

# Lab Tutorial

Shuvadeep Kundu, skundu@calstatela.edu; Ankita Paul, apaul11@calstatela.edu;

Kevin Xu, hxu9@calstatela.edu

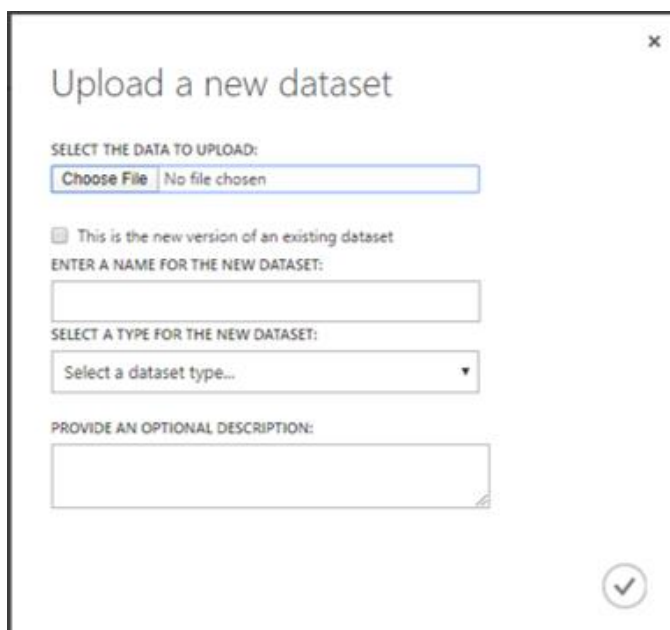
## Iowa Liquor Sales Predictive Analytics

This tutorial will show you how to set up the following machine learning algorithms in Microsoft Azure ML to build models to predict liquor sales in the state of Iowa with Linear Regression, Boosted Decision Tree Regression, and Decision Forest Regression.

Data Preparation ➡ Splitting the data ➡ Training the model ➡ Evaluating the model

### Creating a Sample dataset from the original big dataset

- Open a browser and go to <https://studio.azureml.net> and sign into your Azure ML account.
- In the Azure ML Studio, go to the **Experiments** page, click **New** at the bottom of the page. Then in the collection of Microsoft samples, select **Blank Experiment**. This creates a blank experiment. Give it the title **Sampling**.
- With the **Sampling** experiment open, at the bottom left, click **NEW**. Then in the **NEW** dialog box, click the **DATASET** tab.
- Go to **FROM LOCAL FILE**, choose file **iowa\_Liquor\_Sales.csv** and click **Ok** to upload a new dataset.



The screenshot shows the 'Upload a new dataset' dialog box in Azure ML Studio. The dialog has a title bar with a close button (X). The main content area includes the following elements:

- SELECT THE DATA TO UPLOAD:** A button labeled 'Choose File' and a text field showing 'No file chosen'.
- ☐ This is the new version of an existing dataset
- ENTER A NAME FOR THE NEW DATASET:** A text input field.
- SELECT A TYPE FOR THE NEW DATASET:** A dropdown menu with the text 'Select a dataset type...' and a downward arrow.
- PROVIDE AN OPTIONAL DESCRIPTION:** A text input field.

At the bottom center of the dialog, there is a circular button with a checkmark (✓).

- Once the dataset is uploaded, in the **Sampling** experiment items pane on the left, expand **Saved Datasets**, expand **My Datasets**, and drag **iowa\_Liquor\_Sales.csv** to the experiment canvas in the middle of the page.
- Search for **Partition and Sample** module and drag it to the canvas below the dataset module. Connect the output of the **iowa\_Liquor\_Sales.csv** dataset to the **Dataset** input of the **Partition and Sample** module. In the **Properties** pane of the **Partition and Sample** module, select the fields as following:

**Partition or sample mode:** Sampling

**Rate of sampling:** 0.01

**Random seed for sampling:** 1234

**Stratified split for sampling:** True

**Launch Column Selector:** Sale (Dollars)

---

Properties

Project

▲ Partition and Sample

Partition or sample mode

Sampling ▼

Rate of sampling

0.01

Random seed for samp...

1234

Stratified split for sampling

True ▼

Stratification key column fo...

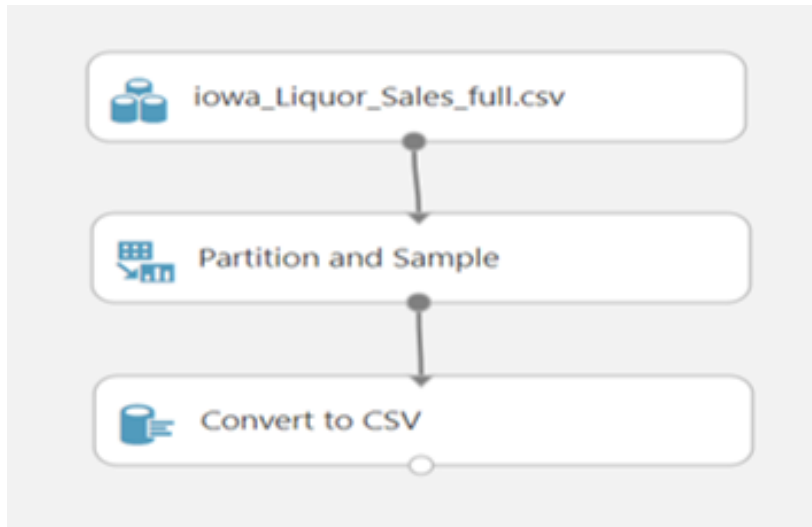
Selected columns:

Column names: Sale (Dollars)

Launch column selector

- Search for **Convert to CSV** module and drag it to the canvas. Connect the output port (**Results Dataset**) of *Partition and Sample* module to the **Dataset** port of **Convert to CSV** module.

