Maestría en Tecnología Informática y de Comunicaciones

Cohorte 2018 – 2019

Título del Trabajo Final:

"Desarrollo de una aplicación para la predicción de síntesis de geles moleculares"

Autor: *Alexis Emmanuel Federico Davila*

Director del Trabajo Final: ***Dr. Tomás Enrique Tecce.***

Institución a la que pertenece: UADE

Fecha de entrega: *29/11/2019*

ABSTRACT

This work is framed in the *Data Science into Food Science* research project, which is developed in the research field of the Universidad Argentina de la Empresa. The focus of this project is on studying the behavior of molecular gels and the application from new data management technologies to the field of food engineering.

Molecular gels are materials formed by the combination of a low molecular weight organic gel and a solvent; Due to the particular characteristics of them they are potentially useful for the food industry. Despite the rapid growth of the studies dedicated to molecular gels, the identification of good quality molecular gelants, that is to say, those capables of generating stable gels; it is purely empirical and many of them are discovered coincidentally, making it a slow and expensive process.

In this sense, the Data Science into Food Science project has Excel spreadsheets with information on a total of 22 molecular gelants and 115 solvents. In these spreadsheets, the result of the gelling process among some of them was methodically and manually recorded, which were obtained from a combination of literature and laboratory experiments.

The aim of this work seeks to contribute to the development of molecular gels through the application of Data Science methodologies and machine learning techniques to build a web application, which will serve as a tool for researchers and food technicians to make predictions of the result of the combination of different gelants and solvents, as well as identifying the most significant data or variables in the gelling process.

Índice

Índice de figuras.

Índice de tablas.

1. INTRODUCCIÓN

El presente proyecto se enmarca dentro del proyecto de investigación y desarrollo titulado *Data Science into Food Science*, que se desarrolla en el ámbito de investigación de la Universidad Argentina de la Empresa. El foco de dicho proyecto es el estudio de la aplicación de nuevas tecnologías y metodologías de manejo de datos al campo de la ciencia de los alimentos en general, y en particular al estudio del comportamiento de las sustancias denominadas **geles moleculares**.

Un **gel** es una sustancia con propiedades intermedias entre el estado sólido y el líquido. Está compuesto por dos fases: una sólida, que le imparte la estructura y soporte al gel, y otra líquida que queda atrapada en la estructura tridimensional dada por la fase sólida. Es por esto que, aunque los geles muestran propiedades propias de un sólido como el tener forma, resistencia ante ciertos esfuerzos, también tienen una importante proporción de fase líquida. En particular, los geles moleculares son aquellos cuya fase sólida consiste en compuestos orgánicos de bajo peso molecular, comúnmente llamados materiales blandos-inteligentes. Una característica relevante de estos geles es que son capaces de responder reversiblemente a estímulos externos tales como temperatura, energía lumínica, cambios de pH, u ondas ultrasónicas, entre otros. Los geles moleculares tienen un amplio rango de aplicación, incluyendo la producción de alimentos. Su importancia es especialmente grande ya que la demanda de productos bajos en grasa ha potenciado el desarrollo de alimentos donde esta se sustituye parcialmente por sistemas gelificados en base acuosa.

Actualmente, el proyecto de investigación antes mencionado cuenta con varias bases de datos de diferentes moléculas de bajo peso molecular, capaces de formar geles (denominadas simplemente **gelantes**). Para cada gelante se registraron los resultados obtenidos en laboratorio, de manera manual, de distintas propiedades fisicoquímicas. En cada experimento se combina un gelante con un solvente, siendo el resultado de la experiencia uno de tres casos: en primer lugar, puede efectivamente obtenerse un gel,

que es el resultado de interés. También puede suceder que el gelante no forme estructura dentro del solvente y se deposite en el fondo (un **precipitado),** o que la mezcla resultante tenga solamente propiedades de líquido, obteniéndose una **solución**.

A pesar de que las bases de datos de gelantes presentan un alto nivel de detalle se trata de un conjunto de datos muy heterogéneo, lo cual dificulta el análisis e interpretación de la información de manera conjunta. Esto motiva el desarrollo de una herramienta que, aplicando las técnicas más recientes de procesamiento y análisis de datos, permita predecir el resultado del experimento e identificar aquellas características de la combinación de solventes y/o gelantes que resulten efectivamente en el proceso de formación de un gel. Dicha herramienta resultaría de suma utilidad para investigadores y técnicos en el área de Ciencia y Tecnología de los Alimentos, para lograr que la identificación de nuevos gelantes se obtenga a partir de predicciones basadas en datos fisicoquímicos y no se deba únicamente a descubrimientos fortuitos.

1.1. DESCRIPCIÓN DEL PROBLEMA

Los geles moleculares de bajo peso molecular son potencialmente útiles para la industria alimentaria. Por ejemplo, permiten minimizar la migración de aceites en alimentos multicomponentes, reduciendo la necesidad de ácidos grasos saturados o trans. También se han propuesto para mejorar la estabilidad de emulsiones alimenticias y controlar la liberación de productos farmacéuticos y nutracéuticos en el torrente sanguíneo. Sin embargo, a pesar del rápido crecimiento de la literatura dedicada a los geles moleculares en la última década, la determinación de la gelificación sigue siendo una tarea empírica, donde la mayoría de los nuevos gelificadores se descubren por casualidad [María Corradini y otros, 2016]. Esto implica tener que llevar a cabo una extensa cantidad de pruebas de laboratorio, combinando diversos tipos de gelantes con solventes varios, lo que es un proceso laborioso, lento y costoso. El interés por hacer más eficiente todo este proceso motiva la aplicación de los métodos de la disciplina denominada Ciencia de Datos al estudio de este problema.

La Ciencia de Datos (*Data Science*) puede definirse como un proceso que, partiendo de un conjunto de datos y una serie de hipótesis sobre el área de aplicación, emplea métodos estadísticos y numéricos para obtener conocimiento y generar modelos predictivos cuyo objetivo final es la toma de decisiones [Zumel y Mount, 2014]. La Ciencia de Datos posibilita la extracción de conocimiento de los datos mediante el uso de técnicas numéricas robustas cuya validación estadística puede ser cuantificada. El actual auge de esta disciplina se sustenta en las nuevas tecnologías informáticas que permiten la obtención, procesamiento y almacenamiento de enormes volúmenes de datos. La aplicación de las metodologías de la Ciencia de Datos ha generado novedosos productos y servicios, incluyendo avances en disciplinas como biología y química computacional [Beck y otros, 2016].

La adopción de las estrategias multidisciplinarias de Ciencia de Datos en un área de aplicación determinada depende del desarrollo de soluciones analíticas que permitan

tanto obtener y compilar información como la aplicación de técnicas de aprendizaje automático (*machine learning*) para explicar relaciones presentes en conjuntos de datos complejos [Corradini y otros, 2016]. En este sentido, el presente trabajo busca contribuir al desarrollo de geles moleculares mediante la construcción de una aplicación web de Ciencia de Datos que permita a investigadores y técnicos en alimentación colectar, analizar y visualizar datos sobre solventes y gelantes. El componente de machine learning dentro de la aplicación permitirá realizar predicciones del resultado de la combinación de un dado gelante con distintos solventes, como así también la identificación de aquellos datos o predictores más significativos para la predicción del proceso de gelificación.

1.2. OBJETIVO.

Desarrollar una aplicación que permita, mediante la utilización de tecnologías de Ciencia de Datos, predecir la gelificación molecular de compuestos orgánicos de bajo peso molecular para el uso en Tecnología e Industria de Alimentos, en el marco del proyecto *Data Science into Food Science*, durante el año 2019.

1.3. ALCANCES.

- Desarrollar una aplicación Web, accesible desde computadoras personales o dispositivos móviles mediante navegadores web actuales.
- Predecir el resultado de la combinación de un gelante con un solvente (GEL O NO GEL).
- Mostrar la correlación entre las distintas propiedades de los solventes con el resultado del proceso de gelificación.
- No se contempla el desarrollo de una App Mobile (aplicaciones que se ejecutan de manera nativa en sistemas operativos móviles).
- No se contempla el desarrollo de una interfaz entre la aplicación web y el MVP (Minimum Viable Product) de una solución Big Data, propuesta por Verónica Cuello en su trabajo “Arquitectura Big Data para el análisis de organogelantes”, el cual también forma parte del proyecto *Data Sciende into Food Science* y cuyo objetivo es la automatización del proceso de carga, la homogeneización y disponibilizacion de los datos de experimentos con organogelantes. No se contempla la consulta de datos desde esta solución debido a que al momento de llevar a cabo este trabajo final la misma se encontraba en fase de ser mejorada y puesta en producción como parte de otro trabajo de investigación.

1.4. MARCO METODOLÓGICO

El enfoque metodológico utilizado para este trabajo es mixto, dado que se analizarán distintas tecnologías del ámbito de Data Science, utilizando una fuente de datos de naturaleza cualitativa y cuantitativa y proponiendo asimismo una implementación práctica dentro de otro proyecto de investigación.

Los métodos mixtos representan un conjunto de procesos sistemáticos, empíricos y críticos de investigación e implican la recolección y análisis de datos cuantitativos y cualitativos, así como su integración y discusión conjunta, para realizar inferencias producto de toda la información recabada y lograr un mayor entendimiento del fenómeno estudiado

El tipo de diseño es descriptivo, ya que se propone la construcción de una solución a una problemática determinada y dicho proceso de construcción se describe en forma detallada. Con los estudios descriptivos se busca especificar las propiedades, las características y los perfiles de personas, grupos, comunidades, procesos, objetos o cualquier otro fenómeno sometido a análisis [Hernandez Sampieri, y otros, 2010].

La fuente primaria de datos para el desarrollo de la aplicación en este trabajo proviene de bases de datos recopiladas por los investigadores del proyecto *Data Science into Food Science*. Dichas bases de datos almacenan los resultados de las experiencias realizadas en laboratorio en forma manual, y a la fecha consisten principalmente en colecciones de planillas de cálculo en formato Microsoft Excel. Estos datos son de naturaleza heterogénea: proceden de distintos orígenes, y pueden haber sido calculados de diversas formas (por una persona, un instrumento de laboratorio, un algoritmo de terceras personas). Cuando los datos son producidos manualmente es muy frecuente que se introduzcan errores en el proceso de recolección de datos. Es posible que los operadores humanos no sepan usar apropiadamente el instrumental, tengan apuros o cometan descuidos, malinterpreten instrucciones, o no sigan ciertos protocolos preestablecidos [Anderson, 2015].

En cada una de las planillas antes mencionadas se listan los solventes disponibles, el valor de las propiedades fisicoquímicas de estos solventes, y el resultado de combinar un determinado gelante con cada uno de ellos. Si bien la utilización de planillas de cálculos resulta útil para estructurar y almacenar información, especialmente para aquellas personas sin conocimientos especializados, las planillas presentan la desventaja de resultar en estructuras de datos heterogéneas que dificultan realizar un análisis integral entre las distintas bases para encontrar relaciones o patrones entre los datos contenidos en ellas.

2. MARCO CONCEPTUAL Y ESTADO DEL ARTE

A continuación se presenta una breve reseña de conceptos referentes a geles moleculares y Ciencia de Datos, necesarios para una mejor comprensión del presente trabajo.

2.1. GELES

Los geles son una clase de materiales compuestos que están presentes en nuestra vida cotidiana con amplias y variadas aplicaciones como ser la industria alimentaria, medicina, cosmética, o biomateriales, entre otras. Como se indicó en la introducción, se trata de sustancias con propiedades intermedias entre el estado sólido y el líquido, compuestas por la combinación de una fase sólida que le imparte estructura y soporte y otra líquida que queda atrapada en la estructura dada por la fase sólida. Debido principalmente a la diversidad de mecanismos por los cuales se puede establecer una estructura de gel, es mucho más fácil definirlos desde un punto de vista cotidiano que desde una perspectiva estrictamente química.

Mediante la introducción de la Teoría de geles en 1861, el químico Thomas Graham los definió como **sistemas coloidales**: estos sistemas son conformados por dos o más fases. Normalmente una de ellas es fluida, y la otra dispersa en forma de partículas muy pequeñas, entre 10^{-9} y 10^{-5} m de diámetro [Chang 2007]. Más tarde, Paul John Flory proporciona una definición más completa para vincular las propiedades macroscópicas y microscópicas, indicando que una sustancia es un gel si cumple con los dos requisitos siguientes:

1. Poseer una estructura microscópica continua, con dimensiones macroscópicas, que sea permanente en la escala de tiempo de un experimento analítico.
2. Presentar propiedades de sólido en su comportamiento reológico a pesar de ser un líquido el componente mayoritario.

Los geles pueden clasificarse de acuerdo con distintos criterios (ver Figura 1). Según su naturaleza, pueden diferenciarse en geles **naturales** o **artificiales** (sintéticos). Dependiendo de su constitución, se pueden clasificar en geles **macromoleculares** (de tipo polimérico), que pueden formarse a partir del entrecruzamiento químico o interacciones físicas no covalentes, o en geles **supramoleculares**, que en cambio se forman por el autoagregado de pequeñas moléculas de gelante mediante interacciones no covalentes. Por otra parte, si el solvente es un químico orgánico se denominan **organogeles**, mientras que si el solvente es agua se los llama **hidrogeles** [Edelsztein, 2010].

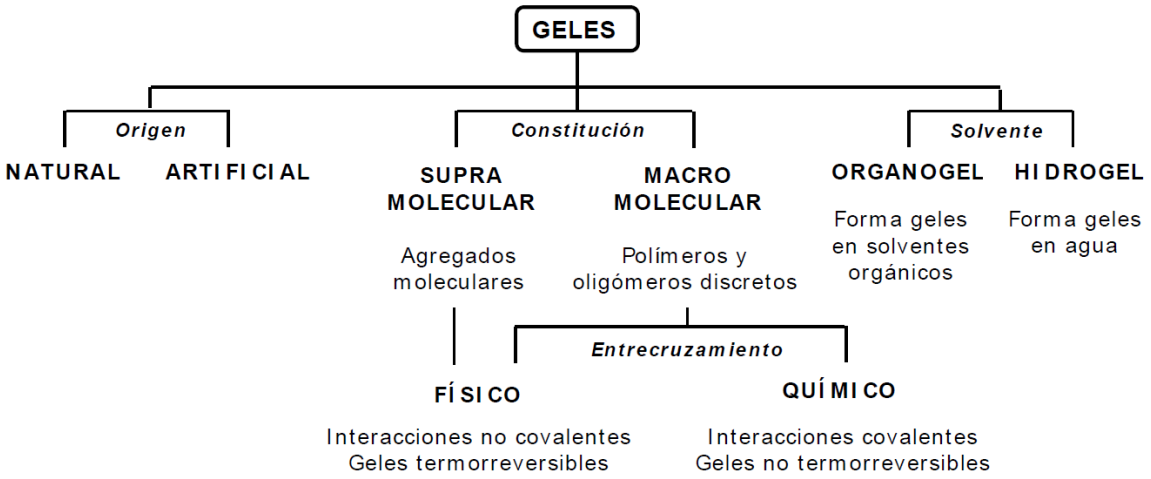


Figura 1 - Clasificación de geles.

2.2. GELES MOLECULARES.

Los geles moleculares están compuestos por gelificadores de bajo peso molecular (LMOGs, por sus siglas en inglés: *Low Molecular mass Organic Gelators*), menor a 3000

Dalton[1]. Las moléculas se autoensamblan a través de enlaces específicos no covalentes como enlaces de Hidrogeno, acoplamiento π - π, interacciones solvofóbicas y de Van der Waals, entre otras, forman estructuras alargadas y dan como resultado la formación de redes tridimensionales (SAFIN, del inglés: *Self-Assembled Fibrillar Networks*).

La formación de geles moleculares involucra varios pasos (ver Figura 2). El primero consiste en la disolución de los LMOGs en condiciones específicas, como por ejemplo temperaturas elevadas, para obtener una solución. Al enfriarse dicha solución se forman los agregados de estas moléculas LMOG, dando origen a un crecimiento preferentemente unidimensional (1D) formando fibras que pueden ser tubos, hebras, cintas u otras. Finalmente, las zonas de unión entre estas fibras 1D actúan como pegamento, generando redes tridimensionales (3D) que impregnan todo el sistema y atrapan el componente líquido en ellas a través de fuerzas capilares y tensión superficial [Corradini y otros, 2015].

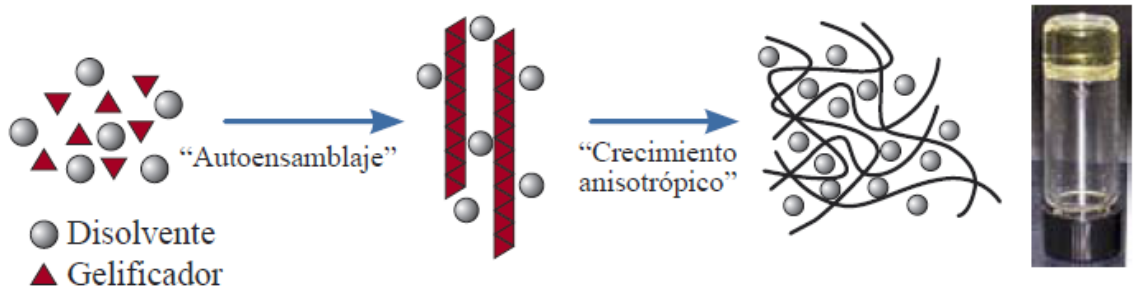


Figura 2 - Proceso de gelificación.

2.2.1. PREDICCIÓN DE LA GELIFICACIÓN.

La gran cantidad de LMOGs y la diversidad de interacciones que impulsan el proceso de autoensamblaje dificultan el estudio detallado de la formación de geles y el desarrollo de

[1] Dalton: unidad de masa estándar de masa equivalente a la doceava parte de un átomo de carbono 12.

modelos para predecir de manera precisa y universal si una dada combinación de gelante y solvente producirá un gel, un precipitado o una solución.

Las propiedades de los solventes, tanto físicas como solvatocrómicas[2] y termodinámicas, afectan las interacciones no covalentes entre gelante-gelante y solvente-gelante que impulsan el proceso de autoensamblaje. Estos parámetros se encuentran disponibles en la literatura o se pueden obtener mediante experimentos en el laboratorio. Una única propiedad aislada de solvente resulta ineficaz para predecir la gelificación molecular; sin embargo, cuando se toman propiedades multitérminos, ya sean solvatocrómicas o termodinámicas, se observa una mejora en la capacidad de predicción [María Corradini y Michael Rogers, 2016].

La disponibilidad de una herramienta predictiva, capaz de considerar múltiples variables, ayudaría a los investigadores a reducir tiempos y costos e identificar los factores más relevantes en el proceso de la gelificación molecular, permitiendo de esta manera que el descubrimiento de nuevos gelantes no se deba únicamente a un hecho fortuito. Por ejemplo, los parámetros de solubilidad de Hansen (*Hansen Solubility Parameters*, HSP) se consideran entre los predictores más eficaces y utilizados para la determinación de gelificación [Corradini y Rogers, 2016]. Los HSP clasifican las interacciones entre moléculas en tres tipos: **enlaces de hidrógeno** (dh), **dispersivas** (dd) y **polares** (dp). Estos parámetros tienen la capacidad de agrupar solventes que conducen ya sea a la solubilización, precipitación o gelificación de gelantes moleculares. A continuación, se listan algunos de los softwares existentes para el cálculo de dichos parámetros.

- HSPiP: desarrollado por el propio Charles Hansen junto con Hiroshi Yamamoto, permite la determinación numérica de esferas de solución, precipitado o gel, proporcionando, además, visualizaciones 2D y 3D de las regiones encerradas por las esferas. Para el cálculo de las esferas utiliza un algoritmo de minimización junto

[2] Se denomina solvatocromismo a la influencia que ejerce un solvente en el espectro de absorción UV/Vis/IR de una molécula.

con una función de deseabilidad, que reduce la probabilidad de clasificaciones erróneas [Hansen CM y Yamamoto H, 2013].

- Software basado en Excel, desarrollado por Bonnet, Suissa, Raynal y Bouteiller. Este utiliza el algoritmo de optimización y una rutina de minimización para calcular las esferas de Hansen. La aplicación solo muestra las esferas en 2D y la distancia al centro de la esfera, lo que reduce la utilidad de este software a la hora de identificar buenos solventes [Bonnet J y otros, 2014].

- Software basado en Mathematica 9, desarrollado por Lan, permite calcular las esferas correspondientes a los resultados de solución, gel y precipitado utilizando un procedimiento de optimización. Este enfoque permite obtener la ubicación del centro de la esfera mientras se calcula el radio mínimo posible [Lan Yaqi y otros, 2014].

A pesar de existir los programas arriba indicados para el cálculo de los HSP, no se encontraron en la bibliografía consultada iniciativas para la construcción de modelos predictivos, aplicados al estudio del comportamiento de geles moleculares, en donde se consideren otras propiedades de gelantes y solventes.

