

**Experiment No. 7**

**Title: Develop a Multiple linear regression model using Microsoft Azure Machine Learning Studio**

**Batch: B1**

**Roll No. 1914122**

**Experiment no.: 7**

**Title:** Develop a Multiple linear regression model using Microsoft Azure Machine Learning Studio

**Describe the following points with respect to the business under consideration,**

---

**1. Problem faced by the business**

The system is designed such that it helps to estimate the price of a Car based upon the different features present, so it helps the business to give the correct amount of pricing to their Car Models, so that the customers are attracted and also are satisfied with price range assigned based upon it's features. So, assigning Price range accordingly is challenging without the use of ML algorithms.

**2. Approach/ Methodology followed by the business**

The methodology which we followed was that we took the dataset of Automobile Price Data. Then we split the dataset into train and test. Then we used Linear Regression and trained the machine learning model with that dataset and in the end we tested that model.

**3. Skillsets , infrastructure and other impact on the business during**

**implementation Skillset:** Cloud computing

**Infrastructure:** Microsoft Azure Cloud services

There were no as such impacts on the business, as any of the business service lines were not being used.

**4. Similar approaches followed by other businesses**

**Step 1.** Sign-in using Microsoft account on studio.azureml.net

**Step 2.** Creating workspace for our Machine Learning project.

**Step 3.** Select New option on bottom right:

**Step 4.** Click on Blank experiment and write name and summary of experiment

**Step 5.** Select From Saved Datasets-> Samples-> dataset of your choice

**Step 6.** Now, search 'Select columns in dataset' from items and drag it

**Step 7.** Now, click on launch column selector-> with rules->exclude column normalized-losses as that column contains many rows/records with empty values.

**Step 8.** Search and select 'Clean Missing Data' from items list

**Step 9.** Now, select cleaning mode -> Remove entire row as it will remove the entire row wherever missing value is found

**Step 10.** Again choose 'select columns in dataset'

**Step 11.** Now, launch column selector and include all the columns based on which prediction is to be done: make, body-style, wheel-base, engine-size, horsepower, peak-rpm, highway-mpg, price

**Step 12.** Now, select 'split data' from list and drag it

**Step 13.** For Split data, enter the fraction of data which is needed for training while rest will be used for testing

**Step 14.** Now, Select 'Linear Regression' as the algorithm to be used and 'Train Model' from list

**Step 15.** For training model, click on launch column selector, include price column as Price is what is to be predicted

**Step 16.** Add Score Model from list drag it and make

connections **Step 17.** Now, Add Evaluate Model from list and

make connections **Step 18.** Now, Click on Run

**Step 19.** To check prediction results, right click on Score Model, select visualize

**Step 20.** To check Evaluation results, right click on Evaluation Model, select visualize

## 1.Problem faced by the business:

The prices of new cars in the industry is fixed by the manufacturer with some additional costs incurred by the Government in the form of taxes. So, customers buying a new car can be assured of the money they invest to be worthy. But due to the increased price of new cars and the incapability of customers to buy new cars due to the lack of funds, used cars sales are on a global increase. There is a need for a used car price prediction system to effectively determine the worthiness of the car using a variety of features. Even though there are websites that offers this service, their prediction method may not be the best. Besides, different models and systems may contribute on predicting power for a used car's actual market value. It is important to know their actual market value while both buying and selling.

## 2.Approach followed by the business

Generally, Car Manufacturers vary the product for the price of cars with the available independent variables. That should help the management to understand how exactly the prices vary with the independent variables. They can accordingly manipulate the design of the cars, the business strategy etc. to meet certain price levels. As the car is made to meet multiple requirements, the prediction is taken under the consideration of multiple facilities and services provided by the Car Manufacturer. Majority of the car manufacturers use machine learning models to predict the car price using multiple features as inputs with the current trends in the society.

## 3.Skillsets, infrastructure and other impact on the business during implementation

The Skillset required is the understanding of the machine learning models, individual parts and overall functioning of each and every part of the automobile. Thorough understanding of the automobile industry and independent parts of the automobile.

The infrastructure required is the Storage, Computational power and the data to implement these models to predict the pricing.

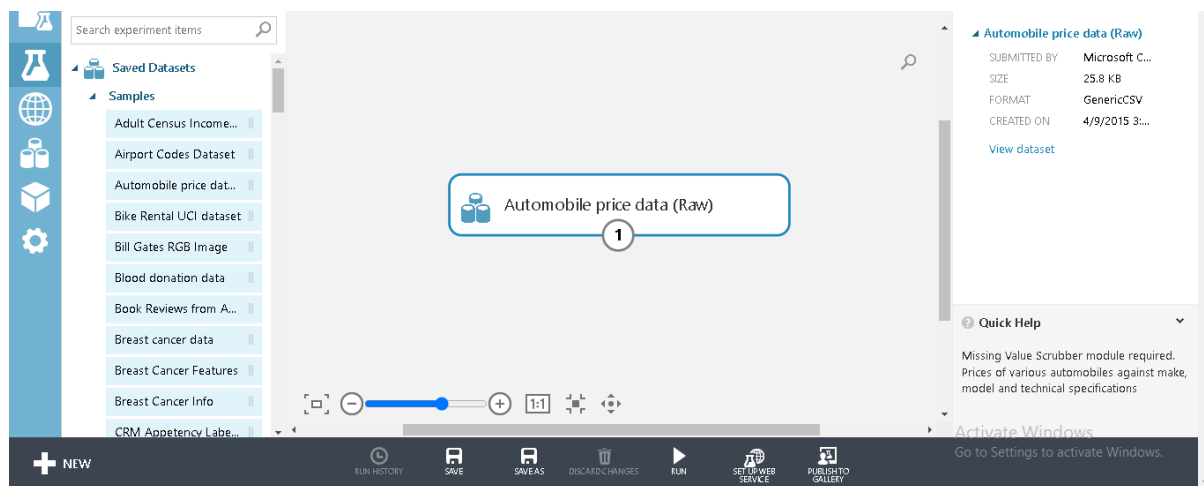
## 4. Similar approaches followed by other businesses

Majority Businesses follow a similar trend of prediction before date to improvise before-hand and produce good quality products. Another approach followed by similar businesses is to determine the brand pricing model of the Car Manufacturer and use it as its base to predict the sale. Current Trends, requirements and feature selection are also used by the car manufacturers and other businesses to predict the prices.

