Evaluación de modelos estadísticos para precisar si un paciente está medicado basado en los datos recolectados en un monitoreo ambulatorio de presión arterial.

Sergio Bermúdez Gómez
Leidy Marcela Castañeda Bedoya
Luis Alberto Escobar Robledo
Yullys María Quintero Martínez
Ana Cristina Urcuqui Henao

Proyecto integrador primer semestre
Maestría en Ciencia de los datos
Universidad EAFIT

Medellín, 2020

Introducción

La prevalencia mundial de hipertensión fue de 1.13 mil millones en 2015, con una prevalencia general en adultos de alrededor de 30 - 45 %. La hipertensión generalmente se clasifica por valores de presión arterial, o por características demográficas subyacentes y factores de riesgo. Aunque la monitorización ambulatoria de la presión arterial las 24 horas (MAPA 24 h) se ha convertido en una herramienta poderosa para diferenciar algunas formas de elevación de la presión arterial (es decir, bata blanca, enmascarada e hipertensión nocturna) ahora se han empezado a aplicar perfiles hemodinámicos específicos de hipertensión.

Medir la presión arterial es una rutina desarrollada por los médicos en toda consulta porque es una medida que puede indicar una enfermedad rápidamente o un simple estado de excitación, para confirmar esto se necesita asociar la medida de la presión arterial a otras variables y estilos de vida.

Para aplicar los conocimientos adquiridos durante el semestre en cada curso utilizaremos una base de datos de cerca de 20 mil pacientes a los cuales se les realiza un MAPA con perfiles hemodinámicos.

Objetivo general

Profundizar a través de la práctica los conceptos y las metodologías desarrolladas durante el primer semestre de la maestría en ciencia de datos y analítica mediante la creación y evaluación de modelos estadísticos para precisar si un paciente está medicado basado en los datos recolectados en un MAPA

Justificación

Si una persona que sufre de presión arterial se deja sin tratamiento, esto le puede provocar afecciones a largo plazo. Estas incluyen enfermedades del corazón, accidente cerebrovascular, insuficiencia renal, entre otros.

Precisar si un paciente toma medicamentos o no basados solamente en los datos del MAPA puede ayudar a mejorar la precisión a la hora del tratamiento, identificar errores en la toma o administración del tratamiento.
Además esto abre la puerta a realizar modelos prescriptivos para identificar el tratamiento necesario para cada tipo de paciente.

Github:

Este es el link del github del proyecto

https://github.com/Proyecto-Integrador-SYALL/modelos_estadisticos_MAPA_medicamentos

Metodologías usadas

El notebook “0 get_data_from_api.ipynb” tiene el código encargado de recolectar la información desde la url. Dado que hay información protegida por la relación médico paciente, no se puede compartir la URL ni los códigos de accesos.

Utilizando request para obtener los datos del API y ray para paralelizar, este código recolecta los datos de los pacientes y los guarda en la carpeta “api-data” (la cual simula un bucket de un datalake). La información de los pacientes son guardados en un formato json, con información semiestructurada.

El notebook “1 build_base_dataset.ipynb” se encarga de cargar todos los pacientes que se encuentren en “api-data” y generar datasets. Aquí hay un proceso muy importante de ETL que es la función “hypertension_medicine” esta función toma la información de los medicamentos de los pacientes que al ser digitada manualmente por diferentes enfermeras, no es consistente en identificar los medicamentos que se usan para la presión arterial. Además aquí se recolectan todos los medicamentos que toma el paciente, sin importar su finalidad. Esta función permite obtener un valor booleano que indica si el paciente está o no bajo tratamiento para la presión arterial. Este notebook también carga el resto de los datos que consideramos necesarios para el análisis de los pacientes. Al final el dataset se guarda como base_dataset.csv

El notebook “2 build_working_dataset.ipynb” se encarga de tomar el dataset de base y realizar más procesos de ETL. el principal es agrupar las variables para que se tenga un solo dato por paciente pues en la carga original se toma más de un dato de presión arterial debido a que se cargan todos los registros del MAPA de 24h. Aquí se promedian las variables que se tienen de cada registro, y las invariantes se toma el máximo. Al final, la base de datos queda con las siguientes variables.

