PROTECO - Machine Learning

Proyecto - Dataset sobre el cáncer de mama mediante los métodos PCA, de regresión logística y K-Means

Pérez Ruiz Daniel Michell
Domingo 20 de septiembre 2020.
Github: https://github.com/danielmichellpr
https://github.com/danielmichellpr/PROTECO-Machine-Learning

Resumen

En este proyecto se trabaja con el conjunto de datos proporcionado por Wisconsin sobre el cáncer de mamá, en el cuál se tienen 30 variables para determinar si el tipo un tumor es de tipo maligno o benigno, es decir, es un problema de dimensionalidad 30 con sólo 2 clases. Se realiza un método de análisis de componentes principales para reducir las variables a sólo dos, con lo cual es posible identificar las clases en un espacio bidimensional. Una vez realizado este método, sobre los datos obtenidos se efectuan dos métodos más, el de regresión logística y uno de K-Means, con el primero se obtuvo la separación de los datos, clasificando si el tumor es benigno o maligno, mientras que con K-Means se retiraron las etiquetas y se logró determinar a que clase pertenece cada dato, obteniendo así resultados satisfactorios para ambos métodos, demostrando una aplicación de algoritmos de Machine Learning.

1. Introducción

El dataset del cáncer de mamá, es un dataset clásico al aprender Machine Learning, ya que es consiste en conjunto de datos simples, pero en el cual se maneja una cantidad considerable de datos, con los cuales se plantea describir si un tumor en la mama es maligno o benigno, tal como se muestra en la Fig 1.

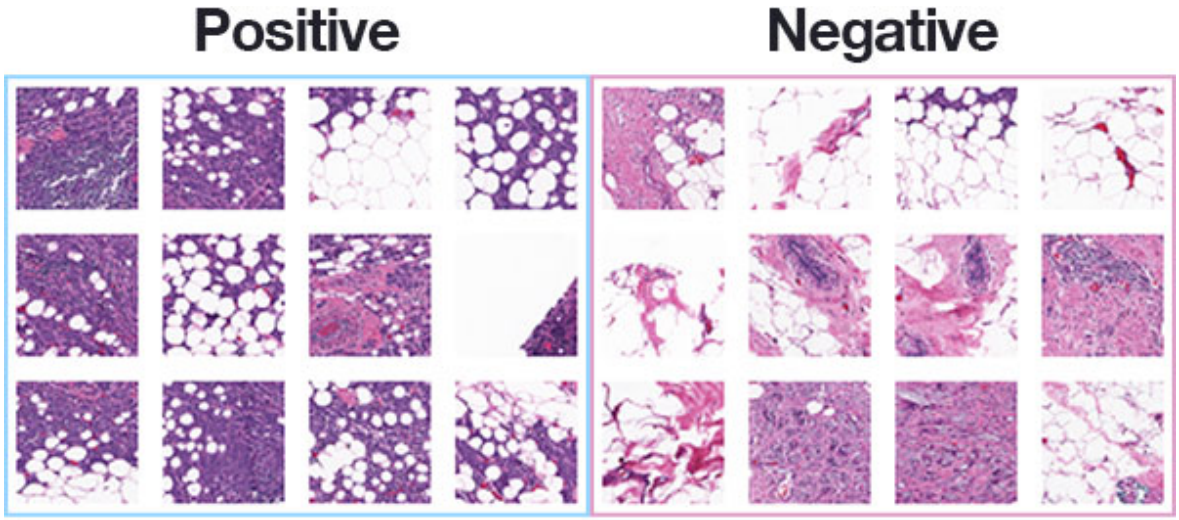

Figura 1: Visualización de un tumor en la mama, tumor maligno (positivo) y benigno (negativo)

El dataset contiene la siguiente información [1]:

- 2 Clases: Maligno(M) o Benigno(B), es decir, es un problema binario, asociando el valor 0 a maligno y 1 a benigno.
- El total de muestras/casos son 569, en los cuales 212 pertenecen a un tumor maligno o cáncer de mama, mientras que 357 son de un tumor benigno.
- El dataset es de dimensional 30, por lo que hay 30 características a tomar en cuenta para determinar el tipo de clase en cada caso: el radio, la textura, perímetro, área, suavidad, compacidad, concavidad, simetría, entre otros.

