



# CIS5560 Term Project Tutorial



**Authors:** [Sanjana Bodireddy](#); [Samyuktha Muralidharan](#); [Savita Yadav](#); [Rahbar Far Farnood](#)

**Instructor:** [Jongwook Woo](#)

**Date:** 05/21/2021

## Lab Tutorial

Sanjana Bodireddy ([sboddir@calstatela.edu](mailto:sboddir@calstatela.edu))

Samyuktha Muralidharan ([smurali2@calstatela.edu](mailto:smurali2@calstatela.edu))

Savita Yadav ([syadav5@calstatela.edu](mailto:syadav5@calstatela.edu))

Rahbar Far Farnood ([frahbar@calstatela.edu](mailto:frahbar@calstatela.edu))

05/21/2021

## Airbnb Predictive Analysis using Machine Learning Models in Azure ML Studio

---

### Objectives

The objective of this lab is to build a model that predicts the optimal price and rating of a property considering the features of the listings using the following machine learning algorithms:

#### Price Prediction

- Bayesian Linear Regression
- Decision Forest Regression
- Boosted Decision Tree Regression

## Rating Prediction

- Two-Class Boosted Decision Tree
- Two-Class Decision Forest
- Two-Class Logistic Regression

## Platform Specifications

- Microsoft Azure Machine Learning Studio
- Number of nodes: 1
- Total Memory Size: 10 GB

## Steps to create an experiment using ML studio:

- a) Data Preparation
- b) Train the model
- c) Evaluating the model

## Airbnb Price Prediction

## Bayesian Linear Regression

### a) Data Preparation

1. Open a browser and browse to <https://studio.azureml.net>. Then sign in using the Microsoft account associated with your Azure ML account.
2. Create a new blank experiment and give it the title **Airbnb Price Prediction**.
3. Upload the `airbnb_sample.csv` file and drag it to canvas
4. Search for the **Edit Metadata (Metadata Editor)** module and drag it onto the canvas.
5. Connect the output of the **Airbnb Dataset** to the **Dataset** input of the **Edit Metadata (Metadata Editor)**. Configure the properties of the **Edit Metadata (Metadata Editor)** as:
  - **Launch Column selector** and select following columns: Host Listings Count, Host Total Listings Count, Accommodates, Bathrooms, Bedrooms, Beds, Square Feet, Price, Weekly Price, Monthly Price, Security Deposit, Cleaning Fee, Guests Included, Extra People, Minimum Nights, Maximum Nights, Number of Reviews, Review Scores Rating, Review Scores Cleanliness, Review Scores Accuracy, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Sentiment, Reviews per month, Calculated host listings count.
  - **Data Type:** Integer
  - **Categorical:** Unchanged

- **Fields:** Unchanged

6. Search for the **Select Columns in Dataset** module and drag it onto the canvas.
7. Connect the output of the **Edit Metadata** to the input of the **Select Columns in Dataset**
8. In the properties pane of the **Select Columns in Dataset**

- **Launch Column selector** and select the column names: Neighborhood, Latitude, Longitude, Property Type, Room Type, Accommodates, Bathrooms, Bedrooms, Bed Type, Price, Monthly Price, Security Deposit, Cleaning Fee, Guests Included, Extra People, Minimum Nights, Review Scores Accuracy, Review Scores Location, Review Scores Value, Sentiment.

9. Search for **Clip values** and drag it on to the canvas.

10. Connect the input port of **clip values** to the output port of **Select Columns in Dataset**. Configure the properties of the **Clip values** as:

- **Set of thresholds:** ClipPeaksandSubPeaks
- **Threshold:** Percentile
- **Percentile number of upper threshold:** 90
- **Percentile number of lower threshold:** 10
- **Substitute value for peaks:** Missing
- **Substitute value for subpeaks:** Missing
- **List of columns and** select column names: Price, Accommodates, Monthly Price, Security Deposit, Cleaning Fee, Guests Included, Extra People, Minimum Nights, Bedrooms, Bathrooms.

11. Search for **Clean Missing Data** and drag it on to canvas and connect input port of **Clean Missing Data** to the output port of the clip values.

12. Configure the properties of the **Clean Missing Data** as:

- **Launch Column selector:** Select all columns
- **Minimum missing value ratio:** 0
- **Maximum missing value ratio:** 1
- **Cleaning mode:** Remove entire row

13. Connect the output of the **Clean Missing Dataset** to the input of the **Edit Metadata (Metadata Editor)**. Configure the properties of the **Edit Metadata (Metadata Editor)** as:

- **Launch Column selector** and select the column names: Property Type, Bed Type, Neighborhood, Room Type
- **Data Type:** Integer
- **Categorical:** Unchanged
- **Fields:** Unchanged

14. Search for **Normalize Data** module and drag it on to canvas.  
15. Connect the input port of Normalize Data to the Output Port of the Edit Metadata (Metadata Editor). Configure the properties of the **Normalize Data** as:

- **Transformation method:** MinMax
- **Columns to transform:** Column type: Numeric, All  
Exclude column names: Price

16. It would appear as figure given below:



## b) Train the Model

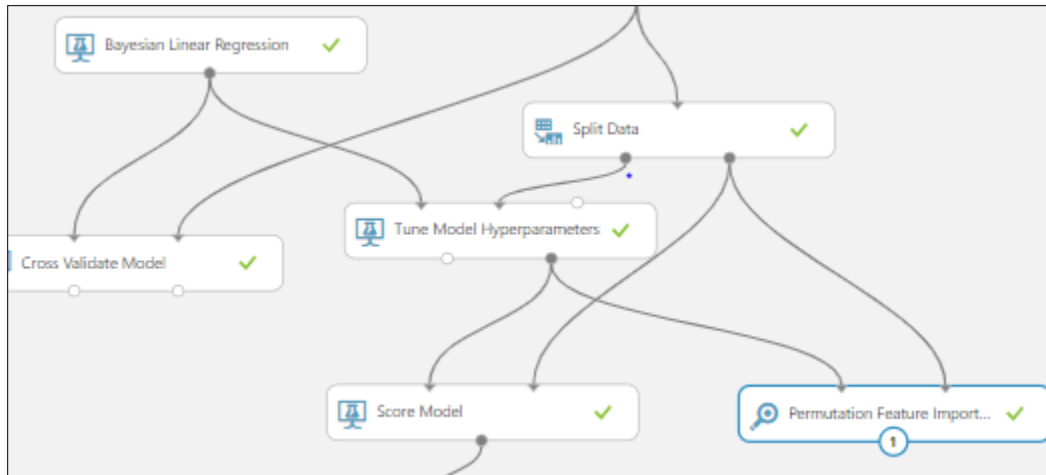
Now the data is prepared, we must train the model.

1. Search for the **Split Data (Split)** module and drag it onto the Canvas.
2. Connect the output of the **Normalize Data** module to the input of the **Split Data (Split)** module.
3. On the properties pane of the **Split Data (Split)** module, configure the properties as shown below:

- **Splitting mode:** Split Rows
- **Fraction of rows in the first output dataset:** 0.7
- **Random seed:** 1234
- **Stratified split:** False

4. Search for the **Tune Model Hyperparameters** module and drag it onto the canvas.

5. Connect the **Results dataset** (left) output of the **Split Data (Split) module** to the input of the **Tune Model Hyperparameters** (right) module.
6. On the properties pane for the **Tune Model Hyperparameters** module, configure the properties as follows:
  - **Specify parameter sweeping mode:** Random Sweep
  - **Maximum number of runs on random sweep:** 40
  - **Random seed:** 4567
  - **Label column:** Price
  - **Metric for measuring performance for classification:** Accuracy
  - **Metric for measuring performance for regression:** Coefficient of determination
7. Search for **Score Model** and drag it on to the canvas.
8. Connect the **Results dataset2** (right) output of the **Split Data (Split) module** to the right input of the **Score Model** and to the right input of the **Permutation Feature Importance**.
9. Search for the **Bayesian Linear Regression** module and drag it onto the canvas. Connect output port of **Bayesian Linear Regression to the input port of Tune Model Hyperparameters(right)** and Set the property as:
  - **Regularization weight:** 4
10. Search for the **Permutation Features Importance** model and drag it onto the canvas. Configure the properties of **Permutation Features Importance** model as:
  - **Random Seed:** 1234
  - **Metric for measuring performance:** Regression-Coefficient of Determination
11. **Connect the Tune Model Hyperparameters** module output (right) to the input ports of **Score Model(left)** and **Permutation Feature Importance(right)**.
12. Search for the **Cross Validate Model** and drag it onto the canvas.
13. Connect the output of **Bayesian Linear Regression** model to the left inputs of the **Cross Validate Model** and connect output port of **Normalize Data** module to the right input port of **Cross Validate Model**. Configure the properties of the **Cross Validate Model** as:
  - **Label column:** Price
  - **Random seed:** 1234
14. It would appear as given below:



### c) Evaluating the Model

1. Search for the **Evaluate** module and drag it onto the canvas.
2. Connect the left input of the Evaluate model from the output of Score Model.
3. It should appear as given below



4. Save and run the experiment.
5. When the experiment has finished, Visualize the output form the **Evaluate** module.

rows	columns					
1	6					
		Negative Log Likelihood	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error
						Coefficient of Determination
view as						
		2306.056582	21.414341	29.248719	0.514505	-0.327992
						0.672008

