AIT – 580 Final Project
Yemen: Cholera Outbreak Epidemiology Data

About Data: Everything You need to Know!

The dataset consists of data from Yemen's Cholera outbreak in the year 2017. It was a time of great misery, which was considered as one of the most awful events mankind has ever witnessed. With over 12,000 cases being added to the affected and diseased list every week, the count reached up to 1.2 million in total with almost 2,600 deaths in total due to the outbreak.

The Yemen Cholera Outbreak data was made available by the Yemen Governorate with the help of World Health Organization (W.H.O). The dataset contains **5432 Rows** and **7 Columns**.

Metadata: Description of Dataset Columns

Column	Description
Date	Date when the figures were reported.
Governorate	The Governorate name as reported in the WHO epidemiology bulletin.
Cases	Number of cases recorded in the governorate since 27 April 2017.
Deaths	Number of deaths recorded in the governorate since 27 April 2017.
CFR (%)	The case fatality rate in governorate since 27 April 2017.
Attack Rate (per 1000)	The attack rate per 1,000 of the population in the governorate since 27 April 2017.
COD Gov Pcode	The PCODE name for the governorate according to the Inter Agency Standing Committee (IASC) Common Operation Datasets (CODs) for Yemen.

Data Dictionary

S.No.	Column: Variables	Data Type
1.	Date	Date
2.	Governorate	String
3.	Cases	Integer
4.	Deaths	Integer
5.	CFR (Case Fatality Ratio)	Decimal
6.	Attack Rate (Per 1000)	Decimal
7.	COD Gov Pcode	Integer

Analysis

The dataset was now subjected to a series of phases of analysis, to determine a number of aspects that were hidden in the data. The data made to go through Python programming in the initial phase, then was subjected to R analysis, then was queried upon using PostgreSQL and lastly was visualized with the help of all the three; namely, Python, R and Tableau.

Phase I: Programming in Python: Dataset Analysis – Using Python

[1]The Yemen dataset was provided online at https://data.world/hdx/c12b27ea-5be4-4064-b87c-624cbbb7b1e1. The dataset was in a ".csv" format. The data was downloaded and was read into python. The steps below explain all that was done in python to analyze data. [2]

a) Data reading from .csv file.

```
#Importing all Libraries and Dependencies
import pandas as pd
import os
import matplotlib.pyplot as plt
import re
import numpy as np
```

```
#Setting the Work Directory
os.getcwd()
os.chdir('/Users/sakshamarora/Documents/AIT 580/AIT - 580 - Final Data Analysis Project - Saksham Arora - G01157124 ')
os.getcwd()
```

```
#Reading the file using Pandas in Python
dataframe = pd.read_csv("CholeraOutbreak.csv")
print(dataframe)
```

Output:

Date	Governorate	Cases	Deaths	CFR (%)	Attack Rate (per 1000)	COD Gov Pcode
11/26/17	Amran	94581	174	0.18	81.496	29
11/26/17	Al Mahwit	56447	148	0.26	77.303	27
11/26/17	Al Dhale'e	47004	81	0.17	64.257	30
11/26/17	Abyan	28103	35	0.12	49.232	12
11/26/17	Sana'a	68453	122	0.18	46.556	23
11/26/17	Hajjah	106933	417	0.39	45.899	17
11/26/17	Dhamar	90560	160	0.18	45.004	20
11/26/17	Al Hudaydah	139145	271	0.19	42.97	18
11/26/17	Al Bayda	26793	33	0.12	35.282	14
11/26/17	Amanat Al Asimah	91799	70	0.08	32.463	13
11/26/17	Al Jawf	14689	22	0.15	25.388	16
11/26/17	Raymah	14497	117	0.81	23.892	31
11/26/17	Lahj	22596	21	0.09	22.397	25

b) Checking for NULL Values and dealing with them – Cleaning Data

```
#Showing Data with columns having Null Values
print("Showing Total Number of null values in each column : ")
print(dataframe.isnull().sum())
```

```
#Dealing with Nulll values in the dataset
dataframe.fillna(0, inplace=True)
print(dataframe)
print(dataframe.dtypes)
```

c) Data type conversion for ease of Analysis – Handling Missing Data

```
#Conversion of String(Default) columns to Numeric Colums for Analysis
dataframe["Date"] = pd.to_datetime(dataframe["Date"], errors='coerce')

dataframe["Cases"] = pd.to_numeric(dataframe["Cases"], errors='coerce')
dataframe["Cases"] = dataframe["Cases"].fillna(dataframe["Cases"].mean()).astype(np.int64)

dataframe["Deaths"] = pd.to_numeric(dataframe["Deaths"], errors='coerce')
dataframe["Deaths"] = dataframe["Deaths"].fillna(dataframe["Deaths"].mean()).astype(np.int64)

dataframe["CFR (%)"] = pd.to_numeric(dataframe["CFR (%)"], errors='coerce')
dataframe["CFR (%)"] = dataframe["CFR (%)"].fillna(dataframe["CFR (%)"].mean())

dataframe["Attack Rate (per 1000)"] = pd.to_numeric(dataframe["Attack Rate (per 1000)"],
errors='coerce')
dataframe["Attack Rate (per 1000)"] = dataframe["Attack Rate (per 1000)"].fillna(dataframe["Attack
Rate (per 1000)"].mean())

print("Mean = ",dataframe["COD Gov Pcode"].mean())
dataframe["COD Gov Pcode"] = dataframe["COD Gov Pcode"]
dataframe["COD Gov Pcode"] = dataframe["COD Gov Pcode"].replace(0,dataframe["COD Gov
Pcode"].mean()).astype(np.int64)
```

d) Checking the Null Value's status after cleaning data

```
#Checking the Null value status again after dealing with Null Values
print("\nValues of the columns after replacing the null values with the mean value for each column :
")
print(dataframe.isnull().sum())
print(dataframe.dtypes)
print(dataframe)
```

Output:

```
RESTART: /Users/sakshamarora/Documents/AIT 580/AIT - 580 - Final Data Analysis Project - Saksham Arora - G01157124 /Analysis.py
Squeezed text (64 lines).

Showing Total Number of null values in each column :
Date                       0
Governorate                0
Cases                      0
Deaths                     0
CFR (%)                    0
Attack Rate (per 1000)     0
COD Gov Pcode            366
dtype: int64
Squeezed text (64 lines).
Date                      object
Governorate               object
Cases                     object
Deaths                     int64
CFR (%)                  float64
Attack Rate (per 1000)   float64
COD Gov Pcode            float64
dtype: object
Mean =  19.664334376726202

Values of the columns after replacing the null values with the mean value for each column :
Date                     0
Governorate              0
Cases                    0
Deaths                   0
CFR (%)                  0
Attack Rate (per 1000)   0
COD Gov Pcode            0
dtype: int64
Date                     datetime64[ns]
Governorate                      object
Cases                             int64
Deaths                            int64
CFR (%)                         float64
Attack Rate (per 1000)          float64
COD Gov Pcode                     int64
dtype: object
Squeezed text (64 lines).
```

e) Finding the Statistical Summary

```
#Statistical Summary
print("Statistic Summary for the Dataset - Yemen Cholera Epidemiology : ")
print(dataframe.describe())
```

