Análisis de clasificación para identificar características relevantes en la detección de operaciones sospechosas en Bancolombia

Cristian David Mariaca Rueda

Universidad de Antioquia
Facultad de Ciencias Exactas y Naturales
Instituto de Matemáticas
Medellín, Colombia
2022

Análisis de clasificación para identificar características relevantes en la detección de operaciones sospechosas en Bancolombia

Cristian David Mariaca Rueda

Trabajo de grado presentado como requisito parcial para optar al título de:
Estadístico

Emerson Yarce Carmona
Orientador Externo, Bancolombia
eyarce@bancolombia.com.co

María Eugenia Castañeda López
Orientador Interno, Instituto de Matemáticas
maria.castaneda@udea.edu.co

Universidad de Antioquia
Facultad de Ciencias Exactas y Naturales
Instituto de Matemáticas
Medellín, Colombia
2022

Dedicado a:
Mi amparo y fortaleza, Dios.

Agradecimientos

Agradezco a Dios, mi familia, y especialmente a mi abuela, sin su apoyo y amor no hubiera sido posible culminar estar carrera.

Además agradezco a la Universidad de Antioquia, tanto a mis compañeros como a los profesores, especialmente a la profesora María Eugenia Castañeda que me acompaño esta etapa final como estudiante.

También agradezco al Grupo Bancolombia, por brindarme la oportunidad de realizar las practicas con ellos, y seguir creyendo en mi dándome mi primera oportunidad como profesional.

Finalmente,a mi jefe, Emerson Yarce Carmona y a mis compañeros de trabajo, Carlos Andrés Gallón y Laura María Barrientos, quiero agradecerles por haberme compartido la idea de este proyecto, y por el acompañamiento en mi etapa como practicante.

Resumen

Grupo Bancolombia participa activamente en la lucha contra el Lavado de Activos y Financiación de terrorismo (LAFT), por este motivo, desde la gerencia de pymes y empresas que hace parte de la vicepresidencia de Cumplimiento, se deseaban conocer las variables, atributos o características más influyentes de los estados financieros en los clientes con Reportes de Operaciones Sospechosas (ROS) y los clientes no reportados, con el fin de generar alertas que permitieran llevar a cabo un monitoreo a los clientes con posibles operaciones sospechosas. Se investigaron técnicas de análisis multivariado como ACP, t-SNE y UMAP usando machine learning para hacer la clasificación de los clientes de interés según los estados financieros, y para identificar las características, variables o atributos más influyentes se empleó la prueba no paramétrica U de Mann-Whitney.

Palabras clave: ACP, análisis multivariado, estados financieros, LAFT, machine learning, ROS, t-SNE, U de Mann-Whitney, UMAP.

Abstract

Grupo Bancolombia actively participates in the fight against Money Laundering and Financing of Terrorism (LAFT), for this reason, from the management of SMEs and companies that is part of the Vice Presidency of Cumplimiento, they wanted to know the variables, attributes or characteristics more influencers of financial statements in clients with Suspicious Transaction Reports (ROS) and unreported clients, in order to generate alerts that would allow monitoring clients with possible suspicious transactions. Multivariate analysis techniques such as PCA, t-SNE and UMAP were investigated using machine learning to classify the clients of interest according to the financial statements, and to identify the most influential characteristics, variables or attributes, the non-parametric test U Mann-Whitney.

Keywords: financial statements, LAFT, machine learning, multivariate analysis, PCA, ROS, t-SNE, Test U Mann-Whitney, UMAP.

Análisis de clasificación para identificar características relevantes en la detección de operaciones sospechosas en Bancolombia

Cristian David Mariaca Rueda*

Abril de 2022

Contenido

1. Introducción

Con el paso de los años y la propagación de la tecnología, los delincuentes detrás del Lavado de Activos y Financiación de terrorismo (LAFT) han desarrollado nuevas formas de delinquir que dificultan su identificación, haciéndolos en muchas ocasiones, imperceptibles ante los diferentes organismos encargados de hacer frente a esta problemática. Esta situación, ha llevado a las distintas entidades bancarias a desarrollar nuevos métodos que permitan identificar, anticipar y desmantelar operaciones sospechosas de delitos relacionados con LAFT, y de este modo se logran llevar a cabo desvinculaciones y procesos judiciales en los implicados con todas estas actividades criminales.

Bancolombia es un grupo financiero multinacional que participa activamente en la lucha contra el LAFT, velando por la seguridad no solo de sus clientes, sino que también por la seguridad del país. Este compromiso con la seguridad, lo llevan a cabo por medio de monitoreos, en los cuales se aseguran de que las operaciones que se realizan a diario por los diferentes canales del banco cumplan con las políticas de seguridad, y que además se ajusten al comportamiento habitual de cada cliente.

Actualmente, Grupo Bancolombia cuenta con dos tipos de clientes, aquellos que poseen estados financieros y los clientes que no tienen estados financieros. Los estados financieros son informes que sirven para observar el estado de una empresa en un periodo determinado, pues por medio de estos informes se puede conocer la rentabilidad y solvencia de las compañías, haciendo un análisis tanto de la información económica como patrimonial [4]. A su vez, los clientes con estados financieros se dividen en dos subgrupos, los que tienen Reportes de Operaciones Sospechosas (ROS) y los que no tienen este tipo de reportes. Los ROS son *"aquellas operaciones que por su número, cantidad o características no se enmarcan en el sistema y prácticas normales del negocio, de una industria o de un sector determinado y, además que de acuerdo con los usos y costumbres de la actividad que se trate, no ha podido ser razonablemente justificada"* [2]. Los ROS no son denuncias formales, y se basan en la evidencia de los analistas para reportar clientes sospechosos a la Unidad de Información y Análisis Financiero (UIAF). A su vez, la UIAF envía alertas a la fiscalía general de la nación para que finalmente se determine una sentencia penal.

La gerencia de cumplimiento para PYMES y empresas de Bancolombia tiene un área encargada del análisis e investigación de los clientes con posibles operaciones sospechosas. Desde esta área de investigación se deseaban conocer las variables, atributos o características más influyentes de los estados financieros en los clientes con ROS y los clientes no reportados, esto con el fin de generar alertas que permitieran llevar a cabo un monitoreo a los clientes con posibles operaciones sospechosas.

Para cumplir con este objetivo, se investigaron metodologías de análisis multivariado y machine learning, a fin de encontrar una técnica que pudiera clasificar de forma correcta los grupos de interés. Para el análisis multivariado se emplearon tres algoritmos de reducción de dimensionalidad los cuales fueron, análisis de componentes principales, t-SNE y UMAP. Además, se empleó la prueba no paramétrica U de Mann-Whitney para identificar las variables en las que se podrían estar basando los algoritmos al momento de hacer la clasificación de los clientes.

