Project 2

Prediction of average temperature and humidity of Austin/Texas weather based on time series analysis

Abstract

In this analysis, time series data frame for weather conditions of Austin/Texas has been studied. In fact, the dataset includes many features based on timestamp. The data wrangling for this project has been applied to remove all null values and meaningless data and filled out some blanks with appropriate categorical variables for specific feature (Column of events). Data visualization and statistical analysis including test statistics and linear regression have been performed to observe some correlations and trends in the dataset. In the next step, machine learning has been implemented to predict two important quantities which are average temperature and average humidity. Different models were assessed for these response variables for making prediction. Performance evaluation based on Bayesian Information Criterion (BIC) was utilized for every model to find the optimum order of the function. Actually, it was observed that the modelling with linear regression included multiple features was the best function to have precise prediction in this study.

1-Introduction

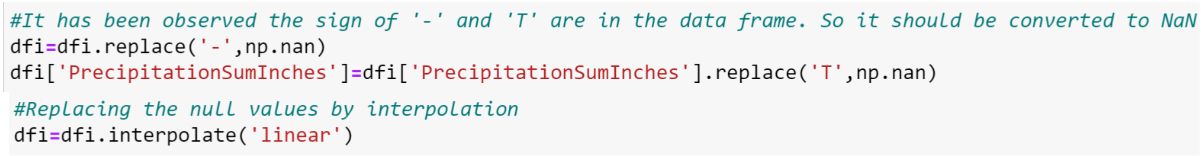
Weather conditions and its forecasting have specific importance for human health, economy, airplane flights, transportation, tourism and etc. So, it is required to focus on different parameters of weather and prediction of key quantities in this regard to improve the safety and reliability of different industries. In fact, weather forecasting is essential particularly by computer technology. In this project, statistical analysis and modelling have been implemented for Austin/Texas weather condition. According to this project, data visualization included scatter plot, histogram and line plot were utilized. In addition, test statistics related to null hypothesis about correlation between average temperature and pressure was studied. Furthermore, linear regression analysis to propose a straight line between these maximum and minimum temperatures was performed to observe the correlation. In next step, machine learning with different models have been applied. Aa a matter of fact, important quantities which are average temperature and average humidity were focused for prediction of last 21 days of July 2017. Five models were applied for these predictions and comparison was made to observe the model performance. Two methods were implemented for this project included modelling with single feature and modelling with multiple features. AR, MA, ARMA and ARIMA models have been utilized regarding single feature and LinearRegression model has been applied regarding multiple features. Based on analysis and model performance in this project, it is concluded that the ARIMA has the best performance rather than other models which have single feature and LinearRegression model with multiple features have the best results with respect to all of models utilized in the project.

2-Method and analysis:

A data frame regarding collected statistical values for Austin/Texas weather is analyzed in this study. Data wrangling, exploratory data analysis, modelling and prediction based on this dataset were applied in the analysis.

2-1-Data wrangling:

According to the dataset, data cleaning for EDA and modelling has been performed. This data frame has 1319 rows and 21 columns. The features of this dataset include many important quantities like: maximum temperature, average temperature, maximum humidity, average humidity, high air pressure and etc. According to this analysis, all of blanks of the data frame have been found and filled out by interpolation. In addition, based on some searches and assessment of the table, some meaningless values like '-' and 'T' have been observed. So, they should be removed, these values are erased and filled out by interpolation. Figure 1 shows the code for replacing meaningless values with NaN and replacing NaN by interpolation.

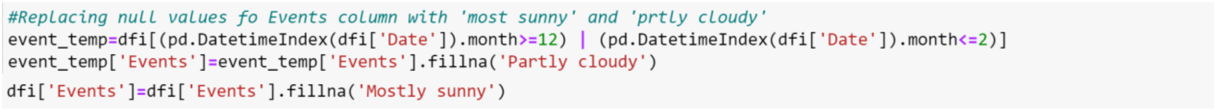


```
#It has been observed the sign of '-' and 'T' are in the data frame. So it should be converted to NaN
dfi=dfi.replace('-',np.nan)
dfi['PrecipitationSumInches']=dfi['PrecipitationSumInches'].replace('T',np.nan)

#Replacing the null values by interpolation
dfi=dfi.interpolate('linear')
```

Figure 1-Applied code to find meaningless values and interpolation

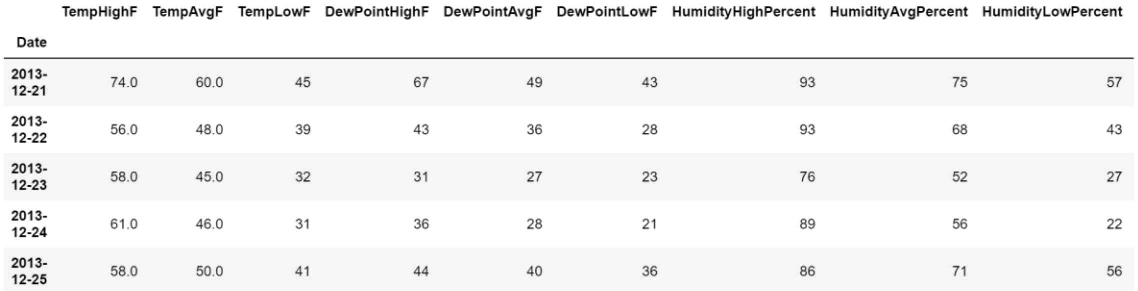
In next step, there is a column in the table called 'Events'. This feature includes some status regarding weather condition like 'Rain', 'Thunderstorm', 'Fog', 'Snow' and etc. On the other hand; there are many blanks for this feature. These blanks should be filled out by appropriate phrase. So, for this purpose research was performed to make decision about these blanks. Based on the research from some valid sources, most probably the weather status for months of December to February is mostly cloudy in Austin. In addition, most of the weather status for months of March to November is mostly sunny. Consequently, it is possible to fill out the blanks of 'Events' with appropriate status based on classification of months. By this method the blanks of this feature have been filled by categorical variables. In fact, the column of 'Events' includes 10 unique categorical variables after filling. Figure 2 depicts the code for making decision regarding weather status of Austin/Texas.



```
#Replacing null values fo Events column with 'most sunny' and 'prtly cloudy'
event_temp=dfi[(pd.DatetimeIndex(dfi['Date']).month>=12) | (pd.DatetimeIndex(dfi['Date']).month<=2)]
event_temp['Events']=event_temp['Events'].fillna('Partly cloudy')
dfi['Events']=dfi['Events'].fillna('Mostly sunny')
```

Figure 2-Applied code for making decision to put appropriate phrase for weather status

In addition, the index of the data table has been set to timestamp for time series assessment. Figure 3 displays first 5 rows of this dataset after cleaning.