Es de interés realizar modelos de Machine Learning que describan el comportamiento de los tumores, para determinar las características que hacen que un tumor se vuelva maligno o benigno. Es por esto que se implementan tres diferentes métodos: *Análisis de Componentes Principales* (PCA), *Regresión Logística* y *K-Means*. A continuación se muestran los resultados sobre cada método empleado; cada uno se realiza desde **Scratch** y desde la paquetería de **Sklearn**.

2. Análisis de Componentes Principales (PCA)

El análisis de componentes principales evita el uso de muchos target, a partir de la *combinación* de las diferentes características, esto disminuye el tiempo de entrenamiento, elimina el ruido de los datos y permite su visualización. Es un algoritmo de aprendizaje no supervisado ya que las variables de salida no se centran en predecir, sino en re-interpretar las entradas. Los pasos en este modelos son [2]:

1.- Cargar los datos a analizar.

2.- Escalar los datos (normalizarlos), para que así un dato no valga más que otro sólo por sus magnitudes.

3.- Crear una matriz de covarianza a partir de los datos.

4.- Calcular los eigenvalores y eigenvectores de la matriz creada.

5.- Almacenar el número de eigenvectores del que queremos nuestro nuevo dataset.

6.- Graficar.

En un principio esto se realizó desde Scratch, en la Fig. 2 se observa a la izquierda los datos sin etiquetas, a la derecha se muestran con etiqueta donde cero representa los casos malignos y uno representa los casos benignos.

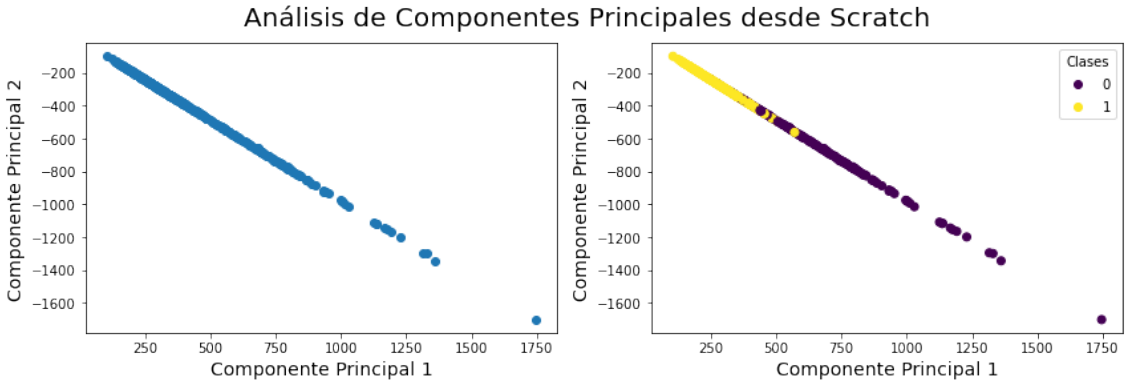

Figura 2: Análisis de componentes principales desde Scratch, mostrando de manera bidimensional si una persona posee o no un tumor maligno o benigno, datos con etiquetas y sin etiquetas.

De igual manera se implementó con Sklearn, este segundo forma es más precisa con los datos y permite una mejor visualización, donde se obtiene una notable separación entre los datos de las personas con un tumor benigno y uno maligno.

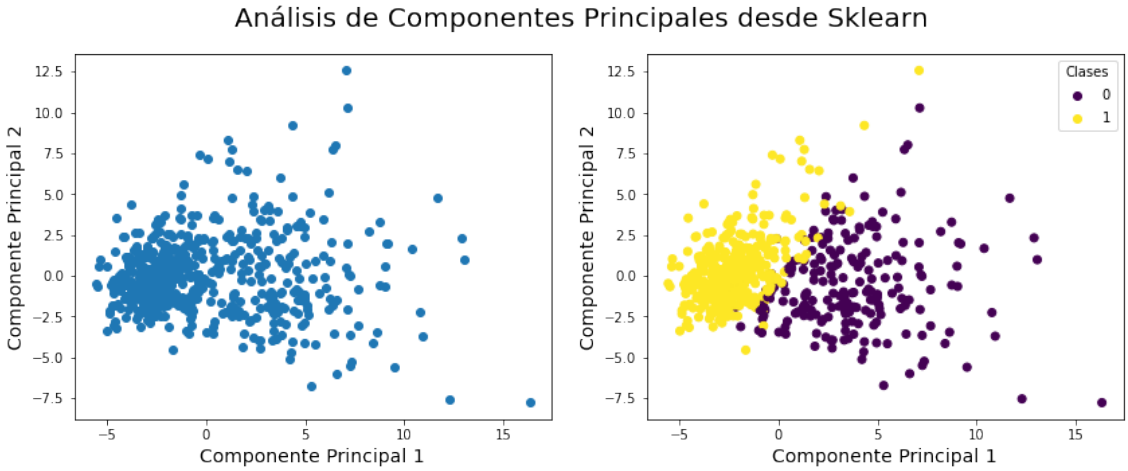

Figura 3: Análisis de componentes principales desde Sklearn, mostrando de manera bidimensional si una persona posee o no cáncer de mama.