2.3. DATA SCIENCE.

Recientes avances en tecnologías de la información han hecho posible recopilar, almacenar y procesar conjuntos de datos masivos, a menudo altamente complejos. El objetivo principal de analizar estos datos masivos es la búsqueda de patrones e información valiosa, que se pueden utilizar para mejorar la toma de decisiones y aumentar las posibilidades de éxito, o simplemente la supervivencia, de muchas organizaciones [Turban, 2007]. Como se mencionó antes, la Ciencia de Datos se trata de

la gestión de un proceso que puede transformar hipótesis y datos en predicciones y percepciones accionables [Zumel y Mount, 2014]. Este proceso es multidisciplinario, e involucra distintos perfiles y tecnologías desde el manejo de datos (arquitectura de la información) pasando por la comprensión de negocio y la aplicación de machine learning hasta la obtención de un **producto de datos**, que es aquella pieza de software, visualización o reporte que permite el aprovechamiento del conocimiento obtenido. La Figura 3 muestra un esquema simplificado de los pasos de dicho proceso.

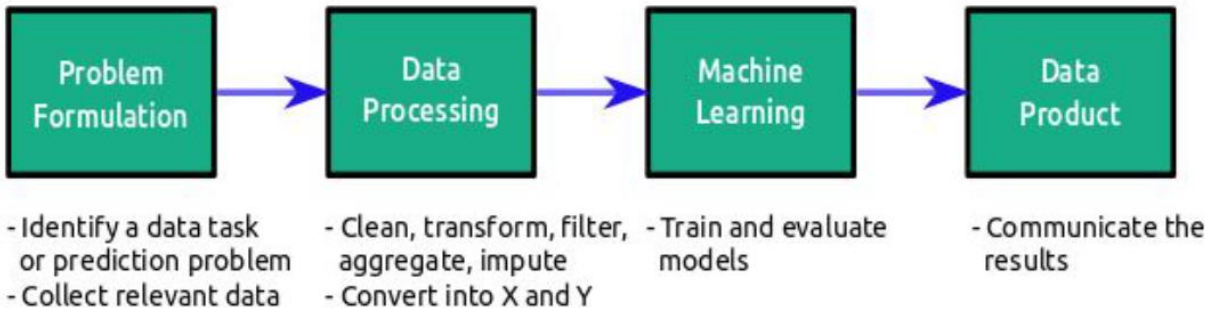


Figura 3 – Proceso simplificado de Ciencia de Datos.

Las áreas centrales de la Ciencia de Datos son el manejo de datos que consiste en métodos para organizar, clasificar y procesar gran cantidad de datos complejos, el aprendizaje estadístico y automático (machine learning) basado en el desarrollo de modelos predictivos que permiten explicar relaciones presentes en datos complejos, y la visualización para facilitar la toma de decisiones y expresar los resultados de una manera accesible [María Corradini y otros, 2017].

2.3.1. MACHINE LEARNING.

El aprendizaje automático, más comúnmente llamado por su nombre en inglés **machine learning**, fue definido por el científico de la computación Arthur Samuel en 1959 como el campo de estudio que brinda a las computadoras la capacidad de aprender la forma de resolver un problema sin ser explícitamente programadas con reglas para obtener dicha

solución. El enfoque fundamental del machine learning, que se basa en aprendizaje a partir de un conjunto de datos, lo diferencia de otros métodos que parten de primeros principios o conjuntos de reglas.

Por ejemplo, para crear un software que sea capaz de pilotar un helicóptero, uno podría escribir un programa que considere instrucciones detalladas sobre cómo encender los motores, elevarse y descender, esquivar edificios o pájaros, ajustarse a distintas condiciones de viento, etc. No es difícil darse cuenta de que un programa así, ante las muchísimas situaciones que podrían suceder durante un vuelo real, alcanzaría rápidamente una enorme complejidad que lo haría muy difícil de mantener por un programador humano que tenga que revisar errores o incluir explícitamente nuevas situaciones. En cambio, en un enfoque basado en machine learning lo que se hace no es dar instrucciones detalladas sino especificar unas condiciones de éxito y un conjunto de datos. Los algoritmos de machine learning exploran estos datos, llamados conjunto de entrenamiento, y determinan el mejor comportamiento posible para lograr los objetivos. Esto se hace minimizando una función de costo, que por ejemplo podría ser la cantidad de errores cometidos.

En machine learning, los problemas que se encaran se suelen dividir en tres grandes clases [Zumel y Mount, 2014]:

a) **Aprendizaje supervisado.** En este tipo de problemas, se busca que el algoritmo aprenda a predecir un valor numérico o una categoría, a partir de datos que contienen una o más variables predictoras (o simplemente **predictores**). El valor o categoría a predecir se denomina **variable objetivo**. En los casos de aprendizaje supervisado, cada punto de datos contiene también la variable objetivo que indica el valor o categoría real que corresponde a ese punto.

b) **Aprendizaje no supervisado.** En este tipo de problemas los datos no contienen una variable objetivo, y lo que se busca es encontrar patrones o características

presentes en los datos. El ejemplo más típico de este tipo de problemas es el clustering para encontrar conjuntos de puntos de datos similares entre sí.

c) **Aprendizaje por refuerzo.** Más que un tipo de problemas se refiere a una técnica en machine learning que involucra entrenar una inteligencia artificial (un “agente”) a través de la repetición de acciones en un “entorno” que simula las condiciones del problema. El agente aprende a partir de recibir recompensas por sus acciones.

El problema del que trata este trabajo constituye un ejemplo de aprendizaje supervisado, ya que los datos de experimentos disponibles indican si se logró o no formar un gel molecular bajo las condiciones indicadas. En consecuencia, en el resto del trabajo no se considerarán los otros tipos de problemas ni algoritmos asociados a ellos.

Entre los lenguajes de programación más utilizados para ciencia de datos se encuentran R y Python. Ambos son lenguajes de programación de alto nivel, de código abierto y libres. El lenguaje de programación R está orientado a estadísticas y es preferido por matemáticos y estadísticos, mientras que Python es un lenguaje potente y de propósito general caracterizado por su simplicidad.

2.4. ANTECEDENTES ACADÉMICOS.

En el área de la Tecnología e Ingeniería de los Alimentos (TIA) la utilización de los métodos de Ciencia de Datos para manipular y analizar bases de datos complejas no está muy difundida, existiendo sólo aplicaciones para casos puntuales que se mencionan a continuación [Corradini María y Otros, 2017].

CLASIFICACIÓN DE ACEITES VEGETALES MEDIANTE REDES NEURONALES.

La Universidad Estatal del Sudoeste de Bahía, Brasil, desarrolló una metodología en la cual, mediante el uso de los parámetros más relevantes del espectro de la fluorescencia de las sustancias como datos de entrada de una red neuronal artificial, es posible diferenciar aceites vegetales de canola, soja, maíz y girasol. Esta es una tecnología prometedora desde el punto de vista del control de calidad de los aceites vegetales, y puede considerarse como un punto de partida para diseñar futuros estudios centrados en la identificación de mezclas reales de aceites de diferentes cultivos [da Silva y otros, 2015].

HUELLA DIGITAL FITOQUÍMICA Y QUIMIOMETRÍA PARA EL RECONOCIMIENTO DE PATRONES DE PREPARACIÓN DE ALIMENTOS NATURALES

Entre las principales técnicas de huellas digitales analíticas para abordar la identidad y la calidad de productos botánicos se encuentran el análisis cromatógráfico, el análisis por cromatógrafo líquido de alta resolución (HPLC) [Donno, 2013], cromatografía de gases (GC) [Pan, 2011], cromatografía liquida de ultra rendimiento (UPLC) [Dan et al. 2009] y electroforesis capilar. Sin embargo, en algunos casos la información limitada proporcionada por la huella digital convencional puede no ser suficiente para revelar de manera integral las características de calidad de algunos productos herbales extremadamente complejos [Peng y otros. 2011]. Por otro lado, con el fin de eliminar o reducir las fuentes de variaciones no deseadas debido a diferentes variables o respuestas instrumentales, la huella digital analítica debe combinarse con un análisis multivariado [Gad y otros, 2013]. El análisis de componentes principales (*principal component analysis*, PCA), una técnica de aprendizaje no supervisado para la exploración de datos, se aplicó para racionalizar la información analítica de las preparaciones a base de hierbas consideradas. Con esto se mostró que las huellas digitales cromatográficas muestran potencial para determinar la identidad, autenticidad y consistencia de lotes de hierbas medicinales [Donno y otros, 2016].

HUELLA DIGITAL DE FLUORESCENCIA COMO UNA EVALUACIÓN INSTRUMENTAL DE LA CALIDAD SENSORIAL DE JUGO DE TOMATE.

Una huella digital de fluorescencia es una serie de espectros de fluorescencia en varias longitudes de onda de excitación, y se ha utilizado para caracterizar diversas calidades alimentarias con la ayuda de métodos quimiométricos [Christensen J y otros, 2006]. El análisis sensorial es un estándar importante para la evaluación de productos alimenticios; sin embargo, es un proceso que requiere personal capacitado e insume tiempo. Mediante este estudio, se demostró que usando PCA, regresión de mínimos cuadrados parciales (PLS) y la Huella digital de Fluorescencia como predictor es posible estimar características sensoriales generales como el aroma y gusto de los jugos de tomates [Trivittayasil V y otros, 2016].

3. DESARROLLO DEL TRABAJO

En este capítulo se detallas las actividades abordadas para el desarrollo de la aplicación web, incluyendo aquellas destinadas a la obtención y transformación de los datos, construcción de un modelo de predicción y selección de infraestructura de implementación.

3.1. FUENTE DE DATOS.

En este trabajo se toman como fuente de datos las planillas Excel con datos fisicoquímicos y resultados del proceso de formación de geles recopiladas por el proyecto *Data Science into Food Science*. Estas planillas contienen información sobre un total de 22 gelantes moleculares y 115 solventes, proveniente de una combinación entre datos de literatura y resultados de pruebas de laboratorio. Los datos registran propiedades físicas, solvatocrómicas y térmicas de los solventes relacionadas con el proceso de gelificación. Un problema de este conjunto de datos es que no se dispone de resultados del proceso de gelificación para todos los posibles pares gelante-solvente [Maria G Corradini, 2016].

Los datos arriba mencionados carecen de información sobre la estructura molecular de los gelantes, la cual tiene influencia sobre la capacidad de formar un gel. Incluir esta información no es trivial; una posibilidad de hacerlo sería mediante la notación SMILES (Simplified Molecular-Input Line-Entry System), que es una notación para describir la estructura de una especie química mediante cadenas cortas de caracteres ASCII [Weininger, 1988].

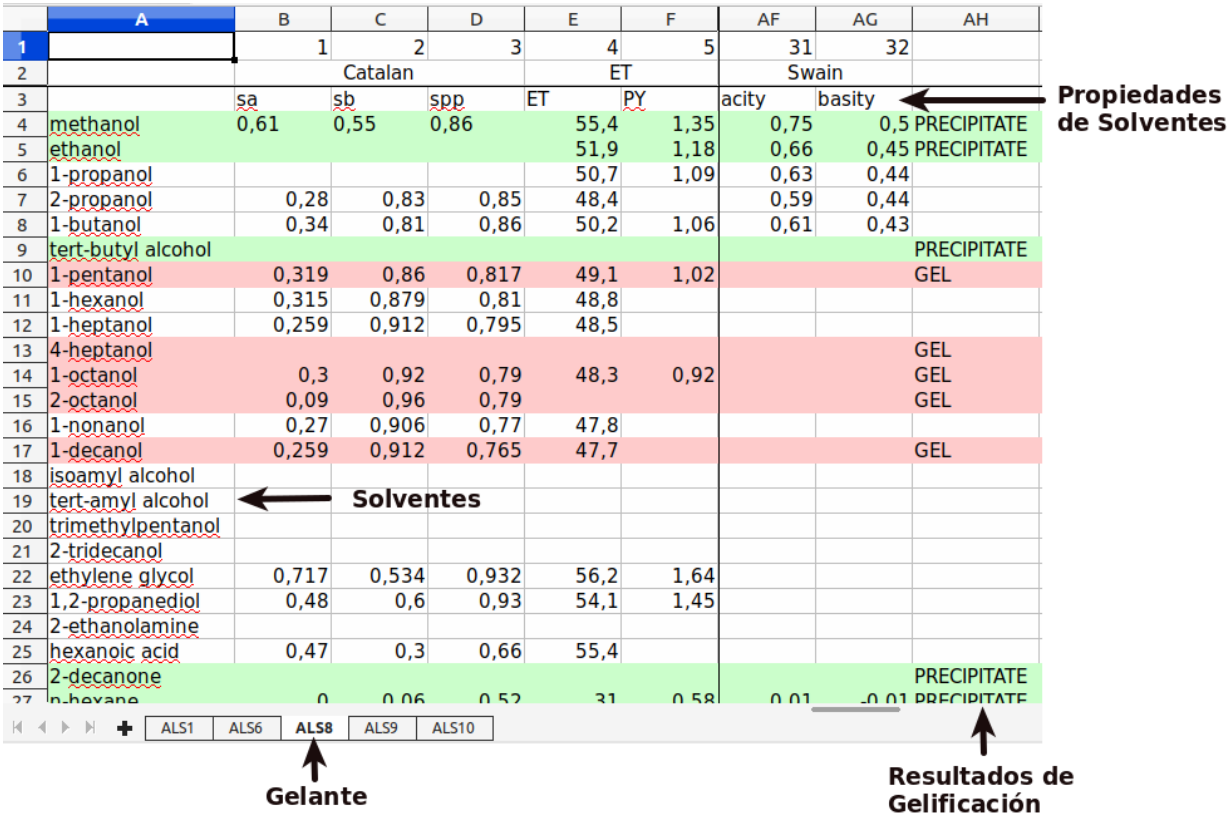
A continuación, se listan las planillas de datos disponibles. Cada una de ellas contiene datos correspondientes a un tipo particular de gelante.

Archivo	Descripción
All_ALS_Data_VF.xlsx	Compuestos ALS (siglas en inglés de Aromatic Linking Steroidal).
All_AZO_Data_VF.xlsx	Compuestos AZO (el término AZO proviene de *azote*, que significa nitrógeno en francés).
All_DBS_Data_VF.xlsx	Compuestos misceláneos.
All_Sugar_Data_VF.xlsx	Compuestos MBD (siglas en inglés de Methyl Benzylidene Derivatives).

Tabla 1 – Planillas de datos disponibles.

Cada una de estas planillas contiene lo siguiente (ver la estructura en la Figura 4):

- **Gelantes:** representados por el nombre de cada hoja de la planilla.
- **Solventes:** identificados en la primera columna (A) de cada hoja.
- **Propiedades de solventes:** presentes entre la segunda columna (B) y la columna AF, AG o AH, dependiendo de la cantidad de propiedades incluidas de los solventes listados para un gelante particular.
- **Resultados de gelificación:** en la columna AG, AH o AI, dependiendo el caso. Los valores posibles para esta columna son SOLUTION, GEL o PRECIPITATE.



	A	B	C	D	E	F	AF	AG	AH
1		1	2	3	4	5	31	32	
2		Catalan			ET		Swain		
3		sa	sb	spp	ET	PY	acity	basity	
4	methanol	0,61	0,55	0,86	55,4	1,35	0,75	0,5	PRECIPITATE
5	ethanol				51,9	1,18	0,66	0,45	PRECIPITATE
6	1-propanol				50,7	1,09	0,63	0,44	
7	2-propanol	0,28	0,83	0,85	48,4		0,59	0,44	
8	1-butanol	0,34	0,81	0,86	50,2	1,06	0,61	0,43	
9	tert-butyl alcohol								PRECIPITATE
10	1-pentanol	0,319	0,86	0,817	49,1	1,02			GEL
11	1-hexanol	0,315	0,879	0,81	48,8				
12	1-heptanol	0,259	0,912	0,795	48,5				
13	4-heptanol								GEL
14	1-octanol	0,3	0,92	0,79	48,3	0,92			GEL
15	2-octanol	0,09	0,96	0,79					GEL
16	1-nonanol	0,27	0,906	0,77	47,8				
17	1-decanol	0,259	0,912	0,765	47,7				GEL
18	isoamyl alcohol								
19	tert-amyl alcohol								
20	trimethylpentanol								
21	2-tridecanol								
22	ethylene glycol	0,717	0,534	0,932	56,2	1,64			
23	1,2-propanediol	0,48	0,6	0,93	54,1	1,45			
24	2-ethanolamine								
25	hexanoic acid	0,47	0,3	0,66	55,4				
26	2-decanone								PRECIPITATE
27	n-hexane	0	0,06	0,52	31	0,58	0,01	-0,01	PRECIPITATE

ALS1 | ALS6 | **ALS8** | ALS9 | ALS10

Figura 4 – Estructura de las planillas de datos.

Como ya se mencionó en el alcance, existe otro proyecto de trabajo final “Arquitectura Big Data para análisis de organogelantes” también asociado al proyecto de investigación *Data Science into Food Science.* Aunque dicha infraestructura no será considerada como fuente de datos para este trabajo final, la mencionamos aquí porque será de interés como desarrollo futuro del presente trabajo la construcción de una interfaz de integración.

3.1.1. DICCIONARIO DE DATOS.

El diccionario de datos es un documento sobre los datos (metadatos) compilado por el analista en sistemas y usado en el proceso de análisis y durante el diseño. El diccionario de datos recopila y coordina términos de datos específicos, además de confirmar lo que significa cada término para las distintas personas en la organización. [Kendall y Kendall, 2011].

En la siguiente tabla se describen las estructuras de los datos utilizadas para este trabajo. Se emplea la siguiente notación:

1. Un signo de igual (=) significa “está compuesto de”.
2. Un signo positivo (+) significa “y”.
3. Las llaves { } indican elementos repetitivos, también conocidos como grupos repetitivos o tablas repetitivas.
4. Los corchetes [ ] representan una situación del tipo cualquiera/o. Puede estar presente cualquiera de los elementos, pero solo uno, los elementos que se listan entre corchetes son mutuamente excluyentes.
5. Los paréntesis ( ) representan un elemento opcional.

Estructura lógica	Elementos
Gelante	Gelante = Id + Nombre + Tipo Gelante
Tipo Solvente	Tipo Solvente = Id + Nombre
Tipo Gelante	Tipo Gelante = Id + Nombre
Propiedad	Propiedad = Id + Nombre + [Cota superior] + [Cota inferior]
Valor Propiedad	Valor Propiedad = Id + Id Propiedad + Valor
Resultado	Resultado = Id + Id Solvente + Id Gelante + [GEL, SOLUTION, PRECIPITATE, NULL]

Tabla 2 - Estructura lógica de datos.

A continuación se describen los elementos de las estructuras lógicas especificadas en la tabla 2.

Estructura	Elemento	Descripción	Tipo	Longitud	Restricción
Solvente	Nombre	Nombre de una sustancia en la que se diluye un soluto	Texto	100	Obligatorio
Gelante	Nombre	Nombre de una sustancia que combinada con un solvente, puede formar un gel	Texto	100	Obligatorio
Tipo Solvente	Nombre	Definición del tipo de solvente	Texto	100	Obligatorio
Tipo Gelante	Nombre	Definición del tipo de gelante	Texto	100	Obligatorio
Propiedad	Nombre	Definición de la propiedad de un solvente	Texto	100	Obligatorio
Propiedad	Cota Superior	Cota superior del valor de una propiedad	Número real		Opcional
Propiedad	Cota Inferior	Cota inferior del valor de una propiedad	Número real		Opcional
Valor propiedad	Valor	Valor obtenido de la propiedad	Número real		Opcional
Resultado	Resultado	Resultado obtenido de la experiencia	Texto	50	Opcional

Tabla 3 - Elementos de datos.

3.1.2. EXTRACCIÓN, TRANSFORMACIÓN Y CARGA.

Para comenzar el proceso de unificación y homogeneización de la información disponible, los datos son tomados desde los archivos originales y almacenados en una base de datos relacional MySQL. La solución desarrollada en este trabajo utiliza los datos exclusivamente desde esta base de datos. Cada uno de los datos pasa por un proceso ETL (por sus siglas en inglés: *Extract, Transform and Load*). Se toma como base el diagrama de flujo de ETL propuesto por Verónica Cuello en su

trabajo “Arquitectura Big Data para análisis de organogelantes”. Consultar anexo – Diagrama de procesamiento ETL.

El proceso ETL es llevado a cabo por un script en el lenguaje de programación Python (ver Figura 5), el cual recibe como parámetros los *path* o rutas en donde se encuentran las planillas Excel con los datos y luego realiza los siguientes pasos:

- **Extracción:** se recorre archivo por archivo, hoja por hoja dentro de cada archivo, y dentro de cada hoja se hace una lectura fila por fila para obtener **tipo de gelante, nombre de gelante, nombre de solvente, nombre de propiedad de solvente y valor de propiedad de solvente.**

- **Transformación:** debido a que los nombres de solventes y de las propiedades, como así también los valores de estas últimas, pueden variar por distintos motivos (e.g. errores en la carga manual de los datos en las planillas), se realizan las siguientes comprobaciones y adecuaciones para homogeneizar los datos antes de ser almacenados.
    - Comprobación de nombre de solvente, nombre de gelante y nombre de propiedad de solvente: cada vez que se leen estos datos, se comprueba su existencia en las tablas mapping_solvente, mapping_gelante o mapping_propiedad, dependiendo el caso. Si el nombre coincide con alguno almacenado en estas tablas, se lo reemplaza por el nombre homologado, en caso de no encontrar coincidencia se genera una excepción y no se cargan los datos.
    - Comprobación de valor de propiedad: la comprobación del valor leído se hace contra la tabla propiedad y solo le cargará en los casos en que se encuentre dentro de los límites de cota superior y cota inferior especificados en esta tabla para cada propiedad. Si no se cumple con esta condición se genera una excepción y no se carga dato.

