Informe Práctica 2

Mario Polaino, Antonio Moya, Leon Skalczynski, Fernando García

mariopolaino@uma.es, amoyam05@uma.es, leonskal@uma.es, nano_2005@uma.es
3º Software. UMA.

En esta práctica exploramos la normalización, selección de características y reducción de la dimensionalidad, para posteriormente configurar una validación cruzada de 5 iteraciones, para asegurarnos de que los resultados no dependen de cómo se dividen los datos entre entrenamiento y test. Tenemos un archivo de ejemplo donde se nos muestra cómo estandarizar un conjunto de datos para acto seguido aplicar el algoritmo de PCA.

1 Representación de datos

1.1 Importación de librerías y carga de datos original

Importamos las librerías matplotlib y numpy, además de scikit-learn de la cual obtenemos el dataset que vamos a manipular. Del mismo obtenemos como características principales el ancho y la longitud del sépalo y representamos el diagrama de dispersión por clases. Podemos apreciar que la setosa tiene un ancho del sépalo generalmente superior y diferenciado de las demás, así como las longitudes de sépalo más grandes se las atribuimos a la virgínica.

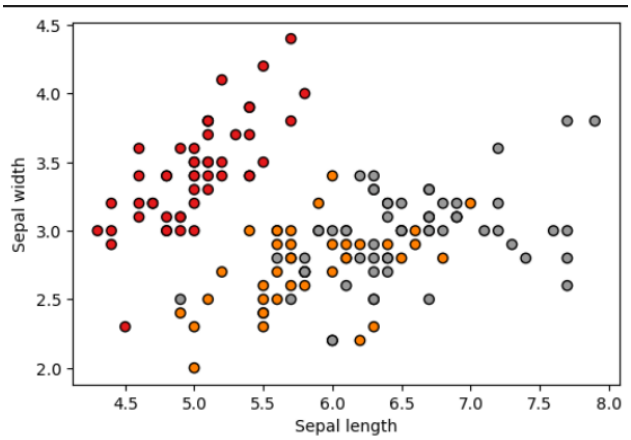

Fig. 1. Diagrama del conjunto original

1.2 Conjunto de datos estandarizado

Importamos StandardScaler de sklearn.preprocessing para aplicarle la estandarización al conjunto de datos original usado anteriormente. Mostramos la distribución de estas mismas características del nuevo conjunto resultante. Observamos el mismo resultado que en el caso anterior pero en una distinta escala.

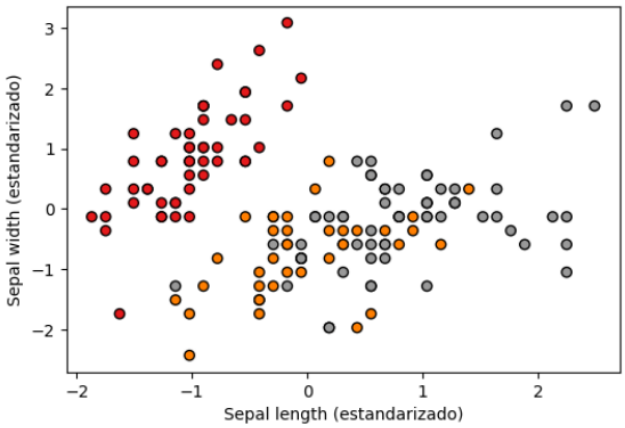

Fig. 2. Diagrama del conjunto estandarizado

1.3 Conjunto de datos normalizado

Importamos MinMaxScaler de sklearn.preprocessing y, al igual que en el caso de estandarización, procedemos a la normalización del dataset original en un rango fijo entre 0 y 1. Visualizamos el diagrama de este nuevo conjunto, cuya única variación es el rango en el que se encuentra.

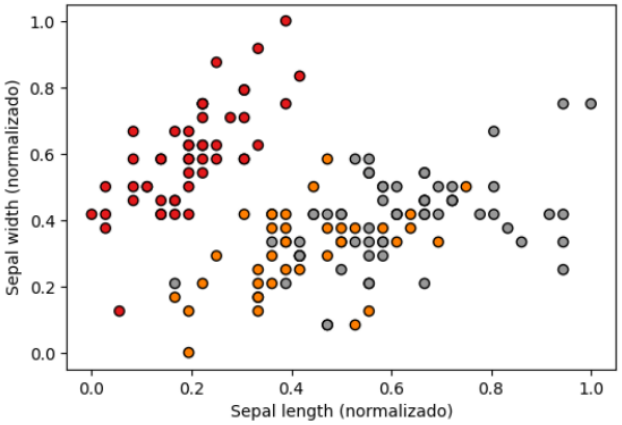

Fig. 3. Diagrama del conjunto normalizado

2 Análisis de componentes principales

2.1 PCA al conjunto original

Utilizamos el algoritmo de PCA como herramienta para reducir la dimensionalidad. Importamos PCA de la librería sklearn. Aplicamos el algoritmo al conjunto original y mostramos qué porcentaje de varianza explica cada componente. Ésto nos servirá para conocer el número de componentes necesario para llegar al 80% y 95% de varianza que se nos pedía. Vemos que la primera componente explica un 92,46% de ésta, y la segunda un 5,31%. Ésto indica que para alcanzar el 80 requerido nos valdría con una única componente, y para el 95

añadiríamos una segunda. Mostramos el diagrama de dispersión y boxplot de las dos versiones del algoritmo realizadas. Podemos concluir que las setosas son las flores más similares entre sí, mientras que las virgínicas se aprecia que son más heterogéneas. Para el caso del PCA 80, al sólo requerir una componente mostramos en el diagrama su componente respecto a las muestras.

