Advances in Data Science & Architecture Assignment 3

Index

Goal:

To build out a web service for energy models for Finland. There will be one web service which will invoke the rest api of all the models. We have the following functions: -

1.Regression with 4 models (Linear regression, Neural Network, Random Forest and KNN) to predict the Normalized energy consumption.

2.Classification with 4 models (Logistic regression, Neural Network, Random Forest and KNN) to predict the Normalized energy consumption.

3.Clustering with 2 models (K means and Hierarchical) to predict the Building ID cluster.

We have trained the data without the year so the trained model can be used to for any year and can also be retrained for the future data. With our current capacity we will be taking the weather input and the building parameter input from the user.

For all the three we have a retraining model and a prediction model. The prediction model has been made into a web service.

Methodology

1) We have our data from Midterm

<table>
<tr><th>Variables</th><th>Class</th><th>Comment</th></tr>
<tr><td>Time</td><td>integer</td><td rowspan="10">Weather Data</td></tr>
<tr><td>TemperatureF</td><td>numeric</td></tr>
<tr><td>Dew.PointF</td><td>numeric</td></tr>
<tr><td>Humidity</td><td>integer</td></tr>
<tr><td>Sea.Level.PressureIn</td><td>numeric</td></tr>
<tr><td>VisibilityMPH</td><td>numeric</td></tr>
<tr><td>Wind.SpeedMPH</td><td>numeric</td></tr>
<tr><td>Gust.SpeedMPH</td><td>numeric</td></tr>
<tr><td>Events</td><td>integer</td></tr>
<tr><td>WindDirDegrees</td><td>integer</td></tr>
<tr><td>sta_code</td><td>integer</td><td rowspan="3">Building Data</td></tr>
<tr><td>day</td><td>integer</td></tr>
<tr><td>month</td><td>integer</td></tr>
</table>

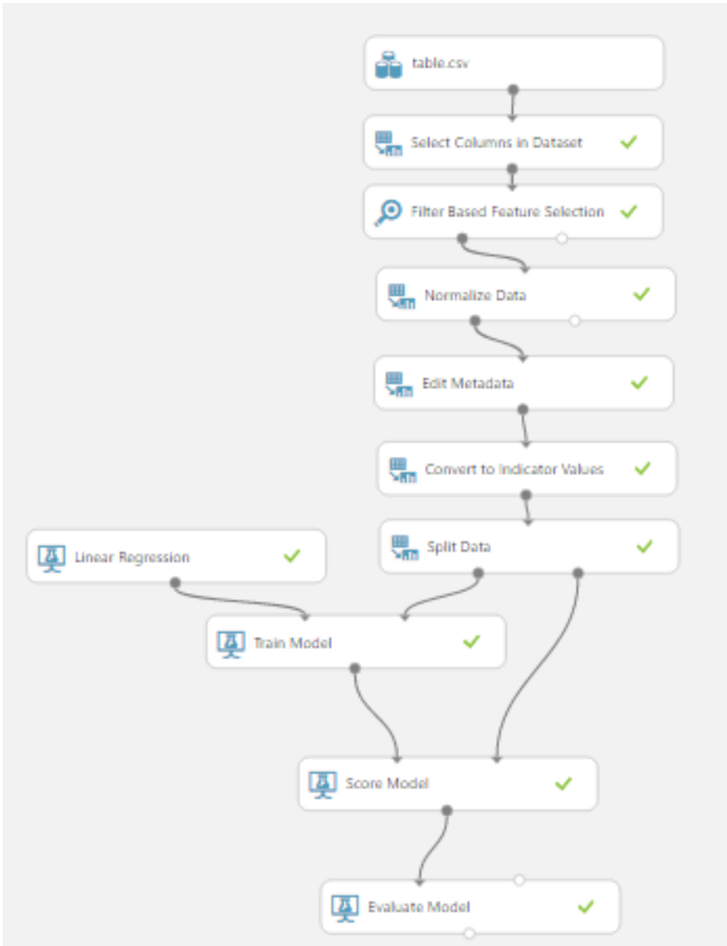
BuildingID	integer	
meternumb	integer	
type	integer	
Consumption	integer	
area_floor _m.sqr	integer	
DayofWeek	integer	
Weekday	integer	
Peakhour	integer	
mont	integer	
Normalised	numeric	Dependent Variable for regression
BaseHourRate	integer	Dependent Variable for classification

We have added this dataset to Azure ML

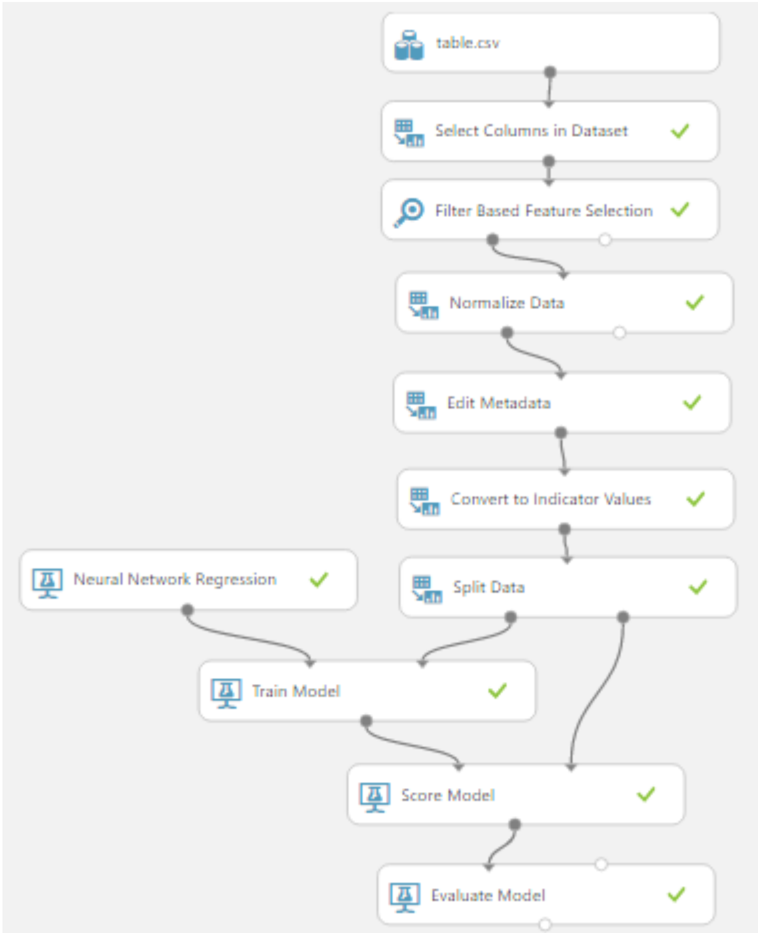
2) The next step is to make 4 predictive model

Regression

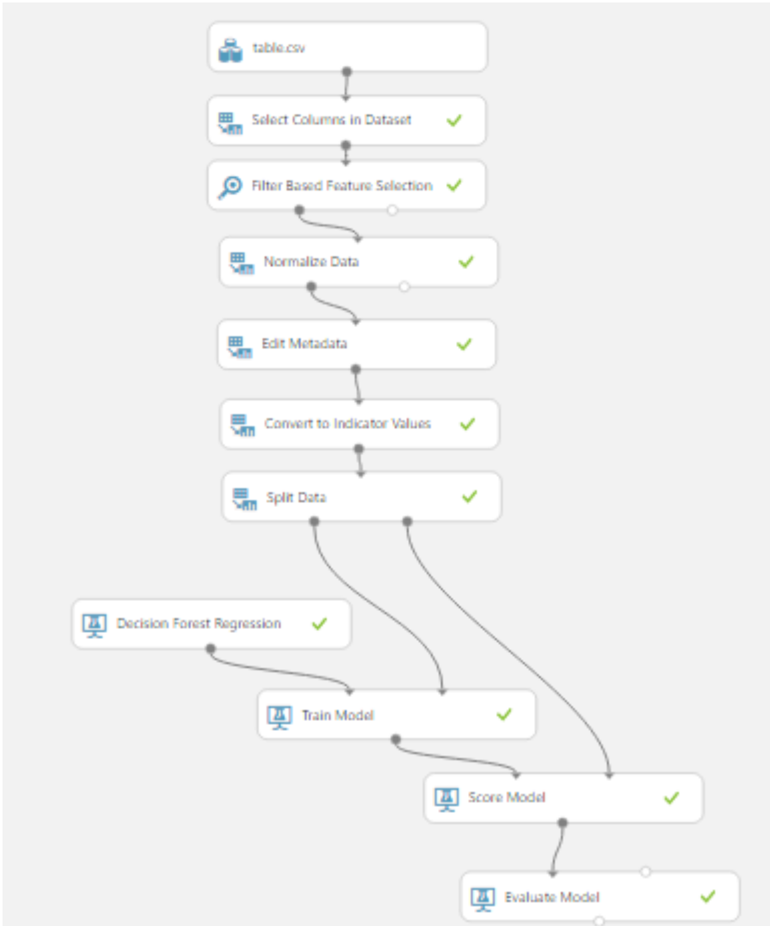
1. Linear Regression



2. Neural Network



3. Random Decision Forest



4. KNN

There is no module for KNN so we use the R script

Below mentioned: - # Map 1-based optional input ports to variables

dataset1 <- maml.mapInputPort(1) # class: data.frame

dataset2 <- maml.mapInputPort(2) # class: data.frame

library(kknn)

model <- train.kknn(Normalised ~ ., data = dataset1, kmax = 9)

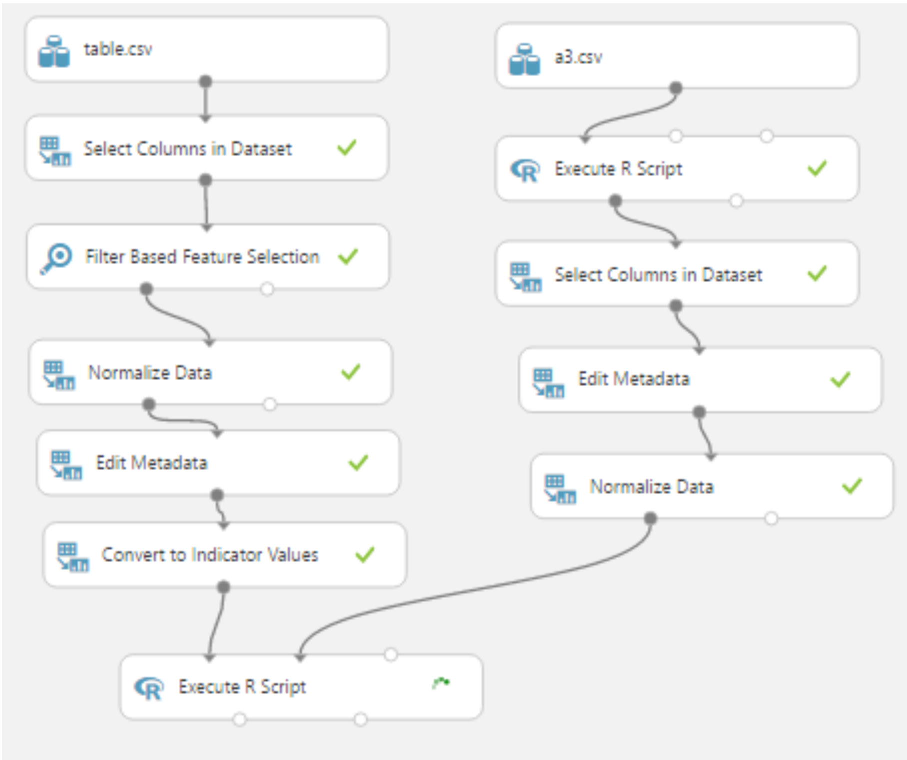
prediction <- predict(model, dataset2)

