REPORT

Info 7390
Assignment 3
Team 8

- **LINK FOR MICROSOFT AZURE MACHINE LEARNING ENVIRONMENT WEB HOSTING:**

  http://rainfallpredict.azurewebsites.net/

- **MOQUPS TOOL LINK TO BUILD MOCKUP**

  https://classic.moqups.com/jain.tan/glFnXgg4

- **WEB HOSTING LINK**

  http://rainfallpredictionteam8.azurewebsites.net/

OVERVIEW

The title of the project is "How Much Did It Rain?" The goal is to predict hourly rainfall using data from Polarimetric radars. The dataset consists of NEXRAD and MADIS data collected on 20 days between Apr and Aug 2014 over mid-western corn-growing states.

Rainfall is highly variable across space and time, making it notoriously tricky to measure. Rain gauges can be an effective measurement tool for a specific location, but it is impossible to have them everywhere. In order to have widespread coverage, data from weather radars is used to estimate rainfall nationwide.

Recently, in an effort to improve their rainfall predictors, the U.S. National Weather Service upgraded their radar network to be polarimetric. These polarimetric radars are able to provide higher quality data than conventional Doppler Radar because they transmit radio wave pulses with both horizontal and vertical orientations. Dual pulses make it easier to infer the size and type of precipitation because rain drops become flatter as they increase in size, whereas ice crystals tend to be elongated vertically.

DATASET

There are multiple radar observations over the course of an hour, and only one gauge observation (the 'Expected'). That is why there are multiple rows with the same 'Id'.

The columns in the datasets are:

- Id: A unique number for the set of observations over an hour at a gauge.
- minutes_past: For each set of radar observations, the minutes past the top of the hour that the radar observations were carried out. Radar observations are snapshots at that point in time.
- radardist_km: Distance of gauge from the radar whose observations are being reported.
- Ref: Radar reflectivity in dBZ
- Ref_5x5_10th: 10th percentile of reflectivity values in 5x5 neighborhood around the gauge.
- Ref_5x5_50th: 50th percentile
- Ref_5x5_90th: 90th percentile
- RefComposite: Maximum reflectivity in the vertical column above gauge. In dBZ.
- RefComposite_5x5_10th
- RefComposite_5x5_50th
- RefComposite_5x5_90th
- RhoHV: Correlation coefficient (unitless)
- RhoHV_5x5_10th
- RhoHV_5x5_50th
- RhoHV_5x5_90th
- Zdr: Differential reflectivity in dB
- Zdr_5x5_10th
- Zdr_5x5_50th
- Zdr_5x5_90th
- Kdp: Specific differential phase (deg/km)
- Kdp_5x5_10th
- Kdp_5x5_50th
- Kdp_5x5_90th
- Expected: Actual gauge observation in mm at the end of the hour.

A1 fx Id

	A	B	C	D	E	F	G	H	I	J	K
1	Id	minutes_past	radardist_km	Ref	Ref_5x5_10th	Ref_5x5_50th	Ref_5x5_90th	RefComposite	RefComposite_5x5_10th	RefComposite_5x5_50th	RefComposite_5x5_90th
2	1	3	10								
3	1	16	10								
4	1	25	10								
5	1	35	10								
6	1	45	10								
7	1	55	10								
8	2	1	2	9	5	7.5	10.5	15	10.5	16.5	23.5
9	2	6	2	26.5	22.5	25.5	31.5	26.5	26.5	28.5	32
10	2	11	2	21.5	15.5	20.5	25	26.5	23.5	25	27
11	2	16	2	18	14	17.5	21	20.5	18	20.5	23
12	2	21	2	24.5	16.5	21	24.5	24.5	21	24	28
13	2	26	2	12	12	16	20	16.5	17	19	21
14	2	31	2	22.5	19	22	25	26	23.5	25.5	27.5
15	2	37	2	14	14	18.5	21	19.5	20	21	23
16	2	42	2	12	11	12.5	17	19.5	18	21	23
17	2	47	2	1.5	3.5	7	10.5	18	16.5	18.5	21.5
18	2	53	2	16	14.5	18	23.5	28	23.5	26.5	29.5
19	2	58	2	22	16.5	22.5	26.5	31.5	26.5	29	32
20	3	4	10								
21	3	9	10								
22	3	14	10								
23	3	18	10								
24	3	23	10								
25	3	28	10								
26	3	33	10								
27	3	38	10								
28	3	43	10				8.5				8.5

