INFO7390_ Assignment3

Group 8:

Zhao Wang;

Lezi Wang;

Qiaoyi He

1. Problem Statement & Requirements

- Nowadays, energy consumption has become a critical requirement in multiple industry fields. Therefore, we are planning to explore the datasets, which contained the power consumption of 9 different industries, and 58 different locations in City of Boston, and then develop short-term prediction models (kWh, Power Factor, and KVarh) for each location, in order to meet following requirements:
- 1.Supply and demand balancing: forecast the future energy usage and peak leveling to avoid overload.
- 2.Market making: help government to set power price.
- 3.Electricity grid operational optimization: government can response according to the demand for avoids energy waste.
- 4.Fault and anomaly record detection.

2. Business Goal & Solutions

– Business Goal

Predict short-term power consumption for each location in City of Boston, in order to

1) Balance supply and demand;

2) Predict the daily peak leveling for avoid some emergency event like power failure, and lowest leveling for avoid the power waste;

3) Help different industries to plan for future power consumption, reduce power cost and save energy.

– Solutions

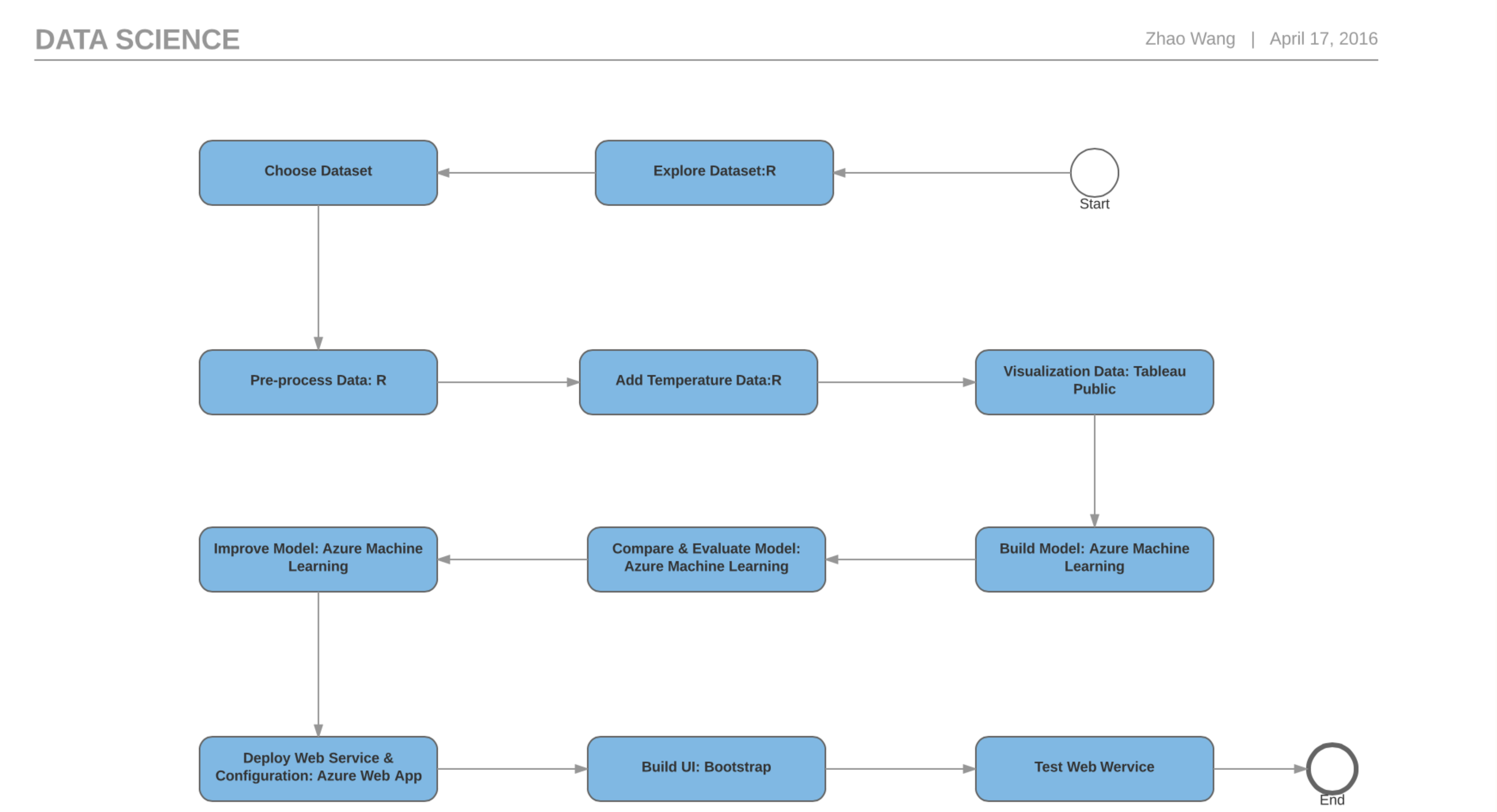
Implement Micro Analysis to predict following power consumption for one day in future.

1) Total kWh for one day.

2) Average Power Factor for one day.

3) Total KVarh for one day.

3. Process Steps & Technologies



4. Data process
4.1 Calculate the value

– After thorough explore the data, we think temperature has a great impact on energy consumption, so we add three different parameters relatively the temperature into dataset – maximum temperature, minimum temperature and average temperature. we need the calculate the sum of kWh and kVARh and the mean of power factor.

```
#select the power factor value
subset2<-subset(data1,Units=="Power Factor")
#calculate the mean of power factor
temp<-subset2[,5:292]
power_factor<-apply(temp,1,mean)
#add mean of power factor as a now column
energydata<-data.frame(energydata,power_factor)
```

```
#select the kWh value
subset1<-subset(data1,Units=="kWh")
#calculate the sum of kWh
temp<-subset1[,5:292]
kWh<-apply(temp,1,sum)
#add total kWh as a now column
energydata<-data.frame(subset1,kWh)
```

```
#select the KVARH value
subset3<-subset(data1,Units=="kVARh")
#calculate the sum of kwh
temp<-subset3[,5:292]
kVARh<-apply(temp,1,sum)
#add sum of kVARh as a now column
energydata<-data.frame(energydata,kVARh)
```

4.2 Add temperature parameter

- After thorough explore the data, we think temperature has a great impact on energy consumption, so we add three different parameters relatively the temperature into dataset – maximum temperature, minimum temperature and average temperature.
- We got the Daily Temperature Record 2014 from following URL and calculate the average temperature based it:

http://www.usclimatedata.com/climate/boston/massachusetts/united-states/usma0046/2014/1

```
#import the temperature date
datatem<-read.csv("~/Desktop/temperature.csv")
#Add temperature data into dataset
tem<-datatem[,2:4]
energydata<-data.frame(energydata,tem)
energydata<-energydata[,-c(4:292)]
```

4.3 Clean the data

- After we restructure the data set, we need to clean abnormal or missing data for improving the usability and stability of data. The following picture is the R code which we use to process clean the data. We delete rows which contain “0” value or missing value.
-

```
#clear 0 and NA row
energydata[energydata == 0] = NA
energydata<-na.omit(energydata)
```

4.3 Clean the data

– Dataset format after cleaning

	A	B	C	D	E	F	G	H	I	J	K
1	Account	Date	Channel	Industry	Location	kWh	high	low	average	power_facto	kVARh
2	########	01/01/2014	605106493 :	BPL	Harvard.st	497.052	29.1	19.2	24.15	0.95061441	168.381
3	2.6812E+10	01/02/2014	605106493 :	BPL	Harvard.st	570.789	26.2	2.1	14.15	0.96164955	163.458
4	2.6812E+10	01/03/2014	605106493 :	BPL	Harvard.st	465.48	14.2	2.1	8.15	0.95305582	154.134
5	2.6812E+10	01/04/2014	605106493 :	BPL	Harvard.st	571.365	27.1	2.1	14.6	0.96432472	155.538
6	2.6812E+10	01/05/2014	605106493 :	BPL	Harvard.st	346.239	37.9	24.3	31.1	0.96655633	91.836
7	2.6812E+10	01/06/2014	605106493 :	BPL	Harvard.st	575.172	55.9	29.1	42.5	0.97101479	136.359
8	2.6812E+10	01/07/2014	605106493 :	BPL	Harvard.st	549.918	29.1	9.1	19.1	0.96259224	156.348
9	2.6812E+10	01/08/2014	605106493 :	BPL	Harvard.st	632.259	22.1	7.2	14.65	0.96290678	176.049
10	2.6812E+10	01/09/2014	605106493 :	BPL	Harvard.st	666.963	30.2	19.2	24.7	0.96754126	170.145
11	2.6812E+10	01/10/2014	605106493 :	BPL	Harvard.st	584.991	37	18.1	27.55	0.96294836	163.116
12	2.6812E+10	01/11/2014	605106493 :	BPL	Harvard.st	486.441	59	34	46.5	0.97746106	101.025
13	2.6812E+10	01/12/2014	605106493 :	BPL	Harvard.st	407.187	54	37	45.5	0.96874735	103.356
14	2.6812E+10	01/13/2014	605106493 :	BPL	Harvard.st	673.947	51.1	31.1	41.1	0.96602897	177.282
15	2.6812E+10	01/14/2014	605106493 :	BPL	Harvard.st	595.521	50	44.1	47.05	0.96454975	160.317
16	2.6812E+10	01/15/2014	605106493 :	BPL	Harvard.st	628.929	48	35.1	41.55	0.96249629	173.925
17	2.6812E+10	01/16/2014	605106493 :	BPL	Harvard.st	595.656	39	33.1	36.05	0.96492173	159.759
18	2.6812E+10	01/17/2014	605106493 :	BPL	Harvard.st	583.767	46	33.1	39.55	0.96534736	156.771
19	2.6812E+10	01/18/2014	605106493 :	BPL	Harvard.st	558.342	37.9	32	34.95	0.96432248	153.189
20	2.6812E+10	01/19/2014	605106493 :	BPL	Harvard.st	338.724	35.1	30.2	32.65	0.96552862	91.287
21	2.6812E+10	01/20/2014	605106493 :	BPL	Harvard.st	508.23	41	26.2	33.6	0.95015148	172.431
22	2.6812E+10	01/21/2014	605106493 :	BPL	Harvard.st	643.932	26.2	11.1	18.65	0.96242735	179.541
23	2.6812E+10	01/22/2014	605106493 :	BPL	Harvard.st	721.566	18.1	7.2	12.65	0.95321607	219.771
24	2.6812E+10	01/23/2014	605106493 :	BPL	Harvard.st	734.337	22.1	6.3	14.2	0.95355806	220.338
25	2.6812E+10	01/24/2014	605106493 :	BPL	Harvard.st	723.951	19.2	7.2	13.2	0.95272836	220.671
26	2.6812E+10	01/25/2014	605106493 :	BPL	Harvard.st	715.806	37.9	15.3	26.6	0.95323208	215.955
27	2.6812E+10	01/26/2014	605106493 :	BPL	Harvard.st	605.61	26.2	16.2	21.2	0.94513609	209.196
28	2.6812E+10	01/27/2014	605106493 :	BPL	Harvard.st	728.334	46.9	21.2	34.05	0.95508175	216.9
29	2.6812E+10	01/28/2014	605106493 :	BPL	Harvard.st	723.429	21.2	13.1	17.15	0.95236007	221.463
30	2.6812E+10	01/29/2014	605106493 :	BPL	Harvard.st	727.596	27.1	15.3	21.2	0.95475169	215.685