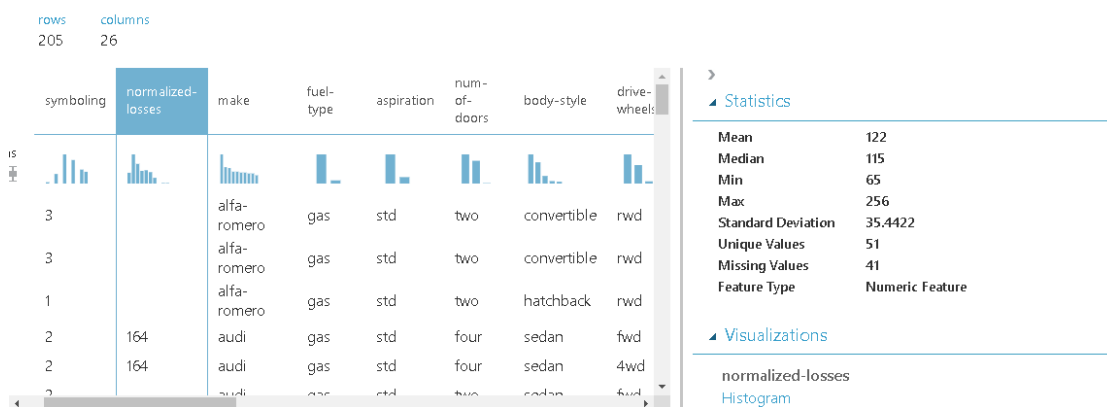
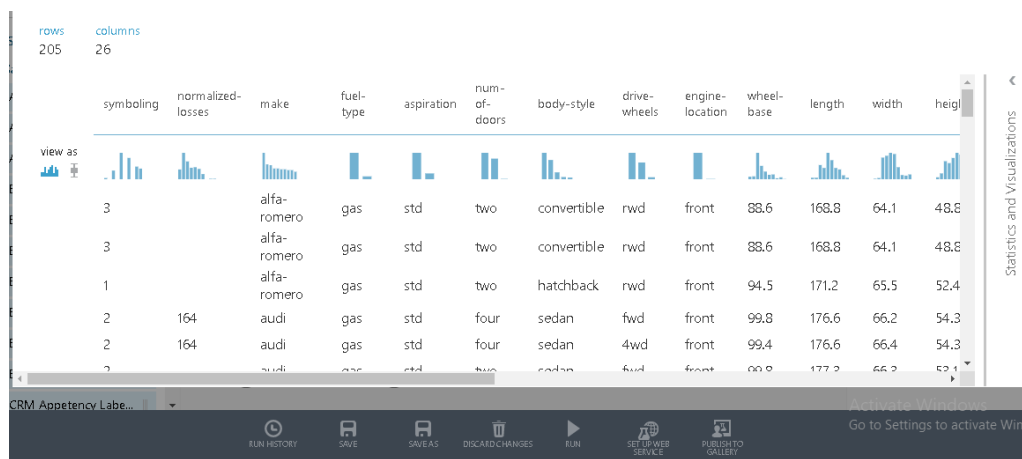
## 5. Follow Machine learning studio for developing an application

### Solution:

Step1: Select the dataset and get the data



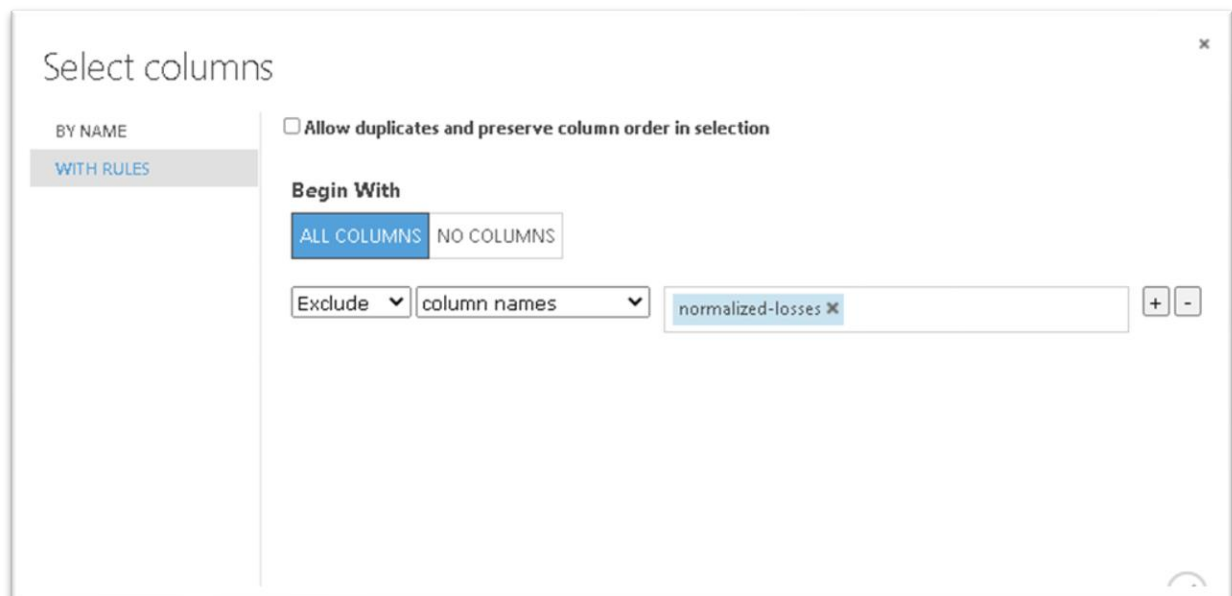
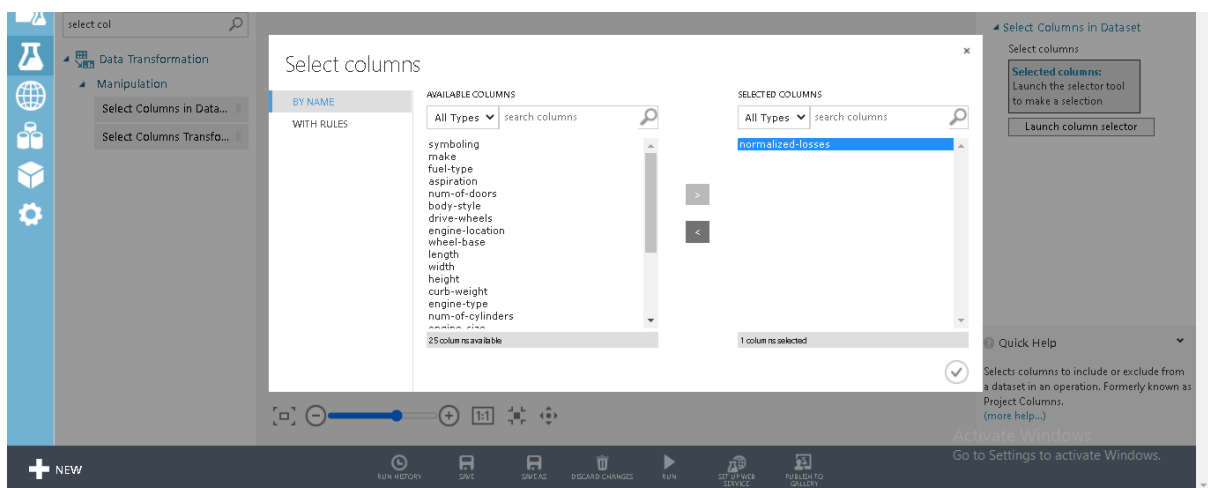
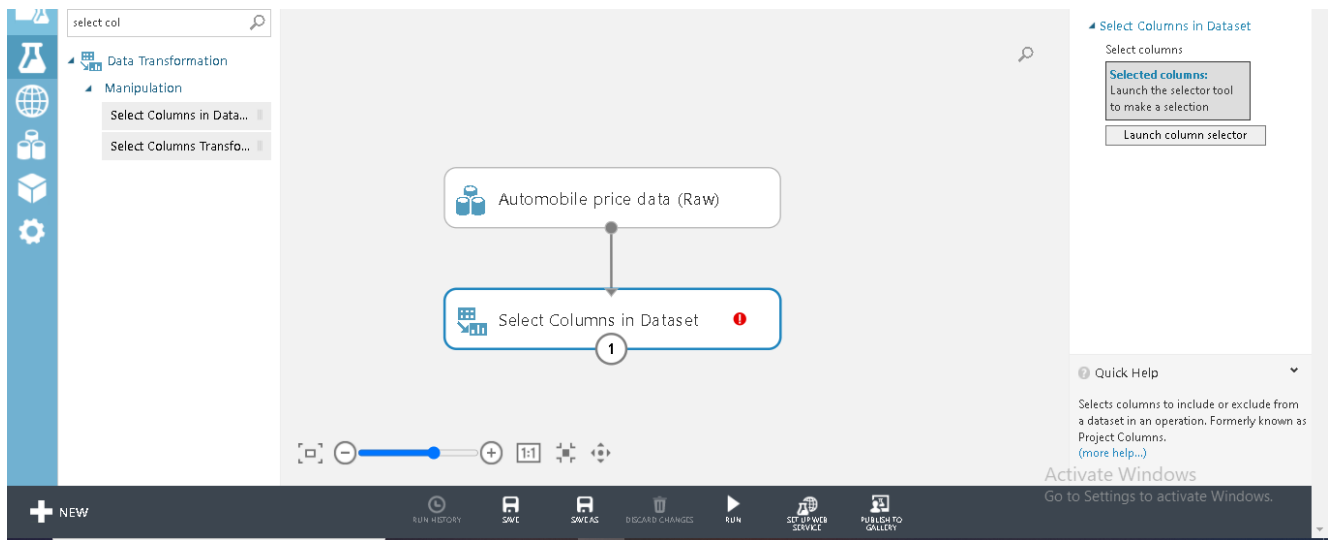
Step2: Visualize the dataset

