Nota de Aceptación

Aprobado por el Comité de Trabajo de Grado en cumplimiento de los requisitos exigidos por la Pontificia Universidad Javeriana para optar el título de Ingeniero de Sistemas y Computación.

Dr. Camilo Rocha
Decano de la Facultad de Ingeniería

ING. Gerardo M. Sarria
Director Carrera Ingeniería Sistemas y Computación.

Dr. Diego Linares
Director(a) Trabajo

Dra. Gloria Inés Álvarez
Director(a) Trabajo

ING. Juan Carlos Martínez Arias
Jurado 1

ING. Julián Gil Gonzales
Jurado 2

Pontificia Universidad JAVERIANA Cali | Facultad de Ingeniería y Ciencias

Acta de Correcciones al Proyecto de Grado
Ingeniería de Sistemas y Ciencias de la Computación

Fecha: 27/02/2023

Autores: Juan Pablo Luna y Naim Sadeghian

Nombre del Proyecto de Grado: Técnicas de Clustering Aplicadas en un Conjunto Metabolitos Perteneciente a Pacientes de Leishmaniasis Cutánea para Predecir La Efectividad del Tratamiento Glucantime a Través de Modelos de Aprendizaje Automático Clásicos

Director: Diego Linares y Gloria Inés Álvarez

Como indica el artículo 2.27 de las Directrices de Trabajo de Grado, he verificado que los estudiantes indicados arriba han implementado todas las correcciones que los Jurados del Proyecto de Grado definieron que se efectuaran, como consta en el Acta de Calificación correspondiente.

_____

Firma de Director(a) del Proyecto de Grado

Pontificia Universidad Javeriana Cali
Facultad de Ingeniería.
Ingeniería de Sistemas y Computación.
Proyecto de Grado.

Técnicas de Clustering Aplicadas en un Conjunto Metabolitos Perteneciente a Pacientes de Leishmaniasis Cutánea para Predecir La Efectividad del Tratamiento Glucantime a Través de Modelos de Aprendizaje Automático Clásicos

Juan Pablo Luna Mejia
Naim Samuel Sadeghian Perskie

Directores: Dr. Diego Linares y Dra. Gloria Inés Álvarez Vargas

27 de Febrero del 2023

Santiago de Cali, 27 de Febrero del 2023.

Señores
Pontificia Universidad Javeriana Cali.
Dr. Gerardo Mauricio Sarria
Director Carrera de Ingeniería de Sistemas y Computación.
Cali.

Cordial Saludo.

Por medio de la presente me permito informarle que los estudiantes de Ingeniería de Sistemas y Computación Naim Samuel Sadeghian Perskie (cod: 8946207) y Naim Samuel Sadeghian Perskie(cod: 8946207) trabajan bajo nuestra dirección en el proyecto de grado titulado "Técnicas de Clustering Aplicadas en un Conjunto Metabolitos Perteneciente a Pacientes de Leishmaniasis Cutánea para Predecir La Efectividad del Tratamiento Glucantime a Través de Modelos de Aprendizaje Automático Clásicos".

Atentamente,

Dr. Diego Luis Linares Ospina

Dra. Gloria Inés Álvarez Vargas

Santiago de Cali, 27 de Febrero del 2023.

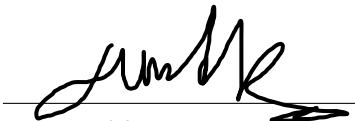
Señores
Pontificia Universidad Javeriana Cali.
Dr. Gerardo Mauricio Sarria
Director Carrera de Ingeniería de Sistemas y Computación.
Cali.

Cordial Saludo.

Nos permitimos presentar a su consideración el proyecto de grado titulado "Técnicas de Clustering Aplicadas en un Conjunto Metabolitos Perteneciente a Pacientes de Leishmaniasis Cutánea para Predecir La Efectividad del Tratamiento Glucantime a Través de Modelos de Aprendizaje Automático Clásicos" con el fin de cumplir con los requisitos exigidos por la Universidad para llevar a cabo el proyecto de grado y posteriormente optar al título de Ingeniero de Sistemas y Computación.

Al firmar aquí, damos fe que entendemos y conocemos las directrices para la presentación de trabajos de grado de la Facultad de Ingeniería aprobadas el 26 de Noviembre de 2009, donde se establecen los plazos y normas para el desarrollo del proyecto y del trabajo de grado.

Atentamente,



Juan Pablo Luna Mejia
Código: 8946272

Naim Samuel Sadeghian Perskie
Código: 8946207

Capítulo 1

Abstract

Los medicamentos usados para el tratamiento de la leishmaniasis pueden ser tóxicos y detrimentales para la salud. Peor aún, estos tratamientos no prometen curar al paciente en todos los casos. Para evitar recetar estos tratamientos a pacientes, a quienes no van a recibir beneficios, se han hecho varios estudios [4][7] para tratar de predecir, por medio de muestras de metabolitos en la sangre, en qué pacientes el tratamiento será efectivo. En este proyecto se hizo una continuación de estos estudios, basados en los mismos datos usados. Estos datos observaron 535 atributos/metabolitos para solo 36 pacientes. El grueso de este proyecto estaba en reducir la dimensionalidad del conjunto de datos (2 a 5 metabolitos) y poder llegar a resultados cercanos o mejores a los ya existentes. Se entrenaron 4 diferentes modelos de clustering para encontrar posibles grupos y de cada uno escoger un representante. Para cada modelo se buscaron los parámetros los cuales llegaban a clusters con un mejor grado de separación. En la fase de escoger los representantes de cada cluster se usaron diferentes métricas como: cercanía al centro del cluster, o probabilidad de ser miembro del cluster, para decidir cuáles podrían ser los mejores representantes. Después de tener los representantes de cada grupo, se pasó a la fase de predicción, donde se observó qué tan buena era la predicción con este pequeño conjunto de atributos. Finalmente se llegó a un modelo con 3 metabolitos y un puntaje *f1* de 0.82 el cual fue muy prometedor para una forma de reducción de la dimensionalidad tan particular y descriptiva como lo es el la selección por representantes de un agrupamiento.

Palabras Clave: Aprendizaje de Máquina, Leishmaniasis, Clustering, Gaussian Mixtures, KMeans, DBSCAN, Agglomerative, Reducción de Dimensionalidad.

Índice general

Capítulo 2

Descripción del Problema

2.1. Planteamiento del Problema

La leishmaniasis cutánea es una enfermedad parasitaria que afecta la piel y las mucosas, común en el trópico y zonas subtropicales, sobre todo en zonas boscosas. En Colombia se puede ver en el Chocó, Nariño, Amazonas . . . etc, donde hay estas zonas con bosques densos. Los parásitos que afligen al infectado -Leishmania- son transmitidas en su mayoría por la picadura de flebotomos, una subfamilia de moscas pequeñas de las cuales se conocen 53 especies involucradas en la transmisión de la enfermedad [1]. Entre las manifestaciones clínicas mas características de esta enfermedad se encuentran las llagas de 2 a 5 centímetros las cuales pueden crecer hasta los 10 centímetros. Además, estas se pueden presentar en cualquier parte del cuerpo y pueden aparecer muchas al mismo tiempo.

El tratamiento de esta enfermedad se basa en uno de 2 medicamentos. El primero: el antimonio, el cual es el medicamento de primera linea en este caso, viene en presentación de inyecciones dolorosas que se toman por 20 días y las cuales son tóxicas para el hombre y pueden generar daños en el corazón y el hígado. Aparte de ser tóxico, este tratamiento no asegura una cura y entre el 20 % al 30 % de la población tratada no se beneficia en absoluto del tratamiento. El segundo: la miltefosina, se toma por vía oral, lo cual lo hace más asequible y fácil de tratar en casa, el cual se toma por 28 días y aunque no amenazan con dañar el corazón, siguen siendo tóxicas a nivel gástrico y para embarazadas. Este tratamiento tampoco es tan efectivo y entre el 10 % al 40 % de los tratados no se benefician.

El estado del tratamiento de la leishmaniasis cutánea no es favorable en varios aspectos. Por un lado, se pone en riesgo la salud del paciente para poder combatir esta enfermedad. Se observa como ninguno de los tratamientos aseguran la cura del paciente. Más aún, el antimonio que se usa como primera medida es tan tóxico que puede causar daños colaterales al combatir la enfermedad. En pocas palabras, se pone en juego la salud del paciente sin poder asegurarle que el tratamiento tendrá resultados positivos. Por otro lado, este tratamiento presenta dificultades en otros aspectos de la vida del tratado. Como la población principal en la cual se presenta la leishmaniasis vive en áreas rurales y alejadas de las urbes, ir a un hospital se hace una tarea difícil, no solo por la lejanía de los hospitales; sino por la pérdida en tiempo productivo que implica la movilización para el afectado. Para recibir las 20 inyecciones que implica el tratamiento con antimonio, el tratado debe movilizarse al hospital 20 días seguidos.

Aunque la investigación acerca de la enfermedad se dificulta por la lejanía de la población a las ciudades, si se ha visto un movimiento de investigadores tratando de entender la enfermedad mejor y cómo tratarla más adecuadamente. En el International Journal of Infectious Diseases (Revista Internacional de Enfermedades Infecciosas) [2] se hace una análisis del progreso del tratamiento de

esta enfermedad desde hace más de medio siglo y cómo ha evolucionado el tratamiento hasta llegar al que se usa en la actualidad que sigue siendo tóxico. Además, se hace un vistazo al horizonte de las nuevas tecnologías que podrían ser implementadas por las farmacéuticas para un mejor tratamiento.