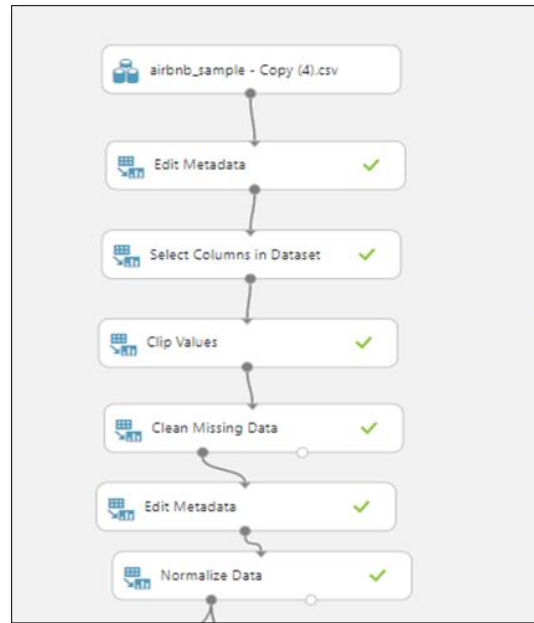
# Decision Forest Regression

## a) Data Preparation

1. Open a browser and browse to <https://studio.azureml.net>. Then sign in using the Microsoft account associated with your Azure ML account.
2. Create a new blank experiment and give it the title **Airbnb Price Prediction**.
3. Upload the `airbnb_sample.csv` file and drag it to canvas
4. Search for the **Edit Metadata (Metadata Editor)** module and drag it onto the canvas.
5. Connect the output of the **Airbnb Dataset** to the **Dataset** input of the **Edit Metadata (Metadata Editor)**. Configure the properties of the **Edit Metadata (Metadata Editor)** as:
  - **Launch Column selector** and select the columns: Host Listings Count, Host Total Listings Count, Accommodates, Bathrooms, Bedrooms, Beds, Square Feet, Price, Weekly Price, Monthly Price, Security Deposit, Cleaning Fee, Guests Included, Extra People, Minimum Nights, Maximum Nights, Number of Reviews, Review Scores Rating, Review Scores Cleanliness, Review Scores Accuracy, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Sentiment, Reviews per month, Calculated host listings count
  - **Data Type:** Integer
  - **Categorical:** Unchanged
  - **Fields:** Unchanged
6. Search for the **Select Columns in Dataset** module and drag it onto the canvas.
7. Connect the output of the **Edit Metadata** to the input of the **Select Columns in Dataset**
8. In the properties pane of the **Select Columns in Dataset**
  - Configure the properties of **Select Columns in Dataset** as:
    - **Launch Column selector** and select the columns: Price, Monthly Price, Host Listings Count, Host Total Listings Count, Longitude, Property Type, Room Type, Bed Type, Accommodates, Bathrooms, Bedrooms, Guests Included, Extra People, Review Scores Rating, Review Scores Accuracy, Review Scores Cleanliness, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Security Deposit, Cleaning Fee, Sentiment
9. Search for **Clip values** and drag it on to the canvas.
10. Connect the input port of **clip values** to the output port of **Select Columns in Dataset**. Configure the properties of the **Clip values** as:
  - **Set of thresholds:** ClipPeaksandSubPeaks

- **Threshold:** Percentile
  - **Percentile number of upper threshold:** 90
  - **Percentile number of lower threshold:** 10
  - **Substitute value for peaks:** Missing
  - **Substitute value for subpeaks:** Missing
  - **List of columns and** select column names: Price, Host Listings Count, Host Total Listings Count, Accommodates, Bathrooms, Bedrooms, Monthly Price, Security Deposit, Cleaning Fee, Guests Included, Extra People
11. Search for **Clean Missing Data** and drag it on to canvas and connect input port of **Clean Missing Data** to the output port of the clip values. Configure the properties of the **Clean Missing Data** as:
- **Launch Column selector:** Select all columns
  - **Minimum missing value ratio:** 0
  - **Maximum missing value ratio:** 1
  - **Cleaning mode:** Remove entire row
12. Connect the output of the **Clean Missing Dataset** to the input of the **Edit Metadata (Metadata Editor)**. Configure the properties of the **Edit Metadata (Metadata Editor)** as:
- **Launch Column selector** and select the column names: Property Type, Bed Type, Room Type
  - **Data Type:** Integer
  - **Categorical:** make categorical
  - **Fields:** Unchanged
13. Search for **Normalize Data** and drag it on to canvas.
14. Connect the input port of Normalize Data to the Output Port of the Edit Metadata (Metadata Editor). Configure the properties of the **Normalize Data** as:
- **Transformation method:** MinMax
  - **Columns to transform:** Column type: Numeric, All, Exclude column names: Price
15. It would appear as figure given below:



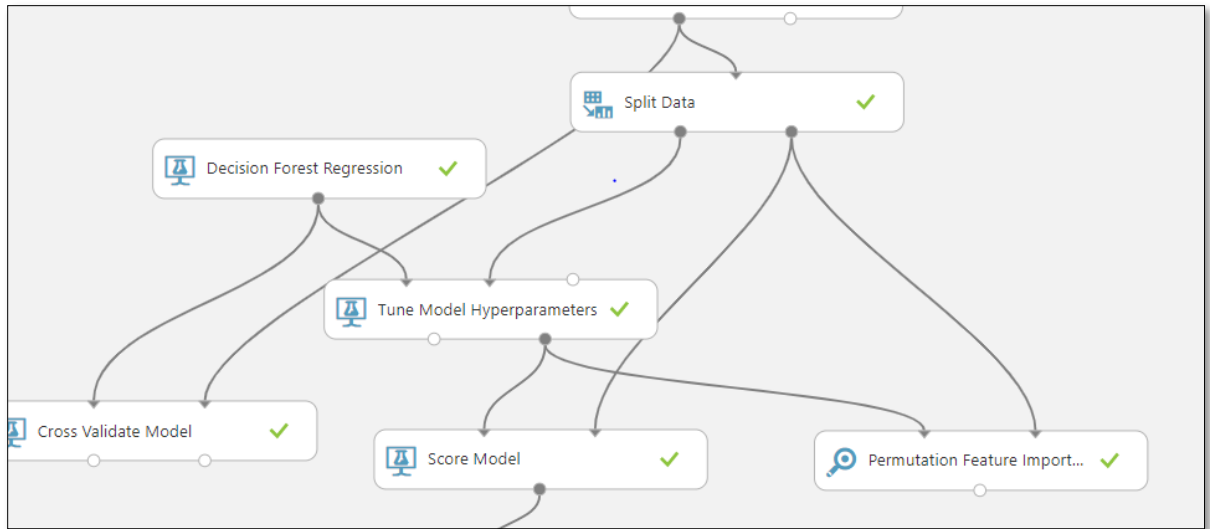


Note: Now the data is prepared, we must train the model.

## b) Train the model

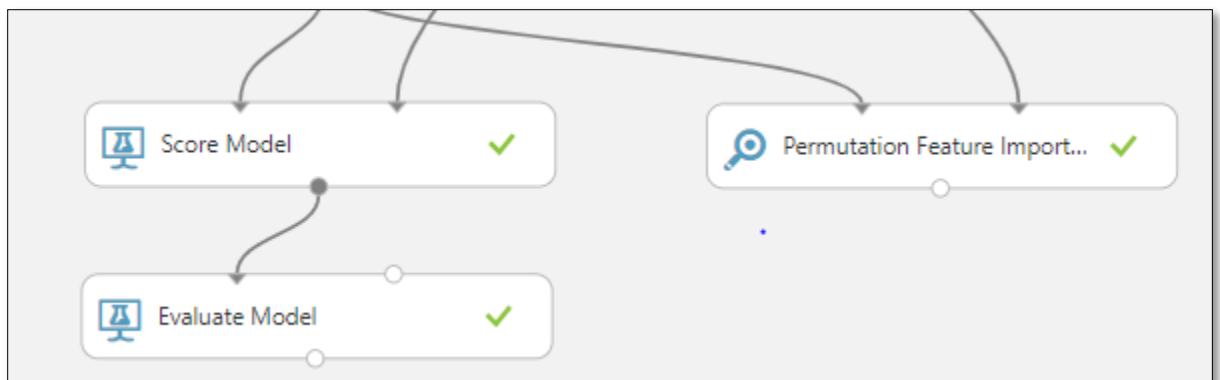
1. Search for the **Split Data (Split)** module and drag it onto the Canvas.
2. Connect the output of the **Normalize Data** module to the input of the **Split Data (Split)** module.
3. On the properties pane of the **Split Data (Split)** module, configure the properties as shown below:
  - **Splitting mode:** Split Rows
  - **Fraction of rows in the first output dataset:** 0.7
  - **Random seed:** 1234
  - **Stratified split:** False
4. Search for the **Tune Model Hyperparameters** module and drag it onto the canvas.
5. Connect the **Results dataset** (left) output of the **Split Data (Split)** module to the input of the **Tune Model Hyperparameters** (right) module.
6. On the properties pane for the **Tune Model Hyperparameters** module, configure the properties as follows:
  - **Specify parameter sweeping mode:** Random Sweep
  - **Maximum number of runs on random sweep:** 20
  - **Random seed:** 4567
  - **Label column:** Price
  - **Metric for measuring performance for classification:** Accuracy

- **Metric for measuring performance for regression:** Coefficient of determination
7. Search for **Score Model** and drag it on to the canvas.
  8. Connect the **Results dataset2** (right) output of the **Split Data (Split) module** to the right input of the **Score Model** and to the right input of the **Permutation Feature Importance**.
  9. Search for the **Decision Forest Regression** module and drag it onto the canvas. Connect output port of **Decision Forest Regression** to the input port of **Tune Model Hyperparameters(right)** and Set the property as:
    - **Resampling method single parameter:** Bagging
    - **Create trainer mode:** Single Parameter
    - **Number of decision trees:** 10
    - **Maximum depth of the decision trees:** 50
    - **Number of random splits per node:** 128
    - **Minimum number of samples per leaf node:** 3
    - **Allow unknown values for categorical features:** Unchecked
  10. Search for the **Permutation Features Importance** model and drag it onto the canvas. Configure the properties of **Permutation Features Importance** model as:
    - **Random Seed:** 1234
    - **Metric for measuring performance:** Regression-Coefficient of Determination
  11. Connect the **Tune Model Hyperparameters** module output (right) to the input ports of **Score Model(left)** and **Permutation Feature Importance(right)**.
  12. Search for the **Cross Validate Model** and drag it onto the canvas.
  13. Connect the output of **Decision Forest Regression** model to the left inputs of the **Cross Validate Model** and connect output port of **Normalize Data** module to the right input port of **Cross Validate Model**. Configure the properties of the **Cross Validate Model** as:
    - **Label column:** Price
    - **Random seed:** 1234
  14. It would appear as given below:



### c) Evaluating the Model

1. Search for the **Evaluate** module and drag it onto the canvas.
2. Connect the left input of the Evaluate model from the output of Score Model.
3. It should appear as given below



4. Save and run the experiment.
5. When the experiment has finished, Visualize the output form the **Evaluate** module.

	Negative Log Likelihood	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
view as						
	2655.924668	23.766136	32.411958	0.573994	0.39695	0.60305

