Inteligencia Artificial

1er Cuatrimestre 2024

Trabajo Práctico:

Aprendizaje Automático II

Facultad de Ciencias de la Administración

Fecha límite de entrega: Viernes 07/06/2024 a las 23:59.

Condiciones de entrega: el trabajo práctico deberá ser realizado en forma individual. Se deberá subir en la sección del Campus Virtual correspondiente un único archivo comprimido con formato `zip, rar, tar.gz` u otro. El mismo contendrá los archivos `.py` separados por cada punto/consigna. En cada archivo `.py` se debe presentar el código solución a la consigna y algunas líneas adicionales de código que sirvan para testear la solución presentada. Además, pueden incluir un `.pdf` que presente las respuestas, suposiciones y aclaraciones pertinentes de cada punto.

1. **Dataset: autenticacion_moneda.csv**
   El dataset cuenta con datos obtenidos a partir de imágenes de diferentes monedas tanto genuinas como falsas. Tenemos información sobre la varianza, asimetría, curtosis, entropía de la imagen y la clase a la que pertenece (genuinas o falsas).

   a) Obtener la curva **Elbow** para determinar la cantidad de arboles. Adjuntar imagen.

   b) En un mismo código fuente .py hacer un clasificador por **Regresión Logística** y un clasificador por **Random Forest** conforme a lo que indico en el punto anterior. Utilizar validación cruzada **K-fold** con **5** folds para entrenamiento y testeo.

   c) Calcular y comparar las métricas score, accuracy, precision, recall y F1 para ambos clasificadores.

2. **Dataset: StudentsPerformance**
   El dataset contiene datos que abordan el rendimiento de un grupo de alumnos de nivel secundario. Los atributos de los datos incluyen las calificaciones de los alumnos, características demográficas, sociales y relacionadas con el centro escolar, y se recogieron mediante informes escolares y cuestionarios. Se proporcionan dos conjuntos de datos relativos al rendimiento en dos asignaturas distintas: Matemáticas (**student-mat.csv**) y Lengua portuguesa (**student-por.csv**). Queremos clasificar a los estudiantes en tres categorías, "bueno", "regular" y "malo", según su rendimiento en el examen final (*final_score*).

   Pre-procesamiento

   a) Generar un único dataset. Utilizar `concat` de la librería Pandas.

   b) Generar una nueva variable categórica *final_grade* a partir de los valores de *final_score* que asigne "bueno" a valores entre 15-20, "regular" a valores entre 10-14 y "malo" para los valores entre 0-9.

   c) Graficar el mapa de calor para ver relación entre variables.

   d) Analizar las características para ver si tienen una influencia significativa en el rendimiento final de los estudiantes.

   Modelos

   a) Utilizar `labelEncoder` para la variable *final_grade*

   b) De ser necesario aplicar la función `get_dummies`.

   c) Dividir el dataset en entrenamiento y testeo, dejando el 30 % de los datos para testeo.

   d) Crear un modelo Random Forest para predecir el rendimiento académico de los estudiantes en función de diferentes características.

   e) Crear un modelo SVM.

   f) Comparar el rendimiento de los modelos basándose en la precisión.

3. **Dataset: teleCust.csv**
   El dataset contiene la base de clientes de un proveedor de telecomunicaciones segmentada por servicio, categorizando a los clientes en cuatro grupos. Si los datos demográficos se pueden usar para predecir

la pertenencia de grupo, la compañía podría personalizar las ofertas. Este ejemplo hace foco en datos demográficos, sean región, edad, estado civil, para predecir patrones de uso. El campo objetivo llamado **custcat**, tiene cuatro valores posibles que corresponden a los cuatro grupos de clientes: 1-Servicio Básico 2- E-Servicio 3- Servicio Plus 4- Servicio Total Utilizar un clasificador K vecinos más cercanos para predecir el grupo de los casos desconocidos o de testeo.

a) Construir un programa Python para resolver el ejercicio. De ser necesario utilizar `sklearn.preprocessing.` para preparar los datos.

b) Separar el dataset original en dos porciones correspondiente a entrenamiento y testeo. Tener en cuenta que debo usar los mismos datos para todos los clasificadores que deba hacer (ver Incs (d)).

c) Construir el clasificador **KNN** con **7** vecinos más cercanos.

d) Obtener una versión alternativa del clasificador que difieran en la forma de la importancia relativa de los vecinos conforme a la distancia.

e) Utilizar acurracy como métrica de comparación entre ambos clasificadores. ¿Cuál es más preciso?

4. **Dataset: Wine** Este dataset contiene información sobre las características químicas de diferentes vinos provenientes de tres diferentes cultivares en la región italiana de Piamonte. El objetivo es clasificar los vinos en tres categorías distintas basadas en estas características químicas.

a) Realizar gráficos que permitan visualizar la relación entre las variables del dataset.

b) Obtener la curva **Elbow** para determinar la cantidad de clusters. Adjuntar imagen.

c) Construir el modelo **K-means** conforme a lo que indico en el punto anterior.

Referencias

[1] *https://scikit-learn.org/*

[2] *https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html*

[3] *https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html*

[4] D. Poole, A. Mackworth., *Artificial Intelligence. Foundations of Computational Agents. 2010. Cambridge University Press.*

[5] S. Russell, P. Norvig., *Artificial Intelligence: A Modern Approach. 3rd Edition. 2010. Prentice Hall.*