Universidad de Córdoba
Escuela Politécnica Superior de Córdoba

Ingeniería Informática
Especialidad: Computación
Cuarto curso. Primer cuatrimestre

Introducción a la minería de datos

Práctica 1: Datos y exploración de datos

Valentín Gabriel Avram Aenachioei
03524931C
p92avavv@uco.es

Curso académico 2022-2023
Córdoba, 9 de junio de 2023

Índice

Índice de figuras II

Índice de tablas III

1. Preguntas práctica 1 **1**

1.1. Evaluar el árbol de decisión y el vecino más cercano sobre los datos originales 1

1.2. Estudie el efecto de la normalización (reescalar en el intervalo [0, 1]) y la estandarización $\mu = 0, \sigma = 1$ sobre el error de clasificación usando el árbol de decisión y el vecino más cercano 3

1.3. Estudie el efecto del análisis en componentes principales sobre el árbol de decisión y el vecino más cercano. 4

1.4. Estudie el efecto del muestreo aleatorio del 10 % de las instancias sin reemplazamiento sobre el árbol de decisión y el vecino más cercano.Compare los resultados con un muestreo del mismo porcentaje pero estratificado. 6

1.5. Seleccione un conjunto de datos con valores perdidos y dos métodos de imputación. Estudie el efecto de los métodos de imputación sobre los dos clasificadores. 7

1.6. Seleccione un conjunto de datos y un método de discretización. Estudie el efecto del método sobre los dos clasificadores 7

1.7. Realice un box plots de los datasets. Indique qué información se puede obtener de los gráficos. En al menos un ejemplo construya el mismo gráfico por clases. ¿Se puede obtener información adicional para este caso? 9

1.8. Realice un scatter plot matricial de los datasets. Indique qué información se puede obtener de los gráficos 12

1.9. Represente la matriz de correlación entre las variables usando un mapa de calor e indique la información que se puede extraer del gráfico 14

1.10. Realice el mismo mapa de calor usando la correlación entre las instancias (esta operación es equivalente a realizar la correlación en la matriz de datos traspuesta). Indique qué información se puede obtener de los gráficos 17

Índice de figuras

1. Árbol de decisión para el dataset Iris 2
2. Precisión con dos componentes principales 5
3. Precisión con tres componentes principales 5
4. Precisión con tres componentes principales 8
5. BoxPlot dataset Iris 9
6. BoxPlot dataset CPU 10
7. BoxPlot dataset Diabetes 10
8. BoxPlot dataset Ionosphere 11
9. BoxPlot dataset Segment Challenge 11
10. BoxPlot por clases dataset Iris 12
11. Scatter Plot dataset Iris 13
12. Scatter Plot dataset CPU 13
13. Scatter Plot dataset Diabetes 14
14. Matriz Correlación dataset Iris 15
15. Matriz Correlación dataset CPU 15
16. Matriz Correlación dataset Diabetes 16
17. Scatter Plot dataset Ionosphere 16
18. Matriz Correlación dataset Segment Challenge 17
19. Matriz Correlación instancias dataset Iris 18

Índice de tablas

1. Resultados al aplicar arboles de decisión 1
2. Resultados al aplicar K-vecinos más cercanos 1
3. Precision arboles de decisión al normalizar 3
4. Precision kNN al normalizar 3
5. Precision arboles de decisión al estandarizar 3
6. Precision kNN al estandarizar 4
7. Precision arboles de decisión con PCA de dos componentes . . 4
8. Precision kNN con PCA de dos componentes 4
9. Precision arboles de decisión con PCA de tres componentes . . 6
10. Precision kNN con PCA de tres componentes 6
11. Precisión de los clasificadores con muestreo aleatorio 10 % . . . 6
12. Precisión de los clasificadores con muestreo estratificado 10 % . 7
13. Precisión de los clasificadores con diferente imputación 7
14. Precisión con diferentes umbrales de binarizador 8

En este documento se recogerán los resultados obtenidos realizando las pruebas indicadas en el guión de la práctica 1, así como el respectivo análisis de esos resultados. No se hará mención a las implementaciones en código necesarias para resolver cada problema.

1. Preguntas práctica 1

1.1. Evaluar el árbol de decisión y el vecino más cercano sobre los datos originales

Para esta evaluación, se han usado 5 datasets: *Iris, CPU, Diabetes, Ionosphere y Segment Challenge*. La precision obtenida al aplicar Arboles de decisión y K-vecinos mas cercanos sobre los datasets se representan en las tablas 1 y 2 respectivamente. Para los K-vecinos más cercanos, se ha usado un valor $vecinos = 3$

Dataset	**Accuracy**
Iris	*95.555 %*
CPU	*3.174 %*
Diabetes	*67.965 %*
Ionosphere	*83.962 %*
Segment	*96 %*