# Boosted Decision Tree Regression

## a. Data Preparation

1. Open a browser and browse to <https://studio.azureml.net>. Then sign in using the Microsoft account associated with your Azure ML account.
2. Create a new blank experiment and give it the title **Airbnb Price Prediction**.
3. Upload the `airbnb_sample.csv` file and drag it to canvas
4. Search for the **Edit Metadata (Metadata Editor)** module and drag it onto the canvas.
5. Connect the output of the **Airbnb Dataset** to the **Dataset** input of the **Edit Metadata (Metadata Editor)**. Configure the properties of the **Edit Metadata (Metadata Editor)** as:
  - **Launch Column selector** and select the columns: Host Listings Count, Host Total Listings Count, Accommodates, Bathrooms, Bedrooms, Beds, Square Feet, Price, Weekly Price, Monthly Price, Security Deposit, Cleaning Fee, Guests Included, Extra People, Minimum Nights, Maximum Nights, Number of Reviews, Review Scores Rating, Review Scores Cleanliness, Review Scores Accuracy, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Sentiment, Reviews per month, Calculated host listings count
  - **Data Type:** Integer
  - **Categorical:** Unchanged
  - **Fields:** Unchanged
6. Search for the **Select Columns in Dataset** module and drag it onto the canvas.
7. Connect the output of the **Edit Metadata** to the input of the **Select Columns in Dataset**
8. In the properties pane of the **Select Columns in Dataset**. Configure the properties of **Select Columns in Dataset** as:
  - **Launch Column selector** and select the columns: Price, Monthly Price, Host Listings Count, Host Total Listings Count, Longitude, Property Type, Room Type, Bed Type, Accommodates, Bathrooms, Bedrooms, Guests Included, Extra People, Review Scores Rating, Review Scores Accuracy, Review Scores Cleanliness, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Security Deposit, Cleaning Fee, Sentiment.
9. Search for **Clip values** and drag it on to the canvas.
10. Connect the input port of **clip values** to the output port of **Select Columns in Dataset**. Configure the properties of the **Clip values** as:
  - **Set of thresholds:** ClipPeaksandSubPeaks
  - **Threshold:** Percentile

- **Percentile number of upper thresholds:** 90
  - **Percentile number of lower thresholds:** 10
  - **Substitute value for peaks:** Missing
  - **Substitute value for subpeaks:** Missing
  - **List of columns and** select column names: Price, Accommodates, Bathrooms, Bedrooms, Monthly Price, Security Deposit, Cleaning Fee, Guests Included, Extra People, Minimum Nights
11. Search for **Clean Missing Data** and drag it on to canvas and connect input port of **Clean Missing Data** to the output port of the clip values. Configure the properties of the **Clean Missing Data** as:
- **Launch Column selector:** Select all columns
  - **Minimum missing value ratio:** 0
  - **Maximum missing value ratio:** 1
  - **Cleaning mode:** Remove entire row
12. Connect the output of the **Clean Missing Dataset** to the input of the **Edit Metadata (Metadata Editor)**. Configure the properties of the **Edit Metadata (Metadata Editor)** as:
- **Launch Column selector** and select the column names: Bed Type, Neighborhood, Room Type, Property Type
  - **Data Type:** Integer
  - **Categorical:** make categorical
  - **Fields:** Unchanged
13. Search for Normalize Data and drag it on to canvas.
14. Connect the input port of Normalize Data to the Output Port of the Edit Metadata (Metadata Editor). Configure the properties of the **Normalize Data** as:
- **Transformation method:** MinMax
  - **Columns to transform:** Column type: Numeric, All and Exclude column names: Price
15. It would appear as figure given below:

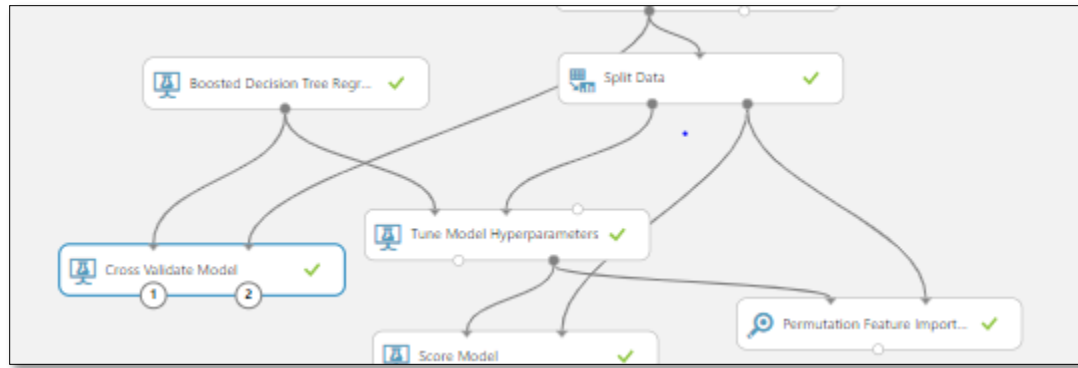


## b) Train the Model

Now the data is prepared, we must train the model.

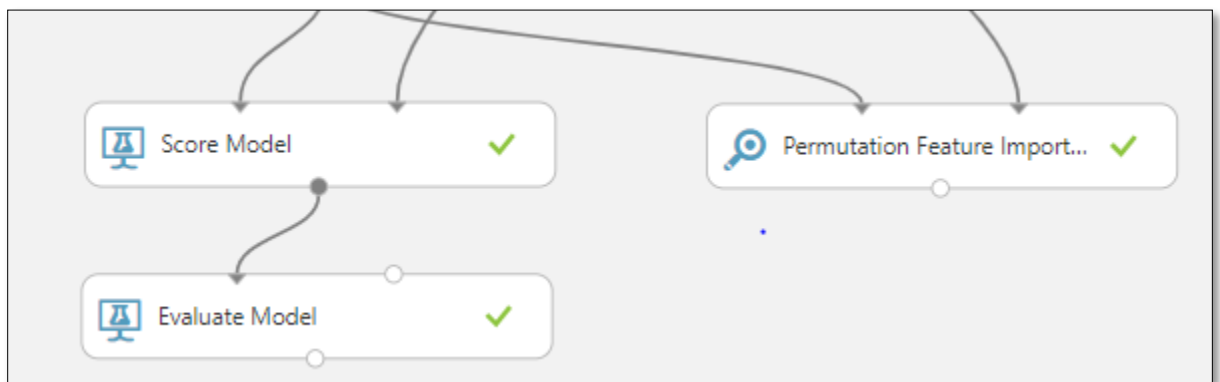
1. Search for the **Split Data (Split)** module and drag it onto the Canvas.
2. Connect the output of the **Normalize Data** module to the input of the **Split Data (Split)** module.
3. On the properties pane of the **Split Data (Split)** module, configure the properties as shown below:
  - **Splitting mode:** Split Rows
  - **Fraction of rows in the first output dataset:** 0.7
  - **Random seed:** 1234
  - **Stratified split:** False
4. Search for the **Tune Model Hyperparameters** module and drag it onto the canvas.
5. Connect the **Results dataset** (left) output of the **Split Data (Split)** module to the input of the **Tune Model Hyperparameters** (right) module.
6. On the properties pane for the **Tune Model Hyperparameters** module. Configure the properties as follows:
  - **Specify parameter sweeping mode:** Random Sweep
  - **Maximum number of runs on random sweep:** 10
  - **Random seed:** 4567
  - **Label column:** Price

- **Metric for measuring performance for classification:** Accuracy
  - **Metric for measuring performance for regression:** Coefficient of determination
7. Search for **Score Model** and drag it on to the canvas.
  8. Connect the **Results dataset2** (right) output of the **Split Data (Split) module** to the right input of the **Score Model** and to the right input of the **Permutation Feature Importance**.
  9. Search for the **Boosted Decision tree Regression** module and drag it onto the canvas. Connect output port of **Boosted Decision tree Regression** to the input port of **Tune Model Hyperparameters(right)** and Set the property as:
    - **Create trainer mode:** single parameter
    - **Maximum number of leaves per tree:** 20
    - **Minimum number of samples per leaf node:** 10
    - **Learning rate:** 0.2
    - **Total number of trees constructed:** 100
    - **Random number seed:** 2345
    - **Allow unknown categorical levels:** checked
  10. Search for the **Permutation Features Importance** model and drag it onto the canvas. Configure the properties of **Permutation Features Importance** model as:
    - **Random Seed:** 1234
    - **Metric for measuring performance:** Regression-Coefficient of Determination
  11. Connect the **Tune Model Hyperparameters** module output (right) to the input ports of **Score Model(left)** and **Permutation Feature Importance(right)**.
  12. Search for the **Cross Validate Model** and drag it onto the canvas.
  13. Connect the output of **Boosted Decision tree Regression** model to the left inputs of the **Cross Validate Model** and connect output port of **Normalize Data** module to the right input port of **Cross Validate Model**. Configure the properties of the **Cross Validate Model** as:
    - **Label column:** Price
    - **Random seed:** 1234
  14. It should appear as given below:



### c) Evaluating the Model

1. Search for the **Evaluate** module and drag it onto the canvas.
2. Connect the left input of the Evaluate model from the output of Score Model.
3. It should appear as given below



4. Save and run the experiment.
5. When the experiment has finished, Visualize the output form the **Evaluate** module.

Metrics	
Mean Absolute Error	21.597769
Root Mean Squared Error	29.234137
Relative Absolute Error	0.518912
Relative Squared Error	0.327665
Coefficient of Determination	0.672335



# RATING PREDICTION

## Two-Class Boosted Decision Tree

### a) Data Preparation

1. Open a browser and browse to <https://studio.azureml.net>. Then sign in using the Microsoft account associated with your Azure ML account.
2. Create a new blank experiment and give it the title **Airbnb Rating Prediction-Two Class BDT**
3. Upload the `airbnb_sample.csv` file and drag it to canvas.
4. Search for the **Select Columns in Dataset (Project Columns)** module and drag it onto the canvas. Include the following columns:

**Column names:** Host Response Time, Host Response Rate, Host Acceptance Rate, Host Neighborhood, Host Listings Count, Host Total Listings Count, Property Type, Room Type, Price, Weekly Price, Monthly Price, Maximum Nights, Review Scores Rating, Review Scores Accuracy, Review Scores Cleanliness, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Cancellation Policy, Calculated host listings count, Neighborhood Cleansed, Neighborhood Group Cleansed, Bedrooms, Bathrooms, Beds, Security Deposit, Cleaning Fee, Extra People, Minimum Nights, Calendar Updated, Availability 30, Availability 60, Availability 90, Amenities.

