Trabajo Práctico 3

Haciendo Ciencia de Datos

En este trabajo final se propone que cada grupo de alumnos se enfrente a un problema real de ciencia de datos, que trabaje en cada una de las etapas del proceso y que pueda resolverlo aplicando todos los contenidos que vimos en la materia.

La entrega del trabajo práctico consistirá en una máquina Júpiter (Google Colab, Kaggle o una propia) en donde se cumplan todas las consignas del enunciado. Se debe indicar claramente qué tareas realizaron, qué supuestos consideraron, qué decisiones tomaron y se deben detallar todos los resultados obtenidos al final de cada etapa. Para ello se utilizarán las celdas de texto, intercaladas con las de código.

Fecha máxima de entrega: 15/08/2022 (última fecha de coloquio)
Defensa del trabajo: se realizará en las fechas de coloquio, el plazo máximo es de dos cuatrimestres desde que finaliza la cursada.

Parte 1 – Conjunto de Datos

Se utilizará un conjunto de datos provisto por la empresa **American Express** a través del sitio de Kaggle (link), el cual contiene millones de registros de transacciones, con un total de 190 columnas. El objetivo es obtener la probabilidad que un cliente pague o no la deuda de su tarjeta de crédito.

Cómo el conjunto de datos de entrenamiento es demasiado grande (16.39 GB), cada grupo trabajará con un subconjunto del mismo, tomando sólo un 5% del total de datos (839 MB).
Para obtener los registros correspondientes, cada grupo realizará un muestreo aleatorio sin repeticiones con la función de **Scikit-learn**: `sample_without_replacement`

Para hacer reproducible el experimento, cada grupo va a usar un valor entero como semilla en el parámetro: `random_state`, de la función anterior. Dicho valor semilla será calculado con la siguiente fórmula:

(31416 × <número-grupo>) mod 1000

En donde “<número-grupo>” es el número de grupo, 1,2,3…,10,11,12, etc

Parte 2 – Ciencia de Datos

1. Exploración, preprocesamiento y transformación de datos

a) **Visualización de los datos**: en esta sección se espera que puedan realizar una primera aproximación a los datos apoyándose en visualizaciones, por ejemplo: gráficos de dispersión entre variables, histogramas, heatmaps, exploración de las columnas y cualquier otro gráfico adicional que se considere útil justificando su utilización.

b) **Ingeniería de características:** esta sección se espera que se trabaje como mínimo en los siguientes ítems:

   a. Revisar los datos faltantes o mal ingresados y tomar una decisión sobre estos: reemplazo de valores, eliminación de registros incompletos, etc.
   b. Verificar columnas a descartar e identificar nuevas columnas que podrían crearse a partir de otras o añadiendo información externa.
   c. Identificación de outliers, utilizar métodos como el *Z-Score* y gráficos de caja. ¿Qué significan? ¿Se los elimina? ¿Se los deja?
   d. Evaluar si es posible y si se justifica una reducción en la dimensionalidad.
   e. Determinar si es necesario normalizar o unificar valores en alguna columna
   f. Balancear el conjunto de datos
   g. Identificar y documentar cualquier otra tarea de limpieza de datos que considere necesaria.

c) **Descripción de los datos**: se espera que puedan explicar qué se ha podido averiguar hasta ahora sobre el conjunto de datos. Esta información debe ser complementaria a la que se describe en la página de Kaggle. En este punto se debe definir qué **métricas** se utilizarán para medir el desempeño de los modelos.

2. Generación y evaluación de modelos

En esta sección cada grupo podrá optar por partir el conjunto de datos en un set de entrenamiento y otro de prueba, o utilizar *cross validation* para resolver los siguientes puntos:

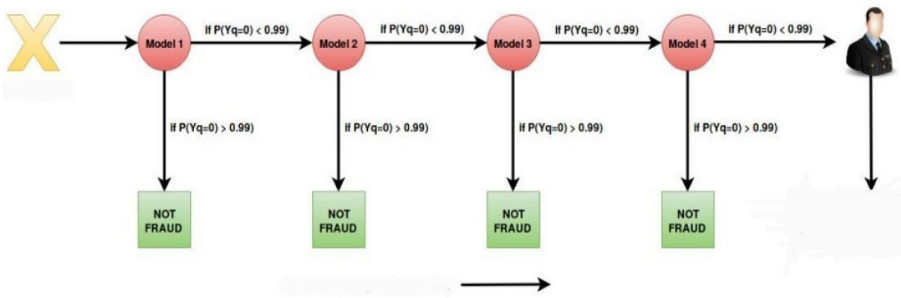
a) **Modelos:** probar varios modelos e ir ajustando distintos hiperparámetros para intentar mejorar el rendimiento de cada uno de ellos. Los modelos a probar deberán incluir:
   i. Random Forest
   ii. xGBoost

iii. SVM

b) **Ensamble de modelos:** crear un ensamble con los modelos anteriores de tipo *VotingClassifier*, y analizar cómo se comporta.

c) **Redes Neuronales**: crear una red neuronal que mejore, o al menos iguale, el desempeño de los modelos anteriores incluido el modelo ensamblado en el item b).

d) **Ensamble en cascada**: partiendo de la red neuronal como primer modelo, crear otro ensamble, esta vez de tipo *cascading*, replicando el siguiente diseño:



3. Conclusiones

Finalmente se pide seleccionar el mejor modelo, ajustarlo para obtener el mejor rendimiento posible y escribir luego algunas conclusiones que hayan podido obtener de este trabajo.

Dejamos a continuación algunas preguntas, a modo de guía, que les pueden resultar útiles para desarrollar las conclusiones:

a. ¿Es posible detectar si un cliente va a dejar de pagar?
b. ¿En qué casos sí? ¿En qué casos no?
c. ¿Qué información adicional creen que sería útil agregar?
d. ¿Qué peso tendría la intervención humana si el sistema entrase en producción?
e. ¿Qué habría que tener en cuenta si se decide mantener actualizado el sistema a lo largo del tiempo?