The experiment would look like this:



- **Save** the experiment and **Run** it. Once it has finished running, right click on the **Results dataset** (output) port of the **Convert to CSV** module and **Download** the sampled dataset. Name it as **sample.csv** and save it. Upload this new dataset **sample.csv** to the Datasets page of Azure ML Studio in the same way as the *iowa\_Liquor\_Sales.csv* dataset was uploaded. We would use this **sample.csv** dataset to build our Machine Learning models and run experiment on it.

## Linear Regression

### Data Preparation

- Open a browser and browse to <https://studio.azureml.net>. Then sign in using the Microsoft account associated with your Azure ML account.
- Create a new blank experiment and give it the title **IOWA – Linear Regression – Cross validation + Tune Model Hyperparameters**.
- In the experiment items pane on the left, expand **Saved Datasets**, expand **My Datasets**, and drag **sample.csv** to the experiment canvas.
- Search for **Partition and Sample** module and connect the output port of **sample.csv** module to the **Dataset** port of **Partition and Sample** module. In the **Properties** pane of **Partition and Sample** module, set the configurations as follows:

## Properties Project

### Partition and Sample

Partition or sample mode

Sampling

Rate of sampling

0.095

Random seed for sampl...

1234

Stratified split for sampling

False

- Search for **Edit Metadata** module and drag it to the canvas. Connect the **Results Dataset** (output) port of *Partition and Sample* module to the **Dataset** port of *Edit Metadata* module. In the Properties pane of *Edit Metadata* module, **Launch Column Selector. With Rules > Begin With > All Columns** and select **Include column names**. Select the columns *Store Number*, *Zip Code*, *County Number*, *Category*, *Item Number* as follows :

Select columns

BY NAME

WITH RULES

☐ Allow duplicates and preserve column order in selection

Begin With

ALL COLUMNS NO COLUMNS

Include column names

Store Number X Zip Code X County Number X

Category X Item Number X

+

-

✓

Select **Ok**.

Set the rest of the configurations as follows:

Properties Project

Edit Metadata

Column

Selected columns:  
All columns  
Column names: Store  
Number, Zip  
Code, County  
Number, Category, Item  
Number

Launch column selector

Data type

Unchanged

Categorical

Make categorical

Fields

Unchanged

- Search for **Select Columns in Dataset** module. Connect the **Results Dataset** port of *Edit Metadata* module to the **Dataset** port of **Select Columns in Dataset** module. **Launch column selector** and select **All columns**, **All features** as shown below:

## Select columns

BY NAME

WITH RULES

☐ Allow duplicates and preserve column order in selection

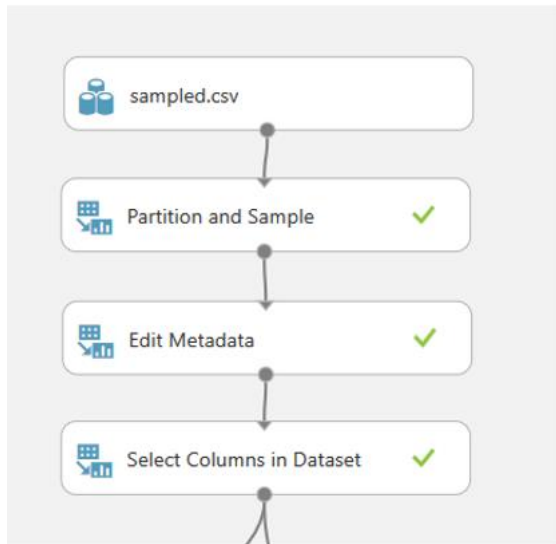
Begin With

ALL COLUMNS NO COLUMNS

Include all features + -

Save the experiment and run it.

The experiment would appear like this till now:



## Splitting the data

Now that the data is prepared, we would split the data into Training dataset and Testing dataset.

- Search for the **Split Data** module and drag it onto the Canvas.
- Connect the **Results dataset** output of the **Select Columns in Dataset** module to the input of the **Split Data** module.

On the properties pane of the **Split Data** module, configure the properties as shown below:

Properties Project

### Split Data

Splitting mode

Split Rows

Fraction of rows in the fi...

0.7

☒ Randomized split

Random seed

5416

Stratified split

False

## Training the model

Now that the data has been split, we can introduce the algorithm and then train the model.

- Search for **Linear Regression** module and drag it to the canvas. Set the property as shown below:

Properties Project

#### Linear Regression

Solution method

Ordinary Least Squares

L2 regularization weight

0.001

☒ Include intercept te...

Random number seed

☒ Allow unknown cat...

- Search for the **Cross Validate Model** module and drag it onto the canvas under the **Linear Regression** module. Connect the **Untrained model** output port of the **Linear Regression** module to the **Untrained model** input port (Left input port) of the **Cross Validate Model** module. Connect the **Results dataset** output of the **Select Columns in Dataset** module to the **Dataset** (right input port) port of the **Cross Validate Model** module. Set the property as shown below:

Properties Project

#### Cross Validate Model

Label column

**Selected columns:**  
**Column names:** Sale  
(Dollars)

Launch column selector

Random seed

3467

- Search for the **Train Model** module and drag it onto the canvas under the **Cross Validate Model** module. Connect the **Untrained model** output port of the **Linear Regression** module to the **Untrained model** input port (Left input port) of the **Train Model** module. Connect the **Results dataset1 (left output port) port** of the **Split Model** module to the **Dataset port (right input port)** of the **Train Model** module. In the properties pane, **Launch column selector** and select the column **Sale (Dollars)** as below:

Select a single column

BY NAME

WITH RULES

Include ▾

column names ▾

Sale (Dollars) ✕

- Search for the **Tune Model Hyperparameters** module and drag it onto the canvas below the **Split Data** module. Connect the **Untrained model** output port of the **Linear Regression** module to the **Untrained model** input port (**Left input port**) of the **Tune Model Hyperparameters** module. Connect the **Results dataset1** (**left output port**) port of the **Split Model** module to the **Training dataset port** (**middle port**) of the **Tune Model Hyperparameters** module. Set the configurations of the **Tune Model Hyperparameters** module as:
  - **Specify parameter sweeping mode** : Random Sweep
  - **Maximum number of runs on random sweep** : 5
  - **Random seed** : 4567
  - **Label column** : Sale (Dollars)
  - **Metric for measuring performance of classification** : Accuracy
  - **Metric for measuring performance of regression** : Root of mean squared error

Properties Project

▲ Tune Model Hyperparameters...

