



**UADE**  
POSGRADOS

**MAESTRÍA EN TECNOLOGÍA INFORMÁTICA  
Y DE COMUNICACIONES**

**DETECCIÓN DE OUTLIERS EN GRANDES BASES  
DE DATOS**

**ADRIÁN ALFREDO DE ARMAS – LU 101230**

**COHORTE: TIC 2012/2013**

**Director del Trabajo Final**

**Dr. Horacio Kuna, Facultad de Ciencias Exactas.**

**Universidad Nacional de Misiones**

**Mag. Ing. Bibiana D. Rossi, Universidad Argentina de la Empresa**

**23 de Febrero del 2015**

**UNIVERSIDAD ARGENTINA DE LA EMPRESA  
FACULTAD DE INGENIERÍA Y CIENCIAS EXACTAS**

*Dedicado a Maria José y Aimee.*

## 1. AGRADECIMIENTOS

Este trabajo final no hubiese sido posible sin la colaboración de mis dos tutores Bibiana Rossi y Horacio Kuna. Fue Bibiana –una gran amiga que es una de las grandes incorporaciones de mi reciente historia- que, a instancias de Horacio, me propuso un tema tan apasionante como la detección de outliers en grandes bases de datos. Con este trabajo final pude integrar mi interés natural por la programación y la investigación de un tema apasionante, del cual no tenía conocimiento previo, como es la detección de outliers. En mis dos tutores pude encontrar interlocutores que me permitieron compartir ideas para delinear las líneas de investigación, forma de trabajo y presentación de este documento. Quiero agradecerles a ambos su acompañamiento y buena predisposición constante, este trabajo no habría sido posible sin su guía.

A mis padres, hermanos y amigos que me han acompañado toda la vida durante mi formación, todos son parte, todos merecen un “*Gracias*”

A la Universidad Argentina de la Empresa le debo las gracias porque gracias a la institución tuve una experiencia muy enriquecedora cursando la maestría TIC. El roce con mis compañeros y docentes me ha brindado amigos que hoy son parte de mi vida, así como también anécdotas y experiencias que hoy me son útiles en mi vida profesional.

## 2. RESUMEN

Este trabajo final se trata sobre la detección de outliers en grandes bases de datos.

Una de las definiciones más citadas en la bibliografía respecto a lo que es un outlier es la enunciada por David Hawkins en sus monografías sobre estadística y probabilidad aplicadas del año 1980: “*Un outlier es una observación que se desvía tanto de otras observaciones que despierta la sospecha de haber sido generado por un mecanismo diferente*” [Hawkings, 1980].

Con el objetivo de proponer un algoritmo que permita identificar eficaz y eficientemente outliers en grandes bases de datos se seleccionó la aproximación por celdas propuesta por Edwin Knorr y Raymond NG en 1998 en el trabajo “*Algorithms for Mining Distance-Based Outliers in Large Datasets*” [Knorr y otros, 1998]. Este método puede procesar de forma muy eficiente hasta 4 dimensiones (5 en algunos casos) pero luego decrece su rendimiento e incluso puede imposibilitarse su ejecución.

Se implementaron distintas versiones del mismo algoritmo, cada una pensada para un escenario diferente, manteniendo la eficacia del 100% de detección de outliers basados en distancia. Con el fin de aumentar la eficiencia de la implementación del algoritmo, se propone la detección probabilística de outliers basada en la aproximación por celdas que mejora la eficiencia a costa de la reducción de la eficacia manifestada por la detección de falsos positivos. Los resultados de laboratorio arrojan que el porcentaje de disminución de la eficacia del algoritmo es siempre menor que el porcentaje de aumento de la eficiencia.

Los experimentos se realizaron sobre datos sintéticos y, finalmente, se probó el algoritmo con datos reales de todos los vuelos de cabotaje en USA entre los años 1998 y 2003.

### 3. ABSTRACT

This final work deals with finding outliers in large, multidimensional datasets. Edwin Knorr and Raymond NG back in 1998 proposed a cell based algorithm for finding outliers in “*Algorithms for Mining Distance-Based Outliers in Large Datasets*” [Knorr y otros, 1998]. The algorithm named **FindAllOutsM** showed to be by far the best for  $k \leq 4$ . Various improvements were made for different implementations of the algorithm such as virtual hypercube for cells processing, parallel processing and probabilistic outlier determination making **FindAllOutsM** suitable for large datasets with  $k \geq 4$  attributes.

Synthetic and real world data were used to test the implementations of the algorithm with datasets containing from 100.000 to 30.000.000 tuples with 3 to 6 attributes showing good performance for heterogeneous domains. Dataset manager has determined which attributes were handled, however, more attributes can be included if they were available. Performance of **FindAllOutsM** is improved as object count increases.

## 4. CONTENIDO

|   |    |
|---|----|
| 1. AGRADECIMIENTOS.....   | 0  |
| 2. RESUMEN.....   | 2  |
| 3. ABSTRACT .....   | 3  |
| 4. CONTENIDO .....  | 4  |
| 5. INTRODUCCIÓN.....  | 7  |
| 6. PROBLEMÁTICA ACTUAL.....   | 10 |
| 7. MARCO TEÓRICO .....  | 11 |
| 7.1. DATA WAREHOUSE.....  | 11 |
| 7.1.1. DIFERENCIAS ENTRE SISTEMAS DE BASE DE DATOS OPERACIONALES<br>Y DATA WAREHOUSES ..... | 12 |
| 7.2. DATA MINING .....  | 14 |
| 7.2.1. PROCESO DE GENERACIÓN DE UN DATA MINING .....  | 16 |
| 7.2.2. FUNCIONALIDADES DEL DATA MINING.....   | 17 |
| 7.2.2.1. CONCEPTO/CLASE: CARACTERIZACIÓN Y DISCRIMINACIÓN .....                             | 18 |
| 7.2.2.2. PATRONES FRECUENTES, ASOCIACIÓN Y CORRELACIÓN.....                                 | 19 |
| 7.2.2.3. CLASIFICACION Y PREDICCIÓN .....   | 19 |
| 7.2.2.4. ANÁLISIS DE CLUSTERS .....   | 20 |
| 7.2.2.5. ANÁLISIS DE EVOLUCIÓN .....  | 20 |
| 7.2.3. METODOLOGÍA DE DESARROLLO DE DATA MINING .....                                       | 21 |
| 7.2.3.1. SEMMA.....   | 21 |
| 7.2.3.2. CRISP-DM .....   | 22 |
| 7.2.4. ALGORITMOS DE DATA MINING.....   | 24 |
| 7.2.5. HERRAMIENTAS DE DATA MINING .....  | 25 |
| 7.3. NOTACIÓN “O” GRANDE y $\Omega$ .....   | 27 |
| 7.4. OUTLIERS.....  | 30 |
| 7.4.1. MODELOS BÁSICOS PARA OUTLIERS.....   | 31 |
| 7.4.1.1. ANÁLISIS DE VALORES EXTREMOS .....   | 32 |
| 7.4.1.2. ANÁLISIS DE VALORES EXTREMOS DE DATOS MULTIVARIADOS.....                           | 34 |
| 7.4.1.3. MODELOS ESTADÍSTICOS Y PROBABILÍSTICOS .....                                       | 35 |
| 7.4.1.4. MÉTODOS BASADOS EN PROFUNDIDAD .....   | 35 |
| 7.4.1.5. MÉTODOS BASADOS EN DESVÍO.....   | 39 |
| 7.4.1.6. MÉTODOS BASADOS EN ÁNGULOS.....  | 39 |

|            |   |     |
|------------|---|-----|
| 7.4.1.7.   | MÉTODOS BASADOS EN DISTANCIA .....                              | 41  |
| 7.4.1.7.1. | DETERMINACIÓN DE LOS VALORES INICIALES PARA $p$ Y $D$ .....     | 43  |
| 7.4.1.8.   | MODELOS LINEALES .....  | 44  |
| 7.4.1.9.   | MODELOS BASADOS EN PROXIMIDAD .....                             | 47  |
| 7.4.1.9.1. | BASADOS EN CELDAS (DISTANCIA) .....                             | 48  |
| 7.4.1.9.2. | BASADOS EN ÍNDICE (DISTANCIA) .....                             | 50  |
| 7.4.1.9.3. | LOF (DENSIDAD).....   | 51  |
| 7.4.1.9.4. | LOCI (DENSIDAD) .....   | 54  |
| 7.4.2.     | OUTLIERS Y MINERÍA DE DATOS .....                               | 56  |
| 8.         | ESTADO DEL ARTE .....   | 57  |
| 8.1.       | DETECCIÓN DE OUTLIERS BASADA EN CELDAS.....                     | 61  |
| 8.1.1.     | ESTRUCTURA DE CELDAS Y PROPIEDADES EN DOS DIMENSIONES ...       | 62  |
| 8.1.2.     | ALGORITMO PARA CONJUNTO DE DATOS RESIDENTES EN MEMORIA.         | 64  |
| 8.1.3.     | ANÁLISIS DE COMPLEJIDAD PARA DOS DIMENSIONES .....              | 66  |
| 8.1.4.     | GENERALIZACIÓN A DIMENSIONES MAYORES .....                      | 67  |
| 8.1.5.     | ANÁLISIS DE COMPLEJIDAD PARA DIMENSIONES MAYORES A 2 .....      | 68  |
| 9.         | DESARROLLO DEL TRABAJO .....                                    | 69  |
| 9.1.       | MATERIALES Y MÉTODOS.....                                       | 70  |
| 9.2.       | IMPLEMETACION INICIAL: CELDAS Y DATOS EN MEMORIA .....          | 73  |
| 9.3.       | ACCEDIENDO AL BANCO DE DATOS A TRAVÉS DE UNA BD .....           | 80  |
| 9.4.       | FINDALLOUTSM AL LÍMITE: PROCESANDO 6 DIMENSIONES.....           | 84  |
| 9.5.       | PROCESANDO MAYORES VOLÚMENES DE DATOS .....                     | 95  |
|            | OBJETIVO 1: REDUCIR A CERO LA CANTIDAD DE CELDAS VACIAS .....   | 105 |
|            | OBJETIVO 2: ALOJAR EN MEMORIA SOLO LOS OBJETOS RELEVANTES ..... | 106 |
| 9.6.       | DOS IMPLEMENTACIONES, UN ALGORITMO .....                        | 110 |
| 9.7.       | FINDALLOUTSM PARALELO .....                                     | 113 |
| 9.8.       | DETECCIÓN PROBABILÍSTICA DE OUTLIERS .....                      | 124 |
| 9.9.       | RESUMEN DEL DESARROLLO DEL TRABAJO .....                        | 129 |
| 10.        | UTILIZACION DE FINDOUTSM EN BANCO DE DATOS REAL .....           | 133 |
| 11.        | FUTURAS LINEAS DE INVESTIGACIÓN.....                            | 137 |
| 12.        | CONCLUSIONES .....  | 139 |
| 13.        | REFERENCIAS BIBLIOGRÁFICAS.....                                 | 141 |
| 14.        | ANEXOS.....   | 145 |

|         |  |     |
|---------|--|-----|
| 14.1.   | LOGS DE MAPEOS Y CONTROL DE CELDAS BLANCAS PARA CASO DE 2 DIMENSIONES .....        | 145 |
| 14.1.1. | LOGS DE MAPEOS ENTRE CELDAS Y OBJETOS.....   | 145 |
| 14.1.2. | LOG DE CONTROL DE CELDAS BLANCAS .....   | 146 |
| 14.2.   | ARCHIVO NHL94 UTILIZADO EN EL PAPER ORIGINAL .....                                 | 150 |
| 14.3.   | LOGS DE MAPEO DE OBJETOS Y CONTROL DE CELDAS BLANCAS PARA 3D Y N=871 (NHL94) ..... | 166 |
| 14.3.1. | LOGS DE MAPEOS ENTRE CELDAS Y OBJETOS.....   | 167 |
| 14.3.2. | LOG DE CONTROL DE CELDAS BLANCAS .....   | 190 |
| 14.4.   | OUTLIERS DETECTADOS. FINDALLOUTSM EN BANCO DE DATOS REAL                           | 237 |
| 14.4.1. | VERSION QUE EJECUTA TODO EL PROCESO EN MEMORIA. 100% DE EFICACIA                   | 237 |
| 14.4.2. | VERSION MEMORIA/PROBABILÍSTICA (129 FALSOS POSITIVOS) .....                        | 245 |



## 5. INTRODUCCIÓN

Un outlier es cualquier dato que parece estar fuera de lugar respecto al resto de los datos, es por este motivo que a lo largo de la bibliografía sobre el tema puede encontrarse sinónimos tales como excepción, contaminante, disidente, desvío, anomalía, valor discordante, sorprendente o inesperado [Knorr, 2002].

Muchas veces los outliers son vistos como ruido, como dato descartable porque su excepcionalidad genera desvíos a la hora de detectar el comportamiento general de un conjunto de datos.

El ruido de una persona puede ser el foco de interés de otra. En efecto, para algunas aplicaciones, los eventos raros son más interesantes. Aplicaciones como la detección de fraude con tarjeta de crédito y el monitoreo de actividades criminales en el comercio electrónico se benefician con la detección de outliers. Por ejemplo, en el comercio electrónico se espera la existencia de muchas transacciones de bajo valor sin embargo son los casos excepcionales (importe de la transacción, tipo de compra, hora en que se realiza, ubicación o alguna combinación de las anteriores) lo que interesa tanto para la detección de fraude o por motivos de marketing.

Algoritmos de minería de datos sofisticados y automatizados son necesarios para dar soporte al trabajo de científicos, estadísticos y expertos de aplicaciones en ciertos dominios. Dado que la cantidad de datos existente crece exponencialmente, queda claro que una buena herramienta de minería de datos es necesaria para explotar estos datos para interés estratégico científico o comercial [Knorr, 2002].

El objetivo de este trabajo final es proponer la implementación de un algoritmo capaz de analizar cualquier banco de datos con cualquier cantidad de atributos, el trabajo está organizado de la siguiente manera:

- En el capítulo 6, expone con mayor detalle la problemática abordada en este trabajo final.

- En el capítulo 7 se contextualiza el ámbito de desempeño del tipo de algoritmo propuesto. Los grandes volúmenes de datos pueden proceder de repositorios del tipo Data Warehouse más que de repositorios de sistemas de producción. El Data Mining es un campo de las ciencias de la computación en donde se proponen y desarrollan herramientas para analizar los datos contenidos en un gran banco de datos. En este capítulo se documenta el proceso de generación de un Data Mining, las características del Data Mining, metodología de desarrollo, algoritmos y herramientas. Luego se presenta un repaso de distintos métodos y modelos para la detección de outliers para finalizar con la vinculación de esta práctica con el Data Mining.
- En el capítulo 8 se desarrolla la justificación teórica el algoritmo **FindAllOutsM** propuesto por Ed Knorr y Raymond NG en 1998 en el trabajo “*Algorithms for Mining Distance-Based Outliers in Large Datasets*” [Knorr y otros, 1998]. Junto con la base teórica, se analiza la complejidad del algoritmo para dos dimensiones para facilitar la visualización de los casos de ejemplo. Una vez desarrollada la base teórica se generaliza el algoritmo para dimensiones mayores junto con el análisis de complejidad adecuado.
- En el capítulo 9 se desarrollan distintas implementaciones del algoritmo **FindAllOutsM**. Cada implementación es conveniente un escenario diferente. La cantidad de tuplas a procesar, la cantidad de atributos, la memoria RAM disponible o cuantos procesadores dispone del equipo donde se ejecutan los programas son algunas variables a tener en cuenta a la hora de seleccionar qué implementación es la mejor para un escenario determinado. Mejorar la eficiencia del algoritmo, independientemente del escenario donde se lo utiliza, es lo que motivó la creación de las distintas implementaciones. En este capítulo se introduce una variante inexistente en el trabajo original [Knorr y otros, 1998] que es la determinación probabilística de outliers utilizando la aproximación por celdas.

El desarrollo de las distintas implementaciones del algoritmo se realizó sobre datos sintéticos, por este motivo, para finalizar con este trabajo final, en el capítulo 9, se utilizan dos de todas las implementaciones realizadas del algoritmo sobre un banco

de datos real. El banco de datos seleccionado consiste en detalles de arribos y partidas para todos los vuelos comerciales dentro de USA desde octubre de 1987 hasta abril del 2008. El banco de datos consta de aproximadamente 120 millones de registros en total que ocupan 1.6GB de espacio en estado comprimido y aproximadamente 12 GB una vez descomprimidos. El banco de datos de prueba es un subconjunto del banco original con todos los vuelos realizados desde el año 1998 al año 2003.

## 6. PROBLEMÁTICA ACTUAL

Algoritmos de minería de datos sofisticados y automatizados son necesarios para dar soporte al trabajo de científicos, estadísticos y expertos de aplicaciones en ciertos dominios. Dado que la cantidad de datos existente crece exponencialmente, queda claro que una buena herramienta de minería de datos será necesaria para explotar estos datos para interés estratégico científico o comercial [Knorr, 2002].

El tamaño del conjunto de datos y el número de dimensiones han comprobado ser obstáculos claves para el análisis de los datos [Han y otros, 2006]. La minería de datos provee algoritmos que son escalables tanto en tamaño del conjunto de datos como en las dimensiones (variables a evaluar).

Este trabajo final aborda la identificación de outliers en grandes y multidimensionales conjuntos de datos. Se comparten los desafíos que enfrentan los algoritmos de data mining respecto de tamaño del conjunto de datos y dimensionalidad. De hecho, se trabaja con bancos de datos que, generalmente, no pueden ser estudiados en su conjunto en memoria principal.

El algoritmo basado en celdas propuesto por [Knorr y otros, 1998] resulta intuitivo y simple de interpretar [Aggarwal, 2013], a su vez sus limitaciones generan desafíos que dan lugar a la innovación ya que este método puede procesar de forma muy eficiente hasta 4 dimensiones (5 en algunos casos) pero luego decrece su rendimiento e incluso puede imposibilitarse su ejecución debido su complejidad exponencial (la complejidad del tamaño del “cubo”). El algoritmo genera oportunidades a la hora de extender su funcionamiento más allá de sus limitaciones.

El objetivo es proponer una implementación del algoritmo que permita identificar eficaz y eficientemente outliers en grandes bases de datos independientemente del tamaño del banco de datos y la cantidad de atributos a procesar.

## 7. MARCO TEÓRICO

La detección de outliers en grandes bases de datos puede responder a la necesidad de detectar actividad infrecuente como también controlar el armado de repositorios de datos unificados. Es más probable que una base de datos con cientos de millones de tuplas provenga de repositorios para análisis como un Data Warehouse que de repositorios de datos de producción. El Data Mining provee herramientas para el análisis de estos repositorios aunque frecuentemente elimina los outliers por no poder calificarlos. La información que pueden brindarnos los outliers se perdería de no ser por algoritmos específicos para detectar outliers que complementan los algoritmos de Data Mining. A continuación se desarrolla la teoría y conceptos teóricos referidos a la formación de grandes bases de datos, su análisis, algoritmos disponibles, modelos básicos para la detección de outliers y se presenta una notación con la cual se podrá analizar la complejidad de los algoritmos que se presentan.

### 7.1. DATA WAREHOUSE

Un sistema data warehouse, en términos generales, refiere a una base de datos mantenida en forma separada de las bases de datos operacionales de una organización.

Un sistema data warehouse generaliza y consolida datos en un espacio multidimensional. Los sistemas data warehouse permiten la integración de varios sistemas de aplicación y su construcción involucra la limpieza, integración y transformación de datos [Han y otros, 2006].

William H. Inmon, arquitecto líder en la construcción de sistemas data warehouse, los define como “*Colección de datos orientados a un tema, integrados, variables en el tiempo y no volátiles que da soporte el proceso de toma de decisiones de la gerencia*” [Inmon, 2006].

Lo que distingue a los sistemas data warehouse de otros sistemas de repositorios de datos, como los sistemas de base de datos relacionales, sistemas de procesamiento

de transacciones y sistemas de archivos son las cuatro palabras claves que encontramos en la definición de William H. Inmon:

**Orientados a un tema:** son organizados alrededor de temas generales como clientes, proveedores, productos y ventas. Típicamente se provee una vista simple y concisa sobre un tema en particular. Excluye datos que no son útiles para el proceso de toma de decisiones.

**Integrados:** se integran fuentes heterogéneas múltiples como pueden ser bases de datos relacionales, archivos planos, registros de transacciones en línea, entre otras. La limpieza e integración de datos son técnicas aplicadas para asegurar la consistencia de los datos almacenados que son puestos en un único repositorio.

**Variables en el tiempo:** los datos almacenados proveen información desde una perspectiva histórica. Cada estructura en los sistemas data warehouse contienen implícita o explícitamente, un elemento de tiempo.

**No volátiles:** los datos transformados de un data warehouse siempre se almacenan separadamente de los ambientes operacionales. Debido a esta separación sólo se requieren dos operaciones para el acceso a datos: la carga inicial de datos (con las técnicas de limpieza e integración) y el acceso a los datos a través de consultas.

Finalmente, un sistema data warehouse puede ser visto como una arquitectura construida a partir de orígenes heterogéneos para darle soporte a consultas ad hoc, reportes de análisis y toma de decisiones [Han y otros, 2006].

### 7.1.1. DIFERENCIAS ENTRE SISTEMAS DE BASE DE DATOS OPERACIONALES Y DATA WAREHOUSES

Las características más representativas de las diferencias entre sistemas de base de datos operacionales y data warehouses pueden ser resumidas en la tabla 1 parcialmente basada en la tabla presentada en *Data Mining – Concepts and techniques* [Han y otros, 2006].

| Característica                 | BD Operacionales                                  | Data Warehouse  |
|--------------------------------|---|---|
| <b>Uso</b>                     | Procesamiento operacional                         | Procesamiento analítico   |
| <b>Orientación</b>             | Transacción                                       | Análisis  |
| <b>Perfil de usuario</b>       | DBA, profesional de BD                            | Managers, ejecutivos, analistas                                       |
| <b>Función</b>                 | Operaciones diarias                               | Requerimientos de información a largo plazo. Soporte a las decisiones |
| <b>Diseño de base de datos</b> | Orientado a la aplicación. Diseño ER. Normalizado | Orientado a temas. Desnormalizada                                     |
| <b>Tiempo de datos</b>         | Actuales. Actualización constante                 | Históricos. Exactitud mantenida en el tiempo                          |
| <b>Detalle de los datos</b>    | Datos primitivos. Muy detallado                   | Resumido, consolidado   |
| <b>Vistas</b>                  | Detallada. Relacional                             | Resumidas, multidimensional   |
| <b>Unidad de trabajo</b>       | Corta. Transacción simple                         | Consulta compleja   |
| <b>Nivel de análisis</b>       | Bajo  | Alto  |
| <b>Acceso frecuente</b>        | Lectura/Escritura                                 | Casi siempre lectura  |
| <b>Foco</b>                    | Ingreso de datos                                  | Salida de información   |
| <b>Tipos de acceso</b>         | Casi siempre basada en la clave primaria          | Búsquedas complejas, iterativas                                       |
| <b>Registros accedidos</b>     | Decenas   | Millones  |
| <b>Tablas por operación</b>    | Muchas  | Pocas   |
| <b>Número de usuarios</b>      | Miles   | Cientos   |
| <b>Tamaño</b>                  | 100 Mb a GB                                       | 100 GB a TB   |
| <b>Prioridad</b>               | Alto rendimiento. Alta disponibilidad             | Alta flexibilidad. Autonomía del usuario final                        |
| <b>Métrica</b>                 | Rendimiento por transacción                       | Rendimiento por consulta. Tiempo de respuesta                         |

Tabla 1 – Comparación entre bases de datos operaciones y data warehouses

El motivo principal para la separación de un sistema transaccional y un data warehouse es promover el alto rendimiento de ambos sistemas ya que se basan en estructuras diferentes, contenidos diferentes y el uso que se le da a los datos es diferente. El soporte a la toma de decisiones requiere de información histórica mientras que las bases operacionales típicamente no mantienen registros históricos y es insuficiente para la toma de decisiones. La toma de decisiones requiere consolidación de datos provenientes de fuentes heterogéneas que se convierten en datos limpios, integrados y de alta calidad [Han y otros, 2006].

## 7.2. DATA MINING

*“Data mining es el análisis de (usualmente) grandes volúmenes de datos observacionales con el fin de encontrar relaciones insospechadas y resumir los datos en formas novedosas que son entendibles y útiles para quien posee los datos”* [Hand y otros, 2001].

La definición dada hace referencia a “datos observacionales” en oposición a datos experimentales. Data mining típicamente trabaja con datos que fueron previamente recolectados por algún otro propósito distinto al de data mining. Esto significa que los objetivos del data mining no tiene nada que ver con la estrategia de recolección de datos. [Hand y otros, 2001].

La definición también menciona que el volumen de datos a examinar con data mining es usualmente grande. Esto es relevante ya que a medida que el volumen de datos crece se presentan nuevos problemas. Algunos de estos problemas se relacionan con la limpieza de los datos, cómo almacenarlos o accederlos pero otros se relacionan con problemas más fundamentales relacionados a la representatividad de los datos, el análisis de datos en un período de tiempo razonable o cómo decidir si una relación aparente es pura coincidencia y no refleja la realidad subyacente [Hand y otros, 2001].

Las relaciones y resúmenes derivados de un ejercicio de data mining son referenciados como modelos o patrones.

Estos patrones se pueden recopilar y definir como un modelo de minería de datos. Los modelos de minería de datos se pueden aplicar en escenarios como los siguientes [Microsoft, 2013]:

- *Pronóstico*: cálculo de las ventas y predicción de las cargas del servidor o del tiempo de inactividad del servidor.
- *Riesgo y probabilidad*: elección de los mejores clientes para la distribución de correo directo, determinación del punto de equilibrio probable para los escenarios de riesgo, y asignación de probabilidades a diagnósticos y otros resultados.



- *Recomendaciones*: determinación de los productos que se pueden vender juntos y generación de recomendaciones.
- *Búsqueda de secuencias*: análisis de los artículos que los clientes han introducido en el carrito de la compra y predicción de posibles eventos.
- *Agrupación*: distribución de clientes o eventos en grupos de elementos relacionados, y análisis y predicción de afinidades.

Los términos de *data mining* y *descubrimiento del conocimiento* usualmente son intercambiables [Knorr, 2002].

Esta actividad ha recibido mucha atención de parte de la de la sociedad y de la industria de la información, en particular, en los últimos años debido a la disponibilidad de enormes cantidades de datos y la inminente necesidad de convertir esos datos en información útil y conocimiento.

La información y el conocimiento que se adquiere por esta técnica pueden ser usados para aplicaciones como análisis de mercado, detección de fraude y retención de clientes, control de producción y exploración científica.

Data mining puede ser visto como el resultado de la evolución natural de la tecnología de la información [Han y otros, 2006].

Definir una disciplina científica es una tarea siempre controversial; los investigadores suelen estar en desacuerdo respecto del rango y límites precisos del campo de estudio. Teniendo en cuenta que puede haber desacuerdos con detalles de la definición dada, a continuación se presentan varias definiciones de data mining [Molina López y otros, 2006]:

- “Data mining es el proceso de extracción y refinamiento de conocimiento útil desde grandes bases de datos”
- “Data mining es el proceso de extracción de información previamente desconocida, válida y procesable desde grandes bases de datos para luego ser utilizada en la toma de decisiones”

- "Data mining es la exploración y análisis, a través de medios automáticos y semiautomáticos, de grandes cantidades de datos con el fin de descubrir patrones y reglas significativos"
- "Data mining es el proceso de planteamiento de distintas consultas y extracción de información útil, patrones y tendencias previamente desconocidas desde grandes cantidades de datos posiblemente almacenados en bases de datos"
- "Data mining es el proceso de descubrir modelos en los datos"
- "Data mining es la extracción no trivial de información implícita, desconocida previamente, y potencialmente útil desde los datos" [Frawley y otros, 1992]

Respecto de la definición dada por William Frawley, Gregory Piatetsky-shapiro y Christopher Matheus, Rakesh Agrawal recomendaba, en el ámbito de la 6ta conferencia IBM CASCON del año 1996 en Toronto, que el adjetivo "eficiente" fuera agregado a la definición para enfatizar el tipo de procesamiento que el data mining debería realizar dado el gran volumen de información con el que se debía tratar [Knorr, 2002].

### 7.2.1. PROCESO DE GENERACIÓN DE UN DATA MINING

Muchas veces se trata al data mining como un sinónimo de otro término popularmente usado como el KDD (Knowledge Discovery from Data). Alternativamente, otros ven al data mining simplemente como un paso esencial en el proceso KDD y esta es la forma en que data mining es interpretado en este trabajo.

KDD como un proceso se representa en la figura 1 y consiste en una secuencia iterativa de los siguientes pasos [Han y otros, 2006]:

1. *Limpieza de datos*: remover ruido y datos inconsistentes
2. *Integración de datos*: donde muchos orígenes de datos se combinan almacenándose en data warehouses (aunque no es excluyente)
3. *Selección de datos*: donde datos relevantes para el análisis son importados desde la base de datos
4. *Transformación de datos*: donde los datos son transformados y consolidados en formas apropiadas para la minería al realizar resúmenes y operaciones de

- agregación. El resultado de este proceso es lo que típicamente alimenta un data warehouse (aunque no es excluyente)
5. *Data Mining*: proceso esencial donde métodos inteligentes son aplicados con el fin de extraer patrones
  6. *Evaluación de patrones*: identificar los patrones realmente interesantes que representen en conocimiento basado en medidas de interés
  7. *Presentación del conocimiento*: Donde técnicas de visualización y representación del conocimiento son usadas para presentar el conocimiento adquirido al usuario

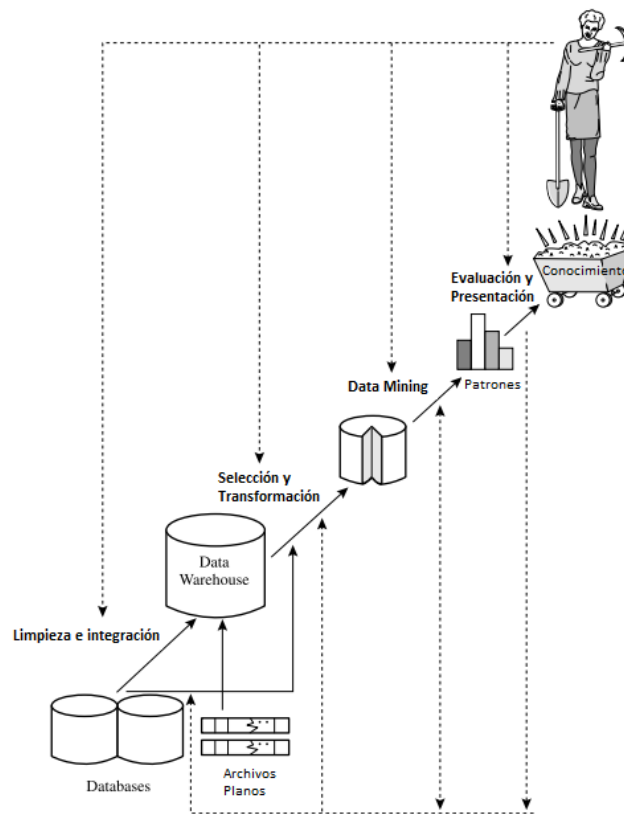


Figura 1 – Proceso KDD [Han y otros, 2006]

### 7.2.2. FUNCIONALIDADES DEL DATA MINING

Las funcionalidades de un data mining son usadas para especificar el tipo de patrón a encontrar en una tarea de data mining. En general, las tareas de data mining puede ser clasificadas en dos categorías: *descriptiva* y *predictiva*. Las tareas de mining descriptivas caracterizan las propiedades generales de los datos en la base

de datos. Las tareas de mining predictivas realizan inferencias en los datos actuales con el fin de realizar predicciones [Han y otros, 2006].

Muchas veces los patrones que se pueden encontrar en una base de datos no son algo que el usuario conozca con antelación por lo tanto es importante tener un sistema de data mining capaz de buscar distintos tipos de patrones para ajustarse a las diferentes expectativas de usuarios.

Las funcionalidades del data mining y los tipos de patrones que pueden descubrir son descriptos a continuación.

#### **7.2.2.1. CONCEPTO/CLASE: CARACTERIZACIÓN Y DISCRIMINACIÓN**

Los datos pueden ser asociados a clases o conceptos. Por ejemplo para una casa de electrónica, hay ítems de distintas clases para la venta como *computadoras* e *impresoras*; también hay conceptos de clientes como *mayoristas* y *minoristas*. Puede ser muy útil describir el comportamiento individual de clases y conceptos de forma concisa, resumida y en términos precisos. Las descripciones de una clase o concepto son llamadas *descripciones* [Han y otros, 2006].

Estas descripciones pueden derivar en [Han y otros, 2006]:

1. *Caracterización de los datos*: resumiendo los datos de la clase bajo estudio (o clase objetivo)
2. *Discriminación de los datos*: comparando la clase objetivo con una o un conjunto de clases comparables (usualmente llamadas clases de contraste)
3. *Ambas*: caracterización y discriminación

La caracterización de los datos es un resumen de las características generales de la clase objetivo.

Hay varios métodos para generar los resúmenes o para caracterizar de manera efectiva desde el uso de medidas estadísticas o utilizando cubos con OLAP (On Line Analytical Processing).

El resultado final de la caracterización de los datos puede ser presentado de varias formas, por ejemplo gráficos de torta, gráficos de barra, curvas, cubos de datos multidimensionales o tablas multidimensionales [Han y otros, 2006].

#### **7.2.2.2. PATRONES FRECUENTES, ASOCIACIÓN Y CORRELACIÓN**

Los patrones frecuentes, son patrones que ocurren a menudo en el conjunto de datos. Hay muchos tipos de patrones frecuentes incluyendo *conjuntos de ítems*, *sub secuencias* y *sub estructuras*. Un *conjunto de ítems* se refiere a ítems que frecuentemente aparecen juntos en un conjunto de datos transaccionales como son el pan y la leche. Una *sub secuencia* frecuente es un patrón como aquel en que los clientes tienden a comprar primero una PC, luego una cámara digital y, finalmente, una tarjeta de memoria. Una *sub estructura* puede referirse a diferentes formas de estructuras tales como gráficos, árboles los cuales pueden ser combinados con *conjuntos de ítems* o *sub secuencias*. Buscar patrones frecuentes lleva al descubrimiento de asociaciones y correlaciones interesantes dentro de los datos [Han y otros, 2006].

#### **7.2.2.3. CLASIFICACION Y PREDICCIÓN**

Clasificación es el proceso de encontrar un modelo (o función) que describe y distingue clases de datos o conceptos, con el propósito de ser capaz de usar el modelo para predecir la clase de los objetos cuya clase es desconocida. El modelo derivado se basa en el análisis de un conjunto de datos de entrenamientos (objetos cuya clase es conocida).

Los modelos derivados pueden ser presentados de distintas maneras como ser las reglas de clasificación (*IF/THEN*), *árboles de decisión*, *formulaciones matemáticas* o *redes neuronales*. Un árbol de decisión es diagrama de flujo con forma de estructura de árbol, donde cada nodo denota una prueba sobre el valor de algún atributo, cada rama representa el resultado del test y las hojas del árbol representan clases o distribuciones de clases. Los árboles de decisión pueden ser fácilmente convertidos en reglas de clasificación. Una red neuronal, cuando se usa para clasificación, es típicamente una colección de unidades de procesamiento como neuronas con conexiones ponderadas entre unidades. Hay muchos otros métodos para la

construcción de modelos de clasificación como bayesiano ingenuo (naive bayes) o el vecindario  $k$  más cercano. Mientras que la clasificación predice etiquetas categóricas (discretas, desordenadas), la predicción modela funciones continuas de las que se puede extraer un valor concreto. Esto quiere decir que es usada para predecir valores numéricos perdidos o no disponibles en vez de una etiqueta de clase. El análisis de regresión es una metodología estadística que se usa con frecuencia para la predicción numérica. La predicción también abarca la identificación de las tendencias de una distribución desconocida en base a los datos disponibles.

Clasificación y predicción requieren ser precedidas por un análisis de relevancia el cual intenta identificar los atributos que no contribuyen al proceso de clasificación o predicción. Estos atributos deben ser excluidos [Han y otros, 2006].

#### **7.2.2.4. ANÁLISIS DE CLUSTERS**

A diferencia de la clasificación y predicción que analizan objetos de datos etiquetados con clases, el análisis de clusters trabaja sin conocer una etiqueta de clase.

En general, las etiquetas de clases no están presentes en el conjunto de datos de entrenamiento porque simplemente no son conocidas. Armar clusters puede ser utilizado para generar dichas etiquetas. Los objetos son agrupados en base al principio de *maximizar la similitud dentro de la clase y minimizar la similitud entre clases*. Los objetos son agrupados de forma tal que los objetos dentro de un mismo cluster son fuertemente similares en comparación al resto pero muy distintos respecto a los objetos de otros clusters. Cada cluster que se forma puede ser visto como una clase de objetos desde la cual se pueden derivar reglas. Armar clusters también facilita la formación de taxonomías, eso es, organizar las observaciones en jerarquías de clases que agrupan eventos similares [Han y otros, 2006].

#### **7.2.2.5. ANÁLISIS DE EVOLUCIÓN**

Los análisis de evolución de datos describen y modelan las tendencias o distintas regularidades de objetos cuyo comportamiento cambia con el tiempo. Aunque esto puede incluir análisis de caracterización, discriminación, análisis de asociación y correlación, clasificación, predicción o armados de clusters de datos relacionados en

el tiempo, se adicionan distintas características como un análisis que contiene una serie temporal, secuencias o patrones de periodicidad [Han y otros, 2006].

### 7.2.3. METODOLOGÍA DE DESARROLLO DE DATA MINING

En la búsqueda del establecimiento de un estándar en el área de data mining han existido esfuerzos que se han destacado como el de SEMMA y CRISP-DM. Ambos han crecido como estándares industriales y definen un conjunto de pasos secuenciales que pretenden guiar la implementación de aplicaciones de data mining [Azevedo y otros, 2008].

SEMMA fue desarrollado por el SAS Institute, CRISP-DM fue desarrollado por medio de los esfuerzos de un consorcio inicialmente compuesto por Daimler Chrysler, SPSS y NCR. Ambas metodologías son en particular un curso de acción destinado a lograr un resultado.

El trabajo “*KDD, SEMMA AND CRISP-DM: A PARALLEL OVERVIEW*” presentado por Ana Azevedo y Manuel Filipe Santos en el año 2008 en el ámbito de IADIS (Conferencia Europea sobre data mining) hace un análisis comparativo entre los procesos KDD, SEMMA y CRISP-DM concluyendo que tanto SEMMA como CRISP-DM pueden ser vistos como una implementación del proceso KDD [Azevedo y otros, 2008].

#### 7.2.3.1. SEMMA

SEMMA es un acrónimo para *Sample, Explore, Modify, Model, Assess* (Muestrear, Explorar, Modificar, Modelar, Evaluar) y se refiere al proceso de conducir un proyecto de data mining. El Instituto SAS considera un ciclo de 5 etapas para el proceso:

1. *Muestrear*: Consiste en tomar una muestra de los datos extrayendo una porción grande de datos, suficientemente grande para contener información significativa y al mismo tiempo suficientemente pequeña para manipularla rápidamente.
2. *Explorar*: Consiste en la exploración de los datos buscando tendencias no anticipadas o anomalías de forma de ganar ideas y entendimiento de los datos.

3. *Modificar*: Consiste en modificar los datos creando, seleccionado y transformando las variables para enfocarse en el proceso de selección del modelo.
4. *Modelar*: Consiste en modelar los datos permitiendo que el software busque automáticamente una combinación de los datos que de manera fiable pueda predecir la salida deseada.
5. *Evaluar*: Consiste en evaluar los datos teniendo en cuenta la usabilidad y confiabilidad de los resultados del proceso de data mining y estimar que tan bien ha rendido.

SEMMA ofrece un proceso fácil de entender, permitiendo un organizado y adecuado desarrollo y mantenimiento de proyectos de data mining. Así otorga una estructura para esta concepción, creación y evolución esperando presentar soluciones a problemas de negocio como así encontrar objetivos de negocio [Azevedo y otros, 2008].

### 7.2.3.2. CRISP-DM

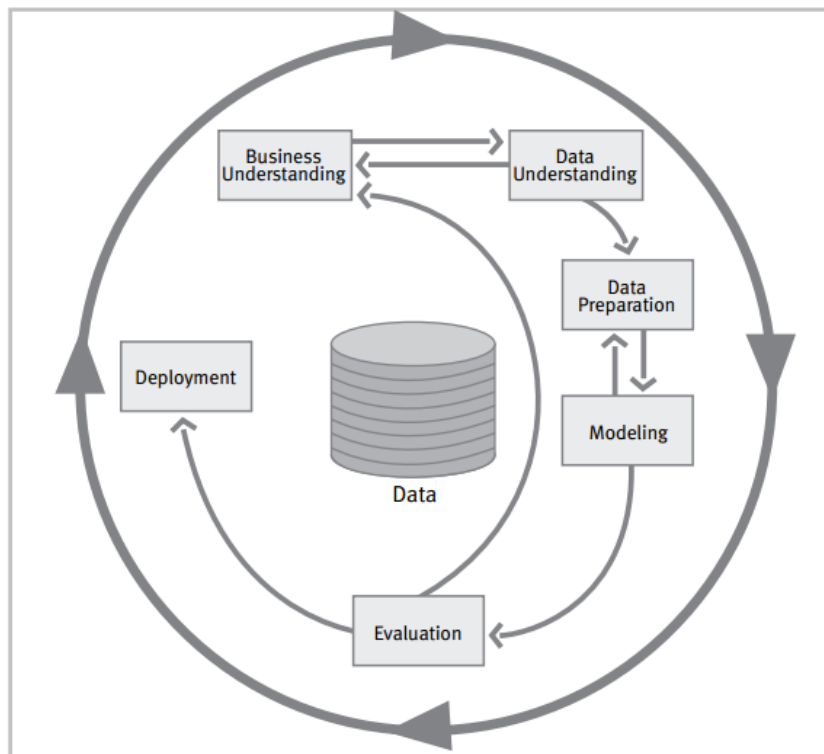


Figura 2 – Ciclo de vida de CRISP-DM [Chapman y otros, 2000]



CRISP-DM representa **C**Ross-**I**ndustry **S**tandard **P**rocess for **D**ata **M**ining. Consiste en un ciclo de vida que comprende seis etapas [Chapman y otros, 2000]:

1. *Comprender el negocio* (Business understanding): fase inicial cuyo objetivo es entender los objetivos del proyecto y sus requerimientos desde una perspectiva de negocio, luego se convierte este conocimiento en una definición de problema de data mining y se diseña un plan preliminar para alcanzar estos objetivos
2. *Entender los datos* (Data understanding): esta fase comienza con una colección de datos inicial y se procede con actividades tendientes a lograr familiarizarse con los datos, identificar problemas de calidad de datos, descubrir las primeras ideas sobre los datos o para detectar subconjuntos interesantes para formar hipótesis de información oculta
3. *Preparación de los datos* (Data preparation): esta fase cubre todas las actividades para construir el conjunto de datos final tomando como inicio los datos en crudo
4. *Modelar* (Modeling): en esta fase varias técnicas se seleccionan y aplican, luego sus parámetros son calibrados hasta llegar a valores óptimos con el fin de poder construir el modelo
5. *Evaluación* (Evaluation): en esta etapa el modelo (o modelos) obtenidos son evaluados más a fondo y los pasos realizados para construir el modelo son revisados para asegurarse que se alcanzan los objetivos de negocio apropiadamente
6. *Instalación/Despliegue* (Deployment): la creación del modelo, generalmente, no es el fin del proyecto. Aún si los propósitos del modelo es incrementar el conocimiento de los datos, el conocimiento obtenido debe ser organizado y presentado de forma tal que el cliente pueda usarlo

En la figura 2 se presentan las seis etapas del ciclo de vida de CRISP-DM y su relación. CRISP-DM es extremadamente completo y documentado. Todas sus etapas están debidamente organizadas, estructuradas y definidas, permitiendo que un proyecto pueda ser fácilmente entendido y revisado [Azevedo y otros, 2008].

#### 7.2.4. ALGORITMOS DE DATA MINING

A continuación se enumeran los 10 algoritmos principales de data mining identificados por IEEE presentados en la conferencia internacional de data mining (ICDM) del año 2006 [Wu y otros, 2007]. Estos 10 algoritmos se encuentran entre los más influyentes para la comunidad de investigación y cubren clasificación, clustering, aprendizaje estadístico, análisis de asociación y link mining, que se encuentran entre los tópicos más importantes en la investigación y desarrollo del data mining.

En un esfuerzo por identificar los algoritmos más influyentes que han sido ampliamente usados en la comunidad del data mining, como primer paso del proceso de identificación, en septiembre del 2006 se invitó a quienes obtuvieron premios por innovación o por investigación a que nominaran los 10 algoritmos más conocidos en data mining. Todos, excepto uno, de estos distinguidos ganadores respondieron la invitación. Se solicitó que cada algoritmo nominado debía ser ampliamente citado y utilizado por otros investigadores del campo. Una vez recibidas las nominaciones se controló con Google Scholar (en el año 2006) y se eliminó todas aquellas nominaciones con menos de 50 citas. El proceso de refinamiento continuó y está debidamente documentado en el trabajo “*Top 10 algorithms in data mining*” [Wu y otros, 2007]. El resultado es el que se presenta a continuación.

Los 10 algoritmos más importantes del data mining:

1. C4.5
2.  $k$ -means algorithm
3. Support vector machines
4. The Apriori algorithm
5. The EM algorithm
6. PageRank
7. AdaBoost
8.  $k$ NN:  $k$ -nearest neighbor classification
9. Naive Bayes
10. CART

En el trabajo “*Top 10 algorithms in data mining*” cada algoritmo está debidamente descrito, se discute su impacto y se realiza una revisión sobre el estado actual y futuras investigaciones sobre el algoritmo [Wu y otros, 2007]. Se recomienda su lectura para ahondar en las diferencias de cada uno de los algoritmos que componen la lista.

### 7.2.5. HERRAMIENTAS DE DATA MINING

Para llevar a cabo el proceso de data mining es necesaria la utilización de distintas herramientas que nos asistirán a la hora de aplicar los métodos inteligentes para extraer patrones y también a la hora de presentar el conocimiento adquirido.

A continuación se presentan algunas herramientas creadas con el fin de asistir el proceso de data mining. La lista está basada en el artículo “5 de los mejores softwares para datamining libres y open source” del sitio TECHSOURCE [Auza, 2010]:



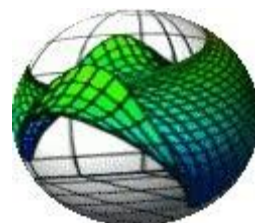
Orange es una suite de software basada en componentes para minería de datos y aprendizaje de máquina con características amigables pero al mismo tiempo poderosas. Posee una interfaz gráfica rápida y versátil para visualización y análisis de datos exploratorios. Permite escribir scripts en lenguaje Python. Contiene un completo conjunto de componentes para pre procesamiento de datos, filtrado y puntuado, modelado, evaluación de modelos y técnicas de exploración. Está escrito en C++ y Python, y su interfaz gráfica está basada en el framework multiplataforma QT.



RapidMiner, también conocido como YALE (*Yet Another Learning Environment – Otro entorno de aprendizaje más-*), es un entorno para experimentar con aprendizaje de máquina y minería de datos y es utilizado tanto para investigación como para tareas de minería de datos del mundo real. Permite construir experimentos hechos de un gran número de operadores unidos arbitrariamente, los cuales son detallados en archivos XML que son construidos con la interfaz gráfica de RapidMiner. Provee más de 500 operadores para todos los principales procedimientos de aprendizaje de máquina, y también combina evaluadores de esquemas y atributos del entorno Weka.



Escrito en Java, Weka (*Waikato Environment for Knowledge Analysis*) es una suite bien conocida para aprendizaje de máquina que soporta varias tareas típicas de la minería de datos, particularmente el pre procesamiento de datos, clustering, clasificación, regresión, visualización, y selección de características. Sus técnicas están basadas en la hipótesis que los datos están disponibles como un solo archivo plano o relación, donde cada punto de información está etiquetado por un número fijo de atributos. Weka provee acceso a bases de datos SQL utilizando “Java Database connectivity” y puede procesar el resultado de una query.



Diseñado para científicos, ingenieros y estudiantes, JHepWork es un framework libre y de código abierto para el análisis de datos.



KNIME (Konstanz Information Miner) herramienta amigable, inteligible y completa que provee una plataforma para la integración de datos, procesamiento, análisis y exploración.

La lista presentada es parcial y reducida respecto a lo que podemos encontrar en el mercado. Software propietario como IBM SPSS Modeler, SAS Enterprise Miner, Angoss Knowledge STUDIO, Microsoft Analysis Services, Oracle Data Mining, Think Analytics, Viscovery Data Mining Suite, IBM DB2 Intelligent Miner, STATISTICA Data Miner, Forte Wares QIWare o software libre y open source como TANAGRA, C4.5/C5.0/See5 y OCTAVE engrosan la oferta disponible para asistir el proceso de data mining.

### 7.3. NOTACIÓN “O” GRANDE y $\Omega$

Para medir las características de rendimiento de un algoritmo se requiere de una notación compacta y fácil de interpretar.

Como la eficiencia de un algoritmo puede medirse por varios factores como el tiempo de uso de CPU, de memoria, de disco o de red, en este caso, sólo se tiene en cuenta la complejidad asociada al tiempo (uso de CPU).

La complejidad está relacionada a cómo el tiempo requerido de CPU de un algoritmo en particular, escala a medida que el tamaño del problema a resolver se vuelve más grande. La computación ya tiene dicha notación y se denomina *notación “O” grande* [BuckNall, 2012].

Para esta notación, se trabaja con la función matemática de  $n$  que representa el número de ítems sobre el cual el rendimiento del algoritmo es proporcional.

Dada las funciones  $f(n)$  y  $g(n)$  se dice que  $f(n)$  es  $O(g(n))$  si existe una constante positiva  $c > 0$  y  $n_0 \geq 1$  tal que:

$$f(n) \leq cg(n) \text{ para todo } n, n \geq n_0$$

Esta definición formal es la que se usa para referenciar a la notación “*O*” *grande*. [Jones y otros, 2004]

De manera alternativa se suele decir que  $f(n)$  **es de orden**  $g(n)$ .

Esta notación brinda el límite superior de la tasa de crecimiento de una función. La afirmación “ $f(n)$  es  $O(g(n))$ ” significa que el la tasa de crecimiento de  $f(n)$  no es mayor que la tasa de crecimiento de  $g(n)$  como se visualiza en la figura 3.

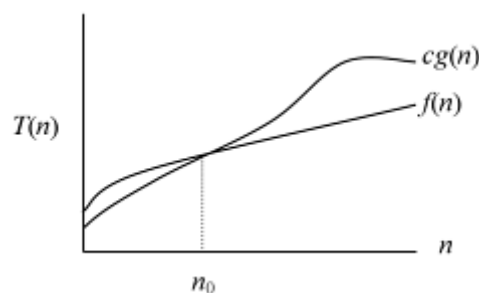


Figura 3 -  $f(n)$  es  $O(g(n))$  [Muniswamy, 2009]

Así como la notación “*O*” *grande* gobierna el límite superior del crecimiento de una función, la relación que refleja el límite inferior del crecimiento de una función se denomina  $\Omega$ .

Una función  $f(n)$  es  $\Omega(g(n))$  si existen constantes reales positivas  $c$  y  $n_0$  tal que

$$f(n) \geq cg(n) \text{ para todo } n, n \geq n_0$$

Si  $f(n) = \Omega(g(n))$ , entonces se dice que  $f$  crece más rápido que  $g$  [Jones y otros, 2004]. Esta relación puede visualizarse en la figura 4.

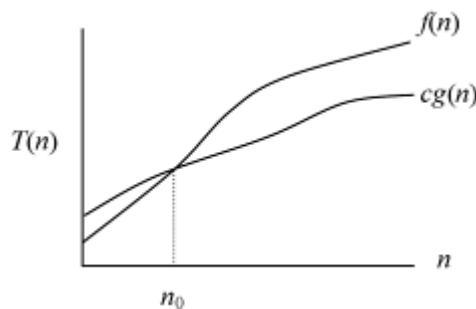


Figura 4 -  $f(n)$  es  $\Omega(g(n))$  [Muniswamy, 2009]

En computación se categorizan los algoritmos de acuerdo a sus funciones asintóticas en tiempo de ejecución. El comportamiento en tiempo de ejecución de la mayoría de los algoritmos cae dentro de una de las siguientes siete categorías primarias  $O(1)$ ,  $O(\log n)$ ,  $O(n \log n)$ ,  $O(n^2)$ ,  $O(n^3)$  y  $O(2^n)$ . Estas son conocidas como *clases de complejidad* o *clases de crecimiento asintóticas* [Muniswamy, 2009] y todas pueden visualizarse en la figura 5.

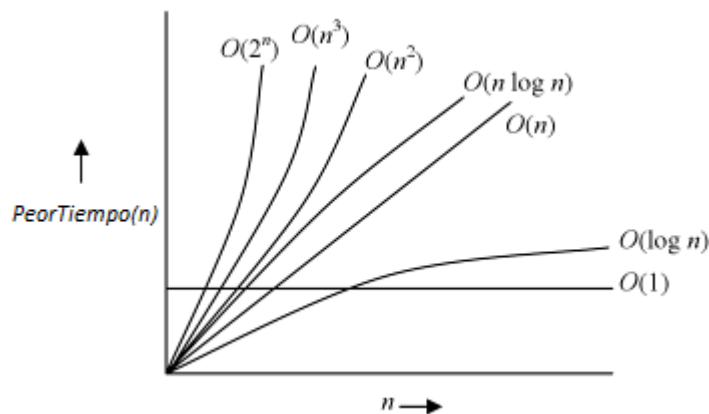


Figura 5 – Comparación del peor tiempo para los siete órdenes de funciones.

La notación “O” grande es sucinta y compacta. Si mediante experimentación se descubre que el algoritmo  $X$  es  $O(n^2 + n)$  significa, en otras palabras, que su rendimiento es proporcional a  $n^2 + n$ . Por “proporcional a” se refiere a que se puede encontrar una constante  $k$  tal que la siguiente ecuación se mantiene cierta:

$$\text{Rendimiento} = k (n^2 + n)$$

Por esta ecuación, y otras derivadas de la notación “O” grande se puede observar que multiplicar la función matemática dentro del paréntesis de la “O” grande por un valor constante no tiene ningún efecto. Por ejemplo,  $O(3f(n))$  es igual a  $O(f(n))$ ; el 3 puede sacarse de la ecuación y multiplicarse fuera de la constante de proporcionalidad, la cual se puede, convenientemente, ignorar.

Si el valor de  $n$  es suficientemente grande cuando se prueba el algoritmo  $X$ , se puede afirmar con seguridad que los efectos del término “ $+n$ ” son absorbidos por el término  $n^2$ . En otras palabras, provistos de un  $n$  suficientemente grande,  $O(n^2 + n)$  es muy similar a  $O(n^2)$ . Esto es aplicable a cualquier término adicional en  $n$ , esto

quiere decir que se puede ignorar ese término adicional de forma segura ya que para un valor suficientemente grande de  $n$ , sus efectos son absorbidos por otro término de  $n$ . Por lo tanto, un término  $n^2$  será absorbido por un término  $n^3$ ; un término  $\log(n)$  será absorbido por un término en  $n$ ; etc. [BuckNall, 2012].

**Advertencia:** *“Se debe prestar atención antes de sacar conclusiones respecto de los valores “O” grande. Los valores de “O” grande son representativos de lo que pasa con grandes valores de  $n$ . Para valores de  $n$  pequeños, la notación se quiebra completamente ya que otros factores comienzan a entrar en juego y el resultado general muta”* [BuckNall, 2012].

#### 7.4. OUTLIERS

Un outlier es cualquier dato que parece estar fuera de lugar respecto al resto de los datos, es por este motivo que a lo largo de la bibliografía sobre el tema pueden encontrarse sinónimos tales como excepción, contaminante, disidente, desvío, anomalía, valor discordante, sorprendente o inesperado [Knorr, 2002].

Una de las definiciones más citadas en la bibliografía es la enunciada por David Hawkins en sus monografías sobre estadística y probabilidad aplicadas del año 1980: *“Un outlier es una observación que se desvía tanto de otras observaciones que despierta la sospecha de haber sido generado por un mecanismo diferente”* [Hawkings, 1980].

Barnet y Lewis definen a los outliers como *“una observación (o subconjunto de observaciones) que parecen ser inconsistentes respecto del resto de ese conjunto de datos”* [Barnet y otros, 1994]. En otras palabras, las observaciones que no siguen el mismo modelo estadístico que el resto de los datos son considerados outliers. Charu Aggarwal escribe *“Virtualmente todos los algoritmos de detección de outliers crean un modelo del patrón normal de los datos, luego teniendo en cuenta un dato dado, calcula una puntuación de outlier en base a la desviación de ese patrón”* [Aggarwall, 2013]. La puntuación del dato dado se calcula evaluando la calidad de ajuste del punto al modelo.



Dependiendo en la distribución subyacente de los datos, un outlier puede ser una de las siguientes entidades [Knorr, 2002]:

- Un valor extremo o relativamente extremo
- Un contaminante, es decir, una observación de otra (posiblemente desconocida) distribución
- Un valor legítimo pero sorprendente/inesperado
- Un valor que ha sido medido o grabado incorrectamente

#### 7.4.1. MODELOS BÁSICOS PARA OUTLIERS

La mayoría de los trabajos existentes sobre la detección de outliers yace en el campo de la estadística. Consistentemente, acerca de 100 formas de detectar outliers han sido desarrolladas para diferentes circunstancias, dependiendo de [Barnet y otros, 1994]:

- Distribución de los datos
- Si los parámetros de la distribución son conocidos o no
- El número de outliers esperados
- El tipo de outliers esperados

Sin embargo, estas formas de detección presentan dos inconvenientes

1. Casi todas son univariadas. Esta restricción hace que sean inaplicables para grupos de datos multidimensionales.
2. Todas son basadas en la distribución de los datos. Existen numerosas situaciones donde se desconoce si un atributo en particular sigue una distribución normal, una distribución gama, o cualquier otra, por lo cual, se deben realizar pruebas extensas para encontrar una distribución que se ajuste al atributo.

A continuación se presentan distintos métodos para la detección de outliers y sus características. La taxonomía de los métodos para la detección de outliers está

parcialmente basada en el trabajo “*Outlier Analysis*” realizado por Charu Aggarwal del año 2013.

#### 7.4.1.1. ANÁLISIS DE VALORES EXTREMOS

La forma más básica de detección de outliers es el análisis de valores extremos en datos de una dimensión. Estos outliers son de un tipo muy específico, en el cual se asume que los valores que son muy grandes o muy pequeños son outliers.

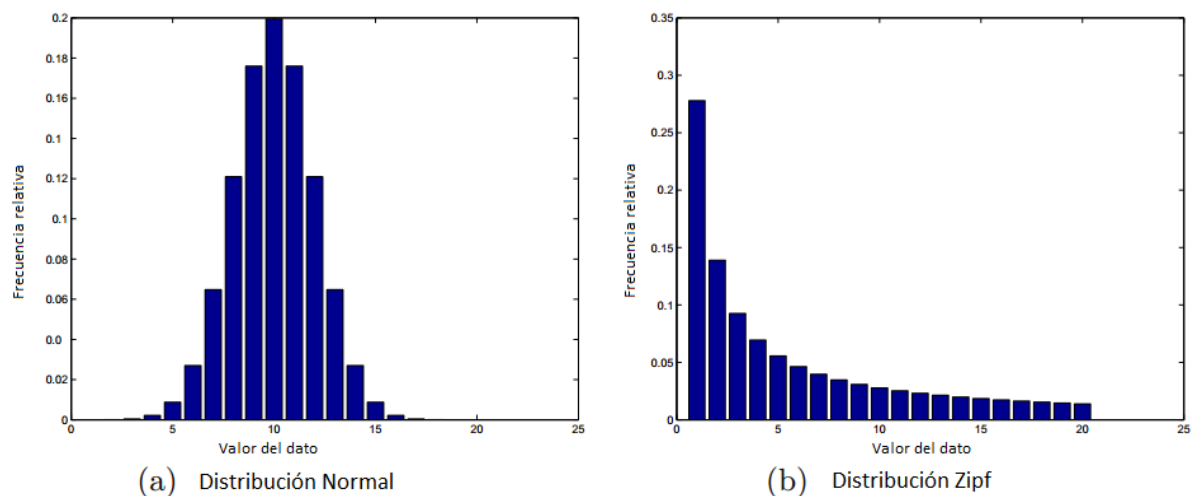


Figura 6 – Distribuciones normal y Zipf [Aggarwal, 2013]

La clave es determinar las colas estadísticas de la distribución subyacente. Como se ilustra en la figura 6, la naturaleza de las colas puede variar considerablemente dependiendo de la distribución subyacente de los datos. La distribución normal es la más sencilla de analizar, porque la mayoría de los test (tal como el test del valor  $Z$ ) puede ser interpretado directamente en término de probabilidad de significancia. Sin embargo, aun para distribuciones arbitrarias dichos test pueden proveer de una buena idea heurística de los valores outliers de los datos analizados aun cuando no hayan sido analizados estadísticamente [Aggarwal, 2013].

La mayoría de los primeros métodos univariados para la detección de outliers asumen una distribución subyacente conocida de los datos, idéntica e independientemente distribuida [Ben Gal, 2005]. Por otra parte, muchos de los test de discordancia para detectar outliers univariados además asumen que los

parámetros de la distribución y el tipo de outliers que se espera son conocidos [Barnet y otros, 1994].

Una asunción central en los métodos estadísticos para la detección de outliers, es que el modelo generador permite el muestreo de un pequeño número de observaciones aleatorias de las distribuciones  $G_1 \dots G_k$ , diferentes de la distribución objetivo  $F$ , la cual, generalmente, se toma como una distribución normal  $N(\mu, \sigma^2)$ . Luego, el problema de identificación de outliers se traslada al problema de identificar aquellas observaciones que yacen en el la llamada “región outlier” [Ben Gal, 2005]. Esto nos lleva a la siguiente definición:

Para cualquier intervalo de confianza  $\alpha, 0 < \alpha < 1$ , la región  $\alpha$  – outlier de la distribución  $N(\mu, \sigma^2)$  se define como:

$$out(\alpha, \mu, \sigma^2) = \{x: |x - \mu| > z_{1-\alpha/2}\sigma\},$$

Donde  $Z_q$  es el  $q$  quintil de  $N(0, 1)$ . Un número  $x$  es un  $\alpha$  – outlier respecto de  $F$  si  $x \in out(\alpha, \mu, \sigma^2)$  [Davies y otros, 1993].

Considerar el análisis de valores extremos como outliers dista de la definición tradicional provista por Hawkins [Hawkins, 1980]. Hawkins define los outliers por su probabilidad de generación, no por su pertenencia a las extremidades del conjunto de valores. Confundir el análisis de valores extremos y el análisis de outliers es bastante común especialmente en el contexto de datos multivariados. Esto sucede a menudo ya que muchos modelos de valores extremos también usan modelos probabilísticos para cuantificar la probabilidad de que un dato dado sea un valor extremo.

El modelado de valores extremos juega un rol importante en la mayoría de los algoritmos de detección de outliers. Esto sucede porque la mayoría de los algoritmos para detección de outliers cuantifica la desviación de los datos respecto del patrón normal en la forma de un valor numérico. El análisis de valores extremos usualmente es utilizado como el último paso sobre estas desviaciones modeladas, ya que ahora son representadas como valores univariados en los cuales los valores extremos corresponden a outliers. Este método en particular no ha tenido mucha utilidad en la

literatura para análisis genérico de outliers por su incapacidad de descubrir outliers en las regiones interiores de un conjunto de datos [Aggarwal, 2013].

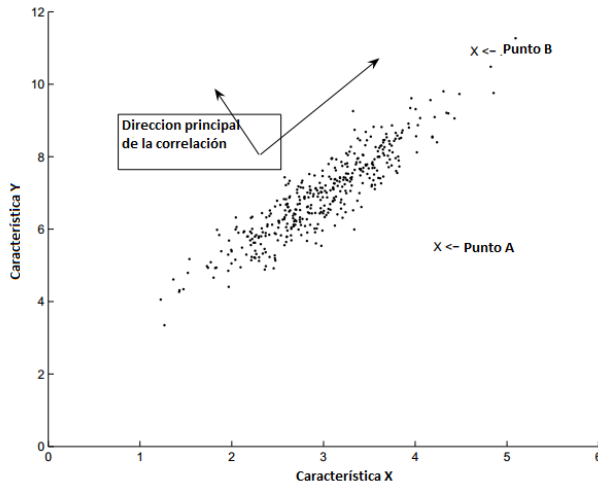


Figura 7 – Banco de datos 1 [Aggarwal, 2013]

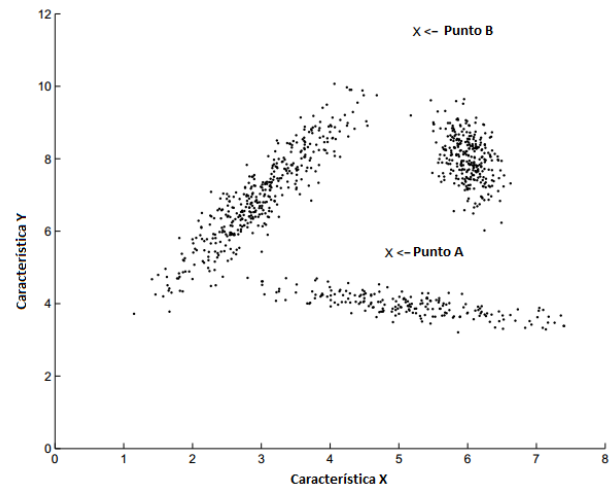


Figura 7 – Banco de datos 2 [Aggarwal, 2013]

Estos métodos pueden encontrar de forma efectiva outliers para los casos ilustrados en la figura 4 pero no para el **punto A** del caso ilustrado en la figura 7.

#### 7.4.1.2. ANÁLISIS DE VALORES EXTREMOS DE DATOS MULTIVARIADOS

El análisis de valores extremos también puede ser aplicado a datos multivariados en una variedad de formas. Algunas de estas formas tratan de modelar la distribución subyacente de forma explícita, mientras que otras se basan en un análisis estadístico más general, el cual no asume ninguna distribución particular.

Sin embargo, la determinación de tales outliers puede ser útil en muchos escenarios especializados. Por ejemplo, en casos donde desviaciones múltiples pueden ser asociadas con registros previos, el análisis de extremos multivariados puede ser útil. Un ejemplo es el clima donde múltiples atributos como la temperatura, presión y humedad son medidas en distintas ubicaciones espaciales y la desviación espacial local respecto a los valores esperados es calculada como un paso intermedio. Estas desviaciones respecto de los valores esperados sobre distintos atributos podría ser necesario transformarlas a una sola puntuación significativa [Aggarwal, 2013].

### 7.4.1.3. MODELOS ESTADÍSTICOS Y PROBABILÍSTICOS

En los modelos estadísticos y probabilísticos, los datos son modelados en la forma de una distribución de probabilidad concreta, y los parámetros de este modelo son aprendidos en base al banco de datos procesado. La asunción principal en estos modelos es la distribución de los datos con la cual será realizado el modelo subyacente y no es un tema menor ya que puede determinar el éxito o el fracaso del análisis a realizar.

Una de las mayores ventajas que tienen los modelos probabilísticos es que pueden ser fácilmente aplicados a, virtualmente, cualquier tipo de dato (o mezcla de tipos), siempre que un modelo generativo apropiado pueda ser encontrado para cada componente. Como estos modelos trabajan con probabilidades, el problema de la normalización de los datos se tiene en cuenta en las decisiones generativas asumidas.

Una desventaja de los modelos probabilísticos es que intentan ajustarse a una distribución particular la cual, no siempre puede ser la apropiada para los datos. Esto suele ser un inconveniente importante para el propósito de la detección de anomalías ya que se presupone un buen conocimiento del proceso generativo de los datos y la elección del modelo influye directamente sobre los resultados obtenidos [Aggarwal, 2013].

### 7.4.1.4. MÉTODOS BASADOS EN PROFUNDIDAD

En los métodos basados en profundidad, las envolventes convexas, como la presentada en la figura 8, son usadas para encontrar outliers.

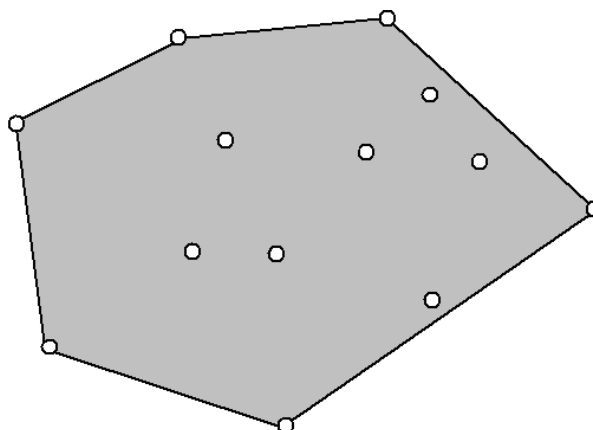


Figura 8 – Envolvente convexa

La envolvente convexa de un conjunto de puntos  $S$  (figura 9a) es la región convexa más pequeña que contiene todos los puntos en  $S$  (figura 9b).

La idea es que aquellos puntos en los límites exteriores de los datos yacen en las esquinas de las envolventes convexas.

En base a alguna definición de profundidad propia del método, los datos son organizados en capas en el espacio con la expectativa que las capas más superficiales son las que contendrán los datos outliers con mayor probabilidad que las capas más profundas [Knorr, 2002].

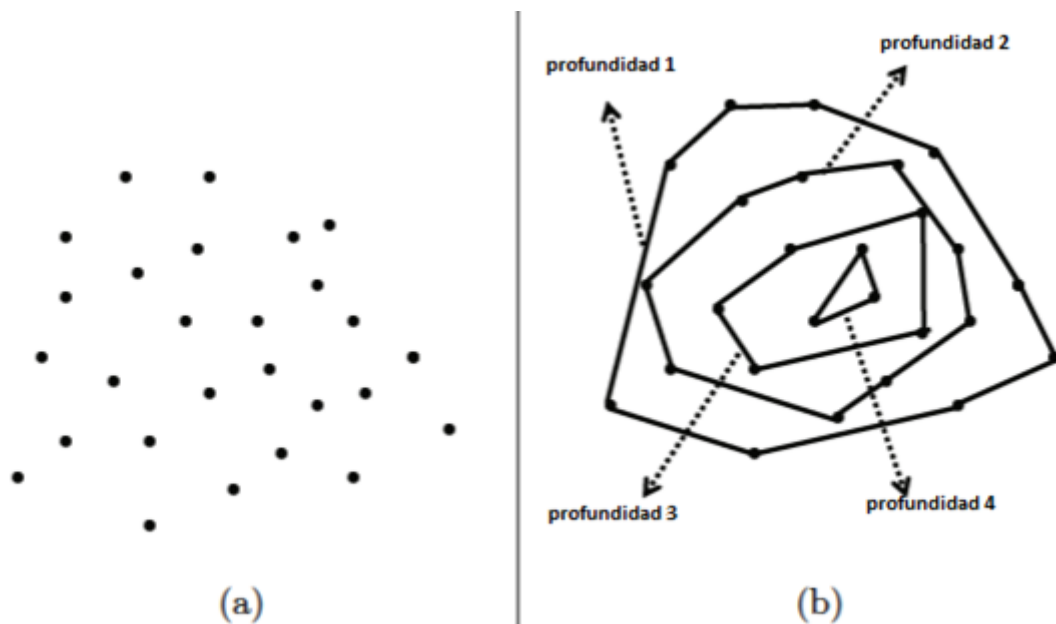


Figura 9 – Detección de outliers basados en profundidad [Aggarwal, 2013]

Estos métodos evaden el problema previamente mencionado de tener que ajustar los datos a una distribución específica y, conceptualmente, permiten el procesamiento de datos multidimensionales. Sin embargo, en la práctica, el procesamiento de capas  $k$ -dimensionales implica el procesamiento de  $k$ -dimensionales envolventes convexas [Knorr, 2002].

La idea es que los puntos en los límites externos de los datos yacerán en las esquinas de la envolvente convexa y serán outliers con mayor probabilidad.

*Pseudocódigo* [Aggarwal, 2013]:

```

algoritmo EncontrarOutliersPorProfundidad( D: ConjuntoDeDatos;
                                             r: UmbralDePuntuacion)
begin
    k=1;
    repeat
        Encontrar conjunto S de esquinas de la envolvente convexa de D;
        Asignar profundidad k a los puntos en S;
        D=D-S;
        k=k+1;
    until (D está vacío)
    Reportar puntos con profundidad (como máximo) r como outliers;
end;
    
```

Un algoritmo basado en profundidad procede en una forma iterativa. En la iteración número  $k$ , todos los puntos en las esquinas de la envolvente convexa se remueven del conjunto de datos. Estos puntos son asignados a una profundidad de  $k$ , Estos pasos se repiten hasta que el conjunto de datos queda vacío. Todos los puntos con profundidad de máximo  $r$  son reportados como outliers.

La complejidad computacional de los métodos basados en una envolvente convexa se incrementa exponencialmente de acuerdo a la dimensión de los datos. Además, con una dimensionalidad creciente, una porción más grande de los puntos de datos yacen en las esquinas de la envolvente convexa. Esto es así porque una envolvente convexa de dimensión  $d$  contiene al menos  $2^d$  puntos. Esto no sólo hace que los métodos sean computacionalmente imprácticos, sino también ineficaces en dimensionalidades más grandes [Aggarwal, 2013].

*Peeling* y *profundidad del contorno* son ideas estudiadas por Preparata y Shamos en el libro “Computational Geometry – Springer” (1988); y Ruts y Rousseeuw el trabajo “Computing depth contours of bivariate point clouds” (1996). En 1-D *peeling* (o pelar) se refiere al borrado sucesivo de orden más bajo y más alto del conjunto de datos mientras que en dimensiones más grandes, se refiere al borrado sucesivo de los puntos que yacen en la envolvente convexa de los puntos restantes. La *profundidad del contorno* también es un conjunto convexo, pero no cada vértice en la profundidad del contorno necesita estar presente en el conjunto de datos. Hay exactamente  $k$  puntos fuera de cada borde de la  $k$  – *ésima* profundidad de contorno [Knorr, 2002].

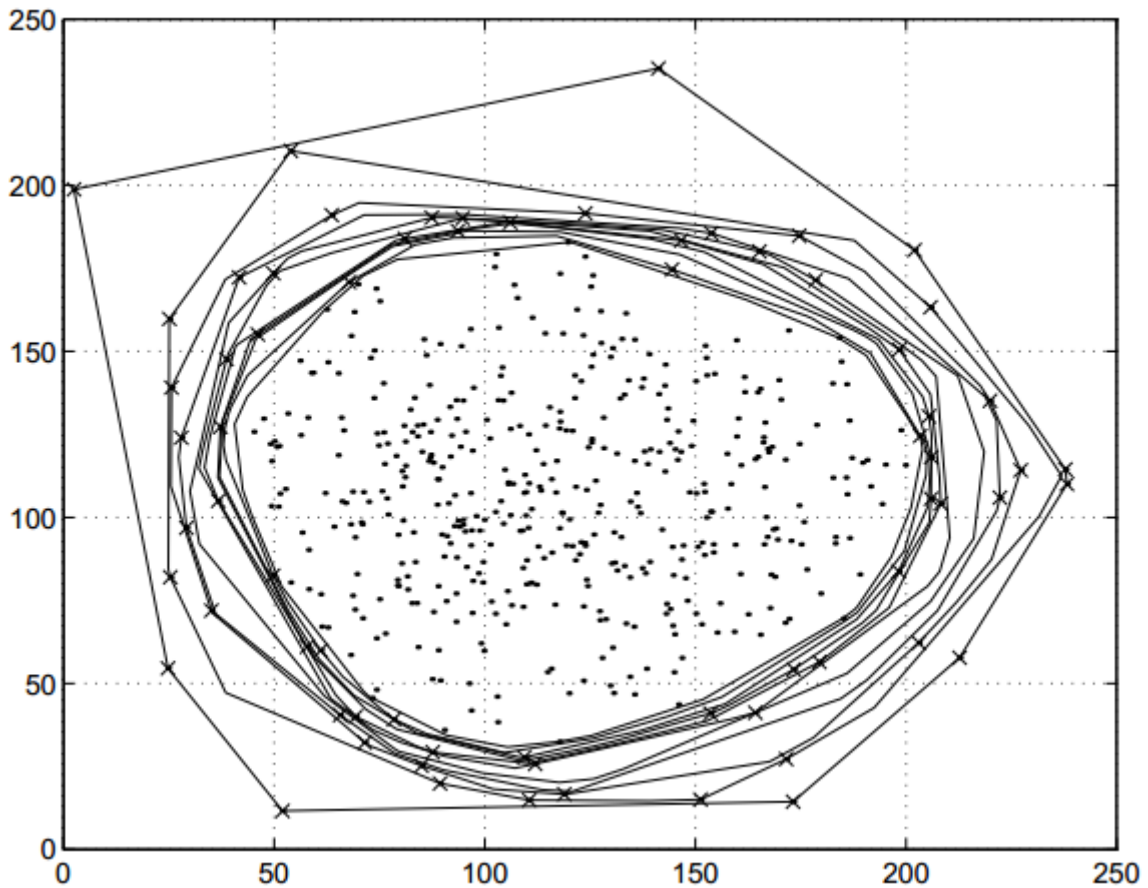


Figura 10 – Profundidades de contorno para un conjunto de datos de 500 puntos [Knorr,2002]

La figura 10 muestra las primeras profundidades de contorno para un conjunto de datos de 500 puntos, producido por el algoritmo FDC [Johnson y otros, 1998]. La mediana es el punto más profundo del conjunto de datos.

Los métodos existentes basados en profundidad sólo logran un rendimiento aceptable para  $k \leq 2$ . El algoritmo FDC (de Johnson y otros) puede procesar rápidamente profundidades de contorno en 2-D y significó una mejora importante de rendimiento sobre otros métodos. FDC permite procesar conjuntos de datos que contengan 100.000 puntos o más y aún mantener un buen rendimiento, sin embargo, el cálculo basado en profundidad se limita a valores muy pequeños de  $k$  (por ejemplo 2 o 3) [Knorr, 2002].



#### 7.4.1.5. MÉTODOS BASADOS EN DESVÍO

Los métodos basados en desvío miden el impacto de los outliers en la varianza de los datos. Por ejemplo un método busca medir cuánto se reduce la varianza de un conjunto de datos cuando un dato en particular se elimina. La asunción básica es que el outlier se ubica en los límites de los datos, se supone que al eliminar dicho dato la varianza se reducirá de manera significativa. Esto es esencialmente lo que realiza el método, examina la reducción de la complejidad cuando un dato se elimina. Dada la reducción se dice que el conjunto de datos se “suaviza”. De esta manera, se define un factor de suavización para un conjunto de datos  $R$  de la siguiente manera

**Factor de suavización:** El factor de suavización (smoothing factor)  $SF(R)$  para un conjunto de datos  $R$  es la reducción en la varianza del conjunto de datos, cuando el conjunto de puntos en  $R$  se elimina de la muestra original.

Los outliers son definidos como conjunto de excepciones  $E$  tales que su eliminación causa la máxima reducción en la varianza de la muestra. En otras palabras, para cualquier subconjunto de datos  $R$ , se debe cumplir

$$SF(E) \geq SF(R)$$

Si más de un conjunto tiene la misma reducción de la varianza, entonces se elige el más pequeño. Esto sigue el principio estándar de encontrar el conjunto de datos que incrementa la descripción de la muestra todo lo posible, en el menor tamaño posible. La determinación del tamaño óptimo de  $E$  es un problema muy complicado porque existen  $2^N$  posibilidades para un conjunto de datos de  $N$  puntos. Un buen aspecto de este enfoque es que es independiente de la distribución y puede ser aplicado a cualquier conjunto de datos, siempre que se pueda construir una buena definición del factor de suavización [Aggarwal, 2013].

#### 7.4.1.6. MÉTODOS BASADOS EN ÁNGULOS

La idea detrás de los métodos basados en ángulos es que los puntos en los límites de los datos probablemente encierren los datos completos dentro de un ángulo menor mientras que los puntos en el interior tendrán, con mayor probabilidad, datos

alrededor en diferentes ángulos. Por ejemplo, si consideramos los puntos  $A$  y  $B$  de la figura 11, en la cual el punto  $A$  es un outlier y el punto  $B$  se encuentra en el interior del conjunto de datos. Puede observarse claramente que todos los datos yacen dentro de un ángulo limitado centrado en el punto  $A$ . Por otro lado, no es el caso del punto  $B$ . En el ejemplo presentado, los ángulos entre los diferentes pares de puntos pueden variar extensamente, de hecho, mientras más aislado se encuentre un punto respecto del resto, más pequeño será el ángulo subyacente. Así, los puntos que contengan menor espectro de ángulo serán outliers, mientras que aquellos con espectro de ángulo mayor no lo serán.

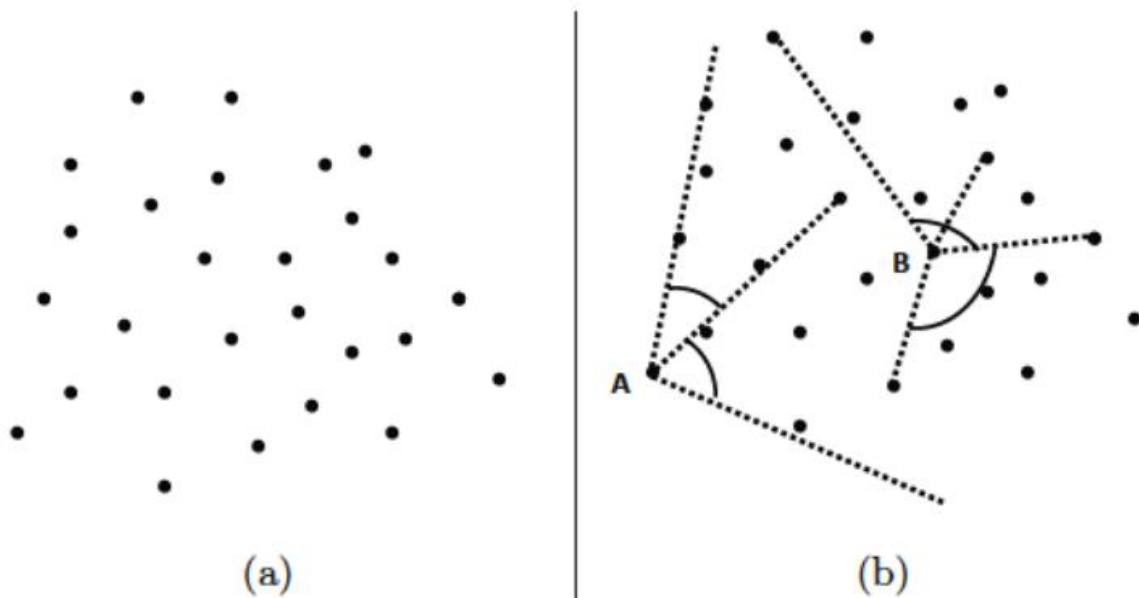


Figura 11 – Detección de outliers basados en ángulos [Aggarwal, 2013]

Los métodos basados en ángulos calculan para un punto  $\bar{X}$  el denominado *factor outlier basado en ángulo* (ABOF – por “*angle based outlier factor*”) y se lo escribe como  $ABOF(\bar{X})$ .

El  $ABOF(\bar{X})$  puede ser calculado de diversas maneras. La forma más ingenua es tomar de a tres puntos de todas las formas posible y calcular los  $O(N^3)$  ángulos entre los diferentes vectores. Los valores  $ABOF$  pueden ser explícitamente calculados con estos valores. Sin embargo una aproximación como esa es

impráctica para conjuntos de datos muy grandes. Se han propuesto mejoras basadas en distintas optimizaciones [Aggarwal, 2013].

Los detalles de las distintas optimizaciones fueron discutidos en el trabajo “Angle-Based Outlier Detection in High-dimensional Data” presentado en la 14ª conferencia internacional sobre KDD y Data mining (ACM SIGKDD) por Hans Peter Kriegel, Matthias Schubert y Arthur Zimek en el año 2008 [Kriegel y otros, 2008].

En el año 2012 Ninh Pham y Rasmus Pagh propusieron un algoritmo para este problema en su trabajo “A Near-linear time Approximation Algorithm for Angle-based Outlier Detection in High-dimensional Data”. Este trabajo se presentó en el contexto de la Conferencia ACM KDD del año 2012 y se introduce un algoritmo basado en proyecciones aleatorias para aproximar la diferencia de ángulos entre pares de puntos en un conjunto de datos, lo que constituye una puntuación robusta de outliers para detectar patrones de outliers de grandes dimensiones. Los experimentos empíricos en datos sintéticos y del mundo real demuestran la escalabilidad, efectividad y eficiencia de la aproximación presentada al detectar outliers en conjuntos de datos grandes [Pham y otros, 2012].

#### 7.4.1.7. MÉTODOS BASADOS EN DISTANCIA

Los métodos basados en distancia son una clase popular de algoritmos para la detección de outliers a lo largo de una amplia variedad de dominios y definen el puntaje outlier basándose en las distancias de los vecinos más cercanos.

Los métodos de análisis basados en distancia trabajan con la asunción de que las distancias a los  $k$  vecinos más cercanos son mucho más grandes que la de los puntos normales [Aggarwal, 2013].

En este trabajo se abordan los métodos basados en distancia (Distance Based o *DB*) bajo la noción que un outlier se define como:

*“Un objeto  $O$  en un grupo de datos  $T$  es un outlier  $DB(p, D)$  si al menos una fracción  $p$  de los objetos en  $T$  se encuentran a una distancia mayor que la distancia  $D$  respecto de  $O$ ”* [Knorr y otros, 1999].

Respecto a esta definición Aggarwal, en el 2013, hace la siguiente reflexión:

“Esta definición es casi idéntica a la definición del  $k$ -vecindario más cercano al elegir como valor de  $p$  a  $(N - k)/N$  para un conjunto de datos que contenga  $N$  puntos. Como  $k$  típicamente es mucho más chico que  $N$ , el valor de  $p$  necesita estar muy cercano a la unidad para obtener resultados razonables. La mayoría de los algoritmos basados en distancia, por lo tanto, trabajan con el parámetro  $k$  porque es más simple y más intuitivo de entender” [Aggarwal, 2013].

Para abordar el estudio de los outliers basados en distancia, teniendo en cuenta las distintas distribuciones, se propone la siguiente generalización:

**$DB(p, D)$**  Unifica o generaliza una nueva definición  **$Def$**  para outliers si existen valores  $p_0$  y  $D_0$  tal que el objeto  $O$  es un outlier de acuerdo a  **$Def$**  si y solo si  $O$  es un outlier  **$DB(p_0, D_0)$**  [Knorr y otros, 1999]

Por ejemplo, para una distribución normal, los outliers son considerados puntos que se encuentran alejados tres o más desvíos estándar ( $\geq 3\sigma$ ) respecto a la media  $\mu$ .

*Definición:*

Sea  $T$  una variable aleatoria distribuida normalmente con media  $\mu$  y desvío estándar  $\sigma$ . Se define  **$Def_{normal}$**  de la siguiente manera:

$$t \in T \text{ es un outlier si y sólo si } \frac{t - \mu}{\sigma} \geq 3 \text{ o } \frac{t - \mu}{\sigma} \leq -3$$

Y se complementa la definición con el siguiente lema:

**$DB(p, D)$**  Unifica  **$Def_{normal}$**  con  $p_0 = 0,9988$  y  $D_0 = 0,13\sigma$ , esto significa que,  $t$  es un outlier de acuerdo a  **$Def_{normal}$**  si y solo si  $t$  es un outlier  **$DB(0,9988, 0,13\sigma)$**  [Knorr y otros, 1999]

La misma aproximación aplica a la distribución  $T$  de Student, la cual tiene colas más anchas que la distribución normal. El principio de usar la cola para identificar outliers también aplica a una distribución Poisson.

La correspondiente demostración de todos los lemas, el de la distribución normal, más los correspondientes para la distribución  $T$  de Student o la distribución Poisson

han sido estudiados, demostrados y documentados en el trabajo “A unified notion of outliers: properties and computation:” [Knorr y otros, 1997].

Sea  $N$  el número de objetos en el conjunto de datos de entrada  $T$ , y sea  $F$  la función que provee la distancia entre cualquier par de objetos en  $T$ . Para un objeto  $O$ , el  $D$ -vecindario de  $O$  contiene el conjunto de objetos  $Q \in T$  que se encuentran dentro de la distancia  $D$  de  $O$ . ( $\{Q \in T | F(O, Q) \leq D\}$ ). La fracción  $p$  es la mínima fracción de objetos en  $T$  que deben estar fuera del  $D$ -vecindario de un outlier. Para simplificar la discusión, se conviene que  $M$  es el número máximo de objetos dentro del  $D$ -vecindario de un outlier, es decir,  $M = N(1 - p)$ .

De la formulación anterior puede observarse que dado  $p$  y  $D$ , el problema de encontrar todos los outliers  $DB(p, D)$  puede ser resuelto al detectar el vecindario más cercano tomando como centro cada objeto  $O$ . Más específicamente, basado en una estructura multidimensional indexada, podemos ejecutar una búsqueda de rango con radio  $D$  para cada objeto  $O$ . Una vez que al menos  $(M + 1)$  vecinos son incluidos en el vecindario de  $O$ , se detiene la búsqueda y se descarta a  $O$  como outlier, de otra forma, se puede reportar  $O$  como un outlier.

La relevancia que adquieren los métodos basados en distancia es su granularidad. Para un mismo banco de datos, si se establecen distintos valores de  $p$  y  $D$ , se habilita una mejor capacidad para distinguir entre outliers débiles y fuertes [Aggarwal, 2013].

#### **7.4.1.7.1. DETERMINACIÓN DE LOS VALORES INICIALES PARA $p$ Y $D$**

Para abordar esta discusión, lo primero que hay que dejar en claro es que no hay un valor universalmente correcto para  $p$  y  $D$ . La idea es permitir al usuario determinar, durante el lanzamiento de los procesos para buscar outliers, los valores específicos de los parámetros de la búsqueda. La forma de trabajo propuesta en este trabajo final incorpora al usuario y a su conocimiento sobre el dominio de los datos en el proceso de exploración. Esto es intencional porque sólo un usuario con conocimiento del dominio de los datos a procesar puede determinar si las tuplas retornadas son significativas o no. Directamente relacionada con esta noción de

“significativa” esta la elección del usuario de  $p$  y  $D$ . Se consideran algunas heurísticas para la elección apropiada de estos dos parámetros.