Por el momento, se puede hacer un análisis del funcionamiento del tratamiento actual. En cuanto a esto, existen estudios los cuales tratan de predecir en qué personas podría ser más probable que funcione el tratamiento, para no poner en riesgo la salud y ahorrar tiempo a la persona. Se han hecho predicciones basadas en el ADN de las personas a las cuales les funcionó el tratamiento y aquellos que no. También se han hecho sobre los metabolitos en una muestra de sangre, los cuales son entidades moleculares relativamente pequeñas y su cantidad es mucho menor al del ADN, por tanto es más fácil de analizar y encontrar correlaciones. Además, estos se pueden analizar de una forma no sesgada; es decir, se pueden tomar muestras de plasma en la sangre y mirar cientos de metabolitos a la misma vez, diferenciándolos por su peso y su carga. Un análisis desde este horizonte podría resultar más fácil y acertado que el análisis genético ya existente. Con técnicas de agrupamiento se podrían identificar metabolitos de interés (aquellos que tengan alguna correlación entre ellos y más aún si tienen alguna correlación con el resultado del tratamiento). Sobre este agrupamiento se podrían utilizar técnicas de aprendizaje automático y definir un modelo para saber si ciertos metabolitos son predictores del desenlace terapéutico.

2.1.1. Formulación

Cómo identificar los metabolitos que pueden ser predictores de la cura o fallo del tratamiento de la leishmaniasis cutánea utilizando técnicas de clustering y aprendizaje supervisado.

2.1.2. Sistematización

- ¿Cómo debe ser el pre-procesamiento de los datos?
- ¿Cuales metabolitos de la base de datos serán tenidos en cuenta?
- ¿Cuales técnicas de clustering y de aprendizaje supervisado se modelarán?
- ¿Cómo se evaluará que modelos y conjuntos de datos es mejor predictor?

2.2. Objetivos

2.2.1. Objetivo General

Utilizar modelos de aprendizaje automático sobre un conjunto de metabolitos, obtenido mediante técnicas de clustering al corpus de todos los metabolitos, para predecir el desenlace terapéutico de los pacientes con leishmaniasis.

2.2.2. Objetivos Específicos

- Identificar cuáles técnicas de clustering serán usadas para el agrupamiento.

- Pre-procesar los datos de entrada para que puedan ser usados por distintos métodos y sus resultados sean estandarizados.

- Construir los modelos usando las diferentes técnicas de clustering

- Evaluar los resultados obtenidos, siendo capaz de identificar cuál modelo agrupa a los metabolitos que son mejores predictores.

2.3. Justificación

La leishmaniasis cutánea es una enfermedad desatendida por los gobiernos porque afecta a una población rural, relativamente pequeña, y que en su mayoría no cuentan con influencia política, social o económica. Esta enfermedad, como ya se mencionó, tiene un impacto muy fuerte a aquellos afectados ya que cuentan con pocos recursos económicos, ellos dependen de su labor física para sus ingresos y se ubican lejos de centros de atención médica. Todo esto, junto con la realidad de que el tratamiento no es 100 % efectivo y muy doloroso, pone a aquellos afectados en una situación incómoda y con opciones limitadas. Por un lado, si deciden exponerse al tratamiento, esto implica desplazarse a centros médicos por 20 días, perdiendo tiempo de trabajo y estar en un tratamiento doloroso que no garantiza la recuperación y que es tóxico para el cuerpo. En contraste, la persona puede decidir optar por no recibir el tratamiento y reducir significativamente su calidad de vida, en ciertos casos llegando hasta la muerte.

Al observar la situación, las opciones para mejorarla son limitadas. Tras varios años de investigación para mejores alternativas para el tratamiento y con poco éxito, hay que recurrir a otras alternativas. Se han optado por alternativas predictivas para observar al paciente y de antemano decirle si se va a recuperar si se expone al tratamiento o no. Este tipo de identificación de patrones utilizando estudios estadísticos sobre pacientes existe y se ha llevado a cabo en investigaciones que miran el perfil de metabolitos en la sangre de los pacientes para la predicción [4]. Además, se han hecho modelos de aprendizaje automático sobre estos mismos datos para mejorar los resultados, con métodos más complejos que los estadísticos [7].

Lo que se ve aquí es una oportunidad para aportar a esta investigación y generar, a través de la computación, otros resultados para ayudar a través de la identificación de metabolitos útiles para la predicción, a predecir el resultado del tratamiento. Un reto que busca utilizar el aprendizaje automático para encontrar nuevas alternativas para ayudar y facilitar el tratamiento a las personas afectadas. Hay espacio para probar nuevos modelos y formas de predecir el funcionamiento del tratamiento utilizando métodos menos usuales como lo es la selección de atributos con modelos de agrupamiento.

Un proyecto así sería viable ya que utiliza datos que ya existen y fueron colectados y procesados; por tanto, solo requiere de la implementación de estas técnicas, las cuales han sido implementadas y documentadas en estudios similares y son fácilmente replicables.

2.4. Delimitaciones y Alcances

1. Se busca implementar al menos 4 métodos distintos de clustering para poder hacer un análisis y comparar los resultados entre estos
2. Tendrá un plazo de 4 meses
3. La muestra se hará sobre el estudio de metabolitos recopilado por el CIDEIM

Capítulo 3

Desarrollo del Proyecto

3.1. Marco de Referencia

3.1.1. Áreas Temáticas

Las áreas principales que se tocarán para este trabajo de investigación son varias. La primera es: la leishmaniasis como enfermedad y cómo impacta su tratamiento a los pacientes. La segunda, es la predicción de resultados, a manera general, basado en atributos biológicos. Siguiendo qué procesos se han llevado a cabo y cuales modelos se han usado para la predicción. Por último, está el tema de agrupamiento: cuales son las distintas técnicas, como varían y qué técnicas pueden ser más útiles y apropiadas para la realización de predicciones sobre datos biológicos como los metabolitos.

3.1.2. Marco Teórico

3.1.2.1. Leishmaniasis

La leishmaniasis es una enfermedad complicada y poco fácil de tratar. Esta enfermedad cuenta con pocas alternativas para su tratamiento. En América Latina, su tratamiento recae principalmente en el antimonio, con el cual se deben tener en cuenta varios factores como lo son: la forma clínica de la enfermedad, la región geográfica (especie de parásito involucrada), enfermedad subyacente, medicamento adecuado y disponibilidad de medicamentos. El 65 % de los pacientes presentan eventos adversos, siendo la mayoría leves o moderados y no impiden la continuación del tratamiento, entre ellos dolor en el sitio de aplicación intramuscular, vómito, náuseas, mialgias, artrologías, y cefalea. Además, se presentan efectos tóxicos sobre riñón, hígado, corazón y páncreas donde su punto máximo de presentación ocurre entre los días 7 y 14 del tratamiento. [1]

El páncreas también es otro órgano susceptible a toxicidad por los antileishmaniasicos. Entre el 33 y el 75 % de los pacientes tratados registran elevación de las enzimas pancreáticas (amilasa y/o lipasa séricas). Pacientes de cualquier edad con aumento de hasta 5 veces el valor basal pueden hacer manifestaciones clínicas de pancreatitis. Por tal razón, partiendo de una línea de base de amilasa y/o lipasa, se debe hacer un estricto seguimiento clínico y para-clínico del paciente en tratamiento y repetir en los días 7 y 12 de tratamiento, momento en que se presentan las mayores elevaciones. En aquellos pacientes con elevaciones superiores a 10 veces el valor basal se debe suspender el tratamiento.

Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large `n_samples`, medium `n_clusters` with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters	Distances between points
Affinity propagation	damping, sample preference	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with `n_samples`	Many clusters, uneven cluster size, non-flat geometry	Distances between points
Spectral clustering	number of clusters	Medium `n_samples`, small `n_clusters`	Few clusters, even cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters or distance threshold	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints	Distances between points
Agglomerative clustering	number of clusters or distance threshold, linkage type, distance	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints, non Euclidean distances	Any pairwise distance
DBSCAN	neighborhood size	Very large `n_samples`, medium `n_clusters`	Non-flat geometry, uneven cluster sizes	Distances between nearest points
OPTICS	minimum cluster membership	Very large `n_samples`, large `n_clusters`	Non-flat geometry, uneven cluster sizes, variable cluster density	Distances between points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation	Mahalanobis distances to centers
Birch	branching factor, threshold, optional global clusterer.	Large `n_clusters` and `n_samples`	Large dataset, outlier removal, data reduction.	Euclidean distance between points

Figura 3.1: Características de modelos de agrupamiento[4]

3.1.2.2. Métodos de Predicción Sobre Datos Biológicos

Para el tratamiento de la leishmaniasis se han realizado varios estudios sin embargo no ha habido avances significantes. Una alternativa que se toma es la predicción de cura para los pacientes. Hay varios métodos que se pueden utilizar para la predicción de tratamientos como lo son el aprendizaje automático, estudios estadísticos o simples observaciones de síntomas. Así mismo como existen varios métodos estos se pueden realizar sobre varios elementos como lo son la genética, los metabolitos o historias clínicas [2]. Uno de estos estudios fue el estudio estadístico [4] de Vargas que habla acerca de como ciertos metabolitos pueden ser predicadores de la cura o falla para la investigación acerca de si el antimonio del tratamiento afecta ciertos metabolitos que pueden ser claves para la recuperación del paciente.

3.1.2.3. Algoritmos de Agrupamiento

Los algoritmos de agrupamiento se usan para clasificar un grupo de datos según sus atributos y dividirlos en grupos con características similares. Cada algoritmo de agrupamiento utiliza diferentes procesos y métricas para definir cómo se agrupan los datos según su cercanía. La cercanía se refiere a que tan parecidos son los atributos de un dato a comparación con otro. En el caso de este estudio, se agrupan los metabolitos para encontrar grupos con características similares y poder escoger un

representante de cada grupo, y así poder reducir la dimensión de los datos a la hora de hacer la predicción.