Specify parameter sweepin...

Random sweep ▾

Maximum number of ...

5

Random seed

4567

Label column

Selected columns:  
Column names: Sale  
(Dollars)

Launch column selector

Metric for measuring ...

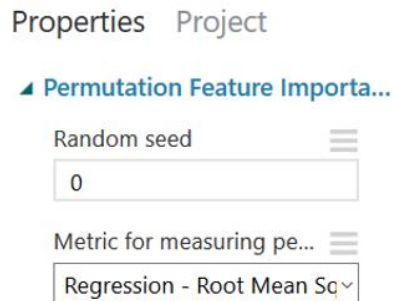
Accuracy ▾

Metric for measuring ...

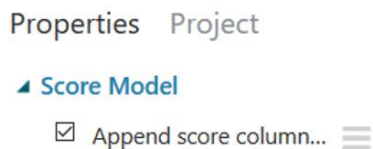
Root of mean squared e ▾



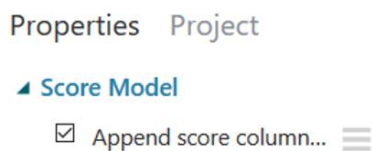
- Search for the **Permutation Features Importance** module and drag it onto the canvas. Connect the **Trained best model port (right output port)** of the **Tune Model Hyperparameters** module to the **Trained model input port (left input port)** of the **Permutation Features Importance** module. Connect the **Results dataset2 port (right output port)** of the **Split Data** module to the **Test data port (right input port)** of the **Permutation Features Importance** module. Set the property of the **Permutation Features Importance** module as shown below:



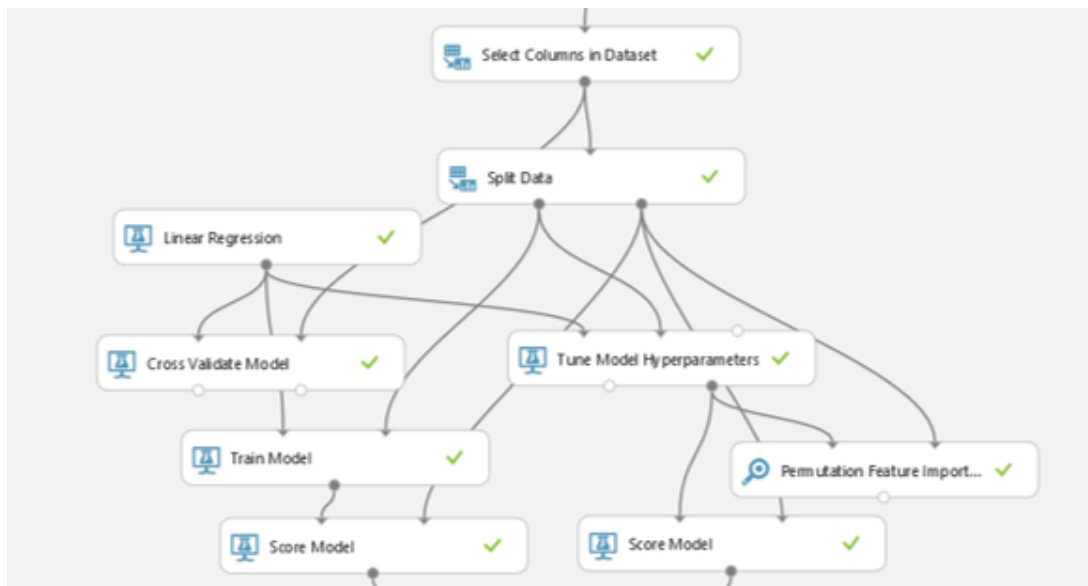
- Search for **Score Model** module and drag it onto the canvas under the **Train Model** module. Connect the **output port** of the **Train Model** module to the **Trained model input port (left input port)** of the **Score Model** module. Connect the **Results dataset2 port (right output port)** of the **Split Data** module to the **Dataset input port (right input port)** of the **Score Model** module. Set the properties as follows:



- Search for **Score Model** module again and drag this second **Score Model** module onto the canvas under the **Tune Model Hyperparameters** module. Connect the **Trained best model port (right output port)** of the **Tune Model Hyperparameters** module to the **Trained model input port (left input port)** of this second **Score Model** module. Connect the **Results dataset2 port (right output port)** of the **Split Data** module to the **Dataset input port (right input port)** of this **Score Model** module. Set the properties same as the first **Score Model** module, i.e.,



The experiment would look like this till now:



## Evaluating the model

- Search for the **Evaluate** module and drag it onto the canvas.
- Connect the **output port** of the **Score Model** module on the left side to the **left input port** of the **Evaluate** module. Connect the **output port** of the second **Score Model** module to the **right input port** of the **Evaluate** module. It would appear as below:



- Save and run the experiment.
- When the experiment has finished, **Visualize** the output from the **Evaluate** module. You would see that both the **Cross Validate Model** and **Tune Model Hyperparameters** have performed equally.