Intuitivamente un outlier ocurre de forma relativamente infrecuente, y es por lo tanto, razonable seleccionar un valor de  $p$  bastante cercano a la unidad. Supongamos un conjunto de datos de tamaño  $N = 1000$ , y  $p = 0,995$ . Dada la fórmula  $M = N(1 - p)$ , donde  $M$  es el máximo número de tuplas en el  $D$ -vecindario (tuplas a  $D$  distancia) de un outlier  $Q$ , significa solamente que  $M = 5$  de las 1000 tuplas en el conjunto de datos pueden encontrarse dentro del  $D$ -vecindario de  $Q$ . Si  $D$  es suficientemente grande, muy pocos outliers se detectarían, si es que alguno se detecta. Claro que, para un valor suficientemente pequeño de  $D$ , es posible que *cada* tupla en el conjunto de datos sea identificada como un outlier. Basados en experiencias previas para valores grandes de  $N$ ,  $p$  debería acercarse a la unidad [Knorr y otros, 1999]. Por ejemplo,  $p = 0,995$  podría bastar cuando  $N \approx 10^3$ , pero puede ser demasiado pequeño cuando  $N \approx 10^6$ . Para este último caso, un valor de  $p = 0,99995$  sería más apropiado.

Un método para determinar un valor inicial razonable para  $D$  dado un  $p$  fijo cercano a la unidad y en ausencia de otra información adicional, es a través de un muestreo. El muestreo puede ser complicado. Debido a la ocurrencia infrecuente e impredecible de los outliers, es difícil muestrear y proveer un alto grado de confianza para  $D$ . En este trabajo se permitirá al usuario incrementar o decrementar el valor de  $D$  pero no se basará una propuesta del valor de  $D$  en ningún muestreo [Knorr y otros, 1999].

#### 7.4.1.8. MODELOS LINEALES

Muchos conjuntos de datos muestran correlaciones significativas entre sus diferentes atributos. En tales escenarios, los modelos lineales pueden proveer una herramienta efectiva para la detección o remoción de los outliers.

Las diferentes dimensiones en los conjuntos de datos reales están fuertemente correlacionadas entre sí. Esto es así porque, generalmente, los distintos atributos son generados por el mismo proceso relacionándolos estrechamente. Esta es la base de los modelos lineales. La asunción de una correlación aproximadamente lineal para el conjunto de datos es crítica para garantizar la efectividad del modelo.

Hay dos principales clases de modelos lineales muy conocidas, la primera clase usa regresión estadística para modelar variables dependientes e independientes a fin de determinar tipos específicos de dependencias en los datos. La segunda clase usa análisis de componentes tratando todos los atributos de forma homogénea para determinar los subespacios dimensionales inferiores de proyección. Técnica y matemáticamente ambas clases son bastante similares, la diferencia principal entre ambas es cómo se formula la función objetivo de los dos modelos.

La mayor limitación de los modelos lineales es que no reconoce que el comportamiento de la correlación puede diferir en distintas localidades e intenta ajustar el conjunto de datos total en un solo modelo global [Aggarwal, 2013].

De los modelos lineales tal vez el más representativo es el modelo de regresión lineal. En los modelos de regresión lineal, los valores observados de los datos son modelados en un sistema de ecuaciones lineales. Específicamente, las diferentes dimensiones en los datos se relacionan entre sí usando un conjunto de coeficientes lineales. Como el número de valores observados, típicamente es mucho más grande que la dimensionalidad de los datos, el sistema de ecuaciones está *sobre-determinado* y no puede ser resuelto de manera exacta. Por lo tanto, estos modelos funcionan prestando atención a las desviaciones de cada punto respecto del valor predicho por el modelo lineal y es la elección de la función del error la que determina si una variable se trata de forma especial o de forma homogénea al resto [Aggarwal, 2013].

Muchas aplicaciones que realizan análisis de outliers no tratan ninguna variable como especial y la definición de outlier, generalmente, se basa sobre la distribución total de los datos. Sin embargo, el caso especial del análisis de regresión con variables dependientes es importante en muchas aplicaciones. Esto es así porque en muchos dominios de la vida real tales como datos temporales y espaciales, los atributos son particionados en contextuales y de comportamiento. En tales casos, un atributo de comportamiento particular se puede predecir tomando en cuenta una función de los atributos de comportamiento en un vecindario contextual. De esta forma se puede determinar las desviaciones respecto los valores esperados,

dándoles una gran relevancia a las variables dependientes. Los outliers se definen sobre la base de cómo variables independientes impactan en las variables dependientes y las anomalías entre las relaciones de las variables independientes se consideran menos importantes [Aggarwal, 2013].

A los modelos de regresión podemos definirlos según un criterio denominado  $Def_{Regresión}$  y demostrar que  $DB(p, D)$  unifica  $Def_{Regresión}$  [Knorr, 1997].

Consideremos un modelo de regresión lineal simple dado por  $y = \alpha + \beta x$ . Supongamos que el conjunto de datos  $T$  consiste en observaciones del tipo  $(x_i, y_i)$  para  $i = 1, 2, \dots, N$ , que ajustan a este modelo. Denotamos el residuo o error, que es, la diferencia entre el valor observado y el valor ajustado para la observación  $i$  –ésima, con  $e_i$ , resultando la ecuación final del modelo en  $y = \alpha + \beta x + e$ . Una forma posible de identificar outliers es encontrando residuales fuera de lo común. Un outlier fuera de lo común puede ser considerado un residual cuyo valor es mucho más grande que el resto en valor absoluto, al menos, tres desviaciones estándares desde la media de los residuos. Si se asume de manera simplificada (cosa que algunos autores hacen) que los residuos individuales son independientes y siguen una distribución normal, entonces se puede identificar fácilmente a los outliers tomando como parámetro su desvío respecto de la media [Knorr y otros, 1999].

Hay dos grandes clases de métodos para la detección de outliers con regresión, los métodos directos e indirectos. La siguiente lista son algoritmos para métodos de regresión lineal que han sido frecuentemente citados o recientemente publicados [Billor y otros, 2008]:

#### Métodos directos

- *HS93*: The sequential point addition algorithm de Hadi y Simonoff (1993)
- *PY*: The influence matrix algorithm de Peña y Yohai (1995)
- *SMR*: The clustering algorithm de Sebert y otros (1998)
- *BR*: The BACON regression method de Billor y otros (2000)
- *RWLS*: The Robust iteratively re-weighted least squares method de Billor y otros (2007)



### Métodos indirectos

- *LMS*: The least median of squares de Rousseeuw (1984)
- *M*: The M-estimator de Huber (1973)

En el año 2007 Nedret Billor y Gulsen Kiral iniciaron extensos experimentos con el fin de comparar distintos métodos generalmente citados o recientemente propuestos y así brindar un estudio comparativo del rendimiento de los algoritmos en situaciones extremas como ser altos niveles de contaminación, grandes dimensiones y grandes tamaños de muestra. Los experimentos realizados, así como sus resultados se encuentran debidamente documentados en el trabajo “*A comparison of Multiple Outlier Detection Methods for Regression Data*” del año 2008 [Billor y otros, 2008].

#### 7.4.1.9. MODELOS BASADOS EN PROXIMIDAD

Las técnicas basadas en proximidad definen un punto cualquiera como outlier si alrededor del punto se encuentra escasamente poblado.

La proximidad de un punto puede ser definido de varias formas sutilmente diferentes pero suficientemente similares. Las formas más comunes de definir la proximidad para el análisis outlier son [Aggarwal, 2013]:

- *Basado en clusters*: la no pertenencia de un punto a un cluster, la distancia a otros clusters o el tamaño del cluster más cercano son algunos de los criterios utilizados para calcular la puntuación outlier.
- *Basado en distancia*: la distancia de un punto a sus  $k$  vecinos más cercanos (u otras variantes) es utilizada para definir la proximidad. Los puntos con distancia grandes entre sus  $k$  vecinos más cercanos son identificados como outliers.
- *Basado en densidad*: el número de otros puntos dentro de una región local específica de un punto, es utilizado para definir la densidad local. Estos valores locales de densidad son utilizados como puntuación outlier. Otros métodos pueden ser utilizados para la estimación de la densidad.

La mayoría de estos métodos funcionan bien cuando los datos forman clusters definidos y los outliers pueden ser claramente distinguidos entre densas regiones de

datos. La diferencia entre estos métodos debe analizarse a nivel de detalle de cómo se define la proximidad. Estos métodos están naturalmente diseñados para detectar tanto ruido como anomalías aunque diferentes métodos son más adecuados dependiendo del tipo de valor atípico que se quiere detectar.

Los métodos basados en proximidad son extremadamente populares debido a que resultan intuitivos y simples de interpretar [Aggarwal, 2013].

#### 7.4.1.9.1. BASADOS EN CELDAS (DISTANCIA)

Los métodos basados en celdas son exponenciales en dimensionalidad pero lineales en los puntos de datos. En esta técnica el espacio de datos se divide en celdas, el ancho es una función del umbral  $D$  y la dimensionalidad de los datos. Específicamente, cada dimensión se divide en celdas de ancho  $\frac{D}{(2\sqrt{d})}$ . La presencia de puntos en una celda dada así, como en celdas adyacentes, satisface ciertas propiedades que son explotadas para una mayor eficiencia de procesamiento [Aggarwal, 2013].

|  |    |    |    |    |    |    |    |
|--|----|----|----|----|----|----|----|
|  |    |    |    |    |    |    |    |
|  | L2 | L2 | L2 | L2 | L2 | L2 | L2 |
|  | L2 | L2 | L2 | L2 | L2 | L2 | L2 |
|  | L2 | L2 | L1 | L1 | L1 | L2 | L2 |
|  | L2 | L2 | L1 | *X | L1 | L2 | L2 |
|  | L2 | L2 | L1 | L1 | L1 | L2 | L2 |
|  | L2 | L2 | L2 | L2 | L2 | L2 | L2 |
|  | L2 | L2 | L2 | L2 | L2 | L2 | L2 |
|  |    |    |    |    |    |    |    |

Figura 12 – Particiones del espacio de datos basado en celdas [Aggarwal, 2013]

El método se explica detalladamente junto con las propiedades y lemas en el trabajo de Ed Knorr “*Outliers and data mining*” del año 2002 [Knorr, 2002].

La aproximación puede explicarse más fácilmente en un caso de dos dimensiones. Se forma una grilla con celdas sucesivas que se encuentran a una distancia máxima

de  $\frac{D}{(2\sqrt{d})}$ . Algo importante que hay que tener en cuenta es que el número de celdas se basa en la partición del espacio de datos y es independiente del número de datos.

Para una celda dada, los vecinos  $L_1$  están definidos por el conjunto de celdas que son alcanzables por esa misma celda cruzando tan solo una celda. Los vecinos  $L_2$  son aquellas celdas que se obtienen al cruzar 2 o 3 celdas. Un caso particular de una celda marcada con  $X$  junto con sus vecinos  $L_1$  y  $L_2$  puede observarse en la figura 12 [Aggarwal, 2013].

De la figura es evidente que la celda marcada con una  $X$  tiene 8 vecinos  $L_1$  y 40 vecinos  $L_2$ . Teniendo esto en cuenta las siguientes propiedades se pueden observar [Knorr, 2002]:

- La distancia entre un par de puntos en una celda es como máximo  $D/2$
- La distancia entre un punto y un punto vecino  $L_1$  es como máximo  $D$
- La distancia entre un punto y un punto vecino  $L_r$  (*donde*  $r > 2$ ) es al menos  $D$

Las únicas celdas por las cuales no se pueden sacar conclusiones inmediatas son aquellas en  $L_2$ . Esta representa la región de incertidumbre para los puntos en esas celdas. Para esos casos, el cálculo de la distancia necesita ser realizado de forma explícita [Aggarwal, 2013]. Al mismo tiempo un número de reglas pueden ser definidas con el fin de inmediatamente declarar algunas fracciones de los datos como outliers o no outliers. Siendo  $k$  la cantidad umbral de objetos para la determinación de outliers, las reglas son [Knorr, 2002]:

- Si más de  $k$  puntos están contenidos en una celda junto con sus vecinos  $L_1$ , entonces *ninguno* de aquellos puntos son outliers
- Si menos de  $k$  puntos están contenidos en una celda  $A$  y sus vecinos  $L_1$  y  $L_2$ , entonces *todos* los puntos en la celda  $A$  son outliers

#### 7.4.1.9.2. BASADOS EN ÍNDICE (DISTANCIA)

El principal inconveniente en los esquemas basados en distancia es el tiempo potencialmente largo que se requiere para realizar los cálculos de a pares entre los puntos. Las estructuras indexadas pueden ser utilizadas para mejorar la complejidad de cálculos basados en distancia.

Sridhar Ramaswamy, Rajeev Rastogi y Kyuseok Shim realizaron una primera aproximación a esta línea de métodos [Ramaswamy y otros, 2000]. En la propuesta los puntos son rankeados en orden decreciente por la distancia a sus  $k$  vecinos más cercanos. Los primeros  $n$  puntos de la lista son reportados como outliers. Por lo tanto, el umbral está definido por el ranking en vez del valor de la distancia. Muchas variantes de esta definición son usadas en este tipo de métodos como ser la distancia absoluta o la distancia promedio a los  $k$  vecinos más cercanos para determinar la puntuación outlier [Aggarwal, 2013].

Sea  $N$  el número de objetos en el conjunto de datos de entrada  $T$ , y sea  $F$  la función de distancia subyacente que provee la distancia entre cualquier par de objetos en  $T$ . Para un objeto  $O$ , el vecindario  $D$  de  $O$  contiene el conjunto de objetos  $Q \in T$  que se encuentran dentro de la distancia  $D$  de  $O$ . ( $\{Q \in T | F(O, Q) \leq D\}$ ). La fracción  $p$  es la mínima fracción de objetos en  $T$  que deben estar fuera del vecindario  $D$  de un outlier. Para simplificar,  $M$  es el número máximo de objetos dentro del vecindario  $D$  de un outlier, es decir  $M = N(1 - p)$ .

De la formulación anterior es obvio que dado  $p$  y  $D$ , el problema de encontrar todos los outliers  $DB(p, D)$  puede ser resuelto al detectar el vecindario más cercano tomando como centro cada objeto  $O$ . Más específicamente, basado en una estructura multidimensional indexada, se puede ejecutar una búsqueda de rango con radio  $D$  para cada objeto  $O$ . Una vez que al menos  $(M + 1)$  vecinos son incluidos en el vecindario de  $O$ , se detiene la búsqueda y se descarta a  $O$  como outlier, de otra forma, se puede reportar  $O$  como un outlier [Knorr, 2002].

Análisis de esquemas de indexado multidimensional revelan que, para variantes de árboles  $R$ , árboles  $k - d$ , y árboles  $X$ , la complejidad menor que puede tener una

búsqueda por rango es  $\Omega(N^{1-\frac{1}{k}})$ . Mientras  $k$  crece, el rango rápidamente se reduce a  $O(N)$ , ofreciendo en el mejor caso, una mejora constante de tiempo pareciéndose a la búsqueda secuencial. Por lo tanto, el procedimiento anteriormente descrito para encontrar todos los outliers  $DB(p, D)$  tiene en el peor de los casos una complejidad de  $O(kN^2)$  [Knorr, 2002]. Vale la pena notar dos puntos:

- Comparado con las aproximaciones basadas en profundidad, las cuales tienen una complejidad mínima de  $\Omega\left(N^{\frac{k}{2}}\right)$ , los outliers  $DB$  escalan mucho mejor con dimensionalidad creciente. El framework de outliers  $DB$  es aplicable y computacionalmente factible para conjunto de datos con muchos atributos ( $k \geq 5$ ).
- El análisis anterior solo considera el tiempo de búsqueda. Cuando se utiliza un algoritmo basado en índices, se asume fuertemente que el índice correcto existe. En otras palabras, un gran costo oculto detrás de un algoritmo basado en índices es el esfuerzo de crear el índice en primer lugar. Teniendo en cuenta el costo de crear el índice, aun sin contar el costo de la búsqueda casi siempre pone a los algoritmos basados en índices como no competitivos.

#### 7.4.1.9.3. LOF (DENSIDAD)

El método LOF (*Local Outlier Factor*) fue presentado en la conferencia ACM SIGMOD del año 2000 en Dallas, Texas. Sus autores Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng y Jörg Sander propusieron un método que le asigna a cada objeto *cierto grado* de outlier. Este grado se llama factor outlier local o LOF (por *local outlier factor*). Se lo denomina local porque el grado depende de que tan aislado se encuentra un objeto respecto del vecindario circundante [Breuning y otros, 2000].

Una observación sobre el método LOF es que mientras ha sido popularmente clasificado en la literatura como una aproximación basada en densidad, puede ser más simplemente entendido como una aproximación relativa de distancia con suavizado [Aggarwal, 2013].

LOF es una cuantificación del aislamiento de cada dato, capaz de ajustarse a las variaciones de las diferentes densidades. Para un punto dado  $\bar{X}$ ,  $D^k(\bar{X})$  será la distancia al  $k$ -vecindario más cercano de  $X$ , y  $L_k(\bar{X})$  el conjunto de puntos dentro de la distancia  $k$ -vecindario más cercano de  $\bar{X}$ . Se nota que, típicamente,  $L_k(\bar{X})$  contendrá  $k$  puntos pero puede pasar que este conjunto tenga más de  $k$  puntos debido a los vínculos que se dan en la distancia al  $k$ -vecindario más cercano.

Luego, la distancia de acceso  $R_k(\bar{X}, \bar{Y})$  del objeto  $\bar{X}$  respecto de  $\bar{Y}$  se define como la máxima distancia  $Dist(\bar{X}, \bar{Y})$ , entre el par  $(\bar{X}, \bar{Y})$  y la distancia de  $\bar{Y}$  a su  $k$ -vecindario más cercano.

$$R_k(\bar{X}, \bar{Y}) = \max\{Dist(\bar{X}, \bar{Y}), D^k(\bar{Y})\}$$

La *distancia de acceso* no es simétrica entre  $\bar{X}$  e  $\bar{Y}$ . Intuitivamente, cuando  $\bar{Y}$  se encuentra en una región densa y la distancia entre  $\bar{X}$  e  $\bar{Y}$  es grande, la distancia de acceso de  $\bar{X}$  es igual a la verdadera distancia  $Dist(\bar{X}, \bar{Y})$ . Por otra parte, cuando las distancias entre  $\bar{X}$  e  $\bar{Y}$  son pequeñas, la distancia de acceso se suaviza por la distancia de  $\bar{Y}$  a su  $k$ -vecindario más cercano. Mientras más grande sea el valor de  $k$ , mayor es como se suaviza, y en consecuencia, las distancias de acceso respecto a los diferentes puntos se volverán similares.

La *distancia de acceso promedio*  $AR_k(\bar{X}, \bar{Y})$  de un punto  $\bar{X}$  respecto a su vecindario  $L_k(\bar{X})$  se define como el promedio de sus distancias de acceso a todos los puntos en su vecindario.

$$AR_k(\bar{X}) = MEDIA_{\bar{Y} \in L_k(\bar{X})} R_k(\bar{X}, \bar{Y})$$

El *factor outlier local* es entonces igual al ratio medio de  $AR_k(\bar{X})$  a los valores correspondientes de todos los puntos en el  $k$ -vecindario más cercano de  $\bar{X}$ .

$$LOF_k(\bar{X}) = MEDIA_{\bar{Y} \in L_k(\bar{X})} \frac{AR_k(\bar{X})}{AR_k(\bar{Y})}$$

El uso de ratios de distancia en la definición asegura que el comportamiento de la distancia local está bien representado en esta definición [Aggarwal, 2013].

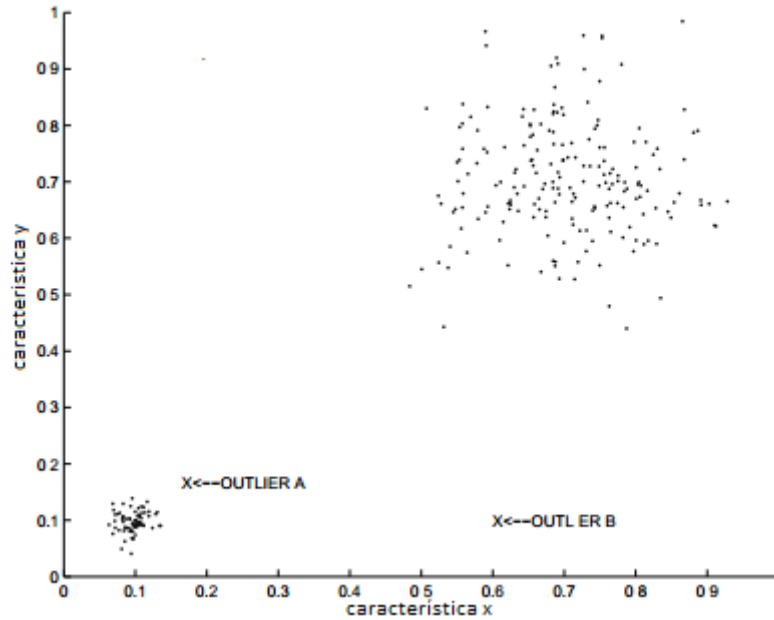


Figura 13 – Impacto de la densidad local en outliers [Breuning y otros, 2000]

Como resultado, los valores de los objetos dentro de un cluster son a menudo cercanos a 1, cuando los puntos en el cluster están homogéneamente distribuidos. En el ejemplo de la figura 13, los valores LOF de los puntos de ambos clusters serán bastante cercanos a 1, aunque las densidades de ambos clusters son diferentes. Por otra parte, los valores LOF de los dos puntos señalados como outliers serán mucho más grandes ya que serán calculados en término de los ratios de la distancia promedio de acceso a sus vecinos. En la práctica, el valor máximo de  $LOF_k(\bar{X})$  sobre un rango de diferentes valores de  $k$  es utilizado como el puntaje outlier con el que se crea un ranking de los diferentes objetos [Breuning y otros, 2000].

Como conclusión de la presentación del método los autores consideran que, para ciertas situaciones, proponer una etiqueta binaria para los outliers, no tiene tanto sentido como asociar un grado de outlier a cada punto, dependiendo del vecindario circundante. LOF captura exactamente este grado relativo de aislamiento [Breuning y otros, 2000].

El método LOF puede ser interpretado tanto como un método basado en distancia o como un algoritmo basado en densidad, dependiendo en la forma en que se lo presenta. La conexión real que tiene LOF con la densidad de los datos yace en su intuitiva capacidad de ajustarse a densidades de datos variables con el uso de distancias relativas [Aggarwal, 2013].

#### 7.4.1.9.4. LOCI (DENSIDAD)

En el año 2003, en el marco de la 19ª conferencia internacional sobre ingeniería de datos en Bangalore, India se propone LOCI (*Local Correlation Integral*) [Papadimitriou y otros, 2003] un método realmente basado en densidad [Aggarwal, 2013].

LOCI define la densidad  $M(\bar{X}, r)$  de un punto  $\bar{X}$  en términos del número de puntos dentro de un radio  $r$  predeterminado alrededor del punto. Esto es referido como el conteo del vecindario del punto  $\bar{X}$ . Correspondientemente la densidad promedio  $AM(\bar{X}, r, \delta)$  en el  $\delta$ -vecindario de  $\bar{X}$  se define como la media de  $M(\bar{X}, r)$  para todos los puntos una distancia máxima  $\delta$  de  $\bar{X}$ . El valor de  $\delta$  también es referido como el vecindario de muestra de  $\bar{X}$ , y es siempre más grande que  $r$ .

Además, el valor de  $r$  siempre se elige como una fracción constante de  $\delta$ , no importa cuál sea el valor de  $\delta$  usado. El valor de  $\delta$  es un parámetro crítico en el análisis y múltiples valores de este parámetro son usados con el fin de proveer intuiciones analíticas a diferentes grados de granularidad. La densidad promedio se define formalmente como:

$$AM(\bar{X}, r, \delta) = \text{MEDIA}_{(\bar{Y}: \text{dist}(\bar{X}, \bar{Y}) \leq \delta)} M(\bar{Y}, r)$$

Respectivamente, el factor de desviación de multigranularidad  $MDEF(\bar{X}, r, \delta)$  al nivel  $\delta$  se expresa en términos del ratio de las densidades en un punto dado y sus vecindarios.

$$MDEF(\bar{X}, r, \delta) = 1 - \frac{M(\bar{X}, r)}{AM(\bar{X}, r, \delta)}$$

Mientras más grande sea el valor de  $MDEF(\bar{X}, r, \delta)$ , mayor es la puntuación outlier que adquiere el punto. Para poder convertir la puntuación adquirida por el punto en



una etiqueta binaria, la desviación  $\sigma(\bar{X}, r, \delta)$  de los diferentes valores de  $M(\bar{X}, r)$  dentro del vecindario de muestra de  $\bar{X}$  es calculado

$$\sigma(\bar{X}, r, \delta) = \frac{STD_{(\bar{Y}:dist(\bar{X},\bar{Y})\leq\delta)}M(\bar{Y}, r)}{AM(\bar{X}, r, \delta)}$$

Aquí el término STD corresponde a la función de desvío estándar calculado sobre todo el vecindario de muestra.

El valor de  $r$  es siempre la mitad de  $\delta$  para habilitar un cálculo rápido de aproximación. Múltiples valores de  $\delta$  es una forma de proveer una aproximación de multigranularidad para el análisis de outliers. Estos métodos pueden variar el radio de muestreo desde un valor mínimo que contenga al menos 20 puntos y un valor máximo que se extiende por la mayoría de los datos. Un punto es un outlier si su valor  $MDEF$  es inusualmente grande en cualquiera de los valores calculados a diferentes niveles de granularidad. Específicamente, el valor de  $MDEF$  necesitar ser al menos  $k \cdot \sigma(\bar{X}, r, \delta)$  donde  $k$  es generalmente 3. Esta elección de  $k$  es común en un análisis estadístico que asume que los datos están normalmente distribuidos ) [Papadimitriou y otros, 2003].

Las principales contribuciones que tuvo este método fueron [Papadimitriou y otros, 2003]:

- Como cualquier método propuesto LOCI puede detectar outliers y grupos de outliers (micro-clusters)
- En contraste a los métodos anteriores que delegan en el usuario la decisión última sobre si un punto es o no un outlier, se propone un método automático, basado en un recorte de los datos.
- El método trata correctamente densidades locales y múltiple granularidad
- El método es tan rápido como cualquier método anterior

### 7.4.2. OUTLIERS Y MINERÍA DE DATOS

Algoritmos de minería de datos sofisticados y automatizados son necesarios para dar soporte al trabajo de científicos, estadísticos y expertos de aplicaciones en ciertos dominios. Dado que la cantidad de datos existente crece exponencialmente, queda claro que una buena herramienta de minería de datos será necesaria para explotar estos datos para interés estratégico científico o comercial [Knorr, 2002].

El tamaño del conjunto de datos y el número de dimensiones han comprobado ser obstáculos clave para el análisis de los datos [Han y otros, 2006]. La minería de datos provee algoritmos que son escalables tanto en tamaño del conjunto de datos como en las dimensiones (variables a evaluar).

Los algoritmos de data mining muchas veces detectan tuplas (u objetos) que no se pueden clasificar porque sus características son tan particulares que no logran ser calificados en ningún grupo que el algoritmo haya definido. Muchos algoritmos de data mining están escritos para descartar estas anomalías perdiéndose la información que ese objeto puede brindar.

El principal problema que este trabajo final aborda es la identificación de outliers en grandes y multidimensionales conjuntos de datos. Se comparten los desafíos que enfrentan los algoritmos de data mining respecto de tamaño del conjunto de datos y dimensionalidad. De hecho, se trabaja con conjuntos de datos que no pueden ser estudiados en su conjunto en memoria principal por restricciones propias del algoritmo utilizado o porque la cantidad de memoria disponible es inferior a la requerida para cargar todo el banco de datos a procesar.

## 8. ESTADO DEL ARTE

Uno de los principales problemas del data mining, como así también de la detección de outliers, son los conjuntos de datos masivos. Algunos conjuntos de datos son tan grandes que muchos algoritmos bien establecidos se vuelven más lentos de usar por una potencial complejidad de tiempo cuadrática (notación  $O$  grande) asociada a la cantidad de objetos a procesar.

Una forma de mejorar el rendimiento de los algoritmos es optimizar el uso del hardware disponible. En el trabajo “*Parallel Algorithms for Distance-Based and Density-Based Outliers*” [Lozano y otros, 2005] se experimentó con la paralelización de dos algoritmos Bay y LOF habilitando la utilización de más de un procesador en la ejecución del algoritmo. Hoy en día, una computadora o dispositivo móvil típico (tablet, teléfono, etc.) tienen múltiples procesadores, pero cuando se trata de programar para obtener los beneficios de trabajar con todos los procesadores disponibles, la tarea se vuelve más complicada. Se consume más tiempo y hay código extra que administrar. A pesar del trabajo adicional de programación que involucra mezclar una técnica para encontrar outliers y técnicas específicas para paralelizar el algoritmo, se pueden obtener algoritmos más eficientes y que aprovechen todo el potencial de un equipo para abordar la problemática de conjuntos de datos masivos en la detección de outliers.

La aproximación por celdas fue propuesta por Ed Knorr y Raymond NG en 1998 en el trabajo “*Algorithms for Mining Distance-Based Outliers in Large Datasets*” [Knorr y otros, 1998]. Lo interesante de la propuesta fue que la detección de outliers basados en la distancia de un objeto a sus vecinos más cercanos, es no paramétrica, es decir, no se basa en alguna distribución específica [Lozano y otros, 2005].

Dada una medida de distancia determinada, un objeto  $x$  en un conjunto de datos  $T$  es un outlier con parámetros  $p$  y  $D$  si al menos una fracción  $p$  de los objetos de  $T$  están a una distancia mayor a  $D$  [Knorr y otros, 1999]. Esta definición tiene ciertas dificultades como la determinación de  $D$  y la falta de un ranking de outliers. Como se ha establecido previamente, la determinación de ambos parámetros  $p$  y  $D$  será delegada en el usuario que es quien conoce el dominio de los datos. Respecto a la

falta de ranking de outliers, esta es una debilidad conocida, en el trabajo “*LOF: Identifying Density-Based local Outliers*” [BREUNIG y otros, 2000] muestra, mediante un ejemplo, la debilidad de los métodos basados en distancia al identificar ciertos tipos de outliers.

En “*Parallel Algorithms for Distance-Based and Density-Based Outliers*” [Lozano y otros, 2005] se tomaron dos algoritmos, uno basado en distancia (Algoritmo de Bay) y otro basado en profundidad (el conocido *Local Outlier Factor* –LOF) ambos con buen rendimiento en tiempo de ejecución y detección de outliers. Se construyó la versión que incorpora paralelismo en la ejecución y se lo probó con cuatro bancos de datos distintos [Lozano y otros, 2005]:

- *Landsat*: consiste en 4.435 objetos. Cada objeto representa una imagen satelital con 36 características. Los datos se clasifican en seis clases
- *Shuttle*: lanzamientos de la NASA, contiene 43.500 objetos de entrenamiento con 9 atributos continuos. Los datos se clasifica en siete clases.
- *Census*: este conjunto de datos consiste en 14 características medidas en 32.561 individuos.
- *Covtype*: Conjunto de datos obtenidos del servicio forestal de Estados Unidos. Contiene medidas de siete tipos de suelos con 581.012 objetos y 54 atributos.

Como puede visualizarse en la figura 14, el algoritmo de Bay mejora su rendimiento cada vez que se habilita el uso de más procesadores con todos los bancos de datos utilizados. Cada procesador ejecuta el mismo algoritmo de Bay pero localmente y sobre su propio conjunto de datos, luego se sincroniza cada hilo de ejecución con un proceso maestro que administra los distintos procesos para la siguiente iteración. La versión “paralela” del algoritmo de Bay alcanza velocidad prácticamente lineal [Lozano y otros, 2005].

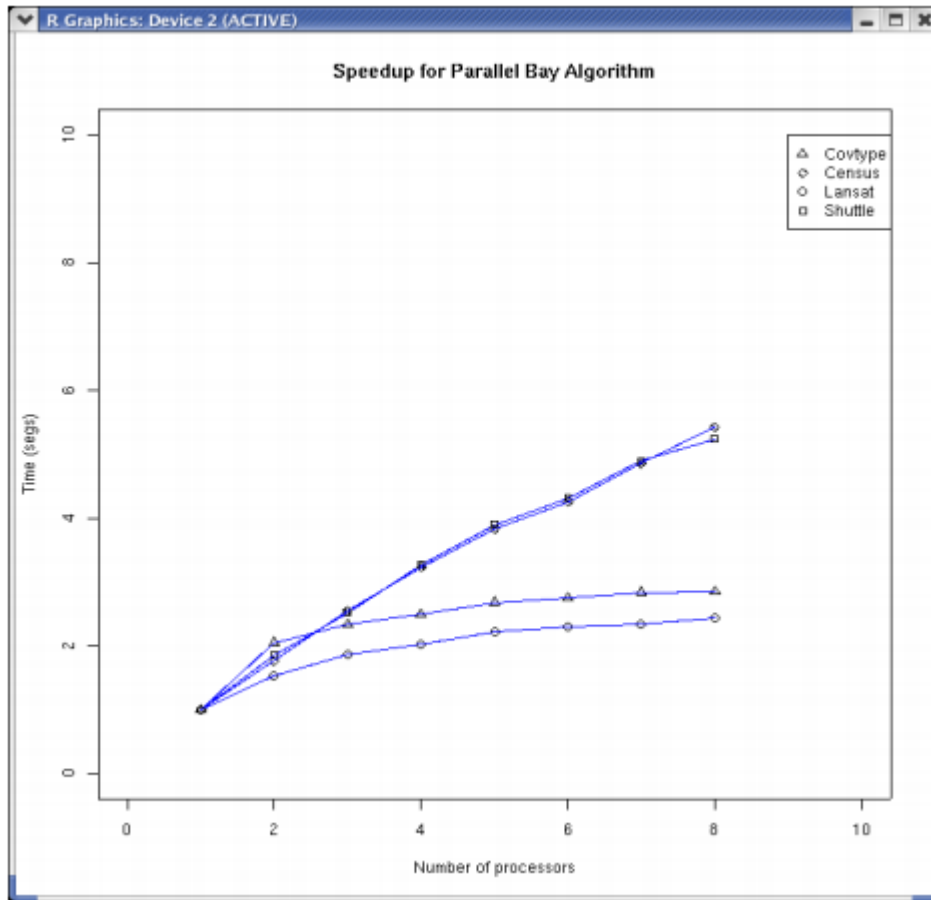


Figura 14 – Mejora de rendimiento para el algoritmo de Bay por procesador [Lozano y otros, 2005]

Por otro lado, el trabajo pesado en el algoritmo de LOF radica en el cálculo de las distancias a los  $k$ -vecindarios más cercanos para cada observación. El proceso principal le envía el conjunto de datos completo a cada proceso hijo y le asigna su tarea administrando una ida y vuelta con los procesos hijos que permiten llevar a cabo todo el proceso. El algoritmo alcanza una buena mejoría en los tiempos de ejecución porque hay poca comunicación entre los procesos [Lozano y otros, 2005]. El rendimiento del algoritmo LOF, a medida que se habilita el uso de más procesadores puede observarse en la figura 15.

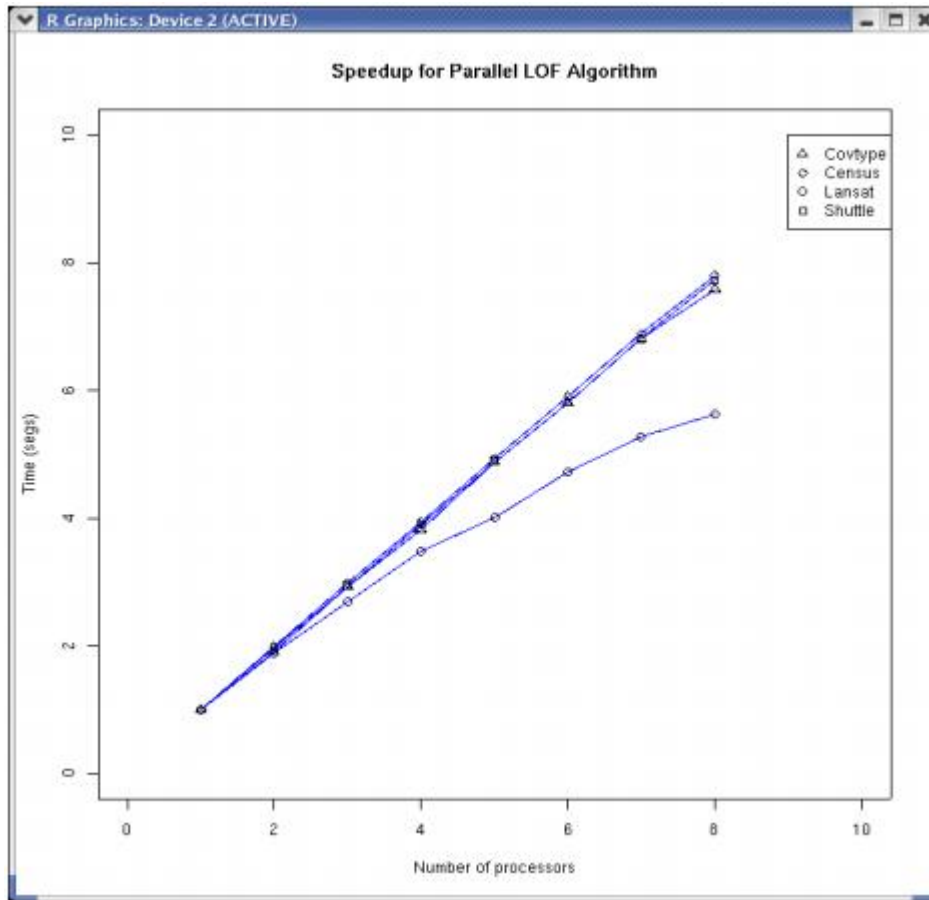


Figura 15 – Mejora de rendimiento para el algoritmo LOF por procesador [Lozano y otros, 2005]

El algoritmo basado en celdas propuesto por [Knorr y otros, 1998] resulta intuitivo y simple de interpretar [Aggarwal, 2013], a su vez sus limitaciones generan desafíos que dan lugar a la innovación ya que este método puede procesar de forma muy eficiente hasta 4 dimensiones (5 en algunos casos) pero luego decrece su rendimiento e incluso puede imposibilitarse su ejecución debido su complejidad exponencial (la complejidad del tamaño del “cubo”). El algoritmo genera oportunidades a la hora de extender su funcionamiento más allá de sus limitaciones iniciales y, respecto al rendimiento, presenta varias instancias que pueden ser aprovechadas para paralelizar el proceso.

1. La ubicación de cada tupla en la celda indicada podría ser realizada en forma paralela. Se debe tener en cuenta que la “ubicación” de una tupla en una

celda podría estar actualizando una zona de memoria compartida entre procesos y debe sincronizarse.

2. Teniendo en cuenta que la mayoría de los datos de un banco de datos no será un outlier, todas las tuplas alojadas en memoria que correspondan a una celda densamente poblada pueden ser liberadas para mantener el rápido acceso a todos aquellos candidatos a outlier.
3. El cálculo final del proceso, que es el cálculo de  $k$ -vecindarios más cercanos para cada observación que no haya podido ser descartada como outlier, puede ser realizado en paralelo.

Los desafíos al paralelizar son los mismos que en cualquier disciplina:

1. Administración correcta de la memoria disponible
2. Sincronización de los procesos que tengan memoria compartida
3. Paralelizar mientras tenga sentido. No se debe pasar más tiempo administrando los distintos procesos que llevando a cabo el algoritmo que interesa

Mientras que en este caso se incorpora un desafío obvio pero muy particular:

4. Se debe mantener la producción de resultados del algoritmo original

A continuación se presenta el desarrollo completo de la aproximación propuesta en [Knorr y otros, 1998] para continuar con la implementación del algoritmo, incorporación de mejoras y adaptación para la ejecución en paralelo del algoritmo basado en celdas.

## 8.1. DETECCIÓN DE OUTLIERS BASADA EN CELDAS

En el año 1998 Edwing Knorr y Raymond NG publicaron el trabajo “*Algorithms for Mining Distance-Based Outliers in Large Datasets*”. En este trabajo se presentan dos algoritmos que implementan la aproximación basada en celdas para la detección de outliers. El primer algoritmo es una simplificación del problema planteado en el trabajo porque presuponen que todos los datos a analizar caben en memoria, el segundo algoritmo ya plantea la premisa inicial del trabajo donde el conjunto de

datos a analizar es un muy grande por lo cual, para el análisis, se debe paginar los datos residentes en disco.

A pesar de que el segundo algoritmo parece ajustarse más a la temática de este trabajo final, se opta por implementar y extender la simplificación del algoritmo ya que, de esta manera, la implementación será más permeable a las mejoras que se desean introducir.

Se comienza por la versión simplificada del algoritmo, esta versión asume que la estructura multidimensional de celdas y todo el conjunto de datos cabe en la memoria.

Para facilitar la presentación se comienza por objetos de dos dimensiones y luego se procede a  $kD$ .

### 8.1.1. ESTRUCTURA DE CELDAS Y PROPIEDADES EN DOS DIMENSIONES

Para comenzar, se supone que los objetos de nuestro conjunto de datos tienen dos dimensiones.

Para utilizar la estrategia basada en celdas para encontrar todos los outliers  $DB(p, D)$  en  $2D$ , se ubica cada uno de los objetos dentro de un espacio  $2D$  que ha sido particionado en celdas o cuadrados de longitud  $l = \frac{D}{2\sqrt{2}}$ . Se denota  $C_{x,y}$  a la celda que está en la intersección de la fila  $x$  y la columna  $y$ . La capa 1 ( $L_1$ —por la palabra layer en inglés—) que se avecina con  $C_{x,y}$  se define como:

$$L_1(C_{x,y}) = \{C_{u,v} | u = x \pm 1, v = y \pm 1, C_{u,v} \neq C_{x,y}\} \quad (1)$$

Una celda típica (excepto por aquellas en los límites de la estructura de celdas) tiene ocho vecinos  $L_1$  [Knorr y otros, 1988].

**Propiedad 1:** Cualquier par de objetos dentro de la misma celda se encuentra como máximo a una distancia de  $\frac{D}{2}$

**Propiedad 2:** Si  $C_{u,v}$  es un vecino  $L_1$  de  $C_{x,y}$ , entonces cualquier objeto  $P \in C_{u,v}$  se encuentra como máximo a una distancia  $D$



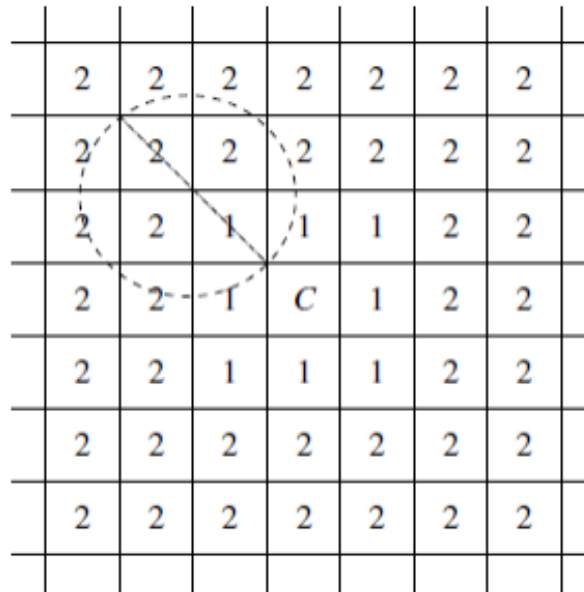


Figura 17 – La celda C y sus celdas vecinas de nivel 1 y nivel 2

La demostración de las propiedades pueden ser observadas en “*Algorithms for Mining Distance-Based Outliers in Large Datasets*” [Knorr y otros, 1988].

La capa  $L_2$  vecinos de  $C_{x,y}$  son aquellas celdas adicionales dentro de 3 celdas de  $C_{x,y}$

$$L_2(C_{x,y}) = \{C_{u,v} | u = x \pm 3, v = y \pm 3, C_{u,v} \notin L_1(C_{x,y}), C_{u,v} \neq C_{x,y}\} \quad (2)$$

Una celda típica (excepto aquellas ubicadas en los límites o cercanos a ellos) tiene  $7^2 - 3^2 = 40$  celdas  $L_2$  como puede visualizarse en la figura 17. Nótese que la capa 1 tiene una celda de espesor mientras que la capa 2 tiene dos celdas de espesor.  $L_2$  fue concebida de esta forma para satisfacer las siguientes propiedades

**Propiedad 3:** Si  $C_{u,v} \neq C_{x,y}$  no es ni un vecino  $L_1$  o  $L_2$  de  $C_{x,y}$ , entonces cualquier objeto  $P \in C_{u,v}$  y cualquier objeto  $Q \in C_{x,y}$  debe estar separados distancia mayor a  $D$ .

Como se puede observar en la figura 17, el espesor combinado de  $L_1$  más  $L_2$  es de tres celdas y la distancia entre  $P$  y  $Q$  debe exceder  $3l = \frac{3D}{2\sqrt{2}} > D$

**Propiedad 4:**

- a) Si hay más de  $M$  objetos en  $C_{x,y}$ , ninguno de los objetos en  $C_{x,y}$  es un outlier.
- b) Si hay más de  $M$  objetos en  $C_{x,y} \cup L_1(C_{x,y})$ , ninguno de los objetos en  $C_{x,y}$  es un outlier
- c) Si hay menos o exactamente  $M$  objetos en  $C_{x,y} \cup L_1(C_{x,y}) \cup L_2(C_{x,y})$ , cada objeto en  $C_{x,y}$  es un outlier

Las propiedades 4a y 4b son consecuencias directas de las propiedades 1 y 2, y la 4c es debido a la propiedad 3 [Knorr y otros, 1988]

**8.1.2. ALGORITMO PARA CONJUNTO DE DATOS RESIDENTES EN MEMORIA**

A continuación se presenta el algoritmo **FindAllOutsM** para detectar todos los outliers  $DB(p, D)$  en conjuntos de datos residentes en memoria. El algoritmo es una adaptación de aquel presentado en [Knorr y otros, 1988] dónde se incorpora una sintaxis orientada a objetos.

```

1  {
2      datos: dataSet - Conjunto de datos a analizar
3      celdas: ColeccionCeldas - Coleccion de celdas
4      umbral: integer - Valor frontera para outliers
5      distancia: integer - Valor frontera de distancia para outliers
6  }
7  procedure FindAllOutsM( datos: dataSet;
8      |                   celdas: ColeccionCeldas;
9      |                   umbral, distancia: integer);
10
11  var
12      i, cantObjetos, cantObjetos2, cantObjetosP: integer;
13      obj, objQ: Objeto;
14      celBlanca, celRoja, celAux1, celAux2: unaCelda;
15
16  begin
17      for i := 1 to celdas.Count do //Contadores en cero y color blanco
18          celdas.obtenerCelda(i).InicializarContador;
19
20      for obj in datos do //Guarda obj en la Celda correspondiente e incrementa
21          celdas.MapearACeldaApropiada(obj);
22
23      for celAux1 in celdas do
24          if celAux1.Count > umbral then celAux1.Color := Rojo;
25
26      for celRoja in celdas.Rojas do
27          for celAux1 in celRoja.vecindarioL1 do
28              if celAux1.Color <> Rojo then celAux1.Color := Rosado;
29
30  end
    
```

} Paso 1

} Paso 2

} Paso 3

} Paso 4



El algoritmo **FindAllOutsM** tiene cuatro parámetros:

- *datos*: es el conjunto de datos a analizar
- *celdas*: es una clase administradora de las celdas como las que se observan en la figura 17. Esta clase conoce todas las propiedades de una celda y las administra
- *umbral*: valor entero que representa el valor de  $M$  (mínima cantidad de puntos en el  $k$ -vecindario más cercano) utilizado para definir si un punto es outlier o no
- *distancia*: valor entero utilizado para definir el  $k$ -vecindario más cercano de un punto (El valor puede ser real en una implementación real)

El código del paso 1 recorre todas las celdas determinadas como necesarias por el algoritmo y las inicializa poniendo el contador de elementos en cero y “pintando” la celda de blanco.

El paso 2 recorrer todo el banco de datos a procesar ubicando cada dato en la celda correspondiente.

El paso 3 colorea de rojo todas aquellas celdas densamente pobladas, es decir, que hubieran superado el umbral de objetos implementando la propiedad  $4_a$ . Luego, en el paso 4, se recorren todas las celdas rojas, marcando cada celda perteneciente a su vecindario  $L_1$  (que no hubiese sido marcada como roja previamente) de color rosado. Estas celdas no pueden contener outliers por la propiedad  $4_b$ . En el paso 5a, otras celdas que satisfacen la propiedad  $4_b$  son detectadas y pintadas.

Finalmente el resto del algoritmo es utilizado para identificar aquellas celdas que cumplan con la propiedad  $4_c$ .

Todas las propiedades mencionadas en esta sección son utilizadas para determinar si un dato es o no un outlier en base a un análisis celda por celda en vez de un análisis objeto a objeto. Esto ayuda a reducir el tiempo de ejecución de forma significativa porque rápidamente encontramos un gran número de objetos que no pueden ser outliers. Para todas aquellas celdas que no satisfagan ninguna de las propiedades  $4_x$  se debe realizar un procesamiento objeto a objeto. Estos objetos están alojados en celdas denominadas blancas. Para cada elemento en una celda blanca se suman todos los elementos de su vecindario  $L_1$  y se calcula la distancia con todos los elementos de su vecindario  $L_2$  con el fin de determinar la cantidad de objetos que están a  $D$  distancia del punto analizado. Todo objeto que no supere el umbral de elementos  $M$  será declarado como un outlier.

### 8.1.3. ANÁLISIS DE COMPLEJIDAD PARA DOS DIMENSIONES

Se analiza la complejidad del algoritmo **FindAllOutsM** para dos dimensiones. El paso 1 (líneas 15 y 16) insume  $O(m)$  de tiempo. Siendo  $m$  el número total de celdas que, cumple con la característica de  $m \ll N$ . Los pasos 2 (líneas 18 y 19) y 3 (líneas 21 y 22) insumen  $O(N)$  y  $O(m)$  respectivamente. Como  $M + 1$  es el mínimo número de objetos que pueden alojarse en una celda roja hay, a lo sumo,  $\frac{N}{M+1}$  celdas rojas, por lo tanto, el paso 4 (líneas 24 a 26) insume  $O(\frac{N}{M+1})$ .

La complejidad del paso 5 (líneas 28 en adelante) es la más complicada de calcular:

- i. En el peor escenario no hay celdas que hayan podido ser etiquetadas como rojas o rosadas en los pasos anteriores
- ii. El paso entre las líneas 38 a 55 (paso 5<sub>b</sub>) puede involucrar todas las celdas. Si ninguna celda fue coloreada, entonces cada celda tiene a lo sumo  $M$  objetos asociados

Cada uno de los objetos en una celda puede requerir el control contra  $M$  objetos en cada una de las 40 celdas en su vecindario  $L_2$ . Esto significa que el tiempo requerido para la tarea será de  $O(40M^2)$ , por lo tanto, todo el paso 5 (líneas 28 en adelante) insume  $O(mM^2)$ . Como por definición  $M = (1 - p)N$ , podemos equiparar la ecuación a  $O(mN^2(1 - p)^2)$ . En la práctica se puede esperar que el valor de  $p$  sea extremadamente cercano a 1, especialmente para grandes conjuntos de datos, por lo tanto  $O(mN^2(1 - p)^2)$  puede ser aproximado a  $O(m)$ . Finalmente la complejidad asociada al algoritmo **FindAllOutsM** en dos dimensiones es  $O(m + N)$  [Knorr y otros, 1998].

Tener en cuenta que la complejidad calculada es muy conservadora ya que, en la práctica, se espera que haya muchas celdas rojas y rosadas. Cuando esto sucede, el paso 5 donde se recorren todas las celdas y sus vecinos  $L_2$  se vuelve menos dominante y el algoritmo requiere menos tiempo.

#### 8.1.4. GENERALIZACIÓN A DIMENSIONES MAYORES

Cuando se analizan dimensiones mayores a  $k = 2$ , el algoritmo **FindAllOutsM** requiere de un solo cambio para incorporar una estructura de celdas que generalice una dimensión  $k$   $D$ . El cambio tiene que ver con la longitud de la celda  $l$ . Cuando se procesan dos dimensiones se utiliza  $l = \frac{D}{2\sqrt{2}}$ , como la longitud de la diagonal de un hipercubo  $kD$  de longitud  $l$  es  $\sqrt{k}$ , la longitud  $l$  para  $k$  dimensiones debe ser modificada a  $l = \frac{D}{2\sqrt{k}}$  para preservar las propiedades 1 y 2 [Knorr y otros, 1998].

El vecindario  $L_1$  de una celda  $C_{x_1, \dots, x_k}$  se define como:

$$L_1(C_{x_1, \dots, x_k}) = \{C_{u_1, \dots, u_k} \mid u_i = x_i \pm 1, \forall 1 \leq i \leq k, C_{u_1, \dots, u_k} \neq C_{x_1, \dots, x_k}\} \quad (3)$$

Lo cual generaliza la definición dada como (1) en la sección 8.1.1. Sin embargo, para preservar la propiedad 3, la definición de los vecinos  $L_2$  debe ser modificada. Específicamente, como  $l = \frac{D}{2\sqrt{k}}$ , la capa  $L_2$  necesita ser más fina de lo que era para  $k = 2$ . Si se denomina  $x$  al espesor de la capa 2, entonces el espesor combinado de la capa 1 y 2 es  $x + 1$ . Por lo tanto, para que la propiedad 3 se mantenga, se requiere que  $(x + 1)l > D$ ; consecuentemente, se selecciona a  $x$  como  $[2\sqrt{k} - 1]$ . El vecinario  $L_2$  de  $C_{x_1, \dots, x_k}$  se define como:

$$L_2(C_{x_1, \dots, x_k}) = \left\{ C_{u_1, \dots, u_k} \left| \begin{array}{l} u_i = x_i \pm [2\sqrt{k}], \forall 1 \leq i \leq k, \\ C_{u_1, \dots, u_k} \notin L_1(C_{x_1, \dots, x_k}), \\ C_{u_1, \dots, u_k} \neq C_{x_1, \dots, x_k} \end{array} \right. \right\} \quad (4)$$

Lo que generaliza la definición (2) de la sección 8.1.1. De esta forma, las propiedades 1 a 4, presentadas en la sección 8.1.1 se preservan [Knorr y otros, 1998].

### 8.1.5. ANÁLISIS DE COMPLEJIDAD PARA DIMENSIONES MAYORES A 2

Para  $k > 2$  la complejidad de los pasos 1 a 4 en el algoritmo **FindAllOutsM** se mantiene sin modificaciones. Sin embargo,  $m$  es exponencial respecto a  $k$  y no necesariamente será mucho menor a  $N$ . También, la complejidad del paso 5 ya no es más  $O(m)$ , sino  $O(m(2[2\sqrt{k}] + 1)^k) \approx O(c^k)$  donde  $c$  es alguna constante dependiente de  $\sqrt{k}$  y  $m^{1/k}$  (que corresponde aproximadamente al número de celdas a lo largo de cada dimensión). Por lo tanto, la complejidad de todo el algoritmo es  $O(c^k + N)$  [Knorr y otros, 1998].

## 9. DESARROLLO DEL TRABAJO

A continuación se presentan distintas versiones del algoritmo con la evidencia concreta de su funcionamiento y limitaciones.

El algoritmo **FindAllOutsM** es del tipo basados en distancia. Se asumen las restricciones del método previamente documentadas [BREUNIG y otros, 2000] y se considera eficaz el funcionamiento del algoritmo si puede reproducir los resultados obtenidos mediante el análisis por fuerza bruta de las distancias entre las filas del banco de dato.

Como el algoritmo basado en celdas es del tipo basado en distancia, se utilizó la definición vista en la sección 7.3.1.7 para establecer si un objeto es un outlier o no:

*“Un objeto  $O$  en un grupo de datos  $T$  es un outlier  $DB(p, D)$  si al menos una fracción  $p$  de los objetos en  $T$  se encuentran a una distancia mayor que la distancia  $D$  respecto de  $O$ ” [Knorr y otros, 1999].*

Por ejemplo, si el banco de datos en el que se buscan outliers contienen 1000 objetos, con  $p = 0.975$  y  $D = 10$ , se calcula el umbral de outliers, esto se realiza mediante la siguiente fórmula:  $M = N(1 - p) = 1000(1 - 0,975) = 1000 * 0,025 = 25$ , luego, todo objeto del banco de datos es un outlier si la cantidad de objetos a una distancia  $D$  del objeto a analizar es menor o igual a 25.

Respecto a la eficiencia del algoritmo, en la sección 8.1.5 se analizó la complejidad del algoritmo para dimensiones mayores a 2 y como se estableció en la sección 6.4 sólo es tomada en cuenta la complejidad asociada al tiempo (uso de CPU).

La complejidad analizada en la sección 8.1.5 es muy conservadora ya que corresponde al peor escenario donde puede desenvolverse el algoritmo pero se espera un mejor rendimiento a medida que el algoritmo sea capaz de generar celdas rojas y rosadas. Se tiene en cuenta la complejidad temporal (cuánto tiempo demora el programa en brindar resultados) y la complejidad espacial (cuánta memoria operativa –RAM– es requerida por el programa). Se considera más eficiente un programa cuando más rápido sea capaz de arrojar resultados correctos. Si dos versiones del mismo algoritmo arrojan resultados en tiempos parecidos (+/- 5%), es

considerado más eficiente aquel algoritmo que menos memoria RAM requiera para brindar los resultados del proceso.

Para poder medir el rendimiento del algoritmo se utilizó un único banco de datos para todas versiones desarrolladas. Sólo se modificaron los parámetros de  $p$  y  $D$  para obtener resultados relevantes.

El objetivo principal del desarrollo del trabajo es brindar una implementación del algoritmo que sea capaz de detectar outliers superando las restricciones que se le habían impuesto al uso del algoritmo en el trabajo original. El algoritmo debe ser capaz de brindar resultados basándose en la cantidad de atributos a procesar, el valor de  $p$  y el valor de  $D$  independientemente de la cantidad de filas que conformen el banco de datos. Qué atributos se utilicen para determinar los outliers no será relevante desde el punto de vista de la ejecución del algoritmo siempre y cuando cumpla con la restricción de ser numérico. La implementación del algoritmo es absolutamente independiente del banco de datos o su dominio.

Todas las mejoras son realizadas sobre la implementación del algoritmo, es decir, se crea un programa (en este trabajo final utilizando Object Pascal como lenguaje de programación) que respeta cada una de las etapas presentadas en la sección 8.1.2 y la innovación se logra mejorando cómo se realiza en el programa cada etapa del algoritmo preservando su esencia.

## 9.1. MATERIALES Y MÉTODOS

En la primera etapa del desarrollo del trabajo se implementó el algoritmo **FindAllOutsM** utilizando Embarcadero RAD Studio XE. El lenguaje utilizado para escribir los programas fue Object Pascal.

Una vez implementado el algoritmo descrito en el trabajo original se procedió a corroborar el correcto funcionamiento del mismo con un banco de datos que permite la comprobación manual. El siguiente experimento estuvo destinado reproducir los experimentos y a explorar los límites del algoritmo documentados en el trabajo original para, una vez reproducido el límite, proponer modificaciones a la implementación que permitieran superar los límites encontrados.



El objetivo del trabajo es documentar el comportamiento de la implementación del algoritmo y completar la tabla 2 con la cantidad de segundos requerida desde el comienzo de la ejecución del algoritmo hasta la detección de todos los outliers.

|                     | 3D | 4D | 5D | 6D |
|---------------------|----|----|----|----|
| Cantidad de Objetos |    |    |    |    |
| 100.000             |    |    |    |    |
| 500.000             |    |    |    |    |
| 1.000.000           |    |    |    |    |
| 2.500.000           |    |    |    |    |
| 5.000.000           |    |    |    |    |
| 10.000.000          |    |    |    |    |

Tabla 2 – Template de los experimentos a realizar

Al completar los datos que exige el llenado de la tabla 2 se podrá observar el crecimiento de complejidad que tiene el algoritmo para dimensionalidades (desde 3 hasta 6 atributos) y volumen de datos (desde 100.000 y 10.000.000 de objetos) crecientes.

Si el algoritmo está bien implementado tiene una eficacia del 100%. Dados los parámetros de  $p$  y  $D$ , **FindAllOutsM** detecta todos aquellos objetos con al menos el  $p$  porcentaje de objetos del banco de datos a una distancia mayor a  $D$  del objeto analizado. Para comprobar el correcto funcionamiento del algoritmo se toma como referencia los outliers encontrados por un algoritmo que detecte outliers por fuerza bruta.

A medida que se encontraron limitaciones para la ejecución del algoritmo se realizaron propuestas para superar dichas limitaciones. La implementación de las propuestas fueron puestas a prueba inmediatamente.

Para reproducir los resultados del trabajo original se utilizó el mismo archivo NHL94 de estadísticas de la liga nacional de hockey de Canadá del año 1994 [Knorr y otros, 1998]. El archivo tiene el siguiente formato:

```
%team P NO. PLAYER(* is rookie)GP G A PTS +/- PIM PP SH GW GT S PCTG
% 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
LA C 99 WAYNE GRETZKY 81 38 92 130 -25 20 14 4 0 1 233 16.3
DET C 91 SERGEI FEDOROV 82 56 64 120 48 34 13 4 10 0 337 16.6
BOS C 12 ADAM OATES 77 32 80 112 10 45 16 2 3 0 197 16.2
```

Los datos que provee pueden ser visualizados en la tabla 3

| Atributo | Significado  |
|----------|--|
| Team     | Equipo para el que jugaba  |
| P        | Posición (G:Arquero, D: Defensa, C: Centro, R y L: wing derecho e izquierdo)                             |
| NO.      | Número de jugador  |
| PLAYER   | Nombre del jugador   |
| GP       | Partidos jugados   |
| G        | Goles  |
| A        | Asistencias  |
| PTS      | Puntos   |
| +/-      | Un jugador recibe un + si su equipo anota un gol estando el en el hielo, un - si su equipo recibe un gol |
| PIM      | Tiempo que el jugador permanece en la caja de penalidad sin poder participar del juego                   |
| PP       | Power play goals – Goles con un adversario en penalidad  |
| SH       | Short handed goals – Goles con un compañero en penalidad   |
| GW       | Game winning goals – Goles que pusieron a su equipo en ventaja   |
| GT       | Game tying goals – Goles que empataron el partido  |
| S        | Disparos al arco   |
| PCTG     | Save percentage – Porcentaje de tiros salvados   |

Tabla 3 – Atributos del archivo NHL94

La utilización de NHL94 responde a la necesidad de reproducir los resultados del trabajo original y, al mismo tiempo, proveer un banco de datos que permita comprobar los resultados de los distintos experimentos.

Los bancos de datos utilizados se dividen en distintos lotes, 871 tuplas del archivo original, 100.000, 500.000, 2.500.000, 5.000.000 y 10.000.000.

El banco de datos con 871 tuplas (el archivo original completo) sólo será utilizado para la comprobación del correcto funcionamiento del algoritmo tomando como

referencia el trabajo original. Este banco de datos no está documentado en la planilla final porque todos los experimentos realizados (para distinta cantidad de atributos) detectan los outliers antes de los 1000 milisegundos, obteniendo 0 segundos como resultado en todos los casos.

El motor de base de datos utilizado para el experimento es FirebirdSQL. Firebird es una base de datos relacional que ofrece muchas de las características del estándar ANSI SQL y puede ejecutarse en Linux, Windows y una variedad de plataformas Unix. Esta base de datos ha sido usada en sistemas de producción bajo distintos nombres desde 1981. Firebird no tiene costos de licenciamiento ni instalación y puede ser distribuida gratuitamente.

Algunas de las propuestas de mejoras a la implementación del programa que implementa el algoritmo **FindAllOutsM** requiere de la implementación de archivos indexados. Esta característica fue implementada mediante el uso de la base de datos SQLite versión 3. SQLite es una librería que implementa una base de datos SQL auto contenida, sin servidor, sin configuración y transaccional. El código de SQLite es de dominio público.

La base de datos Firebird fue instalada en una notebook Dell Studio 1555 (Intel Core 2 Duo P8600 2.4 GHz / ATI Mobility Radeon HD 4570) con 3 Gb de memoria, con 500Gb de disco y con Ubuntu 12.04 como sistema operativo.

La PC de prueba fue una notebook HP Intel® Core™ i7-4700MQ CPU@2.4GHz con 16GB de memoria, 1TB de disco y sistema operativo Windows 8.1 Pro.

## 9.2. IMPLEMETACION INICIAL: CELDAS Y DATOS EN MEMORIA

La primera implementación del algoritmo se escribió bajo la presunción que los datos pueden ubicarse en memoria y todos los cálculos pueden ser realizados sin necesidad de paginación ni otra técnica tendiente a administrar los recursos disponibles.

Para comprobar el funcionamiento del algoritmo se planteó el siguiente escenario:

El banco de datos se compone por los siguientes puntos en el plano: **{(1,1), (2,3), (3,5),(3,6), (4,4), (5.5, 4.5), (6.5, 3), (4,6), (6,6), (5.5, 8), (6.5,8), (7,6), (7,7), (7.5,5.5), (8,1),(9,3), (9.5, 5), (9.5, 7), (10,7), (9.5, 6)}**

Para identificar cada par ordenado es utilizada su ubicación dentro del banco de datos, siendo (1,1) el elemento 1 y (9.5,6) el elemento 20. Los parámetros del algoritmo son:

**$D = 2$**  (distancia máxima para buscar vecinos)

**$M = 1$**  (cantidad de objetos máxima en el vecindario más cercano para que un elemento sea considerado outlier)

Teniendo en cuenta el valor de  $M$  y la cantidad de elementos del banco de datos, se obtiene el valor de  $p = \left(1 - \left(\frac{1}{20}\right)\right) = (1 - 0.05) = 0.95$

La longitud de la celda es calculada por  $l = \frac{D}{2\sqrt{2}} = \frac{2}{2\sqrt{2}} = 0,7071$

Para calcular la dimensión del plano dónde se desarrolla el algoritmo se obtiene el mayor valor de la primera posición del par ordenado, en este caso 10, y el mayor valor de la segunda posición, en este caso 8. Se divide cada valor por  $l$  resultando así una matriz de [15,12]. Esta matriz contiene 180 celdas que fueron alojadas, junto con los datos que contenían, en memoria. Para saber en qué celda se ubicaba cada valor se realizó la división entera de su primer y segunda componente por  $l$ . Una vez realizada esta tarea para cada uno de los 20 valores se obtiene la siguiente distribución de los puntos que se observa en la figura 18

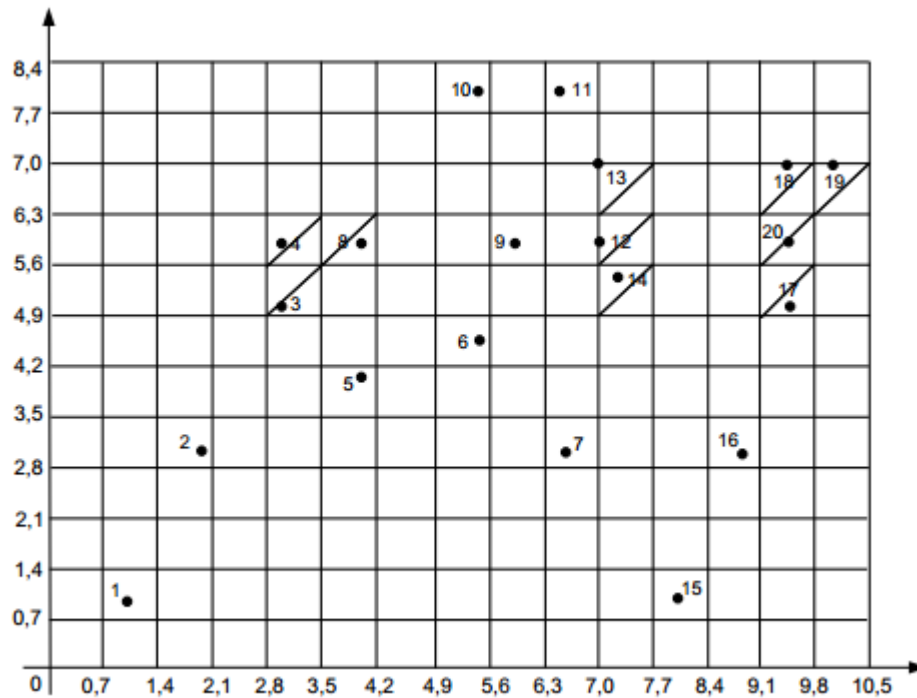


Figura 18 – Distribución del banco de datos en la matriz de celdas

Como se observa los puntos 3, 4, 8, 12, 13, 14, 17, 18, 19 y 20 fueron descartados como outliers por la propiedad 4<sub>b</sub> que indica que si hay más de  $M$  objetos en  $C_{x,y} \cup L_1(C_{x,y})$ , ninguno de los objetos en  $C_{x,y}$  es un outlier.

Con este banco de datos, el algoritmo no obtuvo celdas rojas, todas resultaron blancas por no superar el umbral ( $M = 1$ ) de objetos dentro de una misma celda, sin embargo analizando los puntos junto con su vecindario de nivel 1 ( $L_1$ ) se observan celdas donde sí se supera el umbral, siendo el punto 20 quien más vecinos tiene con 3 (17, 18 y 19). Todos los objetos de aquellas celdas donde la cantidad de objetos en  $C_{x,y} \cup L_1(C_{x,y})$  supera el umbral ( $M = 1$ ) son descartados como outliers.

No hay ninguna celda cuyo número de objetos asociados más sus vecindarios  $L_1$  y  $L_2$  sea menor 2. Para cada objeto contenido en cada celda blanca se debe calcular las distancias objeto a objeto con todos sus vecinos  $L_1$  y  $L_2$ .

Para el caso del objeto 1, su vecindario  $L_2$  está compuesto solamente por el objeto 2 y contra él se calcula la distancia:

$d = \sqrt{((2 - 1)^2 + (3 - 1)^2)} = \sqrt{(1^2 + 2^2)} = \sqrt{5} = 2,232$ , como el valor de la distancia es mayor a 2, el objeto 1 debe ser marcado como outlier.

En el caso del objeto 2, su vecindario  $L_2$  está compuesto por los objetos 3 y 5; contra ellos se calcula la distancia

$$d = \sqrt{((3 - 2)^2 + (5 - 3)^2)} = \sqrt{(1^2 + 2^2)} = \sqrt{5} = 2,232 \text{ y}$$

$d = \sqrt{((4 - 2)^2 + (4 - 3)^2)} = \sqrt{(2^2 + 1^2)} = \sqrt{5} = 2,232$ . Como ambas distancias son mayores a 2, el objeto 2 también debe ser marcado como outlier.

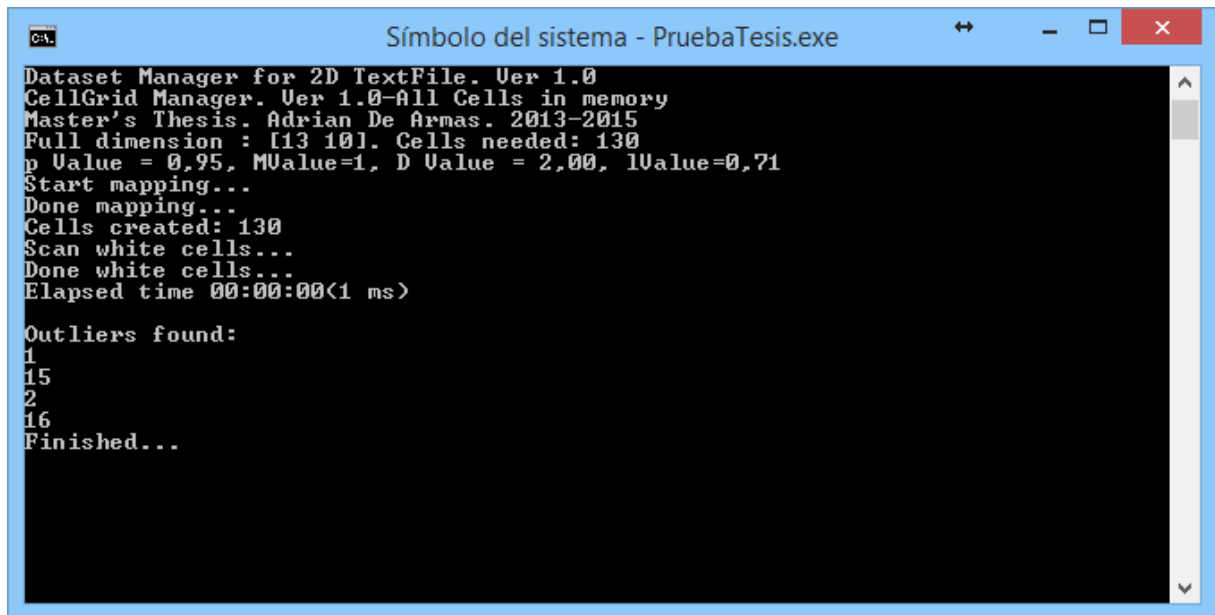
Para el caso del objeto 10, su vecindario  $L_2$  está compuesto por los objetos 4, 9, 11, 12 y 13. Visualmente puede observarse que el objeto 11 es quien más cerca se encuentra del objeto 10 y, por este motivo, se calcula la distancia primero respecto a este objeto ya que con un objeto que se encuentre a una distancia menor a 2, el objeto 10 no es un outlier.

$$d = \sqrt{((6.5 - 5.5)^2 + (8 - 8)^2)} = \sqrt{(1^2 + 0^2)} = \sqrt{1} = 1 < 2$$

Al encontrarse un objeto cuya distancia al objeto 10 es menor a 2, el objeto 10 no debe ser considerado un outlier.

Continuando con los cálculos se concluye que los objetos 1, 2, 15 y 16 son outliers.

La eficacia del algoritmo puede ser demostrada si detecta el 100% de los outliers recientemente calculados. Dicha ejecución puede observarse en la figura 19.



```

Símbolo del sistema - PruebaTesis.exe
Dataset Manager for 2D TextFile. Ver 1.0
CellGrid Manager. Ver 1.0-All Cells in memory
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [13 10]. Cells needed: 130
p Value = 0,95, MValue=1, D Value = 2,00, lValue=0,71
Start mapping...
Done mapping...
Cells created: 130
Scan white cells...
Done white cells...
Elapsed time 00:00:00<1 ms>

Outliers found:
1
15
2
16
Finished...
    
```

Figura 19 – FindAllOutsM para N=20 y D=2

Los puntos 1 [1,1]; 2 [2,3]; 15 [8,1] y 16 [9,3] son declarados como outliers coincidiendo con el cálculo manual resuelto anteriormente.

Respecto a las 180 celdas que debían alojarse en memoria, en la figura 19 puede observarse que el plano utilizado para la ejecución del algoritmo es de [13, 10], es sencillo de ver en la figura 18 que toda la primera fila y toda la primera columna tienen celdas vacías dando oportunidad a la primera optimización del programa. Si se tiene en cuenta no sólo los valores mayores de cada posición del par ordenado, sino que también se tiene en cuenta los valores menores, se crea una matriz que aloja sólo aquellas celdas con posibilidad de ser ocupadas por algún valor del banco de datos. Los valores mayores y menores se pueden obtener por conocimiento de dominio, o recorriendo el banco de datos.

Para calcular la dimensión del plano en donde se desarrolló el algoritmo se extrajo el mayor y menor valor de la primera posición del par ordenado, en este caso 10 y 1, y el mayor y menor valor de la segunda posición, en este caso 8 y 1, luego se realiza la división entera del valor de (mayor-menor) por  $l$ , resultando así en una matriz de [13,10] que alojarán un máximo de 130 celdas.

Para confirmar el correcto funcionamiento del algoritmo se analiza el archivo de log que genera el programa. El primer dato importante que se puede obtener es el estado de cada celda y qué objetos fueron mapeados dentro de ellas. A continuación se presenta el log completo para el escenario planteado:

```

Cell [0,0] = 1 Items White
1
Cell [9,0] = 1 Items White
15
Cell [1,2] = 1 Items White
2
Cell [7,2] = 1 Items White
7
Cell [11,2] = 1 Items White
16
Cell [4,4] = 1 Items White
5
Cell [6,4] = 1 Items White
6
Cell [2,5] = 1 Items White
3
Cell [12,5] = 1 Items White
17
Cell [9,6] = 1 Items White
14
Cell [2,7] = 1 Items White
4
Cell [4,7] = 1 Items White
8
Cell [7,7] = 1 Items White
9
Cell [8,7] = 1 Items White
12
Cell [12,7] = 1 Items Pink
20
Cell [8,8] = 1 Items White
13
Cell [12,8] = 2 Items Red
18
19
Cell [6,9] = 1 Items White
10
Cell [7,9] = 1 Items White
11
    
```

Sólo aquellas celdas con objetos asociados son visualizadas.

Para cada objeto contenido en un espacio ocupado por una celda blanca se calcula la distancia del objeto contra todos sus vecinos de nivel 2 ( $L_2$  – máximo nivel con capacidad de aportar algún ítem cuya distancia al objeto evaluado pueda ser menor



a  $D$ ). Para entender el funcionamiento de la implementación del algoritmo se visualizan algunos ejemplos extraídos del log que el programa genera. Tener en cuenta que las celdas son identificadas con coordenadas en base cero siendo  $[0,0]$  la primer celda de la matriz

```
Analizing Cell [0,0]
[0,1]
[1,0]
[1,1]
L1 Count 1
L2 Count 2
Distance to 2 2,2361
Number of objects withing 2,0000 distance: 1
```

El log lista todas las celdas que fueron tenidas en cuenta como vecinos  $L_2$

Para la celda  $[0,0]$  (ítem 1), la cantidad de elementos en su vecindario  $L_1$  es tan solo un elemento (el mismo) mientras que su vecindario  $L_2$  está compuesto por el punto 2 solamente, contra ese elemento es que se calculará la distancia. Como se puede observar la distancia al punto 2 es mayor a 2 y, como es el único punto a evaluar, el punto 1 es declarado como outlier.

```
Analizing Cell [6,4]
[5,3]
[5,4]
[5,5]
[6,3]
[6,5]
[7,3]
[7,4]
[7,5]
L1 Count 1
L2 Count 7
Distance to 5 1,5811
Number of objects withing 2,0000 distance: 2
```

Para la celda  $[6,4]$  (ítem 6), la cantidad de elementos en su vecindario  $L_1$  se conforma sólo por el elemento seis mientras que en su vecindario  $L_2$  tiene siete

elementos: 3, 4, 5, 8, 9, 12 y 14. Contra todos esos elementos se calcula la distancia dejando de calcular distancias una vez que podemos confirmar que las condiciones para descartar el punto como outlier están dadas.

Analizing Cell [9,6]

[8,5]

[8,6]

[8,7]

[9,5]

[9,7]

[10,5]

[10,6]

[10,7]

L1 Count 2

Para la celda [9,6] (ítem 14), la cantidad de elementos en su vecindario  $L_1$  es 2: los objetos de las celdas 14 y 12. La cantidad de elementos es suficiente para descartar al ítem 14 como outlier.

El log completo de mapeos y control de celdas blancas puede observarse en el anexo 11.1.

La implementación del algoritmo **FindAllOutsM** respeta todas las propiedades enunciadas en la sección 8.1.1. Esta implementación es eficaz ya que, respetando los valores de los parámetros  $p$  y  $D$  logra detectar todos outliers publicando evidencia de cómo se han obtenido.

Respecto a la eficiencia del programa creado, como se puede observar en la figura 19, la detección de los outliers se produjo prácticamente al instante consumiendo apenas 1 milisegundo entre el comienzo del algoritmo y la presentación de los resultados. Según el criterio adoptado en la sección 9, donde se tiene en cuenta el tiempo que consume de CPU el programa, la implementación resulta además de eficaz, eficiente.

### 9.3. ACCEDIENDO AL BANCO DE DATOS A TRAVÉS DE UNA BD

Una vez disponible la implementación del algoritmo **FindAllOutsM** se vinculó con una base de datos para la obtención de los objetos a analizar. Este paso es

indispensable para el alineamiento de las mejoras introducidas en la implementación del algoritmo y el objetivo principal de este trabajo final.

Para la comprobación del correcto funcionamiento del programa generado, se aplica el algoritmo sobre uno de los conjuntos de datos utilizados en el trabajo original [Knorr y otros, 1998] junto con los correspondientes valores de  $p$  y  $D$  documentados para contrastar los resultados obtenidos a lo largo de los experimentos.

El archivo NHL94 contiene estadísticas de jugadores de la liga nacional de hockey de Canadá del año 1994. El contenido completo del archivo se encuentra en la sección 14.2 en el apartado de anexos. Todos los datos del archivo fueron cargados en una tabla de base de datos, en este caso FIREBIRD SQL (base de datos libre) que actúa como origen de datos del experimento. Una vez cargado el archivo NHL94 en la base de datos se obtuvo una tabla con 871 filas.

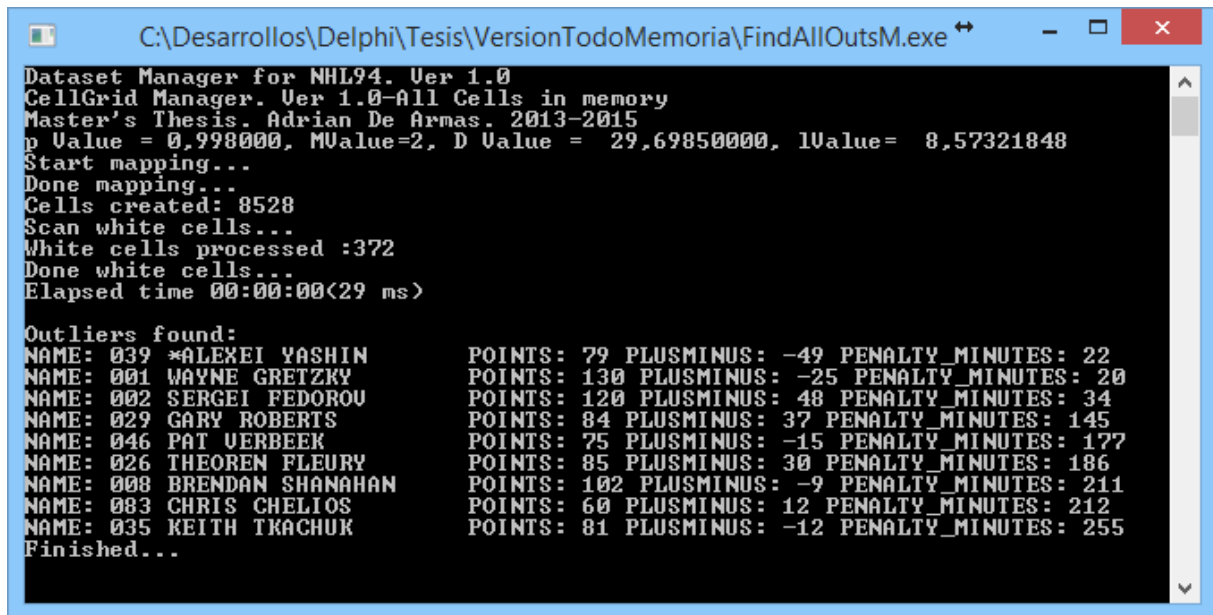
Para el primer experimento se buscó reproducir el experimento mostrado en el trabajo original cuyo resultado puede observar en la figura 21

```
FindOutliers nhl94.data p=0.998 D=29.6985 POINTS PLUSMINUS PENALTY_MINUTES
1) Name = WAYNE GRETZKY, POINTS = 130, PLUSMINUS = -25, PENALTY_MINUTES = 20
2) Name = SERGEI FEDOROV, POINTS = 120, PLUSMINUS = 48, PENALTY_MINUTES = 34
```

Figura 20 – **FindAllOutsM** original para  $N=871$  y  $D=3$  [Knorr y otros, 1998].

Como se puede observar en la figura 20, en el trabajo original se evaluaron tres atributos: POINTS, PLUSMINUS y PENALTY\_MINUTES con  $p = 0.998$  y  $D = 29.6985$ .

El resultado obtenido por la implementación del algoritmo puede observarse en la figura 21 y contrastarse con el resultado presentado en el trabajo original que se puede observar en la figura 20.



```

C:\Desarrollos\Delphi\Tesis\VersionTodoMemoria\FindAllOutsM.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 1.0-All Cells in memory
Master's Thesis. Adrian De Armas. 2013-2015
p Value = 0,998000, MValue=2, D Value = 29,69850000, lValue= 8,57321848
Start mapping...
Done mapping...
Cells created: 8528
Scan white cells...
White cells processed :372
Done white cells...
Elapsed time 00:00:00<29 ms>

Outliers found:
NAME: 039 *ALEXEI YASHIN          POINTS: 79 PLUSMINUS: -49 PENALTY_MINUTES: 22
NAME: 001 WAYNE GRETZKY          POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
NAME: 002 SERGEI FEDOROV        POINTS: 120 PLUSMINUS: 48 PENALTY_MINUTES: 34
NAME: 029 GARY ROBERTS          POINTS: 84 PLUSMINUS: 37 PENALTY_MINUTES: 145
NAME: 046 PAT UERBEEK            POINTS: 75 PLUSMINUS: -15 PENALTY_MINUTES: 177
NAME: 026 THEOREN FLEURY        POINTS: 85 PLUSMINUS: 30 PENALTY_MINUTES: 186
NAME: 008 BRENDAN SHANAHAN      POINTS: 102 PLUSMINUS: -9 PENALTY_MINUTES: 211
NAME: 083 CHRIS CHELIOS         POINTS: 60 PLUSMINUS: 12 PENALTY_MINUTES: 212
NAME: 035 KEITH KACHUK          POINTS: 81 PLUSMINUS: -12 PENALTY_MINUTES: 255
Finished...
    
```

Figura 21 – FindAllOutsM para N=871 y D=3

Si bien los jugadores WAYNE GRETZKY y SERGEI FEDOROV fueron detectados, el resultado de la ejecución del programa arroja muchos más outliers de los esperados ya que el resultado que presenta el trabajo original se pueden observar tan sólo dos jugadores. Ante esta situación donde la eficacia de la implementación del algoritmo se veía comprometida, se comprobó el resultado obtenido según el criterio tomado en la sección 9 que considera como eficaz el funcionamiento del algoritmo sí puede reproducir los resultados obtenidos mediante el análisis por fuerza bruta de las distancias entre las filas del banco de datos. Por lo tanto, se procedió a escribir un programa cuyo único fin es calcular los outliers por fuerza bruta, es decir, tomando cada una de las filas de la tabla y comparándola con el resto de las filas mediante el cálculo de la distancia entre ambas utilizando como parámetros el valor de  $p$  y  $D$  previamente determinado.

El resultado de la ejecución del programa para la detección de outliers por fuerza bruta puede observarse en la figura 22 y en la figura 23 el consumo de memoria de ambos programas.

```

C:\Desarrollos\Delphi\Tesis\VersionTodoMemoria\BruteForceScanNHL94.exe
Brute Force Scan for NHL94 dataset
Params D= 29,698500 p= 0,99800
Processing fields:
PTS
MM
PIM
Dataset count: 871
M value : 2
Outliers found:
NAME: WAYNE GRETZKY          POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
NAME: SERGEI FEDOROV        POINTS: 120 PLUSMINUS: 48 PENALTY_MINUTES: 34
NAME: BRENDAN SHANAHAN     POINTS: 102 PLUSMINUS: -9 PENALTY_MINUTES: 211
NAME: THEOREM FLEURY       POINTS: 85 PLUSMINUS: 30 PENALTY_MINUTES: 186
NAME: GARY ROBERTS         POINTS: 84 PLUSMINUS: 37 PENALTY_MINUTES: 145
NAME: KEITH TKACHUK        POINTS: 81 PLUSMINUS: -12 PENALTY_MINUTES: 255
NAME: *ALEXEI YASHIN        POINTS: 79 PLUSMINUS: -49 PENALTY_MINUTES: 22
NAME: PAT VERBEEK          POINTS: 75 PLUSMINUS: -15 PENALTY_MINUTES: 177
NAME: CHRIS CHELIOS        POINTS: 60 PLUSMINUS: 12 PENALTY_MINUTES: 212
Elapsed time 00:00:00<30 ms>
    
```

Figura 22 – Fuerza bruta para N=871 y D=3.

| Nombre                  | Estado | 1% CPU | 17% Memoria | 3% Disco | 0% Red |
|-------------------------|--------|--------|-------------|----------|--------|
| <b>Aplicaciones (3)</b> |        |        |             |          |        |
| Administrador de tareas |        | 0,2%   | 12,8 MB     | 0 MB/s   | 0 Mbps |
| BruteForceScanNHL94.exe |        | 0%     | 6,4 MB      | 0 MB/s   | 0 Mbps |
| FindAllOutsM.exe        |        | 0%     | 7,7 MB      | 0 MB/s   | 0 Mbps |

Figura 23 – Consumo de memoria para **FindAllOutsM** y Fuerza bruta con N=871 y D=3

Se ha comprobado el correcto funcionamiento de la implementación del algoritmo **FindAllOutsM** mediante la comparación de los resultados obtenidos contra la determinación de outliers calculados por fuerza bruta, es decir, calculando las distancias de cada elemento del banco de datos contra el resto de los elementos del banco de datos.

Es de destacar que para la cantidad de dimensiones y la cantidad de filas procesadas en rendimiento del algoritmo es prácticamente el mismo que aplicar fuerza bruta, siendo que este último es un algoritmo mucho más simple de

implementar e incluso el consumo de memoria es inferior debido a que no tiene que alojar la información de las celdas en memoria.

Hasta el momento se han procesado pocas dimensiones con una cantidad discreta de filas y se ha comprobado, como en el trabajo original, que el algoritmo funciona eficazmente pero la determinación por fuerza bruta, según los parámetros estipulados en la sección 9, es más eficiente ya que utilizando menos memoria (y menor complejidad de implementación) se logran los mismos resultados. El algoritmo **FindAllOutsM** no solo almacena en memoria las filas a procesar, sino que también mantiene un hiper cubo de celdas a las que les asocia las filas previamente leídas. Hasta el momento todas las ejecuciones de algoritmo han requerido tan solo una lectura del banco de datos externo.

Los logs generados por el programa para la comprobación de su funcionamiento pueden observarse en la sección de anexos bajo el número 14.3.

#### **9.4. FINDALLOUTSM AL LÍMITE: PROCESANDO 6 DIMENSIONES**

Para continuar con los experimentos con **FindAllOutsM** no resultaba relevante probar el algoritmo en situaciones donde ya se había determinado su correcto funcionamiento, por lo tanto, el próximo experimento es para determinar la existencia de outliers en un banco de datos de seis atributos, independientemente de la cantidad de filas a procesar. El inconveniente que presenta la cantidad de dimensiones es la estructura de memoria que se requiere para mantener todas las celdas. Los experimentos realizados muestran que el limitante de la cantidad de dimensiones a procesar ocurre más rápidamente que los inconvenientes que puede ocasionar un banco de datos con gran cantidad de filas.

Para realizar el experimento se utilizó el mismo banco de datos y esta vez se procesaron las columnas: POINTS, PLUSMINUS, PENALTY\_MINUTES, GAMES\_PLAYED, GOALS, ASSISTANCES. Para experimentar con el algoritmo no resulta relevante qué atributos se utilizaron sino la cantidad de atributos involucrados en el procesamiento, por lo tanto, la selección de atributos se realizó de forma aleatoria sin buscar otro resultado que no sea la correcta ejecución del algoritmo. No se modificaron los parámetros del algoritmo y se mantuvo el mismo  $p$  y  $D$  ya que el

algoritmo debe poder utilizarse, independientemente de la cantidad de outliers que vaya a detectar o no, con esos parámetros.