	TempHighF	TempAvgF	TempLowF	DewPointHighF	DewPointAvgF	DewPointLowF	HumidityHighPercent	HumidityAvgPercent	HumidityLowPercent
Date									
2013-12-21	74.0	60.0	45	67	49	43	93	75	57
2013-12-22	56.0	48.0	39	43	36	28	93	68	43
2013-12-23	58.0	45.0	32	31	27	23	76	52	27
2013-12-24	61.0	46.0	31	36	28	21	89	56	22
2013-12-25	58.0	50.0	41	44	40	36	86	71	56

Figure 3-First five rows of the data frame

The table has 20 features after cleaning which are quantities for weather conditions measurement. Actually, these features are very important for modelling and prediction. All of features have been collected in the table1. Table 1 shows the columns of dataset.

Column number	Column name
1	TempHighF
2	TempAvgF
3	TempLowF
4	DewPointHighF
5	DewPointAvgF
6	DewPointLowF
7	HumidityHighPercent
8	HumidityAvgPercent
9	HumidityLowPercent
10	SeaLevelPressureHighInches
11	SeaLevelPressureAvgInches
12	SeaLevelPressureLowInches
13	VisibilityHighMiles
14	VisibilityAvgMiles
15	VisibilityLowMiles
16	WindHighMPH
17	WindAvgMPH
18	WindGusMPH
19	PrecipitationSumInchs
20	Events

Table 1- Columns of the dataset

2-2-Data analysis and visualization

Some graphs and curves included maximum and minimum temperatures, maximum and minimum humidities and correlation between average pressure and average temperature have been illustrated after cleaning. In fact, this data visualization clearly gives some relationships, trends and behaviors of important quantities inside the data frame regarding Austin/Texas weather. Figures 4 to 9 show you the different graphs regarding exploratory data analysis.

