



Escuela Superior de Cómputo
Profesora: Consuelo Varinia García Mendoza
Práctica 6. K-NN



1. Para esta práctica se utilizará los dataset *iris.csv* y *emails.csv*
 - En el dataset *iris.csv*
 - Las primeras 4 columnas son las características de las instancias
 - La última columna es la clase
 - En el dataset *emails.csv*
 - La primera columna indica el id del correo
 - La última columna indica si el correo es spam o no
 - El resto de las columnas (3,000) son las palabras más comunes en todos los correos
2. Para cada dataset
 - Mezcla los datos (`random_state=0`) y crea un conjunto de entrenamiento del 70% y 30% de prueba
 - Con el 70% de entrenamiento genera conjuntos de validación con el método de validación cruzada para $k = 3$. Genera la información de la Tabla 1 utilizando las bibliotecas:
 - `sklearn.neighbors.KNeighborsClassifier`
 - `sklearn.metrics.accuracy_score`
 - Selecciona las configuraciones que logran los mejores accuracy promedio para las pruebas finales y llena la Tabla 2. Genera la matriz de confusión y el reporte de clasificación utilizando las bibliotecas:
 - `sklearn.metrics.classification_report`
 - `sklearn.metrics.ConfusionMatrixDisplay`
3. La salida del programa será la información de las tablas 1 y 2. De las pruebas finales de K-NN y Bayes se generarán las matrices de confusión y los reportes de clasificación

Nota: A la Tabla 2 se agregarán los resultados de las pruebas finales de la Práctica de Bayes

Tabla 1. Validación cruzada con $k = 3$ para 1-NN y 10-NN

Dataset	Vecinos	Pesos	Pliegue	Accuracy
Iris.csv	1	-----	1	
			2	
			3	
			Promedio	
	10	uniforme	1	
			2	
			3	
			Promedio	
	10	distancia	1	
			2	
			3	
			Promedio	
emails.csv	1	-----	1	
			2	
			3	
			Promedio	
	10	uniforme	1	
			2	
			3	
			Promedio	
	10	distancia	1	
			2	
			3	
			Promedio	

Tabla 2. Resultados de pruebas finales

Dataset	Clasificador	Vecinos	Pesos	Distribución	Accuracy
iris.csv	Naïve Bayes	-----	-----		
	K-NN			-----	
emails.csv	Naïve Bayes	-----	-----		
	K-NN			-----	