

# Evaluación Módulo 1

## IA para profesionales TIC – 3ª edición

### Introducción

En una red eléctrica, hemos podido tomar 12 características numéricas (**tau1, tau2, tau3, tau4, p1, p2, p3, p4, g1, g2, g3, g4**) y si con esta información podemos saber si la red eléctrica es estable o no (característica llamada **stabf** ser 'estable' o 'inestable').

El conjunto de datos que vamos a utilizar se basa en `electric_grid.csv`, un dataset disponible en Internet, pero ha sido modificado deliberadamente. Así que, ten cuidado ;)

El conjunto de datos de entrenamiento tiene 7500 muestras, para cada una de ellas tenemos los valores numéricos de las variables o características dependientes, y la variable objetivo/clase (**columna stabf**).

Viene como `electrical_grid_train.csv` y viene con las 7500 filas.

### Pasos a seguir

Enlaces a los archivos requeridos

[https://www.dropbox.com/s/n0t1fi534rhycoc/electrical\\_grid\\_train.csv](https://www.dropbox.com/s/n0t1fi534rhycoc/electrical_grid_train.csv)

[https://www.dropbox.com/s/v20v43hwmkj1bol/electrical\\_grid\\_test.csv](https://www.dropbox.com/s/v20v43hwmkj1bol/electrical_grid_test.csv)

### Entregas

Debes entregar un notebook .ipynb exportado desde Colab y un archivo PDF siguiendo el formato que verás a continuación.

Crear un bloc de notas de Google Colab y un pdf de descripción

El Google Colab debe tener este nombre:

`modulo1_nombre_apellido` y tendrás que descargarlo en `.ipynb`.

Crea un documento (`modulo1_nombre_apellido.pdf`) donde expliques tu notebook con cualquier comentario adicional que creas de interés y pueda ayudar en la corrección de tu tarea. En este PDF tendrás que poner también los RESULTADOS de 'accuracy' y otras MÉTRICAS, el modelo final que elijas, y sus hiperparámetros.

Usa celdas de markdown para comentar y explicar cada paso importante, y usa también comentarios de python para dar información adicional sobre el código.

Puedes usar cualquier código que hayamos estado usando en nuestros cuadernos y, por supuesto, podéis hacer uso de ChatGPT, Gemini, etc. Siempre teniendo en cuenta su no infalibilidad.

Usa **scikit-learn** para todos los modelos evaluar diferentes modelos. Mínimo regresión logística, SVM, árbol de decisión y random forest. Se valorará probar XGBOOST.

Lectura de los datos

Realiza las importaciones requeridas.

Carga el archivo csv como un DataFrame de pandas:

```
datos = pd.read_csv('electrical_grid_train.csv')
```

Análisis exploratorio de datos (EDA)

Visualiza la información requerida y común de su conjunto de datos de entrenamiento, incluidas algunas filas, tipos de datos de las columnas, correlación con Seaborn, etc.

¿Alguna decisión interesante que tomar? En caso afirmativo, descríbelo en el PDF. Recuerda que cualquier cambio en el conjunto de datos original debe evaluarse y compararse con la no realización de dicha acción.

### Creación de X e y

Crea X como las variables dependientes del conjunto de datos e y como la **columna stabf** (el destino)

Utiliza varios modelos y evalúa el mejor con **GridSearchCV** teniendo en cuenta los hiperparámetros que consideres en el proceso de búsqueda.

Evalúe los siguientes modelos:

- SVM (SVC – clasificador)
- Árbol de decisión
- Regresión logística
- Random forest (clasificador)
- XGBoost (clasificador) - Opcionals

Con 5 Kfolds (cv=5, es decir, 80% 20%)

Dar la precisión final de ellos que obtenemos en GridSearchCV. Recuerda poner esta información en el PDF para una mejor corrección.

Elije uno con respecto a estos resultados. De nuevo, documenta esto en el PDF.

Deberías obtener una precisión superior a 0,80 fácilmente.

En el proceso EDA, si detectas algún tipo de acción (como eliminar una característica, etc.), hazlo, y también compara si lo que obtienes es mejor, y las ventajas o desventajas de tus acciones. Documenta esto en el PDF.

### Resultado final

Abre como un nuevo dataset el archivo  
`'electrical_grid_test.csv'`.

En este archivo tienes 2500 nuevas muestras reales con sus valores **de stabf correctos**. Haz las predicciones de estas 2500 muestras (sin la columna **stabf**) con tu mejor modelo y compáralas con el valor real de las mismas (la columna **stabf** en `'electrical_grid_test.csv'`)

Calcula su 'accuracy' final con este nuevo conjunto de nuevos valores (**accuracy** y **f-1**).