- Para ello se utilizan las tablas mapping también elaboradas por Verónica Cuello, consultar en el anexo - Tablas mapping.

- **Carga:** en este paso, luego de haber pasado por los procesos de extracción y transformación, se realiza la carga propiamente dicha de la información en la base de datos MySQL.

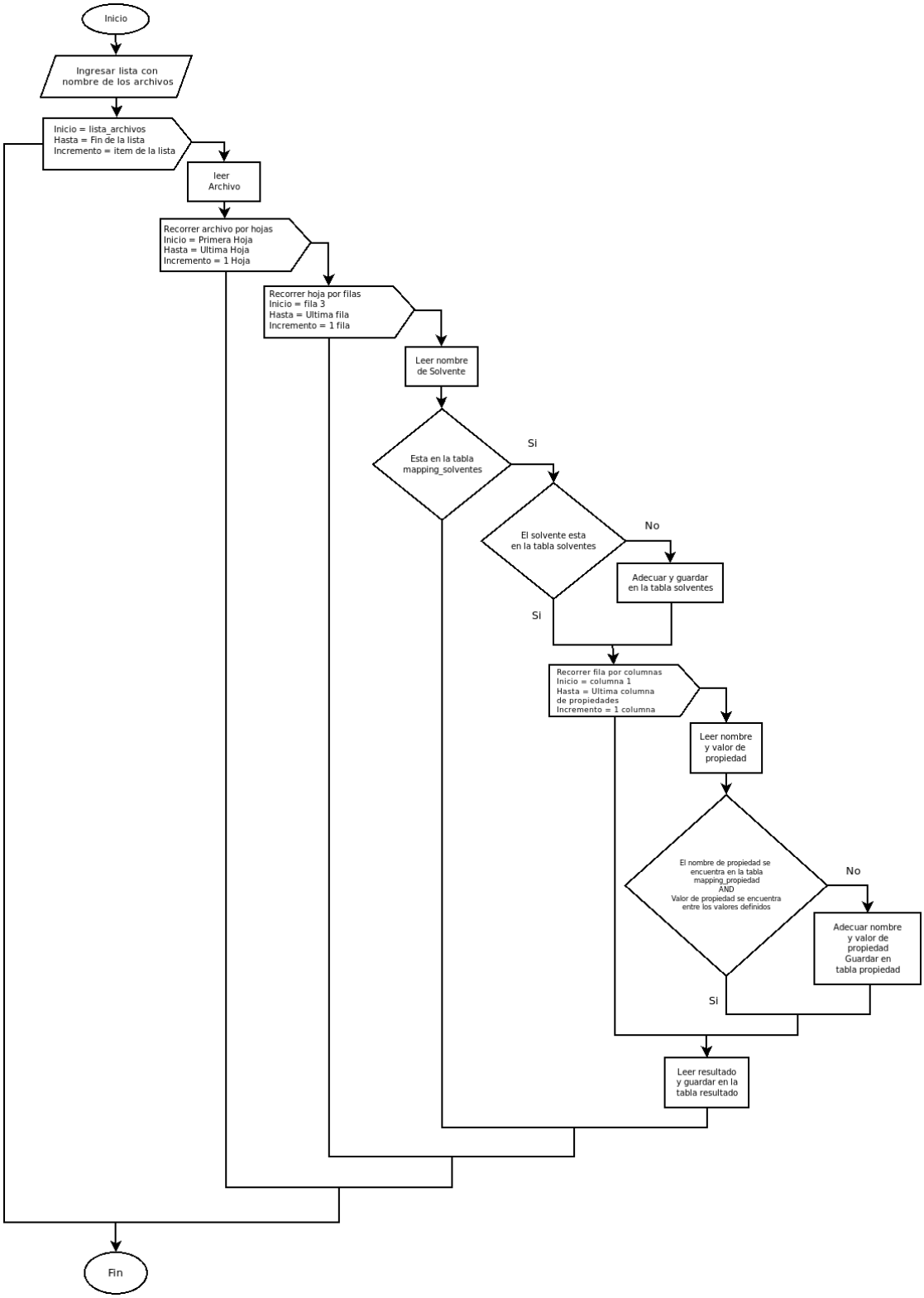


Figura 5 - Diagrama de flujo del proceso ETL.

3.2. MODELO DE MACHINE LEARNING

En esta sección detallamos la construcción del modelo predictivo que aprovecha los datos existentes para intentar determinar si en determinadas condiciones dadas por el usuario se produce efectivamente el proceso de gelificación.

Para la construcción del modelo predictivo, como indicamos anteriormente, este trabajo utiliza la librería de Python **scikit-learn**. Dicha librería proporciona una amplia gama de algoritmos de aprendizaje supervisado y no supervisado. Además, para facilitar el desarrollo y la exploración de datos, se emplea **Jupyter Notebook,** un entorno de trabajo interactivo que permite la evaluación de código Python de manera interactiva, integrando en un mismo documento tanto bloques de código como texto descriptivo, resultados de análisis y gráficos.

Como estructura de datos intermedia para uso de los algoritmos de scikit-learn, usaremos la librería pandas [McKinney, 2011] que provee una estructura de datos llamada DataFrame, equivalente en muchos sentidos a una tabla de base de datos relacional y que puede ser accedida directamente por los métodos de scikit-learn.

3.2.1. DATASET

Los datos son leídos desde la base de datos MySQL (ver sección extracción, transformación y carga) y cargados en un DataFrame de pandas antes de proceder a la exploración, limpieza y análisis de los mismos.

	resultado	gelante	gelante_tipo	solvente	Catalan sa	Catalan sb	Catalan spp	ET PY	ET et30	FH floryhuggins	...	MOSCED mo-lambda	MOSCED mo-q	MOSCED mo-tau	Physical DI	Physical Dipole
0	None	DBS	DBS	Methanol	0.61	0.55	0.86	1.35	55.4	0.430601	...	14.43	3.77	1.0	32.7	2.87
1	None	HSA	DBS	Methanol	0.61	0.55	0.86	1.35	55.4	0.430601	...	14.43	3.77	1.0	32.7	2.87
2	None	DBU	DBS	Methanol	0.61	0.55	0.86	1.35	55.4	0.430601	...	14.43	3.77	1.0	32.7	2.87
3	None	DCHU	DBS	Methanol	0.61	0.55	0.86	1.35	55.4	0.430601	...	14.43	3.77	1.0	32.7	2.87
4	None	CAB	DBS	Methanol	0.61	0.55	0.86	1.35	55.4	0.430601	...	14.43	3.77	1.0	32.7	2.87

5 rows × 34 columns

Figura 6 - Vista de las primeras cinco filas del DataFrame, a modo de ilustración de la estructura de datos.

El conjunto de datos contiene las variables detalladas en la Tabla 5, a razón de una por columna.

Variable	Descripción
resultado	resultado del proceso de gelificación al combinar un gelante y un solvente. (GEL, SOLUTION, PRECIPITATE)
gelante	Nombre del gelante que se diluye en el solvente.
gelante_tipo	Tipo del gelante que se diluye en el solvente. El cual puede ser uno del siguiente: (ALS, AZO, MBD, BDS)
solvente	Nombre del solvente utilizado en el proceso de gelificación.
Catalan sa	Parámetro solvatocrómico que caracteriza la acidez de un solvente.
Catalan sb	Parámetro solvatocrómico que caracteriza la basicidad de un solvente.
Catalan spp	Parámetro solvatocrómico que caracteriza la polarización de un solvente.
ET PY	Coeficiente de absorción molar para agregados de pynere.
ET et30	Parámetro ET de Dimroth-Reichardt. Medida de la potencia ionizante (perdida de polaridad) de un solvente.
FH floryhuggins	Parámetro de interacción de Flory-Huggins. Representa una cantidad adimensional que caracteriza la energía de interacción entre moléculas de polímero con solvente.
HD	Parámetro que indica la capacidad de un solvente para donar enlaces de puente de hidrogeno.
A	Parámetro que indica la capacidad de un solvente para aceptar enlaces de puente de hidrogeno.
HD/A	Cociente entre los parámetros HD y A.
Hansen R_{ij}	Distancia vectorial en el espacio de Hansen desde un solvente (*j*) hasta el centro de la esfera de solubilidad de un gelante (i)
Hansen dsolvent	Parámetro de solubilidad de Hansen dispersivo. δ_d

Hansen hsolvent	Parámetro de solubilidad de Hansen enlace de hidrogeno δ_h
Hansen psolvent	Parámetro de solubilidad de Hansen polar δ_p
Hansen dtotal	Sumatoria de los parámetros de Hansen $\delta_t = \delta_d + \delta_h + \delta_p$
Hildebran d total	Parámetro de solubilidad de Hildebran. Refleja la energía de cohesiva de un material.
Kamlet K-Pi	Parámetro de solubilidad de Kamlet – Taft (π). Caracteriza un solvente con respecto a su polarización.
Kamlet K-alpha	Parámetro de solubilidad de Kamlet – Taft (α). Capacidad de donar puente de hidrogeno.
Kamlet K-Beta	Parámetro de solubilidad de Kamlet – Taft (β). Capacidad de aceptar puente de hidrogeno.
MOSCED mo-alpha	Parámetro de acidez (α) de Mosced. Capacidad de donar puente de hidrogeno.
MOSCED mo-beta	Parámetro de basicidad (β) de Mosced. Capacidad de aceptar puente de hidrogeno
MOSCED mo-lambda	Parámetro de dispersión (λ) de Mosced.
MOSCED mo-q	Parámetro de inducción (q) de Mosced.
MOSCED mo-tau	Parámetro de polaridad (τ) de Mosced.
Physical DI	Constante dieléctrica (permitividad estática relativa), se usa para evaluar la polaridad de un solvente.
Physical Dipole	Momento dipolar de un solvente.
Physical Henry	Constante de la ley de Henry, característica para cada soluto.
Physical logP	Coeficiente de reparto octanol/agua.
Swain Acity	Parámetro Acity en la escala de Swain. Capacidad de solvatación de aniones.
Swain Basity	Parámetro Basity en la escala de Swain. Capacidad de solvatación de cationes.

Tabla 5 – Descripción de Variables.

3.2.2. ANÁLISIS DE DATOS

Previo a la construcción de un modelo predictivo es de suma importancia explorar y visualizar los datos disponibles, de manera de comprender sus características y posibles errores. Luego de este proceso exploratorio se pasa a la selección de aquellas variables que a priori serán útiles como predictores para el modelo.

Se comienza mostrando el número total de datos como así también su tipo - esto es, si son numéricos o no numéricos (ver Figura 7).

```
df_total.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1779 entries, 0 to 1778
Data columns (total 34 columns):
resultado              864 non-null object
gelante                1779 non-null object
gelante_tipo           1779 non-null object
solvente               1779 non-null object
Catalan sa             974 non-null float64
Catalan sb             962 non-null float64
Catalan spp            954 non-null float64
ET PY                  877 non-null float64
ET et30                1154 non-null float64
FH floryhuggins        1416 non-null float64
HAD A                  1415 non-null float64
HAD HD                 1415 non-null float64
HAD HD/A               1235 non-null float64
Hansen Rij             1667 non-null float64
Hansen dsolvent        1779 non-null float64
Hansen dtotal          1779 non-null float64
Hansen hsolvent        1775 non-null float64
Hansen psolvent        1768 non-null float64
Hildebrand d total     1779 non-null float64
Kamlet K-Pi            1072 non-null float64
Kamlet K-alpha         1122 non-null float64
Kamlet K-beta          1096 non-null float64
MOSCED mo-alpha        1046 non-null float64
MOSCED mo-beta         1041 non-null float64
MOSCED mo-lambda       1007 non-null float64
MOSCED mo-q            1041 non-null float64
MOSCED mo-tau          1041 non-null float64
Physical DI            1079 non-null float64
Physical Dipole        1361 non-null float64
Physical RI            1218 non-null float64
Physical henry         1382 non-null float64
Physical logP          1374 non-null float64
Swain acity            946 non-null float64
Swain basity           546 non-null float64
dtypes: float64(30), object(4)
memory usage: 486.4+ KB
```

Figura 7 - Información del DataFrame que se obtiene con el método .info() de pandas.

En nuestro trabajo disponemos, una vez terminada la carga de datos en crudo, de 1779 filas de datos, correspondientes cada una de ellas a las pruebas realizadas en laboratorio entre gelantes y solventes. Para cada una de estas pruebas se registran 34 variables fisicoquímicas, en columnas. El tipo de dato float64 permite almacenar caracteres

numéricos con coma flotante, mientras que el tipo de dato object alberga secuencias de caracteres (números y/o texto).

El DataFrame tiene 34 columnas, de las que 33 corresponden a posibles predictores; la primera, resultado, contiene la variable objetivo o variable a predecir (recordar que se trata de un problema de aprendizaje supervisado). En los archivos de datos de origen, los resultados posibles son tres, a saber: “GEL”, “PRECIPITATE” o “SOLUTION”. El objetivo de nuestra aplicación (por lo menos en su primera versión) es detectar si se produce un gel o no, por lo cual reducimos el problema a una clasificación binaria realizando lo siguiente:

- Se reemplaza el valor del atributo **resultado** por “1” para los casos que contengan “GEL” y por “0” para los casos “PRECIPITATE” y “SOLUTION”.

- Se eliminan los registros cuyo valor del atributo **resultado** sea nulo o vacío. Como se mencionó en la sección Fuente de datos, esto sucede cuando no se tiene resultado del proceso de gelificación, y dichos registros no aportan información para la construcción del modelo.

Como futura línea de investigación se puede considerar un modelo de clasificación multiclase (para contemplar los tres resultados posibles); sin embargo, el caso de clasificación binaria (gel o no gel) es la aplicación de mayor interés para uso en TIA.

Luego de este proceso quedan un total de 864 registros.

Existe propiedades de determinados solventes para los cuales no se tiene un valor, esto se debe a que se desconoce ese valor (no se encuentra disponible en la bibliografía), como así tampoco se obtuvo mediante instrumentos de laboratorio. Como ya se mencionó en este trabajo, estas tareas de laboratorio no siempre son posibles de llevarlas a cabo debido al tiempo y costo que implican.

La figura 8 muestra un subconjunto de 5 registros, tomados de manera aleatoria, en donde se observa la existencia de valores faltantes. La librería **pandas** muestra estos valores como **NaN** (por sus siglas en ingles: *not a number*).

```
df_total.sample(5)
```

solvente	Catalan sa	Catalan sb	Catalan spp	ET PY	ET et30	FH floryhuggins	...	MOSCED mo-q	MOSCED mo-tau	Physical DI	Physical Dipole	Physical RI	Physical henry	Physical logP	Swain acity	Swain basity
Decanal	0.0	0.557	0.883	NaN	39.3	NaN	...	5.49	1.0	17.00	2.82	1.3900	0.000050	0.73	NaN	NaN
Ethyl Acetate	0.0	0.540	0.800	1.37	38.1	0.124516	...	5.74	1.0	6.08	1.88	1.4400	0.000134	0.64	0.21000	0.59
Dipropyl Ether	0.0	0.670	0.680	NaN	34.0	0.357018	...	2.00	1.0	NaN	1.12	1.3810	0.002200	2.03	17.87779	NaN
Octane	0.0	0.079	0.540	1.18	31.1	0.338769	...	0.00	1.0	NaN	0.00	1.3970	3.200000	3.90	NaN	NaN
Toluene	0.0	0.128	0.655	0.60	33.9	0.004274	...	3.22	0.9	36.60	0.37	1.4961	0.006640	3.15	0.13000	0.54

Figura 8 – Vista de 5 filas aleatorias del DataFrame.

En la siguiente figura se muestra de manera gráfica el total de valores no nulos por cada una de las variables.

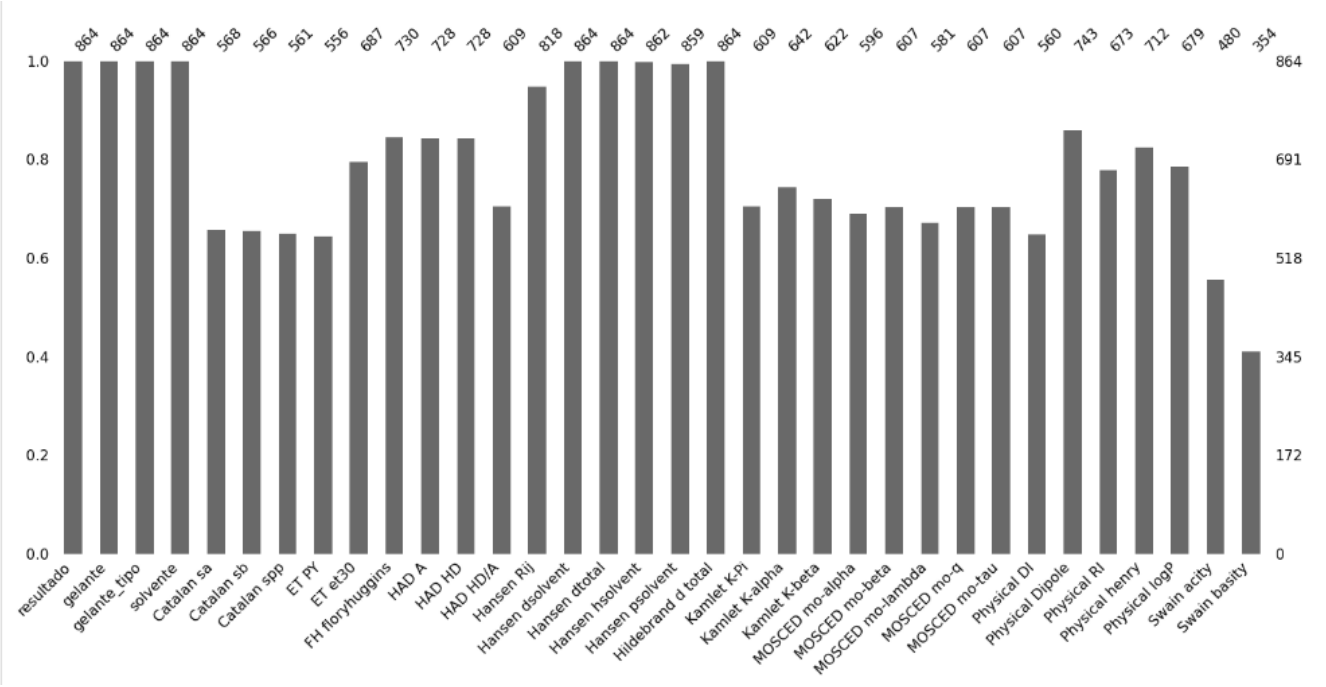


Figura 9 – Valores no nulos por variable.

A través de la utilización del método *isnull()* en combinación con el método *sum(),* ambos de la librería **pandas**, es posible obtener la cantidad de registros que presentan al menos una variable con valor nulo (ver Figura 10).

```
df_total.isnull().any(axis=1).sum()
```

```
738
```

Figura 10 – Cantidad de registros con al menos una variable con valores nulos.

Entre los métodos más utilizados para el tratamiento de datos faltantes se encuentran: **análisis de casos completos, análisis de casos disponibles e imputación de valores unicos** [Pigot, 2001].

El método de **análisis casos completos** consiste en omitir todos los registros que tengan valores faltantes (eliminación por filas). Considerando los 864 registros con los que se cuentan hasta el momento y observando la cantidad de registros con valores nulos (Ver Figura 10), aplicar este método reduciría el DataSet al 14,6% del total. Ajustar nuestro modelo teniendo en cuenta solo los casos completos sería ineficaz.

Otra opción es excluir la variable o conjunto de variables que presenten una alta tasa de valores perdidos, a este método se lo conoce como **análisis de casos disponibles** o también llamado **análisis de variables completas**. Haciendo un análisis similar al anterior, en la Figura 9 se puede apreciar que solo las variables Hansen dsolvent, Hansen hsolvent, Hansen psolvent, Hansen dtotal, Hildebrand d total, gelante, tipo_gelante y solvente tienen valores para todos registros del Dataset, la utilización de este método implicaría eliminar una gran cantidad de variables y dejando de lado información que pueden aportar los valores no nulos de estas.

En el método de **imputación de valores únicos** tiene como objetivo sustituir los valores faltantes por estimaciones estadísticas. Una de las posibilidades es utilizar el valor medio de las variables en lugar de los datos los faltantes de esas variables. A pesar de las limitaciones de este método, como por ejemplo que su aplicación afecta la distribución de la probabilidad de la variable imputada o atenúa la correlación con el resto de las variables, se considera un procedimiento apropiado cuando los datos faltantes siguen el

patrón MCAR[3] [Medina y Galban, 2007]. Teniendo en cuenta el tamaño de nuestro *DataSet,* se considera el **método de imputación de la media** como el más apropiado para el tratamiento de datos faltantes ya que permitiría un mayor aprovechamiento de los datos disponibles.

