NORMALIZACIÓN, SELECCIÓN DE CARACTERÍSTICAS Y REDUCCIÓN DE LA DIMENSIONALIDAD

VALIDACIÓN CRUZADA

En este proyecto aprenderás a familiarizarte con el proceso de preprocesamiento de datos. Específicamente, se trabajará con un conjunto de datos para generar versiones modificadas del mismo aplicando diferentes técnicas. En particular, se aplicarán:

- Normalización: Su objetivo es ajustar los valores medidos en diferentes escalas a una escala teóricamente común.
- Selección de características y reducción de la dimensionalidad: Su objetivo es conservar algunas características importantes y eliminar el ruido y las características irrelevantes, mejorando así la velocidad de procesamiento de datos, y reduciendo la complejidad computacional y la dificultad del proceso de análisis de datos.

También aprenderás a configurar una validación cruzada, que es un método para validar los resultados de un análisis estadístico, asegurando que no dependan de cómo se dividen los datos entre entrenamiento y test. Hay que usarla siempre que se pueda porque eso nos da una idea del rendimiento global estable de un modelo. Cuanto mayor sea k, más seguros estamos de su rendimiento. Si no se hace, se corre el riesgo de tener una visión sesgada (por azar, es posible que los datos de test hayan sido fáciles de clasificar y entonces el modelo parecerá muy bueno cuando puede que en realidad no lo sea). La validación cruzada consiste en repetir el experimento k veces con distinto conjunto de entrenamiento y test, y después se hace la media. De esta forma, se tiene una idea aproximada del rendimiento general del modelo. Cuanto mayor sea el k, mayor confianza tendremos sobre el rendimiento general del modelo.

Vamos a trabajar con el conjunto de datos Iris. Este conjunto es una de las bases de datos más conocida en la literatura sobre reconocimiento de patrones. Contiene tres clases de 50 instancias cada una, cada una de las cuales se refiere a un tipo de planta iris. Dentro de este conjunto de datos, una clase es linealmente separable de las otras dos; estas últimas no lo son entre sí. Información de atributos:

1. Longitud del sépalo en cm
2. Anchura del sépalo en cm
3. Longitud del pétalo en cm
4. Anchura del pétalo en cm
5. Clase:
   - Iris Setosa
   - Iris Versicolor
   - Iris Virginica

https://archive.ics.uci.edu/ml/datasets/iris
https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data

Se proporciona el tutorial normalizacion_pca.ipynb con comentarios y código para aprender diferentes aspectos de una forma práctica.

Una vez realizado y aprendido el tutorial, se pide desarrollar un código (archivo setup.ipynb) que realice los siguientes pasos:

- Cargar el conjunto de datos y visualizarlo. Llamaremos a este conjunto de datos *conjunto de datos original*.
- Estandarizar el conjunto de datos original a escala unitaria (media = 0 y varianza = 1) y visualizarlo. Llamaremos a este conjunto de datos *conjunto de datos estandarizado.*
- Normalizar el conjunto de datos original a un rango fijo [0, 1] y visualizarlo. Llamaremos a este conjunto de datos *conjunto de datos normalizado.*
- Aplicar un PCA de 0.95 al conjunto de datos original y visualizarlo. Llamaremos a este conjunto de datos *conjunto de datos originalPCA95.*
- Aplicar un PCA de 0.80 al conjunto de datos original y visualizarlo. Llamaremos a este conjunto de datos *conjunto de datos originalPCA80.*
- Aplicar un PCA de 0.95 al conjunto de datos estandarizado y visualizarlo. Llamaremos a este conjunto de datos *conjunto de datos estandarizado PCA95.*
- Aplicar un PCA de 0.80 al conjunto de datos original y visualizarlo. Llamaremos a este conjunto de datos *conjunto de datos estandarizado PCA80.*
- Aplicar un PCA de 0.95 al conjunto de datos normalizado y visualizarlo. Llamaremos a este conjunto de datos *conjunto de datos normalizado PCA95.*
- Aplicar un PCA de 0.80 al conjunto de datos original y visualizarlo. Llamaremos a este conjunto de datos *conjunto de datos normalizado PCA80.*

Una vez hecho esto, vamos a configurar una validación cruzada de k=5 iteraciones o pliegues. Esto produce una validación cruzada 80/20, que suele ser la más usada. Este proceso consiste en lo siguiente:

- Dividir el conjunto de datos en k partes, de forma que la primera parte tendrá las n/k primeras muestras del conjunto de datos, la segunda parte tendrá las muestras de la número (n/k)+1 hasta la 2*(n/k), y así hasta la parte k. Es decir, si, por ejemplo, tu conjunto de datos tiene 50 muestras (n=50) y vas a hacer una validación cruzada de 5 pliegues (k=5), tras la partición tendrás 5 partes, donde la primera parte tendrá las 50/5=10 primeras muestras, la segunda parte tendrá las muestras de la número (50/10)+1=11 hasta la 2*(50/5)=20, y así hasta la quinta parte, que tendrá las muestras de la número 41 hasta la 50. Cada parte debe tener el mismo número de elementos por clase.
- Ahora se genera el conjunto de datos de entrenamiento y test correspondiente a la primera iteración. Para ello, junta las muestras de las partes 2 a k, y esto formará tu conjunto de entrenamiento, que estará en un archivo "training1" (y estará formado por (k-1)*(n/k) muestras), mientras que las muestras de la parte 1 formarán tu conjunto de test (y estará formado por n/k muestras), que estarán en un archivo "test1". Guarda ambos archivos.
- Ahora se genera el conjunto de datos de entrenamiento y test correspondiente a la segunda iteración. Para ello, la segunda parte la utilizarás como conjunto de test y las cuatro restantes las utilizarás como entrenamiento. Es decir, en esta segunda iteración de la validación cruzada, junta las muestras de las partes 1, 3, 4... hasta k, y esto formará tu conjunto de entrenamiento, que estará en un archivo "training2", mientras que las muestras de la parte "2" formarán tu conjunto de test, que estarán en un archivo "test2". Guarda ambos archivos.
- …
- Hacer lo mismo hasta la iteración k.

Para ello, se pide desarrollar un código (archivo setup.ipynb) que realice esos pasos para cada uno de los conjuntos de datos considerados: original, estandarizado, normalizado, y las distintas versiones de PCA. Usa sufijos diferentes para los nombres de los archivos cada uno de ellos (por ejemplo, el archivo training3_norm_PCA95 contendrá las muestras de entrenamiento del conjunto de datos normalizado PCA95 correspondientes a la tercera iteración de la validación cruzada). Usa formato CSV para guardar los archivos.

Finalmente, se tiene que redactar un informe con los resultados obtenidos y las conclusiones que se pueden extraer. El informe tiene que estar escrito en Latex usando la plantilla LNCS.

Se tiene que entregar un único archivo comprimido que contenga los siguientes archivos:

- El informe en formato pdf generado por el proyecto Latex.
- Archivo comprimido (.rar o .zip) con todos los archivos del proyecto Latex.
- El código (en formato ipynb) que se ha desarrollado.
- Un video (en formato mp4) con la ejecución del código.
- Declaración explícita en la que se asuma la originalidad del trabajo entregado.