Tabla 1: Resultados al aplicar arboles de decisión

Dataset	**Accuracy**
Iris	*97.777 %*
CPU	*0 %*
Diabetes	*68.831 %*
Ionosphere	*91.509 %*
Segment	*87.078 %*

Tabla 2: Resultados al aplicar K-vecinos más cercanos

Para estos datasets, relativamente sencillos y de pequeño tamaño, la precisión obtenida para ambos procedimientos es relativamente alta, siendo peor a cuanto mayor y mas complejo sea el conjunto de datos. Se puede apreciar una clara diferencia entre los datasets *Iris* y *CPU*, al ser el dataset *CPU* un dataset preparado para problemas de regresión, siendo necesario discretizar

la salida para ser usado en problemas de clasificación. A modo de visualización gráfica, podemos ver el desarrollo del árbol de decisión para el dataset *Iris* en la figura 1.

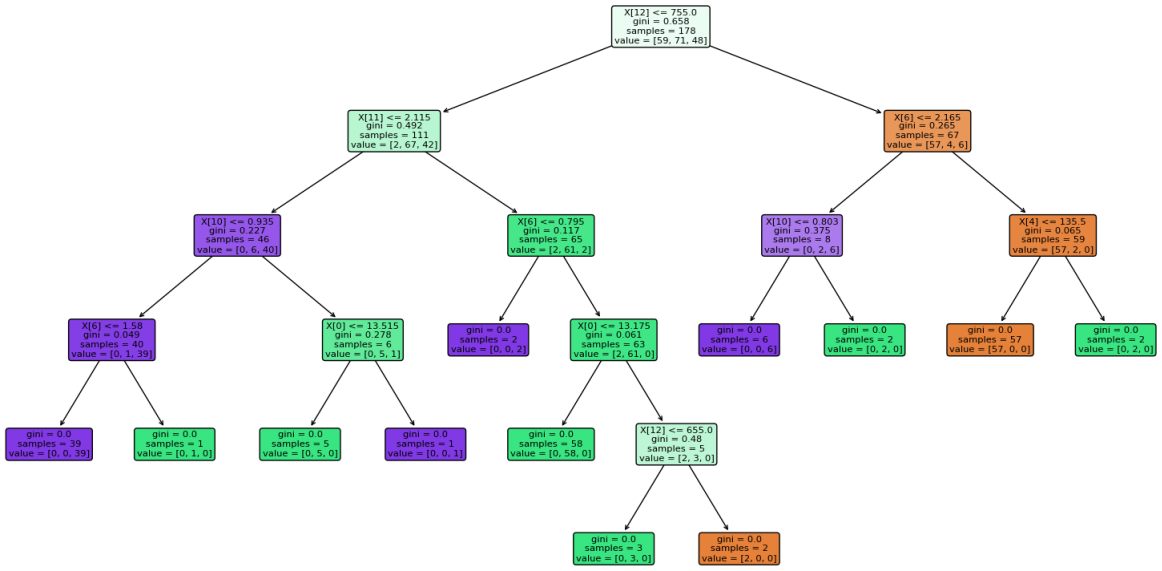

Figura 1: Árbol de decisión para el dataset Iris

1.2. Estudie el efecto de la normalización (reescalar en el intervalo [0, 1]) y la estandarización $\mu = 0, \sigma = 1$ sobre el error de clasificación usando el árbol de decisión y el vecino más cercano

Para poder comparar resultados, se han usados los mismos datasets que en el apartado anterior. La medida de error obtenida tras normalizar los datos se pueden ver en la tabla 3 y tabla 4. La medida de error obtenida tras estandarizar los datos se pueden ver en las tablas 5 y 6.

Dataset	**Error**
Iris	*4.444 %*
CPU	*93.650 %*
Diabetes	*28.138 %*
Ionosphere	*18.867 %*
Segment	*4.222 %*

Tabla 3: Precision arboles de decisión al normalizar

Dataset	**Error**
Iris	*2.222 %*
CPU	*100 %*
Diabetes	*28.138 %*
Ionosphere	*17.924 %*
Segment	*7.777 %*

Tabla 4: Precision kNN al normalizar

Dataset	**Error**
Iris	*0.0 %*
CPU	*98.412 %*
Diabetes	*29.437 %*
Ionosphere	*14.150 %*
Segment	*6.888 %*

Tabla 5: Precision arboles de decisión al estandarizar

Dataset	Error
Iris	*2.222 %*
CPU	*100 %*
Diabetes	*35.064 %*
Ionosphere	*16.037 %*
Segment	*7.333 %*

Tabla 6: Precision kNN al estandarizar

1.3. Estudie el efecto del análisis en componentes principales sobre el árbol de decisión y el vecino más cercano.

Primero, se realiza el estudio aplicando dos componentes. Los resultados de precisión para Arboles de decision y kNN se representan en las tablas 7 y 8 respectivamente. Además, se puede visualizar gráficamente esta precisión en la figura 2.