Output:

```
Statistic Summary for the Dataset - Yemen Cholera Epidemiology :
               Cases       Deaths  ...  Attack Rate (per 1000)  COD Gov Pcode
count    5431.000000  5431.000000  ...             5431.000000    5431.000000
mean    12745.691401   116.638004  ...               33.478465      20.944762
std     17516.364775    96.045295  ...               28.004167       6.000390
min         2.000000     0.000000  ...                0.000000      11.000000
25%      3267.500000    28.000000  ...               10.083000      16.000000
50%      6611.000000   103.000000  ...               24.530000      21.000000
75%     12745.000000   183.000000  ...               53.810000      26.000000
max    139145.000000   417.000000  ...               98.970000      31.000000

[8 rows x 5 columns]
```

```
#CSV formation out of dataframe
dataframe.to_csv('AIT-580-FinalProject.csv', index=False, encoding='utf-8')
```

[3]Phase II: Analysis using R and R Studio – Descriptive Statistics

The data is now clean and ready for analysis. The data is now introduced to various algorithms in R to calculate various aspects of data and find out more about the data like trends or patterns hidden in data.

1. Regression Analysis

The data is now run through a series of regression models to check for relationship among the various variables and check whether any variable is partially or totally dependent on each other or not. The analysis was as follows:

```
#Setting Work Directory
setwd("~/Documents/AIT 580/AIT - 580 - Final Data Analysis Project - Saksham Arora - G01157124
")
```

```
#Reading the file into R
library(readr)
AIT_580_FinalProject <- read_csv("AIT-580-FinalProject.csv")
```

```
#Regression Model - 1
model1 <- lm(AIT_580_FinalProject$`Attack Rate (per 1000)`~Cases + Deaths, data =
AIT_580_FinalProject)
summary(model1)
layout(matrix(c(1,2,3,4),2,2))
plot(model1)
```

```
> summary(model1)

Call:
lm(formula = AIT_580_FinalProject$`Attack Rate (per 1000)` ~
    Cases + Deaths, data = AIT_580_FinalProject)

Residuals:
    Min      1Q  Median      3Q     Max
-50.881 -21.772  -7.617  19.483  75.151

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.481e+01  5.848e-01   42.42   <2e-16 ***
Cases       -2.177e-04  2.126e-05  -10.24   <2e-16 ***
Deaths       9.814e-02  3.877e-03   25.31   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 26.46 on 5428 degrees of freedom
Multiple R-squared:  0.1076,    Adjusted R-squared:  0.1072
F-statistic: 327.1 on 2 and 5428 DF,  p-value: < 2.2e-16
```

Output:

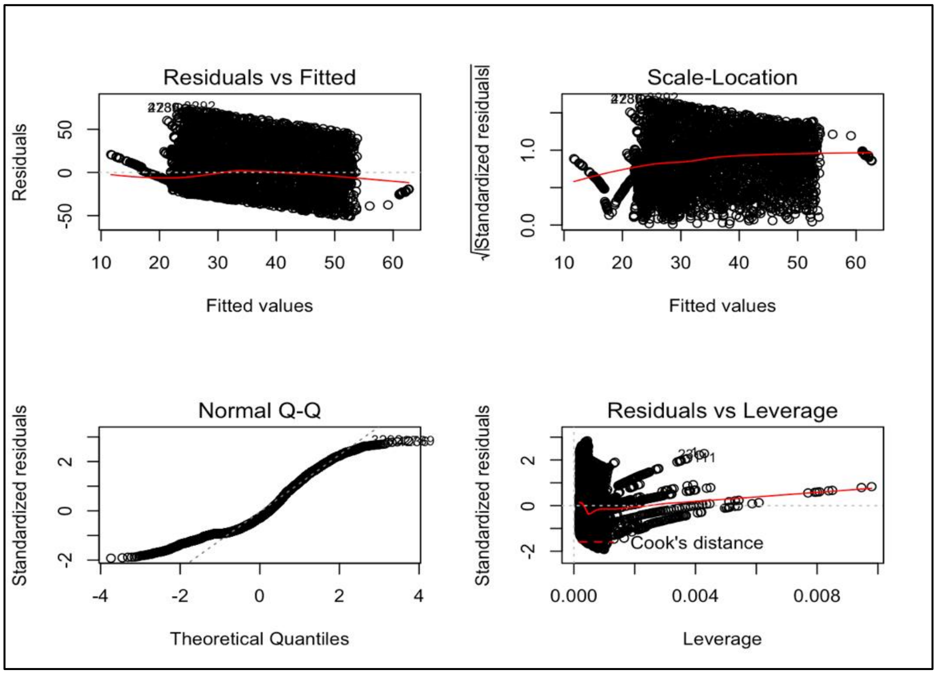


Figure 1: Regression Statistics (Produced by R) – Plot 1

```
#Regression Model - 2
model2 <- lm(AIT_580_FinalProject$`CFR (%)`~Cases + Deaths, data = AIT_580_FinalProject)
layout(matrix(c(1,2,3,4),2,2))
plot(model2)
```

Output:

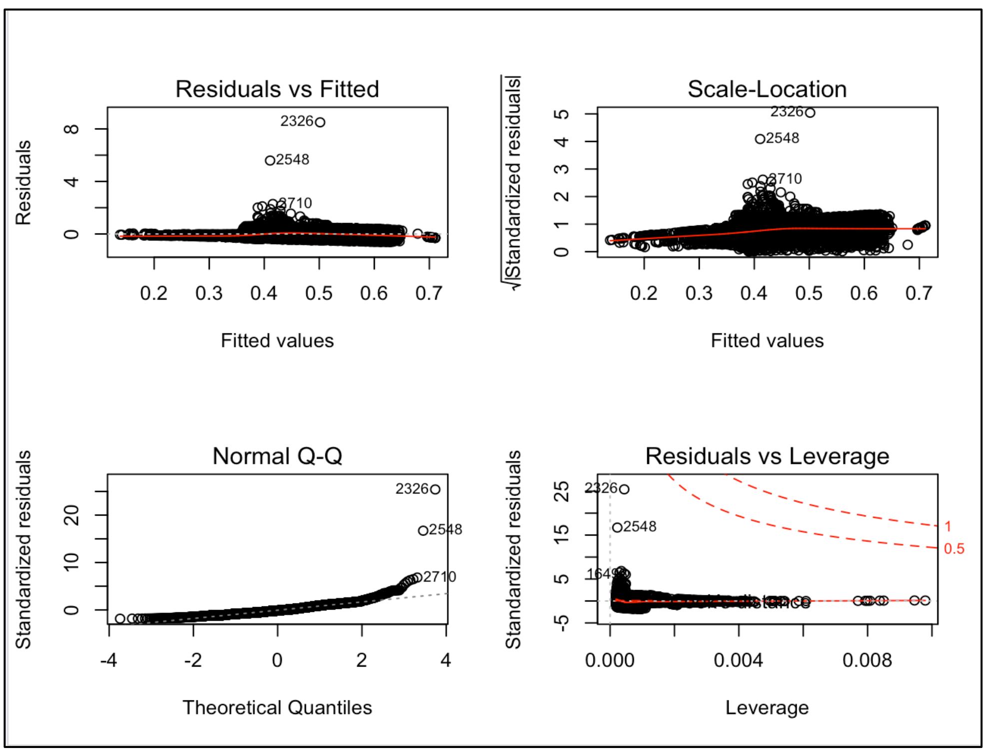


Figure 2: Regression Statistics (Produced by R) – Plot 2

Explanation of the Regression:

