Trabajo Práctico 1 : Reservas de Hotel

Introducción

En este trabajo práctico 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 los contenidos que vemos en la materia.
Vamos a utilizar un conjunto de datos de reservas de hotel provisto por la cátedra. El objetivo principal del trabajo será aplicar técnicas de análisis exploratorio, preprocesamiento de datos y se entrenarán modelos de clasificación para predecir si una reserva va a ser cancelada. En la sección enunciado se detallan los objetivos particulares de cada etapa del trabajo.

Modalidad de entrega

Notebook

El trabajo debe ser realizado en una notebook *Jupyter* de Python, se espera que la misma contenga **todos** los resultados de la ejecución los cuales siempre deben ser **reproducibles**. La notebook debe respetar la siguiente nomenclatura :

7506R_TP1_GRUPOXX_CHPX_ENTREGA

En el caso que sea estrictamente necesario entregar más de una notebook las mismas deben contar con una numeración correlativa manteniendo un orden lógico entre ellas (7506R_TP1_GRUPOXX_CHPX_ENTREGA_**N1**, 7506R_TP1_GRUPOXX_CHPX_ENTREGA_**N2**, etc)
Las secciones del trabajo deben estar claramente diferenciadas en la notebook utilizando celdas de *markdown*. Se debe incluir una sección principal con el título del trabajo, el número de grupo y el nombre de todos los integrantes.
Todo análisis realizado debe estar acompañado de su respectiva explicación y toda decisión tomada debe estar debidamente justificada. Cualquier hipótesis que sea considerada en el desarrollo del trabajo práctico debe ser detallada y debe estar informada en la entrega. Cualquier criterio que se utilice basado en fuentes externas (papers, bibliografía, etc.) debe estar correctamente referenciado en el trabajo.

Visualizaciones

Todos los gráficos que se incorporen deben tener su correspondiente título, leyenda, nombres en los ejes, unidades de medidas, y cualquier referencia que se considere necesaria. Es importante que tengan presente que los gráficos son una herramienta que facilita entender el problema, por lo tanto, deben ser comprensibles por quien los vaya a leer.

Preprocesamiento:
A partir de las tareas de preprocesamiento, y de las diferentes estrategias que se planteen, es posible que se generen nuevos *datasets* sobre los cuales se entrenarán los modelos. Todo conjunto de datos creado debe ser almacenado y debe estar disponible en la entrega para ser utilizado por el equipo docente.

Modelos
Todos los modelos entrenados tanto para clasificación como para regresión deben ser guardados en un archivo (joblib / pickle) y deben estar disponibles en la entrega para ser utilizado por el equipo docente.

Reportes
Todos los reportes/informes solicitados deben respetar la extensión máxima solicitada, deben estar en formato pdf y deben tener la siguiente nomenclatura

7506R_TP1_GRUPOXX_CHPX_REPORTE.pdf

Repositorio
Cada grupo deberá crear su propio repositorio en github con la siguiente nomenclatura:

7506R-2C2023-GRUPOXX

En dicho repositorio deberá estar disponible la notebook, los modelos entrenados, los conjuntos de datos utilizados para el entrenamiento y cualquier archivo que sea necesario para la correcta ejecución del trabajo.

Competencia Kaggle

El trabajo práctico estará enmarcado en una competencia de **Kaggle**, dónde todos los alumnos deberán participar. Para unirse a la misma deben acceder con el siguiente enlace y conformar los grupos correspondientes. Pueden elegir cualquier nombre que represente al equipo.

El objetivo de la competencia es hacer la predicción más precisa posible acerca de si una reserva será cancelada. Para saber qué tan bien se desempeña un modelo, cada grupo hará su predicción sobre el conjunto de test y la subirá (submit) a **Kaggle**.

Kaggle verificará las predicciones contra el archivo de soluciones utilizando la **métrica F1** y mostrará la posición del equipo en la tabla de puntajes (leaderboard). Pero sólo usará para ello el 70% de sus respuestas, hay un 30% que se usarán también para calcular un tablero de puntajes privado que sólo podremos ver los docentes y se revelará al finalizar la competencia (**02/11/2023**).
Antes de finalizar la competencia cada grupo deberá seleccionar sus mejores 2 predicciones las cuales se utilizarán para el ranking final, el cual usa todos los datos.