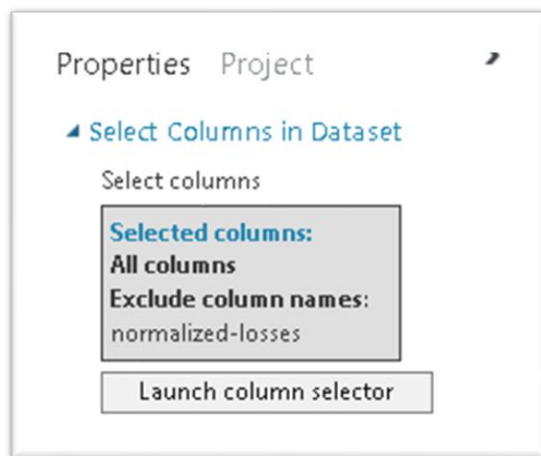


### Step 3: Preparing the data (selecting columns from the dataset)

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).

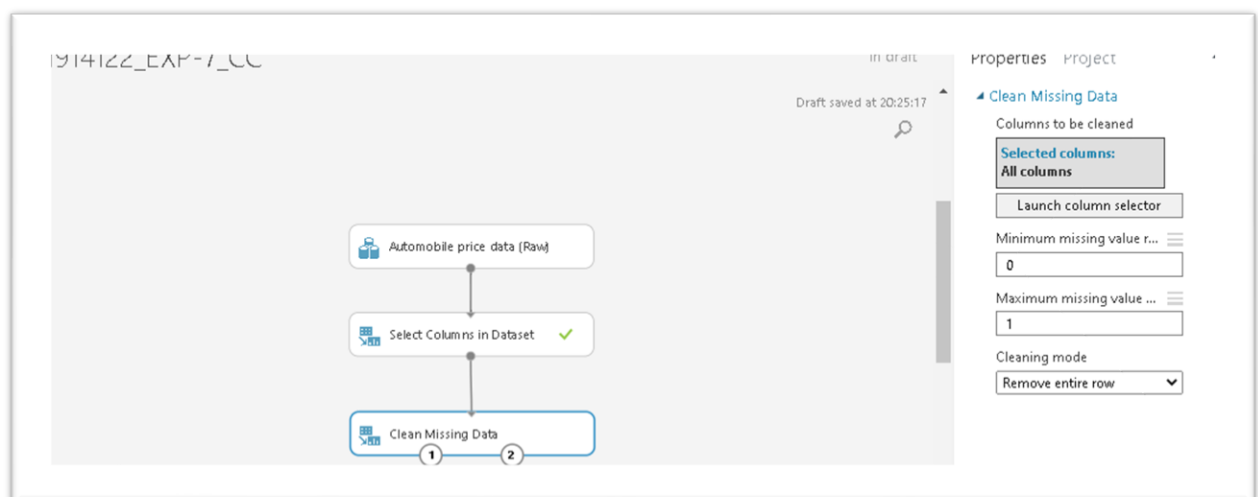




## Step 4: Preprocessing over the data

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. Double-click the module and type the comment "Remove missing value rows."



Properties
Project

Clean Missing Data

Columns to be cleaned

Selected columns:  
All columns

Launch column selector

Minimum missing value r...

0

Maximum missing value ...

1

Cleaning mode

Remove entire row

## Step 5: Define Features

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

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.

clean missing data

Data Transformation

Manipulation

Clean Missing Data

BY NAME

WITH RULES

Allow duplicates and preserve column order in selection

Begin With

ALL COLUMNS NO COLUMNS

Include column names

price engine-type fuel-system make  
body-style wheel-base peak-rpm  
horsepower

Select Columns in Dataset

Select columns

Selected columns:  
All columns  
Column names:  
normalized-losses

Launch column selector

Quick Help

Selects columns to include or exclude from a dataset in an operation. Formerly known as Project Columns.  
(more help...)

Activate Windows

Go to Settings to activate Windows.

