Set-up for hyperparameter tuning

To begin, you need to have access to an Azure Databricks workspace with an interactive cluster. If you do not have a workspace and/or the required cluster, follow the instructions below. Otherwise, you can skip to the **Upload data** section below. Note that you need to upload **two different datasets** to complete the two exercises!

Create Azure Databricks resources

To use Azure Databricks, you first need to deploy an Azure Databricks workspace in your Azure subscription and create a cluster on which you will run notebooks and code. You can then upload the data and notebooks to experiment with in your workspace.

Deploy an Azure Databricks workspace

1. In the [Azure portal](#), create a new **Azure Databricks** resource, specifying the following settings:

   - **Subscription**: *Choose the Azure Subscription in which to deploy the workspace.*
   - **Resource Group**: *Create a new resource group.*
   - **Workspace Name**: *Provide a name for your workspace.*
   - **Region**: *Select a location near you for deployment. For the list of regions supported by Azure Databricks, see [Azure services available by region](#).*
   - **Pricing Tier**: Standard

2. Wait for the workspace to be created. Workspace creation takes a few minutes. During workspace creation, the portal displays the Submitting deployment for Azure Databricks tile on the right side. You may need to scroll right on your dashboard to see the tile. There is also a progress bar displayed near the top of the screen. You can watch either area for progress.

Create a cluster

1. When your Azure Databricks workspace resource has been created, go to it in the portal, and select **Launch Workspace** to open your Databricks workspace in a new tab, signing in if prompted.

2. In the left-hand menu of your Databricks workspace, select **Compute**, and then select **+ Create Cluster** to add a new cluster with the following configuration:

   - **Name**: *Enter a unique name.*
   - **Cluster Mode**: Single Node
   - **Pool**: None
   - **Databricks Runtime Version**: *Select the **ML** edition of the latest available version of the runtime. Ensure that the version selected:*

     - *Does **not** use a GPU*
     - *Includes Scala > **2.11***
     - *Includes Spark > **3.0***
   - **Terminate after**: 120 minutes of inactivity
   - **Node Type**: Standard_DS3_v2

3. Wait for your cluster to be created, which may take several minutes. The cluster will start automatically, and eventually the spinning *Pending* indicator next to the cluster name will change to a solid green circle to indicate a status of *Running*.

Upload dataset exercise 1

1. Download `https://github.com/MicrosoftDocs/ml-basics/blob/master/challenges/data/real_estate.csv` to your computer, saving it as **real_estate.csv** in any folder.

2. On the **Data** page in the Databricks Workspace, select the option to **Create Table**.

3. In the **Files** area, select **browse** and then browse to the **real_estate.csv** file you downloaded.

4. After the file is uploaded to the workspace, select **Create Table with UI**. Then select your cluster and select **Preview Table**.

5. Specify the following table attributes, and then select **Create Table**.

   - **Table Name**: real_estate
   - **Create in Database**: default
   - **File Type**: CSV
   - **Column Delimiter**: , *(comma)*
   - **First row is header**: *checked*
   - **Infer schema**: *checked*
   - **Multi-line**: *unchecked*

6. After the table has been created, view it in the workspace.

Upload dataset exercise 2

1. Download `https://github.com/MicrosoftDocs/ml-basics/blob/master/challenges/data/wine.csv` to your computer, saving it as **wine.csv** in any folder.

2. On the **Data** page in the Databricks Workspace, select the option to **Create Table**.

3. In the **Files** area, select **browse** and then browse to the **wine.csv** file you downloaded.

4. After the file is uploaded to the workspace, select **Create Table with UI**. Then select your cluster and select **Preview Table**.

5. Specify the following table attributes, and then select **Create Table**.

   - **Table Name**: wine
   - **Create in Database**: default
   - **File Type**: CSV
   - **Column Delimiter**: , *(comma)*
   - **First row is header**: *checked*
   - **Infer schema**: *checked*
   - **Multi-line**: *unchecked*

6. After the table has been created, view it in the workspace.

Import Databricks notebooks

1. In the Azure Databricks Workspace, using the command bar on the left, select **Workspace**. Then select **Users**, and 📁 *your_user_name*.

2. In the blade that appears, select the downwards pointing chevron (**v**) next to your name, and select **Import**.

3. On the **Import Notebooks** dialog, import the notebook archive from the following URL, noting that a folder with the archive name is created, containing two notebooks:

   - `https://github.com/MicrosoftLearning/dp-090-databricks-ml/blob/master/05%20-%20Hyperparameter%20Tuning.dbc`

Explore automated MLflow hyperparameter tuning

In this exercise, you will discover how to use automated MLflow for hyperparameter tuning.

1. In the **05 - Hyperparameter tuning** folder in your workspace, open the **1.0 Automated MLflow Hyperparameter Tuning** notebook.

2. In the top left dropdown menu, choose your cluster to attach your notebook to that cluster. *(Alternatively, you will be prompted to attach a cluster when running the first cell in an unattached notebook).*

3. Read the notes in the notebook, running each code cell in turn.

Explore Hyperopt for hyperparameter tuning

In this exercise, you will discover how to use automated MLflow for hyperparameter tuning.

1. In the **05 - Hyperparameter tuning** folder in your workspace, open the **2.0 Automated MLflow Hyperparameter Tuning** notebook.

2. In the top left dropdown menu, choose your cluster to attach your notebook to that cluster. *(Alternatively, you will be prompted to attach a cluster when running the first cell in an unattached notebook)*.

3. Read the notes in the notebook, running each code cell in turn.

Clean-up

If you're finished working with Azure Databricks for now, in Azure Databricks workspace, on the **Compute** page, select your cluster and select ■ **Terminate** to shut it down. Otherwise, leave it running for the next exercise.