5. Connect the output of the **Airbnb sampled** dataset to the **Dataset** input of the **Select Columns in Dataset module**.
6. Search for the **Clean Missing Data** module and drag it onto the canvas. Select the columns. Set cleaning mode as custom substitution value and replacement value as 0 as shown below:

**Clean Missing Data**

Columns to be cleaned

**Selected columns:**

**Column names:** Host Response Time, Host Response Rate, Host Acceptance Rate, Host Neighborhood, Host Listings Count, Host Total Listings Count, Neighborhood Cleansed, Neighborhood Group Cleansed, Property Type, Room Type, Bathrooms, Bedrooms, Beds, Amenities, Price, Weekly Price, Monthly Price, Security Deposit, Cleaning Fee, Extra People, Minimum Nights, Maximum Nights, Calendar Updated, Availability 30, Availability 60, Availability 90, Review Scores Rating, Review Scores Accuracy, Review Scores Cleanliness, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Cancellation

Launch column selector

Minimum missing value ratio

0

Maximum missing value ratio

1

Cleaning mode

Custom substitution value

Replacement value

0

☐ Generate missing value indicator column

7. Search for the **Clip values** module, drag it to canvas, connect it's input with the **cleaned dataset** output of the **Clean missing data** module.

8. Configure the properties of the **Clip Values** module as shown in the figure below:

**Clip Values**

Set of thresholds

ClipPeaksAndSubpeaks

Threshold

Percentile

Percentile number of upper threshold

90

Percentile number of lower threshold

10

Substitute value for peaks

Mean

Substitute value for subpeaks

Mean

List of columns

**Selected columns:**

**Column type:** Numeric, All

Launch column selector

9. Search for the **Normalize Data** module and drag it onto the canvas. Connect its input with the **results dataset** output of the **Clip Values** module.

10. Configure the properties of the **Normalize Data** as shown in the figure below:

**Normalize Data**

Transformation method  
ZScore

☒ Use 0 for constant columns when checked

Columns to transform

**Selected columns:**

Column type: Numeric, All

Exclude column names: Review Scores Rating

Exclude column type: String, All

Launch column selector

11. Search for the **Execute Python Script** module, drag it to canvas, connect its input with the **Transformed dataset** output of the **Normalize Data** module.

12. Add the following code in the **Execute Python Script** module.

### Python Script

**# The script MUST contain a function named azureml\_main**  
**# which is the entry point for this module.**

**# imports up here can be used to**  
import pandas as pd  
import numpy as np

def azureml\_main(dataframe1):

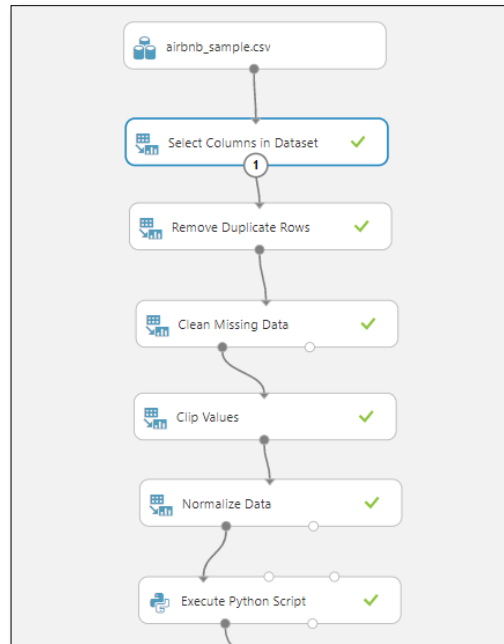
**# Execution logic goes here**

print('Input pandas.DataFrame #1:\r\n\r\n{0}'.format(dataframe1))

dataframe1['Review Scores Rating'] = np.where(dataframe1['Review Scores Rating'] >= 80, 'High',  
'Low')  
dataframe1.head()

**# Return value must be of a sequence of pandas.DataFrame**  
return dataframe1,

It should appear like this:



## b) Train the Model

Now that the data is prepared, you can train the model.

1. Search for the **Split Data** module and drag it onto the Canvas.
2. Connect the **Results dataset** output of the **Execute Python Script** module to the input of the **Split Data** module.
3. On the properties pane of the **Split Data** module, configure the properties as shown below:

Split Data

Splitting mode  
Split Rows

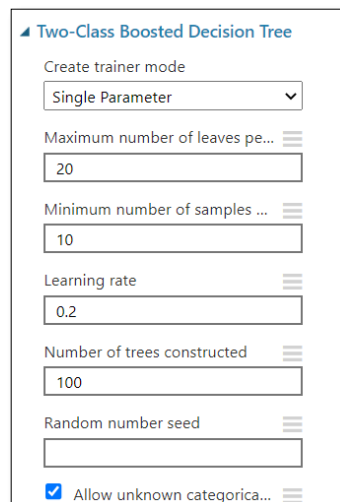
Fraction of rows in the first out...  
0.5

☒ Randomized split

Random seed  
0

Stratified split  
False

4. Search for the **Two - Class Boosted Decision Tree** Classification module and drag it onto the canvas. Set the property as shown below:



▲ Two-Class Boosted Decision Tree

Create trainer mode  
Single Parameter ▼

Maximum number of leaves per tree  
20

Minimum number of samples per leaf  
10

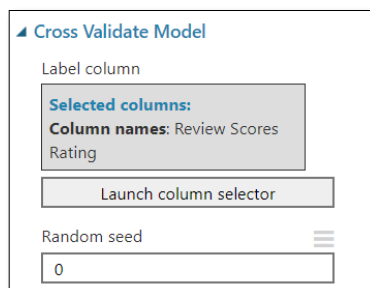
Learning rate  
0.2

Number of trees constructed  
100

Random number seed

☒ Allow unknown categorical values

5. Search for the **Cross Validate Model** and drag it onto the canvas. Set the property as shown below:



▲ Cross Validate Model

Label column  
Selected columns:  
Column names: Review Scores  
Rating

Launch column selector

Random seed  
0

6. Search for the **Tune Model Hyperparameters** and drag it onto the canvas. Set the property as shown below:

**▲ Tune Model Hyperparameters**

Specify parameter sweeping mode

Maximum number of runs on r...

Random seed

Label column  
**Selected columns:**  
 Column names: Review  
 Scores Rating

Metric for measuring performa...

Metric for measuring performa...

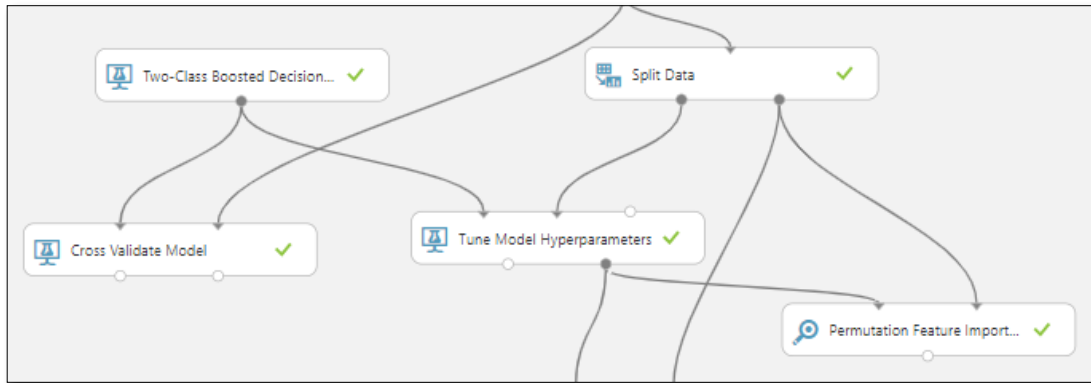
7. Search for the **Permutation Features Importance** model and drag it onto the canvas. Set the property as shown below:

**▲ Permutation Feature Importance**

Random seed

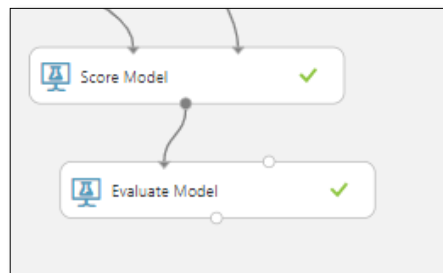
Metric for measuring performance

8. Search for Score model and drag it on canvas.
9. Connect the **Results dataset1** (left) output of the **Split Data** module to the middle input of the **Tune Model Hyperparameters**.
10. Connect the **Results dataset2** (right) output of the **Split Data** module to the right input of the **Score Model** and to the right input of the **Permutation Feature Importance**.
11. Connect the output of **Two - Class Boosted Decision Tree** model to the left inputs of the **Cross Validate Model**, and **Tune Model Hyperparameters**.
12. Connect the output of the **Execute Python Script** module to the right input of the **Cross Validate Model**.
13. Connect the right output of the **Tune Model Hyperparameters** to the left input of the **Score Model** and to the left input of the **Permutation Feature Importance** module.
14. The figure should be like as given below:

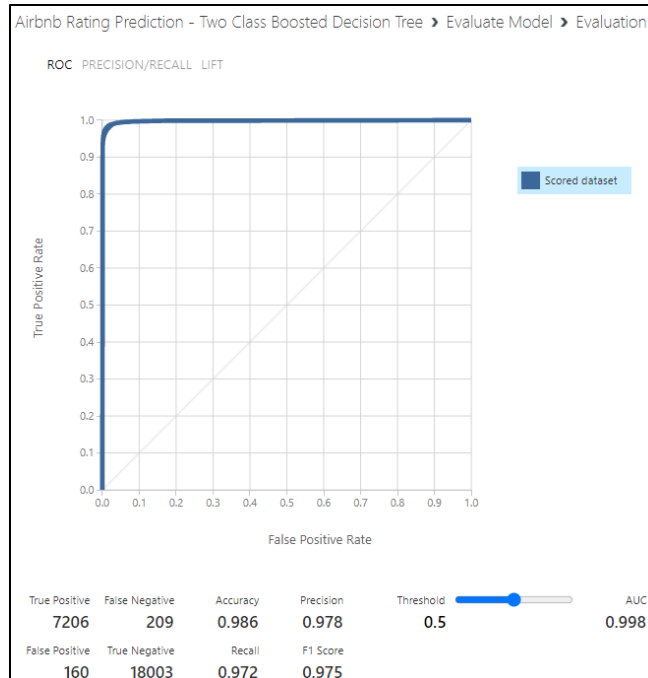


### c) Evaluating the Model

1. Search for the **Evaluate** module and drag it onto the canvas.
2. Connect the left input of the **Evaluate** model from the output of **Score Model**.
3. It would appear as below:



4. Save and run the experiment.
5. When the experiment has finished, Visualize the output form the **Evaluate** module.



## Two-Class Decision Forest

### a) Data Preparation

1. Open a browser and browse to <https://studio.azureml.net>. Then sign in using the Microsoft account associated with your Azure ML account.
2. Create a new blank experiment and give it the title **Airbnb Rating Prediction-Two Class Decision Forest**.
3. Upload the `airbnb_sample.csv` file and drag it to canvas.
4. Search for the **Select Columns in Dataset (Project Columns) module** and drag it onto the canvas. Include the following columns:

**Column names:** Host Response Time, Host Response Rate, Host Acceptance Rate, Host Neighborhood, Host Listings Count, Host Total Listings Count, Property Type, Room Type, Price, Weekly Price, Monthly Price, Maximum Nights, Review Scores Rating, Review Scores Accuracy, Review Scores Cleanliness, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Cancellation Policy, Calculated host listings count, Neighborhood Cleansed, Neighborhood Group Cleansed, Bedrooms, Bathrooms, Beds, Security Deposit, Cleaning Fee, Extra People, Minimum Nights, Calendar Updated, Availability 30, Availability 60, Availability 90, Amenities.