Por lo antes mencionado se realiza lo siguiente:

- Se reemplazan los valores nulos de los atributos que representan propiedades de solventes por el valor medio de cada propiedad. Esto se realiza debido a que el algoritmo no acepta valores nulos para la construcción del modelo.

	resultado	gelante	gelante_tipo	solvente	Catalan sa	Catalan sb	Catalan spp	ET PY	ET et30	FH floryhuggins	...	MOSCED mo-lambda	MOSCED mo-q	MOSCED mo-tau	Physical DI	Physical Dipole
10	0	SUGAR NITRO	MBD	Methanol	0.61	0.55	0.86	1.35	55.4	0.430601	...	14.43	3.77	1.0	32.7	2.87
11	0	ALS1	ALS	Methanol	0.61	0.55	0.86	1.35	55.4	0.430601	...	14.43	3.77	1.0	32.7	2.87
12	0	ALS6	ALS	Methanol	0.61	0.55	0.86	1.35	55.4	0.430601	...	14.43	3.77	1.0	32.7	2.87
13	0	ALS8	ALS	Methanol	0.61	0.55	0.86	1.35	55.4	0.430601	...	14.43	3.77	1.0	32.7	2.87
14	0	ALS9	ALS	Methanol	0.61	0.55	0.86	1.35	55.4	0.430601	...	14.43	3.77	1.0	32.7	2.87

5 rows × 34 columns

Figura 11 - Vista de las cinco primeras filas del DataFrame, post limpieza de datos.

3.2.2.1. ANÁLISIS DE LOS POTENCIALES PREDICTORES.

Para un análisis más detallado de las variables que potencialmente se pueden usar como predictores para el modelo, el primer paso es obtener estadísticas descriptivas básicas para cada columna: promedio, desviación estándar, mínimo/máximo, etc. Para ello se puede utilizar directamente un método de la estructura de datos DataFrame, que genera dicha información estadística y resume la tendencia central, la dispersión y la forma de la distribución del conjunto de datos con el que se trabaja (ver Figura 12).

[3] MCAR: por sus siglas en inglés (*missing completely at random data).* Los datos se consideran faltantes en forma completamente aleatoria cuando la probabilidad de datos faltantes en la variable (Y) no está relacionada con los valores de otras variables ni con los valores de (Y) en si misma.

	Catalan sa	Catalan sb	Catalan spp	ET PY	ET et30	FH floryhuggins	HAD A	HAD HD	HAD HD/A	Hansen Rij	Hansen dsolvent
count	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000
mean	58.515374	130.313681	226.706215	0.748588	33.118750	0.168833	5.695139	3.483912	0.327586	5.528314	16.573380
std	183.787251	255.727648	364.959484	0.631766	18.465237	0.162949	5.119042	5.471318	0.508905	2.634031	1.395508
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	14.000000
25%	0.000000	0.000000	0.000000	0.000000	31.000000	0.037672	0.000000	0.000000	0.000000	4.047221	15.500000
50%	0.000000	0.240000	0.680000	0.920000	37.400000	0.129289	5.500000	0.100000	0.022222	5.173973	16.000000
75%	0.143625	86.000000	616.000000	1.240000	47.700000	0.254918	8.700000	6.400000	0.693250	7.034202	17.800000
max	1062.000000	926.000000	964.000000	1.950000	63.100000	1.281429	21.400000	22.400000	2.333300	13.855320	20.500000

	Hansen dtotal	Hansen hsolvent	Hansen psolvent	Hildebrand d total	Kamlet K-PI	Kamlet K-alpha	Kamlet K-beta	MOSCED mo-alpha	MOSCED mo-beta	MOSCED mo-lambda	MOSCED mo-q
count	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000
mean	20.758021	8.250116	5.548866	20.751183	0.355162	0.208750	0.282361	2.315150	4.552153	10.533692	2.961690
std	5.434912	7.772577	4.608258	5.440918	0.334627	0.359954	0.301213	4.743959	5.874609	7.422857	3.526702
min	14.221460	0.000000	0.000000	14.221460	-0.080000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	17.732740	2.000000	1.400000	17.732740	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	19.528700	6.500000	5.100000	19.528700	0.400000	0.000000	0.120000	0.000000	0.920000	14.900000	1.460000
75%	21.927840	11.900000	8.600000	21.927840	0.600000	0.130000	0.480000	1.610000	8.380000	16.060000	5.145000
max	47.807320	42.300000	18.000000	47.807320	1.090000	1.220000	0.900000	27.150000	26.170000	19.670000	13.360000

	MOSCED mo-tau	Physical henry	Physical RI	Physical DI	Physical Dipole	Physical logP	Swain acity	Swain basity
count	864.000000	8.640000e+02	864.000000	864.000000	864.000000	864.000000	864.000000	864.000000
mean	0.688681	3.236126e-01	1.113418	12.815694	1.161563	1.000764	3.059828	0.203333
std	0.449472	1.188015e+00	0.596282	15.674259	1.098492	1.654443	8.146436	0.305925
min	0.000000	0.000000e+00	0.000000	0.000000	0.000000	-1.800000	0.000000	0.000000
25%	0.000000	3.370000e-07	1.344000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	1.000000	2.670000e-05	1.399000	7.600000	1.130000	0.340000	0.150000	0.000000
75%	1.000000	3.250000e-03	1.430000	17.812500	1.850000	1.860000	0.370000	0.430000
max	1.010000	8.200000e+00	1.628000	80.100000	4.280000	7.200000	36.081330	1.000000

Figura 12 - Información estadística de variables numéricas.

Estos resúmenes numéricos nos permiten comprender cuáles son los valores típicos de las propiedades fisicoquímicas en los datos, y cuál es el rango de variación. Sin embargo, esta información debe complementarse con una visualización de la distribución de dichas propiedades. A continuación, se generan histogramas de algunos atributos para tener una impresión visual de su distribución (Figuras 13 y 14).

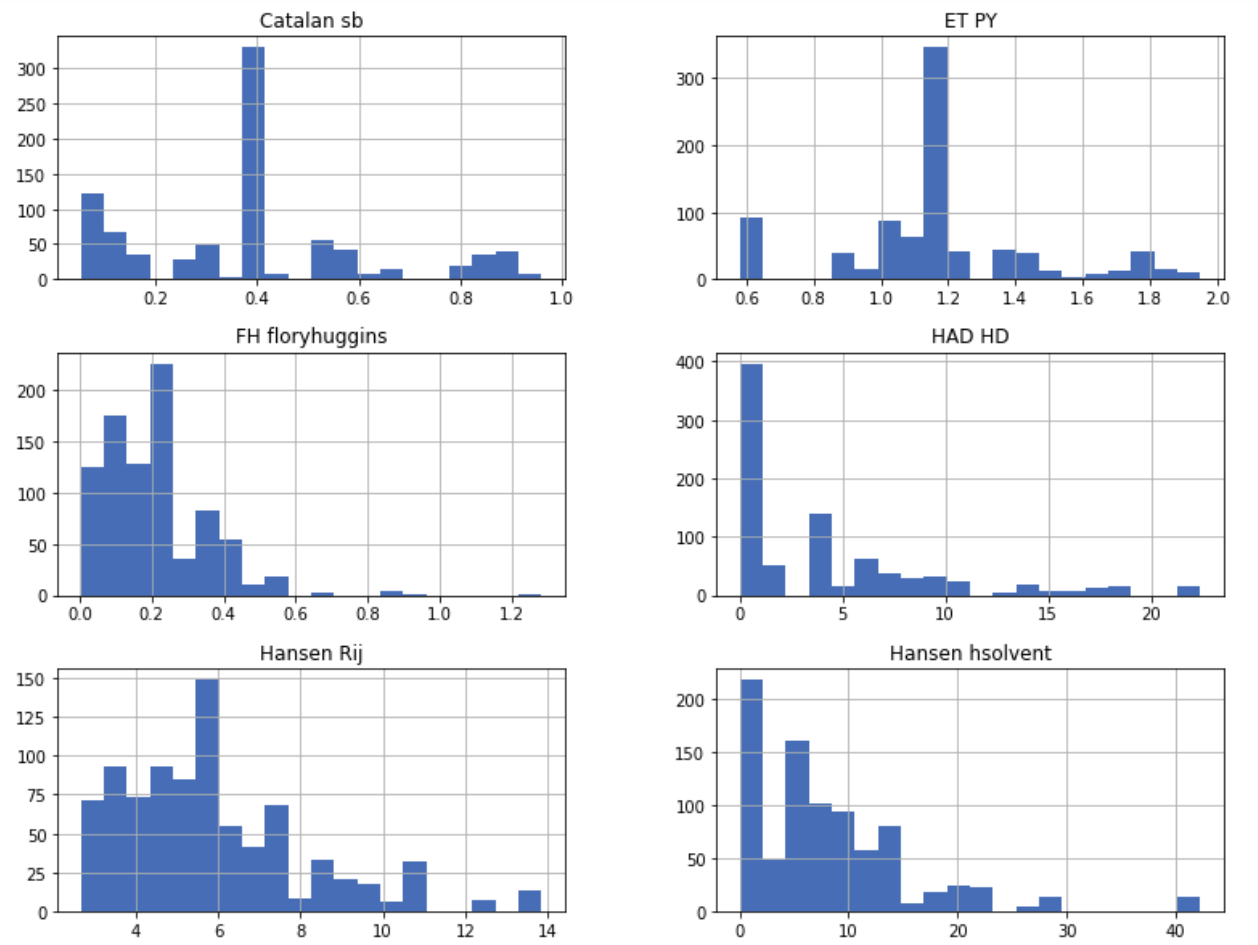


Figura 13 - Distribuciones de los valores de las siguientes propiedades (Catalan sb, ET PY, FH floryhuggins, HAS HD, Hansen Rij, Hansen hsolvent).

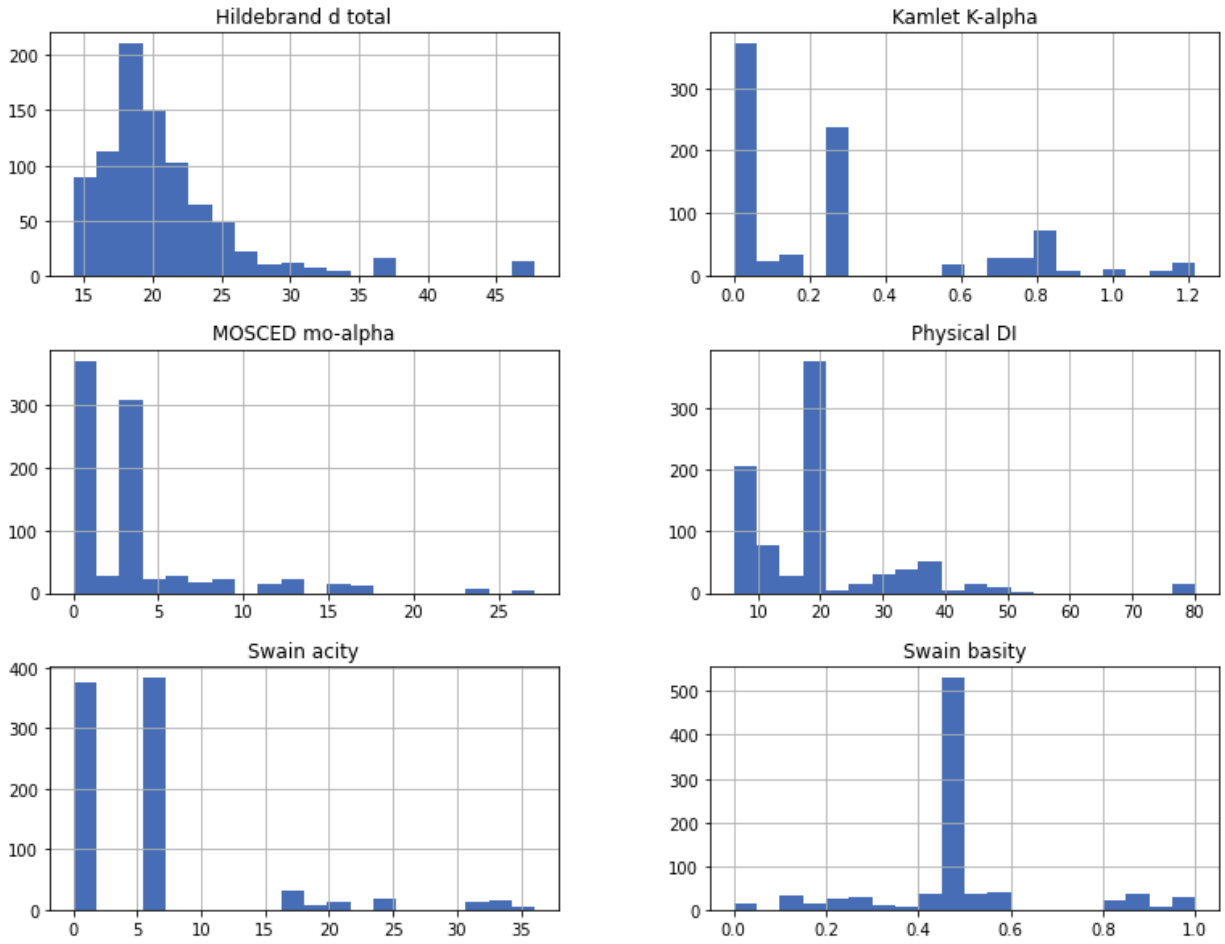


Figura 14 - Distribuciones de los valores de las siguientes propiedades (Hildebrand d total, Kamlet K-alpha, MOSCED mo-alpha, Physical DI, Swain acity, Swain basity).

Los histogramas de las Figuras 13 y 14 muestran que las distintas variables presentan distribuciones muy variadas. Un detalle que se nota en la mayoría de ellos es que para muchas filas de datos hay propiedades con valor cero. Si bien en algunos casos (FH floryhuggins, por ejemplo, a la izquierda y centro en la Figura 13) podría corresponder a valores reales medidos de la propiedad, en otros (como ser Swain acity, abajo a la izquierda en la Figura 14) pareciera indicar la ausencia de datos de esa propiedad para ese experimento, teniendo en cuenta que para los registros no nulos los valores oscilan entre aproximadamente 5 y 35. Otro indicador de la existencia de propiedades con valor nulo , es la alta frecuencia del valor medio en cada gráfico, ya que durante el proceso de

análisis de datos los valores nulos fueron reemplazados por la media aritmética de cada predictor.

Lo anterior muestra una limitación de este proyecto, lamentablemente impuesta por los datos disponibles. Como se indicó en la Introducción [Corradini y otros, 2016, 2017] la aplicación de técnicas de análisis de datos y machine learning a TIA está en su infancia, y no hay registros consistentes en los experimentos de laboratorio. El presente trabajo, juntamente con el llevado a cabo por Verónica Cuello (Arquitectura Big Data para análisis de comportamiento de organogelantes), intenta generar un marco metodológico que permita la generación de un estándar de datos que a futuro optimice la generación de conocimiento; pero, de momento, nos encontramos con que en algunos experimentos hay información sobre algunas variables que faltan en otros. Esto no imposibilita la generación de un modelo predictivo, pero sí afecta la precisión que podremos obtener sobre los resultados, al faltar información que permita determinar mejor los patrones de correlación entre propiedades químicas y el proceso de gelificación.

3.2.2.2. CORRELACIÓN LINEAL DE ATRIBUTOS.

Continuando con el análisis se comprueba cómo se relacionan los datos con lo que se trata de predecir, en este caso la gelificación o no (Figura 15).

El coeficiente de correlación lineal oscila en -1 y 1:

- Coeficientes cercanos a 1 implican una fuerte correlación positiva.
- Coeficientes cercanos a -1 implican una fuerte correlación negativa.
- Coeficientes cercanos a 0 significan que no hay correlación lineal.

```
matriz_cor = df_resultado.corr()
matriz_cor['resultado'].sort_values(ascending=False)
```

```
resultado              1.000000
Physical logP          0.104933
Physical henry         0.104057
FH floryhuggins        0.099481
Hansen Rij             0.065854
Kamlet K-alpha         0.035194
Catalan sb             0.034934
Kamlet K-beta          0.004256
Physical RI            0.003499
MOSCED mo-alpha       -0.008219
HAD A                 -0.050486
Catalan sa            -0.051583
MOSCED mo-beta        -0.052363
Physical Dipole       -0.054672
MOSCED mo-lambda      -0.058947
HAD HD/A              -0.062653
Swain acity           -0.065126
HAD HD                -0.069451
MOSCED mo-tau         -0.073135
Hansen dsolvent       -0.096963
gelante_tipo          -0.097935
Catalan spp           -0.101140
gelante               -0.111583
Hansen hsolvent       -0.113430
Hansen psolvent       -0.131818
Physical DI           -0.142033
ET et30               -0.144133
Hansen dtotal         -0.158557
Hildebrand d total    -0.158679
Kamlet K-Pi           -0.160506
ET PY                 -0.165879
Swain basity          -0.173530
MOSCED mo-q           -0.173951
Name: resultado, dtype: float64
```

Figura 15 - Coeficiente de correlación entre los predictores y el atributo ***resultado.***

Asimismo, es importante considerar la correlación existente entre los predictores. Para ello, las visualizaciones de tipo mapa de calor son de utilidad (ver Figura 16).

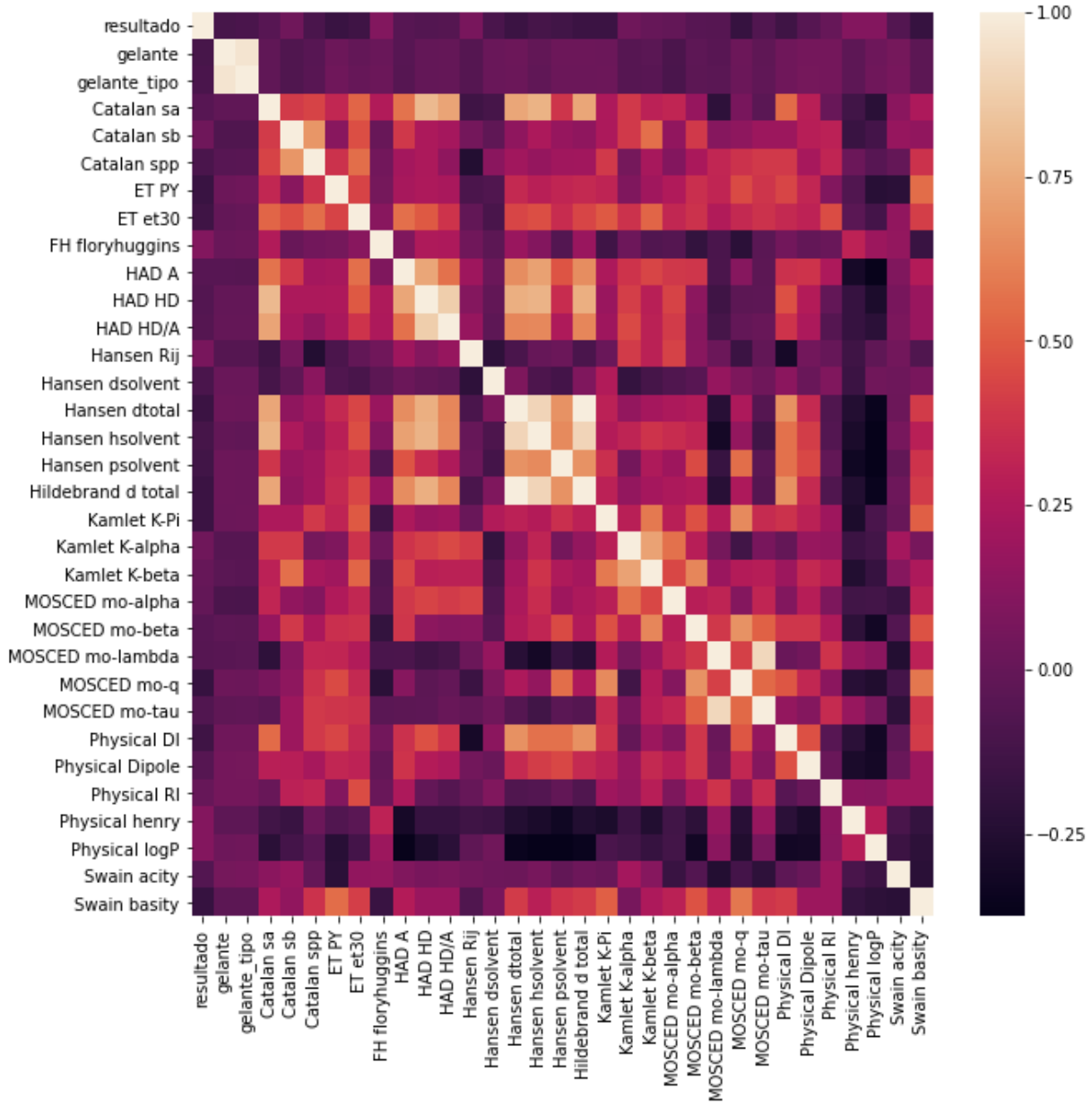


Figura 16 - Correlación de predictores entre sí.

