Análisis de hábitos saludables de las personas mediante Algoritmos de Aprendizaje Supervisado y No Supervisado

Leonardo Garcia Muñoz

16 de noviembre de 2025

1. Introducción

Los habitos con los que cuentan las personas en los tiempos actuales son consideradas una de las principales razones de las enfermedades que actualmente presentan una alta incidencia, siendo estos factores mucho mas relevantes en un monitoreo medico que incluso la predisposicion genetica de las personas. Existiendo distintas maneras de clasificar los distintos habitos con los que cuentan las personas.

El siguiente artículo tiene como principal proposito el aplicar tecnicas de **clustering** o agrupamiento para explorar distintas agrupaciones de los habitos de las personas y las posibles relaciones con los riesgos que tienen las personas de sufrir alguna enfermedad de gravedad. Por lo cual se estaran utilizando principalmente los algoritmos **Optics** y **K-Means** para realizar las agrupaciones.

2. Descripción de los datos

El conjunto de datos se compone principalmente por variables númericas:

- age,bmi,daily steps, sleep hours, water intake, calories consumed, resting hr, systolic bp, diastolic bp, cholesterol

Pero tambien presenta las siguientes variables categoricas:

- gender, smoker, alcohol, family history

Todas estas variables fueron registradas por persona, por lo cual cada registro de estas variables es una persona con distintos habitos, condiciones de vida e historias diferentes. Pero esta informacion para poder ser utilizada en los algoritmos a aplicar, se estandarizaron para evitar problemas de escalabilidad entre variables y presentarlas todas en las mismas dimensiones.

3. Metodología

3.1. Algoritmo *K-Means*

Este algoritmo de **clustering** o agrupamiento agrupa los datos en k grupos o **clusters** basandose en la distancia al punto promedio o **centroide** que representa el centro de un grupo, y a cada dato se le asigna el **centroide** mas cercano. Para despues recalcular los **centroide** de manera iterativa. Buscando que los datos que componen cada grupo sean similares.

Formalmente, la distancia más cercana respecto a los centroides se define como la minimizacion del error cuadrado para cada grupo.

$$SS_k = \sum_{i \in K} (x_i - \hat{x}_k)^2 + (y_i - \hat{y}_k)^2.$$

Teniendo como función objetivo

$$\text{mín} \sum_{i \in K} SS_k.$$

Donde $\hat{c}_k$ es un centroide compuesto por $(\hat{x}, \hat{y})$

3.2. Algoritmo *Optics*

El algoritmo *Ordering Points To Identify the Clustering Structure* o *OPTICS* esta en la identificacion de regiones con una alta densidad de datos. Cada dato o punto tiene una densidad que se calcula como el valor maximo entre la distancia al dato vecino mas cercano y un parámetro de densidad minimo. El algoritmo recorre los datos, expandiendo los clusters desde los puntos más densos y ordenandolos de manera que refleja la estructura de densidades.

Formalmente se define de la siguiente manera

$$reachability\_distance(p, o) = max(core\_distance(o), dist(p, o))$$

Donde $core\_distance(o)$ es el parametro de densidad minimo.

3.3. Metrica *Inercia*

Esta métrica que se puede utilizar en *K-Means* mide que tan compactos están los clusters, calculando la suma de las distancias al cuadrado de cada punto a su centroide. Para elegir el número óptimo de clusters, se prueba K-Means con varios k y se grafica la inercia contra k. Mediante el método del codo se busca el punto de quiebre, en donde la disminución de la inercia se vuelve menos pronunciada en escala de valores.

Formalmente para k clusters o grupos

$$Inercia = \sum_{i=1}^{k} \sum_{x_j \in C_i} \|x_j - \mu_i\|^2$$

Donde C_i son los puntos que componen el grupo o cluster i y μ_i su centroide.

3.4. Metrica *Calinski-Harabasz*

El índice *Calinski-Harabasz* es una métrica para evaluar clusters o grupos que compara la separación entre grupos con que tan cerca estan los puntos entre si, dentro de cada grupo. Se usa para elegir el número óptimo de clusters k. Calculando CH para varios k, el valor más alto indica la mejor separación y cohesión.