Lo que se busca es observar diferentes modelos de agrupamiento para identificar metabolitos que puedan ser indicadores de la cura del tratamiento. En otros proyectos, dentro del área biológica, también se han utilizado modelos de agrupamiento para agrupar genes, elementos de la sangre y metabolitos para el desarrollo de la investigación. "A biologically-inspired validity measure for compasión of clustering methods over metabolic datasets"[5] habla de cómo los algoritmos más populares para agrupar datos biológicos son K-vecinos y agrupamientos por jerarquía. Aparte de estos existen varios algoritmos de agrupamiento que cuentan con sus características únicas como se observa en la figura 2.1. Los algoritmos seleccionados son descritos a mayor detalle en la etapa de agrupamiento.

3.1.3. Trabajos Relacionados

- Pharmacometabolomics of Meglumine Antimoniate in Patients With Cutaneous Leishmaniasis[4]

  Este articulo habla de un estudio estadistico sobre una base de datos de metabolitos de pacientes antes y después del trátamelo de la leishmaniasis donde su objetivo es identificar que metabolitos son indicadores de la recuperación de la enfermedad.

- Cluster analysis and prediction of treatment outcomes for chronic rhinosinusitis[3]

  Este artículo utiliza técnicas de agrupamiento no supervisado para agrupar fenotipos de pacientes con rhinusitis crónica (la cual tiene malos pronósticos en cuanto al resultado del tratamiento, igual que le leishmaniasis). Por tanto, buscan determinar a qué subgrupos les podría ser útil pasar a cirugía y cuáles deberían seguir en manejo médico.

- GUÍA PARA LA ATENCIÓN CLÍNICA INTEGRAL DEL PACIENTE CON LEISHMANIASIS[1]

  Este extenso documento habla de lo que se sabe y se entiende por la leishmaniasis en Colombia, donde de un contexto acerca de todas la etapas de la enfermedad desde el incio y los tipos de parásitos hasta los tratamientos y sus efectos.

- Drug-Drug Interactions prediction from enzyme action crossing through machine learning approaches[2]

  Este articulo habla acerca de como la interacción entre diferentes drogas puede ser detrimental para el tratamiento de una enfermedad. Por tanto, busca ver como afecta el metabolismo mirando diferentes enzimas y las agrupa por medio de k-Nearest Neighbors y Redes Neuronales y Support Vector Machines.

- A Biologically Inspired Validity Measure for Comparison of Clustering Methods over Metabolic Data Sets[5]

Habla de como los genes y los metabolitos son una expresión de procesos biológicos comunes y de como analizar los patrones de miembros agrupados puede ser de gran utilidad desde la perspectiva biológica. El artículo compara varios métodos de clustering sobre data sets de metabolitos y el conocimiento a priori sobre estos elementos biológicos. Además, propone una manera de medir la importancia biológica que tienen las agrupaciones encontradas.

Capítulo 4

Preprocesamiento

4.1. Descripción de los datos

Los datos utilizados en el proyecto fueron tomados de unos estudios hechos en 2019 [4]. En estos se lleva el registro de 35 pacientes que pasaron por el tratamiento para la leishmaniasis. La tabla tiene 536 atributos, donde 535 de son los metabolitos en el perfil de la sangre y el último atributo es un valor booleano que dice si el tratamiento curó o no al paciente. En este caso 22 de los pacientes tienen valor positivo, es decir que el tratamiento fue exitoso, y el resto fueron negativos.

4.2. Escalamiento

Los datos indican que había una gran diferencia entre algunos metabolitos, donde algunos podían variar desde 0 a más de 10,000. Estos rangos tan grandes pueden sesgar el modelo al darle más importancia -o más peso- a algunos valores cuando no debería. Por esto, para el escalamiento de datos, se decidió utilizar el escalado de Pareto. Esto se decidió porque el uso de este escalado en situaciones similares había dado buenos resultados, además fue el usado por Zuluaga [7] en su estudio. Este escalado consiste en dividir el valor por la raíz cuadrada de la desviación estándar de la columna a la que corresponde [7]. Esto redujo de forma muy significativa la enorme diferencia entre los datos de miles a solo unos cientos.

4.3. Análisis de datos correlacionados

Para limpiar los datos correlacionados se hizo una matriz de correlación entre todos los metabolitos. Mirando los rangos superiores e inferiores de la matriz se tiene que los 500 metabolitos más correlacionados tenían valores entre 0.995978 a 0.755847 y los 500 metabolitos menos correlacionados estaban muy cercanos a ser cero; 0.000994 a 0.000001. Como todos los datos representan la cantidad de un metabolito en la sangre, no se registran datos negativos. Al ver que habían datos altamente correlacionados, fijamos 0.875 como el valor mínimo de correlación entre 2 parejas para decidir si se podía eliminar un metabolito de la pareja. Este número representa una alta correlación entre 2 atributos y por tanto se tiene que cada miembro de la pareja sería un buen representante del otro. Usualmente se eligen un numero entre 0.7 a 0.90 como valores mínimos para ser considerados altamente correlacionados, se observó una gran cantidad de datos por encima de 0.875 el cual es considerablemente alto y, a su vez, abarca un número sustancial de datos.

Después de establecer este conjunto de parejas por encima del umbral propuesto, se encontró que habían componentes fuertemente conexos. Habían grupos de metabolitos en los que se podían trazar ciclos donde existía cierta transitividad; si la pareja (a,b) y (b,c) estaban encima del umbral, era muy probable que (a,c) también estuviera en ese rango. Por tanto se construyó una tabla donde, para cada metabolito se guardaba los metabolitos con los que estaba correlacionado por encima del umbral y a que valor. Por consiguiente, se ordenó la tabla descendentemente según la cantidad de metabolitos relacionados a un metabolito dado, y si eran de la misma longitud, primaba el que el promedio de sus correlaciones fuese más alto. Ordenar de esta manera aseguraba que el primer metabolito que se mira seria un mejor representante ya que representa más metabolitos y con un mejor promedio de correlación. A continuación se encuentran los metabolitos que se eliminaron y cual seria su representante:

Metablito Representante	Metabolito(s) eliminados
270	145, 149, 155, 177, 347
520	4, 18, 128, 163, 246
373	27, 30, 316 , 325
451	148, 208, 247, 406
176	62, 131, 370
151	12, 15, 171, 307
46	24, 58, 158
114	115, 116, 239
464	5, 26, 53
142	230, 297
3	165, 481
324	276, 412
361	366, 531
136	201, 516
108	6
339	341
500	502
322	334
434	504
167	456
285	353
215	381
102	399
306	311
386	449
210	211
377	518
526	529
71	77
111	228
180	383
74	79
327	360
188	349
291	372
72	85
112	498
41	130
88	134
432	533
272	342
214	258

Cuadro 4.1: Representantes de metabolitos altamente correlacionados

Capítulo 5

Algoritmos de Agrupamiento

Para el agrupamiento se utilizaron 4 algoritmos en total, cada uno de naturaleza diferente para variar los resultados. Por naturaleza esto hace alusión como el modelo agrupa datos con un modelo por distancia, por densidad, por jerarquía o un modelo probabilístico. Para esto los 4 algoritmos que se eligieron fueron *Kmeans*, *DBSCAN*, *Gaussian Mixtures* y agrupamiento jerárquico aglomerativo. para la implementación de estos modelos se utilizaron las librerías respectivas de sklearn en python[6].

5.1. KMeans

El algoritmo de KMeans es un modelo de agrupamiento relativamente simple para clusters con forma circular/esférica. Consiste en escoger una cantidad de puntos k los cuales serán los centros de cada cluster. Estos centros se ponen al azar entre los puntos que representan cada dato. Luego, cada punto/dato es asignado a un cluster según el centro que se halle más cercano. Iterativamente, se re-posicionan los centros según la media de todos los puntos asignados a él, hasta que ya no haya cambio. Como el posicionamiento inicial de los centros puede afectar mucho los resultados, se repite este proceso con diferentes posicionamientos de los centros para encontrar el resultado con la menor varianza entre datos de un mismo cluster. Este algoritmo en particular escala de forma muy efectiva para una cantidad de datos grande o mediana. También es útil a la hora de generar pocos clusters en comparación de muchos.

5.1.1. Selección de parámetros

Para el algoritmo de agrupamiento de *KMeans* se observó solo un parámetro que es el número de clusters que tendrán los agrupamientos: k. La librería de sklearn ofrece más parámetros, pero se trabajó únicamente con el número de clusters como parámetro los demás se dejaron como valores fijos que selecciona por defecto en la librería de sklearn.

Esta selección se realizó con un grilla donde se evaluaban los resultados de clusters con diferentes métricas para determinar qué valor era el mejor. En esta etapa se realizaron análisis con el método del codo, la métrica de Calinski Harabazs, la métrica de Shilhouette y la métrica de Davies Bouldin.