Residuals vs Fitted plot

The dotted line at the 0 point represents the line of exact fit, while the red line is a smoothed polynomial curve which represents the pattern of residual movement. The points as per the line are as below:
1. Above the line - Positive residual
2. Below the line - Negative residual
3. On the line - Zero residual

The above plots have been obtained by using the cleaned dataset.
Plot 1 - In this plot the red line is quite near to the fitted line. Therefore, this plot can be considered to be nearing accuracy. Complete accuracy can be achieved by using a quadratic term.
Plot 2 - This plot is better fitted than the plot 1. The slight curve in the mid is because of the outliers like, 2326 and 2548. Otherwise, this plot is quite a good fit.

Scale-Location plot

This plot signifies the spread of the points that lie between the range of the predicted values. If the red line in the plot would have been horizontally straight, it would have signified uniform variance in the residuals. Homoscedasticity is one of the major assumptions of regression and it denotes that variance in the plot should be equal within the range of predictors.
Plot 1 - This plot mostly seems to be Homoscedastic.
Plot 2 - This plot seems to be even more Homoscedastic, excluding the range of 0.4 to 0.5, which is slightly Heteroscedastic.

Normal Q-Q plot

The Normal Q-Q plot, as the name suggests is used to check if the residuals follow normal distribution or not. The points which follow the dotted line signify normal distribution.
Plot 1 - The graph that has been obtained by plotting the dataset, clears the test of Normality, as is visible from the plot. All the points are normally distributed except for 1 point at the upper right corner of the plot.
Plot 2 - This plot does not completely clear the test of normality. Majority of the points are normally distributed, except for a few - 2326, 2548, 2710.

Residuals vs Leverage plot

There are two important terms for the Residuals vs Leverage plot:
Influence - A particular observation's influence can be very well understood by the fact that unto extent will its exclusion affect the predicted values.
Leverage - This can be termed as the difference between the mean of the predictor variable and the observation's value.
Both these terms go hand in hand, as in if one increases the other one automatically increases with it.
The cook's distance is the red dashed line.
Plot 1 - The dotted line seems to near perfection, and there are no outliers that are present in the plot.
Plot 2 - In this graph there are no outliers that exist in the area of concern (above the curved red line in the top right section). Therefore, this graph is quite a well fitted graph

2. Correlation Analysis

The data is now run through a series of correlation models to check whether any two variables are correlated or not, and if they are, we check how strong or weak the correlation is, between them.

```
#Correaltion between Cases and Deaths
cor(AIT_580_FinalProject$Cases,AIT_580_FinalProject$Deaths, method = 'spearman')
```

```
#Correaltion between Cases and Attack Rate (per 1000)
cor(AIT_580_FinalProject$Cases,AIT_580_FinalProject$`Attack Rate (per 1000)`, method =
'kendall')
```

```
#Correaltion between Cases and Case Fatality Ratio Percentage
cor(AIT_580_FinalProject$Cases,AIT_580_FinalProject$`CFR (%)`, method = 'kendall')
```

```
#Correaltion between Deaths and Case Fatality Ratio Percentage
cor(AIT_580_FinalProject$Deaths,AIT_580_FinalProject$`CFR (%)`, method = 'spearman')
```

```
#Correaltion between Deaths and Attack Rate (per 1000)
cor(AIT_580_FinalProject$Deaths,AIT_580_FinalProject$`Attack Rate (per 1000)`, method =
'spearman')
```

Output:

```
> cor(AIT_580_FinalProject$Cases,AIT_580_FinalProject$Deaths, method = 'spearman')
[1] 0.2874436
> cor(AIT_580_FinalProject$Cases,AIT_580_FinalProject$`Attack Rate (per 1000)`, method = 'kendall')
[1] 0.08482019
> cor(AIT_580_FinalProject$Cases,AIT_580_FinalProject$`CFR (%)`, method = 'kendall')
[1] -0.006346657
> cor(AIT_580_FinalProject$Deaths,AIT_580_FinalProject$`CFR (%)`, method = 'spearman')
[1] 0.3055641
> cor(AIT_580_FinalProject$Deaths,AIT_580_FinalProject$`Attack Rate (per 1000)`, method = 'spearman')
[1] 0.4287188
```

Note: The correlation is checked to be strong, normal or weak based on the correlation coefficient value between two variables. Thus, a coefficient near to (+1 or -1) is known to be highly correlated and other normally correlated and the values further away from 1 are weakly correlated. Both -1 and +1 i.e. Positive and Negative correlation are as good as each other.

In the above correlation analysis, we found that Deaths and Attack Rate are the ones which are the most correlated, while Cases and Case Fatality Ratio are highly non-correlated.

3. Hypothesis Testing

The data is now subjected to a number of hypothesis to check whether any hypothesis comes out to be true or not.

a) Single Variable T-test Hypothesis

```
t.test(AIT_580_FinalProject$Deaths)
t.test(AIT_580_FinalProject$Cases)
t.test(AIT_580_FinalProject$`CFR (%)`)
t.test(AIT_580_FinalProject$`Attack Rate (per 1000)`)
```

Output:

```
> t.test(AIT_580_FinalProject$Deaths)

	One Sample t-test

data:  AIT_580_FinalProject$Deaths
t = 89.496, df = 5430, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 114.0831 119.1929
sample estimates:
mean of x
  116.638
```

```
> t.test(AIT_580_FinalProject$Cases)

	One Sample t-test

data:  AIT_580_FinalProject$Cases
t = 53.624, df = 5430, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 12279.73 13211.65
sample estimates:
mean of x
 12745.69
```

```
> t.test(AIT_580_FinalProject$`Attack Rate (per 1000)`)

	One Sample t-test

data:  AIT_580_FinalProject$`Attack Rate (per 1000)`
t = 88.101, df = 5430, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 32.73351 34.22342
sample estimates:
mean of x
 33.47846
```

```
> t.test(AIT_580_FinalProject$`CFR (%)`)

	One Sample t-test

data:  AIT_580_FinalProject$`CFR (%)`
t = 95.26, df = 5430, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 0.4383335 0.4567539
sample estimates:
mean of x
0.4475437
```

b) Correlation Hypothesis

```
#Running Correlation Hypothesis between Deaths and Cases
cor.test(AIT_580_FinalProject$Deaths, AIT_580_FinalProject$Cases)

#Running Correlation Hypothesis between Deaths and Case Fatality Ratio Percentage
cor.test(AIT_580_FinalProject$Deaths, AIT_580_FinalProject$`CFR (%)`)

#Running Correlation Hypothesis between Cases and Case Fatality Ratio Percentage
cor.test(AIT_580_FinalProject$Cases, AIT_580_FinalProject$`CFR (%)`)

#Running Correlation Hypothesis between Deaths and Cases
cor.test(AIT_580_FinalProject$Cases, AIT_580_FinalProject$`Attack Rate (per 1000)`)

#Running Correlation Hypothesis between Deaths and Cases
cor.test(AIT_580_FinalProject$Deaths, AIT_580_FinalProject$`Attack Rate (per 1000)`)
```