En la figura 23 puede observarse la pantalla de ejecución del experimento que analiza seis atributos y en la figura 24 puede observarse el error ocurre una vez que el programa intenta procesar las celdas blancas producto del mapeo de objetos

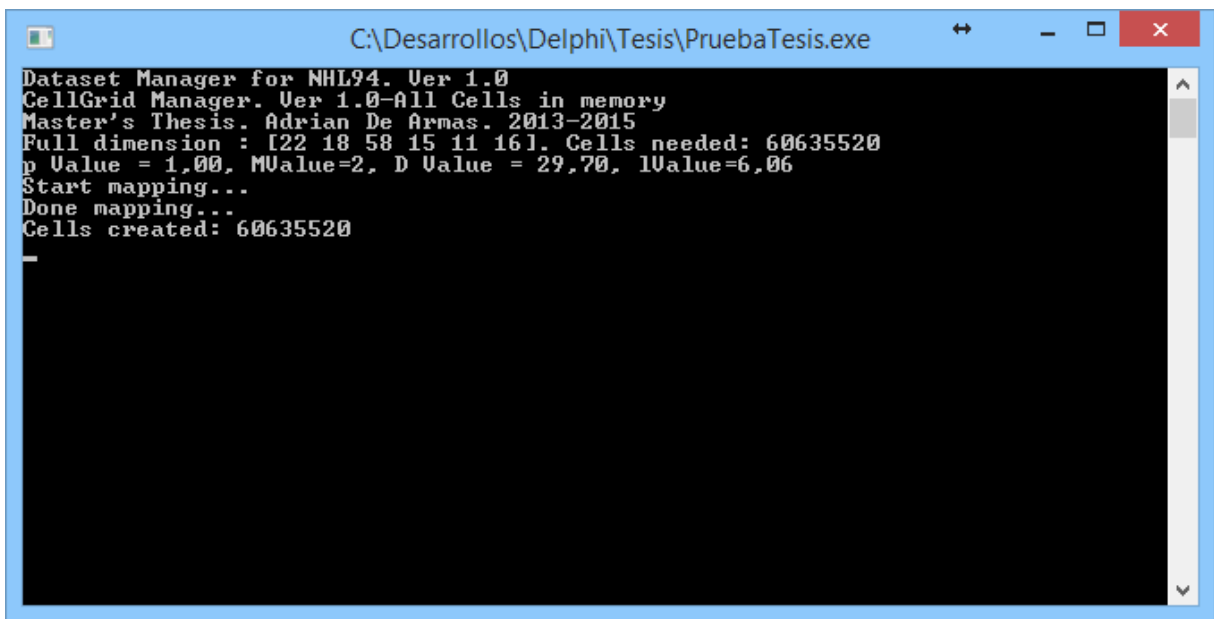


Figura 23 – FindAllOutsM para N=871 y D=6.

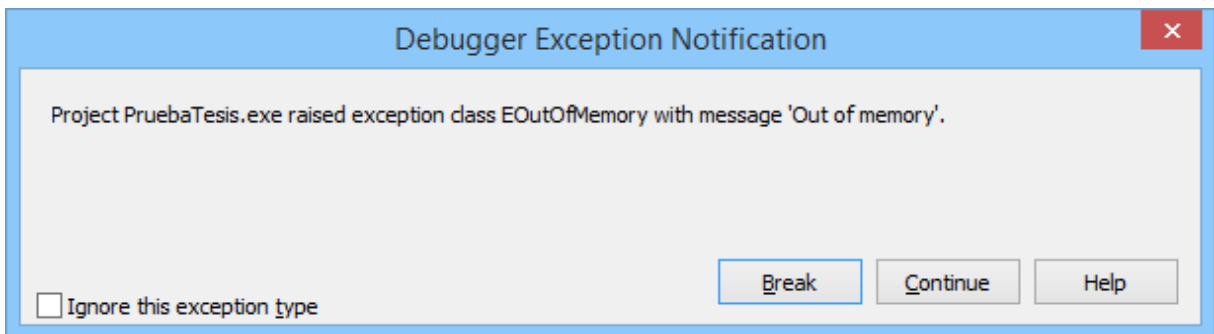


Figura 24 – Error por la ejecución de FindAllOutsM para D=6.

Si bien no se dispone del código fuente utilizado para crear la implementación del algoritmo **FindAllOutsM** en el trabajo original [Knorr y otros, 1998], se infiere que, al reproducir las limitaciones documentadas en el trabajo, ambas implementaciones (la del trabajo original y la de este trabajo final) se hicieron con decisiones de diseño parecidas.

Como se puede observar en la figura 23, la implementación del algoritmo requiere más de 60 millones de celdas para las 6 dimensiones elegidas y sus correspondientes valores. Es notable cómo crece la cantidad de celdas a utilizar, teniendo en cuenta que para 3 dimensiones se requirieron poco más de 8500 celdas.

El programa logra reservar la memoria necesaria para la dirección de todas las celdas aunque aún no se hayan creado ninguna de ellas. Cada celda se implementa como un objeto que sólo es instanciado al momento de requerirse su utilización. Esta sencilla forma de administrar los recursos no es suficiente para subsanar el obstáculo que se genera por la cantidad de celdas requeridas ya que al ir creando las primeras celdas y, manteniendo los objetos de jugadores asociados a ellas, rápidamente la aplicación se queda sin posibilidad de reservar más memoria. Esto sucede porque el programa ejecutado estaba compilado para 32 bits. Si tan sólo se compilase la misma versión del programa para 64 bits, la cantidad de memoria que se puede direccionar crece en forma exponencial.

A pesar de tener una solución a mano para extender el alcance de la aproximación por celdas para la búsqueda de outliers, esta solución está muy lejos de ser eficiente ya que tan sólo se extendería la capacidad del algoritmo para reservar más memoria y, según lo establecido en la sección 9, mientras menos memoria utilice el algoritmo, más eficiente será considerado. El uso de 32 bits fue utilizado para mejorar la visibilidad de los inconvenientes que acarrear grandes dimensiones de datos o bancos de datos con muchas tuplas ¿Acaso no llama la atención que para procesar un banco de datos de 871 filas haya que reservar memoria para 60 millones de celdas? En el peor escenario, cada uno de los 871 objetos ocuparía 1 celda (lo que obligaría a una determinación de outliers por fuerza bruta), ¿para qué reservar memoria para casi 60 millones de celdas?

Bajo esta premisa se desarrolló la segunda mejora a la implementación realizada al algoritmo basado en celdas. El algoritmo en sí, permanece sin modificaciones.

La mejora implica sólo crear las celdas necesarias (sin reservar memoria previamente), esto es, sólo aquellas que tengan objetos asociados o deban ser



marcadas con un color diferente al blanco (eliminando así la posibilidad de alojar outliers en esa celda).

Cada celda que se aloje en memoria se identifica por una dirección de memoria o una posición relativa dentro de un vector de celdas. Ninguna de las dos opciones resulta práctica a la hora de identificar inequívocamente a la celda teniendo en cuenta las características del algoritmo.

Tomando como modelo la ejecución realizada en el experimento anterior cuando se procesó 871 filas y tres atributos, se observa que la asignación de objetos a celdas produjo el siguiente mapeo:

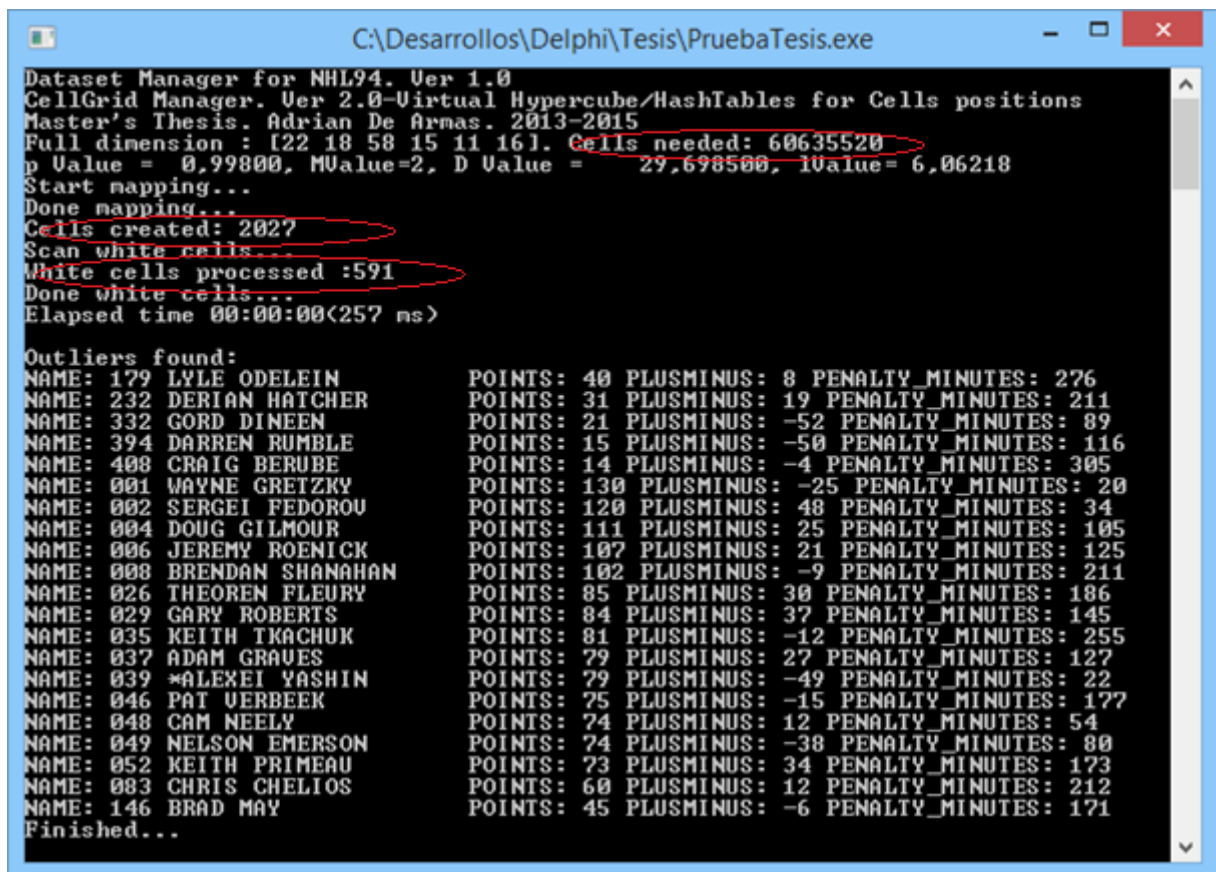
```
Cell [0,4,0] = 4 Items Red
521 BRAD LAUER
574 DEREK MAYER
720 *ANDREW MCKIM
829 RADEK HAMR
```

Claramente lo que permite respetar la definición original del algoritmo y su interpretación es identificar cada celda con su posición relativa dentro del hipercubo, en el caso del ejemplo [0,4,0]. Dado que el hipercubo no debe ser distribuido en memoria (por el inconveniente anteriormente visto) y la idea es sólo crear las celdas que sean absolutamente necesarias, se implementó una lista de doble entrada: por un lado una cadena que representa las coordenadas de la celda y por el otro la posición (índice) que ocupa la celda dentro de una lista de celdas (objetos). La idea es obtener de forma ágil la posición de una celda (identificada por sus coordenadas) dentro de la lista de celdas.

Para llevar a cabo la tarea de identificar la ubicación de una celda mediante sus coordenadas se utilizó un diccionario. Un diccionario es, en cierto modo, similar a un vector. En un vector se tiene una serie de valores indexados (generalmente) por un valor entero. Este índice tiene un valor de mínima y otro de máxima. El diccionario implementado, tiene por característica, estar definido como un “hashtable”. Los hashtables representan una colección de pares de datos definidos como clave y valor, que se organizan en base al código hash de la clave. Los hashtables están optimizados para realizar búsquedas veloces. Cuando un par clave-valor se agrega

a un hastable, el hash de la clave se calcula y se almacena junto con el par agregado.

La implementación del diccionario fue realizada usando como clave las coordenadas (respecto a ese hipercubo virtual) de la celda y el índice que representa su posición relativa dentro de la lista de celdas como valor.



```

C:\Desarrollos\Delphi\Tesis\PruebaTesis.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [22 18 58 15 11 16]. Cells needed: 60635520
p Value = 0,99800, MValue=2, D Value = 29,678500, lValue= 6,06218
Start mapping...
Done mapping...
Cells created: 2027
Scan white cells
White cells processed :591
Done white cells...
Elapsed time 00:00:00(257 ms)

Outliers found:
NAME: 179 LYLE ODELEIN          POINTS: 40 PLUSMINUS: 8 PENALTY_MINUTES: 276
NAME: 232 DERIAN HATCHER      POINTS: 31 PLUSMINUS: 19 PENALTY_MINUTES: 211
NAME: 332 GORD DINEEN        POINTS: 21 PLUSMINUS: -52 PENALTY_MINUTES: 89
NAME: 394 DARREN RUMBLE      POINTS: 15 PLUSMINUS: -50 PENALTY_MINUTES: 116
NAME: 408 CRAIG BERUBE       POINTS: 14 PLUSMINUS: -4 PENALTY_MINUTES: 305
NAME: 001 WAYNE GREITZKY     POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
NAME: 002 SERGEI FEDOROV    POINTS: 120 PLUSMINUS: 48 PENALTY_MINUTES: 34
NAME: 004 DOUG GILMOUR      POINTS: 111 PLUSMINUS: 25 PENALTY_MINUTES: 105
NAME: 006 JEREMY ROENICK    POINTS: 107 PLUSMINUS: 21 PENALTY_MINUTES: 125
NAME: 008 BRENDAN SHANAHAN  POINTS: 102 PLUSMINUS: -9 PENALTY_MINUTES: 211
NAME: 026 THEOREN FLEURY    POINTS: 85 PLUSMINUS: 30 PENALTY_MINUTES: 186
NAME: 029 GARY ROBERTS     POINTS: 84 PLUSMINUS: 37 PENALTY_MINUTES: 145
NAME: 035 KEITH TRACHUK     POINTS: 81 PLUSMINUS: -12 PENALTY_MINUTES: 255
NAME: 037 ADAM GRAVES       POINTS: 79 PLUSMINUS: 27 PENALTY_MINUTES: 127
NAME: 039 *ALEXEI YASHIN    POINTS: 79 PLUSMINUS: -49 PENALTY_MINUTES: 22
NAME: 046 PAT UERBEEK       POINTS: 75 PLUSMINUS: -15 PENALTY_MINUTES: 177
NAME: 048 CAM NEELY        POINTS: 74 PLUSMINUS: 12 PENALTY_MINUTES: 54
NAME: 049 NELSON EMERSON    POINTS: 74 PLUSMINUS: -38 PENALTY_MINUTES: 80
NAME: 052 KEITH PRIMEAU    POINTS: 73 PLUSMINUS: 34 PENALTY_MINUTES: 173
NAME: 083 CHRIS CHELIOS    POINTS: 60 PLUSMINUS: 12 PENALTY_MINUTES: 212
NAME: 146 BRAD MAY         POINTS: 45 PLUSMINUS: -6 PENALTY_MINUTES: 171
Finished...
    
```

Figura 25 – FinAllOutsM con hipercubo virtual para N=871 y D=6

Como se puede observar en la figura 25, que muestra la ejecución del programa mejorado, la cantidad de celdas que requiere el algoritmo pasó de 60 millones por definición a 2.027 que son las necesarias para alojar todos las filas de datos y sus vecinos  $L_1$  y  $L_2$  (para esta ejecución en particular).

Finalmente fueron tan sólo 591 celdas blancas a las que se le practicó el cálculo de distancias contra sus vecinos  $L_2$ .

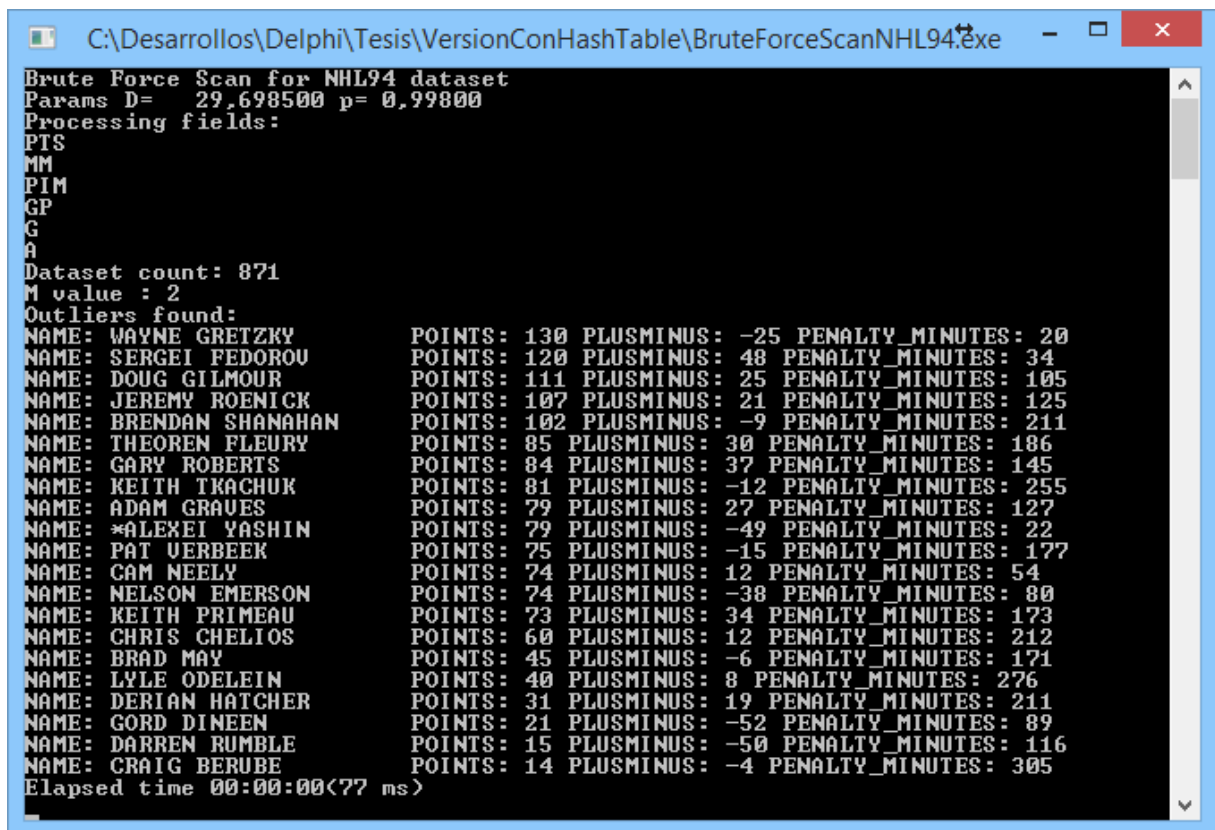
El programa detectó, para el banco de datos original NHL94 con parámetros  $p = 0.998$  y  $D = 29.6985$ , 21 outliers:

|                            |  |
|----------------------------|--|
| NAME: 001 WAYNE GRETZKY    | POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20 |
| NAME: 002 SERGEI FEDOROV   | POINTS: 120 PLUSMINUS: 48 PENALTY_MINUTES: 34  |
| NAME: 004 DOUG GILMOUR     | POINTS: 111 PLUSMINUS: 25 PENALTY_MINUTES: 105 |
| NAME: 006 JEREMY ROENICK   | POINTS: 107 PLUSMINUS: 21 PENALTY_MINUTES: 125 |
| NAME: 008 BRENDAN SHANAHAN | POINTS: 102 PLUSMINUS: -9 PENALTY_MINUTES: 211 |
| NAME: 026 THEOREN FLEURY   | POINTS: 85 PLUSMINUS: 30 PENALTY_MINUTES: 186  |
| NAME: 029 GARY ROBERTS     | POINTS: 84 PLUSMINUS: 37 PENALTY_MINUTES: 145  |
| NAME: 035 KEITH TKACHUK    | POINTS: 81 PLUSMINUS: -12 PENALTY_MINUTES: 255 |
| NAME: 037 ADAM GRAVES      | POINTS: 79 PLUSMINUS: 27 PENALTY_MINUTES: 127  |
| NAME: 039 *ALEXEI YASHIN   | POINTS: 79 PLUSMINUS: -49 PENALTY_MINUTES: 22  |
| NAME: 046 PAT VERBEEK      | POINTS: 75 PLUSMINUS: -15 PENALTY_MINUTES: 177 |
| NAME: 048 CAM NEELY        | POINTS: 74 PLUSMINUS: 12 PENALTY_MINUTES: 54   |
| NAME: 049 NELSON EMERSON   | POINTS: 74 PLUSMINUS: -38 PENALTY_MINUTES: 80  |
| NAME: 052 KEITH PRIMEAU    | POINTS: 73 PLUSMINUS: 34 PENALTY_MINUTES: 173  |
| NAME: 083 CHRIS CHELIOS    | POINTS: 60 PLUSMINUS: 12 PENALTY_MINUTES: 212  |
| NAME: 146 BRAD MAY         | POINTS: 45 PLUSMINUS: -6 PENALTY_MINUTES: 171  |
| NAME: 179 LYLE ODELEIN     | POINTS: 40 PLUSMINUS: 8 PENALTY_MINUTES: 276   |
| NAME: 232 DERIAN HATCHER   | POINTS: 31 PLUSMINUS: 19 PENALTY_MINUTES: 211  |
| NAME: 332 GORD DINEEN      | POINTS: 21 PLUSMINUS: -52 PENALTY_MINUTES: 89  |
| NAME: 394 DARREN RUMBLE    | POINTS: 15 PLUSMINUS: -50 PENALTY_MINUTES: 116 |
| NAME: 408 CRAIG BERUBE     | POINTS: 14 PLUSMINUS: -4 PENALTY_MINUTES: 305  |

Para confirmar que los outliers detectados son correctos se vuelve a comparar por fuerza bruta. Los outliers detectados por ambos programas son los mismos. El resultado obtenido por fuerza bruta puede observarse en la figura 27, mientras que el resultado obtenido por la implementación del algoritmo **FindAllOutsM** se observa en la figura 25. En la figura 26 puede observarse el consumo de memoria de cada programa.

| Nombre                  | Estado | CPU  | Memoria | Disco  | Red    |
|-------------------------|--------|------|---------|--------|--------|
| <b>Aplicaciones (5)</b> |        |      |         |        |        |
| Administrador de tareas |        | 0,4% | 12,3 MB | 0 MB/s | 0 Mbps |
| BruteForceScanNHL94.exe |        | 0%   | 6,4 MB  | 0 MB/s | 0 Mbps |
| FindAllOutsM.exe        |        | 0%   | 8,3 MB  | 0 MB/s | 0 Mbps |

Figura 26 – Consumo de memoria para **FindAllOutsM** y Fuerza bruta con  $N=871$  y  $D=6$



```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\BruteForceScanNHL94.exe
Brute Force Scan for NHL94 dataset
Params D= 29.698500 p= 0.99800
Processing fields:
PTS
MM
PIM
GP
G
A
Dataset count: 871
M value : 2
Outliers found:
NAME: WAYNE GRETZKY          POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
NAME: SERGEI FEDOROU        POINTS: 120 PLUSMINUS: 48 PENALTY_MINUTES: 34
NAME: DOUG GILMOUR          POINTS: 111 PLUSMINUS: 25 PENALTY_MINUTES: 105
NAME: JEREMY ROENICK        POINTS: 107 PLUSMINUS: 21 PENALTY_MINUTES: 125
NAME: BRENDAN SHANAHAN     POINTS: 102 PLUSMINUS: -9 PENALTY_MINUTES: 211
NAME: THEOREN FLEURY        POINTS: 85 PLUSMINUS: 30 PENALTY_MINUTES: 186
NAME: GARY ROBERTS          POINTS: 84 PLUSMINUS: 37 PENALTY_MINUTES: 145
NAME: KEITH TRACHUK         POINTS: 81 PLUSMINUS: -12 PENALTY_MINUTES: 255
NAME: ADAM GRAVES           POINTS: 79 PLUSMINUS: 27 PENALTY_MINUTES: 127
NAME: *ALEXEI YASHIN        POINTS: 79 PLUSMINUS: -49 PENALTY_MINUTES: 22
NAME: PAT UERBEEK           POINTS: 75 PLUSMINUS: -15 PENALTY_MINUTES: 177
NAME: CAM NEELY             POINTS: 74 PLUSMINUS: 12 PENALTY_MINUTES: 54
NAME: NELSON EMERSON        POINTS: 74 PLUSMINUS: -38 PENALTY_MINUTES: 80
NAME: KEITH PRIMEAU         POINTS: 73 PLUSMINUS: 34 PENALTY_MINUTES: 173
NAME: CHRIS CHELIOS         POINTS: 60 PLUSMINUS: 12 PENALTY_MINUTES: 212
NAME: BRAD MAY              POINTS: 45 PLUSMINUS: -6 PENALTY_MINUTES: 171
NAME: LYLE ODELEIN          POINTS: 40 PLUSMINUS: 8 PENALTY_MINUTES: 276
NAME: DERIAN HATCHER        POINTS: 31 PLUSMINUS: 19 PENALTY_MINUTES: 211
NAME: GORD DINEEN           POINTS: 21 PLUSMINUS: -52 PENALTY_MINUTES: 89
NAME: DARREN RUMBLE         POINTS: 15 PLUSMINUS: -50 PENALTY_MINUTES: 116
NAME: CRAIG BERUBE          POINTS: 14 PLUSMINUS: -4 PENALTY_MINUTES: 305
Elapsed time 00:00:00<?? ms>
    
```

Figura 27 – Ejecución por Fuerza bruta para N=871 y D=6.

Lo relevante de este experimento es que el algoritmo **FindAllOutsM** pudo ser ejecutado superando el limitante establecido en el trabajo original de un máximo de 5 dimensiones.

Ambos programas encontraron los mismos outliers arrojando evidencia suficiente de la eficacia de la implementación del algoritmo **FindAllOutsM**.

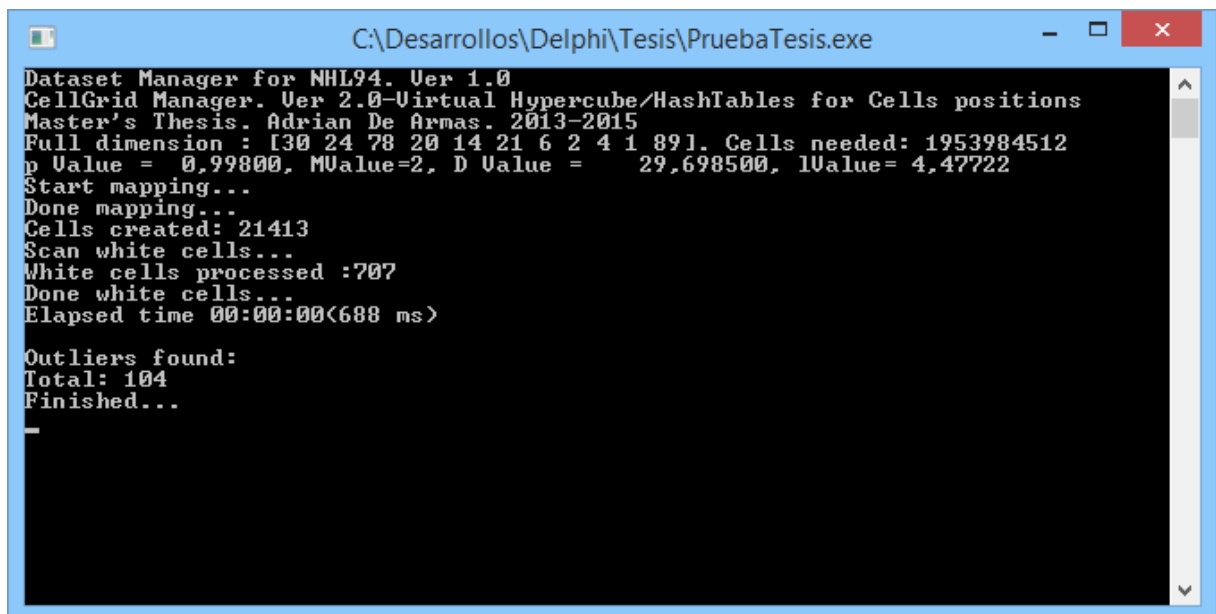
Respecto a la eficiencia del algoritmo, para esta cantidad de tuplas y dimensiones, la determinación de outliers por fuerza bruta es mucho más eficiente ya que no sólo consume menos memoria sino que resuelve la detección de outliers en  $\frac{1}{3}$  del tiempo que insumió el algoritmo **FindAllOutsM**. El resultado es el esperado ya que la complejidad del algoritmo por fuerza bruta es  $O(N^2)$ , por lo tanto, no es afectada por la cantidad de atributos a procesar mientras que la complejidad de **FindAllOutsM** sí se ve afectada por la dimensión y la cantidad creciente de celdas influye en un mayor consumo de RAM.

La pregunta que surge luego de este experimento es ¿Cuántas dimensiones puede procesar la implementación del algoritmo? En principio no hay una limitante real, la posibilidad de ejecutar el algoritmo dependerá de la cantidad de dimensiones y la cantidad de elementos a procesar en memoria.

El banco de datos NHL94 proporciona 12 columnas de datos que pueden ser utilizadas para la determinación de outliers, la última columna es un promedio, por lo tanto es una variable dependiente y no es tenida en cuenta. Al ejecutar el algoritmo con 11 propiedades a analizar se extienden los escenarios posibles dónde utilizar esta técnica ya que en el trabajo original se recomienda el uso de este algoritmo para 4 dimensiones o 5 (a lo sumo) [Knorr y otros, 1998].

En la figura 28 puede observarse el resultado de la ejecución del programa que procesa 11 atributos con parámetros  $p = 0.998$  y  $D = 29.6985$ . La comprobación del resultado obtenido fue realizada por fuerza bruta y puede observarse en la figura 29.

En ambos casos se eliminó la salida de los jugadores identificados como outliers porque, para los parámetros cargados en el programa, los outliers fueron muchos y no podían listarse en una sola pantalla.



```

C:\Desarrollos\Delphi\Tesis\PruebaTesis.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [30 24 78 20 14 21 6 2 4 1 89]. Cells needed: 1953984512
p Value = 0,99800, MValue=2, D Value = 29,698500, lValue= 4,47722
Start mapping...
Done mapping...
Cells created: 21413
Scan white cells...
White cells processed :707
Done white cells...
Elapsed time 00:00:00<688 ms>

Outliers found:
Total: 104
Finished...
    
```

Figura 28 – FindAllOutsM para N=871 y D=11.



Figura 29 – Fuerza bruta para N=871 y D=11.

Los dos programas detectaron 104 outliers, los jugadores detectados como anómalos coinciden y se listan a continuación.

|           |                  |            |                |                      |
|-----------|------------------|------------|----------------|----------------------|
| NAME: 179 | LYLE ODELEIN     | POINTS: 40 | PLUSMINUS: 8   | PENALTY_MINUTES: 276 |
| NAME: 191 | DALE HUNTER      | POINTS: 38 | PLUSMINUS: -4  | PENALTY_MINUTES: 131 |
| NAME: 194 | SCOTT PEARSON    | POINTS: 37 | PLUSMINUS: -4  | PENALTY_MINUTES: 165 |
| NAME: 195 | JOE SACCO        | POINTS: 37 | PLUSMINUS: -11 | PENALTY_MINUTES: 61  |
| NAME: 196 | MARIO LEMIEUX    | POINTS: 37 | PLUSMINUS: -2  | PENALTY_MINUTES: 32  |
| NAME: 201 | MARK LAMB        | POINTS: 36 | PLUSMINUS: -44 | PENALTY_MINUTES: 72  |
| NAME: 232 | DERIAN HATCHER   | POINTS: 31 | PLUSMINUS: 19  | PENALTY_MINUTES: 211 |
| NAME: 234 | MARTY MCSORLEY   | POINTS: 31 | PLUSMINUS: -12 | PENALTY_MINUTES: 194 |
| NAME: 240 | ROB PEARSON      | POINTS: 30 | PLUSMINUS: -6  | PENALTY_MINUTES: 189 |
| NAME: 255 | RONNIE STERN     | POINTS: 29 | PLUSMINUS: 6   | PENALTY_MINUTES: 243 |
| NAME: 258 | ULF SAMUELSSON   | POINTS: 29 | PLUSMINUS: 23  | PENALTY_MINUTES: 199 |
| NAME: 271 | RANDY MCKAY      | POINTS: 27 | PLUSMINUS: 24  | PENALTY_MINUTES: 244 |
| NAME: 307 | TROY MALLETT     | POINTS: 23 | PLUSMINUS: -33 | PENALTY_MINUTES: 166 |
| NAME: 320 | JEFF ODGERS      | POINTS: 21 | PLUSMINUS: -13 | PENALTY_MINUTES: 222 |
| NAME: 329 | DAVE MANSON      | POINTS: 21 | PLUSMINUS: -14 | PENALTY_MINUTES: 191 |
| NAME: 331 | KELLY BUCHBERGER | POINTS: 21 | PLUSMINUS: -20 | PENALTY_MINUTES: 199 |
| NAME: 332 | GORD DINEEN      | POINTS: 21 | PLUSMINUS: -52 | PENALTY_MINUTES: 89  |
| NAME: 340 | MIKE PELUSO      | POINTS: 20 | PLUSMINUS: 19  | PENALTY_MINUTES: 238 |
| NAME: 344 | ANDREW MCBAIN    | POINTS: 19 | PLUSMINUS: -41 | PENALTY_MINUTES: 64  |
| NAME: 346 | WARREN RYCHEL    | POINTS: 19 | PLUSMINUS: -19 | PENALTY_MINUTES: 322 |
| NAME: 349 | TIE DOMI         | POINTS: 19 | PLUSMINUS: -8  | PENALTY_MINUTES: 347 |
| NAME: 358 | TODD EWEN        | POINTS: 18 | PLUSMINUS: -7  | PENALTY_MINUTES: 272 |
| NAME: 370 | BOB PROBERT      | POINTS: 17 | PLUSMINUS: -1  | PENALTY_MINUTES: 275 |
| NAME: 394 | DARREN RUMBLE    | POINTS: 15 | PLUSMINUS: -50 | PENALTY_MINUTES: 116 |
| NAME: 408 | CRAIG BERUBE     | POINTS: 14 | PLUSMINUS: -4  | PENALTY_MINUTES: 305 |
| NAME: 424 | SHANE CHURLA     | POINTS: 13 | PLUSMINUS: -8  | PENALTY_MINUTES: 333 |
| NAME: 457 | SYLVAIN LEFEBVRE | POINTS: 11 | PLUSMINUS: 33  | PENALTY_MINUTES: 79  |
| NAME: 474 | KEN DANAYKO      | POINTS: 10 | PLUSMINUS: 27  | PENALTY_MINUTES: 176 |

|           |                  |             |                |                      |
|-----------|------------------|-------------|----------------|----------------------|
| NAME: 519 | ROB RAY          | POINTS: 7   | PLUSMINUS: 2   | PENALTY_MINUTES: 274 |
| NAME: 556 | MARC POTVIN      | POINTS: 5   | PLUSMINUS: -8  | PENALTY_MINUTES: 272 |
| NAME: 568 | MICK VUKOTA      | POINTS: 4   | PLUSMINUS: -5  | PENALTY_MINUTES: 237 |
| NAME: 721 | JAMIE HUSCROFT   | POINTS: 1   | PLUSMINUS: -2  | PENALTY_MINUTES: 144 |
| NAME: 001 | WAYNE GRETZKY    | POINTS: 130 | PLUSMINUS: -25 | PENALTY_MINUTES: 20  |
| NAME: 002 | SERGEI FEDOROV   | POINTS: 120 | PLUSMINUS: 48  | PENALTY_MINUTES: 34  |
| NAME: 003 | ADAM OATES       | POINTS: 112 | PLUSMINUS: 10  | PENALTY_MINUTES: 45  |
| NAME: 004 | DOUG GILMOUR     | POINTS: 111 | PLUSMINUS: 25  | PENALTY_MINUTES: 105 |
| NAME: 005 | PAVEL BURE       | POINTS: 107 | PLUSMINUS: 1   | PENALTY_MINUTES: 86  |
| NAME: 006 | JEREMY ROENICK   | POINTS: 107 | PLUSMINUS: 21  | PENALTY_MINUTES: 125 |
| NAME: 007 | MARK RECCHI      | POINTS: 107 | PLUSMINUS: -2  | PENALTY_MINUTES: 46  |
| NAME: 008 | BRENDAN SHANAHAN | POINTS: 102 | PLUSMINUS: -9  | PENALTY_MINUTES: 211 |
| NAME: 009 | JAROMIR JAGR     | POINTS: 99  | PLUSMINUS: 15  | PENALTY_MINUTES: 61  |
| NAME: 010 | DAVE ANDREYCHUK  | POINTS: 98  | PLUSMINUS: 22  | PENALTY_MINUTES: 98  |
| NAME: 011 | BRETT HULL       | POINTS: 97  | PLUSMINUS: -3  | PENALTY_MINUTES: 38  |
| NAME: 012 | ERIC LINDROS     | POINTS: 97  | PLUSMINUS: 16  | PENALTY_MINUTES: 103 |
| NAME: 013 | ROD BRIND'AMOUR  | POINTS: 97  | PLUSMINUS: -9  | PENALTY_MINUTES: 85  |
| NAME: 015 | RAY SHEPPARD     | POINTS: 93  | PLUSMINUS: 13  | PENALTY_MINUTES: 26  |
| NAME: 016 | MIKE MODANO      | POINTS: 93  | PLUSMINUS: -8  | PENALTY_MINUTES: 54  |
| NAME: 017 | ROBERT REICHEL   | POINTS: 93  | PLUSMINUS: 20  | PENALTY_MINUTES: 58  |
| NAME: 019 | JOE SAKIC        | POINTS: 92  | PLUSMINUS: -8  | PENALTY_MINUTES: 18  |
| NAME: 021 | RAY BOURQUE      | POINTS: 91  | PLUSMINUS: 26  | PENALTY_MINUTES: 58  |
| NAME: 022 | SERGEI ZUBOV     | POINTS: 89  | PLUSMINUS: 20  | PENALTY_MINUTES: 39  |
| NAME: 023 | KEVIN STEVENS    | POINTS: 88  | PLUSMINUS: -24 | PENALTY_MINUTES: 155 |
| NAME: 024 | LUC ROBITAILLE   | POINTS: 86  | PLUSMINUS: -20 | PENALTY_MINUTES: 86  |
| NAME: 026 | THEOREN FLEURY   | POINTS: 85  | PLUSMINUS: 30  | PENALTY_MINUTES: 186 |
| NAME: 028 | JOE JUNEAU       | POINTS: 85  | PLUSMINUS: 11  | PENALTY_MINUTES: 41  |
| NAME: 029 | GARY ROBERTS     | POINTS: 84  | PLUSMINUS: 37  | PENALTY_MINUTES: 145 |
| NAME: 030 | MARK MESSIER     | POINTS: 84  | PLUSMINUS: 25  | PENALTY_MINUTES: 76  |
| NAME: 031 | CRAIG JANNEY     | POINTS: 84  | PLUSMINUS: -1  | PENALTY_MINUTES: 24  |
| NAME: 033 | AL MACINNIS      | POINTS: 82  | PLUSMINUS: 35  | PENALTY_MINUTES: 95  |
| NAME: 034 | STEVE YZERMAN    | POINTS: 82  | PLUSMINUS: 11  | PENALTY_MINUTES: 36  |
| NAME: 035 | KEITH TKACHUK    | POINTS: 81  | PLUSMINUS: -12 | PENALTY_MINUTES: 255 |
| NAME: 037 | ADAM GRAVES      | POINTS: 79  | PLUSMINUS: 27  | PENALTY_MINUTES: 127 |
| NAME: 039 | *ALEXEI YASHIN   | POINTS: 79  | PLUSMINUS: -49 | PENALTY_MINUTES: 22  |
| NAME: 040 | BRIAN LEETCH     | POINTS: 79  | PLUSMINUS: 28  | PENALTY_MINUTES: 67  |
| NAME: 041 | SCOTT STEVENS    | POINTS: 78  | PLUSMINUS: 53  | PENALTY_MINUTES: 112 |
| NAME: 043 | PAUL COFFEY      | POINTS: 77  | PLUSMINUS: 28  | PENALTY_MINUTES: 106 |
| NAME: 044 | WENDEL CLARK     | POINTS: 76  | PLUSMINUS: 10  | PENALTY_MINUTES: 115 |
| NAME: 045 | STEVE THOMAS     | POINTS: 75  | PLUSMINUS: -9  | PENALTY_MINUTES: 139 |
| NAME: 046 | PAT VERBEEK      | POINTS: 75  | PLUSMINUS: -15 | PENALTY_MINUTES: 177 |
| NAME: 048 | CAM NEELY        | POINTS: 74  | PLUSMINUS: 12  | PENALTY_MINUTES: 54  |
| NAME: 049 | NELSON EMERSON   | POINTS: 74  | PLUSMINUS: -38 | PENALTY_MINUTES: 80  |
| NAME: 052 | KEITH PRIMEAU    | POINTS: 73  | PLUSMINUS: 34  | PENALTY_MINUTES: 173 |
| NAME: 054 | STEPHANE RICHER  | POINTS: 72  | PLUSMINUS: 31  | PENALTY_MINUTES: 16  |
| NAME: 057 | BOB KUDELSKI     | POINTS: 70  | PLUSMINUS: -33 | PENALTY_MINUTES: 24  |
| NAME: 058 | JOE MULLEN       | POINTS: 70  | PLUSMINUS: 9   | PENALTY_MINUTES: 41  |
| NAME: 059 | JOHN MACLEAN     | POINTS: 70  | PLUSMINUS: 30  | PENALTY_MINUTES: 95  |
| NAME: 060 | JOE MURPHY       | POINTS: 70  | PLUSMINUS: 1   | PENALTY_MINUTES: 111 |
| NAME: 063 | GEOFF COURTNALL  | POINTS: 70  | PLUSMINUS: 15  | PENALTY_MINUTES: 123 |
| NAME: 064 | GARRY GALLEY     | POINTS: 70  | PLUSMINUS: -11 | PENALTY_MINUTES: 91  |
| NAME: 066 | ULF DAHLEN       | POINTS: 69  | PLUSMINUS: -1  | PENALTY_MINUTES: 10  |
| NAME: 070 | ROB BLAKE        | POINTS: 68  | PLUSMINUS: -7  | PENALTY_MINUTES: 137 |
| NAME: 071 | GEOFF SANDERSON  | POINTS: 67  | PLUSMINUS: -13 | PENALTY_MINUTES: 42  |

|                             |  |
|-----------------------------|--|
| NAME: 075 MIKE GARTNER      | POINTS: 64 PLUSMINUS: 20 PENALTY_MINUTES: 62   |
| NAME: 081 SCOTT MELLANBY    | POINTS: 60 PLUSMINUS: 0 PENALTY_MINUTES: 149   |
| NAME: 083 CHRIS CHELIOS     | POINTS: 60 PLUSMINUS: 12 PENALTY_MINUTES: 212  |
| NAME: 090 AL IAFRATE        | POINTS: 58 PLUSMINUS: 16 PENALTY_MINUTES: 163  |
| NAME: 091 GLEN WESLEY       | POINTS: 58 PLUSMINUS: 1 PENALTY_MINUTES: 64    |
| NAME: 093 VALERI ZELEPUKIN  | POINTS: 57 PLUSMINUS: 36 PENALTY_MINUTES: 70   |
| NAME: 095 ZDENO CIGER       | POINTS: 57 PLUSMINUS: -11 PENALTY_MINUTES: 8   |
| NAME: 096 VLADIMIR MALAKHOV | POINTS: 57 PLUSMINUS: 29 PENALTY_MINUTES: 80   |
| NAME: 098 ALEXEI KOVALEV    | POINTS: 56 PLUSMINUS: 18 PENALTY_MINUTES: 154  |
| NAME: 100 IGOR LARIONOV     | POINTS: 56 PLUSMINUS: 20 PENALTY_MINUTES: 40   |
| NAME: 101 NICKLAS LIDSTROM  | POINTS: 56 PLUSMINUS: 43 PENALTY_MINUTES: 26   |
| NAME: 107 TEEMU SELANNE     | POINTS: 54 PLUSMINUS: -23 PENALTY_MINUTES: 22  |
| NAME: 117 ALEXEI ZHITNIK    | POINTS: 52 PLUSMINUS: -11 PENALTY_MINUTES: 101 |
| NAME: 121 SCOTT YOUNG       | POINTS: 51 PLUSMINUS: -4 PENALTY_MINUTES: 14   |
| NAME: 123 ALEXANDRE DAIGLE  | POINTS: 51 PLUSMINUS: -45 PENALTY_MINUTES: 40  |
| NAME: 124 THOMAS STEEN      | POINTS: 51 PLUSMINUS: -38 PENALTY_MINUTES: 32  |
| NAME: 125 SYLVAIN COTE      | POINTS: 51 PLUSMINUS: 30 PENALTY_MINUTES: 66   |
| NAME: 140 STEVE CHIASSON    | POINTS: 46 PLUSMINUS: 17 PENALTY_MINUTES: 122  |
| NAME: 146 BRAD MAY          | POINTS: 45 PLUSMINUS: -6 PENALTY_MINUTES: 171  |
| NAME: 151 PETER BONDRA      | POINTS: 43 PLUSMINUS: 22 PENALTY_MINUTES: 40   |
| NAME: 154 DAVE MCLLWAIN     | POINTS: 43 PLUSMINUS: -40 PENALTY_MINUTES: 48  |
| NAME: 173 KEVIN HATCHER     | POINTS: 40 PLUSMINUS: -13 PENALTY_MINUTES: 108 |

Teniendo en cuenta el tiempo requerido por cada programa para determinar los outliers del banco de datos, el algoritmo de fuerza bruta es claramente más eficiente ya que los resultados se obtienen en mucho menos tiempo y se consume menos memoria porque no es necesario mantener celdas, tan sólo se mantienen los objetos.

En esta sección pudo reproducirse la restricción del trabajo original que propone aplicar el algoritmo **FindAllOutsM** en conjuntos de datos de hasta 5 propiedades. Esta restricción se manifiesta en la imposibilidad de reservar memoria suficiente para mantener la complejidad del hipercubo que propone el algoritmo. Con la virtualización del hipercubo se superó dicha restricción eliminando la necesidad de reservar memoria para todas las celdas que se proponen en el trabajo original [Knorr y otros, 1998]. El programa respeta cada paso del algoritmo que sigue funcionando de la misma manera, mapea objetos contra celdas que se encuentran identificadas por su posición relativa en el hipercubo.

Creando sólo las celdas que son requeridas por el algoritmo se redujo drásticamente la cantidad de memoria necesaria para procesar el banco de datos a pesar de haber agregado listas para la administración de las celdas. En la figura 28 se puede



observar que para 11 dimensiones el algoritmo hubiese requerido casi dos mil millones de celdas (1.953.984.512 en total) y el experimento se llevó a cabo con tan solo 21.413. El programa que implementa el algoritmo **FindAllOutsM** es eficaz ya que encuentra todos los outliers que deben detectar los algoritmos basados en distancia. Con las últimas modificaciones realizadas a la implementación del algoritmo se mejoró la eficiencia.

Hay que destacar que a pesar de haber modificado la cantidad de atributos no se han cambiado en ningún momento los parámetros de la ejecución del programa. Para cambiar los parámetros de ejecución del programa sería necesario, como se planteó oportunamente en la sección 7.4.1.7, la asistencia del usuario para que, con su conocimiento del dominio de los datos, pudiese ajustar los parámetros para obtener resultados relevantes.

## 9.5. PROCESANDO MAYORES VOLÚMENES DE DATOS

El primer inconveniente que surge a la hora de procesar grandes volúmenes de datos es disponer de algún banco de datos que sea capaz de proveer una gran cantidad de filas para procesar. No siempre hay una relación directa entre el tamaño de la base de datos y la cantidad de tuplas que se pueden procesar. El ejemplo más cercano al que se tuvo acceso, y que se puede mencionar, es una base de datos de 480 GB de análisis médicos cuyo tamaño podía ser explicado en un 90% por archivos PDFs e imágenes almacenadas en la base de datos. La tabla que más información podía proveer tenía tan sólo 350 mil tuplas de las cuales sólo 3 atributos podían ser utilizados en el contexto de este trabajo final (campos numéricos). Por lo tanto, una gran base de datos, no siempre implica un desafío a la hora de buscar outliers (al menos del tipo planteados en este trabajo) ya que versiones menos complejas del algoritmo **FindAllOutsM** podrían procesarlas sin demasiados inconvenientes.

Para experimentar con los límites del algoritmo **FindAllOutsM** se creó un banco de datos artificial basado en NHL94. Este nuevo banco de datos de laboratorio se creó con nuevas tuplas utilizando la técnica descrita en el trabajo original [Knorr y otros, 1998] para la creación de tuplas sintéticas. El crecimiento que se le impuso al banco

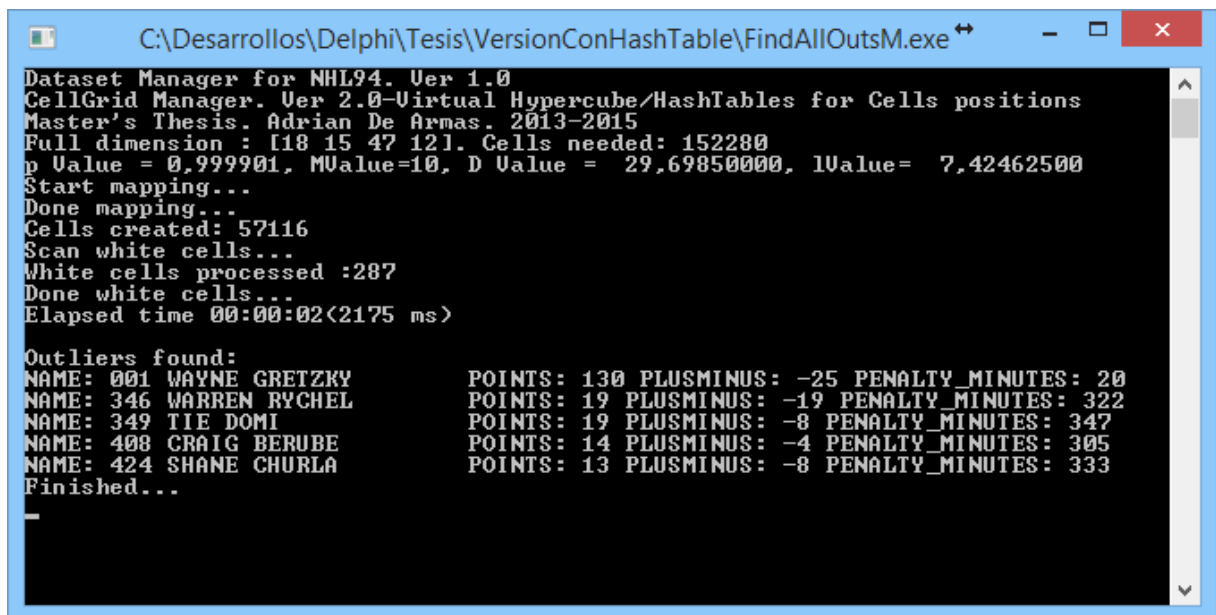
fue 100.000, 500.000, 1.000.000, 2.500.000 y 5.000.000 y en todos los casos se volvió a correr las pruebas con distinta suerte y, a partir de los inconvenientes presentados, se realizaron mejoras que fueron incorporadas.

**$N = 100.871$**

Para este experimento se procedió a disminuir la cantidad de atributos a procesar a 4 (una cantidad discreta pero la máxima que se podía utilizar en el trabajo original) para poder sacar conclusiones respecto al funcionamiento del algoritmo a medida que el tamaño del banco de datos aumentaba.

El primer experimento fue realizado con 100.871 filas a procesar, se estipuló que si no se encontraban 10 jugadores dentro del vecindario más cercano un jugador sería considerado un outlier, los atributos considerados fueron POINTS, PLUSMINUS, PENALTY\_MINUTES y GAMES\_PLAYED. La decisión tomada respecto a qué atributos tener en cuenta y el valor de  $M$  utilizado son absolutamente arbitrarios según lo planteado en la sección 9 respecto a atributos, valor de  $p$  y valor de  $D$  para la ejecución del algoritmo.

El resultado de la ejecución puede observarse en la figura 30



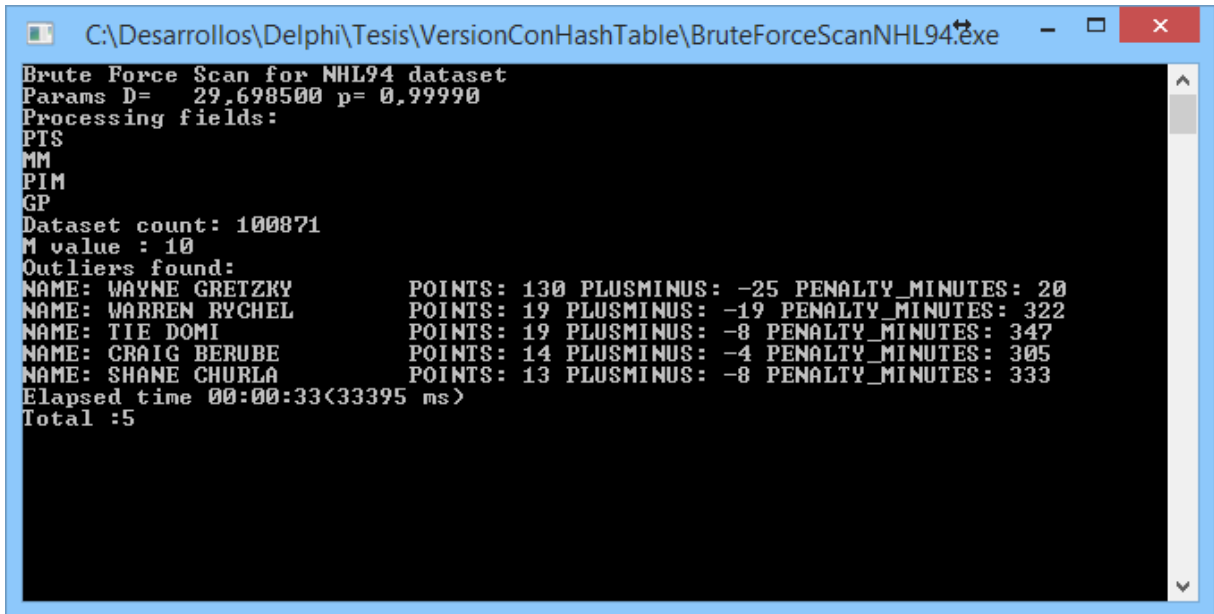
```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\FindAllOutsM.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [18 15 47 12]. Cells needed: 152280
p Value = 0,999901, MValue=10, D Value = 29,69850000, lValue= 7,42462500
Start mapping...
Done mapping...
Cells created: 57116
Scan white cells...
White cells processed :287
Done white cells...
Elapsed time 00:00:02(2175 ms)

Outliers found:
NAME: 001 WAYNE GRETZKY      POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
NAME: 346 WARREN RYCHEL     POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
NAME: 349 TIE DOMI         POINTS: 19 PLUSMINUS: -8 PENALTY_MINUTES: 347
NAME: 408 CRAIG BERUBE     POINTS: 14 PLUSMINUS: -4 PENALTY_MINUTES: 305
NAME: 424 SHANE CHURLA     POINTS: 13 PLUSMINUS: -8 PENALTY_MINUTES: 333
Finished...
    
```

Figura 30 – FindAllOutsM para  $N=100.0871$  y  $D=4$ .

La correspondiente confirmación de resultados se realizó por fuerza bruta y puede observarse en la figura 31 y en la figura 32 puede observarse la memoria consumida por los dos programas



```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\BruteForceScanNHL94.exe
Brute Force Scan for NHL94 dataset
Params D= 29.698500 p= 0.99990
Processing fields:
PTS
MM
PIM
GP
Dataset count: 100871
M value : 10
Outliers found:
NAME: WAYNE GRETZKY          POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
NAME: WARREN RYCHEL         POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
NAME: TIE DOMI              POINTS: 19 PLUSMINUS: -8  PENALTY_MINUTES: 347
NAME: CRAIG BERUBE         POINTS: 14 PLUSMINUS: -4  PENALTY_MINUTES: 305
NAME: SHANE CHURLA         POINTS: 13 PLUSMINUS: -8  PENALTY_MINUTES: 333
Elapsed time 00:00:33(33395 ms)
Total :5
    
```

Figura 31 – Fuerza bruta para N=100.871 y D=4.

| Nombre                  | Estado | CPU  | Memoria  | Disco  | Red    |
|-------------------------|--------|------|----------|--------|--------|
| <b>Aplicaciones (9)</b> |        |      |          |        |        |
| Administrador de tareas |        | 0,3% | 14,1 MB  | 0 MB/s | 0 Mbps |
| BruteForceScanNHL94.exe |        | 0%   | 359,6 MB | 0 MB/s | 0 Mbps |
| FindAllOutsM.exe        |        | 0%   | 62,5 MB  | 0 MB/s | 0 Mbps |

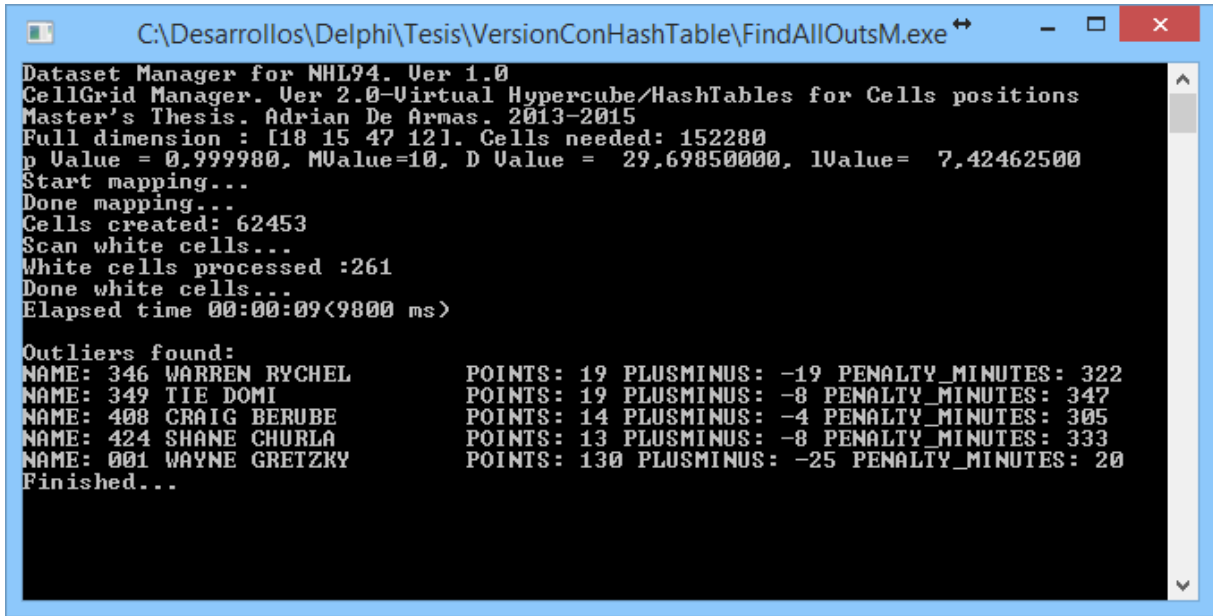
Figura 32 – Consumo de memoria para **FindAllOutsM** y Fuerza bruta con N=100.871 y D=4.

Es importante notar que el algoritmo **FindAllOutsM** por primera vez tuvo mejor rendimiento que la comprobación por fuerza bruta. Para las condiciones dadas, no fue necesario cambiar la implementación realizada hasta el momento.

**FindAllOutsM**, al obtener los mismos resultados que la fuerza bruta, demuestra su eficacia mientras que respecto a la eficiencia, siguiendo el criterio establecido en la sección 9, **FindAllOutsM** es mucho más eficiente ya que obtuvo los mismos resultados en apenas el 6.5% del tiempo insumido por la fuerza bruta y consumiendo

apenas el 17% de memoria consumido por la determinación por fuerza bruta, si bien el segundo criterio de eficiencia se descarta como se estableció en la sección 9.

$$N = 500.871$$



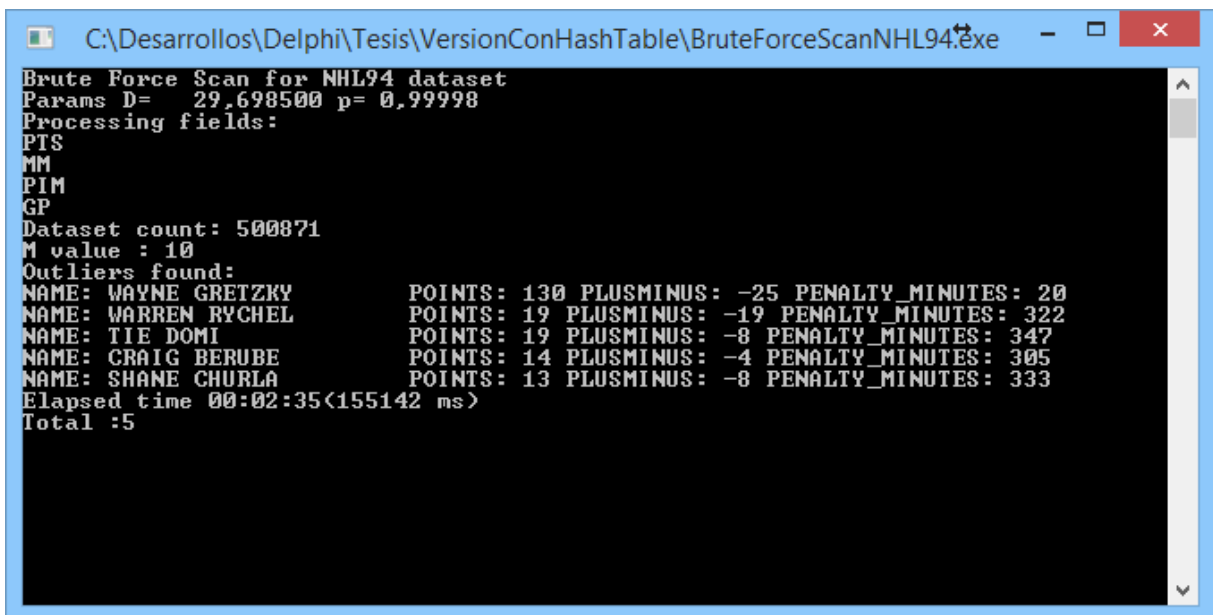
```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\FindAllOutsM.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [18 15 47 12]. Cells needed: 152280
p Value = 0,999980, MValue=10, D Value = 29,69850000, lValue= 7,42462500
Start mapping...
Done mapping...
Cells created: 62453
Scan white cells...
White cells processed :261
Done white cells...
Elapsed time 00:00:09<9800 ms>

Outliers found:
NAME: 346 WARREN RYCHEL          POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
NAME: 349 TIE DOMI              POINTS: 19 PLUSMINUS: -8  PENALTY_MINUTES: 347
NAME: 408 CRAIG BERUBE         POINTS: 14 PLUSMINUS: -4  PENALTY_MINUTES: 305
NAME: 424 SHANE CHURLA        POINTS: 13 PLUSMINUS: -8  PENALTY_MINUTES: 333
NAME: 001 WAYNE GRETZKY       POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
Finished...
    
```

Figura 33 – FindAllOutsM para N=500.0871 y D=4.

En la figura 33 puede observarse el resultado de procesar más de medio millón de filas y la comprobación por fuerza bruta puede observarse en la figura 34, mientras que en la figura 35 puede observarse el consumo de memoria de ambos programas



```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\BruteForceScanNHL94.exe
Brute Force Scan for NHL94 dataset
Params D= 29,698500 p= 0,99998
Processing fields:
PTS
MM
PIM
GP
Dataset count: 500871
M value : 10
Outliers found:
NAME: WAYNE GRETZKY          POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
NAME: WARREN RYCHEL         POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
NAME: TIE DOMI              POINTS: 19 PLUSMINUS: -8  PENALTY_MINUTES: 347
NAME: CRAIG BERUBE         POINTS: 14 PLUSMINUS: -4  PENALTY_MINUTES: 305
NAME: SHANE CHURLA        POINTS: 13 PLUSMINUS: -8  PENALTY_MINUTES: 333
Elapsed time 00:02:35<155142 ms>
Total :5
    
```

Figura 34 – Fuerza bruta para N=500.871 y D=4.

| Nombre                  | Estado | CPU  | Memoria    | Disco  | Red    |
|-------------------------|--------|------|------------|--------|--------|
| <b>Aplicaciones (8)</b> |        |      |            |        |        |
| Administrador de tareas |        | 0,2% | 14,1 MB    | 0 MB/s | 0 Mbps |
| BruteForceScanNHL94.exe |        | 0%   | 1.771,4 MB | 0 MB/s | 0 Mbps |
| FindAllOutsM.exe        |        | 0%   | 201,3 MB   | 0 MB/s | 0 Mbps |

 Figura 35 – Consumo de memoria para **FindAllOutsM** y Fuerza bruta con N=500.871 y D=4.

En este experimento la cantidad de outliers detectados es la misma y los resultados coinciden. **FindAllOutsM** es igual de eficaz que el algoritmo por fuerza bruta pero resuelve la detección de outliers en una fracción del tiempo de insume la fuerza bruta y con mucho menos consumo de memoria. Tener en cuenta que el algoritmo de fuerza bruta no tiene ningún tipo de optimización lo cual puede influir en los números presentados pero nunca en el resultado final.

El algoritmo **FindAllOutsM** demostró un gran rendimiento al detectar todos los outliers en casi 9 segundos, mientras que la comprobación por fuerza bruta insumió 2 minutos, 35 segundos.

Esta diferencia se explica en que la complejidad del algoritmo **FindAllOutsM** es (en el peor escenario)  $O(c^k + N)$  donde  $c$  es alguna constante dependiente de  $\sqrt{k}$  y  $m^{1/k}$  (que corresponde aproximadamente al número de celdas a lo largo de cada dimensión) mientras que la complejidad de la fuerza bruta es  $O(N^2)$  de crecimiento exponencial. En la sección 8.1 se puede leer sobre el algoritmo **FindAllOutsM** y su análisis de complejidad.

$$N = 1.000.871$$

Con un millón de registros a procesar ambos algoritmos encontraron los mismos outliers como puede observarse en las figuras 36 y 37, mientras que el consumo de memoria de ambos programas puede observarse en la figura 38.

```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\FindAllOutsM.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [18 15 47 12]. Cells needed: 152280
p Value = 0,999990, MValue=10, D Value = 29,69850000, lValue= 7,42462500
Start mapping...
Done mapping...
Cells created: 64369
Scan white cells...
White cells processed :257
Done white cells...
Elapsed time 00:00:20(20627 ms)

Outliers found:
NAME: 346 WARREN RYCHEL          POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
NAME: 349 TIE DOMI              POINTS: 19 PLUSMINUS: -8 PENALTY_MINUTES: 347
NAME: 408 CRAIG BERUBE         POINTS: 14 PLUSMINUS: -4 PENALTY_MINUTES: 305
NAME: 424 SHANE CHURLA        POINTS: 13 PLUSMINUS: -8 PENALTY_MINUTES: 333
NAME: 001 WAYNE GRETZKY       POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
Finished...
  
```

Figura 36 – FindAllOutsM para N=1.000.0871 y D=4.

```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\BruteForceScanNHL94.exe
Brute Force Scan for NHL94 dataset
Params D= 29,698500 p= 0,99998
Processing fields:
PTS
MM
PIM
GP
Dataset count: 1000871
M value : 10
Outliers found:
NAME: WAYNE GRETZKY          POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
NAME: WARREN RYCHEL        POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
NAME: TIE DOMI             POINTS: 19 PLUSMINUS: -8 PENALTY_MINUTES: 347
NAME: CRAIG BERUBE        POINTS: 14 PLUSMINUS: -4 PENALTY_MINUTES: 305
NAME: SHANE CHURLA       POINTS: 13 PLUSMINUS: -8 PENALTY_MINUTES: 333
Elapsed time 00:05:03(303788 ms)
Total :5
  
```

Figura 37 – Fuerza bruta para N=1.000.871 y D=4.

| Nombre                  | Estado | CPU  | Memoria    | Disco  | Red    |
|-------------------------|--------|------|------------|--------|--------|
| <b>Aplicaciones (8)</b> |        |      |            |        |        |
| Administrador de tareas |        | 0,4% | 14,5 MB    | 0 MB/s | 0 Mbps |
| BruteForceScanNHL94.exe |        | 0%   | 3.536,8 MB | 0 MB/s | 0 Mbps |
| FindAllOutsM.exe        |        | 0%   | 372,9 MB   | 0 MB/s | 0 Mbps |

Figura 38 – Consumo de memoria para **FindAllOutsM** y Fuerza bruta con  $N=1.000.871$  y  $D=4$ .

Nuevamente el algoritmo **FindAllOutsM** logró detectar outliers consumiendo, como era de esperar, mucha más memoria para el proceso, pero mucha menos que la determinación por fuerza bruta. El tiempo de procesamiento de un millón de filas es bueno (comparado con la fuerza bruta).

A partir de estos resultados se consideró irrelevante seguir comprobando el correcto funcionamiento de la implementación de **FindAllOutsM** realizada contra una comprobación hecha por fuerza bruta. Se observa que ambos algoritmos son igualmente eficaces pero la implementación de **FindAllOutsM** es decididamente más eficiente. Del algoritmo por fuerza bruta (que no dispone de ningún tipo de optimización) sólo se podía esperar mayor consumo de memoria y un incremento del tiempo necesario para la detección de todos los outliers a medida que el número de tuplas fuera incrementándose.

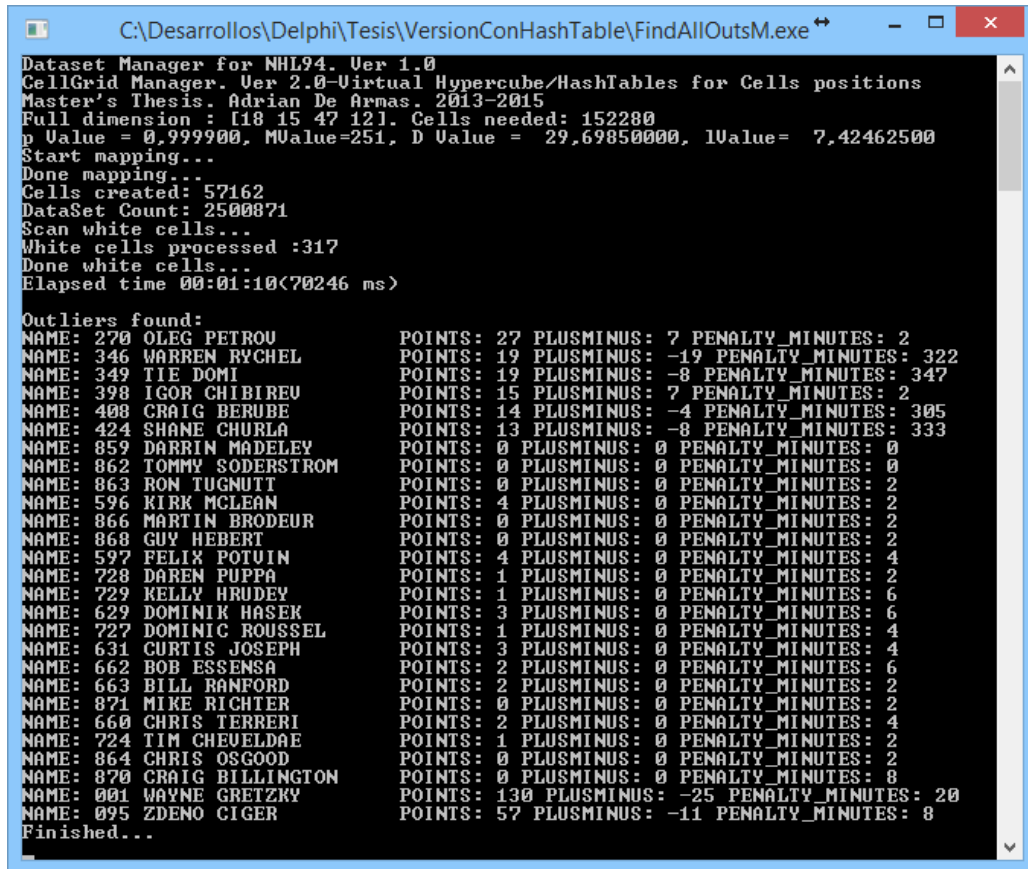
**$N = 2.500.871$  en 4D**

Para esta cantidad de tuplas a procesar, se modificó el valor de  $p$  a 0,9999 para reproducir (parcialmente) el experimento del trabajo original para  $N = 2.000.000$  como se puede observar en la figura 39, el consumo de memoria de la ejecución del algoritmo puede observarse en la figura 41, mientras que el resultado obtenido se puede observar en la figura 40.

CPU + I/O Times (in Seconds) for a Variable Number of Tuples, Dimensions, and Cells—for  $p = 0.9999$ .

| $N$       | 3-D          |         | 4-D          |         | 5-D          |             |             |         |
|-----------|--------------|---------|--------------|---------|--------------|-------------|-------------|---------|
|           | CS( $10^3$ ) | NL      | CS( $10^4$ ) | NL      | CS( $10^5$ ) | CS( $8^5$ ) | CS( $6^5$ ) | NL      |
| 100,000   | 10.77        | 93.96   | 23.32        | 45.79   | 93.40        | 217.04      | 205.63      | 17.30   |
| 500,000   | 57.10        | 490.62  | 114.00       | 223.51  | 695.37       | 997.11      | 1061.33     | 148.44  |
| 2,000,000 | 253.90       | 2332.10 | 606.56       | 1421.16 | >2147        | >2147       | >2147       | 1555.78 |

Figura 39 – Resultados del trabajo original para  $N=2.000.000$  y  $D=4$  [Knorr y otros, 1998].



```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\FindAllOutsM.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [18 15 47 12]. Cells needed: 152280
p Uvalue = 0,999900, MValue=251, D Uvalue = 29,69850000, lUvalue= 7,42462500
Start mapping...
Done mapping...
Cells created: 57162
DataSet Count: 2500871
Scan white cells...
White cells processed :317
Done white cells...
Elapsed time 00:01:10<70246 ms>

Outliers found:
NAME: 270 OLEG PETROU          POINTS: 27 PLUSMINUS: 7 PENALTY_MINUTES: 2
NAME: 346 WARREN RYCHEL       POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
NAME: 349 TIE DOMI            POINTS: 19 PLUSMINUS: -8 PENALTY_MINUTES: 347
NAME: 398 IGOR CHIBIREU       POINTS: 15 PLUSMINUS: 7 PENALTY_MINUTES: 2
NAME: 408 CRAIG BERUBE        POINTS: 14 PLUSMINUS: -4 PENALTY_MINUTES: 305
NAME: 424 SHANE CHURLA        POINTS: 13 PLUSMINUS: -8 PENALTY_MINUTES: 333
NAME: 859 DARRIN MADELEY       POINTS: 0 PLUSMINUS: 0 PENALTY_MINUTES: 0
NAME: 862 TOMMY SODERSTROM     POINTS: 0 PLUSMINUS: 0 PENALTY_MINUTES: 0
NAME: 863 RON TUGNUTT          POINTS: 0 PLUSMINUS: 0 PENALTY_MINUTES: 2
NAME: 596 KIRK MCLEAN          POINTS: 4 PLUSMINUS: 0 PENALTY_MINUTES: 2
NAME: 866 MARTIN BRODEUR       POINTS: 0 PLUSMINUS: 0 PENALTY_MINUTES: 2
NAME: 868 GUY HEBERT           POINTS: 0 PLUSMINUS: 0 PENALTY_MINUTES: 2
NAME: 597 FELIX POTVIN         POINTS: 4 PLUSMINUS: 0 PENALTY_MINUTES: 4
NAME: 728 DAREN PUPPA          POINTS: 1 PLUSMINUS: 0 PENALTY_MINUTES: 2
NAME: 729 KELLY HRUDEY         POINTS: 1 PLUSMINUS: 0 PENALTY_MINUTES: 6
NAME: 629 DOMINIK HASEK        POINTS: 3 PLUSMINUS: 0 PENALTY_MINUTES: 6
NAME: 727 DOMINIC ROUSSEL      POINTS: 1 PLUSMINUS: 0 PENALTY_MINUTES: 4
NAME: 631 CURTIS JOSEPH        POINTS: 3 PLUSMINUS: 0 PENALTY_MINUTES: 4
NAME: 662 BOB ESSENSA          POINTS: 2 PLUSMINUS: 0 PENALTY_MINUTES: 6
NAME: 663 BILL RANFORD         POINTS: 2 PLUSMINUS: 0 PENALTY_MINUTES: 2
NAME: 871 MIKE RICHTER         POINTS: 0 PLUSMINUS: 0 PENALTY_MINUTES: 2
NAME: 660 CHRIS TERRERI        POINTS: 2 PLUSMINUS: 0 PENALTY_MINUTES: 4
NAME: 724 TIM CHEVELDAE        POINTS: 1 PLUSMINUS: 0 PENALTY_MINUTES: 2
NAME: 864 CHRIS OSGOOD         POINTS: 0 PLUSMINUS: 0 PENALTY_MINUTES: 2
NAME: 870 CRAIG BILLINGTON     POINTS: 0 PLUSMINUS: 0 PENALTY_MINUTES: 8
NAME: 001 WAYNE GRETZKY        POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
NAME: 095 ZDENO CIGER          POINTS: 57 PLUSMINUS: -11 PENALTY_MINUTES: 8
Finished...
    
```

Figura 40 –FindAllOutsM para N=2.500.871 y D=4.

Aunque los experimentos no sean comparables por los 17 años transcurridos desde que se presentó el algoritmo y la mejora que experimentó el hardware de los equipos en este lapso, con este experimento se superó la cantidad máxima de tuplas procesadas en el trabajo original que fue 2 millones. En aquel momento (año 1998) se procesó un archivo binario de 2 millones de tuplas alojado en el mismo equipo donde se estaba experimentando en, aproximadamente, 10 minutos, mientras que con la implementación de **FindAllOutsM** se obtuvieron resultados (confirmados mediante fuerza bruta) en 70 segundos extrayendo los datos a analizar desde una base de datos.



| Nombre                   | Estado | CPU   | Memoria    | Disco  | Red    |
|--------------------------|--------|-------|------------|--------|--------|
| <b>Aplicaciones (10)</b> |        |       |            |        |        |
| Administrador de tareas  |        | 0,7%  | 14,6 MB    | 0 MB/s | 0 Mbps |
| BruteForceScanNHL94.exe  |        | 18,4% | 8.848,0 MB | 0 MB/s | 0 Mbps |
| FindAllOutsM.exe         |        | 0%    | 872,0 MB   | 0 MB/s | 0 Mbps |

Figura 41 – Consumo de memoria para **FindAllOutsM** y Fuerza bruta con  $N=2.500.871$  y  $D=4$ .

$N = 5.000.871$

En la figura 42 puede observarse el resultado de procesar más de 5 millones de tuplas. Ya se comprobó la eficacia del algoritmo **FindAllOutsM** y su eficiencia se pone en manifiesto teniendo en cuenta que la relación entre el tiempo insumido hasta obtener resultados y la cantidad de tuplas que el algoritmo procesó arroja un promedio de procesamiento de 25 objetos por milisegundo.

```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\FindAllOutsM.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [18 15 47 12]. Cells needed: 152280
p Uvalue = 0,999998, MValue=10, D Uvalue = 29,69850000, lUvalue= 7,42462500
Start mapping...
Done mapping...
Cells created: 64369
DataSet Count: 5000871
Scan white cells...
White cells processed :257
Done white cells...
Elapsed time 00:03:18<198876 ms>

Outliers found:
NAME: 346 WARREN RYCHEL          POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
NAME: 349 TIE DOMI              POINTS: 19 PLUSMINUS: -8  PENALTY_MINUTES: 347
NAME: 408 CRAIG BERUBE          POINTS: 14 PLUSMINUS: -4  PENALTY_MINUTES: 305
NAME: 424 SHANE CHURLA          POINTS: 13 PLUSMINUS: -8  PENALTY_MINUTES: 333
NAME: 001 WAYNE GRETZKY         POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
Total : 5
Finished...
    
```

Figura 42 –**FindAllOutsM** para  $N=5.000.871$  y  $D=4$ .

| Nombre                  | Estado | CPU  | Memoria    | Disco  | Red    |
|-------------------------|--------|------|------------|--------|--------|
| <b>Aplicaciones (8)</b> |        |      |            |        |        |
| Administrador de tareas |        | 0,2% | 13,4 MB    | 0 MB/s | 0 Mbps |
| FindAllOutsM.exe        |        | 0%   | 1.746,1 MB | 0 MB/s | 0 Mbps |

Figura 43 – Consumo de memoria para **FindAllOutsM** y Fuerza bruta con  $N=5.000.871$  y  $D=4$ .

Si bien en la sección 9 se estableció que la cantidad de memoria requerida no sería utilizada para medir la eficiencia del algoritmo a menos que se estuvieran comparando dos algoritmos distintos con rendimiento parecido en tiempo de ejecución, para 5 millones de tuplas el consumo de memoria comienza a ser crítico como se observa en la figura 43. Se considera crítico porque un sistema operativo de 32 bits puede direccionar hasta 4GB de memoria y si el programa, con cantidades crecientes de tuplas, requiere cada vez más memoria, es tan sólo cuestión de llegar al límite del sistema operativo para que el algoritmo, implementado como está, no pueda ser utilizado para detectar outliers.

Del análisis del caso  $N = 5.000.871$  en  $4D$  se observó que de las celdas creadas la gran mayoría pertenecía a celdas necesarias por la consulta de las celdas vecinas  $L_1$  y, permanecían vacías lo cual le agregaba complejidad al algoritmo como se analizó en la sección 8.1.5.

Otra situación que explica la cantidad de memoria utilizada y que es un factor determinante a la hora de poner en riesgo la ejecución del algoritmo, es la cantidad de memoria utilizada por los objetos de jugadores. Para el caso analizado son 5.000.871 objetos de jugadores distribuidos a lo largo de las celdas creadas, todas en memoria. El objetivo final (respecto a **FindAllOutsM**) es brindarle al algoritmo la mejor posibilidad de ejecución en cualquier escenario. Sigue siendo una alternativa compilar el programa para 64 bits pero de esta forma trasladamos el inconveniente de poder (o no) obtener resultados con el programa a la cantidad de memoria disponible en el equipo.

Los nuevos objetivos quedan bien expuestos:

1. Reducir a 0 la cantidad de celdas vacías administradas en memoria.
2. Crear un mecanismo que permita alojar en memoria sólo aquellos objetos que sean relevantes.

### **OBJETIVO 1: REDUCIR A CERO LA CANTIDAD DE CELDAS VACIAS**

Durante la etapa de mapeo de los objetos a su respectiva celda, cada celda conoce sus vecinos  $L_1$  (en realidad sólo calcula sus coordenadas) y si la cantidad de elementos alojados en la propia celda supera el umbral  $M$ , la celda se colorea de rojo y, consecuentemente, colorea como rosado a su vecindario  $L_1$  para indicar que en ninguna de ellas podría alojarse un outlier. En este proceso las celdas del vecindario  $L_1$ , de no existir, se creaban sólo para mantener la propiedad del color rosa mientras que el resto de las propiedades como ser índice, coordenadas (mayor dimensión, mayor consumo de memoria), y colección de objetos asociados se crean y pasan a consumir memoria en el proceso.

Si al continuar con el mapeo de objetos, alguna celda de las coloreadas como rosas, se ocupaba con objetos, la celda tenía sentido para el algoritmo, pero el resultado de los experimentos arroja que son muchas las celdas que se creaban innecesariamente sobretodo en bancos de datos multidimensionales con muchas dimensiones (más de 5).

Se modificó la implementación del algoritmo para no crear celdas cuando el algoritmo está accediendo un vecindario  $L_1$  para colorearlo (si existe se colorea), para esto se lleva un inventario de todas aquellas celdas que no fueron creadas pero que, de crearse, deberían llevar el color rosado. Esto es relevante porque todas las celdas rosadas no son tenidas en cuenta en el último proceso del algoritmo (el paso 5) donde una celda puede terminar calculando la distancia contra todos sus vecinos  $L_2$  (el proceso más caro, en tiempo de ejecución, de todo el algoritmo). Para llevar dicho inventario se creó otro diccionario como el que permitió virtualizar el hipercubo del algoritmo **FindAllOutsM** y fue descrito en la sección 9.3

**OBJETIVO 2: ALOJAR EN MEMORIA SOLO LOS OBJETOS RELEVANTES**

La dificultad que presenta este desafío es que no es fácil de determinar cuáles son aquellos “objetos relevantes”. Si se consideran como relevantes a todos aquellos objetos que se encuentran en celdas blancas (por lo tanto, aquellos con posibilidades de ser outliers) y, consecuentemente, se liberan todos los objetos alojados en celdas rojas y rosadas, se liberarían objetos que pueden ser necesarios para determinar si un objeto en particular es o no un outlier ya que una celda roja puede ser parte del vecindario  $L_2$  de una celda blanca.

Aunque desconocida, es de esperar que todos los bancos de datos que se procesen sigan alguna distribución determinada (aunque sea por partes) como ejemplifica la figura 44 dónde se pueden visualizar distintas zonas en las que se agrupan los datos del banco que se está procesando.

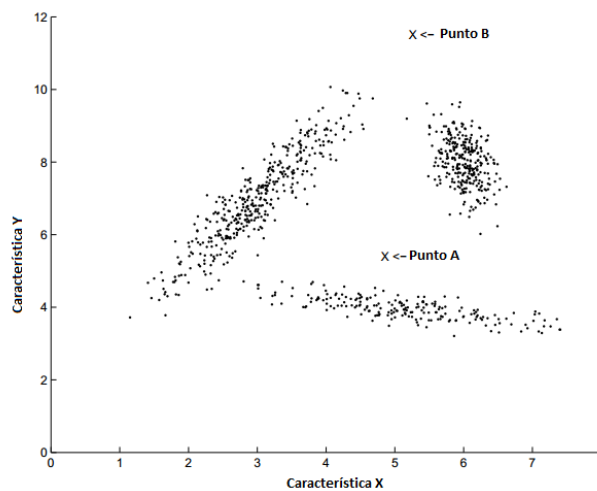


Figura 44 – Banco de datos modelo para detección de outliers [Aggarwal, 2013]

Es de esperar que si los datos siguen alguna distribución específica las celdas rojas y rosadas se encuentren con mayor probabilidad rodeadas de celdas rojas y rosadas salvo en la periferia de los clusters.

Teniendo en cuenta esto, la implementación del algoritmo podría liberar de memoria todos aquellos objetos que se encuentran en celdas rojas o rosadas dejando disponibles sólo aquellos con posibilidades de ser outliers, es decir, aquellos alojados en celdas blancas. Esta última opción deja abierta la posibilidad de ejecutar

el algoritmo con un banco de datos multidimensional con una cantidad grande de atributos donde los objetos se encuentren muy dispersos entre sí, generando de esta manera, numerosas celdas blancas. Esta situación puede poner en riesgo la conclusión del proceso por falta de memoria.

Los datos que se utilizan para determinar si un objeto es o no es un outlier, generalmente no identifican la tupla inequívocamente, como por ejemplo en el banco de datos que se está usando en este trabajo final (el archivo NHL94), los jugadores se identifican por el nombre y no por el resto de sus atributos. Es por eso que cada objeto, más allá de sus atributos numéricos utilizados para determinar si es o no outlier, tienen un atributo extra obligatorio que es una clave que identifica la tupla. Hasta el momento la clave se utilizaba únicamente para exponer (de forma clara) el outlier detectado. Como parte de la solución para el desafío planteado se requiere que la clave sirva para volver a obtener los datos del objeto a demanda. Así, cuando una celda requiera todos los objetos que se encuentran en otra celda, el programa pueda volver a obtener, a voluntad, los datos necesarios para calcular las distancias objeto a objeto como lo requiere el algoritmo. Por las características propias del algoritmo la cantidad de objetos que deban ser releídos debe ser muy inferior al número total de tuplas del banco de datos.

Para guardar la relación celda/clave del objeto es necesario un almacenamiento secundario distinto a la memoria, por lo tanto se utiliza un archivo indexado para guardar esta información y poder recuperarla a voluntad.

Finalmente, la respuesta dada a este desafío es la de instanciar en un archivo externo la relación de todos los objetos (independientemente del color de celda donde se alojen) y se recuperan sólo aquellos objetos que sean requeridos en el paso final del algoritmo.

Las modificaciones que se realizaron a la implementación del algoritmo con el fin de extender su uso son:

1. Virtualización del hipercubo propuesto por el algoritmo
2. Virtualización de celdas de vecindarios  $L_1$

3. Persistencia de la relación objeto/celda en un almacenamiento secundario para disponer de más memoria
4. Mecanismo para recuperar los objetos asociados a una celda a voluntad

Como última mejora el programa que implementa el algoritmo **FindAllOutsM** pasa a ser de 64 bits para maximizar las posibilidades de asignación de memoria. En este caso toda la memoria se utiliza para asignar celdas solamente.

La nueva versión del programa, habiendo procesado más de 4 millones de tuplas, como se puede observar en la figura 45, producía un consumo de memoria acorde a lo esperado por a las modificaciones realizadas.



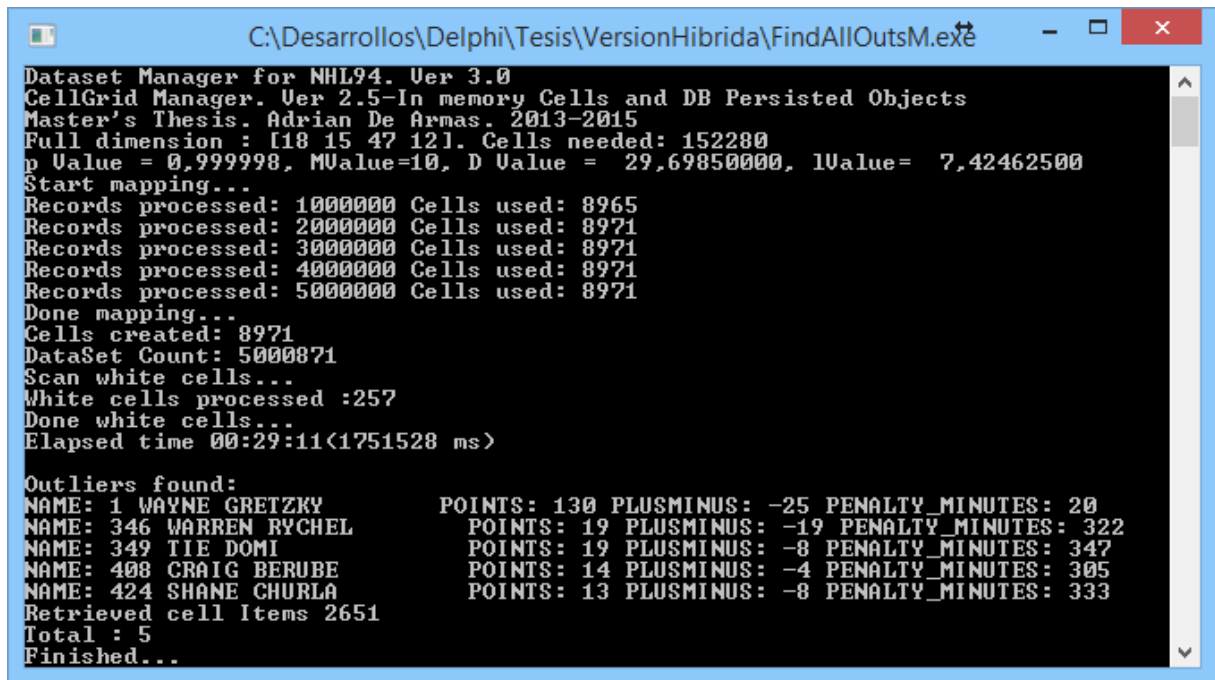
| Nombre   | Estado | CPU   | Memoria  | Disco     | Red    |
|--|--------|-------|----------|-----------|--------|
| <b>Aplicaciones (10)</b>   |        |       |          |           |        |
| ▶  Administrador de tareas |        | 0,1%  | 14,7 MB  | 0 MB/s    | 0 Mbps |
| ▶  FindAllOutsM.exe       |        | 19,1% | 182,8 MB | 21,5 MB/s | 0 Mbps |

Figura 45 – Memoria consumida por **FindAllOutsM** versión híbrida para N=5.000.871 y D=4.

