

## Pre-requisitos

1. **Nota:** defina un nombre para su grupo, defínalo claramente en **reporte** y use este nombre como parte inicial de **todos los recursos que cree**.
2. **Nota 2:** la entrega de este taller consiste en un **reporte** y unos **archivos de soporte**. Cree el archivo de su **reporte** como un documento de texto en el que pueda fácilmente incorporar capturas de pantalla, textos y similares. Puede ser un archivo de word, libre office, markdown, entre otros.
3. La sección de Databricks la deben realizar los dos miembros del equipo e incluir soportes de los dos.

## 1. Opción 1: Instale y pruebe MLflow localmente

1. En su ambiente local active un ambiente de python. Por ejemplo, usando Anaconda en consola puede activar el ambiente base con el comando

```
activate base
```

2. En este ambiente instale mlflow

```
conda install -c conda-forge mlflow
```

3. En este ambiente instale sklearn

```
conda install -c conda-forge scikit-learn
```

4. En una terminal aparte lance la interfaz gráfica de MLflow con el comando

```
mlflow ui
```

Tome un pantallazo de la salida de la terminal e inclúyalo en su **reporte**.

5. La interfaz web debe quedar disponible a través del navegador en el puerto 5000. Abra la ubicación <http://localhost:5000> y compruebe que funciona.
6. Ahora ejecute el notebook **mlflow-diab** que encontrará en Bloque Neón. Estudie con cuidado el código y describa en su **reporte** qué hace.
7. Modifique alguno de los parámetros del modelo y vuelva a correr el modelo.
8. Ahora visite la interfaz web de MLflow e identifique el experimento y las dos corridas realizadas. Compare los dos modelos en términos de MSE (mean squared error) usando las funciones de la interfaz. Incluya un pantallazo de la comparación en su **reporte**.

## 2. Opción 2: Instale y pruebe MLflow en la nube

1. Lance una instancia en AWS EC2. Se recomienda una máquina t2.small, con sistema operativo Ubuntu y 20GB de disco. Incluya un pantallazo de la consola de AWS EC2 con la máquina en ejecución en su **reporte**.

2. Conéctese a la máquina con SSH.

3. Actualice las ubicaciones de los paquetes

```
sudo apt update
```

4. Instale pip para python3

```
sudo apt install python3-pip
```

5. Instale sklearn

```
pip install scikit-learn
```

6. Instale mlflow

```
pip install mlflow
```

Incluya la dirección `/home/ubuntu/.local/bin` en el PATH para facilitar su ejecución

```
PATH=$PATH:/home/ubuntu/.local/bin
```

7. Descargue el archivo **mlflow-diab.py** que encontrará en Bloque Neón (es un archivo de python, no un cuaderno). Estudie con cuidado el código y describa en su **reporte** qué hace.

8. Comente la línea de registro de mlflow (línea 17)

```
mlflow.set_tracking_uri('http://0.0.0.0:5000')
```

9. En una terminal aparte, suba el archivo **mlflow-diab.py** a la máquina con un comando como

```
scp -i llave.pem mlflow-diab.py ubuntu@IP:/home/ubuntu
```

10. En la terminal de la máquina ejecute mlflow como servidor y exponga el servicio por el puerto 8050

```
mlflow server -h 0.0.0.0 -p 8050
```

Tome un pantallazo de la salida de la terminal e inclúyalo en su **reporte**.

11. La interfaz web debe quedar disponible a través del navegador en el puerto 8050. Abra el puerto en el grupo de seguridad de la máquina y compruebe que funciona.

12. Ejecute el archivo de python para entrenar y evaluar el modelo

```
python3 mlflow-diab.py
```

13. Modifique alguno de los parámetros del modelo y vuelva a correr el script.
14. Ahora visite la interfaz web de MLflow e identifique el experimento y las dos corridas realizadas. Compare los dos modelos en términos de MSE (mean squared error) usando las funciones de la interfaz. Incluya un pantallazo de la comparación en su **reporte**.

### 3. MLflow en Databricks

1. Inicie creando una cuenta en Databricks **Community Edition** <https://www.databricks.com/try-databricks#account>
2. Provea sus datos y un correo.
3. En la sección **Choose a cloud provider** en la parte inferior seleccione **Get started with Community Edition**.
4. Valide su correo y genere su contraseña.
5. Una vez haya creado la cuenta, ingrese con sus credenciales y familiarícese con la consola principal.
6. En el primer ítem del menú de la izquierda seleccione Data Science and Engineering.
7. Luego de click en Create - Cluster.
8. Asigne un nombre al cluster, seleccione un **Runtime de ML**, en su versión más reciente.
9. Click en crear cluster y espere a que el cluster se haya creado. Incluya un pantallazo de su clúster operando en su **reporte**.
10. En el primer ítem del menú de la izquierda seleccione Machine Learning.
11. En el menú de la izquierda click en Create - Notebook.
12. Asigne un nombre al notebook, con lenguaje python por defecto y asígnelo al cluster que acaba de crear.
13. Con el notebook abierto, copie las instrucciones del notebook **mlflow-diab-db** que encontrará en Bloque Neón.
14. Revise y comprenda las instrucciones de cada celda.
15. Ejecute cada celda y explore los resultados.
16. En el menú de la izquierda seleccione Experiments.

17. Seleccione el experimento realizado y navegue la documentación asociada (versión del modelo, librerías, python).
18. Cambie los parámetros del entrenamiento del modelo y ejecute un nuevo experimento.
19. Revise los resultados en Experiments, genere gráficas comparativas. Incluya un pantallazo de las gráficas en su **reporte**.
20. Repita este procedimiento varias veces.
21. Repita este procedimiento con el notebook **mlflow-mnist** que encontrará en Bloque Neón. Incluya un pantallazo de las gráficas en su **reporte**.
22. En su **reporte** explique en qué consiste este último notebook y los modelos que allí se entrenan. Documente alguna observación sobre sus resultados.