COB-BPL. HARVARD. ST. 2014

4.4 Upload Dataset

- After pre-process, we upload all data set through COE serve. So customers can easy change the data source by change the URL of reader. You can find all representative datasets from following URL:

http://www1.ece.neu.edu/~zwang3/datatest/

Index of /~zwang3/datatest

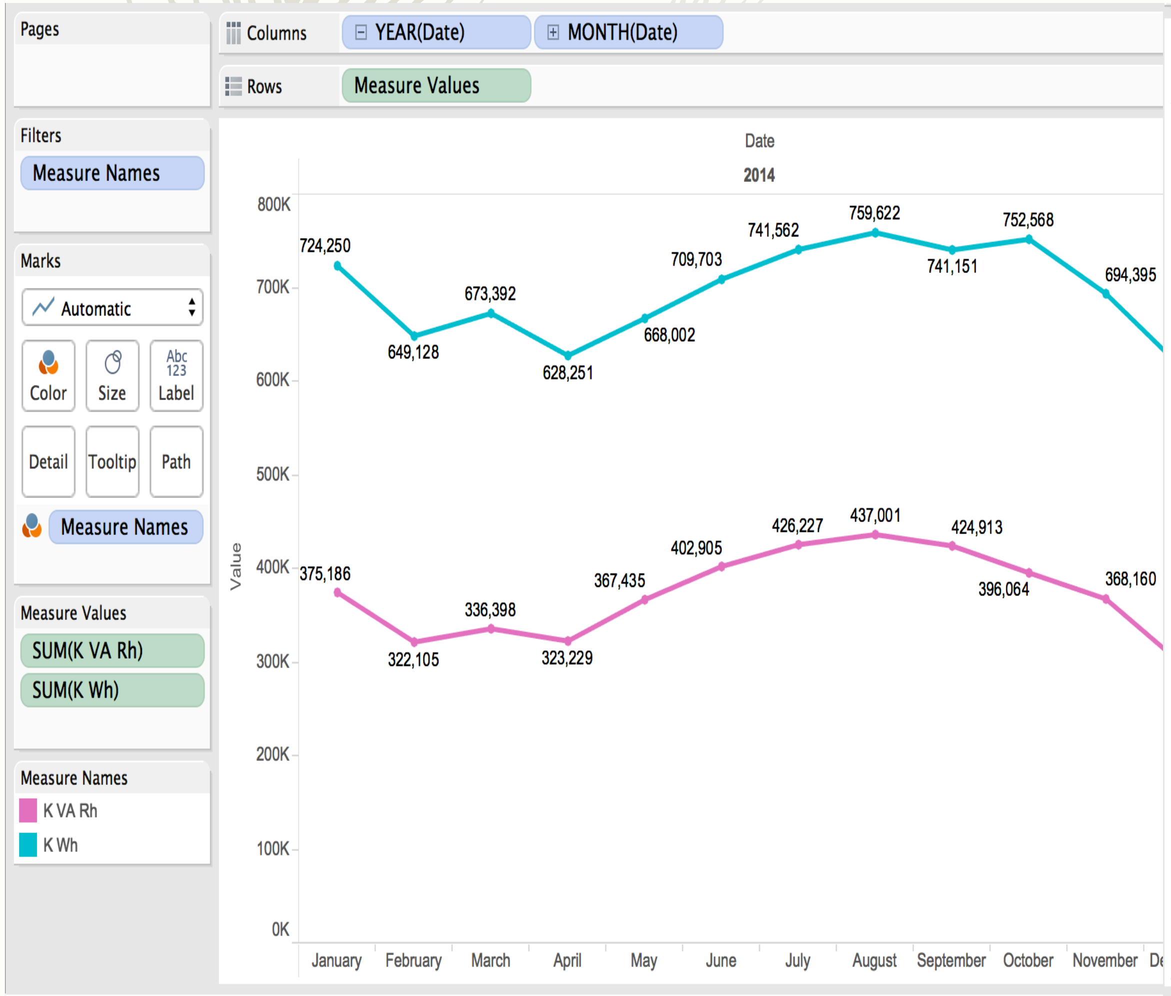
Name	Last modified	Size	Description
Parent Directory		-	
COB-BCYF.CURLEY.CMTY.CTR.2014.csv	2016-04-15 14:44	38K	
COB-BFD.HEADQUARTERS.2014.csv	2016-04-15 14:44	35K	
COB-BPD.DUDLEY.SQ.2014.csv	2016-04-15 14:44	33K	
COB-BPD.HEADQUARTERS.D.2014.csv	2016-04-15 14:44	35K	
COB-BPD.STATION.11.AREA.C.2014.csv	2016-04-15 14:44	37K	
COB-BPL.COPLEY.SQUARE.2014.csv	2016-04-15 14:44	38K	
COB-BPL.HARVARD.ST.2014.csv	2016-04-15 14:44	32K	
COB-BPL.HYDE.PARK2014.csv	2016-04-15 14:44	34K	
COB-BPL.MATTAPAN.2014.csv	2016-04-15 14:44	33K	
COB-DND.HEMENWAY.2014.csv	2016-04-15 14:44	33K	
COB-PROP.MGMT.43.HAWKINS.2014.csv	2016-04-15 14:44	34K	
COB-PROP.MGMT.1481TREMONT.2014.csv	2016-04-15 14:44	34K	
COB-PROP.MGMT.CITY.HLL.SQ.2014.csv	2016-04-15 14:44	37K	

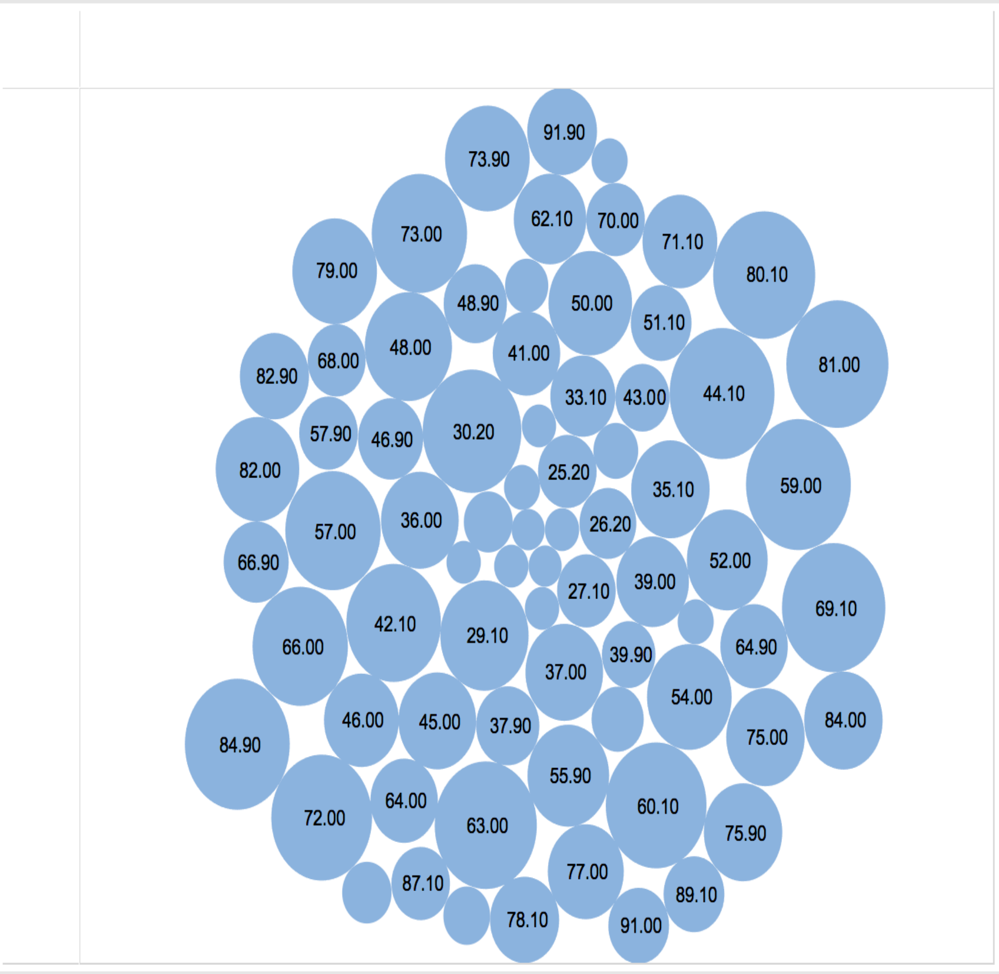
5. Visualization Data

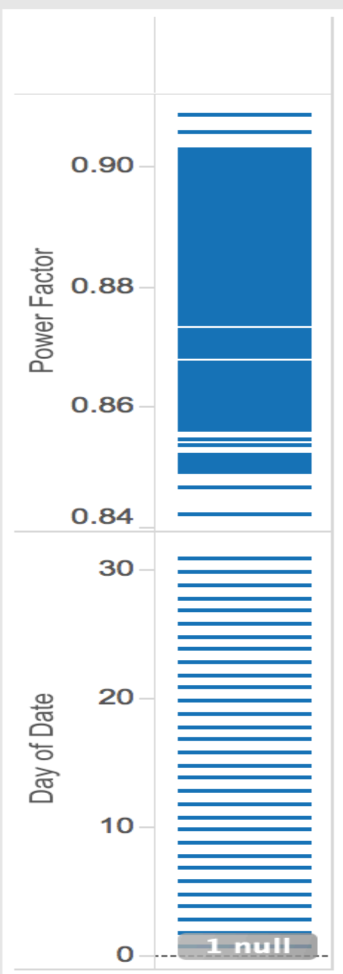
– For Visualization, we use Tableau Public tools.

– We choose 22 dataset in total, and we do visualization for each dataset separately. Here we just show two typical analysis for **BPL at Copley Square** and **SCH Arts**, and **all others can be viewed on our website**, under visualization tab.

