

Desafío - Random Forest

- Para realizar este desafío debes haber estudiado previamente todo el material disponibilizado correspondiente a la unidad.
- Una vez terminado el desafío, comprime la carpeta que contiene el desarrollo de los requerimientos solicitados y sube el `.zip` en el LMS.
- Desarrollo desafío:
 - El desafío se debe desarrollar de manera Individual.
 - Para la realización del desafío necesitarás apoyarte del archivo *Apoyo Desafío - Random Forest*.

Requerimientos

- Para esta sesión trabajaremos con una base de datos sobre los precios de inmuebles en la ciudad de Ames, Iowa. La base se compone de 2930 registros y contiene un gran número de atributos.
- Nuestro objetivo es generar un modelo que prediga de forma adecuada los precios de inmuebles, medidos con la variable `Sale_Price`.
- A diferencia de otras sesiones donde implementamos el preprocesamiento a mano, ahora haremos uso de los archivos serializados en la sesión pasada.

Ejercicio 1: Preparación del ambiente de trabajo

- Importe las librerías clásicas a utilizar.
- Para este ejercicio Random Forest de Regresión.
- De manera adicional importe las funciones y clases necesarias para generar un desempeño de métricas en problemas de regresión.
- Elimine la columna 'Unnamed: 0' cuando cargue los datos.

Ejercicio 2: Importación de archivos serializados

- Importe su modelo entrenado y sus conjuntos de entrenamiento y validación serializados la sesión pasada y evalúe el desempeño del modelo de su compañero con el suyo ¿Cuál es mejor? Si no puede ejecutar el modelo, comente cuáles pueden ser los causantes.
- No es necesario que realice diagnósticos gráficos sobre el modelo, sólo reporte sus métricas.
- Para importar un archivo serializado, puede ocupar la siguiente línea de código:

```
import pickle  
read_model = pickle.load(open("ignacio-soto-act07.sav", "rb"))
```

Ejercicio 3: Evaluación Random Forest

- En base a los conjuntos de entrenamiento serializados, genere un modelo utilizando `sklearn.ensemble.RandomForest` sin modificar hiper parámetros, sólo declarando la semilla pseudoaleatoria.
- Reporte su desempeño y compárelo con su modelo de la sesión pasada, así como con el de su compañero.

Ejercicio 4: Reporte las métricas de desempeño

- Para afinar el comportamiento de nuestro modelo, evalúe su desempeño los siguientes hiper parámetros:
 - `max_features`: `None`, `log2`, `sqrt`.
 - `n_estimators`: Evaluar entre 20 y 1000 en pasos de 50.
- Guarde la tasa de error en la exactitud medida con OOB.
- Grafique el comportamiento.

Ejercicio 5: Refactoriza el modelo

- En base a la mejor combinación de hiper parámetros, reentrene el modelo y comente su desempeño.