En grafico anterior se observa lo mismo que en la figura 15, una baja correlación entre la variable objetivo **resultado** y los distintos predictores, siempre menores a 0.17, tanto positivos como negativos. Permite identificar predictores fuertemente correlacionados, como es el caso de los predictores **gelante** y **gelante_tipo**, la correlación es tan significativa, cercana a 0.9, que básicamente proporcionan la misma información. Otro punto interesante es la alta correlación que se puede apreciar entre los predictores

multitérminos como los de Hansen (**Hansen hsolvent** y **Hansen psolven**t) o los de Kamlet (**Kamlet K-alpha** y **Kamlet K-beta**).

3.2.3. CONSTRUCCIÓN DEL MODELO

Para la construcción del modelo de aprendizaje supervisado se utiliza el algoritmo de regresión logística, que es uno de muchos ya implementados en la librería scikit-learn. Este algoritmo es ampliamente usado para problemas de clasificación binaria, y suele ser una buena primera elección para este tipo de problemas [Zumel y Mount 2014, Raschka y Mirjalili 2017].

La regresión logística es el miembro más comúnmente utilizado de una clase de modelos llamados **modelos lineales generalizados**. Sea y la variable objetivo, que queremos predecir en términos de un conjunto de predictores x_1, x_2, … x_N. Un modelo lineal generalizado busca modelar a y en términos de los x_i en la siguiente forma:

$$g(y) = a_0 + a_1 x_1 + a_2 x_2 + \ldots + a_N x_N + e,$$

donde a_i son los coeficientes del modelo, e es un término de error y $g(y)$ se denomina **función de enlace**.
En el caso de la regresión logística, la función de enlace correspondiente se denomina *función logit* y es el logaritmo natural de la probabilidad de obtener la categoría 1 (producir un gel) dividida por la probabilidad de obtener la categoría 0 (no gel):

$$ln\left[\frac{P(y = 1 \mid x)}{1 - P(y = 1 \mid x)}\right] = a_0 + a_1 x_1 + \ldots$$

A ese cociente de probabilidades se lo denomina razón de oportunidades (*odds ratio*, OR, en inglés). A diferencia de la regresión lineal (en la cual $g(y) = y$), la regresión logística predice directamente valores restringidos al intervalo [0; 1], con lo cual sus valores de

salida pueden interpretarse directamente como la probabilidad de que una instancia pertenezca a una categoría específica (producir un gel, en nuestro caso).

Entrenar un modelo de regresión logística consiste en encontrar los valores de los coeficientes a_i tales que minimicen el error de clasificación. De la ecuación anterior se puede ver que el coeficiente a_i de la regresión logística da el cambio en el logaritmo de OR ante un cambio de la variable predictora x_i, manteniendo todas las demás constantes.

Existen otros algoritmos que pueden usarse para resolver problemas de clasificación, como por ejemplo los árboles de decisión, las máquinas de soporte vectorial (*support vector machines*, SVM) o las redes neuronales [ver una reseña en Zumel y Mount 2014]; en el presente trabajo se considera sólo la regresión logística porque a) constituye un benchmark interpretable y ampliamente usado para problemas de clasificación y b) las limitaciones en los datos hacen que no tenga mucho sentido aplicar modelos de mayor sofisticación (que, por otra parte, pierden en interpretabilidad).

Una vez entrenado un modelo, para verificar la performance del mismo se deben realizar dos comprobaciones:

1. Medir la calidad del modelo como resultado del proceso de entrenamiento.
2. Asegurar que el modelo funcionará igual de bien cuando se lo ponga en fase de producción, expuesto a datos que nunca ha visto antes.

Estas dos tareas se denominan **evaluación** y **validación**, respectivamente. Para poder realizarlas, antes de comenzar con el entrenamiento del modelo se divide el conjunto de datos en dos subconjuntos (ver Figura 17):

- Datos de entrenamiento (train set): con los que se entrenará el modelo.
- Datos de prueba (test set): estos datos se mantienen separados, y nunca se usan durante el entrenamiento. Su función es modelar datos previamente desconocidos,

con los que se comprobará la capacidad del modelo de generalizar a situaciones nuevas.

Para esto, se separa en primer lugar el conjunto de datos total en dos DataFrames: uno denominado all_x que contiene los predictores y otro denominado all_y que solo contiene los valores de la variable **resultado,** es decir los valores a predecir. Luego, cada uno de estos DataFrames se parte en dos grupos: 75% de los datos, tomados al azar, constituyen los datos de entrenamiento y el 25% restante son los datos de prueba a utilizar para la evaluación y ajuste del modelo.

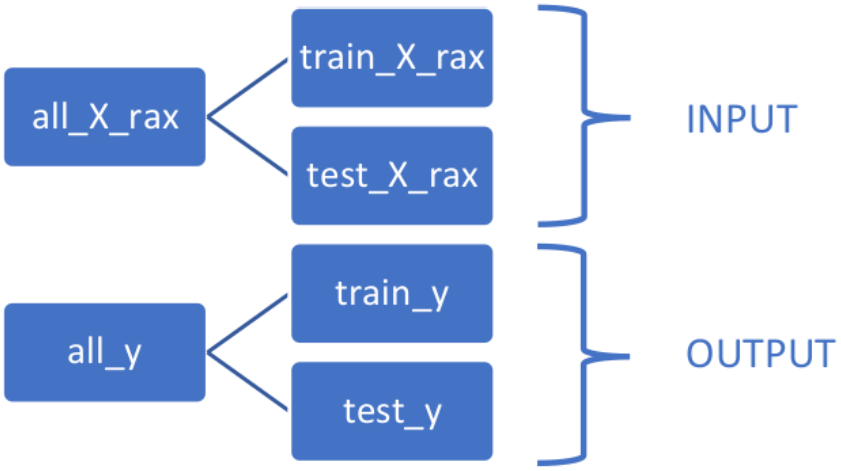


Figura 17 - Esquema de separación de datos en conjuntos de entrenamiento y de prueba.

3.2.3.1. ENTRENAMIENTO DEL MODELO

Para entrenar el modelo se genera una instancia del objeto *LogisticRegression()* de scikit-learn para definir el modelo. La clase LogisticRegression tiene un método fit() que es el que realiza el entrenamiento, recibiendo como parámetros de entrada los DataFrames con los datos de entrenamiento (tanto los predictores como las variables objetivo correspondientes):

```
#Modelo de Regresion Logistica

modelo_log = LogisticRegression(solver='lbfgs', max_iter=400)

modelo_log.fit(x_train[predictores], y_train,)
```

Figura 18 – Entrenamiento del modelo.

Utilizando el método .coef_() de la clase LogisticRegression se construye una matriz de coeficientes que muestra el peso o incidencia que tiene cada predictor en nuestro modelo (Figura 19).

```
#Matriz de coeficientes
pd.DataFrame(list(zip(train_x.columns, np.transpose(modelo_log.coef_))))
```

	0	1
0	gelante	[-0.09284702642210942]
1	gelante_tipo	[0.4553723556598678]
2	Catalan sa	[1.5710730805220663]
3	Catalan sb	[0.6906618834679549]
4	Catalan spp	[-0.6273211490077185]
5	ET PY	[-0.34301242044006613]
6	ET et30	[-0.015049601291481982]
7	FH floryhuggins	[0.5843016032819135]
8	HAD A	[0.02997635461883278]
9	HAD HD	[0.05933029667591569]
10	HAD HD/A	[-0.5882629773363777]
11	Hansen Rij	[0.002820308880598895]
12	Hansen dsolvent	[0.012452827727245432]
13	Hansen dtotal	[-0.24845390398869843]
14	Hansen hsolvent	[-0.09035052963760691]
15	Hansen psolvent	[-0.028086056251745074]
16	Hildebrand d total	[0.23874060199162503]
17	Kamlet K-Pi	[-0.5731908448170788]
18	Kamlet K-alpha	[0.16726948265157582]
19	Kamlet K-beta	[0.7633531375714857]
20	MOSCED mo-alpha	[0.03945577315171471]
21	MOSCED mo-beta	[0.015156753468803057]
22	MOSCED mo-lambda	[-0.04356885386564341]
23	MOSCED mo-q	[0.0632774850294533]
24	MOSCED mo-tau	[-0.022754189746591734]
25	Physical DI	[0.000314317907298877]
26	Physical Dipole	[0.1134686768929288]
27	Physical RI	[0.24606272311010927]
28	Physical henry	[0.12711328609034506]
29	Physical logP	[0.037508612031643795]
30	Swain acity	[-0.0333238684249018]
31	Swain basity	[-0.28602248232089195]

Figura 19 - Coeficientes obtenidos del entrenamiento de un modelo de regresión logística con scikit-learn usando los datos disponibles.

En un modelo de regresión logística la interpretación de los coeficientes, aunque más compleja que en una regresión lineal, es más sencilla que en el caso de otros modelos de clasificación como por ejemplo SVM o redes neuronales (estas últimas, notoriamente

difíciles de interpretar). Si el coeficiente de la variable x_i es a_i, entonces las oportunidades (odds) de un resultado positivo se multiplican por un factor $exp(a_i)$ por cada unidad de cambio en la variable.

Por ejemplo: según nuestro modelo, el coeficiente de la variable Physical Dipole es, redondeado al segundo decimal, igual a 0,11. De la Figura 12 podemos ver que, en promedio, esta variable tiene un valor de 1,16 con una desviación estándar de 1,09, por lo que podemos decir que los valores posibles mayormente están entre 0 y 2. El valor obtenido para su coeficiente asociado en el modelo quiere decir que, por cada incremento en una unidad de esta variable, las oportunidades de obtener un gel se multiplican por un factor

$$OR = exp(0{,}11) = 1{,}12$$

o sea, un incremento de aproximadamente 12% en las chances de obtener un gel con un mayor Physical Dipole respecto a las chances de no obtener gel, manteniendo las demás condiciones iguales.

En otro caso, el coeficiente para MOSCED mo-lambda (con una media de 7,42 y un rango de valores entre 0 y 10,57) es -0,04. Esto implica que, según el modelo, cada incremento en una unidad de esta variable implica multiplicar las oportunidades de obtener gel por un factor 0,96; o sea, una disminución de aproximadamente 4% en las chances de obtener un gel al aumentar el valor de MOSCED mo-lambda.

3.2.3.2. EVALUACIÓN DEL MODELO.

El modelo entrenado resuelve un problema de clasificación binaria, es decir, intenta predecir los casos positivos (clase 1, se obtiene un gel como resultado del proceso de gelificación) y los casos negativos (clase 0, no se obtiene un gel como resultado).

Mediante el método .predict() del modelo se realizan predicciones utilizando los datos prueba (test_x).

```
#Predicciones con los datos de prueba

predicciones = modelo_log.predict(test_x)
```

Figura 20 - Predicciones con datos de prueba.

Comparando estas predicciones con los resultados reales de los datos de prueba (test_y), se pueden obtener 4 resultados posibles:

- **Verdaderos Positivos** (True Positive): son los casos en donde realmente se produce un gel, y se lo predice como positivo (clase 1).
- **Falsos Positivos** (False Positive): son los casos en donde no se produce un gel pero se lo predice como positivo.
- **Verdaderos Negativos** (True Negative): son los casos en donde no se produce un gel y se los predice como negativos.
- **Falsos Negativos** (False Negative): son los casos en donde no se produce un gel pero se lo predice como positivo.

La **matriz de confusión** es una representación que se usa para mostrar estos valores (Figura 21).Los valores de la diagonal principal de dicha matriz (65, 82), indican las predicciones correctas para verdaderos positivos y verdaderos negativos. La diagonal secundaria (36, 33), muestra las predicciones incorrectas.

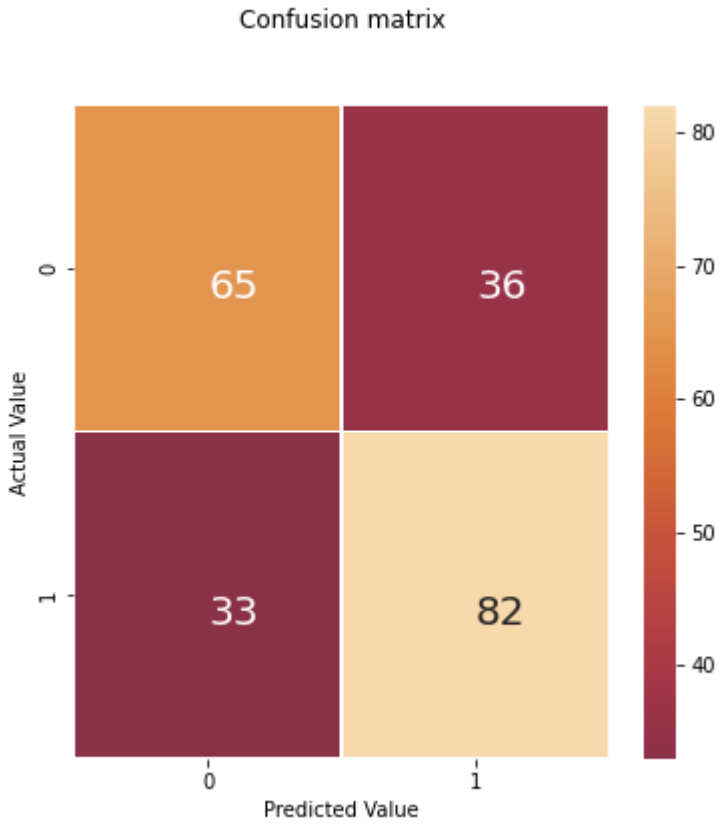


Figura 21 - Matriz de confusión.

Continuando con las tareas de evaluación del modelo se calculan las siguientes métricas:

- **Exactitud**: porcentaje de casos clasificados correctamente.
- **Precisión**: porcentaje de casos clasificados como positivos que realmente son positivos.
- **Sensibilidad**: porcentaje de casos positivos detectados o alcanzados por el modelo.

La precisión es una medida de la capacidad del modelo de encontrar solamente la categoría de interés, positiva, mientras que la sensibilidad es una medida de utilidad (cuántos del total de los casos positivos es capaz de encontrar el modelo). Ambas medidas son complementarias.

```
#Calculo de metricas - Exactitud, Precisión y Sensibilidad
print("Exactitud: %s"%(metrics.accuracy_score(test_y, predicciones),))
print("Precisión: %s"%(metrics.average_precision_score(test_y, predicciones),))
print("Sensibilidad: %s"%(metrics.recall_score(test_y, predicciones),))
```

```
Exactitud: 0.6805555555555556
Precisión: 0.6482825677556703
Sensibilidad: 0.7130434782608696
```

Figura 22 - Métricas del modelo.

En este caso, y considerando las características de los datos antes detalladas, se obtienen valores para las métricas de evaluación que, si bien no son extremadamente buenos, aún pueden constituir un avance. El valor de precisión indica que cuando el modelo predice una gelificación, acierta en casi un 65% de las veces, y es capaz de encontrar un 71% de las situaciones en que se produce gelificación (Figura 22). Que la calidad del modelo no sea mejor puede deberse a la cantidad de datos disponibles (mientras mayor es la cantidad, más tiene el modelo de dónde aprender), o como ya se mencionó anteriormente, a que no todos los registros presentan valores para la totalidad de las variables.

3.3. DESARROLLO APLICACIÓN WEB

Como producto de datos para este trabajo se eligió la creación de una aplicación Web, debido a las múltiples ventajas que ofrece frente a las aplicaciones de escritorio. Entre otras, podemos mencionar las siguientes: independencia respecto al sistema operativo, el no requerir proceso de instalación, y almacenamiento de copias de seguridad en los servidores o infraestructura cloud, accesibles en forma simultánea por varios usuarios.

3.3.1. ESTRUCTURA DE LA APLICACIÓN.

El lenguaje de programación Python cuenta con una gran cantidad de librerías y herramientas para Data Science y machine learning [Raschka y Mirjalili, 2017]. Esto, nos permite combinar de manera nativa librerías de machine learning con un framework de Python bien establecido para desarrollo web, Django. Este framework, como muchos otros, se encarga de gran parte de las complicaciones generales de desarrollo web, permitiendo al desarrollador concentrarse en las particularidades de la aplicación. Django sigue el patrón de diseño MTV, el cual agrupa las funcionalidades relacionadas en componentes reutilizables llamados Modelo, Vista y Template (plantilla).

Modelo: en esta capa de abstracción los modelos (clases Python) tienen como objetivo la comunicación de la aplicación con la base de datos. Los modelos cuentan con los métodos o funciones necesarias para la interacción con la base de datos (alta, baja y modificación de registros, consultas, etc), lo que permite abstraerse de la tecnología y particularidades del motor de base de datos.

Los modelos creados para la aplicación web son los siguientes:

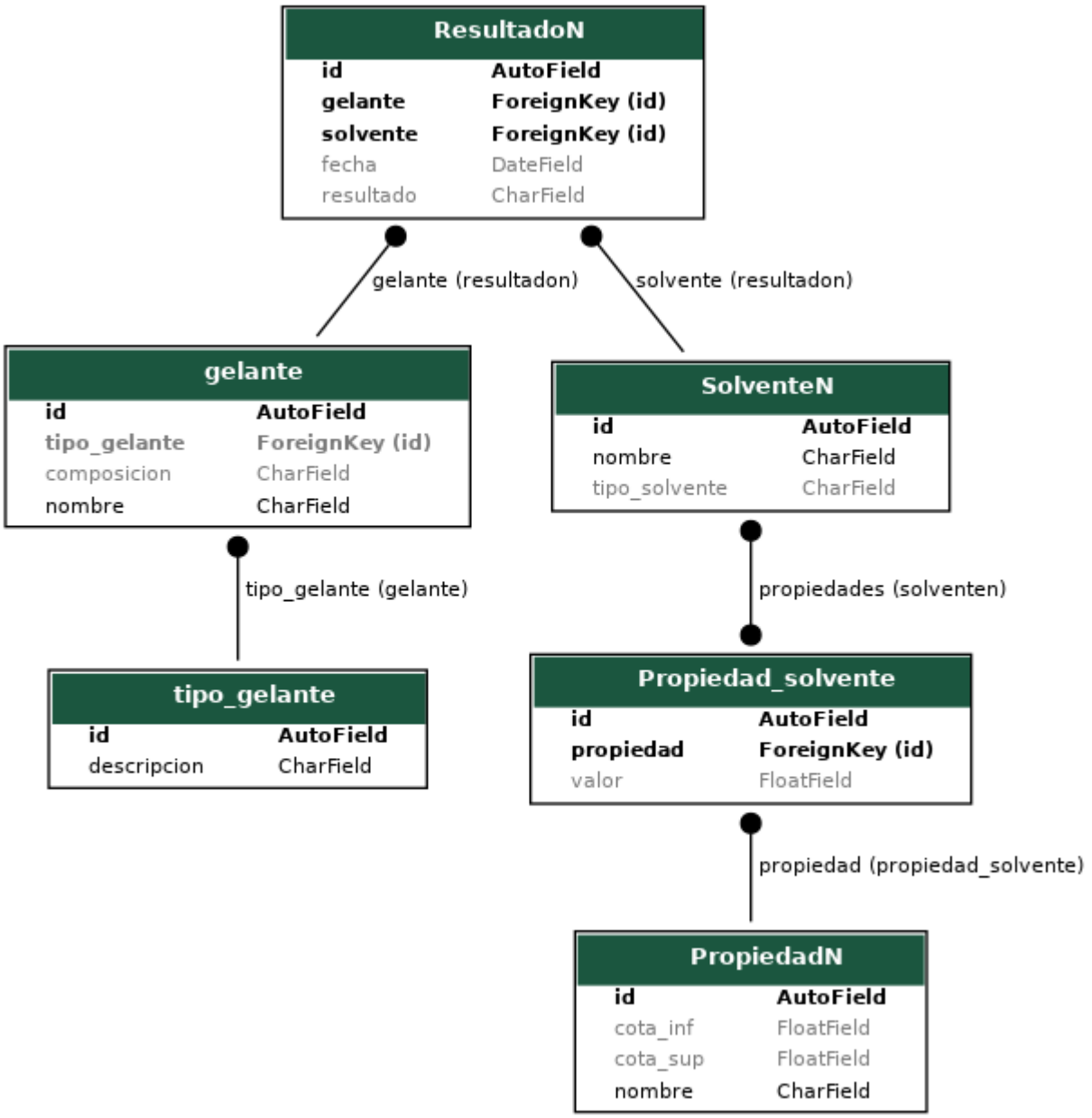


Figura 23 – Modelos Django de la aplicación web.