IOWA - Linear Regression - Cross Validation +Tuning... > Evaluate Model > Evaluation results

### Metrics

Mean Absolute Error	91.888803
Root Mean Squared Error	137.145669
Relative Absolute Error	0.215055
Relative Squared Error	0.074045
Coefficient of Determination	0.925955

### Metrics

Mean Absolute Error	91.888803
Root Mean Squared Error	137.145669
Relative Absolute Error	0.215055
Relative Squared Error	0.074045
Coefficient of Determination	0.925955

- You can also visualize the output from the **Permutation Feature Importance** module to see features with their scores of importance and can prune features with less importance. This might improve the model.



- Save a copy of this experiment by clicking **Save as** in the bottom of the page and name the experiment copy as **IOWA – Linear Regression – Cross validation + Tune Model Hyperparameters – New**.
- In the experiment **IOWA – Linear Regression – Cross validation + Tune Model Hyperparameters – New**, on the properties pane of **Select Columns in Dataset** module, **exclude** the columns *Invoice/Item Number*, *Store Number* and *Store Location* which had 0 and negative scores of feature importance.

## Properties Project

### ▲ Select Columns in Dataset

Select columns

**Selected columns:**

**All columns**

**Exclude column names:**

Invoice/Item  
Number,Store  
Number,Store Location

Launch column selector

It can be done as follows:

**Launch column selector > WITH RULES > Begin With > ALL COLUMNS > Exclude > column names > Invoice/Item Number, Store Number, Store Location**

## Select columns

BY NAME  
WITH RULES

☐ Allow duplicates and preserve column order in selection

Begin With  
**ALL COLUMNS** NO COLUMNS

Exclude

Invoice/Item Number ✕ Store Number ✕  
Store Location ✕

- Save and run the experiment.
- When the experiment has finished, **Visualize** the output from the **Evaluate** module to see if any improvement is there. It turned out that the **RMSE** value has actually increased and the **Coefficient of Determination** value decreased.

IOWA - Linear Regression - Cross Validation... > Evaluate Model > Evaluation results

### Metrics

Mean Absolute Error	176.678729
Root Mean Squared Error	240.403611
Relative Absolute Error	0.413495
Relative Squared Error	0.227516
Coefficient of Determination	0.772484

### Metrics

Mean Absolute Error	176.678729
Root Mean Squared Error	240.403611
Relative Absolute Error	0.413495
Relative Squared Error	0.227516
Coefficient of Determination	0.772484

So, the model performed better before pruning the features.

## Boosted Decision Tree Regression

### Data Preparation

- Open a browser and browse to <https://studio.azureml.net>. Then sign in using the Microsoft account associated with your Azure ML account.
- Create a new blank experiment and give it the title **IOWA - Boosted Decision Tree Regression**.
- In the experiment items pane on the left, expand **Saved Datasets**, expand **My Datasets**, and drag **sample.csv** to the experiment canvas.
- Search for **Partition and Sample** module and connect the output port of **sample.csv** module to the **Dataset** port of **Partition and Sample** module. In the **Properties** pane of **Partition and Sample** module, set the configurations as follows:

Properties Project

Partition and Sample

Partition or sample mode

Sampling

Rate of sampling

0.095

Random seed for sampl...

1234

Stratified split for sampling

False

- Search for **Edit Metadata** module and drag it to the canvas. Connect the **Results Dataset** (output) port of *Partition and Sample* module to the **Dataset** port of *Edit Metadata* module. In the Properties pane of *Edit Metadata* module, **Launch Column Selector. With Rules > Begin With > All Columns** and select **Include column names**. Select the columns *Store Number*, *Zip Code*, *County Number*, *Category*, *Item Number* as follows :

## Select columns

BY NAME

WITH RULES

☐ Allow duplicates and preserve column order in selection

Begin With

ALL COLUMNS

NO COLUMNS

Include

column names

Store Number X Zip Code X County Number X

Category X Item Number X

Select **Ok**.

Set the rest of the configurations as follows:

Properties Project

Edit Metadata

Column

Selected columns:  
All columns  
Column names: Store  
Number, Zip  
Code, County  
Number, Category, Item  
Number

Launch column selector

Data type

Unchanged

Categorical

Make categorical

Fields

Unchanged

- Search for **Select Columns in Dataset** module. Connect the **Results Dataset** port of *Edit Metadata* module to the **Dataset** port of **Select Columns in Dataset** module. **Launch column selector** and select **All columns**, **All features** as shown below:

Select columns

BY NAME

WITH RULES

☐ Allow duplicates and preserve column order in selection

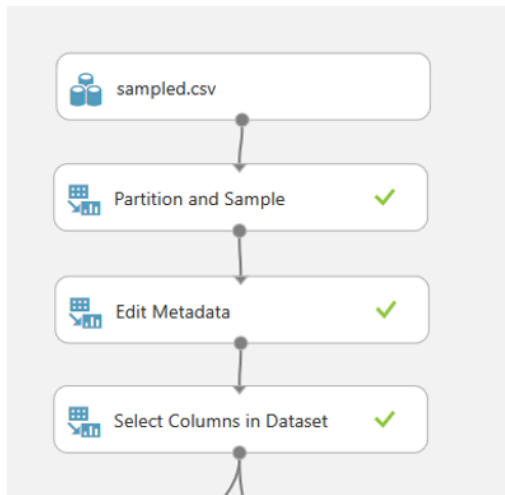
Begin With

ALL COLUMNS NO COLUMNS

Include all features + -

Save the experiment and run it.

The experiment would appear like this till now:



## Splitting the data

Now that the data is prepared, we would split the data into Training dataset and Testing dataset.

- Search for the **Split Data** module and drag it onto the Canvas.
- Connect the **Results dataset** output of the **Select Columns in Dataset** module to the input of the **Split Data** module.

On the properties pane of the **Split Data** module, configure the properties as shown below

Properties Project

### Split Data

Splitting mode

Split Rows

Fraction of rows in the fi...

