CENTRO DE E-LEARNING UTN BA
Medrano 951 CABA, Buenos Aires Argentina
(1179) // tel +54 11 7078 – 8073 / fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Curso
Machine Learning con Python

Trabajo Práctico Final

Objetivos

Aplicar conocimientos presentados en el curso

Consigna

Ejercicio 1 (50 puntos)

Para la realización de este ejercicio se utilizará el dataset "Wine recognition dataset" de Scikit-Learn.

Desarrolle un Jupyter notebook que

1) (5 ptos) Cargue los datos y normalice dichos datos restando la media y dividiendo por la desviación estándar
2) (15 ptos) Aplique el algoritmo PCA a los datos normalizados para reducir su dimensionalidad a 2. Realice un gráfico de dispersión de los datos obtenidos luego de aplicar PCA, utilizando marcadores de diferentes colores para las diferentes clases
3) (5 ptos) Utilice el comando train_test_split para separar el dataset obtenido en el apartado anterior, en conjuntos de entrenamiento y prueba. La fracción de datos de prueba debe estar entre 20% y 30%.
4) (10 ptos) Sin utilizar las etiquetas del dataset, aplique el algoritmo KMeans a los datos de entrenamiento. Considere valores de K (número de clusters) de

2, 3, 4 y 5, y utilice la función adjusted_rand_score para evaluar el desempeño obtenido sobre los datos de prueba, para los diferentes valores de K

5) (10 ptos) Sin utilizar las etiquetas del dataset, aplique el algoritmo de propagación de afinidad sobre los datos de entrenamiento. Utilice la función adjusted_rand_score para evaluar el desempeño obtenido sobre los datos de prueba. La precisión obtenida debe ser mayor o igual al 90%.
6) (5 ptos) Comente sobre los resultados obtenidos

Ejercicio 2 (50 puntos)

Para la realización de este ejercicio se utilizará el dataset Labeled Faces in the Wild (LFW), disponible en scikit-learn. Específicamente, se utilizarán las 7 categorías más representativas, que incluyen un total de 1288 datos que corresponden a los personajes para los que el dataset tiene al menos 70 imágenes. El objetivo es obtener diferentes modelos de clasificación para este conjunto de datos. Pueden obtener más información del dataset en

https://scikit-learn.org/0.19/datasets/labeled_faces.html

En el Jupyter Notebook suministrado, que realiza una visualización simple del conjunto de datos, agregue las instrucciones necesarias para

1) (10 ptos) Aplicar el algoritmo PCA a los datos. En este caso, es importante tomar en consideración que:
   - Antes de aplicar PCA, las imágenes deben convertirse a vectores 1D
   - El número de componentes principales debe seleccionarse de manera que el porcentaje de varianza explicada sea mayor al 80%. Este porcentaje puede determinarse sumando las varianzas explicadas por las componentes principales, que pueden obtenerse aplicando el método explained_variance_ratio_ al objeto pca

   El dataset obtenido aplicando el algoritmo PCA es el que va a utilizar para determinar los modelos

2) (5 ptos) Separar el dataset en conjuntos de entrenamiento y prueba utilizando el comando train_test_split. La fracción de datos de prueba debe estar entre 20% y 30%
3) (10 ptos) Determinar un modelo de clasificación basado en regresión logística, utilizando el conjunto de entrenamiento. Evalúe el desempeño del modelo sobre el conjunto de prueba, determinando la precisión (accuracy) y la matriz de confusión
4) (10 ptos) Determinar un modelo de clasificación basado en KNN, utilizando el conjunto de entrenamiento. Evalúe el desempeño del modelo sobre el conjunto de prueba, determinando la precisión (accuracy) y la matriz de confusión
5) (10 ptos) Determinar un modelo de clasificación basado en SVM, utilizando el conjunto de entrenamiento. Evalúe el desempeño del modelo sobre el conjunto de prueba, determinando la precisión (accuracy) y la matriz de confusión

   Al menos dos de los modelos deben tener una precisión (accuracy) por encima del 80% sobre el conjunto de prueba. Las etiquetas de las matrices de confusión deben ser los nombres de los personajes
6) (5 ptos) Comentar sobre los resultados obtenidos

Formato de presentación:

- El trabajo puede realizarse individualmente o en grupos de 2 participantes
- Los participantes deben entregar un archivo comprimido que contenga los Jupyter notebooks desarrollados. El nombre del archivo comprimido debe tener el formato TPF_Apellidos. Por ejemplo, TPF_Canelon o TPF_Barreto_Canelon. El nombre de los Jupyter notebooks debe tener el formato TPF_Apellidos_Ejer#.ipynb, donde # debe reemplazarse por el número de ejercicio. Por ejemplo, TPF_Canelon_Ejer1.ipynb o TPF_Barreto_Canelon_Ejer1.ipynb
- El archivo comprimido debe generarse con la aplicación 7-zip, que puede descargarse gratuitamente desde https://www.7-zip.org/.

Fecha límite de entrega:

Nominal: 23/10/24 - 23:59 hrs

Recuperatorio: 30/10/24 - 23:59 hrs

Las fechas de entrega son inapelables, ya que están configuradas automáticamente en el Campus. La plataforma no permitirá que los participantes entreguen fuera de la fecha/hora indicada. Si no pueden entregarlo en la primera fecha, podrán hacerlo en el recuperatorio.

Criterios de evaluación

- La calificación total del trabajo estará en función del número de consignas realizadas correctamente. Si alguna consigna no funciona de manera correcta o genera un error en el Jupyter notebook, se restarán puntos del total correspondiente a esa consigna

Bibliografía utilizada y sugerida

scikit-learn. Documentación oficial.
Disponible desde: URL: https://scikit-learn.org/stable/

NumPy. Documentación oficial.
Disponible desde: URL: https://numpy.org/

Matplotlib. Documentación oficial.
Disponible desde: URL: https://matplotlib.org/