Formalmente para k clusters o grupos.

$$CH = \frac{B_k/(k-1)}{W_k/(n-k)}$$

Donde B_k es la varianza entre clusters, W_k la varianza dentro de los clusters, n el número de puntos y k el numero de clusters o grupos. Un CH alto indica grupos altos y bien separados de otros.

3.5. Modelo de *Regresión Lineal*

El modelo de Regresión Lineal busca describir la relación entre una variable dependiente y y un conjunto de variables independientes $x_1, x_2, ..., x_n$. Su objetivo es encontrar los coeficientes de β_i que minimicen el error cuadrático entre los valores observados y los valores predichos.

Formalmente, el modelo se define de la siguiente manera.

$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n$$

3.6. Modelo *Lasso* (Least Absolute Shrinkage and Selection Operator

El modelo de Lasso es una extensión de la regresión lineal que introduce una penalización sobre el tamaño de los coeficientes del modelo para controlar la complejidad y evitar un sobreajuste.

Formalmente, la funcion de penalizacion que utiliza se define de la siguiente manera.

$$\min_{\beta} \left(\sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right) \tag{1}$$

Donde y_i es el valor observado de la variable dependiente (y), $\hat{y}_i$ es el valor estimado por el modelo, β_j es el coeficiente que toma el modelo y λ es el parámetro de penalización.

3.7. Algoritmo *Random Forest*

El algoritmo Random Forest es un método de aprendizaje supervisado que combina múltiples árboles de decisión para mejorar la precisión del modelo y reducir asi el sobreajuste. Cada uno de los árboles que componen el algoritmo se entrena con muestras aleatorias de datos y de las variables predictoras (x) generando asi una serie de modelos diferentes (árboles de decisiones diferentes entre si).

Formalmente, si existen n árboles individuales y cada uno produce una predicción $\hat{y}_t$ el algoritmo se define de la siguiente manera.

$$\hat{y} = \frac{1}{n} \sum_{t=1}^{n} \hat{y}_t \tag{2}$$

Donde $\hat{y}$ es el valor pronosticado del modelo Random Forest, n es el número total de árboles de decision que componen el bosque, $\hat{y}_t$ es la predicción realizada por el árbol t-ésimo.

3.8. Métrica *MAE* (Mean Absolute Error)

El Error Absoluto Medio (MAE) es una métrica utilizada para evaluar el desempeño de los modelos de Regresión. Mide la diferencia promedio entre los valores reales y los valores predichos por el modelo. Esta métrica es la que se utilizara para medir el desempeño de los modelos a utilizar.

Formalmente, se define como:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{3}$$

Donde n es el número total de observaciones, y_i representa el valor real observado, $\hat{y}_i$ es el valor estimado por el modelo a evaluar.

4. Diseño del Experimento

Para evaluar el desempeño de los modelos de regresión empleados, se diseño un procedimiento experimental que asegura que este mismo sea replicable ó reproducible. Ademas de contar con una comparación objetiva entre algoritmos.

Utilizando el siguiente flujo experimental:

1. Estandarización de las variables
2. División de la informacion en subconjuntos: 70 % para entrenamiento y 30 % para prueba.
3. Entrenamiento del modelo con el subconjunto de entrenamiento
4. Evaluación del modelo con los datos de prueba
5. Comparación de la métrica de error entre modelos

De tal manera que se garantiza una estimación clara del error del modelo para poder comparar el desempeño del mismo con otros, bajo las mismas condiciones experimentales.

5. Resultados

Primero realizando la tecnica de *k-Means* y utilizando la metrica *Inercia* obtenemos los siguientes resultados:

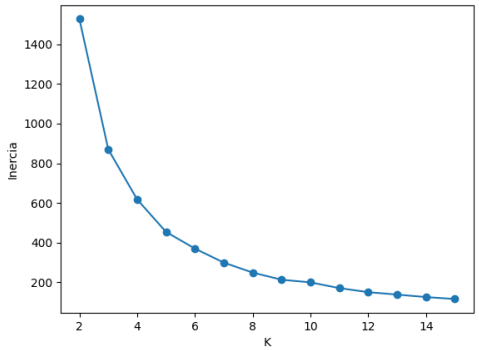

Figura 1. Gráfico generado en Python mediante la metrica *Inercia*.

Para obtener el numero de k grupos a utilizar primero debemos aplicar el Método del codo.