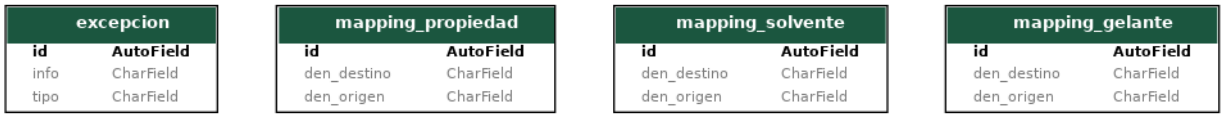


Figura 24 – Modelos Django utilizados para el proceso ETL.

Template: este componente contempla todo lo referido a la lógica de presentación (front end), es decir, la manera en que se presentan y muestran los datos extraídos de la base de datos a través de los modelos. La plantilla es básicamente un archivo HTML que también puede incluir código XML, CSS, JavaScript, entre otros.

Para este trabajo se emplea en particular la plantilla web Paper Dashboard, obtenible desde el sitio https://www.creative-tim.com/. Dicho sitio ofrece recursos y herramientas destinados al desarrollo web, donde algunos de los mismos se pueden utilizar de manera gratuita, aunque con funcionalidades limitadas. En nuestro caso se realizaron cambios en esta plantilla para adaptarla a las necesidades del proyecto (ver Figura 25).

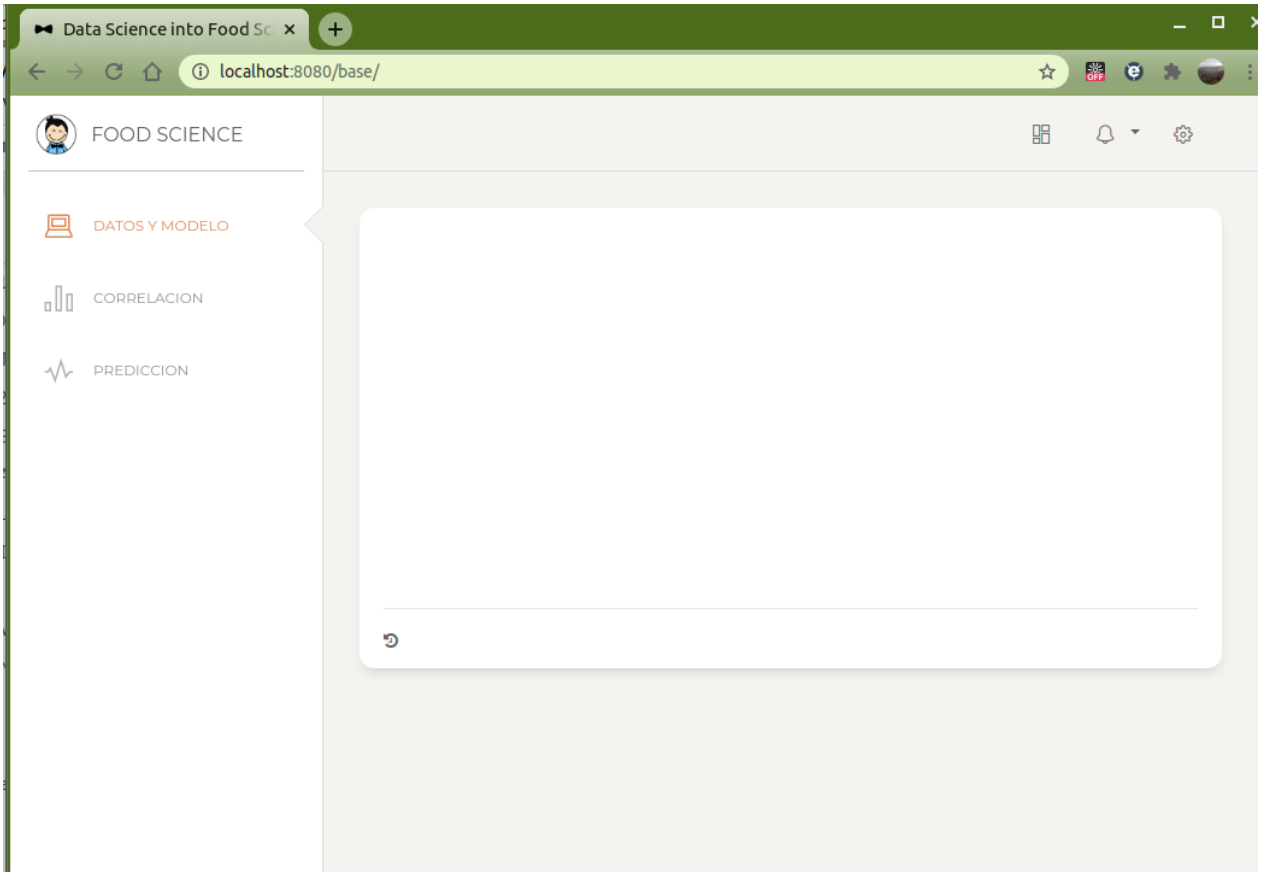


Figura 25 – Plantilla Paper Dashboard adaptada

Vistas: es la capa designada a la lógica de negocios, y la que suele realizar la mayor parte del trabajo de back end. La vista es simplemente una función Python que tiene como objetivo recibir los requerimientos enviados desde el navegador, si es necesario, realizar una petición al Modelo para extraer un valor de la base de datos, y finalmente mostrar los resultados a través de las Templates.

Los algoritmos de machine learning son provistos por la librería scikit-learn [Pedregosa y otros, 2011].

La aplicación se encuentra constituida por 3 vistas principales, donde cada una de ella se ocupa de proporcionar una funcionalidad determinada.

1) Datos y modelo.

   Esta vista se ocupa de mostrar información referente al modelo de predicción como así también del conjunto de datos utilizados para la construcción de este. Está constituida por las siguientes pestañas:

   a. Información del modelo.

   En esta sección se muestra el tipo de aprendizaje y el tipo de algoritmo utilizado para la construcción del modelo. A continuación de esto, el usuario puede observar las métricas de exactitud, precisión y sensibilidad, lo que le permitirá evaluar el modelo y determinar qué tan confiable es en sus predicciones.

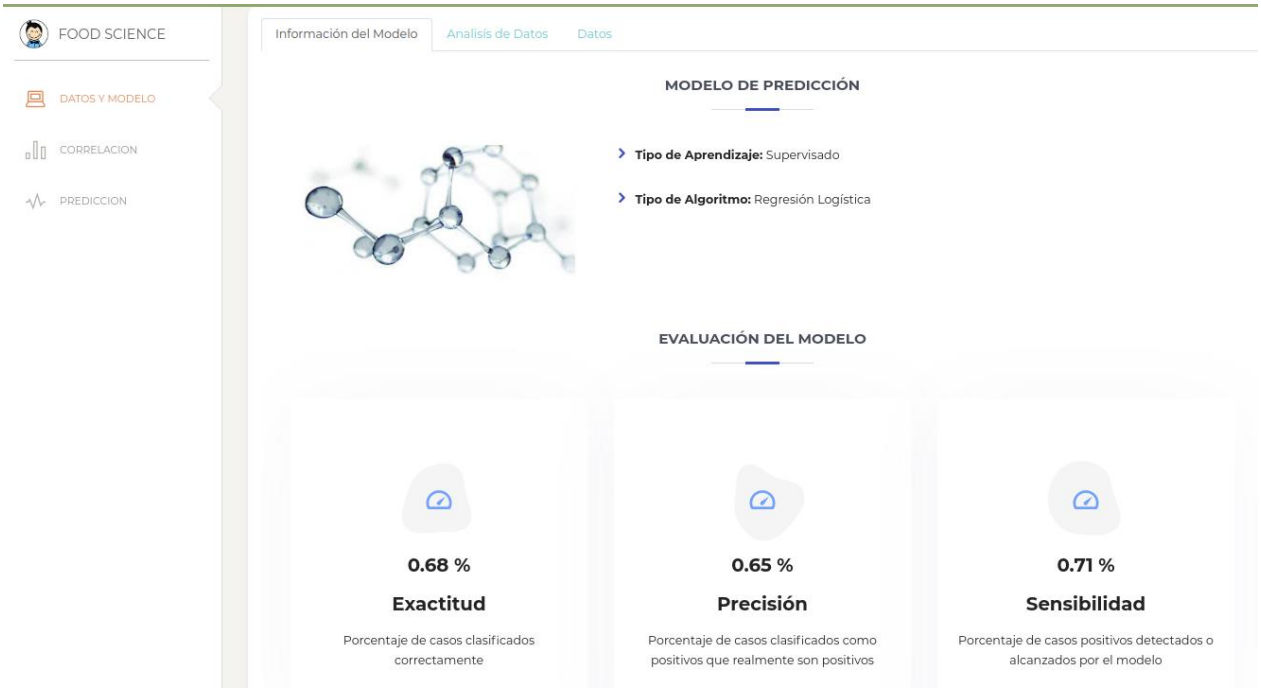


Figura 26 - Pestaña "Información del modelo"

b. Análisis de datos.

Esta pestaña tiene como objetivo mostrar información y representaciones gráficas de los predictores, lo que ayuda a comprender la naturaleza y la calidad de los mismos. Para cada uno de los predictores se muestra la cantidad de registros, media, desvío estándar, valor máximo, valor mínimo, gráfico de distribución y gráfico de cajas (Figura 27).

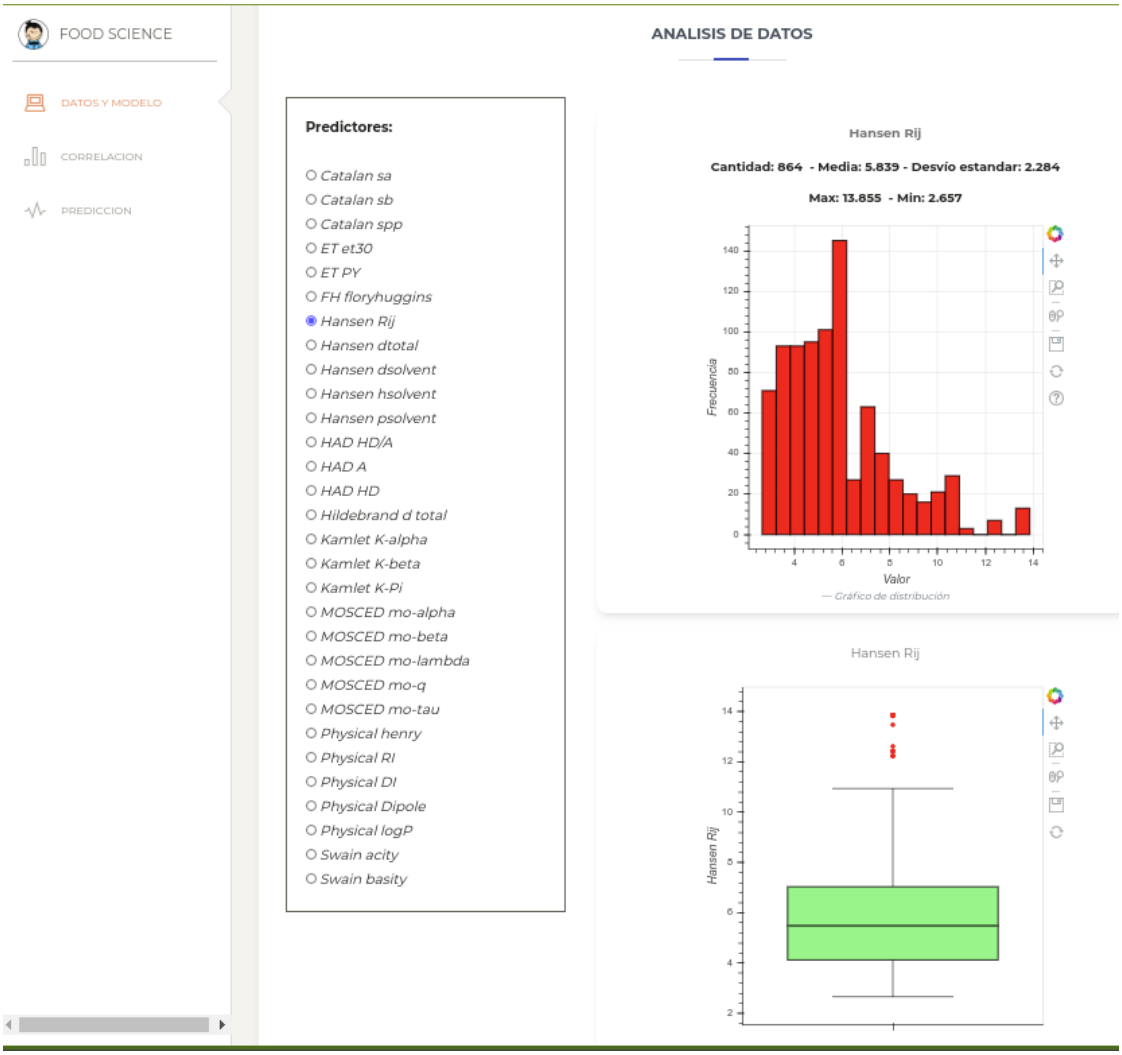


Figura 27 - Pestaña "Análisis de Datos"

c. Datos.

A través de esta opción, se podrá recorrer el conjunto de datos almacenado en la base de datos y realizar búsquedas, como así también, agregar, editar o eliminar algún registro.

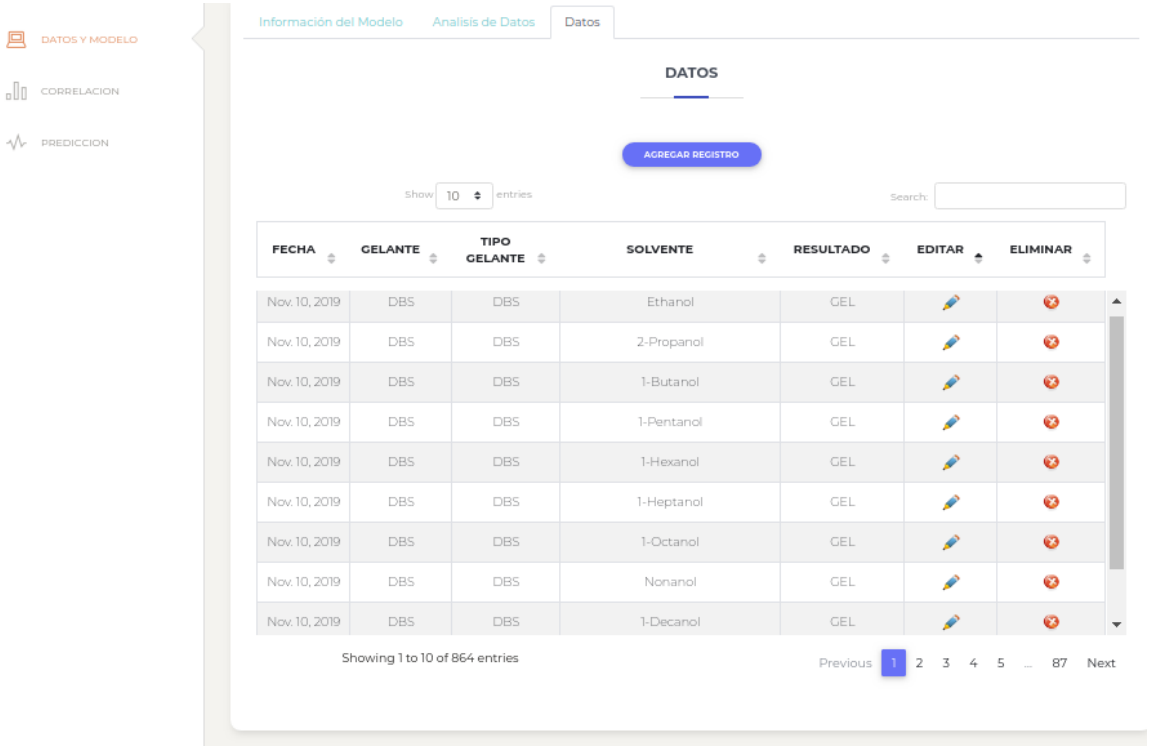


Figura 28 - Pestaña "Datos"

2) Correlación.

Esta vista le da la posibilidad al usuario de seleccionar una propiedad de solvente y mostrar que resultado de gelificación se obtuvo para los distintos valores de la propiedad. Los resultados se muestran en gráficos distintos dependiendo del tipo de gelante, permitiendo en cada gráfico elegir entre los distintos gelantes de ese tipo. Esto le permitirá al usuario observar de manera gráfica cómo se relacionan las distintas propiedades de solventes con los resultados del proceso de gelificación y los distintos gelantes.

Figura 29 - Gráficos de correlación.

3) Predicción.

Es la vista más importante de la aplicación, ya que esta funcionalidad está directamente relacionada con el objetivo de este trabajo. Se encarga de mostrar y dar la posibilidad de elegir el gelante y solvente, solicitando estos datos a los usuarios. Permite, además, modificar el valor de las propiedades del solvente, validando que estos valores no excedan la cota superior e inferior de cada propiedad (ver Figura 30). Finalmente, mediante la opción “Predecir” y utilizando el modelo de machine learning construido para este trabajo (ver sección Construcción del modelo), se muestra el resultado de la predicción al combinar esos elementos (Figura 31).

Figura 30 - Selección de gelante y solvente, edición de propiedades de solvente.

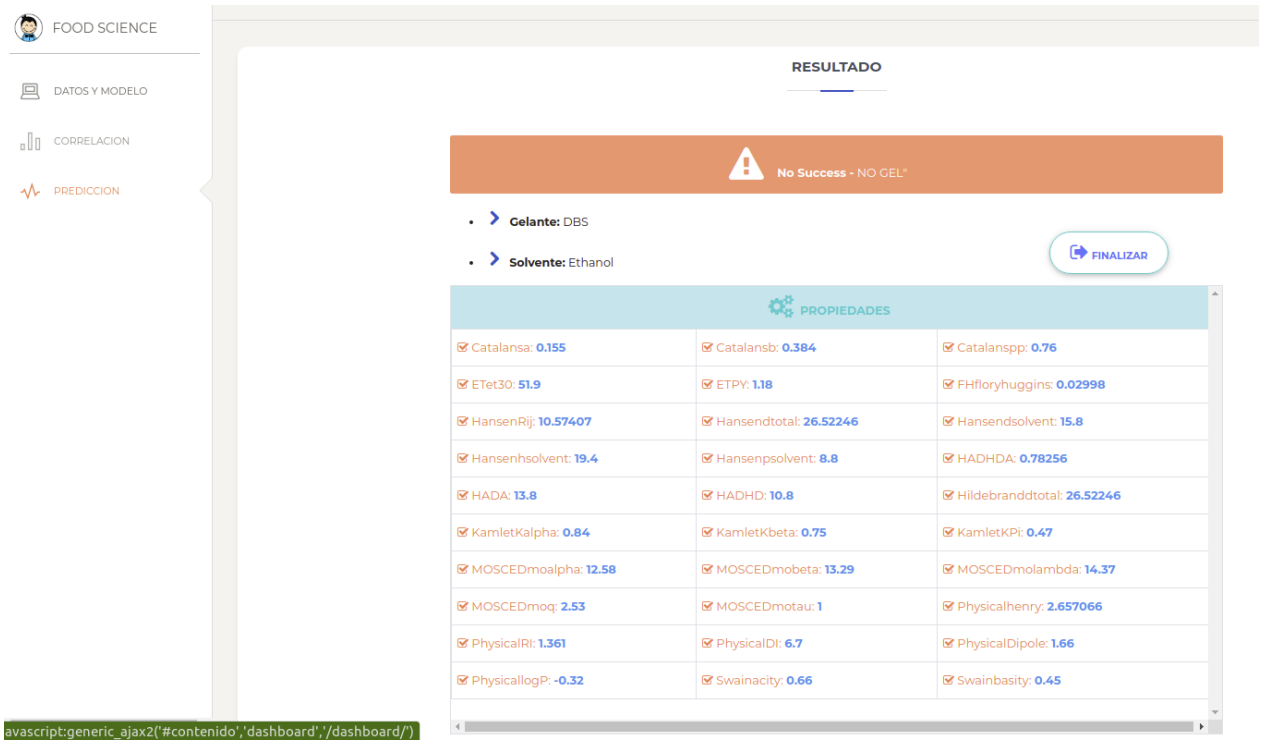


Figura 31 - Resultado de predicción.

3.4. INFRAESTRUCTURA DE IMPLEMENTACIÓN

Para la implementación de la aplicación web, se optó por seleccionar una infraestructura cloud IaaS (Infraestructure as a Service), teniendo en cuenta por un lado los requerimientos en cuanto a software y hardware necesarios para la solución, como así también a las ventajas que ofrecen este tipo de infraestructuras sobre las on premise.

IaaS provee, bajo demanda a los consumidores, las capacidades de procesamiento, almacenamiento, redes y funciones principales de cómputo, y puede incluir sistemas operativos y aplicaciones. Este tipo de servicio pone al alcance de muchas empresas pequeñas y PyMES ventajas y funcionalidades tecnológicas que de otra manera serían extremadamente costosas [Incibe, 2019].

Este tipo de modelo trae aparejado ventajas e inconvenientes como se muestran en las siguientes tablas.

