Submitted by: Indira Priyadarsini J , (PhD Scholar-HCU)

Ph no: 8309262338, indira.jagiripu@gmail.com

**DATA ANALYSIS ON DATASET\_Null Innovations**

**Step 1** : Loading the required libraries (ex Pandas, etc) and then reading the csv file into a variable

**Observation**: we have a mix of nominal, ration and interval data. Month, dayofthemonth, Day\_ofthe\_week are ordinal values and the rest can be ration/interval values. (Ex : temperature is a ratio scale)

**Step 2**: Checking for missing data, if any :

print(dataset.isnull().sum())

Month 0

Day\_of\_month 0

Day\_of\_week 0

ozone\_reading 5

pressure\_height 0

Wind\_speed 0

Humidity 15

Temperature\_Sandburg 2

Temperature\_ElMonte 139

Inversion\_base\_height 15

Pressure\_gradient 1

Inversion\_temperature 14

Visibility 0

Unnamed: 13 366

Unnamed: 14 366

Unnamed: 15 361

Unnamed: 16 362

Unnamed: 17 366

Unnamed: 18 366

Unnamed: 19 366

Unnamed: 20 366

Unnamed: 21 366

dtype: int64

**Observation** : It can been seen that the dataset has missing values.

**Step 3** : % of missing data in each label

miss\_val\*100/len(dataset)

Month 0.000000

Day\_of\_month 0.000000

Day\_of\_week 0.000000

ozone\_reading 1.366120

pressure\_height 0.000000

Wind\_speed 0.000000

Humidity 4.098361

Temperature\_Sandburg 0.546448

Temperature\_ElMonte 37.978142

Inversion\_base\_height 4.098361

Pressure\_gradient 0.273224

Inversion\_temperature 3.825137

Visibility 0.000000

Unnamed: 13 100.000000

Unnamed: 14 100.000000

Unnamed: 15 98.633880

Unnamed: 16 98.907104

Unnamed: 17 100.000000

Unnamed: 18 100.000000

Unnamed: 19 100.000000

Unnamed: 20 100.000000

Unnamed: 21 100.000000

dtype: float64

**Observation** : It is seen that the rows which name start with Unnamed 13-21 have 100% missing values. So its better to drop those from the dataset. The rest which has certain percentage of missing values (ex : Ozone\_reading =1%, Humidity = 4%) have to be treated for various missing value treatments.

**Step 4**: Conversion of few row labels into datatype int, for simplification of the further analysis.

**Observation** : It is observed that few rows like pressure\_height, wind\_speed are integer datatypes and few rows like Humidity, Ozone\_reading are float datatypes. All rows which have certain percentage of missing values are converted to integer datatype. (Actually not necessary to convert the datatypes, but it gave an error that cnt impute mean to float for a particular instance. So converted them to integer for smooth execution).

**Step 5**: Imputing all the qualitative rows with mean.( Dealing with missing data)

dataset1['Humidity'] = dataset1['Humidity'].fillna(dataset1.Humidity.mean()).astype(np.int64)

dataset1.Humidity.isnull().sum()

0

dataset1.isnull().sum()

Month 0

Day\_of\_month 0

Day\_of\_week 0

ozone\_reading 0

pressure\_height 0

Wind\_speed 0

Humidity 0

Temperature\_Sandburg 0

Temperature\_ElMonte 0

Inversion\_base\_height 0

Pressure\_gradient 0

Inversion\_temperature 0

Visibility 0

**Observation** : Since all the missing value variables are in interval scale , imputation with mean is possible. But in cases with categorical variables , missing values are treated in other ways like using KNN means , filling with the nearest possible value, etc. For metric values we can use other methods like imputing with median, mode, and also the other methods mentioned for categorical variables depending on the values of the data. A dataset eliminated of missing values is generated.

**Step 6** : Exploring the data . The descriptive statistics of the dataset after cleaning the dataset is generated.

dataset1.describe()

Month Day\_of\_month Day\_of\_week ozone\_reading pressure\_height Wind\_speed Humidity Temperature\_Sandburg Temperature\_ElMonte Inversion\_base\_height Pressure\_gradient Inversion\_temperature Visibility

count 366.000000 366.000000 366.000000 366.000000 366.000000 366.000000 366.000000 366.000000 366.000000 366.000000 366.000000 352.000000 366.000000

mean 6.513661 15.756831 4.002732 11.016393 5764.316940 4.868852 58.456284 61.909836 56.232240 2590.904372 17.795082 60.927330 123.300546

std 3.455958 8.823592 1.997942 7.840167 85.149011 2.116928 19.349192 14.237518 9.174303 1759.520761 36.061045 13.871084 80.280142

min 1.000000 1.000000 1.000000 0.000000 5510.000000 0.000000 19.000000 25.000000 27.000000 111.000000 -69.000000 27.500000 0.000000