K1 fx RefComposite_5x5_90th

	L	M	N	O	P	Q	R	S	T	U	V	W	X
1	RhoHV	RhoHV_5x5_10th	RhoHV_5x5_50th	RhoHV_5x5_90th	Zdr	Zdr_5x5_10th	Zdr_5x5_50th	Zdr_5x5_90th	Kdp	Kdp_5x5_10th	Kdp_5x5_50th	Kdp_5x5_90th	Expected
2													0.2540001
3													0.2540001
4													0.2540001
5													0.2540001
6													0.2540001
7													0.2540001
8	0.9983333	0.99833333	0.99833333	0.99833333	0.375	-0.125	0.3125	0.875	1.06	-1.4100037	-0.3500061	1.0599976	1.0160005
9	1.0016667	0.9816667	0.99833333	1.005	0.063	-0.1875	0.25	0.6875				1.4099884	1.0160005
10	1.0016667	0.995	0.99833333	1.0016667	0.313	-0.0625	0.3125	0.625	0.35		-0.3500061	1.7599945	1.0160005
11	0.995	0.995	0.99833333	1.0016667	0.25	0.125	0.375	0.6875	0.35	-1.0599976	0	1.0599976	1.0160005
12	0.9983333	0.995	0.99833333	0.99833333	0.25	0.0625	0.1875	0.5625	-0.35	-1.0599976	-0.3500061	1.7599945	1.0160005
13	0.9983333	0.995	0.99833333	0.99833333	0.563	0.25	0.4375	0.6875	-1.76	-1.7600098	-0.3500061	0.70999146	1.0160005
14	0.9983333	0.995	0.99833333	1.0016667	0	-0.1875	0.25	0.625	-1.06	-2.1200104	-0.7100067	0.34999084	1.0160005
15	0.9983333	0.9916667	0.99833333	0.99833333	0.5	0.1875	0.4375	0.8125	0	-1.7600098	-0.3500061	1.0599976	1.0160005
16	0.9983333	0.995	0.99833333	0.99833333	0.625	0.375	0.625	0.875	-0.35	-0.3500061	0	0.34999084	1.0160005
17	0.9983333	0.995	0.99833333	0.99833333	0.375	0.1875	0.5	0.6875	0.35	-2.1100006	-0.3500061	1.0599976	1.0160005
18	0.9983333	0.9916667	0.99833333	0.99833333	0.875	0.625	0.9375	1.375	-0.35	-1.4100037	-0.3500061	2.119995	1.0160005
19	0.9983333	0.995	0.99833333	1.0016667	0.375	0.1875	0.375	0.875	-1.41		-0.3500061	0.69999695	1.0160005
20													26.162014
21													26.162014
22													26.162014
23													26.162014
24													26.162014
25													26.162014
26													26.162014
27													26.162014
28													26.162014

CLEANING THE DATA:

Following is the code for cleaning the data in R. The R-Script is executed in azure machine learning environment.

1) **Prediction – Regression using Neural Network**

```
#Map 1-based optional input ports to variables
dataset1 <- maml.mapInputPort(1)  # class: data.frame

#Removing all the rows where the value of Ref column is 0
df_removedRef<-dataset1[!(is.na(dataset1$Ref) | dataset1$Ref==""), ]

#Replacing all NAs with 0 so that there is no missing data as NA values are missing real time data that cannot be averaged by other values
df_removedRef[is.na(df_removedRef)]<-0

#Mean of column "Expected"
mean_Expected<-mean(df_removedRef$Expected)

#Standard Deviation of column "Expected"
sd_Expected<-sd(df_removedRef$Expected)

#Range of values (covering 95% of values) by Normal Distribution
#Positive Maxima
result_more<-mean_Expected+(1.5*sd_Expected)

#Negative Minima
result_less<-mean_Expected-(1.5*sd_Expected)

#Removing the outlier values by removing rows with Expected values more than result_more or less than result_less
variable_removedrows<-df_removedRef[with(df_removedRef,  !((Expected > result_more | Expected < result_less))), ]

maml.mapOutputPort("variable_removedrows");
```