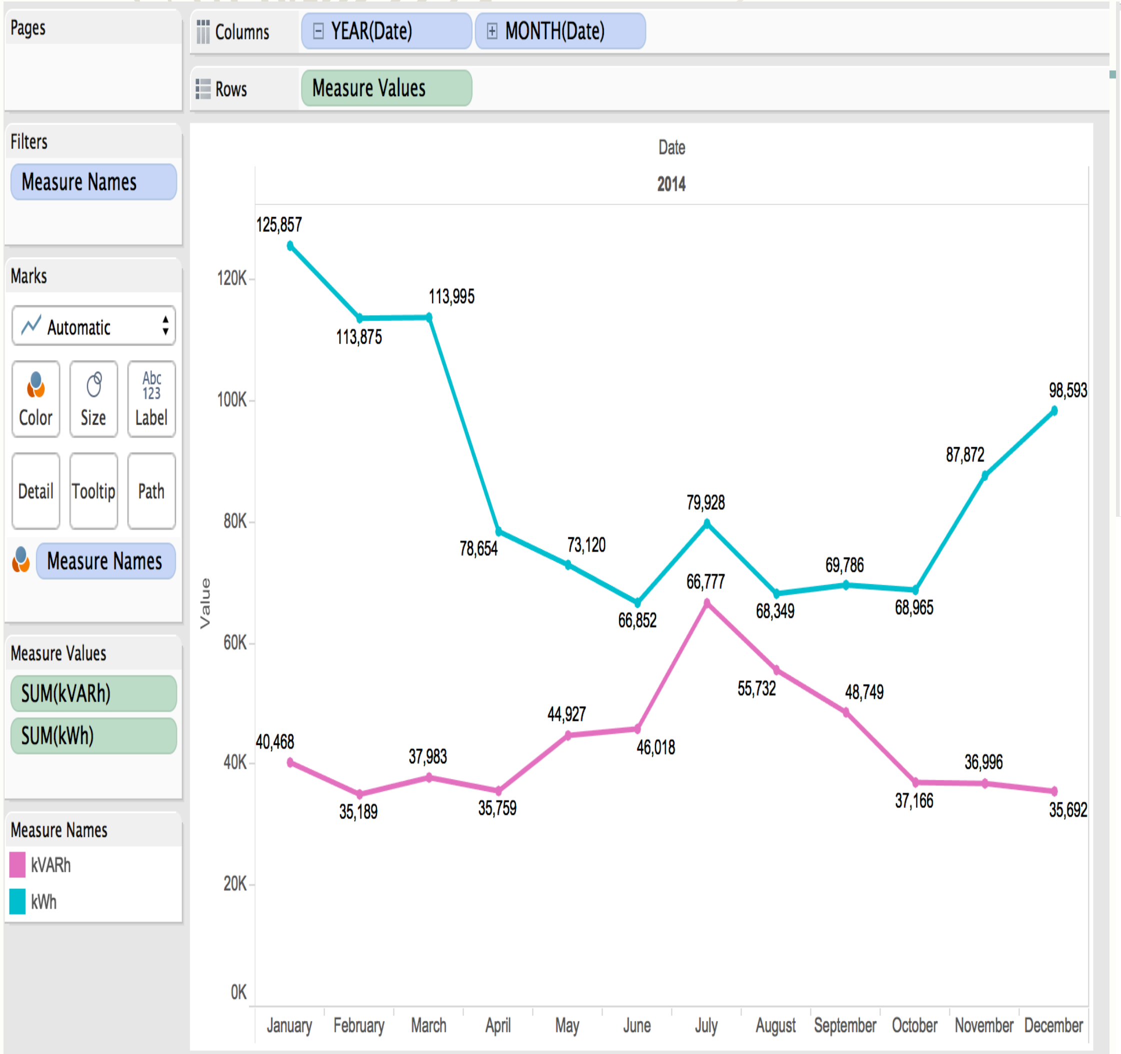
5.1 Visualization for BPL at Copley Square Energy Consumption

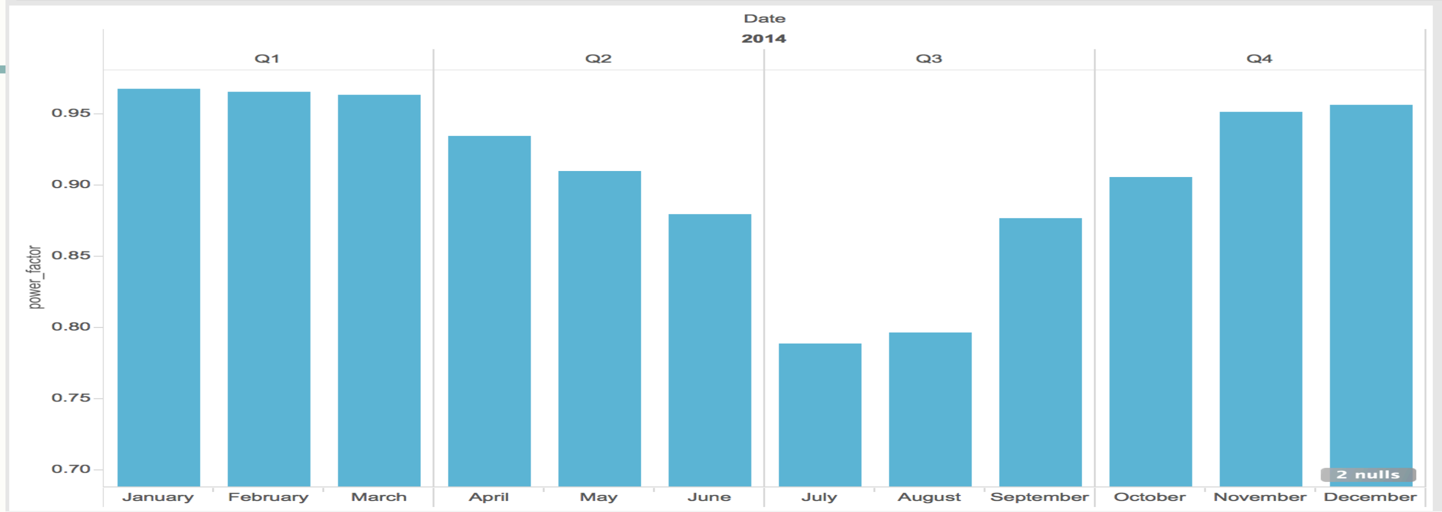


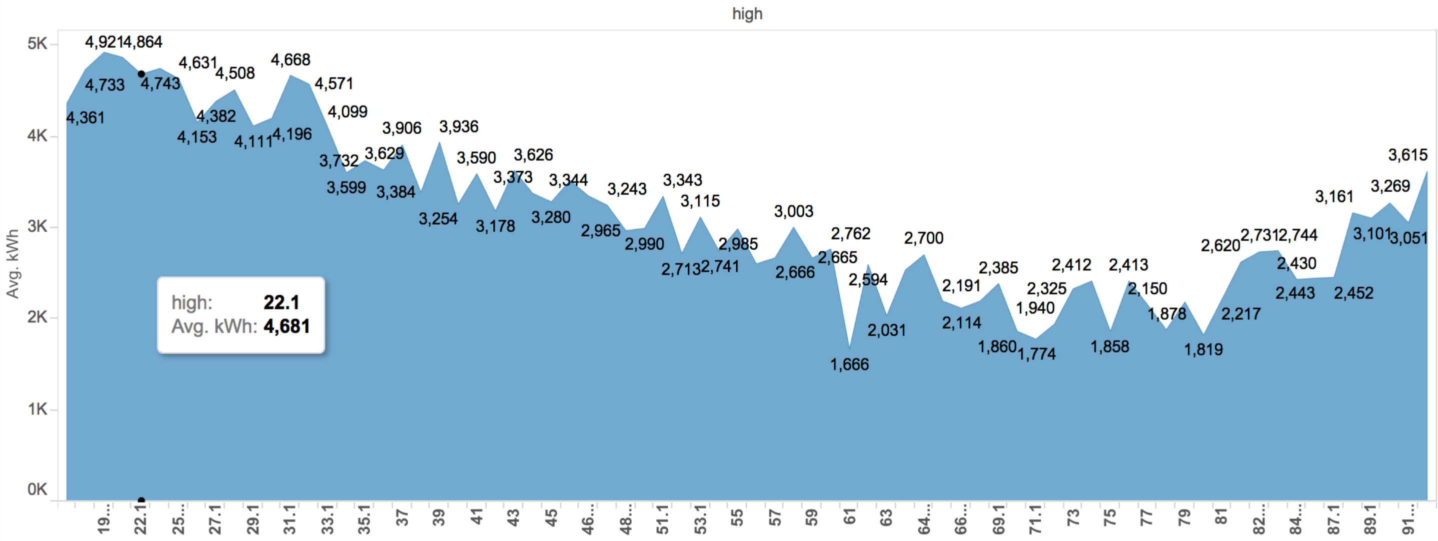




5.2 Visualization for SCH Arts Energy Consumption



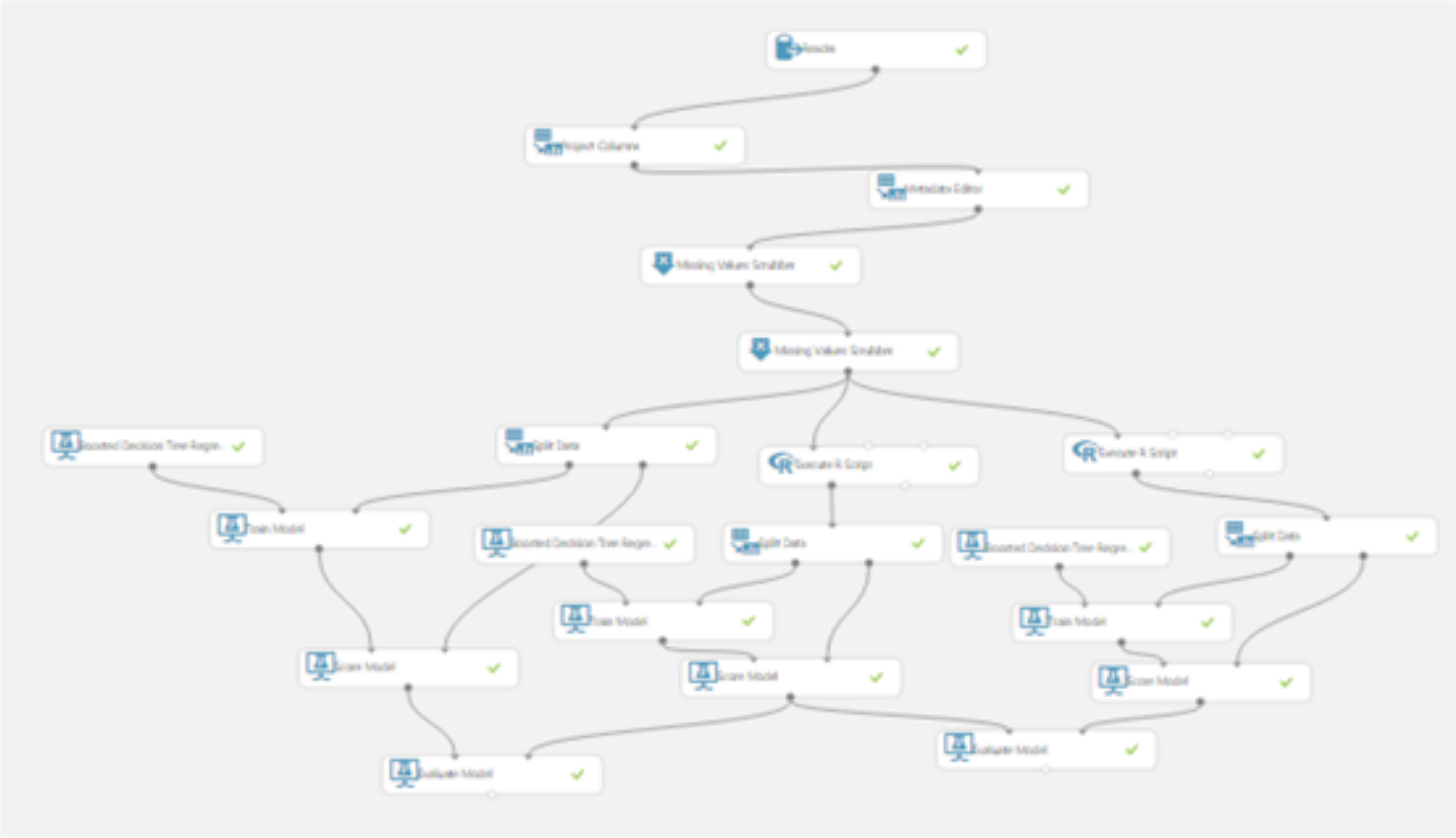




6. Azure Machine Learning

- We created three machine learning models in Azure Machine Learning Workspace to predict value. The first is the total kWh value for one day, the second is the average power factor for one day, and the third is the total Kvarh for one day.
- We created three models for each result in Azure Machine Learning. In first experiment, we created three models to select the best variables subset. In the second experiment, we created two models to choose a better performance algorithm. In the third experiment, we use the best variables subset and a better performance algorithm.