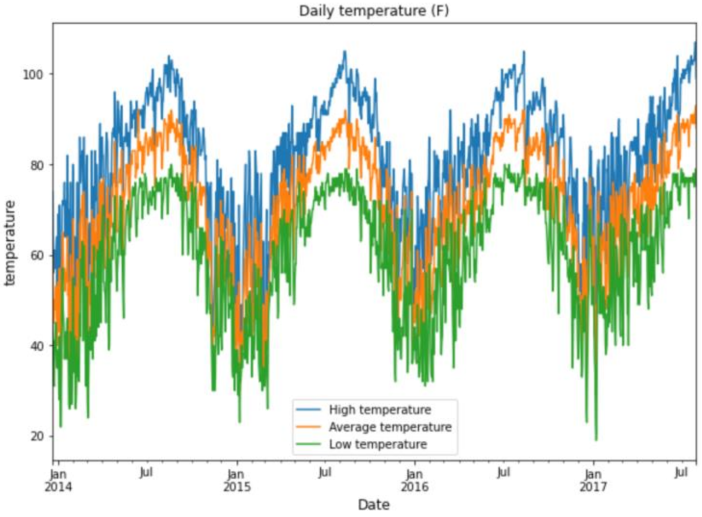


Figure 4-Illustration of high, average and low temperatures

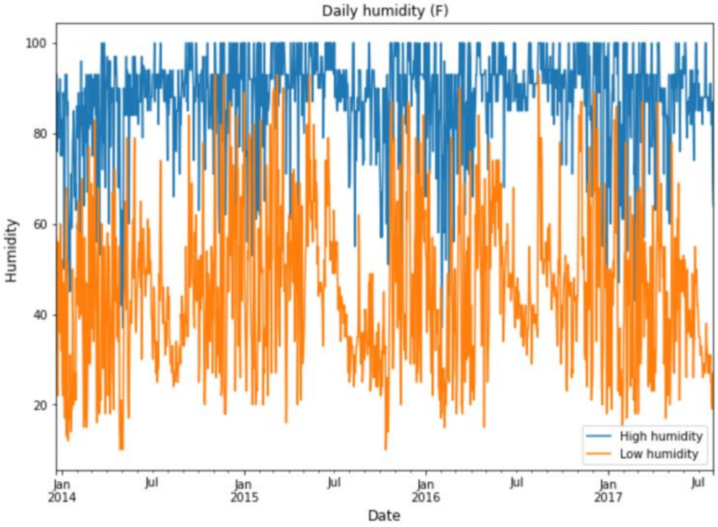


Figure 5-Illustration of high and low humidities

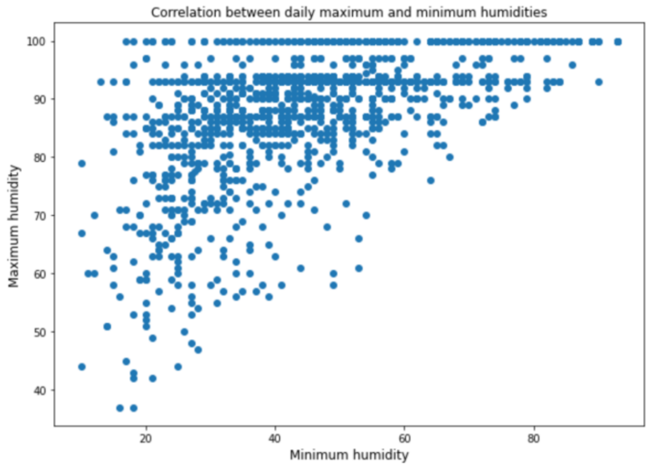


Figure6-Scatter plot between maximum and minimum humidities

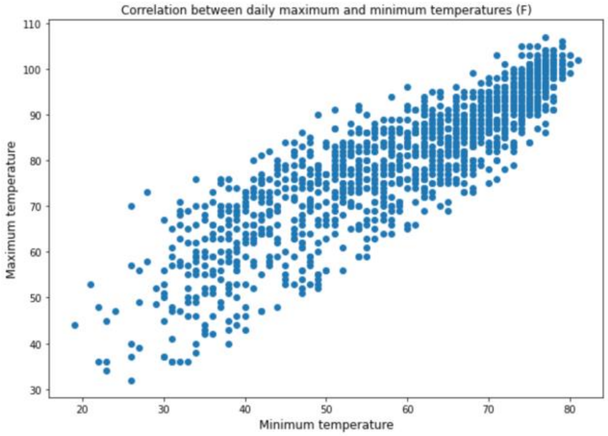


Figure7-Scatter plot between maximum and minimum temperatures

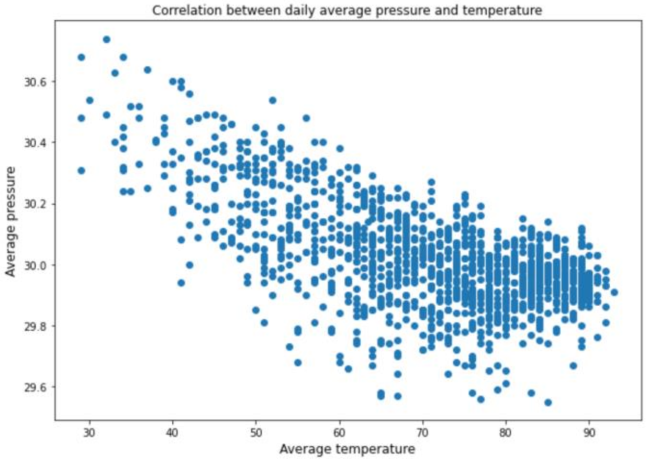


Figure 8-Scatter plot between average pressure and average temperature

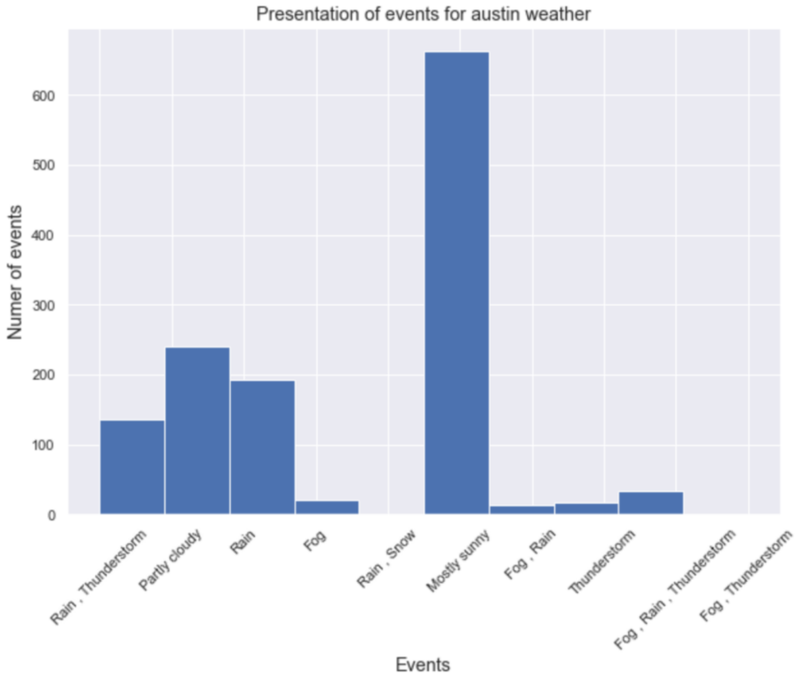


Figure 9-Events frequency for different weather status

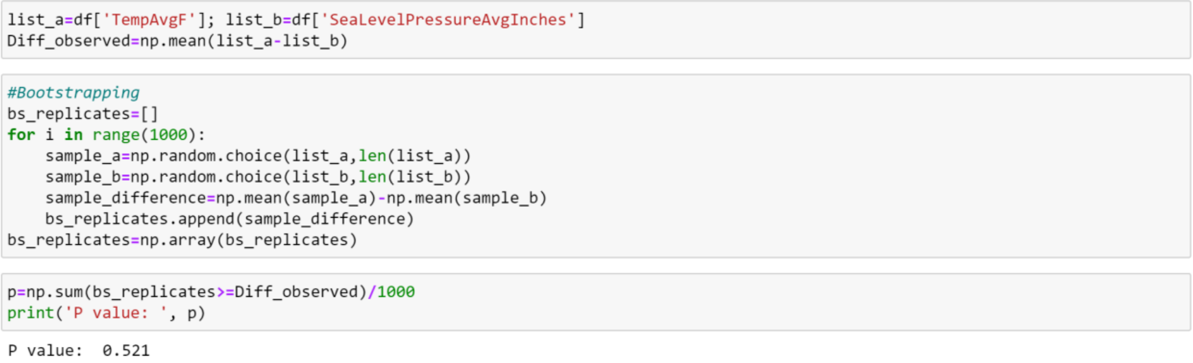
According to figure 6 related to correlation between maximum and minimum humidities for Austin weather; based on this scatter plot, it is concluded that the trend is approximately ascending. But it is observed that most collocation is in the upper area, so it is concluded that for most cases, maximum humidity is dominant factor. In figure 7 the ascending trend is observed which means if the maximum temperature increases the corresponding minimum temperature increases. But there are some points on the graph which have high maximum temperatures with corresponding low minimum temperatures. These points are unlike the general trend of the graph, so they can be outliers of the data. According to the figure 8, it is deduced that there is an inverse proportionality between average temperature and average pressure. Again, some outliers can be seen on this graph. In figure 9, events frequency for different weather status have been depicted by histogram. It is observed that most frequently events are mostly sunny, partly cloudy, rain and thunderstorm-rain respectively.

2-2-1-Test statistics and null hypothesis for average temperature and average pressure:

In this step, null hypothesis has been assessed for average temperature and pressure. For this purpose, observed difference between these two features was determined by making difference between mean value of pressure and mean value of temperature. So:

Observed difference=T_{mean}-P_{mean}

The next step, bootstraping for temperature and pressure data have been determined. So, the mean values for these two random choices have been calculated and making difference between these two mean values have been performed and it should be stored in a list. This procedure must be repeated for 1000 times. So, there is a list with 1000 elements which are mean values called boostraping replicates. Consequently, the number of elements from boostraping replicates which are higher than the observed difference have been found and it should be divided by the number of elements of this list which is1000. The probability of this selection from bootstraping replicates called P-value. Consequently, P-value can be determined and it is 0.5. Figure 10 shows you the code for null hypothesis.



```
list_a=df['TempAvgF']; list_b=df['SeaLevelPressureAvgInches']
Diff_observed=np.mean(list_a-list_b)
```

```
#Bootstrapping
bs_replicates=[]
for i in range(1000):
    sample_a=np.random.choice(list_a,len(list_a))
    sample_b=np.random.choice(list_b,len(list_b))
    sample_difference=np.mean(sample_a)-np.mean(sample_b)
    bs_replicates.append(sample_difference)
bs_replicates=np.array(bs_replicates)
```

```
p=np.sum(bs_replicates>=Diff_observed)/1000
print('P value: ', p)
```

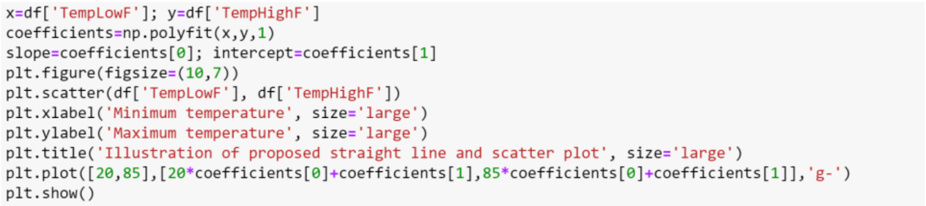
```
P value:  0.521
```

Figure 10-Applied code for null hypothesis for average pressure and average temperature

So, it is deduced that there is correlation between average temperature and pressure.