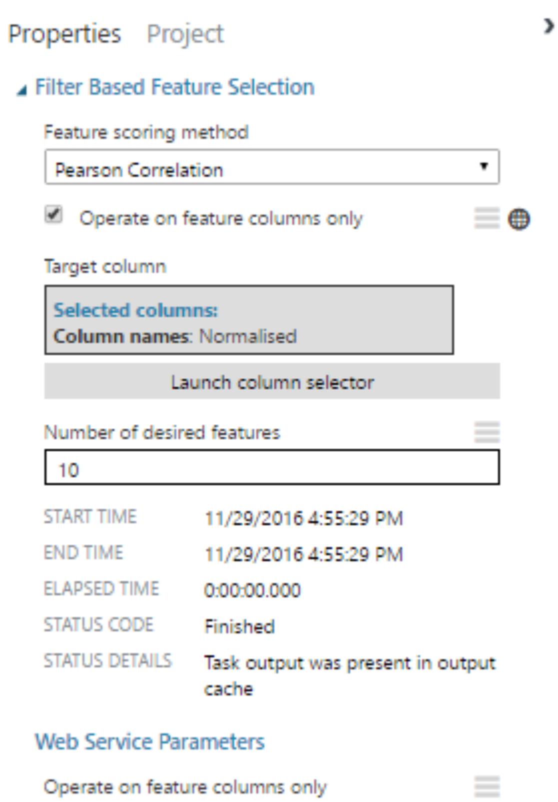
model <- train.kknn(Normalised ~ ., data = dataset1, kmax = 9)

prediction<-as.data.frame(prediction)

madel<-as.data.frame(model)

maml.mapOutputPort("prediction")





Perform feature selection for each model to reduce the independent variable and to get high co efficient of Determination for regression and Area under the curve (AUC) for Classification.

Pearson Correlation has been chosen to eliminate collinearity.

The number of variables have been chosen by iteration (starting from 25 and reducing it to 10)

Our Model after selection is

Normalised	sta_code	VisibilityMPH	Dew.PointF	TemperatureF	month	Events	Peakhour	Weekday	Wind.SpeedMPH	meternumb-1	meternumb-2	meternumb-3	meternumb-4	meternumb-8
0	0	0.2	0.396226	0.366667	0.181818	0.857143	1	1	0.333333	1	0	0	0	0
0.006911	0	0.2	0.867925	0.783333	0.454545	0.428571	0	1	0.101449	1	0	0	0	0
0.000549	0	0.2	0.641509	0.55	0.909091	0.428571	1	1	0.133333	0	1	0	0	0
0.112918	0	0.2	0.641509	0.6	0.272727	0.428571	1	1	0.234783	0	0	0	1	0
0.01626	0	0.2	0.924528	0.866667	0.454545	0	0	1	0	1	0	0	0	0
0.027451	0	0.2	0.849057	0.733333	0.454545	0.428571	0	1	0.066667	1	0	0	0	0
0.016941	0	0.13871	0.528302	0.55	0.272727	0	0	1	0.066667	1	0	0	0	0
0.00546	0	0.080645	0.849057	0.7	0.818182	0	1	1	0.168116	1	0	0	0	0
0.00259	0	0.2	0.867925	0.766667	0.454545	0.428571	0	1	0.2	1	0	0	0	0
0.005956	0	0.180645	0.528302	0.416667	0.090909	0.857143	0	0	0	1	0	0	0	0
0	0	0.2	0.773585	0.716667	0.363636	0	1	1	0.066667	1	0	0	0	0
0	0	0.2	0.566038	0.466667	1	0.428571	1	1	0	1	0	0	0	0
0.007894	0	0.2	0.811321	0.716667	0.545455	0.428571	1	0	0.168116	1	0	0	0	0
0.038462	0	0.2	0.622642	0.566667	0.909091	0.428571	1	1	0.168116	1	0	0	0	0
0.005956	0	0.2	0.339623	0.4	0.181818	0.857143	1	0	0.133333	1	0	0	0	0
0.333333	1	0.2	0.641509	0.55	0.272727	0.428571	0	1	0.2	1	0	0	0	0
0.025562	0	0.2	0.90566	0.8	0.545455	0.428571	1	1	0.133333	1	0	0	0	0

And the Coefficient of determination for

1. Linear regression

Metrics

Mean Absolute Error	0.021247
Root Mean Squared Error	0.040998
Relative Absolute Error	0.839013
Relative Squared Error	0.653143
Coefficient of Determination	0.346857

2. Neural Network

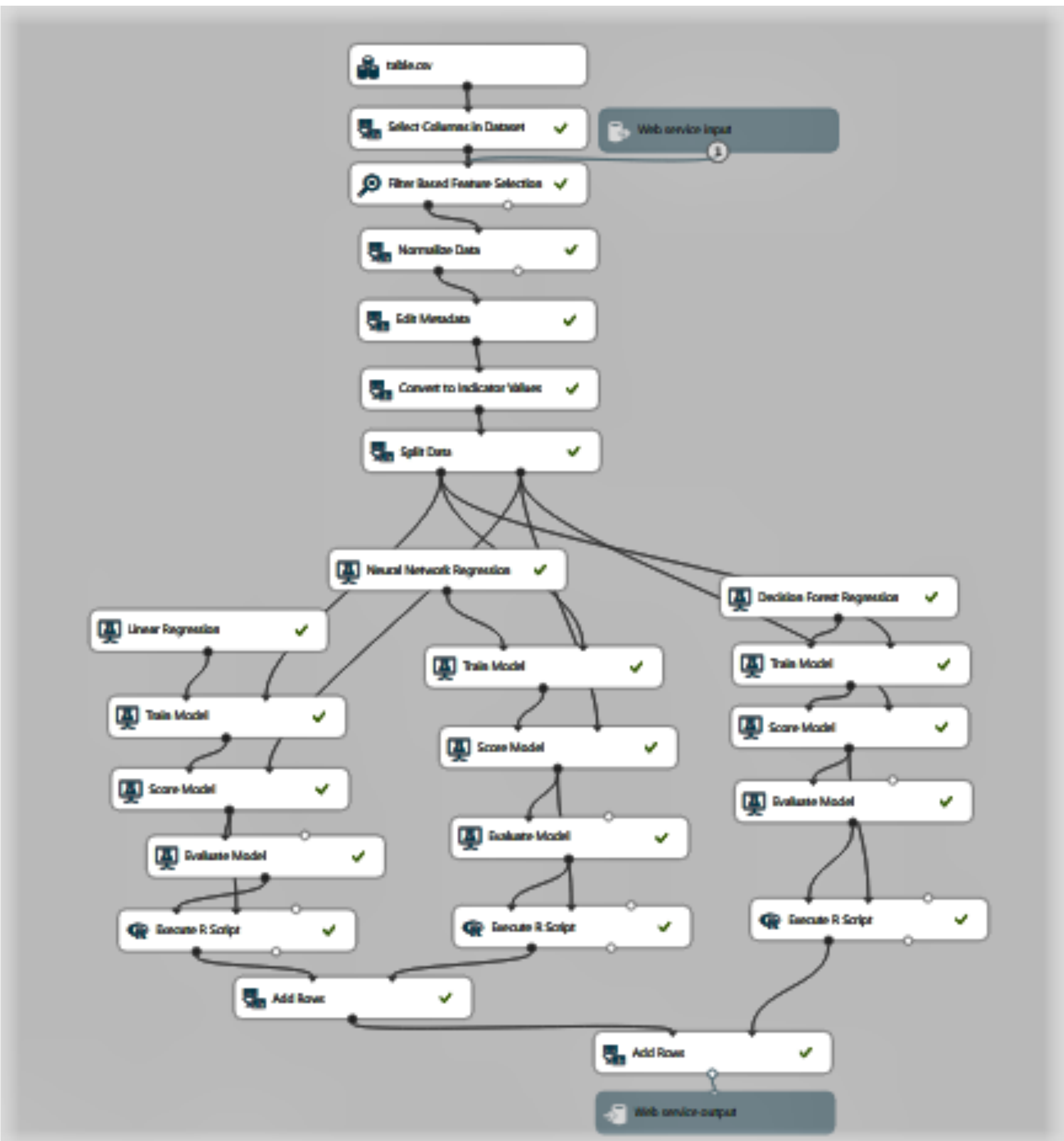
Metrics

Mean Absolute Error	0.020141
Root Mean Squared Error	0.039193
Relative Absolute Error	0.795328
Relative Squared Error	0.5969
Coefficient of Determination	0.4031

3. Random Forest

view as	Negative Log Likelihood	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
	11204748.62927	0.021708	0.049093	0.857201	0.936506	0.063494

3) Next merge the 3 models into one predictive web service

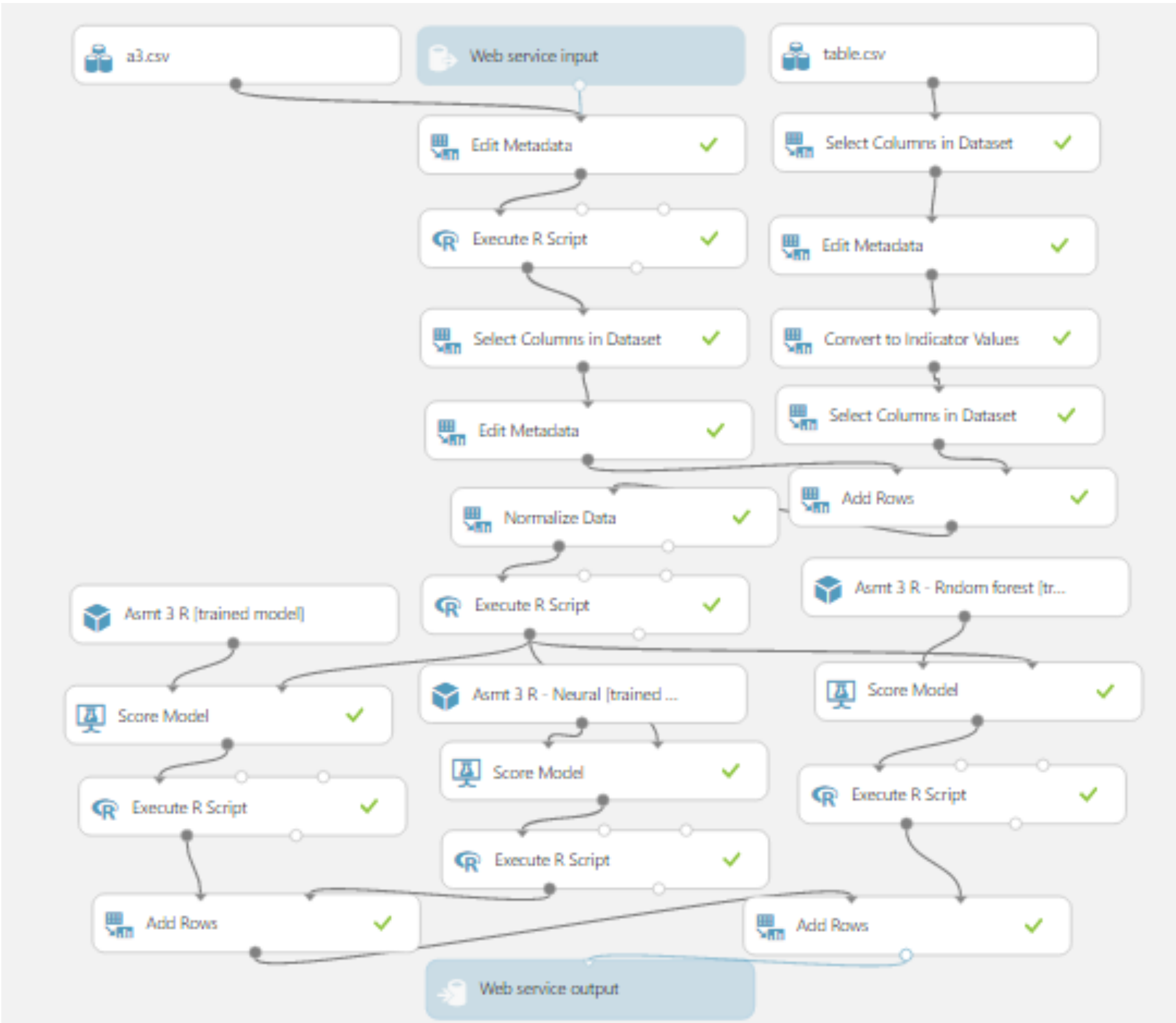


you can retrain it with new data and we get a comparative result of accuracy for all three

Algorithm	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Linear Regression	0.021247	0.040998	0.839013	0.653143	0.346857
Neural Network Regression	0.020092	0.039206	0.793386	0.597293	0.402707
Random Forest Regression	0.021708	0.049093	0.857201	0.936506	0.063494

The best regression model is Neural Network as you can see from the co-efficient of Determination.