0.7

☒ Randomized split

Random seed

5416

Stratified split

False

## Training the model

Now that the data has been split, we can introduce the algorithm and then train the model.

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

Properties Project

▲ Boosted Decision Tree Reg...

Create trainer mode

Single Parameter

Maximum number of ...

20

Minimum number of ...

10

Learning rate

0.02

Total number of trees ...

100

Random number seed

0

☒ Allow unknown c...

- Search for the **Cross Validate Model** module and drag it onto the canvas under the **Boosted Decision Tree Regression** module. Connect the **Untrained model** output port of the **Boosted Decision Tree Regression** module to the **Untrained model input port (Left input port)** of the **Cross Validate Model** module. Connect the **Results dataset** output of the **Select Columns in Dataset** module to the **Dataset (right input port) port** of the **Cross Validate Model** module. Set the property as shown below:

Properties Project

▲ Cross Validate Model

Label column

**Selected columns:**

**Column names:** Sale  
(Dollars)

Launch column selector

Random seed

3467

- Search for the **Train Model** module and drag it onto the canvas under the **Cross Validate Model** module. Connect the **Untrained model** output port of **Boosted Decision Tree Regression** module to the **Untrained model input port (Left input port)** of the **Train Model** module. Connect the **Results dataset1 (left output port) port** of the **Split Model** module to the **Dataset port (right input port)** of the **Train Model** module. In the properties pane, **Launch column selector** and select the column **Sale (Dollars)**.



- Search for the **Tune Model Hyperparameters** module and drag it onto the canvas below the **Split Data** module. Connect the **Untrained model** output port of the **Boosted Decision Tree Regression** module to the **Untrained model** input port (**Left input port**) of the **Tune Model Hyperparameters** module. Connect the **Results dataset1 (left output port)** port of the **Split Model** module to the **Training dataset port (middle input port)** of the **Tune Model Hyperparameters** module. Connect the **Results dataset2 (right output port)** port of the **Split Model** module to the **Optional validation dataset port (right input port)** of the **Tune Model Hyperparameters** module. Set the configurations of the **Tune Model Hyperparameters** module as below :

Properties Project

▲ Tune Model Hyperparam...

Specify parameter sweepin...

Random sweep ▼

Maximum number of ...

10

Random seed

4567

Label column

Selected columns:  
Column names: Sale  
(Dollars)

Launch column selector

Metric for measuring ...

Accuracy ▼

Metric for measuring ...

Root of mean squared e ▼

- Search for the **Permutation Features Importance** module and drag it onto the canvas. Connect the **Trained best model port (right output port)** of the **Tune Model Hyperparameters** module to the **Trained model input port (left input port)** of the **Permutation Features Importance** module. Connect the **Results dataset2 port (right output port)** of the **Split Data** module to the **Test data port (right input port)** of the **Permutation Features Importance** module. Set the property of the **Permutation Features Importance** module as shown below:

Properties Project

▲ Permutation Feature Importa...

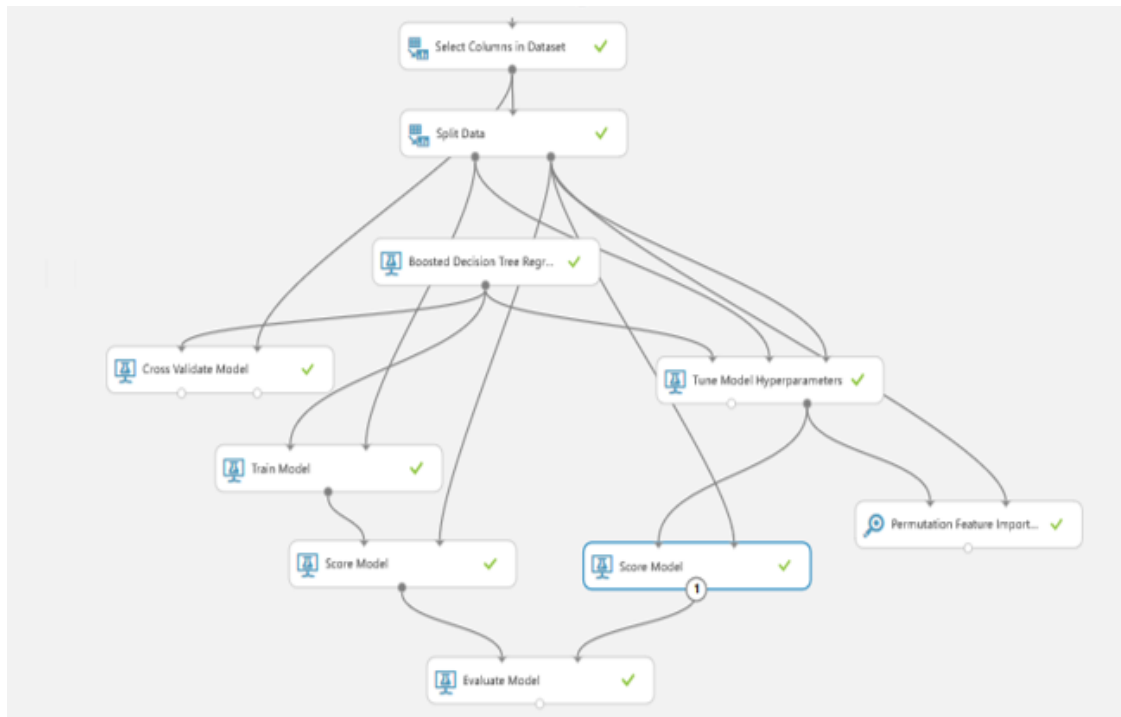
Random seed

0

Metric for measuring pe...