Output:

```
> cor.test(AIT_580_FinalProject$Deaths, AIT_580_FinalProject$Cases)

	Pearson's product-moment correlation

data:  AIT_580_FinalProject$Deaths and AIT_580_FinalProject$Cases
t = 20.229, df = 5429, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2398366 0.2893039
sample estimates:
      cor
0.2647444
```

```
> cor.test(AIT_580_FinalProject$Deaths, AIT_580_FinalProject$`CFR (%)`)

	Pearson's product-moment correlation

data:  AIT_580_FinalProject$Deaths and AIT_580_FinalProject$`CFR (%)`
t = 15.048, df = 5429, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.1744317 0.2254964
sample estimates:
      cor
0.2000999
```

```
> cor.test(AIT_580_FinalProject$Cases, AIT_580_FinalProject$`CFR (%)`)

	Pearson's product-moment correlation

data:  AIT_580_FinalProject$Cases and AIT_580_FinalProject$`CFR (%)`
t = -7.9011, df = 5429, p-value = 3.323e-15
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.1328412 -0.0802523
sample estimates:
       cor
-0.1066213
```

```
> cor.test(AIT_580_FinalProject$Cases, AIT_580_FinalProject$`Attack Rate (per 1000)`)

	Pearson's product-moment correlation

data:  AIT_580_FinalProject$Cases and AIT_580_FinalProject$`Attack Rate (per 1000)`
t = -3.471, df = 5429, p-value = 0.0005225
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.07356098 -0.02048553
sample estimates:
        cor
-0.04705647
```

```
> cor.test(AIT_580_FinalProject$Deaths, AIT_580_FinalProject$`Attack Rate (per 1000)`)

	Pearson's product-moment correlation

data:  AIT_580_FinalProject$Deaths and AIT_580_FinalProject$`Attack Rate (per 1000)`
t = 23.217, df = 5429, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2761452 0.3245370
sample estimates:
      cor
0.3005345
```

Phase III: Querying Using PostgreSQL [4]

Now that we are done with analysis using both R and Python, we introduce the data to Querying language PostgreSQL, where we use the method of writing queries to extract various outputs from the data to answer various different questions.

We start the analysis by all the basic steps:

#Create table Query

```
[AIT_FinalProject=# create table cholera(Date date, Governorate varchar(20), Cases int, Deaths int,CFR_Percent float, Attack_Rate_Per_1000 float,COD_Gov_Code int);
CREATE TABLE
```

```
[AIT_FinalProject=# \dt
           List of relations
 Schema |  Name   | Type  |    Owner
--------+---------+-------+--------------
 public | cholera | table | sakshamarora
(1 row)
```

#Describe Table Query

```
[AIT_FinalProject-# \d+ cholera
                                              Table "public.cholera"
       Column        |         Type          | Collation | Nullable | Default | Storage  | Stats target | Description
---------------------+-----------------------+-----------+----------+---------+----------+--------------+-------------
 date                | date                  |           |          |         | plain    |              |
 governorate         | character varying(20) |           |          |         | extended |              |
 cases               | integer               |           |          |         | plain    |              |
 deaths              | integer               |           |          |         | plain    |              |
 cfr_percent         | double precision      |           |          |         | plain    |              |
 attack_rate_per_1000 | double precision     |           |          |         | plain    |              |
 cod_gov_code        | integer               |           |          |         | plain    |              |
```

#Adding/Importing Records from CSV file.

```
[AIT_FinalProject=# copy cholera(Date, Governorate, Cases, Deaths, CFR_Percent, Attack_Rate_Per_1000, COD_Gov_Code) from '/Users/sakshamarora/Documents/AIT 580/AIT - 580 - Final Data Analysis Project - Sak]
sham Arora - G01157124 /AIT-FinalProject-Report.csv' DELIMITER ',' CSV HEADER;
COPY 5431
```

Now once, the records were inserted in a database table, I wrote queries to get answers to various questions from the dataset.

The Questions and Queries are as follows:

1. Find the total number of cases of Cholera.

```
[AIT_FinalProject=# SELECT SUM(Cases) as "Total Cases of Cholera"  from cholera;
 Total Cases of Cholera
------------------------
               69221850
(1 row)
```

2. Find the deaths among the total number of cases.

```
[AIT_FinalProject=# SELECT SUM(Deaths) as "Total Cases of Cholera which resulted in Deaths"  from cholera;
 Total Cases of Cholera which resulted in Deaths
-------------------------------------------------
                                          633461
(1 row)
```

3. Find the rate of attack of cholera in each governorate with the total number of deaths.

```
AIT_FinalProject=# select governorate, attack_rate_per_1000, deaths from cholera;
```

governorate	attack_rate_per_1000	deaths
Amran	81.496	174
Al Mahwit	77.303	148
Al Dhale'e	64.257	81
Abyan	49.232	35
Sana'a	46.556	122
Hajjah	45.899	417
Dhamar	45.004	160
Al Hudaydah	42.97	271
Al Bayda	35.282	33
Amanat Al Asimah	32.463	70
Al Jawf	25.388	22
Raymah	23.892	117
Lahj	22.397	21
Aden	21.978	62
Ibb	20.267	284
Taizz	19.419	184
Marib	19.236	7
Sa'ada	10.741	5
Al Maharah	7.86	1
Shabwah	2.31	3
Moklla	1.417	2

4. Find the Governorate(s) which were the most affected by number of cases of Cholera.

```
AIT_FinalProject=# select governorate as "Most Affected Governorates", cases as "Number of Cases" from cholera where cases = (select MAX(cases) from cholera) group by governorate,cases;
```

Most Affected Governorates	Number of Cases
Al Hudaydah	139145

(1 row)

5. Find the maximum number of cases in each of the distinct Governorates.

```
AIT_FinalProject=# select governorate as "Distinct Governorates",MAX(Cases) as "Number of Cases" from cholera group by governorate;
```

Distinct Governorates	Number of Cases
Abyan	28103
Raymah	14497
Marib	12745
Hajjah	106933
Dhamar	90560
Al Mahwit	56447
Al Jawf	14689
Lahj	22596
Al Bayda	26793
Taizz	58223
Aden	20286
Say'on	9184
AL Mahrah	9114
Moklla	9216
Al_Jawf	12745
Ibb	59932
Amanat Al Asimah	91799
Sa'ada	12745
Ma'areb	12745
Shabwah	12745
Al Hudaydah	139145
Al Dhale'e	47004
Al Maharah	12745
Amran	94581
Al-Hudaydah	55409
Sana'a	68453