2-2-2-Proposing a straight line for correlation between maximum and minimum temperatures:

A linear regression analysis should be applied in order to suggest a straight line which shows the relationship between maximum and minimum temperatures. For this purpose, an equation of line with specific slope and intercept must be assigned. This analysis can be performed by polyfit() command. By running this method, it is possible to find the slope and intercept of the straight line between these two quantities. Figure 11 depicts the applied code and figure 12 displays the linear regression graph.



```
x=df['TempLowF']; y=df['TempHighF']
coefficients=np.polyfit(x,y,1)
slope=coefficients[0]; intercept=coefficients[1]
plt.figure(figsize=(10,7))
plt.scatter(df['TempLowF'], df['TempHighF'])
plt.xlabel('Minimum temperature', size='large')
plt.ylabel('Maximum temperature', size='large')
plt.title('Illustration of proposed straight line and scatter plot', size='large')
plt.plot([20,85],[20*coefficients[0]+coefficients[1],85*coefficients[0]+coefficients[1]],'g-')
plt.show()
```

Figure11- The code utilized for linear regression analysis

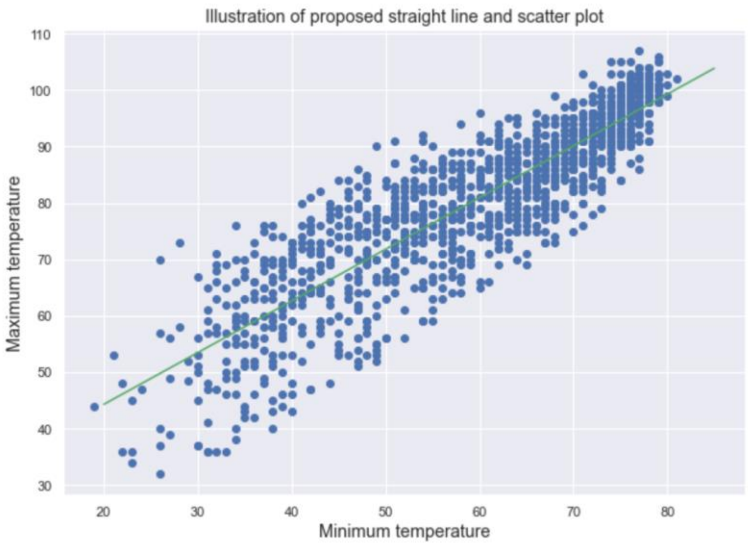


Figure 12-Linear regression analysis

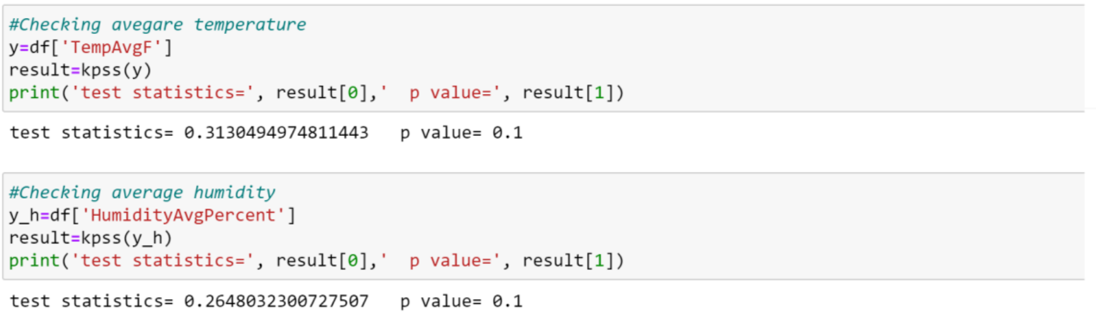
2-3-Modelling and performance evaluation:

Some functions for prediction of two key quantities (average temperature and average humidity) are utilized. In fact, these two parameters are predicted for last 21 days of July 2017 in Austi/Texas. Model performance based on Bayesian Information Criterion (BIC) is implemented to find the best case for prediction. As a matter of fact, two types of approach are used for this analysis:

-Forecasting by single feature

-Prediction by multiple features

Actually, these two approaches are applied to for both average temperature and average humidity. In addition, a test has to be performed for these quantities for stationarity. Consequently, KPSS method is used to check this condition. In other words, the result of the KPSS is the P-value for the null hypothesis that the series is stationary. If the P-value is less than 0.05, the null hypothesis can be rejected. Figure 13 gives the code for this test statistics.



```
#Checking avegare temperature
y=df['TempAvgF']
result=kpss(y)
print('test statistics=', result[0],'  p value=', result[1])
```

```
test statistics= 0.3130494974811443   p value= 0.1
```

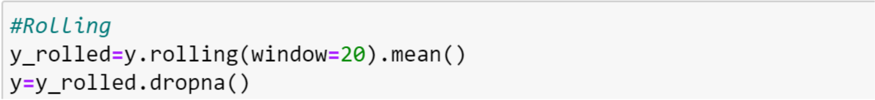
```
#Checking average humidity
y_h=df['HumidityAvgPercent']
result=kpss(y_h)
print('test statistics=', result[0],'  p value=', result[1])
```

```
test statistics= 0.2648032300727507   p value= 0.1
```

Figure 13-The code for checking the stationarity

According to the test, the P-values for both series are 0.1. It is higher than 0.05 and null hypothesis cannot be rejected, so it is concluded that both series are stationary. Consequently, it is possible to have prediction for both quantities.

