Facultad de Ciencias Exactas, Ingeniería y Agrimensura

Tecnicatura en Inteligencia Artificial

Aprendizaje Automático 1

Trabajo Práctico: Predicción de lluvia en Australia.

Objetivos

Familiarizarse con la biblioteca scikit-learn y las herramientas que brinda para el pre-procesamiento de datos, la implementación de modelos de clasificación y la evaluación de métricas, con TensorFlow para el entrenamiento de redes neuronales y con streamlit y Docker para la puesta en producción del modelo seleccionado como el más adecuado, entre otras.

Dataset

El dataset se llama weatherAUS.csv y contiene información climática de Australia de los últimos años, incluyendo si para el día siguiente llovió o no en la columna '**RainTomorrow**'. El objetivo es la predicción de esta variable en función del resto de las características.

Tiene una columna 'Location' que indica la ciudad y el objetivo es predecir la condición de lluvia en una cierta cantidad de ciudades. Pueden considerarse como una única ubicación. **Descartar los datos del resto de las ciudades no elegidas.**

Para elegir las ciudades, obtener los valores únicos de la columna Location del DataFrame y ejecutar una línea de código donde se elijan de manera aleatoria 10 ciudades de esos valores. Quedarse con estas ciudades para todo el trabajo práctico.

Para todos los ítems, incorporar una cantidad de texto adecuado en forma de comentarios/conclusiones, ya sea para la comprensión del código (usualmente una línea de comentario por cada celda) como para explicar las decisiones tomadas a lo largo del trabajo (por ejemplo, la justificación de la imputación de valores faltantes, la elección de las métricas adecuadas, entre otros). Mantener la coherencia con los comentarios.

Consignas

1. Realizar un análisis descriptivo, que ayude a la comprensión del problema, de cada una de las variables involucradas, detallando: características, comportamiento y rango de variación. **¿Qué es cada variable?**
   Debe incluir (estos ítems no están necesariamente en orden):
   - Análisis y decisión sobre datos faltantes.
   - Visualización de datos (por ejemplo histogramas, scatterplots entre variables, diagramas de caja)
   - ¿Está *balanceado* el dataset? ¿Por qué cree que hacemos esta pregunta?
   - Codificación de variables categóricas (si se van a utilizar para predicción).
   - Matriz de correlación de variables.
   - Estandarización/escalado de datos.
   - Validación cruzada train - test. Realizar una división del conjunto de datos en conjuntos de entrenamiento y prueba (y si se quiere, se puede incluir validación, que luego será útil) **en el MOMENTO donde lo crean adecuado.**

2. Implementar la solución del problema de clasificación con regresión logística.
    - Obtener las **métricas adecuadas** (accuracy, precision, recall, F1 score, entre otras, ¡investiguen adicionales!). Graficar matrices de confusión para cada modelo. Analizar “falsos negativos” y “falsos positivos”, ¿qué significa cada uno?
    - Trazar curvas ROC para cada modelo. Comenten cuáles serían los umbrales adecuados a utilizar; ¿cómo podrían calcular el mejor umbral? ¿Es 0.5 el mejor?
    - ¿Creen que han conseguido un buen fitting?

3. Implementar un modelo base.
    - Repetir los pasos del ítem 2.

4. Optimizar la selección de hiperparámetros.
    - Probar validación cruzada k-folds, si corresponde.
    - Utilizar grid search, random search u optuna. **Justificar su uso. Justificar los hiperparámetros que se están optimizando.**

5. Implementar explicabilidad de los modelos.
    - Utilizar SHAP o similar. Implementar al menos dos gráficas a nivel local y dos gráficas a nivel global. **¡Escribir lo que se observa!**
    - ¿Cuáles son las variables más importantes? ¿Cuáles son las menos?

6. Implementar un modelo de AutoML con scikit-learn. ¿Qué observa?

7. Implementar las soluciones con una red neuronal.
    - Obtener las métricas adecuadas. ¿Consigue un mejor fitting que con la regresión logística?
    - Repetir los pasos 4 y 5 para las redes neuronales. ¿Qué diferencias observan con los modelos de regresión logística?

8. Comparación de modelos.
    - Incluyan en su análisis una comparación de todos los modelos de clasificación, ¿cuál es el mejor? **Escoger UNA métrica adecuada para poder compararlos**.

9. **MLOps: puesta en producción.**

    Para realizar el deployment del trabajo deben utilizar Docker. Dentro del repo del TP, ademas del notebook con el desarrollo, deben agregar una carpeta "docker" que contenga:

- script de inferencia (debe llamarse **inferencia.py**)
- requeriments.txt (unicamente incluir librerias necesarias para la inferencia)
- binarios con pipeline o modelo, imputers y scalers (.pkl, .joblib, pueden escoger el formato de serializacion que consideren adecuado)
- Dockerfile
- readme.md con instrucciones para poder construir la imagen de docker y ejecutar el container (docker build y docker run)

- todo lo que consideren necesario para realizar la inferencia, **no incluir archivos/carpetas innecesarios.**

10. Escribir una conclusión del trabajo práctico.

11. **Preparar una defensa del trabajo práctico**: la defensa consiste en preguntas hechas por el cuerpo docente que pueden ser: explicar una parte del código, explicar alguno de los métodos utilizados, preguntas de índole teórica, preguntas de índole práctica. Son tanto grupales como individuales. Abarcan el notebook y la aplicación de MLOps. **Tener en cuenta la duración sugerida por el cuerpo docente para las defensas.**

Entregas parciales y condiciones
Las entregas parciales se realizan mediante GitHub (suben el código y nos envían el link del repositorio **mediante este formulario: https://forms.gle/CW85afc7NQUB4JFe9)**

El repositorio debe llamarse “AA1-TUIA-Apellido1-Apellido2-Apellido3” **sin excepciones. No crear carpetas dentro que contengan alguno de los entregables.**

Respetar los siguientes nombres:

Notebook de trabajo: TP-clasificacion-AA1.ipynb

Se deben hacer commits con el asunto “Entrega hasta ítem x” (se pueden hacer commits parciales -de hecho se recomienda-).

1. Hasta el 25/10: ítem 1, 2 y 3.
2. Hasta el 15/11: ítem 4, 5 y 6.
3. Hasta el 29/11: ítems 7, 8, 9 y 10.

Cada entrega puede demorarse hasta dos días después de la fecha pactada, **con disminución de la nota final del trabajo práctico.**

No se aceptan entregas finales con fecha posterior al 01/12/24. En caso de no tener todos los ítems entregados para esta fecha, la condición es automáticamente de libre.

La defensa del TP se hará desde el lunes 02/12/24, en horarios de clase, separados por turnos que la cátedra asignará según el orden en el que se fue entregando. En caso de detectar errores o una presentación en la que falten conocimientos sobre el trabajo realizado que se consideren lo suficientemente graves, se pactará una fecha para una segunda defensa (donde deberán estar realizadas las correcciones y más acertada la presentación). *En caso de reprobar en esta segunda instancia de defensa, la condición es de libre*.

Los ítems se pueden ir perfeccionando a medida que se va avanzando, aunque no se tiene la misma consideración para la nota si fue editado luego de la fecha de entrega. **Pero sí importa!**