3. Regresión Logística

Una vez implementado el método PCA, se continuo con los datos obtenidos de dicho método, realizando una regresión logística. Una regresión logística es un método de aprendizaje supervisado el cual utiliza un algoritmo para predecir entre dos opciones. La función de hipótesis en la regresión logística es una función sigmoidal:

$$h_\theta(X) = \frac{1}{1 + e^{-(\theta^T X)}} \tag{1}$$

El método consta de los siguientes pasos:

1.- Separación de datos de test y entrenamiento.

2.- Definimos la función de hipótesis.

3.- Continuamos con la función de costos y el gradiente de descenso.

4.- Finalmente se gráfica el borde decisión que determina la separación entre los dos tipos de datos.

Los datos resultantes por Sklearn en el método PCA mostraron una mejor visualización y entendimiento de los datos, por lo que fueron tomados para la implementación del código desde Scratch y Sklearn en el método de Regresión Logística.
En este caso se obtuvieron resultados similares en ambas implementaciones. A continuación se muestran los datos obtenidos, así como la línea de decisión que nos clasifica unos datos de otros.

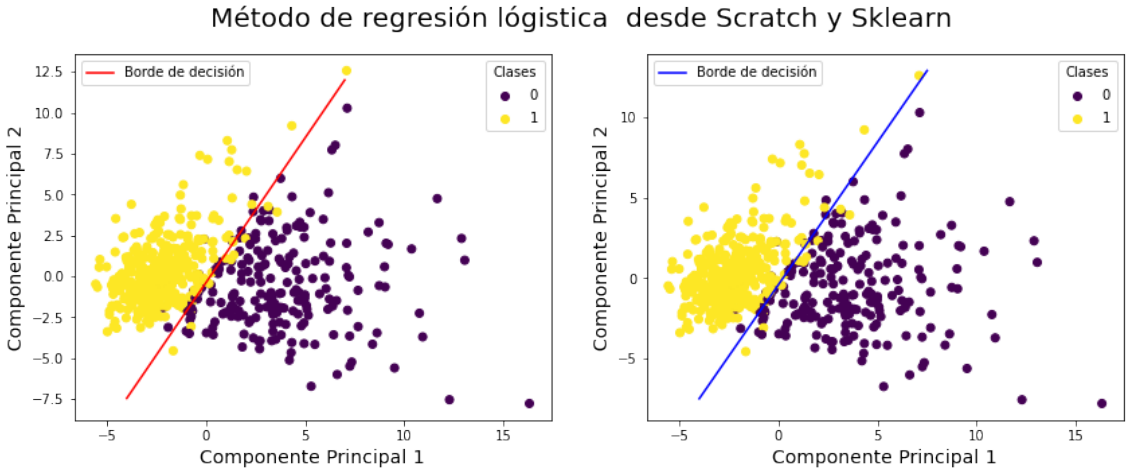

Figura 4: Método de Regresión logística, el borde de decisión representa la clasficación de las dos clases.

4. K-Means

El último método implementado fue el de K-Means, para esto primero retiramos las etiquetas de los datos, ya que se trata de método de aprendizaje no supervisado. El algoritmo K-Means nos permite empaquetar datos, a través de ciertas similitudes que encuentre en ellos, los pasos para realizarlo son [3, 4]:

1.- Se especifica el número de grupos que se quieren identificar, K.

2.- Se realiza un proceso de inicialización (se elige un centroide para cada grupo de datos aleatoriamente).

3.- Continuamos con un proceso de asignación (se asigna el centroide a cada dato cercano).

4.- Se realiza actualización (el centroide se ve desplazado hasta hallar un lugar optimo).