(26 rows)

6. Find the Governorate Code for the governorate where maximum number of people died.

```
AIT_FinalProject=# select cod_gov_code as "Government PCode", deaths as "Total number of Deaths" from cholera where deaths = (select MAX(deaths) from cholera) group by deaths,cod_gov_code;
 Government PCode | Total number of Deaths
------------------+------------------------
               17 |                    417
(1 row)
```

7. Find the dates and name of governorate when there were maximum number of cases and maximum number of deaths.

```
AIT_FinalProject=# select date, governorate,cases,deaths from cholera where cases = (select MAX(cases) from cholera) OR deaths = (select MAX(deaths) from cholera) group by date, governorate,cases,deaths
;
    date    | governorate | cases  | deaths
------------+-------------+--------+--------
 2017-11-26 | Al Hudaydah | 139145 |    271
 2017-11-26 | Hajjah      | 106933 |    417
(2 rows)
```

8. Find all the governorates and their max deaths.

```
AIT_FinalProject=# select governorate as "Distinct Governorates",MAX(Deaths) as "Number of Deaths" from cholera group by governorate;
 Distinct Governorates | Number of Deaths
-----------------------+------------------
 Abyan                 |              293
 Raymah                |              298
 Marib                 |              289
 Hajjah                |              417
 Dhamar                |              300
 Al Mahwit             |              299
 Al Jawf               |              295
 Lahj                  |              296
 Al Bayda              |              297
 Taizz                 |              294
 Aden                  |              300
 Say'on                |              295
 AL Mahrah             |              300
 Moklla                |              300
 Al_Jawf               |              290
 Ibb                   |              298
 Amanat Al Asimah      |              300
 Sa'ada                |              298
 Ma'areb               |              294
 Shabwah               |              297
 Al Hudaydah           |              292
 Al Dhale'e            |              300
 Al Maharah            |              297
 Amran                 |              299
 Al-Hudaydah           |              298
 Sana'a                |              300
(26 rows)
```

9. Find the name of the governorate, date and total number of deaths that happened on 13th July 2017.

```
AIT_FinalProject=# select date, governorate, deaths from cholera where date = '2017-07-13';
    date    |   governorate    | deaths
------------+------------------+--------
 2017-07-13 | Amanat Al Asimah |     56
 2017-07-13 | Al-Hudaydah      |    200
 2017-07-13 | Hajjah           |    344
 2017-07-13 | Amran            |    149
 2017-07-13 | Ibb              |    227
 2017-07-13 | Sana'a           |    111
 2017-07-13 | Taizz            |    154
 2017-07-13 | Dhamar           |    116
 2017-07-13 | Al Dhale'e       |     71
 2017-07-13 | Al Mahwit        |    113
 2017-07-13 | Aden             |     48
 2017-07-13 | Al Bayda         |     24
 2017-07-13 | Abyan            |     30
 2017-07-13 | Raymah           |     80
 2017-07-13 | Lahj             |     16
 2017-07-13 | Al_Jawf          |     13
 2017-07-13 | Ma'areb          |      4
 2017-07-13 | Sa'ada           |      1
 2017-07-13 | AL Mahrah        |      1
 2017-07-13 | Shabwah          |      1
 2017-07-13 | Say'on           |      0
(21 rows)
```

10. Find all the records where the case fatality ratio is less than 0.15% and date was 30th October 2017.

```
AIT_FinalProject=# select * from cholera where cfr_percent < 0.15 and date = '2017-10-30';
    date    |   governorate    | cases | deaths | cfr_percent | attack_rate_per_1000 | cod_gov_code
------------+------------------+-------+--------+-------------+----------------------+--------------
 2017-10-30 | Abyan            | 27804 |     35 |        0.13 |               48.708 |           12
 2017-10-30 | Al Bayda         | 25369 |     30 |        0.12 |               33.407 |           14
 2017-10-30 | Amanat Al Asimah | 85073 |     68 |        0.08 |               30.084 |           13
 2017-10-30 | Lahj             | 22481 |     21 |        0.09 |               22.283 |           25
 2017-10-30 | Marib            |  5943 |      7 |        0.12 |               16.575 |           26
 2017-10-30 | Sa'ada           |  7968 |      5 |        0.06 |                8.803 |           22
 2017-10-30 | Al Maharah       |  1164 |      1 |        0.09 |                 7.84 |           28
 2017-10-30 | Say'on           |    18 |      0 |           0 |                0.082 |           19
(8 rows)
```

[5]Phase IV: Exploration and Visualizations of the data and its aspects

1. Plots using Python[5]

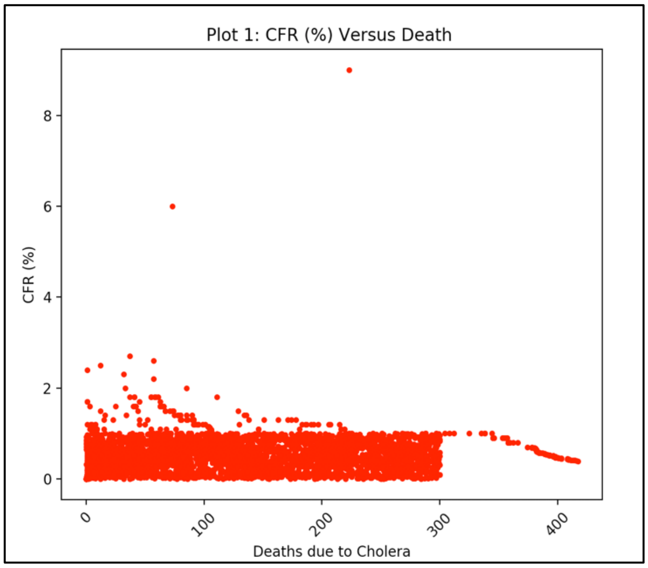


Figure 1: Case Fatality Ratio Vs Death Scatterplot

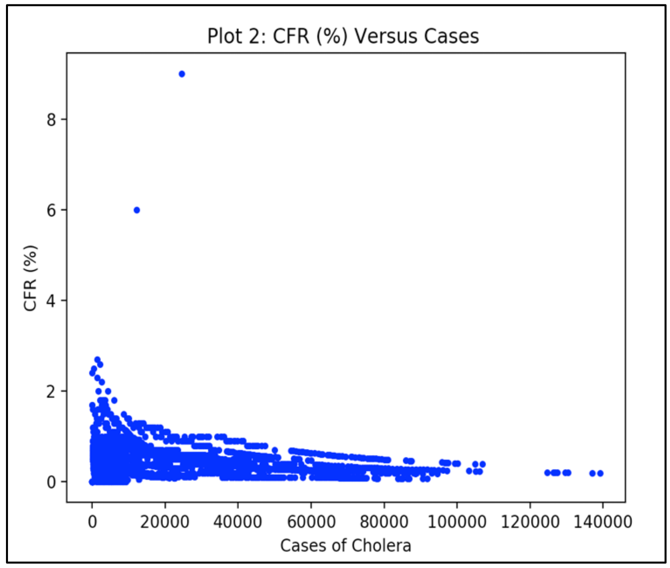


Figure 2: Case Fatality Ratio Vs Cases Scatterplot

Data Exploration – Figure 1:
The plot above shows us a relationship between Case fatality ratio and the number of deaths due to cholera. The relationship is following a formula,

$$\textbf{Case fatality ratio (CFR)} = \frac{\text{Total Number Of Deaths due to Cholera}}{\text{Total Number of Cases of Cholera}}$$

