

Introduction to Inference and Interpretation

Project Report

On

Analysis of Hyderabad's Air Quality Index Data (2016-2022)

By

K.V.H. KASHYAP	-	160121729038
J. SAI RITHWIK	-	160121729037
PRANAV RAJ B.	-	160121729050
V. BHANU SAI	-	160121729061
M. SHYAM KUMAR	-	160121729066



Branch: Artificial Intelligence and Machine Learning (J)

BE (AI&ML) III Semester

Submitted to

Department of Computer Science and Engineering

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY (A)

(Affiliated to Osmania University)

Gandipet, Hyderabad- 500075

2022 – 2023

Under the guidance of:

Dr. M. Swamy Das, Professor, Dept. of CSE

Smt. Ch. Vijaya Lakshmi, Asst. Professor, Dept. of CSE

INTRODUCTION

1.1 Motivation and Background

An air quality index (AQI) is used by government agencies to communicate to the public how polluted the air currently is or how polluted it is forecast to become. AQI information is obtained by averaging readings from an air quality sensor, which can increase due to vehicle traffic, forest fires, or anything that can increase air pollution. Pollutants tested include ozone, nitrogen dioxide, sulphur dioxide, among others.

With the increasing population in the country, there has been an increase in the number of migrations to the metropolitan cities, the commutators in these cities have increased which leads us to review the pollution in these cities.

The situation of Delhi has turned out to be alarming as the Air Quality Index of many places in Delhi have been marked “Unhealthy.” This might also be the plight of Hyderabad soon if no preventive measures are taken by the government immediately.

Hence, we need to analyse the trends of the Air Quality Index and also predict the future Air Quality. This would help us find the reasons for the higher observations and to control the pollution in those areas.

**IMPLEMENTATION OF
ANALYSIS OF HYDERABAD'S AIR QUALITY INDEX DATA
(2016-2022) USING R LANGUAGE**

2.1 Introduction to the Dataset^[1]

The dataset has been collected from the official website of Telangana State Pollution Control Board. The yearly data was in the form of an Excel File. Each Excel File contained two sheets, one regarding the AQI of localities of Hyderabad and the other being the AQI of Localities in Telangana. Since the problem statement was restricted to Hyderabad, the data was filtered manually and the unwanted rows and columns were discarded. All the empty data fields were considered as NA.

2.2 R Code

```
setwd("C:/Users/WoW/Downloads/MonthlyAQI")
```

```
#install.packages("readxl")  
  
library("readxl")  
  
#install.packages('tidyverse')  
  
library(tidyverse)  
  
#install.packages('plotly')  
  
library(plotly)  
  
#install.packages("ggfortify")  
  
library(ggfortify)  
  
#install.packages("NbClust")  
  
library(NbClust)  
  
#install.packages("cluster")  
  
library(cluster)  
  
#install.packages("plotrix")
```

```
library(plotrix)

#install.packages("factoextra")

library(factoextra)

aqi2016<-read_excel("AQI2016.xlsx")

aqi2017<-read_excel("aqi2017.xlsx")

aqi2018<-read_excel("aqi2018.xlsx")

aqi2019<-read_excel("aqi2019.xlsx")

aqi2020<-read_excel("aqi2020.xlsx")

aqi2021<-read_excel("aqi2021.xlsx")

aqi2022<-read_excel("aqi2022.xlsx")



#Getting Area Name

area_name<- function(i) {

  a<-as.data.frame(aqi2016[i,1])

  area<-a$Location

  return(area)

}

#Creating a Vector of AQI of the Area

area_vector<- function(i) {

  aqi2016df<-as.data.frame(aqi2016[i,2:13])

  aqi2016vec<-

  c(aqi2016df$Jan,aqi2016df$Feb,aqi2016df$Mar,aqi2016df$Apr,aqi2016df$May,aqi2
```

```
016df$Jun,aqi2016df$Jul,aqi2016df$Aug,aqi2016df$Sep,aqi2016df$Oct,aqi2016df$  
Nov,aqi2016df$Dec)  
  
aqi2017df<-as.data.frame(aqi2017[i,2:13])  
  
aqi2017vec<-  
c(aqi2017df$Jan,aqi2017df$Feb,aqi2017df$Mar,aqi2017df$Apr,aqi2017df$May,aqi2  
017df$Jun,aqi2017df$Jul,aqi2017df$Aug,aqi2017df$Sep,aqi2017df$Oct,aqi2017df$  
Nov,aqi2017df$Dec)  
  
aqi2018df<-as.data.frame(aqi2018[i,2:13])  
  
aqi2018vec<-  
c(aqi2018df$Jan,aqi2018df$Feb,aqi2018df$Mar,aqi2018df$Apr,aqi2018df$May,aqi2  
018df$Jun,aqi2018df$Jul,aqi2018df$Aug,aqi2018df$Sep,aqi2018df$Oct,aqi2018df$  
Nov,aqi2018df$Dec)  
  
aqi2019df<-as.data.frame(aqi2019[i,2:13])  
  
aqi2019vec<-  
c(aqi2019df$Jan,aqi2019df$Feb,aqi2019df$Mar,aqi2019df$Apr,aqi2019df$May,aqi2  
019df$Jun,aqi2019df$Jul,aqi2019df$Aug,aqi2019df$Sep,aqi2019df$Oct,aqi2019df$  
Nov,aqi2019df$Dec)  
  
aqi2020df<-as.data.frame(aqi2020[i,2:13])  
  
aqi2020vec<-  
c(aqi2020df$Jan,aqi2020df$Feb,aqi2020df$Mar,aqi2020df$Apr,aqi2020df$May,aqi2  
020df$Jun,aqi2020df$Jul,aqi2020df$Aug,aqi2020df$Sep,aqi2020df$Oct,aqi2020df$  
Nov,aqi2020df$Dec)  
  
aqi2021df<-as.data.frame(aqi2021[i,2:13])  
  
aqi2021vec<-  
c(aqi2021df$Jan,aqi2021df$Feb,aqi2021df$Mar,aqi2021df$Apr,aqi2021df$May,aqi2  
021df$Jun,aqi2021df$Jul,aqi2021df$Aug,aqi2021df$Sep,aqi2021df$Oct,aqi2021df$  
Nov,aqi2021df$Dec)  
  
aqi2022df<-as.data.frame(aqi2022[i,2:13])
```

```

aqi2022vec<-
c(aqi2022df$Jan,aqi2022df$Feb,aqi2022df$Mar,aqi2022df$Apr,aqi2022df$May,aqi2
022df$Jun,aqi2022df$Jul,aqi2022df$Aug,aqi2022df$Sep,aqi2022df$Oct,aqi2022df$Nov,aqi2022df$Dec)

suppressWarnings(area_aqi<-
as.integer((c(aqi2016vec,aqi2017vec,aqi2018vec,aqi2019vec,aqi2020vec,aqi2021vec,
aqi2022vec)))))

return(area_aqi)
}