NEW

RUN HISTORY

SAVE

SWAPS

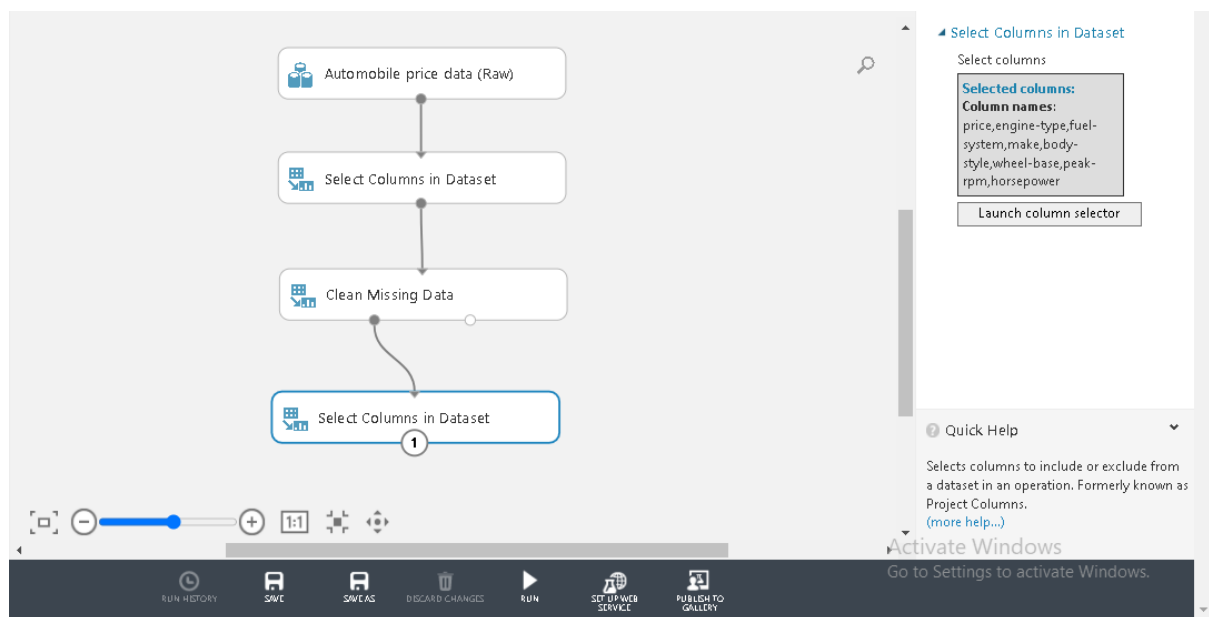
DISCARD CHANGES

RUN

STOP OPERATOR SERVICE

PUBLISH TO GALLERY

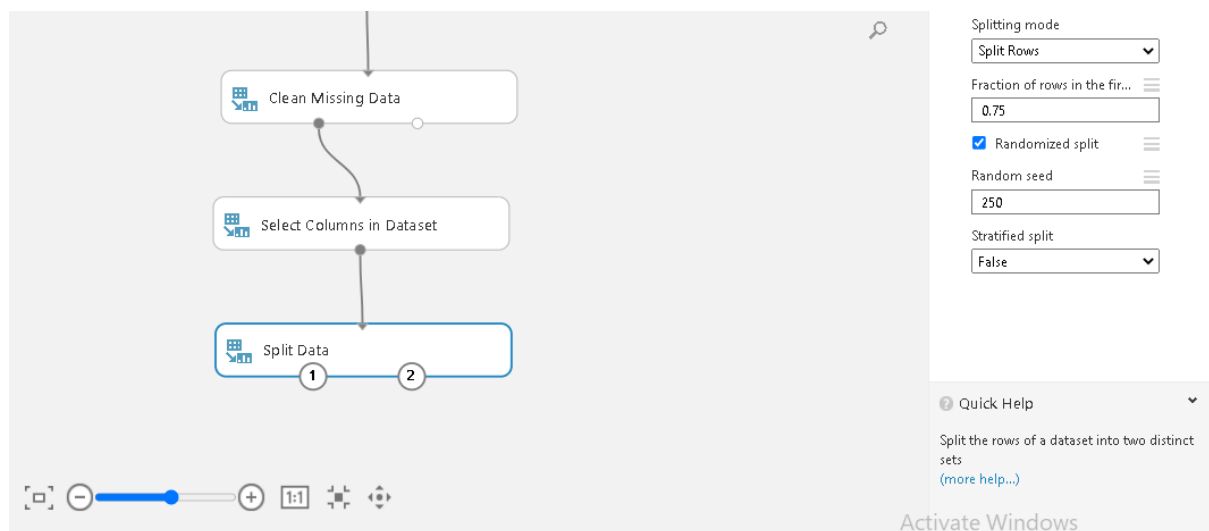




## Step 6: Split data

Select and drag the Split Data module to the experiment canvas and connect it to the last Select Columns in Dataset module.

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. And then set Random seed to 250.



### Split Data

Splitting mode

Split Rows 

Fraction of rows in the fir... 