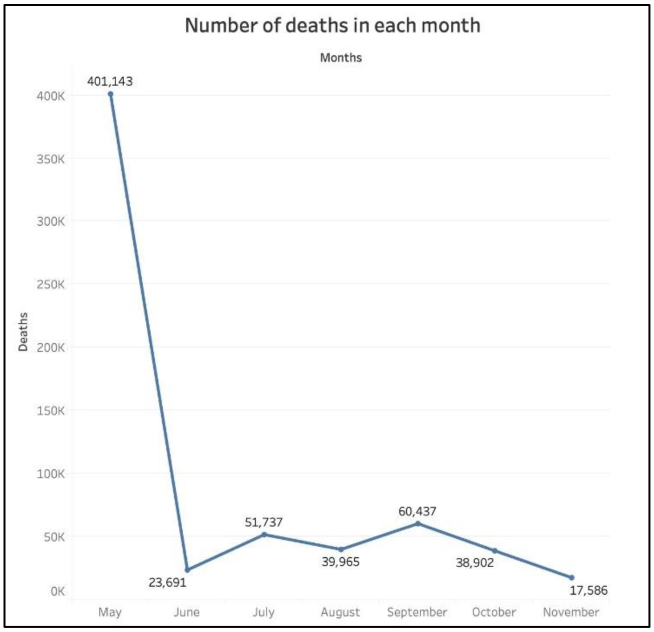
Now, the plot shows the exact relationship, i.e., Indirect or Inverse relationship, as it should with respect to the formula, i.e., with the increase in number of deaths the case fatality ratio increased. This happened up to a certain level, but then it started to decrease. Although this might seem unconventional, but a very important aspect can be made out, from this visualization, that each governorate worked to prevent any more deaths as time passed, also, they came out with various health and disease related aid so to help the people deal with those difficult times.

Data Exploration – Figure 2:
The plot above shows us a relationship between Case fatality ratio and the number of cases of cholera. The relationship is following a formula,

$$\textbf{Case fatality ratio (CFR)} = \frac{\text{Total Number Of Deaths due to Cholera}}{\text{Total Number of Cases of Cholera}}$$

Now, the plot shows the exact relationship i.e., Direct Relationship', as it should with respect to the formula, i.e., with the decrease in number of cases of cholera, the case fatality ratio decreased. Important aspects can be made out, from this visualization, that the health services and aid provided by each governorate helped the people to get better and as time passed prevented any more people from getting affected by the virus and the disease.

2. Plots using Tableau[6]



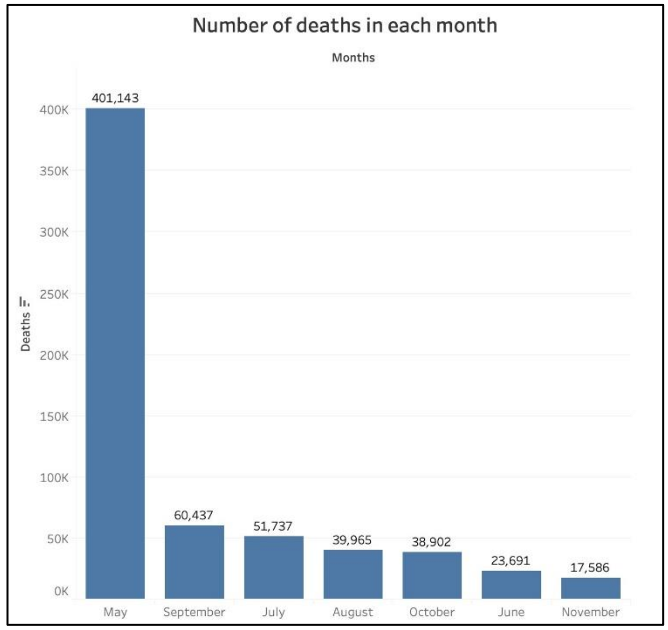
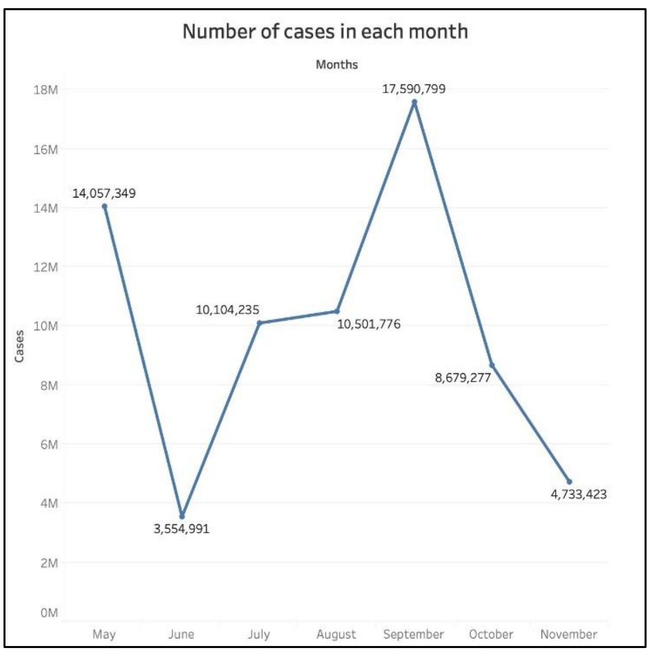


Figure 3: Line Plot for Number of deaths in each month **Figure 4: Bar Plot for Number of deaths in each month**

Both the above visualizations show the sum of number of deaths in each month. It can be clearly seen that in the start of the cholera outbreak in May 2017, there were a vast number of deaths that were recorded in the respective month, then there was a rapid fall in the sum of number of deaths in the constituent months, which kept on increasing and decreasing and then till November, it fell down with a steep, which means that the health services in the governorates helped people recover from the disease.



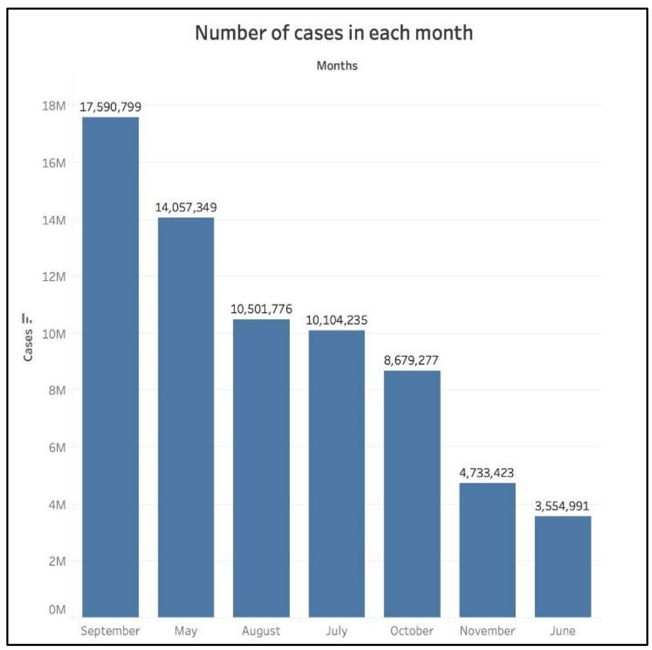


Figure 5: Line Plot for Number of cases in each month **Figure 6: Bar Plot for Number of cases in each month**

Both the above visualizations show the sum of number of cases in each month. It can be clearly seen that many cases of cholera were registered in the beginning of the outbreak in May 2017, then there was a rapid fall in the sum of number of cases, which then increased in a breaking inclination.