25% 4.000000 8.000000 2.000000 4.250000 5710.000000 3.000000 50.000000 51.250000 54.000000 899.000000 -9.750000 51.260000 70.000000

50% 7.000000 16.000000 4.000000 9.000000 5770.000000 5.000000 64.000000 62.000000 56.000000 2380.000000 24.000000 62.240000 110.000000

75% 9.750000 23.000000 6.000000 16.000000 5830.000000 6.000000 73.000000 72.000000 59.750000 5000.000000 45.000000 70.520000 150.000000

max 12.000000 31.000000 7.000000 37.000000 5950.000000 11.000000 93.000000 93.000000 82.000000 5000.000000 107.000000 91.760000 500.000000

**Observation** : The mean, standard deviation(SD), quartiles , minimum and maximum values of each row are generated. The SD is a measure of spread of data from the mean. A low SD = numbers are close to average. High SD = numbers are spread out.

Pressure\_gradient is the row with high SD, data with high variation. Day of the week, etc are those with less SD, data with low variation.

(Detecting outliers using descriptive statistics is explained in Outliers detection part(step 8))

**Step 7 :**Visual exploration of the data . Generating graphs.

**Graph 1** : A scatter plot for Temperature\_ElMonte and Temperature\_Sandburg.

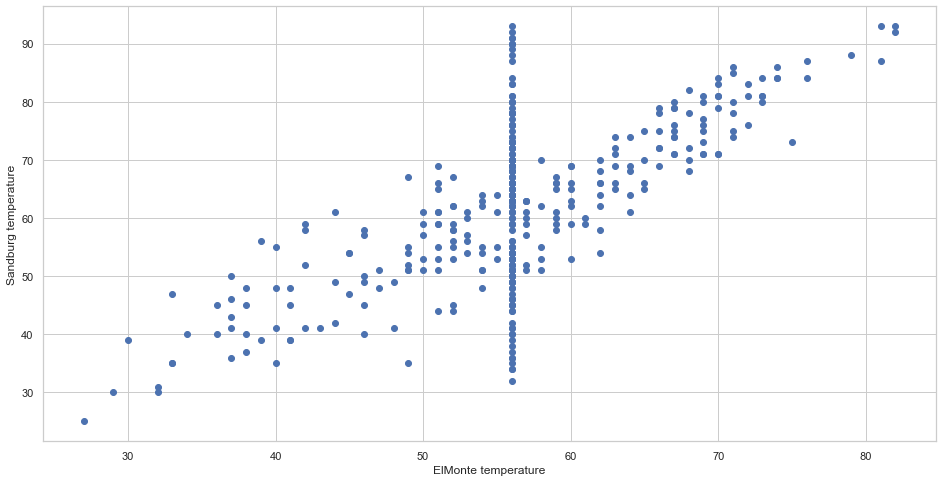
fig, ax = plt.subplots(figsize=(16,8))

ax.scatter(dataset1['Temperature\_ElMonte'], dataset1['Temperature\_Sandburg'])

ax.set\_ylabel('Sandburg temperature')

ax.set\_xlabel('ElMonte temperature')

plt.show()

**Observation** : The temperatures of both the cities Sandburg and Elmonte are positively corelated. It can be inferred that both the cities have similar climatic conditions. Or also the distance between both the cities might be less. The straight line formed in between 50-60 label on X-axis can be noise. It is desirable to remove the noise in a data for better results during model development.

**Graph 2**: A boxplot for the Wind\_speed data

import matplotlib.pyplot as plt

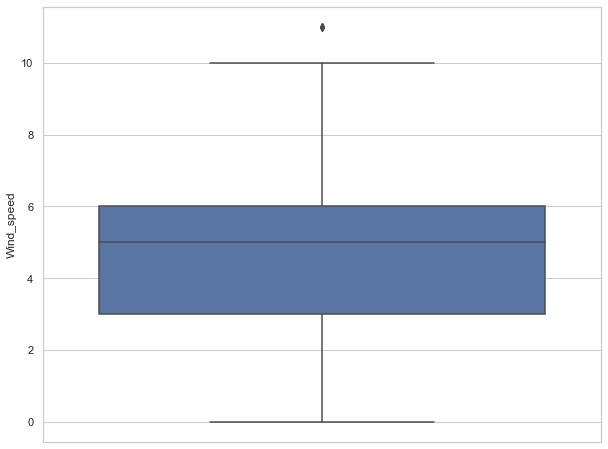
import seaborn as sns

%matplotlib inline

sns.set(style="whitegrid")

plt.figure(figsize=(10,8))

ax = sns.boxplot(x='Wind\_speed',data=dataset1, orient="v")