Dataset	Precisión
Iris	*88.888 %*
CPU	*1.587 %*
Diabetes	*66.666 %*
Ionosphere	*69.811 %*
Segment	*64.444 %*

Tabla 7: Precision arboles de decisión con PCA de dos componentes

Dataset	Precisión
Iris	*88.888 %*
CPU	*3.174 %*
Diabetes	*74.891 %*
Ionosphere	*82.075 %*
Segment	*66.222 %*

Tabla 8: Precision kNN con PCA de dos componentes

Posteriomente, se realiza el mismo análisis con tres componentes. Los resultados de precisión obtenidos son representados en las tablas 9 y 10 para Árboles de decisión y kNN respectivamente. Además, se pueden visualizar los resultados de precisión obtenidos para cada dataset en la figura 3.

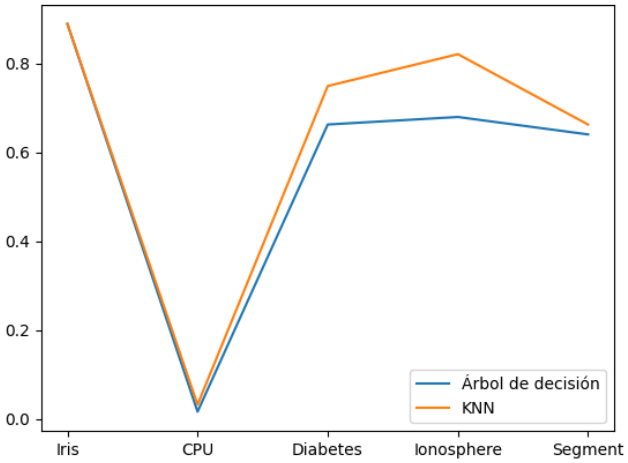

Figura 2: Precisión con dos componentes principales

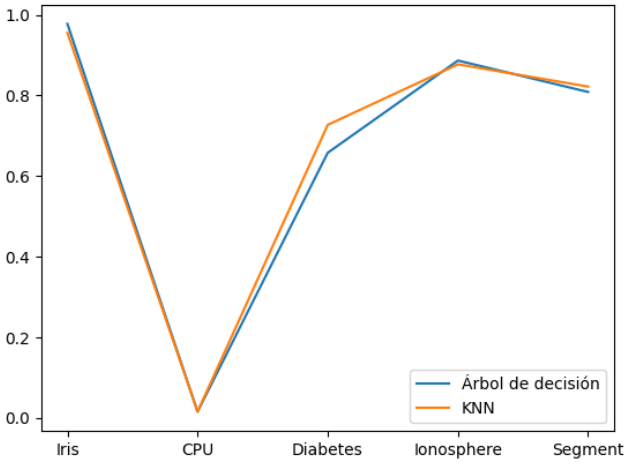

Figura 3: Precisión con tres componentes principales

Dataset	Precisión
Iris	*97.777 %*
CPU	*1.587 %*
Diabetes	*65.801 %*
Ionosphere	*88.679 %*
Segment	*90.888 %*

Tabla 9: Precision arboles de decisión con PCA de tres componentes

Dataset	Precisión
Iris	*97.777 %*
CPU	*1.587 %*
Diabetes	*72.727 %*
Ionosphere	*87.735 %*
Segment	*82.222 %*

Tabla 10: Precision kNN con PCA de tres componentes

1.4. Estudie el efecto del muestreo aleatorio del 10 % de las instancias sin reemplazamiento sobre el árbol de decisión y el vecino más cercano.Compare los resultados con un muestreo del mismo porcentaje pero estratificado.

Primeramente estudiamos el efecto de aplicar el muestreo aleatorio del 10 % de las instancias sin reemplazamiento, tanto sobre el árbol de decisión y kNN. La precisión de ambos clasificadores se recogen en la tabla 11.

Dataset	Precisión árbol	Precisión kNN
Iris	*100 %*	*100 %*
Diabetes	*75.324 %*	*75.324 %*
Ionosphere	*91.666 %*	*88.888 %*
Segment	*92 %*	*91.333 %*

Tabla 11: Precisión de los clasificadores con muestreo aleatorio 10 %

Posteriomente, se comprueba la precisión de ambos clasificadores usando un muestreo estratificado del 10 %. De nuevo, las precisiones obtenidas en ambos clasificadores se recogen en la tabla 12.

Dataset	Precisión árbol	Precisión kNN
Iris	*93.333 %*	*100 %*
Diabetes	*75.324 %*	*76.6234 %*
Ionosphere	*91.666 %*	*77.777 %*
Segment	*97.333 %*	*93.333 %*

Tabla 12: Precisión de los clasificadores con muestreo estratificado 10 %

1.5. Seleccione un conjunto de datos con valores perdidos y dos métodos de imputación. Estudie el efecto de los métodos de imputación sobre los dos clasificadores.

