Tarea 2 EL4106 - Semestre Otoño 2021

Profesor: Javier Ruiz del Solar
Auxiliar: Patricio Loncomilla

Fecha enunciado: Miércoles 14 de Abril de 2021
Plazo entrega tarea: Domingo 25 de Abril de 2021

El objetivo de esta tarea es implementar un clasificador de hadrones v/s rayos gamma usando *Support Vector Machine* (SVM). Se usará el conjunto de datos MAGIC Gamma Telescope Data Set, el cual corresponde a un conjunto de simulaciones de *air showers* generados por rayos gamma primarios v/s hadrones. El conjunto de datos forma parte del *UC Irvine Machine Learning Repository*. El conjunto de datos contiene 10 características, además de una clase, la cual puede ser h (hadrón) o g (no-hadrón). Hay 12.332 ejemplos de rayos gamma y 6.688 ejemplos de hadrones. El conjunto de datos contiene 11 columnas: las primeras 10 son las características y la última es la clase. Los datos están disponibles tanto en u-cursos como en la página del repositorio.

Se pide utilizar la metodología de clasificación estadística SVM para entrenar y validar un clasificador de hadrones. El trabajo por realizar incluye analizar detalladamente el efecto en el rendimiento del clasificador mediante la utilización de distintos tipos de *kernel*s. En esta tarea tendrán que entrenar y calibrar 4 clasificadores SVM binarios, que utilicen distintos tipos de Kernels/grados en caso polinomial.

Los kernels del SVM, incluyendo sus hiperparámetros, son:

- Lineal: $K(x_i, x_j) = x_i^T x_j$
- Polinomial: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d$
- RBF: $K(x_i, x_j) = \exp\left(-\gamma \left\| x_i^T x_j \right\|^2\right)$

Se pide:

1) Implementar un código que lea el dataset, usando pandas.

2) Re-muestrear el dataset usando pandas, de modo que contenga 3.000 ejemplos de la clase positiva, y 3.000 ejemplos de la clase negativa.

3) Dividir la base de datos en 3 conjuntos representativos: entrenamiento (60%), validación (20%) y prueba (20%). Compruebe la representatividad de éstos, verificando si la proporción de cada clase se mantiene cercana a la proporción del conjunto completo.

4) Entrene un StandardScaler usando las características del conjunto de entrenamiento para poder normalizar las características. Luego, aplíquelo a las características del conjunto de entrenamiento, validación y prueba.

5) Entrenar un clasificador SVM lineal que permita discriminar hadrones de no-hadrones. Para obtener un buen clasificador, se debe usar una grilla para buscar los mejores hiper parámetros para el clasificador. Se debe usar la función GridSearchCV( ) con 5 *folds*, considerando distintos valores del parámetro C (5 valores distintos). El clasificador base SVM a usar se puede construir de la siguiente manera: svm.SVC(kernel='linear', probability=False).

6) Evaluar sobre el conjunto de validación y generar la matriz de confusión, tanto en su versión normalizada como no normalizada. Se recomienda usar metrics.confusión_matrix( )

7) Generar una curva ROC que muestre el desempeño del clasificador y calcular el área bajo la curva. Se debe usar las funciones decision_function( ), metrics.roc_curve( ) y metrics.auc( )

8) Generar una curva precisión-recall y calcular el average precision. Se debe usar las funciones decision_function( ), metrics.precision_recall_curve( ) y metrics.average_precision_score( )

9) Repetir los pasos (5), (6), (7), (8) para el caso de un SVM con kernel polinomial. Usar dos grados distintos del polinomio y 4 valores de C y 4 valores de gamma (parámetro del kernel) para la grilla de búsqueda (de parámetros).

10) Repetir los pasos (5), (6), (7), (8) para el caso de un SVM con kernel RBF. Usar 5 valores de C y 4 valores de gamma (parámetro del kernel) para la grilla de búsqueda (de parámetros)..

11) Evaluar el mejor clasificador obtenido sobre el conjunto de prueba, reportando las métricas indicadas en (6), (7) y (8). Considerar los casos de SVM lineal, SVM con kernel polinomial (2 grados distintos de polinomio) y SVM con kernel RBF.

Los informes y códigos deben ser subidos a u-cursos a más tardar el día domingo 25 de Abril a las 23:59. Incluir un breve archivo de texto explicando cómo se utiliza su programa. Las tareas atrasadas serán penalizadas con un punto base por cada día de atraso.

Nota: Dado que el entrenamiento del SVM con grillas puede tomar varios minutos, se puede usar un número menor de muestras mientras se verifica el correcto funcionamiento del código.

Importante: La evaluación de la tarea considerará el correcto funcionamiento del programa, la inclusión de los resultados de los pasos pedidos en el informe, la calidad de los experimentos realizados y de su análisis, la inclusión de las partes importantes del código en el informe, así como la forma, prolijidad y calidad del mismo. Se debe usar el formato indicado en material docente.

Nota: El informe, en formato pdf, debe ser subido a turnitin