**Observation**: A box plot is a visual presentation of the descriptive statistics of a data.

Here 10 is the maximum value , 0 is the minimum value in the data. So range of the data is (10-0) = 10. Median is around 5. An outlier is observed one above the maximum number. It is better to exclude the outliers, because they might effect the final results. The more far away the outliers are from the median and more the number of outliers, the accuracy of the model will reduce. The outlier here at around 11 might be because of the measurement/data recording error. Inconsistency in weather conditions also can be the reason for that outlier.

**Step 8**: Detecting and removal of outliers from the ozone\_reading data.

import matplotlib.pyplot as plt

import seaborn as sns

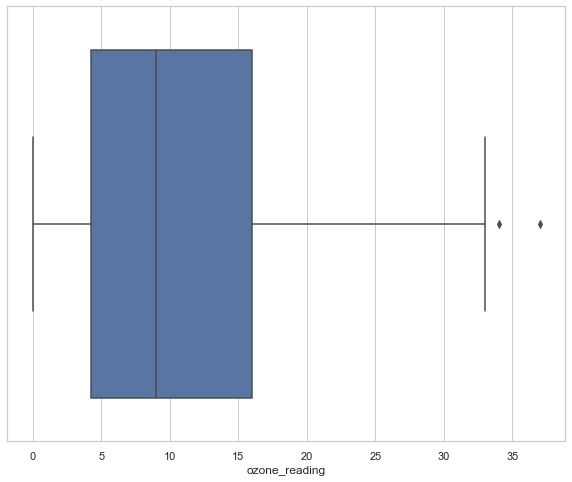
%matplotlib inline

sns.set(style="whitegrid")

plt.figure(figsize=(10,8))

ax = sns.boxplot(x='ozone\_reading',data=dataset1, orient="h")

**Graph 3.1** : A box plot for ozone\_reading with outliers



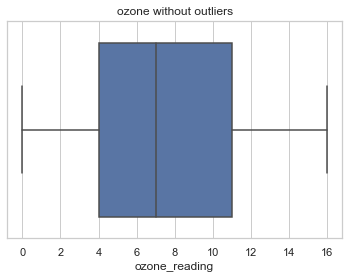
**Observation** : It is visible that there are two outliers for the ozone\_reading data. Reasons can be as mention for the wind\_speed data. Since outliers are better to be excluded from the data seet, lets exclude them and check again with another box plot for ozone\_reading.

dataset1=dataset1[(dataset1.ozone\_reading <= 16)]

dataset1.shape

sns.boxplot(dataset1['ozone\_reading']).set\_title("ozone without outliers")

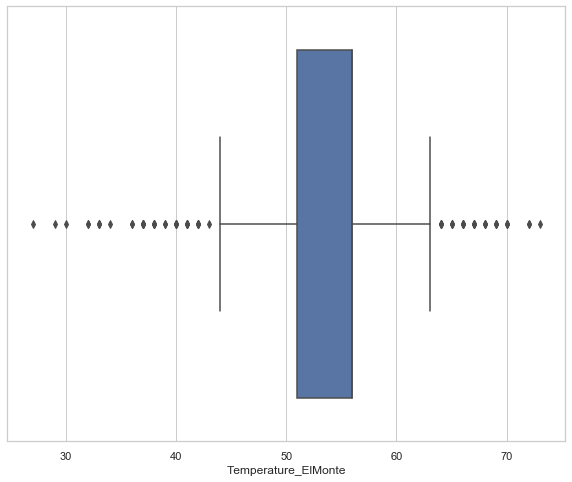
**Graph 3.2** : box plot for ozone\_reding without outliers.



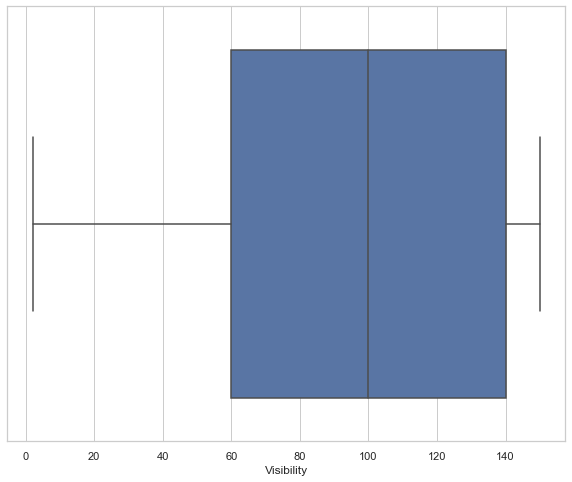
**Observation** : its visible that there are no more outliers in the ozone\_ reading. In the code it is mentioned that the values <=16 should be considered as outliers. If we observe the descriptive statistics table above, we can infer from it directly weather there are outliers or not. If Q1 value >= minimum value OR if Q3 is less than maximum values that means the data has outliers. The maximum value for ozone reading is 36 and the Q3 is 16.