El documento se guía por el siguiente esquema. En la sección 2 se describe la parte teórica de los métodos ya mencionados. En la sección 3 se explica la metodología empleada y el detalle del uso de los algoritmos. En la sección 4 se muestran los resultados las técnicas, y finalmente en la sección 5 se exponen las conclusiones y sugerencias del trabajo realizado.

2. Marco teórico

ACP: análisis de componentes principales

El método de componentes principales permite transformar un conjunto de variables, en un nuevo conjunto de variables denominadas componentes principales que son combinación lineal de las variables originales. Las componentes principales (CP) se caracterizan por estar incorrelacionadas entre sí. La cantidad de información incorporada en cada componente depende de su varianza, es decir, entre mayor sea la varianza de la componente, mayor será la cantidad de información explicada por la componente. La cantidad de componentes que se necesitan para explicar la mayor parte de la variabilidad de un conjunto de datos multivariado depende del grado de correlación de las variables, pues a mayor correlación, se necesitara un menor número de componentes. Se puede decir que el objetivo que se busca al aplicar el ACP, es la reducción de dimensionalidad de los datos [3].

La utilidad de ACP se puede resumir así [10]:

1. Representar óptimamente en un espacio de dimensión más reducido que el original, observaciones de un espacio general p-dimensional.
2. Transformar las variables originales, en nuevas variables no correlacionadas, facilitando la interpretación y representación de los datos.

t-SNE: t-Distributed Stochastic Neighbor Embedding

t-SNE es una técnica no lineal, empleada para la exploración y visualización de datos de alta dimensión [12]. Bajo este método se mide la distancia entre cada observación del conjunto de datos y cualquier otra observación, y luego se aleatoriza la posición de estas observaciones generalmente en dos nuevos ejes. Las observaciones se mezclan iterativamente alrededor de estos nuevos ejes hasta que sus distancias entre sí, en el espacio bidimensional, sean lo más similares posibles a las distancias en el espacio de alta dimensión. Tanto en el espacio de alta dimensión como en el de baja dimensión las distancias se convierten en probabilidades, y el objetivo es optimizar la posición de las observaciones en baja dimensión minimizando la divergencia entre los dos vectores de probabilidad.

Descripción del método

Dados x_i y x_j como dos puntos en coordenadas cartesianas, la distribución de densidad de probabilidad de datos vecinos para x_i se asume como una función gaussiana centrada en x_i con varianza σ_i. La probabilidad de que x_j sea seleccionado como vecino de x_i es una probabilidad condicional calculada como:

$$p_{j|i} = \frac{exp(-\left\|x_i - x_j\right\|^2/2\sigma_i^2)}{\sum_{k\neq i} exp(-\|x_i - x_k\|^2/2\sigma_i^2)}$$

La probabilidad condicional anterior es una medida no simétrica, por lo tanto, como $p_{i|j}$, y $p_{j|i}$ suelen ser diferentes, la similitud de los puntos x_i y x_j se calcula como la probabilidad conjunta definida como:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}$$

En el espacio de baja dimensión, la probabilidad conjunta que describe la similitud se calcula para y_i e y_j como contrapartes de las estructuras de alta dimensión x_i y x_j. En el método t-SNE, la distribución t de Student con un grado de libertad se emplea para calcular la probabilidad conjunta entre y_i y y_j, como:

$$q_{ij} = \frac{(1+\|y_i - y_j\|^2)^{-1}}{\sum_{k \neq i}(1+\|y_i - y_k\|^2)^{-1}}$$

Si los puntos del mapa y_i e y_j modelan correctamente la similitud entre los puntos de alta dimensión x_i y x_j, la probabilidad conjunta q_{ij} debería ser cercana a p_{ij}. Por lo tanto, el objetivo del método t-SNE es encontrar una representación de baja dimensión que minimice la diferencia entre q_{ij} y p_{ij} para todos los puntos i y j.

La forma de comparar las diferencias entre los datos de alta dimensión y las representaciones de baja dimensión es por medio de la divergencia de Kullback-Leibler (KL) en todos los puntos para construir las funciones de costo **C**, y así poder evaluar la proyección de la estructura de alta dimensión a la baja. Las funciones de costo están dadas por [14]:

$$C = KL(P||Q) = \sum_{i \neq j} p_{ij} \cdot log \frac{p_{ij}}{q_{ij}}$$

Teniendo en cuenta la teoría explicada anteriormente, se puede decir que t-SNE se ejecuta en dos pasos [6]:

1. Se construye una distribución de probabilidad sobre parejas de muestras en el espacio original, de forma tal que las muestras más semejantes reciben alta probabilidad de ser escogidas, mientras que las muestras muy diferentes o con distancias muy amplias reciben baja probabilidad de ser escogidas. Este es el paso que consume más tiempo y requiere de una mayor capacidad computacional en el método t-SNE.

2. t-SNE lleva los puntos del espacio de alta dimensionalidad al espacio de baja dimensionalidad, y minimiza la denominada divergencia Kullback-Leibler entre las dos distribuciones con respecto a las posiciones de los puntos en el mapa.

Para este proyecto, los argumentos de interés en el algoritmo de t-SNE fueron [7]:

- **n components:** Dimensión del conjunto transformado.
- **perplexity:** La perplejidad está relacionada con el número de vecinos más cercanos. Los conjuntos de datos más grandes generalmente requieren una mayor perplejidad. Generalmente este valor esta entre 5 y 50.
- **n iter:** Número máximo de iteraciones para la optimización. Debería ser, por lo menos, 250.

UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction

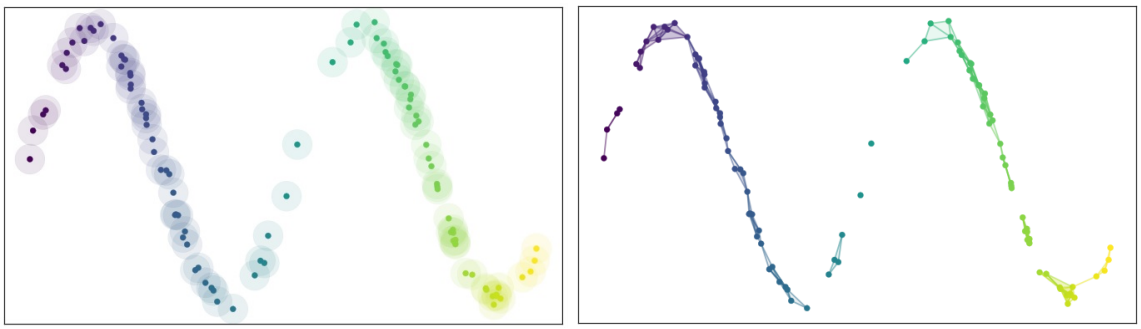
UMAP es un algoritmo de reducción de dimensionalidad no lineal [8]. UMAP se basa en la construcción de dos representaciones topológicas de los datos, una en el espacio original de alta

dimensión y otra en un espacio de baja dimensión, generalmente bidimensional. En el espacio original, la representación topológica busca aproximar el Mainfold sobre el cual se supone se encuentran los datos. En el espacio bidimensional la representación topológica inicia con valores aleatorios. A partir de estas dos representaciones, se optimiza la posición de los puntos en baja dimensión minimizando la entropía cruzada de las dos representaciones.