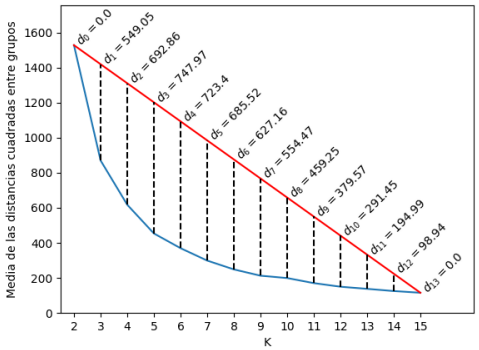

Figura 2. Gráfico generado en Python para aplicar el Método del codo.

Se aprecia como el punto de quiebre ocurre en $k = 5$ debido a que apartir de ahi, la disminucion ya no es tan pronunciada.

Por lo tanto se utiliza $k = 5$ para *K-Means*.

Ahora, utilizando la metrica *Calinski-Harabasz*

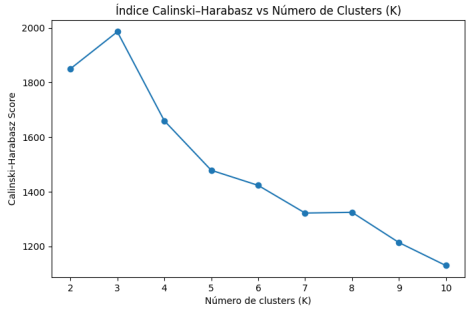

Figura 3. Gráfico generado en Python mediante la metrica *Calinski-Harabasz*.

Obtenemos que el CH mas alto pertenece a $k = 3$, por lo que bajo esta metrica tendrian que utilizarse 3 clusters.

Pero tambien aplicaremos al algoritmo *OPTICS* el cual obtiene los clusters necesarios

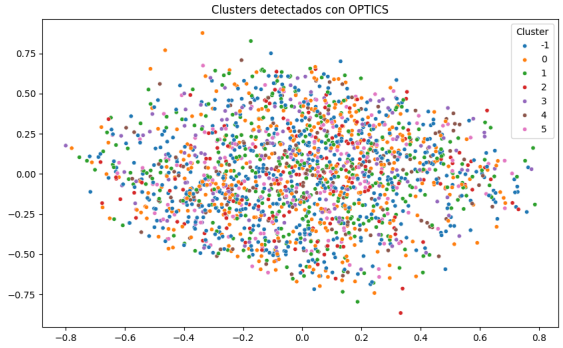

Figura 4. Gráfico generado en Python mediante *OPTICS*.

Calibrando un modelo de regresion lineal para realizar predicciones sobre la variable de *Presión Arterial diastólica* obtenemos los siguientes resultados reflejados en el grafico.

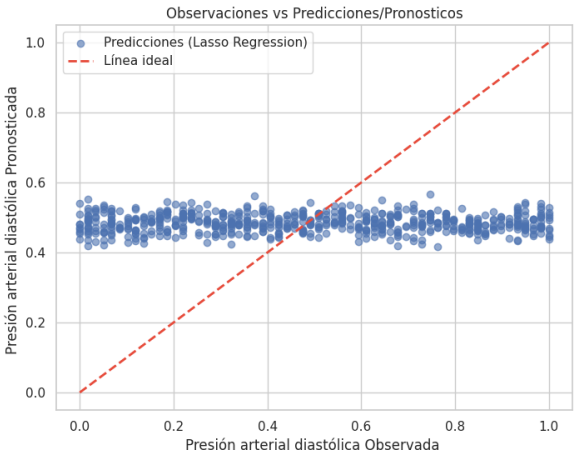

Figura 5. Gráfico generado en Python con base a una Regresion Lineal.

Se observa como los valores pronosticados oscilan entre el 0.4 y el 0.6 mientras que los valores observados se encuentran en un intervalo mucho mas amplio de valores, lo cual nos podria hablar de un posible sobreajuste en el modelo. Ademas de que muy pocos valores realmente se ajustan a la *Línea ideal*

En la siguiente figura se estan comparando distintos algoritmos supervisados para evaluar su desempeño y elegir el mejor modelo a utilizar, adicionalmente del modelo de regresion lineal previamente calibrado.