The process is repeated for other rows also. Only the before after graphs are posted which are as follows.

**Graph 4.1** : box plot for Visibility with outliers



**Graph 4.2** : Box plot of Visibility without Outliers



**Step 9** : Exploring data of other rows

fig, ax = plt.subplots(figsize=(16,8))

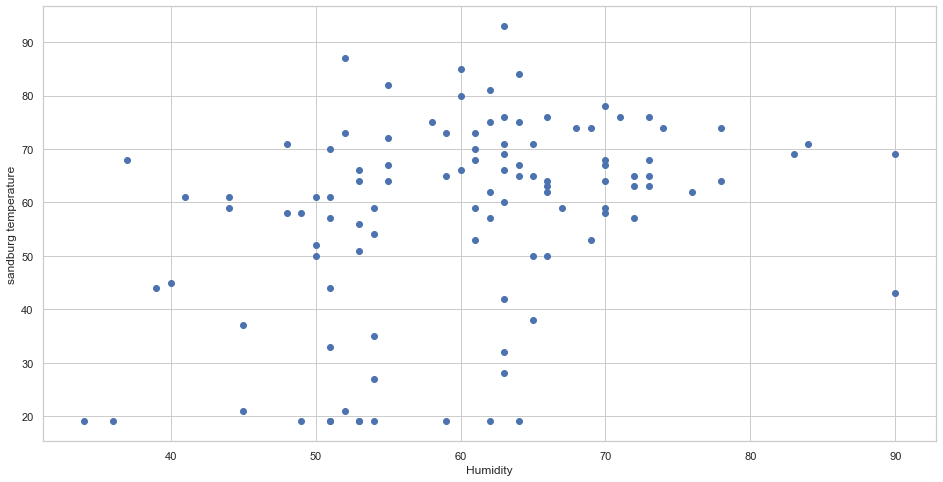
ax.scatter(dataset1['Temperature\_ElMonte'], dataset1['Humidity'])

ax.set\_ylabel('Sandburg temperature')

ax.set\_xlabel('Humidity')

plt.show()

**Graph 4.1** : scatter plot for Temperature\_Sandberg vs Humidity



**Graph 4.2** : Scatter plot for Temperature\_ElMonte and Humidity

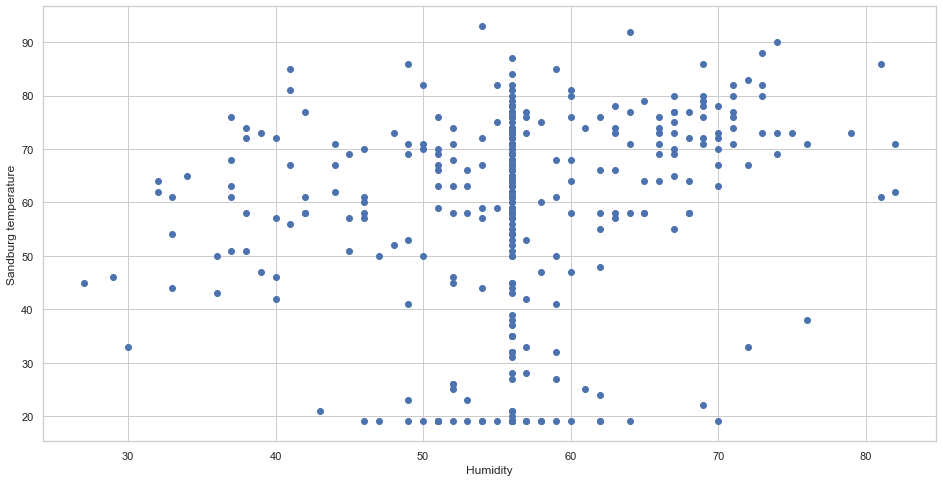
fig, ax = plt.subplots(figsize=(16,8))

ax.scatter(dataset1['Temperature\_ElMonte'], dataset1['Humidity'])

ax.set\_ylabel('Elmonte temperature')

ax.set\_xlabel('Humidity')

plt.show()



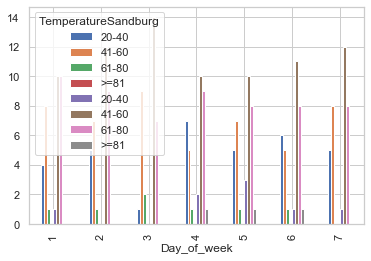
**Observation** : The data is dispersed and there doesn’t seem to have any relation. The reason is as follows , stated from the literature review of scientific studies conducted on temperature and humidity.