Descripción del método

Los complejos simples son la base de la construcción de los espacios topológicos, pues estos complejos se unen entre sí, formando bloques combinatorios que conforman el espacio topológico. Estos bloques también son llamados simplex, y puede construir objetos K dimensionales.

Ahora, una cubierta abierta es una familia de conjuntos cuya unión es el espacio completo y, un complejo de Cech es una forma combinatoria de convertir estas cubiertas en un complejo simplicial. Como los datos de origen se encuentran en un espacio métrico, una forma de aproximar una cubierta abierta es crear círculos de radio fijo alrededor de cada dato (imagen 1). De este modo, se puede representar los complejos 0-simplex, 1-simplex y 2-simplex (imagen 2). Para encontrar una representación de baja dimensión de la similitud topológica de los datos, se debe emplear el complejo Vietoris-Rips.

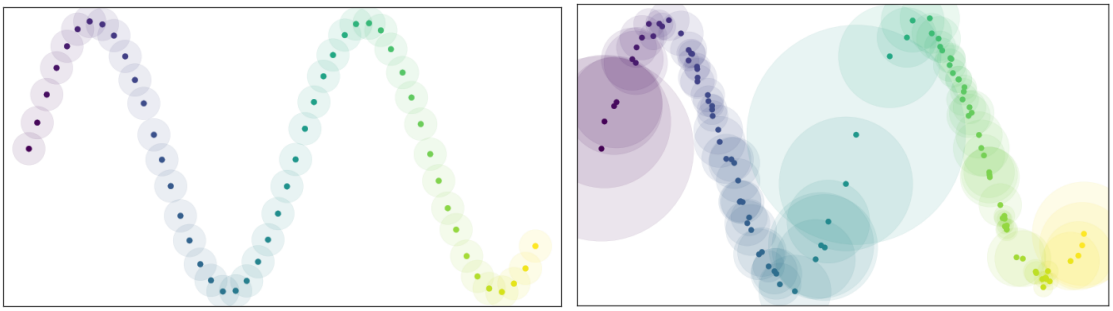


(a) Imagen 1 (b) Imagen 2

Figura 1: McInnes, L. (2018). How UMAP Works [Ilustración]. Recuperado de https://umap-learn.readthedocs.io/en/latest/how_umap_works.html

En teoría, lo explicado anteriormente funciona bastante bien, pero en la práctica con datos reales se presentan algunos problemas. Uno de esos problemas tiene que ver con el radio del círculo que se forma alrededor de cada dato, pues si el radio llega a ser muy pequeño o grande, no se cumpliría con la agrupación adecuada para los datos. Para este problema hay una solución, y es que los datos estén distribuidos uniformemente, pues en este caso seleccionar el radio sería mucho más fácil empleando las distancia promedio entre los datos (imagen 3).

De aquí surge otro problema, los datos reales no se comportan de esta forma. Por medio de la geometría riemanniana se parte de la suposición de que efectivamente los datos están distribuidos uniformemente, y esto nos permite calcular una noción local de distancia para cada punto. Esta distancia está determinada por un círculo de radio 1 alrededor de cada punto y se extiende hasta el K-esimo vecino más cercano de ese punto (imagen 4). De este modo, cada punto tendría una función de distancia única.



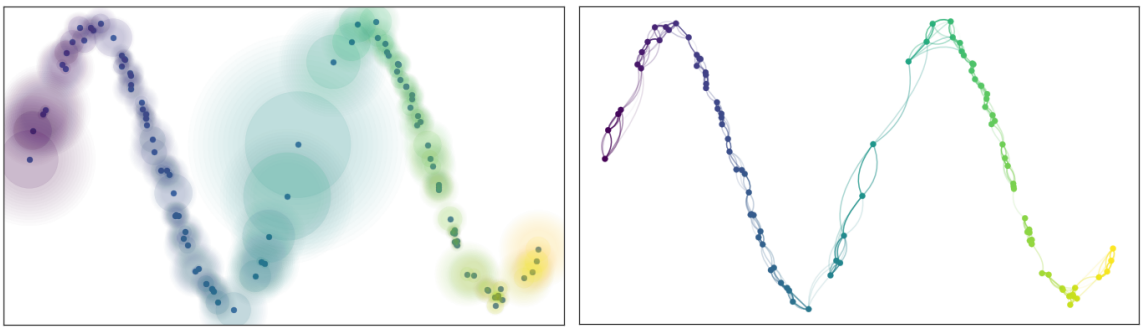
(a) Imagen 3 (b) Imagen 4

Figura 2: McInnes, L. (2018). How UMAP Works [Ilustración]. Recuperado de https://umap-learn.readthedocs.io/en/latest/how_umap_works.html

Luego, la elección de K determina que tan localmente se desea estimar la métrica de Riemann. Dependiendo de si K es grande o pequeño, se conservan los detalles más finos de las agrupaciones. Otra de las ventajas de usar esta métrica es que en realidad se tiene un espacio métrico local asociado a cada punto. La elección de este K depende en gran medida al tamaño del conjunto de datos que se desea analizar.

Otro de los problemas que surge, es que al aplicar el proceso que hasta el momento se ha explicado, muchos de los puntos quedan totalmente aislados. Para darle solución a este problema se emplea la conectividad local que se centra en la diferencia de distancias entre vecinos más cercanos en lugar de la distancia absoluta, por lo tanto, cada punto debe estar relacionado con otro (imagen 5).

El ultimo problema que surge es que las métricas locales no son compatibles. Como cada punto tiene su propia métrica, la distancia puede variar de la perspectiva en que se analice. Se tienen los puntos a y b, entonces se debe tener un solo borde de peso combinando $a+b-a\cdot b$, por lo tanto, se puede emplear la idea de combinación de peso del borde para unir todos los conjuntos simpliciales y terminar con un único complejo simplicial (imagen 6). De esta forma se obtiene una representación de los datos.



(a) Imagen 5 (b) Imagen 6

Figura 3: McInnes, L. (2018). How UMAP Works [Ilustración]. Recuperado de https://umap-learn.readthedocs.io/en/latest/how_umap_works.html

Ahora se debe convertir esto en una representación de baja dimensión. En este punto surgen dos preguntas:

1. ¿Como representamos la estructura topológica de una representación de baja dimensión?
2. ¿Como encontrar una buena representación?