4) The next step is to convert the models into trained modules to be used for prediction. We have modules for Linear regression, Random Forest and Neural network.



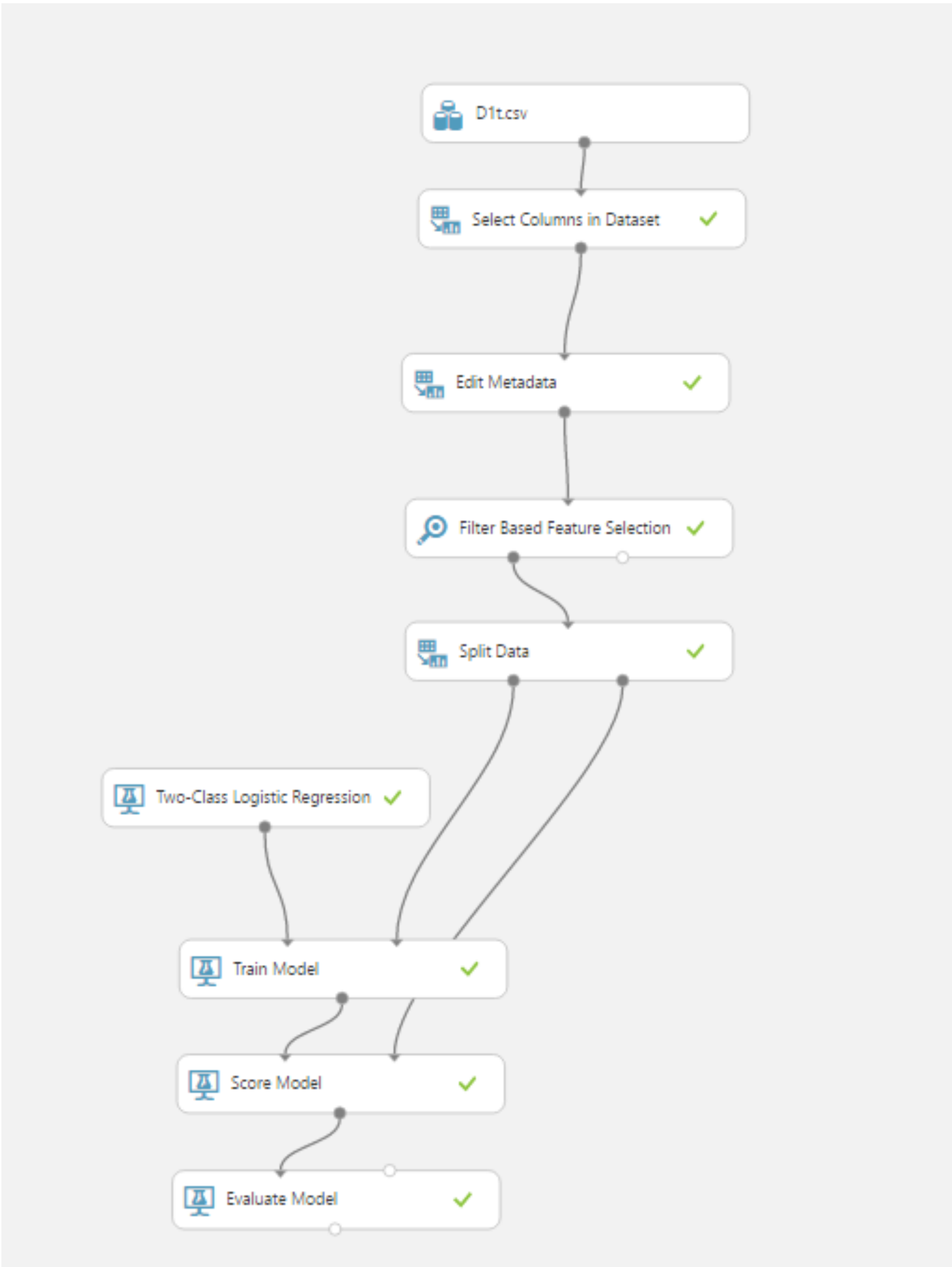
The Web Service Output will have the following output rows 3 columns 2

Algorithm	dataset[1, 15]
Linear Regression	0.063563
Neural Network Regression	0.314257
Random Forest Regression	0.033811

Here the data set is the predicted Normalised (consumption in KWh/Sq mt)

Classification:

1. Logistic Regression:



1a. Feature selection: Perform feature selection for each model to reduce the independent variable and to get high Area under the curve (AUC) for Classification. *Pearson Correlation has been chosen to eliminate collinearity. The number of variables have been chosen by iteration (starting from 25 and reducing it to 10)*

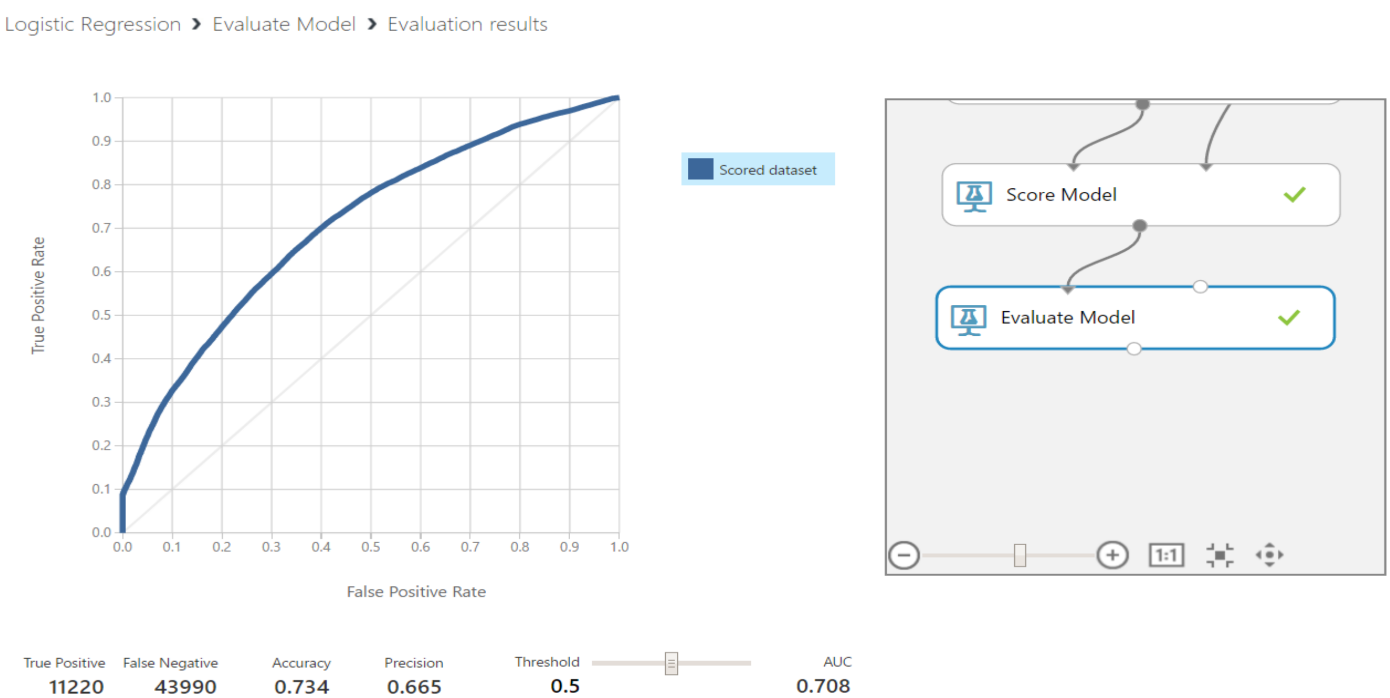
Logistic Regression › Filter Based Feature Selection › Filtered dataset