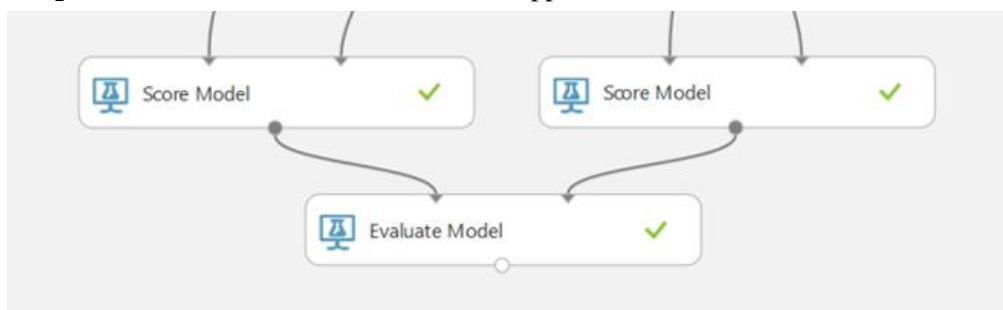
Regression - Root Mean Sq ▼

- Search for **Score Model** module and drag it onto the canvas under the **Train Model** module. Connect the **output port** of the **Train Model** module to the **Trained model input port (left input port)** of the **Score Model** module. Connect the **Results dataset2 port (right output port)** of the **Split Data** module to the **Dataset input port (right input port)** of the **Score Model** module.
- Search for **Score Model** module again and drag this second **Score Model** module onto the canvas under the **Tune Model Hyperparameters** module. Connect the **Trained best model port (right output port)** of the **Tune Model Hyperparameters** module to the **Trained model input port (left input port)** of this second **Score Model** module. Connect the **Results dataset2 port (right output port)** of the **Split Data** module to the **Dataset input port (right input port)** of this **Score Model** module. The experiment figure would look like this till now:



## Evaluating the model

- Search for the **Evaluate** module and drag it onto the canvas.
- Connect the **output port** of the **Score Model** module on the left side to the **left input port** of the **Evaluate** module. Connect the **output port** of the second **Score Model** module to the **right input port** of the **Evaluate** module. It would appear as below:



- Save and run the experiment.
- When the experiment has finished, **Visualize** the output from the **Evaluate** module. We would see that the **Tune Model Hyperparameter** has performed better, with lower **RMSE** value and higher **Coefficient of Determination** value.

IOWA - Boosted Decision Tree Regression > Evaluate Model > Evaluation results

#### Metrics

Mean Absolute Error	206.890108
Root Mean Squared Error	267.974043
Relative Absolute Error	0.484202
Relative Squared Error	0.282693
Coefficient of Determination	0.717307

#### Metrics

Mean Absolute Error	119.92497
Root Mean Squared Error	173.140216
Relative Absolute Error	0.28067
Relative Squared Error	0.118012
Coefficient of Determination	0.881988

- You can also visualize the output from the **Permutation Feature Importance** module to see features with their scores of importance and can prune features with less importance. This might improve the model.

IOWA - Boosted Decision Tree Regression > Permutation F

rows: 16  
columns: 2

Feature	Score
Bottles Sold	454.818329
Pack	81.371853
Vendor Number	75.608935
Category	54.132026
Bottle Volume (ml)	39.98075
Volume Sold (Liters)	6.885084
State Bottle Cost	2.324521
Store Number	0.948541
City	0.295601
Item Number	0.219255
Zip Code	0.167606
State Bottle Retail	0.015421
Invoice/Item Number	0
Date	0
Store Location	0
County Number	-0.122389

- Save a copy of this experiment by clicking **Save as** in the bottom of the page and name the experiment copy as **IOWA - Boosted Decision Tree Regression – New**.
- In the experiment **IOWA - Boosted Decision Tree Regression – New**, on the properties pane of **Select Columns in Dataset** module, **exclude** the columns *Invoice/Item Number*, *Date*, *Store Location*, *County Number*, *Zip Code*.

It can be done as follows:

**Launch column selector > WITH RULES > Begin With > ALL COLUMNS > Exclude > column names > Invoice/Item Number, Date, Store Location, County Number, Zip Code**

## Properties Project

### Select Columns in Dataset

Select columns

#### Selected columns:

All columns

#### Exclude column names:

Invoice/Item

Number,Date,Store

Location,County

Number,Zip Code

Launch column selector

- Save and run the experiment.
- When the experiment has finished, **Visualize** the output from the **Evaluate** module to see if any improvement is there. It turned out that the **RMSE** value has decreased by very negligible amount and the **Coefficient of Determination** has increased by very negligible amount.

IOWA - Boosted Decision Tree Regression - ... > Evaluate Model > Evaluation results

#### Metrics

Mean Absolute Error	206.890108
Root Mean Squared Error	267.974043
Relative Absolute Error	0.484202
Relative Squared Error	0.282693
Coefficient of Determination	0.717307

#### Metrics

Mean Absolute Error	119.084776
Root Mean Squared Error	171.97299
Relative Absolute Error	0.278704
Relative Squared Error	0.116426
Coefficient of Determination	0.883574

## Decision Forest Regression

### Data Preparation

- Open a browser and browse to <https://studio.azureml.net>. Then sign in using the Microsoft account associated with your Azure ML account.
- Create a new blank experiment and give it the title **IOWA - Decision Forest Regression**.
- In the experiment items pane on the left, expand **Saved Datasets**, expand **My Datasets**, and drag **sample.csv** to the experiment canvas.
- Search for **Partition and Sample** module and connect the output port of **sample.csv** module to the **Dataset** port of **Partition and Sample** module. In the **Properties** pane of **Partition and Sample** module, set the configurations as follows:

Properties Project

#### Partition and Sample

Partition or sample mode

Sampling

Rate of sampling

0.095

Random seed for sampl...