2) **Classification – Two Class Boosted Tree Algorithm**

```
# Map 1-based optional input ports to variables
dataset1 <- maml.mapInputPort(1)  # class: data.frame

#Removing all the rows where the value of Ref column is 0
df_removedRef<-dataset1[!(is.na(dataset1$Ref)  | dataset1$Ref==""), ]

#Replacing all NAs with 0 so that there is no missing data as NA values are missing real time data that cannot be averaged by other values
```

```
df_removedRef[is.na(df_removedRef)]<-0

#Mean of column "Expected"
mean_Expected<-mean(df_removedRef$Expected)

#Standard Deviation of column "Expected"
sd_Expected<-sd(df_removedRef$Expected)

#Range of values (covering 95% of values) by Normal Distribution
#Positive Maxima
result_more<-mean_Expected+(1.5*sd_Expected)
#Negative Minima
result_less<-mean_Expected-(1.5*sd_Expected)

#Removing the outlier values by removing rows with Expected values more than result_more or less than result_less
variable_removedrows<-df_removedRef[with(df_removedRef, !((Expected > result_more | Expected < result_less))), ]

mean_Expected<- mean(variable_removedrows$Expected)
View(mean_Expected)

Expected_Class<-
ifelse(variable_removedrows$Expected>mean_Expected,"Above_Normal","Optimal"
)
df_final<-data.frame(variable_removedrows,Expected_Class)

maml.mapOutputPort("df_final");
```

MODEL BUILDING

- **Regression Model (Neural Network Algorithm)**

1) We had train.csv file which contain very large dataset (13765201 rows and 24 attributes).

2) We cleaned the data. Firstly, we remove all the rows where the value of Ref column was 0

3) We replaced all NA's with 0 so that there is no missing data as NA values are missing real time are missing real time real time data cannot be averaged by other values.

4) We wanted to remove outliers for the file. For that we took mean and standard deviation of column "Expected" and removed all the outliers with Expected values more or less than appropriate one.

5) After cleaning the data, we were left with around 60lakhs rows

6) We took filter based feature selector to select the best features for our model. After doing this we found 5 features that were very important.

7) Then we split our input cleaned Data into training and testing.

8) Then we applied different algorithms and found neural network to be the best one.

9) After that, we got predicted values for test file and evaluated MAE, RMSE values.

OUTPUT:

- **Classification Model (Two-Class Boosted Decision Tree)**

1) We had train.csv file which contain very large dataset (130 million rows and 24 attributes).

2) We cleaned the data. Firstly, we remove all the rows where the value of Ref column was 0

3) We replaced all NA's with 0 so that there is no missing data as NA values are missing real time are missing real time real time data cannot be averaged by other values.

4) We wanted to remove outliers for the file. For that we took mean and standard deviation of column "Expected" and removed all the outliers with Expected values more or less than appropriate one.

5) After cleaning the data, we were left with around 60lakhs rows

6) We also classified the expected column into above normal and optimal by taking the mean of the expected column.

7) We then removed the expected column by taking select column in Dataset block.

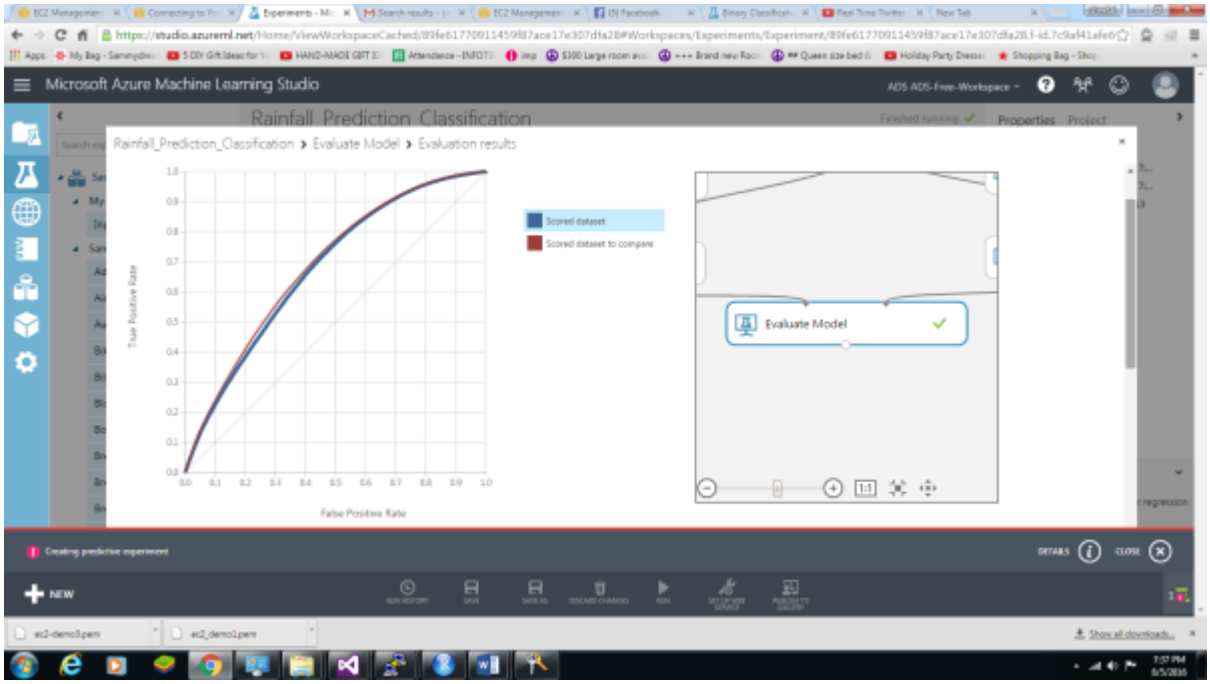
8) We took filter based feature selector to select the best features for our model. After doing this we found 5 features that were very important.