La primera métrica que se implementó fue la del método del codo, donde se mide la distorsión de los datos a medida que el número de clusters aumenta. Para cada clusters se mide la suma de cuadrados total dentro del cluster (WCSS). Estos valores se grafican, tal que, para cada número de cluster en el eje x se representa su puntaje en el eje y y se observa el punto de quiebre en la

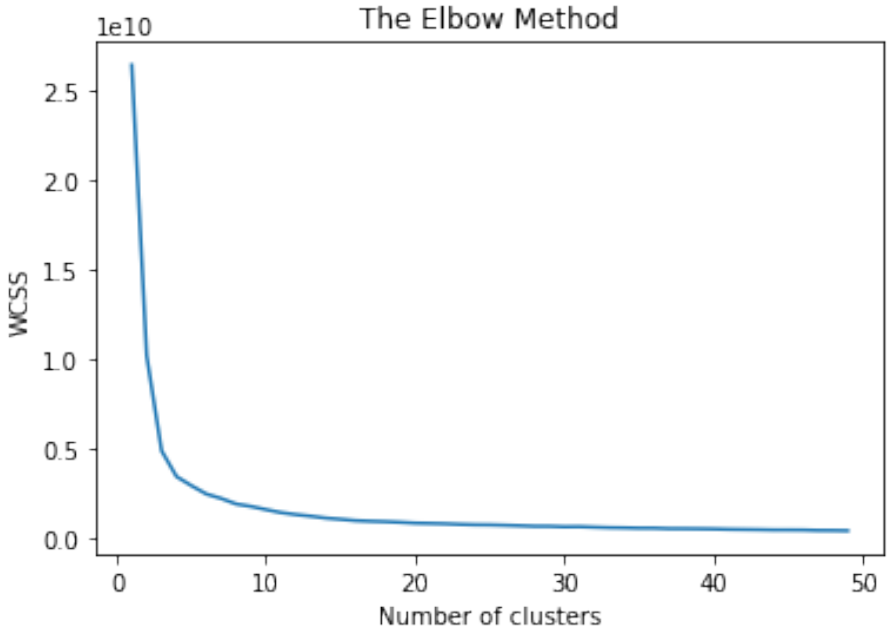


Figura 5.1: Grafica de metodo de codo por numero de clusters

curvatura, donde la pendiente pasa de vertical a horizontal. Lo que se observó fue que el número ideal de clusters estaría entre 3 y 8 como se aprecia en la gráfica en la parte inferior.

La siguiente métrica que se evaluó fue la de Calinski Harabasz. Esta métrica también conocida como criterio de relación de varianza, se calcula como una relación de la suma de la dispersión entre grupos y la suma de la dispersión dentro de los grupos para todos los grupos (donde la dispersión es la suma de las distancias al cuadrado). Por lo que, a medida que este valor sea más alto, implica que hay clusters mejores formados y más densos. Utilizando esta métrica, se observó un comportamiento similar que con el método del codo; donde, en los primeros clusters, se ve una puntuación mucho más alta y a medida que se van aumentando los clusters el score va decreciendo. Por tanto, lo ideal, según esta métrica, es tomar un número de clusters más bajos entre 3 y 6. Se observa que hay un constante decrecimiento en la puntuación, donde parece decrecer de forma logarítmica con excepción de un pico que ocurre alrededor del cluster 17.

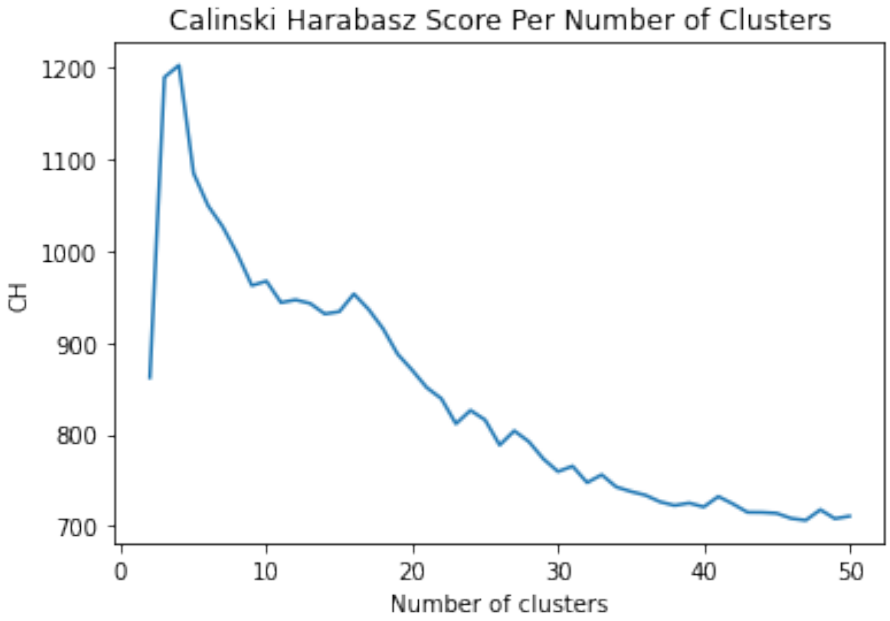


Figura 5.2: Grafica de Calinski Harabasz Score por numero de clusters

La siguiente métrica que se utilizó para observar los clusters fue la de Davies Bouldin. Esta métrica es una de las medidas de evaluación de los algoritmos de agrupamiento que se usa más para evaluar la división del split de K-Means, para una cantidad determinada de clústeres. Este método mide la dispersión interna de los clusters, la dispersión entre clusters y su semejanza entre ellos. Al final, entre más bajo sea el score mejores serán las agrupaciones. Para la métrica de Davies Bouldin se observó como los mejores resultados, en comparación, están en valores de clusters bajos entre 2 y 6. Después de esto, hay otros números de clusters que tienen un valor razonable como los valores entre 13 y 18.

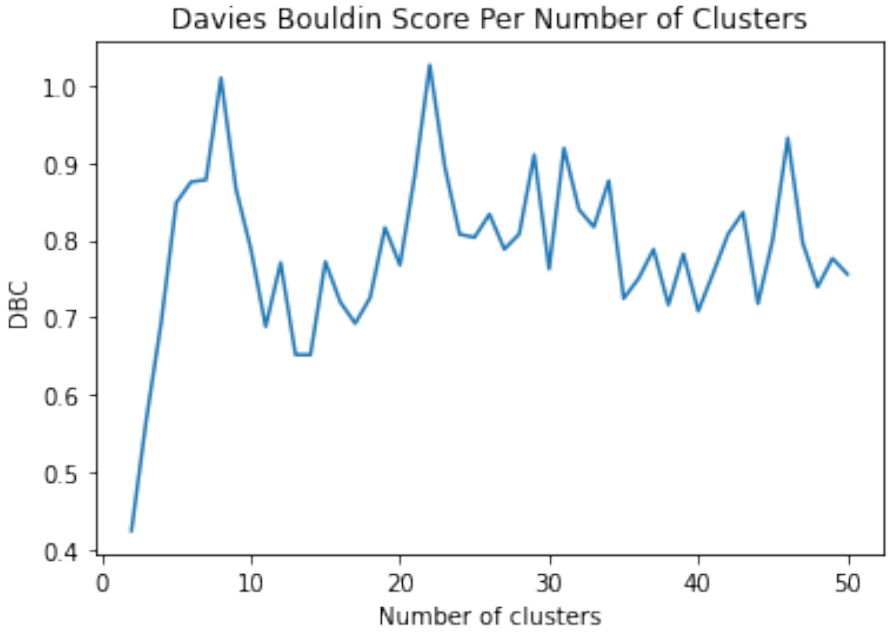


Figura 5.3: Grafica de Davies Bouldin Score por numero de clusters

La última métrica que se utilizó fue la Silhouette Score. Esta también es una métrica utilizada para evaluar la agrupación. Su valor varía de -1 a 1 siendo 1 lo mejor y -1 lo peor. Este valor se calcula tomando la distancia promedio entre todos los clusters y restando la distancia promedio entre cada punto dentro de su cluster y dividiéndola por el valor máximo de ambos. Para la métrica de Silhouette se clasificó el score promedio para cada número de clusters de 2 a 50 y se ve como los mejores resultados están en los valores bajos entre 3 y 10. Aquí, para entender mejor cómo se estaban distribuyendo los datos, se grafició el score de cada atributo por cluster y esto se realizó para cada tipo de agrupamiento, con un k de 5 y 10 para hacer una idea general de cómo se estaban formando los clusters y qué tan bien se formaban estos. Como se puede observar cuando k es 5, hay un cluster de proporciones mayores, que agrupa la gran mayoría de los elementos, y otros 4 clusters más pequeños; donde cada uno de estos tenía un Silhouette score bueno pero no excelente en comparación con el cluster grande. A medida que fuimos aumentando el número de clusters en 10, 15, 20 ,25 este patrón se conservó: donde había un cluster que agrupa muchísimos más elementos que los demás y a medida que íbamos aumentando los clusters, aquellos que no eran el cluster más grande iban empeorando. Esto es justo como esperábamos por los resultados de las métricas anteriores, donde a mayor número de clusters peor la agrupación.