- número identificador del paciente
- Fecha de nacimiento
- fecha de inicio de la prueba
- fecha de inicio de la noche (hora a la que el paciente se va a dormir)
- fecha de fin de la noche (hora a la que el paciente se despierta)
- genero (m = masculino, f = femenino)
- Talla (en metros)
- Peso (en Kg)
- Fuma (0 = no, 1 = si)
- frecuencia de ejercicio (1 = sedentario, 2 = menos de 70 minutos a la semana de actividad leve, 3 = entre 70 y 210 minutos a la semana, 4 = entre 210 y 500 minutos a la semana, 5 = más de 500 minutos a la semana)
- consumo de cerveza (0 = no, 1 = si)
- consumo de vino (0 = no, 1 = si)
- consumo de licor fuerte (0 = no, 1 = si)
- consumo de cualquier tipo de licor (0 = no, 1 = si)
- medicamentos (lista que contiene los medicamentos que consume el paciente)
- fecha de la medición
- presion sistólica (en mmHg)
- presion diastólica (en mmHg)
- frecuencia cardiaca (en latidos/min)
- resistencia vascular sistémica (en dyn·s/cm−5)
- presión de pulso (en mmHg)
- velocidad de onda de pulso (en m/s)
- gasto cardiaco (en L/min)
- indice cardiaco (en L/min/m2)
- volumen latido (en mL)
- presion arterial media (en mmHg)

Como estos datos tienen información de los pacientes no se pueden compartir libremente. Para desarrollar el proyecto integrador, se generó una base de datos con la misma matriz de covarianza de la original, pero con datos aleatorizados bajo una distribución gaussiana. Este permite el desarrollo de códigos por parte de los integrantes del grupo que después serán usados en la base de datos final. Para realizar esto, se tiene un código que sigue estos pasos.

1. filtrar por todas las posibles combinaciones de las variables cualitativas
2. estandarizar este subgrupo
3. calcular la matriz de covarianza de este subgrupo
4. generar un array aleatorio de la misma cantidad de datos que el subgrupo
5. transformar inversamente la estandarización
6. repetir los pasos 2 al 5 por cada subgrupo

Al final se graba como random_dataset.csv (este archivo se comparte en el github)

El notebook “3 mahalproyecto.ipynb” tiene como finalidad realizar la limpieza y depuración de los datos, detectando los valores atípicos mediante la distancia de Mahalanobis robusta, debido a que la matriz de covarianzas es una matriz muy mal condicionada. De esta matriz se graba la base de datos filtrada donde se eliminan los outliers. Esta se graba como filtered_dataset.csv

El notebook “4 Normas_deter.ipynb” es el código que permite encontrar después de eliminar los valores atípicos, la matriz de correlaciones de las variables, la cual permite identificar las variables que se encuentran altamente correlacionadas y de esta manera identificar las variables que son causantes de multicolinealidad, estas variables son eliminadas del estudio. Se puede encontrar también el cálculo de la matriz de covarianzas de las variables que no presentan altas correlaciones, permitiendo encontrar las varianzas total y generalizadas de los datos.

El notebook “5 Número de Condición.ipynb” calcula el número de condición al dataset de datos final (dataset fltrado de outliers y variables dependientes).

En los notebooks “6.0 Descomposición en Valores Singulares.ipynb” y “6.1 PCA” se realiza la descomposición en valores singulares, la descomposición SVD y el análisis de componentes principales (PCA). Cuando se trabaja en problemas de Machine Learning, muchas veces se encuentran conjuntos de datos grandes que contienen además muchas características o features. Una forma simple de reducir las dimensiones de estas características es aplicar alguna técnica de factorización de matrices.

En el notebook “7.0 Análisis Descriptivo.ipynb” se puede encontrar un análisis descriptivo de cada una de las variables categóricas y métricas consideradas en el estudio. Se pueden observar, las formas y distribuciones de cada una de ellas.

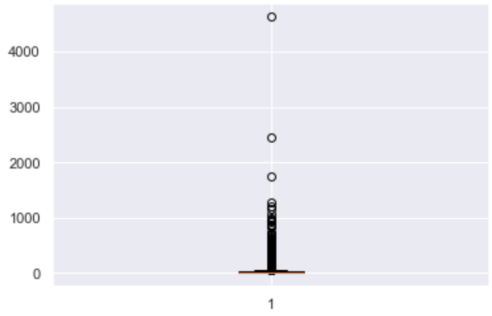
En los notebooks “8.0 KMeans”, “8.1 DBSCAN” y “8.2 Aglomerative” se realizaron modelos de aprendizaje no supervisado para identificar clusters dentro de los datos. Estos se procesaron con múltiples algoritmos de agrupación, KMeans, agrupación espacial basada en densidad de aplicaciones con ruido (DBSCAN), agrupación jerárquica, Para cada uno de estos métodos, se realizó una configuración de hiperparámetros. Para encontrar un número adecuado de grupos en KMeans se utilizó el “método del codo” para encontrar el número óptimo sugerido de clusters.