Para elegir el número correcto de clusters se implementa un algoritmo de codo, en el cual nos dice la distorsión que se presenta por número de clusters, cuando la distorsión ya no es significativa, se toma el número de cluster más cercano. En nuestro caso ya sabemos que tenemos dos tipos de clases, pero de no ser así con el método de codo es posible realizar una aproximación, tal como se muestra en la Fig. 5.

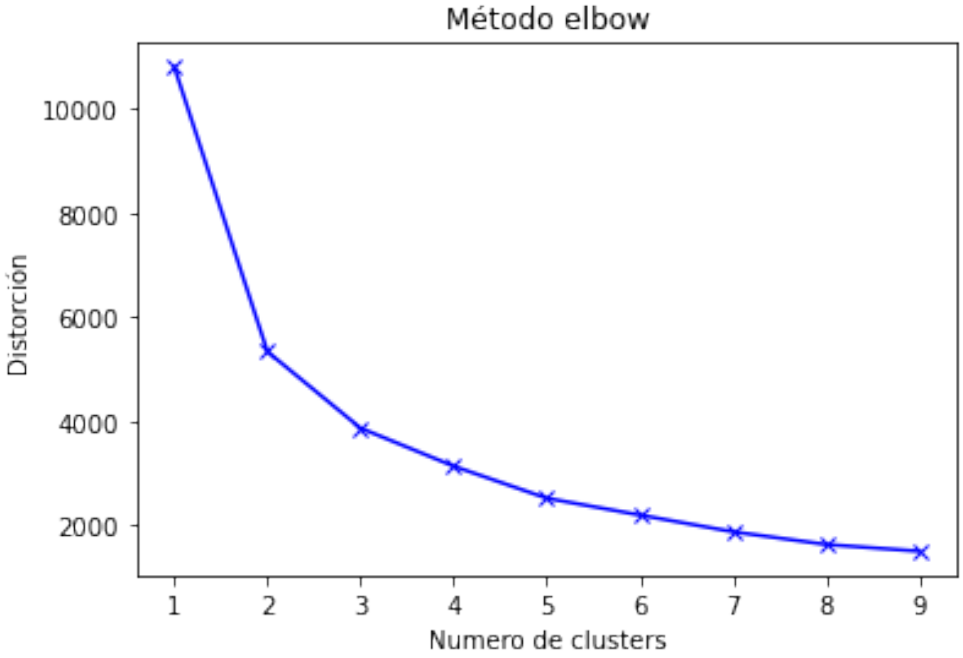

Figura 5: Método de codo para determinar el número correcto de clusters que deben implementarse en KMeans.

En este método podemos observar que la distorsión empieza a dejar de ser significativa a partir de 4 o 5 clusters, por lo que esto nos podría sugerir que existen más de dos clases y no tan sólo el tumor es maligno o benigno, sino que también podemos tener situaciones en las que el tumor se encuentra fuera de las posibilidades de ser maligno, alguna otra clase en las que el tumor es poco probable, otra en el que es potencialmente probable, una más donde sea maligno y una donde se tenga que atender de inmediato. Con esto notamos que es posible obtener una mayor información de la proporcionada para este tipo de algoritmo.
A continuación se muestran las gráficas obtenidas desde Scratch y Sklearn para este método, con 2 y 5 clusters.

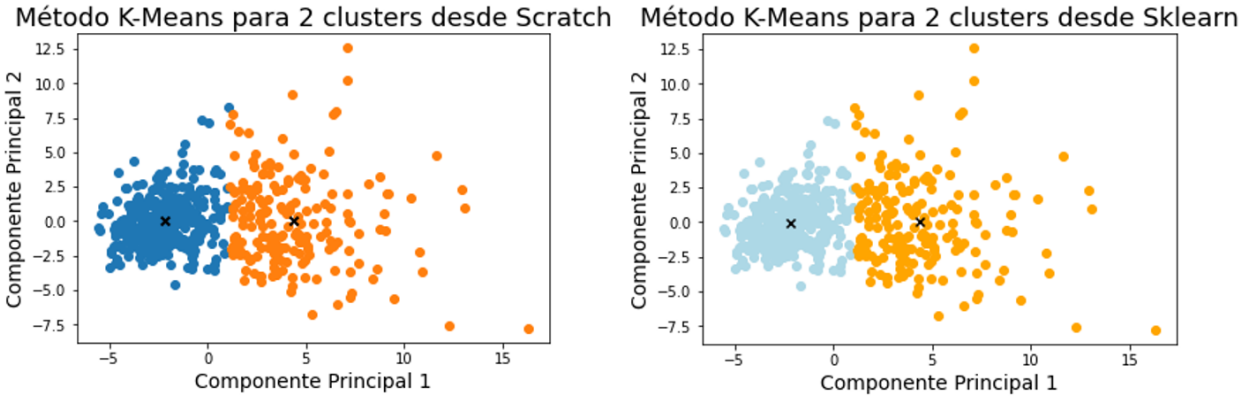

Figura 6: Método de K-Means para 2 clusters, mostrando si una persona tiene un tumor maligno o benigno.