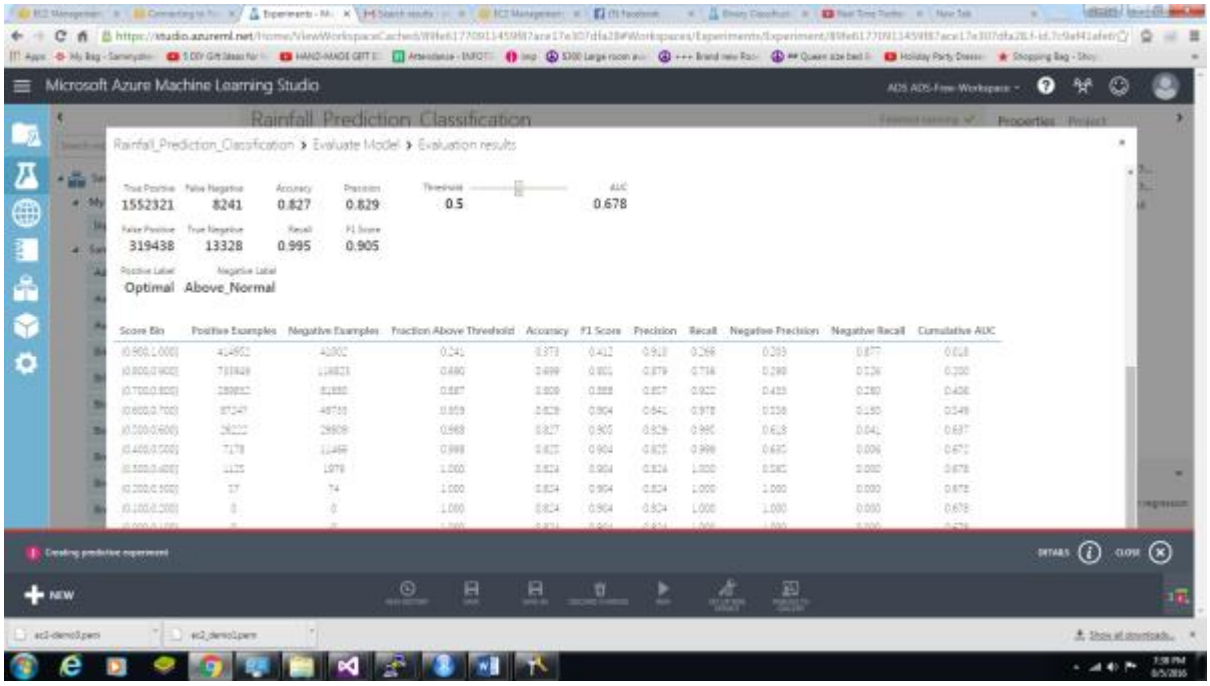
9) Then we split our input cleaned Data into training and testing.

10) Then we applied different algorithms and found Two-Class Boosted Decision Tree to be the best one.

11) After that, we got predicted values for test file and evaluated MAE, RMSE values, Accuracy.

