**Create your first data science experiment in Azure Machine Learning Studio**

In this lab session, you will create a machine learning experiment in Azure Machine Learning Studio that predicts **the price of a car based** on **different variables such as make and technical specifications.**

<https://docs.microsoft.com/en-us/azure/machine-learning/studio/what-is-ml-studio>

https://studio.azureml.net/

**Note that you should be able to login to Azure ML Studio with a Microsoft Live account (or your SLIIT student account).**

How much? or How many? uses regression algorithms

Regression algorithms make **numerical predictions**, such as:

What will the temperature be next Tuesday?

What will my fourth quarter sales be?

They help answer any question that asks for a number.

This lab Session follows the default workflow for an experiment:

1. Create a model

* Get the data
* Prepare the data
* Define features

1. Train the model

* Choose and apply an algorithm

1. Score and test the model

* Predict new automobile prices

**Get the data**

The first thing you need in machine learning is data. We'll use the sample dataset, **Automobile price data (Raw),** that's included in your workspace. This dataset includes entries for various **individual automobiles, including information such as make, model, technical specifications, and price.**

Here's how to get the dataset into your experiment.

1. Create a new experiment by clicking **+NEW** at the bottom of the Machine Learning Studio window. Select **EXPERIMENT** > **Blank Experiment**.
2. The experiment is given a default name that you can see at the top of the canvas. Select this text and rename it to something meaningful, for example, **Automobile price prediction**. The name doesn't need to be unique.

A screenshot of a cell phone

Description automatically generated

1. To the left of the experiment canvas is a palette of datasets and modules. Type automobile in the Search box at the top of this palette to find the dataset labeled **Automobile price data (Raw)**. Drag this dataset to the experiment canvas.

**Prepare the data**

A dataset usually requires some preprocessing before it can be analyzed. You might have noticed the missing values present in the columns of various rows. These missing values need to be cleaned so the model can analyze the data correctly.

We'll remove any rows that have missing values. Also, the normalized-losses column has a large proportion of missing values, so we'll exclude that column from the model altogether.

First, we’ll add a module that removes the normalized-losses column completely. Then we can add another module that removes any row that has missing data.

1. Type select columns in the search box at the top of the module palette to find the **Select Columns** in Dataset module. Then drag it to the experiment canvas. This module allows us to select which columns of data we want to include or exclude in the model.
2. Connect the output port of the Automobile price data (Raw) dataset to the input port of the Select Columns in Dataset.

A screenshot of a cell phone

Description automatically generated

1. Click the Select Columns in Dataset module and click **Launch column selector** in the **Properties** pane.

On the left, click **With rules**

Under **Begin With**, click **All columns**. These rules direct Select Columns in Dataset to pass through all the columns (except those columns we're about to exclude).

From the drop-downs, select **Exclude** and **column names**, and then click inside the text box. A list of columns is displayed. Select **normalized-losses**, and it's added to the text box.

Click the check mark (OK) button to close the column selector (on the lower right).

A screenshot of a cell phone

Description automatically generated

Now the properties pane for Select Columns in Dataset indicates that it will pass through all columns from the dataset except normalized-losses.

A screenshot of a cell phone

Description automatically generated

1. Drag the **Clean Missing Data** module to the experiment canvas and connect it to the Select Columns in Dataset module. In the Properties pane, select **Remove entire row under Cleaning mode**. These options direct Clean Missing Data to clean the data by removing rows that have any missing values.

A screenshot of a cell phone

Description automatically generated

1. Run the experiment by clicking **RUN** at the bottom of the page.

When the experiment has finished running, all the modules have a **green** check mark to indicate that they finished successfully. Notice also the Finished running status in the upper-right corner.

Now we have clean data. If you want to view the cleaned dataset, click the left output port of the Clean Missing Data module and select Visualize. **Notice that the normalized-losses column is no longer included, and there are no missing values.**

Now that the data is clean, we're ready to specify what features we're going to use in the predictive model.

**Define features**

In machine learning, features are **individual measurable properties** of something you’re interested in. In our dataset, **each row represents one automobile**, and **each column is a feature of that automobile.**

Let's build a model that uses a subset of the features in our dataset. You can come back later and select different features, run the experiment again, and see if you get better results. But to start, let's try the following features.

**make, body-style, wheel-base, engine-size, horsepower, peak-rpm, highway-mpg, price**

1. Drag another **Select Columns in Dataset module** to the experiment canvas. Connect the **left output port** of the Clean Missing Data module to the input of the Select Columns in Dataset module.
2. Click Launch column selector in the Properties pane.

Click With rules. Under Begin With, click **No columns**. In the filter row, select **Include** and column names and **select our list of column names in the text box.** This filter directs the module to not pass through any columns (features) except the ones that we specify.

Click the check mark (OK) button.

A screenshot of a cell phone

Description automatically generated

This module produces a filtered dataset containing only the features we want to pass to the learning algorithm we'll use in the next step. Later, you can return and try again with a different selection of features.

**Choose and apply an algorithm**

Now that the data is ready, constructing a predictive model consists of training and testing. We'll use our data to train the model, and then we'll test the model to see how closely it's able to predict prices.

Because we want to predict price, which is a number, we'll use a regression algorithm. For this example, we'll use a linear regression model.

We train the model by giving it a set of data that includes the price. The model scans the data and look for correlations between an automobile's features and its price. Then we'll test the model - we'll give it a set of features for automobiles we're familiar with and see how close the model comes to predicting the known price.

We'll use our data for both training the model and testing it by **splitting** the data into separate training and testing datasets.

1. Select and drag the **Split Data module** to the experiment canvas and connect it to the last Select Columns in Dataset module.
2. Click the Split Data module to select it. Find the **Fraction of rows in the first output dataset** (in the Properties pane to the right of the canvas) **and set it to 0.75**. This way, we'll use **75 percent** of the data to **train the model** and hold back **25 percent for testing.**

**A screenshot of a cell phone

Description automatically generated**

**By changing the Random seed parameter, you can produce different random samples for training and testing. This parameter controls the seeding of the pseudo-random number generator.**

1. Run the experiment
2. To select the learning algorithm, expand the **Machine Learning category** in the module palette to the left of the canvas, and then expand **Initialize Model**. This displays several categories of modules that can be used to initialize machine learning algorithms. For this experiment, select the **Linear Regression** module under the Regression category, and drag it to the experiment canvas. (You can also find the module by typing "linear regression" in the palette Search box.)
3. Find and drag the **Train Model** module to the experiment canvas. Connect the output of the **Linear Regression module to the left input of the Train Model module** and connect the **training data output (left port) of the Split Data module to the right input of the Train Model module.**

**A close up of a device

Description automatically generated**

1. Click the Train Model module, click Launch column selector in the Properties pane, and then select the **price** column. **Price is the value that our model is going to predict**.

You select the price column in the column selector by moving it from the **Available** **columns list** to the **Selected columns list**.

A screenshot of a cell phone

Description automatically generated

1. Run the experiment.

A picture containing screenshot

Description automatically generated

**Predict new automobile prices**

1. Find and drag the **Score Model** module to the experiment canvas. Connect the **output of the Train Model module to the left input port of Score Model**. Connect the **test data** **output (right port) of the Split Data module to the right input port of Score Model.**

A close up of a map

Description automatically generated

1. Run the experiment and view the output from the Score Model module by clicking the output port of Score Model and select Visualize. **The output shows the predicted values for price and the known values from the test data.**

**A screenshot of a cell phone

Description automatically generated**

**Submission: You may upload a screenshot of the prediction view as an image. The file name should be submission should be your registration number.**

**E.g. ITXXXXXX.jpg**