In next step, the curves of temperature and humidity should be smooth by rolling command. So, by this method both graphs will be smooth. Figure 14 shows the python code for rolling. Figures 15 and 16 display the initial curve and smooth graph for temperature and humidity.



```
#Rolling
y_rolled=y.rolling(window=20).mean()
y=y_rolled.dropna()
```

Figure 14-The code for making curves smooth

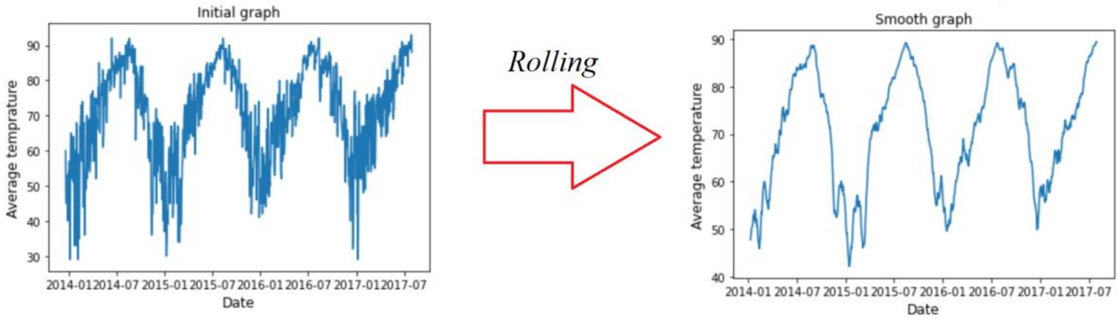


Figure 15-Rolling for average temperature curve

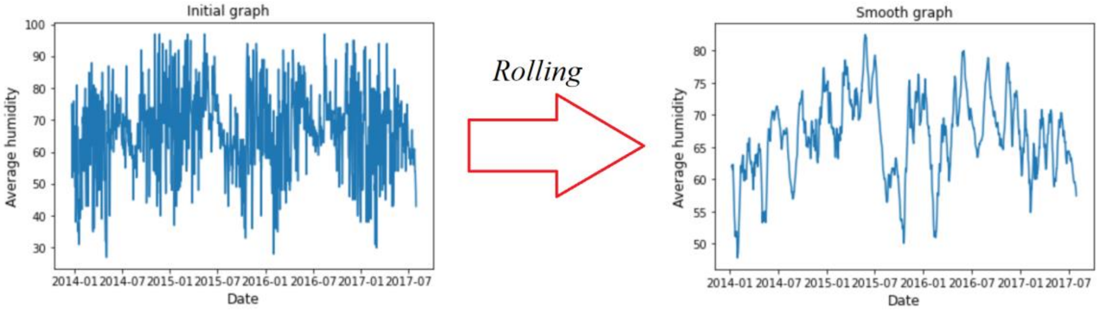


Figure 16-Rolling for average humidity curve

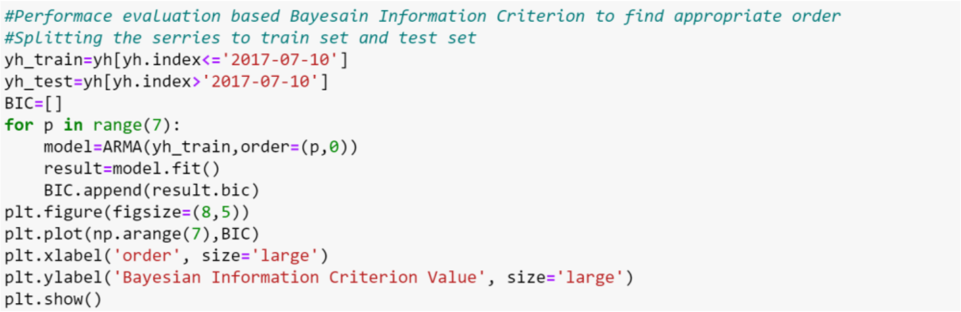
Now the curves are smooth and ready for machine learning. Therefore, we can apply different models and their assessment. After the performance evaluation for different models, it is possible to compare the results and make decision regarding the more precise model to predict these quantities. In addition, two sets extracted from the data frame for training and testing. Train set is used for fitting and test set is applied for prediction. Consequently, test set of the data frame includes the last 21 days of June 2017 for average temperature and average humidity.

2-3-1-Prediction of average temperature and humidity in Austin/Texas:

2-3-1-1-Forecasting by single feature:

2-3-1-1-1-Auto Regressive model (AR model):

According to this model, model performance should be applied to find the appropriate order based on Bayesian Information Criterion. Figure 17 shows the python code for this assessment.



```
#Performace evaluation based Bayesain Information Criterion to find appropriate order
#Splitting the serries to train set and test set
yh_train=yh[yh.index<='2017-07-10']
yh_test=yh[yh.index>'2017-07-10']
BIC=[]
for p in range(7):
    model=ARMA(yh_train,order=(p,0))
    result=model.fit()
    BIC.append(result.bic)
plt.figure(figsize=(8,5))
plt.plot(np.arange(7),BIC)
plt.xlabel('order', size='large')
plt.ylabel('Bayesian Information Criterion Value', size='large')
plt.show()
```

Figure 17-AR model performance evaluation

Based on this analysis, it is observed that p=2 has minimum BIC value which means it has the best prediction. So, this order should be used for AR model for function and forecasting. Figure 18 depicts the graphs about prediction of average temperature for last 21 days of June 2017. Similarly, for humidity p=2 has the best case for prediction by Auto Regressive model. Figure 19 illustrates the comparison between two curves.