```

```
date<-seq(as.Date("2016-01-01"),as.Date("2022-12-01"),by="month")
```

```

df<-
data.frame(date,area_vector(1),area_vector(2),area_vector(3),area_vector(4),area_vec
tor(5),area_vector(6),area_vector(7),area_vector(8),area_vector(9),area_vector(10),ar
ea_vector(11),area_vector(12),area_vector(13),area_vector(14),area_vector(15),area_
vector(16),area_vector(17),area_vector(18),area_vector(19),area_vector(20),area_vect
or(21),area_vector(22),area_vector(23),area_vector(24),area_vector(25))

areanames<-
c("Balanagar","Uppal","JubileeHills","Paradise","Charminar","Jeedimetla","Abids",""
KBRNPark","LangarHouse","Madhapur","MGBS","Chikkadapally","Kukatpally","N
acharam","Rajendranagar","Sainikpuri","BPPA","Shameerpet","HCU","Panjagutta",""
Sanathnagar","ZooPark","Pashamylaram","Bollaram","ICRISAT")

colnames(df)<-c("Date",areanames)

summary(df)

```

```
df$MGBS[which(is.na(df$MGBS))]<-as.integer(mean(df$MGBS,na.rm=TRUE))
```

```
df$Nacharam[which(is.na(df$Nacharam))]<-
as.integer(mean(df$Nacharam,na.rm=TRUE))
```

```

df$Sainikpuri[which(is.na(df$Sainikpuri))]<-
as.integer(mean(df$Sainikpuri,na.rm=TRUE))

df$Shameerpet[which(is.na(df$Shameerpet))]<-
as.integer(mean(df$Shameerpet,na.rm=TRUE))

df$Panjagutta[which(is.na(df$Panjagutta))]<-
as.integer(mean(df$Panjagutta,na.rm=TRUE))

df$ZooPark[which(is.na(df$ZooPark))]<-
as.integer(mean(df$ZooPark,na.rm=TRUE))

df$Bollaram[which(is.na(df$Bollaram))]<-
as.integer(mean(df$Bollaram,na.rm=TRUE))

df$ICRISAT[which(is.na(df$ICRISAT))]<-
as.integer(mean(df$ICRISAT,na.rm=TRUE))

is.na(df)

```

#For Clustering, we need unlabelled data

```
mydata=select(df,c(2:26))
```

#Getting the transpose of the data for forming clusters

```
mydatat<-as.data.frame(t(mydata))

colnames(mydatat)<-c(date)
```