Finalmente, el resultado de procesar más de 5 millones de tuplas puede observarse en la figura 46 cuyos resultados coinciden con la versión que procesa todas las tuplas en memoria (validada contra una determinación de outliers por fuerza bruta) como se observa en la figura 42.



```

Dataset Manager for NHL94. Ver 3.0
CellGrid Manager. Ver 2.5-In memory Cells and DB Persisted Objects
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [18 15 47 12]. Cells needed: 152280
p Value = 0,9999998, MValue=10, D Value = 29,69850000, IValue= 7,42462500
Start mapping...
Records processed: 1000000 Cells used: 8965
Records processed: 2000000 Cells used: 8971
Records processed: 3000000 Cells used: 8971
Records processed: 4000000 Cells used: 8971
Records processed: 5000000 Cells used: 8971
Done mapping...
Cells created: 8971
DataSet Count: 5000871
Scan white cells...
White cells processed :257
Done white cells...
Elapsed time 00:29:11<1751528 ms>

Outliers found:
NAME: 1 WAYNE GRETZKY          POINTS: 130 PLUSMINUS: -25 PENALTY_MINUTES: 20
NAME: 346 WARREN RYCHEL       POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
NAME: 349 TIE DOMI            POINTS: 19 PLUSMINUS: -8 PENALTY_MINUTES: 347
NAME: 408 CRAIG BERUBE        POINTS: 14 PLUSMINUS: -4 PENALTY_MINUTES: 305
NAME: 424 SHANE CHURLA        POINTS: 13 PLUSMINUS: -8 PENALTY_MINUTES: 333
Retrieved cell Items 2651
Total : 5
Finished...
    
```

Figura 46 –FindAllOutsM híbrido para N=5.000.871 y D=4.

Las modificaciones realizadas alcanzaron sus objetivos y disminuyeron considerablemente el consumo de memoria del programa. 5 millones de objetos (y sus celdas asociadas) consumieron no más de 190 megas de memoria al momento de realizar el proceso. Esta modificación extendió considerablemente la cantidad de filas que se pueden procesar ya que toda la memoria se utiliza para alojar celdas. Con el programa compilado para 64 bits, el único limitante de la cantidad de celdas a crear es la cantidad de memoria disponible en el equipo.

Esta nueva versión de la implementación del algoritmo **FindAllOutsM** resulta mucho menos eficiente que su predecesora. Los motivos de esta situación tiene muchos orígenes:

- El uso del almacenamiento secundario (un disco duro es notablemente más lento que la RAM)
- Se torna mucho más relevante la capacidad del motor de bases de datos para entregar las tuplas solicitadas
- El esquema de base de datos que se esté procesando debe tener los índices necesarios para poder recuperar las tuplas rápidamente

- El rendimiento por requerir al motor de base de datos varias veces cientos de tuplas (las tuplas pertenecientes a una celda en particular) es inferior a requerir pocas veces miles de tuplas y procesarlas

Otro factor que influye negativamente en el rendimiento del programa es la cantidad de celdas blancas que resulten de la etapa de mapeo ya que por cada celda que se lea y por cada celda que compone su vecindario  $L_2$ , se vuelve a leer las tuplas que la celda tiene asociadas para su comparación. Esta situación se puede observar en la figura 46. Al momento de procesar las 257 celdas blancas resultantes del mapeo de objetos, el programa requirió releer las tuplas correspondientes a 2651 celdas de las 8971 creadas. Si bien este proceso enlentece el proceso final del algoritmo, la cantidad de memoria ahorrada bien vale el esfuerzo teniendo en cuenta que toda la determinación de outliers se hizo leyendo tan solo una vez todo el banco de datos más una relectura aproximada del 30%. La última relectura se aloja completamente en memoria pero podría agregarse un proceso de liberación de memoria que para esta versión de la implementación del algoritmo no fue llevada a cabo.

Esta versión permitiría aplicar el algoritmo **FindAllOutsM** prácticamente en cualquier escenario sin la restricción de los 4GB de memoria que dispone un sistema operativo de 32 bits y su rendimiento estaría estrechamente asociado al banco de datos utilizado, mientras más celdas hubiese que releer, más tiempo se requerirá para completar el proceso.

## 9.6. DOS IMPLEMENTACIONES, UN ALGORITMO

Una vez implementadas todas las mejoras surgen dos versiones del programa posibles:

1. Ejecución completa en memoria
2. Ejecución híbrida: memoria y almacenamiento alternativo (disco)

Cada versión tiene distintas virtudes y limitantes

Versión de procesamiento completo en memoria

- Eficacia del 100% para la detección de outliers basados en distancia y mayor rendimiento (comparado con la versión híbrida)



- Limitada por la cantidad de memoria RAM disponible en el equipo que la ejecuta
- Recomendada para equipos de 64 bits

Versión de procesamiento híbrido: memoria + disco

- Puede ser utilizado tanto en equipos de 32 o 64 bits
- Menor rendimiento (comparado con la versión de proceso completo en memoria), misma eficacia del 100% para la detección de outliers basados en distancia
- Limitado por la cantidad de disco disponible (esta limitación conlleva mucho menor riesgo que el límite de memoria RAM disponible)

Ambas versiones contienen las dos primeras mejoras introducidas: hipercubo virtual para el manejo de celdas y celdas rosadas virtuales.

Una vez identificadas ambas versiones se corrieron distintos experimentos para distintas dimensiones y distintos tamaños del banco de datos como se había estipulado en la sección 9.1 y se establecía en la tabla 2.

Los resultados obtenidos se pueden visualizar en la tabla 4. En todos los casos se ajustó el valor de  $p$  para maximizar las posibilidades de obtener celdas rojas.

| N          | 3D      |         | 4D      |         | 5D      |         | 6D       |            |
|------------|---------|---------|---------|---------|---------|---------|----------|------------|
|            | Memoria | Híbrido | Memoria | Híbrido | Memoria | Híbrido | Memoria  | Híbrido    |
| 100.000    | 1s      | 44s     | 3s      | 9s      | 12s     | 1m 10s  | 5m 13s   | 1h 35m 28s |
| 500.000    | 8s      | 53s     | 9s      | 1m 5s   | 24s     | 1m 22s  | 55m 56s  | >100.000   |
| 1.000.000  | 19s     | 1m 52s  | 20s     | 2m 30s  | 36s     | 2m 49s  | >500.000 | >500.000   |
| 2.500.000  | 1m 7s   | 4m 54s  | 1m 9s   | 6m 16s  | 1m 27s  | 7m 54s  | 2m 37s   | 15m 56s    |
| 5.000.000  | 3m 8s   | 11m 36s | 3m 10s  | 15m 12s | 3m 40s  | 16m 23s | 5m 30s   | 25m 54s    |
| 10.000.000 | 10m 13s | 24m 41s | 10m 16s | 37m 38s | 10m 52s | 43m 58s | 14m 2s   | 1h 0m 38s  |

Tabla 4 – Resultados de los experimentos ejecutados para dos versiones de **FindAllOutsM**

En la figura 47 puede observarse como crece el tiempo de procesamiento a medida que el número de tuplas se incrementa en el banco de datos procesado. En la sección 8.1 se analizó el algoritmo **FindAllOutsM** y su análisis de complejidad. La complejidad del algoritmo es  $O(c^k + N)$  donde  $c$  es alguna constante dependiente de

$\sqrt{k}$  y  $m^{1/k}$  (que corresponde aproximadamente al número de celdas a lo largo de cada dimensión). Si bien la complejidad  $O(c^k + N)$  corresponde al peor escenario posible, el rendimiento del algoritmo se explica mejor si se tiene en cuenta la posibilidad de crear celdas rojas y rosadas para el valor de  $p, D$  y  $N$ . Es de esperar que cuanto mayor sea el banco de datos y la dimensionalidad, mayor sea probabilidad de requerir más celdas para contener todos los objetos presentes.

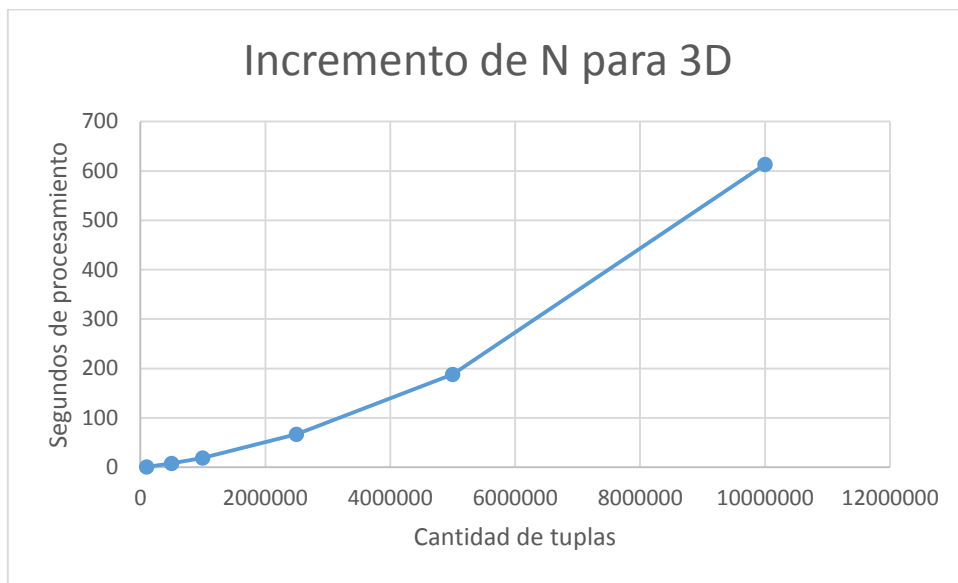


Figura 47 – Incremento de complejidad para 3D e incremento de N

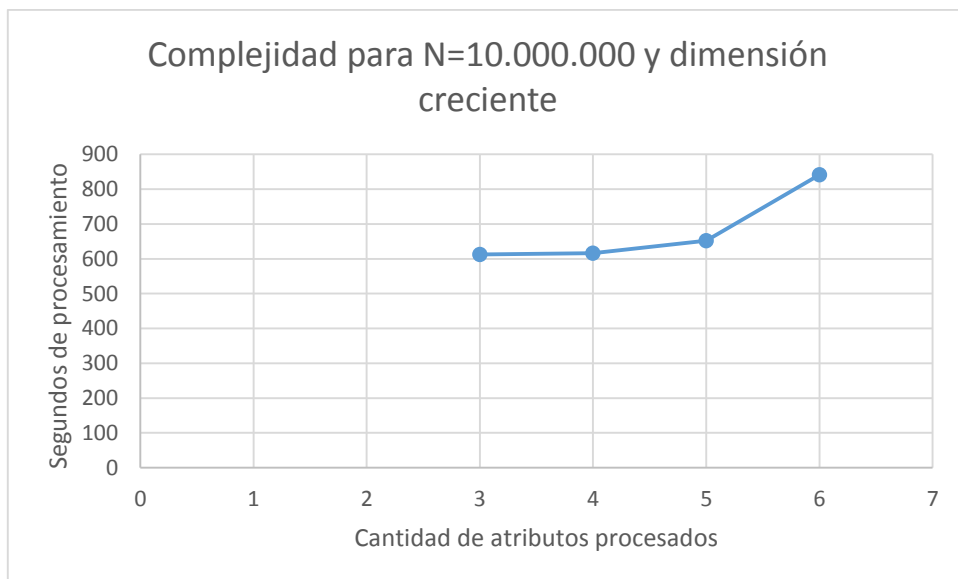


Figura 48 – N=10.000.000 e incremento de D

En la figura 48 puede observarse cómo se incrementa de manera notoria el tiempo requerido para encontrar outliers de 5 a 6 dimensiones. Esto se produce por el crecimiento exponencial que tienen las celdas requeridas para ejecutar el algoritmo, cuanto mayor es la cantidad de celdas requeridas, mayor es la cantidad de celdas blancas resultantes al final del proceso y mayor es el tiempo necesario para obtener un resultado.

En la columna 6D de la tabla 4 puede observarse cómo, a medida que se incrementa el valor de  $N$ , el algoritmo mejora su rendimiento para dimensiones grandes lo que hace de **FindAllOutsM** un algoritmo a tener en cuenta para grandes bases de datos.

En la tabla 4, para 10 millones de registros y 6D el casillero fue marcado con un color más oscuro. Esto se hizo así para llamar la atención sobre el consumo de memoria. Cuando se ejecutó el experimento, el programa llegó a consumir 5,3GB de memoria para 10 millones de registros, si bien el equipo donde se llevó a cabo el experimento posee 16GB de memoria, claramente la cantidad de memoria instalada es un limitante importante para el éxito de los experimentos.

## 9.7. FINDALLOUTSM PARALELO

En la sección 9 se estableció que la eficiencia del algoritmo se evalúa según el tiempo requerido para detectar todos los outliers de un banco de datos y, en caso insumir el mismo tiempo dos algoritmos distintos, el consumo de memoria de cada programa es la medida para determinar qué algoritmo era más eficiente.

Esta forma de medir la eficiencia fue seleccionada porque es comparable con la medida adoptada en el trabajo original [Knorr y otros, 1998].

Si como medida de eficiencia de un algoritmo se tiene en cuenta qué tan bien aprovecha todos los recursos del equipo en el que se ejecuta, se introduce una nueva oportunidad de mejora. En esta oportunidad se optó por experimentar con la capacidad de procesamiento que brinda un equipo con varios procesadores.

En la figura 49 puede observarse el uso típico de CPU que hace la implementación del algoritmo **FindAllOutsM** en la versión que procesa celdas y objetos en memoria.

Como puede observarse el uso de los recursos de la CPU son poco aprovechados ya que se utiliza tan sólo 1 procesador de los 8 disponibles en el equipo de pruebas. Para mejorar la utilización de las capacidades del equipo donde se ejecuta el algoritmo se optó por paralelizar la implementación del algoritmo **FindAllOutsM**. En el trabajo “*Parallel Algorithms for Distance-Based and Density-Based Outliers*” [Lozano y otros, 2005] se experimentó con la paralelización del algoritmo de Bay y el algoritmo LOF obteniendo mejoras en el rendimiento de cada algoritmo.

Una versión paralela del algoritmo **FindAllOutsM** sería similar a la versión no paralela, sólo se diferenciaría por el hecho de que soporta más de un hilo de ejecución concurrente, esto quiere decir, que es capaz de ejecutar simultáneamente múltiples secuencias de instrucciones. Cada secuencia de instrucciones tiene su propio control de flujo de ejecución que es independiente de los otros controles de flujo. Cada secuencia de instrucciones independiente se conoce como hilo de ejecución (o simplemente hilo).

En un ambiente donde se ejecutan varios hilos al mismo tiempo, si se accede al mismo recurso para leer y escribir, el valor accedido puede no ser el correcto. Por ejemplo, con dos hilos, uno que escribe y otro que lee el contenido de un archivo, si el hilo que escribe intenta escribir y, al mismo tiempo, el hilo que lee, intenta leer, los datos del archivo pueden corromperse. Por este motivo los hilos son mucho más efectivos si cada hilo tiene sus propios recursos con que trabajar y estos recursos son independientes del resto de los datos de los otros hilos. Utilizar recursos independientes para cada hilo es la aproximación utilizada en el trabajo “*Parallel Algorithms for Distance-Based and Density-Based Outliers*” [Lozano y otros, 2005]. Si el uso de recursos compartidos fuera mandatorio, los hilos deben sincronizarse.



Figura 49 –Uso de CPU típico para **FindAllOutsM** (versión memoria)

El algoritmo **FindAllOutsM** tiene dos etapas bien diferenciadas que pueden verse beneficiadas por la paralelización. La primera etapa que se puede agrupar bajo el nombre de “Carga” que involucra los pasos 2, 3 y 4 del algoritmo y la segunda etapa que se compone por todo el paso 5 del algoritmo que es la “determinación fina” de outliers.

Entre la “carga” y la “determinación fina” hay una diferencia relevante y es que la carga debe leer y actualizar una zona común de datos (el hipercubo de celdas) mientras que la determinación fina accede al mismo hipercubo pero sólo en modo lectura.

La estrategia utilizada para paralelizar la carga de objetos fue dividir el banco de datos en lotes que cada hilo podría procesar individualmente mientras que la estrategia utilizada para paralelizar la determinación fina de outliers fue que cada hilo tomara una celda blanca y procesara la celda junto con su entorno para determinar si todos los objetos de la celda son outliers o no. Cada hilo de ejecución

solicita que se le entregue una celda blanca que procesar. El hilo termina cuando el proceso que administra todos los hilos no tiene más celdas que entregar.

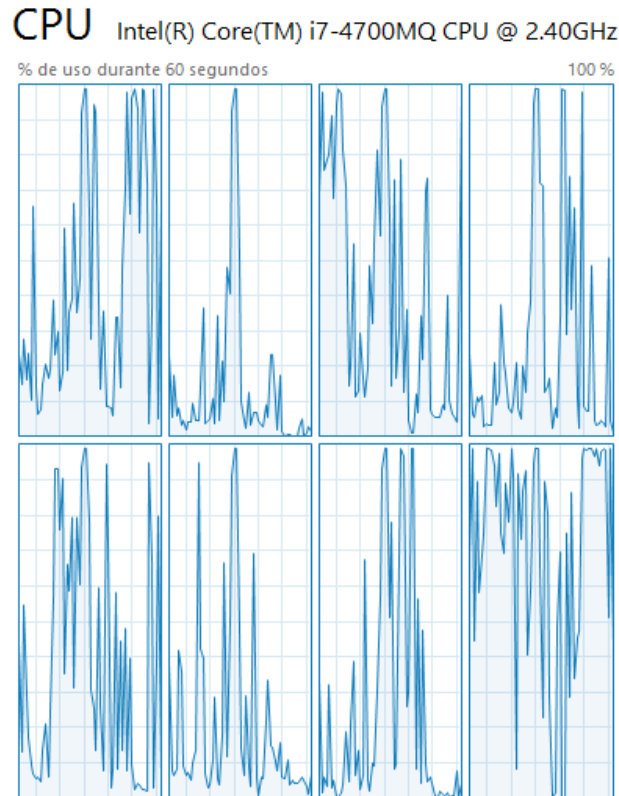


Figura 50 –Uso de CPU para **FindAllOutsM** (paralelo) en la etapa de mapeo

El objetivo al paralelizar el algoritmo es reducir un 25% el tiempo de ejecución del algoritmo al menos en el 70% de los casos. Para lograr el objetivo se creó versiones paralelas de las dos implementaciones del algoritmo presentadas en este trabajo final: la que procesa todo en memoria y aquella con una solución mixta.

Los experimentos arrojaron como resultado que hay una mejora en la eficiencia de los algoritmos manteniéndose la eficacia de detección del 100% de los outliers del banco de datos teniendo en cuenta los valores de  $p$  y  $D$ .

Analizando el rendimiento de las modificaciones introducidas por etapas pudo observarse que tanto la etapa de carga como la de determinación fina de outliers se benefician por la ejecución de varios hilos en paralelo.

En la figura 50 y 51 puede observarse cómo hay mayor actividad por el uso de todos los procesadores cuando el algoritmo **FindAllOutsM** (versión paralela) se ejecuta tanto en la etapa de mapeo celda/objeto o en la determinación fina de outliers. Al momento de ejecutarse estos experimentos, el programa que implementa el algoritmo **FindAllOutsM** era el único ejecutándose.

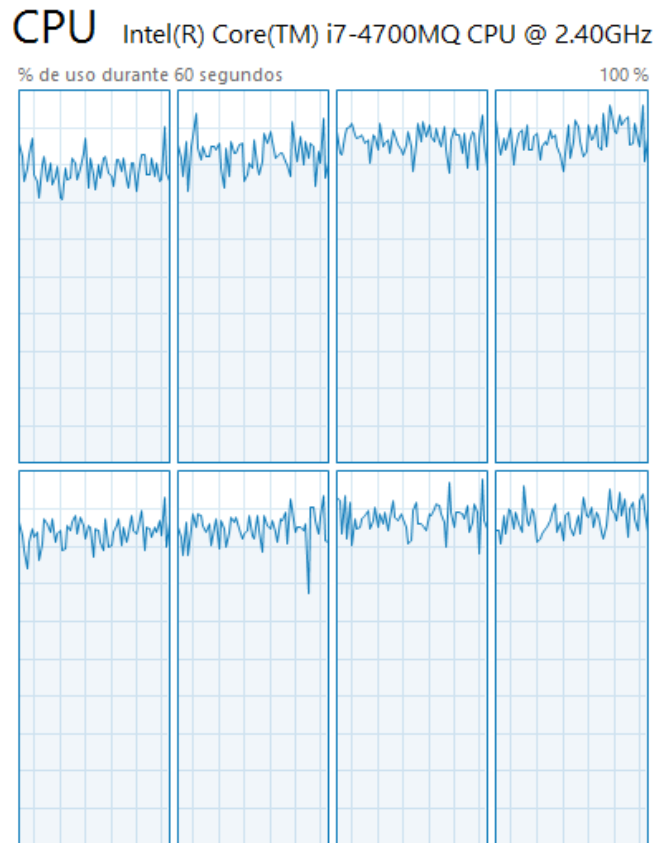


Figura 51 –Uso de CPU para **FindAllOutsM** (paralelo) en la etapa de determinación de outliers

Es esperable que mayor actividad de procesador pudiera traducirse en un mejor rendimiento de los programas creados. Para averiguar si se cumple el objetivo de disminución del 25% del tiempo de ejecución se diseñó la tabla 5 que debe ser completada con los resultados de los experimentos.

| Versión todo en memoria para $p=XX$ y $D=XX$ |        |          |        |          |        |          |        |          |
|--|--------|----------|--------|----------|--------|----------|--------|----------|
| N  | 3D     |          | 4D     |          | 5D     |          | 6D     |          |
|  | Normal | Paralela | Normal | Paralela | Normal | Paralela | Normal | Paralela |
| 2.500.000                                    |        |          |        |          |        |          |        |          |
| 5.000.000                                    |        |          |        |          |        |          |        |          |
| 10.000.000                                   |        |          |        |          |        |          |        |          |

 Tabla 5 – Template para experimentar con **FindAllOutsM** paralelo

Para determinar si hay una mejora en el rendimiento del algoritmo **FindAllOutsM** por el uso de varios hilos de ejecución se ejecutaron experimentos que hicieran visibles las mejoras teóricas que introducían las modificaciones realizadas. Para esto se experimentó con bancos de datos de 2.500.000, 5.000.000 y 10.000.000 de tuplas y con 3, 4, 5 y 6 atributos.

Los resultados de la ejecución del algoritmo para  $p = 0,9999$  y  $D = 29,6985$  pueden observarse en las tablas 5 y 6.

| Versión todo en memoria para $p=0,9999$ y $D=29.6985$ |          |          |          |          |          |          |
|---|----------|----------|----------|----------|----------|----------|
| N   | 3D       |          | 4D       |          | 5D       |          |
|   | Original | Paralela | Original | Paralela | Original | Paralela |
| 2.500.000   | 1m 28s   | 1m 4s    | 1m 31s   | 1m 5s    | 1m 46s   | 1m 13s   |
| 5.000.000   | 4m 41s   | 2m 33s   | 4m 27s   | 2m 21s   | 4m 56s   | 2m 38s   |
| 10.000.000  | 14m 13s  | 5m 31s   | 14m 38s  | 5m 10s   | 15m 28s  | 5m 42s   |

 Tabla 5 – Resultados de los experimentos ejecutados **FindAllOutsM** en memoria original y paralelo

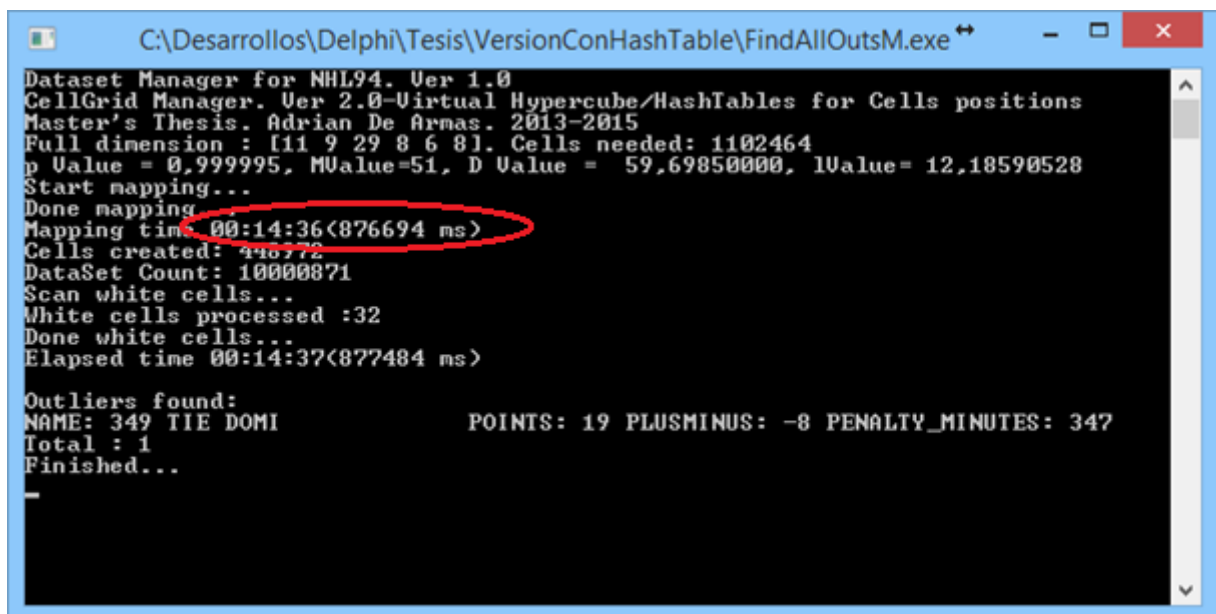
| Versión híbrida para $p=0,9999$ y $D=29.6985$ |           |          |           |          |            |          |
|---|-----------|----------|-----------|----------|------------|----------|
| N   | 3D        |          | 4D        |          | 5D         |          |
|   | Original  | Paralela | Original  | Paralela | Original   | Paralela |
| 2.500.000                                     | 34m 54s   | 13m 5s   | 27m 22s   | 10m 43s  | 11m 48s    | 8m 45s   |
| 5.000.000                                     | 47m 9s    | 20m 39s  | 38m 59s   | 16m 48s  | 31m 32s    | 13m 48s  |
| 10.000.000                                    | 1h 49m 4s | 30m 6s   | 1h 6m 47s | 34m 40   | 1h 17m 35s | 31m 45s  |

 Tabla 6 – Resultados de los experimentos ejecutados **FindAllOutsM** híbrido original y paralelo

Para 6 atributos los parámetros utilizados en los experimentos anteriores no permitían desarrollar el potencial del algoritmo **FindAllOutsM** porque del análisis del experimento se pudo observar que se habían generado, producto del mapeo celda/objetos, todas celdas blancas, ninguna rosa ni roja. Esto se explica por la cantidad de atributos a procesar que influye en la formación del hipercubo con más de 60 millones de celdas, donde se tienen que alojar todas las tuplas. Como



resultado se obtienen celdas blancas con objetos cuya etiqueta de “outlier” (o no) será determinada por un proceso de comparación objeto a objeto, es decir, el algoritmo **FindAllOutsM** se comporta como una determinación de outliers por fuerza bruta más allá que la comprobación se restringe a sus vecinos  $L_2$ . Se decidió no repetir el experimento con los mismos parámetros para poder obtener resultados relevantes.



```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\FindAllOutsM.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [11 9 29 8 6 8]. Cells needed: 1102464
p Value = 0,999995, MValue=51, D Value = 59,69850000, lValue= 12,18590528
Start mapping...
Done mapping...
Mapping time 00:14:36<876694 ms>
Cells created: 440772
DataSet Count: 10000871
Scan white cells...
White cells processed :32
Done white cells...
Elapsed time 00:14:37<877484 ms>

Outliers found:
NAME: 349 TIE DOMI POINTS: 19 PLUSMINUS: -8 PENALTY_MINUTES: 347
Total : 1
Finished...
    
```

Figura 51 –**FindAllOutsM** (original) para  $N=10.000.000$ ,  $p=0,999995$  y  $D=59,6985$

El rendimiento de **FindAllOutsM** se incrementa notablemente a medida que el algoritmo puede crear mayor cantidad de celdas rojas y rosadas (para lo cual es importante determinar correctamente los parámetros de ejecución), como para  $p = 0.9999, D = 29.6985$  con seis atributos, el rendimiento es malo, los parámetros de ejecución deben ser replanteados junto al usuario. Se modifican los parámetros de ejecución del experimento sólo para  $N = 10.000.000$  y *atributos* = 6. El resultado se puede observar en las figuras 51 y 52. Lo notable de este experimento es que permite visualizar cómo la etapa de mapeo puede verse beneficiada por la ejecución en paralelo.

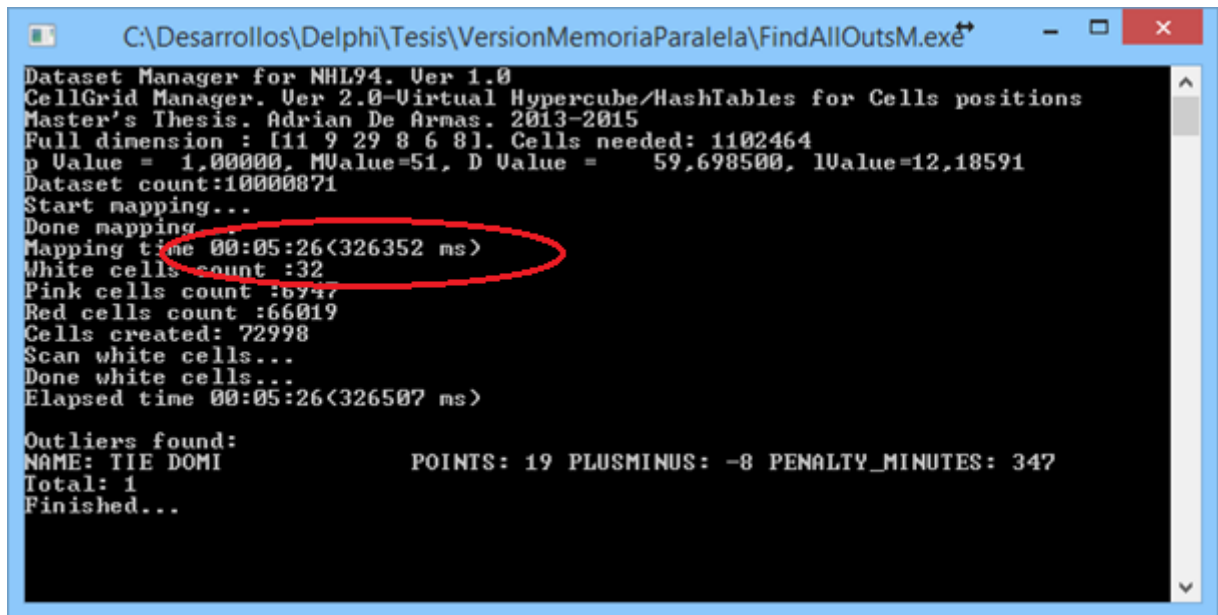


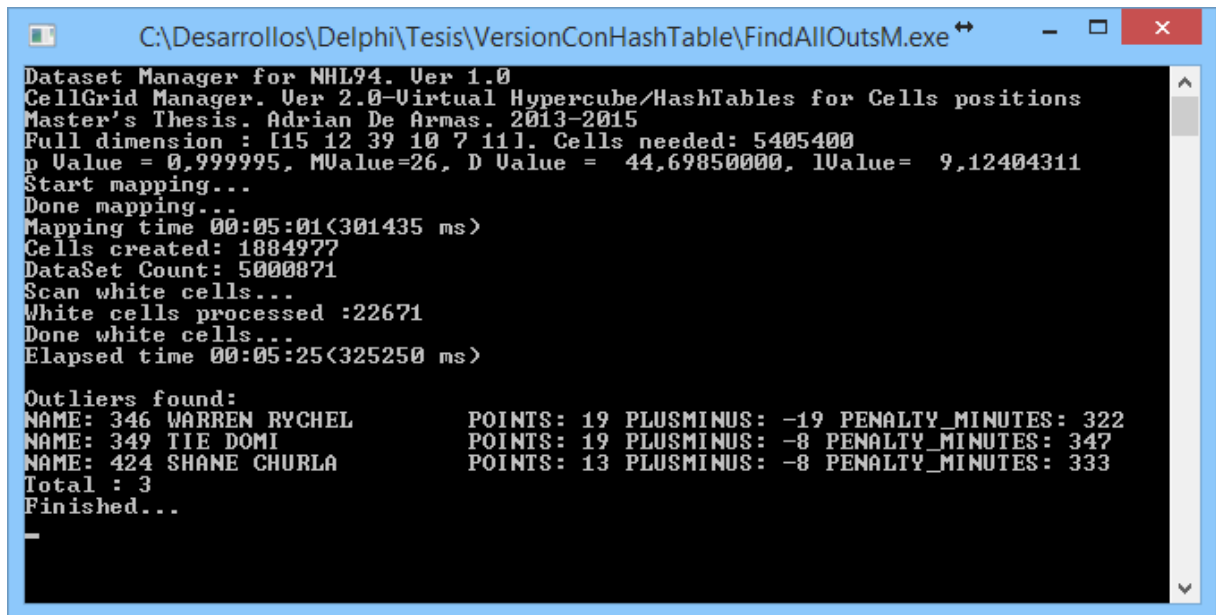
Figura 52 –FindAllOutsM (paralelo) para N=10.000.000, p=0,999995 y D=59,6985

La versión paralela pudo realizar el mapeo celda/objetos en  $\frac{1}{3}$  del tiempo que lo hizo la versión normal. Se repitieron los experimentos para 6 atributos pero anteriormente se establecieron nuevos parámetros para la ejecución ( $p = 0,999995$  y  $D = 44,6985$ ) y el resultado para  $N = 5000871$  puede observarse en las figuras 53 y 54. Un resumen de los experimentos realizados para 6 atributos puede observarse en la tabla 7 y en la figura 55 un gráfico de barra que resume la experiencia.

| p=0,999995 y D=44.6985 |            |          |            |            |          |            |
|------------------------|------------|----------|------------|------------|----------|------------|
| N                      | 6D memoria |          |            | 6D Híbrida |          |            |
|                        | Normal     | Paralela | Diferencia | Normal     | Paralela | Diferencia |
| 2.500.000              | 2m 26s     | 1m 51s   | 23,97%     | 13m 31s    | 9m 58s   | 26,26%     |
| 5.000.000              | 5m 25s     | 3m 4s    | 43,38%     | 23m 12s    | 15m 55s  | 31,39%     |
| 10.000.000             | 11m 15s    | 5m 37s   | 50,07%     | 42m 27s    | 27m 19s  | 35,65%     |

Tabla 7 – Resultados de los experimentos ejecutados para FindAllOutsM original y paralelo

Estos experimentos permiten demostrar la relevancia de seleccionar correctamente los parámetros de cada ejecución del algoritmo con el fin de aprovechar las ventajas que brinda FindAllOutsM como así también, para obtener resultados relevantes.

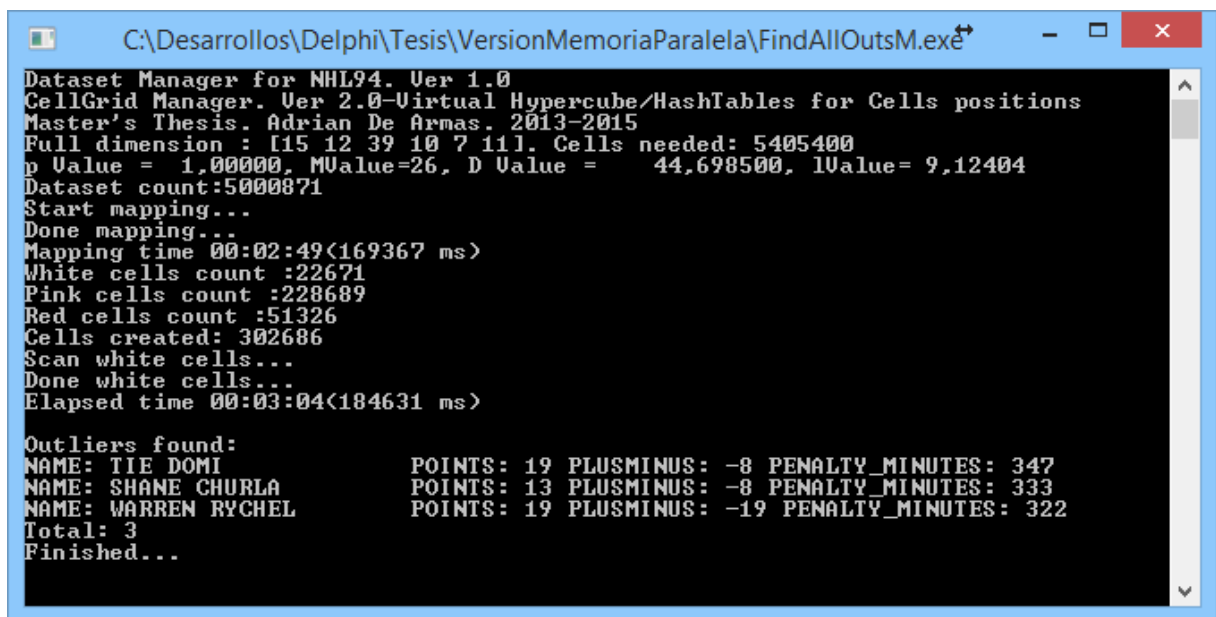


```

C:\Desarrollos\Delphi\Tesis\VersionConHashTable\FindAllOutsM.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [15 12 39 10 7 11]. Cells needed: 5405400
p Value = 0,999995, MValue=26, D Value = 44,69850000, lValue= 9,12404311
Start mapping...
Done mapping...
Mapping time 00:05:01(301435 ms)
Cells created: 1884977
DataSet Count: 5000871
Scan white cells...
White cells processed :22671
Done white cells...
Elapsed time 00:05:25(325250 ms)

Outliers found:
NAME: 346 WARREN RYCHEL          POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
NAME: 349 TIE DOMI              POINTS: 19 PLUSMINUS: -8  PENALTY_MINUTES: 347
NAME: 424 SHANE CHURLA         POINTS: 13 PLUSMINUS: -8  PENALTY_MINUTES: 333
Total : 3
Finished...
    
```

Figura 53 –FindAllOutsM (normal) para N=5.000.000, p=0,999995 y D=44,6985



```

C:\Desarrollos\Delphi\Tesis\VersionMemoriaParalela\FindAllOutsM.exe
Dataset Manager for NHL94. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [15 12 39 10 7 11]. Cells needed: 5405400
p Value = 1,000000, MValue=26, D Value = 44,6985000, lValue= 9,12404
DataSet count:5000871
Start mapping...
Done mapping...
Mapping time 00:02:49(169367 ms)
White cells count :22671
Pink cells count :228689
Red cells count :51326
Cells created: 302686
Scan white cells...
Done white cells...
Elapsed time 00:03:04(184631 ms)

Outliers found:
NAME: TIE DOMI                  POINTS: 19 PLUSMINUS: -8  PENALTY_MINUTES: 347
NAME: SHANE CHURLA             POINTS: 13 PLUSMINUS: -8  PENALTY_MINUTES: 333
NAME: WARREN RYCHEL           POINTS: 19 PLUSMINUS: -19 PENALTY_MINUTES: 322
Total: 3
Finished...
    
```

Figura 54 –FindAllOutsM (paralelo) para N=5.000.000, p=0,999995 y D=44,6985

En esta sección se buscó mejorar la eficiencia del algoritmo **FindAllOutsM** mediante la incorporación de la ejecución de varios hilos en paralelo. Los experimentos permitieron confirmar que el algoritmo puede verse beneficiado con el uso de varios procesadores a medida que la cantidad de tuplas crece.

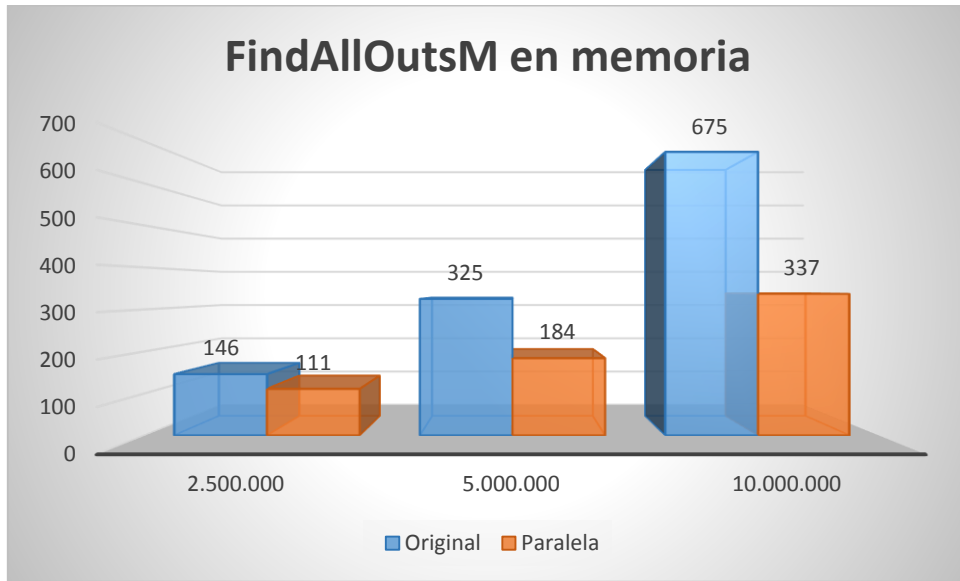


Figura 55 –FindAllOutsM para 6 atributos, p=0,999995 y D=44,6985

En la figura 56 puede observarse cómo el uso de varios procesadores pasa inadvertido para cantidades pequeñas de tuplas ya que el tiempo ganado por la ejecución en paralelo se ve compensado por la complejidad interna del programa dedicado a administrar los distintos hilos de ejecución. A medida que la cantidad de tuplas aumenta, los beneficios de paralelizar son más notorios ya que permite suavizar el incremento de tiempo requerido hasta arrojar resultados.

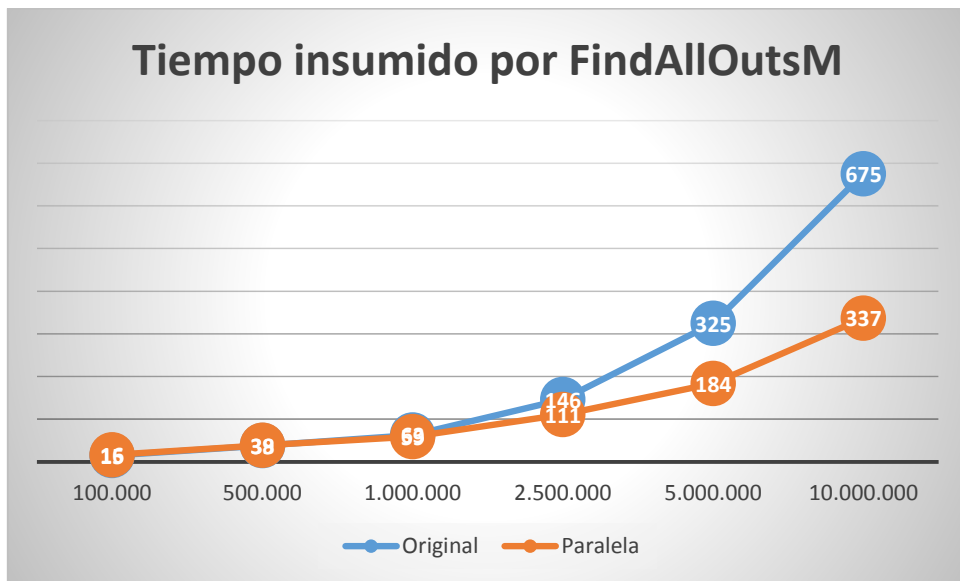


Figura 56 –FindAllOutsM para 6 atributos, p=0,999995 y D=44,6985

Los experimentos de esta sección permiten notar qué tan relevante es determinar con qué parámetros se ejecuta el programa, ya que de esto depende el aprovechamiento del algoritmo **FindAllOutsM**. La relevancia de los resultados obtenidos reposan en qué parámetros de  $p$  y  $D$  fueron propuestos.

Es posible que dado un banco de datos, la mayoría de los objetos se agrupen en pocas celdas. Si esto sucede, el algoritmo pasará mucho tiempo administrando el acceso a memoria compartida, lo cual puede ocasionar que los distintos hilos se dispongan en serie para agregar un objeto a una celda en particular perdiendo, de esta manera, el beneficio de paralelizar el proceso de carga.

Las versiones híbridas de la implementación del algoritmo pueden mejorar su rendimiento si realizan la carga de tuplas en paralelo como se puede observar con los experimentos documentados en la tabla 7. Se debe tener en cuenta que cuando la carga se realiza en paralelo cada hilo tiene un conjunto reducido de datos extraídos del banco que se esté procesando, esto implica que el proceso consumirá más memoria porque todos esos subconjuntos de datos están residentes al mismo tiempo. Tal incremento de memoria es incompatible con aquello que motivó la creación de las versiones híbridas que es mantener al algoritmo consumiendo poca memoria para asegurar la obtención de resultados independientemente de la cantidad de tuplas y atributos a procesar. Si se reduce el consumo de memoria haciendo que cada hilo procese pequeñas cantidades de tuplas, se consigue reducir la cantidad de memoria que se consume en un instante dado pero al mismo tiempo se afecta el rendimiento del proceso. Paralelizar la carga de datos, para las versiones híbridas, es posible pero no se recomienda.

La determinación fina de outliers, al acceder a la zona de memoria compartida en modo sólo lectura, no tiene el inconveniente que sí tiene la carga. Para decidir si era posible utilizar varios hilos en la carga (o no), en el ámbito de este trabajo final, se realizaron experimentos controlados en los cuales se utiliza un banco de datos de 2 millones de tuplas y se corre la carga con código paralelo y normal. La versión más rápida es la que se utiliza para el resto de los experimentos con ese banco de datos.

Teniendo en cuenta que la eficacia del algoritmo mejora a medida que crece la cantidad de tuplas a procesar como se puede observar en las tablas 5, 6 y 7, se da por cumplido el objetivo de la propuesta realizada en esta sección.

## 9.8. DETECCIÓN PROBABILÍSTICA DE OUTLIERS

Todas las modificaciones propuestas a la implementación del algoritmo **FindAllOutsM** a lo largo de este trabajo final fueron pensadas para mejorar la eficiencia del algoritmo manteniendo la característica de detectar el 100% de los outliers basados en distancia.

Existen situaciones dónde es más importante obtener un rápido resultado que una eficacia del 100%. En la sección 7 se han visto modelos para la detección de outliers que asignan cierto “grado de outlier” a cada tupla del banco de datos como LOF [Breuning y otros, 2000]. La determinación de qué elementos son considerados outliers queda en manos del usuario. **FindAllOutsM** determina de forma booleana si cada elemento del banco de datos es o no outlier y la determinación se hace con una eficacia del 100% respetando los valores de los parámetros  $p$  y  $D$ .

La versión híbrida del algoritmo **FindAllOutsM**, desarrollada para equipos con limitaciones de memoria tiene una alta penalidad de rendimiento respecto a la versión que procesa todas las tuplas en memoria aunque puede ser utilizada prácticamente en cualquier escenario. Si se puede sacrificar eficacia, el algoritmo puede ganar eficiencia realizando una determinación probabilística de outliers.

En la versión híbrida del algoritmo el proceso de determinación fina de outliers puede ser costosa (respecto al tiempo requerido para obtener resultados), a medida que, para analizar los objetos de una celda, se recuperan los vecinos  $L_2$  de la celda. La recuperación de los objetos asociados a una celda se realiza para poder comparar las distancias objeto a objeto con el fin de determinar la cantidad de elementos en el vecindario más cercano de un objeto en particular. Cada recuperación requiere el acceso a la memoria auxiliar (un archivo indexado implementado con SQLite en el ámbito de este trabajo final) para obtener todas aquellas claves que pertenecen a una celda y luego, por cada clave, se vuelve a

acceder al motor de base de datos con el fin de recuperar los datos asociados a los objetos a evaluar. Para estos casos se propone una determinación probabilística de outliers. La determinación probabilística de outliers se basa en obtener probabilísticamente cuántos elementos aporta una celda en particular al vecindario más cercano de un objeto cualquiera. Como prueba de concepto la probabilidad se calcula con una simple regla de tres simple y se asume que los objetos asociados a una celda se encuentran uniformemente distribuidos.

Para realizar esta modificación sí hay que modificar el paso 5<sub>b</sub> del algoritmo de la siguiente manera: Por cada celda contra la cual hay que calcular la distancia objeto a objeto, se calcula la distancia máxima y mínima que puede tener el objeto a analizar contra cualquier objeto perteneciente a esa celda, luego, utilizando la regla de tres simple, el porcentaje de objetos que esa celda puede aportar de vecinos más cercanos queda expresado de la siguiente manera:

$$\text{porcentaje} = (D - \text{DistMin}) / (\text{DistMax} - \text{DistMin})$$

Donde  $D$  es el valor de la distancia obtenido como parámetro para la ejecución del algoritmo. Una vez calculado ese porcentaje, la cantidad de elementos que aporta la celda al objeto se calcula con  $\text{vecinos} = \text{Cantidad objetos} * \text{porcentaje}$ , siendo  $\text{Cantidad objetos}$  los objetos asociados a la celda  $L_2$  procesada.

Esta forma de procesar el vecindario más cercano de un objeto reduce la eficacia del algoritmo al brindar la posibilidad de detectar falsos positivos o no detectar outliers reales. Por este motivo, el objetivo de esta modificación es lograr una reducción de tiempo (en porcentaje) siempre mayor al porcentaje de errores introducidos por el algoritmo (no encontrados + falsos positivos).

De esta manera el paso 5 del algoritmo **FindAllOutsM** debe ser modificado a

```

for celBlanca in celdas.Blancas
  if celBlanca.Items.Count > 0 then
  begin
    cantObjetos := celBlanca.Items.Count;
    for celAux1 in celBlanca.vecindarioL1 do
      cantObjetos := cantObjetos + celAux1.Items.Count;
    if cantObjetos > umbral then
      celBlanca.Color := Rosado
    else
    begin
      cantObjetos2 := cantObjetos;
      for celAux2 in celBlanca.vecindarioL2 do
        cantObjetos2 := cantObjetos2 + celAux2.Items.Count;
      if cantObjetos2 <= umbral then
        celBlanca.MarcarTodosComoOutliers
      else
      begin
        for obj in celBlanca.Items do
        begin
          cantObjetosP := cantObjetos;
          for celAux2 in celBlanca.vecindarioL2 do
          begin
            distMin := CalcularDistanciaMinima(obj, celAux2);
            distMax := CalcularDistanciaMaxima(obj, celAux2);
            porcentaje=(distancia-distMin)/(distMax-distMin);
            cantObjetosP := celAux2.Items.Count * porcentaje;
          end
          if cantObjetosP <= umbral then
            obj.MarcarComoOutlier;
          end
        end
      end
    end
  end
end
end
end
end
    
```

La modificación sólo afecta al proceso de determinación fina de outliers. El costo de mapear cada objeto a una celda es un tiempo compartido por ambos algoritmos (el que tiene la modificación y la referencia que no lo tiene) y no se tiene en cuenta para definir si el objetivo establecido fue logrado o no. El tiempo de mapeo celda/objeto será eliminado del registro de los experimentos.

Se toma como versión de referencia la versión híbrida paralela.



Para maximizar la posibilidad de obtener resultados relevantes esta técnica sólo es utilizada cada vez que la celda  $L_2$  a analizar sea una celda roja por dos motivos:

1. Cuánto más objetos tenga asociado la celda roja, más tiempo se requiere para recuperar los datos de los objetos asociados
2. Cuánto más elementos tenga la celda, mayor probabilidad de obtener resultados que se asemejen a la realidad (los objetos se distribuyen uniformemente en la celda)

En la tabla 8 y 9 se puede observar el resultado de los experimentos. Se dividieron los experimentos para utilizar distintos parámetros según la cantidad de atributos.

| p=0,9999 d=29.6985 |         |                |          |    |    |         |                |          |    |    |
|--------------------|---------|----------------|----------|----|----|---------|----------------|----------|----|----|
| N                  | 3D      |                |          |    |    | 4D      |                |          |    |    |
|                    | Híbrida | Probabilística | OUTLIERS | FP | NE | Híbrida | Probabilística | OUTLIERS | FP | NF |
| 10.000.000         | 34m 24s | 3m 2s          | 216      | 71 | 0  | 27m 55s | 10m 37s        | 230      | 44 | 0  |

Tabla 8 – Resultados de los experimentos ejecutados para **FindAllOutsM** híbrida y probabilística

| p=0,999995 d=44.6985 |         |                |          |    |    |         |                |          |    |    |
|----------------------|---------|----------------|----------|----|----|---------|----------------|----------|----|----|
| N                    | 5D      |                |          |    |    | 6D      |                |          |    |    |
|                      | Híbrida | Probabilística | OUTLIERS | FP | NE | Híbrida | Probabilística | OUTLIERS | FP | NF |
| 2.500.000            | 3M 23S  | 25S            | 3        | 0  | 0  | 3m 33s  | 23s            | 3        | 0  | 0  |
| 5.000.000            | 8m 51s  | 42s            | 3        | 8  | 0  | 3m 17s  | 2m 45s         | 3        | 0  | 0  |
| 10.000.000           | 20m 52s | 2m 41s         | 3        | 15 | 0  | 5m 55   | 5m             | 3        | 0  | 0  |

Tabla 9 – Resultados de los experimentos ejecutados para **FindAllOutsM** híbrida y probabilística

Las columnas FP y NE se refieren a los falsos positivos detectados y los outliers no encontrados respectivamente.

Con el análisis del resultado de los experimentos puede concluirse el cumplimiento del objetivo planteado. El porcentaje de mejora en tiempo obtenido por el algoritmo **FindAllOutsM** (en detrimento de la eficacia del mismo) siempre ha superado el porcentaje de falsos positivos detectados. Durante todos los experimentos nunca se registró una situación en donde el algoritmo no detectase algún outlier real (aunque es una posibilidad cierta), la única anomalía registrada fue la detección de falsos positivos como se puede observar en la figura 57.

En los casos en que la diferencia de tiempo no es relevante, como por ejemplo en la tabla 9 el caso  $N = 10.000.000$  o  $N = 5.000.000$  y 6 atributos, el algoritmo no arrojó

falsos positivos. Esto es así porque toda (o casi toda) la determinación fina de outliers se hizo analizando celdas blancas y/o rosas.

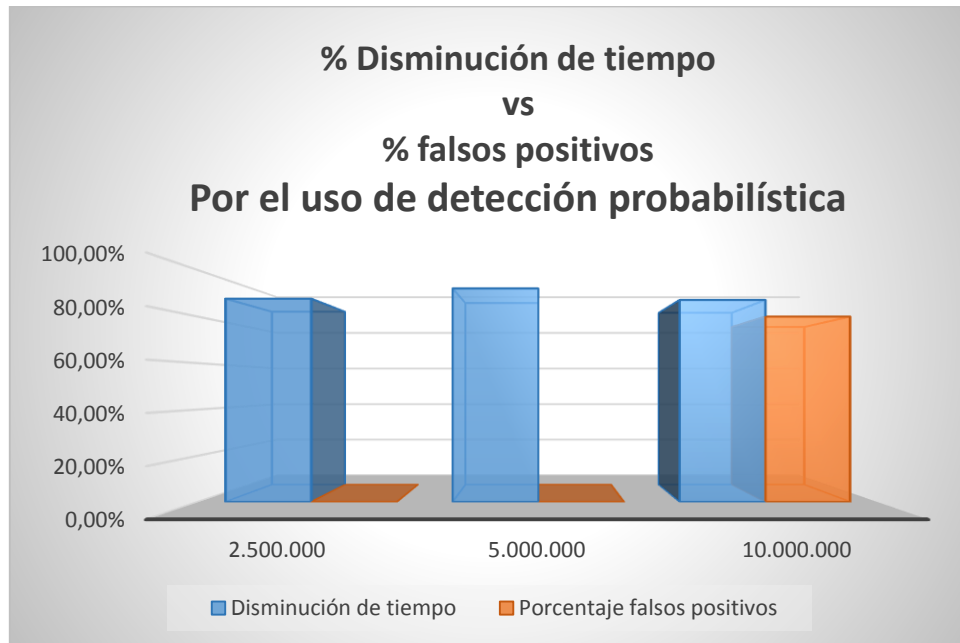


Figura 57 –FindAllOutsM para 5 atributos,  $p=0,999995$ ,  $D=44,6985$ ,  $N=10.000.000$

Si bien esta modificación se introdujo sólo en la versión híbrida paralela, el concepto puede utilizarse en cualquiera de las versiones presentadas durante el desarrollo del trabajo. El tiempo de mapeo es un tema a tener muy en cuenta porque si se toma el tiempo de procesamiento total, puede resultar poco relevante introducir la modificación si el beneficio total es poco visible, sobretodo, en versiones ejecutadas totalmente en memoria. Un ejemplo dónde sí es relevante la modificación es para  $N = 5.000.000$  y 5 atributos de la tabla 9, ambas ejecuciones comparten un proceso de carga que tardó entre 10m 30s y 11m, el tiempo requerido hasta obtener resultados es del doble para aquella versión que no tiene la modificación.

Las versiones basadas en el cálculo de outliers completamente en memoria se ven beneficiadas por la determinación probabilística porque como los objetos de las celdas rojas no son tenidos en cuenta, estos objetos pueden ser liberados de memoria permitiendo alojar más celdas y objetos lo que extiende su capacidad de procesamiento a más tuplas y atributos.

## 9.9. RESUMEN DEL DESARROLLO DEL TRABAJO

El algoritmo **FindAllOutsM** propuesto por Edwin Knorr en el año 1998 [Knorr y otros, 1998] fue un algoritmo novedoso y muy simple de comprender. Destacaba porque su implementación se independizaba del conocimiento de la distribución subyacente de los datos.

En el trabajo original se propone el uso de esta técnica para 4 dimensiones (cantidad de atributos utilizados) o, a lo sumo, 5. En este trabajo final se partió de la simplificación del algoritmo y se lo implementó. Mediante experimentos se pudo comprobar sus limitaciones y, sin modificar el algoritmo, se introdujo mejoras a las distintas implementaciones realizadas con la idea de extender la cantidad de escenarios en los que se puede usar **FindAllOutsM**

Las mejoras introducidas son las siguientes:

1. Virtualización del hipercubo propuesto por el algoritmo
2. Virtualización de celdas de vecindarios  $L_1$
3. Persistencia de la relación objeto/celda en un almacenamiento secundario para disponer de más memoria
4. Mecanismo para recuperar los objetos asociados a una celda a voluntad
5. Ajustes varios para mejorar el rendimiento del programa
6. Compilación del programa para 64 bits para mejorar las posibilidades de alojar celdas en memoria

Estas mejoras permiten utilizar el algoritmo en una variedad mucho más amplia de escenarios. Mediante experimentación se pudo probar el algoritmo con muchas dimensiones (hasta 11) y con una gran cantidad de tuplas (hasta 10 millones).

El algoritmo mostró muy buen rendimiento para bancos de datos pequeños (hasta 5 mil tuplas) y cualquier cantidad de atributos o con prácticamente cualquier cantidad de datos (se probó con 10 millones de filas) y hasta 6 atributos para superar las limitaciones detectadas en el trabajo original.

Aumentando la cantidad de atributos a evaluar (por ejemplo 8, 9, 10) y manteniendo una cantidad relativamente discreta de tuplas (1, 2, 3 o 4 millones) es altamente

probable que haya una proliferación de celdas blancas, donde cada celda contiene una sola tupla asociada, ocasionando que el rendimiento del algoritmo decaiga considerablemente. El rendimiento del algoritmo depende fuertemente de poder crear muchas celdas rosas y rojas para descartar como outliers sin ningún análisis posterior, a todos los objetos alojados en estas celdas.

Mediante la etapa de experimentación se procesó un banco de datos con 100.000 tuplas y se evaluaron 9 atributos. El programa tuvo que crear poco más de 97000 celdas, casi 1 por cada tupla, sin llegar a colorear como roja a ninguna. En este contexto, el proceso final del algoritmo se comportó casi como una determinación de outliers por fuerza bruta.

Muchos atributos (entre 6 y 10) y una cantidad media de tuplas a analizar (entre 3.5 y 6 millones) es una zona “gris” donde no se recomienda el uso de este algoritmo pero si la cantidad de datos crece (10, 20, 30, 50 millones de tuplas) y con los parámetros adecuados, **FindAllOutsM** vuelve a verse beneficiado ya que con mayor cantidad de tuplas y parámetros adecuados (el valor de  $p$  debe establecerse muy cercano a la unidad) habrá una proliferación de celdas rojas lo cual mejorará mucho el rendimiento de la detección de outliers. Por este motivo la aproximación por celdas es un excelente algoritmo a aplicar para la búsqueda de outliers en grandes bases de datos, con una gran cantidad de tuplas, ya que un mayor valor de  $N$  implica mayor posibilidad de obtener celdas rojas.

La utilización de un hiper cubo virtual para resolver el mapeo de celdas y objetos hace innecesaria la determinación inicial de la dimensión del hiper cubo. En términos generales esto permite ahorrar una lectura completa del banco de datos.

Se ha presentado un algoritmo implementado de tal forma que supera las limitaciones previamente establecidas capaz de, dados dos parámetros  $p$  y  $D$ , detectar eficazmente el 100% de aquellos objetos que, para los algoritmos basados en distancia, serían considerados outliers. El algoritmo tiene un gran rendimiento para poca cantidad de tuplas y un rendimiento creciente a medida que la cantidad de tuplas crece lo que lo hace una excelente alternativa para procesar grandes bases de datos.

El algoritmo fue implementado de forma genérica para cualquier dimensión, el mismo código base ajusta su comportamiento dependiendo de la dimensionalidad del banco de datos asociado al proceso de detección de outliers. El trabajo original se realizó creando versiones específicas para cada una de las distintas dimensiones con las que se experimentó.

Se experimentó con dos versiones del mismo algoritmo, una basada en el procesamiento de todas las tuplas en memoria y otra con mecanismos para la utilización de memorias alternativas para la persistencia de la relación celda/objeto y recuperación de tuplas con el objetivo de maximizar la cantidad de celdas en memoria.

La versión que procesa todo el banco de datos en memoria tiene un rendimiento muy superior que el de su par híbrida (hasta un 500%) pero tiene el limitante de la cantidad de memoria instalada en el equipo donde se ejecuta el programa. La gran diferencia entre ambas implementaciones encuentra su explicación en la velocidad de los soportes que se utilizan para realizar los cálculos. Para bancos de datos grandes (millones de tuplas), es mandatorio que el algoritmo sea compilado para 64 bits para mejorar la posibilidad de poder brindar un resultado.

La versión híbrida tiene un buen rendimiento y permite obtener resultados en buenos tiempos como se puede observar a lo largo de todos los experimentos documentados. Esta versión puede compilarse tanto para 32 bits o 64 y obtener resultados para cualquier cantidad de tuplas en bancos de datos multidimensionales. A diferencia de la versión del algoritmo que utiliza sólo memoria, esta versión utiliza como soporte alternativo el disco de la PC donde se utiliza el programa permitiéndole instanciar y procesar mayor cantidad de celdas.

Se modificaron las implementaciones del algoritmo para que manejen varios hilos simultáneos de ejecución para aumentar la efectividad del programa. Todas las versiones (memoria e híbrida) se vieron beneficiadas por esta técnica.

Si un 100% de eficacia no es mandatorio para el dominio de donde se extraen los datos, se introdujo la posibilidad de determinar los outliers de banco de datos de forma probabilística lo que permitió mejoras sustanciales (hasta un 80%) en la

eficiencia del algoritmo y con un costo aceptable (según lo establecido como objetivo) de falsos positivos.

La elección de una u otra versión dependerá del dominio donde se vaya a utilizar el programa. Se debe tener en cuenta la cantidad de dimensiones a procesar, el tamaño del banco de datos, la arquitectura de la PC donde se ejecuta el programa (32 o 64 bits), cantidad de memoria disponible, velocidad de disco y el nivel de eficacia esperado para saber qué versión se ajusta a mejor a cada situación particular.

## 10. UTILIZACION DE FINDOUTSM EN BANCO DE DATOS REAL

Una vez superados los distintos experimentos a los que se sometió el algoritmo y sus distintas implementaciones con una base de datos de laboratorio se debe comprobar su correcto funcionamiento en un banco de datos real.

El banco de datos seleccionado consiste en detalles de arribos y partidas para todos los vuelos comerciales dentro de USA desde octubre de 1987 hasta abril del 2008 [BTS, 2013]. El banco de datos consta de aproximadamente 120 millones de registros en total que ocupan 1.6GB de espacio en estado comprimido y aproximadamente 12 GB una vez descomprimidos.

La información proviene originalmente de la Oficina de Estadísticas de Transporte del Departamento de Transporte de los Estados Unidos donde se encuentran los archivos para su descarga y los campos que compone cada archivo están debidamente explicados [BTS, 2013].

En la tabla 10 puede observarse el detalle de los campos que componen el banco de datos.

El interés de este trabajo final es detectar outliers analizando datos numéricos. Los atributos que cumplen con esta característica y fueron usados para el experimento son ArrDelay, DepDelay, CarrierDelay, WeatherDelay, NASDelay, SecurityDelay. El atributo LateAircraftDelay fue descartado porque del vuelco de datos se pudo observar que el campo siempre tenía “NA” (Not available – No disponible) lo cual no lo hacía relevante para analizar.

Todos los atributos hacen referencia a distintos tipos de retrasos que puede afectar la puntualidad de un vuelo. El banco de datos se compone por los datos de todos los vuelos de cabotaje realizados en EE UU en los años 1998 hasta el 2003 inclusive brindando más de 34 millones de tuplas.

| Campo             | Descripción  |
|-------------------|--|
| Year              | Año del vuelo  |
| Month             | Mes del vuelo  |
| DayofMonth        | Día del vuelo  |
| DayOfWeek         | Día de la semana   |
| DepTime           | Horario real de partida                                    |
| CRSDepTime        | Horario de partida esperado                                |
| ArrTime           | Horario real de llegada                                    |
| CRSArrTime        | Horario de llegada esperado                                |
| UniqueCarrier     | Número de avión  |
| FlightNum         | Número de vuelo  |
| TailNum           |  |
| ActualElapsedTime | Tiempo real de vuelo en minutos                            |
| CRSElapsedTime    | Tiempo esperado de vuelo                                   |
| AirTime           | Tiempo real de vuelo en minutos                            |
| ArrDelay          | Diferencia entre el tiempo programado de arribo y el real  |
| DepDelay          | Diferencia entre el tiempo programado de partida y el real |
| Origin            | Aeropuerto origen  |
| Dest              | Aeropuerto destino   |
| Distance          | Distancia (en millas) entre aeropuertos                    |
| TaxiIn            | Tiempo hasta desembarcar                                   |
| TaxiOut           | Tiempo hasta despegar                                      |
| Cancelled         | 1=Vuelo cancelado  |
| CancellationCode  | Código de razón específica de cancelación                  |
| Diverted          | 1=Vuelo desviado   |
| CarrierDelay      | Demora por el avión  |
| WeatherDelay      | Demora por clima   |
| NASDelay          | Demora por National Air System                             |
| SecurityDelay     | Demora por Seguridad                                       |
| LateAircraftDelay | Demora por aviones   |

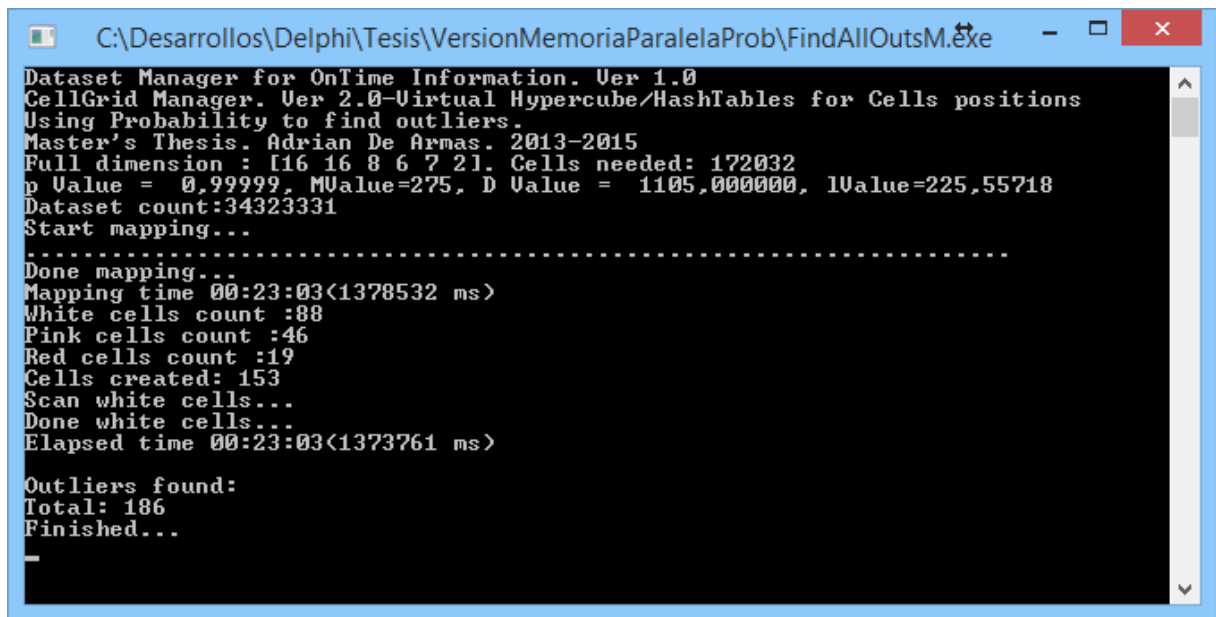
Tabla 10 – Estructura del banco de datos real (detalles de arribos y partidas)

Se seleccionan dos versiones del algoritmo, ambas se ejecutan completamente en memoria. La primer versión es la que ejecuta en forma paralela el algoritmo, y la segunda es igual a la primera sólo que incorpora la determinación probabilística de outliers.



La versión de memoria probabilística es una versión que libera todos los objetos pertenecientes a celdas rojas porque no serán utilizados en la determinación fina de outliers, permitiendo de esta manera, alojar todas las celdas y sólo aquellos objetos que se no hayan podido ser descartados como outliers.

En la figura 58 puede observarse el resultado de la ejecución del algoritmo **FindAllOutsM** en su variante Memoria/Probabilística.



```

C:\Desarrollos\Delphi\Tesis\VersionMemoriaParalelaProb\FindAllOutsM.exe
Dataset Manager for OnTime Information. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Using Probability to find outliers.
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [16 16 8 6 7 21. Cells needed: 172032
p Value = 0,99999, MValue=275, D Value = 1105,000000, lValue=225,55718
Dataset count:34323331
Start mapping...
.....
Done mapping...
Mapping time 00:23:03<1378532 ms>
White cells count :88
Pink cells count :46
Red cells count :19
Cells created: 153
Scan white cells...
Done white cells...
Elapsed time 00:23:03<1373761 ms>

Outliers found:
Total: 186
Finished...
    
```

Figura 58 –**FindAllOutsM** para 6 atributos,  $p=0,999992$ ,  $D=1105$ ,  $N=34.323.331$

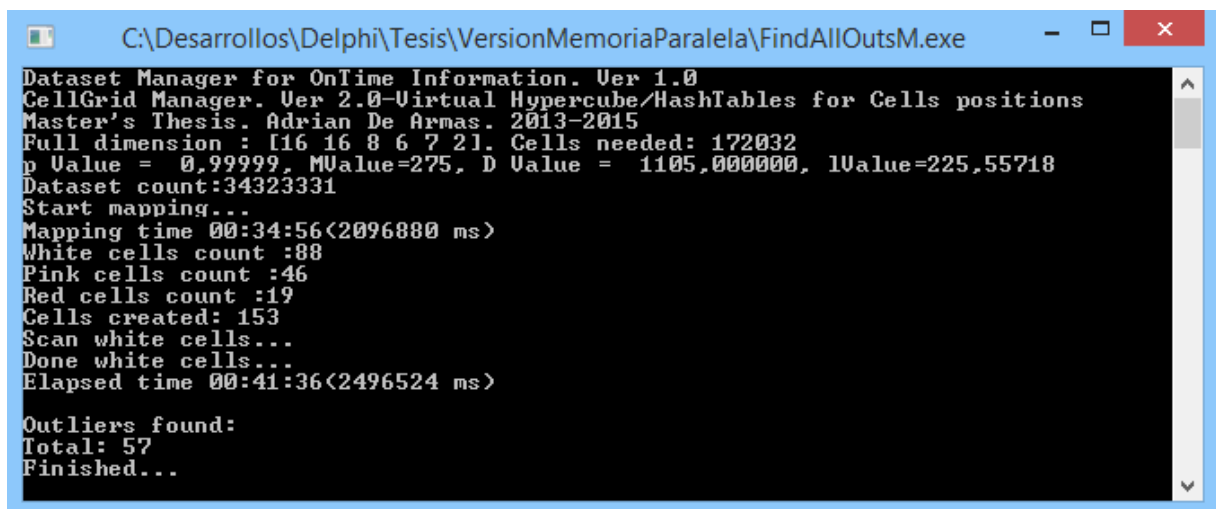
La versión Memoria/Probabilística es la más eficiente ya que en 23 minutos pudo detectar 186 outliers (129 falsos positivos) en un banco de datos de más de 34 millones de registros.

La versión más eficaz fue la versión de memoria paralela, esta versión fue afectada en su rendimiento por la cantidad de objetos (y celdas) que debían mantener residentes en memoria. Cuando se promediaba el 80% de la carga del banco de datos, el sistema operativo comenzó a realizar “swaps” de memoria que afectaron el rendimiento del proceso. De todas formas esta versión es relevante porque es la que detecta el 100% de los outliers del banco de datos (teniendo en cuenta los parámetros establecidos para el algoritmo).

| p=0,999992 d=1105 N=34.323.331 |         |          |     |    |
|--------------------------------|---------|----------|-----|----|
| Versión                        | 6D      |          |     |    |
|                                | Memoria | OUTLIERS | FP  | NF |
| Memoria Paralela               | 41m 36s | 57       | 0   | 0  |
| Memoria Probabilística         | 23m 3s  | 186      | 129 | 0  |

Tabla 11 – Estructura del banco de datos real (detalles de arribos y partidas)

Como puede observarse en la figura 59, el programa detectó, en casi 42 minutos, 57 outliers pertenecientes a un banco de datos de más de 34 millones de tuplas. El tiempo de mapeo entre la versión probabilística y la versión 100% eficaz, en condiciones normales, debería ser el mismo porque comparten el mismo código así que como se puede observar en la figura 58, el tiempo de mapeo fue de 23 minutos, mientras que la versión de memoria paralela ocupó 34 minutos. Si el equipo en el que se ejecutó el programa hubiese tenido más memoria disponible, los resultados podrían haberse obtenido diez minutos antes con la versión de memoria paralela. Un resumen de este experimento puede visualizarse en la tabla 11.



```

C:\Desarrollos\Delphi\Tesis\VersionMemoriaParalela\FindAllOutsM.exe
Dataset Manager for OnTime Information. Ver 1.0
CellGrid Manager. Ver 2.0-Virtual Hypercube/HashTables for Cells positions
Master's Thesis. Adrian De Armas. 2013-2015
Full dimension : [16 16 8 6 7 2]. Cells needed: 172032
p Uvalue = 0,99999, MValue=275, D Uvalue = 1105,000000, lUvalue=225,55718
Dataset count:34323331
Start mapping...
Mapping time 00:34:56<2096880 ms>
White cells count :88
Pink cells count :46
Red cells count :19
Cells created: 153
Scan white cells...
Done white cells...
Elapsed time 00:41:36<2496524 ms>

Outliers found:
Total: 57
Finished...
    
```

Figura 59 –FindAllOutsM para 6 atributos, p=0,999992, D=1105, N=34.323.331

Todos los vuelos detectados como outliers, junto con el valor de los atributos evaluados puede observarse en el anexo 12.4.

## 11. FUTURAS LINEAS DE INVESTIGACIÓN

Este trabajo final puede ser continuado en varias líneas de investigación.

Respecto al algoritmo **FindAllOutsM**:

- Detección de outliers mediante funciones de distancia personalizables para ajustar el algoritmo a la necesidad de los distintos dominios de aplicación. En este trabajo final la función de distancia utilizada es la distancia euclidiana. Dependiendo del dominio de aplicación, es probable que otras funciones de distancia más específicas aporten resultados más relevantes.
- Detección de outliers probabilística con porcentaje controlado de falsos positivos. La determinación probabilística de outliers propuesta en la sección 9.8 no tiene control sobre el porcentaje de falsos positivos generados, tener control sobre los falsos positivos creados es una buena mejora a la propuesta.
- Procesamiento de vecindarios  $L_2$  por capas incrementales. El espesor de capas del vecindario  $L_2$  de una celda en  $3D$  es de longitud 2 y se compone por las celdas que se encuentran a 2 y 3 celdas de distancia del objeto analizado. Respecto al objeto que se está procesando, todos los objetos que se encuentran en la capa 2 están más cerca que todos los objetos de la capa 3. Esta priorización para el análisis objeto a objeto puede aportar una mejora importante de rendimiento del algoritmo en casos donde, producto del mapeo celda/objeto, se hayan creado muchas celdas blancas.

Respecto de la detección de outliers en grandes bases de datos

- Paralelización de distintos algoritmos para la detección de outliers en grandes bases de datos. En el trabajo "*Parallel Algorithms for Distance-Based and Density-Based Outliers*" [Lozano y otros, 2005] se experimentó con la paralelización de dos algoritmos Bay y LOF con bancos de datos muy discretos como se puede observar en la sección 8. Los desafíos que presentan el procesamiento de grandes bancos de datos merecen ser investigados.

- Comparacion de distintos métodos de detección de outliers en grandes bases de datos para crear una guía de qué método puede ser más adecuado según el dominio de aplicación.
- Integrar la implementación de los algoritmos de detección de outliers con tecnología específica de Big Data para documentar las individualidades del caso.

## 12. CONCLUSIONES

Encontrar outliers es una tarea importante para muchas aplicaciones. Un outlier es cualquier dato que parece estar fuera de lugar respecto al resto de los datos, resultan raros, distintos. Para algunas aplicaciones, los eventos raros son los más interesantes.

En este trabajo final se implementó el algoritmo **FindAllOutsM** propuesto por Edwin Knorr y Raymond NG en 1998 en el trabajo “*Algorithms for Mining Distance-Based Outliers in Large Datasets*” [Knorr y otros, 1998].

El objetivo con la que se inició este trabajo fue proponer un algoritmo que pudiera ser tenido en cuenta para procesar grandes bases de datos con gran cantidad de atributos con el objetivo de detectar outliers. Disponer de algoritmos capaces de ser utilizados en estos escenarios es importante ya que el tamaño del conjunto de datos y el número de dimensiones han comprobado ser obstáculos clave para el análisis de los datos [Han y otros, 2006].

En el trabajo original [Knorr y otros, 1998], el algoritmo tiene documentadas ciertas restricciones que no lo hacía seleccionable para detectar outliers en banco de datos con muchos atributos (más de 5) principalmente por la complejidad del hipercubo que se requería para su implementación.

La implementación hecha del algoritmo fue dotada de varias características:

1. Se eliminó la restricción de la cantidad de atributos. El algoritmo puede ser usado con cualquier cantidad de atributos
2. Ya no es relevante la cantidad de celdas requeridas por el algoritmo sino la cantidad de celdas realmente utilizadas en la ejecución
3. Se implementaron dos versiones
  - a. Una muy veloz que procesa todos los objetos en memoria, su única limitante es la cantidad de memoria instalada en el equipo que se ejecuta
  - b. Otra más discreta en cuanto a velocidad porque utiliza una memoria secundaria (como un disco duro) pero que puede ser utilizada en cualquier escenario, con cualquier cantidad de tuplas y atributos

4. La eficacia del algoritmo se maneja de dos formas
  - a. Eficacia del 100% de detección de outliers basados en distancia
  - b. Detección probabilística de outliers. Menor eficacia por la detección de falsos positivos. La disminución de la eficacia es siempre menor que el aumento eficiencia lograda
5. Utilización de todos los procesadores instalados para mejorar la efectividad del algoritmo
6. La implementación del algoritmo requiere, a lo sumo, dos lecturas del banco de datos. Una lectura completa para el mapeo celda/objetos en cualquier versión. Las versiones híbridas (aquellas que usan un almacenamiento alternativo como el disco duro) deben releer ciertas tuplas para la determinación fina de outliers. En el peor escenario deben releerse todas las tuplas haciendo que la determinación completa de outliers se haga con dos lecturas. En los experimentos realizados, la relectura no superó nunca el 30% de los datos
7. La implementación del algoritmo es 100% genérica. El uso de otros bancos de datos sólo requiere la escritura de una clase que administre el banco de datos específico a analizar

Cada banco de datos debe ser analizado en profundidad ya que el rendimiento del algoritmo depende mucho de la distribución subyacente de los datos. La selección de la versión a utilizar depende mucho del banco de datos a procesar.

La utilización del algoritmo **FindAllOutsM** con distintos bancos de datos (de laboratorio y reales) de distintos tamaños y cantidad de atributos en experimentos demuestra que es viable usar la implementación de este algoritmo en grandes bases de datos dado por cumplido el objetivo inicial de este trabajo.

Las distintas versiones desarrolladas a lo largo de este trabajo brindan distintas alternativas de aplicación del algoritmo según el escenario que se presente.

### 13. REFERENCIAS BIBLIOGRÁFICAS

**AGGARWAL C.** *Outliers Analysis*. Springer, IBM T.J. Watson Research Center, Yorktown Heights, New York, USA. 2013.

**AUZA J.**, página web TECH SOURCE. *5 of the Best Free and Open Source Data Mining Software* [En línea]. junauza.com, 2010. Consulta realizada en 15/11/2013. <http://www.junauza.com/2010/11/free-data-mining-software.html>.

**AZEVEDO A.; SANTOS M.** *KDD, SEMMA AND CRISP-DM: A PARALLEL OVERVIEW* [En línea]. IADIS European Conference Data Mining. 2008. Consulta realizada en 19/11/2014. [https://www.researchgate.net/profile/Ana\\_Azevedo9/publication/220969845\\_KDD\\_SEMMA\\_and\\_CRISP-DM\\_a\\_parallel\\_overview/links/02bfe50cbb21f029f1000000.pdf](https://www.researchgate.net/profile/Ana_Azevedo9/publication/220969845_KDD_SEMMA_and_CRISP-DM_a_parallel_overview/links/02bfe50cbb21f029f1000000.pdf)

**BARNET V.; LEWIS T.** *Outliers in Statical Data*. John Wiley, 3a edición. Elsevier Inc, USA. 1994.

**BEN-GAL I.** *Outlier detection* [En línea]. Department of Industrial Engineering Tel-Aviv University. Israel. 2005. Consulta realizada en 08/10/2014. <http://www.eng.tau.ac.il/~bengal/outlier.pdf>

**BILLOR N.; KIRAL G.** *A comparison of Multiple Outlier Detection Methods for Regression Data*. Communication in Statistics – Simulation and Computation. Taylor & Francis Group, LLC. USA. 2008.

**BREUNIG M.; KRIEGEL H.; NG R.; SANDER J.** *LOF: Identifying Density-Based local Outliers* [En línea]. ACM SIGMOD Conference 2000. USA. 2000. Consulta realizada en 10/03/2014. [www.dbs.ifi.lmu.de/Publikationen/Papers/LOF.pdf](http://www.dbs.ifi.lmu.de/Publikationen/Papers/LOF.pdf)

**BTS (Bureau of Transportation Statistics).** *Airline On-Time Statistics and Delay Causes* [En línea]. USA: transtats.bts.gov, 2014. Consulta realizada el 20/01/2015. [http://www.transtats.bts.gov/Fields.asp?Table\\_ID=236](http://www.transtats.bts.gov/Fields.asp?Table_ID=236)

**BUCKNALL J.** *The tomes of Delphi. Algorithms and Data Structures*. 3a edición. Wordware Publishing Inc, Texas, USA. 2012.

**CHAPMAN P.; CLINTON J.; KERBER R; KHABAZA T.;REINARTZ T.;SHEARER C.;WIRTH R.** *CRISP-DM 1.0 Step by step mining guide* [En línea]. CRISP-DM consortium. 2000. Consulta realizada el 12/08/2014. <http://public.dhe.ibm.com/common/ssi/ecm/yt/en/ytw03084usen/YTW03084USEN.PDF>

**DAVIES, L.;GATHER, U.** *The identification of multiple outliers (with discussion)*. Journal of the American Statistical Association 88. Taylor & Francis Group, USA. 1993

**FRAWLEY W.;PIATETSKY-SHAPIO G.;MATHEUS C.** *Knowledge Discovery in Data Bases: An Overview*. AI Magazine Volume 13 Number 3. ASSOCIATION FOR THE ADVANCEMENT OF ARTIFICIAL INTELLIGENCE, USA.1992.

**HAN J.;KAMBER M..** *Data Mining – Concepts and techniques*. 2a edición. The Morgan Kaufmann Series in Data Management Systems. Elsevier Inc, USA. 2006.

**HAND D.;MANNILA H.;SMYTH P.** *Principles of Data Mining*. The MIT Press. Massachusetts Institute of Technology, USA. 2001.

**HAWKINS D..** *Identification of Outliers (Monographs on Statistics and Applied Probability) vol 3*. Chapman and Hall, London. 1980.

**INMON W.H..** *Building the Data Warehouse*. 3a ed. John Wiley & Sons, USA.1996.

**JONSON T.;KWOK I.;NG R.** *Fast Computation of 2-Dimensional Depth Contours* [En línea]. AT&T Research Center, USA. Departament of Computer Science – University of British Columbia. 1998. . Consulta realizada el 13/08/2014. [www.aaai.org/Papers/KDD/1998/KDD98-038.pdf](http://www.aaai.org/Papers/KDD/1998/KDD98-038.pdf)

**JONES N.C.;PAVEL A.P.** *An Introduction to Bioinformatics Algorithms*. MIT Press. Massachusetts Institute of Technology, USA. 2004.

**KRIEDEL H.;SCHUBERT M;ZIMEK A.** *Angle-Based Outlier Detection in High-dimensional Data* [En línea]. Department of Computer Science & Database Systems Group - Ludwig-Maximilians-Universität München. 2008. Consulta realizada el 21/08/2013. [www.dbs.ifi.lmu.de/~zimek/.../KDD08-ABOD.pdf](http://www.dbs.ifi.lmu.de/~zimek/.../KDD08-ABOD.pdf)

**KNORR E.M.;NG R.T.** *A unified notion of outliers: Properties and Computation* [En línea]. Department of Computer Science, University of British Columbia, Vancouver,



BC, V6T 1Z4, Canada. 1997. Consulta realizada el 13/08/2013.  
[www.aaai.org/Papers/KDD/1997/KDD97-044.pdf](http://www.aaai.org/Papers/KDD/1997/KDD97-044.pdf)

**KNORR E.M.;NG R.T.** *Algorithms for Mining Distance-Based Outliers in Large Datasets* [En línea]. Department of Computer Science, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada. 1998. Consulta realizada el 13/08/2013.  
[www.vldb.org/conf/1998/p392.pdf](http://www.vldb.org/conf/1998/p392.pdf)

**KNORR E.M.;NG R.T.;TUCAKOV V.** *Distance-based outliers: algorithms and applications* [En línea]. Department of Computer Science, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada. 1999. Consulta realizada el 13/08/2013.  
<http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=D4B0CBF365D06AB31F6F74F7F596566E?doi=10.1.1.43.1842&rep=rep1&type=pdf>

**KNORR, E. M.** *Outliers and Data Mining: Finding Exceptions in Data* [En línea]. Department of Computer Science, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada. 2002. Consulta realizada el 13/08/2013.  
[https://www.cs.ubc.ca/grads/resources/thesis/May02/Ed\\_Knorr.pdf](https://www.cs.ubc.ca/grads/resources/thesis/May02/Ed_Knorr.pdf)

**MICROSOFT.** *Conceptos de minería de datos* [En línea]. USA: microsoft.com, 2013. Consulta realizada 01/01/2013. <http://msdn.microsoft.com/es-es/library/ms174949.aspx>.

**MOLINA LÓPEZ J.M.;GARCÍA HERRERO J.** *Técnicas de análisis de datos*. Universidad Carlos III, Madrid. 2006. Consulta realizada el 26/02/2013.  
[www.giaa.inf.uc3m.es/docencia/II/ADatos/apuntesAD.pdf](http://www.giaa.inf.uc3m.es/docencia/II/ADatos/apuntesAD.pdf)

**MUNISWAMY V.V.** *Design and analysis of algorithms*. I.K.Internacional Publishing House Pvt. Ltd, New Delphi, India. 2009

**LOZANO E.;ACUÑA E.** *Parallel Algorithms for Distance-Based and Density-Based Outliers* [En línea]. IEEE International Conference on Data Mining (ICDM). 2005. Consulta realizada el 26/06/2014. [academic.uprm.edu/eacuna/elioedgariee.pdf](http://academic.uprm.edu/eacuna/elioedgariee.pdf)

**PAPADIMITRIOU S.;KITAGAWA H.;GIBBONS P.; FALOUTSOS C.** *LOCI: Fast Outlier Detection using the Local Correlation Integral* [En línea].

International Conference on Data Engineering (ICDE). 2003. Consulta realizada el 28/06/2014. [http://www.informedia.cs.cmu.edu/documents/loci\\_icde03.pdf](http://www.informedia.cs.cmu.edu/documents/loci_icde03.pdf)

**PHAM N.; PAGH R.** *A Near-linear Time Approximation Algorithm for*

*Angle-based Outlier Detection in High-dimensional Data* [En línea]. IT University of Copenhagen, Dinamarca. 2012. Consulta realizada el 12/09/2013. [www.itu.dk/people/pagh/papers/outlier.pdf](http://www.itu.dk/people/pagh/papers/outlier.pdf)

**RAMASWAMY S.; RASTOGI R.; SHIM K.** *Efficient Algorithms for Mining Outliers from Large Data Sets*. ACM SIGMOD International Conference on Management of Data. 2000.