Para dar solución a la primera pregunta se debe seguir el mismo procedimiento que se explicó anteriormente, simplemente que en esta ocasión los datos no estarán en una variedad. Así se tendrá una representación de baja dimensión que se encuentra en una variedad muy particular en el espacio euclidiano. La idea también es que la distancia al vecino más cercano fuera globalmente verdadera en la variedad a medida que se hace una buena optimización hacia una representación de baja dimensión.

La segunda pregunta depende de que tan cerca se haya encontrado una coincidencia de las estructuras topológicas difusas. Básicamente, esto se convierte en problema de optimización.

Ambas estructuras topológicas que se comparan comparten los mismo 0-simplex, por lo tanto, se están comparando los dos vectores de probabilidad indexados por los 1-simplex. Como estas variables son Bernoulli se debe emplear la entropía cruzada.

Suponga que el conjunto de todos los 1-simplex posibles es **E**, y se tienen funciones de peso tales que $\omega_h(e)$ es el peso de 1-simplex e en el caso de alta dimensión y $\omega_l(e)$ es el peso e en el caso de baja dimensión, de este modo, la entropía cruzada seria:

$$\sum_{e\in\mathbf{E}} \omega_h(e)\cdot log(\tfrac{\omega_h(e)}{\omega_l(e)}) + (1-\omega_h(e))\cdot log(\tfrac{1-\omega_h(e)}{1-\omega_l(e)})$$

donde:

- $\omega_h(e)\cdot log(\frac{\omega_h(e)}{\omega_l(e)})$ proporciona una fuerza de atracción entre los e.
- $(1-\omega_h(e))\cdot log(\frac{1-\omega_h(e)}{1-\omega_l(e)})$ proporciona una fuerza repulsiva entre los puntos e cuando $\omega_h(e)$ es pequeño.

Este proceso permitirá que la representación de baja dimensión muestre con precisión la topología de los datos de origen.

Finalmente, se tienen todos los elementos necesarios para construir el algoritmo de UMAP. Esto se hace en dos fases. En la primera fase se construye la representación topológica, y en la segunda fase se optimiza la representación de baja dimensión.

Los parámetros de interés en este proyecto son:

- **n neighbors:** por medio de este parámetro se equilibra la estructura local frente a la global en los datos. Valores bajos de este parámetro hacen que UMAP se concentre en una estructura muy local, caso contrario el algoritmo buscara vecindarios más grandes, pero se pierden los detalles más finos de los datos.
- **min dist:** este parámetro controla la restricción en la agrupación de los puntos, es decir, proporciona la distancia mínima entre los datos. Valores bajos del parámetro dan lugar a agrupaciones más concentradas, caso contrario, UMAP se centrará en la preservación de la estructura topológica amplia.

Test U de Mann-Whitney

U de Mann-Whitney es una prueba no paramétrica y se emplea para comparar las medianas de dos muestras independientes de tamaño arbitrario. Para aplicar la prueba se deben cumplir dos condiciones [5]:

1. No se necesita una distribución especifica
2. Las observaciones son variables ordinales

Las hipótesis por probar son las siguientes:

H_0: no hay diferencias en las medianas de ambos grupos
H_1: hay diferencias en las medianas de ambos grupos

El estadístico U se construye a partir de la suma de rangos R_i de las observaciones de las dos muestras [13]:

$$U_1 = n_1 n_2 + \frac{n_1(n_1+1)}{2} - R_1$$
$$U_2 = n_1 n_2 + \frac{n_2(n_2+1)}{2} - R_2$$

Donde n_1 y n_2 son los tamaños de muestra.De este modo, la prueba U está dada por:

$$U = min(U_1, U_2)$$

Si los tamaños de muestra son grandes, el estadístico U se puede aproximar a una distribución normal con parámetros:

$$\mu_U = \frac{n_1 n_2}{2}$$
$$\sigma_U^2 = \frac{n_1 n_2(n_1+n_2+1)}{12}$$

Finalmente, el estadístico de prueba es:

$$Z = \frac{U-\mu_U}{\sigma_U}$$

3. Metodología

La metodología del trabajo se llevo a cabo en ocho etapas. La primera etapa consistió en el entendimiento del problema y la recepción de la base de datos. La segunda etapa consistió en un análisis exploratorio de los datos. Las etapas que van de la tres a la siete consistieron en la aplicación de los algoritmos de reducción de dimensionalidad. La ultima etapa fue la prueba no paramétrica para la identificación de variables con diferencias estadísticamente significativas.

3.1. Entendimiento del problema y base de datos

Para el entendimiento del problema, se realizaron sesiones de estudio sobre lectura e interpretación de estados financieros. Posterior a esto, se recibieron los datos a analizar. La base en un inicio constaba de 1'533,885 de registros y 36 variables, incluida la variable etiqueta de los clientes que tenían o no tenían reportes de operaciones sospechosas. Por cuestiones de confidencialidad las variables fueron llamadas V1, V2, V3, así sucesivamente hasta V37 que sería la etiqueta de los datos.

3.2. Análisis exploratorio de datos

El análisis exploratorio de la base de datos, al igual que la aplicación de los algoritmos de reducción de dimensionalidad y la prueba no paramétrica, se realizaron empleando el lenguaje de programación Python desde Jupyer Notebook.

En principio, se hizo un recuento de los registros con ROS y los registros sin ROS, para evidenciar que porcentaje de datos pertenecían a cada uno de los grupos. además, se empleó la prueba de Kolmogorov-Smirnov para verificar si las variables se distribuían normalmente.

También, se empleó una matriz de correlación para identificar las variables que tenían mayor correlación y, que podrían estar aportando información redundante. Se estableció un criterio empírico en un intervalo de 0.9 a 1, donde se eliminaron las variables cuya correlación lineal estaba en este intervalo.

Posteriormente, se eliminaron las filas que tenían registros con valores faltantes. Luego de esto, la base de datos se redujo a 749,253 filas.

Como se tenían variables con diferentes tipos de medidas (razones, porcentajes, efectivo, días), se hizo una estandarización de los datos (a excepción de la etiqueta), de forma que su media era cero y su varianza uno.

3.3. ACP

El primer algoritmo que se implemento fue ACP, donde no se introdujo la variable etiqueta de los ROS y no ROS. Como se deseaba visualizar el conjunto de variables en un plano bidimensional, se seleccionaron dos componentes y se realizó el grafico para ver como lucían los datos y, ver si el algoritmo era capaz de hacer una correcta discriminación de ambos grupos.

3.4. t-SNE

Posteriormente, se empleó t-SNE donde tampoco se introdujo la variable etiqueta de los datos. Para un correcto funcionamiento de t-SNE, se extrajo una muestra de 30,000 datos con los cuales se entrenó el algoritmo. También se seleccionaron dos componentes para la visualización de los datos y, además, por el tamaño de la muestra, se seleccionó una perplejidad de 50 y un número máximo de 1000 iteraciones para la optimización. Al igual que con ACP, la idea era observar si t-SNE podía clasificar de forma correcta los grupos de interés.