5. Connect the output of the **Airbnb sampled** dataset to the **Dataset** input of the **Select Columns in Dataset module**.



6. Search for the **Clean Missing Data** module and drag it onto the canvas. Select the columns. Set cleaning mode as custom substitution value and replacement value as 0 as shown below:

The screenshot shows the configuration window for the 'Clean Missing Data' module. It includes a list of columns to be cleaned, a 'Launch column selector' button, and input fields for 'Minimum missing value ratio' (0), 'Maximum missing value ratio' (1), 'Cleaning mode' (Custom substitution value), and 'Replacement value' (0). There is also a checkbox for 'Generate missing value indicator column'.

Clean Missing Data

Columns to be cleaned

Selected columns:

Column names: Host Response Time, Host Response Rate, Host Acceptance Rate, Host Neighborhood, Host Listings Count, Host Total Listings Count, Neighborhood Cleansed, Neighborhood Group Cleansed, Property Type, Room Type, Bathrooms, Bedrooms, Beds, Amenities, Price, Weekly Price, Monthly Price, Security Deposit, Cleaning Fee, Extra People, Minimum Nights, Maximum Nights, Calendar Updated, Availability 30, Availability 60, Availability 90, Review Scores Rating, Review Scores Accuracy, Review Scores Cleanliness, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Cancellation Policy, Cancellation Rate, Cancellation Rate 30, Cancellation Rate 60, Cancellation Rate 90, Cancellation Rate 180, Cancellation Rate 360, Cancellation Rate 720, Cancellation Rate 1440, Cancellation Rate 2880, Cancellation Rate 5760, Cancellation Rate 11520, Cancellation Rate 23040, Cancellation Rate 46080, Cancellation Rate 92160, Cancellation Rate 184320, Cancellation Rate 368640, Cancellation Rate 737280, Cancellation Rate 1474560, Cancellation Rate 2949120, Cancellation Rate 5898240, Cancellation Rate 11796480, Cancellation Rate 23592960, Cancellation Rate 47185920, Cancellation Rate 94371840, Cancellation Rate 188743680, Cancellation Rate 377487360, Cancellation Rate 754974720, Cancellation Rate 1509949440, Cancellation Rate 3019898880, Cancellation Rate 6039797760, Cancellation Rate 12079595520, Cancellation Rate 24159191040, Cancellation Rate 48318382080, Cancellation Rate 96636764160, Cancellation Rate 193273528320, Cancellation Rate 386547056640, Cancellation Rate 773094113280, Cancellation Rate 1546188226560, Cancellation Rate 3092376453120, Cancellation Rate 6184752906240, Cancellation Rate 12369505812480, Cancellation Rate 24739011624960, Cancellation Rate 49478023249920, Cancellation Rate 98956046499840, Cancellation Rate 197912092999680, Cancellation Rate 395824185999360, Cancellation Rate 791648371998720, Cancellation Rate 1583296743997440, Cancellation Rate 3166593487994880, Cancellation Rate 6333186975989760, Cancellation Rate 12666373951979520, Cancellation Rate 25332747903959040, Cancellation Rate 50665495807918080, Cancellation Rate 101330991615836160, Cancellation Rate 202661983231672320, Cancellation Rate 405323966463344640, Cancellation Rate 810647932926689280, Cancellation Rate 1621295865853378560, Cancellation Rate 3242591731706757120, Cancellation Rate 6485183463413514240, Cancellation Rate 12970366926827028480, Cancellation Rate 25940733853654056960, Cancellation Rate 51881467707308113920, Cancellation Rate 103762935414616227840, Cancellation Rate 207525870829232455680, Cancellation Rate 415051741658464911360, Cancellation Rate 830103483316929822720, Cancellation Rate 1660206966633859645440, Cancellation Rate 3320413933267719290880, Cancellation Rate 6640827866535438581760, Cancellation Rate 13281655733070877163520, Cancellation Rate 26563311466141754327040, Cancellation Rate 53126622932283508654080, Cancellation Rate 106253245864567017308160, Cancellation Rate 212506491729134034616320, Cancellation Rate 425012983458268069232640, Cancellation Rate 850025966916536138465280, Cancellation Rate 1700051933833072276930560, Cancellation Rate 3400103867666144553861120, Cancellation Rate 6800207735332289107722240, Cancellation Rate 13600415470664578215444480, Cancellation Rate 27200830941329156430888960, Cancellation Rate 54401661882658312861777920, Cancellation Rate 108803323765316625723555840, Cancellation Rate 217606647530633251447111680, Cancellation Rate 435213295061266502894223360, Cancellation Rate 870426590122533005788446720, Cancellation Rate 1740853180245066011576893440, Cancellation Rate 3481706360490132023153786880, Cancellation Rate 6963412720980264046307573760, Cancellation Rate 13926825441960528092615147520, Cancellation Rate 27853650883921056185230295040, Cancellation Rate 55707301767842112370460590080, Cancellation Rate 111414603535684224740921180160, Cancellation Rate 222829207071368449481842360320, Cancellation Rate 445658414142736898963684720640, Cancellation Rate 891316828285473797927369441280, Cancellation Rate 1782633656570947595854738882560, Cancellation Rate 3565267313141895191709477765120, Cancellation Rate 7130534626283790383418955530240, Cancellation Rate 14261069252567580766837911060480, Cancellation Rate 28522138505135161533675822120960, Cancellation Rate 57044277010270323067351644241920, Cancellation Rate 114088554020540646134703288483840, Cancellation Rate 228177108041081292269406576967680, Cancellation Rate 456354216082162584538813153935360, Cancellation Rate 912708432164325169077626307870720, Cancellation Rate 1825416864328650338155252615741440, Cancellation Rate 3650833728657300676310505231482880, Cancellation Rate 7301667457314601352621010462965760, Cancellation Rate 14603334914629202705242020925931520, Cancellation Rate 29206669829258405410484041851863040, Cancellation Rate 58413339658516810820968083703726080, Cancellation Rate 116826679317033621641936167407452160, Cancellation Rate 233653358634067243283872334814904320, Cancellation Rate 467306717268134486567744669629808640, Cancellation Rate 934613434536268973135489339259617280, Cancellation Rate 1869226869072537946270978678519234560, Cancellation Rate 3738453738145075892541957357038469120, Cancellation Rate 7476907476290151785083914714076938240, Cancellation Rate 14953814952580303570167829428153876480, Cancellation Rate 29907629905160607140335658856307752960, Cancellation Rate 59815259810321214280671317712615505920, Cancellation Rate 119630519620642428561342635425231011840, Cancellation Rate 239261039241284857122685270850462023680, Cancellation Rate 478522078482569714245370541700924047360, Cancellation Rate 957044156965139428490741083401848094720, Cancellation Rate 1914088313930278856981482166803696189440, Cancellation Rate 3828176627860557713962964333607392378880, Cancellation Rate 7656353255721115427925928667214784757760, Cancellation Rate 15312706511442230855851857334429569515520, Cancellation Rate 30625413022884461711703714668859139031040, Cancellation Rate 61250826045768923423407429337718278062080, Cancellation Rate 122501652091537846846814858675436556124160, Cancellation Rate 245003304183075693693629717350873112248320, Cancellation Rate 490006608366151387387259434701746224496640, Cancellation Rate 980013216732302774774518869403492448993280, Cancellation Rate 1960026433464605549549037738806984897986560, Cancellation Rate 3920052866929211099098075477613969795973120, Cancellation Rate 7840105733858422198196150955227939591946240, Cancellation Rate 15680211467716844396392301910455879183892480, Cancellation Rate 31360422935433688792784603820911758367784960, Cancellation Rate 62720845870867377585569207641823516735569920, Cancellation Rate 125441691741734755171138415283647033471139840, Cancellation Rate 250883383483469510342276830567294066942279680, Cancellation Rate 501766766966939020684553661134588133884559360, Cancellation Rate 1003533533933878041369107322269176267769118720, Cancellation Rate 2007067067867756082738214644538352535538237440, Cancellation Rate 4014134135735512165476429289076705071076474880, Cancellation Rate 8028268271471024330952858578153410142152949760, Cancellation Rate 16056536542942048661905717156306820284305899520, Cancellation Rate 32113073085884097323811434312613640568611799040, Cancellation Rate 64226146171768194647622868625227281137223598080, Cancellation Rate 128452292343536389295245737250454562274447196160, Cancellation Rate 256904584687072778590491474500909124548894392320, Cancellation Rate 513809169374145557180982949001818249097788784640, Cancellation Rate 1027618338748291114361965898003636498195577569280, Cancellation Rate 2055236677496582228723931796007272996391155138560, Cancellation Rate 4110473354993164457447863592014545992782310277120, Cancellation Rate 8220946709986328914895727184029091985564620554240, Cancellation Rate 16441893419972657829791454368058183971129241108480, Cancellation Rate 32883786839945315659582908736116367942258482216960, Cancellation Rate 65767573679890631319165817472232735884516964433920, Cancellation Rate 131535147359781262638331634944465471769033928867840, Cancellation Rate 263070294719562525276663269888930943538067857735680, Cancellation Rate 526140589439125050553326539777861887076135715471360, Cancellation Rate 1052281178878250101106653079555723774152271430942720, Cancellation Rate 2104562357756500202213306159111447548304542861885440, Cancellation Rate 4209124715513000404426612318222895096609085723770880, Cancellation Rate 8418249431026000808853224636445790193218171447541760, Cancellation Rate 16836498862052001617706449272891580386436342895083520, Cancellation Rate 33672997724104003235412898545783160772872685790167040, Cancellation Rate 67345995448208006470825797091566321545745371580334080, Cancellation Rate 134691990896416012941651594183132643091490743160668160, Cancellation Rate 269383981792832025883303188366265286182981486321336320, Cancellation Rate 538767963585664051766606376732530572365962972642672640, Cancellation Rate 1077535927171328103533212753465061144731925945285345280, Cancellation Rate 2155071854342656207066425506930122289463851890570690560, Cancellation Rate 4310143708685312414132851013860244578927703781141381120, Cancellation Rate 8620287417370624828265702027720489157855407562282762240, Cancellation Rate 17240574834741249656531404055440978315710815124565524480, Cancellation Rate 34481149669482499313062808110881956631421630249131048960, Cancellation Rate 68962299338964998626125616221763913262843260498262097920, Cancellation Rate 137924598677929997252251232443527826525686520996524195840, Cancellation Rate 275849197355859994504502464887055653051373041993048391680, Cancellation Rate 551698394711719989009004929774111306102746083986096783360, Cancellation Rate 1103396789423439978018009859548222612205492167972193566720, Cancellation Rate 2206793578846879956036019719096445224410984335944387133440, Cancellation Rate 4413587157693759912072039438192890448821968671888774266880, Cancellation Rate 8827174315387519824144078876385780897643937343777548533760, Cancellation Rate 17654348630775039648288157752771561795287874687555097067520, Cancellation Rate 35308697261550079296576315505543123590575749375110194135040, Cancellation Rate 70617394523100158593152631011086247181151498750220388270080, Cancellation Rate 141234789046200317186305262022172494362302997500440776540160, Cancellation Rate 282469578092400634372610524044344988724605995000881553080320, Cancellation Rate 564939156184801268745221048088689977449211990001763106160640, Cancellation Rate 1129878312369602537490442096177379954898423980003526212321280, Cancellation Rate 2259756624739205074980884192354759909796847960007052424642560, Cancellation Rate 4519513249478410149961768384709519819593695920014104849285120, Cancellation Rate 9039026498956820299923536769419039639187391840028209698570240, Cancellation Rate 18078052997913640599847073538838079278374783680056419397140480, Cancellation Rate 36156105995827281199694147077676158556749567360112838794280960, Cancellation Rate 72312211991654562399388294155352317113499134720225677588561920, Cancellation Rate 144624423983309124798776588310704634226998269440451355177123840, Cancellation Rate 289248847966618249597553176621409268453996538880902710354247680, Cancellation Rate 578497695933236499195106353242818536907993077761805420708495360, Cancellation Rate 1156995391866472998390212706485637073815986155523610841416990720, Cancellation Rate 2313990783732945996780425412971274147631972311047221682833981440, Cancellation Rate 4627981567465891993560850825942548295263944622094443365667962880, Cancellation Rate 9255963134931783987121701651885096590527889244188886731335925760, Cancellation Rate 18511926269863567974243403303770193181055778488377773462671851520, Cancellation Rate 37023852539727135948486806607540386362111556976755546925343703040, Cancellation Rate 74047705079454271896973613215080772724223113953511093850687406080, Cancellation Rate 148095410158908543793947226430161545448446227907022187701374812160, Cancellation Rate 296190820317817087587894452860323090896892455814044375402749624320, Cancellation Rate 592381640635634175175788905720646181793784911628088750805499248640, Cancellation Rate 1184763281271268350351577811441292363587569823256177501610998497280, Cancellation Rate 2369526562542536700703155622882584727175139646512355003221996994560, Cancellation Rate 4739053125085073401406311245765169454350279293024710006443993989120, Cancellation Rate 9478106250170146802812622491530338908700558586049420012887987978240, Cancellation Rate 18956212500340293605625244983060677817401117172098840025775975956480, Cancellation Rate 37912425000680587211250489966121355634802234344197680051551951912960, Cancellation Rate 75824850001361174422500979932242711269604468688395360103103903825920, Cancellation Rate 151649700002722348845001959864485422539208937376790720206207807651840, Cancellation Rate 303299400005444697690003919728970845078417874753581440412415615303680, Cancellation Rate 606598800010889395380007839457941690156835749507162880824831230607360, Cancellation Rate 1213197600021778790760015678915883380313671499014325761649662461214720, Cancellation Rate 2426395200035557581520031357831766760627342998028651523299324922429440, Cancellation Rate 4852790400071115163040062715663533521254685996057303046598649844858880, Cancellation Rate 9705580800142230326080125431327067042509371992114606093197299689717760, Cancellation Rate 19411161600284460652160250862654134085018743984229212186394599379435520, Cancellation Rate 38822323200568921304320501725308268170037487968458424372789198758871040, Cancellation Rate 77644646401137842608641003450616536340074975936916848745578397517742080, Cancellation Rate 155289292802275685217282006901233072680149951873833697491156795035484160, Cancellation Rate 310578585605511370434564013802466145360299903747667394982313590070968320, Cancellation Rate 621157171211022740869128027604932290720599807495334789964627180141936640, Cancellation Rate 1242314342422045481738256055209864581441199614990669579929254360283873280, Cancellation Rate 2484628684844090963476512110419729162882399229981339159858508720567746560, Cancellation Rate 4969257369688181926953024220839458325764798459962678319717017441135493120, Cancellation Rate 9938514739376363853906048441678916651529596919925356639434034882270986240, Cancellation Rate 19877029478752727707812096883357833303059193839850713278868069764541972480, Cancellation Rate 39754058957505455415624193766715666606118387679701426557736139529083944960, Cancellation Rate 79508117915010910831248387533431333212236775359402853115472279058167889920, Cancellation Rate 159016235830021821662496775066862666424473550718805706230944558116335779840, Cancellation Rate 318032471660043643324993550133725332848947101437611412461889116232671559680, Cancellation Rate 636064943320087286649987100267450665697894202875222824923778232465343119360, Cancellation Rate 1272129886640174573299974200534901331395788405750445649847556464930686238720, Cancellation Rate 2544259773280349146599948401069802662791576811500891299695112929861372477440, Cancellation Rate 5088519546560698293199896802139605325583153623001782599390225859722744954880, Cancellation Rate 10177039093121396586399793604279210651166307246003565198780451719445489909760, Cancellation Rate 20354078186242793172799587208558421302332614492007130397560903438890979819520, Cancellation Rate 40708156372485586345599174417116842604665228984014260795121806877781959639040, Cancellation Rate 81416312744971172691198348834233685209330457968028521590243613755563919278080, Cancellation Rate 162832625489942345382396697668467370418660915936057043180487227511127838556160, Cancellation Rate 325665250979884690764793395336934740837321831872114086360974455022255677112320, Cancellation Rate 651330501959769381529586790673869481674643663744228172721948910044511354224640, Cancellation Rate 1302661003919538763059173581347738963349287327488456345443897820089022708449280, Cancellation Rate 2605322007839077526118347162695477926698574654976912690887795640178045416898560, Cancellation Rate 5210644015678155052236694325390955853397149309953825381775591280356090833797120, Cancellation Rate 10421288031356310104473388650781911706794298619907650763551182560712181667594240, Cancellation Rate 20842576062712620208946777301563823413588597239815301527102365121424363335188480, Cancellation Rate 41685152125425240417893554603127646827177194479630603054204730242848726670376960, Cancellation Rate 83370304250850480