Para esta prueba, se ha usado el dataset *breast_cancer*, que contiene valores perdidos. Se han usado dos métodos de imputación, siendo el primero *SimpleImputer*, propio de *sklearn*, en el que los valores perdidos se rellenan con valores simples, en este caso, con la media de los valores.

El otro método de imputación usado ha sido el *Iterative Imputer*, propio de sklearn. Este método estima los posibles valores a través de un algoritmo de regresión para estimar los valores perdidos. Los valores de precisión para ambos clasificadores y ambos imputadores se recogen en la tabla 13.

Método Imputación	Precisión árbol	Precisión kNN
Simple Imputer	*89.473 %*	*87.719 %*
Iterative Imputer	*91.228 %*	*87.719 %*

Tabla 13: Precisión de los clasificadores con diferente imputación

1.6. Seleccione un conjunto de datos y un método de discretización. Estudie el efecto del método sobre los dos clasificadores

Como método de discretización se ha usado el *Binarizer*, propio de *sklearn*, el cual binariza el conjunto de datos a partir de un cierto umbral. A modo de experimento, se ha comprobado el valor de la precisión para ambos clasificadores con distintos valores del umbral del binarizador. Los resultados de precisión obtenidos están reflejados en la tabla 14. Además, se puede ver de

forma gráfica en la figura 4. Para estas pruebas, el dataset *breast_ cancer* ha sido usado.

Umbral Binarizador	**Precisión árbol**	**Precisión kNN**
0.1	*92.982 %*	*87.719 %*
0.2	*87.719 %*	*87.719 %*
0.3	*84.210 %*	*84.210*
0.4	*82.456 %*	*85.964 %*
0.5	*77.192 %*	*82.456 %*
0.6	*77.192 %*	*77.192 %*
0.7	*70.175 %*	*70.175 %*
0.8	*66.666 %*	*66.666 %*
0.9	*66.666 %*	*66.666 %*

Tabla 14: Precisión con diferentes umbrales de binarizador

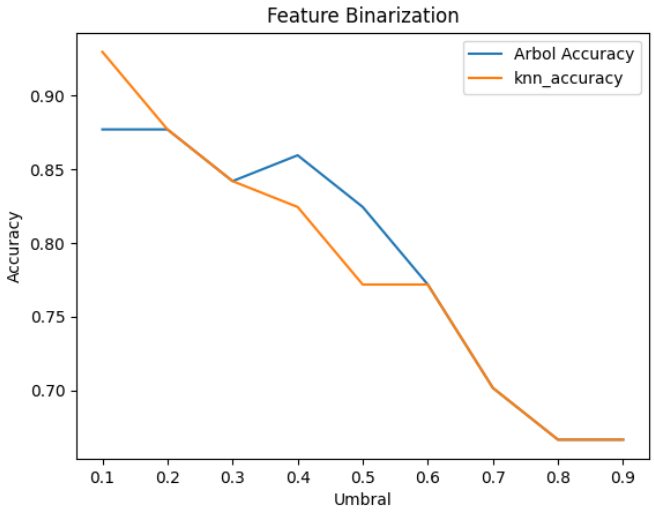

Figura 4: Precisión con tres componentes principales

1.7. Realice un box plots de los datasets. Indique qué información se puede obtener de los gráficos. En al menos un ejemplo construya el mismo gráfico por clases. ¿Se puede obtener información adicional para este caso?

Para esta prueba se han usado como datasets *Iris, CPU, Diabetes, Ionosphere y Segment Challenge*. Como se puede apreciar en las gráficas, dependiendo del dataset se puede apreciar mas o menos información, pues en datasets de grandes valores con instancias muy diferenciadas, ciertos valores o atributos pueden ser gráficamente poco apreciables. Un box plot por clases nos proporciona la distribución de los datos por clases, permitiendonos visualizar gráficamente que valores y atributos son los que mas diferencias las clases.