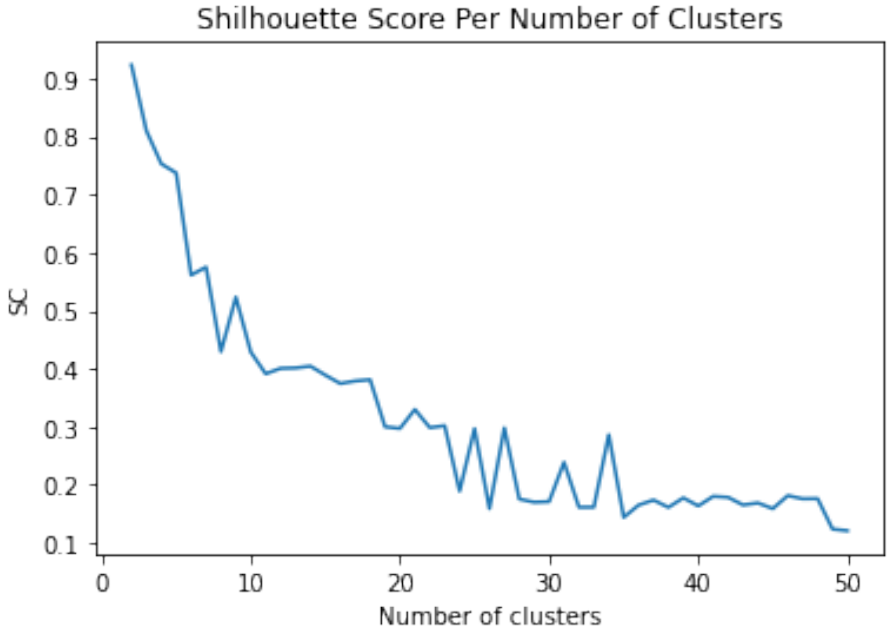


Figura 5.4: Grafica de Shilhouette Score por numero de clusters

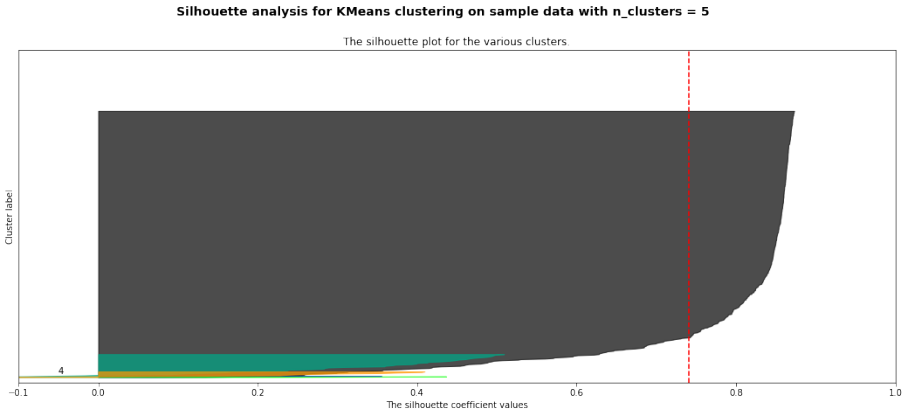


Figura 5.5: Distribución de Shilouette score por cada metabolito con 5 clusters

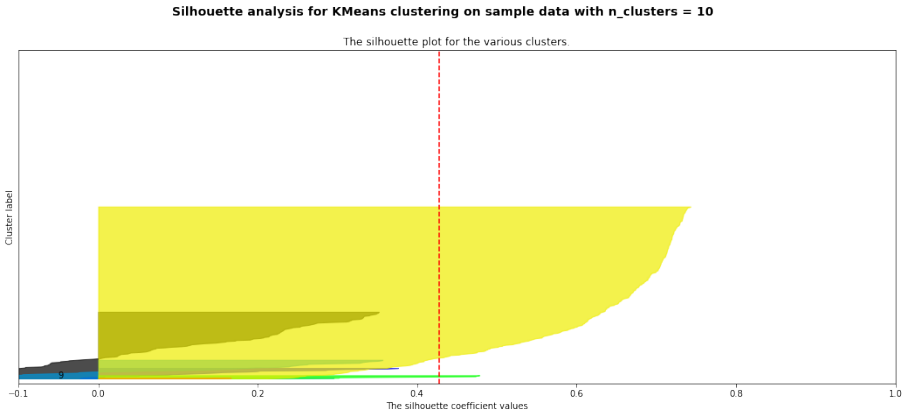


Figura 5.6: Distribución de Shilouette score por cada metabolito con 10 clusters

5.2. DBSCAN

El algoritmo de DBSCAN busca resolver el problema de encontrar clusters los cuales no se encuentran dentro de zonas circulares, sino por la densidad de diferentes grupos. Primero, se escogen registros al azar y se usan sus atributos como dimensiones en el espacio. Para cada registro se busca en un diámetro de tamaño *epsilon*, dentro del cual se contaron los otros registros que estén cercanos. Si una cantidad *min_samples* de vecinos se encuentran dentro del diámetro, el punto se le denomina como punto clave. Esto se hace para cada registro hasta definir cuales son puntos clave. Luego se toma un punto clave al azar y se le asigna un cluster. Iterativamente, se marcan todos los puntos clave dentro de diámetro como parte del mismo cluster y partiendo de estos se marcan sus vecinos; así se repite, hasta agotar los puntos clave dentro del diámetro. Todos los puntos visitados que no sean clave también se marcan como parte del cluster al final, pero no se itera sobre ellos. Si todavía hay puntos clave, estos hacen parte de otro cluster y se hace el mismo proceso de marcar los. Si al final quedan puntos que no estaban en el radio de ningún punto clave, estos se toman como ruido.

5.2.1. Selección de parámetros

DBSCAN cuenta con 2 hiperparámetros principales: *min_samples*, el cual define la cantidad mínima de vecinos que debe haber alrededor de un punto para que se considere el centro de un grupo; *epsilon*, define la distancia máxima a la cual puede estar un dato para que sea considerado un vecino. El heurístico utilizado para escoger los valores de los hiperpárametros[8], consiste en escoger un valor n para *min_samples* y para cada uno de los datos, encontrar la distancia a la cual se encuentre el nésimo vecino. Luego, se ordenan las distancias de forma ascendente; en el cual, se busca tener una expresión gráfica del aumento de la distancia del vecino más lejano. Una vez están ordenados y graficados los datos, se busca el codo del gráfico y éste será el valor que tomará *epsilon*. La idea detrás de este heurístico es que se tome el punto intermedio/quiebre donde el radio no es lo suficientemente grande para que cualquier dato pueda ser centro, pero tampoco se formen grupos y centros muy pequeños y el resto de datos sean considerados como ruido.

Para hacer esto se utilizó la librería de Sklearn y la función de *NearestNeighbors* para encontrar los n vecinos más cercanos y luego poder saber a qué distancia se encontraba el nésimo. Se ordenaron los datos y luego se utilizó la libreria de kneed y la funcionalidad de *kneelocator*[9], la cual encuentra el "codo" (o en esta caso "rodilla") de un arreglo de datos. Es decir el punto, o puntos de mayor curvatura. En este caso, se tomó un solo punto. En la figura 5.7 se pueden observar los gráficos de las distancias ordenadas para los valores de 2 a 20 de *min_ samples*. En el eje x está el número de dato y en el eje y esta la distancia del nésimo vecino. Como se puede observar, todas las gráficas presentan un comportamiento altamente similar: alrededor de los primeros 450 datos tienen su n-ésimo vecino a casi la misma distancia y esta empieza a crecer exponencialmente hacia el final. Esto indica que hay un grupo grande de datos que están muy cercanos y luego unos cuantos que se encuentran a la periferia. Además, los datos del grupo grande están tan cercanos que la distancia del vecino segundo es similar a la del vecino veinteavo.

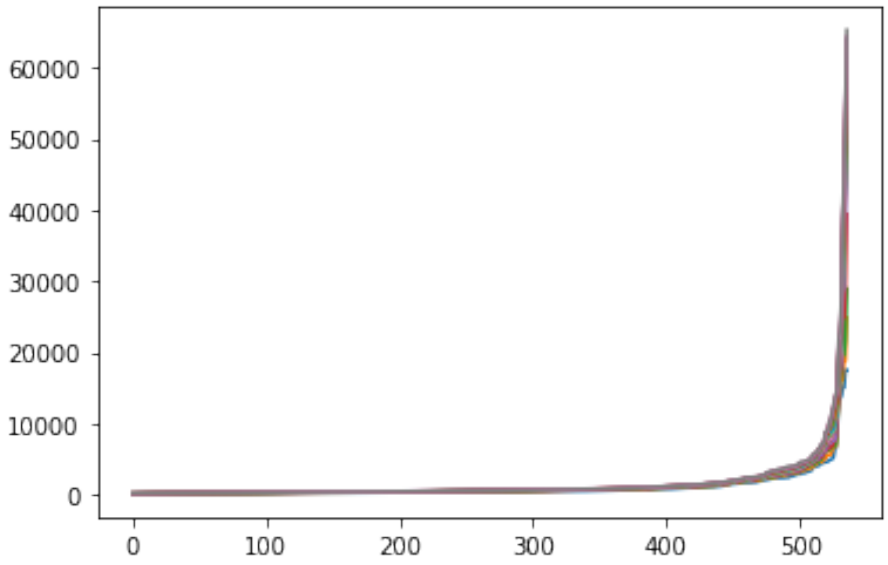


Figura 5.7: Distancia del n-ésimo vecino (eje: y) para cada metabolito.

5.2.2. Resultados

Una vez encontrados los valores de *epsilon* respectivos para los *min_ samples* de 2 a 20 se hizo una clasificación de los datos. Para los valores de *min_ samples* de 4 a 20, solo había un gran cluster y el resto de los datos eran tomados como datos ruidosos. Solo para los valores de 2 y 3 se obtuvieron resultados donde había la presencia de un cluster más 5.1. Este comportamiento va de la mano con lo presenciado en la extracción de los parámetros: se podía observar que el gran cúmulo de datos estaban muy cercano entre sí y que el resto de los datos se encontraba a una distancia mucho mayor a comparación. Como el algoritmo busca agrupar zonas de alta densidad, era de esperarse

que hubiera un una clase mayor con el gran cúmulo de datos y que el resto parecieran ruido.

***min_ samples*: 2**	
Metabolitos en grupo pequeño	77, 352
***min_ samples*: 3**	
Metabolitos en grupo pequeño	379, 393, 398

Cuadro 5.1: Metabolitos que no fueron tomados como ruidosos

5.3. Mixturas Gausianas

El modelo de Mixturas Gaussianas es similar al de K Means, en cuanto a que utiliza centros de masa para agrupar registros. Pero difiere en que no utiliza solo la media de los datos, sino que utiliza EM (Maximización de la Esperanza) y distribuciones Gaussianas para escoger los centros. Primero coloca una cantidad *n_ componentes* de centros en el espacio, donde cada centro es el centro de una distribución normal. Cada uno de estos centros sera el centro de un cluster, aunque en primera instancia no estén ubicados de la mejor manera. Además, se mira la co-varianza de los distintos atributos para definir la forma del área alrededor del centro ya que estos no necesariamente son circulares/esféricos. Para cada punto se calcula la probabilidad de que pertenezca a cada cluster, según su distancia a cada centro, y se señala cual es el que tiene la probabilidad mas alta. Luego hay una iteración de actualizar, tanto los centros, como la matriz de co-varianza. Esto se hace moviendo los centros hacia donde están los puntos que son mas probables de pertenecer a un dicho cluster. Este proceso se repite hasta que los centros y su forma tiendan a converger; es decir, que ya no presentan cambios sustanciales.