0.75

☒ Randomized split Random seed 

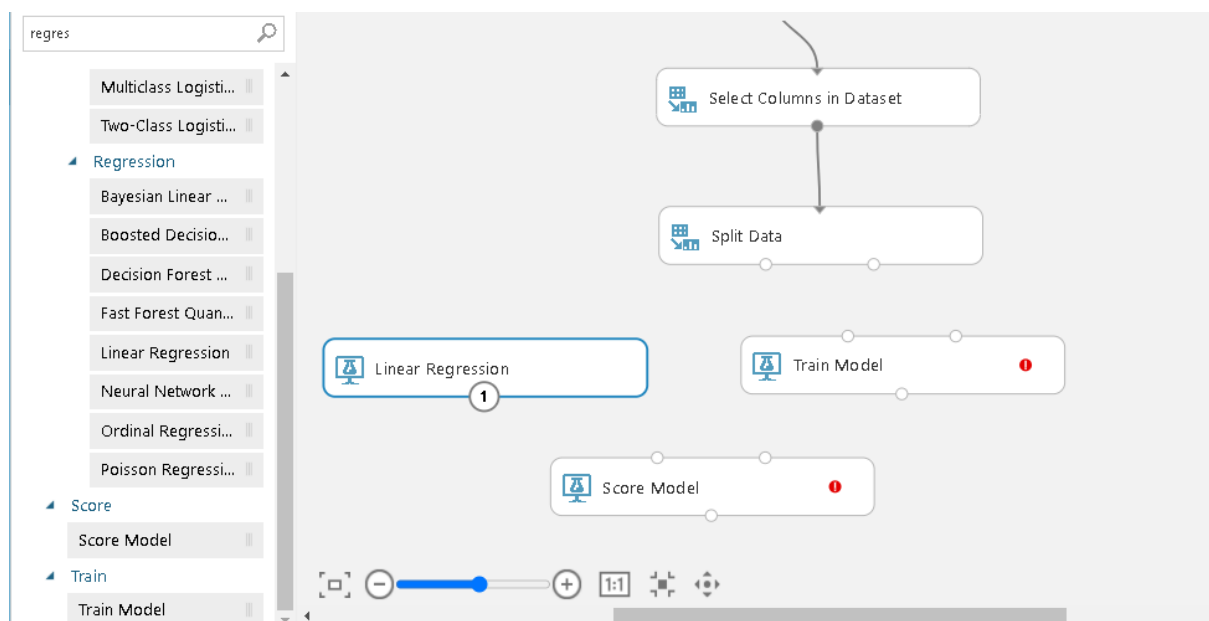
250

Stratified split

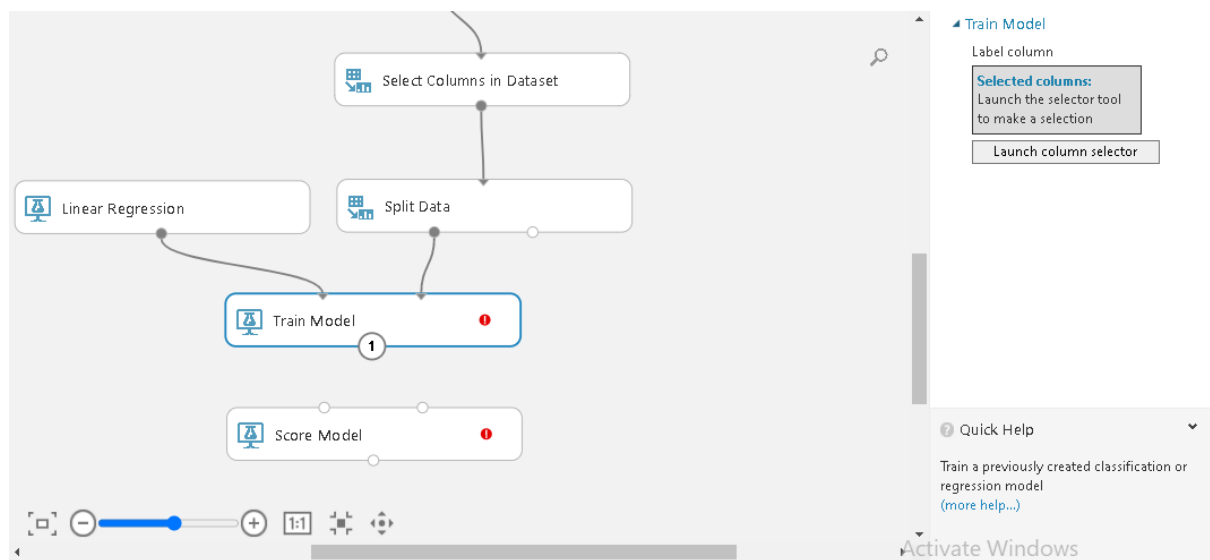
False 

## Step 7: Choose an Algorithm

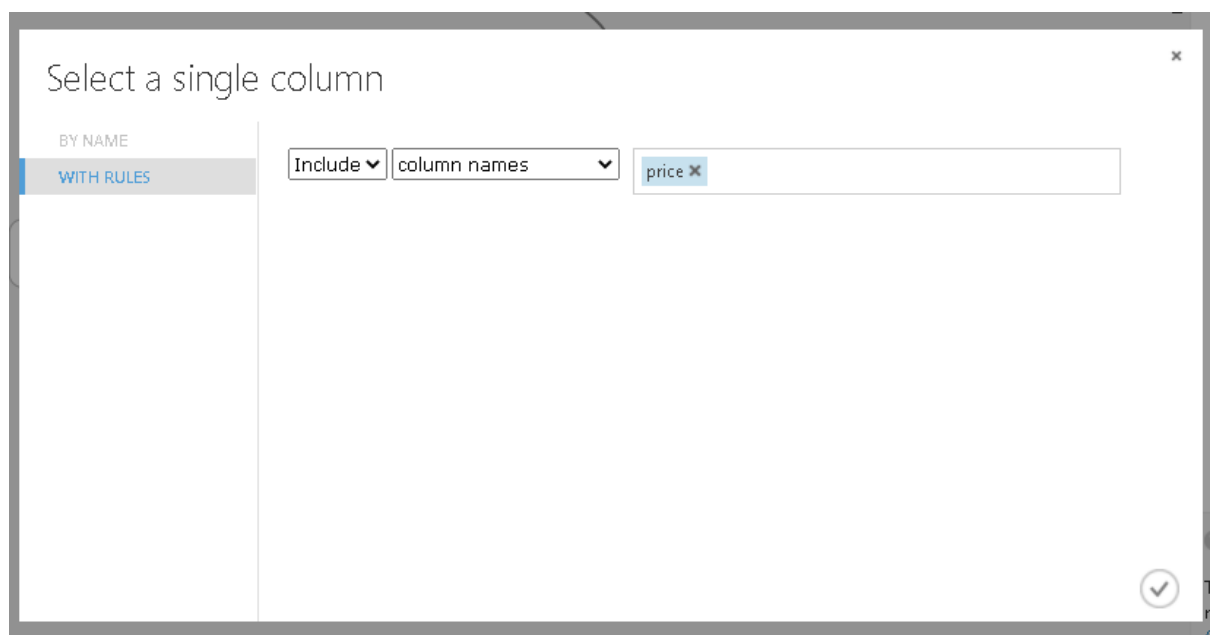
Select the Linear Regression module under the Regression category, and drag it to the experiment canvas



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.



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.