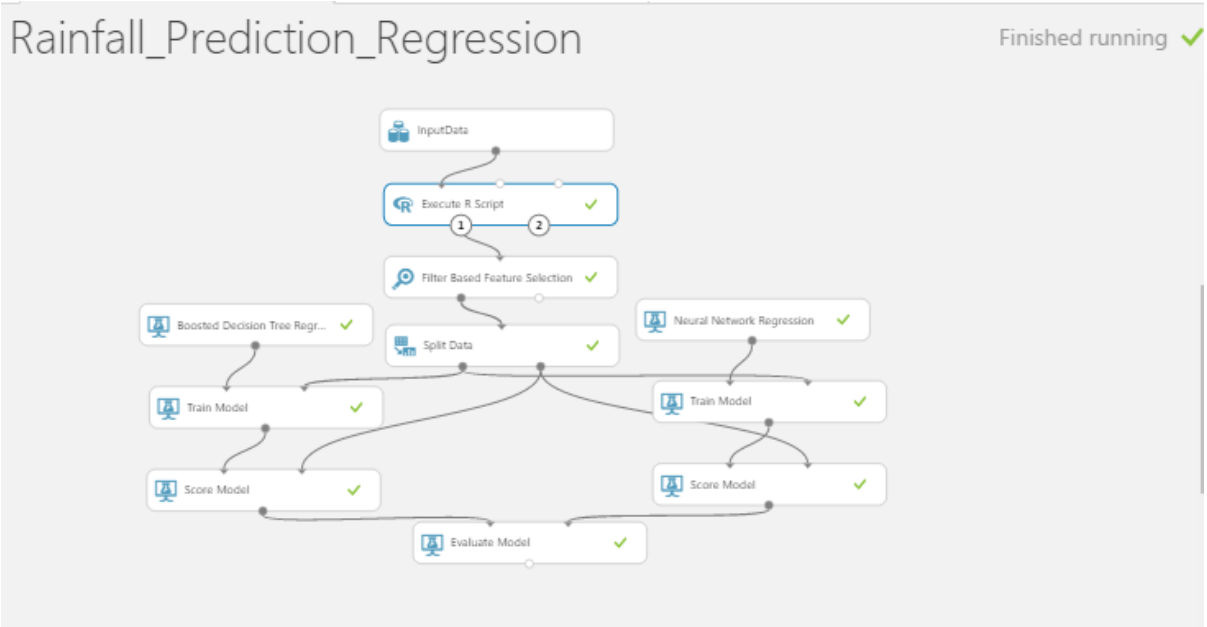
OUTPUT:



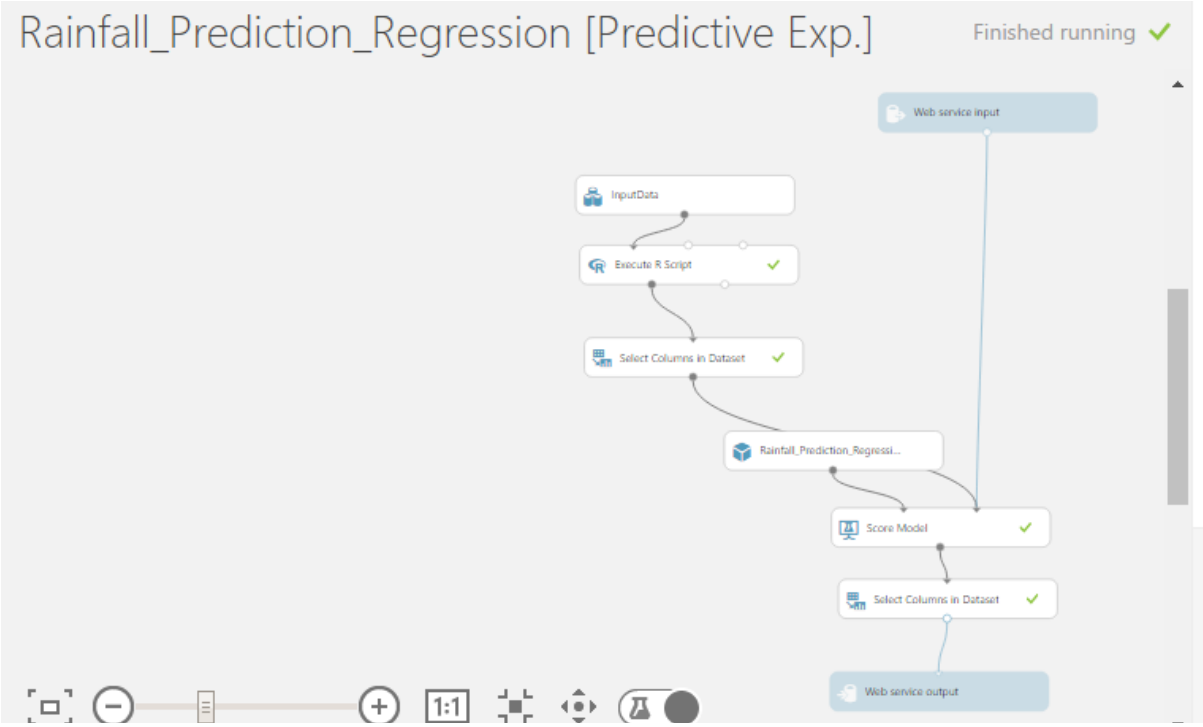

Microsoft Azure Machine Learning Studio
Rainfall_Prediction_Classification
Rainfall_Prediction_Classification › Evaluate Model › Evaluation results
True Positive 1552321
False Negative 8241
Accuracy 0.827
Precision 0.829
Threshold 0.5
AUC 0.678
False Positive 319438
True Negative 13328
Recall 0.995
F1 Score 0.905
Positive Label Optimal
Negative Label Above_Normal
Score Bin
Positive Examples
Negative Examples
Fraction Above Threshold
Accuracy
F1 Score
Precision
Recall
Negative Precision
Negative Recall
Cumulative AUC
NEW

MODELS ON AZURE MACHINE LEARNING ENVIRONMENT:

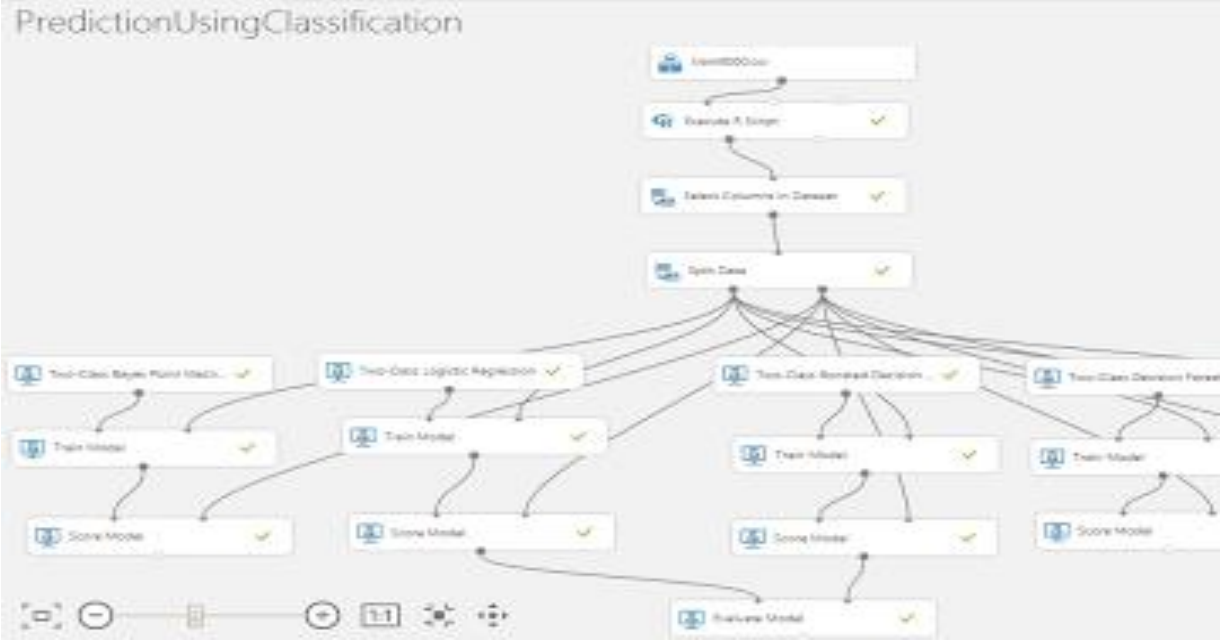
1) Regression Model



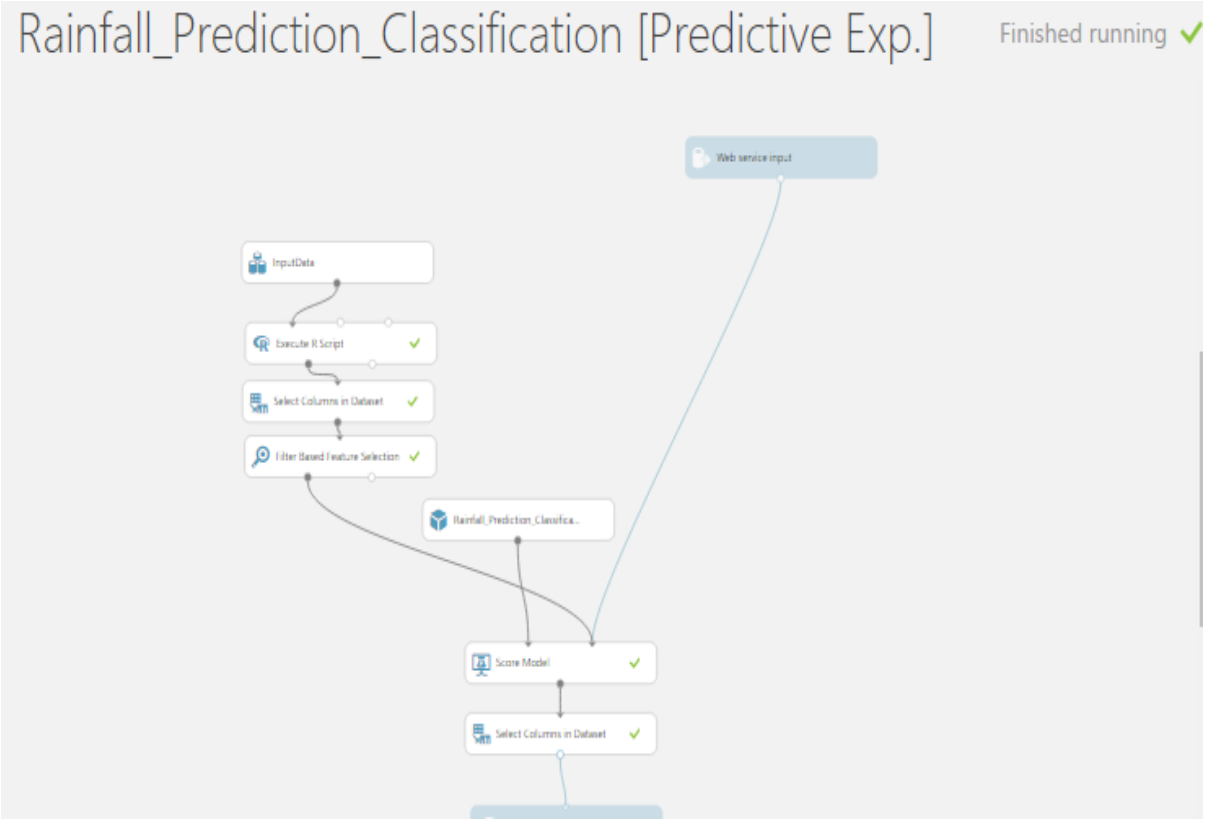
2) Predictive Experiment of Regression (Deploying as web service)



3) Classification Predictive model



4) Predictive Experiment of Classification (Deploying as web service)



BUSINESS PROSPECT

Climate variability leads to economic and food security risks throughout the world because of its major influences on agriculture. Accurate forecasts of climate 3–6 months ahead of time can potentially allow farmers and others in agriculture to make decisions to reduce unwanted impacts or take advantage of expected favorable climate. However, potential benefits of climate forecasts vary considerably because of many physical, biological, economic, social, and political factors.

- Wholesalers and retailers of production inputs (for example, improved marketing strategies, inventory management and advice to clients)
- Grain/fibre handling and marketing organisations, and processors (better forecasts of production and quality)
- Forward sellers and purchasers of products (more accurate estimations of future prices)
- Water resource management.

REFERENCES:

1) Kaggle dataset
   https://www.kaggle.com/c/how-much-did-it-rain-ii

2) Mockup Tool - moqups
   https://moqups.com/#!/edit/jain.tan/glFnXgg4

3) Web Hosting Help
   https://studio.azureml.net/Home/ViewWorkspaceCached/89fe61770911459f87ace17e307dfa28#Workspace/Experiments/ListExperiments

4) Visual Studio
   https://www.visualstudio.com/products/visual-studio-community-vs