En los notebooks “9.0 Bosques Aleatorios.ipynb”, “9.1 Naive Bayes.ipynb”, “9.2 Regresion Lineal.ipynb”, “Regresion Logistica.ipynb” y “GLM.ipynb” se pueden encontrar algoritmos de modelos de aprendizajes supervisados con el objetivo de clasificar a los pacientes en pacientes medicados o no medicados. Tambiés se hizo un ajuste de un modelo de regresión con variable respuesta métrica para determinar si existe alguna relación que se considere importante entre ellas.

Resultados

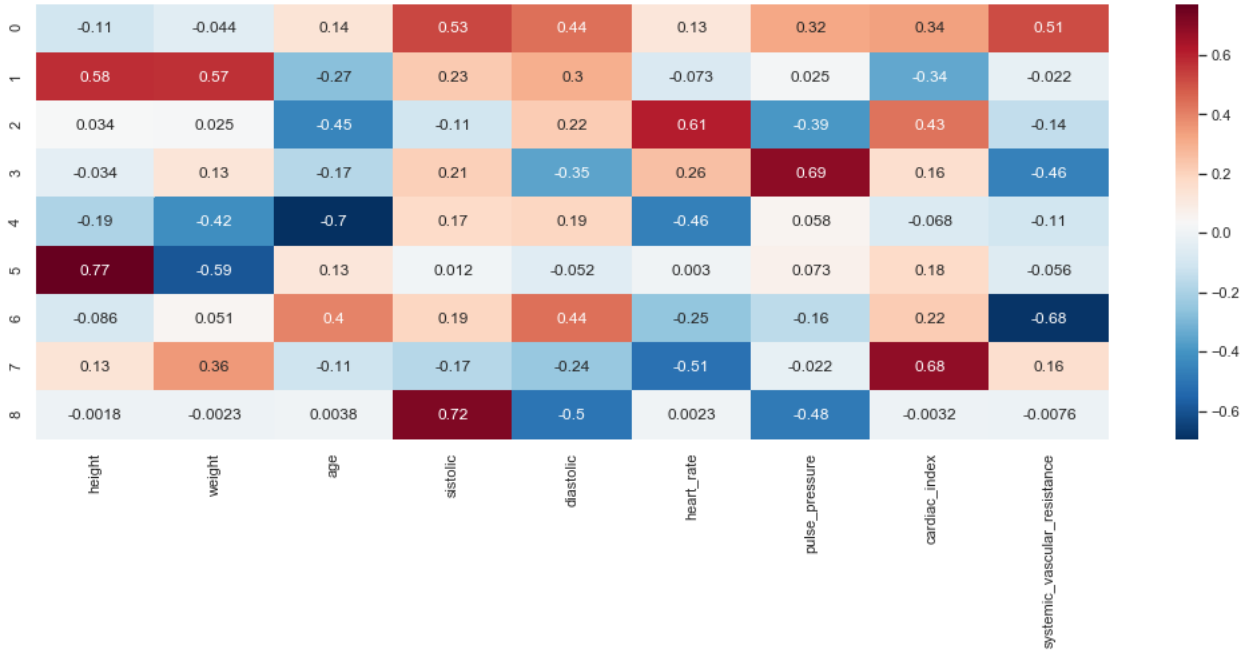
Realizando la distancia de Mahalanobis :

