DANA Assignment – 1

1. Introduction:

* This is the report based on Ulaanbaatar city situated in Mongolia. This data set consisted of 26665 rows and 14 columns. It has 4 years of data from 2018 to 2021 which has Raw\_Conc, NowCast\_Conc & AQI as the main column for analysis. It also has measured data time columns like Date\_LT, month, year, day, hour and other categorical columns like AQI\_Category. The main objective is to clean the data and replace -999 values with its true values and then calculate AQI\_Category using the table mentioned in the report pdf. After that the outliers can be deduced and cleaned from data to get the output file. From there the graphs can be plotted to get more insights which is listed below in the later sections.

1. Categorical vs Numerical:

* These are subdivided below further with Nominal, Ordinal, Interval and Ratio with the table below. The further explanation can be found below the table.

Table: Categories division with contents vs data columns.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Contents/Categories** | **Categorical** | **Numerical** | **Nominal** | **Ordinal** | **Interval** | **Ratio** | **Dtype** |
| **Site** | Yes | No | Yes | No | No | No | object |
| **Parameter** | Yes | No | Yes | No | No | No | object |
| **Date\_LT** | Yes | No | No | No | Yes | No | Datetime[ns] |
| **Year** | Yes | No | No | No | Yes | No | Int64 |
| **Month** | Yes | No | No | No | Yes | No | Int64 |
| **Day** | Yes | No | No | No | Yes | No | Int64 |
| **Hour** | Yes | No | No | No | Yes | No | Int64 |
| **NowCast\_Conc** | No | Yes | No | No | No | No | Float64 |
| **AQI** | No | Yes | No | No | No | No | Int64 |
| **AQI\_Category** | Yes | No | No | Yes | No | No | object |
| **Raw\_Conc** | No | Yes | No | No | No | No | Int64 |
| **Conc\_Unit** | Yes | No | No | No | No | No | object |
| **Duration** | Yes | No | No | No | No | No | object |
| **QC\_Name** | Yes | No | Yes | No | No | No | object |

1. **Site** -> As Site variable just has one information and has no numbers associated with it, it is Categorical type. As it has only one label variable with no ranks, it is Nominal variable type. It has its data type as object.
2. **Parameter**-> As Parameter has one value in entire column with no numbers, it is Categorical. This has one label with no further subdivision or ranking and hence this is further classified as Nominal. It has a dtype as object.
3. **Date\_LT**-> This is a date time variable with equal intervals throughout and even though it’s numbers it has categories associated with it and hence it is Categorical type further classified to Interval variable. It has dtype as datetime64[ns].
4. **Year**-> This one has a number with values 2018, 2019, 2020 & 2021 which can be categorized through data and hence this is Categorical variable type. These categories can be ranked in equal intervals and hence this is falls under Interval type. It has a dtype as int64.
5. **Month**-> Month column has uniquely 1 to 12 as the values and can be classified as fixed or in equal intervals and hence this is Categorical data with uniquely categorized intervals as Interval variable type. It has a data type as int64.
6. **Day**-> This column has uniquely 1 to 31 as the values and can be classified as fixed or in equal intervals and hence this is Categorical data with uniquely categorized intervals as Interval variable type. It has a data type as int64.
7. **Hour**-> Hour column has uniquely 0 to 23, and even though a few data is not consistent, it has a pattern of consistency throughout the data. Hence this is Categorical with further classified as Interval type as it has fixed timeframe and is consistent. It has dtype as int64.
8. **NowCast\_Conc**-> NowCast\_Conc has continuous values with almost no order and hence this is Numerical variable type. It has a dtype as float64.
9. **AQI**-> This is continuous numerical variable type with no categories or ranks and hence will not fall into Nominal, Ordinal, Interval or Ratio type. This has a dtype as int64.
10. **AQI\_Category**-> This column has no numbers and hence this is Categorical variable type. On further analysis, it has ranks categories in order, meaning if the value of AQI is less than or equal to 50 this will be categorized as “Good”, and hence this is Ordinal variable type as well. This has a dtype as object.
11. **Raw\_conc**-> This column has continuous values and hence this is Numerical variable type. It has dtype as int64.
12. **Conc\_Unit**-> As Conc\_Unit has one value in entire column with no numbers, it is Categorical. This has one label with no further subdivision or ranking and hence this is further classified as Nominal. It has a dtype as object.
13. **Duration**-> As Duration has one value in entire column with no numbers, it is Categorical. This has one label with no further subdivision or ranking and hence this is further classified as Nominal. It has a dtype as object.
14. **QC\_Name**: QC\_Name column has no numbers and hence this is Categorical. While observing it has few categories and has certain label variables with almost no ranks and hence this is Nominal variable type. It has object data type.
15. Accuracy:
16. The min & max values are already included in the python code file. Here is the snapshot.