**WU X.; KUMAR V.; QUINLAN J.; GHOSH J.; YANG Q.; MOTODA H.; MCLACHLAN G.; NG A.; LIU B.; YU P.; ZHOU Z.; STEINBACH M.; HAND D.; STEINBERG D.** *Top ten algorithms in data mining* [En línea]. Springer-Verlag London Limited. 2007. Consulta realizada el 12/09/2014. <http://www.cs.umd.edu/~samir/498/10Algorithms-08.pdf>

## 14. ANEXOS

### 14.1. LOGS DE MAPEOS Y CONTROL DE CELDAS BLANCAS PARA CASO DE 2 DIMENSIONES

#### 14.1.1. LOGS DE MAPEOS ENTRE CELDAS Y OBJETOS

Cell [0,0] = 1 Items White  
1  
Cell [9,0] = 1 Items White  
15  
Cell [1,2] = 1 Items White  
2  
Cell [7,2] = 1 Items White  
7  
Cell [11,2] = 1 Items White  
16  
Cell [4,4] = 1 Items White  
5  
Cell [6,4] = 1 Items White  
6  
Cell [2,5] = 1 Items White  
3  
Cell [12,5] = 1 Items White  
17  
Cell [9,6] = 1 Items White  
14  
Cell [2,7] = 1 Items White  
4  
Cell [4,7] = 1 Items White  
8  
Cell [7,7] = 1 Items White  
9  
Cell [8,7] = 1 Items White  
12  
Cell [12,7] = 1 Items Pink  
20  
Cell [8,8] = 1 Items White  
13  
Cell [12,8] = 2 Items Red  
18  
19  
Cell [6,9] = 1 Items White  
10  
Cell [7,9] = 1 Items White  
11  
Cell [0,0] = 1 Items White  
1  
Cell [9,0] = 1 Items White  
15  
Cell [1,2] = 1 Items White  
2  
Cell [7,2] = 1 Items White  
7

Cell [11,2] = 1 Items White  
16  
Cell [4,4] = 1 Items White  
5  
Cell [6,4] = 1 Items White  
6  
Cell [2,5] = 1 Items White  
3  
Cell [12,5] = 1 Items White  
17  
Cell [9,6] = 1 Items White  
14  
Cell [2,7] = 1 Items White  
4  
Cell [4,7] = 1 Items White  
8  
Cell [7,7] = 1 Items White  
9  
Cell [8,7] = 1 Items White  
12  
Cell [12,7] = 1 Items Pink  
20  
Cell [8,8] = 1 Items White  
13  
Cell [12,8] = 2 Items Red  
18  
19  
Cell [6,9] = 1 Items White  
10  
Cell [7,9] = 1 Items White  
11

#### 14.1.2. LOG DE CONTROL DE CELDAS BLANCAS

Analizing Cell [0,0]  
[0,1]  
[1,0]  
[1,1]  
L1 Count 1  
L2 Count 2  
Distance to 2 2,2361  
Number of objects withing 2,0000 distance: 1  
Analizing Cell [9,0]  
[8,0]  
[8,1]  
[9,1]  
[10,0]  
[10,1]  
L1 Count 1  
L2 Count 3  
Distance to 7 2,5000  
Distance to 16 2,2361  
Number of objects withing 2,0000 distance: 1  
Analizing Cell [1,2]

```

[0,1]
[0,2]
[0,3]
[1,1]
[1,3]
[2,1]
[2,2]
[2,3]
L1 Count 1
L2 Count 4
Distance to 1 2,2361
Distance to 3 2,2361
Distance to 5 2,2361
Number of objects withing 2,0000 distance: 1
Analizing Cell [7,2]
[6,1]
[6,2]
[6,3]
[7,1]
[7,3]
[8,1]
[8,2]
[8,3]
L1 Count 1
L2 Count 4
Distance to 5 2,6926
Distance to 6 1,8028
Number of objects withing 2,0000 distance: 2
Analizing Cell [11,2]
[10,1]
[10,2]
[10,3]
[11,1]
[11,3]
[12,1]
[12,2]
[12,3]
L1 Count 1
L2 Count 3
Distance to 15 2,2361
Distance to 17 2,0616
Number of objects withing 2,0000 distance: 1
Analizing Cell [4,4]
[3,3]
[3,4]
[3,5]
[4,3]
[4,5]
[5,3]
[5,4]
[5,5]
L1 Count 1
L2 Count 8
Distance to 2 2,2361

```

Distance to 3 1,4142  
Number of objects withing 2,0000 distance: 2  
Analizing Cell [6,4]  
[5,3]  
[5,4]  
[5,5]  
[6,3]  
[6,5]  
[7,3]  
[7,4]  
[7,5]  
L1 Count 1  
L2 Count 7  
Distance to 5 1,5811  
Number of objects withing 2,0000 distance: 2  
Analizing Cell [2,5]  
[1,4]  
[1,5]  
[1,6]  
[2,4]  
[2,6]  
[3,4]  
[3,5]  
[3,6]  
L1 Count 1  
L2 Count 5  
Distance to 2 2,2361  
Distance to 4 1,0000  
Number of objects withing 2,0000 distance: 2  
Analizing Cell [12,5]  
[11,4]  
[11,5]  
[11,6]  
[12,4]  
[12,6]  
[13,4]  
[13,5]  
[13,6]  
L1 Count 1  
L2 Count 6  
Distance to 14 2,0616  
Distance to 16 2,0616  
Distance to 20 1,0000  
Number of objects withing 2,0000 distance: 2  
Analizing Cell [9,6]  
[8,5]  
[8,6]  
[8,7]  
[9,5]  
[9,7]  
[10,5]  
[10,6]  
[10,7]  
L1 Count 2

```
Analizing Cell [2,7]
[1,6]
[1,7]
[1,8]
[2,6]
[2,8]
[3,6]
[3,7]
[3,8]
L1 Count 1
L2 Count 4
Distance to 3 1,0000
Number of objects withing 2,0000 distance: 2
Analizing Cell [4,7]
[3,6]
[3,7]
[3,8]
[4,6]
[4,8]
[5,6]
[5,7]
[5,8]
L1 Count 1
L2 Count 8
Distance to 3 1,4142
Number of objects withing 2,0000 distance: 2
Analizing Cell [7,7]
[6,6]
[6,7]
[6,8]
[7,6]
[7,8]
[8,6]
[8,7]
[8,8]
L1 Count 3
Analizing Cell [8,7]
[7,6]
[7,7]
[7,8]
[8,6]
[8,8]
[9,6]
[9,7]
[9,8]
L1 Count 4
Analizing Cell [8,8]
[7,7]
[7,8]
[7,9]
[8,7]
[8,9]
[9,7]
[9,8]
```

```

[9,9]
L1 Count 4
Analizing Cell [6,9]
[5,8]
[5,9]
[5,10]
[6,8]
[6,10]
[7,8]
[7,9]
[7,10]
L1 Count 2
Analizing Cell [7,9]
[6,8]
[6,9]
[6,10]
[7,8]
[7,10]
[8,8]
[8,9]
[8,10]
L1 Count 3
Finished
    
```

### 14.2. ARCHIVO NHL94 UTILIZADO EN EL PAPER ORIGINAL

| %team | P | NO. | PLAYER(* is rookie) | GP | G  | A  | PTS | +/- | PIM | PP | SH | GW | GT | S   | PCTG |
|-------|---|-----|---------------------|----|----|----|-----|-----|-----|----|----|----|----|-----|------|
| % 1   | 2 | 3   | 4                   | 5  | 6  | 7  | 8   | 9   | 10  | 11 | 12 | 13 | 14 | 15  | 16   |
| LA    | C | 99  | WAYNE GRETZKY       | 81 | 38 | 92 | 130 | -25 | 20  | 14 | 4  | 0  | 1  | 233 | 16.3 |
| DET   | C | 91  | SERGEI FEDOROV      | 82 | 56 | 64 | 120 | 48  | 34  | 13 | 4  | 10 | 0  | 337 | 16.6 |
| BOS   | C | 12  | ADAM OATES          | 77 | 32 | 80 | 112 | 10  | 45  | 16 | 2  | 3  | 0  | 197 | 16.2 |
| TOR   | C | 93  | DOUG GILMOUR        | 83 | 27 | 84 | 111 | 25  | 105 | 10 | 1  | 3  | 2  | 167 | 16.2 |
| VAN   | R | 10  | PAVEL BURE          | 76 | 60 | 47 | 107 | 1   | 86  | 25 | 4  | 9  | 0  | 374 | 16.0 |
| CHI   | C | 27  | JEREMY ROENICK      | 84 | 46 | 61 | 107 | 21  | 125 | 24 | 5  | 5  | 1  | 281 | 16.4 |
| PHI   | R | 8   | MARK RECCHI         | 84 | 40 | 67 | 107 | -2  | 46  | 11 | 0  | 5  | 0  | 217 | 18.4 |
| STL   | L | 19  | BRENDAN SHANAHAN    | 81 | 52 | 50 | 102 | -9  | 211 | 15 | 7  | 8  | 1  | 397 | 13.1 |
| PIT   | R | 68  | JAROMIR JAGR        | 80 | 32 | 67 | 99  | 15  | 61  | 9  | 0  | 6  | 2  | 298 | 10.7 |
| TOR   | L | 14  | DAVE ANDREYCHUK     | 83 | 53 | 45 | 98  | 22  | 98  | 21 | 5  | 8  | 0  | 333 | 15.9 |
| STL   | R | 16  | BRETT HULL          | 81 | 57 | 40 | 97  | -3  | 38  | 25 | 3  | 6  | 1  | 392 | 14.5 |
| PHI   | C | 88  | ERIC LINDROS        | 65 | 44 | 53 | 97  | 16  | 103 | 13 | 2  | 9  | 1  | 197 | 22.3 |
| PHI   | C | 17  | ROD BRIND'AMOUR     | 84 | 35 | 62 | 97  | -9  | 85  | 14 | 1  | 4  | 0  | 230 | 15.2 |
| NYI   | C | 77  | PIERRE TURGEON      | 69 | 38 | 56 | 94  | 14  | 18  | 10 | 4  | 6  | 0  | 254 | 15.0 |
| DET   | R | 26  | RAY SHEPPARD        | 82 | 52 | 41 | 93  | 13  | 26  | 19 | 0  | 5  | 0  | 260 | 20.0 |
| DAL   | C | 9   | MIKE MODANO         | 76 | 50 | 43 | 93  | -8  | 54  | 18 | 0  | 4  | 2  | 281 | 17.8 |
| CAL   | C | 26  | ROBERT REICHEL      | 84 | 40 | 53 | 93  | 20  | 58  | 14 | 0  | 6  | 0  | 249 | 16.1 |
| PIT   | C | 10  | RON FRANCIS         | 82 | 27 | 66 | 93  | -3  | 62  | 8  | 0  | 2  | 1  | 216 | 12.5 |
| QUE   | C | 19  | JOE SAKIC           | 84 | 28 | 64 | 92  | -8  | 18  | 10 | 1  | 9  | 1  | 279 | 10.0 |
| MON   | L | 25  | VINCENT DAMPHOUSSE  | 84 | 40 | 51 | 91  | 0   | 75  | 13 | 0  | 10 | 1  | 274 | 14.6 |
| BOS   | D | 77  | RAY BOURQUE         | 72 | 20 | 71 | 91  | 26  | 58  | 10 | 3  | 1  | 1  | 386 | 5.2  |
| NYR   | D | 21  | SERGEI ZUBOV        | 78 | 12 | 77 | 89  | 20  | 39  | 9  | 0  | 1  | 0  | 222 | 5.4  |
| PIT   | L | 25  | KEVIN STEVENS       | 83 | 41 | 47 | 88  | -24 | 155 | 21 | 0  | 4  | 0  | 284 | 14.4 |
| LA    | L | 20  | LUC ROBITAILLE      | 83 | 44 | 42 | 86  | -20 | 86  | 24 | 0  | 3  | 0  | 267 | 16.5 |
| BUF   | C | 10  | DALE HAWERCHUK      | 81 | 35 | 51 | 86  | 10  | 91  | 13 | 1  | 7  | 0  | 227 | 15.4 |
| CAL   | R | 14  | THEOREN FLEURY      | 83 | 40 | 45 | 85  | 30  | 186 | 16 | 1  | 6  | 0  | 278 | 14.4 |



|     |   |    |                   |    |    |    |    |     |     |    |   |    |   |     |       |
|-----|---|----|-------------------|----|----|----|----|-----|-----|----|---|----|---|-----|-------|
| QUE | C | 13 | MATS SUNDIN       | 84 | 32 | 53 | 85 | 1   | 60  | 6  | 2 | 4  | 0 | 226 | 14.2  |
| WAS | C | 90 | JOE JUNEAU        | 74 | 19 | 66 | 85 | 11  | 41  | 6  | 0 | 2  | 1 | 164 | 11.6  |
| CAL | L | 10 | GARY ROBERTS      | 73 | 41 | 43 | 84 | 37  | 145 | 12 | 3 | 5  | 1 | 202 | 20.3  |
| NYR | C | 11 | MARK MESSIER      | 76 | 26 | 58 | 84 | 25  | 76  | 6  | 2 | 5  | 0 | 216 | 12.0  |
| STL | C | 15 | CRAIG JANNEY      | 69 | 16 | 68 | 84 | -1  | 24  | 8  | 0 | 7  | 0 | 95  | 16.8  |
| PHI | L | 19 | *MIKAEL RENBERG   | 83 | 38 | 44 | 82 | 8   | 36  | 9  | 0 | 1  | 0 | 195 | 19.5  |
| CAL | D | 2  | AL MACINNIS       | 75 | 28 | 54 | 82 | 35  | 95  | 12 | 1 | 5  | 0 | 324 | 8.6   |
| DET | C | 19 | STEVE YZERMAN     | 58 | 24 | 58 | 82 | 11  | 36  | 7  | 3 | 3  | 1 | 217 | 11.1  |
| WIN | L | 7  | KEITH TKACHUK     | 84 | 41 | 40 | 81 | -12 | 255 | 22 | 3 | 3  | 1 | 218 | 18.8  |
| DAL | R | 26 | RUSS COURTNALL    | 84 | 23 | 57 | 80 | 6   | 59  | 5  | 0 | 4  | 0 | 231 | 10.0  |
| NYR | L | 9  | ADAM GRAVES       | 84 | 52 | 27 | 79 | 27  | 127 | 20 | 4 | 4  | 1 | 291 | 17.9  |
| BUF | R | 89 | ALEXANDER MOGILNY | 66 | 32 | 47 | 79 | 8   | 22  | 17 | 0 | 7  | 1 | 258 | 12.4  |
| OTT | C | 19 | *ALEXEI YASHIN    | 83 | 30 | 49 | 79 | -49 | 22  | 11 | 2 | 3  | 0 | 232 | 12.9  |
| NYR | D | 2  | BRIAN LEETCH      | 84 | 23 | 56 | 79 | 28  | 67  | 17 | 1 | 4  | 0 | 328 | 7.0   |
| NJ  | D | 4  | SCOTT STEVENS     | 83 | 18 | 60 | 78 | 53  | 112 | 5  | 1 | 4  | 0 | 215 | 8.4   |
| LA  | L | 17 | JARI KURRI        | 81 | 31 | 46 | 77 | -24 | 48  | 14 | 4 | 3  | 1 | 198 | 15.7  |
| DET | D | 77 | PAUL COFFEY       | 80 | 14 | 63 | 77 | 28  | 106 | 5  | 0 | 3  | 0 | 278 | 5.0   |
| TOR | L | 17 | WENDEL CLARK      | 64 | 46 | 30 | 76 | 10  | 115 | 21 | 0 | 8  | 0 | 275 | 16.7  |
| NYI | L | 32 | STEVE THOMAS      | 78 | 42 | 33 | 75 | -9  | 139 | 17 | 0 | 5  | 2 | 249 | 16.9  |
| HAR | R | 16 | PAT VERBEEK       | 84 | 37 | 38 | 75 | -15 | 177 | 15 | 1 | 3  | 1 | 226 | 16.4  |
| CAL | C | 25 | JOE NIEUWENDYK    | 64 | 36 | 39 | 75 | 19  | 51  | 14 | 1 | 7  | 1 | 191 | 18.9  |
| BOS | R | 8  | CAM NEELY         | 49 | 50 | 24 | 74 | 12  | 54  | 20 | 0 | 13 | 1 | 185 | 27.0  |
| WIN | C | 19 | NELSON EMERSON    | 83 | 33 | 41 | 74 | -38 | 80  | 4  | 5 | 6  | 1 | 282 | 11.7  |
| EDM | C | 39 | DOUG WEIGHT       | 84 | 24 | 50 | 74 | -22 | 47  | 4  | 1 | 1  | 0 | 188 | 12.8  |
| DET | C | 13 | VYACHESLAV KOZLOV | 77 | 34 | 39 | 73 | 27  | 50  | 8  | 2 | 6  | 0 | 202 | 16.8  |
| DET | L | 55 | KEITH PRIMEAU     | 78 | 31 | 42 | 73 | 34  | 173 | 7  | 3 | 4  | 2 | 155 | 20.0  |
| PIT | D | 55 | LARRY MURPHY      | 84 | 17 | 56 | 73 | 10  | 44  | 7  | 0 | 4  | 0 | 236 | 7.2   |
| NJ  | R | 44 | STEPHANE RICHER   | 80 | 36 | 36 | 72 | 31  | 16  | 7  | 3 | 9  | 3 | 217 | 16.6  |
| MON | L | 23 | BRIAN BELLOWS     | 77 | 33 | 38 | 71 | 9   | 36  | 13 | 0 | 2  | 1 | 251 | 13.2  |
| WIN | C | 10 | ALEXEI ZHAMNOV    | 61 | 26 | 45 | 71 | -20 | 62  | 7  | 0 | 1  | 1 | 196 | 13.3  |
| FLO | R | 22 | BOB KUDELSKI      | 86 | 40 | 30 | 70 | -33 | 24  | 17 | 0 | 3  | 1 | 251 | 15.9  |
| PIT | R | 7  | JOE MULLEN        | 84 | 38 | 32 | 70 | 9   | 41  | 6  | 2 | 9  | 0 | 231 | 16.5  |
| NJ  | R | 15 | JOHN MACLEAN      | 80 | 37 | 33 | 70 | 30  | 95  | 8  | 0 | 4  | 0 | 277 | 13.4  |
| CHI | R | 17 | JOE MURPHY        | 81 | 31 | 39 | 70 | 1   | 111 | 7  | 4 | 4  | 0 | 222 | 14.0  |
| NYI | L | 27 | DEREK KING        | 78 | 30 | 40 | 70 | 18  | 59  | 10 | 0 | 7  | 1 | 171 | 17.5  |
| WAS | C | 17 | MIKE RIDLEY       | 81 | 26 | 44 | 70 | 15  | 24  | 10 | 2 | 4  | 3 | 144 | 18.1  |
| VAN | L | 14 | GEOFF COURTNALL   | 82 | 26 | 44 | 70 | 15  | 123 | 12 | 1 | 2  | 0 | 264 | 9.9   |
| PHI | D | 3  | GARRY GALLEY      | 81 | 10 | 60 | 70 | -11 | 91  | 5  | 1 | 0  | 1 | 186 | 5.4   |
| NYI | C | 33 | BENOIT HOGUE      | 83 | 36 | 33 | 69 | -7  | 73  | 9  | 5 | 3  | 0 | 218 | 16.5  |
| SJ  | R | 22 | ULF DAHLEN        | 78 | 25 | 44 | 69 | -1  | 10  | 15 | 0 | 5  | 1 | 190 | 13.2  |
| EDM | C | 7  | *JASON ARNOTT     | 78 | 33 | 35 | 68 | 1   | 104 | 10 | 0 | 4  | 1 | 194 | 17.01 |
| SJ  | R | 24 | SERGEI MAKAROV    | 80 | 30 | 38 | 68 | 11  | 78  | 10 | 0 | 5  | 0 | 155 | 19.35 |
| VAN | C | 7  | CLIFF RONNING     | 76 | 25 | 43 | 68 | 7   | 42  | 10 | 0 | 4  | 1 | 197 | 12.69 |
| LA  | D | 4  | ROB BLAKE         | 84 | 20 | 48 | 68 | -7  | 137 | 7  | 0 | 6  | 0 | 304 | 6.58  |
| HAR | C | 8  | GEOFF SANDERSON   | 82 | 41 | 26 | 67 | -13 | 42  | 15 | 1 | 6  | 2 | 266 | 15.41 |
| SJ  | C | 27 | TODD ELIK         | 79 | 25 | 41 | 66 | -3  | 95  | 9  | 0 | 4  | 1 | 185 | 13.51 |
| VAN | D | 22 | JEFF BROWN        | 74 | 14 | 52 | 66 | -11 | 56  | 7  | 0 | 3  | 1 | 237 | 5.91  |
| QUE | L | 31 | VALERI KAMENSKY   | 76 | 28 | 37 | 65 | 12  | 42  | 6  | 0 | 1  | 0 | 170 | 16.47 |
| TOR | R | 11 | MIKE GARTNER      | 81 | 34 | 30 | 64 | 20  | 62  | 11 | 5 | 4  | 0 | 275 | 12.36 |
| PIT | C | 82 | MARTIN STRAKA     | 84 | 30 | 34 | 64 | 24  | 24  | 2  | 0 | 6  | 1 | 130 | 23.08 |
| SJ  | D | 6  | SANDIS OZOLINSH   | 81 | 26 | 38 | 64 | 16  | 24  | 4  | 0 | 3  | 0 | 157 | 16.56 |
| TB  | C | 19 | BRIAN BRADLEY     | 78 | 24 | 40 | 64 | -8  | 56  | 6  | 0 | 2  | 0 | 180 | 13.33 |
| DAL | C | 15 | DAVE GAGNER       | 76 | 32 | 29 | 61 | 13  | 83  | 10 | 0 | 6  | 1 | 213 | 15.02 |
| VAN | R | 16 | TREVOR LINDEN     | 84 | 32 | 29 | 61 | 6   | 73  | 10 | 2 | 3  | 0 | 234 | 13.68 |

|     |   |    |                   |    |    |    |    |     |     |    |   |   |   |     |       |
|-----|---|----|-------------------|----|----|----|----|-----|-----|----|---|---|---|-----|-------|
| FLO | R | 27 | SCOTT MELLANBY    | 80 | 30 | 30 | 60 | 0   | 149 | 17 | 0 | 4 | 1 | 204 | 14.71 |
| NYR | R | 28 | STEVE LARMER      | 68 | 21 | 39 | 60 | 14  | 41  | 6  | 4 | 7 | 0 | 146 | 14.38 |
| CHI | D | 7  | CHRIS CHELIOS     | 76 | 16 | 44 | 60 | 12  | 212 | 7  | 1 | 2 | 0 | 219 | 7.31  |
| BUF | R | 28 | DONALD AUDETTE    | 77 | 29 | 30 | 59 | 2   | 41  | 16 | 1 | 4 | 0 | 207 | 14.01 |
| WAS | C | 8  | DIMITRI KHRISTICH | 83 | 29 | 29 | 58 | -2  | 73  | 10 | 0 | 4 | 1 | 195 | 14.87 |
| BUF | L | 13 | YURI KHYLEV       | 72 | 27 | 31 | 58 | 13  | 49  | 11 | 0 | 4 | 0 | 171 | 15.79 |
| PIT | R | 17 | TOMAS SANDSTROM   | 78 | 23 | 35 | 58 | -7  | 83  | 4  | 0 | 3 | 1 | 193 | 11.92 |
| WIN | L | 34 | DARRIN SHANNON    | 77 | 21 | 37 | 58 | -18 | 87  | 9  | 0 | 2 | 0 | 124 | 16.94 |
| HAR | C | 21 | ANDREW CASSELS    | 79 | 16 | 42 | 58 | -21 | 37  | 8  | 1 | 3 | 0 | 126 | 12.70 |
| BOS | D | 43 | AL IAFRATE        | 79 | 15 | 43 | 58 | 16  | 163 | 6  | 0 | 4 | 0 | 299 | 5.02  |
| BOS | D | 26 | GLEN WESLEY       | 81 | 14 | 44 | 58 | 1   | 64  | 6  | 1 | 1 | 1 | 265 | 5.28  |
| DET | R | 22 | DINO CICCARELLI   | 66 | 28 | 29 | 57 | 10  | 73  | 12 | 0 | 1 | 2 | 153 | 18.30 |
| NJ  | L | 25 | VALERI ZELEPUKIN  | 82 | 26 | 31 | 57 | 36  | 70  | 8  | 0 | 0 | 0 | 155 | 16.77 |
| MON | L | 11 | KIRK MULLER       | 76 | 23 | 34 | 57 | -1  | 96  | 9  | 2 | 3 | 0 | 168 | 13.69 |
| EDM | R | 8  | ZDENO CIGER       | 84 | 22 | 35 | 57 | -11 | 8   | 8  | 0 | 1 | 2 | 158 | 13.92 |
| NYI | D | 23 | VLADIMIR MALAKHOV | 76 | 10 | 47 | 57 | 29  | 80  | 4  | 0 | 2 | 0 | 235 | 4.26  |
| NYI | C | 18 | MARTY MCINNIS     | 81 | 25 | 31 | 56 | 31  | 24  | 3  | 5 | 3 | 1 | 136 | 18.38 |
| NYR | C | 27 | ALEXEI KOVALEV    | 76 | 23 | 33 | 56 | 18  | 154 | 7  | 0 | 3 | 0 | 184 | 12.50 |
| BUF | C | 26 | *DEREK PLANTE     | 77 | 21 | 35 | 56 | 4   | 24  | 8  | 1 | 2 | 0 | 147 | 14.29 |
| SJ  | C | 7  | IGOR LARIONOV     | 60 | 18 | 38 | 56 | 20  | 40  | 3  | 2 | 2 | 1 | 72  | 25.00 |
| DET | D | 5  | NICKLAS LIDSTROM  | 84 | 10 | 46 | 56 | 43  | 26  | 4  | 0 | 3 | 0 | 200 | 5.00  |
| TB  | R | 85 | PETR KLIMA        | 75 | 28 | 27 | 55 | -15 | 76  | 10 | 0 | 2 | 0 | 167 | 16.77 |
| PIT | R | 24 | DOUG BROWN        | 77 | 18 | 37 | 55 | 19  | 18  | 2  | 0 | 1 | 0 | 152 | 11.84 |
| VAN | L | 32 | MURRAY CRAVEN     | 78 | 15 | 40 | 55 | 5   | 30  | 2  | 1 | 3 | 0 | 115 | 13.04 |
| CAL | C | 92 | MICHAEL NYLANDER  | 73 | 13 | 42 | 55 | 8   | 30  | 4  | 0 | 1 | 2 | 95  | 13.68 |
| VAN | D | 21 | JYRKI LUMME       | 83 | 13 | 42 | 55 | 3   | 50  | 1  | 3 | 3 | 0 | 161 | 8.07  |
| WIN | R | 13 | TEEMU SELANNE     | 51 | 25 | 29 | 54 | -23 | 22  | 11 | 0 | 2 | 0 | 191 | 13.09 |
| EDM | L | 9  | SHAYNE CORSON     | 64 | 25 | 29 | 54 | -8  | 118 | 11 | 0 | 3 | 1 | 171 | 14.62 |
| NYR | L | 10 | ESA TIKKANEN      | 83 | 22 | 32 | 54 | 5   | 114 | 5  | 3 | 4 | 0 | 257 | 8.56  |
| BOS | C | 21 | TED DONATO        | 84 | 22 | 32 | 54 | 0   | 59  | 9  | 2 | 1 | 1 | 158 | 13.92 |
| SJ  | R | 17 | PAT FALLOON       | 83 | 22 | 31 | 53 | -3  | 18  | 6  | 0 | 1 | 0 | 193 | 11.40 |
| NYI | C | 20 | RAY FERRARO       | 82 | 21 | 32 | 53 | 1   | 83  | 5  | 0 | 3 | 3 | 136 | 15.44 |
| SJ  | L | 10 | JOHAN GARPENLOV   | 80 | 18 | 35 | 53 | 9   | 28  | 7  | 0 | 3 | 0 | 125 | 14.40 |
| ANA | C | 25 | TERRY YAKE        | 82 | 21 | 31 | 52 | 2   | 44  | 5  | 0 | 2 | 0 | 188 | 11.17 |
| MON | D | 27 | MATT SCHNEIDER    | 75 | 20 | 32 | 52 | 15  | 62  | 11 | 0 | 4 | 0 | 193 | 10.36 |
| DAL | C | 7  | NEAL BROTEN       | 79 | 17 | 35 | 52 | 10  | 62  | 2  | 1 | 1 | 0 | 153 | 11.11 |
| LA  | D | 2  | ALEXEI ZHITNIK    | 81 | 12 | 40 | 52 | -11 | 101 | 11 | 0 | 1 | 1 | 227 | 5.29  |
| PHI | D | 29 | YVES RACINE       | 67 | 9  | 43 | 52 | -11 | 48  | 5  | 1 | 1 | 1 | 142 | 6.34  |
| BOS | C | 20 | *BRYAN SMOLINSKI  | 83 | 31 | 20 | 51 | 4   | 82  | 4  | 3 | 5 | 0 | 179 | 17.32 |
| QUE | C | 9  | MIKE RICCI        | 83 | 30 | 21 | 51 | -9  | 113 | 13 | 3 | 6 | 1 | 138 | 21.74 |
| QUE | R | 48 | SCOTT YOUNG       | 76 | 26 | 25 | 51 | -4  | 14  | 6  | 1 | 1 | 0 | 236 | 11.02 |
| ANA | R | 20 | BOB CORKUM        | 76 | 23 | 28 | 51 | 4   | 18  | 3  | 3 | 0 | 1 | 180 | 12.78 |
| OTT | C | 91 | *ALEXANDRE DAIGLE | 84 | 20 | 31 | 51 | -45 | 40  | 4  | 0 | 2 | 0 | 168 | 11.90 |
| WIN | C | 25 | THOMAS STEEN      | 76 | 19 | 32 | 51 | -38 | 32  | 6  | 0 | 1 | 1 | 137 | 13.87 |
| WAS | D | 3  | SYLVAIN COTE      | 84 | 16 | 35 | 51 | 30  | 66  | 3  | 2 | 2 | 0 | 212 | 7.55  |
| HAR | C | 18 | ROBERT KRON       | 77 | 24 | 26 | 50 | 0   | 8   | 2  | 1 | 3 | 0 | 194 | 12.37 |
| NJ  | C | 10 | COREY MILLEN      | 78 | 20 | 30 | 50 | 24  | 52  | 4  | 0 | 3 | 1 | 132 | 15.15 |
| FLO | C | 26 | *JESSE BELANGER   | 70 | 17 | 33 | 50 | -4  | 16  | 11 | 0 | 3 | 1 | 104 | 16.35 |
| WAS | C | 20 | MICHAL PIVONKA    | 82 | 14 | 36 | 50 | 2   | 38  | 5  | 0 | 4 | 0 | 138 | 10.14 |
| EDM | D | 21 | IGOR KRAVCHUK     | 81 | 12 | 38 | 50 | -12 | 16  | 5  | 0 | 2 | 0 | 197 | 6.09  |
| PHI | L | 42 | JOSEF BERANEK     | 80 | 28 | 21 | 49 | -2  | 85  | 6  | 0 | 2 | 0 | 182 | 15.38 |
| NYR | C | 13 | SERGEI NEMCHINOV  | 76 | 22 | 27 | 49 | 13  | 36  | 4  | 0 | 6 | 0 | 144 | 15.28 |
| STL | R | 14 | KEVIN MILLER      | 75 | 23 | 25 | 48 | 6   | 83  | 6  | 3 | 5 | 0 | 154 | 14.94 |
| FLO | C | 14 | STU BARNES        | 77 | 23 | 24 | 47 | 4   | 38  | 8  | 1 | 3 | 0 | 172 | 13.37 |

|     |   |    |                   |    |    |    |    |     |     |    |   |   |   |     |       |
|-----|---|----|-------------------|----|----|----|----|-----|-----|----|---|---|---|-----|-------|
| FLO | L | 19 | ANDREI LOMAKIN    | 76 | 19 | 28 | 47 | 1   | 26  | 3  | 0 | 2 | 0 | 139 | 13.67 |
| CAL | D | 33 | ZARLEY ZALAPSKI   | 69 | 10 | 37 | 47 | -6  | 74  | 1  | 0 | 1 | 0 | 156 | 6.41  |
| NJ  | C | 9  | BERNIE NICHOLLS   | 61 | 19 | 27 | 46 | 24  | 86  | 3  | 0 | 1 | 1 | 142 | 13.38 |
| TB  | C | 9  | DENIS SAVARD      | 74 | 18 | 28 | 46 | -1  | 106 | 2  | 1 | 2 | 0 | 181 | 9.94  |
| MON | R | 12 | MIKE KEANE        | 80 | 16 | 30 | 46 | 6   | 119 | 6  | 2 | 2 | 1 | 129 | 12.40 |
| DET | D | 3  | STEVE CHIASSON    | 82 | 13 | 33 | 46 | 17  | 122 | 4  | 1 | 2 | 0 | 238 | 5.46  |
| NJ  | D | 27 | SCOTT NIEDERMAYER | 81 | 10 | 36 | 46 | 34  | 42  | 5  | 0 | 2 | 1 | 135 | 7.41  |
| DAL | D | 12 | GRANT LEDYARD     | 84 | 9  | 37 | 46 | 7   | 42  | 6  | 0 | 1 | 0 | 177 | 5.08  |
| CAL | C | 13 | GERMAN TITOV      | 76 | 27 | 18 | 45 | 20  | 28  | 8  | 3 | 2 | 0 | 153 | 17.65 |
| MON | L | 45 | GILBERT DIONNE    | 74 | 19 | 26 | 45 | -9  | 31  | 3  | 0 | 5 | 2 | 162 | 11.73 |
| ANA | L | 18 | GARRY VALK        | 78 | 18 | 27 | 45 | 8   | 100 | 4  | 1 | 5 | 0 | 165 | 10.91 |
| BUF | L | 27 | BRAD MAY          | 84 | 18 | 27 | 45 | -6  | 171 | 3  | 0 | 3 | 0 | 166 | 10.84 |
| NJ  | R | 12 | BILL GUERIN       | 81 | 25 | 19 | 44 | 14  | 101 | 2  | 0 | 3 | 0 | 195 | 12.82 |
| NJ  | R | 22 | CLAUDE LEMIEUX    | 79 | 18 | 26 | 44 | 13  | 86  | 5  | 0 | 5 | 0 | 181 | 9.94  |
| DAL | D | 14 | PAUL CAVALLINI    | 74 | 11 | 33 | 44 | 13  | 82  | 6  | 0 | 3 | 0 | 145 | 7.59  |
| DAL | C | 16 | DEAN EVASON       | 80 | 11 | 33 | 44 | -12 | 66  | 3  | 2 | 2 | 1 | 118 | 9.32  |
| WAS | R | 12 | PETER BONDRA      | 69 | 24 | 19 | 43 | 22  | 40  | 4  | 0 | 2 | 0 | 200 | 12.00 |
| TB  | R | 24 | DANTON COLE       | 81 | 20 | 23 | 43 | 7   | 32  | 8  | 1 | 4 | 0 | 149 | 13.42 |
| MON | C | 17 | JOHN LECLAIR      | 74 | 19 | 24 | 43 | 17  | 32  | 1  | 0 | 1 | 0 | 153 | 12.42 |
| OTT | C | 17 | DAVE MCLLWAIN     | 66 | 17 | 26 | 43 | -40 | 48  | 1  | 1 | 1 | 0 | 115 | 14.78 |
| ANA | L | 8  | TIM SWEENEY       | 78 | 16 | 27 | 43 | 3   | 49  | 6  | 1 | 2 | 0 | 114 | 14.04 |
| FLO | D | 5  | GORD MURPHY       | 84 | 14 | 29 | 43 | -11 | 71  | 9  | 0 | 2 | 3 | 172 | 8.14  |
| EDM | D | 2  | BOB BEERS         | 82 | 11 | 32 | 43 | -22 | 86  | 6  | 0 | 0 | 1 | 187 | 5.88  |
| TOR | D | 4  | DAVE ELLETT       | 68 | 7  | 36 | 43 | 6   | 42  | 5  | 0 | 1 | 1 | 146 | 4.79  |
| WAS | L | 18 | RANDY BURRIDGE    | 78 | 25 | 17 | 42 | -1  | 73  | 8  | 1 | 5 | 0 | 150 | 16.67 |
| LA  | L | 11 | MIKE DONNELLY     | 81 | 21 | 21 | 42 | 2   | 34  | 4  | 2 | 3 | 0 | 177 | 11.86 |
| PIT | C | 15 | SHAWN MCEACHERN   | 76 | 20 | 22 | 42 | 14  | 34  | 0  | 5 | 1 | 0 | 159 | 12.58 |
| PHI | R | 11 | KEVIN DINEEN      | 71 | 19 | 23 | 42 | -9  | 113 | 5  | 1 | 2 | 1 | 156 | 12.18 |
| CHI | R | 10 | TONY AMONTE       | 79 | 17 | 25 | 42 | 0   | 37  | 4  | 0 | 4 | 0 | 195 | 8.72  |
| TB  | C | 77 | *CHRIS GRATTON    | 84 | 13 | 29 | 42 | -25 | 123 | 5  | 1 | 2 | 1 | 161 | 8.07  |
| NYI | R | 26 | PATRICK FLATLEY   | 64 | 12 | 30 | 42 | 12  | 40  | 2  | 1 | 2 | 0 | 112 | 10.71 |
| WAS | D | 6  | CALLE JOHANSSON   | 84 | 9  | 33 | 42 | 3   | 59  | 4  | 0 | 1 | 0 | 141 | 6.38  |
| NJ  | L | 17 | TOM CHORSKE       | 76 | 21 | 20 | 41 | 14  | 32  | 1  | 1 | 4 | 0 | 131 | 16.03 |
| NYR | R | 36 | GLENN ANDERSON    | 85 | 21 | 20 | 41 | -5  | 62  | 7  | 0 | 3 | 0 | 149 | 14.09 |
| NYR | R | 16 | BRIAN NOONAN      | 76 | 18 | 23 | 41 | 7   | 69  | 10 | 0 | 6 | 1 | 160 | 11.25 |
| BUF | D | 42 | RICHARD SMEHLIK   | 84 | 14 | 27 | 41 | 22  | 69  | 3  | 3 | 1 | 1 | 106 | 13.21 |
| NYI | C | 39 | TRAVIS GREEN      | 83 | 18 | 22 | 40 | 16  | 44  | 1  | 0 | 2 | 1 | 164 | 10.98 |
| TB  | R | 14 | JOHN TUCKER       | 66 | 17 | 23 | 40 | 9   | 28  | 2  | 0 | 6 | 1 | 126 | 13.49 |
| WAS | D | 4  | KEVIN HATCHER     | 72 | 16 | 24 | 40 | -13 | 108 | 6  | 0 | 3 | 0 | 217 | 7.37  |
| QUE | C | 22 | RON SUTTER        | 73 | 15 | 25 | 40 | 2   | 90  | 5  | 0 | 2 | 0 | 108 | 13.89 |
| FLO | C | 20 | BRIAN SKRUDLAND   | 79 | 15 | 25 | 40 | 13  | 136 | 0  | 2 | 1 | 0 | 110 | 13.64 |
| PIT | R | 22 | RICK TOCCHET      | 51 | 14 | 26 | 40 | -15 | 134 | 5  | 1 | 2 | 1 | 150 | 9.33  |
| SJ  | L | 14 | RAY WHITNEY       | 61 | 14 | 26 | 40 | 2   | 14  | 1  | 0 | 0 | 1 | 82  | 17.07 |
| WIN | C | 18 | DALLAS DRAKE      | 62 | 13 | 27 | 40 | -1  | 49  | 1  | 2 | 3 | 0 | 112 | 11.61 |
| MON | D | 24 | LYLE ODELEIN      | 79 | 11 | 29 | 40 | 8   | 276 | 6  | 0 | 2 | 0 | 116 | 9.48  |
| NYI | D | 28 | TOM KURVERS       | 66 | 9  | 31 | 40 | 7   | 47  | 5  | 0 | 1 | 0 | 141 | 6.38  |
| SJ  | D | 8  | JEFF NORTON       | 64 | 7  | 33 | 40 | 16  | 36  | 1  | 0 | 0 | 0 | 92  | 7.61  |
| ANA | D | 23 | BILL HOULDER      | 80 | 14 | 25 | 39 | -18 | 40  | 3  | 0 | 3 | 0 | 187 | 7.49  |
| WAS | L | 10 | KELLY MILLER      | 84 | 14 | 25 | 39 | 8   | 32  | 0  | 1 | 3 | 0 | 138 | 10.14 |
| BUF | D | 8  | DOUG BODGER       | 75 | 7  | 32 | 39 | 8   | 76  | 5  | 1 | 1 | 0 | 144 | 4.86  |
| BUF | L | 19 | RANDY WOOD        | 84 | 22 | 16 | 38 | 11  | 71  | 2  | 2 | 5 | 0 | 161 | 13.66 |
| PHI | R | 18 | BRENT FEDYK       | 72 | 20 | 18 | 38 | -14 | 74  | 5  | 0 | 1 | 0 | 104 | 19.23 |
| NYR | L | 32 | STEPHANE MATTEAU  | 77 | 19 | 19 | 38 | 15  | 57  | 3  | 0 | 2 | 1 | 135 | 14.07 |
| DAL | R | 29 | TRENT KLATT       | 61 | 14 | 24 | 38 | 13  | 30  | 3  | 0 | 2 | 0 | 86  | 16.28 |

|     |   |    |                      |    |    |    |    |     |     |   |   |   |   |     |       |
|-----|---|----|----------------------|----|----|----|----|-----|-----|---|---|---|---|-----|-------|
| MON | C | 21 | GUY CARBONNEAU       | 79 | 14 | 24 | 38 | 16  | 48  | 0 | 0 | 1 | 0 | 120 | 11.67 |
| CAL | C | 17 | WES WALZ             | 53 | 11 | 27 | 38 | 20  | 16  | 1 | 0 | 0 | 0 | 79  | 13.92 |
| WAS | C | 32 | DALE HUNTER          | 52 | 9  | 29 | 38 | -4  | 131 | 1 | 0 | 1 | 0 | 61  | 14.75 |
| CHI | C | 12 | BRENT SUTTER         | 73 | 9  | 29 | 38 | 17  | 43  | 3 | 2 | 0 | 0 | 127 | 7.09  |
| VAN | D | 24 | JIRI SLEGR           | 78 | 5  | 33 | 38 | 0   | 86  | 1 | 0 | 0 | 0 | 160 | 3.13  |
| EDM | L | 33 | SCOTT PEARSON        | 72 | 19 | 18 | 37 | -4  | 165 | 3 | 0 | 7 | 0 | 160 | 11.88 |
| ANA | R | 14 | JOE SACCO            | 84 | 19 | 18 | 37 | -11 | 61  | 3 | 1 | 2 | 1 | 206 | 9.22  |
| PIT | C | 66 | MARIO LEMIEUX        | 22 | 17 | 20 | 37 | -2  | 32  | 7 | 0 | 4 | 0 | 92  | 18.48 |
| QUE | C | 38 | *IAIN FRASER         | 60 | 17 | 20 | 37 | -5  | 23  | 2 | 0 | 2 | 0 | 109 | 15.60 |
| FLO | L | 10 | DAVE LOWRY           | 80 | 15 | 22 | 37 | -4  | 64  | 3 | 0 | 3 | 1 | 122 | 12.30 |
| VAN | L | 8  | GREG ADAMS           | 68 | 13 | 24 | 37 | -1  | 20  | 5 | 1 | 2 | 0 | 139 | 9.35  |
| DAL | R | 20 | MIKE CRAIG           | 72 | 13 | 24 | 37 | -14 | 139 | 3 | 0 | 2 | 0 | 150 | 8.67  |
| PHI | C | 22 | MARK LAMB            | 85 | 12 | 24 | 36 | -44 | 72  | 4 | 1 | 2 | 0 | 124 | 9.68  |
| TOR | D | 15 | DMITRI MIRONOV       | 76 | 9  | 27 | 36 | 5   | 78  | 3 | 0 | 0 | 2 | 147 | 6.12  |
| DAL | L | 17 | MIKE MCPHEE          | 79 | 20 | 15 | 35 | 8   | 36  | 1 | 3 | 1 | 1 | 115 | 17.39 |
| WAS | R | 26 | KEITH JONES          | 68 | 16 | 19 | 35 | 4   | 149 | 5 | 0 | 1 | 0 | 97  | 16.49 |
| SJ  | R | 37 | ROB GAUDREAU         | 84 | 15 | 20 | 35 | -10 | 28  | 6 | 0 | 4 | 0 | 151 | 9.93  |
| TOR | R | 16 | NIKOLAI BORSCHESVSKY | 45 | 14 | 21 | 35 | 6   | 10  | 7 | 0 | 1 | 0 | 105 | 13.33 |
| CHI | L | 14 | PAUL YSEBAERT        | 71 | 14 | 21 | 35 | -7  | 26  | 3 | 0 | 1 | 0 | 151 | 9.27  |
| MON | D | 28 | ERIC DESJARDINS      | 84 | 12 | 23 | 35 | -1  | 97  | 6 | 1 | 3 | 0 | 193 | 6.22  |
| EDM | D | 15 | FREDRIK OLAUSSON     | 73 | 11 | 24 | 35 | -7  | 30  | 7 | 0 | 1 | 0 | 126 | 8.73  |
| CAL | D | 6  | JAMES PATRICK        | 68 | 10 | 25 | 35 | -5  | 40  | 5 | 1 | 2 | 1 | 91  | 10.99 |
| LA  | D | 25 | DARRYL SYDOR         | 84 | 8  | 27 | 35 | -9  | 94  | 1 | 0 | 0 | 0 | 146 | 5.48  |
| ANA | L | 16 | PETER DOURIS         | 74 | 12 | 22 | 34 | -5  | 21  | 1 | 0 | 1 | 0 | 142 | 8.45  |
| TB  | D | 22 | SHAWN CHAMBERS       | 66 | 11 | 23 | 34 | -6  | 23  | 6 | 1 | 1 | 0 | 142 | 7.75  |
| PIT | D | 4  | GREG HAWGOOD         | 64 | 6  | 28 | 34 | 9   | 36  | 4 | 0 | 2 | 0 | 112 | 5.36  |
| NYR | C | 14 | CRAIG MACTAVISH      | 78 | 20 | 13 | 33 | -14 | 91  | 1 | 0 | 2 | 1 | 122 | 16.39 |
| QUE | R | 51 | ANDREI KOVALENKO     | 58 | 16 | 17 | 33 | -5  | 46  | 5 | 0 | 4 | 0 | 92  | 17.39 |
| CHI | R | 33 | DIRK GRAHAM          | 67 | 15 | 18 | 33 | 13  | 45  | 0 | 2 | 5 | 0 | 122 | 12.30 |
| MON | C | 15 | PAUL DI PIETRO       | 70 | 13 | 20 | 33 | -2  | 37  | 2 | 0 | 0 | 0 | 115 | 11.30 |
| NJ  | L | 16 | BOBBY HOLIK          | 70 | 13 | 20 | 33 | 28  | 72  | 2 | 0 | 3 | 0 | 130 | 10.00 |
| DET | D | 16 | VLAD. KONSTANTINOV   | 80 | 12 | 21 | 33 | 30  | 138 | 1 | 3 | 3 | 0 | 97  | 12.37 |
| NJ  | L | 19 | BOB CARPENTER        | 76 | 10 | 23 | 33 | 7   | 51  | 0 | 2 | 1 | 0 | 125 | 8.00  |
| ANA | C | 22 | SHAUN VAN ALLEN      | 80 | 8  | 25 | 33 | 0   | 64  | 2 | 2 | 1 | 0 | 104 | 7.69  |
| FLO | R | 21 | TOM FITZGERALD       | 83 | 18 | 14 | 32 | -3  | 54  | 0 | 3 | 1 | 0 | 144 | 12.50 |
| EDM | R | 12 | STEVEN RICE          | 63 | 17 | 15 | 32 | -10 | 36  | 6 | 0 | 1 | 1 | 129 | 13.18 |
| QUE | L | 25 | MARTIN RUCINSKY      | 60 | 9  | 23 | 32 | 4   | 58  | 4 | 0 | 1 | 0 | 96  | 9.38  |
| NJ  | D | 23 | BRUCE DRIVER         | 66 | 8  | 24 | 32 | 29  | 63  | 3 | 1 | 0 | 1 | 109 | 7.34  |
| VAN | D | 44 | DAVE BABYCH          | 73 | 4  | 28 | 32 | 0   | 52  | 0 | 0 | 2 | 0 | 96  | 4.17  |
| BOS | R | 44 | GLEN MURRAY          | 81 | 18 | 13 | 31 | -1  | 48  | 0 | 0 | 4 | 2 | 114 | 15.79 |
| DAL | C | 41 | BRENT GILCHRIST      | 76 | 17 | 14 | 31 | 0   | 31  | 3 | 1 | 5 | 0 | 103 | 16.50 |
| LA  | R | 19 | JOHN DRUCE           | 55 | 14 | 17 | 31 | 16  | 50  | 1 | 1 | 0 | 0 | 104 | 13.46 |
| STL | D | 28 | STEVE DUCHESNE       | 36 | 12 | 19 | 31 | 1   | 14  | 8 | 0 | 1 | 0 | 115 | 10.43 |
| DAL | D | 2  | DERIAN HATCHER       | 83 | 12 | 19 | 31 | 19  | 211 | 2 | 1 | 2 | 0 | 132 | 9.09  |
| TB  | D | 23 | CHRIS JOSEPH         | 76 | 11 | 20 | 31 | -21 | 136 | 8 | 0 | 0 | 0 | 179 | 6.15  |
| LA  | D | 33 | MARTY MCSORLEY       | 65 | 7  | 24 | 31 | -12 | 194 | 1 | 0 | 1 | 1 | 160 | 4.38  |
| EDM | D | 20 | *BORIS MIRONOV       | 79 | 7  | 24 | 31 | -33 | 110 | 5 | 0 | 0 | 1 | 145 | 4.83  |
| CHI | L | 16 | MICHEL GOULET        | 56 | 16 | 14 | 30 | 1   | 26  | 3 | 0 | 6 | 0 | 120 | 13.33 |
| TOR | C | 19 | JOHN CULLEN          | 53 | 13 | 17 | 30 | -2  | 67  | 2 | 0 | 4 | 1 | 80  | 16.25 |
| SJ  | L | 12 | BOB ERREY            | 64 | 12 | 18 | 30 | -11 | 126 | 5 | 0 | 2 | 0 | 89  | 13.48 |
| WAS | L | 21 | TODD KRYGIER         | 66 | 12 | 18 | 30 | -4  | 60  | 0 | 1 | 3 | 0 | 146 | 8.22  |
| TOR | R | 12 | ROB PEARSON          | 67 | 12 | 18 | 30 | -6  | 189 | 1 | 0 | 4 | 0 | 119 | 10.08 |
| SJ  | L | 11 | GAETAN DUCHESNE      | 84 | 12 | 18 | 30 | 8   | 28  | 0 | 1 | 3 | 0 | 121 | 9.92  |
| ANA | C | 19 | ANATOLI SEMENOV      | 49 | 11 | 19 | 30 | -4  | 12  | 4 | 0 | 2 | 0 | 103 | 10.68 |

|     |   |    |                    |    |    |    |    |     |     |   |   |   |   |     |       |
|-----|---|----|--------------------|----|----|----|----|-----|-----|---|---|---|---|-----|-------|
| NYI | R | 15 | BRAD DALGARNO      | 73 | 11 | 19 | 30 | 14  | 62  | 3 | 0 | 1 | 0 | 97  | 11.34 |
| MON | L | 22 | BENOIT BRUNET      | 71 | 10 | 20 | 30 | 14  | 20  | 0 | 3 | 1 | 0 | 92  | 10.87 |
| CAL | C | 11 | KELLY KISIO        | 51 | 7  | 23 | 30 | -6  | 28  | 1 | 0 | 1 | 0 | 62  | 11.29 |
| FLO | D | 7  | BRIAN BENNING      | 73 | 6  | 24 | 30 | -7  | 107 | 2 | 0 | 0 | 0 | 112 | 5.36  |
| PHI | D | 2  | DIMITRI YUSHKEVICH | 75 | 5  | 25 | 30 | -8  | 86  | 1 | 0 | 2 | 0 | 136 | 3.68  |
| HAR | D | 44 | *CHRIS PRONGER     | 81 | 5  | 25 | 30 | -3  | 113 | 2 | 0 | 0 | 0 | 174 | 2.87  |
| TOR | D | 34 | JAMIE MACOUN       | 82 | 3  | 27 | 30 | -5  | 115 | 1 | 0 | 1 | 0 | 122 | 2.46  |
| VAN | L | 29 | GINO ODJICK        | 76 | 16 | 13 | 29 | 13  | 271 | 4 | 0 | 5 | 0 | 121 | 13.22 |
| NJ  | C | 20 | ALEXANDER SEMAK    | 54 | 12 | 17 | 29 | 6   | 22  | 2 | 2 | 2 | 0 | 88  | 13.64 |
| HAR | L | 26 | BRIAN PROPP        | 65 | 12 | 17 | 29 | 3   | 44  | 3 | 1 | 2 | 0 | 108 | 11.11 |
| WAS | C | 14 | *PAT PEAKE         | 49 | 11 | 18 | 29 | 1   | 39  | 3 | 0 | 1 | 1 | 91  | 12.09 |
| CHI | C | 22 | CHRISTIAN RUUTTU   | 54 | 9  | 20 | 29 | -4  | 68  | 1 | 1 | 1 | 0 | 96  | 9.38  |
| CAL | R | 22 | RONNIE STERN       | 71 | 9  | 20 | 29 | 6   | 243 | 0 | 1 | 3 | 0 | 105 | 8.57  |
| DET | C | 12 | MIKE SILLINGER     | 62 | 8  | 21 | 29 | 2   | 10  | 0 | 1 | 1 | 0 | 91  | 8.79  |
| FLO | L | 18 | MIKE HOUGH         | 78 | 6  | 23 | 29 | 3   | 62  | 0 | 1 | 1 | 0 | 106 | 5.66  |
| PIT | D | 5  | ULF SAMUELSSON     | 80 | 5  | 24 | 29 | 23  | 199 | 1 | 0 | 0 | 1 | 106 | 4.72  |
| LA  | L | 15 | PAT CONACHER       | 77 | 15 | 13 | 28 | 0   | 71  | 0 | 3 | 1 | 0 | 98  | 15.31 |
| VAN | L | 23 | MARTIN GELINAS     | 64 | 14 | 14 | 28 | -8  | 34  | 3 | 0 | 1 | 2 | 107 | 13.08 |
| CHI | L | 44 | PATRICK POULIN     | 67 | 14 | 14 | 28 | -8  | 51  | 2 | 0 | 3 | 0 | 96  | 14.58 |
| TB  | R | 15 | PAT ELYNUIK        | 67 | 13 | 15 | 28 | -21 | 64  | 4 | 1 | 1 | 0 | 111 | 11.71 |
| QUE | C | 28 | BOB BASSEN         | 83 | 13 | 15 | 28 | -17 | 99  | 1 | 1 | 0 | 1 | 129 | 10.08 |
| QUE | C | 47 | CLAUDE LAPOINTE    | 59 | 11 | 17 | 28 | 2   | 70  | 1 | 1 | 1 | 0 | 73  | 15.07 |
| VAN | C | 17 | JIMMY CARSON       | 59 | 11 | 17 | 28 | -15 | 24  | 3 | 0 | 1 | 0 | 129 | 8.53  |
| EDM | D | 10 | ILYA BYAKIN        | 44 | 8  | 20 | 28 | -3  | 30  | 6 | 0 | 3 | 0 | 51  | 15.69 |
| CHI | D | 2  | ERIC WEINRICH      | 62 | 4  | 24 | 28 | 1   | 35  | 2 | 0 | 2 | 0 | 115 | 3.48  |
| HAR | L | 23 | JOCELYN LEMIEUX    | 82 | 18 | 9  | 27 | -3  | 82  | 0 | 0 | 2 | 1 | 151 | 11.92 |
| VAN | L | 27 | SERGIO MOMESSO     | 68 | 14 | 13 | 27 | -2  | 149 | 4 | 0 | 1 | 0 | 112 | 12.50 |
| MON | R | 6  | *OLEG PETROV       | 55 | 12 | 15 | 27 | 7   | 2   | 1 | 0 | 1 | 1 | 107 | 11.21 |
| NJ  | R | 21 | RANDY MCKAY        | 78 | 12 | 15 | 27 | 24  | 244 | 0 | 0 | 1 | 1 | 77  | 15.58 |
| HAR | L | 14 | PAUL RANHEIM       | 82 | 10 | 17 | 27 | -18 | 22  | 0 | 2 | 2 | 0 | 131 | 7.63  |
| ANA | D | 6  | SEAN HILL          | 68 | 7  | 20 | 27 | -12 | 78  | 2 | 1 | 1 | 0 | 165 | 4.24  |
| EDM | C | 37 | *DEAN MCAMMOND     | 45 | 6  | 21 | 27 | 12  | 16  | 2 | 0 | 0 | 0 | 52  | 11.54 |
| CHI | D | 5  | STEVE SMITH        | 57 | 5  | 22 | 27 | -5  | 174 | 1 | 0 | 1 | 0 | 89  | 5.62  |
| TOR | D | 23 | TODD GILL          | 45 | 4  | 23 | 27 | 8   | 44  | 2 | 0 | 1 | 0 | 74  | 5.41  |
| ANA | C | 47 | STEPHAN LEBEAU     | 56 | 15 | 11 | 26 | -4  | 22  | 6 | 0 | 3 | 1 | 98  | 15.31 |
| FLO | R | 12 | JODY HULL          | 69 | 13 | 13 | 26 | 6   | 8   | 0 | 1 | 5 | 1 | 100 | 13.00 |
| WAS | C | 22 | STEVE KONOWALCHUK  | 62 | 12 | 14 | 26 | 9   | 33  | 0 | 0 | 0 | 0 | 63  | 19.05 |
| CHI | R | 15 | RICH SUTTER        | 83 | 12 | 14 | 26 | -8  | 108 | 0 | 0 | 2 | 0 | 122 | 9.84  |
| OTT | L | 61 | SYLVAIN TURGEON    | 47 | 11 | 15 | 26 | -25 | 52  | 7 | 0 | 2 | 2 | 116 | 9.48  |
| FLO | C | 44 | *ROB NIEDERMAYER   | 65 | 9  | 17 | 26 | -11 | 51  | 3 | 0 | 2 | 0 | 67  | 13.43 |
| DET | R | 25 | *DARREN MCCARTY    | 67 | 9  | 17 | 26 | 12  | 181 | 0 | 0 | 2 | 0 | 81  | 11.11 |
| WIN | D | 4  | STEPHANE QUINTAL   | 81 | 8  | 18 | 26 | -25 | 119 | 1 | 1 | 1 | 0 | 154 | 5.19  |
| BUF | R | 18 | WAYNE PRESLEY      | 65 | 17 | 8  | 25 | 18  | 103 | 1 | 5 | 1 | 0 | 93  | 18.28 |
| STL | L | 25 | VITALI PROKHOROV   | 55 | 15 | 10 | 25 | -6  | 20  | 3 | 0 | 1 | 0 | 85  | 17.65 |
| TB  | L | 34 | MIKAEL ANDERSSON   | 76 | 13 | 12 | 25 | 8   | 23  | 1 | 1 | 2 | 1 | 136 | 9.56  |
| EDM | C | 25 | MIKE STAPLETON     | 81 | 12 | 13 | 25 | -5  | 46  | 4 | 0 | 0 | 1 | 102 | 11.76 |
| BUF | C | 20 | BOB SWEENEY        | 60 | 11 | 14 | 25 | 3   | 94  | 3 | 3 | 1 | 0 | 76  | 14.47 |
| TOR | L | 21 | MARK OSBORNE       | 73 | 9  | 16 | 25 | 2   | 145 | 1 | 1 | 2 | 0 | 103 | 8.74  |
| STL | C | 36 | PHILIPPE BOZON     | 80 | 9  | 16 | 25 | 4   | 42  | 0 | 1 | 1 | 0 | 118 | 7.63  |
| WAS | C | 9  | DAVE POULIN        | 63 | 6  | 19 | 25 | -1  | 52  | 0 | 1 | 0 | 0 | 64  | 9.38  |
| SJ  | D | 41 | TOM PEDERSON       | 74 | 6  | 19 | 25 | 3   | 31  | 3 | 0 | 1 | 1 | 185 | 3.24  |
| QUE | D | 5  | ALEXEI GUSAROV     | 76 | 5  | 20 | 25 | 3   | 38  | 0 | 1 | 0 | 0 | 84  | 5.95  |
| BOS | L | 18 | BRENT HUGHES       | 77 | 13 | 11 | 24 | 10  | 143 | 1 | 0 | 1 | 0 | 100 | 13.00 |
| CHI | L | 19 | RANDY CUNNEYWORTH  | 79 | 13 | 11 | 24 | -1  | 100 | 0 | 1 | 2 | 0 | 154 | 8.44  |

|     |   |    |                    |    |    |    |    |     |     |   |   |   |   |     |       |
|-----|---|----|--------------------|----|----|----|----|-----|-----|---|---|---|---|-----|-------|
| DAL | R | 21 | PAUL BROTEN        | 64 | 12 | 12 | 24 | 18  | 30  | 0 | 0 | 3 | 0 | 76  | 15.79 |
| DAL | D | 24 | MARK TINORDI       | 61 | 6  | 18 | 24 | 6   | 143 | 1 | 0 | 0 | 0 | 112 | 5.36  |
| DET | D | 4  | MARK HOWE          | 44 | 4  | 20 | 24 | 16  | 8   | 1 | 0 | 0 | 0 | 72  | 5.56  |
| STL | D | 7  | ALEXEI KASATONOV   | 63 | 4  | 20 | 24 | -3  | 62  | 1 | 0 | 1 | 0 | 87  | 4.60  |
| BUF | D | 41 | KEN SUTTON         | 78 | 4  | 20 | 24 | -6  | 71  | 1 | 0 | 0 | 0 | 95  | 4.21  |
| SJ  | C | 9  | V. BUTSAYEV        | 59 | 12 | 11 | 23 | 0   | 68  | 2 | 0 | 3 | 0 | 85  | 14.12 |
| ANA | C | 21 | *PATRIK CARNBACK   | 73 | 12 | 11 | 23 | -8  | 54  | 3 | 0 | 2 | 0 | 81  | 14.81 |
| PHI | C | 20 | ROB DIMAIO         | 53 | 11 | 12 | 23 | -4  | 46  | 2 | 0 | 2 | 0 | 81  | 13.58 |
| CAL | C | 29 | JOEL OTTO          | 81 | 11 | 12 | 23 | -17 | 92  | 3 | 1 | 1 | 0 | 108 | 10.19 |
| BOS | R | 22 | *JOZEF STUMPEL     | 59 | 8  | 15 | 23 | 4   | 14  | 0 | 0 | 1 | 0 | 62  | 12.90 |
| OTT | L | 18 | TROY MALLETT       | 82 | 7  | 16 | 23 | -33 | 166 | 0 | 0 | 0 | 0 | 100 | 7.00  |
| BOS | L | 17 | DAVE REID          | 83 | 6  | 17 | 23 | 10  | 25  | 0 | 2 | 1 | 0 | 145 | 4.14  |
| WIN | D | 27 | TEPPO NUMMINEN     | 57 | 5  | 18 | 23 | -23 | 28  | 4 | 0 | 1 | 0 | 89  | 5.62  |
| OTT | D | 4  | BRAD SHAW          | 66 | 4  | 19 | 23 | -41 | 59  | 1 | 0 | 0 | 0 | 113 | 3.54  |
| OTT | D | 22 | NORM MACIVER       | 53 | 3  | 20 | 23 | -26 | 26  | 0 | 0 | 0 | 0 | 88  | 3.41  |
| MON | D | 43 | PATRICE BRISEBOIS  | 53 | 2  | 21 | 23 | 5   | 63  | 1 | 0 | 0 | 0 | 71  | 2.82  |
| CAL | D | 7  | MICHEL PETIT       | 63 | 2  | 21 | 23 | 5   | 110 | 0 | 0 | 0 | 0 | 103 | 1.94  |
| DET | L | 11 | SHAWN BURR         | 51 | 10 | 12 | 22 | 12  | 31  | 0 | 1 | 1 | 0 | 64  | 15.63 |
| LA  | C | 12 | KEVIN TODD         | 47 | 8  | 14 | 22 | -3  | 24  | 4 | 0 | 1 | 0 | 65  | 12.31 |
| STL | D | 6  | PHIL HOUSLEY       | 26 | 7  | 15 | 22 | -5  | 12  | 4 | 0 | 1 | 1 | 60  | 11.67 |
| QUE | D | 7  | CURTIS LESCHYSHYN  | 72 | 5  | 17 | 22 | -2  | 65  | 3 | 0 | 2 | 0 | 97  | 5.15  |
| HAR | D | 5  | ALEXANDER GODYNYUK | 69 | 3  | 19 | 22 | 13  | 75  | 0 | 0 | 1 | 0 | 110 | 2.73  |
| CAL | D | 39 | DAN KECZMER        | 69 | 1  | 21 | 22 | -8  | 60  | 0 | 0 | 0 | 0 | 116 | 0.86  |
| SJ  | R | 36 | JEFF ODGERS        | 81 | 13 | 8  | 21 | -13 | 222 | 7 | 0 | 0 | 1 | 73  | 17.81 |
| BOS | R | 23 | STEPHEN HEINZE     | 77 | 10 | 11 | 21 | -2  | 32  | 0 | 2 | 1 | 0 | 183 | 5.46  |
| STL | L | 12 | *VITALI KARAMNOV   | 59 | 9  | 12 | 21 | -3  | 51  | 2 | 0 | 1 | 0 | 66  | 13.64 |
| NYI | D | 4  | UWE KRUPP          | 41 | 7  | 14 | 21 | 11  | 30  | 3 | 0 | 0 | 0 | 82  | 8.54  |
| LA  | L | 21 | TONY GRANATO       | 50 | 7  | 14 | 21 | -2  | 150 | 2 | 0 | 0 | 0 | 117 | 5.98  |
| CAL | D | 18 | TRENT YAWNEY       | 58 | 6  | 15 | 21 | 21  | 60  | 1 | 1 | 1 | 1 | 62  | 9.68  |
| BOS | D | 32 | DON SWEENEY        | 75 | 6  | 15 | 21 | 29  | 50  | 1 | 2 | 2 | 0 | 136 | 4.41  |
| BUF | C | 14 | DAVE HANNAN        | 83 | 6  | 15 | 21 | 10  | 53  | 0 | 3 | 1 | 0 | 40  | 15.00 |
| HAR | D | 4  | FRANTISEK KUCERA   | 76 | 5  | 16 | 21 | -3  | 48  | 3 | 0 | 0 | 0 | 122 | 4.10  |
| WIN | D | 3  | DAVE MANSON        | 70 | 4  | 17 | 21 | -14 | 191 | 1 | 0 | 0 | 0 | 180 | 2.22  |
| TB  | D | 44 | ROMAN HAMRLIK      | 64 | 3  | 18 | 21 | -14 | 135 | 0 | 0 | 0 | 0 | 158 | 1.90  |
| EDM | R | 16 | KELLY BUCHBERGER   | 84 | 3  | 18 | 21 | -20 | 199 | 0 | 0 | 0 | 0 | 93  | 3.23  |
| OTT | D | 6  | GORD DINEEN        | 77 | 0  | 21 | 21 | -52 | 89  | 0 | 0 | 0 | 0 | 62  | 0.00  |
| TB  | C | 10 | ADAM CREIGHTON     | 53 | 10 | 10 | 20 | -7  | 37  | 2 | 0 | 1 | 0 | 77  | 12.99 |
| ANA | D | 2  | BOBBY DOLLAS       | 77 | 9  | 11 | 20 | 20  | 55  | 1 | 0 | 1 | 0 | 121 | 7.44  |
| OTT | R | 11 | EVGENY DAVYDOV     | 61 | 7  | 13 | 20 | -9  | 46  | 1 | 0 | 0 | 1 | 66  | 10.61 |
| STL | C | 93 | PETR NEDVED        | 19 | 6  | 14 | 20 | 2   | 8   | 2 | 0 | 0 | 1 | 63  | 9.52  |
| STL | C | 10 | *JIM MONTGOMERY    | 67 | 6  | 14 | 20 | -1  | 44  | 0 | 0 | 1 | 0 | 67  | 8.96  |
| VAN | D | 5  | DANA MURZYN        | 80 | 6  | 14 | 20 | 4   | 109 | 0 | 1 | 0 | 0 | 79  | 7.59  |
| ANA | D | 4  | DAVID WILLIAMS     | 56 | 5  | 15 | 20 | 8   | 42  | 2 | 0 | 0 | 0 | 74  | 6.76  |
| NJ  | L | 8  | MIKE PELUSO        | 69 | 4  | 16 | 20 | 19  | 238 | 0 | 0 | 0 | 0 | 44  | 9.09  |
| DAL | C | 6  | PELLE EKLUND       | 53 | 3  | 17 | 20 | -2  | 10  | 0 | 0 | 0 | 0 | 53  | 5.66  |
| ANA | L | 24 | TROY LONEY         | 62 | 13 | 6  | 19 | -5  | 88  | 6 | 0 | 1 | 0 | 93  | 13.98 |
| BOS | L | 10 | DMITRI KVARTALNOV  | 39 | 12 | 7  | 19 | -9  | 10  | 4 | 0 | 0 | 0 | 68  | 17.65 |
| OTT | R | 20 | ANDREW MCBAIN      | 55 | 11 | 8  | 19 | -41 | 64  | 8 | 0 | 0 | 0 | 91  | 12.09 |
| EDM | R | 18 | *KIRK MALTBY       | 68 | 11 | 8  | 19 | -2  | 74  | 0 | 1 | 1 | 0 | 89  | 12.36 |
| LA  | L | 10 | WARREN RYCHEL      | 80 | 10 | 9  | 19 | -19 | 322 | 0 | 0 | 3 | 0 | 105 | 9.52  |
| LA  | C | 13 | *ROBERT LANG       | 32 | 9  | 10 | 19 | 7   | 10  | 0 | 0 | 0 | 0 | 41  | 21.95 |
| OTT | C | 26 | *SCOTT LEVINS      | 62 | 8  | 11 | 19 | -26 | 162 | 4 | 0 | 1 | 0 | 77  | 10.39 |
| WIN | R | 20 | TIE DOMI           | 81 | 8  | 11 | 19 | -8  | 347 | 0 | 0 | 1 | 0 | 98  | 8.16  |
| TOR | L | 10 | BILL BERG          | 83 | 8  | 11 | 19 | -3  | 93  | 0 | 0 | 1 | 0 | 99  | 8.08  |

|     |   |    |                  |    |    |    |    |     |     |   |   |   |   |     |       |
|-----|---|----|------------------|----|----|----|----|-----|-----|---|---|---|---|-----|-------|
| NYR | D | 4  | KEVIN LOWE       | 71 | 5  | 14 | 19 | 4   | 70  | 0 | 0 | 1 | 0 | 50  | 10.00 |
| HAR | C | 89 | DARREN TURCOTTE  | 32 | 4  | 15 | 19 | -13 | 17  | 0 | 0 | 0 | 0 | 60  | 6.67  |
| EDM | L | 23 | VLADIMIR VUJTEK  | 40 | 4  | 15 | 19 | -7  | 14  | 1 | 0 | 0 | 0 | 66  | 6.06  |
| QUE | D | 55 | GARTH BUTCHER    | 77 | 4  | 15 | 19 | -7  | 143 | 0 | 2 | 1 | 0 | 66  | 6.06  |
| NJ  | D | 6  | TOMMY ALBELIN    | 62 | 2  | 17 | 19 | 20  | 36  | 1 | 0 | 1 | 0 | 62  | 3.23  |
| EDM | L | 34 | *BRENT GRIEVE    | 27 | 13 | 5  | 18 | 4   | 21  | 4 | 0 | 0 | 1 | 54  | 24.07 |
| OTT | C | 15 | DAVID ARCHIBALD  | 33 | 10 | 8  | 18 | -7  | 14  | 2 | 0 | 1 | 0 | 65  | 15.38 |
| ANA | R | 36 | TODD EWEN        | 76 | 9  | 9  | 18 | -7  | 272 | 0 | 0 | 2 | 0 | 59  | 15.25 |
| TOR | C | 32 | MIKE EASTWOOD    | 54 | 8  | 10 | 18 | 2   | 28  | 1 | 0 | 2 | 0 | 41  | 19.51 |
| CHI | D | 20 | GARY SUTER       | 41 | 6  | 12 | 18 | -12 | 38  | 4 | 1 | 0 | 0 | 86  | 6.98  |
| HAR | C | 17 | *TED DRURY       | 50 | 6  | 12 | 18 | -15 | 36  | 0 | 1 | 1 | 1 | 80  | 7.50  |
| BUF | C | 16 | PAT LAFONTAINE   | 16 | 5  | 13 | 18 | -4  | 2   | 1 | 0 | 0 | 0 | 40  | 12.50 |
| OTT | C | 38 | VLADIMIR RUZICKA | 42 | 5  | 13 | 18 | -21 | 14  | 4 | 0 | 0 | 1 | 64  | 7.81  |
| WIN | C | 38 | LUCIANO BORSATO  | 75 | 5  | 13 | 18 | -11 | 28  | 1 | 1 | 2 | 0 | 65  | 7.69  |
| LA  | D | 22 | CHARLIE HUDDY    | 79 | 5  | 13 | 18 | 4   | 71  | 1 | 0 | 0 | 0 | 134 | 3.73  |
| OTT | D | 5  | KERRY HUFFMAN    | 62 | 4  | 14 | 18 | -28 | 40  | 2 | 1 | 0 | 1 | 112 | 3.57  |
| NYR | D | 25 | *A. KARPOVTSEV   | 67 | 3  | 15 | 18 | 12  | 58  | 1 | 0 | 1 | 0 | 78  | 3.85  |
| HAR | D | 6  | ADAM BURT        | 63 | 1  | 17 | 18 | -4  | 75  | 0 | 0 | 0 | 0 | 91  | 1.10  |
| SJ  | C | 13 | JAMIE BAKER      | 65 | 12 | 5  | 17 | 2   | 38  | 0 | 0 | 2 | 0 | 68  | 17.65 |
| DET | R | 24 | BOB PROBERT      | 66 | 7  | 10 | 17 | -1  | 275 | 1 | 0 | 0 | 0 | 105 | 6.67  |
| DET | C | 23 | *GREG JOHNSON    | 52 | 6  | 11 | 17 | -7  | 22  | 1 | 1 | 0 | 0 | 48  | 12.50 |
| QUE | D | 59 | *DAVE KARPA      | 60 | 5  | 12 | 17 | 0   | 148 | 2 | 0 | 0 | 0 | 48  | 10.42 |
| QUE | D | 29 | STEVEN FINN      | 80 | 4  | 13 | 17 | -9  | 159 | 0 | 0 | 1 | 0 | 74  | 5.41  |
| NJ  | D | 5  | *JAROSLAV MODRY  | 41 | 2  | 15 | 17 | 10  | 18  | 2 | 0 | 0 | 0 | 35  | 5.71  |
| WAS | D | 29 | JOE REEKIE       | 85 | 1  | 16 | 17 | 15  | 156 | 0 | 0 | 0 | 0 | 98  | 1.02  |
| WIN | D | 5  | IGOR ULANOV      | 74 | 0  | 17 | 17 | -11 | 165 | 0 | 0 | 0 | 0 | 46  | 0.00  |
| BUF | L | 22 | CRAIG SIMPSON    | 22 | 8  | 8  | 16 | -3  | 8   | 2 | 0 | 2 | 0 | 28  | 28.57 |
| TOR | C | 25 | PETER ZEZEL      | 41 | 8  | 8  | 16 | 5   | 19  | 0 | 0 | 0 | 0 | 47  | 17.02 |
| DET | R | 20 | *MARTIN LAPOINTE | 50 | 8  | 8  | 16 | 7   | 55  | 2 | 0 | 0 | 0 | 45  | 17.78 |
| NYR | D | 23 | JEFF BEUKEBOOM   | 68 | 8  | 8  | 16 | 18  | 170 | 1 | 0 | 0 | 0 | 58  | 13.79 |
| WAS | D | 28 | *JOHN SLANEY     | 47 | 7  | 9  | 16 | 3   | 27  | 3 | 0 | 1 | 0 | 70  | 10.00 |
| TOR | L | 18 | KENT MANDERVILLE | 67 | 7  | 9  | 16 | 5   | 63  | 0 | 0 | 1 | 0 | 81  | 8.64  |
| QUE | D | 6  | CRAIG WOLANIN    | 63 | 6  | 10 | 16 | 16  | 80  | 0 | 0 | 0 | 0 | 78  | 7.69  |
| HAR | L | 24 | *JIM STORM       | 68 | 6  | 10 | 16 | 4   | 27  | 1 | 0 | 0 | 0 | 84  | 7.14  |
| STL | C | 38 | IGOR KOROLEV     | 73 | 6  | 10 | 16 | -12 | 40  | 0 | 0 | 1 | 0 | 93  | 6.45  |
| STL | C | 26 | PETER STASTNY    | 17 | 5  | 11 | 16 | -2  | 4   | 2 | 0 | 1 | 0 | 30  | 16.67 |
| TOR | D | 3  | BOB ROUSE        | 63 | 5  | 11 | 16 | 8   | 101 | 1 | 1 | 0 | 0 | 77  | 6.49  |
| WIN | D | 6  | WAYNE MCBEAN     | 50 | 3  | 13 | 16 | -34 | 40  | 2 | 0 | 0 | 0 | 114 | 2.63  |
| CHI | C | 11 | *JEFF SHANTZ     | 52 | 3  | 13 | 16 | -14 | 30  | 0 | 0 | 0 | 0 | 56  | 5.36  |
| BUF | D | 7  | PETR SVOBODA     | 60 | 2  | 14 | 16 | 11  | 89  | 1 | 0 | 0 | 0 | 80  | 2.50  |
| TB  | D | 25 | MARC BERGEVIN    | 83 | 1  | 15 | 16 | -5  | 87  | 0 | 0 | 1 | 0 | 76  | 1.32  |
| LA  | R | 9  | DIXON WARD       | 67 | 12 | 3  | 15 | -22 | 82  | 4 | 0 | 1 | 0 | 90  | 13.33 |
| TB  | C | 28 | MARC BUREAU      | 75 | 8  | 7  | 15 | -9  | 30  | 0 | 1 | 1 | 0 | 110 | 7.27  |
| OTT | D | 34 | DARREN RUMBLE    | 70 | 6  | 9  | 15 | -50 | 116 | 0 | 0 | 0 | 0 | 95  | 6.32  |
| NJ  | C | 11 | *JIM DOWD        | 15 | 5  | 10 | 15 | 8   | 0   | 2 | 0 | 0 | 0 | 26  | 19.23 |
| BOS | R | 27 | STEPHEN LEACH    | 42 | 5  | 10 | 15 | -10 | 74  | 1 | 0 | 1 | 1 | 89  | 5.62  |
| MON | R | 26 | GARY LEEMAN      | 31 | 4  | 11 | 15 | 5   | 17  | 0 | 0 | 0 | 0 | 53  | 7.55  |
| HAR | C | 32 | IGOR CHIBIREV    | 37 | 4  | 11 | 15 | 7   | 2   | 0 | 0 | 1 | 0 | 30  | 13.33 |
| PIT | C | 19 | BRYAN TROTTIER   | 41 | 4  | 11 | 15 | -12 | 36  | 0 | 0 | 0 | 0 | 45  | 8.89  |
| HAR | D | 27 | BRYAN MARCHMENT  | 55 | 4  | 11 | 15 | -14 | 166 | 0 | 1 | 1 | 0 | 92  | 4.35  |
| PHI | C | 14 | DAVE TIPPETT     | 73 | 4  | 11 | 15 | -20 | 38  | 0 | 2 | 1 | 0 | 45  | 8.89  |
| NYR | L | 17 | GREG GILBERT     | 76 | 4  | 11 | 15 | -3  | 29  | 1 | 0 | 0 | 1 | 64  | 6.25  |
| QUE | D | 45 | *MIKE MCKEE      | 48 | 3  | 12 | 15 | 5   | 41  | 2 | 0 | 0 | 0 | 60  | 5.00  |
| VAN | D | 28 | BRIAN GLYNN      | 64 | 2  | 13 | 15 | -19 | 53  | 1 | 0 | 0 | 0 | 71  | 2.82  |

|     |   |    |                    |    |   |    |    |     |     |   |   |   |   |     |       |
|-----|---|----|--------------------|----|---|----|----|-----|-----|---|---|---|---|-----|-------|
| NJ  | D | 2  | VIACHESLAV FETISOV | 52 | 1 | 14 | 15 | 14  | 30  | 0 | 0 | 0 | 0 | 36  | 2.78  |
| VAN | R | 20 | JOSE CHARBONNEAU   | 30 | 7 | 7  | 14 | -3  | 49  | 1 | 0 | 0 | 0 | 28  | 25.00 |
| WIN | C | 15 | RANDY GILHEN       | 60 | 7 | 7  | 14 | -12 | 50  | 0 | 0 | 0 | 0 | 95  | 7.37  |
| WAS | L | 27 | CRAIG BERUBE       | 84 | 7 | 7  | 14 | -4  | 305 | 0 | 0 | 0 | 0 | 48  | 14.58 |
| QUE | L | 17 | CHRIS LINDBERG     | 37 | 6 | 8  | 14 | -1  | 12  | 0 | 0 | 0 | 0 | 42  | 14.29 |
| BUF | D | 4  | *PHILIPPE BOUCHER  | 38 | 6 | 8  | 14 | -1  | 29  | 4 | 0 | 1 | 0 | 67  | 8.96  |
| MON | R | 31 | ED RONAN           | 61 | 6 | 8  | 14 | 3   | 42  | 0 | 0 | 1 | 0 | 49  | 12.24 |
| NYI | L | 25 | DAVE VOLEK         | 32 | 5 | 9  | 14 | 0   | 10  | 2 | 0 | 0 | 0 | 56  | 8.93  |
| STL | D | 34 | MURRAY BARON       | 77 | 5 | 9  | 14 | -14 | 123 | 0 | 0 | 0 | 0 | 73  | 6.85  |
| NYI | D | 7  | SCOTT LACHANCE     | 74 | 3 | 11 | 14 | -5  | 70  | 0 | 0 | 1 | 0 | 59  | 5.08  |
| MON | D | 34 | *PETER POPOVIC     | 47 | 2 | 12 | 14 | 10  | 26  | 1 | 0 | 0 | 0 | 58  | 3.45  |
| PIT | D | 32 | PETER TAGLIANETTI  | 60 | 2 | 12 | 14 | 5   | 142 | 0 | 0 | 0 | 0 | 57  | 3.51  |
| MON | D | 48 | J.J. DAIGNEAULT    | 68 | 2 | 12 | 14 | 16  | 73  | 0 | 0 | 1 | 0 | 61  | 3.28  |
| BUF | D | 5  | CRAIG MUNI         | 82 | 2 | 12 | 14 | 31  | 66  | 0 | 1 | 2 | 0 | 45  | 4.44  |
| DAL | D | 3  | CRAIG LUDWIG       | 84 | 1 | 13 | 14 | -1  | 123 | 1 | 0 | 0 | 0 | 65  | 1.54  |
| PHI | C | 36 | *ANDRE FAUST       | 37 | 8 | 5  | 13 | -1  | 10  | 0 | 0 | 1 | 0 | 33  | 24.24 |
| BUF | L | 43 | *JASON DAWE        | 32 | 6 | 7  | 13 | 1   | 12  | 3 | 0 | 1 | 0 | 35  | 17.14 |
| TOR | R | 11 | MARK GREIG         | 44 | 6 | 7  | 13 | -5  | 41  | 0 | 0 | 0 | 0 | 55  | 10.91 |
| DET | R | 15 | SHELDON KENNEDY    | 61 | 6 | 7  | 13 | -2  | 30  | 0 | 1 | 0 | 0 | 60  | 10.00 |
| DAL | R | 27 | SHANE CHURLA       | 69 | 6 | 7  | 13 | -8  | 333 | 3 | 0 | 0 | 1 | 62  | 9.68  |
| DET | C | 33 | KRIS DRAPER        | 39 | 5 | 8  | 13 | 11  | 31  | 0 | 1 | 0 | 0 | 55  | 9.09  |
| PIT | D | 28 | KJELL SAMUELSSON   | 59 | 5 | 8  | 13 | 18  | 118 | 1 | 0 | 0 | 0 | 57  | 8.77  |
| TB  | L | 17 | GERARD GALLANT     | 51 | 4 | 9  | 13 | -6  | 74  | 1 | 0 | 2 | 0 | 45  | 8.89  |
| MON | D | 14 | KEVIN HALLER       | 68 | 4 | 9  | 13 | 3   | 118 | 0 | 0 | 1 | 0 | 72  | 5.56  |
| NYI | D | 37 | DENNIS VASKE       | 65 | 2 | 11 | 13 | 21  | 76  | 0 | 0 | 0 | 0 | 71  | 2.82  |
| BUF | D | 24 | RANDY MOLLER       | 78 | 2 | 11 | 13 | -5  | 154 | 0 | 0 | 0 | 0 | 77  | 2.60  |
| TB  | L | 7  | ROB ZAMUNER        | 59 | 6 | 6  | 12 | -9  | 42  | 0 | 0 | 1 | 0 | 109 | 5.50  |
| FLO | L | 11 | BILL LINDSAY       | 84 | 6 | 6  | 12 | -2  | 97  | 0 | 0 | 0 | 0 | 90  | 6.67  |
| DAL | L | 23 | ALAN MAY           | 51 | 5 | 7  | 12 | -3  | 115 | 0 | 0 | 1 | 0 | 40  | 12.50 |
| WIN | L | 21 | RUSS ROMANIUK      | 24 | 4 | 8  | 12 | -11 | 6   | 3 | 0 | 0 | 0 | 36  | 11.11 |
| CHI | D | 4  | KEITH CARNEY       | 37 | 4 | 8  | 12 | 14  | 39  | 0 | 0 | 0 | 0 | 37  | 10.81 |
| FLO | D | 4  | KEITH BROWN        | 51 | 4 | 8  | 12 | 11  | 60  | 1 | 0 | 0 | 0 | 52  | 7.69  |
| WIN | C | 36 | MIKE EAGLES        | 73 | 4 | 8  | 12 | -20 | 96  | 0 | 1 | 0 | 0 | 53  | 7.55  |
| WIN | L | 17 | KRIS KING          | 83 | 4 | 8  | 12 | -22 | 205 | 0 | 0 | 1 | 0 | 86  | 4.65  |
| ANA | D | 39 | DON MCSWEEN        | 32 | 3 | 9  | 12 | 4   | 39  | 1 | 0 | 2 | 0 | 43  | 6.98  |
| CHI | D | 23 | NEIL WILKINSON     | 72 | 3 | 9  | 12 | 2   | 114 | 1 | 0 | 0 | 0 | 72  | 4.17  |
| MON | C | 8  | RON WILSON         | 48 | 2 | 10 | 12 | -2  | 12  | 0 | 0 | 0 | 0 | 39  | 5.13  |
| HAR | C | 22 | MARK JANSSENS      | 84 | 2 | 10 | 12 | -13 | 137 | 0 | 0 | 0 | 0 | 52  | 3.85  |
| QUE | D | 2  | TOMMY SJODIN       | 29 | 1 | 11 | 12 | 4   | 22  | 1 | 0 | 0 | 0 | 54  | 1.85  |
| CAL | D | 5  | CHRIS DAHLQUIST    | 77 | 1 | 11 | 12 | 5   | 52  | 0 | 0 | 0 | 0 | 57  | 1.75  |
| VAN | D | 3  | BRET HEDICAN       | 69 | 0 | 12 | 12 | -7  | 64  | 0 | 0 | 0 | 0 | 88  | 0.00  |
| ANA | R | 17 | *STEVEN KING       | 36 | 8 | 3  | 11 | -7  | 44  | 3 | 0 | 1 | 0 | 50  | 16.00 |
| HAR | C | 39 | ROBERT PETROVICKY  | 33 | 6 | 5  | 11 | -1  | 39  | 1 | 0 | 0 | 0 | 33  | 18.18 |
| STL | C | 18 | TONY HRKAC         | 36 | 6 | 5  | 11 | -11 | 8   | 1 | 1 | 1 | 0 | 43  | 13.95 |
| WAS | R | 11 | TIM BERGLAND       | 54 | 6 | 5  | 11 | -15 | 10  | 0 | 0 | 0 | 0 | 65  | 9.23  |
| TOR | C | 26 | MIKE KRUSHELNYSKI  | 54 | 5 | 6  | 11 | -5  | 28  | 1 | 0 | 1 | 0 | 71  | 7.04  |
| NYR | C | 15 | MIKE HUDSON        | 48 | 4 | 7  | 11 | -5  | 47  | 0 | 0 | 1 | 0 | 48  | 8.33  |
| EDM | C | 17 | SCOTT THORNTON     | 61 | 4 | 7  | 11 | -15 | 104 | 0 | 0 | 0 | 0 | 65  | 6.15  |
| FLO | D | 24 | BRENT SEVERYN      | 67 | 4 | 7  | 11 | -1  | 156 | 1 | 0 | 1 | 0 | 93  | 4.30  |
| PIT | R | 29 | *MARKUS NASLUND    | 71 | 4 | 7  | 11 | -3  | 27  | 1 | 0 | 0 | 0 | 80  | 5.00  |
| PIT | D | 34 | GREG BROWN         | 36 | 3 | 8  | 11 | 1   | 28  | 1 | 0 | 0 | 0 | 37  | 8.11  |
| CAL | L | 12 | PAUL KRUSE         | 68 | 3 | 8  | 11 | -6  | 185 | 0 | 0 | 0 | 0 | 52  | 5.77  |
| TOR | D | 2  | SYLVAIN LEFEBVRE   | 84 | 2 | 9  | 11 | 33  | 79  | 0 | 0 | 0 | 1 | 96  | 2.08  |
| VAN | D | 4  | GERALD DIDUCK      | 55 | 1 | 10 | 11 | 2   | 72  | 0 | 0 | 0 | 0 | 50  | 2.00  |