5.3.1. Selección de parámetros

Los 2 parámetros principales los cuales se tuvieron en cuenta para este modelo son; *n_ componentes*, que es el número de centros Gaussianos que serán usados para la clasificación; *covariance_ type*, el tipo de covarianza que será usado. Para determinar cuál sería la mejor clasificación se usó la métrica *silhouette*[6], la cual asigna un valor de -1 a 1, teniendo en cuenta para cada miembro su cercania a su cluster y al cluster mas cercano al cual pertenece. Un puntaje cercano a 1 significa que hay una buena separación entre los clusters, es decir; que la distancia del cluster más cercano es considerablemente mayor que su propio cluster. En el cuadro 5.3 se observa un barrido por estos 2 parámetros, anotando el puntaje de silhouette para cada uno. En general, entre mayor el número de componentes, menor el puntaje. De igual manera, el tipo de covarianza esférico ("spherical") y diagonal ("diag") presentaron los peores resultados en todos los casos. En contraste, los puntajes más altos fueron con 2 y 3 componentes y la covarianza atada ("tied"). Por tanto, se procede a tomar

estos dos agrupamientos con los puntajes más altos, para la reducción de la dimensionalidad.

Los grupos formados por los 2 mejores agrupamientos presentaron características similares; aunque ambas eran de una cantidad diferente de componentes, en ambas persistía la existencia de un cluster mucho más grande que contenía más del 90 % de los metabolitos. Más interesante aún, es que el cluster más pequeño de ambos agrupamientos es el mismo, a excepción de que uno tiene un miembro extra (Cuadro 5.2).

Centros	3
Covarianza	"tied"
Metabolitos	77, 252, 352, 365, 386

Centros	2
Covarianza	"tied"
Metabolitos	77, 229, 252, 352, 365, 386

Cuadro 5.2: Metabolitos del cluster más pequeño, en los modelos con los mejores puntajes

Centros	Covarianza	Silhouette Score
2	tied	0.9272
2	spherical	0.6468
2	diag	0.6440
2	full	0.9272
3	tied	0.8109
3	spherical	0.3109
3	diag	0.3176
3	full	0.8096
4	tied	0.7478
4	spherical	0.2763
4	diag	0.2813
4	full	0.4993
5	tied	0.7199
5	spherical	0.1468
5	diag	0.2779
5	full	0.7204
6	tied	0.6902
6	spherical	0.2594
6	diag	0.2801
6	full	0.4985

Cuadro 5.3: Silhouette Score de modelos con diferentes parámetros

5.4. Agrupamiento Aglomerativo

Por último, está el algoritmo jerárquico aglomerativo el cual funciona de manera "de abajo hacia arriba". Es decir, cada objeto se considera inicialmente como un grupo de un solo elemento (hoja). En cada paso del algoritmo, los dos grupos que son más similares se combinan en un nuevo grupo más grande (nodos). Para unir los grupos se utilizan parámetros de linkage, affinity y número de clusters.

5.4.1. Selección de parámetros

El algoritmo jerárquico aglomerativo cuenta con varios parámetros de los cuales elegir, entre los cuales están los parámetros:

- Número de clusters.
- Afinidad: haciendo referencia a cómo se calculan las distancias (*Manhattan, euclidiano, coseno, L1 y L2*).
- Vinculación: en donde el criterio de vinculación determina qué distancia usar entre conjuntos de observación (average complete single y ward).

Para esto se realizó una grilla donde, para cada número de clusters de 2 a 50, se probó cada vinculación con con cada afinidad y se observó el *silhouette score*. Al finalizar, se determinaron cuales afinidades y linkage eran las más óptimas para un número x de clusters. Aparte de esto, se tomó en cuenta la selección del número de clusters que se hizo con *K-Means* considerando que un número bajo de clusters daría mejores resultados. En la tabla se pueden observar algunos de los resultados de la grilla 5.4.

Numero de Clusters	Afinidad	Vinculación	Shilhouette Score
2	Euclidian	Single	0.9369191909466
3	Euclidian	Single	0.9285805879970
4	Euclidian	Single	0.9196569845429
5	Euclidian	Single	0.9140885520336
6	Euclidian	Single	0.8826348403682
7	Euclidian	Single	0.8677880971622
8	Euclidian	Single	0.8662221000548
9	Euclidian	Average	0.8292241605980
10	Euclidian	Average	0.8233411700437

Cuadro 5.4: Puntaje Silhouette para diferentes parámetros

Como se observó la mejor distancia fue la euclidiana para todos los casos y en la mayoría la mejor vinculación fue Single donde únicamente para los clusters 9 y 10 fue Average. Previamente

a los resultados se tomó la decisión de solo observar aquellos modelos que tuvieran un score de 0.8 o mayor ya que este número es un buen punto de quiebre para decidir si un clustering tiene una buena separación. Los únicos modelos con ese score fueron de los clusters 2 al 10. Estos son los que se usarán para realizar la predicción en la siguiente etapa.

Capítulo 6

Selección de Representantes

Después de ajustar los modelos de agrupamiento y encontrar sus clusters y respectivos centros, se llevó a cabo un proceso de selección de representantes. Como se decidió no utilizar los resultados del modelo de *DBSCAN* para la predicción, el cual era el único modelo no basado en centros, los algoritmos restantes contaban con la noción de centroides para agrupar. Se aprovechó este hecho para la selección de representantes, donde el heurístico para seleccionar el representante de cada cluster se hará según su cercanía al centro del cluster. Por tanto, para los modelos entrenados se encontró 1 representante para cada cluster; tal que, si se entrenó un modelo el cual agrupó en 3 clusters, entonces se seleccionaron 3 representantes en total. En la figura **??** se ejemplifica a manera gráfica cómo se vería este proceso en 2 dimensiones (a diferencia de 35): donde se diferencian los grupos según su color y se señala el centroide en color rojo.

De antemano, no había forma de saber si este heurístico de selección de representantes llevaría a los mejores resultados. Por tanto, para tener un grupo de control, se seleccionaron otros miembros del grupo, que eran los siguientes más cercanos al centro, y se tomó 1 de los mejores 5 al azar. Como el modelo de *Gaussian Mixtures* también observa la probabilidad de que cada dato esté en un cluster y agrupa según el más probable, se decidió que también se utilizará el heurístico del miembro más probable de cada grupo para la selección. La noción de probabilidad de quedar en un cluster agrega otro factor ya que no solo mide qué tan cerca está al centro del cluster, sino que tan alejada está del resto.

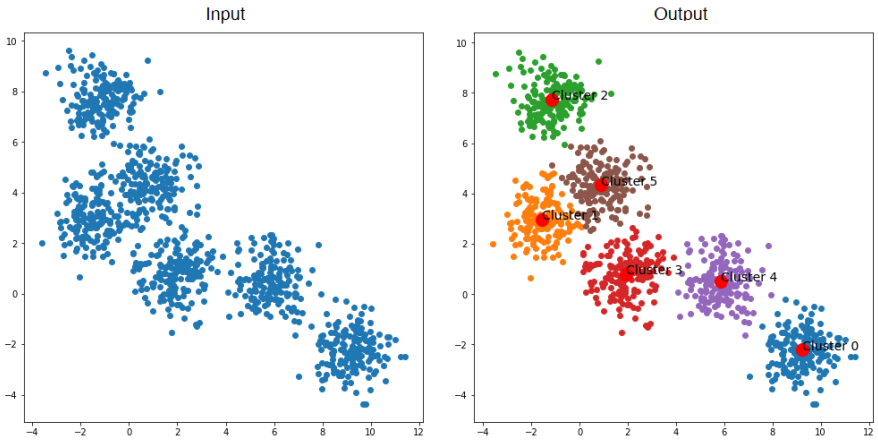


Figura 6.1: Visualización de Agrupamiento y Centroides en 2 Dimensiones [10]

Capítulo 7

Predicción y Resultados

Una vez se seleccionaron algunos metabolitos de interés -o más representativos- se podía hacer una reducción del ancho de los atributos de la tabla utilizando solo este pequeño grupo. Para comprobar que tanto información nos brindan estos metabolitos en cuanto a la predicción del funcionamiento del tratamiento se seleccionaron 3 modelos de clasificación para observar los resultados: *K-Nearest Neighbors, Logistic Regression y Random Forest*. Además, se comparan los resultados obtenidos al hacer la clasificación con los diferentes conjuntos de metabolitos seleccionados en la etapa anterior. Esto lleva a que se haga una grilla de todos los conjuntos con todos los modelos de clasificación para saber cual obtiene los mejores resultados. La implementación de estos algoritmos fue tomada de la librería de sklearn [6].

Para cada uno de los modelos se hizo una búsqueda de grilla para encontrar los mejores parámetros. Los valores inlcuidos en la grilla se pueden ver en la tabla 7.1 Además se hizo un proceso de validación cruzada de *K-Folds* con 7 splits (grupos de 5) y 3 repeticiones. Al final de la búsqueda, para cada modelos se registró cual fue la iteración con el mejor resultado comparando así el puntaje *f1*.

Kmeans	
C	logspace(-3,3,7)
penalty	l1, l2
solver	liblinear
max_iter	300
Logistic Regression	
n_neighbors	2, 3, 4, 5, 6
weights	uniform, distance
p	1, 2, 3
Random Forest	
clf_n_estimators	10, 100, 200
clf_criterion	gini, entropy
clf_max_dept'	10, 100, 1000

Cuadro 7.1: Parámetros usados para cada modelo de predicción.