Properties Project

#### Train Model

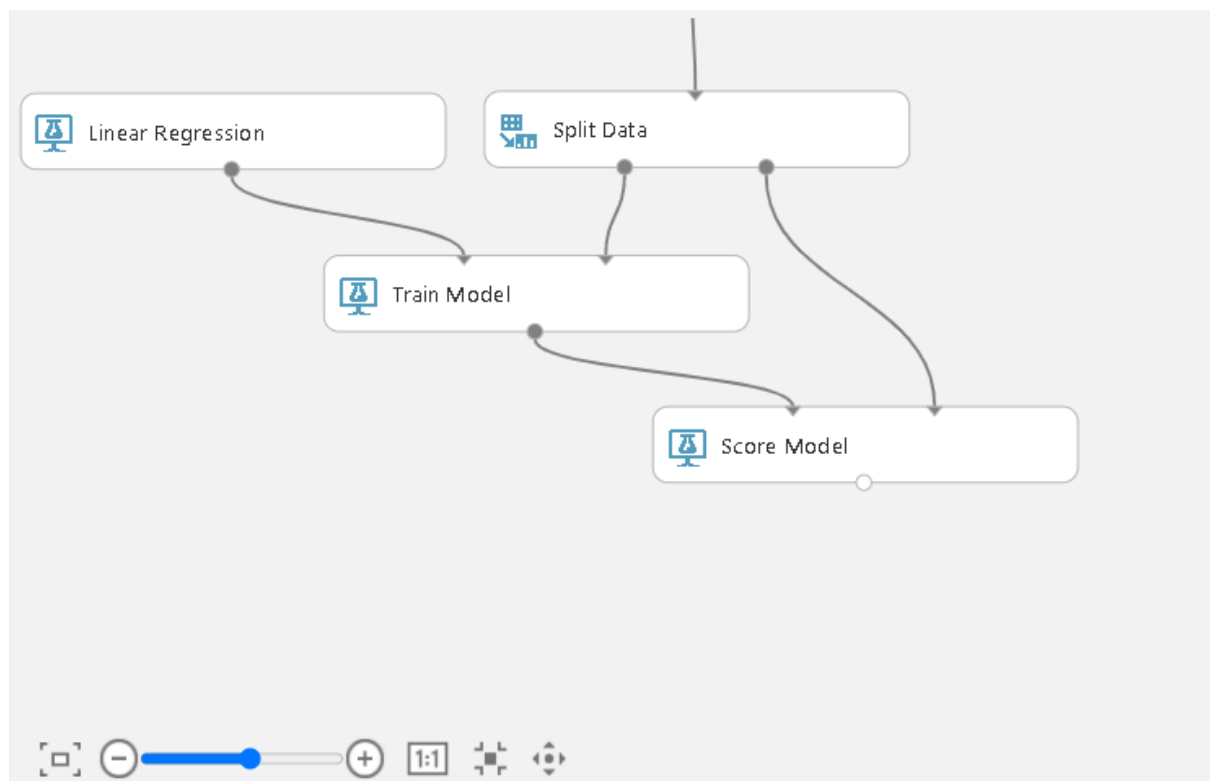
Label column

**Selected columns:**  
Column names: price

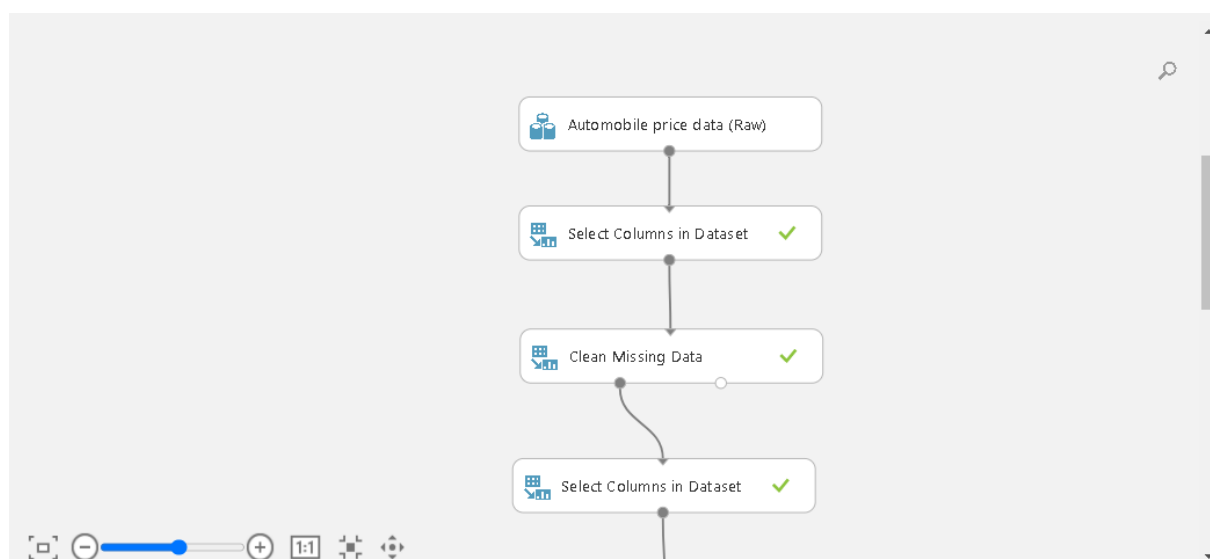
Launch column selector

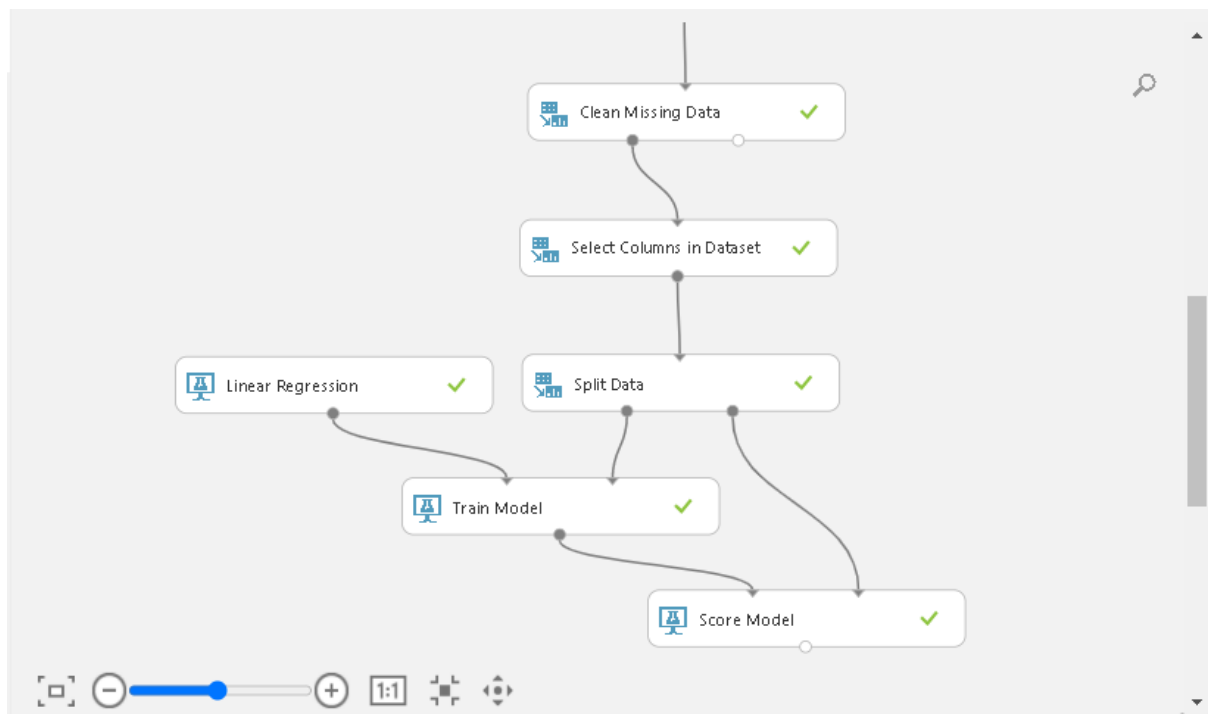
## Step 8: Predict new prices

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.



## Step 9: Run the model





## Step 10: Visualize the score and the predicted score

The screenshot shows the 'Score Model' node selected, with a context menu open. The menu options are: Delete, Copy, Cut, Paste, Scored dataset (selected), View Log, Edit Comment, Run selected, and Help. The 'Scored dataset' sub-menu is open, showing options: Download, Save as Dataset, Save as Trained Model, Save as Transform, Visualize (highlighted), and Generate Data Access Code... The 'Visualize' option is the target for Step 10. On the right, a 'Score Model' panel displays metadata: START TIME (1/26/2022 8...), END TIME (1/26/2022 8...), ELAPSED TIME (0:00:01.860), STATUS CODE (Finished), and STATUS DETAILS (None). A 'View output log' link is also present. The bottom toolbar includes icons for Run History, Save, Save As, Discard Changes, Run, Set Up Web Service, and Publish to Gallery. An 'Activate Windows' watermark is visible at the bottom.

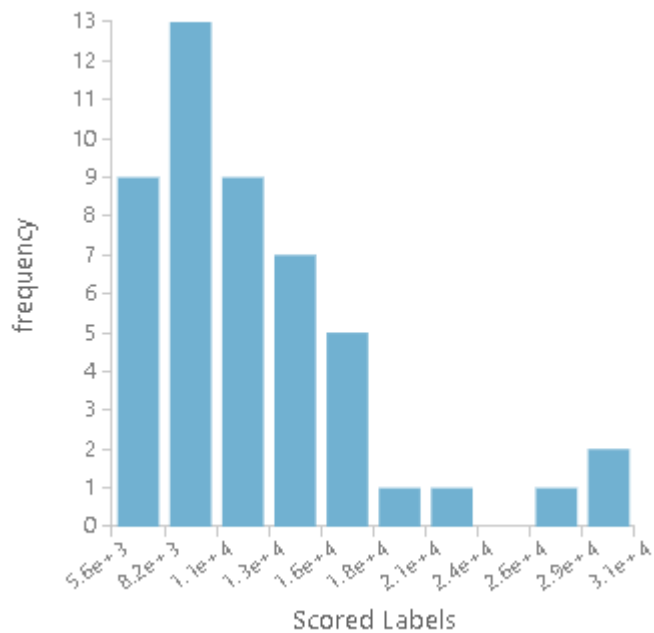
rows  
48

columns  
9

	make	body-style	wheel-base	engine-type	fuel-system	horsepower	peak-rpm	price	Scored Labels
view as									
	toyota	hatchback	102.4	ohc	mpfi	92	4200	9988	12256.369643
	alfa-romero	convertible	88.6	dohc	mpfi	111	5000	16500	13375.416626
	volkswagen	sedan	100.4	ohc	idi	68	4500	13845	10197.449813
	mazda	hatchback	93.1	ohc	2bbl	68	5000	6095	7627.592396
	toyota	hardtop	98.4	ohc	mpfi	116	4800	8449	15765.87598
	isuzu	hatchback	96	ohc	spfi	90	5000	11048	10634.530319
	volkswagen	sedan	97.3	ohc	mpfi	85	5250	7975	9317.749017
	volkswagen	sedan	100.4	ohc	mpfi	110	5500	13295	12419.449042
	subaru	wagon	96.9	ohcf	2bbl	82	4800	8013	8464.665942
	mazda	hatchback	98.8	ohc	2bbl	84	4800	10595	11159.504051
	toyota	hatchback	98.4	ohc	mpfi	116	4800	9989	12591.325507