rows 621816 columns 11

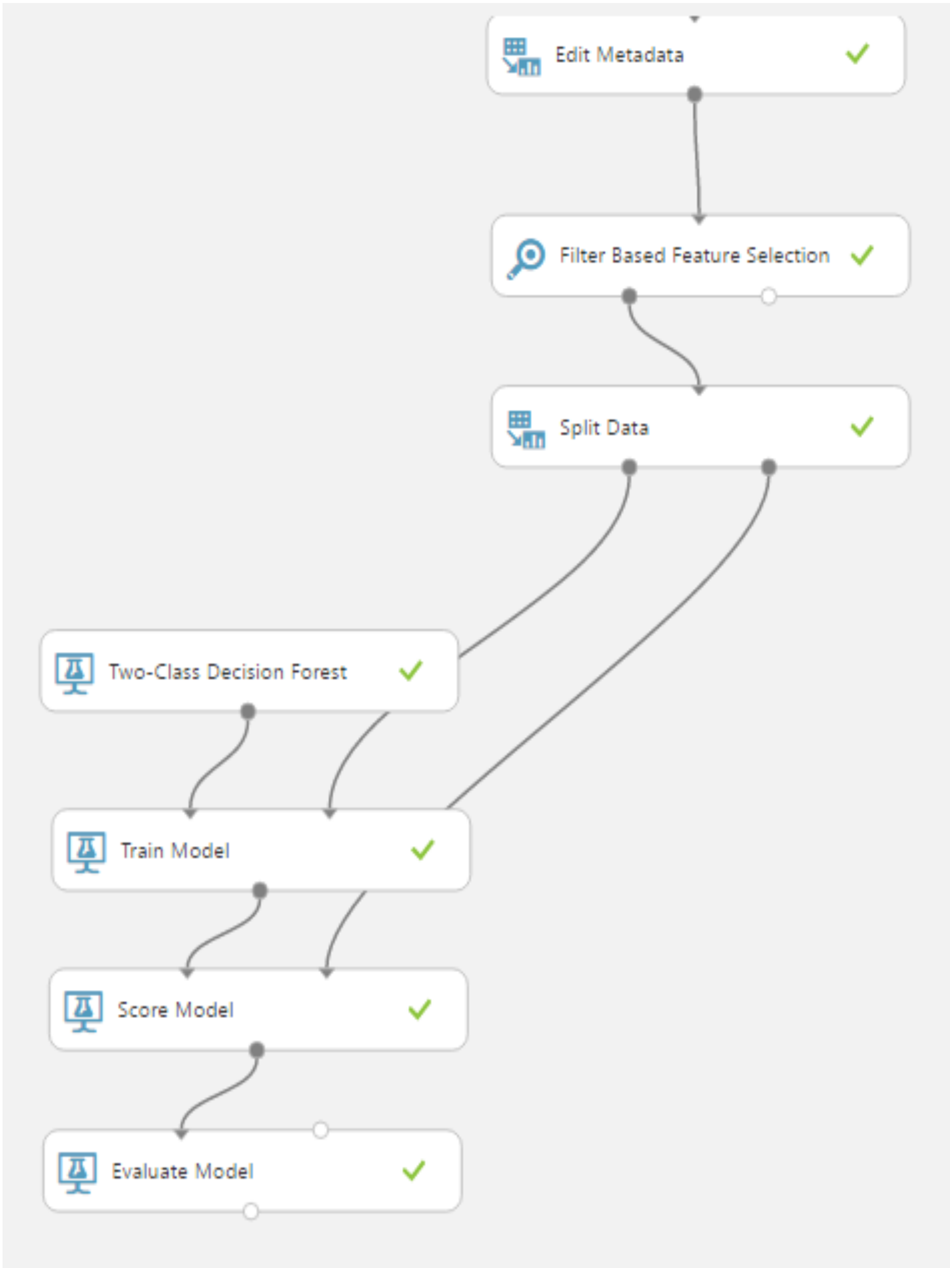
view as

BaseHourRate	sta_code	Dew.PointF	TemperatureF	month	Peakhour	Events	VisibilityMPH	Weekday	type	Wind.SpeedMPH
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
1	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
1	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
1	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459

1b. Evaluation Results: A model with an AUC above 70% is considered to be a good model.



2.Decision forest:



2a. Feature Selection: Perform feature selection for each model to reduce the independent variable and to get high Area under the curve (AUC) for Classification. *Pearson Correlation has been chosen to eliminate collinearity. The number of variables have been chosen by iteration (starting from 25 and reducing it to 10)*

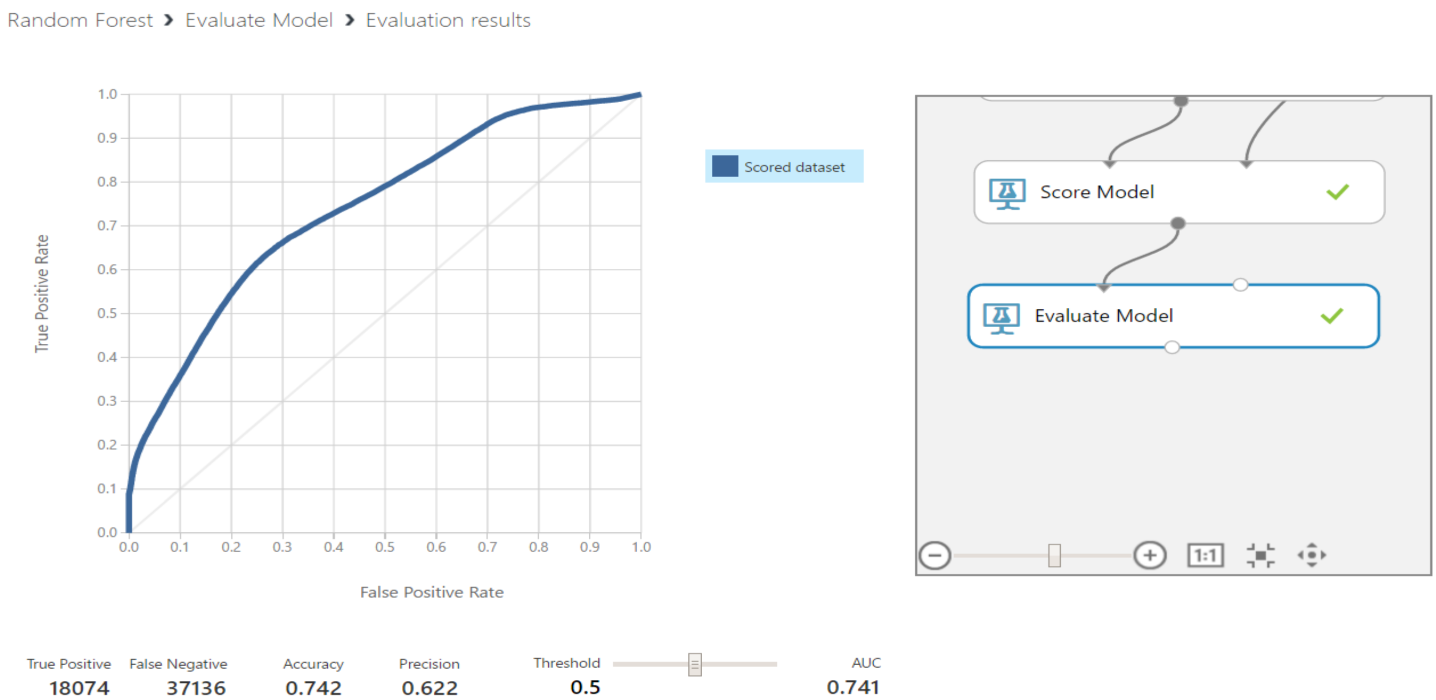
Random Forest › Filter Based Feature Selection › Filtered dataset

rows 621816 columns 11

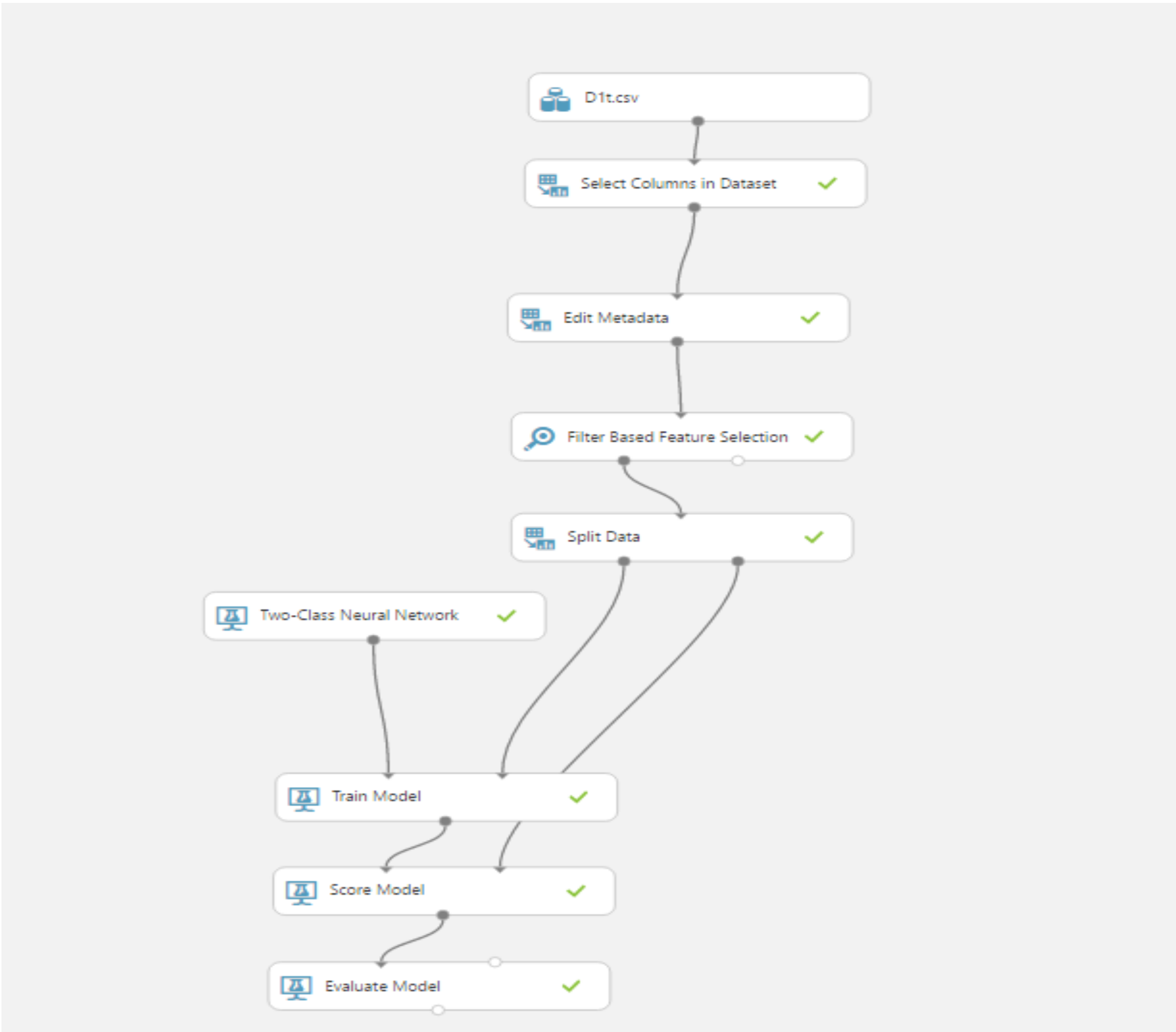
view as

BaseHourRate	sta_code	Dew.PointF	TemperatureF	month	Peakhour	Events	VisibilityMPH	Weekday	type	Wind.SpeedMPH
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
1	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
1	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
1	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459

2b. Evaluation Results: A model with an AUC above 70% is considered to be a good model.



3.Neural Networks:



3a. Feature Selection: Perform feature selection for each model to reduce the independent variable and to get high Area under the curve (AUC) for Classification. *Pearson Correlation has been chosen to eliminate collinearity. The number of variables have been chosen by iteration (starting from 25 and reducing it to 10)*

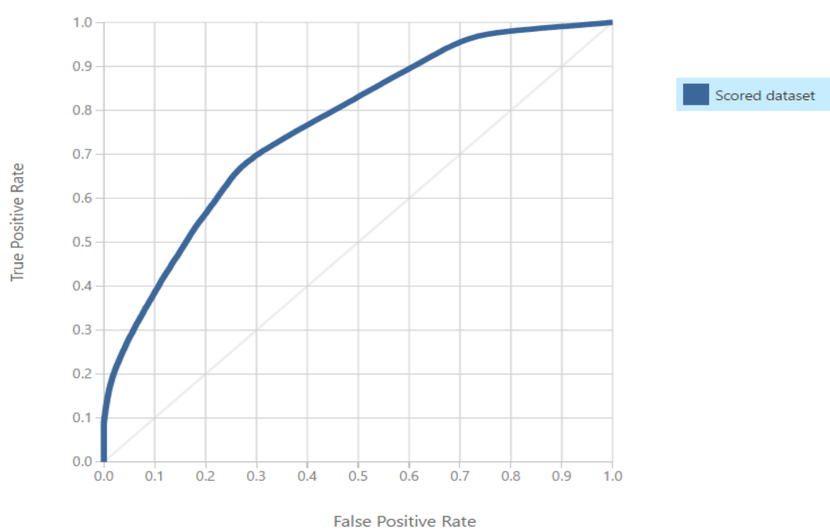
Neural Network › Filter Based Feature Selection › Filtered dataset

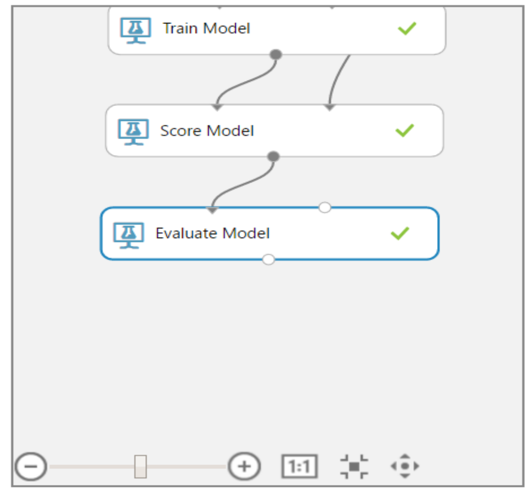
rows 621816 columns 11

BaseHourRate	sta_code	Dew.PointF	TemperatureF	month	Peakhour	Events	VisibilityMPH	Weekday	type	Wind.SpeedMPH
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
1	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
1	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
1	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000335	0.000459
0	0.000335	0.000744	0.000744	0.000335	0.000323	0.000371	0.00036	0.000335	0.000347	0.000459

3b. Evaluation Results: A model with an AUC above 70% is considered to be a good model.