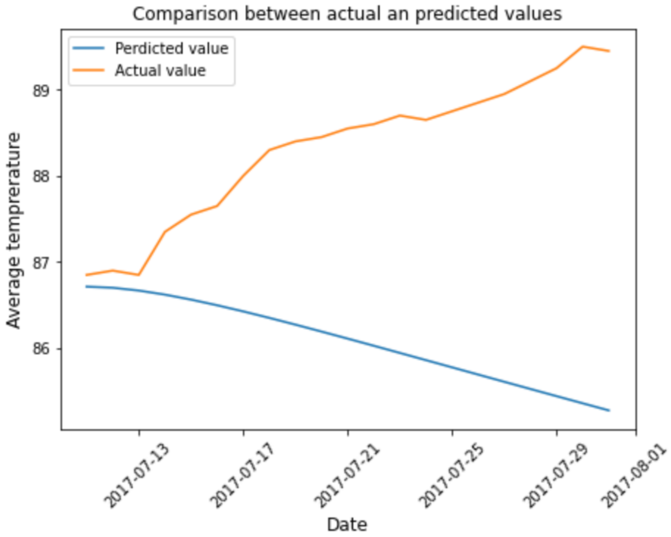


Figure 18-Forecasting of average temperature with AR model

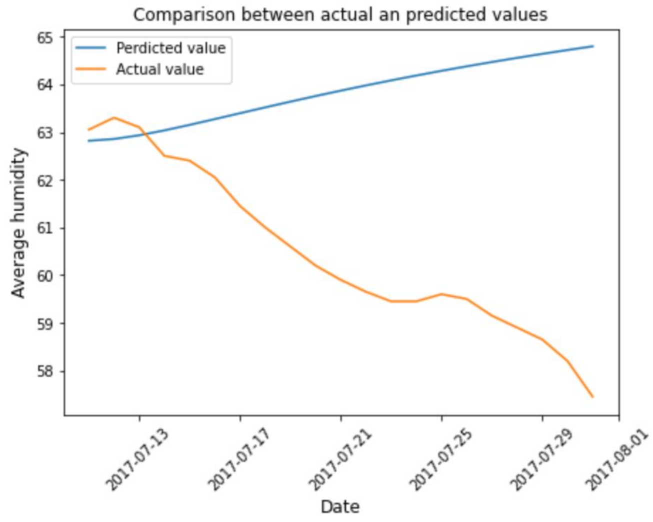
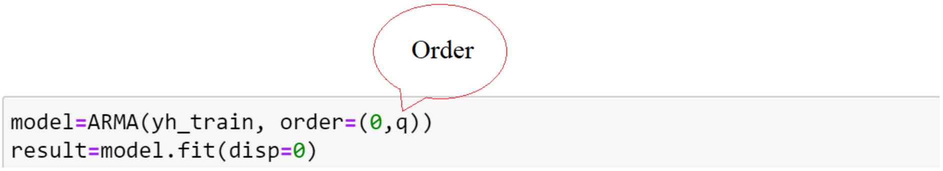


Figure 19-Forecasting the average humidity with second order of Auto Regressive model

Obviously, it is concluded that the prediction is not acceptable. Since, the predicted graph is ascending, on the contrary; the actual graph is descending. The predictions only for time less than '2017-07-13' are approximately good but for other moments there is no appropriate prediction. You can observe the trend the prediction of humidity in figure 18 as well. Again, there is much difference between two graphs except the dates less than '2017-07-14'

2-3-1-1-2- Moving Average model (MA model):

In this step, moving average model is applied. At first the model performance is required to find the best order related to q value. Figure 20 shows the code for this function and its order.



```
model=ARMA(yh_train, order=(0,q))
result=model.fit(disp=0)
```

Figure 20-The program code for MA model and its order

But according to this program code running, there is divergence for q>1 for both temperature and humidity. Consequently, there is no choice to apply this function unless if first order is assigned for MA model. Therefore, the first order of this function is run to observe the result. Figures 21 and 22 display the comparison between actual and predicted values.

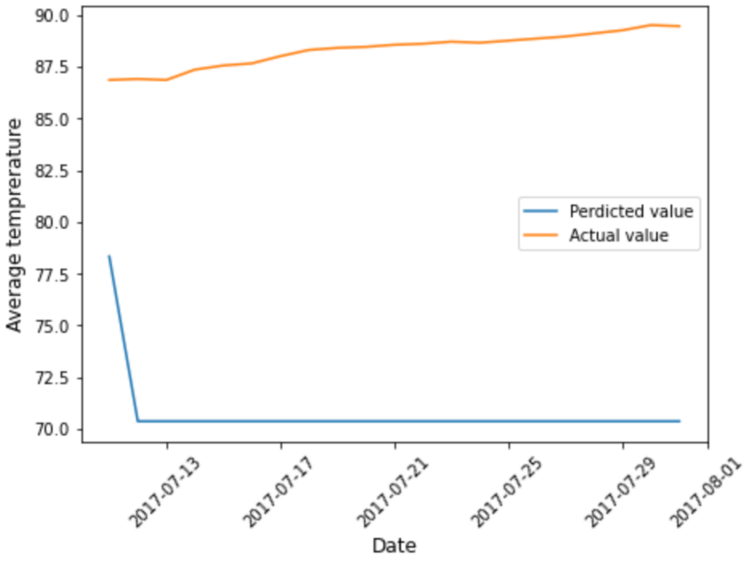


Figure 21-Average temperature prediction by first order of MA model

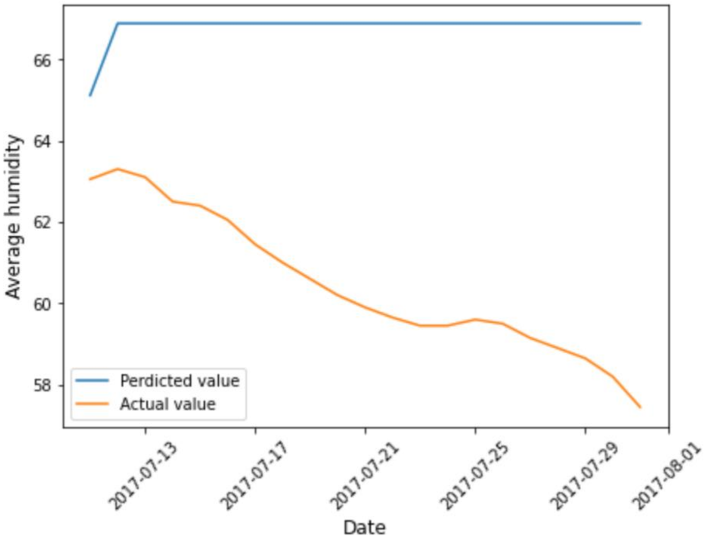


Figure 22-Average humidity prediction by first order of MA model

2-3-1-1-3- - ARMA model:

This is combination of AR and MA models. So, it has both p and q for function order. Therefore, performance evaluation should be implemented for p and q. According to this assessment for average temperature, it is concluded p=2 and q=3 have the best result for temperature, furthermore p=4 and q=4 are appropriate for average humidity. Figures 23 and 24 show you the results.