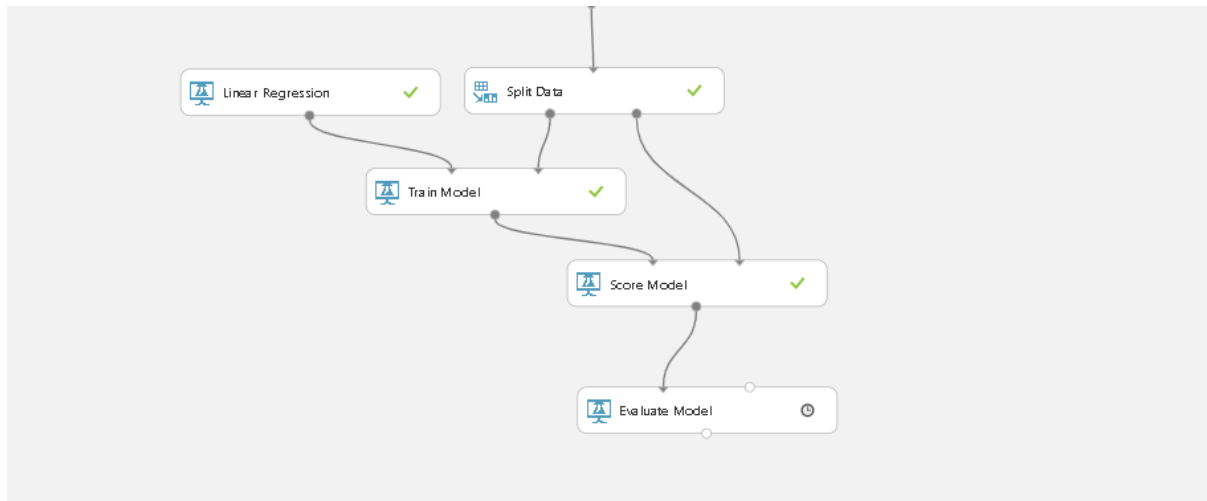
### Statistics

Mean	12866.2491
Median	11336.0762
Min	5618.0109
Max	31195.7762
Standard Deviation	6009.4892
Unique Values	45
Missing Values	0
Feature Type	Numeric Score



## Step 11: Evaluating the model and the quality of the result

Finally, we test the quality of the results. Select and drag the Evaluate Model module to the experiment canvas, and connect the output of the Score Model module to the left input of Evaluate Model.



**Evaluate Model**

START TIME	1/26/2022 8...
END TIME	1/26/2022 8...
ELAPSED TIME	0:00:01.829
STATUS CODE	Finished
STATUS DETAILS	None

[View output log](#)

Download  
Save as Dataset  
Save as Trained Model  
Save as Transform  
**Visualize**  
Generate Data Access Code...

Evaluates a scored classification or regression model with standard metrics ([more help...](#))

Activate Windows  
Go to Settings to activate Windows.

The following statistics are shown for our model:

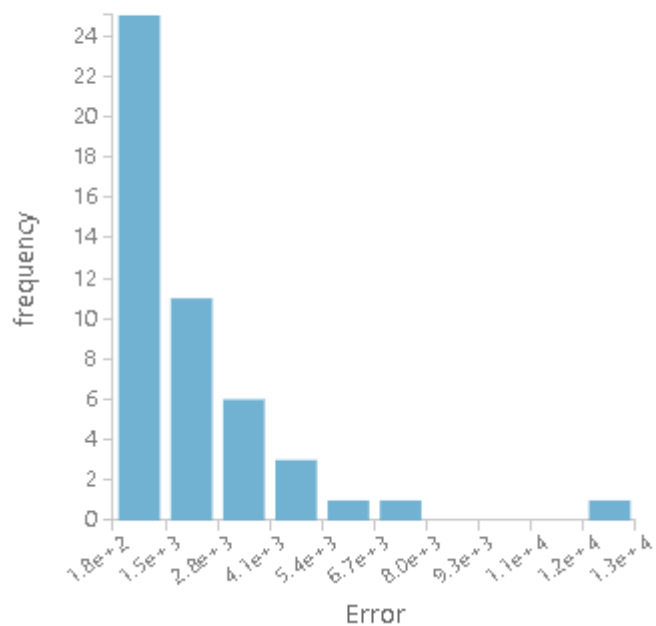
- **Mean Absolute Error (MAE):** The average of absolute errors (an *error* is the difference between the predicted value and the actual value).
- **Root Mean Squared Error (RMSE):** The square root of the average of squared errors of predictions made on the test dataset.
- **Relative Absolute Error:** The average of absolute errors relative to the absolute difference between actual values and the average of all actual values.
- **Relative Squared Error:** The average of squared errors relative to the squared difference between the actual values and the average of all actual values.
- **Coefficient of Determination:** Also known as the R squared value, this is a statistical

metric indicating how well a model fits the data.

#### Metrics

Mean Absolute Error	2082.639432
Root Mean Squared Error	3103.22062
Relative Absolute Error	0.378845
Relative Squared Error	0.163133
Coefficient of Determination	0.836867

#### Error Histogram



## Using the Custom Python Script

We've used the custom python script to predict the Cost per horsepower as a regression model



Microsoft Machine Learning Studio (classic) 1914122\_TY\_SINDHAVAD ...

Search experiment items

1914122

Finished running selected items

Properties Project

Experiment Properties

START TIME 4/19/202...  
END TIME 4/19/202...  
STATUS CODE Finished  
STATUS DETAILS None

Prior Run

Summary

Enter a few sentences describing your experiment (up to 140 characters).

Quick Help

Activate Windows  
Go to Settings to activate Windows.

NEW RUN HISTORY SAVE SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

Automobile price data (Raw)

Select Columns in Dataset

Execute Python Script

Clean Missing Data

Select Columns in Dataset

Split

Microsoft Machine Learning Studio (classic) 1914122\_TY\_SINDHAVAD ...

Search experiment items

1914122

Finished running selected items

Properties Project

Python script

```
1 import pandas as pd
2
3 def azureml_main(dataframe1 = None, dataframe2 = None):
4     dataframe1['Dollar/HP'] = dataframe1.price / dataframe1.horsepower
5     return dataframe1
```

CRM Appetency Labe...

Activate Windows  
Go to Settings to activate Windows.

NEW RUN HISTORY SAVE SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

Microsoft Machine Learning Studio (classic) 1914122\_TY\_SINDHAVAD ...

Search experiment items

1914122

Finished running selected items

Properties Project

1914122 > Execute Python Script > Results dataset

rows	columns
205	27

compression-ratio	horsepower	peak-rpm	city-mpg	highway-mpg	price	Dollar/HP
9	111	5000	21	27	13495	121.576577
9	111	5000	21	27	16500	148.648649
9	154	5000	19	26	16500	107.142857
10	102	5500	24	30	13950	136.764706
8	115	5500	18	22	17450	151.73913
9	110	5500	10	25	15250	129.626264

Statistics

Visualizations

To view, select a column in the table.

Activate Windows  
Go to Settings to activate Windows.