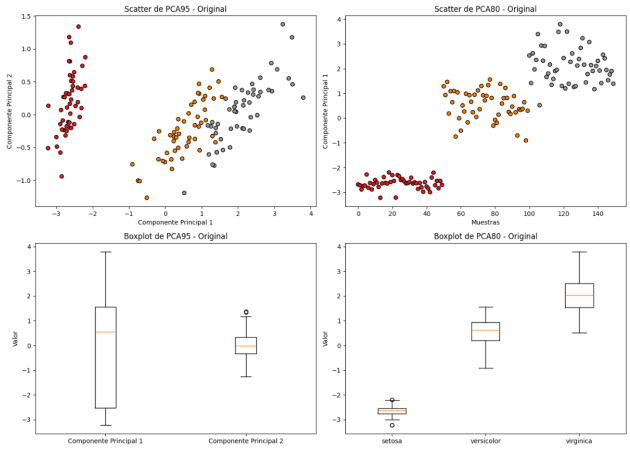

Fig. 4. PCA conjunto original

2.2 PCA al conjunto estandarizado

Implementamos el algoritmo al conjunto estandarizado y de igual forma revelamos el porcentaje de varianza explicada por componentes y apreciamos que la primera componente esta vez explica un 72,96% y la segunda un 22,85%. Ésto quiere decir que vamos a necesitar las dos componentes para llegar a los porcentajes exigidos. De nuevo, presentamos los diagramas de dispersión para cada versión. Concluimos que, como previamente dedujimos las especies son claramente separables con sólo dos dimensiones.

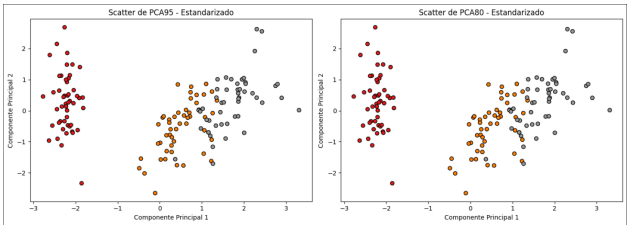

Fig. 5. PCA conjunto estandarizado

2.3 PCA al conjunto normalizado

Empleamos una vez más el proceso con el conjunto normalizado y continuamos en la línea de averigüar el porcentaje que nos revela cada componente. La primera data del 84,14% y la segunda del 11,75%. Para realizar el PCA 80 requerido utilizaremos sólo la primera y echaremos mano de la segunda para el 95. Finalmente, volvemos a ilustrar los diagramas de dispersión y boxplot para éstos casos. La información que podemos obtener de éstas figuras es la mencionada anteriormente.

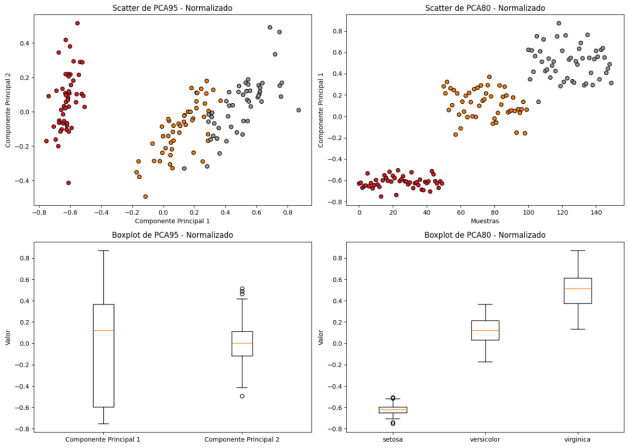

Fig. 6. PCA conjunto normalizado

3 Validación cruzada

Para finalizar esta práctica, hemos implementado una validación cruzada manual de k=5 pliegues, lo que nos da la división estándar de 80% para entrenamiento y 20% para prueba en cada iteración. El enunciado requería que cada pliegue tuviera el mismo número de muestras por clase (es decir, que fuera estratificada), así que no podíamos simplemente cortar el dataset de 150 muestras en 5 trozos de 30, ya que los datos de Iris están ordenados por clase.
Lo que hicimos fue, primero, separar todo el conjunto de datos (conjunto_datos_original) en tres grupos distintos, uno para cada clase (Setosa, Versicolor y Virginica), quedándonos con 50 muestras en cada grupo. Después, dividimos cada uno de esos grupos de 50 en 5 "trozos" más pequeños de 10 muestras cada uno. Por ejemplo, la clase Setosa se partió en "Trozo 1 Setosa" (muestras 1-10), "Trozo 2 Setosa" (muestras 11-20), y así hasta 5.
Finalmente, construimos nuestros 5 pliegues (o "partes") finales. El Pliegue 1 se creó juntando el "Trozo 1" de Setosa, el "Trozo 1" de Versicolor y el "Trozo 1" de Virginica, dándonos un pliegue de 30 muestras perfectamente balanceado. Repetimos esto para los demás trozos hasta tener los 5 pliegues.
Una vez con los 5 pliegues listos, generamos los archivos:
Iteración 1: El test1.csv fue el Pliegue 1 (30 muestras), y el training1.csv fue la unión de los Pliegues 2, 3, 4 y 5 (120 muestras).
Iteración 2: El test2.csv fue el Pliegue 2, y el training2.csv fue la unión de los Pliegues 1, 3, 4 y 5.
...y así sucesivamente hasta completar las 5 iteraciones, guardando todos los archivos CSV como se pedía.

4 Uso de inteligencia artificial

Nos hemos valido del uso de ésta herramienta para la ayuda y corrección de errores y repaso del código. Además lo utilizamos para conocer las importaciones necesarias para llevar a cabo la práctica y entender cómo funcionaban.

UNIVERSIDAD
DE MÁLAGA