#Finding Optimum m=number of Clusters

#Method 1

#WSS Plot Function

```
wssplot <- function(data, nc=15, seed=1234){

  wss <- (nrow(data)-1)*sum(apply(data,2,var))
```

```
for (i in 2:nc){\n  set.seed(seed)\n\n  wss[i] <- sum(kmeans(data, centers=i)$withinss)}\n\n  plot(1:nc, wss, type="b", xlab="Number of Clusters",\n    ylab="Within groups sum of squares")\n\n  wss\n}\n\nwssplot(mydatat)\n\n#Method 2\n\nset.seed(1234)\n\nnc <- NbClust(mydata, min.nc=2, max.nc=15, method="kmeans")\n\nKM = kmeans(mydatat,3)\n\n#KM1= kmeans(mydata,3)\n\nfviz_cluster(KM,data=mydatat)\n\n#autoplot(KM1,mydata,frame=TRUE)\n\nKM$centers\n\nKM$cluster\n\n#Plots of Cluster 1\n\nc1plot <- ggplot(data=df,mapping=aes(x=Date))+
```

```

geom_line(mapping=aes(y=KBRNPark,color="KBRNPark"),linewidth=0.5)+

geom_line(mapping=aes(y=Madhapur,color="Madhapur"),linewidth=0.5)+

geom_line(mapping=aes(y=MGBS,color="MGBS"),linewidth=0.5)+

geom_line(mapping=aes(y=Chikkadapally,color="Chikkadapally"),linewidth=0.5)+

geom_line(mapping=aes(y=Rajendranagar,color="Rajendranagar"),linewidth=0.5)+

geom_line(mapping=aes(y=Sainikpuri,color="Sainikpuri"),linewidth=0.5)+

geom_line(mapping=aes(y=BPPA,color="BPPA"),linewidth=0.5)+

geom_line(mapping=aes(y=Shameerpet,color="Shameerpet"),linewidth=0.5)+

theme_bw()+
  labs(title=paste("Cluster 1 Plots"))+
  xlab("Date")+
  ylab('AQI')

ggplotly(c1plot)

```

#Plots of Cluster 2

```

c2plot <- ggplot(data=df,mapping=aes(x=Date))+

  geom_line(mapping=aes(y=Balanagar,color="Balanagar"),linewidth=0.5)+

  geom_line(mapping=aes(y=Uppal,color="Uppal"),linewidth=0.5)+

  geom_line(mapping=aes(y=JubileeHills,color="JubileeHills"),linewidth=0.5)+

  geom_line(mapping=aes(y=Paradise,color="Paradise"),linewidth=0.5)+

  geom_line(mapping=aes(y=Charminar,color="Charminar"),linewidth=0.5)+

  geom_line(mapping=aes(y=Jeedimetla,color="Jeedimetla"),linewidth=0.5)+
```

```

geom_line(mapping=aes(y=Abids,color="Abids"),linewidth=0.5)+

geom_line(mapping=aes(y=LangarHouse,color="LangarHouse"),linewidth=0.5)+

geom_line(mapping=aes(y=Kukatpally,color="Kukatpally"),linewidth=0.5)+

geom_line(mapping=aes(y=Nacharam,color="Nacharam"),linewidth=0.5)+

geom_line(mapping=aes(y=HCU,color="HCU"),linewidth=0.5)+

geom_line(mapping=aes(y=Panjagutta,color="Panjagutta"),linewidth=0.5)+

theme_bw()+
  labs(title=paste("Cluster 2 Plots"))+
  xlab("Date")+
  ylab('AQI')

ggplotly(c2plot)

```

#Plots of Cluster 3

```

c3plot <- ggplot(data=df,mapping=aes(x=Date))+

geom_line(mapping=aes(y=Sanathnagar,color="Sanathnagar"),linewidth=0.5)+

geom_line(mapping=aes(y=ZooPark,color="ZooPark"),linewidth=0.5)+

geom_line(mapping=aes(y=Pashamylaram,color="Pashamylaram"),linewidth=0.5)+

geom_line(mapping=aes(y=Bollaram,color="Bollaram"),linewidth=0.5)+

geom_line(mapping=aes(y=ICRISAT,color="ICRISAT"),linewidth=0.5)+

theme_bw()+
  labs(title=paste("Cluster 3 Plots"))+

```

```

xlab("Date")+
ylab('AQI')

ggplotly(c3plot)

```

#Pie Chart of Cluster 1

```

x1<-
c(mean(df$KBRNPark),mean(df$Madhapur),mean(df$MGBS),mean(df$Chikkadapal
ly),mean(df$Rajendranagar),mean(df$Sainikpuri),mean(df$BPPA),mean(df$Shameer
pet))

pct=round(x1/sum(x1)*100)

names1<-c("KBRNPark", "Madhapur", "MGBS",
"Chikkadapally","Rajendranagar","Sainikpuri","BPPA","Shameerpet")

label1<-paste(names1,"-",pct,"%",sep="")

col1<-c("#FFBBA9", "#FFF74", "#9AFF4D",
"#A05FFF","#FF5067","#6139FF","#71FFE3","#0EFF0E")

pie3D(x1, labels = label1, main = "Cluster 1 AQI Average
Distribution",col=col1,explode=0.1)

legend("bottom", label1, fill = col1)