▲ Normalize Data

Transformation method  
ZScore

☒ Use 0 for constant columns when checked

Columns to transform

Selected columns:  
Column type: Numeric, All  
Exclude column names: Review Scores Rating  
Exclude column type: String, All

Launch column selector

11. Search for the **Execute Python Script** module, drag it to canvas, connect its input with the **Transformed dataset** output of the **Normalize Data** module.

12. Add the following code in the **Execute Python Script** module.

### Python Script

**# The script MUST contain a function named azureml\_main  
# which is the entry point for this module.**

**# imports up here can be used to**

```
import pandas as pd
import numpy as np
```

```
def azureml_main(dataframe1):
```

**# Execution logic goes here**

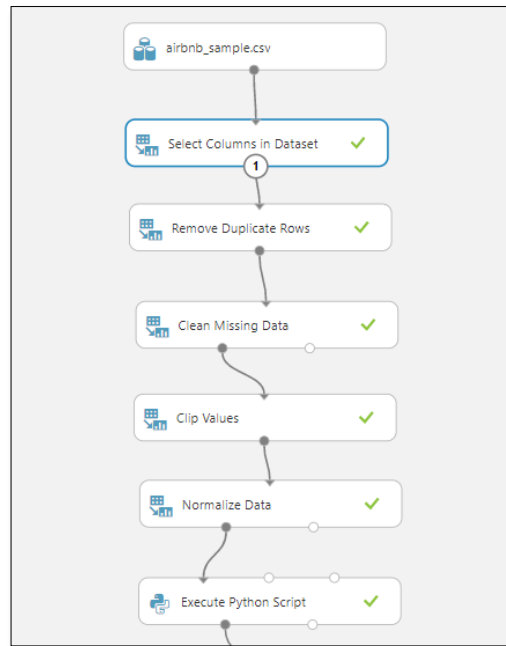
```
print('Input pandas.DataFrame #1:\r\n\r\n{0}'.format(dataframe1))
```

```
dataframe1['Review Scores Rating'] = np.where(dataframe1['Review Scores Rating'] >= 80, 'High',
'Low')
dataframe1.head()
```

**# Return value must be of a sequence of pandas.DataFrame**

```
return dataframe1,
```

It should appear like this:



## b) Train the Model

Now that the data is prepared, you can train the model.

1. Search for the **Split Data** module and drag it onto the Canvas.
2. Connect the **Results dataset** output of the **Execute Python Script** module to the input of the **Split Data** module.
3. On the properties pane of the **Split Data** module, configure the properties as shown below:

▲ Split Data

Splitting mode  
Split Rows ▼

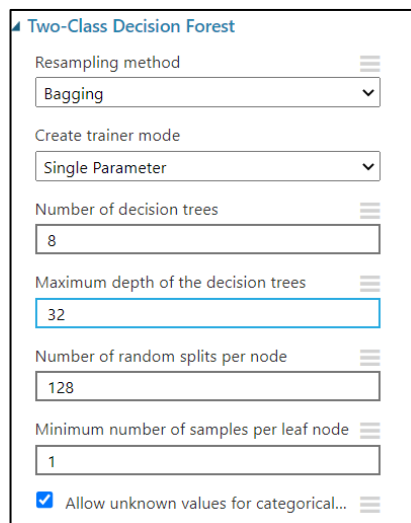
Fraction of rows in the first out...  
0.5

☒ Randomized split

Random seed  
0

Stratified split  
False ▼

4. Search for the **Two-Class Decision Forest** Classification module and drag it onto the canvas. Set the property as shown below:



Two-Class Decision Forest

Resampling method  
Bagging

Create trainer mode  
Single Parameter

Number of decision trees  
8

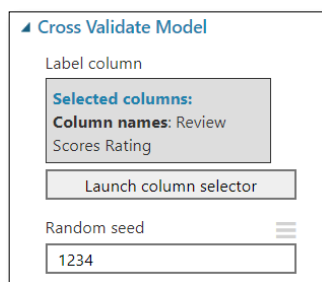
Maximum depth of the decision trees  
32

Number of random splits per node  
128

Minimum number of samples per leaf node  
1

☒ Allow unknown values for categorical...