6.1 Prediction Model for kWh



6.2 Reader Console in Azure

For user Convenience, in Azure Workspace, we use Reader console to get the input dataset instead of using saved dataset. The reason is, if we used the dataset saved in Azure, the dataset is fixed, the model is only for this dataset, and user only can use this model for only one time. However, if we use Reader-Web URL via HTTP, we can input a dataset URL. Also, we can set URL as the Web Service Parameter. In this way, user can input the URL of one dataset they want to use, and use our models to predict values based on their preferred dataset. Our models are more flexible and more useful.

6.3 Compared Variables Subsets

- In this experiment, we created three models which use same machine learning algorithm-Boosted Decision Tree Regression as Training Algorithm. But we use three different variables subsets for each compared models.
- The three training models that we constructed were all based on the same input data, but we added different additional features to each training set. Because this model is for predicting kWh value, so the output of this model is kWh.

Set A = Date + Highest Temperature + Lowest Temperature+ Average Temperature

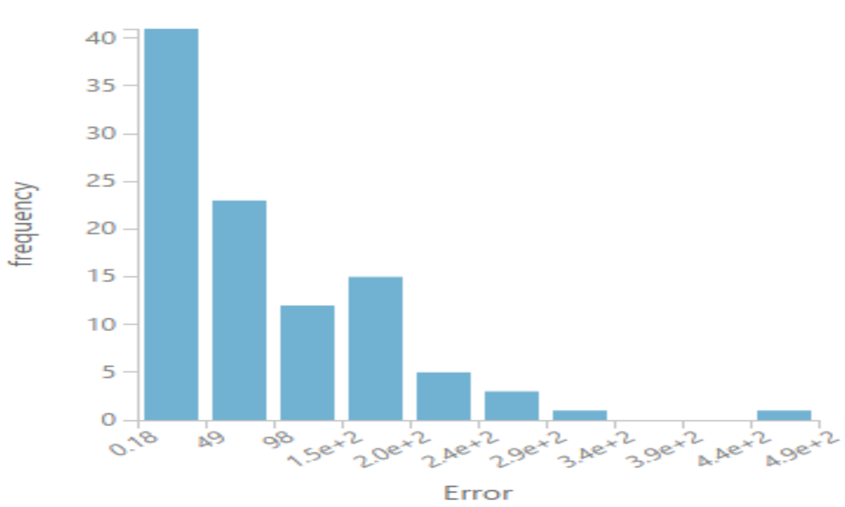
Set B = Date + Average Temperature

Set C = Date + Highest Temperature +Lowest Temperature

6.4 Evaluate Models

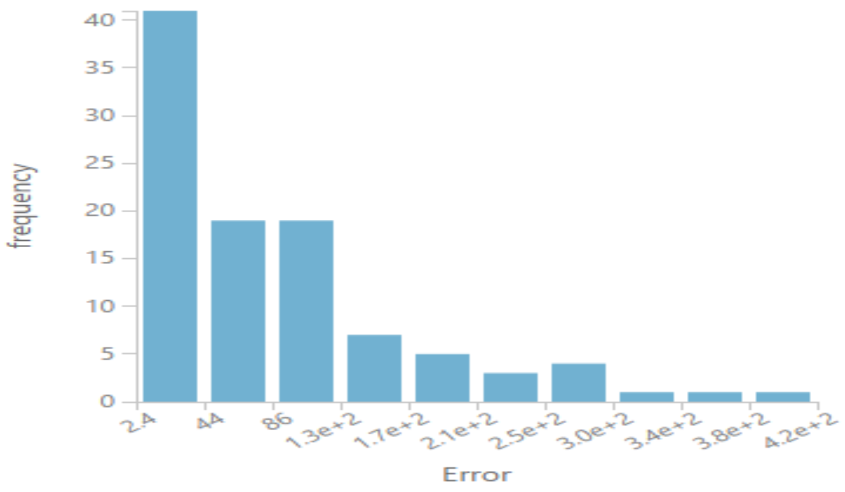
Metrics

Mean Absolute Error	91.863546
Root Mean Squared Error	124.553907
Relative Absolute Error	0.905576
Relative Squared Error	0.70127
Coefficient of Determination	0.29873



Metrics

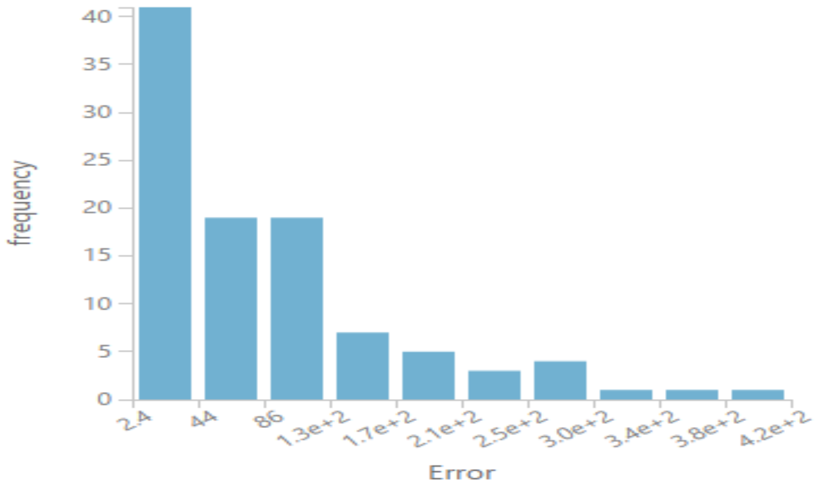
Mean Absolute Error	88.376551
Root Mean Squared Error	122.185869
Relative Absolute Error	0.871202
Relative Squared Error	0.674859
Coefficient of Determination	0.325141



6.4 Evaluate Models (CON'T)

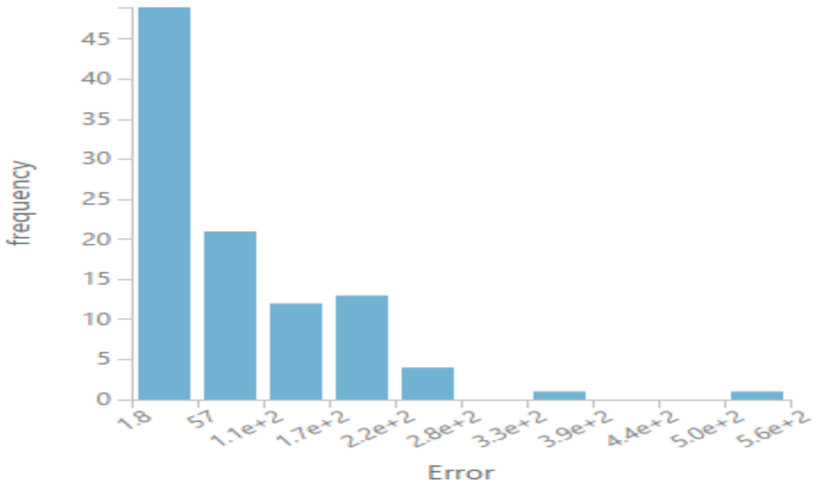
Metrics

Mean Absolute Error	88.376551
Root Mean Squared Error	122.185869
Relative Absolute Error	0.871202
Relative Squared Error	0.674859
Coefficient of Determination	0.325141

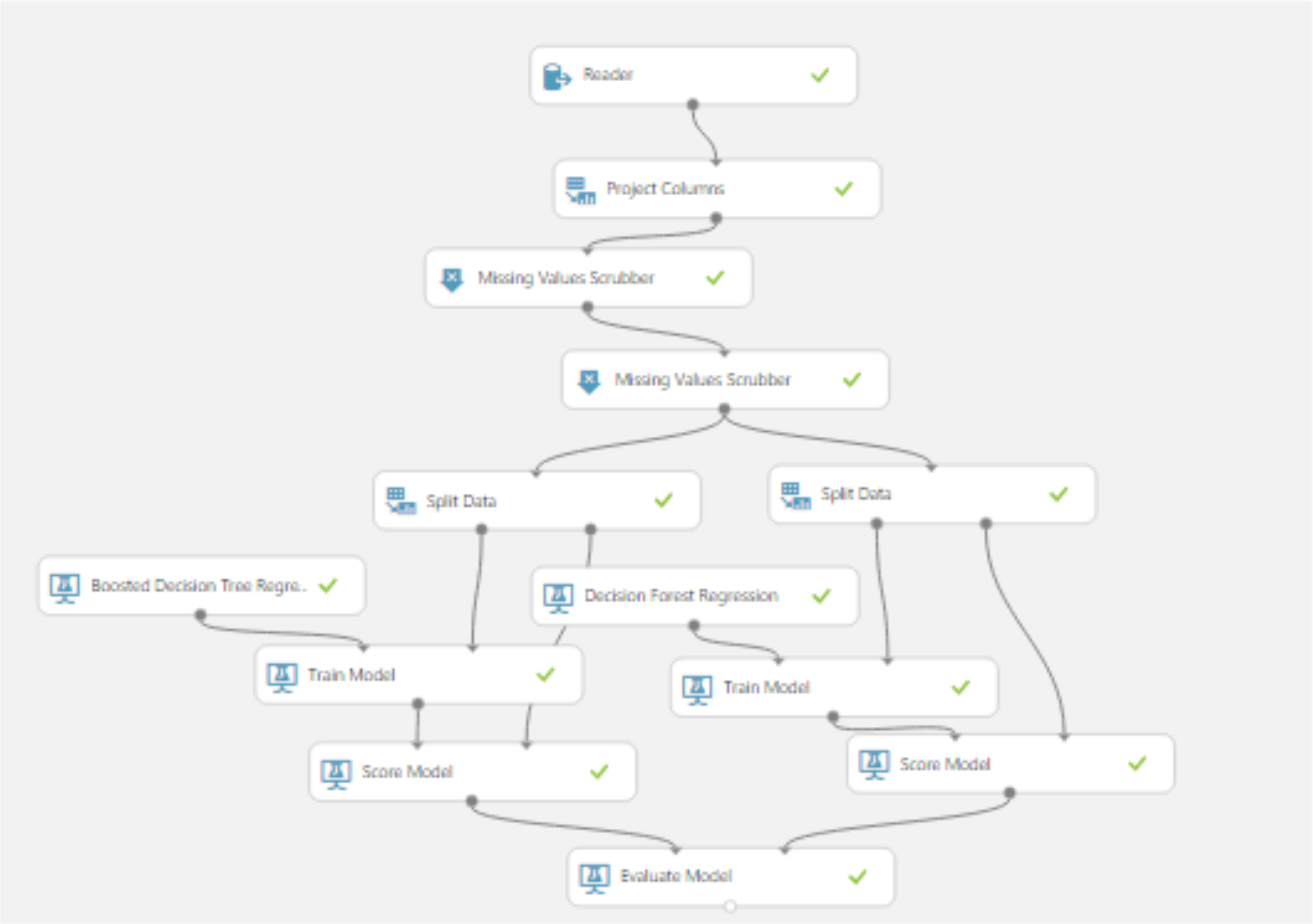


Metrics

Mean Absolute Error	90.008469
Root Mean Squared Error	125.155236
Relative Absolute Error	0.887289
Relative Squared Error	0.708058
Coefficient of Determination	0.291942