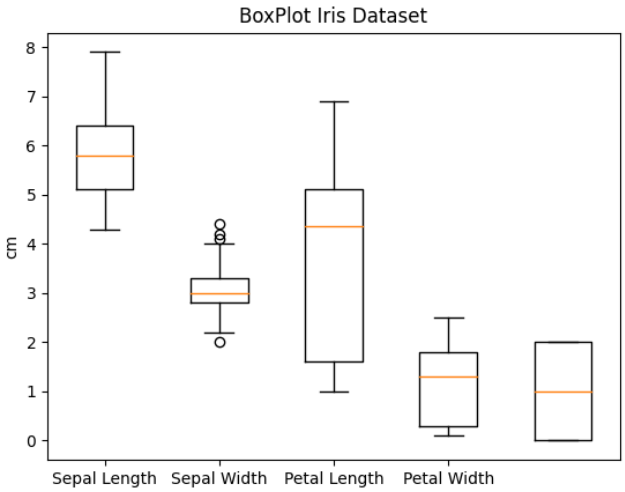

Figura 5: BoxPlot dataset Iris

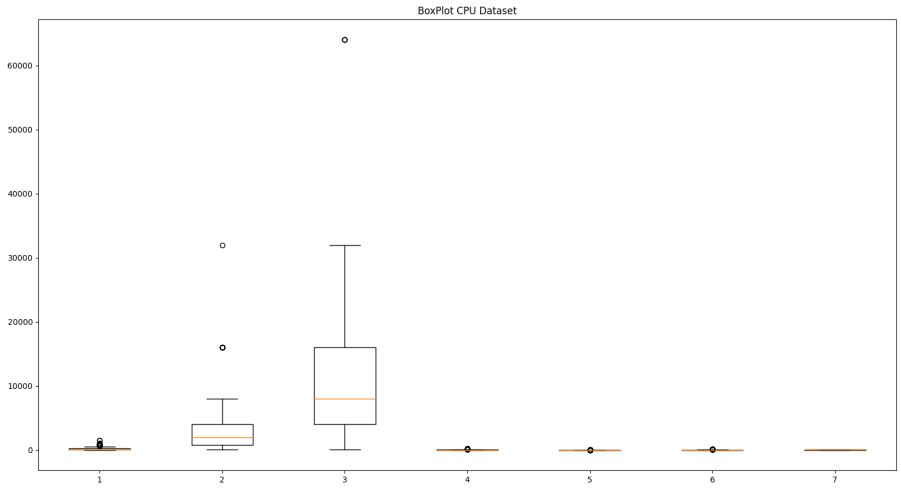

Figura 6: BoxPlot dataset CPU

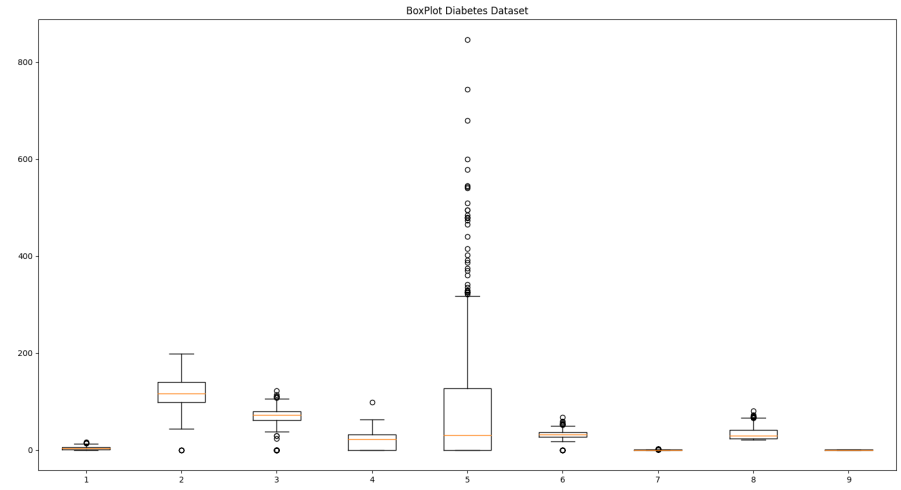

Figura 7: BoxPlot dataset Diabetes

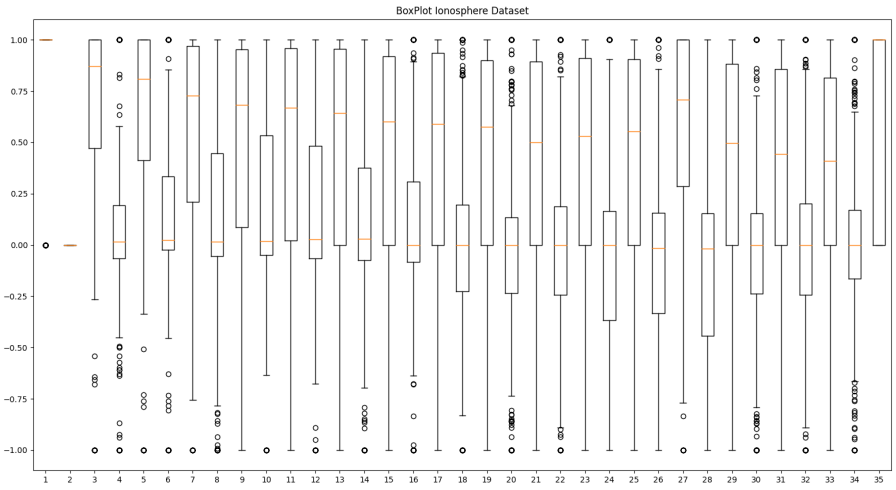

Figura 8: BoxPlot dataset Ionosphere

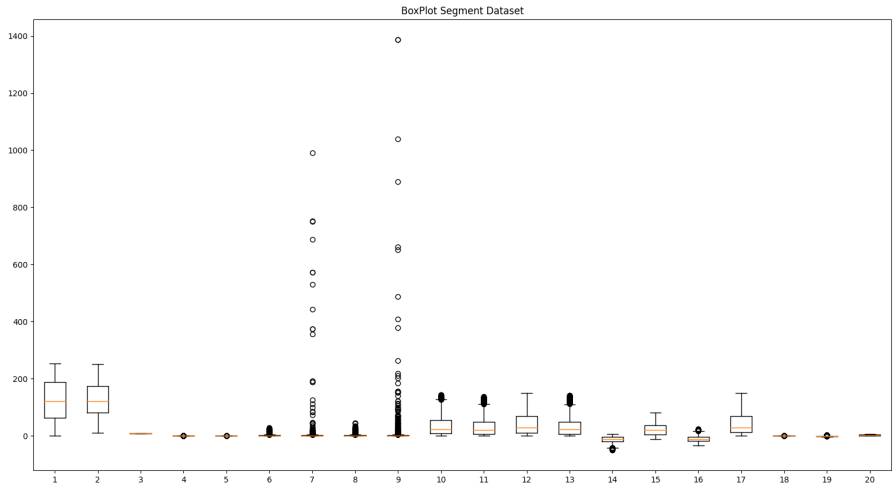

Figura 9: BoxPlot dataset Segment Challenge

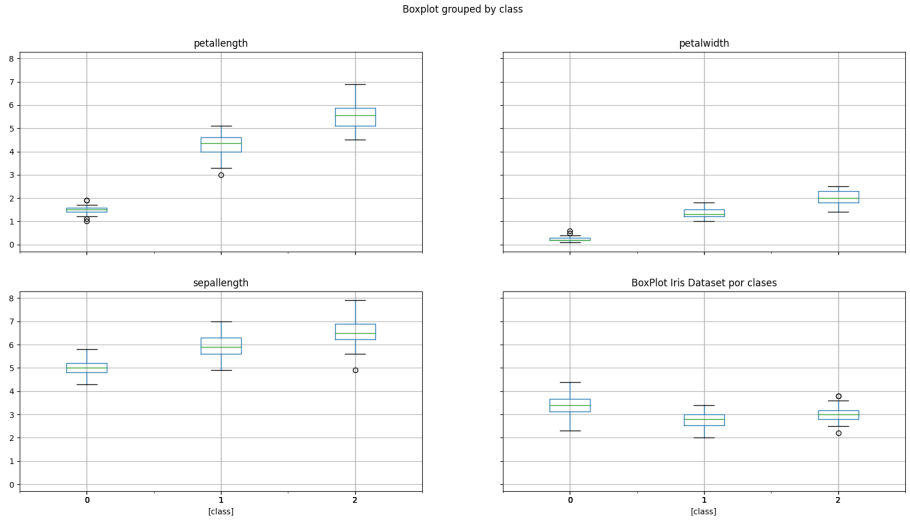

Figura 10: BoxPlot por clases dataset Iris

1.8. Realice un scatter plot matricial de los datasets. Indique qué información se puede obtener de los gráficos

Los scatter plots nos dan información sobre la relación entre dos variables, comparando la relación entre todas las variables. Para esta prueba, solo se mostrarán los scatter plots de los datasets *Iris, CPU y Diabetes*, pues otros datasets más grandes con los que se han trabajado anteriormente, son demasiado grandes y no serían apreciables en una única imagen.