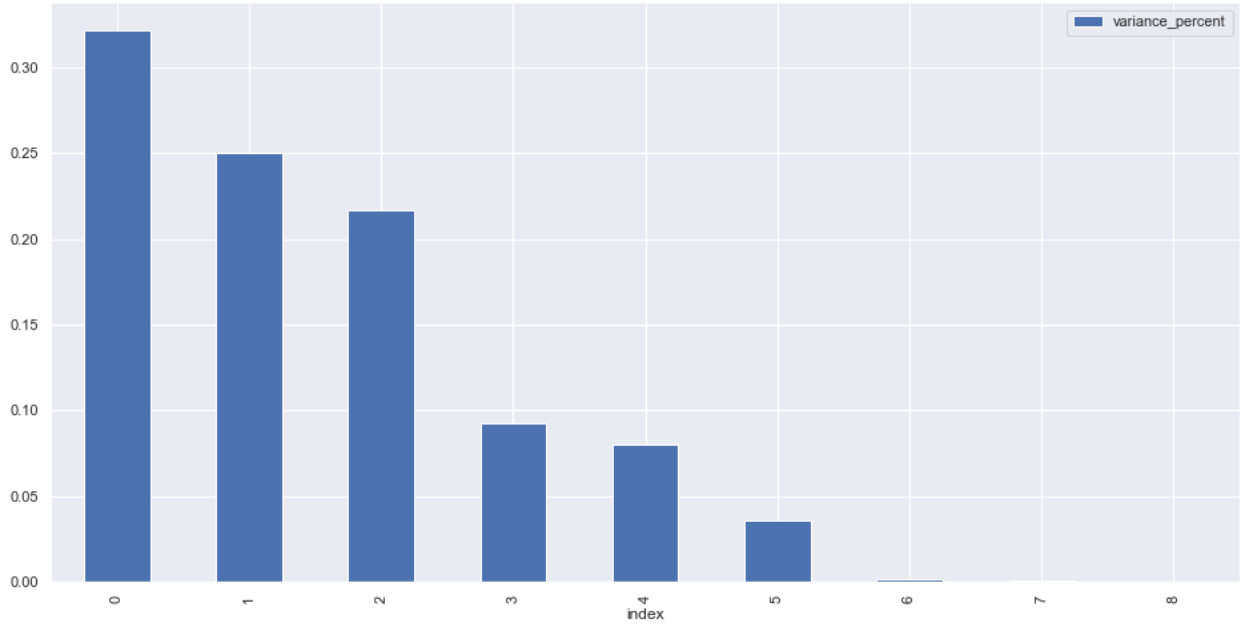
Para identificar outliers se observa en la gráfica a la izquierda el boxplot de la base de datos -sin extraer los outliers- que el valor máximo de la distancia de mahalanobis es 4300 y al retirarlos -la gráfica de la derecha - ese valor desciende a 60. La base de datos ahora se llama Filtered_dataset.csv y se va a trabajar con esta para los siguientes pasos de MAPA.

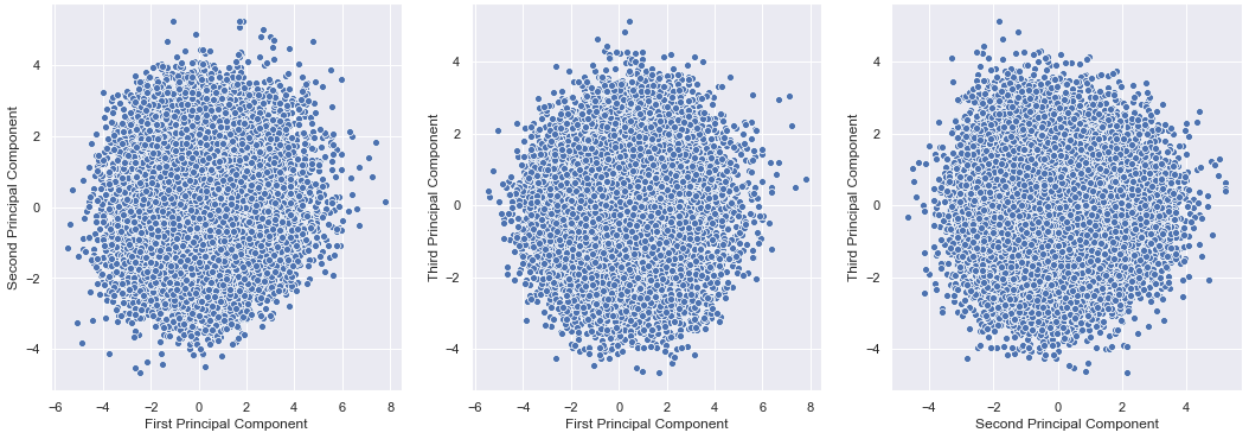


Resultados de la PCA:

Aquí se muestra un heatmap de cada componente principal con las variables utilizadas. Un gráfico de barras que explica el porcentaje de variación que explica cada componente y finalmente un scatterplot de los tres primeros componentes que explican el 79% de la variación de los datos

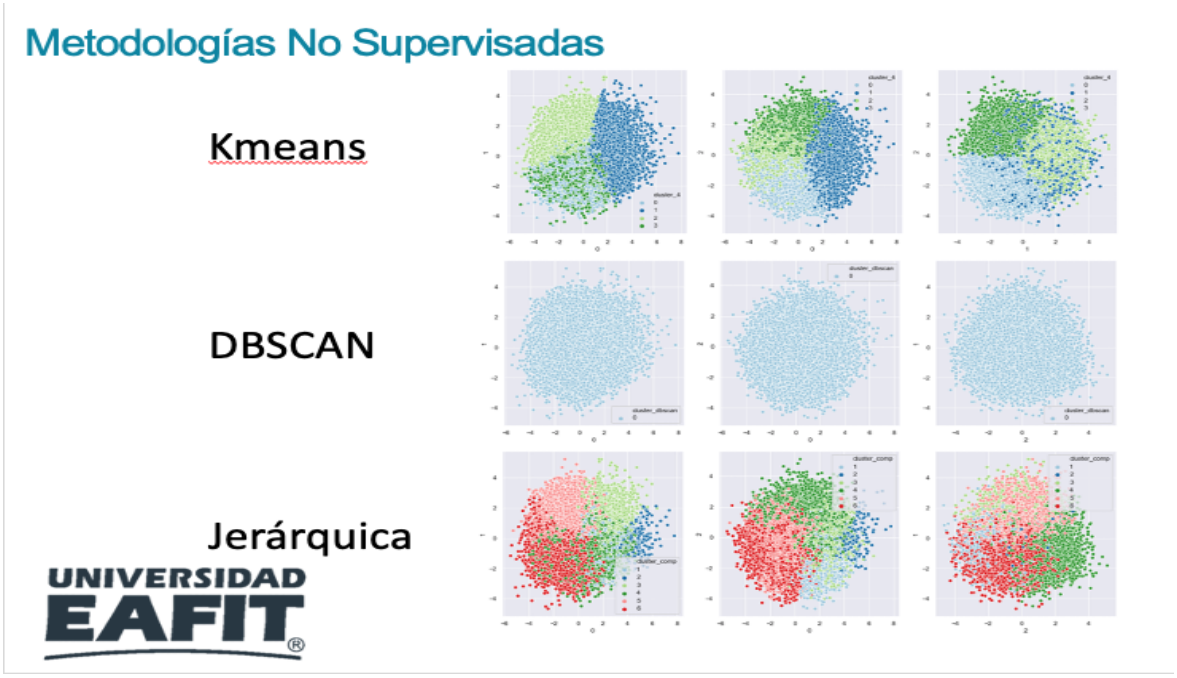


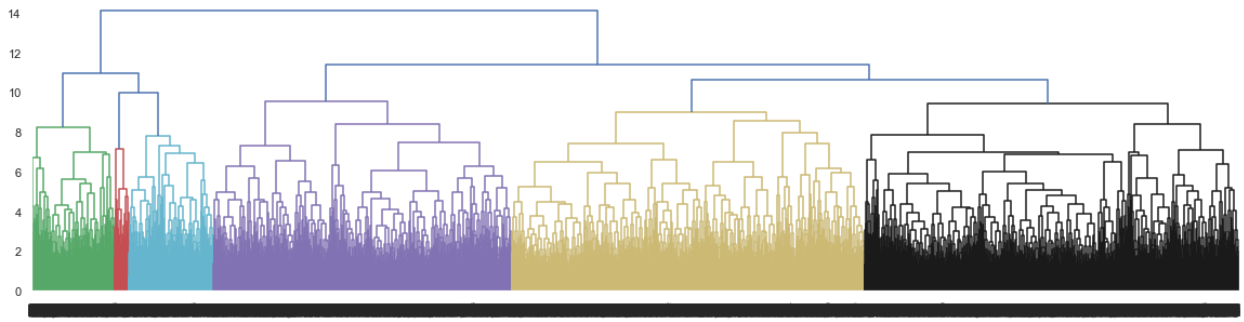




Resultados de los métodos no supervisados:

Aquí se evidencia que los datos no tienen clusters discretos definidos.