7.1. Resultados Kmeans

Una vez se determinaron los parámetros, se corrieron dos modelos de Kmeans. El primero, donde se iban a recolectar los centroides de los agrupamientos para realizar predicciones y, el segundo, donde no se tomaba el centroide sino que se tomaban elementos aleatorios del cluster para la predicción. Esto con el propósito de verificar si el representante más cercano al centro era tambien el mejor representante para la predicción. Para ambos, se hizo un barrido con los conjuntos de datos que entregó el agrupamiento de Kmeans desde 2 a 6. Para las predicciones de los 3 modelos que se registró el mejor modelo de todos fue cuando el Kmeans generó 5 clusters y el modelo de predicción fue random forest. Para este el F1 score fue de 0.826 bastante buen score. en comparación con los demás. Cuando se realizó la predicción con los metabolitos que no eran los centroides del cluster sino que eran elegidos aleatoriamente la mejor F1 score que se obtuvo fue de 0.649 con un Kmeans de 3 clusters utilizando random forest.

Kmeans de 2 a 6 Clusters Centriodes			
Metabolitos	94, 352, 393 (para 3 centros) y 252, 94, 468, 391, 77 (5 centros)		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 2, p: 1, weights: distance	0.6761904762	3
Random Forest	clf_criterion: gini, clf_max_depth: 10, clf_n_estimators: 200	0.8266666667	3
Logistic Reg	C: 1000, penalty: l1	0.7823809524	5

Cuadro 7.2: Resultados de predicción del algoritmo Kmeans con mejores resultados con clusters del 2 a 6 con representantes centiodes

Kmeans de 3 Clusters con Representantes Aleatorios			
Metabolitos	352, 80, 175, 386, 85, 471		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 5, p: 2, weights: uniform	0.52380952	3
Random Forest	clf_criterion: gini, clf_max_depth: 10, clf_n_estimators: 200	0.64904762	3
Logistic Reg	C: 0.001, penalty: l1	0.62857143	3

Cuadro 7.3: Resultados de predicción del algoritmo Kmeans de 5 clusters con representantes escogidos aleatoriamente

Kmeans de 3 Clusters con Representantes Aleatorios			
Metabolitos	352, 80, 175, 386, 85, 471		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 5, p: 2, weights: uniform	0.5619047619	5
Random Forest	clf_criterion: gini, clf_max_depth: 10, clf_n_estimators: 200	0.6466666667	5
Logistic Reg	C: 0.01, penalty: l2	0.5238095238	5

Cuadro 7.4: Resultados de predicción del algoritmo Kmeans de 3 clusters con representantes escogidos aleatoriamente

7.2. Resultados Mixturas Gaussianas

Para la validación de las Mixturas Gaussianas se tuvieron en cuenta diferentes formas de escoger representantes y por tanto, se hicieron pruebas en 6 conjuntos de datos distintos. Se usaron 3 agrupamientos: según su cercanía al centro y según su probabilidad de ser escogidos (con 2 y 3 centros), y se tomó el mejor de cada uno. Es decir, los más cercanos a cada centro o los más probables en quedar en un grupo. Luego para cada uno de los 3 se hizo una verificación con otros metabolitos, escogidos al azar, los cuales no son los más cercanos al centro, pero están entre los primeros 5. Esto se hizo con la intención de ver si en efecto el mejor representante arrojaría los mejores resultados. O si los mejores resultados serían un grupo más grande y no un único representante.

En general los metabolitos escogidos según su probabilidad tuvieron un mejor desempeño que aquellos que se escogieron según su cercanía al centro. El mejor puntaje fue el de *Random Forest* sobre los datos de 2 representantes de 2 grupos, el cual tuvo un puntaje *f1* de 0.8167. Se podría atribuir que estos tengan un mejor mejor desempeño el hecho de que los miembros con mayor probabilidad de ser de un grupo, no solo están más cercanos al centro de este, pero también tienen menos influencia de los centros de otros grupos. Además, se podría decir con cierta certeza que los datos escogidos como mejores representantes si se desempeñaron mejor que aquellos escogidos al azar, pero aun entre los mejores 5. También se puede observar que en todos los casos, los mejores resultados fueron del modelo de *Random Forest*.

Escogidos por cercania al centro			
Metabolitos	94, 352, 393		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 2, p: 1, weights: distance	0.6761904762	3
Random Forest	clf_criterion: entropy, clf_max_depth: 10, clf_n_estimators: 200	0.7257142857	3
Logistic Reg	C: 0.001, penalty: l1	0.6	3

Escogidos por probabilidad de quedar en el cluster			
Metabolitos	80, 252, 520		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 5, p: 3, weights: distance	0.6761905	3
Random Forest	clf_criterion: gini, clf_max_depth: 10, clf_n_estimators: 100	0.7866667	3
Logistic Reg	C: 10, penalty: l1	0.69524	3

Escogidos por probabilidad de quedar en el cluster			
Metabolitos	252, 520		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 3, p: 3, weights: uniform	0.6095238	2
Random Forest	clf_criterion: gini, clf_max_depth: 10, clf_n_estimators: 100	0.8166667	2
Logistic Reg	C: 0.001, penalty: l1	0.6285714	2

Cuadro 7.5: Resultados de predicción con los mejores representantes

7.3. Resultados Jerárquico

A la hora de validar el modelo del algoritmo jerárquico se ejecutaron varias instancias. Para el algoritmo jerárquico se decidió hacer con 3 clusters únicamente. Con estos tres clusters se ejecutaron

Escogidos por cercania al centro			
Metabolitos	135, 386, 384		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 5, p: 3, weights: uniform	0.51428571	3
Random Forest	clf_criterion: entropy, clf_max_depth: 10, clf_n_estimators: 100	0.68333333	3
Logistic Reg	C: 0.001, penalty: l2	0.55238095	3

Escogidos por probabilidad de quedar en el cluster			
Metabolitos	316, 270, 451		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 5, p: 3, weights: uniform	0.51428571	3
Random Forest	clf_criterion: gini, clf_max_depth: 10, clf_n_estimators: 100	0.67333333	3
Logistic Reg	C: 0.001, penalty: l1	0.62857143	3

Escogidos por probabilidad de quedar en el cluster			
Metabolitos	77, 175		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 5, p: 1, weights: uniform	0.60952381	2
Random Forest	clf_criterion: gini, clf_max_depth: 10, clf_n_estimators: 100	0.62	2
Logistic Reg	C: 0.001, penalty: l1	0.62857143	2

Cuadro 7.6: Resultados de predicción con los representantes escogidos al azar del top 20 %

3 instancias: la primera, donde se realizaron las predicciones con los metabolitos centroides (uno de cada cluster), la segunda con 3 metabolitos aleatorios (uno de cada cluster) y la última con 6 metabolitos representantes (2 centros de cada cluster). Se utilizaron únicamente 3 clusters porque

Jerarquico Aglomerativo Centroides			
Metabolitos	386, 85, 471		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 2, p: 1, weights: distance	0.67619047	3
Random Forest	clf_criterion: gini, clf_max_depth: 10, clf_n_estimators: 100	0.7557142857	3
Logistic Reg	C: 100, penalty: l1	0.6761904762	3

Cuadro 7.7: Resultados de predicción del algoritmo jerárquico con tres centriodes de 3 clusters

Jerarquico Aglomerativo No Centriodes			
Metabolitos	352, 80, 175		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 4, p: 1, weights: uniform	0.71428571	3
Random Forest	clf_criterion: entropy, clf_max_depth: 10, clf_n_estimators: 10	0.76238095	3
Logistic Reg	C: 100, penalty: l1	0.63809524	3

Cuadro 7.8: Resultados de predicción del algoritmo jerárquico con tres centriodes de 3 clusters

al momento de tener 5 o más se generaban clusters de un solo elemento. Estos se podrían llegar a considerarse datos atípicos; por tanto, lo que se decidió hacer fue reducir el número de clusters a 3 donde se iba a tener clusters más poblados, así fuesen menos. Los parámetros que se utilizaron para el algoritmo aglomerativo fueron Euclidean como afinidad y Single como vinculación. El mejor resultado lo tuvo la iteración con 6 representantes cuando se predijo con regresión lineal con un *F1* score de 0.7823809524. Aparte de este, los siguientes mejores resultados fueron todos de random forest donde con 3 centroides tuvo un score de 0.7557142857, con 3 representantes aleatorios tuvo 0.7623809524. A pesar de ser un resultado bueno no es el mejor en comparación comparado con los demás resultados que se obtuvieron de otros algoritmos de agrupamiento.

Jerarquico Aglomerativo Combinado			
Metabolitos	352, 80, 175, 386, 85, 471		
Método	Parametros	Puntaje *f1*	Número de Clusters
K-vecinos	n_neighbors: 4, p: 2, weights: distance	0.6952380952	3
Random Forest	clf_criterion: entropy, clf_max_depth: 10, clf_n_estimators: 100	0.7357142857	3
Logistic Reg	C: 1000, penalty: l1	0.7823809524	3

Cuadro 7.9: Resultados de predicción del algoritmo jerárquico con 6 representantes 2 por cluster

Capítulo 8

Análisis

8.0.1. Mejores Resultados

Una vez terminados todos los modelos se observó que el mejor resultado vino del agrupamiento hecho por *K-Means* con 3 grupos, el cual obtuvo un puntaje *f1* de 0.8267. Resulta interesante que el algoritmo, el cual es relativamente el más sencillo en complejidad, seleccionará los metabolitos con los mejores resultados. El modelo que precede en puntaje a este, es el de *Gaussian Mixtures* con dos centros y un puntaje *f1* de 0.81667. Estos dos modelos, tienen un comportamiento similar en su forma de agrupar, ya que lo hacen por cercanía a una media. Más aún, el modelo de *Gaussian Mixtures* implementa a K-Medias para la selección de centros. El hecho de que estos modelos con un comportamiento similar obtuvieron los mejores resultados no es gratuito. Significa que, gracias a la forma del conjunto de datos, resulta conveniente agrupar según la cercanía a una media. Además, en mayoría de los casos, la predicción hecha por *Random Forest* obtuvo los mejores resultados en comparación con los otros métodos, sin importar el agrupamiento. En la tabla 8.1 se registran los nombres de los metabolitos que usaron para la predicción en los dos modelos mencionados.