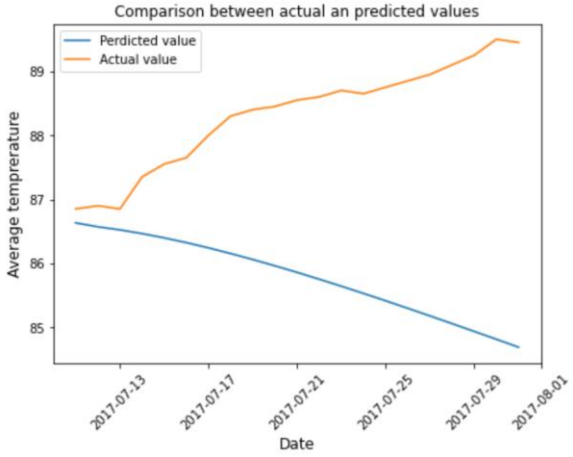


Figure 23-Average temperature prediction of ARMA model with order of p=2 and q=3

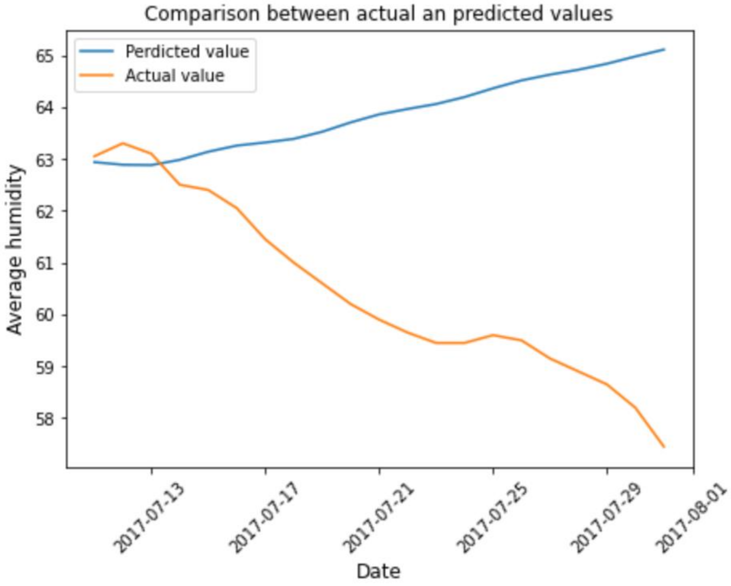


Figure 24-Average humidity prediction of ARMA model with order of p=4 and q=4

2-3-1-1-4- ARIMA model:

In this model, there are three parameters for the function which are p, d and q. For the performance assessment, three loops should be defined for this function to check the results based on Bayesian Information Criterion. By this assessment it is observed that p=5, d=1 and q=5 have minimum value which means best result. By the same method, it is concluded that p=6, d=1 and q=7 have most precise results. Figures 25 and 26 illustrate the comparison of actual and predicted values for average temperature and humidity.

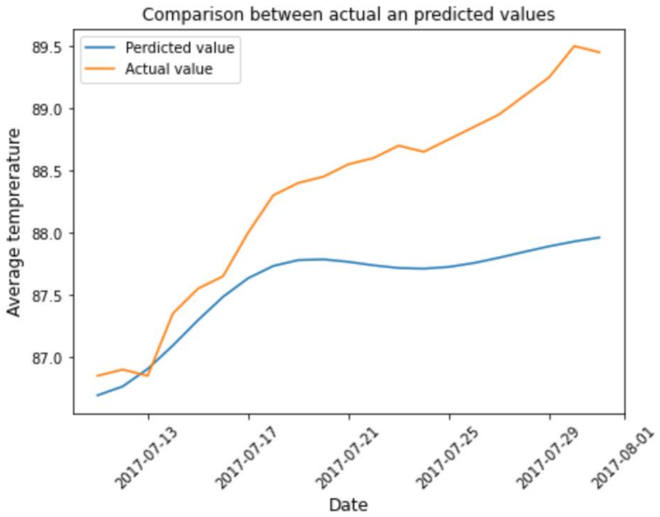


Figure 25-Comparison of two graphs based on ARIMA for temperature

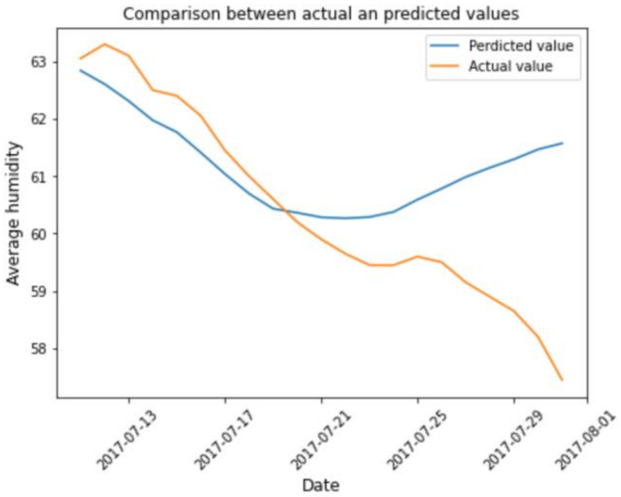


Figure 26-Comparison of two graphs based on ARIMA for humidity

According to the figures 25 and 64, it is concluded that the ARIMA model is much better rather than other functions. Figure 25 is the prediction for average temperature, it can be seen that forecasting related to times less than 2017-07-18 is acceptable, on the other hand; for time higher than 2017-07-18 there is no good agreement between two curves. Correspondingly, figure 26 is the prediction for average humidity and the result based on ARMA model has the best performance rather than previous functions. In addition, it is observed the forecasting of humidity less than 2017-07-24 is significantly acceptable. But, for times higher than 2017-07-24 the deviation between two graphs takes place.

2-3-1-2-Prediction by multiple features:

A function by multiple features for machine learning can be applied for prediction of weather conditions which depend on time. Therefore, it is possible to utilize a function which includes multiple fields for time series analysis. In this step, linear regression function is implemented for fitting and prediction. For this purpose, the data frame should be modified because it has categorical variables. In fact, the column of ‘Event’ includes some unique values which are categorical variables. These values are following strings:

-Rain, Thunderstorm

-Partly cloudy

-Rain

-Fog

-Rain, Snow

-Mostly sunny

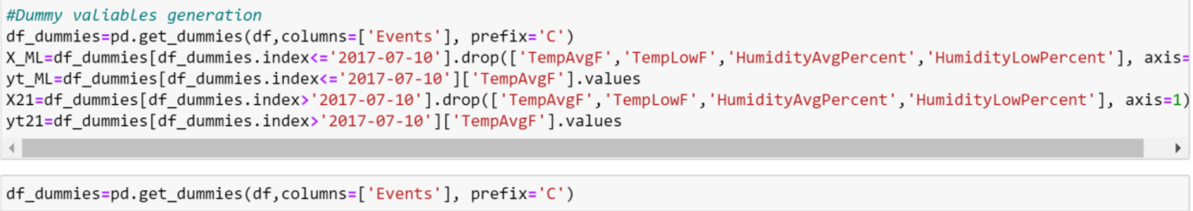
-Fog, Rain

-Thunderstorm

-Fog, Rain, Thunderstorm

-Fog, Thunderstorm

By get_dummies method, it is possible to convert categorical variables to dummy variables. Figure 27 shows the applied code for dummy variables generation.



```
#Dummy valiables generation
df_dummies=pd.get_dummies(df,columns=['Events'], prefix='C')
X_ML=df_dummies[df_dummies.index<='2017-07-10'].drop(['TempAvgF','TempLowF','HumidityAvgPercent','HumidityLowPercent'], axis=
yt_ML=df_dummies[df_dummies.index<='2017-07-10']['TempAvgF'].values
X21=df_dummies[df_dummies.index>'2017-07-10'].drop(['TempAvgF','TempLowF','HumidityAvgPercent','HumidityLowPercent'], axis=1)
yt21=df_dummies[df_dummies.index>'2017-07-10']['TempAvgF'].values
```

```
df_dummies=pd.get_dummies(df,columns=['Events'], prefix='C')
```