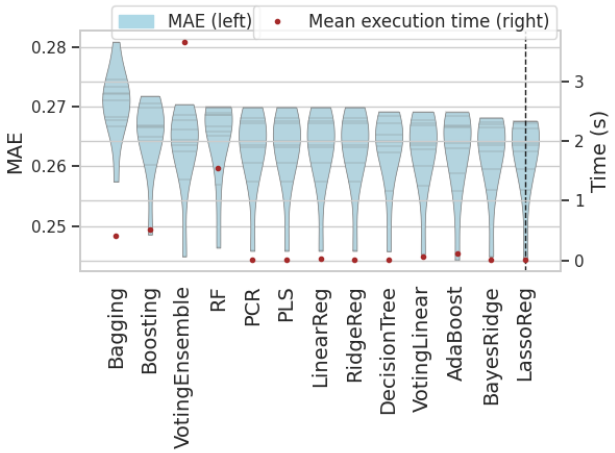

Figura 6. Gráfico de violín generado en Python

El modelo con menor error (MAE) y el tiempo de ejecucion mas optimo es el modelo Lasso, por lo que se calibra un modelo Lasso con el cual se pueda predecir la *Presión Arterial diastólica*.

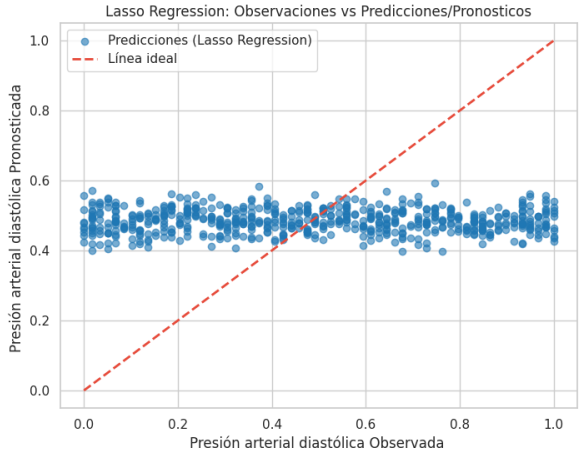

Figura 7. Gráfico generado en Python en base a un modelo Lasso.

En este modelo obtenemos una mayor variacion en las predicciones que realiza el modelo, pero sigue existiendo una concentracion de los valores pronosticados justo en el rango de **(0.38 , 0.6]**. Ademas de contar con una menor cantidad de valores que sigan la *Línea ideal*.

Ahora, aplicando y calibrando el algoritmo de aprendizaje supervisado *Random Forest* obtenemos los siguientes resultados.

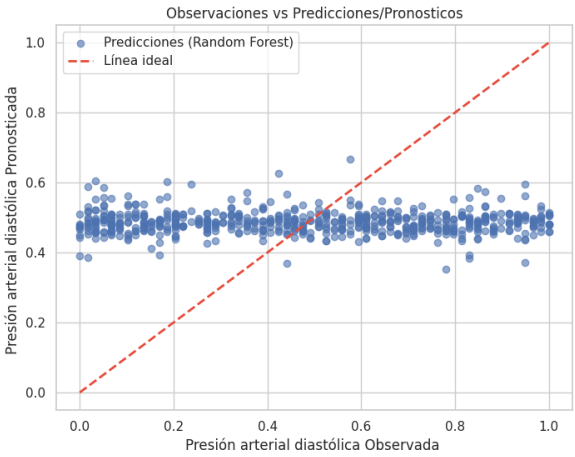

Figura 8. Gráfico generado en Python en base a un Random Forest.

Comparandolo con los otros dos modelos previamente calibrados, observamos como este es el que presenta la mayor variabilidad en las predicciones que realiza, pues ahora estas cubren un rango mayor de valores. Tomando los valores **(0.35 , 0.66)**. Resaltando que aun y con mayor variacion, existen menos puntos que se ajusten a la *Línea ideal*.

Con todos los modelos calibrados y utilizados para pronosticar, podemos realizar la comparacion de la metrica para medir los errores de los datos.

Modelo	MAE
Regresion Lineal	**0.2646**
Regresion Lasso	**0.2605**
Random Forest	**0.2666**

Cuadro1. Comparación de los modelos según la métricas de error.