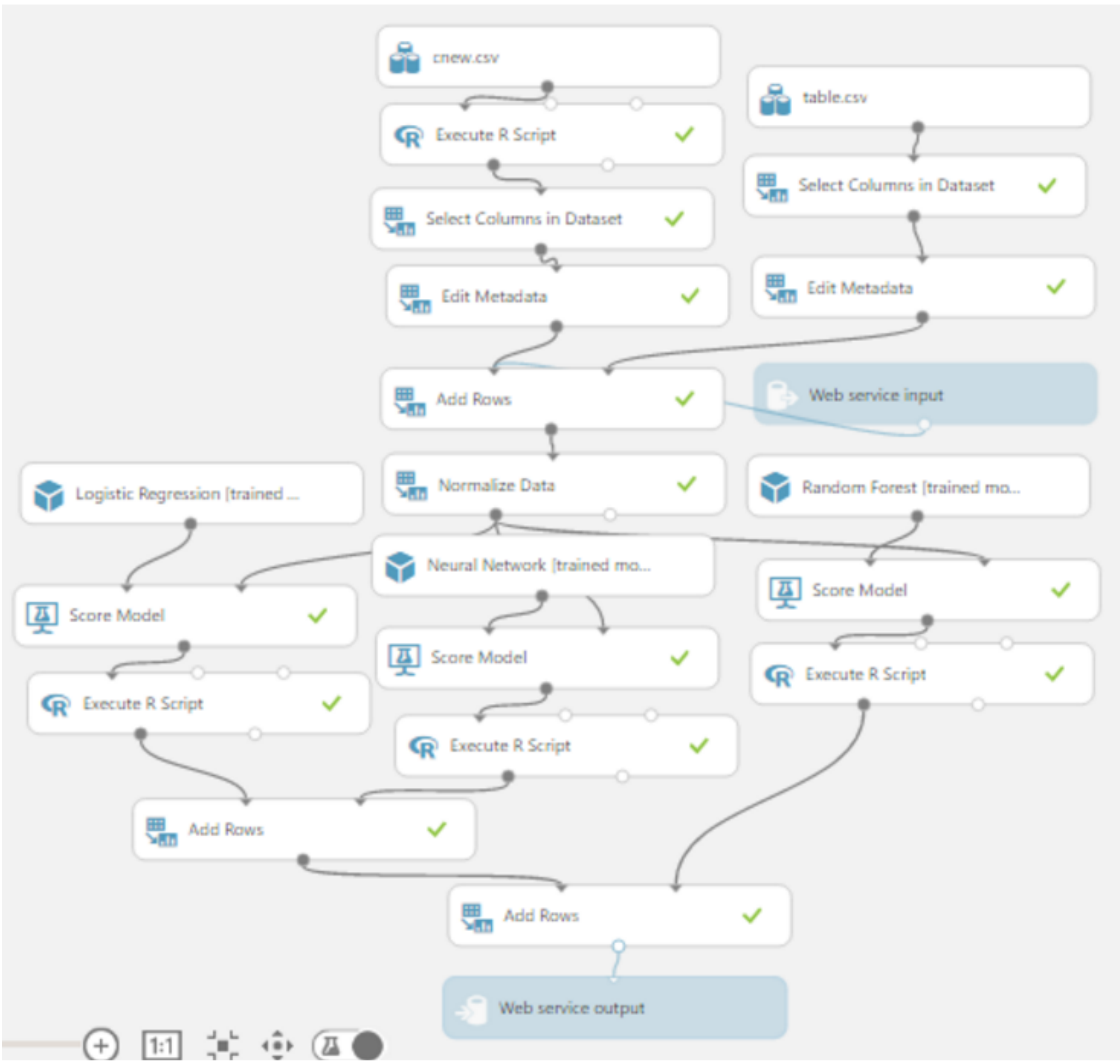
Neural Network › Evaluate Model › Evaluation results





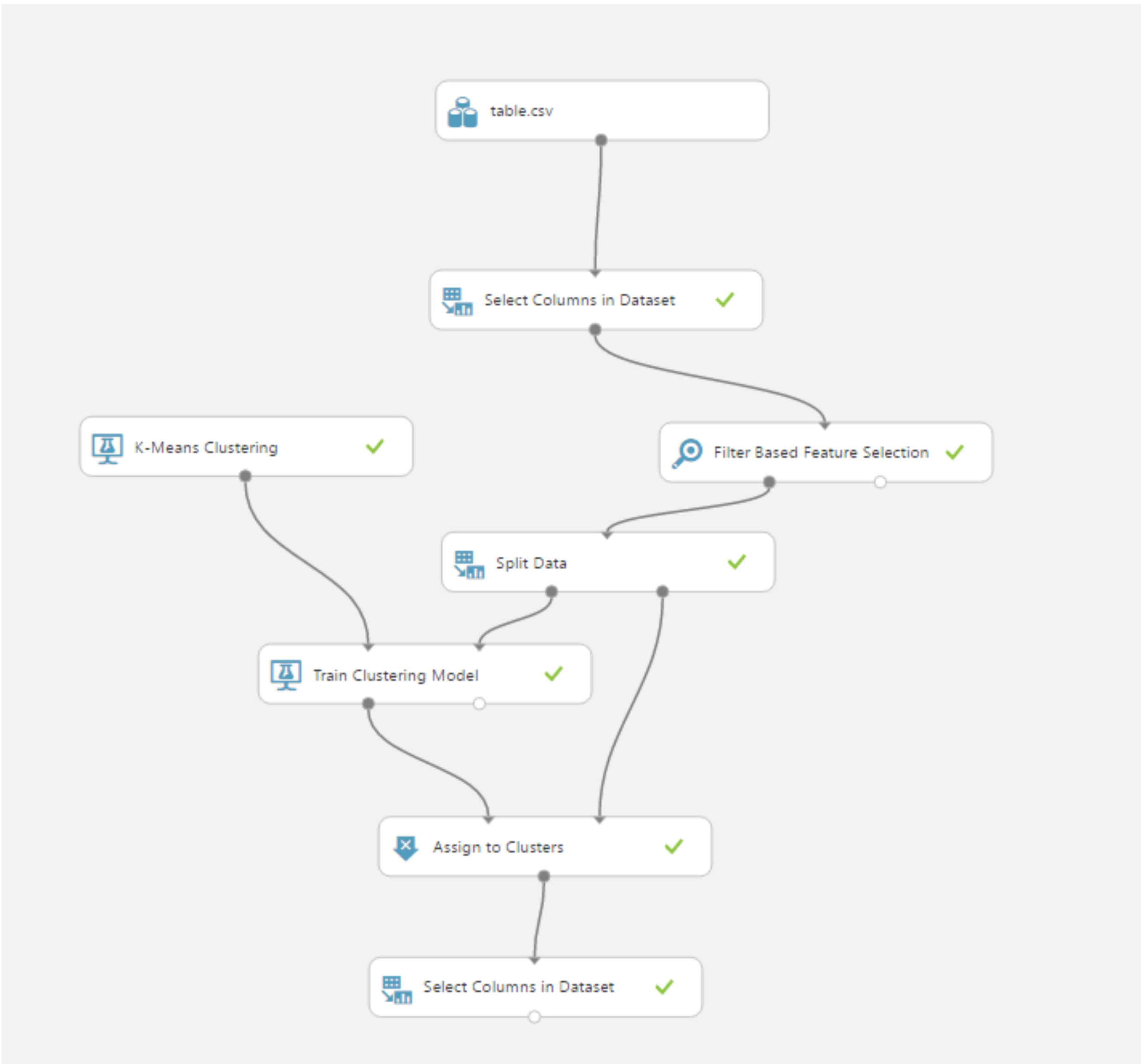
True Positive	False Negative	Accuracy	Precision	Threshold	AUC
31687	23523	0.729	0.540	0.5	0.766

4. The next step is to convert the models into trained modules to be used for classification. We have modules for Linear regression, Random Forest and Neural network.

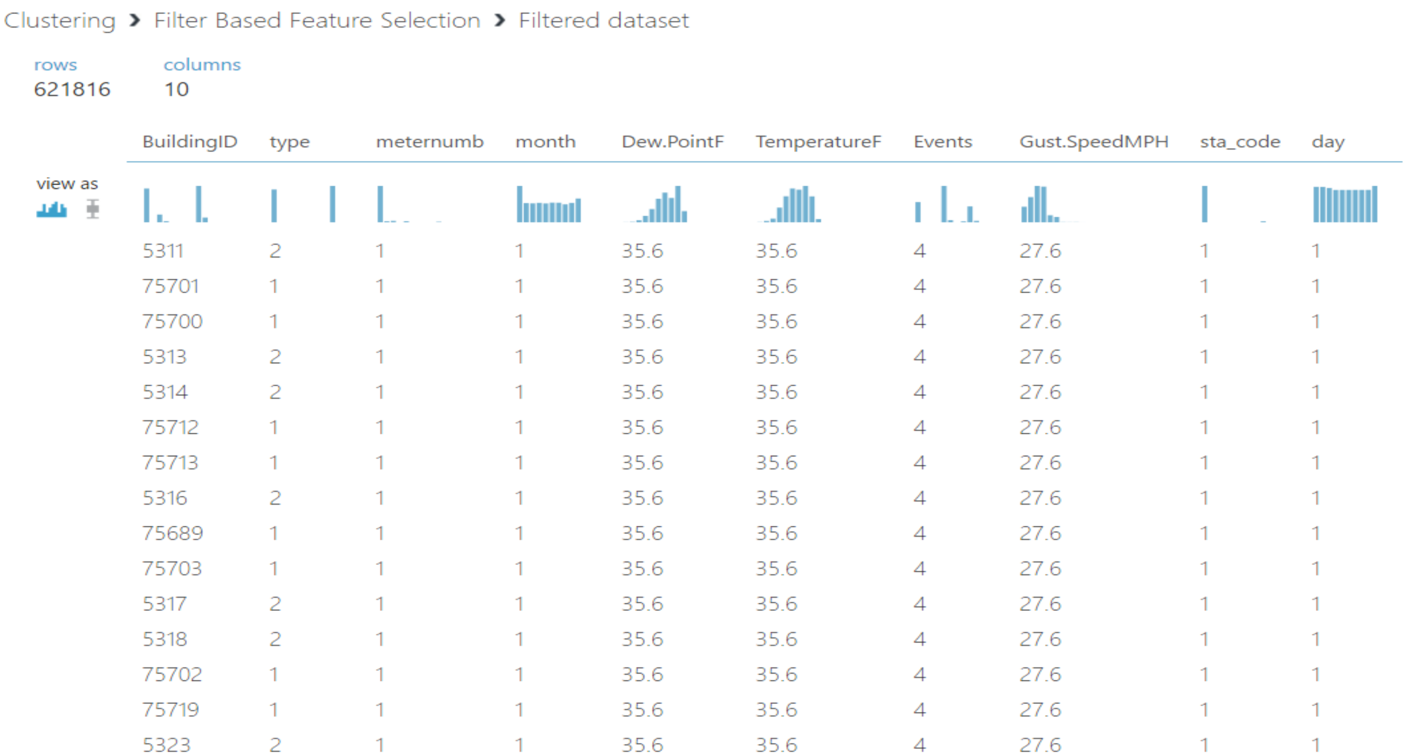


Clustering:

5. K means Clustering: Training Experiment



5a. Feature Selection: Perform feature selection for each model to reduce the independent variable and to get high Area under the curve (AUC) for Classification. *Pearson Correlation has been chosen to eliminate collinearity. The number of variables have been chosen by iteration (starting from 25 and reducing it to 10)*



Clustering › Filter Based Feature Selection › Filtered dataset

rows 621816 columns 10

view as

BuildingID	type	meternumb	month	Dew.PointF	TemperatureF	Events	Gust.SpeedMPH	sta_code	day
5311	2	1	1	35.6	35.6	4	27.6	1	1
75701	1	1	1	35.6	35.6	4	27.6	1	1
75700	1	1	1	35.6	35.6	4	27.6	1	1
5313	2	1	1	35.6	35.6	4	27.6	1	1
5314	2	1	1	35.6	35.6	4	27.6	1	1
75712	1	1	1	35.6	35.6	4	27.6	1	1
75713	1	1	1	35.6	35.6	4	27.6	1	1
5316	2	1	1	35.6	35.6	4	27.6	1	1
75689	1	1	1	35.6	35.6	4	27.6	1	1
75703	1	1	1	35.6	35.6	4	27.6	1	1
5317	2	1	1	35.6	35.6	4	27.6	1	1
5318	2	1	1	35.6	35.6	4	27.6	1	1
75702	1	1	1	35.6	35.6	4	27.6	1	1
75719	1	1	1	35.6	35.6	4	27.6	1	1
5323	2	1	1	35.6	35.6	4	27.6	1	1

5b. Evaluation Results: The buildings are assigned to their respective clusters based on the building data.

Clustering › Select Columns in Dataset › Results dataset

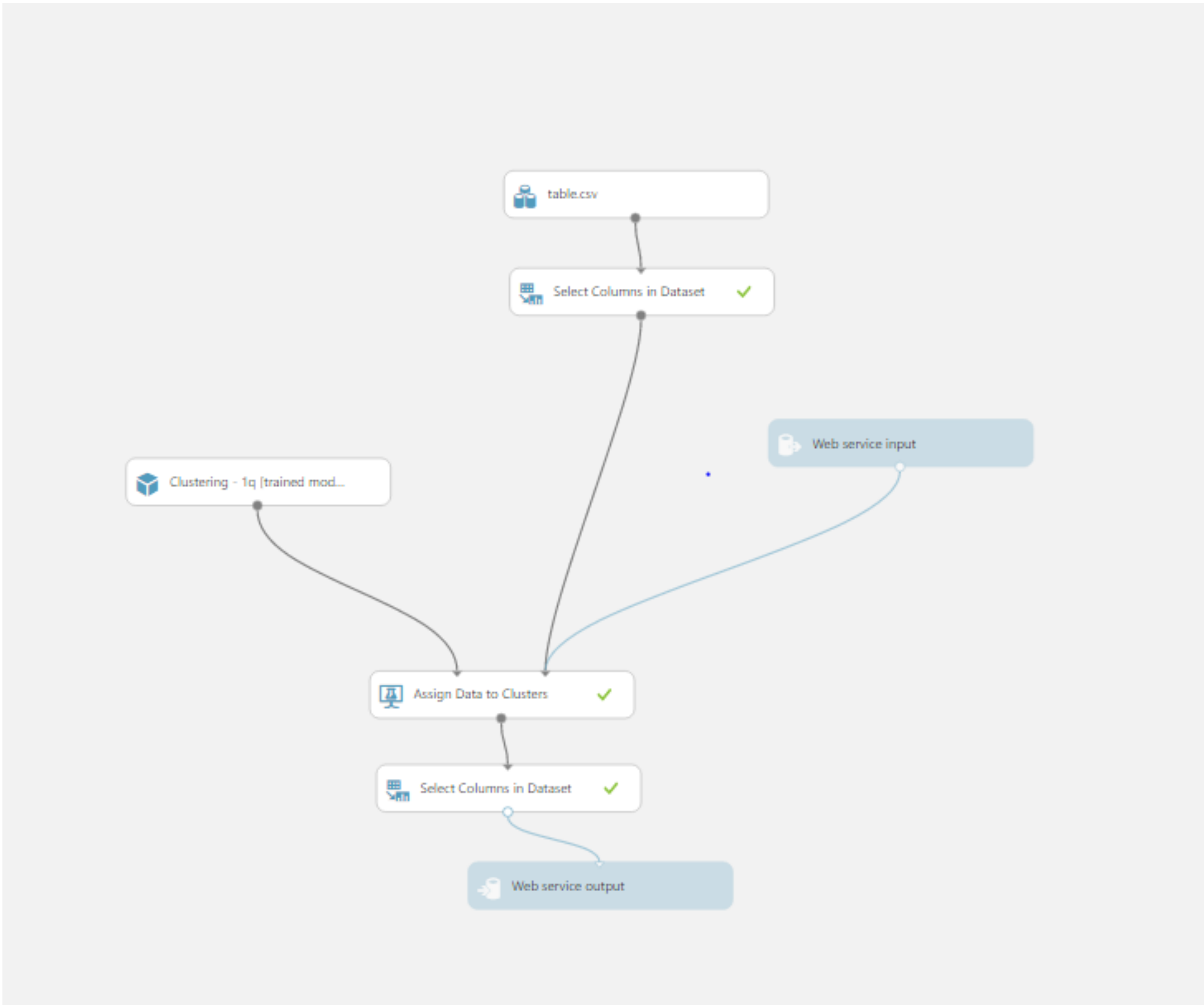
rows 186545

columns 2

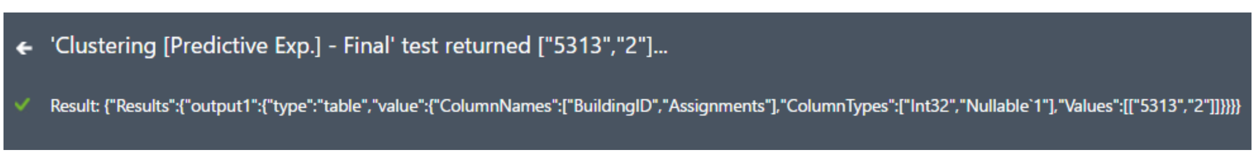
view as

BuildingID	Assignments
5357	2
75709	1
75703	2
5314	2
28096	2
75715	1
75720	1
75719	2
75698	0
5329	2
5332	1
5322	1
5325	1
75698	2
75705	1

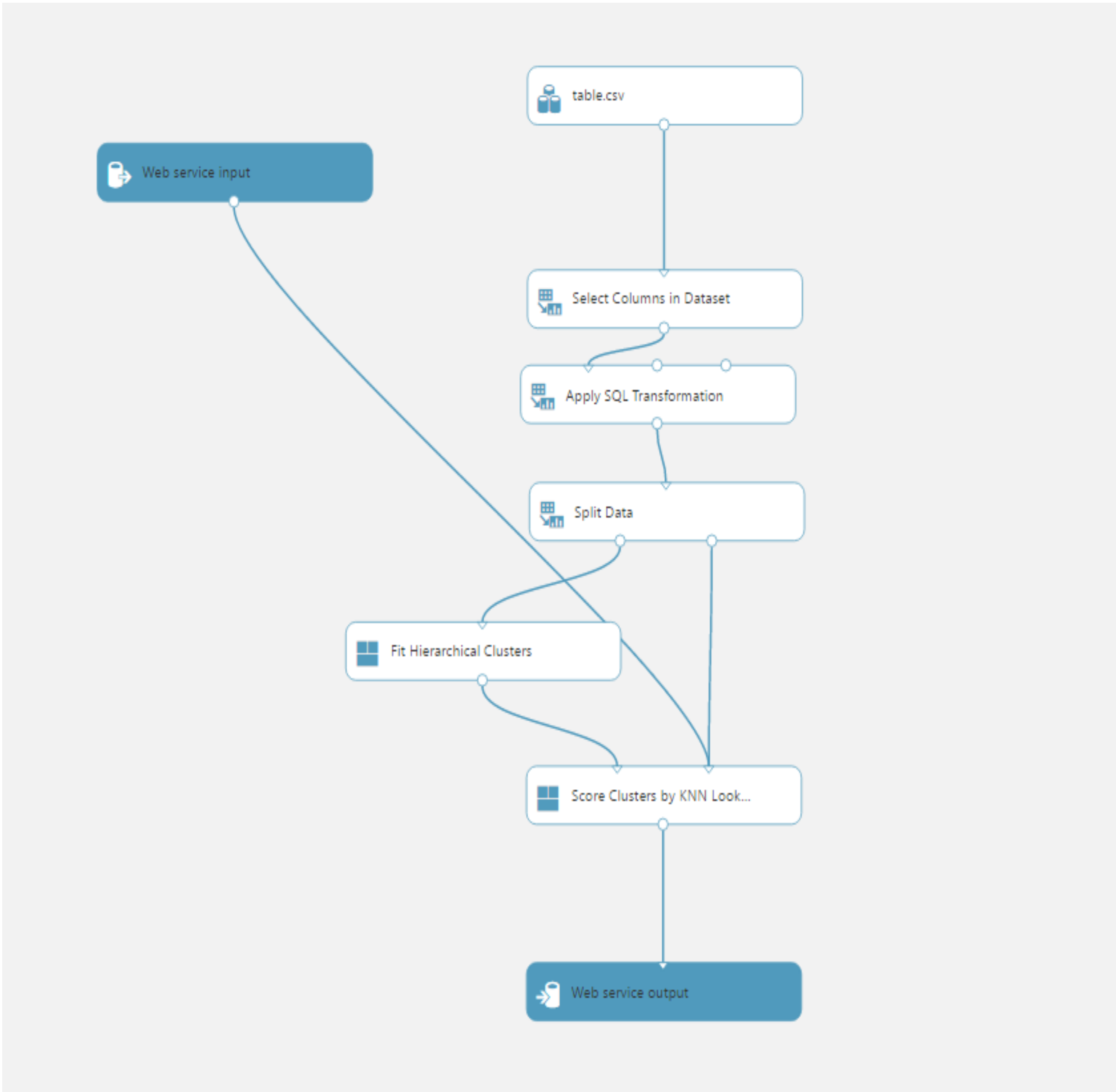
5c. K means Clustering: Predictive Experiment



5d. We deploy the predictive experiment as a web service and enter inputs as part of the building data and the result is as shown below.