“ It has been known that the exceptionally low temperatures at the tropopause constrain the amount of water entering the stratosphere. Most of the stratosphere is subsaturated due the radiative effects of ozone leading to temperatures higher than at the tropical tropopause.” [Fueglistaler, S., Liu, Y. S., Flannaghan, T. J., Haynes, P. H., Dee, D. P., Read, W. J., ... & Bernath, P. F. (2013). The relation between atmospheric humidity and temperature trends for stratospheric water. *Journal of Geophysical Research: Atmospheres*, *118*(2), 1052-1074.]

It means because of effects of ozone decreasing, temperature and humidity are no more inversely related. The readings change according to the impact of global warning might be.

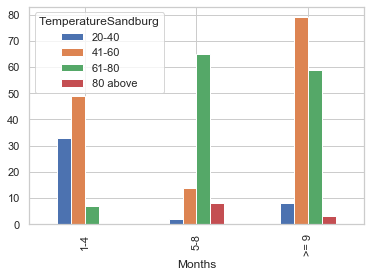
**Graph 5** : Graph for temperature\_sandberg distribution during the day of the week

Here the temperatures are divided into frequencies sets 0f interval 20 starting from 20 to above 81.



**Observation** : At any day of the week the temperature of >=81 followed by 60-80 seems to be occurring more as compared to other temperature frequencies. So the usual temperature would be something more than 60 most of the times. (Units of the variables not mentioned in the given dataset). So assuming it to be farenheit.

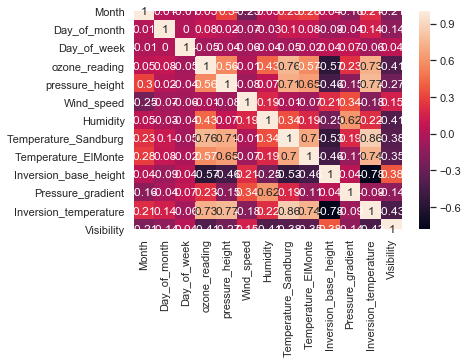
**Graph 6** : Graph for Temperature\_Sandberg vs month



**Observation** : It is observed that Sanberg has temperatures 20-40 in the first quarter ,increase in the 2nd quarter to 61-80 and in the 3rd quarter again slightly decreases to 41-60 (Most of the month frequency)/ Since the data follows the usual weather patter , we can ascertain that the data can be true.

**Step 10**: Finding the co -relation between all the variables (initial step of model development)

**GRAPH 7**: Co-relation matrix of the given dataset after addressing the missing values and eliminating the outliers.

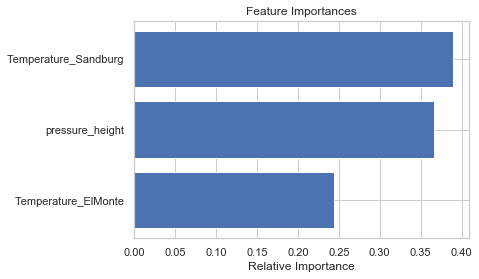


**Observation** : Inversion\_temperature seems to have some co-relation with most the other variables. So if we go for some prediction, Inversion\_temperature can be considered as the dependent variable and Inversion\_base\_height ,Temperature\_ElMonte, Temperature\_Sandberg, Pressure\_height, ozone\_reading can be considered as independent variables.

Co-relation matrix depicts only the relation between the attributes. Above 0.7 can be considered positive co-relation and as the correlation value reduces , so does the relation.

**Step 11**: A Random forest method is used to predict the ozone readings. But due to the lack of feature selection and descriptions of the variable, the output generated is not much accurate.. But the code can be referred for implementation of the algorithm and coding skills.

**Graph 8** : Bar graph relative importance of various features



The above graph explains the importance of the features Temperatures of Sandburg, Elmonthe and Pressure height on the Ozone readings. It is visible that Temperature of Sandburg has more importance than Elmonte on the Ozone readings. This is a visible proof that the model performance is not good. This happened because the features are selected based on their corelations with the dependent variable (ozone\_readings) rather than a appropriate mathematical procedure of feature selection.

Since model development is not the main aim of the assignment, it is done for the sake of implementation and awareness purpose.