Modelo	**Nombre de metabolito**
Gaussian Mixtures (2 centros, escogido según probabilidad)	Creatinine, trans2Dodecenoylcarnitine
KMeans (3 centros, escogidos según cercanía a la media)	[PC (18:2/22:6)] 1-(9Z_12Z-octadecadienoyl)-2-(4Z_7Z_10Z_13Z_16Z_19Z-docosahexaenoyl)-sn-glycero-3-phosphocholine, L-Carnitine, L-Tyrosine

Cuadro 8.1: Representantes de metabolitos altamente correlacionados

8.0.2. Posible forma de conjunto de datos

La exploración de los modelos de agrupamiento ayudó a formar una idea de la distribución espacial de los metabolitos. Sobre todo, el algoritmo de *DBSCAN* y la búsqueda sobre sus parámetros ayudó a entender una posible forma del dataset. Se observó que el gran cúmulo de datos están en gran cercanía, tal que, la distancia del tercer vecino es casi que la misma que el veinteavo. En comparación, hubo un grupo mucho menor, de menos del 20 %, que se encontraba bastante alejado de este centro. Por tanto, la selección por densidad no resulta beneficiosa ya que estos metabolitos en la periferia

del gran cúmulo eran tomados como ruidosos. Se decidió que los metabolitos de interés podrían ser aquellos que están alejados y que presentan características más únicas, las cuales podrían ser mejores predictores del funcionamiento del tratamiento. Este comportamiento también se evidenció en la exploración de *K-Means* y de *Gaussian Mixtures* ya que el tamaño de los conjuntos era casi el mismo. Uno grande, en el cual estaba más del 80 % de los metabolitos y el resto en pequeños.

8.0.3. Metabolitos recurrentes

Durante las diferentes etapas de agrupamiento y en la selección de representantes se observo que algunos metabolitos persistían en diferentes métodos de agrupamiento. A continuación se señalan los metabolitos que se repetían en diferentes modelos:

Nombre de metabolito
Gaussian Mixutres (2 centros, escogido según probabilidad)
Creatinine
trans2Dodecenoylcarnitine
[PC (16:0)] 1-hexadecanoyl-sn-glycero-3-phosphocholine
[PC (18:2/22:6)] 1-(9Z_12Z-octadecadienoyl)-2-(4Z_7Z_10Z_13Z_16Z_19Zdocosahexaenoyl)-snglycero-3-phosphocholine
[PC (16:0/18:2)] 1-hexadecanoyl-2-(9Z_12Z-octadecadienoyl)-sn-glycero-3-phosphocholine
L-Proline
L-Citrulline

Cuadro 8.2: Metabolitos frecuentemente observados

Capítulo 9

Conclusiones

9.0.1. Objetivos

Se logró la implementación de cuatro modelos de agrupamiento para la selección de metabolitos. De cada uno de estos agrupamientos se hicieron selecciones de los metabolitos representantes teniendo en cuenta diferentes métricas como: distancia al centro más cercano o probabilidad de ser clasificado en un cierto. Así mismo, se logró reducir el ancho del conjunto de entrenamiento de 535 atributos a unos 2 a 5 atributos y aun así conseguir resultados prometedores. Se seleccionaron e implementaron diferentes modelos de clasificación para predecir los resultados. En base a esto, se pudo hacer un análisis de cuales agrupamientos y cuales formas de selección de los mejores representantes obtuvieron mejores resultados en la etapa de predicción. En general se logró cumplir con los objetivos estipulados en la sección de actividades.

9.0.2. Comparación con estudios anteriores

Cuando se obtuvieron todos los resultados, el mejor modelo que se pudo producir fue un modelo de *Random Forest* con un *F1* score fue de 0.826. Este es un puntaje prometedor, ya que el proyecto se enfocaba en la etapa de agrupamiento por encima de la de predicción. En términos de modelos de predicción, se eligieron tres modelos de baja complejidad; más como una métrica para comparar resultados del agrupamiento, que como un fin para el mejor modelo de predicción. Por tanto, al mirar el estudio de Zuluaga[7], el cual trabaja con el mismo conjunto de metabolitos, pero con modelos de mayor complejidad (Naive Bayes, Decision Trees y Support Vector Classification), se observan puntajes de hasta 0.93 para el *F1* score. Cabe resaltar que ese estudio utilizó un proceso para la reducción de la dimensionalidad más tradicional a comparación de la extracción de atributos con agrupamiento. En contraste, es interesante que se hayan logrado resultados prometedores sin tanto enfoque en la predicción y con un proceso de selección menos común y no supervisado. Además, el proceso de selección con agrupamiento tiene un gran aporte a entender mejor los datos, su forma, su relación con otros datos y su comportamiento. Teniendo esto en cuenta, sería de gran valor poder seguir con la predicción usando modelos de mayor complejidad.

Los resultados obtenidos también se pueden comparar con la investigación del Vargas[4] donde se hace un análisis estadístico para observar qué metabolitos podrían ser indicadores de mejora. Al hacer una comparación de estos metabolitos y los metabolitos que fueron utilizados como representantes en la predicción se observó que había un metabolito que fue representante del mejor modelo y fue uno de los que en la investigación de Vargas sobresale. El metabolito es concretamente L-carnitine que fue representante en el modelo de predicción de *Kmeans*.En la investigación de Vargas se encontró que este metabolito consistentemente decrecía en cantidad en los casos exitosos. Esto

explicaría porque esté modelo tuvo buenos resultados. Aparte de este metabolito no se encontraron más similitudes.

9.0.3. Valor de resultados

Aunque las métricas resultan prometedoras, con puntajes que no son muy altos, pero dan a creer que sí puede haber una correlación entre la cantidad de un metabolito y el funcionamiento del tratamiento, es difícil decir que los resultados de la predicción dan espacio a extraer conclusiones certeras o con suficiente peso. No solo porque no es un puntaje muy alto, sino porque el conjunto de entrenamiento es muy poco; con solo unos 35 datos la población. A pesar de esto, lo que sí ofrece este análisis es la facilidad de poder seguir llevando a cabo un entrenamiento del modelo con muy pocos parámetros. A comparación de los resultados de estudios anteriores, no destacan mucho los números que se lograron. El estudio también presenta otras etapas como la extracción de metabolitos altamente correlacionados los cuales podría proveer información interesante.

9.0.4. Trabajo futuro y consideraciones

Después de observar los resultados de *DBSCAN*, donde casi todos los metabolitos quedaron en un gran conjunto y el resto fueron tomados como ruido, podría resultar de interés hacer un estudio solo sobre este conjunto grande de metabolitos. Podría ser el caso que existan subconjuntos dentro de este que se pierden en los otros modelos, ya que los metabolitos del conjunto grande están tan cercanos entre sí. A futuro, también se podría énfasis en la etapa de predicción utilizando modelos que no se tuvieron en cuenta en este estudio, pero si en el de Zuluaga como: Naive Bayes, Decision Trees o Support Vector Classification. Además, valdría la pena otros métodos de selección de atributos, que no estén basados en el mejor representante de cada grupo, ya que hay una posibilidad de que los métodos usados para la selección de mejores representantes para cada grupo, no asegurara un mejor desempeño en la fase de predicción. Es posible que los mejores metabolitos para la predicción no son aquellos que representan un cluster, sino los que presentan características mas únicas y se encuentran mas alejados del resto.

Bibliografía

[1] Ministerio de Saludo, "GUÍA PARA LA ATENCIÓN CLÍNICA INTEGRAL DEL PACIENTE CON LEISHMANIASIS", Convenio de Cooperación Técnica con el Ministerio de la Protección Social Nro. 256 de 2023 y Nro. 237 de 2010

[2] S. Hunta, N. Aunsri and T. Yooyativong, "Drug-Drug Interactions prediction from enzyme action crossing through machine learning approaches,"2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2015, pp. 1-4, doi: 10.1109/ECTICon.2015.7207126

[3] Bombardier, C.H., Divine, G.W., Jordan, J.S. et al. Minnesota Multiphasic Personality Inventory (MMPI) cluster groups among chronically ill patients: Relationship to illness adjustment and treatment outcome. J Behav Med 16, 467–484 (1993). https://doi.org/10.1007/BF00844817

[4] Vargas DA, Prieto MD, Martínez-Valencia AJ, Cossio A, Burgess KEV, Burchmore RJS and Gómez MA (2019) Pharmacometabolomics of Meglumine Antimoniate in Patients With Cutaneous Leishmaniasis. Front. Pharmacol. 10:657. doi: 10.3389/fphar.2019.00657

[5] G. Stegmayer, D. H. Milone, L. Kamenetzky, M. G. Lopez and F. Carrari, .ª Biologically Inspired Validity Measure for Comparison of Clustering Methods over Metabolic Data Sets,ïn IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 9, no. 3, pp. 706-716, May-June 2012, doi: 10.1109/TCBB.2012.10

[6]F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[7] S. Zuluaga, "Aprendizaje de Máquina Aplicado a la Predicción del Éxito del Tratamiento de la Leishmaniasis," Pontificia Universidad Javeriana Cali, 2022.

[8] Ester, M., Kriegel, H. P., Sander, J., Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In kdd (Vol. 96, No. 34, pp. 226-231). [9]K. Arvai, "Knee LocatorÇopyright Revision 13b9c17d. 2020. [10] Centroid Neural Network: An Efficient and Stable Clustering Algorithm Accessed Feb. 18, 2023 [Online] Available: https://towardsai.net/p/l/centroid-neural-network-an-efficient-and-stable-clustering-algorithm