6. Hierarchical Clustering:



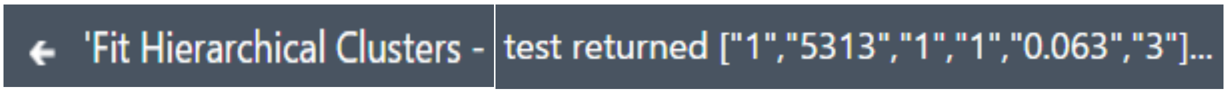
6a. Evaluation Results: The results produced by the Score Clusters by KNN lookup is as follows,

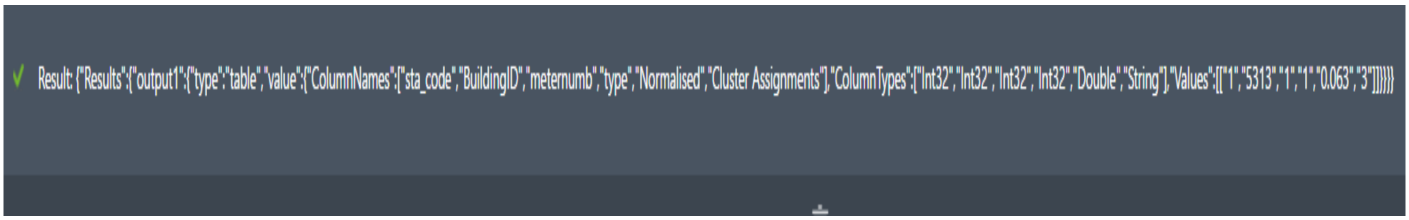
rows 2484 columns 6

view as

sta_code	BuildingID	meternumb	type	Normalised	Cluster Assignments
1	5317	1	2	0.053275	3
1	75690	1	1	1.885206	1
1	5314	1	2	0.619608	3
1	5313	1	2	0.995098	3
1	5317	1	2	0.027413	3
1	5314	1	2	0.719608	3
1	5313	1	2	0.124118	3
1	5304	1	2	0.027093	3
1	5355	1	2	0.713545	3
1	5314	1	2	0.643137	3
1	75701	1	1	0.377546	1
1	5317	1	2	0.013577	3
1	5314	1	2	0.661111	3
1	75705	1	1	0.100385	1

6b. We deploy the predictive experiment as a web service and enter inputs as part of the building data and the result is as shown below.





Web Application:

Applications Used:

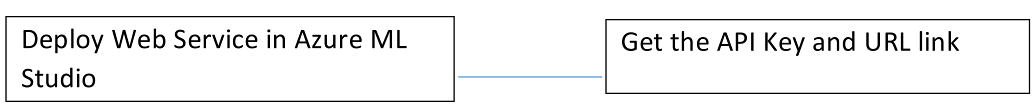
- Sublime Text 3 - Text Editor (Mac)
- Postman – Chrome Extension App (Mac)
- Terminal – Command Prompt (Mac)

Languages:

- HTML
- JavaScript
- jQuery – AJAX

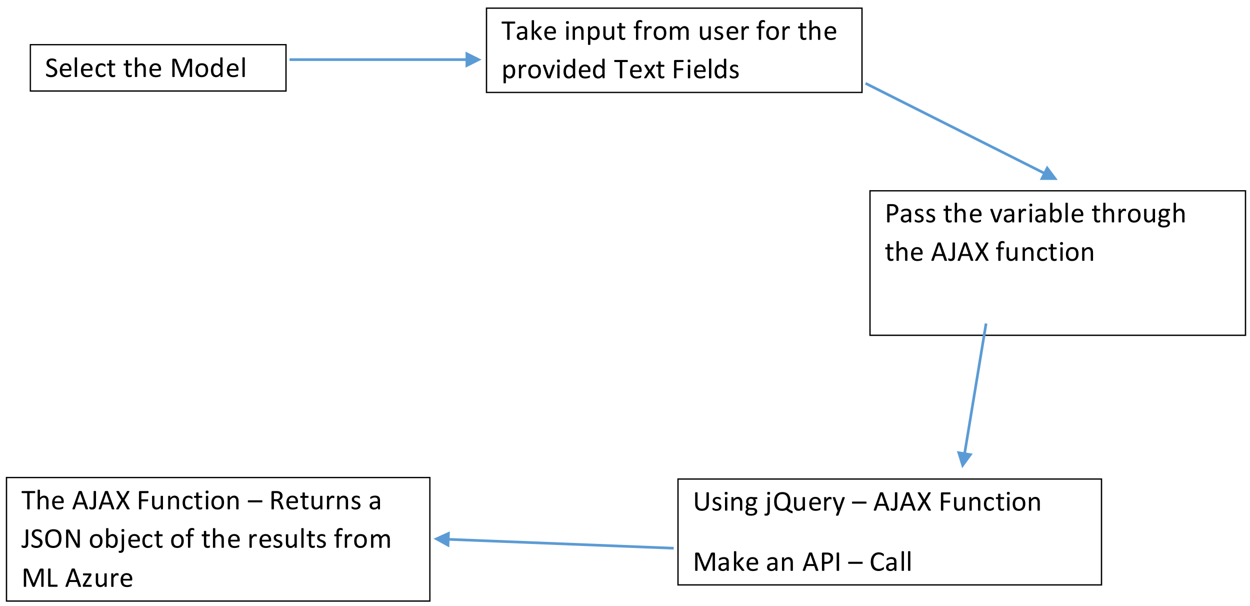
Work Flow:

In ML Azure Studio,



We have created 3 Regression Models which are deployed as a single web service. The API key and URL generated are feed in to the jQuery – Ajax function in HTML Code to get back the JSON object from ML Studio.

Code Modules:



Points to Note:

- There are 4 HTML files (Prediction, Classification, K-means, Hierarchical Clustering).
- The functionality of all the 4 files remain the same.

CODE BREAKDOWN:

```
var textbox = document.getElementsByClassName('a');
var input =[];
for(var i=0;i<textbox.length;i++){
  input.push(textbox[i].value);
}
data.Inputs.input1.Values = [input];
document.getElementById('output').innerHTML=JSON.stringify(data);

var settings = {
  "async": true,
  "crossDomain": true,
  "url": "https://ussouthcentral.services.azureml.net/workspaces/a5d415105df14267a75151b4aca0b5df/services/e6b344524ad5483d86c752ff2d3f715b/
    execute?api-version=2.0&details=true",
  "method": "POST",
  contentType: 'application/json',
  data: JSON.stringify(data),
  "headers": {
    "authorization": "Bearer SiAi0040pgTujtCynAJMdFIsW0cnNyZCSqqsBE51n0Q+KBxeyh3HuWHWl8/T/CSozbMgvxLvX6IHa4elrIztQw==",
    "content-type": "application/json",
    "cache-control": "no-cache",
    "postman-token": "d6e372f6-20fa-ad7f-fd03-a027a3948a4e"
 }
}

  document.getElementById('progressBar').style.display="block";
  $.ajax(settings).done(function (response) {
    document.getElementById('progressBar').style.display="none";
    console.log(response);
    console.log(response.Results.output1.value.Values);
    var lr = response.Results.output1.value.Values[0][1];
    var nn = response.Results.output1.value.Values[1][1];
    var rf = response.Results.output1.value.Values[2][1];

              document.getElementById('output').innerHTML="Linear Regression (KwH/sq.m): "+lr ; // Output Linear Reg
              document.getElementById('outputNeuNet').innerHTML="Neural Network: " + nn;//output For NeuNet
              document.getElementById('outputRanFor').innerHTML="Random Forest: "+ rf;  //output for RanFor
```

In the above code,

- var settings is the variable that is passed to the $.ajax(settings).done(function(response){}

Setting async to false means that the statement you are calling should complete before the next statement in your function can be called. If you set async: true, then that statement will begin its execution and the next statement will be called regardless of whether the async statement has completed yet.

CrossDomain is set to true for server-side redirection to another domain.

URL: From ML Azure studio

Method: Post (Denotes that it's a Post API Request) which sends the content body (data) and content type: application / json.

Response variable gets back the JSON object after hitting the API.

```
<body>
<h1>Assignment 3 - TEAM 8</h1>


<p>Please select the model you want to run</p>

<input type="button" onclick="displayForm()" value="Regression">
<form id=regression style="display:none;">
  Temperature (F):<input class="a" id="input1" name="TemperatureF" onKeyPress ="return isNumberKey(event)"> <br>
  Dew Point (F):<input class="a" id="input2" name="DewPointF" onKeyPress ="return isNumberKey(event)"><br>
  Visibility (MPH):<input class="a" id="input3" name="VisibilityMPH" onKeyPress ="return isNumberKey(event)"><br>
  Wind Speed (MPH):<input class="a" id="input4" name="WindSpeedMPH" onKeyPress ="return isNumberKey(event)"><br>
  Events:<input class="a" id="input5" name="Events" onKeyUp="handleEvents(this)"><br>
  Sta code:<input class="a" id="input6" name="sta_code" onKeyUp="handleSta(this)"><br>
  Date:<input class="a" id="input7" name="Date"><br>
  Meternumb-1:<input class="a" type="radio" id="input8" name="Meternumb" value="0" onclick="setRadioValue()"/><br>
  Meternumb-2:<input class="a" type="radio" id="input9" name="Meternumb" value="0" onclick="setRadioValue()"/><br>
  Meternumb-3:<input class="a" type="radio" id="input10" name="Meternumb" value="0" onclick="setRadioValue()"/><br>
  Meternumb-4:<input class="a" type="radio" id="input11" name="Meternumb" value="0" onclick="setRadioValue()"/><br>
  Meternumb-8:<input class="a" type="radio" id="input12" name="Meternumb" value="0" onclick="setRadioValue()"/><br>
  Time:<input class="a" id="input13" name="Time" onKeyUp="handleTime(this)" /><br>
  <button type="button" onclick="predictionRegression()">Predict!</button>
</form>
<h3 id="progressBar" style="display:none">Loading...</h3>
<br>
<div id="input"></div><br>
<div id="output"></div><br>
<div id="outputNeuNet"></div><br>
<div id="outputRanFor"></div><br>
</body>
```

The HTML body consists of all the Input we require from the user that should be passed on to the ML azure model to get back the result for Regression.