Ventajas	
Reducción de costes	No requiere una gran inversión inicial en infraestructura. Se paga solamente por los recursos insumidos.
Menor esfuerzo en mantenimiento y gestión.	Todas estas tareas son cubiertas por el proveedor del servicio. Lo que le permite al consumidor enfocar sus esfuerzos en tareas propias de su negocio.
Mayor escalabilidad	Permite escalar la infraestructura de manera rápida y fácil aumentado las prestaciones para adecuarse a la creciente demanda.
Menores tiempo de implementación.	Los tiempos de implementación de estas soluciones se llevan a cabo en cuestión de horas o muchas veces de manera casi inmediata, dependiendo del nivel de personalización.
Fiabilidad	Su redundancia permite la continuidad y recuperación ante cualquier tipo de eventualidad.
Facilidad de acceso	Solo se necesita de un navegador y una conexión de internet para acceder al servicio desde cualquier lugar y momento.

Tabla 6. Ventajas de IaaS.

Desventajas	
Dependencia de internet	El acceso al servicio está condicionado a conexión de internet con la que se cuente.
Pérdida de control.	El cliente del servicio no tiene acceso a las instalaciones en donde se ejecutas las aplicaciones. Dejando los datos y aplicaciones en manos del proveedor.
Disponibilidad	La disponibilidad de un IaaS es únicamente función del proveedor del servicio con lo que, de ocurrir algún fallo, el usuario no lo puede solucionar sino deberá esperar a el administrador del servicio.

Tabla 7. Desventajas de IaaS.

3.4.1. PROVEEDOR DE SERVICIOS CLOUD.

Entre los principales proveedores que ocupan el mercado de servicios cloud se encuentran Amazon Web Services (AWS), Microsoft Azure y Google Cloud Platform(GCP), líderes en el mercado según Gartner en julio de 2019. (ver Figura 32).

Figura 32 – Cuadrante mágico de Gartner para IaaS a julio de 2019.

A continuación, se analiza las alternativas ofrecidas por estos proveedores, teniendo en cuenta el objetivo de este trabajo que es la implementación de una aplicación web.

3.4.1.1. AMAZON EC2

Amazon Elastic Compute Cloud (EC2) proporciona capacidad computacional en la nube de AWS, eliminando de esta manera la necesidad de una inversión inicial en hardware y

permitiendo el desarrollo e implementación de aplicaciones en menor tiempo. A través de este servicio se pueden lanzar tantos servidores virtuales como sea necesario, configurando la seguridad, redes y almacenamiento de cada uno de ellos. Amazon EC2 permite escalar hacia arriba o abajo para adaptarse a los cambios de requerimientos o cambios en la demanda del servicio [Amazon, 2019].

Principales características ofrecidas por Amazon EC2:

- Entornos informáticos virtuales (instancias).
- Plantillas preconfiguradas para las instancias, conocidas como imágenes de maquina Amazon (AMI), que empaqueta las partes necesarias para un servidor, incluyendo sistema operativo y software adicional.
- Varias configuraciones de CPU, memoria, almacenamiento y capacidad de red la instancia.
- Información de inicio de sesión segura para las instancias con pares de claves (AWS almacena la clave pública y usted guarda la clave privada en una ubicación segura).
- Un firewall que permite especificar los protocolos, los puertos y los rangos de direcciones IP que pueden alcanzar las instancias mediante el uso de grupos de seguridad.

3.4.1.2. AZURE VIRTUAL MACHINE.

Azure Virtual Machine permite ejecutar software y aplicaciones informática de alto rendimiento, dando la posibilidad de elegir la distribución preferida de Linux o Windows Server [Microsoft Azure, 2019].

Principales características de Azure Virtual Machine:

- Escalar de una a miles de instancias en cuestión de minutos con el servicio Virtual Machine Scale Sets.

- En cuestiones de seguridad permite cifrar la información confidencial, proteger las máquinas virtuales frente a amenazas malintencionadas, proteger el tráfico de la red y satisfacer un gran número de normas internacionales.
- Permite elegir entre imágenes de máquinas virtuales de plataformas Linux o Windows, y los modelos de consumo que mejor se adapten a las necesidades del cliente.

3.4.1.3. GOOGLE COMPUTE ENGINE.

Google Compute Engine es el componente de Infraestructura como Servicio de Google Cloud Platform, permite a los usuarios lanzar máquinas virtuales a pedido y dándole la posibilidad de elegir entre plantillas de configuración predefinidas (combinaciones de uso general de memoria y CPU) o maquinas personalizadas, seleccionando el número de CPUs y cantidad de memoria necesarias para satisfacer su carga de trabajo [Google Cloud, 2019].

Características principales de Google Compute Engine:

- Permite ejecutar sistemas operativos Linux y Windows, como así también usar una imagen compartida por la comunidad de Google Cloud o una propia del usuario.
- Migración activa de máquinas virtuales entre sistemas host sin necesidad de reiniciar, y de esta manera no se interrumpiría la ejecución de las aplicaciones.
- Integración con servicios de Google Cloud como los de inteligencia artificial (IA), aprendizaje automático (ML) y analítica de datos.
- Las máquinas virtuales son protegidas contra defectos y vulnerabilidades con el servicio de Gestión de Parches de SO, permitiendo recibir todos los parches del entorno, aplicar los parches de sistema operativo a todo el conjunto de máquinas virtuales o automatizar su instalación.
- Permite añadir GPUs para acelerar las cargas de trabajo que consuman muchos recursos computaciones. Pueden ser agregas y quitadas en cualquier

momento en función a la necesidad del cliente y pagar por el recurso solo cuando lo utilice.

Luego de analizar las alternativas de estos proveedores, se observa desde el punto de vista funcional y considerando los requerimientos de nuestro desarrollo, que la aplicación web podría ser implementado de igual manera en cualquiera de ellas.Sin embargo, AWS presenta una alternativa llamada AWS Educate, la cual ofrece a instituciones educativas, docentes y estudiantes un conjunto de recursos orientados a desarrollar habilidades en la nube, tales como acceso a contenidos, formación técnica, métodos o servicios de AWS. Todo esto sin costo, o crédito disponible para el caso de algunos servicios [Amazon Educate, 2019]. Aprovechando que UADE es miembro de AWS Educate y que como alumno de la institución el autor puede hacer uso de esto beneficios, se opta por el servicio **Amazon EC2** para la implementación de la aplicación web.

3.4.2. DESPLIEGUE DE LA INFRAESTRUCTURA

Para implementar la aplicación web se utiliza el servicio de Amazon Web Services **EC2** (Elastic Compute Cloud) y se crea una instancia de un servidor virtual Ubuntu (Figura 33).

Recursos de la instancia Amazon EC2:

- Sistema Operativo: Ubuntu Server 18.04.2 LTS
- Procesador: Intel(R) Xeon(R) CPU E5-2676 v3 @ 2.40GHz
- Memoria: 1007524 kB
- Disco: ext4 8G

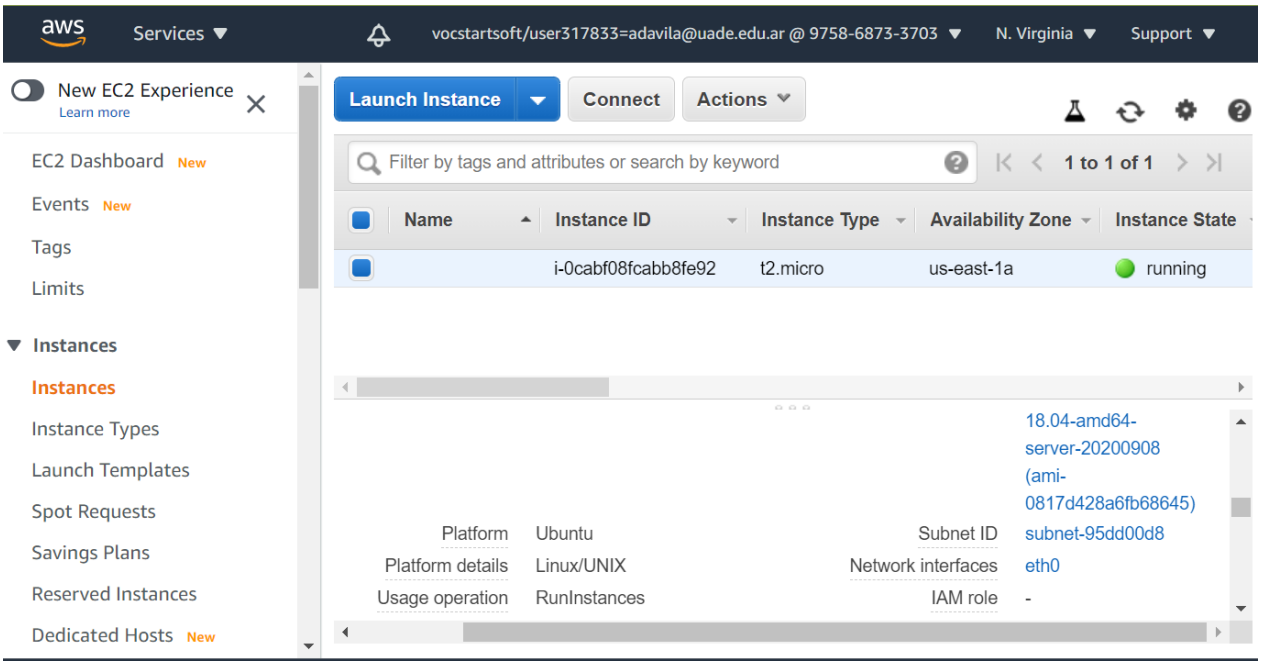


Figura 33 – Instancia EC2 Ubuntu.

Posteriormente se instalan en el servidor virtual los paquetes de software necesarios para poner en funcionamiento los servicios Web HTTP “Apache2” y de base de datos “MySQL” (ver figura 34 y 35). Asimismo, se instala el framework web Django y las librerías Python para machine learning.

Lista de paquetes de software instalados

- apache2 - Apache HTTP Server
- apache2-utils - Apache HTTP Server (utility programs for web servers)
- libapache2-mod-wsgi-py3 - Python 3 WSGI adapter module for Apache
- mysql-server-5.7 - MySQL database server binaries and system database setup
- mysql-client-5.7 - MySQL database client binaries
- phpmyadmin - MySQL web administration tool
- Django 2.2.1 - High-level Python web development framework (Python 3 version)
- scikit-learn 0.21.2 - Python modules for machine learning and data mining

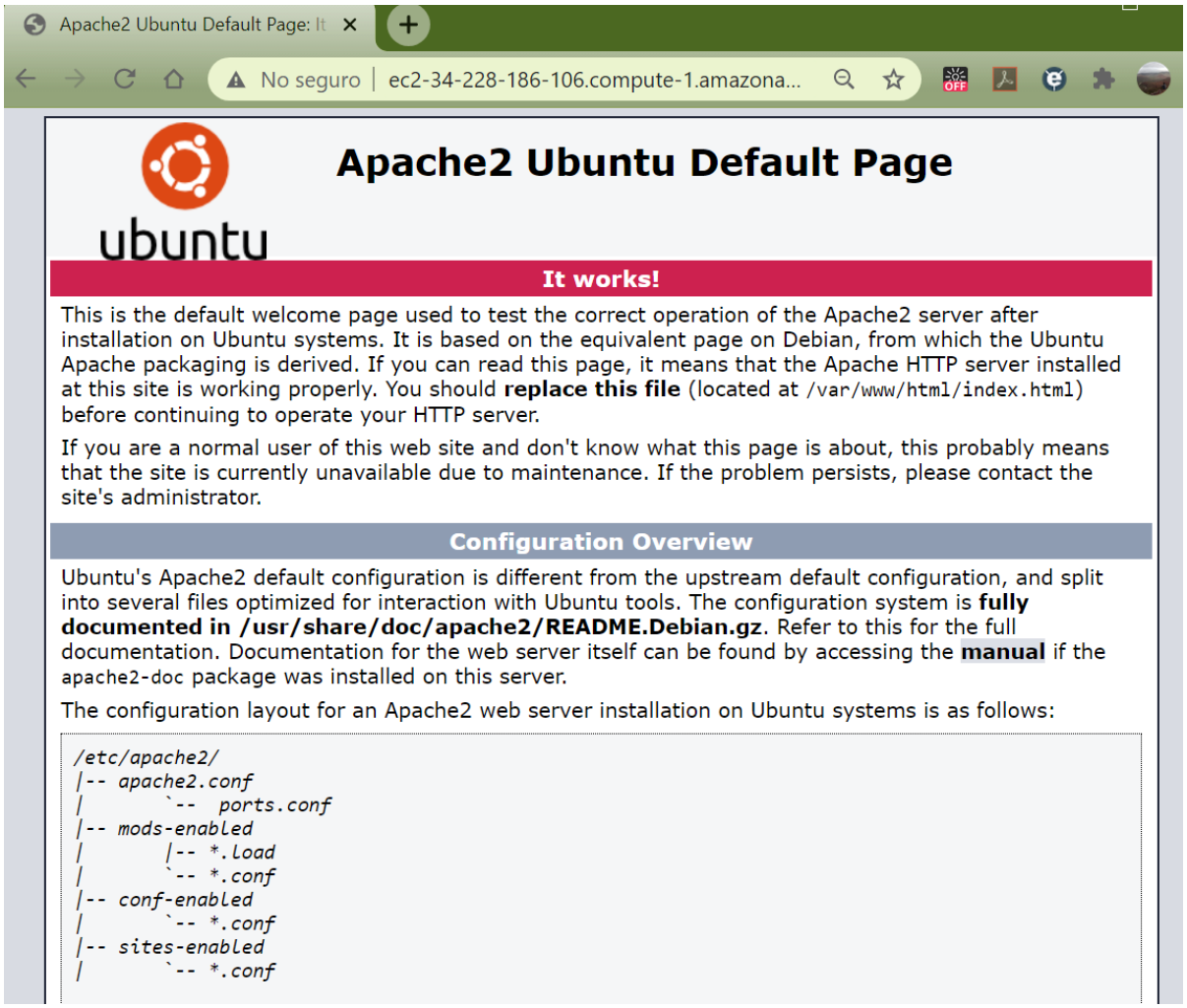


Figura 34 - Página de bienvenida del servidor Apache2, muestra la correcta operación del servicio.

Se crea la base de datos y exportan los datos utilizando el script descripto en la sección extracción, transformación y carga.

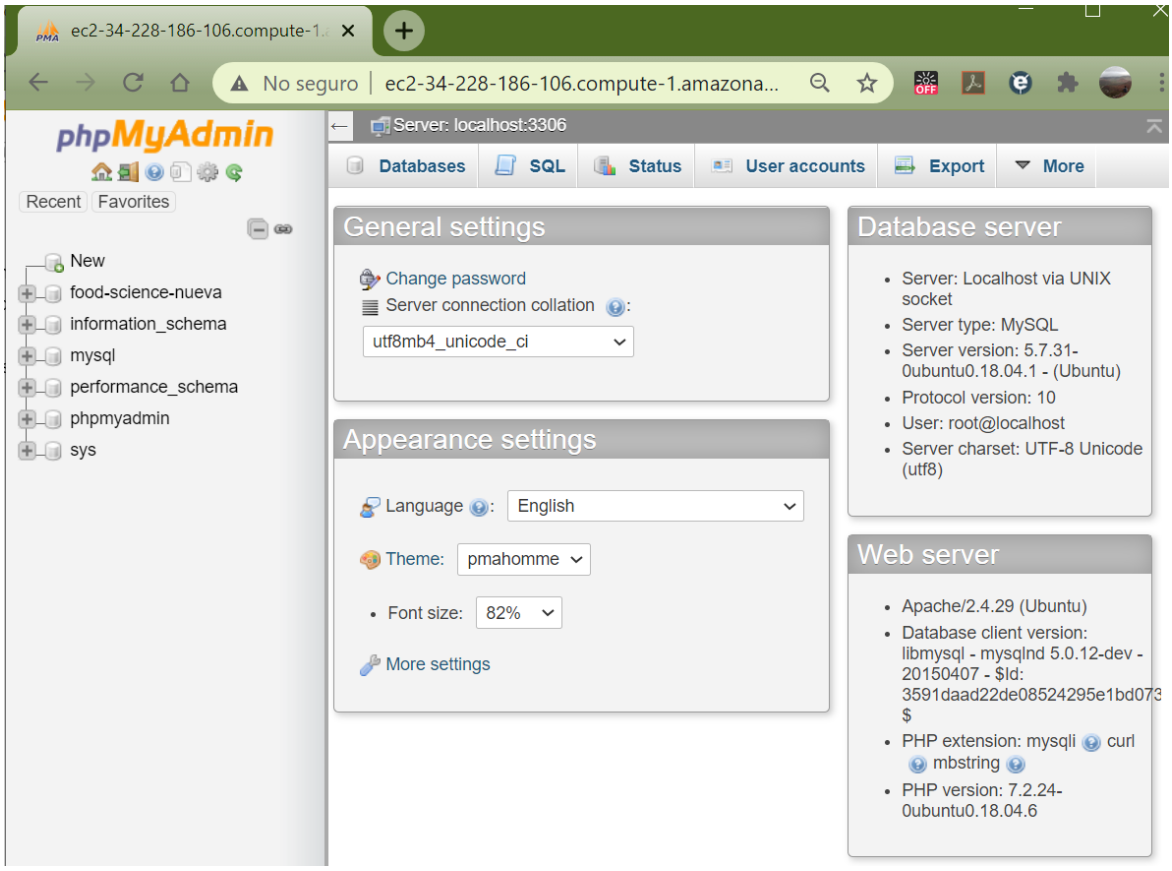


Figura 35 - Interfaz de administración web para MySQL.

Una vez instalados los servicios y paquetes que se mencionaron anteriormente y cargada una copia del código fuente, el servidor virtual ya se encuentra en condiciones de ejecutar la aplicación (Figura 36).

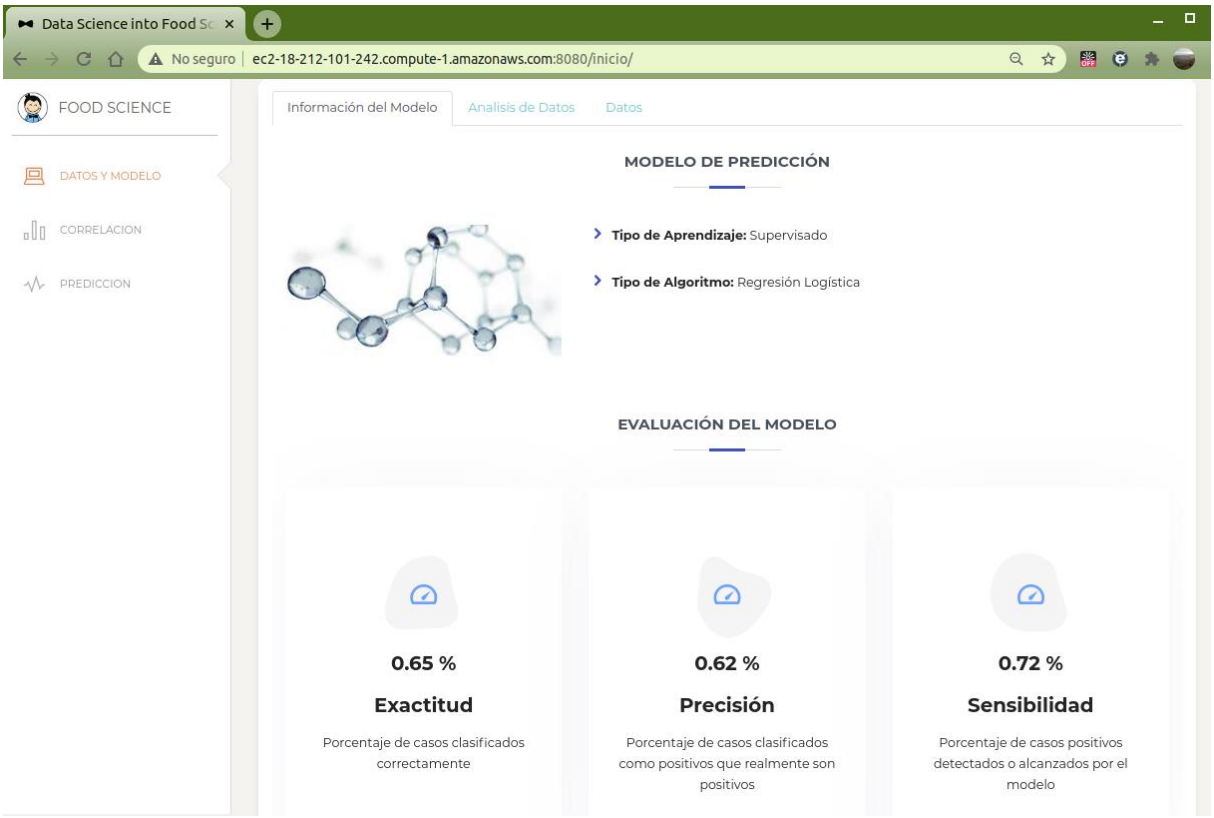


Figura 36 - Aplicación Web en infraestructura Amazon.