Para ambos modelos se dividio el conjunto de datos de entrenamiento y prueba en una proporcion del 70 % y del 30 % respectivamente.

6. Conclusiones y discusiones

6.1. Algoritmos No Supervisados para clustering

Principalmente podemos concluir que el algoritmo *OPTICS* genera 6 clusters distintos, mientras que *Calinski-Harabasz* nos dio 3 y en *K-Means* utilizamos 5. Pero sin importar el algoritmo vemos que los clusters presentan valores muy parecidos entre ellos, aunque se pueden apreciar comportamientos notorios.

El cluster 0 agrupa personas que consumen alcohol con frecuencia, duermen menos y presentan niveles de presión y colesterol moderados, pero no bajos. Mientras que en el cluster 4 se agrupan a personas que no consumen alcohol, con una higiene del sueño similar, pero con una presion sistólica mayor, aunque un IMC menor. Lo cual podría hacer sentido debido a que la presion sistólica es la principal en verse afectada al tener una mala noche de sueño.

Mientras que en el cluster 3 se encuentran las personas con hábitos mas saludables, debido a que presentan valores altos en pasos diarios, horas de sueño y agua consumida, teniendo asi valores de colesterol bajos.

Concluyendo con que K-Means es un algoritmo muy eficiente bajo la metrica de Inercia para generar grupos en la información, mientras que OPTICS va un paso adelante pues generó un numero mas alto de grupos, pero no todo lo que brilla es oro. Pues el hecho de tener mas agrupaciones no siempre es lo mas eficiente y K-Means lo demuestra, con sus agrupaciones da unas buenas descripciones del comportamiento de la informacion.

6.2. Algoritmos Supervisados para Pronosticos

Concluimos con los resultados obtenidos que en el modelo Lasso las predicciones estan mas dispersas, tienen un rango un poco mas amplio que el que presenta el modelo de Regresión lineal, pero en ambos muy pocos puntos siguen la diagonal o emphLínea ideal aunque el modelo de Regresión al presentar una menor variabilidad, se podria inferir que mas puntos de los pronosticados se encuentran siguiendo esta diagonal, en proporciones generales ambos modelos tienen un numero muy reducido de puntos que la siguen.

Tambien, con base al algoritmo de Random Forest aplicado, que las predicciones si pueden presentar una mayor variacion en valores pronosticados, pero tambien presenta una mayor incidencia de **outliers** que se pueden detectar a primera vista.

Mediante la metrica de MAE obtenemos que el modelo Lasso esta ligeramente más cerca de los valores reales que el modelo de Regresion Lineal, aunque es una mejora en el rendimiento muy pequeña, sigue siendo mejor, adicionalmente de que se presenta una mayor variacion, lo cual imita mejor el comportamiento real de la presión diastólica, resaltando que el Random Forest calibrado presenta un mayor error, en comparacion a los otros modelos utilizados, lo cual explicaria el porque existe una mayor variabilidad en los pronosticos y por ende una mayor incidencia de **outliers**.

Por lo que se considera que el modelo Lasso presenta un desempeño mejor que el modelo de Regresión lineal y el Random Forest.

Referencias

Fernández, D., Pezzi, J. P., & Caruso, G. (s. f.). *Apnea del sueño e hipertensión arterial. Diagnóstico y terapéutica.*https://www.saha.org.ar/uploads/pdf/Cap.%2E096.pdf

Scikit-learn. (s. f.). *Lasso — scikit-learn 1.5.2 documentation.* https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html

Mayo Clinic. (2025, marzo 07). *¿La falta de sueño puede provocar presión arterial alta?* https://www.mayoclinic.org/es/diseases-conditions/high-blood-pressure/expert-answers/sleep-deprivation/faq-20057959

Filippi, A., … (2024). *Smoking cessation decreases arterial blood pressure in hypertensive smokers: A subgroup analysis of the randomized controlled trial GENTSMOKING.* https://pubmed.ncbi.nlm.nih.gov/38756738/

Benavides, Alberto. (2025). Aprendizaje Automático. Repositorio en GitHub: https://github.com/albertobenavides/aprendizaje_autom/tree/master

Cappuccio, F. P., ... (2023). *Examining Daily Associations Among Sleep, Stress, and Blood Pressure Across Adulthood.* https://pubmed.ncbi.nlm.nih.gov/36680526/