TALLER 4

Objetivo

Evaluar las capacidades del estudiante para desplegar modelos analíticos mediante una API REST como mecanismo para disponibilizar/integrar un proceso básico de inferencia dentro de una aplicación web o móvil.

Planteamiento del problema

El área de fidelización de una empresa de telecomunicaciones tiene como uno de sus objetivos disminuir la tasa de churn (abandono) por parte de sus clientes. Por esta razón, desea construir un modelo de machine learning que permita predecir si un cliente es propenso abandonar los servicios que actualmente tiene contratados con la empresa. Una vez construido, este modelo debe ser disponibilizado a través de una API REST que pueda ser consumida por la plataforma que es usada por los asesores de call center usando dicha predicción para ofrecer nuevos productos o servicios para los clientes más propensos al abandono.

A continuación, se describe todo el conjunto de datos recolectado:

Field	Description
customerID	Customer ID
gender	Whether the customer is a male or a female
SeniorCitizen	Whether the customer is a senior citizen or not (1, 0)
Partner	Whether the customer has a partner or not (Yes, No)
Dependents	Whether the customer has dependents or not (Yes, No)
tenure	Number of months the customer has stayed with the company
PhoneService	Whether the customer has a phone service or not (Yes, No)
MultipleLines	Whether the customer has multiple lines or not (Yes, No, No phone service)
InternetService	Customer’s internet service provider (DSL, Fiber optic, No)
OnlineSecurity	Whether the customer has online security or not (Yes, No, No internet service)

OnlineBackup	Whether the customer has online backup or not (Yes, No, No internet service)
DeviceProtection	Whether the customer has device protection or not (Yes, No, No internet service)
TechSupport	Whether the customer has tech support or not (Yes, No, No internet service)
StreamingTV	Whether the customer has streaming TV or not (Yes, No, No internet service)
StreamingMovies	Whether the customer has streaming movies or not (Yes, No, No internet service)
Contract	The contract term of the customer (Month-to-month, One year, Two year)
PaperlessBilling	Whether the customer has paperless billing or not (Yes, No)
PaymentMethod	The customer's payment method (Electronic check, Mailed check, Bank transfer (automatic), Credit card (automatic))
MonthlyCharges	The amount charged to the customer monthly
TotalCharges	The total amount charged to the customer
Churn	Whether the customer churned or not (Yes or No)

Actividades

A continuación, se describen más a fondo los hitos mínimos esperados por la empresa:

Integración del proceso de transformación de datos, entrenamiento y predicción (25 pts)

Utilizando la función Pipeline de Scikit-Learn, encapsule los procesos secuenciales de transformación, entrenamiento y predicción requeridos para dar solución al problema de predicción del churn a partir de los datos proporcionados en el archivo *DataSet_Entrenamiento_v1.txt*.

Tenga en cuenta que a partir de los datos originales el pipeline debe estar en capacidad de realizar todas las transformaciones de datos necesarias para realizar las labores de entrenamiento y predicción. Es decir, posterior a las fases de exploración y análisis de calidad de los datos, no se deben implementar procesos de transformación mediante el uso de Pandas.

Búsqueda automática del mejor modelo (25 pts)

Realice una búsqueda en grilla con validación cruzada con al menos 3 algoritmos y sus respectivos conjuntos de hiper-parámetros para encontrar el mejor modelo de clasificación de churn optimizando la métrica ROC AUC. Tenga en cuenta lo siguiente:

- Este proceso debe realizarse solamente sobre el archivo *DataSet_Entrenamiento_v1.txt*.
- Recuerde que dentro de la búsqueda en grilla también puede hacer selección de modelo para diferentes alternativas de transformación de datos. Si el espacio de búsqueda se vuelve muy amplio, puede optar por una búsqueda aleatoria.
- Detalle los aspectos más relevantes del mejor modelo encontrado y sus resultados de diferentes métricas para los datasets de entrenamiento, validación y prueba
- Responda la pregunta: **¿Se evidencian problemas de *overfitting* o *underfitting*?**
- Investigue y describa por qué es preferible la métrica ROC AUC sobre las demás métricas como F1, Precision o Recall.
- Exporte el mejor modelo obtenido a un archivo tipo pickle o joblib.

Despliegue mediante API REST (25 pts)

Construya una API REST utilizando el framework de su preferencia la cual cargue el modelo previamente exportado y la cual implemente los siguientes endpoints:

- **Re-entrenamiento:** Pasando nuevos datos de entrenamiento (*DataSet_Entrenamiento_v2.txt*), este endpoint debe estar en capacidad de re-entrenar el modelo utilizando el mismo algoritmo e hiper-parámetros del modelo original. Estos nuevos datos deben ser recibidos dentro del body de la solicitud HTTP en formato JSON y, como respuesta, el endpoint debe devolver las métricas de error del modelo original y las del nuevo. Internamente el endpoint debe almacenar (sin sobre-escribir) la nueva versión del modelo.
- **Predicción:** Pasando datos "futuros" o de prueba (*DataSet_Prediccion.txt*), este endpoint debe estar en la capacidad de generar las predicciones de churn para cada registro ingresado. Estos datos deben ser recibidos dentro del body de la solicitud HTTP en formato JSON y, como respuesta, el endpoint debe devolver para cada registro el **label y la probabilidad de predicción** en el mismo orden en el que

se ingresan los datos. Por defecto, las predicciones deberán realizarse con base en el modelo más actual.

- **Bono (10 pts):** El endpoint de predicción debe estar en la capacidad de recibir como query param la versión del modelo sobre la cual se debe generar la predicción. Si bien par algunos casos puede que el label de predicción no cambie entre modelos, la probabilidad muy seguramente sí será diferente.

Despliegue mediante API REST (25 pts)

Grabe y suba a YouTube un video de máximo 8 minutos en el cual:

- Explique el trabajo realizado en términos de la construcción del Pipeline, la búsqueda del mejor modelo y la implementación de la API REST.
- Ejemplifique el uso del endpoint de re-entrenamiento utilizando los datos del archivo *DataSet_Entrenamiento_v2.txt*.
- Ejemplifique el uso del endpoint de predicción utilizando los datos del archivo *DataSet_Prediccion.txt*. En el caso de haber realizado el bono,, se deben ejemplificar ambos escenarios: con el envío del query param, sin el envío del query param.

Mecanismo de entrega

- El taller debe ser desarrollado en grupos de 2 a 3 estudiantes.
- Debe utilizar únicamente los archivos de datos proporcionados.
- Debe ser entregado en los tiempos estipulados y solo a través de BloqueNeón. No se admiten entregas por otros medios como correo electrónico.
- El entregable debe consistir de un repositorio público de GitHub, el cual debe incluir:
    - Notebook de creación del Pipeline y experimentación con los outputs de la ejecución de cada bloque, pero también deberá poder ser ejecutado en su totalidad.
    - Implementación de la API REST con los endpoints solicitados.
    - Link del video subido a YouTube en el archivo Readme.

  En BloqueNeón se debe subir solo la URL del repositorio, no se admitirán commits posteriores a la fecha máxima de entrega.

- Dentro del notebook, haga uso de celdas de texto tipo markdown para exponer sus resultados y/o conclusiones de cada punto. También puede utilizar el archivo Readme del repositorio para concluir lo que considere necesario.
- **Todos los integrantes del grupo deben participar en el video.**