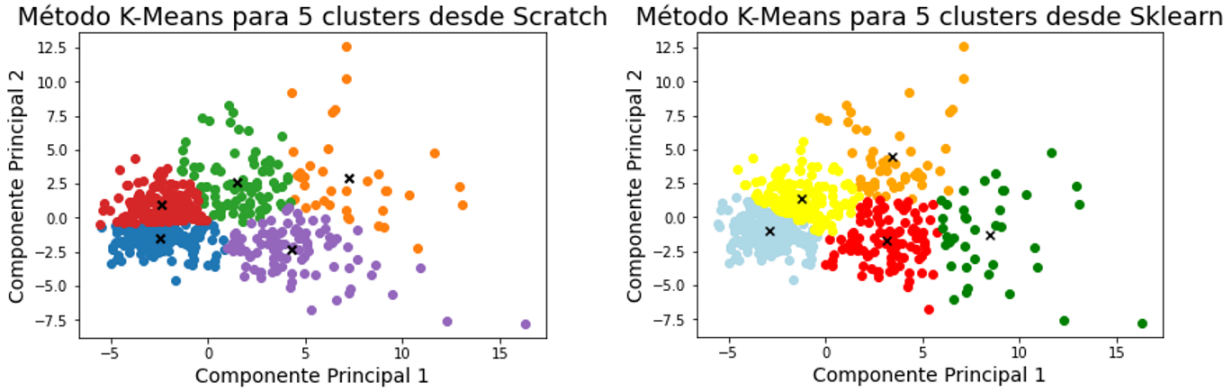

Figura 7: Método de K-Means para 5 clusters, mostrando si una persona tiene un tumor maligno o benigno..

5. Discusión y conclusión

Se exploró el dataset sobre el cáncer de mama mediante tres distintos métodos, en el método PCA, se redujeron las variables del dataset a dos, para obtener una visualización. Los resultados fueron contrastantes en el método implementado con Scratch y con Sklearn, pero esto se debe a distintos factores, dependiendo el algoritmo que sigue cada uno. Podemos observar en la Fig. 8 que en cuanto mayor sea el valor en alguna característica tiende a haber una probabilidad mayor de que el tumor sea maligno, con esto podemos concluir que la gráfica obtenida desde Scratch establece una relación inversa en una variable, ya que mientras menor es la segunda componente principal, mayor es la probabilidad de que el tumor sea maligno, mientras que con el método Sklearn la gráfica es similar a las de la Fig. 8, lo cual nos muestra una tendencia más adecuada en los datos, resultando que mientras mayor sean los datos en una variable, mayor será la probabilidad de que el tumor sea maligno.