Error Handling is done on the text field to prevent the user from typing junk data and to instruct the user to input valid data to feed it into the model.

```
function isNumberKey(evt) {
    var charCode = (evt.which) ? evt.which : event.keyCode;
    if (charCode == 46 && evt.srcElement.value.split('.').length>1) {
        return false;
    }
    if (charCode != 46 && charCode > 31 && (charCode < 48 || charCode > 57))
        return false;
    return true;
}

function handleSta(input) {
    if (/\D/g.test(input.value)) input.value = input.value.replace(/\D/g,'')
    if (input.value < 1 || input.value > 2){
      alert("Input should be 1 or 2");
      input.value = "";
    }
  }

function handleEvents(input) {
  if (/\D/g.test(input.value)) input.value = input.value.replace(/\D/g,'')
    if (input.value < 1 || input.value > 8){
      alert("Input should be between 1-8");
      input.value = "";
    }
  }

function handleTime(input) {
  if (/\D/g.test(input.value)) input.value = input.value.replace(/\D/g,'')
    if (input.value < 0 || input.value > 23){
      alert("Input should be between 0-23");
      input.value = "";
    }
  }
```

For Date Field Validation (Instructing user to use the right format and the Year being 2013) ---- OPTIONAL

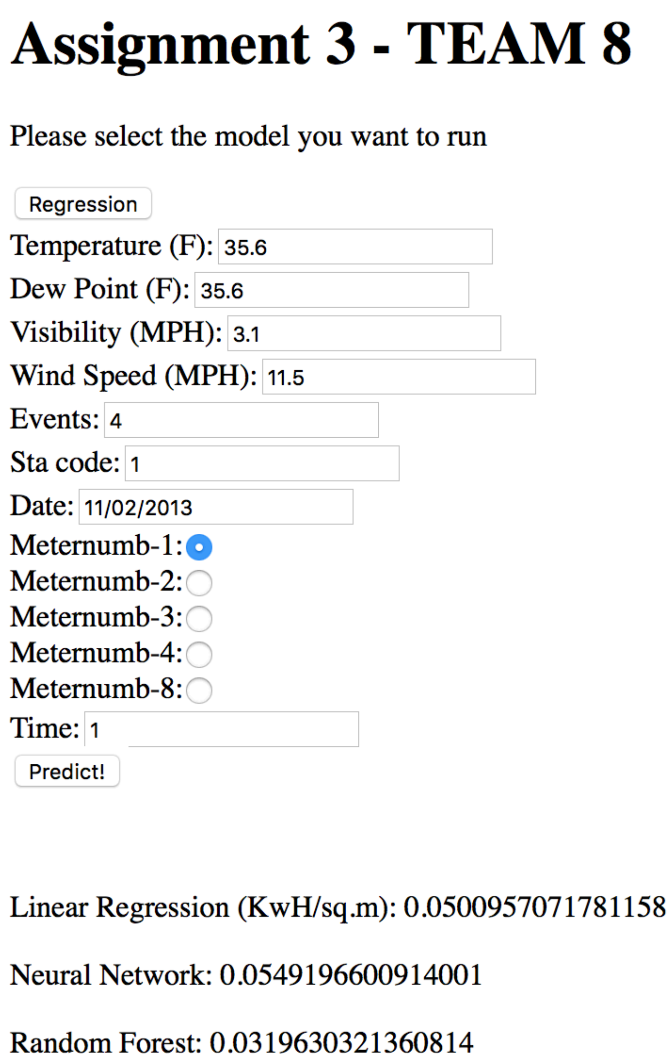
```javascript
// DATE VALIDATION in Date Text Field
var date_regex = /^(0[1-9]|1[0-2])\/(0[1-9]|1\d|2\d|3[01])\/(19|20)\d{2}$/ ;
var datestring = document.getElementsByName('Date')[0]
if(!(date_regex.test(datestring.value)))
{
  console.log(datestring.value)
  alert("Date should be of format mm/dd/yyyy");
  datestring.value = "";
  return false;
}
var parts = datestring.value.split("/");
var day = parseInt(parts[1], 10);
var month = parseInt(parts[0], 10);
var year = parseInt(parts[2], 10);
console.log(year);
if(year != "2013"){
  alert("The year should be 2013");
  datestring.value = "";
  return false;
}
```

```html
<html>
<script
 src="https://code.jquery.com/jquery-3.1.1.js"
 integrity="sha256-16cdPddA6VdVInumRGo6IbivbERE8p7CQR3HzTBuELA="
 crossorigin="anonymous"></script>
 <script src="http://cdn.jsdelivr.net/jquery.validation/1.15.0/jquery.validate.min.js"></script>
 <script src="http://cdn.jsdelivr.net/jquery.validation/1.15.0/additional-methods.min.js"></script>



<style>
.hidden1{
  display: none;
}
.input:invalid {
  border: 1px solid red;
}

.input:valid {
  border: 1px solid green;
}
</style>


<script type="text/javascript" >

function predictionRegression()
{
  var data = {
    "Inputs": {
      "input1":{
        "ColumnNames": ["TemperatureF", "Dew.PointF", "VisibilityMPH", "Wind.SpeedMPH", "Events", "sta_code", "Date", "meternumb-1", "meternumb-2", "
          meternumb-3", "meternumb-4", "meternumb-8", "Time"],
        "Values": [[]]
      }
    },
    "GlobalParameters": {}
  }
```

PredictionRegression() – Function that contains the data that gets passed to the ML Azure Models.

OUTPUT SNIPPETS:

PREDICTION:



The code takes in the input and returns the three Regression models containing the dependent variable (Normalized Consumption/sq.m) for the input feed.

CLASSIFICATION:

Assignment 3 - TEAM 8

Please select the model you want to run

Classification

Temperature (F): 11.5
Dew Point (F): 11.5
Visibility (MPH): 2.2
Wind Speed (MPH): 12.5
Events: 1
Sta code: 1
Month: 1
Type: 1
Weekday: 1
Peakhour: 0

Classification!

Loading...

{"Inputs":{"input1":{"ColumnNames":
["TemperatureF","Dew.PointF","VisibilityMPH","Wind.SpeedMPH","Events","sta_code","month","type","Weekday","Peakhour"],"Values":
[["11.5","11.5","2.2","12.5","1","1","1","1","1","0"]]}},"GlobalParameters":{}}

The Loading … indicates that the HTML code is passing the parameters to the web URL to get the results back.

Clustering output:

Assignment 3 - TEAM 8

Please select the model you want to run

Classification

Temperature (F): 11.5

Dew Point (F): 11.5

Visibility (MPH): 2.2

Wind Speed (MPH): 12.5

Events: 1

Sta code: 1

Month: 1

Type: 1

Weekday: 1

Peakhour: 0

Classification!

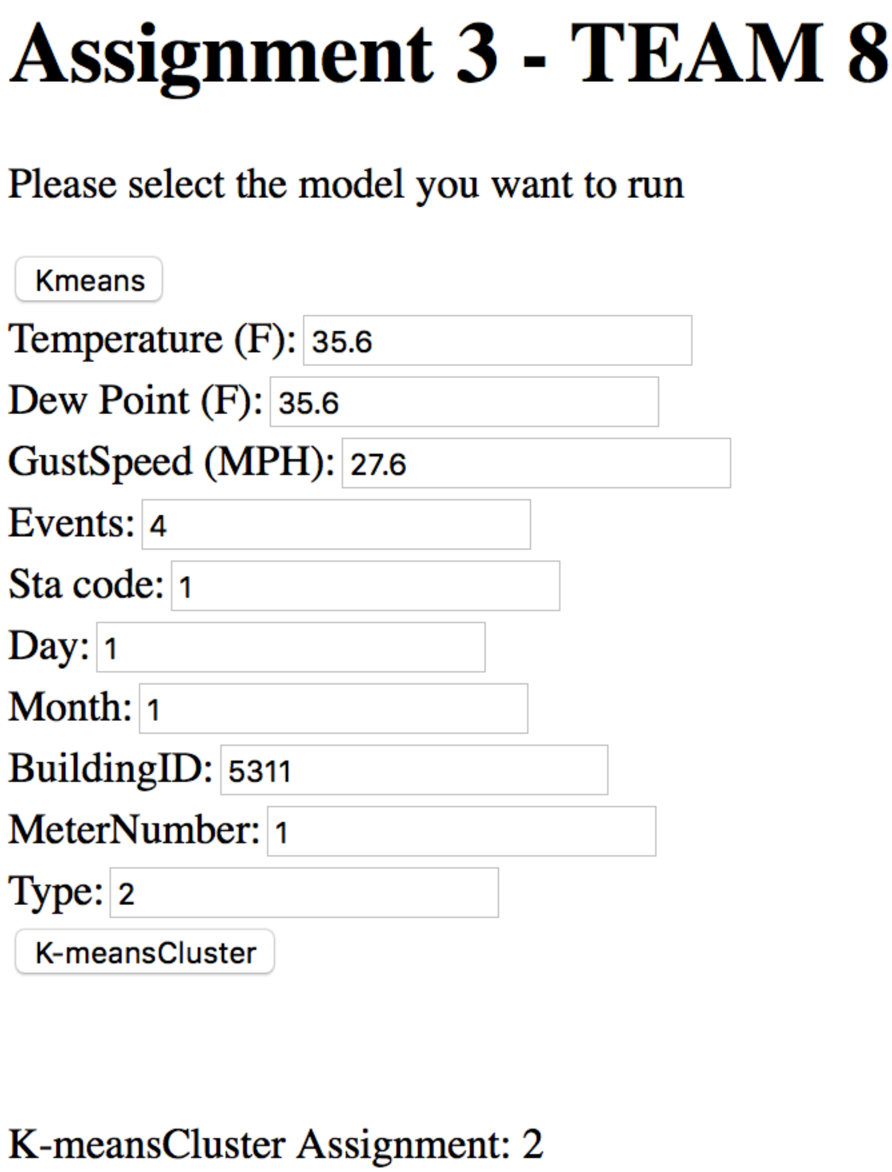
Linear Regression: 0

Neural Network: 1

Random Forest: 0

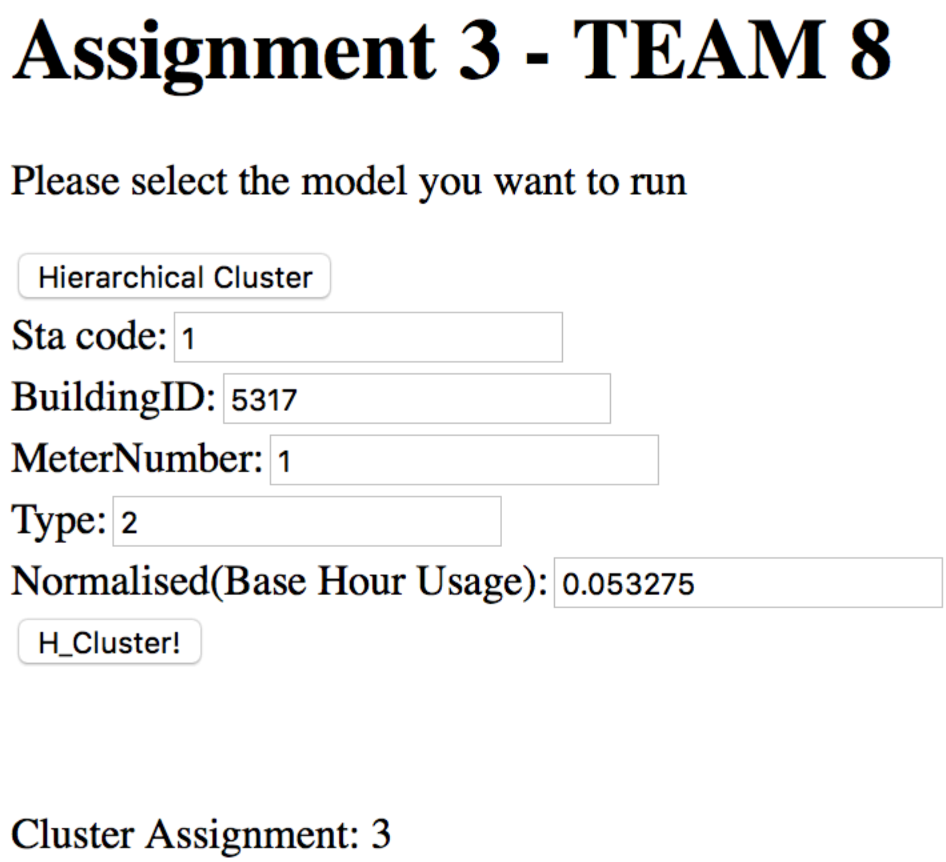
As accuracy of none of the model is 100%, there will be few times when the output of all three will not be same. So, depending on AUC (area under the curve),
Neural Network will be accurate most of the times.

K-Means Clustering OUTPUT :



Depending on the input, it will assign the Buildings to the respective clusters based on the building data entered.

Hierarchical Cluster Output:



Team Contribution:

Ankita – Regression Models & Web service Deployment

Sriniketan – Classification & Clustering Models

Vignesh – Web application development