Then the month of September witnessed the maximum number of cases of cholera in the whole span of the cholera outbreak which then suddenly decreased exponentially in a very steep downfall. The reason behind this rapid change can be the effort of the health services in each governorate to prevent people from getting sick due to cholera.

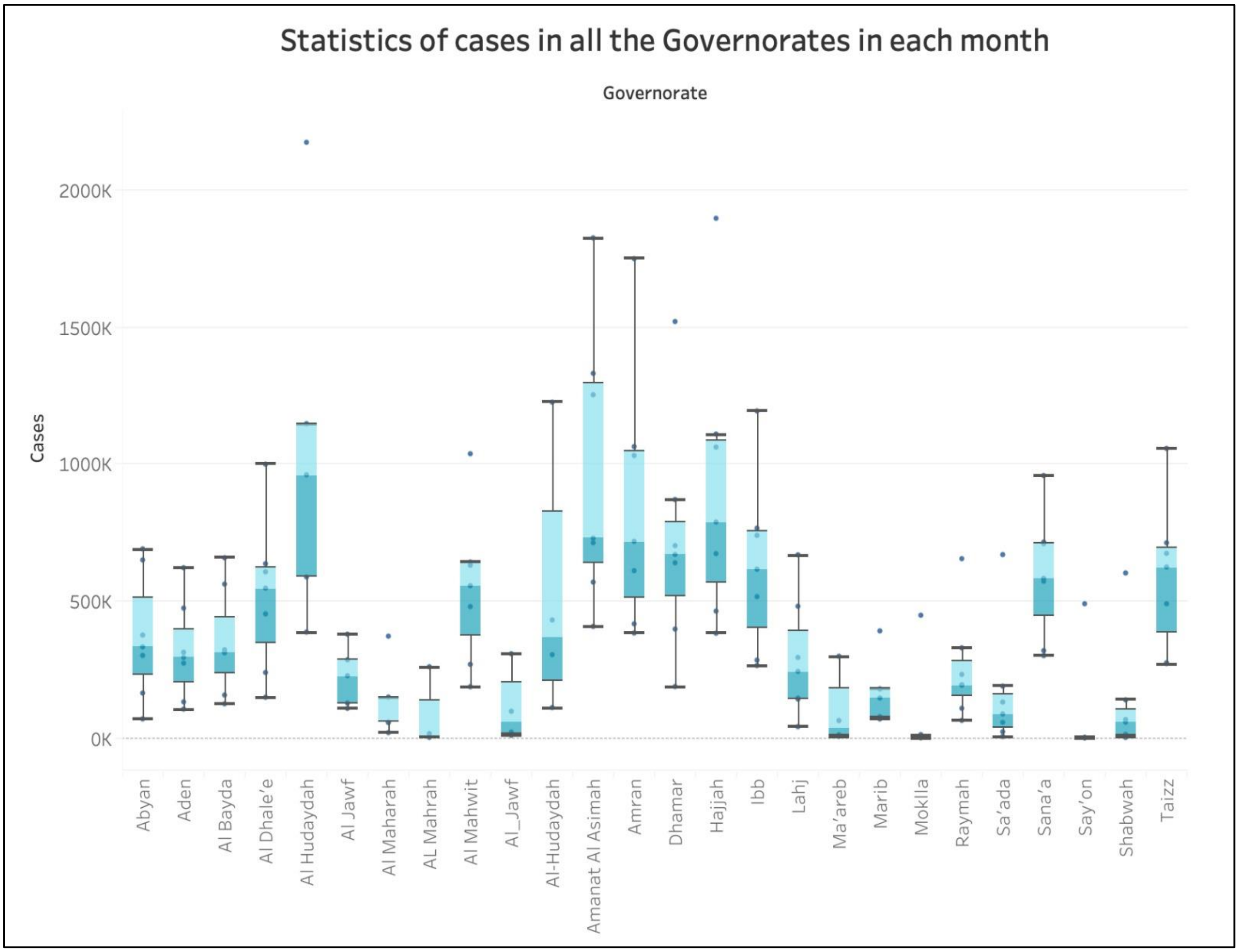


Figure 7: Boxplot statistics of cases in all the Governorates in each month

The above boxplot shows the distribution of data based on the summary statistics of the five-number summary which is the Minimum value, First Quartile(Q1), Median, Third Quartile(Q3) and the Maximum Value. The plot shows the distribution summary of all the Cholera cases spread in distinct governorates.