3.5. ACP con t-SNE

En esta etapa se hizo la combinación de las dos técnicas ejecutadas anteriormente. Con la misma muestra de 30000 registros se aplicó ACP con 5 componentes, pues con esa muestra se alcanzaba a explicar un gran porcentaje de la variabilidad de los datos. Posteriormente, se tomaron esos 5 componentes del ACP y, se llevaron a t-SNE, transformándolos en dos componentes, y haciendo uso de los mismos valores de los argumentos de perplejidad e iteraciones utilizados anteriormente. Se deseaba observar cómo funcionaban los algoritmos en conjunto y, si se podían mejorar los resultados obtenidos de forma individual con cada uno de ellos.

3.6. UMAP no supervisado

Para UMAP no supervisado tampoco se indicó al algoritmo cual era la variable etiqueta de los datos. Para este caso también se empleó la misma muestra de 30,000 datos tomada anteriormente. Se entreno el algoritmo para graficar los resultados en un plano bidimensional y, se compararon los resultados con los obtenidos anteriormente para saber que algoritmo se aproximaba más a la clasificación deseada.

3.7. UMAP supervisado

Se empleo nuevamente UMAP, pero en este caso se hizo un balance de la variable etiqueta para compensar la diferencia que había entre los registros que eran ROS y los que no lo eran ROS. Se extrajeron los $n = 5,249$ ROS que quedaron en la base de datos después de hacer la limpieza de los datos, adicionalmente se tomó una muestra aleatoria del mismo tamaño n para extraer los no ROS. De este modo, la variable etiqueta quedo equilibrada para ambos subgrupos y, se procedió a hacer el análisis de los 10,498 registros extraídos. Como era aprendizaje supervisado, el algoritmo se entrenó con la variable etiqueta. Para este caso, se buscó que la clasificación de los clientes fuera lo mejor posible, agrupándolos en dos conglomerados perfectamente separados para cada grupo. Al algoritmo se le indico un valor de 50 vecinos y una distancia mínima de 0.5. Esto es un punto medio, pues a pesar de que no se concentra en una estructura topológica tan local, tampoco lo hace de forma muy global.

3.8. Prueba U de Mann-Whitney

Finalmente, teniendo en cuenta los resultados obtenidos anteriormente con las técnicas de reducción de dimensionalidad, se empleó la prueba no paramétrica U de Mann-Whitney para hallar las variables que presentaban diferencias estadísticamente significativas entre ambos grupos, y así, poder tener una idea de cuales variables tenían un mayor peso a la hora de hacer la clasificación de los grupos al implementar los algoritmos de reducción de dimensionalidad.

4. Resultados

4.1. Análisis exploratorio de datos

Inicialmente se contaron con 1’533,885 registros y 36 variables. Se hizo un recuento de los registros sin ROS y los registros con ROS:

Etiqueta	Conteo
No ROS	1’524,224
ROS	9,661

Tabla 1: Conteo no ROS y ROS

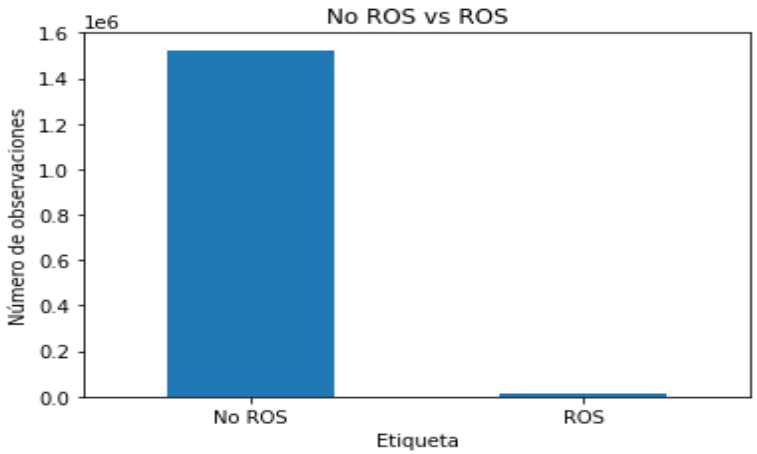


Figura 4: No ROS vs ROS

Evidentemente había un desequilibrio en la variable etiqueta de los datos, pues el porcentaje de ROS era solo un 0.62 % del total de los datos.

Posteriormente se hizo una matriz de correlación lineal entre las 35 variables (no se tuvo en cuenta la variable etiqueta) con el fin de eliminar las variables que podían estar aportando información redundante. Se tomaron solo las variables cuya correlación lineal era mayor a 0.9 y, del par de variables seleccionadas, se eliminaron las que tuvieran menor información, es decir, en las que había más datos faltantes. De este modo se eliminaron las variables V13, V15, V17, V22, V26, V27 y V30, V34. Esta selección dejo 28 variables en la base de datos, contando la variable etiqueta.

Luego, se procedió con el tratamiento de los datos faltantes. Se decidieron eliminar los registros con valores faltantes, lo que dejo un total de 749,253 filas.

Nuevamente se realizó un conteo de los no ROS y los ROS para observar cuantos quedaban después de la eliminación de las filas con valores faltantes:

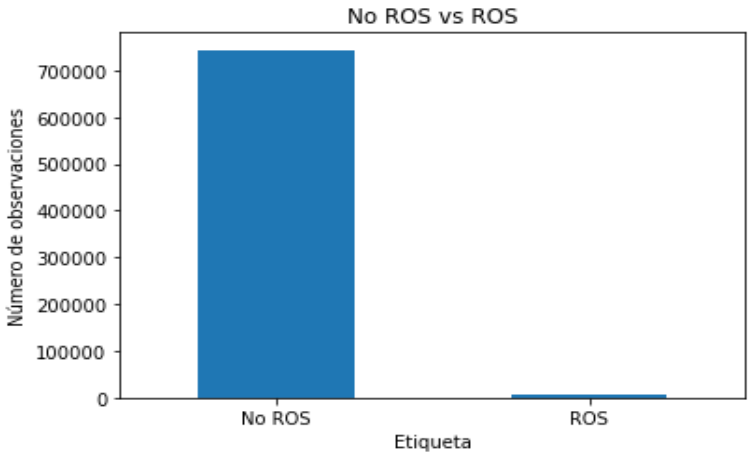


Figura 5: No ROS vs ROS

Etiqueta	Conteo
No ROS	744,004
ROS	5,249

Tabla 2: Conteo no ROS y ROS

luego el porcentaje de ROS era 0.7 % del total de los datos. Similar al porcentaje anterior.

Finalmente, se hizo la estandarización de los datos (sin tener en cuenta la variable etiqueta), pues había diferentes unidades de medida como, días, porcentajes, razones, efectivo, entre otras. Con la estandarización de los datos, los datos quedaron con media cero y varianza uno.