7. Prediction Model For Power Factor



7.1 Compared Algorithm

- In this experiment, we use two different algorithms for each compared model.
- The first algorithm is Boosted Decision Tree Regression, it enables to create an ensemble of regression trees using boosting. Boosting means that each tree is dependent on prior trees, and learns by fitting the residual of the trees that preceded it. Thus, boosting in a decision tree ensemble tends to improve accuracy with some small risk of less coverage. This regression method is a supervised learning method.
- Another algorithm is Decision Forest Regression. It is used to create a regression model using an ensemble of decision trees. Decision trees are non-parametric models that perform a sequence of simple tests for each instance, traversing a binary tree data structure until a leaf node (decision) is reached.

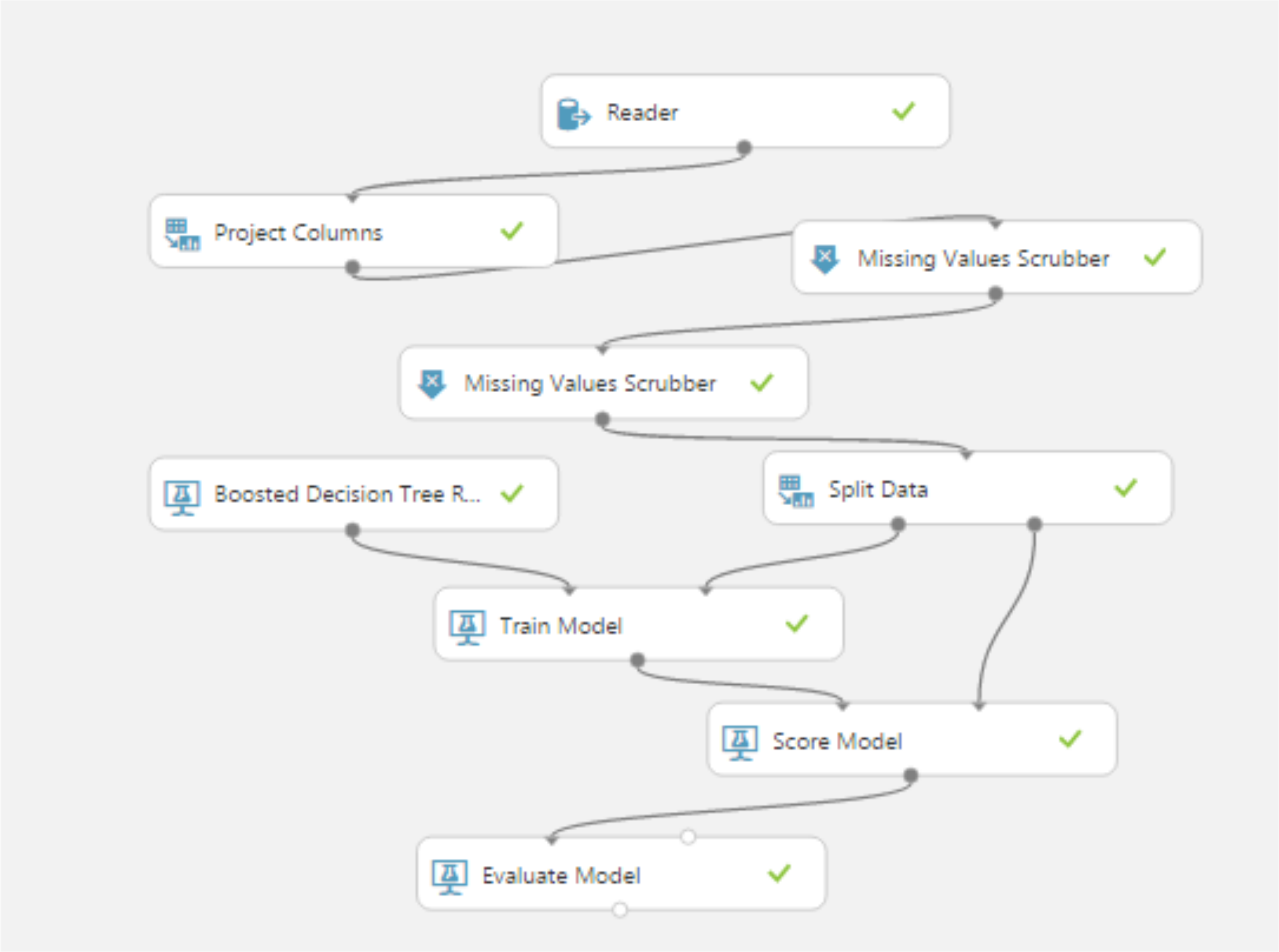
7.2 Evaluate Models

Negative Log Likelihood	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Relative Squared Error	Coefficient of Determination
Infinity	0.006345	0.009598	0.705026	0.655104	0.344896
-15.975275	0.006759	0.010133	0.751055	0.73012	0.26988

– Compared the first and second models, we can see the mean absolute error of second is slightly bigger than second, that means that the error rate of second is bigger than second's, and the average difference of first model between predictive values and real values are bigger than first's. Also, the RMS of second is bigger than first's, RMS stand for same situation as absolute error, but just squared it. For Negative Log Likelihood value, the smaller the better, so infinity negative is better than -15.

– For this result, we can say that first model which used Boosted Decision Tree Algorithm is slightly better than second used Decision Forest Regression Algorithm. So we choose the Boosted Decision model.

8. Prediction Model For KVarh



8.1 Improved Prediction Model

- In this experiment, we created only one model which uses Date, Highest Temperature, and Lowest Temperature. And it uses the Boosted Decision Tree Regression to train the model.
- We have already compared different variables subsets and different training algorithm. And based on prior two experiment, we found the best solution. And used them in this prediction model.

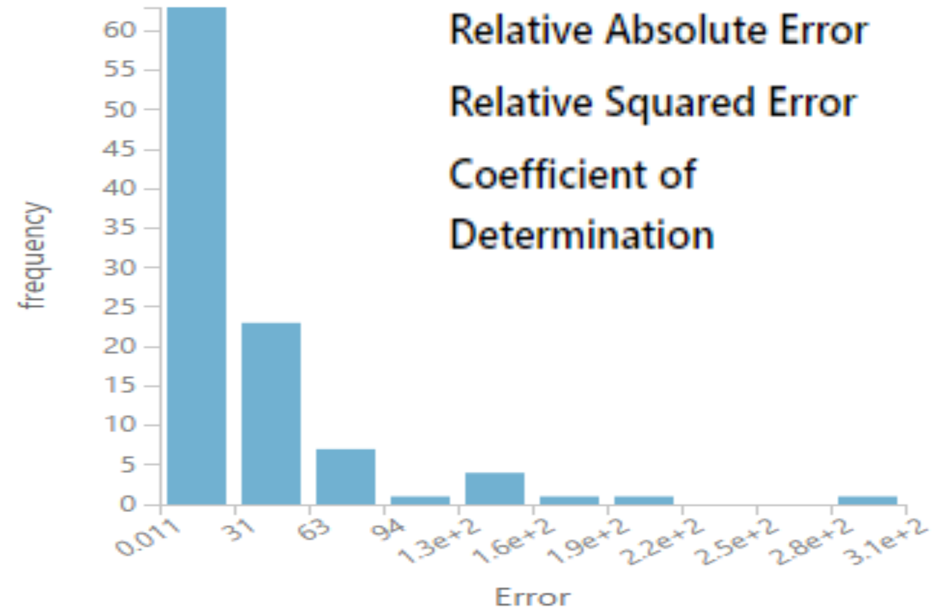
8.2 Evaluate Model

The Absolute Error, RMS and RSE of this model are acceptable. That means that the difference between predictive value and real value is acceptable. So, if these values are small, that means this model can predict kVarh value accurately.

Also, the Error Frequency indicates that this model will not make an error prediction very usual.

Metrics

Mean Absolute Error	35.853465
Root Mean Squared Error	59.188511
Relative Absolute Error	0.534795
Relative Squared Error	0.416622
Coefficient of Determination	0.583378



9. Deploy Web Service

- After creating three training models, Azure Machine Leaning Workspace can created three predictive models automatically. These three predictive models are similar with that three training models we created before.
- After predictive models were created, we deployed Web Service in Azure to get models’ APIs and URIs. These two values are very important for developing an integration of model.

9.1 API Key And URI

API key

kkkTShGdVpm7dR8HHmv2NJ6agfrh8pIsY2NzxdP3k/iHQ1fkSX50bX9n8bISEeDwOGucTyqjxsCNMv/Dbg0UFQ==

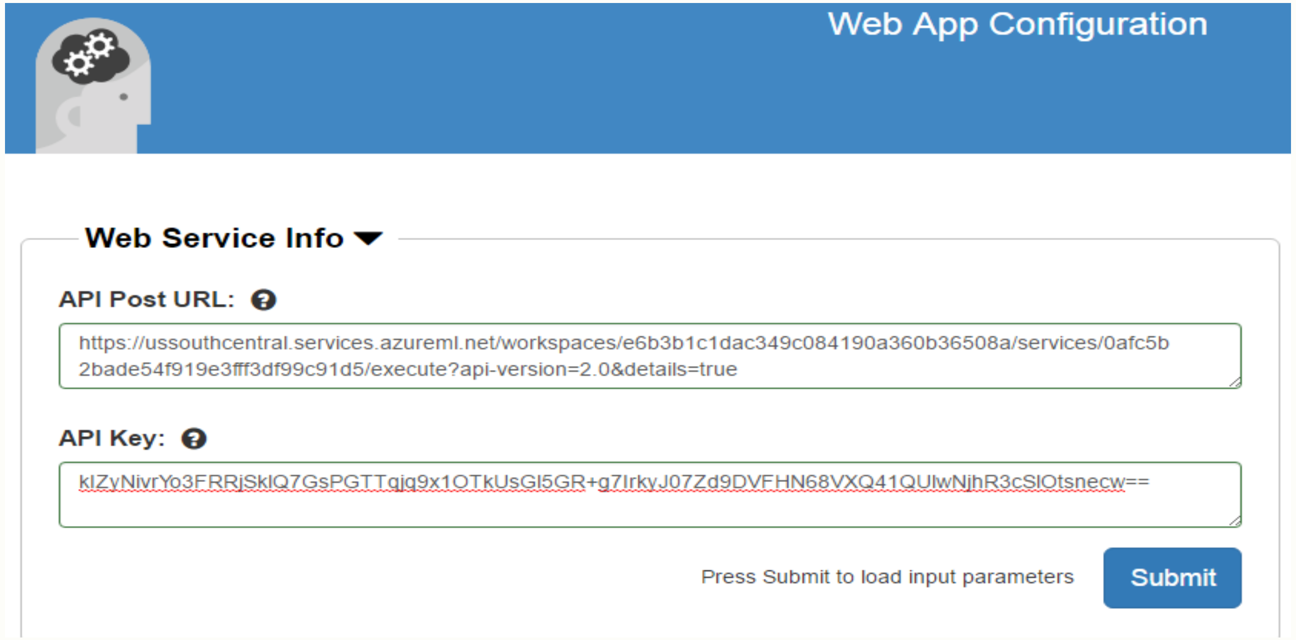
- API stands for application programming interface. It can be helpful to think of the API as a way for different apps to talk to one another. For many users, the main interaction with the API will be through API keys, which allow other apps to access your account without you giving out your password.