1234

Stratified split for sampling

False

- Search for **Edit Metadata** module and drag it to the canvas. Connect the **Results Dataset** (output) port of *Partition and Sample* module to the **Dataset** port of *Edit Metadata* module. In the Properties pane of *Edit Metadata* module, **Launch Column Selector. With Rules > Begin With > All Columns** and select **Include column names**. Select the columns *Store Number*, *Zip Code*, *County Number*, *Category*, *Item Number* as follows :

### Select columns

BY NAME

WITH RULES

☐ Allow duplicates and preserve column order in selection

**Begin With**

ALL COLUMNSNO COLUMNS

Includecolumn names

Store Number XZip Code XCounty Number X

Category XItem Number X

Select **Ok**.

Set the rest of the configurations as follows:

Properties Project

Edit Metadata

Column

Selected columns:  
All columns  
Column names: Store  
Number,Zip  
Code,County  
Number,Category,Item  
Number

Launch column selector

Data type

Unchanged

Categorical

Make categorical

Fields

Unchanged

- Search for **Select Columns in Dataset** module. Connect the **Results Dataset** port of *Edit Metadata* module to the **Dataset** port of **Select Columns in Dataset** module. **Launch column selector** and select **All columns**, **All features** as shown below:

Select columns

BY NAME

WITH RULES

☐ Allow duplicates and preserve column order in selection

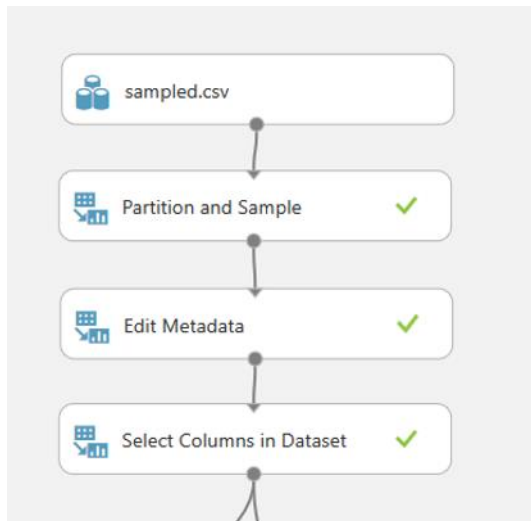
Begin With

ALL COLUMNS NO COLUMNS

Include all features + -

Save the experiment and run it.

The experiment would appear like this till now:



## Splitting the data

Now that the data is prepared, we would split the data into Training dataset and Testing dataset.

- Search for the **Split Data** module and drag it onto the Canvas.
- Connect the **Results dataset** output of the **Select Columns in Dataset** module to the input of the **Split Data** module.

On the properties pane of the **Split Data** module, configure the properties as shown below

[Properties](#) [Project](#)

### Split Data

Splitting mode

Split Rows

Fraction of rows in the fi...

0.7

☒ Randomized split

Random seed

5416

Stratified split

False

## Training the model

Now that the data has been split, we can introduce the algorithm and then train the model.

- Search for the **Decision Forest Regression** module and drag it onto the canvas. Set the property as shown below

Properties Project

#### Decision Forest Regression

Resampling method  
Bagging

Create trainer mode  
Single Parameter

Number of decision tr...  
8

Maximum depth of th...  
32

Number of random sp...  
128

Minimum number of ...  
4

☒ Allow unknown v...

- Search for the **Cross Validate Model** module and drag it onto the canvas under the **Decision Forest Regression** module. Connect the **Untrained model** output port of the **Decision Forest Regression** module to the **Untrained model input port (Left input port)** of the **Cross Validate Model** module. Connect the **Results dataset** output of the **Select Columns in Dataset** module to the **Dataset (right input port)** port of the **Cross Validate Model** module. Set the property as shown below:

Properties Project

#### Cross Validate Model

Label column

**Selected columns:**  
**Column names:** Sale  
(Dollars)

Launch column selector

Random seed  
3467

- Search for the **Train Model** module and drag it onto the canvas under the **Cross Validate Model** module. Connect the **Untrained model** output port of **Decision Forest Regression** module to the **Untrained model input port (Left input port)** of the **Train Model** module. Connect the **Results dataset1 (left output port)** port of the **Split Model** module to the **Dataset port (right input port)** of the **Train Model** module. In the properties pane, **Launch column selector** and select the column **Sale (Dollars)**.
- Search for the **Tune Model Hyperparameters** module and drag it onto the canvas below the **Split Data** module. Connect the **Untrained model** output port of the **Decision Forest Regression** module to the **Untrained model input port (Left input port)** of the **Tune Model Hyperparameters** module. Connect the **Results dataset1 (left output port)** port of the **Split Model** module to the **Training**



**dataset port (middle input port)** of the **Tune Model Hyperparameters** module. Connect the **Results dataset2 (right output port)** of the **Split Model** module to the **Optional validation dataset port (right input port)** of the **Tune Model Hyperparameters** module. Set the configurations of the **Tune Model Hyperparameters** module as below :

Properties Project

▲ Tune Model Hyperparame...