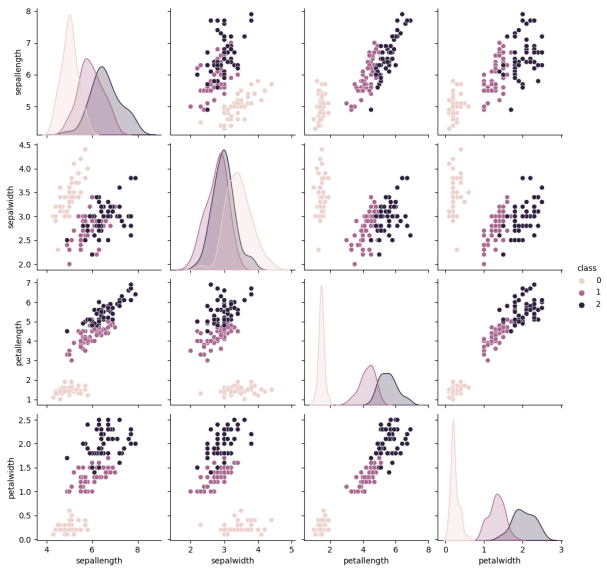

Figura 11: Scatter Plot dataset Iris

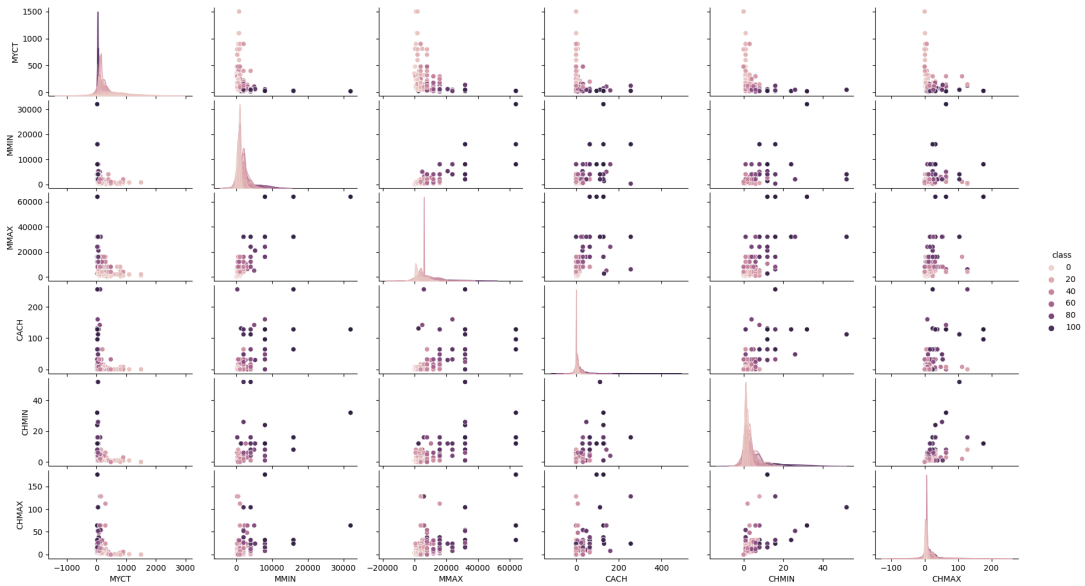

Figura 12: Scatter Plot dataset CPU

Figura 13: Scatter Plot dataset Diabetes

1.9. Represente la matriz de correlación entre las variables usando un mapa de calor e indique la información que se puede extraer del gráfico

Para esta prueba se usarán 5 datasets *Iris, CPU, Diabetes, Ionosphere y Segment Challenge*. De una matriz de correlación se puede obtener la correlación entre dos variables, comparando así todos los pares de variables. Los valores positivos representan que si una variable aumenta, la otra también aumenta de manera consistente, y un valor negativo lo contrario. El mapa de calor sirve para visualizarlo de forma gráfica de forma mas sencilla, usando colores.