|     |   |    |                    |    |   |    |    |     |     |   |   |   |   |     |       |
|-----|---|----|--------------------|----|---|----|----|-----|-----|---|---|---|---|-----|-------|
| NYI | D | 11 | DARIUS KASPARAITIS | 76 | 1 | 10 | 11 | -6  | 142 | 0 | 0 | 0 | 0 | 81  | 1.23  |
| QUE | D | 27 | BRAD WERENKA       | 26 | 0 | 11 | 11 | 3   | 22  | 0 | 0 | 0 | 0 | 28  | 0.00  |
| BOS | R | 33 | DAN MAROIS         | 22 | 7 | 3  | 10 | -4  | 18  | 3 | 0 | 0 | 0 | 32  | 21.88 |
| NJ  | R | 24 | *DAVID EMMA        | 15 | 5 | 5  | 10 | 0   | 2   | 1 | 0 | 2 | 0 | 24  | 20.83 |
| CAL | R | 15 | *SANDY MCCARTHY    | 79 | 5 | 5  | 10 | -3  | 173 | 0 | 0 | 0 | 0 | 39  | 12.82 |
| EDM | L | 29 | LOUIE DEBRUSK      | 48 | 4 | 6  | 10 | -9  | 185 | 0 | 0 | 0 | 0 | 27  | 14.81 |
| BOS | D | 25 | PAUL STANTON       | 71 | 3 | 7  | 10 | -7  | 54  | 1 | 0 | 1 | 0 | 136 | 2.21  |
| TOR | D | 55 | DRAKE BEREHOWSKY   | 49 | 2 | 8  | 10 | -3  | 63  | 2 | 0 | 2 | 0 | 29  | 6.90  |
| STL | D | 4  | RICK ZOMBO         | 74 | 2 | 8  | 10 | -15 | 85  | 0 | 0 | 0 | 0 | 53  | 3.77  |
| WIN | D | 26 | DEAN KENNEDY       | 76 | 2 | 8  | 10 | -22 | 164 | 0 | 0 | 1 | 0 | 38  | 5.26  |
| SJ  | D | 40 | *MIKE RATHJE       | 47 | 1 | 9  | 10 | -9  | 59  | 1 | 0 | 0 | 0 | 30  | 3.33  |
| VAN | D | 6  | ADRIEN PLAVSIC     | 47 | 1 | 9  | 10 | -5  | 6   | 0 | 0 | 1 | 0 | 41  | 2.44  |
| BOS | D | 34 | DAVID SHAW         | 55 | 1 | 9  | 10 | -11 | 85  | 0 | 0 | 0 | 0 | 107 | 0.93  |
| LA  | D | 5  | TIM WATTERS        | 60 | 1 | 9  | 10 | -11 | 67  | 0 | 1 | 0 | 0 | 38  | 2.63  |
| FLO | D | 2  | JOE CIRELLA        | 63 | 1 | 9  | 10 | 8   | 99  | 0 | 0 | 0 | 0 | 63  | 1.59  |
| NJ  | D | 3  | KEN DANEYKO        | 78 | 1 | 9  | 10 | 27  | 176 | 0 | 0 | 1 | 0 | 60  | 1.67  |
| ANA | D | 29 | RANDY LADOUCEUR    | 81 | 1 | 9  | 10 | 7   | 74  | 0 | 0 | 0 | 0 | 66  | 1.52  |
| ANA | C | 10 | JARROD SKALDE      | 20 | 5 | 4  | 9  | -3  | 10  | 2 | 0 | 2 | 0 | 25  | 20.00 |
| FLO | R | 17 | MIKE FOLIGNO       | 43 | 4 | 5  | 9  | 7   | 53  | 0 | 0 | 0 | 0 | 35  | 11.43 |
| MON | L | 20 | *PIERRE SEVIGNY    | 43 | 4 | 5  | 9  | 6   | 42  | 1 | 0 | 1 | 0 | 19  | 21.05 |
| CAL | C | 32 | MIKE SULLIVAN      | 45 | 4 | 5  | 9  | -1  | 10  | 0 | 2 | 1 | 0 | 48  | 8.33  |
| EDM | D | 35 | *ADAM BENNETT      | 48 | 3 | 6  | 9  | -8  | 49  | 1 | 0 | 0 | 0 | 57  | 5.26  |
| BOS | C | 16 | *CAMERON STEWART   | 57 | 3 | 6  | 9  | -6  | 66  | 0 | 0 | 1 | 0 | 55  | 5.45  |
| SJ  | C | 33 | DALE CRAIGWELL     | 58 | 3 | 6  | 9  | -13 | 16  | 0 | 1 | 0 | 0 | 35  | 8.57  |
| VAN | C | 15 | JOHN MCINTYRE      | 62 | 3 | 6  | 9  | -9  | 38  | 0 | 0 | 0 | 0 | 30  | 10.00 |
| TB  | D | 20 | RUDY POESCHEK      | 71 | 3 | 6  | 9  | 3   | 118 | 0 | 0 | 1 | 1 | 46  | 6.52  |
| STL | D | 32 | DOUG CROSSMAN      | 50 | 2 | 7  | 9  | 1   | 10  | 1 | 0 | 0 | 0 | 30  | 6.67  |
| NYI | C | 24 | KEITH ACTON        | 77 | 2 | 7  | 9  | -5  | 71  | 0 | 0 | 0 | 0 | 35  | 5.71  |
| NYR | D | 24 | JAY WELLS          | 79 | 2 | 7  | 9  | 4   | 110 | 0 | 0 | 0 | 0 | 64  | 3.13  |
| PHI | D | 25 | JEFF FINLEY        | 55 | 1 | 8  | 9  | 16  | 24  | 0 | 0 | 0 | 0 | 43  | 2.33  |
| BOS | D | 6  | GLEN FEATHERSTONE  | 58 | 1 | 8  | 9  | -5  | 152 | 0 | 0 | 1 | 0 | 55  | 1.82  |
| CAL | D | 3  | FRANK MUSIL        | 75 | 1 | 8  | 9  | 38  | 50  | 0 | 0 | 0 | 0 | 65  | 1.54  |
| FLO | D | 25 | GEOFF SMITH        | 77 | 1 | 8  | 9  | -13 | 50  | 0 | 0 | 0 | 0 | 67  | 1.49  |
| WAS | D | 36 | SHAWN ANDERSON     | 50 | 0 | 9  | 9  | -1  | 12  | 0 | 0 | 0 | 0 | 31  | 0.00  |
| WIN | R | 37 | JOHN LEBLANC       | 17 | 6 | 2  | 8  | -2  | 2   | 1 | 1 | 1 | 0 | 29  | 20.69 |
| HAR | R | 14 | JIM SANDLAK        | 27 | 6 | 2  | 8  | 6   | 32  | 2 | 0 | 1 | 0 | 32  | 18.75 |
| QUE | L | 12 | *CHRIS SIMON       | 37 | 4 | 4  | 8  | -2  | 132 | 0 | 0 | 1 | 0 | 39  | 10.26 |
| DET | C | 27 | *MICAHA AIVAZOFF   | 59 | 4 | 4  | 8  | -1  | 38  | 0 | 0 | 0 | 0 | 52  | 7.69  |
| TOR | L | 22 | KEN BAUMGARTNER    | 64 | 4 | 4  | 8  | -6  | 185 | 0 | 0 | 0 | 0 | 34  | 11.76 |
| EDM | L | 27 | *PETER WHITE       | 26 | 3 | 5  | 8  | 1   | 2   | 0 | 0 | 0 | 0 | 17  | 17.65 |
| EDM | C | 26 | SHJON PODEIN       | 28 | 3 | 5  | 8  | 3   | 8   | 0 | 0 | 0 | 0 | 26  | 11.54 |
| NYR | C | 12 | ED OLCZYK          | 37 | 3 | 5  | 8  | -1  | 28  | 0 | 0 | 1 | 0 | 40  | 7.50  |
| ANA | D | 3  | MARK FERNER        | 50 | 3 | 5  | 8  | -16 | 30  | 0 | 0 | 0 | 0 | 44  | 6.82  |
| NYR | L | 19 | NICK KYPREOS       | 56 | 3 | 5  | 8  | -16 | 139 | 0 | 0 | 1 | 0 | 34  | 8.82  |
| FLO | L | 23 | JEFF DANIELS       | 70 | 3 | 5  | 8  | -1  | 20  | 0 | 0 | 1 | 0 | 52  | 5.77  |
| PIT | L | 33 | JIM MCKENZIE       | 71 | 3 | 5  | 8  | -7  | 146 | 0 | 0 | 1 | 0 | 33  | 9.09  |
| CHI | C | 32 | *STEVE DUBINSKY    | 27 | 2 | 6  | 8  | 1   | 16  | 0 | 0 | 0 | 0 | 20  | 10.00 |
| QUE | D | 52 | ADAM FOOTE         | 45 | 2 | 6  | 8  | 3   | 67  | 0 | 0 | 0 | 0 | 42  | 4.76  |
| LA  | D | 29 | DONALD DUFRESNE    | 60 | 2 | 6  | 8  | -7  | 58  | 0 | 0 | 0 | 0 | 56  | 3.57  |
| LA  | D | 6  | DOUG HOUDA         | 61 | 2 | 6  | 8  | -19 | 188 | 0 | 0 | 0 | 0 | 32  | 6.25  |
| EDM | D | 22 | LUKE RICHARDSON    | 69 | 2 | 6  | 8  | -13 | 131 | 0 | 0 | 0 | 0 | 92  | 2.17  |
| STL | D | 20 | TOM TILLEY         | 48 | 1 | 7  | 8  | 3   | 32  | 0 | 0 | 0 | 0 | 41  | 2.44  |
| CHI | D | 8  | CAM RUSSELL        | 67 | 1 | 7  | 8  | 10  | 200 | 0 | 0 | 0 | 0 | 41  | 2.44  |
| OTT | C | 7  | DAN QUINN          | 13 | 7 | 0  | 7  | 0   | 6   | 2 | 0 | 3 | 0 | 31  | 22.58 |

|     |   |    |                      |    |   |   |   |     |     |   |   |   |   |    |       |
|-----|---|----|----------------------|----|---|---|---|-----|-----|---|---|---|---|----|-------|
| LA  | R | 18 | DAVE TAYLOR          | 33 | 4 | 3 | 7 | -1  | 28  | 0 | 1 | 2 | 0 | 39 | 10.26 |
| PHI | R | 15 | ALLAN CONROY         | 62 | 4 | 3 | 7 | -12 | 65  | 0 | 1 | 0 | 0 | 40 | 10.00 |
| QUE | C | 33 | *REGGIE SAVAGE       | 17 | 3 | 4 | 7 | 3   | 16  | 1 | 0 | 0 | 0 | 25 | 12.00 |
| VAN | C | 25 | *NATHAN LAFAYETTE    | 49 | 3 | 4 | 7 | -7  | 18  | 0 | 0 | 0 | 0 | 34 | 8.82  |
| LA  | C | 14 | GARY SHUCHUK         | 56 | 3 | 4 | 7 | -8  | 30  | 0 | 0 | 1 | 0 | 55 | 5.45  |
| VAN | R | 19 | TIM HUNTER           | 56 | 3 | 4 | 7 | -7  | 171 | 0 | 1 | 1 | 0 | 41 | 7.32  |
| BUF | L | 32 | ROB RAY              | 82 | 3 | 4 | 7 | 2   | 274 | 0 | 0 | 0 | 0 | 34 | 8.82  |
| DAL | R | 10 | DAVE BARR            | 20 | 2 | 5 | 7 | -6  | 21  | 0 | 0 | 0 | 0 | 20 | 10.00 |
| OTT | R | 16 | BRAD LAUER           | 30 | 2 | 5 | 7 | -15 | 6   | 0 | 1 | 0 | 0 | 45 | 4.44  |
| OTT | D | 21 | DENNIS VIAL          | 55 | 2 | 5 | 7 | -9  | 214 | 0 | 0 | 0 | 0 | 37 | 5.41  |
| STL | R | 39 | KELLY CHASE          | 68 | 2 | 5 | 7 | -5  | 278 | 0 | 0 | 0 | 0 | 57 | 3.51  |
| ANA | R | 48 | *JOHN LILLEY         | 13 | 1 | 6 | 7 | 1   | 8   | 0 | 0 | 1 | 0 | 20 | 5.00  |
| CAL | D | 21 | BRAD SCHLEGEL        | 26 | 1 | 6 | 7 | -4  | 4   | 0 | 0 | 0 | 0 | 24 | 4.17  |
| SJ  | D | 4  | JAY MORE             | 49 | 1 | 6 | 7 | -5  | 63  | 0 | 0 | 0 | 0 | 38 | 2.63  |
| BOS | D | 14 | GORDIE ROBERTS       | 59 | 1 | 6 | 7 | -13 | 40  | 0 | 0 | 0 | 0 | 19 | 5.26  |
| DET | D | 2  | TERRY CARKNER        | 68 | 1 | 6 | 7 | 13  | 130 | 0 | 0 | 0 | 0 | 32 | 3.13  |
| DET | D | 29 | SERGEI BAUTIN        | 60 | 0 | 7 | 7 | -12 | 78  | 0 | 0 | 0 | 0 | 39 | 0.00  |
| WAS | D | 2  | JIM JOHNSON          | 61 | 0 | 7 | 7 | -7  | 63  | 0 | 0 | 0 | 0 | 49 | 0.00  |
| PHI | D | 26 | ROB ZETTLER          | 75 | 0 | 7 | 7 | -26 | 134 | 0 | 0 | 0 | 0 | 55 | 0.00  |
| VAN | R | 36 | *DANE JACKSON        | 12 | 5 | 1 | 6 | 3   | 9   | 0 | 0 | 0 | 0 | 18 | 27.78 |
| CHI | R | 29 | DARIN KIMBLE         | 65 | 4 | 2 | 6 | 2   | 133 | 0 | 0 | 0 | 0 | 17 | 23.53 |
| LA  | D | 42 | DOMINIC LAVOIE       | 8  | 3 | 3 | 6 | -2  | 2   | 2 | 0 | 1 | 0 | 21 | 14.29 |
| TOR | C | 44 | *YANIC PERREAULT     | 13 | 3 | 3 | 6 | 1   | 0   | 2 | 0 | 0 | 0 | 24 | 12.50 |
| TB  | C | 11 | BILL MCDUGALL        | 22 | 3 | 3 | 6 | -4  | 8   | 1 | 0 | 0 | 1 | 26 | 11.54 |
| VAN | R | 25 | *DAN KESA            | 19 | 2 | 4 | 6 | -3  | 18  | 1 | 0 | 1 | 0 | 18 | 11.11 |
| OTT | L | 27 | PHIL BOURQUE         | 27 | 2 | 4 | 6 | -4  | 8   | 0 | 2 | 0 | 0 | 21 | 9.52  |
| OTT | C | 33 | TROY MURRAY          | 27 | 2 | 4 | 6 | 2   | 10  | 0 | 1 | 0 | 0 | 21 | 9.52  |
| BUF | L | 36 | *MATTHEW BARNABY     | 35 | 2 | 4 | 6 | -7  | 106 | 1 | 0 | 0 | 0 | 13 | 15.38 |
| PIT | D | 3  | GRANT JENNINGS       | 61 | 2 | 4 | 6 | -10 | 126 | 0 | 1 | 1 | 0 | 49 | 4.08  |
| OTT | C | 11 | JARMO KEKALAINEN     | 28 | 1 | 5 | 6 | -8  | 14  | 0 | 0 | 0 | 0 | 18 | 5.56  |
| PHI | D | 28 | *JASON BOWEN         | 56 | 1 | 5 | 6 | 12  | 87  | 0 | 0 | 1 | 0 | 50 | 2.00  |
| LA  | D | 7  | JIM PAEK             | 59 | 1 | 5 | 6 | -8  | 18  | 0 | 0 | 0 | 0 | 35 | 2.86  |
| HAR | D | 10 | BRAD MCCRIMMON       | 65 | 1 | 5 | 6 | -7  | 72  | 0 | 0 | 0 | 0 | 39 | 2.56  |
| ANA | L | 32 | STU GRIMSON          | 77 | 1 | 5 | 6 | -6  | 199 | 0 | 0 | 0 | 0 | 34 | 2.94  |
| NYI | L | 8  | DAVID MALEY          | 56 | 0 | 6 | 6 | -7  | 104 | 0 | 0 | 0 | 0 | 23 | 0.00  |
| SJ  | C | 50 | *JAROSLAV OTEVREL    | 9  | 3 | 2 | 5 | -5  | 2   | 1 | 0 | 0 | 0 | 11 | 27.27 |
| BOS | L | 48 | *FRED KNIPSCHER      | 11 | 3 | 2 | 5 | 3   | 14  | 0 | 0 | 1 | 0 | 15 | 20.00 |
| HAR | L | 20 | *KEVIN SMYTH         | 21 | 3 | 2 | 5 | -1  | 10  | 0 | 0 | 0 | 0 | 8  | 37.50 |
| DAL | R | 11 | *JARKKO VARVIO       | 8  | 2 | 3 | 5 | 1   | 4   | 0 | 0 | 1 | 0 | 17 | 11.76 |
| BUF | C | 40 | JAMES BLACK          | 15 | 2 | 3 | 5 | -4  | 2   | 2 | 0 | 0 | 0 | 18 | 11.11 |
| OTT | R | 24 | ROBERT BURAKOWSKY    | 23 | 2 | 3 | 5 | -7  | 6   | 0 | 0 | 0 | 0 | 40 | 5.00  |
| STL | L | 23 | DAVE MACKEY          | 30 | 2 | 3 | 5 | -4  | 56  | 0 | 0 | 0 | 0 | 21 | 9.52  |
| QUE | R | 23 | PAUL MACDERMID       | 44 | 2 | 3 | 5 | -3  | 35  | 0 | 0 | 0 | 0 | 16 | 12.50 |
| HAR | R | 29 | MARC POTVIN          | 54 | 2 | 3 | 5 | -8  | 272 | 0 | 0 | 0 | 0 | 26 | 7.69  |
| CHI | D | 6  | ROBERT DIRK          | 71 | 2 | 3 | 5 | 18  | 131 | 0 | 0 | 0 | 0 | 42 | 4.76  |
| SJ  | D | 38 | *MICHAL SYKORA       | 22 | 1 | 4 | 5 | -4  | 14  | 0 | 0 | 0 | 0 | 22 | 4.55  |
| DET | D | 21 | BOB HALKIDIS         | 28 | 1 | 4 | 5 | -1  | 93  | 0 | 0 | 0 | 0 | 35 | 2.86  |
| NYI | D | 47 | RICHARD PILON        | 28 | 1 | 4 | 5 | -4  | 75  | 0 | 0 | 0 | 0 | 20 | 5.00  |
| OTT | D | 55 | *DIMITRI FILIMONOV   | 30 | 1 | 4 | 5 | -10 | 18  | 0 | 0 | 0 | 0 | 15 | 6.67  |
| PHI | R | 21 | DAVE BROWN           | 71 | 1 | 4 | 5 | -12 | 137 | 0 | 0 | 0 | 0 | 16 | 6.25  |
| DAL | D | 5  | DOUG ZMOLEK          | 75 | 1 | 4 | 5 | -8  | 133 | 0 | 0 | 0 | 0 | 32 | 3.13  |
| WAS | C | 23 | *KEVIN KAMINSKI      | 13 | 0 | 5 | 5 | 2   | 87  | 0 | 0 | 0 | 0 | 9  | 0.00  |
| VAN | D | 2  | *YEVGENY NAMESTNIKOV | 17 | 0 | 5 | 5 | -2  | 10  | 0 | 0 | 0 | 0 | 11 | 0.00  |
| NJ  | D | 26 | *JASON SMITH         | 41 | 0 | 5 | 5 | 7   | 43  | 0 | 0 | 0 | 0 | 47 | 0.00  |

|     |   |    |                     |    |   |   |   |     |     |   |   |   |   |    |       |
|-----|---|----|---------------------|----|---|---|---|-----|-----|---|---|---|---|----|-------|
| PIT | R | 44 | *ED PATTERSON       | 27 | 3 | 1 | 4 | -5  | 10  | 0 | 0 | 0 | 0 | 15 | 20.00 |
| NYI | R | 12 | MICK VUKOTA         | 72 | 3 | 1 | 4 | -5  | 237 | 0 | 0 | 0 | 0 | 26 | 11.54 |
| QUE | R | 11 | OWEN NOLAN          | 6  | 2 | 2 | 4 | 2   | 8   | 0 | 0 | 0 | 0 | 15 | 13.33 |
| SJ  | C | 20 | KIP MILLER          | 11 | 2 | 2 | 4 | -1  | 6   | 0 | 0 | 0 | 0 | 21 | 9.52  |
| TOR | L | 8  | CHRIS GOVEDARIS     | 12 | 2 | 2 | 4 | 4   | 14  | 0 | 0 | 0 | 1 | 16 | 12.50 |
| MON | L | 35 | *DONALD BRASHEAR    | 14 | 2 | 2 | 4 | 0   | 34  | 0 | 0 | 0 | 0 | 15 | 13.33 |
| FLO | D | 6  | PETER ANDERSSON     | 16 | 2 | 2 | 4 | -8  | 2   | 0 | 1 | 1 | 0 | 21 | 9.52  |
| OTT | D | 9  | *DEREK MAYER        | 17 | 2 | 2 | 4 | -16 | 8   | 1 | 0 | 0 | 0 | 29 | 6.90  |
| BUF | R | 21 | *SCOTT THOMAS       | 32 | 2 | 2 | 4 | -6  | 8   | 1 | 0 | 0 | 0 | 26 | 7.69  |
| OTT | L | 28 | BILL HUARD          | 63 | 2 | 2 | 4 | -19 | 162 | 0 | 0 | 0 | 0 | 24 | 8.33  |
| PIT | D | 6  | MIKE RAMSEY         | 65 | 2 | 2 | 4 | -4  | 22  | 0 | 0 | 0 | 0 | 31 | 6.45  |
| PHI | D | 24 | *BOB WILKIE         | 10 | 1 | 3 | 4 | -2  | 8   | 0 | 0 | 0 | 0 | 10 | 10.00 |
| DAL | C | 18 | CHRIS TANCILL       | 12 | 1 | 3 | 4 | -7  | 8   | 0 | 0 | 0 | 0 | 18 | 5.56  |
| VAN | C | 23 | NEIL EISENHUT       | 13 | 1 | 3 | 4 | 0   | 21  | 0 | 0 | 0 | 0 | 13 | 7.69  |
| CAL | C | 42 | GUY LAROSE          | 17 | 1 | 3 | 4 | -5  | 14  | 0 | 0 | 0 | 0 | 12 | 8.33  |
| NYI | D | 2  | CHRIS LUONGO        | 17 | 1 | 3 | 4 | -1  | 13  | 0 | 0 | 0 | 0 | 16 | 6.25  |
| LA  | C | 26 | BRIAN MCREYNOLDS    | 20 | 1 | 3 | 4 | -2  | 4   | 0 | 0 | 0 | 0 | 10 | 10.00 |
| STL | D | 41 | *DANIEL LAPERRIERE  | 20 | 1 | 3 | 4 | -1  | 8   | 1 | 0 | 0 | 0 | 20 | 5.00  |
| SJ  | D | 26 | *VLASTIMIL KROUPA   | 27 | 1 | 3 | 4 | -6  | 20  | 0 | 0 | 0 | 0 | 16 | 6.25  |
| WIN | C | 11 | *DAVE TOMLINSON     | 31 | 1 | 3 | 4 | -12 | 24  | 0 | 0 | 0 | 0 | 29 | 3.45  |
| PHI | D | 27 | RYAN MCGILL         | 50 | 1 | 3 | 4 | -5  | 112 | 0 | 0 | 0 | 0 | 53 | 1.89  |
| PHI | D | 23 | *STEWART MALGUNAS   | 67 | 1 | 3 | 4 | 2   | 86  | 0 | 0 | 0 | 0 | 54 | 1.85  |
| BUF | L | 15 | *SERGEI PETRENKO    | 14 | 0 | 4 | 4 | -3  | 0   | 0 | 0 | 0 | 0 | 7  | 0.00  |
| WIN | D | 24 | DARRYL SHANNON      | 20 | 0 | 4 | 4 | -6  | 18  | 0 | 0 | 0 | 0 | 14 | 0.00  |
| TB  | D | 26 | *CHRIS LIPUMA       | 27 | 0 | 4 | 4 | 1   | 77  | 0 | 0 | 0 | 0 | 20 | 0.00  |
| CAL | G | 37 | *TREVOR KIDD        | 31 | 0 | 4 | 4 | 0   | 4   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| BUF | G | 31 | GRANT FUHR          | 32 | 0 | 4 | 4 | 0   | 16  | 0 | 0 | 0 | 0 | 0  | 0.00  |
| NYI | D | 3  | DEAN CHYNOWETH      | 39 | 0 | 4 | 4 | 3   | 122 | 0 | 0 | 0 | 0 | 26 | 0.00  |
| QUE | L | 15 | TONY TWIST          | 49 | 0 | 4 | 4 | -1  | 101 | 0 | 0 | 0 | 0 | 15 | 0.00  |
| VAN | G | 1  | KIRK MCLEAN         | 52 | 0 | 4 | 4 | 0   | 2   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| TOR | G | 29 | FELIX POTVIN        | 66 | 0 | 4 | 4 | 0   | 4   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| CHI | G | 30 | ED BELFOUR          | 70 | 0 | 4 | 4 | 0   | 61  | 0 | 0 | 0 | 0 | 0  | 0.00  |
| BOS | R | 19 | *MARIVSZ CZERKAWSKI | 4  | 2 | 1 | 3 | -2  | 0   | 1 | 0 | 0 | 0 | 11 | 18.18 |
| BOS | R | 11 | *SERGEI ZHOLTOK     | 24 | 2 | 1 | 3 | -7  | 2   | 1 | 0 | 0 | 0 | 25 | 8.00  |
| NYI | L | 17 | *YAN KAMINSKY       | 24 | 2 | 1 | 3 | 5   | 4   | 0 | 0 | 0 | 0 | 23 | 8.70  |
| OTT | L | 23 | CLAUDE BOIVIN       | 41 | 2 | 1 | 3 | -17 | 95  | 0 | 0 | 0 | 0 | 17 | 11.76 |
| NYR | R | 26 | JOEY KOCUR          | 71 | 2 | 1 | 3 | -9  | 129 | 0 | 0 | 0 | 0 | 43 | 4.65  |
| TB  | L | 8  | *JASON RUFF         | 6  | 1 | 2 | 3 | 2   | 2   | 0 | 0 | 0 | 0 | 14 | 7.14  |
| DET | D | 38 | *JASON YORK         | 7  | 1 | 2 | 3 | 0   | 2   | 0 | 0 | 0 | 0 | 9  | 11.11 |
| MON | D | 5  | *CHRISTIAN PROULX   | 7  | 1 | 2 | 3 | 0   | 20  | 0 | 0 | 0 | 0 | 11 | 9.09  |
| CAL | L | 19 | *VESA VIITAKOSKI    | 8  | 1 | 2 | 3 | 0   | 0   | 1 | 0 | 0 | 0 | 15 | 6.67  |
| EDM | R | 28 | *ROMAN OKSIUTA      | 10 | 1 | 2 | 3 | -1  | 4   | 0 | 0 | 0 | 0 | 18 | 5.56  |
| TB  | C | 49 | *BRENT GRETZKY      | 10 | 1 | 2 | 3 | 0   | 2   | 0 | 0 | 0 | 0 | 14 | 7.14  |
| WAS | D | 25 | JASON WOOLLEY       | 10 | 1 | 2 | 3 | 2   | 4   | 0 | 0 | 0 | 0 | 15 | 6.67  |
| MON | D | 44 | BRYAN FOGARTY       | 13 | 1 | 2 | 3 | -4  | 10  | 0 | 0 | 0 | 0 | 22 | 4.55  |
| HAR | D | 33 | *TED CROWLEY        | 21 | 1 | 2 | 3 | -1  | 10  | 1 | 0 | 0 | 0 | 28 | 3.57  |
| TB  | R | 12 | *JIM CUMMINS        | 26 | 1 | 2 | 3 | -1  | 84  | 0 | 0 | 0 | 0 | 20 | 5.00  |
| OTT | D | 27 | HANK LAMMENS        | 27 | 1 | 2 | 3 | -20 | 22  | 0 | 0 | 0 | 0 | 6  | 16.67 |
| MON | L | 32 | MARIO ROBERGE       | 28 | 1 | 2 | 3 | -2  | 55  | 0 | 0 | 0 | 0 | 5  | 20.00 |
| STL | L | 17 | BASIL MCRAE         | 40 | 1 | 2 | 3 | -7  | 103 | 0 | 0 | 0 | 0 | 23 | 4.35  |
| VAN | L | 18 | *SHAWN ANTOSKI      | 55 | 1 | 2 | 3 | -11 | 190 | 0 | 0 | 1 | 0 | 25 | 4.00  |
| TB  | D | 39 | ENRICO CICCONE      | 57 | 1 | 2 | 3 | -4  | 226 | 0 | 0 | 0 | 0 | 33 | 3.03  |
| CAL | D | 36 | *LEONARD ESAU       | 6  | 0 | 3 | 3 | -1  | 7   | 0 | 0 | 0 | 0 | 4  | 0.00  |
| CHI | R | 25 | DAVE CHRISTIAN      | 9  | 0 | 3 | 3 | 0   | 0   | 0 | 0 | 0 | 0 | 6  | 0.00  |

|     |   |    |                   |    |   |   |   |     |     |   |   |   |   |    |        |
|-----|---|----|-------------------|----|---|---|---|-----|-----|---|---|---|---|----|--------|
| HAR | D | 45 | JOHN STEVENS      | 9  | 0 | 3 | 3 | 4   | 4   | 0 | 0 | 0 | 0 | 3  | 0.00   |
| LA  | D | 24 | MARK HARDY        | 16 | 0 | 3 | 3 | -5  | 27  | 0 | 0 | 0 | 0 | 8  | 0.00   |
| DAL | D | 4  | RICHARD MATVICHUK | 25 | 0 | 3 | 3 | 1   | 22  | 0 | 0 | 0 | 0 | 18 | 0.00   |
| CAL | D | 4  | KEVIN DAHL        | 33 | 0 | 3 | 3 | -2  | 23  | 0 | 0 | 0 | 0 | 20 | 0.00   |
| HAR | D | 25 | BOB MCGILL        | 33 | 0 | 3 | 3 | -7  | 46  | 0 | 0 | 0 | 0 | 14 | 0.00   |
| DAL | D | 18 | MIKE LALOR        | 35 | 0 | 3 | 3 | -10 | 14  | 0 | 0 | 0 | 0 | 22 | 0.00   |
| OTT | L | 10 | DARCY LOEWEN      | 44 | 0 | 3 | 3 | -11 | 52  | 0 | 0 | 0 | 0 | 39 | 0.00   |
| QUE | G | 35 | STEPHANE FISET    | 50 | 0 | 3 | 3 | 0   | 8   | 0 | 0 | 0 | 0 | 0  | 0.00   |
| BUF | G | 39 | DOMINIK HASEK     | 58 | 0 | 3 | 3 | 0   | 6   | 0 | 0 | 0 | 0 | 0  | 0.00   |
| NYI | G | 72 | RON HEXTALL       | 65 | 0 | 3 | 3 | 0   | 52  | 0 | 0 | 0 | 0 | 0  | 0.00   |
| STL | G | 31 | CURTIS JOSEPH     | 71 | 0 | 3 | 3 | 0   | 4   | 0 | 0 | 0 | 0 | 0  | 0.00   |
| CAL | R | 23 | GREG PASLAWSKI    | 15 | 2 | 0 | 2 | -4  | 2   | 0 | 0 | 1 | 0 | 13 | 15.38  |
| FLO | D | 3  | *PAUL LAUS        | 39 | 2 | 0 | 2 | 9   | 109 | 0 | 0 | 1 | 0 | 15 | 13.33  |
| CAL | L | 19 | *DAVID HAAS       | 2  | 1 | 1 | 2 | 2   | 7   | 0 | 0 | 0 | 0 | 3  | 33.33  |
| FLO | L | 28 | *PATRICK LEBEAU   | 4  | 1 | 1 | 2 | 0   | 4   | 1 | 0 | 0 | 0 | 4  | 25.00  |
| QUE | R | 14 | *DWAYNE NORRIS    | 4  | 1 | 1 | 2 | 1   | 4   | 0 | 0 | 0 | 0 | 7  | 14.29  |
| TOR | D | 7  | *DAVID SACCO      | 4  | 1 | 1 | 2 | 2   | 4   | 1 | 0 | 0 | 0 | 4  | 25.00  |
| LA  | L | 37 | *DAN CURRIE       | 5  | 1 | 1 | 2 | -1  | 0   | 0 | 0 | 0 | 0 | 12 | 8.33   |
| VAN | C | 25 | STEPHANE MORIN    | 5  | 1 | 1 | 2 | 0   | 6   | 0 | 0 | 0 | 0 | 6  | 16.67  |
| PHI | D | 41 | *MILOS HOLAN      | 8  | 1 | 1 | 2 | -4  | 4   | 1 | 0 | 0 | 0 | 26 | 3.85   |
| PIT | D | 36 | *PAT NEATON       | 9  | 1 | 1 | 2 | 3   | 12  | 1 | 0 | 0 | 0 | 11 | 9.09   |
| QUE | L | 55 | *RENE CORBET      | 9  | 1 | 1 | 2 | 1   | 0   | 0 | 0 | 0 | 0 | 14 | 7.14   |
| TOR | C | 40 | KEN MCRAE         | 9  | 1 | 1 | 2 | 1   | 36  | 0 | 0 | 0 | 0 | 11 | 9.09   |
| OTT | L | 78 | *PAVOL DEMITRA    | 12 | 1 | 1 | 2 | -7  | 4   | 1 | 0 | 0 | 0 | 10 | 10.00  |
| NYI | C | 10 | CLAUDE LOISELLE   | 17 | 1 | 1 | 2 | -2  | 49  | 0 | 0 | 0 | 0 | 14 | 7.14   |
| NYR | L | 18 | MIKE HARTMAN      | 35 | 1 | 1 | 2 | -5  | 70  | 0 | 0 | 0 | 0 | 19 | 5.26   |
| CHI | D | 3  | GREG SMYTH        | 61 | 1 | 1 | 2 | -4  | 183 | 0 | 0 | 0 | 0 | 36 | 2.78   |
| NYR | D | 8  | *JOBV MESSIER     | 4  | 0 | 2 | 2 | -1  | 0   | 0 | 0 | 0 | 0 | 7  | 0.00   |
| PHI | R | 32 | CHRIS WINNES      | 4  | 0 | 2 | 2 | 1   | 0   | 0 | 0 | 0 | 0 | 4  | 0.00   |
| WIN | D | 55 | *ARTO BLOMSTEN    | 18 | 0 | 2 | 2 | -6  | 6   | 0 | 0 | 0 | 0 | 15 | 0.00   |
| PHI | D | 5  | ROB RAMAGE        | 21 | 0 | 2 | 2 | -12 | 16  | 0 | 0 | 0 | 0 | 23 | 0.00   |
| EDM | D | 6  | IAN HERBERS       | 22 | 0 | 2 | 2 | -6  | 32  | 0 | 0 | 0 | 0 | 16 | 0.00   |
| OTT | D | 24 | STEVE KONROYD     | 27 | 0 | 2 | 2 | -3  | 12  | 0 | 0 | 0 | 0 | 21 | 0.00   |
| FLO | G | 30 | MARK FITZPATRICK  | 28 | 0 | 2 | 2 | 0   | 4   | 0 | 0 | 0 | 0 | 0  | 0.00   |
| NYR | G | 30 | GLENN HEALY       | 29 | 0 | 2 | 2 | 0   | 2   | 0 | 0 | 0 | 0 | 0  | 0.00   |
| LA  | L | 23 | PHILIP CROWE      | 31 | 0 | 2 | 2 | 4   | 77  | 0 | 0 | 0 | 0 | 5  | 0.00   |
| NYR | D | 6  | DOUG LIDSTER      | 34 | 0 | 2 | 2 | -12 | 33  | 0 | 0 | 0 | 0 | 25 | 0.00   |
| SJ  | D | 44 | SHAWN CRONIN      | 34 | 0 | 2 | 2 | 2   | 76  | 0 | 0 | 0 | 0 | 14 | 0.00   |
| DAL | G | 34 | DARCY WAKALUK     | 36 | 0 | 2 | 2 | 0   | 34  | 0 | 0 | 0 | 0 | 0  | 0.00   |
| NJ  | G | 31 | CHRIS TERRERI     | 44 | 0 | 2 | 2 | 0   | 4   | 0 | 0 | 0 | 0 | 0  | 0.00   |
| BOS | G | 30 | JON CASEY         | 57 | 0 | 2 | 2 | 0   | 14  | 0 | 0 | 0 | 0 | 0  | 0.00   |
| DET | G | 35 | BOB ESSENSA       | 69 | 0 | 2 | 2 | 0   | 6   | 0 | 0 | 0 | 0 | 0  | 0.00   |
| EDM | G | 30 | BILL RANFORD      | 71 | 0 | 2 | 2 | 0   | 2   | 0 | 0 | 0 | 0 | 0  | 0.00   |
| SJ  | G | 32 | ARTURS IRBE       | 74 | 0 | 2 | 2 | 0   | 16  | 0 | 0 | 0 | 0 | 0  | 0.00   |
| DET | C | 38 | *TIM TAYLOR       | 1  | 1 | 0 | 1 | -1  | 0   | 0 | 0 | 0 | 0 | 4  | 25.00  |
| FLO | R | 9  | JAMIE LEACH       | 2  | 1 | 0 | 1 | -2  | 0   | 0 | 0 | 0 | 0 | 2  | 50.00  |
| PHI | C | 10 | *TODD HLUSHKO     | 2  | 1 | 0 | 1 | 1   | 0   | 0 | 0 | 0 | 0 | 2  | 50.00  |
| WAS | D | 40 | *TODD NELSON      | 2  | 1 | 0 | 1 | 1   | 2   | 1 | 0 | 1 | 0 | 1  | 100.00 |
| MON | C | 49 | *BRIAN SAVAGE     | 3  | 1 | 0 | 1 | 0   | 0   | 0 | 0 | 0 | 0 | 3  | 33.33  |
| NYI | L | 9  | DAVE CHYZOWSKI    | 3  | 1 | 0 | 1 | -1  | 4   | 0 | 0 | 0 | 0 | 4  | 25.00  |
| SJ  | C | 9  | GARY EMMONS       | 3  | 1 | 0 | 1 | -4  | 0   | 1 | 0 | 0 | 0 | 6  | 16.67  |
| WIN | L | 60 | *MICHAL GROSEK    | 3  | 1 | 0 | 1 | -1  | 0   | 0 | 0 | 0 | 0 | 4  | 25.00  |
| DET | D | 29 | *AARON WARD       | 5  | 1 | 0 | 1 | 2   | 4   | 0 | 0 | 0 | 0 | 3  | 33.33  |
| SJ  | C | 43 | JEFF MCLEAN       | 6  | 1 | 0 | 1 | 1   | 0   | 0 | 0 | 0 | 0 | 5  | 20.00  |

|     |   |    |                     |    |   |   |   |     |     |   |   |   |   |    |       |
|-----|---|----|---------------------|----|---|---|---|-----|-----|---|---|---|---|----|-------|
| STL | R | 9  | *DENNY FELSNER      | 6  | 1 | 0 | 1 | -1  | 2   | 0 | 0 | 0 | 0 | 6  | 16.67 |
| PIT | C | 12 | LARRY DEPALMA       | 7  | 1 | 0 | 1 | 1   | 5   | 0 | 0 | 0 | 0 | 2  | 50.00 |
| QUE | R | 54 | *ED WARD            | 7  | 1 | 0 | 1 | 0   | 5   | 0 | 0 | 0 | 0 | 3  | 33.33 |
| CHI | R | 55 | *SERGEI KRIVOKRASOV | 9  | 1 | 0 | 1 | -2  | 4   | 0 | 0 | 0 | 0 | 7  | 14.29 |
| LA  | L | 45 | *KEITH REDMOND      | 12 | 1 | 0 | 1 | -3  | 20  | 0 | 0 | 0 | 0 | 9  | 11.11 |
| NJ  | R | 14 | *BEN HANKINSON      | 13 | 1 | 0 | 1 | 0   | 23  | 0 | 0 | 1 | 0 | 14 | 7.14  |
| LA  | D | 3  | BRENT THOMPSON      | 24 | 1 | 0 | 1 | -1  | 81  | 0 | 0 | 0 | 0 | 9  | 11.11 |
| TOR | C | 20 | *ALEXEI KUDASHOV    | 25 | 1 | 0 | 1 | -3  | 4   | 0 | 0 | 0 | 0 | 24 | 4.17  |
| WAS | D | 38 | BRIAN CURRAN        | 26 | 1 | 0 | 1 | -2  | 61  | 0 | 0 | 0 | 0 | 11 | 9.09  |
| HAR | R | 15 | *TODD HARKINS       | 28 | 1 | 0 | 1 | -4  | 49  | 0 | 0 | 0 | 0 | 15 | 6.67  |
| DAL | R | 33 | MIKE NEEDHAM        | 30 | 1 | 0 | 1 | -2  | 2   | 0 | 0 | 1 | 0 | 9  | 11.11 |
| FLO | D | 41 | STEPHANE RICHER.    | 2  | 0 | 1 | 1 | -1  | 0   | 0 | 0 | 0 | 0 | 3  | 0.00  |
| MON | C | 46 | *CRAIG FERGUSON     | 2  | 0 | 1 | 1 | 1   | 0   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| WAS | C | 41 | *JASON ALLISON      | 2  | 0 | 1 | 1 | 1   | 0   | 0 | 0 | 0 | 0 | 5  | 0.00  |
| LA  | D | 50 | BOB JAY             | 3  | 0 | 1 | 1 | -2  | 0   | 0 | 0 | 0 | 0 | 2  | 0.00  |
| STL | R | 27 | DENIS CHASSE        | 3  | 0 | 1 | 1 | 1   | 15  | 0 | 0 | 0 | 0 | 5  | 0.00  |
| BOS | L | 56 | DARREN BANKS        | 4  | 0 | 1 | 1 | 0   | 9   | 0 | 0 | 0 | 0 | 3  | 0.00  |
| DET | L | 34 | STEVE MALTAIS       | 4  | 0 | 1 | 1 | -1  | 0   | 0 | 0 | 0 | 0 | 2  | 0.00  |
| EDM | C | 36 | *TODD MARCHANT      | 4  | 0 | 1 | 1 | -2  | 2   | 0 | 0 | 0 | 0 | 6  | 0.00  |
| FLO | L | 29 | JEFF GREENLAW       | 4  | 0 | 1 | 1 | -1  | 2   | 0 | 0 | 0 | 0 | 6  | 0.00  |
| SJ  | L | 27 | DAVE CAPUANO        | 4  | 0 | 1 | 1 | -5  | 0   | 0 | 0 | 0 | 0 | 5  | 0.00  |
| WIN | D | 42 | OLEG MIKULCHIK      | 4  | 0 | 1 | 1 | -2  | 17  | 0 | 0 | 0 | 0 | 3  | 0.00  |
| ANA | D | 44 | MYLES O'CONNOR      | 5  | 0 | 1 | 1 | 0   | 6   | 0 | 0 | 0 | 0 | 7  | 0.00  |
| DAL | C | 37 | NEIL BRADY          | 5  | 0 | 1 | 1 | -1  | 21  | 0 | 0 | 0 | 0 | 1  | 0.00  |
| LA  | C | 28 | *GUY LEVEQUE        | 5  | 0 | 1 | 1 | 1   | 2   | 0 | 0 | 0 | 0 | 3  | 0.00  |
| BOS | D | 29 | *JOHN GRUDEN        | 7  | 0 | 1 | 1 | -3  | 2   | 0 | 0 | 0 | 0 | 8  | 0.00  |
| CAL | D | 34 | BRAD MILLER         | 8  | 0 | 1 | 1 | -2  | 14  | 0 | 0 | 0 | 0 | 2  | 0.00  |
| LA  | C | 8  | ROB MURPHY          | 8  | 0 | 1 | 1 | -3  | 22  | 0 | 0 | 0 | 0 | 4  | 0.00  |
| NYR | D | 5  | *MATTIAS NORSTROM   | 9  | 0 | 1 | 1 | 0   | 6   | 0 | 0 | 0 | 0 | 3  | 0.00  |
| OTT | D | 3  | KENT PAYNTER        | 9  | 0 | 1 | 1 | -6  | 8   | 0 | 0 | 0 | 0 | 8  | 0.00  |
| ANA | C | 26 | ROBIN BAWA          | 12 | 0 | 1 | 1 | -3  | 7   | 0 | 0 | 0 | 0 | 1  | 0.00  |
| CHI | D | 58 | *IVAN DROPPA        | 12 | 0 | 1 | 1 | 2   | 12  | 0 | 0 | 0 | 0 | 13 | 0.00  |
| EDM | D | 32 | GORDON MARK         | 12 | 0 | 1 | 1 | -2  | 43  | 0 | 0 | 0 | 0 | 8  | 0.00  |
| NYI | R | 34 | *DAN PLANTE         | 12 | 0 | 1 | 1 | -2  | 4   | 0 | 0 | 0 | 0 | 9  | 0.00  |
| TOR | D | 33 | *MATT MARTIN        | 12 | 0 | 1 | 1 | 0   | 6   | 0 | 0 | 0 | 0 | 6  | 0.00  |
| STL | C | 37 | *KEVIN MIEHM        | 14 | 0 | 1 | 1 | -3  | 4   | 0 | 0 | 0 | 0 | 5  | 0.00  |
| BUF | C | 17 | *TODD SIMON         | 15 | 0 | 1 | 1 | -3  | 0   | 0 | 0 | 0 | 0 | 14 | 0.00  |
| CAL | D | 6  | LEE NORWOOD         | 16 | 0 | 1 | 1 | 3   | 16  | 0 | 0 | 0 | 0 | 10 | 0.00  |
| HAR | G | 35 | JEFF REESE          | 20 | 0 | 1 | 1 | 0   | 0   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| CHI | G | 31 | JEFF HACKETT        | 22 | 0 | 1 | 1 | 0   | 2   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| NYI | G | 29 | *JAMIE MCLENNAN     | 22 | 0 | 1 | 1 | 0   | 6   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| OTT | D | 29 | *FRANCOIS LEROUX    | 23 | 0 | 1 | 1 | -4  | 70  | 0 | 0 | 0 | 0 | 8  | 0.00  |
| STL | G | 29 | JIM HRIVNAK         | 23 | 0 | 1 | 1 | 0   | 2   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| DAL | D | 43 | GORD DONNELLY       | 25 | 0 | 1 | 1 | -3  | 97  | 0 | 0 | 0 | 0 | 7  | 0.00  |
| BOS | G | 37 | VINCENT RIENDEAU    | 26 | 0 | 1 | 1 | 0   | 0   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| BOS | C | 45 | *ANDREW MCKIM       | 29 | 0 | 1 | 1 | -10 | 4   | 0 | 0 | 0 | 0 | 22 | 0.00  |
| BOS | D | 28 | JAMIE HUSCROFT      | 36 | 0 | 1 | 1 | -2  | 144 | 0 | 0 | 0 | 0 | 13 | 0.00  |
| PIT | G | 31 | KEN WREGGET         | 42 | 0 | 1 | 1 | 0   | 8   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| PIT | G | 35 | TOM BARRASSO        | 44 | 0 | 1 | 1 | 0   | 42  | 0 | 0 | 0 | 0 | 0  | 0.00  |
| WIN | G | 29 | TIM CHEVELDAE       | 44 | 0 | 1 | 1 | 0   | 2   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| WAS | G | 33 | DON BEAUPRE         | 53 | 0 | 1 | 1 | 0   | 16  | 0 | 0 | 0 | 0 | 0  | 0.00  |
| DAL | G | 35 | ANDY MOOG           | 55 | 0 | 1 | 1 | 0   | 16  | 0 | 0 | 0 | 0 | 0  | 0.00  |
| PHI | G | 33 | DOMINIC ROUSSEL     | 60 | 0 | 1 | 1 | 0   | 4   | 0 | 0 | 0 | 0 | 0  | 0.00  |
| TB  | G | 93 | DAREN PUPPA         | 63 | 0 | 1 | 1 | 0   | 2   | 0 | 0 | 0 | 0 | 0  | 0.00  |

|     |   |    |                   |    |   |   |   |    |    |   |   |   |   |   |      |
|-----|---|----|-------------------|----|---|---|---|----|----|---|---|---|---|---|------|
| LA  | G | 32 | KELLY HRUDEY      | 64 | 0 | 1 | 1 | 0  | 6  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| MON | G | 33 | PATRICK ROY       | 68 | 0 | 1 | 1 | 0  | 30 | 0 | 0 | 0 | 0 | 0 | 0.00 |
| ANA | C | 45 | *SCOTT MCKAY      | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| BUF | D | 33 | *MARK ASTLEY      | 1  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 2 | 0.00 |
| CAL | G | 31 | *JASON MUZZATTI   | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| CHI | G | 50 | *CHRISTIAN SOUCY  | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| DAL | R | 25 | ROB BROWN         | 1  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| DAL | L | 23 | DERRICK SMITH     | 1  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| EDM | G | 1  | WAYNE COWLEY      | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| EDM | R | 20 | *JOZEF CIERNY     | 1  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| FLO | D | 8  | DALLAS EAKINS     | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 2 | 0.00 |
| HAR | C | 28 | MIKE TOMLAK       | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 2 | 0.00 |
| HAR | G | 35 | *MIKE LENARDUZZI  | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| LA  | G | 41 | *DAVID GOVERDE    | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| LA  | D | 51 | *JUSTIN HOCKING   | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| MON | R | 42 | *LINDSAY VALLIS   | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| NJ  | R | 18 | SCOTT PELLERIN    | 1  | 0 | 0 | 0 | 0  | 2  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| NYI | C | 14 | *SCOTT SCISSONS   | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| NYI | C | 38 | *DEREK ARMSTRONG  | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 2 | 0.00 |
| OTT | G | 32 | DANIEL BERTHIAUME | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| OTT | D | 2  | KEVIN MACDONALD   | 1  | 0 | 0 | 0 | 0  | 2  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| PHI | D | 40 | *ARIS BRIMANIS    | 1  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| QUE | D | 4  | MIKE HURLBUT      | 1  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| QUE | D | 44 | *AARON MILLER     | 1  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| QUE | L | 24 | *PAXTON SCHULTE   | 1  | 0 | 0 | 0 | 0  | 2  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| SJ  | R | 23 | *ANDREI NAZAROV   | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| STL | L | 21 | *DAVID ROBERTS    | 1  | 0 | 0 | 0 | 0  | 2  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| STL | C | 22 | *IAN LAPERRIERE   | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| TB  | C | 16 | JASON LAFRENIERE  | 1  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 2 | 0.00 |
| WIN | D | 39 | *MARK VISHEAU     | 1  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| BOS | D | 28 | MIKHAIL TATARINOV | 2  | 0 | 0 | 0 | 0  | 2  | 0 | 0 | 0 | 0 | 4 | 0.00 |
| CAL | C | 16 | MARK FREER        | 2  | 0 | 0 | 0 | 0  | 4  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| CAL | D | 38 | PETER AHOLA       | 2  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| DET | L | 18 | MARK PEDERSON     | 2  | 0 | 0 | 0 | -1 | 2  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| EDM | D | 20 | JEFF CHYCHRUN     | 2  | 0 | 0 | 0 | 1  | 0  | 0 | 0 | 0 | 0 | 2 | 0.00 |
| EDM | D | 34 | *DARCY MARTINI    | 2  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| EDM | L | 20 | *BRADLEY ZAVISHA  | 2  | 0 | 0 | 0 | -2 | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| FLO | C | 16 | *LEN BARRIE       | 2  | 0 | 0 | 0 | -2 | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| FLO | G | 33 | ELDON REDDICK     | 2  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| FLO | R | 15 | *DOUG BARRAULT    | 2  | 0 | 0 | 0 | -2 | 0  | 0 | 0 | 0 | 0 | 2 | 0.00 |
| MON | R | 30 | *TURNER STEVENSON | 2  | 0 | 0 | 0 | -2 | 2  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| NYR | R | 16 | JIM HILLER        | 2  | 0 | 0 | 0 | 1  | 7  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| PHI | L | 43 | CLAUDE VILGRAIN   | 2  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| PIT | G | 30 | ROBERTO ROMANO    | 2  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| PIT | G | 1  | ROB DOPSON        | 2  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| SJ  | R | 15 | DAVID BRUCE       | 2  | 0 | 0 | 0 | -2 | 0  | 0 | 0 | 0 | 0 | 5 | 0.00 |
| STL | D | 44 | *TERRY HOLLINGER  | 2  | 0 | 0 | 0 | 1  | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| TOR | D | 38 | *CHRIS SNELL      | 2  | 0 | 0 | 0 | -1 | 2  | 0 | 0 | 0 | 0 | 4 | 0.00 |
| TOR | R | 24 | *PATRIK AUGUSTA   | 2  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 3 | 0.00 |
| WIN | L | 23 | ANDY BRICKLEY     | 2  | 0 | 0 | 0 | -2 | 0  | 0 | 0 | 0 | 0 | 0 | 0.00 |
| ANA | L | 27 | LONNIE LOACH      | 3  | 0 | 0 | 0 | -2 | 2  | 0 | 0 | 0 | 0 | 8 | 0.00 |
| ANA | D | 34 | ANATOLI FEDOTOV   | 3  | 0 | 0 | 0 | -1 | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| ANA | L | 43 | *MAXIM BETS       | 3  | 0 | 0 | 0 | -3 | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |
| DAL | D | 23 | DUANE JOYCE       | 3  | 0 | 0 | 0 | 0  | 0  | 0 | 0 | 0 | 0 | 1 | 0.00 |

|     |   |    |                    |   |   |   |   |     |    |   |   |   |    |      |
|-----|---|----|--------------------|---|---|---|---|-----|----|---|---|---|----|------|
| DET | G | 31 | PETER ING          | 3 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| HAR | L | 11 | YVON CORRIVEAU     | 3 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| NJ  | G | 1  | PETER SIDORKIEWICZ | 3 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| OTT | L | 25 | *CHAD PENNEY       | 3 | 0 | 0 | 0 | -2  | 2  | 0 | 0 | 0 | 2  | 0.00 |
| OTT | R | 33 | *GREG PANKEWICZ    | 3 | 0 | 0 | 0 | -1  | 2  | 0 | 0 | 0 | 3  | 0.00 |
| PIT | D | 4  | *GREG ANDRUSAK     | 3 | 0 | 0 | 0 | -1  | 2  | 0 | 0 | 0 | 4  | 0.00 |
| TB  | G | 30 | J.C. BERGERON      | 3 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| TOR | L | 41 | *ERIC LACROIX      | 3 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 3  | 0.00 |
| BOS | C | 29 | JON MORRIS         | 4 | 0 | 0 | 0 | -2  | 0  | 0 | 0 | 0 | 3  | 0.00 |
| BOS | D | 36 | JIM WIEMER         | 4 | 0 | 0 | 0 | -3  | 2  | 0 | 0 | 0 | 8  | 0.00 |
| BUF | L | 44 | *DOUG MACDONALD    | 4 | 0 | 0 | 0 | -2  | 0  | 0 | 0 | 0 | 3  | 0.00 |
| CAL | C | 33 | *DAVID STRUCH      | 4 | 0 | 0 | 0 | -2  | 4  | 0 | 0 | 0 | 3  | 0.00 |
| LA  | G | 1  | RICK KNICKLE       | 4 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| NYI | L | 38 | *JASON SIMON       | 4 | 0 | 0 | 0 | 0   | 34 | 0 | 0 | 0 | 0  | 0.00 |
| NYR | L | 32 | *DAN LACROIX       | 4 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| PHI | D | 6  | DAN KORDIC         | 4 | 0 | 0 | 0 | 0   | 5  | 0 | 0 | 0 | 0  | 0.00 |
| PIT | R | 37 | *JUSTIN DUBERMAN   | 4 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 2  | 0.00 |
| TB  | D | 3  | *ERIC CHARRON      | 4 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 1  | 0.00 |
| TOR | L | 36 | FRANK BIALOWAS     | 4 | 0 | 0 | 0 | 0   | 12 | 0 | 0 | 0 | 1  | 0.00 |
| VAN | C | 33 | *MIKE PECA         | 4 | 0 | 0 | 0 | -1  | 2  | 0 | 0 | 0 | 5  | 0.00 |
| WIN | C | 50 | *CRAIG FISHER      | 4 | 0 | 0 | 0 | -1  | 2  | 0 | 0 | 0 | 5  | 0.00 |
| CAL | D | 34 | *KEVIN WORTMAN     | 5 | 0 | 0 | 0 | 1   | 2  | 0 | 0 | 0 | 2  | 0.00 |
| EDM | L | 20 | *MARC LAFORGE      | 5 | 0 | 0 | 0 | -2  | 21 | 0 | 0 | 0 | 0  | 0.00 |
| EDM | C | 19 | *TYLER WRIGHT      | 5 | 0 | 0 | 0 | -3  | 4  | 0 | 0 | 0 | 2  | 0.00 |
| EDM | R | 34 | ALEXANDER KERCH    | 5 | 0 | 0 | 0 | -8  | 2  | 0 | 0 | 0 | 4  | 0.00 |
| MON | L | 36 | GERRY FLEMING      | 5 | 0 | 0 | 0 | -4  | 25 | 0 | 0 | 0 | 4  | 0.00 |
| NYI | L | 68 | *ZIGMUND PALFFY    | 5 | 0 | 0 | 0 | -6  | 0  | 0 | 0 | 0 | 5  | 0.00 |
| NYI | L | 17 | *STEVE JUNKER      | 5 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 2  | 0.00 |
| OTT | G | 35 | MARK LAFOREST      | 5 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| PHI | G | 35 | *FREDERIC CHABOT   | 5 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| QUE | G | 1  | *GARTH SNOW        | 5 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0  | 0.00 |
| TB  | D | 4  | *CORY CROSS        | 5 | 0 | 0 | 0 | -3  | 6  | 0 | 0 | 0 | 5  | 0.00 |
| WAS | G | 35 | *BYRON DAFOE       | 5 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| ANA | R | 33 | JIM THOMSON        | 6 | 0 | 0 | 0 | 0   | 5  | 0 | 0 | 0 | 1  | 0.00 |
| MON | G | 40 | *LES KUNTAR        | 6 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0  | 0.00 |
| QUE | D | 73 | ALAIN COTE         | 6 | 0 | 0 | 0 | -2  | 4  | 0 | 0 | 0 | 4  | 0.00 |
| STL | D | 43 | *JEFF BATTERS      | 6 | 0 | 0 | 0 | 1   | 7  | 0 | 0 | 0 | 1  | 0.00 |
| TB  | D | 5  | NORMAND ROCHEFORT  | 6 | 0 | 0 | 0 | -1  | 10 | 0 | 0 | 0 | 4  | 0.00 |
| TOR | D | 28 | *DAVID HARLOCK     | 6 | 0 | 0 | 0 | -2  | 0  | 0 | 0 | 0 | 2  | 0.00 |
| WIN | R | 28 | KEVIN MCCLELLAND   | 6 | 0 | 0 | 0 | 0   | 19 | 0 | 0 | 0 | 1  | 0.00 |
| WIN | C | 12 | ROB MURRAY         | 6 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 1  | 0.00 |
| CHI | L | 34 | TONY HORACEK       | 7 | 0 | 0 | 0 | 1   | 53 | 0 | 0 | 0 | 2  | 0.00 |
| HAR | G | 31 | MARIO GOSSELIN     | 7 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| HAR | D | 41 | ALLEN PEDERSEN     | 7 | 0 | 0 | 0 | -1  | 9  | 0 | 0 | 0 | 1  | 0.00 |
| LA  | L | 27 | DAVE THOMLINSON    | 7 | 0 | 0 | 0 | -6  | 21 | 0 | 0 | 0 | 6  | 0.00 |
| NYI | G | 35 | TOM DRAPER         | 7 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| OTT | D | 5  | *RADEK HAMR        | 7 | 0 | 0 | 0 | -10 | 0  | 0 | 0 | 0 | 5  | 0.00 |
| QUE | D | 42 | *JON KLEMM         | 7 | 0 | 0 | 0 | -1  | 4  | 0 | 0 | 0 | 11 | 0.00 |
| WAS | G | 37 | *OLAF KOLZIG       | 7 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0  | 0.00 |
| BUF | D | 29 | *DENIS TSYGUROV    | 8 | 0 | 0 | 0 | -1  | 8  | 0 | 0 | 0 | 3  | 0.00 |
| DAL | D | 5  | BRAD BERRY         | 8 | 0 | 0 | 0 | -2  | 12 | 0 | 0 | 0 | 4  | 0.00 |
| WIN | C | 14 | *HARIJS VITOLINSH  | 8 | 0 | 0 | 0 | 0   | 4  | 0 | 0 | 0 | 7  | 0.00 |
| DET | D | 44 | *GORD KRUPPKE      | 9 | 0 | 0 | 0 | -4  | 12 | 0 | 0 | 0 | 5  | 0.00 |
| PIT | L | 14 | *LADISLAV KARABIN  | 9 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 3  | 0.00 |

|     |   |    |                    |    |   |   |   |     |    |   |   |   |   |    |      |
|-----|---|----|--------------------|----|---|---|---|-----|----|---|---|---|---|----|------|
| TB  | G | 1  | WENDELL YOUNG      | 9  | 0 | 0 | 0 | 0   | 4  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| ANA | G | 35 | MIKHAIL SHTALENKOV | 10 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| BOS | L | 13 | GRIGORI PANTELEEV  | 10 | 0 | 0 | 0 | -2  | 0  | 0 | 0 | 0 | 0 | 8  | 0.00 |
| OTT | L | 9  | *ANDY SCHNEIDER    | 10 | 0 | 0 | 0 | -6  | 15 | 0 | 0 | 0 | 0 | 4  | 0.00 |
| CAL | G | 1  | *ANDREI TREFILOV   | 11 | 0 | 0 | 0 | 0   | 4  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| MON | G | 37 | ANDRE RACICOT      | 11 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| PIT | D | 2  | *CHRIS TAMER       | 12 | 0 | 0 | 0 | 3   | 9  | 0 | 0 | 0 | 0 | 10 | 0.00 |
| WIN | G | 30 | STEPH BEAUREGARD   | 13 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| OTT | L | 12 | GRAEME TOWNSHEND   | 14 | 0 | 0 | 0 | -7  | 9  | 0 | 0 | 0 | 0 | 5  | 0.00 |
| QUE | G | 32 | JACQUES CLOUTIER   | 14 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| SJ  | G | 29 | JIM WAITE          | 15 | 0 | 0 | 0 | 0   | 6  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| TOR | G | 35 | PAT JABLONSKI      | 15 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| WIN | C | 18 | BRYAN ERICKSON     | 16 | 0 | 0 | 0 | -7  | 6  | 0 | 0 | 0 | 0 | 8  | 0.00 |
| WIN | G | 1  | *MICHAEL O'NEILL   | 17 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| BOS | G | 39 | JOHN BLUE          | 18 | 0 | 0 | 0 | 0   | 7  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| EDM | G | 31 | *FRED BRATHWAITE   | 19 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| HAR | G | 40 | FRANK PIETRANGELO  | 19 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| LA  | G | 35 | ROBB STAUBER       | 22 | 0 | 0 | 0 | 0   | 18 | 0 | 0 | 0 | 0 | 0  | 0.00 |
| TOR | G | 1  | *DAMIAN RHODES     | 22 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| NYI | L | 14 | JOE DAY            | 24 | 0 | 0 | 0 | -7  | 30 | 0 | 0 | 0 | 0 | 16 | 0.00 |
| OTT | R | 25 | HERB RAGLAN        | 29 | 0 | 0 | 0 | -13 | 52 | 0 | 0 | 0 | 0 | 13 | 0.00 |
| QUE | G | 41 | *JOCELYN THIBAUT   | 29 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| OTT | G | 30 | *DARRIN MADELEY    | 32 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| VAN | G | 35 | KAY WHITMORE       | 32 | 0 | 0 | 0 | 0   | 6  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| WAS | G | 31 | RICK TABARACCI     | 32 | 0 | 0 | 0 | 0   | 6  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| PHI | G | 30 | TOMMY SODERSTROM   | 34 | 0 | 0 | 0 | 0   | 0  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| MON | G | 1  | RON TUGNUTT        | 36 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| DET | G | 30 | *CHRIS OSGOOD      | 41 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| HAR | G | 1  | SEAN BURKE         | 47 | 0 | 0 | 0 | 0   | 16 | 0 | 0 | 0 | 0 | 0  | 0.00 |
| NJ  | G | 30 | *MARTIN BRODEUR    | 47 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| CAL | G | 30 | MIKE VERNON        | 48 | 0 | 0 | 0 | 0   | 14 | 0 | 0 | 0 | 0 | 0  | 0.00 |
| ANA | G | 31 | GUY HEBERT         | 52 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0 | 0  | 0.00 |
| FLA | G | 34 | JOHN VANBIESBROUCK | 57 | 0 | 0 | 0 | 0   | 38 | 0 | 0 | 0 | 0 | 0  | 0.00 |
| OTT | G | 1  | CRAIG BILLINGTON   | 63 | 0 | 0 | 0 | 0   | 8  | 0 | 0 | 0 | 0 | 1  | 0.00 |
| NYR | G | 35 | MIKE RICHTER       | 68 | 0 | 0 | 0 | 0   | 2  | 0 | 0 | 0 | 0 | 0  | 0.00 |

### 14.3. LOGS DE MAPEO DE OBJETOS Y CONTROL DE CELDAS BLANCAS

#### PARA 3D Y N=871 (NHL94)

En esta sección pueden visualizarse los logs que genera el programa que implementa el algoritmo **FindAllOutsM** para el control de su funcionamiento. Los dos tipos de logs que se generan son:

- *Mapeo celda/objeto:* Muestra el resultado de mapear todo el banco de datos en las distintas celdas utilizadas por el algoritmo.
- *Control de celdas blancas:* Muestra como se realizó la determinación fina de outliers.



### 14.3.1. LOGS DE MAPEOS ENTRE CELDAS Y OBJETOS

Cell [0,4,0] = 4 Items Red

521 BRAD LAUER

574 DEREK MAYER

720 \*ANDREW MCKIM

829 RADEK HAMR

Cell [1,4,0] = 2 Items Pink

434 RUSS ROMANIUK

448 TONY HRKAC

Cell [6,4,0] = 1 Items White

095 ZDENO CIGER

Cell [0,5,0] = 87 Items Red

493 JOHN LEBLANC

525 BRAD SCHLEGEL

534 DOMINIC LAVOIE

536 BILL MCDUGALL

538 PHIL BOURQUE

548 JAROSLAV OTEVREL

552 JAMES BLACK

553 ROBERT BURAKOWSKY

570 KIP MILLER

573 PETER ANDERSSON

575 SCOTT THOMAS

578 BOB WILKIE

579 CHRIS TANCILL

583 BRIAN MCREYNOLDS

584 DANIEL LAPERRIERE

589 SERGEI PETRENKO

599 MARIVSZ CZERKAWSKI

600 SERGEI ZHOLTOK

608 ROMAN OKSIUTA

619 LEONARD ESAU

632 GREG PASLAWSKI

638 DAN CURRIE

640 MILOS HOLAN

644 PAVOL DEMITRA

648 JOBY MESSIER

650 ARTO BLOMSTEN

665 TIM TAYLOR

666 JAMIE LEACH

670 DAVE CHYZOWSKI

671 GARY EMMONS

672 MICHAL GROSEK

675 DENNY FELSNER

678 SERGEI KRIVOKRASOV

682 ALEXEI KUDASHOV

685 MIKE NEEDHAM

686 STEPHANE RICHER.

689 BOB JAY

692 STEVE MALTAIS

693 TODD MARCHANT

694 JEFF GREENLAW

695 DAVE CAPUANO

700 JOHN GRUDEN  
704 KENT PAYNTER  
705 ROBIN BAWA  
708 DAN PLANTE  
710 KEVIN MIEHM  
711 TODD SIMON  
732 MARK ASTLEY  
735 ROB BROWN  
736 DERRICK SMITH  
738 JOZEF CIERNY  
750 ARIS BRIMANIS  
751 MIKE HURLBUT  
752 AARON MILLER  
757 JASON LAFRENIERE  
762 MARK PEDERSON  
764 DARCY MARTINI  
765 BRADLEY ZAVISHA  
766 LEN BARRIE  
768 DOUG BARRAULT  
769 TURNER STEVENSON  
771 CLAUDE VILGRAIN  
774 DAVID BRUCE  
776 CHRIS SNELL  
778 ANDY BRICKLEY  
779 LONNIE LOACH  
780 ANATOLI FEDOTOV  
781 MAXIM BETS  
786 CHAD PENNEY  
787 GREG PANKIEWICZ  
788 GREG ANDRUSAK  
791 JON MORRIS  
792 JIM WIEMER  
793 DOUG MACDONALD  
794 DAVID STRUCH  
802 MIKE PECA  
803 CRAIG FISHER  
806 TYLER WRIGHT  
807 ALEXANDER KERCH  
809 ZIGMUND PALFFY  
814 CORY CROSS  
818 ALAIN COTE  
821 DAVID HARLOCK  
830 JON KLEMM  
832 DENIS TSYGUROV  
839 GRIGORI PANTELEEV  
849 BRYAN ERICKSON  
Cell [1,5,0] = 3 Items Red  
377 CRAIG SIMPSON  
386 PETER STASTNY  
470 ADRIEN PLAVSIC  
Cell [2,5,0] = 1 Items Pink  
362 PAT LAFONTAINE  
Cell [0,6,0] = 136 Items Red  
498 PETER WHITE

499 SHJON PODEIN  
512 DAN QUINN  
524 JOHN LILLEY  
535 YANIC PERREAULT  
551 JARKKO VARVIO  
569 OWEN NOLAN  
592 TREVOR KIDD  
596 KIRK MCLEAN  
597 FELIX POTVIN  
601 YAN KAMINSKY  
604 JASON RUFF  
605 JASON YORK  
607 VESA VIITAKOSKI  
609 BRENT GRETZKY  
610 JASON WOOLLEY  
620 DAVE CHRISTIAN  
621 JOHN STEVENS  
628 STEPHANE Fiset  
629 DOMINIK HASEK  
631 CURTIS JOSEPH  
634 DAVID HAAS  
635 PATRICK LEBEAU  
636 DWAYNE NORRIS  
637 DAVID SACCO  
639 STEPHANE MORIN  
642 RENE CORBET  
649 CHRIS WINNES  
654 MARK FITZPATRICK  
655 GLENN HEALY  
660 CHRIS TERRERI  
662 BOB ESSENSA  
663 BILL RANFORD  
667 TODD HLUSHKO  
668 TODD NELSON  
669 BRIAN SAVAGE  
673 AARON WARD  
674 JEFF MCLEAN  
676 LARRY DEPALMA  
677 ED WARD  
687 CRAIG FERGUSON  
688 JASON ALLISON  
697 MYLES O'CONNOR  
699 GUY LEVEQUE  
703 MATTIAS NORSTROM  
709 MATT MARTIN  
713 JEFF REESE  
714 JEFF HACKETT  
715 JAMIE MCLENNAN  
717 JIM HRIVNAK  
719 VINCENT RIENDEAU  
722 KEN WREGGET  
724 TIM CHEVELDAE  
727 DOMINIC ROUSSEL  
728 DAREN PUPPA

729 KELLY HRUDEY  
731 SCOTT MCKAY  
733 JASON MUZZATTI  
734 CHRISTIAN SOUCY  
737 WAYNE COWLEY  
739 DALLAS EAKINS  
740 MIKE TOMLAK  
741 MIKE LENARDUZZI  
742 DAVID GOVERDE  
743 JUSTIN HOCKING  
744 LINDSAY VALLIS  
745 SCOTT PELLERIN  
746 SCOTT SCISSIONS  
747 DEREK ARMSTRONG  
748 DANIEL BERTHIAUME  
749 KEVIN MACDONALD  
753 PAXTON SCHULTE  
754 ANDREI NAZAROV  
755 DAVID ROBERTS  
756 IAN LAPERRIERE  
758 MARK VISHEAU  
759 MIKHAIL TATARINOV  
760 MARK FREER  
761 PETER AHOLA  
763 JEFF CHYCHRUN  
767 ELDON REDDICK  
770 JIM HILLER  
772 ROBERTO ROMANO  
773 ROB DOPSON  
775 TERRY HOLLINGER  
777 PATRIK AUGUSTA  
782 DUANE JOYCE  
783 PETER ING  
784 YVON CORRIVEAU  
785 PETER SIDORKIEWICZ  
789 J.C. BERGERON  
790 ERIC LACROIX  
795 RICK KNICKLE  
797 DAN LACROIX  
798 DAN KORDIC  
799 JUSTIN DUBERMAN  
800 ERIC CHARRON  
804 KEVIN WORTMAN  
810 STEVE JUNKER  
811 MARK LAFOREST  
812 FREDERIC CHABOT  
813 GARTH SNOW  
815 BYRON DAFOE  
816 JIM THOMSON  
817 LES KUNTAR  
819 JEFF BATTERS  
823 ROB MURRAY  
825 MARIO GOSSELIN  
828 TOM DRAPER

831 OLAF KOLZIG  
834 HARIJS VITOLINSH  
836 LADISLAV KARABIN  
837 WENDELL YOUNG  
838 MIKHAIL SHTALENKOV  
841 ANDREI TREFILOV  
842 ANDRE RACICOT  
844 STEPH BEAUREGARD  
846 JACQUES CLOUTIER  
847 JIM WAITE  
848 PAT JABLONSKI  
850 MICHAEL O'NEILL  
851 JOHN BLUE  
852 FRED BRATHWAITE  
853 FRANK PIETRANGELO  
855 DAMIAN RHODES  
858 JOCELYN THIBAUT  
859 DARRIN MADELEY  
860 KAY WHITMORE  
861 RICK TABARACCI  
862 TOMMY SODERSTROM  
863 RON TUGNUTT  
864 CHRIS OSGOOD  
866 MARTIN BRODEUR  
868 GUY HEBERT  
870 CRAIG BILLINGTON  
871 MIKE RICHTER  
Cell [1,6,0] = 3 Items Red  
395 JIM DOWD  
398 IGOR CHIBIREV  
462 DAVID EMMA  
Cell [2,6,0] = 1 Items Pink  
336 PETR NEDVED  
Cell [3,6,0] = 2 Items Pink  
270 OLEG PETROV  
278 JODY HULL  
Cell [5,6,0] = 1 Items White  
126 ROBERT KRON  
Cell [2,7,0] = 1 Items Pink  
299 MARK HOWE  
Cell [2,3,1] = 1 Items White  
363 VLADIMIR RUZICKA  
Cell [0,4,1] = 2 Items Pink  
626 MIKE LALOR  
651 ROB RAMAGE  
Cell [1,4,1] = 2 Items Pink  
449 TIM BERGLAND  
482 DALE CRAIGWELL  
Cell [2,4,1] = 1 Items Pink  
352 DARREN TURCOTTE  
Cell [5,4,1] = 1 Items White  
130 IGOR KRAVCHUK  
Cell [0,5,1] = 18 Items Red  
542 JARMO KEKALAINEN

550 KEVIN SMYTH  
558 MICHAL SYKORA  
565 YEVGENY NAMESTNIKOV  
567 ED PATTERSON  
581 GUY LAROSE  
582 CHRIS LUONGO  
611 BRYAN FOGARTY  
612 TED CROWLEY  
653 STEVE KONROYD  
696 OLEG MIKULCHIK  
701 BRAD MILLER  
820 NORMAND ROCHEFORT  
826 ALLEN PEDERSEN  
833 BRAD BERRY  
835 GORD KRUPPKE  
840 ANDY SCHNEIDER  
845 GRAEME TOWNSHEND  
Cell [1,5,1] = 6 Items Red  
409 CHRIS LINDBERG  
420 ANDRE FAUST  
441 RON WILSON  
476 JARROD SKALDE  
479 MIKE SULLIVAN  
492 SHAWN ANDERSON  
Cell [2,5,1] = 5 Items Red  
316 PHIL HOUSLEY  
341 PELLE EKLUND  
343 DMITRI KVARTALNOV  
353 VLADIMIR VUJTEK  
357 DAVID ARCHIBALD  
Cell [3,5,1] = 1 Items Pink  
242 ANATOLI SEMENOV  
Cell [5,5,1] = 2 Items White  
121 SCOTT YOUNG  
128 JESSE BELANGER  
Cell [8,5,1] = 1 Items White  
066 ULF DAHLEN  
Cell [0,6,1] = 20 Items Red  
505 STEVE DUBINSKY  
515 REGGIE SAVAGE  
532 DANE JACKSON  
539 TROY MURRAY  
549 FRED KNIPSCHER  
571 CHRIS GOVEDARIS  
593 GRANT FUHR  
641 PAT NEATON  
661 JON CASEY  
664 ARTURS IRBE  
690 DENIS CHASSE  
691 DARREN BANKS  
706 IVAN DROPPA  
712 LEE NORWOOD  
725 DON BEAUPRE  
726 ANDY MOOG

801 FRANK BIALOWAS  
843 CHRIS TAMER  
865 SEAN BURKE  
867 MIKE VERNON  
Cell [1,6,1] = 4 Items Red  
397 GARY LEEMAN  
412 DAVE VOLEK  
421 JASON DAWE  
485 DOUG CROSSMAN  
Cell [2,6,1] = 2 Items Pink  
306 JOZEF STUMPEL  
347 ROBERT LANG  
Cell [3,6,1] = 2 Items Pink  
231 STEVE DUCHESNE  
256 MIKE SILLINGER  
Cell [4,6,1] = 2 Items Pink  
177 RAY WHITNEY  
206 NIKOLAI BORSCHEVSKY  
Cell [3,7,1] = 1 Items White  
274 DEAN MCAMMOND  
Cell [4,8,1] = 1 Items White  
190 WES WALZ  
Cell [8,9,1] = 1 Items White  
054 STEPHANE RICHER  
Cell [9,0,2] = 1 Items White  
039 \*ALEXEI YASHIN  
Cell [8,2,2] = 1 Items White  
057 BOB KUDELSKI  
Cell [0,3,2] = 1 Items White  
614 HANK LAMMENS  
Cell [3,3,2] = 1 Items White  
272 PAUL RANHEIM  
Cell [6,3,2] = 1 Items White  
107 TEEMU SELANNE  
Cell [15,3,2] = 1 Items White  
001 WAYNE GRETZKY  
Cell [0,4,2] = 2 Items Pink  
561 DIMITRI FILIMONOV  
586 DAVE TOMLINSON  
Cell [3,4,2] = 1 Items Pink  
265 JIMMY CARSON  
Cell [0,5,2] = 15 Items Red  
503 JEFF DANIELS  
516 NATHAN LAFAYETTE  
520 DAVE BARR  
537 DAN KESA  
544 JIM PAEK  
577 MIKE RAMSEY  
585 VLASTIMIL KROUPA  
590 DARRYL SHANNON  
624 KEVIN DAHL  
679 KEITH REDMOND  
698 NEIL BRADY  
702 ROB MURPHY

805 MARC LAFORGE  
808 GERRY FLEMING  
827 DAVE THOMLINSON  
Cell [1,5,2] = 2 Items Pink  
371 GREG JOHNSON  
461 DAN MAROIS  
Cell [2,5,2] = 2 Items Pink  
286 VITALI PROKHOROV  
315 KEVIN TODD  
Cell [3,5,2] = 3 Items Red  
212 PETER DOURIS  
213 SHAWN CHAMBERS  
277 STEPHAN LEBEAU  
Cell [4,5,2] = 2 Items Pink  
197 IAIN FRASER  
199 GREG ADAMS  
Cell [6,5,2] = 1 Items White  
111 PAT FALLOON  
Cell [9,5,2] = 1 Items White  
031 CRAIG JANNEY  
Cell [10,5,2] = 1 Items White  
019 JOE SAKIC  
Cell [0,6,2] = 6 Items Red  
580 NEIL EISENHUT  
606 CHRISTIAN PROULX  
623 RICHARD MATVICHUK  
680 BEN HANKINSON  
822 KEVIN MCCLELLAND  
854 ROBB STAUBER  
Cell [1,6,2] = 3 Items Red  
378 PETER ZEZEL  
443 TOMMY SJODIN  
460 BRAD WERENKA  
Cell [2,6,2] = 2 Items Pink  
287 MIKAEL ANDERSSON  
356 BRENT GRIEVE  
Cell [3,6,2] = 1 Items Pink  
251 ALEXANDER SEMAK  
Cell [5,6,2] = 1 Items Pink  
122 BOB CORKUM  
Cell [6,6,2] = 1 Items White  
099 DEREK PLANTE  
Cell [9,6,2] = 1 Items White  
038 ALEXANDER MOGILNY  
Cell [1,7,2] = 2 Items Pink  
374 JAROSLAV MODRY  
488 JEFF FINLEY  
Cell [2,7,2] = 1 Items Pink  
308 DAVE REID  
Cell [3,7,2] = 1 Items Pink  
244 BENOIT BRUNET  
Cell [7,7,2] = 1 Items White  
077 SANDIS OZOLINSH  
Cell [8,7,2] = 1 Items White



062 MIKE RIDLEY  
Cell [10,7,2] = 1 Items White  
014 PIERRE TURGEON  
Cell [6,8,2] = 1 Items White  
103 DOUG BROWN  
Cell [7,8,2] = 1 Items White  
076 MARTIN STRAKA  
Cell [6,9,2] = 1 Items White  
097 MARTY MCINNIS  
Cell [5,1,3] = 1 Items White  
124 THOMAS STEEN  
Cell [2,3,3] = 2 Items White  
309 TEPPU NUMMINEN  
311 NORM MACIVER  
Cell [0,4,3] = 2 Items Pink  
501 MARK FERNER  
657 DOUG LIDSTER  
Cell [1,4,3] = 1 Items Pink  
389 JEFF SHANTZ  
Cell [2,4,3] = 1 Items Pink  
364 LUCIANO BORSATO  
Cell [4,4,3] = 1 Items Pink  
205 ROB GAUDREAU  
Cell [0,5,3] = 6 Items Red  
500 ED OLCZYK  
513 DAVE TAYLOR  
517 GARY SHUCHUK  
622 MARK HARDY  
652 IAN HERBERS  
856 JOE DAY  
Cell [1,5,3] = 6 Items Red  
393 MARC BUREAU  
402 GREG GILBERT  
410 PHILIPPE BOUCHER  
423 SHELDON KENNEDY  
450 MIKE KRUSHELNYSKI  
454 MARKUS NASLUND  
Cell [2,5,3] = 1 Items Pink  
321 STEPHEN HEINZE  
Cell [3,5,3] = 3 Items Red  
245 KELLY KISIO  
260 MARTIN GELINAS  
266 ILYA BYAKIN  
Cell [4,5,3] = 3 Items Red  
196 MARIO LEMIEUX  
207 PAUL YSEBAERT  
209 FREDRIK OLAUSSON  
Cell [5,5,3] = 1 Items Pink  
144 GILBERT DIONNE  
Cell [0,6,3] = 6 Items Red  
494 JIM SANDLAK  
510 TOM TILLEY  
572 DONALD BRASHEAR  
659 DARCY WAKALUK

730 PATRICK ROY  
796 JASON SIMON  
Cell [1,6,3] = 3 Items Red  
381 JOHN SLANEY  
384 JIM STORM  
455 GREG BROWN  
Cell [2,6,3] = 2 Items Pink  
293 TOM PEDERSON  
359 MIKE EASTWOOD  
Cell [3,6,3] = 3 Items Red  
229 BRENT GILCHRIST  
236 MICHEL GOULET  
241 GAETAN DUCHESNE  
Cell [4,6,3] = 2 Items Pink  
183 KELLY MILLER  
160 MIKE DONNELLY  
Cell [5,6,3] = 2 Items Pink  
135 ANDREI LOMAKIN  
152 DANTON COLE  
Cell [6,6,3] = 2 Items Pink  
104 MURRAY CRAVEN  
105 MICHAEL NYLANDER  
Cell [1,7,3] = 3 Items Red  
405 VIACHESLAV FETISOV  
415 PETER POPOVIC  
425 KRIS DRAPER  
Cell [2,7,3] = 2 Items Pink  
314 SHAWN BURR  
323 UWE KRUPP  
Cell [3,7,3] = 1 Items Pink  
279 STEVE KONOWALCHUK  
Cell [4,7,3] = 4 Items Red  
188 TRENT KLATT  
161 SHAWN MCEACHERN  
167 TOM CHORSKE  
172 JOHN TUCKER  
Cell [6,7,3] = 1 Items Pink  
113 JOHAN GARPENLOV  
Cell [10,7,3] = 1 Items White  
015 RAY SHEPPARD  
Cell [2,8,3] = 1 Items Pink  
297 PAUL BROTEN  
Cell [5,8,3] = 2 Items Pink  
143 GERMAN TITOV  
153 JOHN LECLAIR  
Cell [6,11,3] = 1 Items White  
101 NICKLAS LIDSTROM  
Cell [13,11,3] = 1 Items White  
002 SERGEI FEDOROV  
Cell [5,0,4] = 1 Items White  
123 ALEXANDRE DAIGLE  
Cell [1,2,4] = 1 Items White  
388 WAYNE MCBEAN  
Cell [2,2,4] = 1 Items White

366 KERRY HUFFMAN  
 Cell [1,3,4] = 1 Items White  
 401 DAVE TIPPETT  
 Cell [4,3,4] = 1 Items White  
 182 BILL HOULDER  
 Cell [6,3,4] = 1 Items White  
 089 ANDREW CASSELS  
 Cell [0,4,4] = 1 Items Pink  
 527 GORDIE ROBERTS  
 Cell [1,4,4] = 2 Items Pink  
 385 IGOR KOROLEV  
 399 BRYAN TROTTIER  
 Cell [2,4,4] = 2 Items Pink  
 360 GARY SUTER  
 361 TED DRURY  
 Cell [3,4,4] = 1 Items Pink  
 224 STEVEN RICE  
 Cell [7,4,4] = 1 Items White  
 071 GEOFF SANDERSON  
 Cell [0,5,4] = 2 Items Pink  
 496 MICAH AIVAZOFF  
 555 PAUL MACDERMID  
 Cell [1,5,4] = 4 Items Red  
 422 MARK GREIG  
 431 ROB ZAMUNER  
 447 ROBERT PETROVICKY  
 483 JOHN MCINTYRE  
 Cell [2,5,4] = 1 Items Pink  
 333 ADAM CREIGHTON  
 Cell [3,5,4] = 1 Items Pink  
 218 PAUL DI PIETRO  
 Cell [4,5,4] = 1 Items Pink  
 210 JAMES PATRICK  
 Cell [11,5,4] = 1 Items White  
 011 BRETT HULL  
 Cell [0,6,4] = 3 Items Red  
 643 KEN MCRAE  
 723 TOM BARRASSO  
 869 JOHN VANBIESBROUCK  
 Cell [1,6,4] = 5 Items Red  
 369 JAMIE BAKER  
 403 MIKE MCKEE  
 411 ED RONAN  
 439 DON MCSWEEN  
 478 PIERRE SEVIGNY  
 Cell [2,6,4] = 3 Items Red  
 291 PHILIPPE BOZON  
 294 ALEXEI GUSAROV  
 339 DAVID WILLIAMS  
 Cell [3,6,4] = 2 Items Pink  
 253 PAT PEAKE  
 267 ERIC WEINRICH  
 Cell [4,6,4] = 2 Items Pink  
 203 MIKE MCPHEE

163 TONY AMONTE  
 Cell [5,6,4] = 4 Items Red  
 129 MICHAL PIVONKA  
 134 STU BARNES  
 142 GRANT LEDYARD  
 158 DAVE ELLETT  
 Cell [6,6,4] = 1 Items Pink  
 084 DONALD AUDETTE  
 Cell [7,6,4] = 1 Items White  
 069 CLIFF RONNING  
 Cell [9,6,4] = 1 Items White  
 032 \*MIKAEL RENBERG  
 Cell [1,7,4] = 1 Items Pink  
 435 KEITH CARNEY  
 Cell [3,7,4] = 1 Items Pink  
 214 GREG HAWGOOD  
 Cell [4,7,4] = 2 Items Pink  
 181 JEFF NORTON  
 165 PATRICK FLATLEY  
 Cell [5,7,4] = 1 Items Pink  
 132 SERGEI NEMCHINOV  
 Cell [6,7,4] = 1 Items Pink  
 082 STEVE LARMER  
 Cell [7,7,4] = 1 Items White  
 074 VALERI KAMENSKY  
 Cell [8,7,4] = 2 Items White  
 055 BRIAN BELLOWS  
 058 JOE MULLEN  
 Cell [9,7,4] = 2 Items White  
 028 JOE JUNEAU  
 034 STEVE YZERMAN  
 Cell [2,8,4] = 1 Items Pink  
 355 TOMMY ALBELIN  
 Cell [5,8,4] = 1 Items Pink  
 151 PETER BONDRA  
 Cell [6,8,4] = 1 Items White  
 100 IGOR LARIONOV  
 Cell [10,8,4] = 1 Items White  
 022 SERGEI ZUBOV  
 Cell [5,10,4] = 1 Items White  
 141 SCOTT NIEDERMAYER  
 Cell [5,1,5] = 1 Items White  
 154 DAVE MCLLWAIN  
 Cell [8,3,5] = 2 Items White  
 042 JARI KURRI  
 050 DOUG WEIGHT  
 Cell [1,4,5] = 2 Items Pink  
 407 RANDY GILHEN  
 491 GEOFF SMITH  
 Cell [3,4,5] = 1 Items Pink  
 282 ROB NIEDERMAYER  
 Cell [6,4,5] = 1 Items White  
 118 YVES RACINE  
 Cell [0,5,5] = 4 Items Red

625 BOB MCGILL  
 645 CLAUDE LOISELLE  
 684 TODD HARKINS  
 707 GORDON MARK  
 Cell [1,5,5] = 4 Items Red  
 406 JOSE CHARBONNEAU  
 446 STEVEN KING  
 451 MIKE HUDSON  
 480 ADAM BENNETT  
 Cell [2,5,5] = 6 Items Red  
 288 MIKE STAPLETON  
 304 ROB DIMAIO  
 322 VITALI KARAMNOV  
 328 FRANTISEK KUCERA  
 335 EVGENY DAVYDOV  
 337 JIM MONTGOMERY  
 Cell [3,5,5] = 3 Items Red  
 216 ANDREI KOVALENKO  
 228 GLEN MURRAY  
 261 PATRICK POULIN  
 Cell [4,5,5] = 1 Items Pink  
 178 DALLAS DRAKE  
 Cell [12,5,5] = 1 Items White  
 007 MARK RECCHI  
 Cell [0,6,5] = 1 Items Pink  
 566 JASON SMITH  
 Cell [3,6,5] = 3 Items Red  
 221 BOB CARPENTER  
 252 BRIAN PROPP  
 276 TODD GILL  
 Cell [4,6,5] = 1 Items Pink  
 180 TOM KURVERS  
 Cell [5,6,5] = 1 Items Pink  
 155 TIM SWEENEY  
 Cell [6,6,5] = 2 Items Pink  
 106 JYRKI LUMME  
 114 TERRY YAKE  
 Cell [3,7,5] = 2 Items Pink  
 217 DIRK GRAHAM  
 230 JOHN DRUCE  
 Cell [4,7,5] = 2 Items Pink  
 189 GUY CARBONNEAU  
 171 TRAVIS GREEN  
 Cell [6,7,5] = 1 Items Pink  
 086 YURI KHMYLEV  
 Cell [8,7,5] = 1 Items White  
 053 LARRY MURPHY  
 Cell [13,7,5] = 1 Items White  
 003 ADAM OATES  
 Cell [4,8,5] = 1 Items White  
 192 BRENT SUTTER  
 Cell [8,8,5] = 1 Items White  
 047 JOE NIEUWENDYK  
 Cell [2,9,5] = 1 Items White