5. Search for the **Cross Validate Model** and drag it onto the canvas. Set the property as shown below:

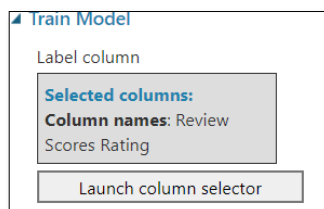


Cross Validate Model

Label column  
Selected columns:  
Column names: Review  
Scores Rating  
Launch column selector

Random seed  
1234

6. Search for the **Train Model** and drag it onto the canvas. Set the property as shown below:



Train Model

Label column  
Selected columns:  
Column names: Review  
Scores Rating  
Launch column selector

7. Search for the **Permutation Features Importance** model and drag it onto the canvas. Set the property as shown below:

**Permutation Feature Importance**

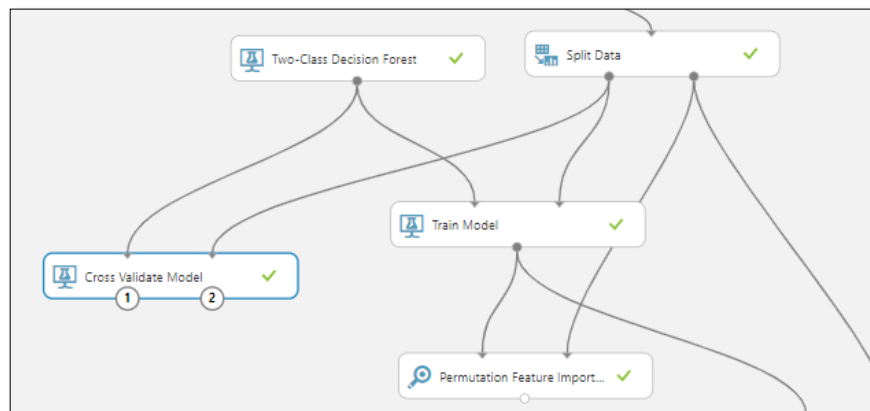
Random seed

1234

Metric for measuring performance

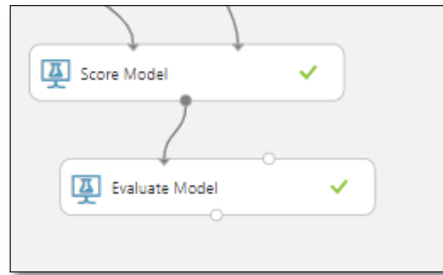
Classification - Accuracy

6. Search for Score model and drag it on canvas.
7. Connect the **Results dataset1** (left) output of the **Split Data** module to the right input of the **Train Model**.
8. Connect the **Results dataset2** (right) output of the **Split Data** module to the right input of the **Score Model** and to the right input of the **Permutation Feature Importance**.
9. Connect the output of **Two-Class Decision Forest** model to the left inputs of the **Cross Validate Model**, and **Train Model**.
10. Connect the output of the **Split Data** module to the right input of the **Cross Validate Model**.
11. Connect the output of the **Train Model** to the left input of the **Score Model** and to the left input of the **Permutation Feature Importance** module.
12. The figure should be like as given below:

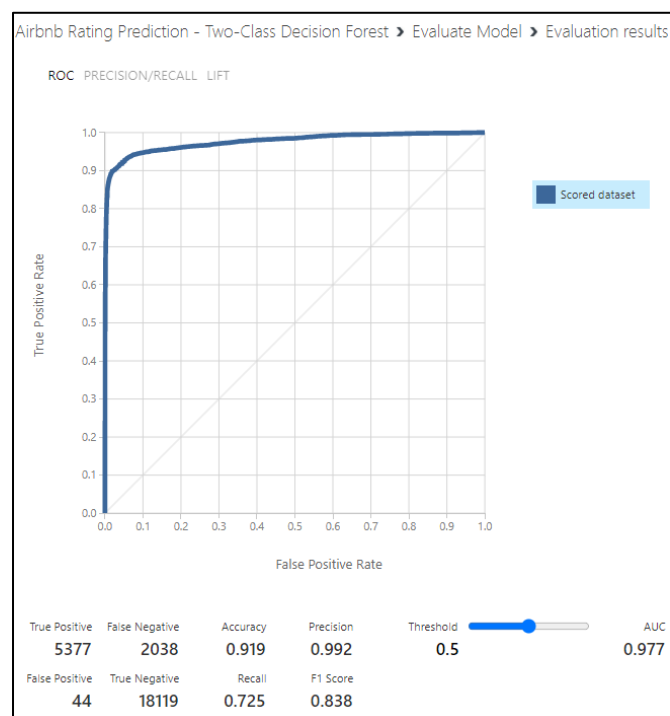


### c) Evaluating the Model

1. Search for the **Evaluate** module and drag it onto the canvas.
2. Connect the left input of the **Evaluate** model from the output of **Score Model**.
3. It would appear as below:



4. Save and run the experiment.
5. When the experiment has finished, Visualize the output form the **Evaluate** module.



# Two-Class Logistic Regression

## a) Data Preparation

1. Open a browser and browse to <https://studio.azureml.net>. Then sign in using the Microsoft account associated with your Azure ML account.
2. Create a new blank experiment and give it the title **Airbnb Rating Prediction-Two Class BDT**
3. Upload the `airbnb_sample.csv` file and drag it to canvas.
4. Search for the **Select Columns in Dataset (Project Columns) module** and drag it onto the canvas. Include the following columns:

**Column names:** Host Response Time, Host Response Rate, Host Acceptance Rate, Host Neighborhood, Host Listings Count, Host Total Listings Count, Property Type, Room Type, Price, Weekly Price, Monthly Price, Maximum Nights, Review Scores Rating, Review Scores Accuracy, Review Scores Cleanliness, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Cancellation Policy, Calculated host listings count, Neighborhood Cleansed, Neighborhood Group Cleansed, Bedrooms, Bathrooms, Beds, Security Deposit, Cleaning Fee, Extra People, Minimum Nights, Calendar Updated, Availability 30, Availability 60, Availability 90, Amenities.

5. Connect the output of the **Airbnb sampled** dataset to the **Dataset** input of the **Select Columns in Dataset module**.
6. Search for the **Clean Missing Data** module and drag it onto the canvas. Select the columns. Set cleaning mode as custom substitution value and replacement value as 0 as shown below:

The screenshot shows the 'Clean Missing Data' module configuration interface. It includes a list of 'Selected columns' with a scrollable list of column names. Below this is a 'Launch column selector' button. The 'Minimum missing value ratio' is set to 0, and the 'Maximum missing value ratio' is set to 1. The 'Cleaning mode' is set to 'Custom substitution value', and the 'Replacement value' is set to 0. There is also a checkbox for 'Generate missing value indicator column' which is currently unchecked.

**Clean Missing Data**

Columns to be cleaned

**Selected columns:**

**Column names:** Host Response Time, Host Response Rate, Host Acceptance Rate, Host Neighborhood, Host Listings Count, Host Total Listings Count, Neighborhood Cleansed, Neighborhood Group Cleansed, Property Type, Room Type, Bathrooms, Bedrooms, Beds, Amenities, Price, Weekly Price, Monthly Price, Security Deposit, Cleaning Fee, Extra People, Minimum Nights, Maximum Nights, Calendar Updated, Availability 30, Availability 60, Availability 90, Review Scores Rating, Review Scores Accuracy, Review Scores Cleanliness, Review Scores Checkin, Review Scores Communication, Review Scores Location, Review Scores Value, Cancellation Policy, Calculated host listings count

Launch column selector

Minimum missing value ratio: 0

Maximum missing value ratio: 1

Cleaning mode: Custom substitution value

Replacement value: 0

☐ Generate missing value indicator column

7. Search for the **Clip values** module, drag it to canvas, connect its input with the **cleaned dataset** output of the **Clean missing data** module.
8. Configure the properties of the **Clip Values** module as shown in the figure below:

The screenshot shows the configuration window for the 'Clip Values' module. It includes the following settings:

- Set of thresholds:** A dropdown menu set to 'ClipPeaksAndSubpeaks'.
- Threshold:** A dropdown menu set to 'Percentile'.
- Percentile number of upper threshold:** A text input field containing '90'.
- Percentile number of lower threshold:** A text input field containing '10'.
- Substitute value for peaks:** A dropdown menu set to 'Mean'.
- Substitute value for subpeaks:** A dropdown menu set to 'Mean'.
- List of columns:** A section showing 'Selected columns:' with 'Column type: Numeric, All'. Below this is a button labeled 'Launch column selector'.

9. Search for the **Normalize Data** module and drag it onto the canvas. Connect it's input with the **results dataset** output of the **Clip Values** module.
10. Configure the properties of the Normalize Data as shown in the figure below:

The screenshot shows the configuration window for the 'Normalize Data' module. It includes the following settings:

- Transformation method:** A dropdown menu set to 'ZScore'.
- Use 0 for constant columns when checked:** A checkbox that is checked.
- Columns to transform:** A section showing 'Selected columns:' with 'Column type: Numeric, All'. It also lists 'Exclude column names: Review Scores Rating' and 'Exclude column type: String, All'. Below this is a button labeled 'Launch column selector'.

11. Search for the **Execute Python Script** module, drag it to canvas, connect it's input with the **Transformed dataset** output of the **Normalize Data** module.
12. Add the following code in the **Execute Python Script** module.