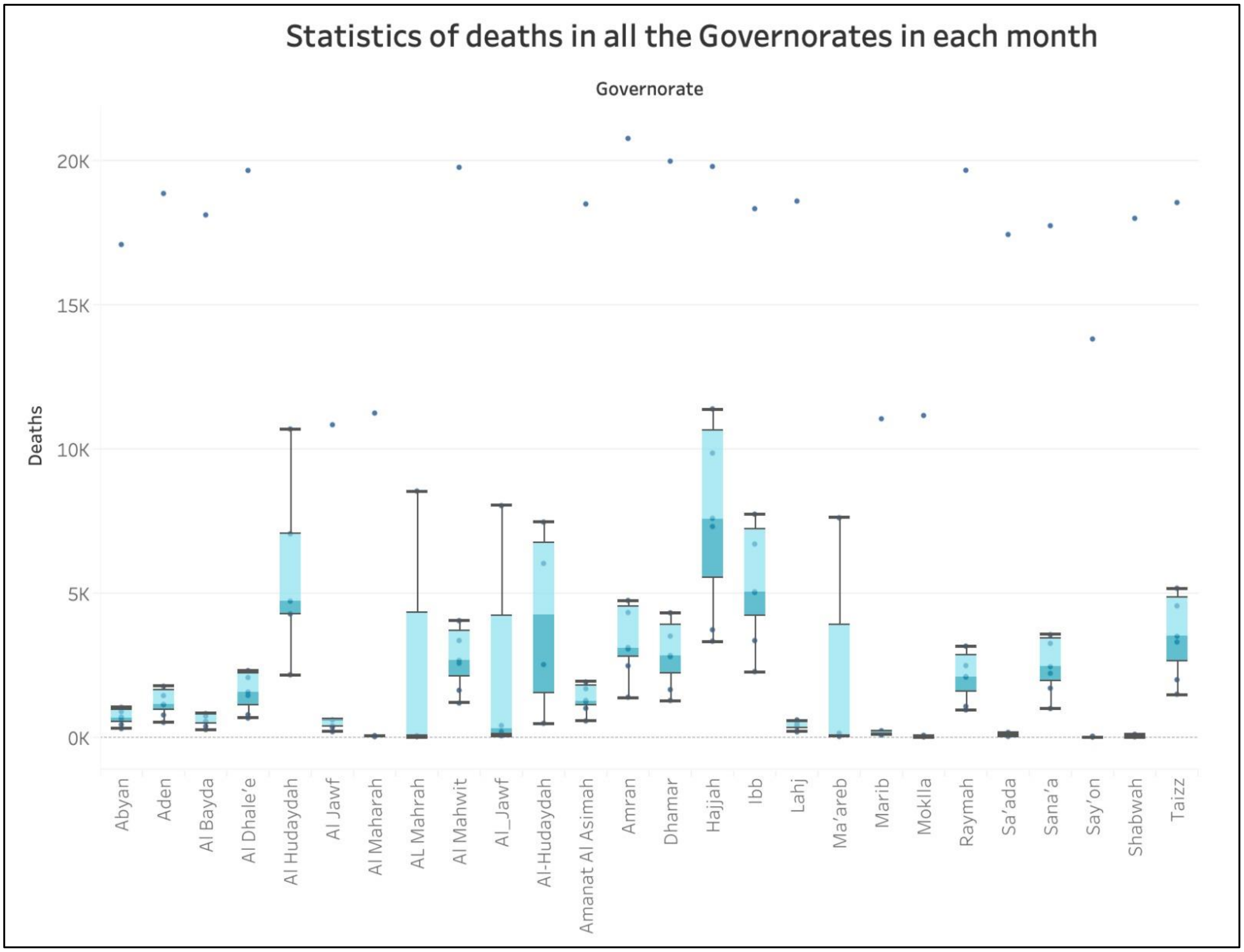


Figure 8: Boxplot statistics of deaths in all the Governorates in each month

The above boxplot shows the distribution of data based on the summary statistics of the five-number summary which is the Minimum value, First Quartile(Q1), Median, Third Quartile(Q3) and the Maximum Value. The plot shows the distribution summary of all the deaths due to cholera in distinct governorates.

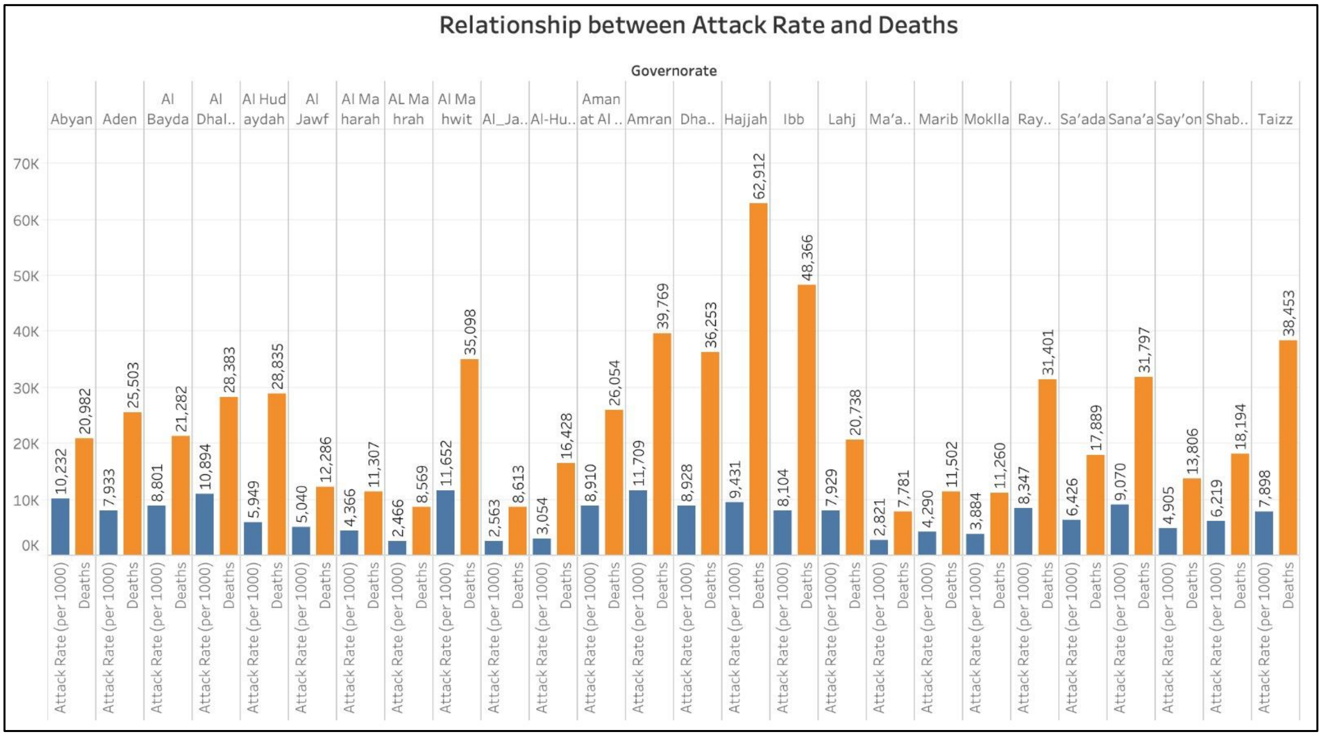


Figure 9: Bar plot showing correlation between Deaths and Attack Rate per 1000

The above relationship has been plotted as the highest correlation coefficient was found to be for the variables deaths and attack rate per 1000. This can very well be verified from the graph.

Conclusion

The motive behind the analysis was to dig deep into the dataset and find out all about the relationships, trends, patterns, and queries related to various aspects of data. We were able to answer several questions related to data and its variables that were not clear or were un-answered before the analysis.

The analysis uncovered a number of misconceptions that were there before the analysis ran its course, like the relationship between the cases and deaths was thought to be linear but it came out to be changing with change in time, other than that we found out that each governorate was working forward to provide aid to the people suffering from the problem with full force, because the plots show a rapid decrease in number of deaths of the affected people.

Conclusively, we found that the cholera outbreak affected a very large set of population in Yemen, but even after the outrageous effect of the disease the various governorates within the country were able to arrange efficient and ample amounts of health services and aid for the people suffering from the disease which helped them recover from that reckless time of misery and neutralise its effect on the people in Yemen.

Citation

[1] Governorates of Yemen and W.H.O, Cholera Outbreak (2017) , Yemen Situation Reports Page, Retrieved from: https://data.world/hdx/c12b27ea-5be4-4064-b87c-624cbbb7b1e1

[2] Python Software Foundation. Python Language Reference, version 2.7. Available, Retrieved from https://www.python.org/

[3] R Core Team (2019). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

[4] PostgreSQL, Retrieved from https://www.postgresql.org

[5] Tableau, Retrieved from https://www.tableau.com/