326 DON SWEENEY  
 Cell [8,9,5] = 1 Items White  
 051 VYACHESLAV KOZLOV  
 Cell [1,10,5] = 1 Items White  
 490 FRANK MUSIL  
 Cell [2,1,6] = 1 Items White  
 310 BRAD SHAW  
 Cell [1,3,6] = 1 Items White  
 404 BRIAN GLYNN  
 Cell [3,3,6] = 1 Items White  
 281 SYLVAIN TURGEON  
 Cell [0,4,6] = 2 Items Pink  
 627 DARCY LOEWEN  
 857 HERB RAGLAN  
 Cell [7,4,6] = 1 Items White  
 073 JEFF BROWN  
 Cell [0,5,6] = 3 Items Red  
 507 DONALD DUFRESNE  
 554 DAVE MACKEY  
 615 MARIO ROBERGE  
 Cell [1,5,6] = 2 Items Pink  
 465 PAUL STANTON  
 469 MIKE RATHJE  
 Cell [2,5,6] = 3 Items Red  
 292 DAVE POULIN  
 303 PATRIK CARNBACK  
 319 DAN KECZMER  
 Cell [3,5,6] = 2 Items Pink  
 223 TOM FITZGERALD  
 239 TODD KRYGIER  
 Cell [7,5,6] = 1 Items White  
 078 BRIAN BRADLEY  
 Cell [10,5,6] = 1 Items White  
 016 MIKE MODANO  
 Cell [0,6,6] = 2 Items Pink  
 630 RON HEXTALL  
 824 TONY HORACEK  
 Cell [1,6,6] = 3 Items Red  
 379 MARTIN LAPOINTE  
 444 CHRIS DAHLQUIST  
 477 MIKE FOLIGNO  
 Cell [3,6,6] = 2 Items Pink  
 225 MARTIN RUCINSKY  
 227 DAVE BABYCH  
 Cell [4,6,6] = 1 Items Pink  
 166 CALLE JOHANSSON  
 Cell [6,6,6] = 1 Items White  
 110 TED DONATO  
 Cell [9,6,6] = 2 Items White  
 027 MATS SUNDIN  
 036 RUSS COURTNALL  
 Cell [1,7,6] = 1 Items Pink  
 436 KEITH BROWN  
 Cell [2,7,6] = 2 Items Pink

327 DAVE HANNAN  
367 A. KARPOVTSEV  
Cell [4,7,6] = 1 Items Pink  
187 STEPHANE MATTEAU  
Cell [8,7,6] = 1 Items White  
048 CAM NEELY  
Cell [2,8,6] = 2 Items White  
325 TRENT YAWNEY  
334 BOBBY DOLLAS  
Cell [5,8,6] = 1 Items White  
127 COREY MILLEN  
Cell [8,8,6] = 1 Items White  
061 DEREK KING  
Cell [10,8,6] = 1 Items White  
017 ROBERT REICHEL  
Cell [10,9,6] = 1 Items White  
021 RAY BOURQUE  
Cell [2,1,7] = 1 Items White  
344 ANDREW MCBAIN  
Cell [3,3,7] = 1 Items White  
262 PAT ELYNUIK  
Cell [8,3,7] = 1 Items White  
056 ALEXEI ZHAMNOV  
Cell [0,4,7] = 1 Items Pink  
514 ALLAN CONROY  
Cell [1,4,7] = 1 Items Pink  
472 TIM WATTERS  
Cell [4,4,7] = 1 Items White  
195 JOE SACCO  
Cell [5,4,7] = 1 Items White  
150 DEAN EVASON  
Cell [0,5,7] = 3 Items Red  
526 JAY MORE  
530 JIM JOHNSON  
683 BRIAN CURRAN  
Cell [1,5,7] = 3 Items Red  
445 BRET HEDICAN  
466 DRAKE BEREHOWSKY  
481 CAMERON STEWART  
Cell [2,5,7] = 2 Items Pink  
300 ALEXEI KASATONOV  
317 CURTIS LESCHYSHYN  
Cell [3,5,7] = 2 Items Pink  
237 JOHN CULLEN  
254 CHRISTIAN RUUTTU  
Cell [4,5,7] = 2 Items White  
198 DAVE LOWRY  
168 GLENN ANDERSON  
Cell [10,5,7] = 1 Items White  
018 RON FRANCIS  
Cell [0,6,7] = 2 Items Pink  
506 ADAM FOOTE  
598 ED BELFOUR  
Cell [1,6,7] = 1 Items Pink

382 KENT MANDERVILLE  
Cell [2,6,7] = 2 Items Pink  
302 V. BUTSAYEV  
312 PATRICE BRISEBOIS  
Cell [3,6,7] = 2 Items Pink  
222 SHAUN VAN ALLEN  
257 MIKE HOUGH  
Cell [6,6,7] = 1 Items White  
091 GLEN WESLEY  
Cell [3,7,7] = 1 Items White  
243 BRAD DALGARNO  
Cell [6,7,7] = 2 Items White  
115 MATT SCHNEIDER  
116 NEAL BROTEN  
Cell [11,7,7] = 1 Items White  
009 JAROMIR JAGR  
Cell [7,8,7] = 1 Items White  
075 MIKE GARTNER  
Cell [1,9,7] = 1 Items White  
418 CRAIG MUNI  
Cell [3,9,7] = 1 Items White  
226 BRUCE DRIVER  
Cell [5,9,7] = 1 Items White  
125 SYLVAIN COTE  
Cell [9,9,7] = 1 Items White  
040 BRIAN LEETCH  
Cell [4,0,8] = 1 Items White  
201 MARK LAMB  
Cell [1,4,8] = 1 Items Pink  
396 STEPHEN LEACH  
Cell [4,4,8] = 1 Items White  
186 BRENT FEDYK  
Cell [5,4,8] = 1 Items White  
156 GORD MURPHY  
Cell [6,4,8] = 1 Items White  
102 PETR KLIMA  
Cell [0,5,8] = 4 Items Red  
545 BRAD MCCRIMMON  
560 RICHARD PILON  
646 MIKE HARTMAN  
716 FRANCOIS LEROUX  
Cell [1,5,8] = 3 Items Red  
414 SCOTT LACHANCE  
427 GERARD GALLANT  
486 KEITH ACTON  
Cell [2,5,8] = 3 Items Red  
301 KEN SUTTON  
345 KIRK MALTBY  
368 ADAM BURT  
Cell [4,5,8] = 1 Items White  
159 RANDY BURRIDGE  
Cell [5,5,8] = 1 Items White  
136 ZARLEY ZALAPSKI  
Cell [6,5,8] = 1 Items White



085 DIMITRI KHRISTICH  
Cell [8,5,8] = 1 Items White  
065 BENOIT HOGUE  
Cell [0,6,8] = 3 Items Red  
591 CHRIS LIPUMA  
656 PHILIP CROWE  
658 SHAWN CRONIN  
Cell [1,6,8] = 2 Items Pink  
458 GERALD DIDUCK  
475 RANDY LADOUCEUR  
Cell [2,6,8] = 2 Items Pink  
351 KEVIN LOWE  
365 CHARLIE HUDDY  
Cell [3,6,8] = 2 Items Pink  
259 PAT CONACHER  
264 CLAUDE LAPOINTE  
Cell [4,6,8] = 2 Items White  
184 DOUG BODGER  
169 BRIAN NOONAN  
Cell [7,6,8] = 1 Items White  
080 TREVOR LINDEN  
Cell [10,6,8] = 1 Items White  
020 VINCENT DAMPHOUSSE  
Cell [1,7,8] = 1 Items Pink  
417 J.J. DAIGNEAULT  
Cell [2,7,8] = 1 Items White  
318 ALEXANDER GODYNYUK  
Cell [4,7,8] = 1 Items White  
185 RANDY WOOD  
Cell [6,7,8] = 1 Items White  
092 DINO CICCARELLI  
Cell [1,8,8] = 1 Items White  
429 DENNIS VASKE  
Cell [4,8,8] = 1 Items White  
170 RICHARD SMEHLIK  
Cell [9,8,8] = 1 Items White  
030 MARK MESSIER  
Cell [3,9,8] = 1 Items White  
219 BOBBY HOLIK  
Cell [6,10,8] = 1 Items White  
093 VALERI ZELEPUKIN  
Cell [8,1,9] = 1 Items White  
049 NELSON EMERSON  
Cell [1,3,9] = 1 Items White  
392 DIXON WARD  
Cell [0,4,9] = 1 Items Pink  
529 SERGEI BAUTIN  
Cell [1,4,9] = 2 Items Pink  
467 RICK ZOMBO  
471 DAVID SHAW  
Cell [3,4,9] = 1 Items Pink  
273 SEAN HILL  
Cell [0,5,9] = 2 Items Pink  
613 JIM CUMMINS

681 BRENT THOMPSON  
Cell [3,5,9] = 1 Items Pink  
268 JOCELYN LEMIEUX  
Cell [5,5,9] = 1 Items White  
131 JOSEF BERANEK  
Cell [6,5,9] = 1 Items White  
087 TOMAS SANDSTROM  
Cell [11,5,9] = 1 Items White  
013 ROD BRIND'AMOUR  
Cell [4,6,9] = 1 Items White  
202 DMITRI MIRONOV  
Cell [5,6,9] = 2 Items White  
119 BRYAN SMOLINSKI  
133 KEVIN MILLER  
Cell [6,6,9] = 1 Items White  
112 RAY FERRARO  
Cell [1,7,9] = 1 Items Pink  
383 CRAIG WOLANIN  
Cell [5,7,9] = 1 Items White  
149 PAUL CAVALLINI  
Cell [7,7,9] = 2 Items White  
068 SERGEI MAKAROV  
079 DAVE GAGNER  
Cell [1,9,9] = 1 Items White  
457 SYLVAIN LEFEBVRE  
Cell [6,9,9] = 1 Items White  
096 VLADIMIR MALAKHOV  
Cell [2,0,10] = 1 Items White  
332 GORD DINEEN  
Cell [5,3,10] = 1 Items White  
157 BOB BEERS  
Cell [6,3,10] = 1 Items White  
088 DARRIN SHANNON  
Cell [10,3,10] = 1 Items White  
024 LUC ROBITAILLE  
Cell [2,4,10] = 1 Items White  
305 JOEL OTTO  
Cell [3,4,10] = 1 Items White  
215 CRAIG MACTAVISH  
Cell [8,4,10] = 1 Items White  
064 GARRY GALLEY  
Cell [0,5,10] = 1 Items White  
559 BOB HALKIDIS  
Cell [1,5,10] = 1 Items White  
391 MARC BERGEVIN  
Cell [2,5,10] = 2 Items White  
342 TROY LONEY  
350 BILL BERG  
Cell [3,5,10] = 1 Items White  
247 DIMITRI YUSHKEVICH  
Cell [4,5,10] = 1 Items White  
211 DARRYL SYDOR  
Cell [0,6,10] = 2 Items White  
564 KEVIN KAMINSKI

588 STEWART MALGUNAS  
Cell [2,6,10] = 1 Items White  
289 BOB SWEENEY  
Cell [4,6,10] = 2 Items White  
193 JIRI SLEGR  
174 RON SUTTER  
Cell [12,6,10] = 1 Items White  
005 PAVEL BURE  
Cell [0,7,10] = 1 Items White  
543 JASON BOWEN  
Cell [1,7,10] = 1 Items White  
390 PETR SVOBODA  
Cell [5,7,10] = 1 Items White  
148 CLAUDE LEMIEUX  
Cell [10,7,10] = 1 Items White  
025 DALE HAWERCHUK  
Cell [5,8,10] = 1 Items White  
137 BERNIE NICHOLLS  
Cell [1,3,11] = 1 Items White  
437 MIKE EAGLES  
Cell [0,4,11] = 1 Items Pink  
602 CLAUDE BOIVIN  
Cell [3,4,11] = 1 Items White  
263 BOB BASSEN  
Cell [6,4,11] = 1 Items White  
117 ALEXEI ZHITNIK  
Cell [0,5,11] = 2 Items Pink  
595 TONY TWIST  
718 GORD DONNELLY  
Cell [1,5,11] = 1 Items Pink  
432 BILL LINDSAY  
Cell [2,5,11] = 1 Items White  
296 RANDY CUNNEYWORTH  
Cell [4,5,11] = 1 Items White  
208 ERIC DESJARDINS  
Cell [6,5,11] = 1 Items White  
094 KIRK MULLER  
Cell [7,5,11] = 1 Items White  
072 TODD ELIK  
Cell [1,6,11] = 2 Items Pink  
387 BOB ROUSE  
473 JOE CIRELLA  
Cell [5,6,11] = 1 Items White  
145 GARRY VALK  
Cell [5,7,11] = 1 Items White  
147 BILL GUERIN  
Cell [11,8,11] = 1 Items White  
010 DAVE ANDREYCHUK  
Cell [8,9,11] = 1 Items White  
059 JOHN MACLEAN  
Cell [9,10,11] = 1 Items White  
033 AL MACINNIS  
Cell [3,2,12] = 1 Items White  
235 BORIS MIRONOV

Cell [1,4,12] = 1 Items Pink  
452 SCOTT THORNTON  
Cell [4,4,12] = 1 Items White  
173 KEVIN HATCHER  
Cell [0,5,12] = 3 Items Red  
540 MATTHEW BARNABY  
547 DAVID MALEY  
616 BASIL MCRAE  
Cell [3,5,12] = 2 Items White  
246 BRIAN BENNING  
280 RICH SUTTER  
Cell [5,5,12] = 1 Items White  
138 DENIS SAVARD  
Cell [1,6,12] = 1 Items Pink  
487 JAY WELLS  
Cell [2,6,12] = 2 Items Pink  
313 MICHEL PETIT  
338 DANA MURZYN  
Cell [7,6,12] = 1 Items White  
067 JASON ARNOTT  
Cell [8,6,12] = 1 Items White  
060 JOE MURPHY  
Cell [0,7,12] = 1 Items Pink  
633 PAUL LAUS  
Cell [11,7,12] = 1 Items White  
012 ERIC LINDROS  
Cell [2,8,12] = 1 Items White  
285 WAYNE PRESLEY  
Cell [12,8,12] = 1 Items White  
004 DOUG GILMOUR  
Cell [8,9,12] = 1 Items White  
043 PAUL COFFEY  
Cell [1,0,13] = 1 Items White  
394 DARREN RUMBLE  
Cell [3,3,13] = 1 Items White  
284 STEPHANE QUINTAL  
Cell [0,5,13] = 1 Items Pink  
587 RYAN MCGILL  
Cell [1,5,13] = 1 Items Pink  
433 ALAN MAY  
Cell [3,5,13] = 2 Items White  
248 CHRIS PRONGER  
249 JAMIE MACOUN  
Cell [4,5,13] = 1 Items White  
162 KEVIN DINEEN  
Cell [5,5,13] = 1 Items White  
120 MIKE RICCI  
Cell [6,5,13] = 1 Items White  
108 SHAYNE CORSON  
Cell [1,6,13] = 3 Items Red  
428 KEVIN HALLER  
440 NEIL WILKINSON  
484 RUDY POESCHEK  
Cell [5,6,13] = 1 Items White

139 MIKE KEANE  
Cell [6,6,13] = 1 Items White  
109 ESA TIKKANEN  
Cell [8,7,13] = 1 Items White  
044 WENDEL CLARK  
Cell [1,8,13] = 1 Items White  
426 KJELL SAMUELSSON  
Cell [9,12,13] = 1 Items White  
041 SCOTT STEVENS  
Cell [4,3,14] = 1 Items White  
164 CHRIS GRATTON  
Cell [0,4,14] = 1 Items Pink  
541 GRANT JENNINGS  
Cell [1,4,14] = 1 Items Pink  
413 MURRAY BARON  
Cell [3,4,14] = 1 Items White  
238 BOB ERREY  
Cell [1,5,14] = 1 Items Pink  
419 CRAIG LUDWIG  
Cell [0,6,14] = 1 Items Pink  
594 DEAN CHYNOWETH  
Cell [8,7,14] = 1 Items White  
063 GEOFF COURTNALL  
Cell [5,8,14] = 1 Items White  
140 STEVE CHIASSON  
Cell [12,8,14] = 1 Items White  
006 JEREMY ROENICK  
Cell [9,9,14] = 1 Items White  
037 ADAM GRAVES  
Cell [0,3,15] = 1 Items White  
531 ROB ZETTLER  
Cell [3,3,15] = 1 Items White  
233 CHRIS JOSEPH  
Cell [0,4,15] = 2 Items Pink  
509 LUKE RICHARDSON  
562 DAVE BROWN  
Cell [1,4,15] = 1 Items Pink  
442 MARK JANSSENS  
Cell [2,4,15] = 1 Items White  
330 ROMAN HAMRLIK  
Cell [4,4,15] = 1 Items White  
176 RICK TOCCHET  
Cell [0,5,15] = 3 Items Red  
495 CHRIS SIMON  
563 DOUG ZMOLEK  
603 JOEY KOEUR  
Cell [4,5,15] = 1 Items White  
191 DALE HUNTER  
Cell [7,5,15] = 1 Items White  
070 ROB BLAKE  
Cell [0,6,15] = 1 Items Pink  
533 DARIN KIMBLE  
Cell [0,7,15] = 1 Items White  
528 TERRY CARKNER

Cell [4,7,15] = 1 Items White  
175 BRIAN SKRUDLAND  
Cell [0,8,15] = 1 Items White  
557 ROBERT DIRK  
Cell [0,4,16] = 1 Items Pink  
502 NICK KYPREOS  
Cell [4,4,16] = 1 Items White  
200 MIKE CRAIG  
Cell [0,5,16] = 1 Items Pink  
721 JAMIE HUSCROFT  
Cell [1,5,16] = 1 Items Pink  
459 DARIUS KASPARAITIS  
Cell [2,5,16] = 1 Items White  
354 GARTH BUTCHER  
Cell [8,5,16] = 1 Items White  
045 STEVE THOMAS  
Cell [1,6,16] = 1 Items Pink  
416 PETER TAGLIANETTI  
Cell [2,6,16] = 2 Items White  
290 MARK OSBORNE  
298 MARK TINORDI  
Cell [2,7,16] = 1 Items White  
295 BRENT HUGHES  
Cell [3,9,16] = 1 Items White  
220 VLAD. KONSTANTINOV  
Cell [9,10,16] = 1 Items White  
029 GARY ROBERTS  
Cell [0,5,17] = 1 Items White  
504 JIM MCKENZIE  
Cell [1,5,17] = 2 Items White  
430 RANDY MOLLER  
489 GLEN FEATHERSTONE  
Cell [2,5,17] = 1 Items White  
324 TONY GRANATO  
Cell [3,5,17] = 1 Items White  
269 SERGIO MOMESSO  
Cell [1,6,17] = 1 Items White  
372 DAVE KARPA  
Cell [4,6,17] = 1 Items White  
204 KEITH JONES  
Cell [6,6,17] = 1 Items White  
081 SCOTT MELLANBY  
Cell [6,8,17] = 1 Items White  
098 ALEXEI KOVALEV  
Cell [0,3,18] = 1 Items White  
576 BILL HUARD  
Cell [2,3,18] = 1 Items White  
348 SCOTT LEVINS  
Cell [10,3,18] = 1 Items White  
023 KEVIN STEVENS  
Cell [1,5,18] = 2 Items White  
373 STEVEN FINN  
453 BRENT SEVERYN  
Cell [1,7,18] = 1 Items White

375 JOE REEKIE  
Cell [2,2,19] = 1 Items White  
307 TROY MALLETTE  
Cell [1,3,19] = 1 Items White  
468 DEAN KENNEDY  
Cell [1,4,19] = 2 Items White  
376 IGOR ULANOV  
400 BRYAN MARCHMENT  
Cell [0,5,19] = 1 Items White  
518 TIM HUNTER  
Cell [4,5,19] = 1 Items White  
194 SCOTT PEARSON  
Cell [5,5,19] = 1 Items White  
146 BRAD MAY  
Cell [6,7,19] = 1 Items White  
090 AL IAFRATE  
Cell [1,8,19] = 1 Items White  
380 JEFF BEUKEBOOM  
Cell [8,4,20] = 1 Items White  
046 PAT VERBEEK  
Cell [1,5,20] = 1 Items White  
463 SANDY MCCARTHY  
Cell [3,5,20] = 1 Items White  
275 STEVE SMITH  
Cell [1,9,20] = 1 Items White  
474 KEN DANEYKO  
Cell [8,10,20] = 1 Items White  
052 KEITH PRIMEAU  
Cell [0,3,21] = 1 Items White  
508 DOUG HOUDA  
Cell [0,5,21] = 2 Items White  
497 KEN BAUMGARTNER  
647 GREG SMYTH  
Cell [1,5,21] = 2 Items White  
456 PAUL KRUSE  
464 LOUIE DEBRUSK  
Cell [3,7,21] = 1 Items White  
283 DARREN MCCARTY  
Cell [9,9,21] = 1 Items White  
026 THEOREN FLEURY  
Cell [0,4,22] = 1 Items White  
617 SHAWN ANTOSKI  
Cell [2,4,22] = 1 Items White  
329 DAVE MANSON  
Cell [3,4,22] = 1 Items White  
234 MARTY MCSORLEY  
Cell [3,5,22] = 1 Items White  
240 ROB PEARSON  
Cell [1,3,23] = 1 Items White  
438 KRIS KING  
Cell [2,3,23] = 1 Items White  
331 KELLY BUCHBERGER  
Cell [0,5,23] = 1 Items White  
546 STU GRIMSON

Cell [0,7,23] = 1 Items White  
 511 CAM RUSSELL  
 Cell [3,8,23] = 1 Items White  
 258 ULF SAMUELSSON  
 Cell [0,5,24] = 1 Items White  
 522 DENNIS VIAL  
 Cell [11,5,24] = 1 Items White  
 008 BRENDAN SHANAHAN  
 Cell [6,7,24] = 1 Items White  
 083 CHRIS CHELIOS  
 Cell [3,8,24] = 1 Items White  
 232 DERIAN HATCHER  
 Cell [2,4,25] = 1 Items White  
 320 JEFF ODGERS  
 Cell [0,5,26] = 1 Items White  
 618 ENRICO CICCONE  
 Cell [0,5,27] = 1 Items White  
 568 MICK VUKOTA  
 Cell [2,8,27] = 1 Items White  
 340 MIKE PELUSO  
 Cell [3,6,28] = 1 Items White  
 255 RONNIE STERN  
 Cell [3,8,28] = 1 Items White  
 271 RANDY MCKAY  
 Cell [9,4,29] = 1 Items White  
 035 KEITH TKACHUK  
 Cell [0,5,31] = 1 Items White  
 556 MARC POTVIN  
 Cell [2,5,31] = 1 Items White  
 358 TODD EWEN  
 Cell [0,6,31] = 1 Items White  
 519 ROB RAY  
 Cell [3,7,31] = 1 Items White  
 250 GINO ODJICK  
 Cell [0,5,32] = 1 Items White  
 523 KELLY CHASE  
 Cell [1,5,32] = 1 Items White  
 370 BOB PROBERT  
 Cell [4,6,32] = 1 Items White  
 179 LYLE ODELEIN  
 Cell [1,5,35] = 1 Items White  
 408 CRAIG BERUBE  
 Cell [2,3,37] = 1 Items White  
 346 WARREN RYCHEL  
 Cell [1,5,38] = 1 Items White  
 424 SHANE CHURLA  
 Cell [2,5,40] = 1 Items White  
 349 TIE DOMI

### 14.3.2. LOG DE CONTROL DE CELDAS BLANCAS

Analizing Cell [6,4,0]  
 L1 Count 4  
 Analizing Cell [5,6,0]  
 L1 Count 5



```

Analizing Cell [2,3,1]
L1 Count 8
Analizing Cell [5,4,1]
L1 Count 8
Analizing Cell [5,5,1]
L1 Count 12
Analizing Cell [8,5,1]
L1 Count 3
Analizing Cell [3,7,1]
L1 Count 17
Analizing Cell [4,8,1]
L1 Count 3
Analizing Cell [8,9,1]
L1 Count 2
L2 Count 78
Distance to 197 IAIN FRASER           50,6952
Distance to 199 GREG ADAMS            47,5920
Distance to 196 MARIO LEMIEUX        50,6952
Distance to 207 PAUL YSEBAERT        53,9722
Distance to 209 FREDRIK OLAUSSON     54,8544
Distance to 210 JAMES PATRICK         56,9298
Distance to 178 DALLAS DRAKE         56,0089
Distance to 177 RAY WHITNEY          43,2319
Distance to 206 NIKOLAI BORSCHEVSKY 45,0555
Distance to 183 KELLY MILLER         43,2897
Distance to 160 MIKE DONNELLY        45,4423
Distance to 203 MIKE MCPHEE          47,9375
Distance to 163 TONY AMONTE          47,9792
Distance to 180 TOM KURVERS          50,6063
Distance to 188 TRENT KLATT          40,9390
Distance to 161 SHAWN MCEACHERN      38,8973
Distance to 167 TOM CHORSKE          38,8072
Distance to 172 JOHN TUCKER          40,6448
Distance to 181 JEFF NORTON          40,6079
Distance to 165 PATRICK FLATLEY      42,8602
Distance to 189 GUY CARBONNEAU       49,0408
Distance to 171 TRAVIS GREEN         45,0888
Distance to 190 WES WALZ             35,7351
Distance to 192 BRENT SUTTER         45,6180
Distance to 121 SCOTT YOUNG          40,8656
Distance to 128 JESSE BELANGER       41,3401
Distance to 144 GILBERT DIONNE       50,5371
Distance to 126 ROBERT KRON          38,8458
Distance to 122 BOB CORKUM           34,2637
Distance to 135 ANDREI LOMAKIN       40,3113
Distance to 152 DANTON COLE          40,9023
Distance to 129 MICHAL PIVONKA       42,5323
Distance to 134 STU BARNES           42,8719
Distance to 142 GRANT LEDYARD        43,9090
Distance to 158 DAVE ELLETT          46,2817
Distance to 155 TIM SWEENEY          52,0961
Distance to 132 SERGEI NEMCHINOV    35,3977
Distance to 143 GERMAN TITOV         31,5278
Distance to 153 JOHN LECLAIR         35,9583

```

Distance to 151 PETER BONDRA 38,7040  
 Distance to 141 SCOTT NIEDERMAYER 36,8917  
 Distance to 111 PAT FALLOON 39,0000  
 Distance to 099 DEREK PLANTE 32,3883  
 Distance to 104 MURRAY CRAVEN 34,0735  
 Distance to 105 MICHAEL NYLANDER 31,8434  
 Distance to 084 DONALD AUDETTE 40,4351  
 Distance to 106 JYRKI LUMME 47,2123  
 Distance to 114 TERRY YAKE 45,0000  
 Distance to 113 JOHAN GARPENLOV 31,4484  
 Distance to 082 STEVE LARMER 32,5269  
 Distance to 086 YURI KHYLEV 40,1123  
 Distance to 103 DOUG BROWN 20,9045  
 Distance to 100 IGOR LARIONOV 30,8707  
 Distance to 097 MARTY MCINNIS 17,8885

Number of players within 29,6985 distance: 3  
 Analizing Cell [9,0,2]

L1 Count 1

L2 Count 14

Distance to 123 ALEXANDRE DAIGLE 33,5261  
 Distance to 124 THOMAS STEEN 31,7017  
 Distance to 154 DAVE MCLLWAIN 45,3100  
 Distance to 130 IGOR KRAVCHUK 47,3920  
 Distance to 107 TEEMU SELANNE 36,0694  
 Distance to 089 ANDREW CASSELS 38,0789  
 Distance to 095 ZDENO CIGER 46,0869  
 Distance to 118 YVES RACINE 53,3760  
 Distance to 071 GEOFF SANDERSON 42,8952  
 Distance to 073 JEFF BROWN 52,6213  
 Distance to 057 BOB KUDELSKI 18,4662  
 Distance to 042 JARI KURRI 36,1248  
 Distance to 050 DOUG WEIGHT 37,1349

Number of players within 29,6985 distance: 1  
 Analizing Cell [8,2,2]

L1 Count 1

L2 Count 63

Distance to 182 BILL HOULDER 37,9737  
 Distance to 205 ROB GAUDREAU 42,0714  
 Distance to 197 IAIN FRASER 43,2897  
 Distance to 199 GREG ADAMS 46,1411  
 Distance to 196 MARIO LEMIEUX 45,9783  
 Distance to 207 PAUL YSEBAERT 43,6463  
 Distance to 209 FREDRIK OLAUSSON 44,0114  
 Distance to 210 JAMES PATRICK 47,5920  
 Distance to 178 DALLAS DRAKE 50,4876  
 Distance to 177 RAY WHITNEY 47,1699  
 Distance to 206 NIKOLAI BORSCHEVSKY 54,2402  
 Distance to 183 KELLY MILLER 52,0192  
 Distance to 160 MIKE DONNELLY 45,9239  
 Distance to 203 MIKE MCPHEE 55,2268  
 Distance to 163 TONY AMONTE 45,1885  
 Distance to 180 TOM KURVERS 55,0364  
 Distance to 166 CALLE JOHANSSON 57,4891  
 Distance to 123 ALEXANDRE DAIGLE 27,5862

Distance to 124 THOMAS STEEN 21,2132  
 Distance to 154 DAVE MCLLWAIN 36,7967  
 Distance to 130 IGOR KRAVCHUK 30,0832  
 Distance to 121 SCOTT YOUNG 36,0832  
 Distance to 128 JESSE BELANGER 36,1248  
 Distance to 144 GILBERT DIONNE 35,3553  
 Distance to 126 ROBERT KRON 41,7732  
 Distance to 122 BOB CORKUM 42,0238  
 Distance to 135 ANDREI LOMAKIN 41,0974  
 Distance to 152 DANTON COLE 48,9183  
 Distance to 129 MICHAL PIVONKA 42,6732  
 Distance to 134 STU BARNES 45,7602  
 Distance to 142 GRANT LEDYARD 50,0000  
 Distance to 158 DAVE ELLETT 50,7346  
 Distance to 155 TIM SWEENEY 51,4782  
 Distance to 107 TEEMU SELANNE 18,9737

Number of players within 29,6985 distance: 3

Analizing Cell [0,3,2]

L1 Count 10

Analizing Cell [3,3,2]

L1 Count 8

Analizing Cell [6,3,2]

L1 Count 2

L2 Count 168

Distance to 310 BRAD SHAW 51,5170  
 Distance to 366 KERRY HUFFMAN 40,5586  
 Distance to 363 VLADIMIR RUZICKA 36,9324  
 Distance to 309 TEPPU NUMMINEN 31,5753  
 Distance to 311 NORM MACIVER 31,4006  
 Distance to 352 DARREN TURCOTTE 36,7423  
 Distance to 364 LUCIANO BORSATO 38,4187  
 Distance to 360 GARY SUTER 40,9023  
 Distance to 361 TED DRURY 39,4462  
 Distance to 362 PAT LAFONTAINE 45,3542  
 Distance to 316 PHIL HOUSLEY 38,0526  
 Distance to 341 PELLE EKLUND 41,7253  
 Distance to 343 DMITRI KVARTALNOV 39,5601  
 Distance to 353 VLADIMIR VUJTEK 39,3065  
 Distance to 357 DAVID ARCHIBALD 40,1995  
 Distance to 286 VITALI PROKHOROV 33,6749  
 Distance to 315 KEVIN TODD 37,7889  
 Distance to 321 STEPHEN HEINZE 40,3733  
 Distance to 333 ADAM CREIGHTON 40,4599  
 Distance to 288 MIKE STAPLETON 41,7253  
 Distance to 304 ROB DIMAIO 43,5660  
 Distance to 322 VITALI KARAMNOV 48,2701  
 Distance to 328 FRANTISEK KUCERA 46,5296  
 Distance to 335 EVGENY DAVYDOV 43,9090  
 Distance to 337 JIM MONTGOMERY 46,0869  
 Distance to 292 DAVE POULIN 47,1699  
 Distance to 303 PATRIK CARNBACK 47,0106  
 Distance to 319 DAN KECZMER 51,8941  
 Distance to 336 PETR NEDVED 44,4635  
 Distance to 306 JOZEF STUMPEL 41,8808

Distance to 347 ROBERT LANG 47,6340  
 Distance to 287 MIKAEL ANDERSSON 42,4617  
 Distance to 356 BRENT GRIEVE 45,0111  
 Distance to 293 TOM PEDERSON 39,9750  
 Distance to 359 MIKE EASTWOOD 44,2380  
 Distance to 291 PHILIPPE BOZON 44,3847  
 Distance to 294 ALEXEI GUSAROV 42,1070  
 Distance to 339 DAVID WILLIAMS 50,1697  
 Distance to 299 MARK HOWE 51,1566  
 Distance to 308 DAVE REID 45,3762  
 Distance to 314 SHAWN BURR 48,2701  
 Distance to 323 UWE KRUPP 48,0521  
 Distance to 327 DAVE HANNAN 56,0268  
 Distance to 367 A. KARPOVTSEV 61,7819  
 Distance to 272 PAUL RANHEIM 27,4591  
 Distance to 281 SYLVAIN TURGEON 41,0853  
 Distance to 265 JIMMY CARSON 27,2764

Number of players within 29,6985 distance: 3  
 Analyzing Cell [15,3,2]

L1 Count 1

L2 Count 158

Distance to 011 BRETT HULL 43,5546  
 Distance to 007 MARK RECCHI 41,6413  
 Distance to 003 ADAM OATES 46,6262  
 Distance to 388 WAYNE MCBEAN 116,0905  
 Distance to 401 DAVE TIPPETT 116,5075  
 Distance to 404 BRIAN GLYNN 119,7915  
 Distance to 434 RUSS ROMANIUK 119,6495  
 Distance to 448 TONY HRKAC 120,4201  
 Distance to 449 TIM BERGLAND 119,8374  
 Distance to 482 DALE CRAIGWELL 121,6594  
 Distance to 389 JEFF SHANTZ 114,9652  
 Distance to 385 IGOR KOROLEV 116,4689  
 Distance to 399 BRYAN TROTTIER 116,8332  
 Distance to 407 RANDY GILHEN 120,5197  
 Distance to 491 GEOFF SMITH 125,2398  
 Distance to 377 CRAIG SIMPSON 116,7219  
 Distance to 386 PETER STASTNY 117,3925  
 Distance to 470 ADRIEN PLAVSIC 122,4582  
 Distance to 409 CHRIS LINDBERG 118,7266  
 Distance to 420 ANDRE FAUST 119,8541  
 Distance to 441 RON WILSON 120,4865  
 Distance to 476 JARROD SKALDE 123,3896  
 Distance to 479 MIKE SULLIVAN 123,7619  
 Distance to 492 SHAWN ANDERSON 123,6163  
 Distance to 371 GREG JOHNSON 114,4421  
 Distance to 461 DAN MAROIS 121,8401  
 Distance to 393 MARC BUREAU 116,5375  
 Distance to 402 GREG GILBERT 117,4308  
 Distance to 410 PHILIPPE BOUCHER 118,7981  
 Distance to 423 SHELDON KENNEDY 119,6578  
 Distance to 450 MIKE KRUSHELNYSKI 120,9339  
 Distance to 454 MARKUS NASLUND 121,2188  
 Distance to 422 MARK GREIG 120,5404

|                 |                    |          |
|-----------------|--------------------|----------|
| Distance to 431 | ROB ZAMUNER        | 121,0950 |
| Distance to 447 | ROBERT PETROVICKY  | 122,8739 |
| Distance to 483 | JOHN MCINTYRE      | 123,3734 |
| Distance to 406 | JOSE CHARBONNEAU   | 121,5771 |
| Distance to 446 | STEVEN KING        | 122,7233 |
| Distance to 451 | MIKE HUDSON        | 123,6527 |
| Distance to 480 | ADAM BENNETT       | 125,5826 |
| Distance to 465 | PAUL STANTON       | 126,0159 |
| Distance to 469 | MIKE RATHJE        | 127,1888 |
| Distance to 395 | JIM DOWD           | 121,3013 |
| Distance to 398 | IGOR CHIBIREV      | 120,7187 |
| Distance to 462 | DAVID EMMA         | 123,8911 |
| Distance to 397 | GARY LEEMAN        | 118,8865 |
| Distance to 412 | DAVE VOLEK         | 119,0840 |
| Distance to 421 | JASON DAWE         | 120,1208 |
| Distance to 485 | DOUG CROSSMAN      | 124,1652 |
| Distance to 378 | PETER ZESEL        | 117,8855 |
| Distance to 443 | TOMMY SJODIN       | 121,5278 |
| Distance to 460 | BRAD WERENKA       | 122,2661 |
| Distance to 381 | JOHN SLANEY        | 117,5968 |
| Distance to 384 | JIM STORM          | 117,8389 |
| Distance to 455 | GREG BROWN         | 122,0697 |
| Distance to 369 | JAMIE BAKER        | 117,5670 |
| Distance to 403 | MIKE MCKEE         | 120,6897 |
| Distance to 411 | ED RONAN           | 121,3425 |
| Distance to 439 | DON MCSWEEN        | 122,9878 |
| Distance to 478 | PIERRE SEVIGNY     | 126,8306 |
| Distance to 379 | MARTIN LAPOINTE    | 123,4706 |
| Distance to 444 | CHRIS DAHLQUIST    | 125,8888 |
| Distance to 477 | MIKE FOLIGNO       | 129,4372 |
| Distance to 374 | JAROSLAV MODRY     | 118,3131 |
| Distance to 488 | JEFF FINLEY        | 127,8202 |
| Distance to 405 | VIACHESLAV FETISOV | 121,8442 |
| Distance to 415 | PETER POPOVIC      | 121,3136 |
| Distance to 425 | KRIS DRAPER        | 122,9065 |
| Distance to 435 | KEITH CARNEY       | 125,7219 |
| Distance to 436 | KEITH BROWN        | 129,6919 |
| Distance to 310 | BRAD SHAW          | 115,0043 |
| Distance to 366 | KERRY HUFFMAN      | 113,8112 |
| Distance to 363 | VLADIMIR RUZICKA   | 112,2319 |
| Distance to 309 | TEPPO NUMMINEN     | 107,3173 |
| Distance to 311 | NORM MACIVER       | 107,1728 |
| Distance to 352 | DARREN TURCOTTE    | 111,6871 |
| Distance to 364 | LUCIANO BORSATO    | 113,1548 |
| Distance to 360 | GARY SUTER         | 114,1797 |
| Distance to 361 | TED DRURY          | 113,5782 |
| Distance to 362 | PAT LAFONTAINE     | 115,3646 |
| Distance to 316 | PHIL HOUSLEY       | 110,1272 |
| Distance to 341 | PELLE EKLUND       | 112,8229 |
| Distance to 343 | DMITRI KVARTALNOV  | 112,5922 |
| Distance to 353 | VLADIMIR VUJTEK    | 112,6099 |
| Distance to 357 | DAVID ARCHIBALD    | 113,5958 |
| Distance to 286 | VITALI PROKHOROV   | 106,7052 |
| Distance to 315 | KEVIN TODD         | 110,2905 |

|                 |                  |          |
|-----------------|------------------|----------|
| Distance to 321 | STEPHEN HEINZE   | 112,0446 |
| Distance to 333 | ADAM CREIGHTON   | 112,7519 |
| Distance to 288 | MIKE STAPLETON   | 110,0045 |
| Distance to 304 | ROB DIMAIO       | 112,0982 |
| Distance to 322 | VITALI KARAMNOV  | 115,4383 |
| Distance to 328 | FRANTISEK KUCERA | 114,6691 |
| Distance to 335 | EVGENY DAVYDOV   | 114,1578 |
| Distance to 337 | JIM MONTGOMERY   | 115,1173 |
| Distance to 292 | DAVE POULIN      | 112,3610 |
| Distance to 303 | PATRIK CARNBACK  | 113,5518 |
| Distance to 319 | DAN KECZMER      | 116,4174 |
| Distance to 336 | PETR NEDVED      | 113,8991 |
| Distance to 306 | JOZEF STUMPEL    | 111,0225 |
| Distance to 347 | ROBERT LANG      | 115,9526 |
| Distance to 287 | MIKAEL ANDERSSON | 110,1045 |
| Distance to 356 | BRENT GRIEVE     | 115,6979 |
| Distance to 293 | TOM PEDERSON     | 109,2245 |
| Distance to 359 | MIKE EASTWOOD    | 115,4859 |
| Distance to 291 | PHILIPPE BOZON   | 111,1306 |
| Distance to 294 | ALEXEI GUSAROV   | 110,1499 |
| Distance to 339 | DAVID WILLIAMS   | 116,9316 |
| Distance to 299 | MARK HOWE        | 114,2847 |
| Distance to 308 | DAVE REID        | 112,6898 |
| Distance to 314 | SHAWN BURR       | 114,6909 |
| Distance to 323 | UWE KRUPP        | 115,2259 |
| Distance to 327 | DAVE HANNAN      | 119,1428 |
| Distance to 367 | A. KARPOVTSEV    | 123,9234 |
| Distance to 297 | PAUL BROTEN      | 114,8260 |
| Distance to 355 | TOMMY ALBELIN    | 120,8387 |
| Distance to 325 | TRENT YAWNEY     | 124,8879 |
| Distance to 334 | BOBBY DOLLAS     | 123,8951 |
| Distance to 272 | PAUL RANHEIM     | 103,2570 |
| Distance to 281 | SYLVAIN TURGEON  | 108,8118 |
| Distance to 265 | JIMMY CARSON     | 102,5671 |
| Distance to 224 | STEVEN RICE      | 100,4241 |
| Distance to 282 | ROB NIEDERMAYER  | 109,4212 |
| Distance to 242 | ANATOLI SEMENOV  | 102,4939 |
| Distance to 212 | PETER DOURIS     | 98,0663  |
| Distance to 213 | SHAWN CHAMBERS   | 97,9081  |
| Distance to 277 | STEPHAN LEBEAU   | 106,1179 |
| Distance to 245 | KELLY KISIO      | 102,1029 |
| Distance to 260 | MARTIN GELINAS   | 104,3504 |
| Distance to 266 | ILYA BYAKIN      | 104,8237 |
| Distance to 218 | PAUL DI PIETRO   | 101,1286 |
| Distance to 216 | ANDREI KOVALENKO | 102,3963 |
| Distance to 228 | GLEN MURRAY      | 105,6456 |
| Distance to 261 | PATRICK POULIN   | 107,9537 |
| Distance to 223 | TOM FITZGERALD   | 106,0377 |
| Distance to 239 | TODD KRYGIER     | 109,7315 |
| Distance to 270 | OLEG PETROV      | 109,3481 |
| Distance to 278 | JODY HULL        | 109,1833 |
| Distance to 231 | STEVE DUCHESNE   | 102,5329 |
| Distance to 256 | MIKE SILLINGER   | 105,0238 |
| Distance to 251 | ALEXANDER SEMAK  | 105,6693 |

Distance to 229 BRENT GILCHRIST 102,6986  
 Distance to 236 MICHEL GOULET 103,4988  
 Distance to 241 GAETAN DUCHESNE 105,6078  
 Distance to 253 PAT PEAKE 106,0094  
 Distance to 267 ERIC WEINRICH 106,3250  
 Distance to 221 BOB CARPENTER 106,7427  
 Distance to 252 BRIAN PROPP 107,5221  
 Distance to 276 TODD GILL 110,7881  
 Distance to 225 MARTIN RUCINSKY 109,0367  
 Distance to 227 DAVE BABYCH 106,0802  
 Distance to 274 DEAN MCAMMOND 109,5171  
 Distance to 244 BENOIT BRUNET 107,3359  
 Distance to 279 STEVE KONOWALCHUK 110,1862  
 Distance to 214 GREG HAWGOOD 103,0922  
 Distance to 217 DIRK GRAHAM 107,1354  
 Distance to 230 JOHN DRUCE 111,2744

Number of players within 29,6985 distance: 0

Analizing Cell [6,5,2]

L1 Count 11

Analizing Cell [9,5,2]

L1 Count 4

Analizing Cell [10,5,2]

L1 Count 3

Analizing Cell [6,6,2]

L1 Count 12

Analizing Cell [9,6,2]

L1 Count 7

Analizing Cell [7,7,2]

L1 Count 8

Analizing Cell [8,7,2]

L1 Count 4

Analizing Cell [10,7,2]

L1 Count 3

Analizing Cell [6,8,2]

L1 Count 7

Analizing Cell [7,8,2]

L1 Count 7

Analizing Cell [6,9,2]

L1 Count 5

Analizing Cell [5,1,3]

L1 Count 2

L2 Count 130

Distance to 388 WAYNE MCBEAN 36,1248

Distance to 401 DAVE TIPPETT 40,6940

Distance to 404 BRIAN GLYNN 45,8039

Distance to 434 RUSS ROMANIUK 54,0925

Distance to 448 TONY HRKAC 53,8981

Distance to 449 TIM BERGLAND 51,1175

Distance to 482 DALE CRAIGWELL 51,4296

Distance to 389 JEFF SHANTZ 42,4853

Distance to 385 IGOR KOROLEV 44,3283

Distance to 399 BRYAN TROTTIER 44,5870

Distance to 407 RANDY GILHEN 48,6724

Distance to 491 GEOFF SMITH 52,0865

|                 |                   |         |
|-----------------|-------------------|---------|
| Distance to 472 | TIM WATTERS       | 60,2910 |
| Distance to 377 | CRAIG SIMPSON     | 55,0091 |
| Distance to 386 | PETER STASTNY     | 57,4891 |
| Distance to 470 | ADRIEN PLAVSIC    | 58,7026 |
| Distance to 409 | CHRIS LINDBERG    | 56,0179 |
| Distance to 420 | ANDRE FAUST       | 57,4195 |
| Distance to 441 | RON WILSON        | 56,7186 |
| Distance to 476 | JARROD SKALDE     | 58,9322 |
| Distance to 479 | MIKE SULLIVAN     | 60,1415 |
| Distance to 492 | SHAWN ANDERSON    | 59,4390 |
| Distance to 371 | GREG JOHNSON      | 47,0850 |
| Distance to 461 | DAN MAROIS        | 55,0727 |
| Distance to 393 | MARC BUREAU       | 46,2709 |
| Distance to 402 | GREG GILBERT      | 50,2991 |
| Distance to 410 | PHILIPPE BOUCHER  | 52,4118 |
| Distance to 423 | SHELDON KENNEDY   | 52,3832 |
| Distance to 450 | MIKE KRUSHELNYSKI | 52,0096 |
| Distance to 454 | MARKUS NASLUND    | 53,3854 |
| Distance to 422 | MARK GREIG        | 51,1273 |
| Distance to 431 | ROB ZAMUNER       | 49,6185 |
| Distance to 447 | ROBERT PETROVICKY | 54,9363 |
| Distance to 483 | JOHN MCINTYRE     | 51,3907 |
| Distance to 406 | JOSE CHARBONNEAU  | 53,6936 |
| Distance to 446 | STEVEN KING       | 52,0096 |
| Distance to 451 | MIKE HUDSON       | 53,9815 |
| Distance to 480 | ADAM BENNETT      | 54,3415 |
| Distance to 465 | PAUL STANTON      | 55,9106 |
| Distance to 469 | MIKE RATHJE       | 57,0175 |
| Distance to 445 | BRET HEDICAN      | 59,2115 |
| Distance to 466 | DRAKE BEREHOWSKY  | 62,1852 |
| Distance to 481 | CAMERON STEWART   | 62,8013 |
| Distance to 310 | BRAD SHAW         | 39,0128 |
| Distance to 344 | ANDREW MCBAIN     | 45,3542 |
| Distance to 366 | KERRY HUFFMAN     | 35,3977 |
| Distance to 363 | VLADIMIR RUZICKA  | 41,2553 |
| Distance to 309 | TEPPO NUMMINEN    | 32,0156 |
| Distance to 311 | NORM MACIVER      | 31,0483 |
| Distance to 352 | DARREN TURCOTTE   | 43,2897 |
| Distance to 364 | LUCIANO BORSATO   | 42,8252 |
| Distance to 360 | GARY SUTER        | 42,4382 |
| Distance to 361 | TED DRURY         | 40,4228 |
| Distance to 362 | PAT LAFONTAINE    | 56,0803 |
| Distance to 316 | PHIL HOUSLEY      | 48,2701 |
| Distance to 341 | PELLE EKLUND      | 52,3546 |
| Distance to 343 | DMITRI KVARTALNOV | 48,4665 |
| Distance to 353 | VLADIMIR VUJTEK   | 48,0521 |
| Distance to 357 | DAVID ARCHIBALD   | 48,7237 |
| Distance to 286 | VITALI PROKHOROV  | 42,9418 |
| Distance to 315 | KEVIN TODD        | 46,1519 |
| Distance to 321 | STEPHEN HEINZE    | 46,8615 |
| Distance to 333 | ADAM CREIGHTON    | 44,1248 |
| Distance to 288 | MIKE STAPLETON    | 44,2832 |
| Distance to 304 | ROB DIMAIO        | 46,2169 |
| Distance to 322 | VITALI KARAMNOV   | 49,8598 |



```

Distance to 328 FRANTISEK KUCERA      48,7955
Distance to 335 EVGENY DAVYDOV      44,6990
Distance to 337 JIM MONTGOMERY      49,7393
Distance to 292 DAVE POULIN         49,4469
Distance to 303 PATRIK CARNBACK      46,5618
Distance to 319 DAN KECZMER         50,2494
Distance to 300 ALEXEI KASATONOV    53,4228
Distance to 317 CURTIS LESCHYSHYN   56,7979
Distance to 272 PAUL RANHEIM        32,8024
Distance to 281 SYLVAIN TURGEON     34,5543
Distance to 262 PAT ELYNUIK         42,9185
Distance to 265 JIMMY CARSON        33,4963
Distance to 224 STEVEN RICE         34,0735
Distance to 282 ROB NIEDERMAYER     41,4126
Distance to 242 ANATOLI SEMENOV    44,6878
Distance to 212 PETER DOURIS       38,7169
Distance to 213 SHAWN CHAMBERS     37,3363
Distance to 277 STEPHAN LEBEAU     43,3705
Distance to 245 KELLY KISIO        38,4838
Distance to 260 MARTIN GELINAS     37,8550
Distance to 266 ILYA BYAKIN        41,9285
Distance to 218 PAUL DI PIETRO     40,5586
Distance to 216 ANDREI KOVALENKO   40,1123
Distance to 228 GLEN MURRAY        45,0000
Distance to 261 PATRICK POULIN     42,3084
Distance to 223 TOM FITZGERALD     45,4973
Distance to 239 TODD KRYGIER       48,7955
Distance to 237 JOHN CULLEN        54,4243
Distance to 254 CHRISTIAN RUUTTU   54,1849
Distance to 182 BILL HOULDER       24,6577
Distance to 205 ROB GAUDREAU       32,4962
Distance to 195 JOE SACCO          42,0238
Distance to 197 IAIN FRASER        36,9594
Distance to 199 GREG ADAMS         41,3401
Distance to 196 MARIO LEMIEUX     38,6264
Distance to 207 PAUL YSEBAERT     35,3977
Distance to 209 FREDRIK OLAUSSON  34,9428
Distance to 210 JAMES PATRICK     37,5366
Distance to 178 DALLAS DRAKE       42,1782
Distance to 198 DAVE LOWRY         48,7442
Distance to 168 GLENN ANDERSON     45,7056
Distance to 154 DAVE MCLLWAIN      18,0000
Number of players within 29,6985 distance: 3
Analizing Cell [2,3,3]
L1 Count 14
Analizing Cell [10,7,3]
L1 Count 7
Analizing Cell [6,11,3]
L1 Count 2
L2 Count 73
Distance to 299 MARK HOWE          45,5741
Distance to 308 DAVE REID          46,6798
Distance to 314 SHAWN BURR        46,2817
Distance to 323 UWE KRUPP         47,5920

```

Distance to 327 DAVE HANNAN 55,1634  
 Distance to 367 A. KARPOVTSEV 58,5577  
 Distance to 297 PAUL BROTEN 40,8044  
 Distance to 355 TOMMY ALBELIN 44,6990  
 Distance to 325 TRENT YAWNEY 53,5257  
 Distance to 334 BOBBY DOLLAS 51,6333  
 Distance to 326 DON SWEENEY 44,6878  
 Distance to 310 BRAD SHAW 96,0937  
 Distance to 344 ANDREW MCBAIN 99,3428  
 Distance to 366 KERRY HUFFMAN 81,7374  
 Distance to 274 DEAN MCAMMOND 43,6119  
 Distance to 244 BENOIT BRUNET 39,4081  
 Distance to 279 STEVE KONOWALCHUK 45,8803  
 Distance to 214 GREG HAWGOOD 41,7133  
 Distance to 217 DIRK GRAHAM 42,3084  
 Distance to 230 JOHN DRUCE 43,9318  
 Distance to 243 BRAD DALGARNO 53,0377  
 Distance to 226 BRUCE DRIVER 46,2709  
 Distance to 188 TRENT KLATT 35,2136  
 Distance to 161 SHAWN MCEACHERN 33,1813  
 Distance to 167 TOM CHORSKE 33,1964  
 Distance to 172 JOHN TUCKER 37,6298  
 Distance to 181 JEFF NORTON 32,9393  
 Distance to 165 PATRICK FLATLEY 36,7831  
 Distance to 189 GUY CARBONNEAU 39,2046  
 Distance to 171 TRAVIS GREEN 36,1801  
 Distance to 187 STEPHANE MATTEAU 45,4863  
 Distance to 190 WES WALZ 30,8707  
 Distance to 192 BRENT SUTTER 35,9026  
 Distance to 132 SERGEI NEMCHINOV 32,3883  
 Distance to 143 GERMAN TITOV 25,5734  
 Distance to 153 JOHN LECLAIR 29,6816

Number of players within 29,6985 distance: 3  
 Analizing Cell [13,11,3]

L1 Count 1

L2 Count 16

Distance to 028 JOE JUNEAU 51,4101  
 Distance to 034 STEVE YZERMAN 53,0754  
 Distance to 040 BRIAN LEETCH 56,3028  
 Distance to 014 PIERRE TURGEON 45,6946  
 Distance to 015 RAY SHEPPARD 44,9222  
 Distance to 022 SERGEI ZUBOV 42,0714  
 Distance to 017 ROBERT REICHEL 45,7056  
 Distance to 021 RAY BOURQUE 43,6005  
 Distance to 009 JAROMIR JAGR 47,5289  
 Distance to 003 ADAM OATES 40,3609  
 Distance to 418 CRAIG MUNI 112,0223  
 Distance to 490 FRANK MUSIL 112,5922  
 Distance to 388 WAYNE MCBEAN 132,5745  
 Distance to 401 DAVE TIPPETT 125,1599  
 Distance to 404 BRIAN GLYNN 125,9960

Number of players within 29,6985 distance: 0  
 Analizing Cell [5,0,4]

L1 Count 3

```

Analizing Cell [1,2,4]
L1 Count 5
Analizing Cell [2,2,4]
L1 Count 5
Analizing Cell [1,3,4]
L1 Count 16
Analizing Cell [4,3,4]
L1 Count 4
Analizing Cell [6,3,4]
L1 Count 3
Analizing Cell [7,4,4]
L1 Count 5
Analizing Cell [11,5,4]
L1 Count 2
L2 Count 47
Distance to 071 GEOFF SANDERSON      31,8748
Distance to 073 JEFF BROWN            36,7287
Distance to 078 BRIAN BRADLEY         37,9210
Distance to 069 CLIFF RONNING         30,9354
Distance to 080 TREVOR LINDEN         51,0098
Distance to 077 SANDIS OZOLINSH      40,5709
Distance to 074 VALERI KAMENSKY      35,5668
Distance to 076 MARTIN STRAKA         44,8776
Distance to 075 MIKE GARTNER          46,8402
Distance to 057 BOB KUDELSKI          42,7200
Distance to 042 JARI KURRI            30,6757
Distance to 050 DOUG WEIGHT           31,1609
Distance to 056 ALEXEI ZHAMNOV       39,2556
Distance to 066 ULF DAHLEN            39,6485
Distance to 065 BENOIT HOGUE         45,0000
Distance to 062 MIKE RIDLEY           35,3412
Distance to 055 BRIAN BELLOWS        28,7054
Distance to 058 JOE MULLEN            29,6985
Number of players within 29,6985 distance: 3
Analizing Cell [7,6,4]
L1 Count 13
Analizing Cell [9,6,4]
L1 Count 7
Analizing Cell [7,7,4]
L1 Count 15
Analizing Cell [8,7,4]
L1 Count 9
Analizing Cell [9,7,4]
L1 Count 9
Analizing Cell [6,8,4]
L1 Count 9
Analizing Cell [10,8,4]
L1 Count 4
Analizing Cell [5,10,4]
L1 Count 2
L2 Count 175
Distance to 395 JIM DOWD                58,3181
Distance to 398 IGOR CHIBIREV          57,3585
Distance to 462 DAVID EMMA              63,6553

```

|                                    |         |
|------------------------------------|---------|
| Distance to 397 GARY LEEMAN        | 49,2646 |
| Distance to 412 DAVE VOLEK         | 56,6039 |
| Distance to 421 JASON DAWE         | 55,4797 |
| Distance to 485 DOUG CROSSMAN      | 59,0085 |
| Distance to 378 PETER ZEZEL        | 47,6445 |
| Distance to 443 TOMMY SJODIN       | 49,5580 |
| Distance to 460 BRAD WERENKA       | 50,8527 |
| Distance to 381 JOHN SLANEY        | 45,6727 |
| Distance to 384 JIM STORM          | 45,0000 |
| Distance to 455 GREG BROWN         | 50,0999 |
| Distance to 369 JAMIE BAKER        | 43,3705 |
| Distance to 403 MIKE MCKEE         | 42,4617 |
| Distance to 411 ED RONAN           | 44,5533 |
| Distance to 439 DON MCSWEEN        | 45,4423 |
| Distance to 478 PIERRE SEVIGNY     | 46,4004 |
| Distance to 379 MARTIN LAPOINTE    | 42,4028 |
| Distance to 444 CHRIS DAHLQUIST    | 45,7930 |
| Distance to 477 MIKE FOLIGNO       | 47,1063 |
| Distance to 382 KENT MANDERVILLE   | 46,7119 |
| Distance to 458 GERALD DIDUCK      | 56,1160 |
| Distance to 475 RANDY LADOUCEUR    | 55,2178 |
| Distance to 374 JAROSLAV MODRY     | 44,6430 |
| Distance to 488 JEFF FINLEY        | 44,9110 |
| Distance to 405 VIACHESLAV FETISOV | 38,7943 |
| Distance to 415 PETER POPOVIC      | 43,0813 |
| Distance to 425 KRIS DRAPER        | 41,7013 |
| Distance to 435 KEITH CARNEY       | 39,5601 |
| Distance to 436 KEITH BROWN        | 44,8219 |
| Distance to 417 J.J. DAIGNEAULT    | 48,0521 |
| Distance to 429 DENNIS VASKE       | 49,1325 |
| Distance to 418 CRAIG MUNI         | 40,1123 |
| Distance to 490 FRANK MUSIL        | 38,0657 |
| Distance to 336 PETR NEDVED        | 53,4416 |
| Distance to 306 JOZEF STUMPPEL     | 47,0425 |
| Distance to 347 ROBERT LANG        | 49,8197 |
| Distance to 287 MIKAEL ANDERSSON   | 38,4448 |
| Distance to 356 BRENT GRIEVE       | 46,0977 |
| Distance to 293 TOM PEDERSON       | 39,0256 |
| Distance to 359 MIKE EASTWOOD      | 44,7661 |
| Distance to 291 PHILIPPE BOZON     | 36,6197 |
| Distance to 294 ALEXEI GUSAROV     | 37,6563 |
| Distance to 339 DAVID WILLIAMS     | 36,7696 |
| Distance to 302 V. BUTSAYEV        | 48,5901 |
| Distance to 312 PATRICE BRISEBOIS  | 42,5558 |
| Distance to 351 KEVIN LOWE         | 49,1223 |
| Distance to 365 CHARLIE HUDDY      | 50,2494 |
| Distance to 299 MARK HOWE          | 44,3170 |
| Distance to 308 DAVE REID          | 37,3363 |
| Distance to 314 SHAWN BURR         | 34,3657 |
| Distance to 323 UWE KRUPP          | 36,0278 |
| Distance to 327 DAVE HANNAN        | 36,3593 |
| Distance to 367 A. KARPOVTSEV      | 39,0384 |
| Distance to 318 ALEXANDER GODYNYUK | 45,8912 |
| Distance to 297 PAUL BROTEN        | 29,7321 |

Distance to 355 TOMMY ALBELIN 31,0000  
 Distance to 325 TRENT YAWNEY 33,4365  
 Distance to 334 BOBBY DOLLAS 32,2645  
 Distance to 326 DON SWEENEY 26,7208  
 Distance to 310 BRAD SHAW 80,2683  
 Distance to 344 ANDREW MCBAIN 82,6922  
 Distance to 270 OLEG PETROV 51,8652  
 Distance to 278 JODY HULL 48,3735  
 Distance to 231 STEVE DUCHESNE 45,8039  
 Distance to 256 MIKE SILLINGER 48,3425  
 Distance to 251 ALEXANDER SEMAK 38,3797  
 Distance to 229 BRENT GILCHRIST 38,7556  
 Distance to 236 MICHEL GOULET 40,0125  
 Distance to 241 GAETAN DUCHESNE 33,5857  
 Distance to 253 PAT PEAKE 37,2424  
 Distance to 267 ERIC WEINRICH 38,2361  
 Distance to 221 BOB CARPENTER 31,2890  
 Distance to 252 BRIAN PROPP 35,4119  
 Distance to 276 TODD GILL 32,2645  
 Distance to 225 MARTIN RUCINSKY 36,7696  
 Distance to 227 DAVE BABYCH 38,1051  
 Distance to 222 SHAUN VAN ALLEN 42,5323  
 Distance to 257 MIKE HOUGH 40,6202  
 Distance to 259 PAT CONACHER 48,1768  
 Distance to 264 CLAUDE LAPOINTE 46,1736  
 Distance to 274 DEAN MCAMMOND 39,0000  
 Distance to 244 BENOIT BRUNET 33,7639  
 Distance to 279 STEVE KONOWALCHUK 33,2566  
 Distance to 214 GREG HAWGOOD 28,3725

Number of players within 29,6985 distance: 3  
 Analizing Cell [5,1,5]

L1 Count 2

L2 Count 146

Distance to 388 WAYNE MCBEAN 28,7924  
 Distance to 401 DAVE TIPPETT 35,8329  
 Distance to 404 BRIAN GLYNN 35,3553  
 Distance to 392 DIXON WARD 47,5815  
 Distance to 449 TIM BERGLAND 55,6147  
 Distance to 482 DALE CRAIGWELL 53,9351  
 Distance to 389 JEFF SHANTZ 41,5812  
 Distance to 385 IGOR KOROLEV 39,7115  
 Distance to 399 BRYAN TROTTIER 41,3763  
 Distance to 407 RANDY GILHEN 40,3609  
 Distance to 491 GEOFF SMITH 43,4626  
 Distance to 472 TIM WATTERS 47,8644  
 Distance to 396 STEPHEN LEACH 48,5798  
 Distance to 467 RICK ZOMBO 55,5248  
 Distance to 471 DAVID SHAW 57,4369  
 Distance to 409 CHRIS LINDBERG 60,4814  
 Distance to 420 ANDRE FAUST 62,1691  
 Distance to 441 RON WILSON 60,8358  
 Distance to 476 JARROD SKALDE 63,0000  
 Distance to 479 MIKE SULLIVAN 64,1950  
 Distance to 492 SHAWN ANDERSON 63,0317

Distance to 371 GREG JOHNSON 49,4065  
 Distance to 461 DAN MAROIS 57,3149  
 Distance to 393 MARC BUREAU 45,4863  
 Distance to 402 GREG GILBERT 50,1398  
 Distance to 410 PHILIPPE BOUCHER 52,1824  
 Distance to 423 SHELDON KENNEDY 51,6527  
 Distance to 450 MIKE KRUSHELNYSKI 51,4684  
 Distance to 454 MARKUS NASLUND 53,2353  
 Distance to 422 MARK GREIG 46,6262  
 Distance to 431 ROB ZAMUNER 44,2493  
 Distance to 447 ROBERT PETROVICKY 51,2445  
 Distance to 483 JOHN MCINTYRE 47,0850  
 Distance to 406 JOSE CHARBONNEAU 47,0213  
 Distance to 446 STEVEN KING 46,1411  
 Distance to 451 MIKE HUDSON 47,4342  
 Distance to 480 ADAM BENNETT 46,7012  
 Distance to 465 PAUL STANTON 47,0532  
 Distance to 469 MIKE RATHJE 46,5940  
 Distance to 445 BRET HEDICAN 48,0208  
 Distance to 466 DRAKE BEREHOWSKY 51,7977  
 Distance to 481 CAMERON STEWART 51,3420  
 Distance to 414 SCOTT LACHANCE 50,4975  
 Distance to 427 GERARD GALLANT 52,2685  
 Distance to 486 KEITH ACTON 53,9444  
 Distance to 310 BRAD SHAW 22,8473

Number of players within 29,6985 distance: 3

Analizing Cell [8,3,5]

L1 Count 4

Analizing Cell [6,4,5]

L1 Count 5

Analizing Cell [12,5,5]

L1 Count 2

L2 Count 40

Distance to 049 NELSON EMERSON 59,5063  
 Distance to 057 BOB KUDELSKI 53,0471  
 Distance to 042 JARI KURRI 37,2559  
 Distance to 050 DOUG WEIGHT 38,6005  
 Distance to 056 ALEXEI ZHAMNOV 43,3128  
 Distance to 066 ULF DAHLEN 52,3546  
 Distance to 065 BENOIT HOGUE 46,8828  
 Distance to 062 MIKE RIDLEY 46,2817  
 Distance to 055 BRIAN BELLOWS 38,9487  
 Distance to 058 JOE MULLEN 38,9230  
 Distance to 053 LARRY MURPHY 36,1109  
 Distance to 048 CAM NEELY 36,7287  
 Distance to 047 JOE NIEUWENDYK 38,6005  
 Distance to 061 DEREK KING 44,0227  
 Distance to 054 STEPHANE RICHER 56,6922  
 Distance to 051 VYACHESLAV KOZLOV 44,8665  
 Distance to 031 CRAIG JANNEY 31,8434  
 Distance to 038 ALEXANDER MOGILNY 38,2099  
 Distance to 032 \*MIKAEL RENBERG 28,7228  
 Distance to 027 MATS SUNDIN 26,2488

Number of players within 29,6985 distance: 3

```

Analizing Cell [8,7,5]
L1 Count 13
Analizing Cell [13,7,5]
L1 Count 1
L2 Count 100
Distance to 031 CRAIG JANNEY           36,6879
Distance to 038 ALEXANDER MOGILNY      40,2741
Distance to 032 *MIKAEL RENBERG        31,3847
Distance to 027 MATS SUNDIN            32,1714
Distance to 036 RUSS COURTNALL         35,1568
Distance to 028 JOE JUNEAU              27,3130
Distance to 034 STEVE YZERMAN           31,3369
Distance to 030 MARK MESSIER            44,3847
Distance to 040 BRIAN LEETCH            43,5546
Distance to 019 JOE SAKIC               38,1182
Distance to 016 MIKE MODANO             27,6767
Distance to 018 RON FRANCIS             28,6182
Number of players within 29,6985 distance: 3
Analizing Cell [4,8,5]
L1 Count 12
Analizing Cell [8,8,5]
L1 Count 10
Analizing Cell [2,9,5]
L1 Count 5
Analizing Cell [8,9,5]
L1 Count 3
Analizing Cell [1,10,5]
L1 Count 2
L2 Count 173
Distance to 505 STEVE DUBINSKY          50,2593
Distance to 515 REGGIE SAVAGE           48,8365
Distance to 532 DANE JACKSON            53,9907
Distance to 539 TROY MURRAY             53,8981
Distance to 549 FRED KNIPSCHEER        50,3686
Distance to 571 CHRIS GOVEDARIS        49,7695
Distance to 593 GRANT FUHR              51,2348
Distance to 641 PAT NEATON              52,1344
Distance to 661 JON CASEY               52,8110
Distance to 664 ARTURS IRBE             51,4684
Distance to 690 DENIS CHASSE            51,5558
Distance to 691 DARREN BANKS            56,4712
Distance to 706 IVAN DROPPA             52,9528
Distance to 712 LEE NORWOOD             49,4469
Distance to 725 DON BEAUPRE             51,6140
Distance to 726 ANDY MOOG               51,6140
Distance to 801 FRANK BIALOWAS          54,4885
Distance to 843 CHRIS TAMER             54,6535
Distance to 865 SEAN BURKE              51,7784
Distance to 867 MIKE VERNON             53,1131
Distance to 580 NEIL EISENHUT           48,0625
Distance to 606 CHRISTIAN PROULX        48,7852
Distance to 623 RICHARD MATVICHUK       46,7868
Distance to 680 BEN HANKINSON           47,2969
Distance to 822 KEVIN MCCLELLAND        49,8598

```

```

Distance to 854 ROBB STAUBER          50,4876
Distance to 494 JIM SANDLAK           36,7287
Distance to 510 TOM TILLEY            39,3700
Distance to 572 DONALD BRASHEAR       41,5331
Distance to 659 DARCY WAKALUK         41,8210
Distance to 730 PATRICK ROY            43,6807
Distance to 796 JASON SIMON           42,2019
Distance to 643 KEN MCRAE             40,1746
Distance to 723 TOM BARRASSO          39,6485
Distance to 869 JOHN VANBIESBROUCK   40,8534
Distance to 566 JASON SMITH           32,0312
Distance to 630 RON HEXTALL           38,5227
Distance to 824 TONY HORACEK          38,1969
Distance to 506 ADAM FOOTE            38,9230
Distance to 598 ED BELFOUR            39,8748
Distance to 591 CHRIS LIPUMA          46,0760
Distance to 656 PHILIP CROWE          43,9773
Distance to 658 SHAWN CRONIN          44,9555
Distance to 397 GARY LEEMAN           47,0532
Distance to 412 DAVE VOLEK            55,3986
Distance to 421 JASON DAWE            53,1883
Distance to 485 DOUG CROSSMAN         54,4885
Distance to 378 PETER ZEZEL           45,8148
Distance to 443 TOMMY SJODIN          44,1475
Distance to 460 BRAD WERENKA         44,8665
Distance to 381 JOHN SLANEY           42,4617
Distance to 384 JIM STORM              41,6413
Distance to 455 GREG BROWN            43,0929
Distance to 369 JAMIE BAKER           38,7814
Distance to 403 MIKE MCKEE            34,7275
Distance to 411 ED RONAN              36,2491
Distance to 439 DON MCSWEEN           35,8608
Distance to 478 PIERRE SEVIGNY        32,9848
Distance to 379 MARTIN LAPOINTE       32,1714
Distance to 444 CHRIS DAHLQUIST       33,1964
Distance to 477 MIKE FOLIGNO          31,1448
Distance to 382 KENT MANDERVILLE       36,1525
Distance to 458 GERALD DIDUCK         42,2374
Distance to 475 RANDY LADOUCEUR       39,2173
Distance to 374 JAROSLAV MODRY        43,2666
Distance to 488 JEFF FINLEY           34,0588
Distance to 405 VIACHESLAV FETISOV    31,8119
Distance to 415 PETER POPOVIC         37,2156
Distance to 425 KRIS DRAPER           33,2566
Distance to 435 KEITH CARNEY          26,5707
Distance to 436 KEITH BROWN           28,9482
Number of players within 29,6985 distance: 3
Analizing Cell [2,1,6]
L1 Count 2
L2 Count 175
Distance to 614 HANK LAMMENS           47,0106
Distance to 561 DIMITRI FILIMONOV     54,4610
Distance to 586 DAVE TOMLINSON        49,2646
Distance to 501 MARK FERNER           41,1218

```



|                                  |         |
|----------------------------------|---------|
| Distance to 657 DOUG LIDSTER     | 44,2493 |
| Distance to 527 GORDIE ROBERTS   | 37,4299 |
| Distance to 627 DARCY LOEWEN     | 36,7287 |
| Distance to 857 HERB RAGLAN      | 36,9053 |
| Distance to 514 ALLAN CONROY     | 33,6601 |
| Distance to 529 SERGEI BAUTIN    | 38,1838 |
| Distance to 503 JEFF DANIELS     | 57,8446 |
| Distance to 516 NATHAN LAFAYETTE | 55,6147 |
| Distance to 520 DAVE BARR        | 54,0833 |
| Distance to 537 DAN KESA         | 58,4294 |
| Distance to 544 JIM PAEK         | 55,3082 |
| Distance to 577 MIKE RAMSEY      | 55,6687 |
| Distance to 585 VLASTIMIL KROUPA | 55,7405 |
| Distance to 590 DARRYL SHANNON   | 57,1577 |
| Distance to 624 KEVIN DAHL       | 56,7186 |
| Distance to 679 KEITH REDMOND    | 58,7282 |
| Distance to 698 NEIL BRADY       | 59,3970 |
| Distance to 702 ROB MURPHY       | 57,4195 |
| Distance to 805 MARC LAFORGE     | 59,1101 |
| Distance to 808 GERRY FLEMING    | 55,2630 |
| Distance to 827 DAVE THOMLINSON  | 56,5509 |
| Distance to 500 ED OLCZYK        | 52,7826 |
| Distance to 513 DAVE TAYLOR      | 53,0754 |
| Distance to 517 GARY SHUCHUK     | 46,7547 |
| Distance to 622 MARK HARDY       | 52,1536 |
| Distance to 652 IAN HERBERS      | 48,9387 |
| Distance to 856 JOE DAY          | 50,2593 |
| Distance to 496 MICAH AIVAZOFF   | 47,6025 |
| Distance to 555 PAUL MACDERMID   | 48,4149 |
| Distance to 625 BOB MCGILL       | 41,5331 |
| Distance to 645 CLAUDE LOISELLE  | 45,4093 |
| Distance to 684 TODD HARKINS     | 44,1928 |
| Distance to 707 GORDON MARK      | 47,5500 |
| Distance to 507 DONALD DUFRESNE  | 37,1753 |
| Distance to 554 DAVE MACKKEY     | 41,2553 |
| Distance to 615 MARIO ROBERGE    | 44,0114 |
| Distance to 526 JAY MORE         | 39,5980 |
| Distance to 530 JIM JOHNSON      | 37,7889 |
| Distance to 683 BRIAN CURRAN     | 44,8219 |
| Distance to 545 BRAD MCCRIMMON   | 40,1746 |
| Distance to 560 RICHARD PILON    | 44,1475 |
| Distance to 646 MIKE HARTMAN     | 43,1045 |
| Distance to 716 FRANCOIS LEROUX  | 44,4297 |
| Distance to 613 JIM CUMMINS      | 51,2348 |
| Distance to 681 BRENT THOMPSON   | 50,6754 |
| Distance to 559 BOB HALKIDIS     | 55,4977 |
| Distance to 388 WAYNE MCBEAN     | 21,4243 |
| Distance to 401 DAVE TIPPETT     | 30,7571 |
| Distance to 404 BRIAN GLYNN      | 24,1661 |

Number of players within 29,6985 distance: 3

Analizing Cell [1,3,6]

L1 Count 7

Analizing Cell [3,3,6]

L1 Count 4

```

Analizing Cell [7,4,6]
L1 Count 6
Analizing Cell [7,5,6]
L1 Count 7
Analizing Cell [10,5,6]
L1 Count 4
Analizing Cell [6,6,6]
L1 Count 9
Analizing Cell [9,6,6]
L1 Count 6
Analizing Cell [8,7,6]
L1 Count 7
Analizing Cell [2,8,6]
L1 Count 11
Analizing Cell [5,8,6]
L1 Count 9
Analizing Cell [8,8,6]
L1 Count 7
Analizing Cell [10,8,6]
L1 Count 4
Analizing Cell [10,9,6]
L1 Count 3
Analizing Cell [2,1,7]
L1 Count 2
L2 Count 154
Distance to 501 MARK FERNER           43,6119
Distance to 657 DOUG LIDSTER          45,7275
Distance to 527 GORDIE ROBERTS        38,7814
Distance to 627 DARCY LOEWEN          36,0555
Distance to 857 HERB RAGLAN            35,9026
Distance to 514 ALLAN CONROY           31,4006
Distance to 529 SERGEI BAUTIN          34,3657
Distance to 602 CLAUDE BOIVIN         42,3438
Distance to 500 ED OLCZYK              54,9272
Distance to 513 DAVE TAYLOR            55,1362
Distance to 517 GARY SHUCHUK           48,8774
Distance to 622 MARK HARDY             54,0463
Distance to 652 IAN HERBERS            50,3786
Distance to 856 JOE DAY                 51,7011
Distance to 496 MICAH AIVAZOFF         48,9592
Distance to 555 PAUL MACDERMID         49,8096
Distance to 625 BOB MCGILL             41,6653
Distance to 645 CLAUDE LOISELLE        45,1110
Distance to 684 TODD HARKINS           43,7950
Distance to 707 GORDON MARK            47,8121
Distance to 507 DONALD DUFRESNE        36,2353
Distance to 554 DAVE MACKKEY           40,3609
Distance to 615 MARIO ROBERGE          43,1045
Distance to 526 JAY MORE                37,9605
Distance to 530 JIM JOHNSON            36,0694
Distance to 683 BRIAN CURRAN           43,0581
Distance to 545 BRAD MCCRIMMON         37,2693
Distance to 560 RICHARD PILON          41,0609
Distance to 646 MIKE HARTMAN           40,2616

```

```

Distance to 716 FRANCOIS LEROUX      41,5812
Distance to 613 JIM CUMMINS           47,4974
Distance to 681 BRENT THOMPSON        47,0425
Distance to 559 BOB HALKIDIS          51,3517
Distance to 595 TONY TWIST            56,5155
Distance to 718 GORD DONNELLY         53,4509
Distance to 388 WAYNE MCBEAN          25,1794
Distance to 401 DAVE TIPPETT          33,6601
Distance to 404 BRIAN GLYNN           24,9199
Number of players within 29,6985 distance: 3
Analizing Cell [3,3,7]
L1 Count 4
Analizing Cell [8,3,7]
L1 Count 2
L2 Count 116
Distance to 201 MARK LAMB              43,6005
Distance to 182 BILL HOULDER           38,8844
Distance to 205 ROB GAUDREAU           50,5173
Distance to 195 JOE SACCO              35,1852
Distance to 186 BRENT FEDYK            35,6230
Distance to 196 MARIO LEMIEUX         48,7852
Distance to 207 PAUL YSEBAERT         52,5452
Distance to 209 FREDRIK OLAUSSON      49,8899
Distance to 210 JAMES PATRICK          44,7772
Distance to 178 DALLAS DRAKE           38,6135
Distance to 198 DAVE LOWRY             37,6298
Distance to 168 GLENN ANDERSON         33,5410
Distance to 159 RANDY BURRIDGE         36,3731
Distance to 211 DARRYL SYDOR           49,4065
Distance to 208 ERIC DESJARDINS       53,6843
Distance to 183 KELLY MILLER           52,0384
Distance to 160 MIKE DONNELLY         45,9239
Distance to 203 MIKE MCPHEE           52,4976
Distance to 163 TONY AMONTE            43,1972
Distance to 180 TOM KURVERS            43,7607
Distance to 166 CALLE JOHANSSON       37,1349
Distance to 184 DOUG BODGER            44,7661
Distance to 169 BRIAN NOONAN           40,9634
Distance to 202 DMITRI MIRONOV        45,8912
Distance to 193 JIRI SLEGR             45,4423
Distance to 174 RON SUTTER             47,2123
Distance to 188 TRENT KLATT            56,5862
Distance to 161 SHAWN MCEACHERN       52,7352
Distance to 167 TOM CHORSKE            54,3691
Distance to 172 JOHN TUCKER            54,3875
Distance to 181 JEFF NORTON            54,1572
Distance to 165 PATRICK FLATLEY        48,4665
Distance to 189 GUY CARBONNEAU        50,8035
Distance to 171 TRAVIS GREEN           50,8035
Distance to 187 STEPHANE MATTEAU      48,3632
Distance to 185 RANDY WOOD             46,1628
Distance to 123 ALEXANDRE DAIGLE       38,8458
Distance to 124 THOMAS STEEN           40,2989
Distance to 154 DAVE MCLLWAIN          37,1484

```

```

Distance to 157 BOB BEERS                36,9324
Distance to 150 DEAN EVASON              28,4429
Distance to 156 GORD MURPHY              30,7571
Distance to 144 GILBERT DIONNE           41,9285
Distance to 136 ZARLEY ZALAPSKI          30,2655
Distance to 131 JOSEF BERANEK            36,5650
Distance to 135 ANDREI LOMAKIN           48,0937
Distance to 152 DANTON COLE              49,1223
Distance to 129 MICHAL PIVONKA           38,7427
Distance to 134 STU BARNES               41,5692
Distance to 142 GRANT LEDYARD            41,8808
Distance to 158 DAVE ELLETT              43,1277
Distance to 155 TIM SWEENEY              38,4968
Distance to 119 BRYAN SMOLINSKI          37,0945
Distance to 133 KEVIN MILLER             40,5709
Distance to 145 GARRY VALK               53,8888
Distance to 132 SERGEI NEMCHINOV        47,4236
Distance to 149 PAUL CAVALLINI           47,0956
Distance to 148 CLAUDE LEMIEUX          48,9285
Distance to 147 BILL GUERIN              58,3609
Distance to 089 ANDREW CASSELS           28,1957
Number of players within 29,6985 distance: 3
Analizing Cell [4,4,7]
L1 Count 14
Analizing Cell [5,4,7]
L1 Count 10
Analizing Cell [4,5,7]
L1 Count 21
Analizing Cell [10,5,7]
L1 Count 5
Analizing Cell [6,6,7]
L1 Count 9
Analizing Cell [3,7,7]
L1 Count 22
Analizing Cell [6,7,7]
L1 Count 8
Analizing Cell [11,7,7]
L1 Count 3
Analizing Cell [7,8,7]
L1 Count 6
Analizing Cell [1,9,7]
L1 Count 4
Analizing Cell [3,9,7]
L1 Count 5
Analizing Cell [5,9,7]
L1 Count 4
Analizing Cell [9,9,7]
L1 Count 5
Analizing Cell [4,0,8]
L1 Count 1
L2 Count 57
Distance to 527 GORDIE ROBERTS           53,1601
Distance to 627 DARCY LOEWEN             50,7740
Distance to 857 HERB RAGLAN              51,5461

```

```

Distance to 514 ALLAN CONROY           43,7493
Distance to 529 SERGEI BAUTIN          43,6005
Distance to 602 CLAUDE BOIVIN          48,4458
Distance to 388 WAYNE MCBEAN           39,0384
Distance to 401 DAVE TIPPETT            46,6154
Distance to 404 BRIAN GLYNN            37,7757
Distance to 392 DIXON WARD              32,0156
Distance to 437 MIKE EAGLES             41,5692
Distance to 385 IGOR KOROLEV            49,4773
Distance to 399 BRYAN TROTTIER          52,5452
Distance to 407 RANDY GILHEN            44,6318
Distance to 491 GEOFF SMITH             46,6262
Distance to 472 TIM WATTERS             42,3084
Distance to 396 STEPHEN LEACH           40,0125
Distance to 467 RICK ZOMBO              41,0609
Distance to 471 DAVID SHAW              43,9773
Distance to 452 SCOTT THORNTON          49,8999
Distance to 332 GORD DINEEN             24,0416
Distance to 310 BRAD SHAW               18,6279
Distance to 344 ANDREW MCBAIN           19,0263
Number of players within 29,6985 distance: 3
Analizing Cell [4,4,8]
L1 Count 14
Analizing Cell [5,4,8]
L1 Count 12
Analizing Cell [6,4,8]
L1 Count 7
Analizing Cell [4,5,8]
L1 Count 22
Analizing Cell [5,5,8]
L1 Count 19
Analizing Cell [6,5,8]
L1 Count 12
Analizing Cell [8,5,8]
L1 Count 2
L2 Count 122
Distance to 182 BILL HOULDER            45,9347
Distance to 195 JOE SACCO                34,4093
Distance to 186 BRENT FEDYK              31,7962
Distance to 173 KEVIN HATCHER            45,8476
Distance to 210 JAMES PATRICK            47,4236
Distance to 178 DALLAS DRAKE             38,1182
Distance to 198 DAVE LOWRY               33,3766
Distance to 168 GLENN ANDERSON           30,1496
Distance to 159 RANDY BURRIDGE           27,6586
Distance to 211 DARRYL SYDOR             40,0125
Distance to 208 ERIC DESJARDINS         42,0476
Distance to 203 MIKE MCPHEE              52,4404
Distance to 163 TONY AMONTE              45,5412
Distance to 180 TOM KURVERS              41,3884
Distance to 166 CALLE JOHANSSON         32,0156
Distance to 184 DOUG BODGER              33,6749
Distance to 169 BRIAN NOONAN            31,5595
Distance to 202 DMITRI MIRONOV          35,4683

```

```

Distance to 193 JIRI SLEGR           34,3366
Distance to 174 RON SUTTER          34,7994
Distance to 181 JEFF NORTON         52,3355
Distance to 165 PATRICK FLATLEY     46,6798
Distance to 189 GUY CARBONNEAU     45,9891
Distance to 171 TRAVIS GREEN        47,0213
Distance to 187 STEPHANE MATTEAU    41,2432
Distance to 185 RANDY WOOD          35,9026
Distance to 192 BRENT SUTTER        49,3660
Distance to 170 RICHARD SMEHLIK     40,5093
Distance to 154 DAVE MCLLWAIN       48,8876
Distance to 157 BOB BEERS           32,7109
Distance to 150 DEAN EVASON         26,4386
Number of players within 29,6985 distance: 3
Analizing Cell [4,6,8]
L1 Count 20
Analizing Cell [7,6,8]
L1 Count 11
Analizing Cell [10,6,8]
L1 Count 4
Analizing Cell [2,7,8]
L1 Count 16
Analizing Cell [4,7,8]
L1 Count 13
Analizing Cell [6,7,8]
L1 Count 12
Analizing Cell [1,8,8]
L1 Count 6
Analizing Cell [4,8,8]
L1 Count 7
Analizing Cell [9,8,8]
L1 Count 2
L2 Count 89
Distance to 150 DEAN EVASON         55,3986
Distance to 156 GORD MURPHY         54,7905
Distance to 136 ZARLEY ZALAPSKI     48,3115
Distance to 131 JOSEF BERANEK       45,1110
Distance to 138 DENIS SAVARD        54,9545
Distance to 129 MICHAL PIVONKA      55,9375
Distance to 134 STU BARNES          57,0438
Distance to 142 GRANT LEDYARD       54,0740
Distance to 158 DAVE ELLETT         56,5509
Distance to 155 TIM SWEENEY         53,7959
Distance to 119 BRYAN SMOLINSKI     39,5727
Distance to 133 KEVIN MILLER        41,3038
Distance to 145 GARRY VALK          48,8467
Distance to 132 SERGEI NEMCHINOV   54,4885
Distance to 149 PAUL CAVALLINI      42,1900
Distance to 148 CLAUDE LEMIEUX     42,9418
Distance to 147 BILL GUERIN         48,4355
Distance to 151 PETER BONDRA        54,6443
Distance to 127 COREY MILLEN        41,6293
Distance to 137 BERNIE NICHOLLS     39,3065
Distance to 125 SYLVAIN COTE        34,8425

```

|                                   |         |
|-----------------------------------|---------|
| Distance to 141 SCOTT NIEDERMAYER | 51,7784 |
| Distance to 118 YVES RACINE       | 55,7136 |
| Distance to 102 PETR KLIMA        | 49,4065 |
| Distance to 117 ALEXEI ZHITNIK    | 54,2679 |
| Distance to 085 DIMITRI KHRISTICH | 37,6032 |
| Distance to 087 TOMAS SANDSTROM   | 41,8210 |
| Distance to 094 KIRK MULLER       | 42,4853 |
| Distance to 084 DONALD AUDETTE    | 48,7750 |
| Distance to 106 JYRKI LUMME       | 44,7325 |
| Distance to 114 TERRY YAKE        | 50,7642 |
| Distance to 110 TED DONATO        | 42,5911 |
| Distance to 091 GLEN WESLEY       | 37,3631 |
| Distance to 112 RAY FERRARO       | 39,8246 |
| Distance to 082 STEVE LARMER      | 43,8406 |
| Distance to 086 YURI KHYLEV       | 39,3573 |
| Distance to 115 MATT SCHNEIDER    | 36,3318 |
| Distance to 116 NEAL BROTEN       | 38,0132 |
| Distance to 092 DINO CICCARELLI   | 31,0322 |
| Distance to 100 IGOR LARIONOV     | 45,8803 |
| Distance to 096 VLADIMIR MALAKHOV | 27,5862 |
| Distance to 093 VALERI ZELEPUKIN  | 29,7658 |
| Distance to 071 GEOFF SANDERSON   | 53,7494 |
| Distance to 073 JEFF BROWN        | 44,9444 |
| Distance to 078 BRIAN BRADLEY     | 43,4626 |
| Distance to 072 TODD ELIK         | 38,3275 |
| Distance to 069 CLIFF RONNING     | 41,6653 |
| Distance to 080 TREVOR LINDEN     | 29,9833 |
| Distance to 067 JASON ARNOTT      | 40,1995 |
| Distance to 074 VALERI KAMENSKY   | 41,0609 |
| Distance to 068 SERGEI MAKAROV    | 21,3542 |

Number of players within 29,6985 distance: 3

Analizing Cell [3,9,8]

L1 Count 3

Analizing Cell [6,10,8]

L1 Count 3

Analizing Cell [8,1,9]

L1 Count 1

L2 Count 41

|                                 |         |
|---------------------------------|---------|
| Distance to 201 MARK LAMB       | 39,2938 |
| Distance to 195 JOE SACCO       | 49,5883 |
| Distance to 186 BRENT FEDYK     | 43,6807 |
| Distance to 173 KEVIN HATCHER   | 50,6458 |
| Distance to 178 DALLAS DRAKE    | 59,0424 |
| Distance to 198 DAVE LOWRY      | 52,7352 |
| Distance to 168 GLENN ANDERSON  | 50,0200 |
| Distance to 159 RANDY BURRIDGE  | 49,4166 |
| Distance to 211 DARRYL SYDOR    | 50,5767 |
| Distance to 208 ERIC DESJARDINS | 56,3826 |
| Distance to 162 KEVIN DINEEN    | 54,3507 |
| Distance to 154 DAVE MCLLWAIN   | 44,5982 |
| Distance to 157 BOB BEERS       | 35,3977 |
| Distance to 150 DEAN EVASON     | 42,0951 |
| Distance to 156 GORD MURPHY     | 42,0833 |
| Distance to 136 ZARLEY ZALAPSKI | 42,2966 |

```

Distance to 131 JOSEF BERANEK           44,1135
Distance to 138 DENIS SAVARD           53,1883
Distance to 120 MIKE RICCI             49,5883
Distance to 088 DARRIN SHANNON         26,5518
Distance to 118 YVES RACINE           47,2969
Distance to 102 PETR KLIMA             30,0998
Distance to 117 ALEXEI ZHITNIK         40,6694
Distance to 085 DIMITRI KHRISTICH     40,0125
Distance to 087 TOMAS SANDSTROM       35,0143
Distance to 094 KIRK MULLER           43,7493
Distance to 108 SHAYNE CORSON         52,3832
Distance to 073 JEFF BROWN            37,0000
Distance to 078 BRIAN BRADLEY         39,6989
Distance to 072 TODD ELIK             38,9102
Distance to 042 JARI KURRI            35,0571
Distance to 050 DOUG WEIGHT           36,6742
Distance to 056 ALEXEI ZHAMNOV       25,6320
Distance to 064 GARRY GALLEY         29,4279
Number of players within 29,6985 distance: 3
Analizing Cell [1,3,9]
L1 Count 6
Analizing Cell [5,5,9]
L1 Count 17
Analizing Cell [6,5,9]
L1 Count 10
Analizing Cell [11,5,9]
L1 Count 3
Analizing Cell [4,6,9]
L1 Count 18
Analizing Cell [5,6,9]
L1 Count 18
Analizing Cell [6,6,9]
L1 Count 13
Analizing Cell [5,7,9]
L1 Count 14
Analizing Cell [7,7,9]
L1 Count 5
Analizing Cell [1,9,9]
L1 Count 2
L2 Count 164
Distance to 625 BOB MCGILL             52,4690
Distance to 645 CLAUDE LOISELLE       46,9681
Distance to 684 TODD HARKINS          48,6724
Distance to 707 GORDON MARK           51,1957
Distance to 507 DONALD DUFRESNE       45,2769
Distance to 554 DAVE MACKKEY          43,9773
Distance to 615 MARIO ROBERGE         43,1856
Distance to 526 JAY MORE               41,4246
Distance to 530 JIM JOHNSON           43,2666
Distance to 683 BRIAN CURRAN          40,6079
Distance to 545 BRAD MCCRIMMON        40,9145
Distance to 560 RICHARD PILON         37,6962
Distance to 646 MIKE HARTMAN          40,0749
Distance to 716 FRANCOIS LEROUX      39,3700