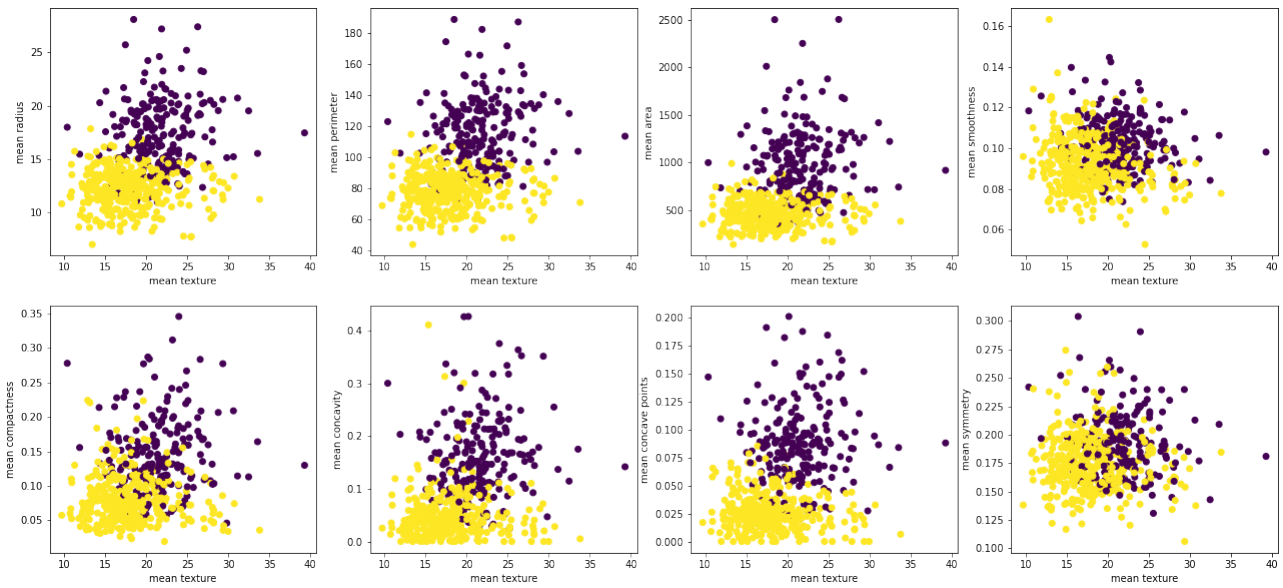

Figura 8: Comparación entre diferentes variables contra el radio.

En el método de regresión logística los resultados fueron muy similares, esto se debe a que nuestras datos ya tenían un primer filtro que el método de PCA, es por eso que al implementarlo desde Scratch no requirió tanto costo y tantas iteraciones para llegar a un resultado óptimo. La gráfica de costo contra iteración se muestra a continuación, en dicha gráfica se obtuvo un último costo de 55.50.

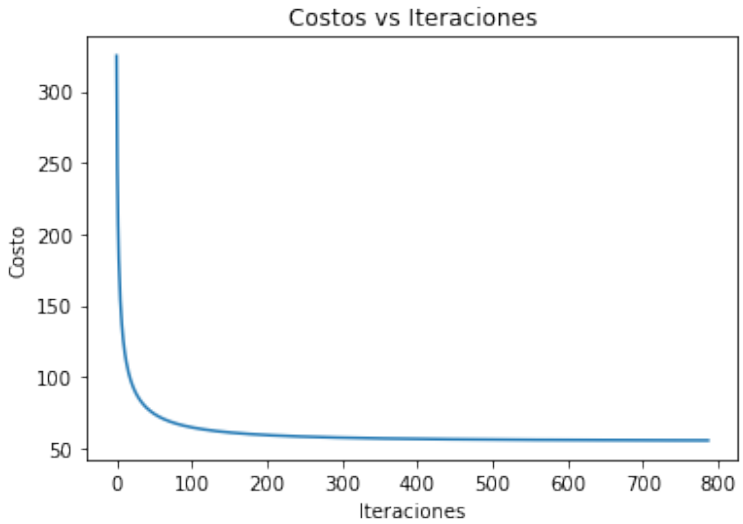

Figura 9: Gráfica de costo contra iteración, las iteraciones necesarias son pocas para obtener un resultado favorable.

Se atribuye al poco costo que realiza el algoritmo a que ambas maneras de realizar el método de regresión logística obtengan resultados similares.

Por último el método K-Means muestra que puede existir más allá de sólo 2 clases, aunque como principalmente el punto de inflexión se halla en 2 clusters, existe una variación significativa hasta 5 clusters. Esto nos muestra que el método de K-Means abre una posibilidad a ir más allá de las clases que tenemos en nuestro target, mostrando en un caso real, que pacientes requieren una atención más delicada que otros.

Referencias

[1] Machine Learning Repository, Breast Cancer Wisconsin (Diagnostic) Data Set, Fecha de Consulta: 17 de septiembre de 2020 https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)

[2] Na8, (2018), Aprende Machine Learning, Comprende Principal Component Analysis, Fecha de Consulta: 17 de septiembre de 2020 https://www.aprendemachinelearning.com/comprende-principal-component-analysis/

[3] Na8, (2018) K-Means en Python paso a paso, Fecha de Consulta: 17 de septiembre de 2020, https://www.aprendemachinelearning.com/k-means-en-python-paso-a-paso/

[4] Martínez H., José, (2020), IArtificial.net, Clustering (Agrupamiento), K-Means con ejemplos en Python, Fecha de Consulta: 17 de septiembre de 2020, url: https://www.iartificial.net/clustering-agrupamiento-kmeans-ejemplos-en-python/