NEW RUN HISTORY SAVE SAVE AS DISCARD CHANGES RUN SET UP WEB SERVICE PUBLISH TO GALLERY

### Questions:

#### 1. Differentiate between linear and nonlinear regression

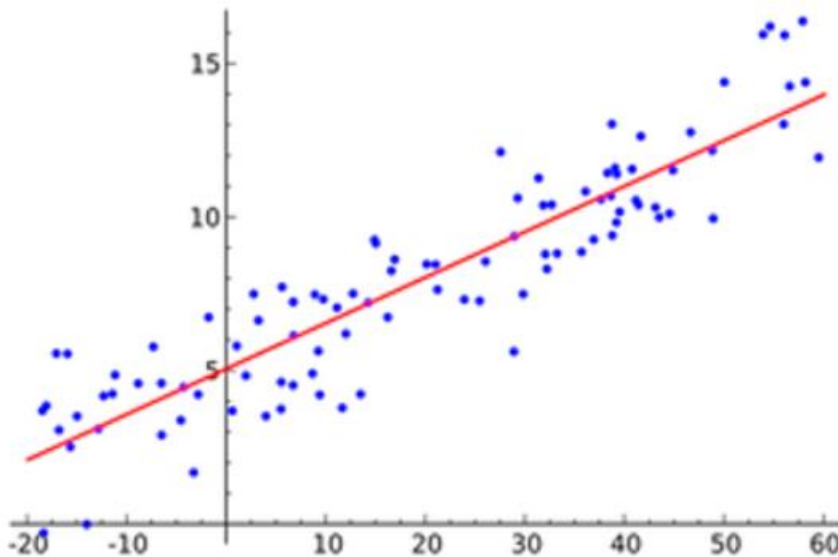
Ans.

### Linear regression:

Linear regression always uses a linear equation,  $Y = a + bx$ , where  $x$  is the explanatory variable and  $Y$  is the dependent variable. In multiple linear regression, multiple equations are added together but the parameters are still linear.

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \dots + \beta_{p-1} X_{i,p-1} + \varepsilon_i$$

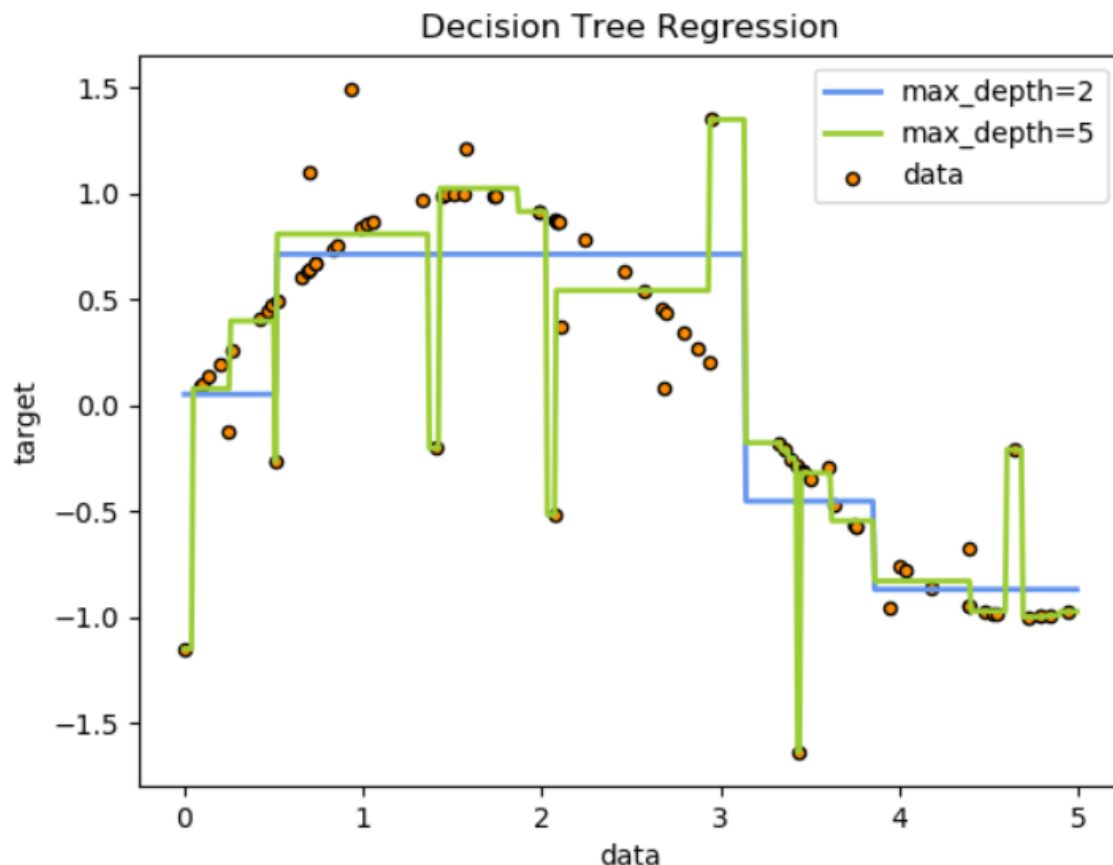
Following is the example of Linear Regression:



### Non Linear regression:

If the model equation does not follow the  $Y = a + bx$  form then the relationship between the dependent and independent variables will not be linear. There are many different forms of non-linear models. A random forest regression is considered a non-linear model. Random forest models are ensemble learning methods for regression which grow a forest of regression trees and then average the outcomes. This cannot be expressed as an equation. In regression trees, the splitting decision is based on minimizing the Residual Sum of Squares (RSS). The variable which has the greatest possible reduction in RSS is chosen as the root node. The tree splitting takes a top-down greedy approach, meaning the algorithm makes the best split at the current step rather than saving a split for better results on future nodes.

Following is the example of Nonlinear regression:



## 2. Write a note on converting non-linear model into linear model.

**Ans.**

Non-linear regression is a functional relationship that does not produce a straight-line in the scattered plot. Some non-linear curves can be transformed into linear regression. Following are the steps to convert a non-linear model to a linear model:

1. Firstly, you plot your data into scattered plot (XY type graph)
2. Examine if there is any non-linear relationship on the scattered plot
3. Guess the model that relate X and Y and transform the model into linear model
4. Compute the parameters and statistical fitness of the model
5. Transform back the parameter to non-linear model.
6. Generally, a logarithmic model is used to convert non linear model to a linear one.

## Outcomes:

**CO2:** Apply concepts of different types of Learning and Neural Network

## Conclusion: (Conclusion to be based on the objectives and outcomes achieved)

In this experiment we learnt about Azure ML Studio and how to create machine learning models using Multiple regression it. Azure ML Studio is a tool as it let us create complicated ML models without writing a single line of code. We implemented a Multiple linear

regression model for automobile dataset which helped us predict price of the automobiles given a set of multiple features from which the price is derived.

**Github Link:** <https://github.com/Adityasindhavad>

**Github Link:** [https://github.com/Adityasindhavad/ML\\_Exp7.git](https://github.com/Adityasindhavad/ML_Exp7.git)

**Grade: AA / AB / BB / BC / CC / CD /DD**

KJSCE/IT/LY-B.Tech/SEM-VII/ML/2022-23

**Signature of faculty in-charge with date**

**References:**

**Books/ Journals/ Websites:**

1. <https://blog.minitab.com/en/adventures-in-statistics-2/what-is-the-difference-between-linear-and-nonlinear-equations-in-regression-analysis>
2. <https://jenmckaig.medium.com/what-is-the-difference-between-linear-regression-and-non-linear-regression-e3b5981fcdc0>
3. <https://statisticsbyjim.com/regression/difference-between-linear-nonlinear-regression-models/>