## Python Script

**# The script MUST contain a function named azureml\_main  
# which is the entry point for this module.**

**# imports up here can be used to**

import pandas as pd

import numpy as np

def azureml\_main(dataframe1):

**# Execution logic goes here**

print('Input pandas.DataFrame #1:\r\n\r\n{0}'.format(dataframe1))

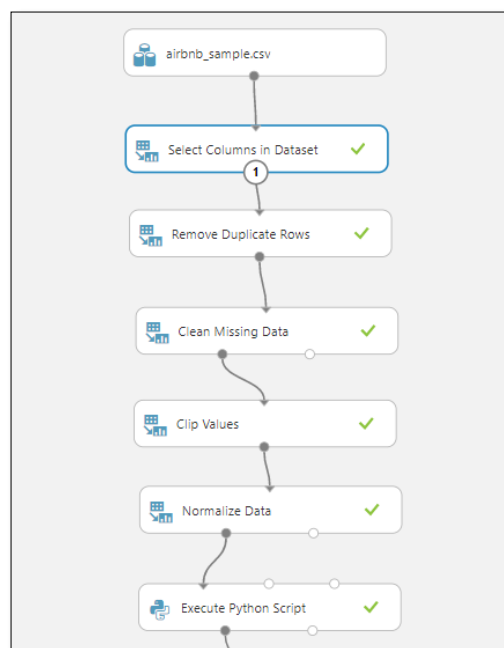
dataframe1['Review Scores Rating'] = np.where(dataframe1['Review Scores Rating'] >= 80, 'High',  
'Low')

dataframe1.head()

**# Return value must be of a sequence of pandas.DataFrame**

return dataframe1,

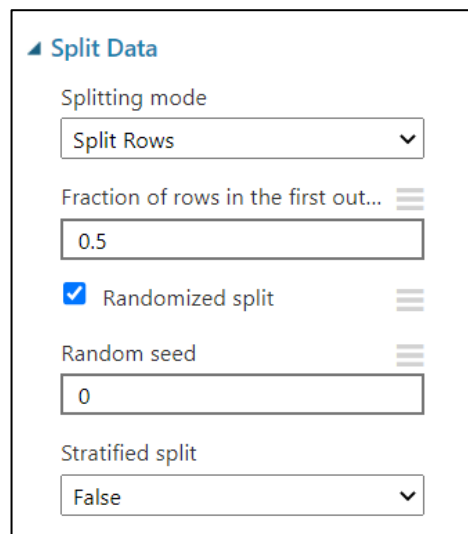
It should appear like this:



## b) Train the Model

Now that the data is prepared, you can train the model.

1. Search for the **Split Data** module and drag it onto the Canvas.
2. Connect the **Results dataset** output of the **Execute Python Script** module to the input of the **Split Data** module.
3. On the properties pane of the **Split Data** module, configure the properties as shown below:



▲ Split Data

Splitting mode  
Split Rows ▼

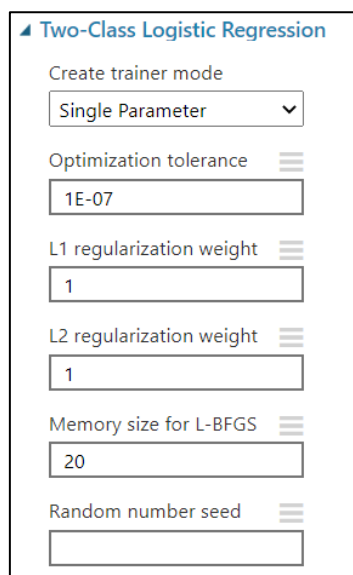
Fraction of rows in the first out...  
0.5

☒ Randomized split

Random seed  
0

Stratified split  
False ▼

4. Search for the **Two-Class Logistic Regression** Classification module and drag it onto the canvas. Set the property as shown below:



▲ Two-Class Logistic Regression

Create trainer mode  
Single Parameter ▼

Optimization tolerance  
1E-07

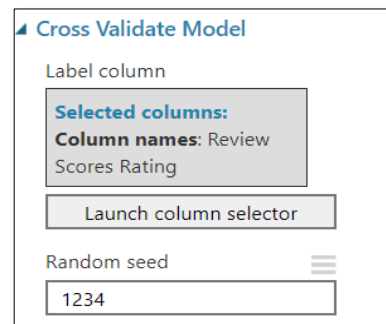
L1 regularization weight  
1

L2 regularization weight  
1

Memory size for L-BFGS  
20

Random number seed

5. Search for the **Cross Validate Model** and drag it onto the canvas. Set the property as shown below:



▲ Cross Validate Model

Label column

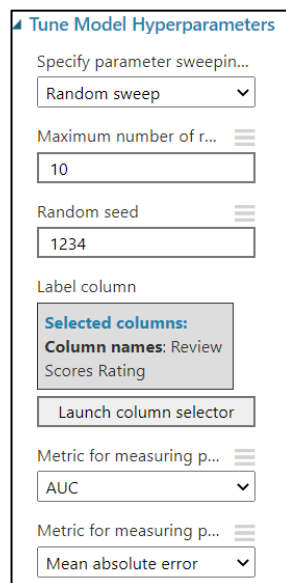
**Selected columns:**  
Column names: Review  
Scores Rating

Launch column selector

Random seed

1234

6. Search for the **Tune Model Hyperparameters** and drag it onto the canvas. Set the property as shown below:



▲ Tune Model Hyperparameters

Specify parameter sweepin...

Random sweep

Maximum number of r...

10

Random seed

1234

Label column

**Selected columns:**  
Column names: Review  
Scores Rating

Launch column selector

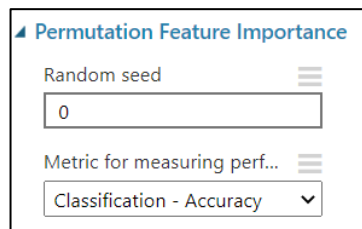
Metric for measuring p...

AUC

Metric for measuring p...

Mean absolute error

7. Search for the **Permutation Features Importance** model and drag it onto the canvas. Set the property as shown below:



▲ Permutation Feature Importance

Random seed

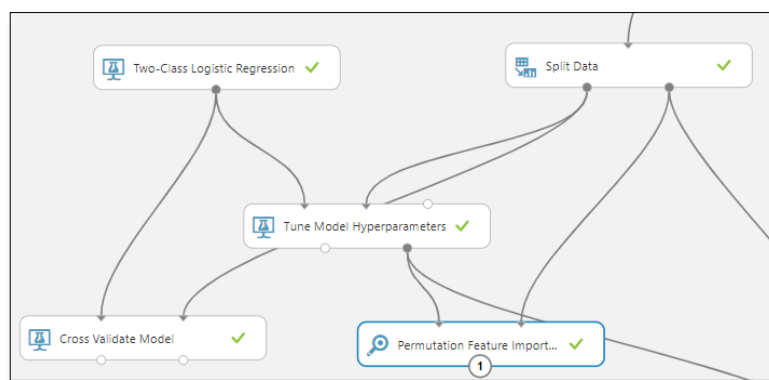
0

Metric for measuring perf...

Classification - Accuracy

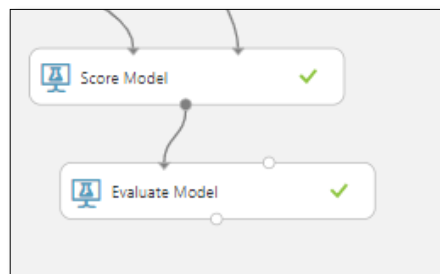
8. Search for Score model and drag it on canvas.
9. Connect the **Results dataset1** (left) output of the **Split Data** module to the middle input of the **Tune Model Hyperparameters**.

10. Connect the **Results dataset2** (right) output of the **Split Data** module to the right input of the **Score Model** and to the right input of the **Permutation Feature Importance**.
11. Connect the output of **Two-Class Logistic Regression** model to the left inputs of the **Cross Validate Model**, and **Tune Model Hyperparameters**.
12. Connect the output of the **Split Data** module to the right input of the **Cross Validate Model**.
13. Connect the right output of the **Tune Model Hyperparameters** to the left input of the **Score Model** and to the left input of the **Permutation Feature Importance** module.
14. The figure should be like as given below:

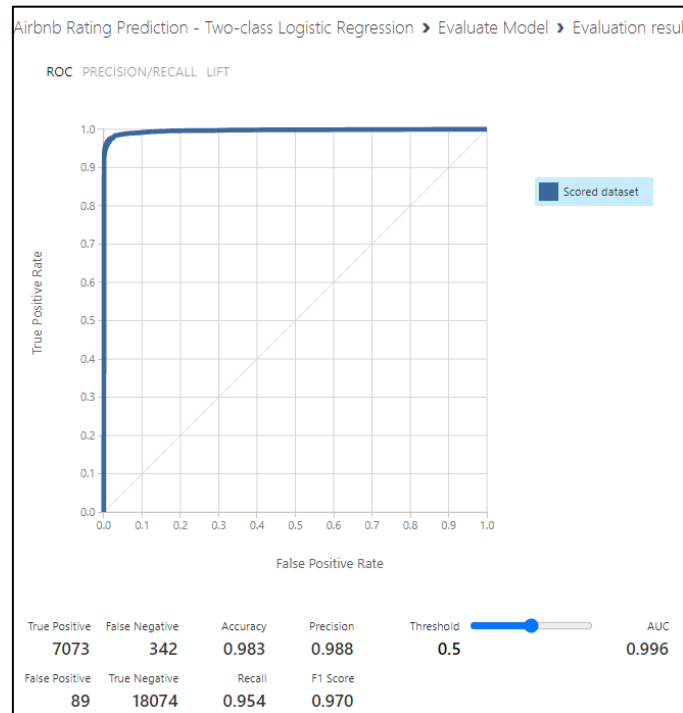


### c) Evaluating the Model

1. Search for the **Evaluate** module and drag it onto the canvas.
2. Connect the left input of the **Evaluate** model from the output of **Score Model**.
3. It would appear as below:



4. Save and run the experiment.
5. When the experiment has finished, Visualize the output form the **Evaluate** module.



## References:

1. URL of Data Source:

- [https://public.opendatasoft.com/explore/dataset/airbnblistings/table/?disjunctive.host\\_verifications&disjunctive.amenities&disjunctive.features](https://public.opendatasoft.com/explore/dataset/airbnblistings/table/?disjunctive.host_verifications&disjunctive.amenities&disjunctive.features)
- <https://www.kaggle.com/samyukthamurali/airbnb-ratings-dataset?select=airbnb-reviews.csv>

2. URL of GitHub: <https://github.com/SYSavy/CIS-5560>

3. URL of References:

- Microsoft's DAT203x, Data Science and Machine Learning Essentials.
- Discover Feature Engineering, How to Engineer Features and How to Get Good at

It: <https://machinelearningmastery.com/discover-feature-engineering-how-to-engineer-features-and-how-to-get-good-at-it/>