A screenshot of a computer program

Description automatically generated with low confidence

Snapshot for data types:

A screenshot of a computer

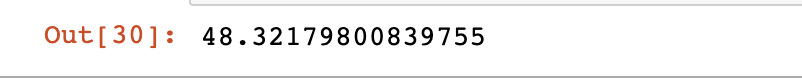
Description automatically generated

1. For null values I’ve removed -999 values throughout to use the algorithm ([*link*](https://usepa.servicenowservices.com/airnow?id=kb_article_view&sys_id=bb8b65ef1b06bc10028420eae54bcb98&spa=1)). Hence my null values look like below (*removed* -*999 from NowCast\_Conc, AQI, Raw\_Conc*). For 217 rows from Raw\_Conc (*in data I observed that there are 217 rows for which the QC\_Name is missing and has all other values in row*) which has AQI & AQI category values assigned, I replaced this null values back to -999 later.

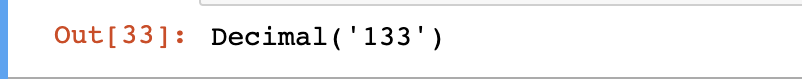
A screenshot of a computer code

Description automatically generated with low confidence

1. Using the Algorithm ([*link*](https://usepa.servicenowservices.com/airnow?id=kb_article_view&sys_id=bb8b65ef1b06bc10028420eae54bcb98&spa=1)), I used Raw\_conc values and then calculated AQI & AQI category and will be explained further in Missing values section. (*The code and formula are in the jupyter notebook file*)
2. The Algorithm was based on fixing the accuracy of NowCast\_Conc from Raw\_Conc and eventually getting its subsequent AQI values.
3. It was a formula to measure the 12-hour reading and then getting the range of those recorded values to generate a cumulative summation of the readings. First, we needed to get the range which was min subtracted from the max value in the 12-hour frame. Then the weight factor was calculated using the range which was then subtracted from 1 and its value ranged from 0.5 to 1. If the value was less than 0.5, we needed to replace it with 0.5 to get at least 0.5 as the weight factor.
4. This weight factor was then raised to the hour before it was measured, multiplied with the Raw\_Conc value and then the summation of all the readings was then added to get the step 6. This was then divided by the weight factor summation raised to the number of hours it was calculated.
5. This value is the true value which was then replaced with null/-999 cell value throughout the data for NowCast\_Conc.



1. Now as I have NowCast\_Conc, I calculated AQI using the python package python-aqi ([*link*](https://pypi.org/project/python-aqi/)). I researched and was not getting the formula to convert directly the value of NowCast\_Conc to AQI and that’s when I got this package in python.



1. From here it was easy to calculate the AQI category which can be discussed in the “Missing Value” section.
2. Missing data:
3. For the missing data, I observed that AQI\_Category values are missing from the file which was needed to get updated. The AQI\_Category is based on AQI column, as mentioned in the report pdf file and hence AQI was calculated. I used NowCast\_Conc values for AQI, which was calculated using Raw\_Conc.
4. I used below code and the blank data was updated in the file. Below is the snapshot:

A screenshot of a data

Description automatically generated with low confidence

1. After this, the null values just remained for Raw\_Conc (*as mentioned earlier I observed 217 rows with -999 which doesn’t need to change as AQI has value*).

A screenshot of a computer program

Description automatically generated with low confidence

1. Outliers:
2. For AQI:

* Here is the scatterplot for AQI with year as x-axis and AQI as y-axis. This has a legend with AQI\_Category.

A picture containing text, screenshot, diagram, font

Description automatically generated

* Here it can be seen that after 400 there is a trend of outliers but it’s hard to get the sum from just scatter plot. Hence, I’ll plot boxplot and get the count via IQR method.

A picture containing diagram, screenshot, rectangle, parallel

Description automatically generated

* In above boxplot, it seems that overall, 300 is the upper bound if seen from 2018 to 2021 but to be double sure, I will use IQR method to get the range and the number of outliers of AQI after data cleaning.
* After using IQR method, the counts above upper bound and below lower bound. Below is the screenshot for the same.