4. CONCLUSIONES

4.1. DISCUSIÓN - RESUMEN FINAL

El objetivo que motivó este trabajo fue desarrollar una aplicación, que utilizando los datos disponibles en el proyecto *Data Science into Food Science*, fuera capaz de generar una predicción sobre si una combinación determinada de gelante y solvente formaría o no un gel molecular. Para ello se almacenaron los datos en una base de datos MySQL, se construyó una interfaz web utilizando el framework de desarrollo Django, se generó y entrenó un modelo de regresión logística con las herramientas proporcionadas por la librería scikit-learn para la clasificación del resultado y por último se implementó la aplicación en una IaaS (Infraestructura como Servicio) del proveedor de servicio Cloud Amazon Web Services.

Los datos disponibles en la actualidad no son abundantes y asimismo presentan faltantes; esto es una posible causa de que las métricas del modelo predictivo no sean de gran calidad. Sin embargo, dada la gran complejidad del problema encarado y las posibilidades a futuro si se adopta una metodología sistemática de colección de información, es posible que ya estos modelos sencillos constituyan un aporte a la mejora de los procesos de investigación, comparados con los laboriosos procesos manuales existentes hoy. Por esto consideramos que, a pesar de las limitaciones, este trabajo constituye un aporte importante al ámbito de la TIA, ya que se provee una infraestructura construida e implementada que constituye una base para la predicción sistemática de gelificación de organogelantes aplicando metodologías de Ciencia de Datos, con la posibilidad de ser mejorada y optimizada mediante futuros desarrollos.

4.2. FUTURAS LÍNEAS DE INVESTIGACIÓN

El presente trabajo puede ser continuado con las siguientes líneas de investigación o evolución:

- Optimizar el modelo utilizando otros algoritmos de machine learning, buscando aumentar la precisión de las predicciones como así también contemplar los tres resultados posibles del proceso de gelificación (GEL, PRECIPITATE, SOLUTION).

- Incorporar nuevas fuentes de datos, como por ejemplo la estructura química de los gelantes utilizando la notación SMILES (Simplified Molecular-Input Line-Entry System), lo que permitiría por un lado aumentar la cantidad de información en las fuentes de datos, y por otro determinar cómo se relacionan estos datos con el resultado de la gelificación.

- Construir una interfaz que posibilite consultar los datos directamente de la “Arquitectura Big Data para el comportamiento de organogelantes”, la cual fue desarrollada como Trabajo Final de Maestría TIC por Verónica Cuello y se encuentra implementada como MVP.

5. BIBLIOGRAFÍA

Amazon, Amazon EC2, [En línea] 2019, [Consultado el 9 de agosto de 2019]. https://aws.amazon.com/es/ec2/.

Amazon Educate, AWS Educate, [En línea] 2019, [Consultado el 9 de agosto de 2019]. https://aws.amazon.com/es/education/awseducate.

Beck DAC, Carothers JM, Subramanian VR, Pfaendtner J. "Data Science: Accelerating innovation and discovery in chemical engineering" *AIChE Journal* 2016, 62:1402-1416.

Bonnet J, Suissa G, Raynal M, Bouteiller L. "Organogel formation rationalized by Hansen solubility parameters: dos and don'ts". Soft Matter 2014, 10:3154-3160.

Chang, R. Química (Novena Edición). México, McGraw Hill, 2007.

Christensen J, Nørgaard L, Bro R and Engelsen SB. "Multivariate autoflu- orescence of intact food systems". *Chemical Reviews* 2006, **106**:1979 – 1994.

Corradini Maria G, Gozzi Marta, Santo Domingo Cinthia, Tecce Tomás, Zarza Gonzalo. "Ciencias de datos: Valor tecnológico para el profesional en alimentos". *Énfasis Alimentación* 2017, 32-34.

Corradini MG, Rogers MA. "Molecular gels: Improving selection and design through computational methods" *Current Opinion in Food Science* 2016, 9:84-92

da Silva CET, Filardi VL, Pepe IM, Chaves MA, Santos CMS. "Classification of food vegetable oils by fluorimetry and artificial neural networks" *Food Control* 2015, 47:86-91

Donno D, Boggia R, Zunin P, Cerutti AK, Guido M, Mellano MG, Prgomet Z, Beccaro GL. "Phytochemical fingerprint and chemometrics for natural food preparation pattern

recognition: an innovative technique in food supplement quality control" *Journal of Food Science and Technology-Mysore* 2016, 53:1071-1083.

Gad Haidy, Sherweit H. El-Ahmady, Mohamed I. Abou-Shoer, Mohamed M. Al-Azizi "Application of chemometrics in authentication of herbal medicines" Phytochem Analysis 2013, 24:1–24.

Google Cloud, Compute Engine, [En línea] 2019. [Consultado el 9 de agosto de 2019] https://cloud.google.com/compute.

Hansen CM, Yamamoto H. "Hansen Solubility Parameters in Practice". 2013.

Hernandez Sampieri, R, Fernández Collado, C y Baptista Lucio, M. 2010. *"Metodología de la Investigación"*. 5ta. ed. McGraw-Hill, 2010.

Incibe, Cloud Computing, Instituto nacional de ciberseguridad de España, [En línea] 2019. [Cosultado el: 9 de agosto de 2019].https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia-cloud-computing_0.pdf.

Kendall, K y Kendall, J. 2011. *"Análisis y Diseño de Sistemas".* Pearson, 2011.

Lan Y, Corradini MG, Liu X, May TE, Borondics F, Weiss RG, Rogers MA. "Comparing and correlating solubility parameters governing the self-assembly of molecular gels using 1,3:2,4-dibenzylidene sorbitol as the gelator". Langmuir 2014, 30:14128-14142.

Microsoft Azure, Virtual Machines, [En línea] 2019, [Consultado el 9 de agosto de 2019], https://azure.microsoft.com/es-es/services/virtual-machines/.

McKinney, W. "pandas: a Foundational Python Library for Data Analysis and Statistics", 2011, contributed paper to PyHPC.

Medina Fernando, Galvan Marcos. “Estudios estadisticos y prospectivos”, CEPAL Santiago de Chile 2007.

Mermelstein NH. “Big data gets bigger and better” *Food Technology* 2017, 71:118-122.

Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M. y Duchesnay E. “Scikit-learn: Machine Learning in Python”, Journal of Machine Learning Research 2011, 2825-2830

Peng L, Wang YZ, Zhu HB, Chen QM. “Fingerprint profile of active components for Artemisia selengensis Turcz by HPLC-PAD combined with chemometrics”. *Food Chemistry* 2011, 125:1064–1071

Pigott Therese D, *“A Review of Methods for Missing Data”.* Educational Research and Evaluation 2001, 4:353-383

Provost, F. y Fawcett, T. *“Data Science for Business: What You Need to Know About Data Mining and Data-Analytic Thinking”*. O'Reilly, 2013

Raschka S. y Mirjalili V. Python Machine Learning (Second Edition). Birmingham, Packt Publishing Ltd., 2017

Trivittayasil V, Tsuta M, Imamura Y, Sato T, Otagiri Y, Obata A, Otomo H, Kokawa M, Sugiyama J, Fujita K, et al. “Fluorescence fingerprint as an instrumental assessment of the sensory quality of tomato juices” *Journal of the Science of Food and Agriculture* 2016, 96:1167-1174

Turban, E., Sharda, R., Aronson, J. & King D. *“Business Intelligence, A Managerial Approach”*. Prentice-Hall, 2017

Weininger D. “SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules”. Journal of Chemical Information and Modeling, 1988, 28:31-36

Zumel N. y Mount, J. “Practical Data Science with R”. Manning Publications, 2014.

6. ANEXOS

6.1. DIAGRAMA DE PROCESAMIENTO ETL.

El siguiente diagrama de flujo fue propuesto por Verónica Cuello en su trabajo “Arquitectura Big Data para análisis de organogelantes”, para el tratamiento de los datos obtenidos desde el origen.

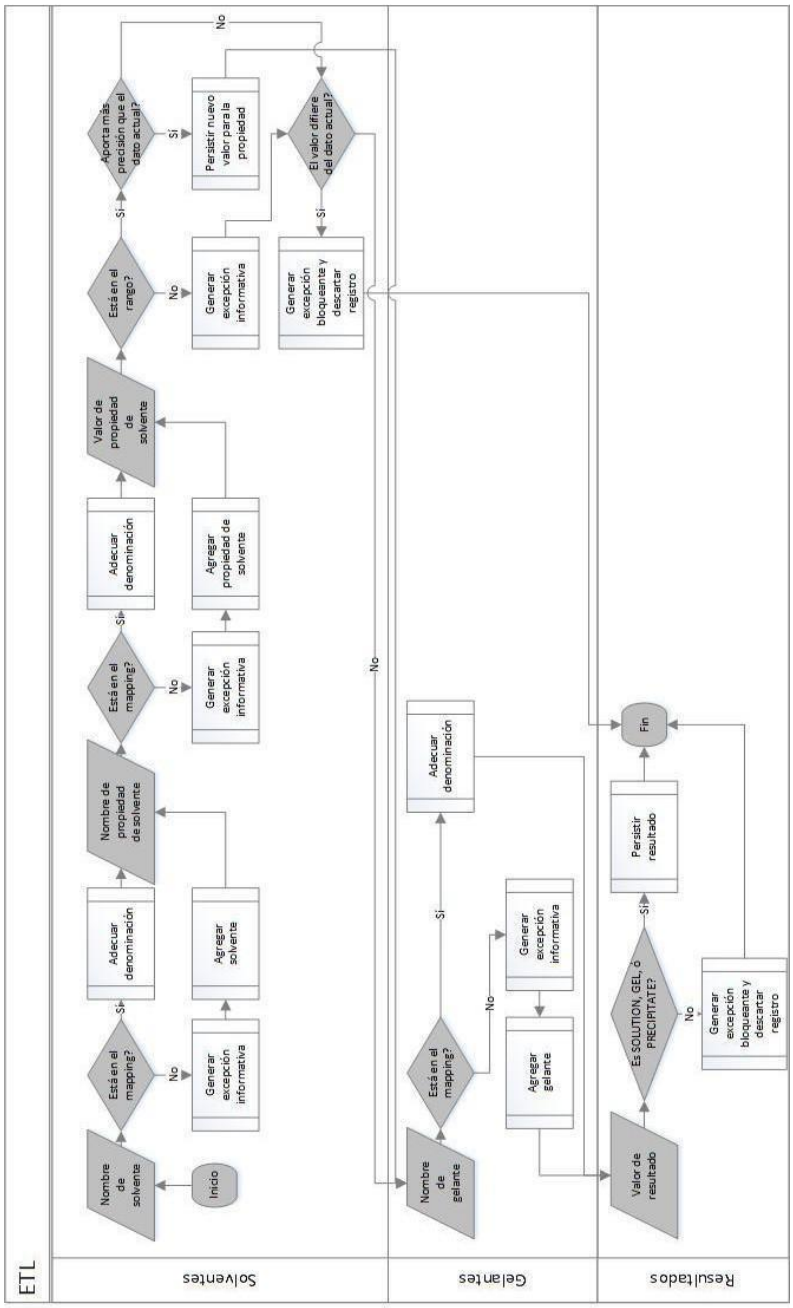


Figura 37 – Diagrama de procesamiento ETL.

6.2. TABLAS MAPPING

En las siguientes tablas, también elaboradas por Verónica Cuello, se pueden observar los mapping definidos para los distintos datos de origen. Para cada tabla, la "denominación de origen" refiere al dato que puede llegar a través de un determinado origen de datos. La "denominación destino" refiere a el dato que reemplazará al de origen en el procesamiento, en caso de que exista el dato en el mapping.

Denominación de origen	Denominación destino
acetic acid	Acetic Acid
acetic anhydride	Acetic Anhydride
acetone	Acetone
acetonitrile	Acetonitrile
acetronitrile	Acetonitrile
aniline	Aniline
Benzaldehyde	Benzaldehyde
benzene	Benzene
Benzyl Alcohol	Benzyl Alcohol
benzyl alcohol	Benzyl Alcohol
benzyl Benzoate	Benzyl Benzoate
benzyl Benzoate*	Benzyl Benzoate
14-Butanediol	14-Butanediol
14-Butanediol*	14-Butanediol
Butanenitrile	Butanenitrile
Butanenitrile*	Butanenitrile
1-Butanol	1-Butanol

1-butanol	1-Butanol
butanol	1-Butanol
butyl methacrylate	Butyl Methacrylate
carbon disulfide	Carbon Disulfide
ccl2	CCl2
CCl3	CCl3
ccl3	CCl3
CCl4	CCl4
ccl4	CCl4
cresol	Cresol
cyclohexane	Cyclohexane
Decanal	Decanal
Decanal*	Decanal
Decane	Decane
decane	Decane
1-Decanethiol	1-Decanethiol
1-Decanol	1-Decanol
1-decanol	1-Decanol
2-decanone	Decanal
Decene	Decene
1,2, dichloroethane	1,2-Dichloroethane
1,2,dichloroethane	1,2-Dichloroethane
diethyl ether	Diethyl Ether
diethylamine	Diethylamine
Diethylene glycol	Diethylene Glycol
Diethylene Glycol Butyl Ether	Diethylene Glycol Butyl Ether

diethylether	Diethyl Ether
1,2 dimethoxyethane	1,2-Dimethoxyethane
26-Dimethyl-4-heptanol	2,6-Dimethyl,4-Heptanol
dioxane	Dioxane
diphenyl ether	Diphenyl Ether
diphenylether	Diphenyl Ether
dipropyl ether	Dipropyl Ether
DMF	DMF
dmf	DMF
DMSO	DMSO
dmso	DMSO
Dodecane	Dodecane
dodecane	Dodecane
Ethanenitrile	Acetonitrile
Ethanol	Ethanol
ethanol	Ethanol
2-ethanolamine	2-Ethanolamine
Ethyl Acetate	Ethyl Acetate
ethyl acetate	Ethyl Acetate
ethyl formate	Ethyl Formate
ethyl malonate	Ethyl Malonate
Ethyl Oleate	Ethyl Oleate
Ethylene carbonate	Ethylene Carbonate
Ethylene Glycol	Ethylene Glycol
ethylene glycol	Ethylene Glycol
Ethylene Glycol Butyl Ether	Ethylene Glycol Butyl Ether

Ethylene glycol monoethyl ether	Ethylene Glycol Monoethyl Ether
Eugenol	Eugenol
Formamide	Formamide
Formic Acid	Formic Acid
Formic Acid*	Formic Acid
Glycerol	Glycerol
glycerol	Glycerol
Guaiacol	Guaiacol
heptane	Heptane
heptane	Heptane
1-heptanol	1-Heptanol
4-heptanol	4-Heptanol
1-Heptanol*	1-Heptanol
2-Heptone	2-Heptone
4-Heptone	4-Heptone
1-Heptylamine	1-Heptylamine
hexadecane	Hexadecane
Hexanenitrile	Hexanenitrile
hexanoic acid	Hexanoic Acid
1-Hexanol	1-Hexanol
1-hexanol	1-Hexanol
1-Hexanol*	1-Hexanol
isoamyl alcohol	Isoamyl Alcohol
Isobutanol	Isobutanol
isooctane	Isooctane
methanol	Methanol

methyl acetate	Methyl Acetate
methyl acrylate	Methyl Acrylate
methyl ethyl ketone(2-butanone)	Methyl Ethyl Ketone (2-Butanone)
Methyl methacrylate	Methyl Methacrylate
methyl methacrylate	Methyl Methacrylate
1-Methyl Naphthalene	1-Methyl Naphthalene
Methyl Oleate	Methyl Oleate
1-methyl-2-prrolidone	NMP
methylcyclohexane	Methylcyclohexane
n hexane	n-Hexane
n-hexane	n-Hexane
nitrobenzene	Nitrobenzene
NMP	NMP
nmp	NMP
Nndimethyl acetamide	Nndimethyl Acetamide
Nonanal	Nonanal
Nonanal*	Nonanal
Nonane	Nonane
Nonanenitrile	Nonanenitrile
1-Nonanol	Nonanol
1-nonanol	Nonanol
5-Nonanone	Nonanone
Octanal	Octanal
Octanal*	Octanal
1-Octanamine	1-Octanamine
Octane	Octane

octane	Octane
Octanenitrile	Octanenitrile
1-Octanol	1-Octanol
1-octanol	1-Octanol
2-octanol	2-Octanol
1-Pentanethiol	1-Pentanethiol
1-Pentanol	1-Pentanol
1-pentanol	1-Pentanol
2-Pentanone	2-Pentanone
3-Pentanone	3-Pentanone
phenylcyclohexane	Phenylcyclohexane
1,2-propanediol	1,2-Propanediol
12-Propanediol	1,2-Propanediol
13-propanediol	1,3-Propanediol
Propanenitrile	Propanenitrile
Propanenitrile*	Propanenitrile
1-propanol	1-Propanol
2-propanol	2-Propanol
2-Propanone	2-Propanone
propylamine	Propylamine
Pyridazine	Pyridazine
pyridine	Pyridine
styrene	Styrene
Sulfolane	Sulfolane
tert-amyl alcohol	Tert-Amyl Alcohol
tert-butyl alcohol	Tert-Butyl Alcohol

Tetradecane	Tetradecane
tetraethoxylsilane	Tetraethoxysilane
tetraethoxysilane	Tetraethoxysilane
tetrahydrofuran	Tetrahydrofuran
toluene	Toluene
2-tridecanol	2-Tridecanol
Triethylamine	Triethylamine
triethylamine	Triethylamine
triethylene glycol	Triethylene Glycol
triethylsilane	Triethylsilane
222 trifluoroethanol	2,2,2-Trifluoroethanol
trimethylpentanol	Trimethylpentanol
6-Undecanone	6-Undecanone
Water	Water
water	Water
xylene	Xylene

Tabla 8 - Mapping de Solventes

Denominación de origen	Denominación destino
sa	Catalan sa
sb	Catalan sb
spp	Catalan spp
et30	ET et30
PY	ET PY
floryhuggins	FH floryhuggins
Rij	Hansen Rij

dtotal	Hansen dtotal
dsolvent	Hansen dsolvent
hsolvent	Hansen hsolvent
psolvent	Hansen psolvent
HD/A	HAD HD/A
A	HAD A
HD	HAD HD
d total	Hildebrand d total
K-alpha	Kamlet K-alpha
K-beta	Kamlet K-beta
K-Pi	Kamlet K-Pi
mo-alpha	MOSCED mo-alpha
mo-beta	MOSCED mo-beta
mo-lambda	MOSCED mo-lambda
mo-q	MOSCED mo-q
mo-tau	MOSCED mo-tau
henry	Physical Henry
RI	Physical RI
DI	Physical DI
Dipole	Physical Dipole
logP	Physical logP
kirk	Physical Kirk
acity	Swain acity
basity	Swain basity

Tabla 9 - Mapping propiedades de solvente.

Denominación de origen	Denominación destino
Azo Bu	Azo Bu
Azo Dec	Azo Dec
Azo Et	Azo Et
Azo Me	Azo Me
Azo Pe	Azo Pe
Azo Pr	Azo Pr
ALS1	ALS1
ALS6	ALS6
ALS8	ALS8
ALS9	ALS9
ALS10	ALS10
ALS 1	ALS1
ALS 6	ALS6
ALS 8	ALS8
ALS 9	ALS9
ALS 10	ALS10
DBS	DBS
HSA	HSA
DBU	DBU
DCHU	DCHU
CAB	CAB
Sugar 01	Sugar 01
Sugar 02	Sugar 02
Sugar 06	Sugar 06

Sugar 09	Sugar 09
Sugar 10	Sugar 10
Sugar Nitro	Sugar Nitro
Sugar01	Sugar 01
Sugar02	Sugar 02
Sugar06	Sugar 06
Sugar09	Sugar 09
Sugar10	Sugar 10
SugarNitro	Sugar Nitro

Tabla 10 - Mapping de gelantes.

Propiedad	Cota inferior	Cota superior
Catalan sa	0	1,062
Catalan sb	0	0,96
Catalan spp	0,52	40
ET et30	30,9	63,1
ET PY	0,58	1,95
FH floryhuggins	0,00357809	1,322877
Hansen Rij	2,657066	15,09735
Hansen dtotal	14,22146	47,80732
Hansen dsolvent	14	20,5
Hansen hsolvent	0	42,3
Hansen psolvent	0	18
HAD HD/A	0,011364	2,333333
HAD A	0	21,4

HAD HD	0	22,4
Hildebrand d total	14,22146	47,80732
Kamlet K-alpha	0	1,22
Kamlet K-beta	0	0,9
Kamlet K-Pi	-0,08	1,09
MOSCED mo-alpha	0	27,15
MOSCED mo-beta	0	26,17
MOSCED mo-lambda	13,71	19,67
MOSCED mo-q	0	13,36
MOSCED mo-tau	0,9	1,01
Physical henry	1,5E^-09	8,2
Physical RI	1,328	1,628
Physical DI	0,604	80,1
Physical Dipole	0	4,28
Physical logP	0,9	7,2
Physical kirk	0,1875	50,54902
Swain acity	0,13	41,43538
Swain basity	0,1	1

Tabla 11 - Umbrales de propiedades de solventes.