```



```

Distance to 613 JIM CUMMINS           35,2846
Distance to 681 BRENT THOMPSON        35,4965
Distance to 559 BOB HALKIDIS          37,2559
Distance to 595 TONY TWIST            41,0974
Distance to 718 GORD DONNELLY         41,4729
Distance to 540 MATTHEW BARNABY       48,5180
Distance to 547 DAVID MALEY           47,4342
Distance to 616 BASIL MCRAE           47,3286
Distance to 587 RYAN MCGILL           50,8134
Distance to 566 JASON SMITH           44,8107
Distance to 630 RON HEXTALL           43,3820
Distance to 824 TONY HORACEK          42,6732
Distance to 506 ADAM FOOTE            32,4500
Distance to 598 ED BELFOUR            38,2361
Distance to 591 CHRIS LIPUMA          32,8177
Distance to 656 PHILIP CROWE          30,4302
Distance to 658 SHAWN CRONIN          32,4191
Distance to 564 KEVIN KAMINSKI        32,5730
Distance to 588 STEWART MALGUNAS      32,5423
Distance to 543 JASON BOWEN           23,0217
Distance to 633 PAUL LAUS             39,4588
Distance to 406 JOSE CHARBONNEAU     46,9574
Distance to 446 STEVEN KING            53,1507
Distance to 451 MIKE HUDSON            49,6790
Distance to 480 ADAM BENNETT          50,8429
Distance to 465 PAUL STANTON          47,1805
Distance to 469 MIKE RATHJE           46,5296
Distance to 445 BRET HEDICAN          42,7317
Distance to 466 DRAKE BEREHOWSKY     39,4081
Distance to 481 CAMERON STEWART       41,1582
Distance to 414 SCOTT LACHANCE        39,1663
Distance to 427 GERARD GALLANT        39,3700
Distance to 486 KEITH ACTON           38,8844
Distance to 391 MARC BERGEVIN         39,1535
Distance to 432 BILL LINDSAY          39,3700
Distance to 433 ALAN MAY              50,9215
Distance to 379 MARTIN LAPOINTE       35,7351
Distance to 444 CHRIS DAHLQUIST       38,9102
Distance to 477 MIKE FOLIGNO          36,8239
Distance to 382 KENT MANDERVILLE      32,6343
Distance to 458 GERALD DIDUCK         31,7805
Distance to 475 RANDY LADOUCEUR       26,4953
Number of players within 29,6985 distance: 3
Analizing Cell [6,9,9]
L1 Count 3
Analizing Cell [2,0,10]
L1 Count 1
L2 Count 39
Distance to 627 DARCY LOEWEN          58,0861
Distance to 857 HERB RAGLAN           57,7148
Distance to 514 ALLAN CONROY          48,7032
Distance to 529 SERGEI BAUTIN         43,7836
Distance to 602 CLAUDE BOIVIN         39,8121
Distance to 541 GRANT JENNINGS        57,9483

```

```

Distance to 394 DARREN RUMBLE          27,7308
Distance to 404 BRIAN GLYNN            49,2037
Distance to 392 DIXON WARD             31,3847
Distance to 437 MIKE EAGLES            33,9706
Distance to 472 TIM WATTERS            47,8121
Distance to 396 STEPHEN LEACH           45,0000
Distance to 467 RICK ZOMBO              38,8072
Distance to 471 DAVID SHAW              42,6380
Distance to 452 SCOTT THORNTON          41,1582
Distance to 413 MURRAY BARON            51,4684
Distance to 310 BRAD SHAW                32,0156
Distance to 344 ANDREW MCBAIN           27,3861
Distance to 305 JOEL OTTO                35,1852
Distance to 235 BORIS MIRONOV           30,0333
Distance to 281 SYLVAIN TURGEON         46,0760
Distance to 262 PAT ELYNUIK             40,4351
Distance to 284 STEPHANE QUINTAL        40,6694
Distance to 273 SEAN HILL                41,9166
Distance to 215 CRAIG MACTAVISH         39,8999
Distance to 263 BOB BASSEN              37,0675
Distance to 238 BOB ERREY               55,9553
Distance to 201 MARK LAMB                24,0416
Number of players within 29,6985 distance: 3
Analizing Cell [5,3,10]
L1 Count 3
Analizing Cell [6,3,10]
L1 Count 3
Analizing Cell [10,3,10]
L1 Count 1
L2 Count 40
Distance to 088 DARRIN SHANNON          28,0891
Distance to 102 PETR KLIMA               32,9545
Distance to 117 ALEXEI ZHITNIK           38,2361
Distance to 085 DIMITRI KHRISTICH        35,7351
Distance to 087 TOMAS SANDSTROM          31,0161
Distance to 094 KIRK MULLER              36,0832
Distance to 108 SHAYNE CORSON            46,8188
Distance to 110 TED DONATO               46,4004
Distance to 091 GLEN WESLEY              41,3401
Distance to 112 RAY FERRARO              39,2301
Distance to 109 ESA TIKKANEN             49,3254
Distance to 115 MATT SCHNEIDER           54,3783
Distance to 116 NEAL BROTEN              51,3030
Distance to 092 DINO CICCARELLI          43,7035
Distance to 073 JEFF BROWN                37,1618
Distance to 078 BRIAN BRADLEY            39,0896
Distance to 072 TODD ELIK                27,7489
Distance to 080 TREVOR LINDEN            38,3406
Distance to 067 JASON ARNOTT             33,0000
Distance to 068 SERGEI MAKAROV           36,7287
Distance to 079 DAVE GAGNER              41,5090
Distance to 049 NELSON EMERSON           22,4499
Number of players within 29,6985 distance: 3
Analizing Cell [2,4,10]

```

```

L1 Count 15
Analizing Cell [3,4,10]
L1 Count 11
Analizing Cell [8,4,10]
L1 Count 2
L2 Count 84
Distance to 201 MARK LAMB           51,0490
Distance to 164 CHRIS GRATTON       44,7661
Distance to 195 JOE SACCO           44,5982
Distance to 186 BRENT FEDYK        36,3593
Distance to 173 KEVIN HATCHER      34,5398
Distance to 198 DAVE LOWRY         43,2088
Distance to 168 GLENN ANDERSON     41,4488
Distance to 159 RANDY BURRIDGE     34,7563
Distance to 211 DARRYL SYDOR       35,1852
Distance to 208 ERIC DESJARDINS    36,8917
Distance to 162 KEVIN DINEEN       35,6651
Distance to 166 CALLE JOHANSSON    44,7661
Distance to 184 DOUG BODGER        39,3319
Distance to 169 BRIAN NOONAN       40,6079
Distance to 202 DMITRI MIRONOV     39,7618
Distance to 193 JIRI SLEGR         34,2053
Distance to 174 RON SUTTER         32,7109
Distance to 187 STEPHANE MATTEAU   53,4416
Distance to 185 RANDY WOOD         43,6807
Distance to 170 RICHARD SMEHLIK    49,1325
Distance to 157 BOB BEERS          29,5804
Distance to 150 DEAN EVASON        36,0832
Distance to 156 GORD MURPHY        33,6006
Distance to 136 ZARLEY ZALAPSKI    29,0345
Number of players within 29,6985 distance: 3
Analizing Cell [0,5,10]
L1 Count 15
Analizing Cell [1,5,10]
L1 Count 20
Analizing Cell [2,5,10]
L1 Count 16
Analizing Cell [3,5,10]
L1 Count 15
Analizing Cell [4,5,10]
L1 Count 14
Analizing Cell [0,6,10]
L1 Count 14
Analizing Cell [2,6,10]
L1 Count 12
Analizing Cell [4,6,10]
L1 Count 14
Analizing Cell [12,6,10]
L1 Count 2
L2 Count 30
Distance to 056 ALEXEI ZHAMNOV     48,0937
Distance to 064 GARRY GALLEY       39,2173
Distance to 065 BENOIT HOGUE       40,9512
Distance to 060 JOE MURPHY         44,6542

```

```

Distance to 048 CAM NEELY           47,2652
Distance to 044 WENDEL CLARK        43,3935
Distance to 063 GEOFF COURTNALL     54,1664
Distance to 061 DEREK KING          48,8569
Distance to 059 JOHN MACLEAN        47,8644
Distance to 043 PAUL COFFEY         45,0444
Distance to 027 MATS SUNDIN         34,0588
Distance to 036 RUSS COURTNALL      38,5097
Distance to 030 MARK MESSIER        34,7131
Distance to 040 BRIAN LEETCH        43,2897
Distance to 037 ADAM GRAVES         56,0446
Distance to 033 AL MACINNIS         43,1509
Distance to 024 LUC ROBITAILLE     29,6985
Distance to 016 MIKE MODANO         36,0694
Distance to 018 RON FRANCIS         28,0713
Number of players within 29,6985 distance: 3
Analizing Cell [0,7,10]
L1 Count 7
Analizing Cell [1,7,10]
L1 Count 8
Analizing Cell [5,7,10]
L1 Count 11
Analizing Cell [10,7,10]
L1 Count 2
L2 Count 54
Distance to 088 DARRIN SHANNON       39,7995
Distance to 102 PETR KLIMA           42,5558
Distance to 117 ALEXEI ZHITNIK      41,1947
Distance to 085 DIMITRI KHRISTICH   35,3836
Distance to 087 TOMAS SANDSTROM     33,7194
Distance to 094 KIRK MULLER         31,4166
Distance to 108 SHAYNE CORSON       45,5741
Distance to 110 TED DONATO          46,3465
Distance to 091 GLEN WESLEY         39,9249
Distance to 112 RAY FERRARO         35,1283
Distance to 109 ESA TIKKANEN        39,7240
Distance to 115 MATT SCHNEIDER      44,9667
Distance to 116 NEAL BROTEN         44,6878
Distance to 092 DINO CICCARELLI     34,1321
Distance to 096 VLADIMIR MALAKHOV   36,3731
Distance to 093 VALERI ZELEPUKIN    44,2493
Distance to 073 JEFF BROWN          45,4533
Distance to 078 BRIAN BRADLEY       45,0888
Distance to 072 TODD ELIK           24,1868
Distance to 080 TREVOR LINDEN       31,0644
Distance to 067 JASON ARNOTT        23,9583
Number of players within 29,6985 distance: 3
Analizing Cell [5,8,10]
L1 Count 5
Analizing Cell [1,3,11]
L1 Count 4
Analizing Cell [3,4,11]
L1 Count 12
Analizing Cell [6,4,11]

```

```

L1 Count 6
Analizing Cell [2,5,11]
L1 Count 18
Analizing Cell [4,5,11]
L1 Count 12
Analizing Cell [6,5,11]
L1 Count 6
Analizing Cell [7,5,11]
L1 Count 6
Analizing Cell [5,6,11]
L1 Count 9
Analizing Cell [5,7,11]
L1 Count 6
Analizing Cell [11,8,11]
L1 Count 4
Analizing Cell [8,9,11]
L1 Count 3
Analizing Cell [9,10,11]
L1 Count 3
Analizing Cell [3,2,12]
L1 Count 2
L2 Count 128
Distance to 531 ROB ZETTLER           34,6554
Distance to 529 SERGEI BAUTIN        45,1774
Distance to 602 CLAUDE BOIVIN        35,5668
Distance to 541 GRANT JENNINGS       37,5500
Distance to 509 LUKE RICHARDSON      37,0135
Distance to 562 DAVE BROWN           42,9651
Distance to 502 NICK KYPREOS         40,7308
Distance to 545 BRAD MCCRIMMON       52,3927
Distance to 560 RICHARD PILON        52,3641
Distance to 646 MIKE HARTMAN         56,7891
Distance to 716 FRANCOIS LEROUX      57,8014
Distance to 613 JIM CUMMINS          49,8397
Distance to 681 BRENT THOMPSON       52,5833
Distance to 559 BOB HALKIDIS        44,5982
Distance to 595 TONY TWIST           42,8252
Distance to 718 GORD DONNELLY       44,3734
Distance to 540 MATTHEW BARNABY     36,2905
Distance to 547 DAVID MALEY          36,5650
Distance to 616 BASIL MCRAE          38,8458
Distance to 587 RYAN MCGILL          38,9487
Distance to 495 CHRIS SIMON          44,4297
Distance to 563 DOUG ZMOLEK         42,7785
Distance to 603 JOEY KOCUR           41,4849
Distance to 721 JAMIE HUSCROFT      54,9272
Distance to 591 CHRIS LIPUMA        54,5344
Distance to 656 PHILIP CROWE        57,4369
Distance to 658 SHAWN CRONIN        56,7627
Distance to 564 KEVIN KAMINSKI      49,2950
Distance to 588 STEWART MALGUNAS    50,2991
Distance to 594 DEAN CHYNOWETH     46,5725
Distance to 533 DARIN KIMBLE        48,7750
Distance to 394 DARREN RUMBLE       24,1039
    
```

```

Distance to 392 DIXON WARD          34,0735
Distance to 437 MIKE EAGLES        26,9444
Number of players within 29,6985 distance: 3
Analizing Cell [4,4,12]
L1 Count 11
Analizing Cell [3,5,12]
L1 Count 11
Analizing Cell [5,5,12]
L1 Count 11
Analizing Cell [7,6,12]
L1 Count 7
Analizing Cell [8,6,12]
L1 Count 4
Analizing Cell [11,7,12]
L1 Count 3
Analizing Cell [2,8,12]
L1 Count 2
L2 Count 138
Distance to 529 SERGEI BAUTIN      43,0000
Distance to 602 CLAUDE BOIVIN     42,1070
Distance to 541 GRANT JENNINGS    40,9145
Distance to 509 LUKE RICHARDSON   45,0999
Distance to 562 DAVE BROWN        49,5580
Distance to 502 NICK KYPREOS      52,3546
Distance to 545 BRAD MCCRIMMON    44,1248
Distance to 560 RICHARD PILON     40,8412
Distance to 646 MIKE HARTMAN      46,3357
Distance to 716 FRANCOIS LEROUX   46,3573
Distance to 613 JIM CUMMINS       34,7275
Distance to 681 BRENT THOMPSON    37,6962
Distance to 559 BOB HALKIDIS     29,3428
Distance to 595 TONY TWIST        28,3901
Number of players within 29,6985 distance: 3
Analizing Cell [12,8,12]
L1 Count 3
Analizing Cell [8,9,12]
L1 Count 3
Analizing Cell [1,0,13]
L1 Count 1
L2 Count 30
Distance to 531 ROB ZETTLER        31,0483
Distance to 529 SERGEI BAUTIN     54,3323
Distance to 602 CLAUDE BOIVIN     40,9145
Distance to 541 GRANT JENNINGS    42,2019
Distance to 509 LUKE RICHARDSON   40,5339
Distance to 562 DAVE BROWN        44,5533
Distance to 502 NICK KYPREOS      41,6413
Distance to 392 DIXON WARD        44,0454
Distance to 437 MIKE EAGLES       36,1801
Distance to 467 RICK ZOMBO        47,0213
Distance to 471 DAVID SHAW        50,0700
Distance to 452 SCOTT THORNTON    37,2156
Distance to 413 MURRAY BARON      36,6879
Distance to 442 MARK JANSSENS     42,6497

```

Distance to 332 GORD DINEEN 27,7308  
 Distance to 305 JOEL OTTO 41,5812  
 Distance to 330 ROMAN HAMRLIK 41,1461  
 Distance to 235 BORIS MIRONOV 24,1039  
 Distance to 284 STEPHANE QUINTAL 27,4773  
 Number of players within 29,6985 distance: 3  
 Analizing Cell [3,3,13]  
 L1 Count 5  
 Analizing Cell [3,5,13]  
 L1 Count 9  
 Analizing Cell [4,5,13]  
 L1 Count 10  
 Analizing Cell [5,5,13]  
 L1 Count 7  
 Analizing Cell [6,5,13]  
 L1 Count 6  
 Analizing Cell [5,6,13]  
 L1 Count 6  
 Analizing Cell [6,6,13]  
 L1 Count 6  
 Analizing Cell [8,7,13]  
 L1 Count 4  
 Analizing Cell [1,8,13]  
 L1 Count 3  
 Analizing Cell [9,12,13]  
 L1 Count 1  
 L2 Count 17  
 Distance to 137 BERNIE NICHOLLS 50,4083  
 Distance to 140 STEVE CHIASSON 49,1935  
 Distance to 157 BOB BEERS 86,7525  
 Distance to 098 ALEXEI KOVALEV 58,9322  
 Distance to 096 VLADIMIR MALAKHOV 45,1774  
 Distance to 088 DARRIN SHANNON 77,8845  
 Distance to 059 JOHN MACLEAN 29,6985  
 Distance to 043 PAUL COFFEY 25,7294  
 Distance to 037 ADAM GRAVES 30,0333  
 Distance to 033 AL MACINNIS 25,0799  
 Number of players within 29,6985 distance: 3  
 Analizing Cell [4,3,14]  
 L1 Count 5  
 Analizing Cell [3,4,14]  
 L1 Count 10  
 Analizing Cell [8,7,14]  
 L1 Count 2  
 L2 Count 49  
 Distance to 164 CHRIS GRATTON 48,8262  
 Distance to 173 KEVIN HATCHER 43,6921  
 Distance to 176 RICK TOCCHET 43,8292  
 Distance to 200 MIKE CRAIG 46,7547  
 Distance to 211 DARRYL SYDOR 51,4004  
 Distance to 208 ERIC DESJARDINS 46,4435  
 Distance to 162 KEVIN DINEEN 38,2099  
 Distance to 191 DALE HUNTER 38,0657  
 Distance to 193 JIRI SLEGR 51,1664

Distance to 174 RON SUTTER 46,4543  
 Distance to 204 KEITH JONES 44,9667  
 Distance to 175 BRIAN SKRUDLAND 32,7567  
 Distance to 157 BOB BEERS 58,8812  
 Distance to 138 DENIS SAVARD 33,4813  
 Distance to 120 MIKE RICCI 32,2025  
 Distance to 145 GARRY VALK 34,6843  
 Distance to 139 MIKE KEANE 25,9422  
 Distance to 148 CLAUDE LEMIEUX 45,2659  
 Distance to 147 BILL GUERIN 34,0735  
 Distance to 137 BERNIE NICHOLLS 45,0111  
 Distance to 140 STEVE CHIASSON 24,1039

Number of players within 29,6985 distance: 3  
 Analizing Cell [5,8,14]

L1 Count 2

L2 Count 86

Distance to 452 SCOTT THORNTON 50,7247  
 Distance to 413 MURRAY BARON 44,5646  
 Distance to 442 MARK JANSSENS 47,7598  
 Distance to 391 MARC BERGEVIN 51,0784  
 Distance to 432 BILL LINDSAY 46,2817  
 Distance to 433 ALAN MAY 40,0625  
 Distance to 419 CRAIG LUDWIG 36,7287  
 Distance to 459 DARIUS KASPARAITIS 46,4112  
 Distance to 430 RANDY MOLLER 50,9608  
 Distance to 489 GLEN FEATHERSTONE 52,4690  
 Distance to 373 STEVEN FINN 53,7215  
 Distance to 453 BRENT SEVERYN 52,0096  
 Distance to 387 BOB ROUSE 37,7094  
 Distance to 473 JOE CIRELLA 43,6578  
 Distance to 487 JAY WELLS 41,0122  
 Distance to 428 KEVIN HALLER 36,0694  
 Distance to 440 NEIL WILKINSON 38,0132  
 Distance to 484 RUDY POESCHEK 39,7618  
 Distance to 416 PETER TAGLIANETTI 39,5980  
 Distance to 372 DAVE KARPA 42,4971  
 Distance to 390 PETR SVOBODA 45,0000  
 Distance to 375 JOE REEKIE 44,7325  
 Distance to 426 KJELL SAMUELSSON 33,2566  
 Distance to 305 JOEL OTTO 50,8429  
 Distance to 330 ROMAN HAMRLIK 41,8927  
 Distance to 342 TROY LONEY 48,6724  
 Distance to 350 BILL BERG 44,3847  
 Distance to 296 RANDY CUNNEYWORTH 35,9444  
 Distance to 354 GARTH BUTCHER 41,7852  
 Distance to 324 TONY GRANATO 42,0714  
 Distance to 289 BOB SWEENEY 37,6962  
 Distance to 313 MICHEL PETIT 28,5832  
 Distance to 338 DANA MURZYN 31,8434  
 Distance to 290 MARK OSBORNE 34,5688  
 Distance to 298 MARK TINORDI 32,3419  
 Distance to 295 BRENT HUGHES 31,2090  
 Distance to 285 WAYNE PRESLEY 28,3373

Number of players within 29,6985 distance: 3



Analizing Cell [12,8,14]

L1 Count 1

L2 Count 17

|                                 |         |
|---------------------------------|---------|
| Distance to 064 GARRY GALLEY    | 59,5735 |
| Distance to 045 STEVE THOMAS    | 46,0435 |
| Distance to 060 JOE MURPHY      | 44,3283 |
| Distance to 044 WENDEL CLARK    | 34,3802 |
| Distance to 063 GEOFF COURTNALL | 37,5366 |
| Distance to 059 JOHN MACLEAN    | 48,4768 |
| Distance to 043 PAUL COFFEY     | 36,1939 |
| Distance to 037 ADAM GRAVES     | 28,7054 |
| Distance to 033 AL MACINNIS     | 41,4849 |
| Distance to 029 GARY ROBERTS    | 34,4238 |
| Distance to 041 SCOTT STEVENS   | 45,0999 |
| Distance to 025 DALE HAWERCHUK  | 41,4488 |
| Distance to 012 ERIC LINDROS    | 24,6779 |
| Distance to 010 DAVE ANDREYCHUK | 28,4781 |

Number of players within 29,6985 distance: 3

Analizing Cell [9,9,14]

L1 Count 1

L2 Count 32

|                                 |         |
|---------------------------------|---------|
| Distance to 138 DENIS SAVARD    | 48,1041 |
| Distance to 120 MIKE RICCI      | 47,7074 |
| Distance to 145 GARRY VALK      | 47,3920 |
| Distance to 139 MIKE KEANE      | 39,9249 |
| Distance to 148 CLAUDE LEMIEUX  | 55,6956 |
| Distance to 147 BILL GUERIN     | 45,4973 |
| Distance to 137 BERNIE NICHOLLS | 52,7162 |
| Distance to 140 STEVE CHIASSON  | 34,8425 |
| Distance to 094 KIRK MULLER     | 47,2123 |
| Distance to 108 SHAYNE CORSON   | 43,9431 |
| Distance to 109 ESA TIKKANEN    | 35,7491 |
| Distance to 081 SCOTT MELLANBY  | 39,6737 |
| Distance to 098 ALEXEI KOVALEV  | 36,5923 |
| Distance to 072 TODD ELIK       | 45,7493 |
| Distance to 070 ROB BLAKE       | 37,1080 |
| Distance to 067 JASON ARNOTT    | 36,4143 |
| Distance to 045 STEVE THOMAS    | 38,1576 |
| Distance to 060 JOE MURPHY      | 31,8277 |
| Distance to 044 WENDEL CLARK    | 21,0238 |
| Distance to 063 GEOFF COURTNALL | 15,5242 |
| Distance to 059 JOHN MACLEAN    | 33,3766 |
| Distance to 043 PAUL COFFEY     | 21,1187 |

Number of players within 29,6985 distance: 3

Analizing Cell [0,3,15]

L1 Count 7

Analizing Cell [3,3,15]

L1 Count 6

Analizing Cell [2,4,15]

L1 Count 8

Analizing Cell [4,4,15]

L1 Count 6

Analizing Cell [4,5,15]

L1 Count 4

Analizing Cell [7,5,15]

L1 Count 2

L2 Count 48

|                 |                    |         |
|-----------------|--------------------|---------|
| Distance to 235 | BORIS MIRONOV      | 52,6688 |
| Distance to 284 | STEPHANE QUINTAL   | 49,1121 |
| Distance to 233 | CHRIS JOSEPH       | 39,5727 |
| Distance to 263 | BOB BASSEN         | 56,0714 |
| Distance to 238 | BOB ERREY          | 39,7618 |
| Distance to 246 | BRIAN BENNING      | 48,4149 |
| Distance to 280 | RICH SUTTER        | 51,0490 |
| Distance to 248 | CHRIS PRONGER      | 45,1221 |
| Distance to 249 | JAMIE MACOUN       | 43,9545 |
| Distance to 269 | SERGIO MOMESSO     | 43,0116 |
| Distance to 220 | VLAD. KONSTANTINOV | 50,9411 |
| Distance to 164 | CHRIS GRATTON      | 34,5832 |
| Distance to 173 | KEVIN HATCHER      | 40,7554 |
| Distance to 176 | RICK TOCCHET       | 29,2746 |
| Distance to 200 | MIKE CRAIG         | 31,8434 |
| Distance to 208 | ERIC DESJARDINS    | 52,2015 |
| Distance to 162 | KEVIN DINEEN       | 35,4401 |
| Distance to 191 | DALE HUNTER        | 30,7409 |
| Distance to 194 | SCOTT PEARSON      | 41,8808 |
| Distance to 204 | KEITH JONES        | 36,7967 |
| Distance to 175 | BRIAN SKRUDLAND    | 34,4238 |
| Distance to 138 | DENIS SAVARD       | 38,4838 |
| Distance to 120 | MIKE RICCI         | 29,4788 |

Number of players within 29,6985 distance: 3

Analizing Cell [0,7,15]

L1 Count 5

Analizing Cell [4,7,15]

L1 Count 2

L2 Count 104

|                 |                 |         |
|-----------------|-----------------|---------|
| Distance to 531 | ROB ZETTLER     | 51,1273 |
| Distance to 576 | BILL HUARD      | 54,7357 |
| Distance to 602 | CLAUDE BOIVIN   | 62,8490 |
| Distance to 541 | GRANT JENNINGS  | 42,2493 |
| Distance to 509 | LUKE RICHARDSON | 41,5331 |
| Distance to 562 | DAVE BROWN      | 43,0232 |
| Distance to 502 | NICK KYPREOS    | 43,2897 |
| Distance to 595 | TONY TWIST      | 52,1249 |
| Distance to 718 | GORD DONNELLY   | 57,4282 |
| Distance to 540 | MATTHEW BARNABY | 49,5580 |
| Distance to 547 | DAVID MALEY     | 50,7937 |
| Distance to 616 | BASIL MCRAE     | 53,4603 |
| Distance to 587 | RYAN MCGILL     | 46,8615 |
| Distance to 495 | CHRIS SIMON     | 35,5668 |
| Distance to 563 | DOUG ZMOLEK     | 40,9268 |
| Distance to 603 | JOEY KOCUR      | 43,6119 |
| Distance to 721 | JAMIE HUSCROFT  | 42,5441 |
| Distance to 504 | JIM MCKENZIE    | 39,0384 |
| Distance to 518 | TIM HUNTER      | 52,0961 |
| Distance to 594 | DEAN CHYNOWETH  | 39,8999 |
| Distance to 533 | DARIN KIMBLE    | 35,8608 |
| Distance to 633 | PAUL LAUS       | 46,7868 |

Distance to 528 TERRY CARKNER 33,5410  
 Distance to 557 ROBERT DIRK 35,7071  
 Distance to 437 MIKE EAGLES 58,9322  
 Distance to 468 DEAN KENNEDY 53,9351  
 Distance to 452 SCOTT THORNTON 51,4684  
 Distance to 413 MURRAY BARON 39,6737  
 Distance to 442 MARK JANSSENS 38,2230  
 Distance to 376 IGOR ULANOV 44,1135  
 Distance to 400 BRYAN MARCHMENT 47,4763  
 Distance to 432 BILL LINDSAY 50,2991  
 Distance to 433 ALAN MAY 38,4838  
 Distance to 419 CRAIG LUDWIG 32,2645  
 Distance to 459 DARIUS KASPARAITIS 35,1852  
 Distance to 430 RANDY MOLLER 37,1080  
 Distance to 489 GLEN FEATHERSTONE 39,2556  
 Distance to 373 STEVEN FINN 39,2683  
 Distance to 453 BRENT SEVERYN 37,9078  
 Distance to 387 BOB ROUSE 42,7317  
 Distance to 473 JOE CIRELLA 47,8957  
 Distance to 487 JAY WELLS 41,4488  
 Distance to 428 KEVIN HALLER 33,9559  
 Distance to 440 NEIL WILKINSON 37,2693  
 Distance to 484 RUDY POESCHEK 37,2156  
 Distance to 416 PETER TAGLIANETTI 27,8568  
 Distance to 372 DAVE KARPA 29,0172

Number of players within 29,6985 distance: 3  
 Analizing Cell [0,8,15]

L1 Count 2

L2 Count 73

Distance to 602 CLAUDE BOIVIN 50,2494  
 Distance to 541 GRANT JENNINGS 28,4605  
 Distance to 509 LUKE RICHARDSON 31,1448  
 Distance to 562 DAVE BROWN 30,5941  
 Distance to 502 NICK KYPREOS 35,0571  
 Distance to 595 TONY TWIST 35,5246  
 Distance to 718 GORD DONNELLY 40,1622  
 Distance to 540 MATTHEW BARNABY 35,3695  
 Distance to 547 DAVID MALEY 36,8103  
 Distance to 616 BASIL MCRAE 37,5899  
 Distance to 587 RYAN MCGILL 29,8496  
 Distance to 495 CHRIS SIMON 20,2485

Number of players within 29,6985 distance: 3  
 Analizing Cell [4,4,16]

L1 Count 5

Analizing Cell [2,5,16]

L1 Count 12

Analizing Cell [8,5,16]

L1 Count 2

L2 Count 32

Distance to 164 CHRIS GRATTON 40,0125  
 Distance to 173 KEVIN HATCHER 46,9255  
 Distance to 176 RICK TOCCHET 35,8608  
 Distance to 200 MIKE CRAIG 38,3275  
 Distance to 162 KEVIN DINEEN 42,0119

```

Distance to 191 DALE HUNTER           38,1838
Distance to 194 SCOTT PEARSON         46,3141
Distance to 204 KEITH JONES           43,2319
Distance to 175 BRIAN SKRUDLAND       41,4488
Distance to 138 DENIS SAVARD          44,6542
Distance to 120 MIKE RICCI            35,3836
Distance to 146 BRAD MAY               43,9659
Distance to 139 MIKE KEANE            38,2884
Distance to 140 STEVE CHIASSON        42,4971
Distance to 108 SHAYNE CORSON         29,7153
Distance to 109 ESA TIKKANEN          35,5246
Distance to 081 SCOTT MELLANBY        20,1494
Distance to 090 AL IAFRATE            38,6005
Distance to 098 ALEXEI KOVALEV        36,2629
Distance to 067 JASON ARNOTT          37,0675
Distance to 046 PAT VERBEEK           38,4708
Distance to 060 JOE MURPHY            30,1496
Distance to 044 WENDEL CLARK          30,6268
Distance to 063 GEOFF COURTNALL      29,2746
Number of players within 29,6985 distance: 3
Analizing Cell [2,6,16]
L1 Count 11
Analizing Cell [2,7,16]
L1 Count 5
Analizing Cell [3,9,16]
L1 Count 1
L2 Count 65
Distance to 540 MATTHEW BARNABY       55,8749
Distance to 547 DAVID MALEY           57,0438
Distance to 616 BASIL MCRAE           59,1101
Distance to 587 RYAN MCGILL           52,3641
Distance to 495 CHRIS SIMON           41,0488
Distance to 563 DOUG ZMOLEK           47,4658
Distance to 603 JOEY KOCUR            50,0200
Distance to 721 JAMIE HUSCROFT        45,6508
Distance to 504 JIM MCKENZIE          45,3652
Distance to 518 TIM HUNTER            55,9821
Distance to 594 DEAN CHYNOWETH        42,7317
Distance to 533 DARIN KIMBLE          39,2173
Distance to 633 PAUL LAUS             47,3603
Distance to 528 TERRY CARKNER         32,0780
Distance to 557 ROBERT DIRK           31,2570
Distance to 433 ALAN MAY              45,3762
Distance to 419 CRAIG LUDWIG          39,3319
Distance to 459 DARIUS KASPARAITIS    42,3792
Distance to 430 RANDY MOLLER          43,3705
Distance to 489 GLEN FEATHERSTONE     44,6878
Distance to 373 STEVEN FINN           47,0956
Distance to 453 BRENT SEVERYN         42,0595
Distance to 463 SANDY MCCARTHY        53,3198
Distance to 487 JAY WELLS             45,1221
Distance to 428 KEVIN HALLER          39,1024
Distance to 440 NEIL WILKINSON        42,4382
Distance to 484 RUDY POESCHEK        41,2916

```

Distance to 416 PETER TAGLIANETTI 31,6544  
 Distance to 372 DAVE KARPA 35,4401  
 Distance to 375 JOE REEKIE 28,3725  
 Distance to 426 KJELL SAMUELSSON 30,7246  
 Distance to 380 JEFF BEUKEBOOM 38,1707  
 Distance to 474 KEN DANAYKO 44,5197  
 Distance to 354 GARTH BUTCHER 39,8748  
 Distance to 324 TONY GRANATO 36,2215  
 Distance to 313 MICHEL PETIT 38,8458  
 Distance to 338 DANA MURZYN 41,0609  
 Distance to 290 MARK OSBORNE 29,9500  
 Distance to 298 MARK TINORDI 26,1151  
 Distance to 295 BRENT HUGHES 22,4944  
 Number of players within 29,6985 distance: 3  
 Analizing Cell [9,10,16]  
 L1 Count 1  
 L2 Count 18  
 Distance to 139 MIKE KEANE 55,5068  
 Distance to 140 STEVE CHIASSON 48,7134  
 Distance to 109 ESA TIKKANEN 53,7122  
 Distance to 081 SCOTT MELLANBY 44,2832  
 Distance to 090 AL IAFRATE 37,9605  
 Distance to 098 ALEXEI KOVALEV 35,0143  
 Distance to 067 JASON ARNOTT 56,8595  
 Distance to 060 JOE MURPHY 51,4587  
 Distance to 044 WENDEL CLARK 41,1461  
 Distance to 063 GEOFF COURTNALL 34,1174  
 Distance to 043 PAUL COFFEY 40,6325  
 Distance to 052 KEITH PRIMEAU 30,2324  
 Distance to 037 ADAM GRAVES 21,1896  
 Distance to 041 SCOTT STEVENS 37,1618  
 Distance to 012 ERIC LINDROS 48,7237  
 Distance to 004 DOUG GILMOUR 49,7293  
 Distance to 006 JEREMY ROENICK 34,4238  
 Number of players within 29,6985 distance: 1  
 Analizing Cell [0,5,17]  
 L1 Count 10  
 Analizing Cell [1,5,17]  
 L1 Count 14  
 Analizing Cell [2,5,17]  
 L1 Count 12  
 Analizing Cell [3,5,17]  
 L1 Count 7  
 Analizing Cell [1,6,17]  
 L1 Count 15  
 Analizing Cell [4,6,17]  
 L1 Count 2  
 L2 Count 84  
 Distance to 531 ROB ZETTLER 43,6921  
 Distance to 576 BILL HUARD 40,7308  
 Distance to 508 DOUG HOUDA 52,7162  
 Distance to 541 GRANT JENNINGS 39,5727  
 Distance to 509 LUKE RICHARDSON 36,6333  
 Distance to 562 DAVE BROWN 36,0555

Distance to 502 NICK KYPREOS 35,0571  
 Distance to 587 RYAN MCGILL 49,1019  
 Distance to 495 CHRIS SIMON 32,4654  
 Distance to 563 DOUG ZMOLEK 36,0555  
 Distance to 603 JOEY KOCUR 39,9124  
 Distance to 721 JAMIE HUSCROFT 34,8855  
 Distance to 504 JIM MCKENZIE 29,3087  
 Distance to 518 TIM HUNTER 37,2693  
 Distance to 497 KEN BAUMGARTNER 46,0977  
 Distance to 647 GREG SMYTH 48,0521  
 Distance to 594 DEAN CHYNOWETH 41,1218  
 Distance to 533 DARIN KIMBLE 33,1813  
 Distance to 528 TERRY CARKNER 35,0143  
 Distance to 557 ROBERT DIRK 37,6829  
 Distance to 468 DEAN KENNEDY 39,0640  
 Distance to 413 MURRAY BARON 37,9605  
 Distance to 442 MARK JANSSENS 31,0161  
 Distance to 376 IGOR ULANOV 28,3725

Number of players within 29,6985 distance: 3  
 Analizing Cell [6,6,17]

L1 Count 1

L2 Count 44

Distance to 307 TROY MALLETTE 52,4118  
 Distance to 348 SCOTT LEVINS 50,2593  
 Distance to 330 ROMAN HAMRLIK 43,7379  
 Distance to 354 GARTH BUTCHER 42,0238  
 Distance to 324 TONY GRANATO 39,0640  
 Distance to 290 MARK OSBORNE 35,2846  
 Distance to 298 MARK TINORDI 36,9865  
 Distance to 295 BRENT HUGHES 37,8418  
 Distance to 284 STEPHANE QUINTAL 51,7784  
 Distance to 233 CHRIS JOSEPH 38,0920  
 Distance to 238 BOB ERREY 39,3700  
 Distance to 248 CHRIS PRONGER 46,9574  
 Distance to 249 JAMIE MACOUN 45,6180  
 Distance to 269 SERGIO MOMESSO 33,0606  
 Distance to 275 STEVE SMITH 41,7013  
 Distance to 283 DARREN MCCARTY 48,2079  
 Distance to 220 VLAD. KONSTANTINOV 41,8330  
 Distance to 164 CHRIS GRATTON 40,3113  
 Distance to 176 RICK TOCCHET 29,1548  
 Distance to 200 MIKE CRAIG 28,7228  
 Distance to 162 KEVIN DINEEN 41,2432  
 Distance to 191 DALE HUNTER 28,7054

Number of players within 29,6985 distance: 3  
 Analizing Cell [6,8,17]

L1 Count 1

L2 Count 39

Distance to 330 ROMAN HAMRLIK 51,0882  
 Distance to 354 GARTH BUTCHER 45,9891  
 Distance to 324 TONY GRANATO 40,5093  
 Distance to 290 MARK OSBORNE 36,0278  
 Distance to 298 MARK TINORDI 35,9026  
 Distance to 295 BRENT HUGHES 34,7707

```

Distance to 238 BOB ERREY                47,9687
Distance to 248 CHRIS PRONGER            52,8961
Distance to 249 JAMIE MACOUN             52,2111
Distance to 269 SERGIO MOMESSO          35,5809
Distance to 275 STEVE SMITH              42,0714
Distance to 283 DARREN MCCARTY           40,8044
Distance to 220 VLAD. KONSTANTINOV      30,4795
Distance to 176 RICK TOCCHET             41,7732
Distance to 200 MIKE CRAIG               40,1248
Distance to 162 KEVIN DINEEN             51,0490
Distance to 191 DALE HUNTER              36,5650
Distance to 194 SCOTT PEARSON            31,0805
Distance to 204 KEITH JONES              25,7294
Distance to 175 BRIAN SKRUDLAND          24,5967
Distance to 120 MIKE RICCI               49,3457
Distance to 146 BRAD MAY                 31,4006
Distance to 139 MIKE KEANE               38,3275
Distance to 140 STEVE CHIASSON           33,5410
Distance to 108 SHAYNE CORSON            44,4522
Distance to 109 ESA TIKKANEN             42,1070
Distance to 081 SCOTT MELLANBY           19,1050
Number of players within 29,6985 distance: 3
Analizing Cell [0,3,18]
L1 Count 4
Analizing Cell [2,3,18]
L1 Count 5
Analizing Cell [10,3,18]
L1 Count 1
L2 Count 7
Distance to 081 SCOTT MELLANBY            37,3631
Distance to 090 AL IAFRATE                50,6360
Distance to 070 ROB BLAKE                 31,8277
Distance to 046 PAT VERBEEK               27,0924
Distance to 045 STEVE THOMAS              25,4951
Distance to 063 GEOFF COURTNALL          53,5630
Number of players within 29,6985 distance: 2
Analizing Cell [1,5,18]
L1 Count 10
Analizing Cell [1,7,18]
L1 Count 3
Analizing Cell [2,2,19]
L1 Count 3
Analizing Cell [1,3,19]
L1 Count 6
Analizing Cell [1,4,19]
L1 Count 9
Analizing Cell [0,5,19]
L1 Count 6
Analizing Cell [4,5,19]
L1 Count 3
Analizing Cell [5,5,19]
L1 Count 2
L2 Count 50
Distance to 468 DEAN KENNEDY              39,1152

```

Distance to 438 KRIS KING 50,0100  
 Distance to 442 MARK JANSSENS 47,8957  
 Distance to 376 IGOR ULANOV 29,0689  
 Distance to 400 BRYAN MARCHMENT 31,4484  
 Distance to 459 DARIUS KASPARAITIS 44,6878  
 Distance to 430 RANDY MOLLER 36,2491  
 Distance to 489 GLEN FEATHERSTONE 40,7185  
 Distance to 373 STEVEN FINN 30,6105  
 Distance to 453 BRENT SEVERYN 37,4967  
 Distance to 463 SANDY MCCARTHY 35,1852  
 Distance to 456 PAUL KRUSE 36,7696  
 Distance to 464 LOUIE DEBRUSK 37,8153  
 Distance to 416 PETER TAGLIANETTI 43,8520  
 Distance to 372 DAVE KARPA 36,7287  
 Distance to 375 JOE REEKIE 38,0789  
 Distance to 380 JEFF BEUKEBOOM 37,6563  
 Distance to 474 KEN DANAYKO 48,3632  
 Distance to 307 TROY MALLETTE 35,1852  
 Distance to 348 SCOTT LEVINS 34,0147  
 Distance to 331 KELLY BUCHBERGER 39,4462  
 Distance to 330 ROMAN HAMRLIK 44,0000  
 Distance to 329 DAVE MANSON 32,2490  
 Distance to 354 GARTH BUTCHER 38,2230  
 Distance to 324 TONY GRANATO 32,1403  
 Distance to 290 MARK OSBORNE 33,7639  
 Distance to 298 MARK TINORDI 37,0000  
 Distance to 295 BRENT HUGHES 38,4838  
 Distance to 233 CHRIS JOSEPH 40,5709  
 Distance to 234 MARTY MCSORLEY 27,5862

Number of players within 29,6985 distance: 3  
 Analizing Cell [6,7,19]

L1 Count 1

L2 Count 34

Distance to 348 SCOTT LEVINS 57,3236  
 Distance to 331 KELLY BUCHBERGER 62,9365  
 Distance to 330 ROMAN HAMRLIK 55,2540  
 Distance to 329 DAVE MANSON 55,2540  
 Distance to 354 GARTH BUTCHER 49,4975  
 Distance to 324 TONY GRANATO 43,1509  
 Distance to 290 MARK OSBORNE 40,1123  
 Distance to 298 MARK TINORDI 40,6940  
 Distance to 295 BRENT HUGHES 39,8999  
 Distance to 233 CHRIS JOSEPH 53,1695  
 Distance to 234 MARTY MCSORLEY 49,7393  
 Distance to 269 SERGIO MOMESSO 38,4838  
 Distance to 275 STEVE SMITH 39,0256  
 Distance to 240 ROB PEARSON 44,0908  
 Distance to 283 DARREN MCCARTY 36,9324  
 Distance to 258 ULF SAMUELSSON 46,7547  
 Distance to 220 VLAD. KONSTANTINOV 38,0263  
 Distance to 176 RICK TOCCHET 46,1086  
 Distance to 200 MIKE CRAIG 43,7836  
 Distance to 191 DALE HUNTER 42,7083  
 Distance to 194 SCOTT PEARSON 29,0689



```

Distance to 204 KEITH JONES          29,4788
Distance to 175 BRIAN SKRUDLAND      32,5883
Distance to 146 BRAD MAY             26,7769
Number of players within 29,6985 distance: 3
Analizing Cell [1,8,19]
L1 Count 3
Analizing Cell [8,4,20]
L1 Count 1
L2 Count 12
Distance to 200 MIKE CRAIG           53,7494
Distance to 194 SCOTT PEARSON        41,3401
Distance to 204 KEITH JONES          52,3927
Distance to 146 BRAD MAY             31,8904
Distance to 081 SCOTT MELLANBY       35,1283
Distance to 090 AL IAFRATE           38,0263
Distance to 083 CHRIS CHELIOS        46,6798
Distance to 098 ALEXEI KOVALEV       44,4860
Distance to 045 STEVE THOMAS         38,4708
Distance to 023 KEVIN STEVENS        27,0924
Distance to 008 BRENDAN SHANAHAN     43,8292
Number of players within 29,6985 distance: 1
Analizing Cell [1,5,20]
L1 Count 8
Analizing Cell [3,5,20]
L1 Count 2
L2 Count 54
Distance to 576 BILL HUARD            29,4788
Distance to 508 DOUG HOUDA           27,4408
Number of players within 29,6985 distance: 3
Analizing Cell [1,9,20]
L1 Count 2
L2 Count 36
Distance to 721 JAMIE HUSCROFT       44,1135
Distance to 504 JIM MCKENZIE         45,3872
Distance to 518 TIM HUNTER           34,4964
Distance to 497 KEN BAUMGARTNER      34,2637
Distance to 647 GREG SMYTH           32,7719
Distance to 546 STU GRIMSON          40,4228
Distance to 522 DENNIS VIAL          52,4309
Distance to 511 CAM RUSSELL          29,4788
Distance to 459 DARIUS KASPARAITIS   47,3920
Distance to 430 RANDY MOLLER         38,9487
Distance to 489 GLEN FEATHERSTONE    40,0125
Distance to 373 STEVEN FINN          40,4228
Distance to 453 BRENT SEVERYN        34,4238
Distance to 463 SANDY MCCARTHY       30,1496
Distance to 456 PAUL KRUSE           34,2199
Distance to 464 LOUIE DEBRUSK       37,1080
Distance to 416 PETER TAGLIANETTI    40,6940
Distance to 372 DAVE KARPA           39,5221
Distance to 375 JOE REEKIE           24,3516
Number of players within 29,6985 distance: 3
Analizing Cell [8,10,20]
L1 Count 2

```

```

L2 Count 8
Distance to 204 KEITH JONES          54,0370
Distance to 081 SCOTT MELLANBY      43,6005
Distance to 090 AL IAFRATE          25,4755
Distance to 083 CHRIS CHELIOS       46,6262
Distance to 098 ALEXEI KOVALEV      30,0998
Distance to 029 GARY ROBERTS        30,2324
Number of players within 29,6985 distance: 2
Analizing Cell [0,3,21]
L1 Count 2
L2 Count 36
Distance to 576 BILL HUARD           26,3059
Distance to 504 JIM MCKENZIE        43,6807
Distance to 518 TIM HUNTER           20,8327
Number of players within 29,6985 distance: 3
Analizing Cell [0,5,21]
L1 Count 6
Analizing Cell [1,5,21]
L1 Count 7
Analizing Cell [3,7,21]
L1 Count 1
L2 Count 44
Distance to 576 BILL HUARD           42,4971
Distance to 508 DOUG HOUDA           36,5240
Distance to 617 SHAWN ANTOSKI        33,7491
Distance to 504 JIM MCKENZIE        43,7035
Distance to 518 TIM HUNTER           28,6705
Distance to 497 KEN BAUMGARTNER      25,7682
Distance to 647 GREG SMYTH           28,9137
Number of players within 29,6985 distance: 3
Analizing Cell [9,9,21]
L1 Count 2
L2 Count 8
Distance to 146 BRAD MAY              55,8659
Distance to 081 SCOTT MELLANBY      53,7959
Distance to 090 AL IAFRATE           38,1314
Distance to 083 CHRIS CHELIOS       40,3113
Distance to 098 ALEXEI KOVALEV      44,8219
Distance to 008 BRENDAN SHANAHAN    49,3457
Number of players within 29,6985 distance: 1
Analizing Cell [0,4,22]
L1 Count 8
Analizing Cell [2,4,22]
L1 Count 7
Analizing Cell [3,4,22]
L1 Count 4
Analizing Cell [3,5,22]
L1 Count 3
Analizing Cell [1,3,23]
L1 Count 4
Analizing Cell [2,3,23]
L1 Count 4
Analizing Cell [0,5,23]
L1 Count 3

```

Analizing Cell [0,7,23]  
 L1 Count 1  
 L2 Count 30  
 Distance to 508 DOUG HOUDA 31,3847  
 Distance to 617 SHAWN ANTOSKI 23,7908  
 Distance to 518 TIM HUNTER 33,6303  
 Distance to 497 KEN BAUMGARTNER 21,9317  
 Distance to 647 GREG SMYTH 22,8254  
 Number of players within 29,6985 distance: 3

Analizing Cell [3,8,23]  
 L1 Count 2  
 L2 Count 29  
 Distance to 617 SHAWN ANTOSKI 43,7379  
 Distance to 518 TIM HUNTER 46,5618  
 Distance to 497 KEN BAUMGARTNER 38,4448  
 Distance to 647 GREG SMYTH 41,4005  
 Distance to 546 STU GRIMSON 37,0135  
 Distance to 522 DENNIS VIAL 41,6293  
 Distance to 618 ENRICO CICCONE 46,1952  
 Distance to 568 MICK VUKOTA 53,4135  
 Distance to 511 CAM RUSSELL 24,7184  
 Distance to 376 IGOR ULANOV 49,5580  
 Distance to 400 BRYAN MARCHMENT 51,5170  
 Distance to 463 SANDY MCCARTHY 41,3884  
 Distance to 456 PAUL KRUSE 36,8917  
 Distance to 464 LOUIE DEBRUSK 39,7618  
 Distance to 380 JEFF BEUKEBOOM 32,1714  
 Distance to 474 KEN DANEYKO 30,0998  
 Distance to 329 DAVE MANSON 38,6911  
 Distance to 320 JEFF ODGERS 43,4626  
 Distance to 340 MIKE PELUSO 40,2244  
 Distance to 234 MARTY MCSORLEY 35,4119  
 Distance to 275 STEVE SMITH 37,5899  
 Distance to 240 ROB PEARSON 30,6920  
 Distance to 283 DARREN MCCARTY 21,3073  
 Number of players within 29,6985 distance: 3

Analizing Cell [0,5,24]  
 L1 Count 2  
 L2 Count 26  
 Distance to 508 DOUG HOUDA 27,8747  
 Distance to 617 SHAWN ANTOSKI 24,4131  
 Number of players within 29,6985 distance: 3

Analizing Cell [11,5,24]  
 L1 Count 1  
 L2 Count 3  
 Distance to 046 PAT VERBEEK 43,8292  
 Distance to 026 THEOREN FLEURY 49,3457  
 Number of players within 29,6985 distance: 0

Analizing Cell [6,7,24]  
 L1 Count 1  
 L2 Count 16  
 Distance to 331 KELLY BUCHBERGER 52,0961  
 Distance to 329 DAVE MANSON 51,3615  
 Distance to 320 JEFF ODGERS 47,3920

```

Distance to 340 MIKE PELUSO           48,2183
Distance to 234 MARTY MCSORLEY        41,7253
Distance to 275 STEVE SMITH           53,1225
Distance to 240 ROB PEARSON           41,8688
Distance to 255 RONNIE STERN          44,2493
Distance to 283 DARREN MCCARTY        46,0109
Distance to 258 ULF SAMUELSSON        35,3695
Distance to 232 DERIAN HATCHER        29,8496
Distance to 271 RANDY MCKAY           47,5079
Distance to 046 PAT VERBEEK           46,6798
Distance to 052 KEITH PRIMEAU         46,6262
Distance to 026 THEOREN FLEURY        40,3113
Number of players within 29,6985 distance: 0
Analizing Cell [3,8,24]
L1 Count 2
L2 Count 24
Distance to 617 SHAWN ANTOSKI         46,0977
Distance to 497 KEN BAUMGARTNER       42,7785
Distance to 647 GREG SMYTH            46,4112
Distance to 546 STU GRIMSON           37,3363
Distance to 522 DENNIS VIAL           37,0000
Distance to 618 ENRICO CICCONE        39,2173
Distance to 568 MICK VUKOTA           44,5084
Distance to 511 CAM RUSSELL           27,0370
Distance to 463 SANDY MCCARTHY        48,6724
Distance to 456 PAUL KRUSE            41,2432
Distance to 464 LOUIE DEBRUSK        43,6005
Distance to 474 KEN DANAYKO           41,5933
Distance to 329 DAVE MANSON           39,8623
Distance to 320 JEFF ODGERS           35,2846
Distance to 340 MIKE PELUSO           29,1548
Number of players within 29,6985 distance: 3
Analizing Cell [2,4,25]
L1 Count 1
L2 Count 24
Distance to 508 DOUG HOUDA            36,8917
Distance to 617 SHAWN ANTOSKI         36,7696
Distance to 497 KEN BAUMGARTNER       39,8372
Distance to 647 GREG SMYTH            44,3058
Distance to 546 STU GRIMSON           28,3373
Distance to 522 DENNIS VIAL           16,6132
Distance to 618 ENRICO CICCONE        20,5183
Number of players within 29,6985 distance: 3
Analizing Cell [0,5,26]
L1 Count 2
L2 Count 17
Distance to 617 SHAWN ANTOSKI         36,6742
Distance to 546 STU GRIMSON           27,2397
Distance to 522 DENNIS VIAL           13,6015
Number of players within 29,6985 distance: 3
Analizing Cell [0,5,27]
L1 Count 2
L2 Count 17
Distance to 546 STU GRIMSON           38,0657

```

```

Distance to 522 DENNIS VIAL                23,5372
Distance to 556 MARC POTVIN                 35,1426
Distance to 519 ROB RAY                    37,7757
Distance to 511 CAM RUSSELL                40,1248
Distance to 438 KRIS KING                  37,1080
Distance to 331 KELLY BUCHBERGER          44,2493
Distance to 320 JEFF ODGERS                24,0416
Number of players within 29,6985 distance: 3
Analizing Cell [2,8,27]
L1 Count 2
L2 Count 16
Distance to 546 STU GRIMSON                48,3942
Distance to 522 DENNIS VIAL                 39,1024
Distance to 618 ENRICO CICCONE             31,0161
Distance to 568 MICK VUKOTA                28,8617
Distance to 556 MARC POTVIN                 45,9347
Distance to 519 ROB RAY                    41,8808
Distance to 511 CAM RUSSELL                40,8534
Distance to 320 JEFF ODGERS                35,7911
Distance to 358 TODD EWEN                  42,8486
Distance to 255 RONNIE STERN               16,5831
Number of players within 29,6985 distance: 3
Analizing Cell [3,6,28]
L1 Count 1
L2 Count 16
Distance to 522 DENNIS VIAL                 39,3700
Distance to 618 ENRICO CICCONE             32,6343
Distance to 568 MICK VUKOTA                27,9643
Distance to 556 MARC POTVIN                 40,1622
Distance to 523 KELLY CHASE                42,7785
Distance to 519 ROB RAY                    38,2230
Distance to 370 BOB PROBERT                34,8855
Distance to 320 JEFF ODGERS                29,4279
Distance to 358 TODD EWEN                  33,6303
Distance to 340 MIKE PELUSO                16,5831
Number of players within 29,6985 distance: 3
Analizing Cell [3,8,28]
L1 Count 2
L2 Count 16
Distance to 522 DENNIS VIAL                 48,8774
Distance to 618 ENRICO CICCONE             41,0366
Distance to 568 MICK VUKOTA                37,6696
Distance to 556 MARC POTVIN                 47,8748
Distance to 523 KELLY CHASE                48,9592
Distance to 519 ROB RAY                    42,2374
Distance to 370 BOB PROBERT                41,0609
Distance to 320 JEFF ODGERS                43,4626
Distance to 358 TODD EWEN                  42,7317
Distance to 255 RONNIE STERN               18,1384
Distance to 250 GINO ODJICK                29,2233
Number of players within 29,6985 distance: 3
Analizing Cell [9,4,29]
L1 Count 1
L2 Count 1

```

```

Analizing Cell [0,5,31]
L1 Count 4
Analizing Cell [2,5,31]
L1 Count 2
L2 Count 12
Distance to 568 MICK VUKOTA           37,7492
Distance to 556 MARC POTVIN           13,0384
Distance to 523 KELLY CHASE          12,6886
Number of players within 29,6985 distance: 3
Analizing Cell [0,6,31]
L1 Count 4
Analizing Cell [3,7,31]
L1 Count 2
L2 Count 12
Distance to 568 MICK VUKOTA           45,8803
Distance to 556 MARC POTVIN           31,9061
Distance to 523 KELLY CHASE          29,2746
Distance to 519 ROB RAY               24,7790
Number of players within 29,6985 distance: 3
Analizing Cell [0,5,32]
L1 Count 4
Analizing Cell [1,5,32]
L1 Count 5
Analizing Cell [4,6,32]
L1 Count 2
L2 Count 10
Distance to 556 MARC POTVIN           38,6911
Distance to 523 KELLY CHASE          35,5246
Distance to 519 ROB RAY              33,6006
Distance to 370 BOB PROBERT           24,7184
Distance to 408 CRAIG BERUBE          40,7554
Distance to 358 TODD EWEN             26,9258
Number of players within 29,6985 distance: 3
Analizing Cell [1,5,35]
L1 Count 1
L2 Count 10
Distance to 556 MARC POTVIN           34,4384
Distance to 523 KELLY CHASE          27,9106
Distance to 519 ROB RAY              32,3419
Distance to 370 BOB PROBERT           30,2985
Distance to 424 SHANE CHURLA          28,3019
Distance to 346 WARREN RYCHEL         23,2164
Number of players within 29,6985 distance: 3
Analizing Cell [2,3,37]
L1 Count 1
L2 Count 4
Distance to 408 CRAIG BERUBE          23,2164
Distance to 424 SHANE CHURLA          16,6733
Distance to 349 TIE DOMI              27,3130
Number of players within 29,6985 distance: 3
Analizing Cell [1,5,38]
L1 Count 1
L2 Count 4
Distance to 408 CRAIG BERUBE          28,3019

```

```

Distance to 346 WARREN RYCHEL          16,6733
Distance to 349 TIE DOMI                15,2315
Number of players within 29,6985 distance: 3
Analizing Cell [2,5,40]
L1 Count 1
L2 Count 3
Distance to 424 SHANE CHURLA           15,2315
Distance to 346 WARREN RYCHEL          27,3130
Number of players within 29,6985 distance: 2
Finished
    
```

#### 14.4. OUTLIERS DETECTADOS. FINDALLOUTSM EN BANCO DE DATOS REAL

En esta sección pueden visualizarse los outliers detectados por dos implementaciones distintas del algoritmo **FindAllOutsM**. La primera versión tiene una eficacia del 100% como es característica del algoritmo original, mientras que la segunda hace una detección probabilística de outliers para ganar eficiencia.

##### 14.4.1. VERSION QUE EJECUTA TODO EL PROCESO EN MEMORIA. 100% DE EFICACIA

```

Flight Number: 1224 From: FLL To: TPA Date: 7/1/2003
ArrDelay:30
DepDelay:-1410
CarrierDelay:-1
WeatherDelay:-1
NASDelay:-1
SecurityDelay:-1
LateAircraftDelay:-1
Flight Number: 391 From: ATL To: STL Date: 18/6/1998
ArrDelay:-1233
DepDelay:168
CarrierDelay:-1
WeatherDelay:-1
NASDelay:-1
SecurityDelay:-1
LateAircraftDelay:-1
Flight Number: 1255 From: ATL To: RSW Date: 30/3/2000
ArrDelay:-989
DepDelay:-990
CarrierDelay:-1
WeatherDelay:-1
NASDelay:-1
SecurityDelay:-1
LateAircraftDelay:-1
Flight Number: 1519 From: DFW To: ELP Date: 25/2/2000
ArrDelay:-1298
DepDelay:25
CarrierDelay:-1
WeatherDelay:-1
NASDelay:-1
SecurityDelay:-1
    
```

LateAircraftDelay:-1  
Flight Number: 1718 From: EWR To: TPA Date: 27/6/2002  
ArrDelay:-987  
DepDelay:-981  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 36 From: JFK To: ROC Date: 23/2/2003  
ArrDelay:304  
DepDelay:-1124  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 1375 From: ORD To: SAN Date: 31/12/2000  
ArrDelay:-1188  
DepDelay:19  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 10 From: JFK To: BUF Date: 23/2/2003  
ArrDelay:264  
DepDelay:-1160  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 392 From: RSW To: JFK Date: 21/3/2003  
ArrDelay:290  
DepDelay:-1132  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 592 From: PDX To: MSP Date: 24/6/2003  
ArrDelay:1246  
DepDelay:1260  
CarrierDelay:1154  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:92  
Flight Number: 1484 From: ATL To: ORD Date: 27/6/2003  
ArrDelay:1205  
DepDelay:1209  
CarrierDelay:1205  
WeatherDelay:0



NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 2088 From: SNA To: ORD Date: 19/6/2003  
ArrDelay:1200  
DepDelay:1211  
CarrierDelay:1200  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 1510 From: RNO To: MSP Date: 20/8/2003  
ArrDelay:1187  
DepDelay:1201  
CarrierDelay:1187  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 365 From: BUF To: JFK Date: 21/7/2003  
ArrDelay:267  
DepDelay:-1180  
CarrierDelay:88  
WeatherDelay:0  
NASDelay:7  
SecurityDelay:0  
LateAircraftDelay:172  
Flight Number: 98 From: DEN To: JFK Date: 14/8/2003  
ArrDelay:367  
DepDelay:-1114  
CarrierDelay:23  
WeatherDelay:0  
NASDelay:41  
SecurityDelay:0  
LateAircraftDelay:303  
Flight Number: 3578 From: DTW To: DFW Date: 31/8/2003  
ArrDelay:1228  
DepDelay:1208  
CarrierDelay:1208  
WeatherDelay:0  
NASDelay:20  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 1680 From: SLC To: STL Date: 17/8/2003  
ArrDelay:1258  
DepDelay:1247  
CarrierDelay:1247  
WeatherDelay:0  
NASDelay:11  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 198 From: SJC To: MSP Date: 9/9/2003  
ArrDelay:1288  
DepDelay:1272

CarrierDelay:1272  
WeatherDelay:0  
NASDelay:16  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 38 From: OGG To: LAX Date: 15/11/2003  
ArrDelay:1195  
DepDelay:1201  
CarrierDelay:1195  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 922 From: HNL To: DTW Date: 31/12/2003  
ArrDelay:1219  
DepDelay:1222  
CarrierDelay:1219  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 610 From: SJU To: JFK Date: 10/12/2003  
ArrDelay:1240  
DepDelay:1245  
CarrierDelay:1240  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 1463 From: BDL To: DFW Date: 29/12/2003  
ArrDelay:1185  
DepDelay:1191  
CarrierDelay:1180  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:5  
Flight Number: 2975 From: ORF To: DFW Date: 7/12/2003  
ArrDelay:1248  
DepDelay:1254  
CarrierDelay:1248  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 352 From: FLL To: JFK Date: 28/11/2003  
ArrDelay:236  
DepDelay:-1173  
CarrierDelay:0  
WeatherDelay:12  
NASDelay:-31  
SecurityDelay:0  
LateAircraftDelay:255  
Flight Number: 97 From: JFK To: DEN Date: 14/8/2003

ArrDelay:308  
DepDelay:-1112  
CarrierDelay:215  
WeatherDelay:0  
NASDelay:-20  
SecurityDelay:0  
LateAircraftDelay:113  
Flight Number: 128 From: JFK To: BTV Date: 28/11/2003  
ArrDelay:244  
DepDelay:-1184  
CarrierDelay:4  
WeatherDelay:0  
NASDelay:-12  
SecurityDelay:0  
LateAircraftDelay:252  
Flight Number: 42 From: MCO To: JFK Date: 28/11/2003  
ArrDelay:257  
DepDelay:-1154  
CarrierDelay:16  
WeatherDelay:0  
NASDelay:-29  
SecurityDelay:0  
LateAircraftDelay:270  
Flight Number: 70 From: PBI To: JFK Date: 15/8/2003  
ArrDelay:323  
DepDelay:-1111  
CarrierDelay:0  
WeatherDelay:0  
NASDelay:-6  
SecurityDelay:0  
LateAircraftDelay:329  
Flight Number: 24 From: TPA To: JFK Date: 28/11/2003  
ArrDelay:240  
DepDelay:-1165  
CarrierDelay:1  
WeatherDelay:0  
NASDelay:-35  
SecurityDelay:0  
LateAircraftDelay:274  
Flight Number: 30 From: FLL To: JFK Date: 5/12/2003  
ArrDelay:299  
DepDelay:-1130  
CarrierDelay:21  
WeatherDelay:0  
NASDelay:-11  
SecurityDelay:0  
LateAircraftDelay:289  
Flight Number: 33 From: ROC To: JFK Date: 5/12/2003  
ArrDelay:321  
DepDelay:-1150  
CarrierDelay:0  
WeatherDelay:0  
NASDelay:31  
SecurityDelay:0

LateAircraftDelay:290  
Flight Number: 837 From: CLT To: MSP Date: 27/8/2002  
ArrDelay:2137  
DepDelay:2119  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 1756 From: DFW To: TUS Date: 26/8/2003  
ArrDelay:1132  
DepDelay:1110  
CarrierDelay:0  
WeatherDelay:1110  
NASDelay:22  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 79 From: JFK To: MCO Date: 21/7/2003  
ArrDelay:222  
DepDelay:-1194  
CarrierDelay:19  
WeatherDelay:0  
NASDelay:-24  
SecurityDelay:0  
LateAircraftDelay:227  
Flight Number: 366 From: JFK To: BUF Date: 14/8/2003  
ArrDelay:241  
DepDelay:-1196  
CarrierDelay:73  
WeatherDelay:0  
NASDelay:-3  
SecurityDelay:0  
LateAircraftDelay:171  
Flight Number: 1224 From: FLL To: TPA Date: 6/12/2002  
ArrDelay:75  
DepDelay:-1370  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 233 From: DCA To: DTW Date: 30/8/2003  
ArrDelay:1340  
DepDelay:1336  
CarrierDelay:1336  
WeatherDelay:0  
NASDelay:4  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 672 From: SLC To: MSP Date: 27/10/2003  
ArrDelay:1304  
DepDelay:1301  
CarrierDelay:1301  
WeatherDelay:0

NASDelay:3  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 264 From: PHX To: DTW Date: 25/11/2003  
ArrDelay:1273  
DepDelay:1297  
CarrierDelay:1234  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:39  
Flight Number: 893 From: PBI To: DTW Date: 22/11/2003  
ArrDelay:1310  
DepDelay:1316  
CarrierDelay:1310  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 5 From: BUF To: JFK Date: 21/7/2003  
ArrDelay:382  
DepDelay:-1052  
CarrierDelay:344  
WeatherDelay:0  
NASDelay:-6  
SecurityDelay:0  
LateAircraftDelay:44  
Flight Number: 614 From: GEG To: MSP Date: 13/8/2003  
ArrDelay:1517  
DepDelay:1532  
CarrierDelay:1517  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 365 From: BUF To: JFK Date: 14/8/2003  
ArrDelay:422  
DepDelay:-1020  
CarrierDelay:420  
WeatherDelay:0  
NASDelay:2  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 89 From: JFK To: ONT Date: 14/8/2003  
ArrDelay:380  
DepDelay:-1028  
CarrierDelay:412  
WeatherDelay:0  
NASDelay:-32  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 509 From: JFK To: SJU Date: 14/8/2003  
ArrDelay:362  
DepDelay:-1055

CarrierDelay:385  
WeatherDelay:0  
NASDelay:-23  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 87 From: JFK To: SLC Date: 14/8/2003  
ArrDelay:336  
DepDelay:-1069  
CarrierDelay:371  
WeatherDelay:0  
NASDelay:-35  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 216 From: LGB To: JFK Date: 15/8/2003  
ArrDelay:403  
DepDelay:-1038  
CarrierDelay:402  
WeatherDelay:0  
NASDelay:1  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 50 From: FLL To: JFK Date: 14/8/2003  
ArrDelay:549  
DepDelay:-960  
CarrierDelay:480  
WeatherDelay:0  
NASDelay:69  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 36 From: JFK To: ROC Date: 14/8/2003  
ArrDelay:245  
DepDelay:-1172  
CarrierDelay:268  
WeatherDelay:0  
NASDelay:-23  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 10 From: JFK To: BUF Date: 15/8/2003  
ArrDelay:225  
DepDelay:-1197  
CarrierDelay:243  
WeatherDelay:0  
NASDelay:-18  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 377 From: BOS To: DTW Date: 10/12/2002  
ArrDelay:1946  
DepDelay:1952  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 62 From: RSW To: JFK Date: 14/8/2003

ArrDelay:823  
DepDelay:-655  
CarrierDelay:785  
WeatherDelay:0  
NASDelay:38  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 2292 From: PDX To: DFW Date: 4/10/2003  
ArrDelay:1385  
DepDelay:1379  
CarrierDelay:0  
WeatherDelay:0  
NASDelay:1385  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 504 From: SJU To: JFK Date: 25/8/2003  
ArrDelay:268  
DepDelay:-1171  
CarrierDelay:269  
WeatherDelay:0  
NASDelay:-1  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 1326 From: BUR To: DFW Date: 26/10/2003  
ArrDelay:1180  
DepDelay:1160  
CarrierDelay:0  
WeatherDelay:1160  
NASDelay:20  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 1325 From: SJU To: MSP Date: 20/12/2003  
ArrDelay:1584  
DepDelay:1582  
CarrierDelay:1582  
WeatherDelay:0  
NASDelay:2  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 43 From: JFK To: MCO Date: 5/12/2003  
ArrDelay:436  
DepDelay:-1049  
CarrierDelay:20  
WeatherDelay:371  
NASDelay:45  
SecurityDelay:0  
LateAircraftDelay:0

#### 14.4.2. VERSION MEMORIA/PROBABILÍSTICA (129 FALSOS POSITIVOS)

Flight Number: 1224 From: FLL To: TPA Date: 7/1/2003  
ArrDelay:30  
DepDelay:-1410  
CarrierDelay:-1  
WeatherDelay:-1

NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 391 From: ATL To: STL Date: 18/6/1998  
ArrDelay:-1233  
DepDelay:168  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 1519 From: DFW To: ELP Date: 25/2/2000  
ArrDelay:-1298  
DepDelay:25  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 1375 From: ORD To: SAN Date: 31/12/2000  
ArrDelay:-1188  
DepDelay:19  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 505 From: MIA To: STL Date: 27/2/2001  
ArrDelay:-1116  
DepDelay:340  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 1255 From: ATL To: RSW Date: 30/3/2000  
ArrDelay:-989  
DepDelay:-990  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 1718 From: EWR To: TPA Date: 27/6/2002  
ArrDelay:-987  
DepDelay:-981  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 1/1/2003  
ArrDelay:1427  
DepDelay:0



CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 33 From: ROC To: JFK Date: 10/2/2003  
ArrDelay:325  
DepDelay:-1100  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 10 From: JFK To: BUF Date: 23/2/2003  
ArrDelay:264  
DepDelay:-1160  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6149 From: LAX To: PSP Date: 1/1/2003  
ArrDelay:1420  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 36 From: JFK To: ROC Date: 23/2/2003  
ArrDelay:304  
DepDelay:-1124  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 365 From: BUF To: JFK Date: 21/7/2003  
ArrDelay:267  
DepDelay:-1180  
CarrierDelay:88  
WeatherDelay:0  
NASDelay:7  
SecurityDelay:0  
LateAircraftDelay:172  
Flight Number: 382 From: PBI To: JFK Date: 23/2/2003  
ArrDelay:351  
DepDelay:-1067  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 352 From: FLL To: JFK Date: 28/11/2003

ArrDelay:236  
DepDelay:-1173  
CarrierDelay:0  
WeatherDelay:12  
NASDelay:-31  
SecurityDelay:0  
LateAircraftDelay:255  
Flight Number: 392 From: RSW To: JFK Date: 21/3/2003  
ArrDelay:290  
DepDelay:-1132  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 128 From: JFK To: BTV Date: 28/11/2003  
ArrDelay:244  
DepDelay:-1184  
CarrierDelay:4  
WeatherDelay:0  
NASDelay:-12  
SecurityDelay:0  
LateAircraftDelay:252  
Flight Number: 3829 From: SLC To: PHX Date: 2/1/2003  
ArrDelay:1435  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 42 From: MCO To: JFK Date: 28/11/2003  
ArrDelay:257  
DepDelay:-1154  
CarrierDelay:16  
WeatherDelay:0  
NASDelay:-29  
SecurityDelay:0  
LateAircraftDelay:270  
Flight Number: 30 From: FLL To: JFK Date: 27/5/2003  
ArrDelay:379  
DepDelay:-1061  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 24 From: TPA To: JFK Date: 28/11/2003  
ArrDelay:240  
DepDelay:-1165  
CarrierDelay:1  
WeatherDelay:0  
NASDelay:-35  
SecurityDelay:0

LateAircraftDelay:274  
Flight Number: 98 From: DEN To: JFK Date: 14/8/2003  
ArrDelay:367  
DepDelay:-1114  
CarrierDelay:23  
WeatherDelay:0  
NASDelay:41  
SecurityDelay:0  
LateAircraftDelay:303  
Flight Number: 6149 From: LAX To: PSP Date: 2/1/2003  
ArrDelay:1427  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 97 From: JFK To: DEN Date: 14/8/2003  
ArrDelay:308  
DepDelay:-1112  
CarrierDelay:215  
WeatherDelay:0  
NASDelay:-20  
SecurityDelay:0  
LateAircraftDelay:113  
Flight Number: 3988 From: SLC To: TUS Date: 3/1/2003  
ArrDelay:1430  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 43 From: JFK To: MCO Date: 14/8/2003  
ArrDelay:312  
DepDelay:-1100  
CarrierDelay:99  
WeatherDelay:0  
NASDelay:-28  
SecurityDelay:0  
LateAircraftDelay:241  
Flight Number: 70 From: PBI To: JFK Date: 15/8/2003  
ArrDelay:323  
DepDelay:-1111  
CarrierDelay:0  
WeatherDelay:0  
NASDelay:-6  
SecurityDelay:0  
LateAircraftDelay:329  
Flight Number: 6149 From: LAX To: PSP Date: 3/1/2003  
ArrDelay:1428  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1

NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 36 From: JFK To: ROC Date: 28/11/2003  
ArrDelay:312  
DepDelay:-1106  
CarrierDelay:58  
WeatherDelay:0  
NASDelay:-22  
SecurityDelay:0  
LateAircraftDelay:276  
Flight Number: 30 From: FLL To: JFK Date: 5/12/2003  
ArrDelay:299  
DepDelay:-1130  
CarrierDelay:21  
WeatherDelay:0  
NASDelay:-11  
SecurityDelay:0  
LateAircraftDelay:289  
Flight Number: 3738 From: SLC To: TUS Date: 5/1/2003  
ArrDelay:1432  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 33 From: ROC To: JFK Date: 5/12/2003  
ArrDelay:321  
DepDelay:-1150  
CarrierDelay:0  
WeatherDelay:0  
NASDelay:31  
SecurityDelay:0  
LateAircraftDelay:290  
Flight Number: 837 From: CLT To: MSP Date: 27/8/2002  
ArrDelay:2137  
DepDelay:2119  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 5/1/2003  
ArrDelay:1428  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3578 From: DTW To: DFW Date: 31/8/2003  
ArrDelay:1228  
DepDelay:1208

CarrierDelay:1208  
WeatherDelay:0  
NASDelay:20  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 3829 From: SLC To: PHX Date: 6/1/2003  
ArrDelay:1427  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 7/1/2003  
ArrDelay:1435  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 1680 From: SLC To: STL Date: 17/8/2003  
ArrDelay:1258  
DepDelay:1247  
CarrierDelay:1247  
WeatherDelay:0  
NASDelay:11  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 3738 From: SLC To: TUS Date: 9/1/2003  
ArrDelay:1430  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 198 From: SJC To: MSP Date: 9/9/2003  
ArrDelay:1288  
DepDelay:1272  
CarrierDelay:1272  
WeatherDelay:0  
NASDelay:16  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 3738 From: SLC To: TUS Date: 11/1/2003  
ArrDelay:1431  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 12/1/2003

ArrDelay:1432  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 12/1/2003  
ArrDelay:1430  
DepDelay:4  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 922 From: HNL To: DTW Date: 31/12/2003  
ArrDelay:1219  
DepDelay:1222  
CarrierDelay:1219  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 3829 From: SLC To: PHX Date: 13/1/2003  
ArrDelay:1430  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 14/1/2003  
ArrDelay:1426  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 610 From: SJU To: JFK Date: 10/12/2003  
ArrDelay:1240  
DepDelay:1245  
CarrierDelay:1240  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 3829 From: SLC To: PHX Date: 14/1/2003  
ArrDelay:1423  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1

LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 16/1/2003  
ArrDelay:1429  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 16/1/2003  
ArrDelay:1432  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 2975 From: ORF To: DFW Date: 7/12/2003  
ArrDelay:1248  
DepDelay:1254  
CarrierDelay:1248  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 3829 From: SLC To: PHX Date: 17/1/2003  
ArrDelay:1435  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 19/1/2003  
ArrDelay:1423  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 20/1/2003  
ArrDelay:1428  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 21/1/2003  
ArrDelay:1421  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1

NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 21/1/2003  
ArrDelay:1418  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 22/1/2003  
ArrDelay:1427  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 22/1/2003  
ArrDelay:1434  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 23/1/2003  
ArrDelay:1426  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 24/1/2003  
ArrDelay:1422  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 24/1/2003  
ArrDelay:1431  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 25/1/2003  
ArrDelay:1426  
DepDelay:0



CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 26/1/2003  
ArrDelay:1428  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 28/1/2003  
ArrDelay:1427  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 28/1/2003  
ArrDelay:1430  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 1756 From: DFW To: TUS Date: 26/8/2003  
ArrDelay:1132  
DepDelay:1110  
CarrierDelay:0  
WeatherDelay:1110  
NASDelay:22  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 3738 From: SLC To: TUS Date: 29/1/2003  
ArrDelay:1424  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 29/1/2003  
ArrDelay:1424  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 79 From: JFK To: MCO Date: 21/7/2003

ArrDelay:222  
DepDelay:-1194  
CarrierDelay:19  
WeatherDelay:0  
NASDelay:-24  
SecurityDelay:0  
LateAircraftDelay:227  
Flight Number: 366 From: JFK To: BUF Date: 14/8/2003  
ArrDelay:241  
DepDelay:-1196  
CarrierDelay:73  
WeatherDelay:0  
NASDelay:-3  
SecurityDelay:0  
LateAircraftDelay:171  
Flight Number: 1224 From: FLL To: TPA Date: 6/12/2002  
ArrDelay:75  
DepDelay:-1370  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 30/1/2003  
ArrDelay:1426  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 5 From: BUF To: JFK Date: 21/7/2003  
ArrDelay:382  
DepDelay:-1052  
CarrierDelay:344  
WeatherDelay:0  
NASDelay:-6  
SecurityDelay:0  
LateAircraftDelay:44  
Flight Number: 365 From: BUF To: JFK Date: 14/8/2003  
ArrDelay:422  
DepDelay:-1020  
CarrierDelay:420  
WeatherDelay:0  
NASDelay:2  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 614 From: GEG To: MSP Date: 13/8/2003  
ArrDelay:1517  
DepDelay:1532  
CarrierDelay:1517  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0

LateAircraftDelay:0  
Flight Number: 89 From: JFK To: ONT Date: 14/8/2003  
ArrDelay:380  
DepDelay:-1028  
CarrierDelay:412  
WeatherDelay:0  
NASDelay:-32  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 233 From: DCA To: DTW Date: 30/8/2003  
ArrDelay:1340  
DepDelay:1336  
CarrierDelay:1336  
WeatherDelay:0  
NASDelay:4  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 3829 From: SLC To: PHX Date: 30/1/2003  
ArrDelay:1432  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 509 From: JFK To: SJU Date: 14/8/2003  
ArrDelay:362  
DepDelay:-1055  
CarrierDelay:385  
WeatherDelay:0  
NASDelay:-23  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 87 From: JFK To: SLC Date: 14/8/2003  
ArrDelay:336  
DepDelay:-1069  
CarrierDelay:371  
WeatherDelay:0  
NASDelay:-35  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 3738 From: SLC To: TUS Date: 31/1/2003  
ArrDelay:1428  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 216 From: LGB To: JFK Date: 15/8/2003  
ArrDelay:403  
DepDelay:-1038  
CarrierDelay:402  
WeatherDelay:0

NASDelay:1  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 3829 From: SLC To: PHX Date: 31/1/2003  
ArrDelay:1430  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6321 From: CVG To: SBN Date: 3/1/2003  
ArrDelay:1428  
DepDelay:-3  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 672 From: SLC To: MSP Date: 27/10/2003  
ArrDelay:1304  
DepDelay:1301  
CarrierDelay:1301  
WeatherDelay:0  
NASDelay:3  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 50 From: FLL To: JFK Date: 14/8/2003  
ArrDelay:549  
DepDelay:-960  
CarrierDelay:480  
WeatherDelay:0  
NASDelay:69  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 49 From: JFK To: FLL Date: 14/8/2003  
ArrDelay:434  
DepDelay:-981  
CarrierDelay:459  
WeatherDelay:0  
NASDelay:-25  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 6197 From: CVG To: SBN Date: 4/1/2003  
ArrDelay:1426  
DepDelay:-6  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 47 From: JFK To: FLL Date: 14/8/2003  
ArrDelay:523  
DepDelay:-875

CarrierDelay:565  
WeatherDelay:0  
NASDelay:-42  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 217 From: JFK To: LGB Date: 14/8/2003  
ArrDelay:417  
DepDelay:-968  
CarrierDelay:472  
WeatherDelay:0  
NASDelay:-55  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 36 From: JFK To: ROC Date: 14/8/2003  
ArrDelay:245  
DepDelay:-1172  
CarrierDelay:268  
WeatherDelay:0  
NASDelay:-23  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 264 From: PHX To: DTW Date: 25/11/2003  
ArrDelay:1273  
DepDelay:1297  
CarrierDelay:1234  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:39  
Flight Number: 6167 From: CVG To: GRR Date: 7/1/2003  
ArrDelay:1434  
DepDelay:-4  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 10 From: JFK To: BUF Date: 15/8/2003  
ArrDelay:225  
DepDelay:-1197  
CarrierDelay:243  
WeatherDelay:0  
NASDelay:-18  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 62 From: RSW To: JFK Date: 14/8/2003  
ArrDelay:823  
DepDelay:-655  
CarrierDelay:785  
WeatherDelay:0  
NASDelay:38  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 195 From: JFK To: LAS Date: 14/8/2003

ArrDelay:425  
DepDelay:-955  
CarrierDelay:485  
WeatherDelay:0  
NASDelay:-60  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 2292 From: PDX To: DFW Date: 4/10/2003  
ArrDelay:1385  
DepDelay:1379  
CarrierDelay:0  
WeatherDelay:0  
NASDelay:1385  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 377 From: BOS To: DTW Date: 10/12/2002  
ArrDelay:1946  
DepDelay:1952  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 893 From: PBI To: DTW Date: 22/11/2003  
ArrDelay:1310  
DepDelay:1316  
CarrierDelay:1310  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 6197 From: CVG To: SBN Date: 7/1/2003  
ArrDelay:1432  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 504 From: SJU To: JFK Date: 25/8/2003  
ArrDelay:268  
DepDelay:-1171  
CarrierDelay:269  
WeatherDelay:0  
NASDelay:-1  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 1325 From: SJU To: MSP Date: 20/12/2003  
ArrDelay:1584  
DepDelay:1582  
CarrierDelay:1582  
WeatherDelay:0  
NASDelay:2  
SecurityDelay:0

LateAircraftDelay:0  
Flight Number: 6203 From: CVG To: TOL Date: 8/1/2003  
ArrDelay:1430  
DepDelay:-10  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 1326 From: BUR To: DFW Date: 26/10/2003  
ArrDelay:1180  
DepDelay:1160  
CarrierDelay:0  
WeatherDelay:1160  
NASDelay:20  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 216 From: LGB To: JFK Date: 24/10/2003  
ArrDelay:855  
DepDelay:-572  
CarrierDelay:0  
WeatherDelay:0  
NASDelay:855  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 43 From: JFK To: MCO Date: 5/12/2003  
ArrDelay:436  
DepDelay:-1049  
CarrierDelay:20  
WeatherDelay:371  
NASDelay:45  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 6203 From: CVG To: TOL Date: 9/1/2003  
ArrDelay:1432  
DepDelay:-6  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 7357 From: IAD To: JAX Date: 26/11/2003  
ArrDelay:-812  
DepDelay:-810  
CarrierDelay:0  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 6197 From: CVG To: SBN Date: 10/1/2003  
ArrDelay:1430  
DepDelay:-10  
CarrierDelay:-1  
WeatherDelay:-1

NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 7426 From: ORD To: JAX Date: 26/11/2003  
ArrDelay:-754  
DepDelay:-739  
CarrierDelay:0  
WeatherDelay:0  
NASDelay:0  
SecurityDelay:0  
LateAircraftDelay:0  
Flight Number: 6179 From: CVG To: LAN Date: 11/1/2003  
ArrDelay:1426  
DepDelay:-10  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 12/1/2003  
ArrDelay:1429  
DepDelay:-10  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 12/1/2003  
ArrDelay:1431  
DepDelay:-7  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6179 From: CVG To: LAN Date: 13/1/2003  
ArrDelay:1423  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 13/1/2003  
ArrDelay:1432  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 14/1/2003  
ArrDelay:1433  
DepDelay:-7



CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6135 From: CVG To: CAE Date: 15/1/2003  
ArrDelay:1419  
DepDelay:-3  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 15/1/2003  
ArrDelay:1427  
DepDelay:-10  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 18/1/2003  
ArrDelay:1428  
DepDelay:-12  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 20/1/2003  
ArrDelay:1433  
DepDelay:-4  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 23/1/2003  
ArrDelay:1432  
DepDelay:-4  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 23/1/2003  
ArrDelay:1431  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6135 From: CVG To: CAE Date: 24/1/2003

ArrDelay:1420  
DepDelay:-6  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 24/1/2003  
ArrDelay:1433  
DepDelay:-10  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 25/1/2003  
ArrDelay:1430  
DepDelay:-7  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 25/1/2003  
ArrDelay:1430  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 29/1/2003  
ArrDelay:1429  
DepDelay:-10  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 30/1/2003  
ArrDelay:1431  
DepDelay:-3  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 30/1/2003  
ArrDelay:1422  
DepDelay:-7  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1

LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 2/2/2003  
ArrDelay:1425  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 2/2/2003  
ArrDelay:1435  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 3/2/2003  
ArrDelay:1431  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 3/2/2003  
ArrDelay:1434  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 5/2/2003  
ArrDelay:1432  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 5/2/2003  
ArrDelay:1431  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 9/2/2003  
ArrDelay:1423  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1

NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 9/2/2003  
ArrDelay:1435  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 10/2/2003  
ArrDelay:1431  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 11/2/2003  
ArrDelay:1422  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 11/2/2003  
ArrDelay:1431  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 14/2/2003  
ArrDelay:1434  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 15/2/2003  
ArrDelay:1423  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 16/2/2003  
ArrDelay:1425  
DepDelay:0

CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 16/2/2003  
ArrDelay:1431  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 18/2/2003  
ArrDelay:1429  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 20/2/2003  
ArrDelay:1432  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3738 From: SLC To: TUS Date: 22/2/2003  
ArrDelay:1431  
DepDelay:5  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 26/2/2003  
ArrDelay:1435  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 27/2/2003  
ArrDelay:1432  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 3829 From: SLC To: PHX Date: 28/2/2003

ArrDelay:1432  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 1/2/2003  
ArrDelay:1429  
DepDelay:-4  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 2/2/2003  
ArrDelay:1429  
DepDelay:-4  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 2/2/2003  
ArrDelay:1433  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 3/2/2003  
ArrDelay:1416  
DepDelay:-19  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6179 From: CVG To: LAN Date: 5/2/2003  
ArrDelay:1426  
DepDelay:-6  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 5/2/2003  
ArrDelay:1428  
DepDelay:-10  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1

LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 8/2/2003  
ArrDelay:1429  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 12/2/2003  
ArrDelay:1431  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 12/2/2003  
ArrDelay:1431  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 13/2/2003  
ArrDelay:1430  
DepDelay:-6  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 19/2/2003  
ArrDelay:1421  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 20/2/2003  
ArrDelay:1431  
DepDelay:-6  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 20/2/2003  
ArrDelay:1428  
DepDelay:-4  
CarrierDelay:-1  
WeatherDelay:-1

NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 26/2/2003  
ArrDelay:1431  
DepDelay:-10  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 1/3/2003  
ArrDelay:1429  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 3/3/2003  
ArrDelay:1419  
DepDelay:-4  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 7/3/2003  
ArrDelay:1428  
DepDelay:-6  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 10/3/2003  
ArrDelay:1423  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 11/3/2003  
ArrDelay:1426  
DepDelay:-6  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6179 From: CVG To: LAN Date: 11/3/2003  
ArrDelay:1426  
DepDelay:-9



CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 12/3/2003  
ArrDelay:1428  
DepDelay:-3  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6179 From: CVG To: LAN Date: 15/3/2003  
ArrDelay:1425  
DepDelay:-6  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 15/3/2003  
ArrDelay:1414  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6194 From: BDL To: RDU Date: 16/3/2003  
ArrDelay:1438  
DepDelay:19  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 16/3/2003  
ArrDelay:1428  
DepDelay:5  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 17/3/2003  
ArrDelay:1426  
DepDelay:-7  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 17/3/2003

ArrDelay:1422  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 18/3/2003  
ArrDelay:1428  
DepDelay:-4  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 19/3/2003  
ArrDelay:1419  
DepDelay:-7  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 22/3/2003  
ArrDelay:1414  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 23/3/2003  
ArrDelay:1425  
DepDelay:-6  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 23/3/2003  
ArrDelay:1428  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 25/3/2003  
ArrDelay:1428  
DepDelay:-9  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1

LateAircraftDelay:-1  
Flight Number: 6167 From: CVG To: GRR Date: 27/3/2003  
ArrDelay:1420  
DepDelay:-4  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6197 From: CVG To: SBN Date: 29/3/2003  
ArrDelay:1425  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1  
Flight Number: 6203 From: CVG To: TOL Date: 29/3/2003  
ArrDelay:1425  
DepDelay:0  
CarrierDelay:-1  
WeatherDelay:-1  
NASDelay:-1  
SecurityDelay:-1  
LateAircraftDelay:-1