Specify parameter sweepin...  
Random sweep ▼

Maximum number of ...  
10

Random seed  
4567

Label column  
Selected columns:  
Column names: Sale (Dollars)

Launch column selector

Metric for measuring ...  
Accuracy ▼

Metric for measuring ...  
Root of mean squared e ▼

- Search for the **Permutation Features Importance** module and drag it onto the canvas. Connect the **Trained best model port (right output port)** of the **Tune Model Hyperparameters** module to the **Trained model input port (left input port)** of the **Permutation Features Importance** module. Connect the **Results dataset2 port (right output port)** of the **Split Data** module to the **Test data port (right input port)** of the **Permutation Features Importance** module. Set the property of the **Permutation Features Importance** module as shown below:

Properties Project

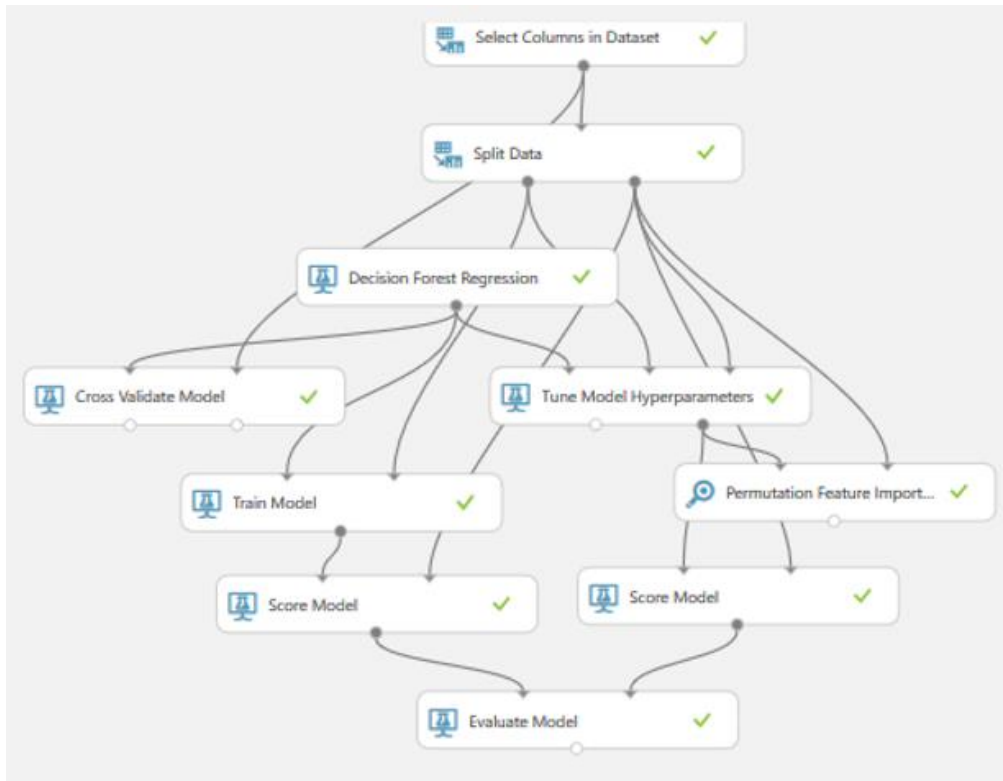
▲ Permutation Feature Importa...

Random seed  
0

Metric for measuring pe...  
Regression - Root Mean Sq ▼

- Search for **Score Model** module and drag it onto the canvas under the **Train Model** module. Connect the **output port** of the **Train Model** module to the **Trained model input port (left input port)** of the **Score Model** module. Connect the **Results dataset2 port (right output port)** of the **Split Data** module to the **Dataset input port (right input port)** of the **Score Model** module.
- Search for **Score Model** module again and drag this second **Score Model** module onto the canvas under the **Tune Model Hyperparameters** module. Connect the **Trained best model port (right**

**output port**) of the **Tune Model Hyperparameters** module to the **Trained model input port (left input port)** of this second **Score Model** module. Connect the **Results dataset2 port (right output port)** of the **Split Data** module to the **Dataset input port (right input port)** of this **Score Model** module. The experiment figure would look like this till now:





## Evaluating the model

- Search for the **Evaluate** module and drag it onto the canvas.
- Connect the **output port** of the **Score Model** module on the left side to the **left input port** of the **Evaluate** module. Connect the **output port** of the second **Score Model** module to the **right input port** of the **Evaluate** module. It would appear as below:



- Save and run the experiment.
- When the experiment has finished, **Visualize** the output from the **Evaluate** module. We would see that the **Tune Model Hyperparameter** has performed better, with lower **RMSE** value and higher **Coefficient of Determination** value.

rows	columns						
2	6						
		Negative Log Likelihood	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
view as							
							
		10152.450678	279.759514	338.365264	0.654744	0.450714	0.549286
		10058.559879	257.659719	314.118924	0.603022	0.388434	0.611566

- IOWA - Decision Forest Regression ▶ Permutation Feature

Feature	Score
Bottles Sold	168.902577
Volume Sold (Liters)	45.997167
Bottle Volume (ml)	27.05422
Pack	15.378501
State Bottle Cost	2.951751
Vendor Number	2.912818
County Number	1.690925
Category	1.339969
Item Number	1.106864
State Bottle Retail	0.719819
City	0.201144
Invoice/Item Number	0
Date	0
Zip Code	-0.030835
Store Location	-0.033807

- It can be done as follows:

**Launch column selector > WITH RULES > Begin With > ALL COLUMNS > Exclude > column names > Store Number, Store Location, Zip Code, Date, Invoice/Item Number**

#### ▲ Select Columns in Dataset



Select columns

**Selected columns:**  
**All columns**  
**Exclude column names:**  
 Store Number,Store  
 Location,Zip  
 Code,Date,Invoice/Item  
 Number

Launch column selector

- Save and run the experiment.
- When the experiment has finished, **Visualize** the output form the **Evaluate** module to see if any improvement is there. It turned out that the **RMSE** value has decreased by considerable amount and the **Coeffiecient of Determination** has increased. However, the **Cross Validate Model** performed better this after pruning the features.

IOWA - Decision Forest Regression - New > Evaluate Model > Evaluation results

rows		columns									
2		6									
		Negative Log Likelihood		Mean Absolute Error		Root Mean Squared Error		Relative Absolute Error		Relative Squared Error	
		Coefficient of Determination									
view as											
 											
		9721.790298		191.971813		246.037976		0.449287		0.238305	
		9866.819041		215.873656		271.932532		0.505227		0.291106	