Method	Request URI	HTTP Version
POST	https://ussouthcentral.services.azureml.net/workspaces/e6b3b1c1dac349c084190a360b36508a/services/cc2cdd8470ba4da48991360db62cf289/execute?api-version=2.0&details=true	HTTP/1.1

- To paraphrase the World Wide Web Consortium, Internet space is inhabited by many points of content. A URI (Uniform Resource Identifier; pronounced YEW-AHR-EYE) is the way you identify any of those points of content, whether it be a page of text, a video or sound clip, a still or animated image, or a program. The most common form of URI is the Web page address, which is a particular form or subset of URI called a Uniform Resource Locator (URL).

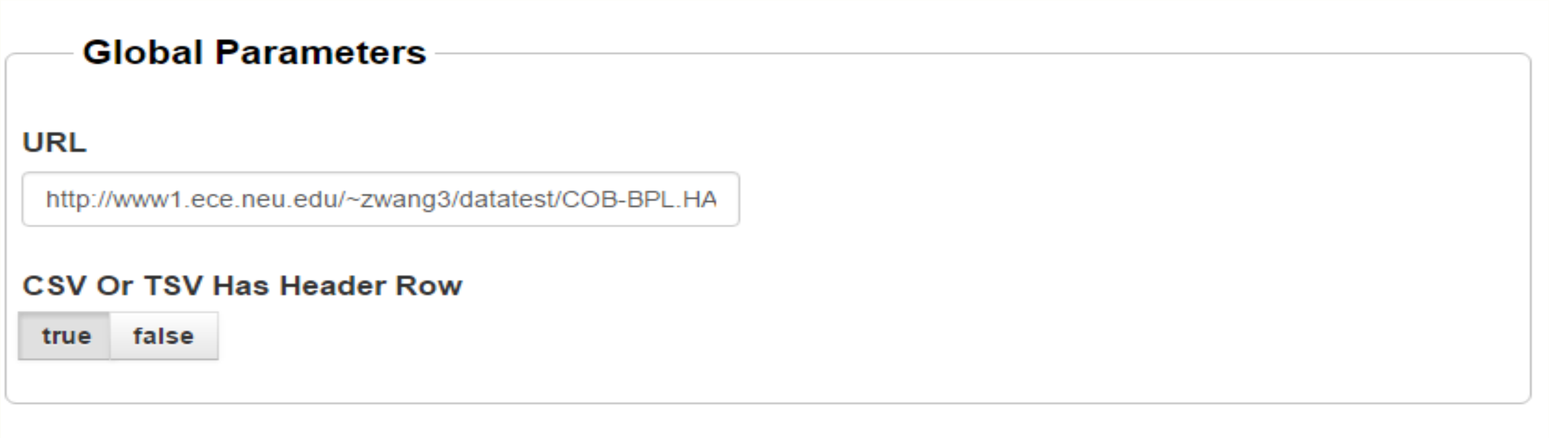
9.2 Models Integration

– We used Azure Web Service to create Web App. In Azure studio, we create an Azure ML Request-Response Service Web App, and input the API key and URL of models.



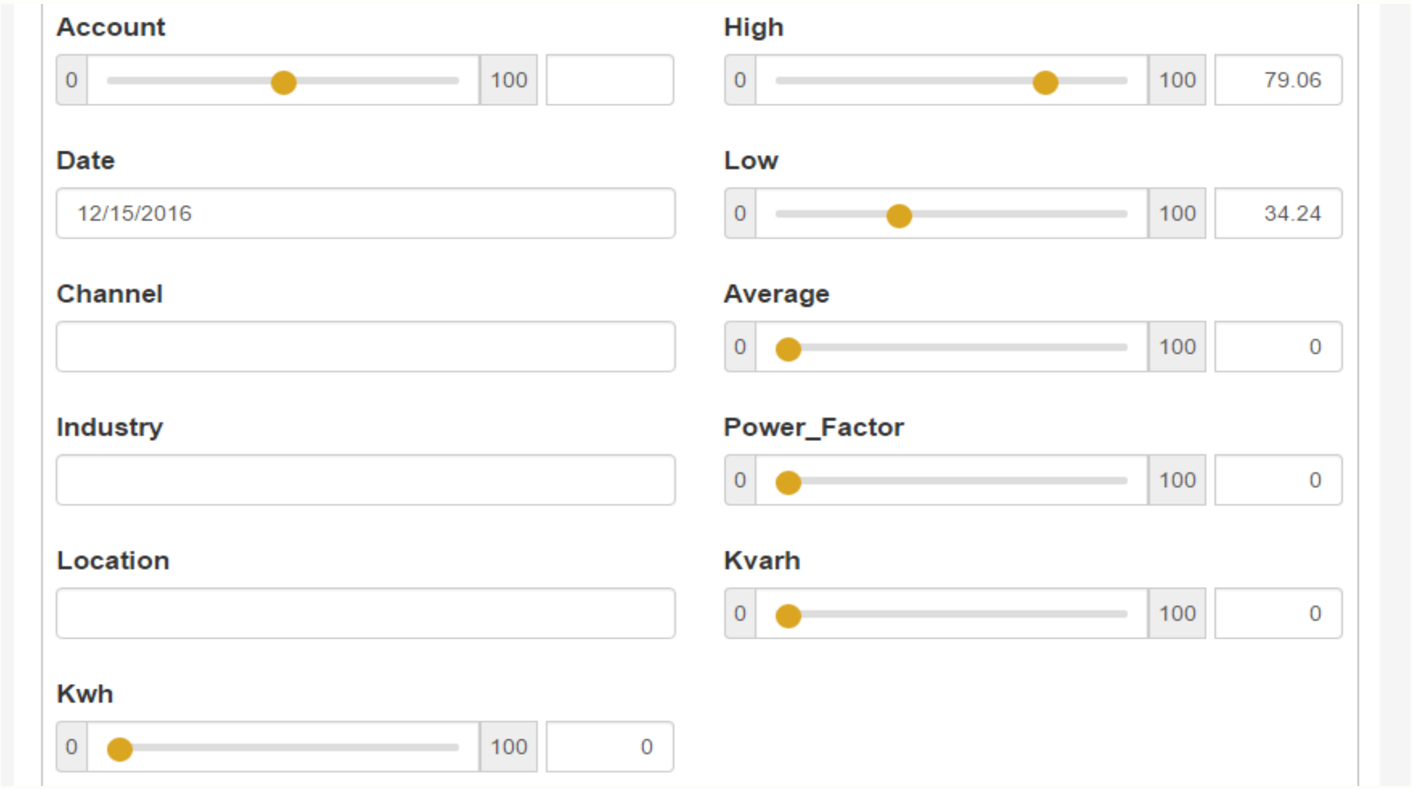
9.3 Result Testing

– We have uploaded all datasets to COE server. Thus, each dataset has its own URL. We set one of dataset's URL as the URL default value. Of course, users can change URL value to the URL of the dataset they want to use.



9.4 Result Testing (CON'T)

- For each dataset, the Location, Industry, and Account are same. So, user need not input these three parameters. Our models do not use Average Temperature and Channel feature. And users should not input the feture they want to predict.
- In conclusion, users only should input the Date they want to predict, the Highest Temperature and Lowest Temperature of that day.



9.4.1 Result of kWh Prediction

– The predicted total kWh value in 12/15/2016 was shown in the last line. This result come from the kWh prediction model created in Azure Machine Learning Workspace. About the result accuracy, we talked about it in Creating Models part before.

Submit

Result

Label	Value
output1	
Date	12/15/2016 12:00:00 AM
High	79.06
Low	34.24
Predicted kWh	563.620300292969

9.4.2 Result of Power Factor Prediction

– After clicking Submit button, we can get a result table. The predicted average power factor value for the 5/13/2017 was shown in the last line. This result come from the power factor prediction model created in Azure Machine Learning Workspace. About the result accuracy, we talked about it in Creating Models part before.

Submit

Result

Label	Value
output1	
Date	5/13/2017 12:00:00 AM
High	44.58
Low	26.35
Predicted Power Factor	0.962981879711151

9.4.3 Result of kVarh Prediction

- After clicking Submit button, we can get a result table. The predicted total kVarh value for 10/21/2018 was shown in the last line. This result come from the kVarh prediction model created in Azure Machine Learning Workspace. About the result accuracy, we talked about it in Creating Models part before.