Enunciado

Los conjuntos de datos a utilizar **hotels_train** y **hotels_test** se encuentran disponibles en la competencia de Kaggle y deberán descargarlos desde allí. Allí mismo encontrarán también un archivo de ejemplo de cómo se deben subir las soluciones.
Se deberán explorar los datos de train, realizar ingeniería de características y entrenar modelos de clasificación para poder predecir si una reserva será cancelada: el ***target*** será la variable **is_canceled.** Las tareas de preprocesamiento deberán replicarse en los datos de test ya que será necesario obtener las predicciones sobre el conjunto de evaluación y subirlas a **Kaggle**.

A continuación se detallan las etapas que deben ser desarrolladas en el trabajo:

Análisis Exploratorio y Preprocesamiento de Datos

Checkpoint 1:

a) **Exploración Inicial :** analizar cada variable, considerando los siguientes aspectos
   - Tipo de variable
   - Variables Cuantitativas: calcular medidas de resumen: media, mediana,moda, etc
   - Variables Cualitativas: reportar los posibles valores que toman y cuán frecuentemente lo hacen.
   - Determinar si existen variables irrelevantes para el análisis
   - Realizar un análisis gráfico de las distribuciones de las variables
   - Analizar las correlaciones existentes entre las variables.
   - Analizar la relación de las variables con el **target**

b) **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.

c) **Datos Faltantes :** analizar la presencia de datos faltantes en el *dataset*
   - Realizar análisis de datos faltantes a nivel de columna. Graficar para cada variable el porcentaje de datos faltantes con respecto al total del dataset
   - Revisar los datos faltantes o mal ingresados y tomar una decisión sobre estos: reemplazo de valores, eliminación de registros incompletos, etc.
   - En caso de realizar imputaciones comparar las distribuciones de cada atributo reparado con la distribución anterior a la imputación de los datos faltantes.

d) **Valores atípicos :** analizar la existencia de valores atípicos
   - Detectar valores atípicos en los datos tanto en forma univariada como multivariada. Realizar gráficos que permitan visualizar los valores atípicos.
   - Explicar qué características poseen los datos atípicos detectados.
   - Decidir el tratamiento a aplicar sobre los mismos.

Nota : Los ítems a, b, c y d son los mínimos requeridos para esta etapa, cada grupo puede realizar cualquier otra tarea de limpieza de datos que considere necesaria, crear nuevas variables derivadas de los atributos existentes o que resulten de incorporar nuevas fuentes de datos.

Clasificación - Entrenamiento y Predicción

El objetivo será predecir el valor del atributo **is_canceled, los modelos deberán entrenarse con los datos de train** y se deberán realizar las tareas de ingeniería de características necesarias para trabajar con cada algoritmo (encoding, normalización, balanceo, etc)

Checkpoint 2 : Árbol de decisión

a. Construir árboles de decisión y optimizar sus hiperparámetros mediante *k-fold Cross Validatio*n para obtener la mejor performance. ¿Cuántos *folds* utilizaron? ¿Qué métrica consideran adecuada para buscar los parámetros?
b. Graficar el árbol de decisión con mejor performance encontrado en el punto anterior. Si es muy extenso mostrar una porción representativa.
c. Analizar el árbol de decisión seleccionado describiendo los atributos elegidos, y decisiones evaluadas (explicar las primeras reglas obtenidas).
d. Evaluar la performance del modelo en entrenamiento y validación, explicar todas las métricas y mostrar la matriz de confusión.
e. Generar predicciones con el conjunto de test y realizar los submits correspondientes en la competencia de **Kaggle**.