Resultados algoritmo supervisado

Para la construcción de los modelos supervisados con el fin de entrenarlos y evaluar si están funcionando bien, se realiza una separación del conjunto de datos inicial en 2: donde se utiliza un conjunto para el entrenamiento del modelo y otro conjunto para realizar pruebas sobre el modelo. Para este caso se divide en 80-20 tomando muestras aleatorias, en donde el 80% del conjunto de datos se utilizará para el entrenamiento y el 20% para el testeo.

En este caso de los 17.644 pacientes, 14.115 nos servirán para entrenar el modelo y 3.529 para realizar las validaciones.

Naive Bayes

- Resultado sobre set entrenamiento

  Se observa de los resultados del modelo sobre el set de entrenamiento y vemos que tiene una precisión del 63%, está prediciendo al 64% de los pacientes que no tienen un tratamiento correctamente y al 63% de los pacientes que si tiene un tratamiento correctamente.

- Resultado sobre set Validación

  Se observa los resultados del modelo sobre el set de validación y vemos que tiene una precisión del 63%, está prediciendo al 63% de los pacientes que no tienen un tratamiento correctamente y al 63% de los pacientes que si tiene un tratamiento correctamente.

En conclusión, se puede observar que es un modelo bastante estable ya que los resultados obtenidos entre el set de entrenamiento y set de validación son muy congruentes.

Regresión Logística

- Resultado sobre set entrenamiento

  Se observa que para el conjunto de entrenamiento la precisión del modelo es de 65%, lo que indica que el modelo clasificará bien los pacientes el 65% de las veces. Al analizar la matriz de confusión:

  - Se observa que para los pacientes sin tratamiento la tasa de buena clasificación es de 70% y para los pacientes con tratamiento es de 59%.
  - También se observa que para los pacientes sin tratamiento la tasa de mala clasificación es de 30% y para los pacientes con tratamiento es de 41%.

- Resultado sobre set Validación

Se observa que para el conjunto de prueba la precisión del modelo es de 63%, lo que indica que el modelo clasificará bien los pacientes el 63% de las veces.
Al analizar la matriz de confusión:

- Se observa que para los pacientes sin tratamiento la tasa de buena clasificación es de 69% y para los pacientes con tratamiento es de 56%.
- También se observa que para los pacientes sin tratamiento la tasa de mala clasificación es de 31% y para los pacientes con tratamiento es de 44%.

En conclusión, se puede observar que es un modelo bastante estable ya que los resultados obtenidos entre el set de entrenamiento y set de validación son muy congruentes.

Random Forest

- Resultado sobre set entrenamiento

Se observa que para el conjunto de entrenamiento la precisión es sospechosamente buena (100%), lo que indica que el modelo clasificará bien los pacientes el 100% de las veces.
Al analizar la matriz de confusión:

- Se observa que, tanto para los pacientes sin tratamiento como los pacientes con tratamiento, la tasa de buena clasificación es de 100%.
- También se observa que para ambas clases la tasa de mala clasificación es de 0%.

- Resultado sobre set Validación

Se observa que para el conjunto de prueba la precisión del modelo es 62%, lo que indica que el modelo clasificará bien los pacientes el 62% de las veces.
Al analizar la matriz de confusión:

- Se observa que para los pacientes sin tratamiento la tasa de buena clasificación es de 72% y para los pacientes con tratamiento es de 52%.
- También se observa que para los pacientes sin tratamiento la tasa de mala clasificación es de 28% y para los pacientes con tratamiento es de 48%.

En conclusión, se puede observar que es un modelo bastante sospechoso, ya que los resultados obtenidos entre el set de entrenamiento son perfectos y en el set de validación estos se caen, lo que indica que no es un modelo que fuera de muestra vaya a tener un buen comportamiento.

Metodologías Supervisadas

Conclusiones

La Regresión Logística y Naive Bayes son los modelos que mejor ajustan los datos disponibles en el MAPA del paciente para determinar si el paciente ingiere o no un medicamento, pues la precisión de ambos es buena tanto para el conjunto de entrenamiento como para el conjunto de prueba.

Sin embargo el modelo ganador será Naive Bayes ya que la tasa de buena clasificación es similar en ambos grupos (Medicamento y No Medicamento), mientras que la Regresión Logística favorece la clasificación de los pacientes que no toman medicamentos.

Se requiere de un proceso con mayor rigor científico para validar estos hallazgos.