Submit

Result

Label	Value
output1	
Date	10/21/2018 12:00:00 AM
High	88.42
Low	31.77
Predicted Kvarh	131.051177978516

10. Web App & UI

– UI Tools

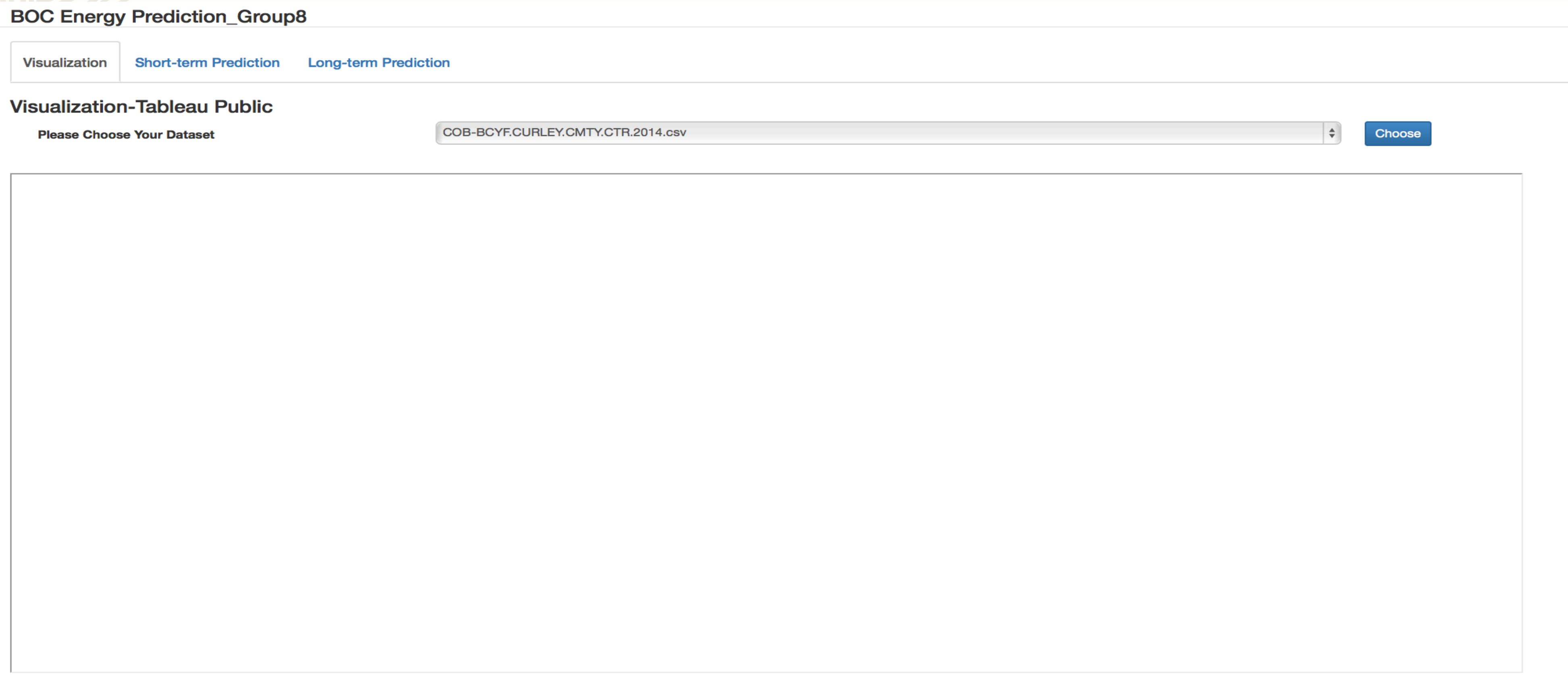
We develop our User Interface using HTML, CSS and Bootstrap. And we also combine our UI with web service UI from Azure.

10.2 Web Pages

– Home Page

– URL: http://www1.ece.neu.edu/~zwang3/energy_prediction_group8.html

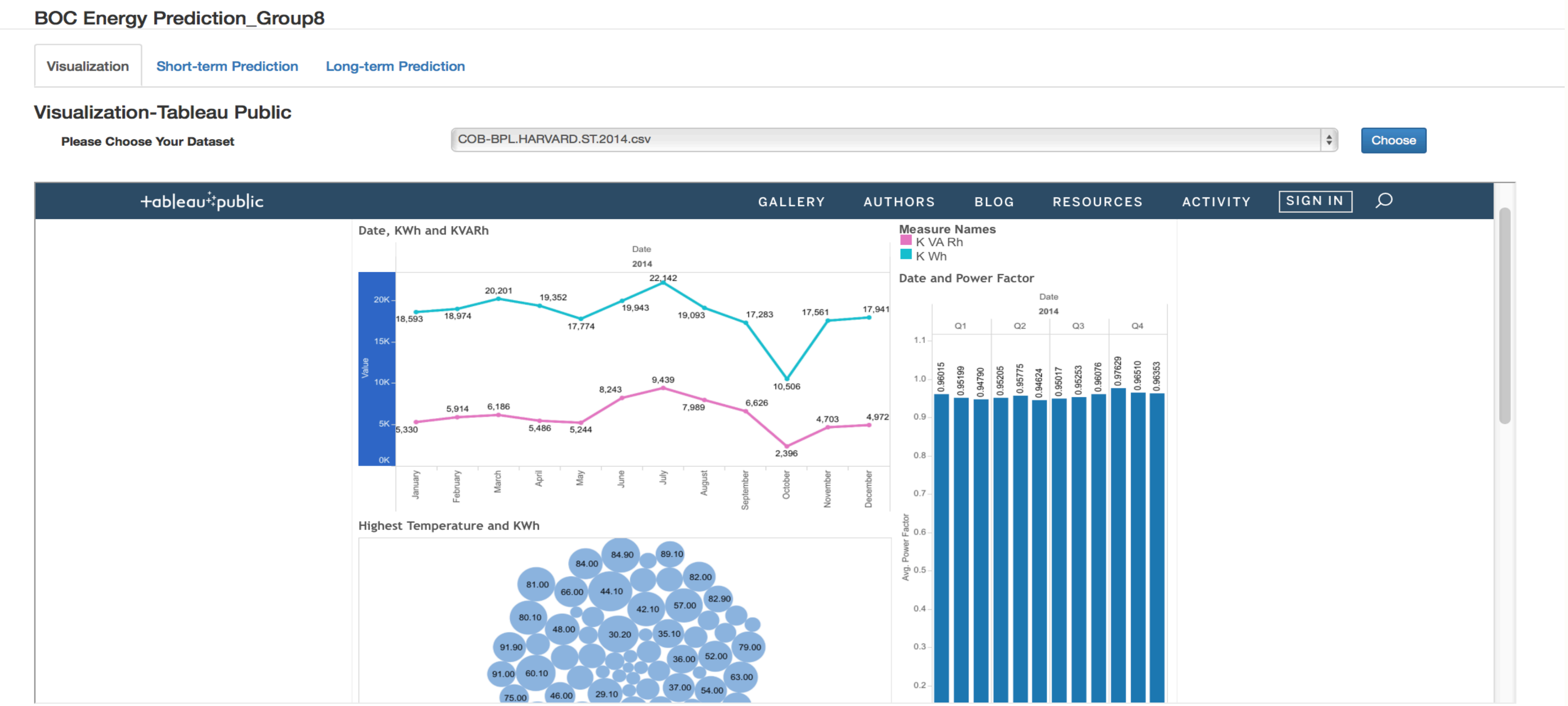
– The home page contains two parts: Visualization, Short-term prediction.



10.2 Web Pages

– Visualization Page

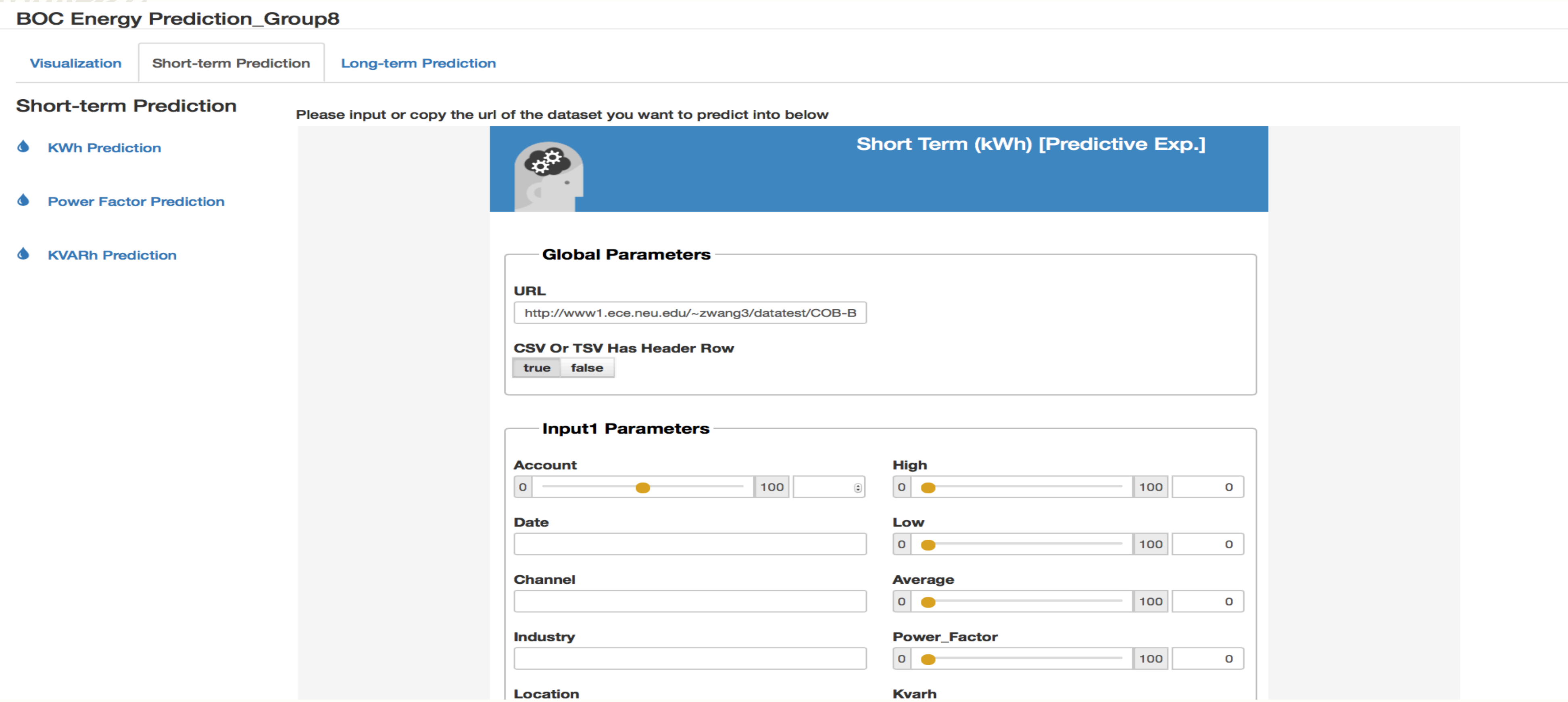
After click the “Visualization” tab, user can choose the dataset he want to analyze and predict with. After choose dataset and click “Choose” button, the visualization analysis for the chosen dataset will occur below. We use Tableau public for that.