Checkpoint 3: Ensambles

a. Construir un clasificador KNN optimizar sus hiperparámetros mediante *k-fold Cross Validation*
b. Construir un clasificador SVM variando el kernel y los parámetros.
c. Construir un clasificador RF y optimizar sus hiperparámetros mediante *k-fold Cross Validation*

d. Construir un clasificador XGBoost y optimizar sus hiperparámetros.
e. Construir un ensamble híbrido tipo Voting y otro tipo Stacking.
f. Evaluar la performance de todos los modelos en entrenamiento y validación, explicar todas las métricas y mostrar la matriz de confusión.
g. Generar predicciones con el set de test y realizar los submits correspondientes en la competencia de **Kaggle**.

Checkpoint 4: Redes Neuronales

a. Construir una red neuronal para clasificación y mejorar su performance mediante la búsqueda de arquitectura e hiperparámetros adecuados.
b. Evaluar la performance de todos los modelos en entrenamiento y validación, explicar todas las métricas y mostrar la matriz de confusión.
c. Generar predicciones con el conjunto de test y realizar los submits correspondientes en la competencia de **Kaggle**.
d. Generar las conclusiones finales del trabajo práctico evaluando la performance de todos los modelos entrenados.

Fechas de entrega

Todas las entregas son **obligatorias**, sólo el ***checkpoint 1*** cuenta con posibilidad de reentregar.
Una vez realizada la entrega cada grupo debe comunicarlo a su corrector por el canal de slack correspondiente.

Checkpoint 1 : 21/09 - Análisis Exploratorio + Ingeniería de Features
Para esta fecha se espera que tengan resueltas las tareas de análisis exploratorio y preprocesamiento de datos. Se debe entregar la notebook más un resumen (máximo dos hojas) que describa las tareas realizadas, la información relevante sobre el *dataset* y que contenga dos visualizaciones que acompañen sus hallazgos. Ejemplo de reporte CHP1 acá.
Hay una única reentrega el 05/10.

Checkpoint 2: 05/10 - Árboles de Decisión
Para esta fecha se espera que hayan entrenado árboles de decisión y hayan participado activamente de la competencia. Se debe entregar la notebook, el modelo entrenado, el archivo con las predicciones (*submit* seleccionado de Kaggle .csv) y un reporte (máx 1 carilla .pdf) con el detalle del modelo construido, resultados obtenidos y cualquier comentario adicional.Ejemplo de reporte CHP2 acá.
Esta fecha es obligatoria sin reentrega.

Checkpoint 3: 19/10 - Ensamble de Modelos
Para esta fecha se espera que hayan entrenado varios modelos con el objetivo de realizar ensambles híbridos y hayan participado activamente de la competencia. Se debe entregar la notebook, el modelo entrenado, el archivo con las predicciones (submit seleccionado de kaggle) más un reporte (máximo 1 carilla) con el detalle del modelo construido, resultados obtenidos y cualquier comentario adicional.Ejemplo de reporte CHP3 acá.
Esta fecha es obligatoria sin reentrega.

Checkpoint 4: **02/11 - Redes Neuronales + Conclusiones (Entrega Final)**
Para esta fecha se espera que hayan entrenado una red neuronal y hayan participado activamente de la competencia. Se debe entregar la notebook, el modelo entrenado, el archivo con las predicciones (submit seleccionado de kaggle) más un reporte (máximo 1 carilla) con el detalle del modelo construido, resultados obtenidos y cualquier comentario adicional.
En esta entrega se deben realizar las conclusiones correspondientes al trabajo realizado en su totalidad, destacando principalmente los aspectos que consideren más relevantes. Comentar brevemente qué otras opciones hubiesen explorado y quedaron fuera del alcance de este trabajo (máximo 2 carillas).Ejemplo de reporte CHP4 acá.
Esta fecha es obligatoria sin reentrega.

Condiciones de Aprobación

- Se deben respetar **todas** las fechas de entrega.
- Se deben respetar **todas** las condiciones de entrega de cada fecha.
- **Todos** los grupos deben participar de la competencia.
- **Todos** los integrantes de los grupos deben participar de la competencia.
- Se deben realizar submits **todas las semanas** luego del checkpoint 1.
- Se debe tener **al menos 1 submit** por modelo/ensamble pedido.