4.2. ACP

Se aplico el ACP a la base de datos normalizada y se seleccionaron dos componentes para el gráfico bidimensional. Se obtuvo la variabilidad explicada por las componentes:

Componente principal	Variabilidad explicada
CP 1	0.148
CP 2	0.08

Tabla 3: variabilidad explicada por las componentes CP1 y CP2

En conjunto, las dos componentes explicaban a penas el 22.91 % de la variabilidad de los datos.

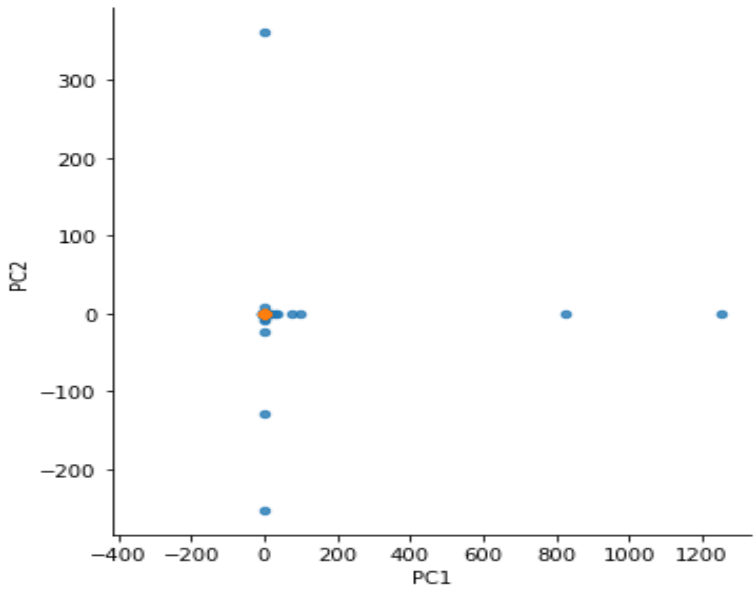


Figura 6: ACP 2D
naranja: ROS - azul: no ROS

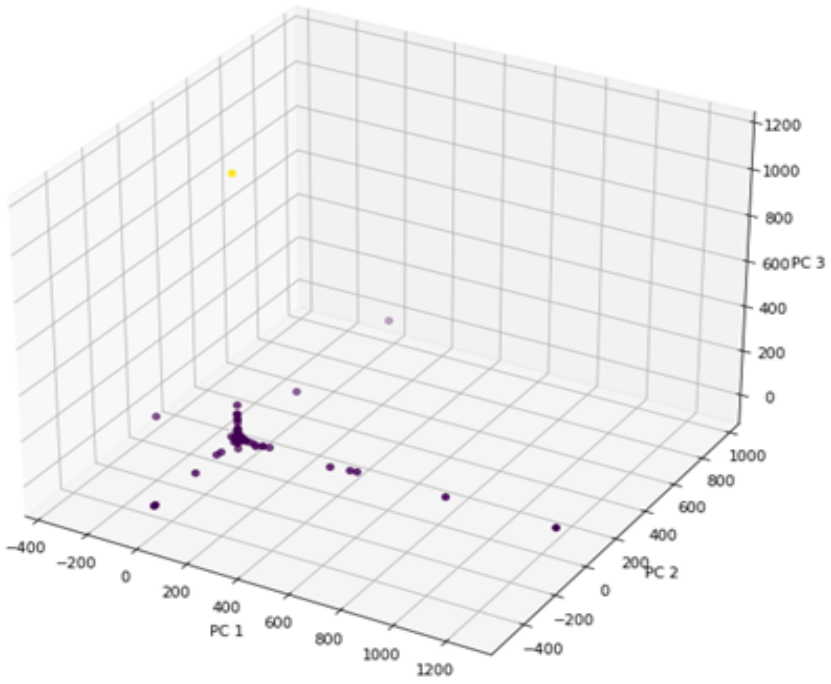


Figura 7: ACP 3D
amarillo: ROS - morado: no ROS

Es claro que por medio del ACP no se pudo hacer la discriminación de los dos grupos de interés. También Se realizo el gráfico, con tres componentes para observar desde otra perspectiva el comportamiento de los datos con el ACP.

4.3. t-SNE

Para un correcto funcionamiento de t-SNE y, evitar problemas con la memoria del sistema, se extrajo una muestra aleatoria de 30,000 datos. realizo por tercera vez un conteo de los no ROS y ROS:

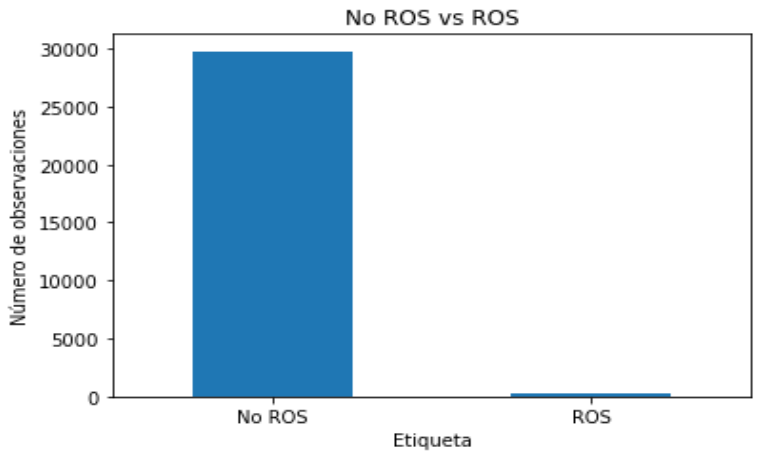


Figura 8: No ROS vs ROS

Etiqueta	Conteo
No ROS	29,777
ROS	223

Tabla 4: Conteo no ROS y ROS

Posteriormente se entrenó t-SNE bajo a aprendizaje no supervisado. Se seleccionaron dos componentes para el gráfico, 50 como el número de vecinos más cercanos y 1000 iteraciones máximas. Se realizo el grafico para la visualización de los datos:

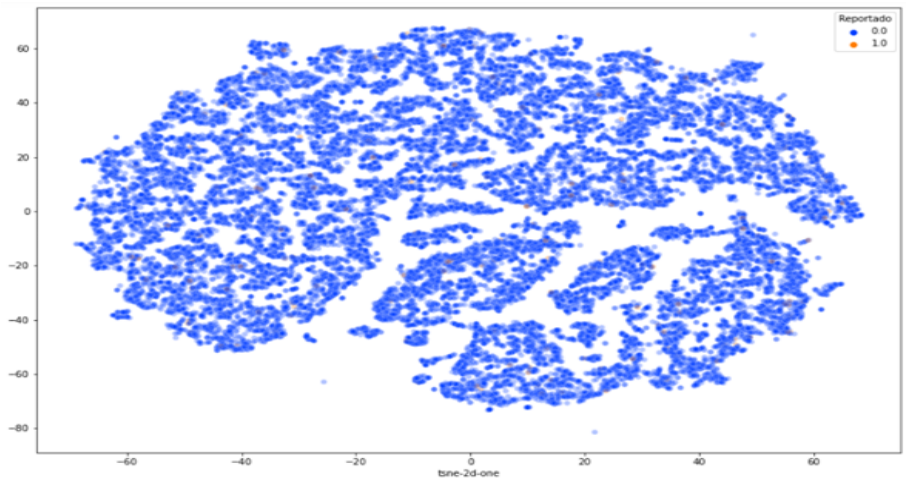


Figura 9: t-SNE
naranja: ROS - azul: no ROS

En este caso el algoritmo tampoco fue capaz de hacer una correcta clasificación de los grupos, de hecho, los puntos naranjas que representa los ROS son prácticamente imperceptibles en el gráfico.

4.4. ACP con t-SNE

En este caso se hizo la combinación de las técnicas anteriores para probar si se podían mejorar los resultados obtenidos anteriormente con cada uno de los algoritmos.

Primero se aplicó ACP y, se hayo el menor número de componentes que expliquen aproximadamente el 85 % de la variabilidad de los datos. Se seleccionaron 5 componentes principales las cuales explicaban el 85.73 % de la variabilidad de los datos.

Con las 5 componentes del ACP se entrenó t-SNE empleando los mismos valores que se usaron anteriormente para los argumentos del algoritmo y se realizó el gráfico:

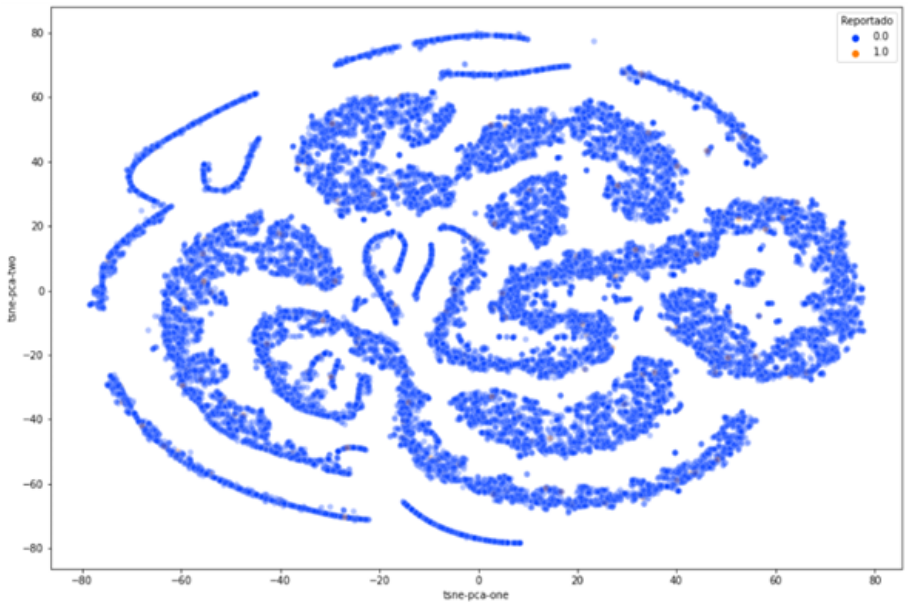


Figura 10: ACP con t-SNE
naranja: ROS - azul: no ROS

La combinación de las técnicas dio como resultado una mejor separación de los puntos en el gráfico, sin embargo, no hay una buena clasificación de los grupos de interés y la presencia de los puntos naranjas que representan los ROS sigue siendo imperceptible.

4.5. UMAP no supervisado

En este caso, tal como en t-SNE, para no tener inconvenientes al entrenar el algoritmo, se tomó la misma muestra de 30,000 datos.

Como en un inicio UMAP se entrenó con aprendizaje no supervisado, al algoritmo no se le especifico la variable etiqueta de los datos. Se obtuvo el siguiente resultado:

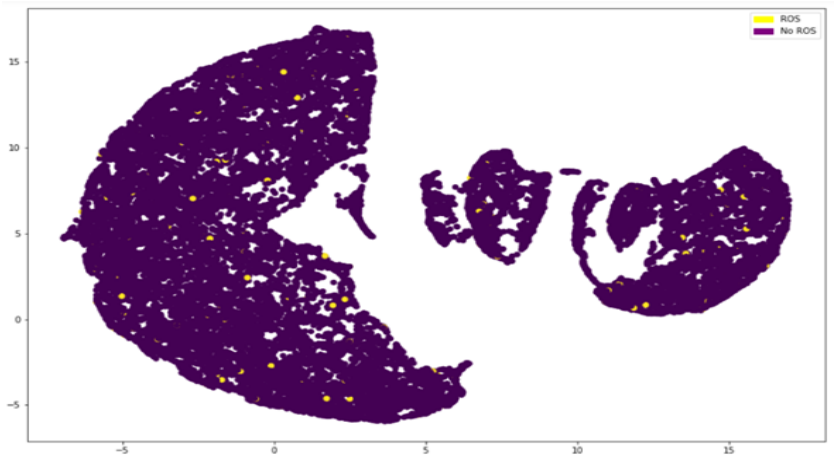


Figura 11: UMAP no supervisado
amarillo: ROS - azul: no ROS

UMAP bajo el aprendizaje no supervisado tampoco dio los resultados esperados, sin embargo, este algoritmo tuvo la capacidad de formar conglomerados mejor estructurados que las técnicas anteriores a pesar de no haber discriminado de forma correcta los grupos de interés.

4.6. UMAP supervisado

Nuevamente se empleó UMAP, pero en este caso se hizo bajo el aprendizaje supervisado y se le dio al algoritmo la etiqueta para que identificara a que grupo pertenecía cada uno de los registros. Además, se suplió el desbalance en la etiqueta de los datos. Se tomaron los $n = 5,249$ registros que quedaron con ROS después de la limpieza de los datos y, se tomó una muestra aleatoria n de la misma cantidad, pero de los no ROS. De este modo, se obtuvo una muestra de 10,498 datos con la etiqueta equilibrada. Para este caso, también se hizo una modificación en dos argumentos del algoritmo y, se seleccionan los 50 vecinos más cercanos y una distancia mínima de 0.5. Se entreno el algoritmo y, se realizó el gráfico:

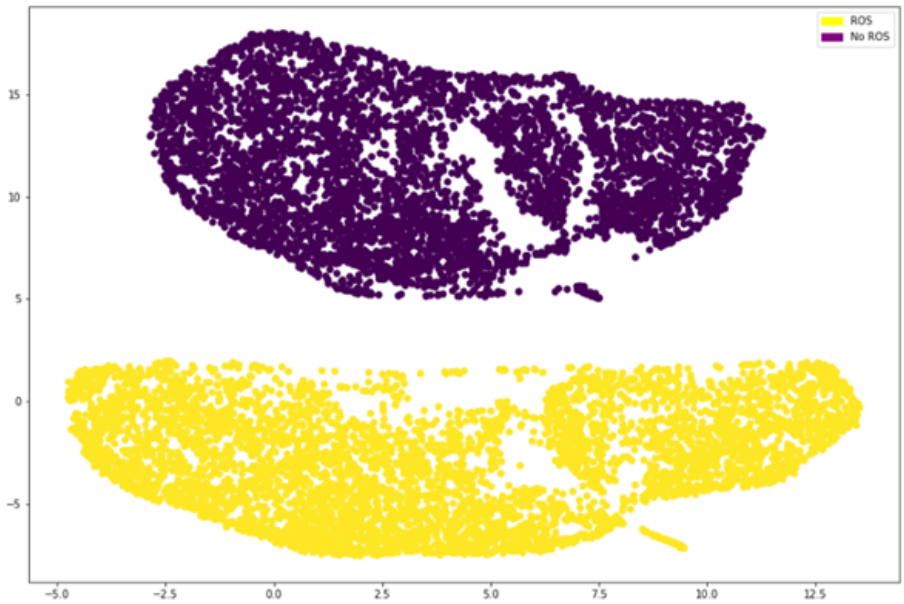


Figura 12: UMAP supervisado

En este caso, bajo el aprendizaje supervisado se logró hacer la clasificación de los grupos de interés en dos conglomerados claramente separados.

4.7. Prueba U de Mann-Whitney

Finalmente, se empleó la prueba no paramétrica U de Mann-Whitney para determinar las variables que tenían diferencias estadísticamente significativas y, en las cuales se podía estar basando el algoritmo para hacer la separación de ambos grupos. Antes de aplicar la prueba, se hizo uso de la prueba de Kolmogorov-Smirnov con la corrección de Lilliefors para verificar que los datos no tuvieran una distribución normal. Posteriormente, se plantearon las hipótesis que se deseaban comprobar mediante la prueba U de Mann-Whitney:

H_0: no hay diferencias en las medianas de la variable **X** de ambos grupos
H_1: hay diferencias en las medianas de la variable **X** de ambos grupos

donde **X** es cada una de las 27 variables que tiene la base de datos.

Las variables para las cuales se rechazó la hipótesis nula fueron las siguientes:

Variables	Valor-p
V1	0.1
V8	0.46
V21	0.16
V23	0.14
V25	0.31
V33	0.41

Tabla 5: variables descartadas

Por lo tanto, de las 27 variables que se tenían, 21 presentaban diferencias estadísticamente significativas. UMAP, podría estarse basando en esas 21 variables para hacer la discriminación entre ambos grupos, por lo tanto, estas variables pueden ser clave para poder identificar por medio de los estados financieros, clientes que deban tener reportes de operaciones sospechosas.

5. Conclusiones y recomendaciones

- UMAP fue el algoritmo que logro hacer una mejor clasificación de los datos (especialmente bajo el aprendizaje supervisado), pues las otras técnicas no eran las más adecuadas para el conjunto de datos.
- El hecho de que la discriminación de los subgrupos solamente haya podido realizarse bajo aprendizaje supervisado, podría sugerir, que, aunque con la variable etiqueta, el algoritmo si puede aprender de los datos, quizá esas características, variables o atributos no sean tan relevantes o influyentes, pues el aprendizaje no supervisado es más potente.
- Hay 21 variables con diferencias estadísticamente significativas en las cuales el algoritmo se puede estar basando para hacer la clasificación de ambos grupos, lo que sugiere que son estas variables las características o atributos más influyentes en los estados financieros de los clientes con ROS y lo no ROS. De esta forma se pueden empezar a tener en cuenta los estados financieros de los clientes, para monitoreos, y así poder identificar clientes que puedan requerir ROS por posibles operaciones sospechosas.
- Las 21 variables con diferencias estadísticamente significativas están relacionadas con endeudamiento y ventas.
- El proyecto fue presentado a contadores públicos de la gerencia los cuales estuvieron de acuerdo con los resultados obtenidos, ya que por experiencia en el campo intuían un resultado similar.
- Se recomienda realizar el mismo proceso de clasificación empleando las mismas técnicas, pero haciendo una segregación de cada uno de los segmentos para obtener mejores resultados.

Referencias

[1] U. de Barcelona. Pruebas para dos muestras independientes. http://www.ub.edu/aplica_infor/spss/cap6-2.htm.

[2] U. de Información y Análisis Financiero. ¿qué es un ros? https://www.uiaf.gov.co/sistema_nacional_ala_cft/que_es_un_ros.

[3] S. de la Fuente Hernández. Análisis de componentes principales. https://www.estadistica.net/Master-Econometria/Componentes_Principales.pdf.

[4] I. escuela de negocios. Estados financieros, ¿qué son y por qué son tan importantes para tu empresa? https://www.ienupm.com/pdd/estados-financieros-que-son/.

[5] F. L. J. García. U de mann-whitney. https://www.rincondepaco.com.mx%2Frincon%2FInicio%2FApuntes%2FProyecto%2Farchivos%2FDocumentos%2FU_Mann.pdfchunk=true.

[6] InteractiveChaos. t-sne. https://interactivechaos.com/es/manual/tutorial-de-machine-learning/t-sne.

[7] S. Learn. sklearn.manifold.tsne. https://scikit-learn.org/stable/generated/sklearn.manifold.TSNE.

[8] L. McInnes, J. Healy, and J. Melville. How umap works. https://umap-learn.readthedocs.io/en/latest/how_umap_works.html.

[9] L. McInnes, J. Healy, and J. Melville. Umap: Uniform manifold approximation and projection for dimension reduction. 2020.

[10] D. Peña. *Análisis de datos multivariante*. McGrawHillEducation, 1 edition, 2002.

[11] R. R. Ruiz, J. M. Palacios, and J. Talaveraa. Investigación clínica xvi diferencias de medianas con la u de mann-whitney. *Revista Médica del Instituto Mexicano del Seguro Social*, 51(4), 2013.

[12] sitiobigdata.com. Algoritmo t-sne con python, una breve introducción. https://sitiobigdata.com/2018/08/27/algoritmo-t-sne-con-python/.

[13] Wikipedia. Prueba u de mann-whitney. https://es.wikipedia.org/wiki/Prueba_U_de_Mann-Whitney.

[14] H. Zhou, F. Wang, and P. Tao. t-distributed stochastic neighbor embedding (t-sne) method with the least information loss for macromolecular simulations. *NCBI*, 14(11):5499–5510, 2018.