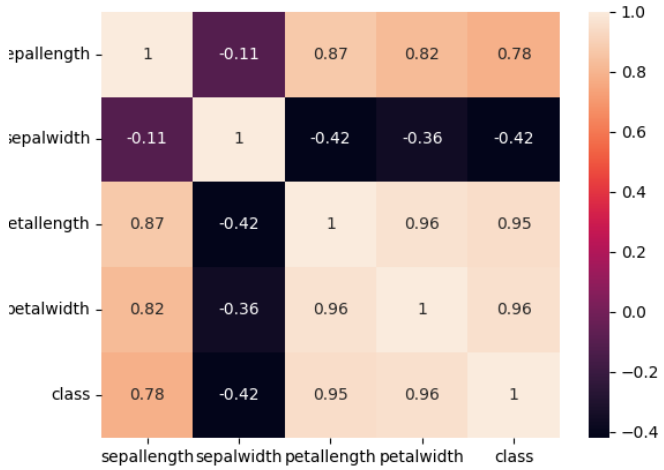

Figura 14: Matriz Correlación dataset Iris

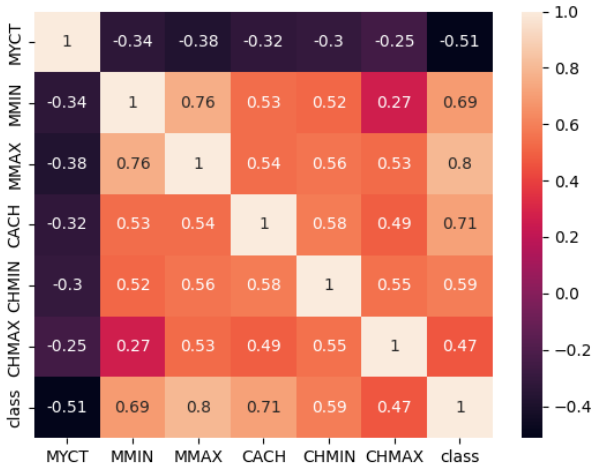

Figura 15: Matriz Correlación dataset CPU

Figura 16: Matriz Correlación dataset Diabetes

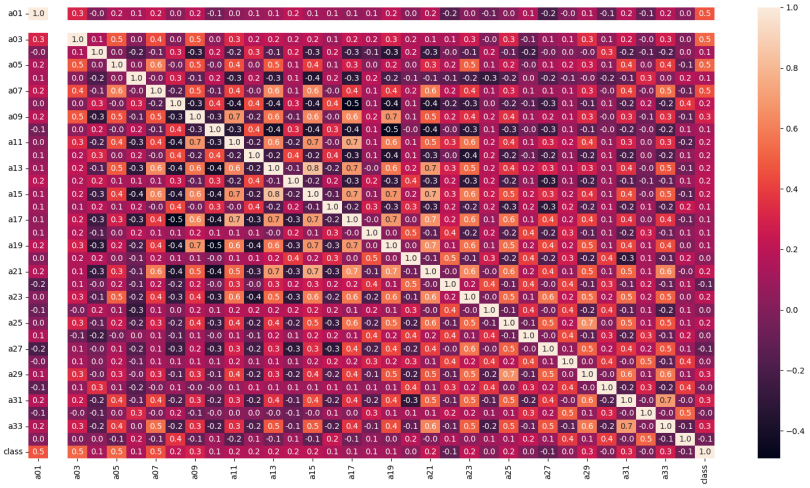

Figura 17: Scatter Plot dataset Ionosphere

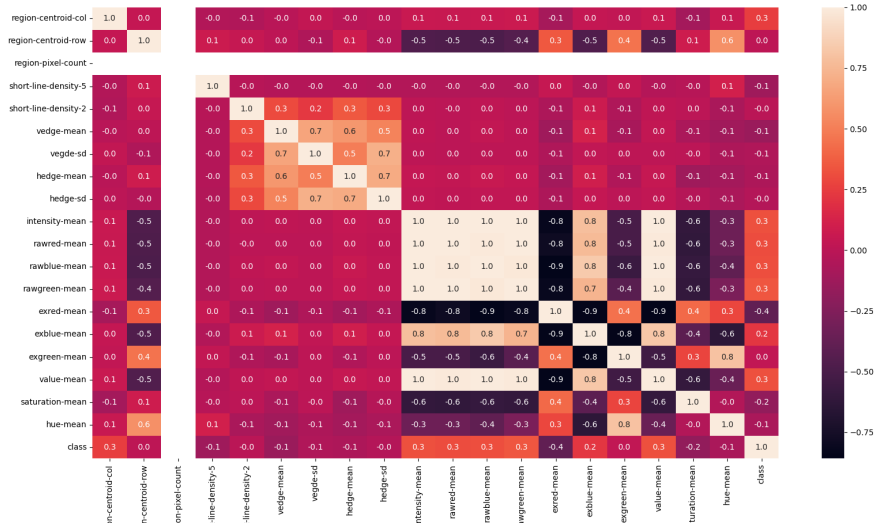

Figura 18: Matriz Correlación dataset Segment Challenge

1.10. Realice el mismo mapa de calor usando la correlación entre las instancias (esta operación es equivalente a realizar la correlación en la matriz de datos traspuesta). Indique qué información se puede obtener de los gráficos

Al ser de nuevo un mapa de calor de correlaciones, obtenemos la misma información que en el caso anterior, pero analizando la correlación entre instancias, en lugar de comparando clases. La información visual es prácticamente inútil, pues no se puede visualizar nada. Para el dataset más pequeño, la matriz de correlación entre instancias se representa en la figura 19.

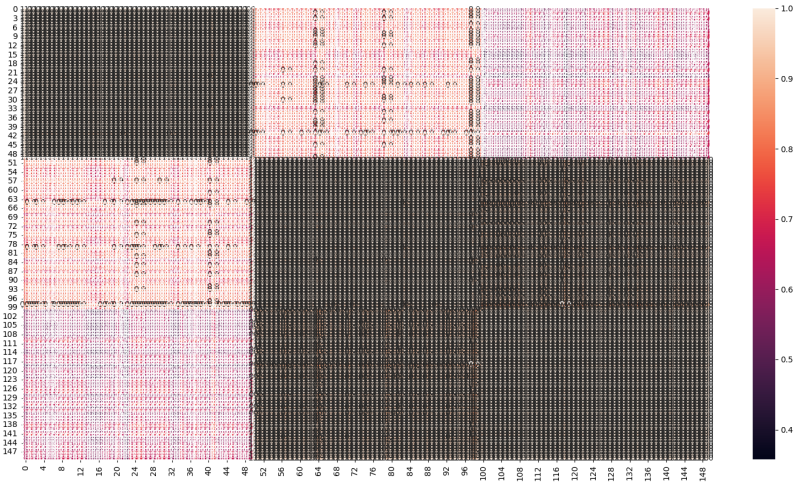

Figura 19: Matriz Correlación instancias dataset Iris