10.2 Web Pages

– Short-term Prediction Page

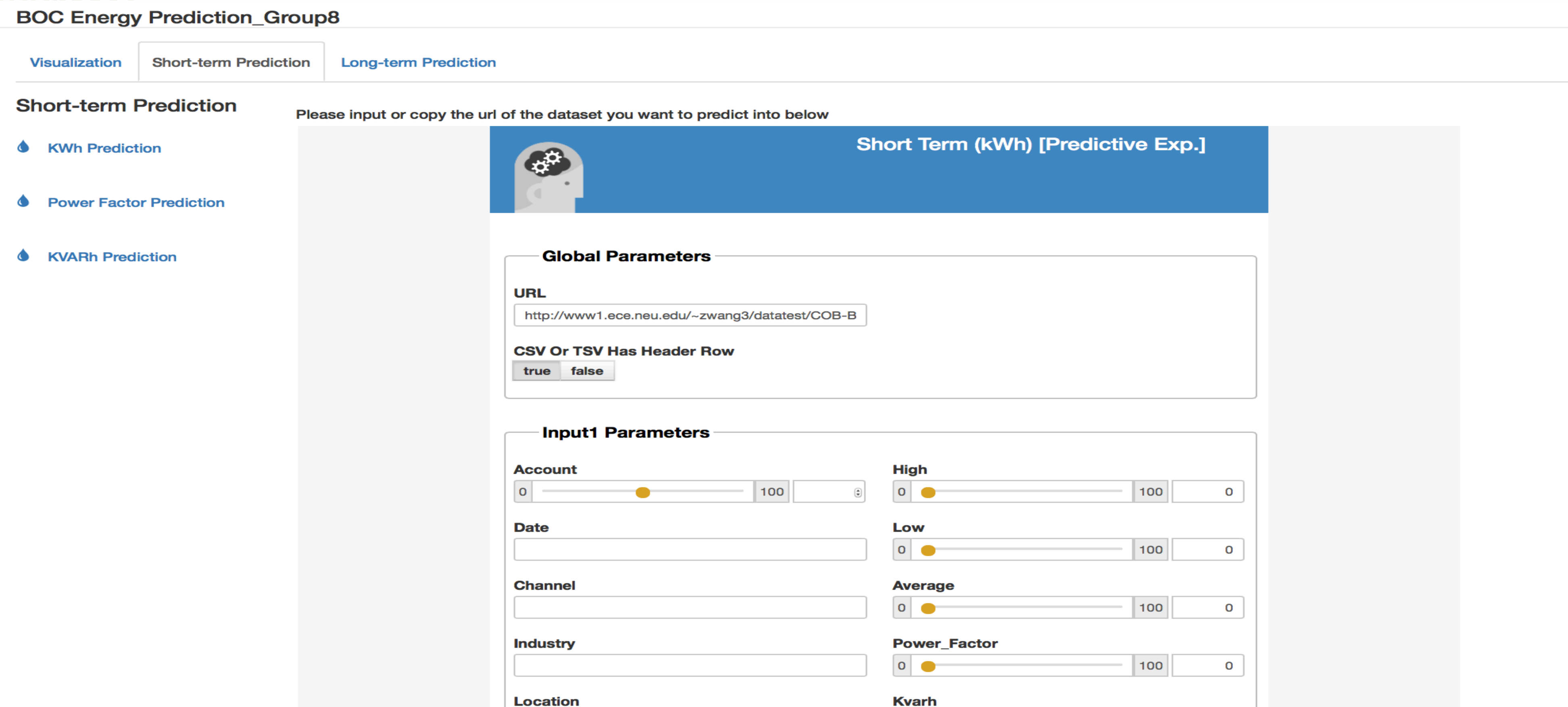
After click “Short-term Prediction” tab, user will enter short-term prediction page. And it contains three prediction models: kWh, Power Factor, KVarh. The default one is for kWh.



10.2 Web Pages

– Short-term kWh Prediction Page

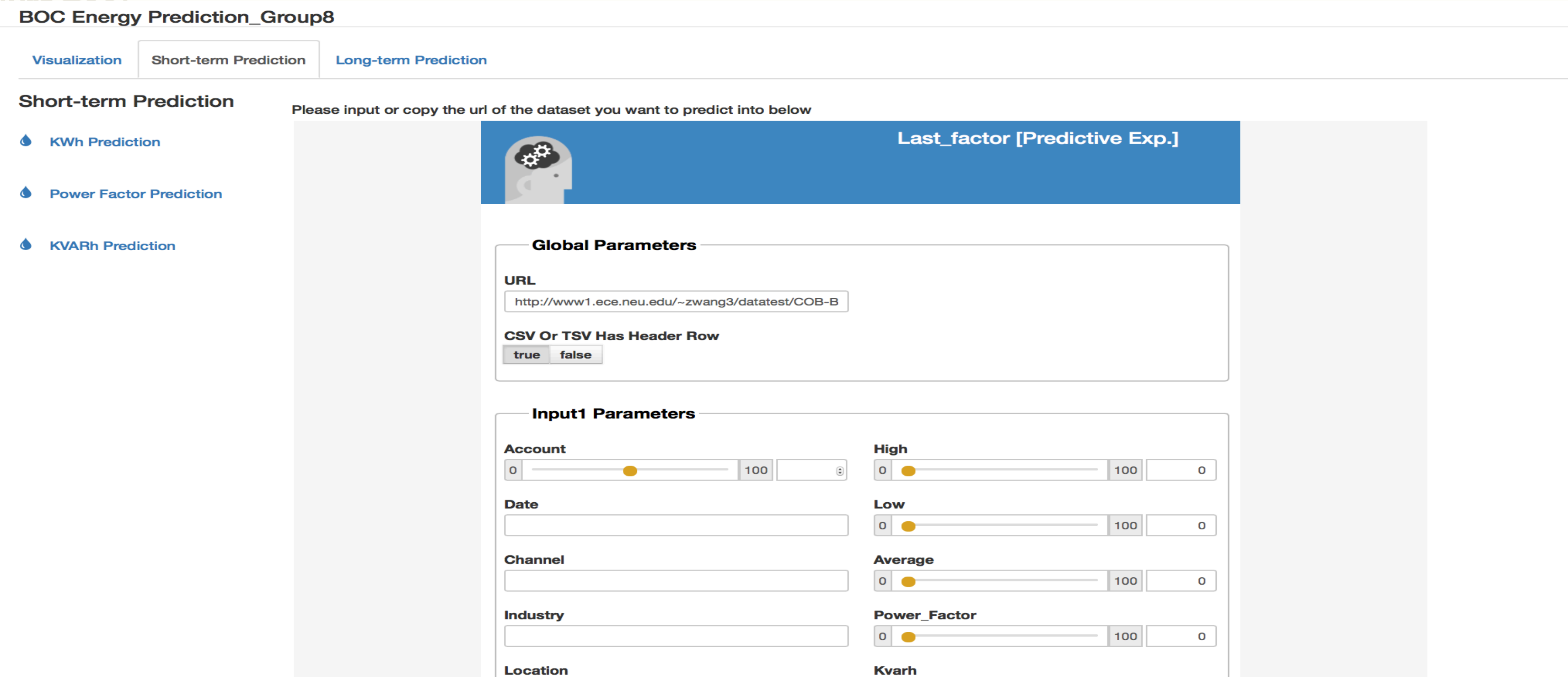
After choosing kWh prediction model, user can input or copy the URL of the dataset he want to predict into input box, and then fill the prediction forms with parameters, and click "submit" button. Then the prediction output will show in the bottom.



10.2 Web Pages

– Short-term Power Factor Prediction Page

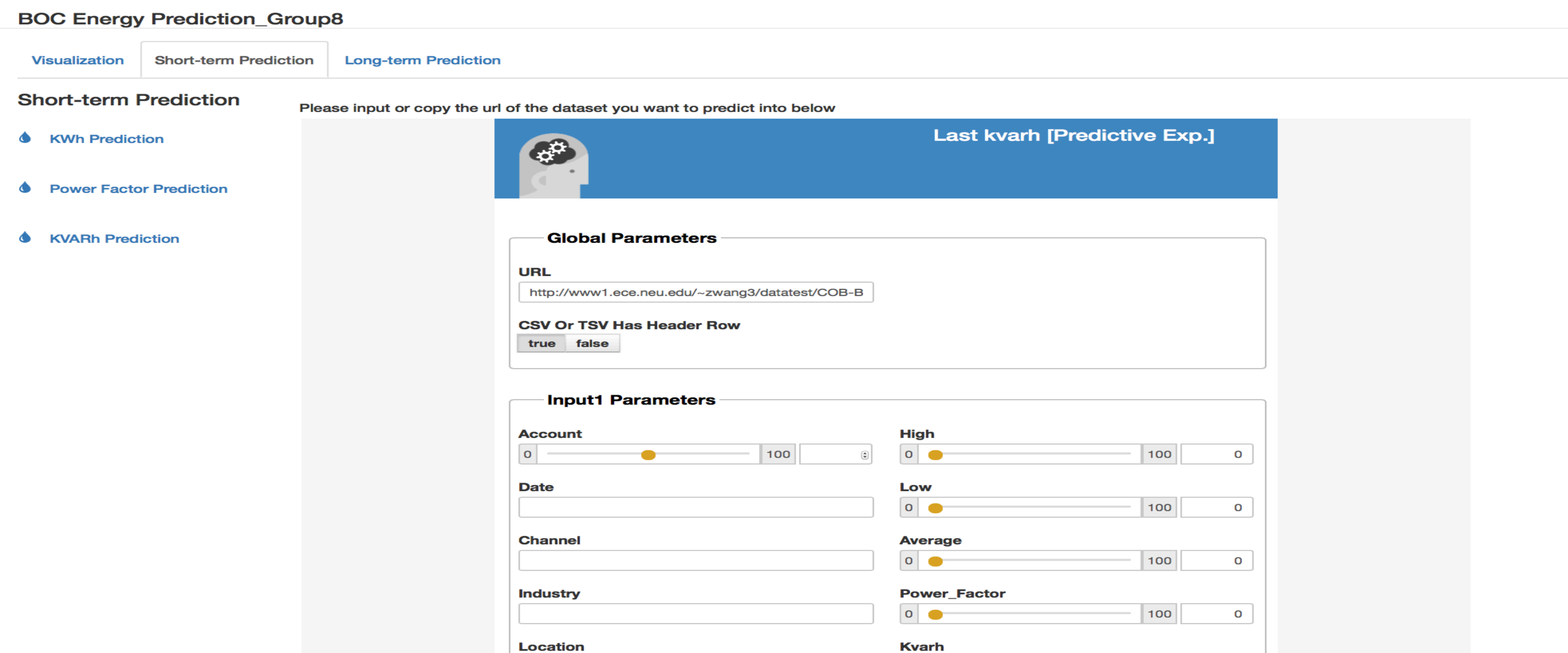
After choosing Power Factor prediction model, user can input or copy the URL of the dataset he want to predict into input box, and then fill the prediction forms with parameters, and click “submit” button. Then the prediction output will show in the bottom.



10.2 Web Pages

– Short-term KVarh Prediction Page

After choosing KVarh prediction model, user can input or copy the URL of the dataset he want to predict into input box, and then fill the prediction forms with parameters, and click “submit” button. Then the prediction output will show in the bottom.



11. Challenges & Improvement in the Future

- We should offer the long-term prediction to the customer. The Macro Analysis can predict energy consumption for future 1 week to 3 months, in order to help different industries to plan for future energy consumption, reduce power cost and save energy.
- The most important challenge to complete that part is data quality and size, since the data set is proved by customers. The customer can easy change the data set by change the URL. For long-term scenarios, it is important to have high quality data that covers a span of multiple years (minimum 3 years, preferable 5-10 years). Unfortunately, data sets which we used now is only cover a single year, so the performance of the long-term predict model will decline and hard to predict the meaningful value.

11. Challenges & Improvement in the Future(CON'T)

– Only the model we used can only prediction a single result each time. But for both short-term and long-term prediction, it will be better, if we can predict multiple result at once. For example, for the short- term prediction, if the customer wants to predict the energy consumption for next 24 hours, the website will general the total amount of energy consumption for a single day. It's will be better if the website can predict hourly amount of energy consumption and draw a line chart to show the trend of energy consumption.

– We should prove the better user experience by design the better user interface and offer better interactive. We should combine the visualization part with the prediction part better.

Thank you