Figure 27-Getting dummy variables and splitting the data frame to two sets

So, there are 29 columns after getting dummy variables. In next step, two columns of average temperature and average humidity should be removed from the dataset. Because they are response variables for making prediction. Consequently, the data set can be managed based on feature engineering. Afterwards, Linear Regression function which uses multiple features is utilized for fitting and prediction. Therefore, daily temperature and humidity of Austin/Texas are predicted. Figures 28 and 29 show the predictions for these key quantities.

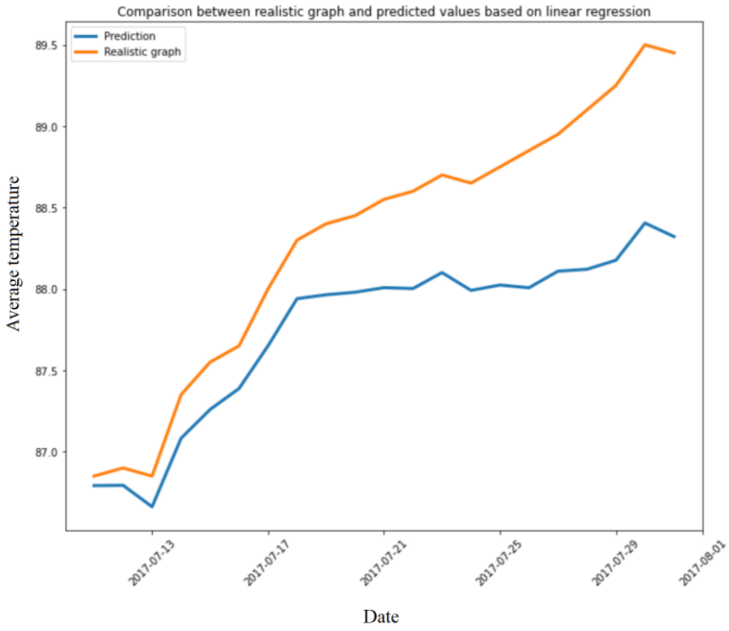


Figure28-Prediction of average temperature by Linear Regression model

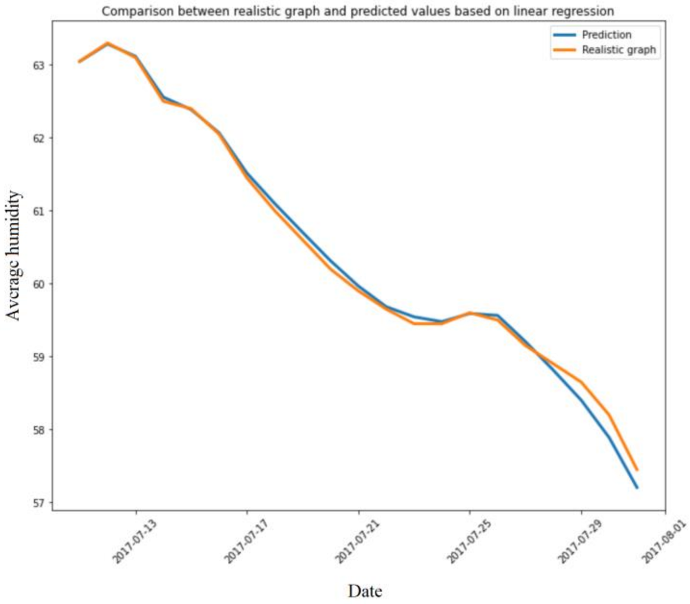


Figure29-Prediction of average humidity by Linear Regression model

So, it is clearly observed that the results of this function are the best among all of models. Consequently, modelling by multiple features is more efficient with respect to models which are utilizing single feature. Therefore, the results by Linear Regression function are more precise for prediction.

3-Conclusion

In this project, the statistical analysis for a data set regarding Austin/Texas weather was performed. After data wrangling by replacing the blanks and meaningless signs with interpolation; visualization and test statistics were accomplished. Based on data visualization, it is observed that trend between daily maximum and minimum temperatures is ascending which means most of the times for a specific day, if there is a higher maximum temperature there should be a higher minimum temperature although there are some outliers on this graph. On the other hand; the correlation for average pressure and temperature is descending. In other words; for specific date if average temperature increases, the average pressure will decrease at that moment. The next finding is related to weather events, according to histogram plot it can be seen the most frequent events related to weather status are mostly sunny, partly cloudy, rain and thunderstorm-rain respectively. In addition, null hypothesis was assessed regarding the correlation between average temperature and average pressure. P-value from this performed test is 0.1 and it is concluded there is a correlation between these quantities. In machine learning science for this project, five functions were implemented to predict average temperature and pressure. Four functions which included AR, MA, ARMA, ARIMA used single feature. Performance evaluation was applied for these four functions based on Bayesian Information Criterion (BIC). According to this assessment, ARIMA model is most precise rather than other three functions. In addition, LinearRegression model which utilized multiple features from the data frame was applied as fifth function in order to predict these quantities. According to the observed results, it is concluded that the LinearRegression model is more precise than ARIMA method. In other words, LinearRegression model has the best results among these functions. According the obtained results by fifth model, it is deduced the average temperature for last 21 days of June 2017 is ascending. On the other hand; average pressure graph is descending and there is very good agreement between actual values and predicted values. Therefore, it is possible to conclude that the modelling with multiple features is more efficient rather than modelling with single feature. If number of rows for the dataset is much more than the current value (1319), probably modelling with single feature has better results but it is not possible to claim it with certainty because there is no guaranty for the models using single features.

Recommendations for new improvements

According to this analysis, it is recommended to find a dataset of Austin/Texas weather which has much more rows like 5000 or more to check the model performance of functions which are using single feature. There is another model called SARIMAX which can be used for making prediction. In fact, this model is using single field. So, using SARIMAX is next recommendation to continue this project. In addition, model performance of this function should be implemented to find appropriate order for prediction. The next recommendation is about average pressure of data frame. So, it is good idea to focus on average pressure and apply some functions to forecast this quantity.