A black text on a white background

Description automatically generated with low confidence

* Here, median is 106.0, whereas upper bound or upper limit is 654 and lower is 0 as it is cleaned data. I have removed these 654 total rows to get the clean file to plot histogram on the same.

1. For Raw\_Conc:

* Here is the scatterplot for Raw\_Conc with year as x-axis and Raw\_Conc as y-axis. This has a legend with AQI\_Category.

A picture containing text, screenshot, diagram, font

Description automatically generated

* From the above graph, it is visible that below 0 and above 500 are the outliers. But to be sure, I have plotted boxplot and used IQR to get the actual numbers. Here -999 are the rows which already has all the data and has QC\_name as “missing” in data.
* Here is the plot boxplot and I will be getting the counts using IQR method.

A picture containing text, diagram

Description automatically generated

* From above boxplot, it seems that the upper range would be around 150 and it also has lower range as 0. Anything below 0 is the same number which I observed in scatter plot. To get the actual number of outliers I am going to use IQR method. After using the formula below is the output for the same.

A black text on a white background

Description automatically generated with low confidence

* Raw\_Conc has 43.0 as median and a total of 2949 upper outlier and 202 lower outliers. I have cleaned the data by removing these outliers from the data to get a clean histogram.

1. For NowCast\_Conc:

* Here is the scatterplot for NowCast\_Conc with year as x-axis and NowCast\_Conc as y-axis. This has a legend with AQI\_Category.

A picture containing text, screenshot, diagram, plot

Description automatically generated

* It can be observed that the outliers are not evenly spread throughout the years and hence all have different max range whereas 0 is the lower range or lower bound which will be used in IQR method for detection of outliers later.
* Here is the boxplot for the same.

A picture containing text, diagram, line, parallel

Description automatically generated

* For the boxplot, the upper bound is around 150 and lower bound is 0. Also, 2019 has the most outlier which I will detect using the formula for IQR method. Here is the output after using the IQR formula.

A screenshot of a computer

Description automatically generated with low confidence

* For NowCast\_Conc, the median value is 42.5 and the outliers are 2695 all of them are from the upper bound as I observed in scatter plot and boxplot.

1. After removing all the outliers from the data to plot histogram in the next section, the remaining data was calculated using the ".shape()” and below is the screenshot of the same. Here I used multiple removal of outliers and hence the last shape is as mentioned below.

A picture containing text, font, white, screenshot

Description automatically generated

1. Univariate Normality:
2. For histograms of continuous variables, I can see that NowCast\_Conc, AQI and Raw\_Conc are the 3 columns that has continuous data. Below are the histograms of each.
3. For AQI, here is the histogram plot:

A picture containing screenshot, diagram, plot, text

Description automatically generated

* For AQI, it can be observed that it is right skewed, meaning the tale is tilting towards right when plotted. It can be observed that the values are plotted in x-axis and their frequencies are on y-axis with 0 to almost 100 as the maximum time placed in data with about 12000 times in data.
* There seems to be skewedness in the graph meaning that there might be something off in the data and hence needs special attention like Raw\_Conc has 217 rows with -999 data and is marked as missing meaning that the values are pseudo values and hence, we might need to figure out a way to correct it to get a symmetrical data.

1. For AQI, below is the histogram plotted.

A picture containing text, screenshot, plot, line

Description automatically generated

* It is plotted with the values as x-axis and frequency as y-axis.
* There still is some value present before 0 which was the lower bound and hence the graph is a bit odd. This could be because of the null values in Raw\_Conc which are labelled as missing.
* Although the graph is not proper, it is right skewed if I ignore the -999 or left graph of 0.

1. For Raw\_Conc here is the histogram.

A picture containing text, screenshot, plot, line

Description automatically generated

* For Raw\_Conc, values are on x-axis and its frequency is on the y-axis.
* It seems that the maximum values are tilted in between 0 to 150 with a frequency of more than 17500.
* The graph is right skewed, and it seems that there are a few values which needs to be observed to make this graph symmetrical and proper. In addition to the cleaned data, it seems that those 217 rows with -999 have affected the whole data set and need a special attention.

1. Analysis & Result:

* In addition to the whole analysis, my summary is that the data after cleaning lost a lot of rows (*about 4k+*) and it seems that still there are certain rows that need special attention like Raw\_Conc 217 rows and then after getting it correct the outlier needs to be detected again. I have created new data frames every time and hence the original data is still preserved to go back to in case I see any need to. In addition to my above analysis, the AQI category with the most values is “Unhealthy for sensitive groups” or “Very unhealthy” and Ulaanbaatar needs special attention for the air quality.