```

#Pie Chart of Cluster 2

```

x2<-
c(mean(df$Balanagar),mean(df$Uppal),mean(df$JubileeHills),mean(df$Paradise),me
an(df$Charminar),mean(df$Jeedimetla),mean(df$Kukatpally),mean(df$Panjagutta))

pct2=round(x2/sum(x2)*100)

names2<-c("Balanagar", "Uppal", "JubileeHills",
"Paradise","Charminar","Jeedimetla","Kukatpally","Panjagutta")

```

```

label2<-paste(names2,"-",pct2,"%",sep="")

col2<-c("#FF8BE4", "#FF567D", "#68FF8B",
"##F7FF2A","#FF7437","#BBFF04","#A0FFA0","#5DC9FF")

pie3D(x2, labels = label2, main = "Cluster 2 AQI Average
Distribution",col=col1,explode=0.1)

legend("bottom", label2, fill = col2)

#Pie Chart of Cluster 3

x3<-
c(mean(df$Sanathnagar),mean(df$ZooPark),mean(df$Pashamylaram),mean(df$Bolla
ram),mean(df$ICRISAT))

pct3=round(x3/sum(x3)*100)

names3<-c("Sanathnagar", "ZooPark", "Pashamylaram", "Bollaram","ICRISAT")

label3<-paste(names3,"-",pct3,"%",sep="")

col3<-c("#BEF7FF", "#FFA000", "#8BFF34", "#A05FFF","#FF5067")

pie3D(x3, labels = label3, main = "Cluster 3 AQI Average
Distribution",col=col3,explode=0.1 )

legend("bottom", label3, fill = col3)

rowMeans(KM$centers)

mean(df$Chikkadapally)

mean(df$Charminar)

mean(df$Bollaram)

cl1_model<-lm(df$Chikkadapally~date)

```

```

predict(cl1_model,newdata=data.frame(date=seq(as.Date("2023-01-
01"),as.Date("2023-12-01"),by="month")),interval="confidence")

ggplot(df,aes(x=Date,y=Chikkadapally))+geom_point()+geom_smooth(method="lm"
)

cl2_model<-lm(df$Charminar~date)

predict(cl2_model,newdata=data.frame(date=seq(as.Date("2023-01-
01"),as.Date("2023-12-01"),by="month")),interval="confidence")

ggplot(df,aes(x=Date,y=Charminar))+geom_point()+geom_smooth(method="lm")

cl3_model<-lm(df$Bollarlam~date)

predict(cl3_model,newdata=data.frame(date=seq(as.Date("2023-01-
01"),as.Date("2023-12-01"),by="month")),interval="confidence")

ggplot(df,aes(x=Date,y=Bollarlam))+geom_point()+geom_smooth(method="lm")

aqi<-cbind(aqi2016$Avg,aqi2017$Avg)

aqi<-cbind(aqi,aqi2018$Avg)

aqi<-cbind(aqi,aqi2019$Avg)

aqi<-cbind(aqi,aqi2020$Avg)

aqi<-cbind(aqi,aqi2021$Avg)

aqi<-cbind(aqi,aqi2022$Avg)

aqi

noRows<-25

for(i in 1:noRows){

  if(sum(is.na(aqi[i,]))){

    aqi[i,][is.na(aqi[i,])] <- mean(aqi[i,], na.rm = TRUE)
}

```

```

        }

    }

aqi<-rowMeans(aqi)

aqi

place <-c(aqi2016[,1])

df2<-data.frame(
  Location<-place,
  aqi_overall_avg <-aqi
)

df2

aqi_range<-c(ifelse(aqi_overall_avg>400,"Severe(>400)",
ifelse(aqi_overall_avg>300,"Very poor(301-400)",
ifelse(aqi_overall_avg>200,"Poor(201-300)",
ifelse(aqi_overall_avg>100,"Moderate(101-200)",
ifelse(aqi_overall_avg>50,"Satisfactory(51-100)","Good(0-
50)")))))

ggplot(df2,aes(x=Location,y=aqi_overall_avg,fill=aqi_range))+

  labs(title = "Overall Mean AQI for Different Places in Hyderabad",face="bold")+

  geom_bar(stat = "identity")+

  scale_fill_manual(values=c('#F9FF50','#25D84A'))+

  theme_bw()+
  coord_flip()

```

2.3 Demonstration

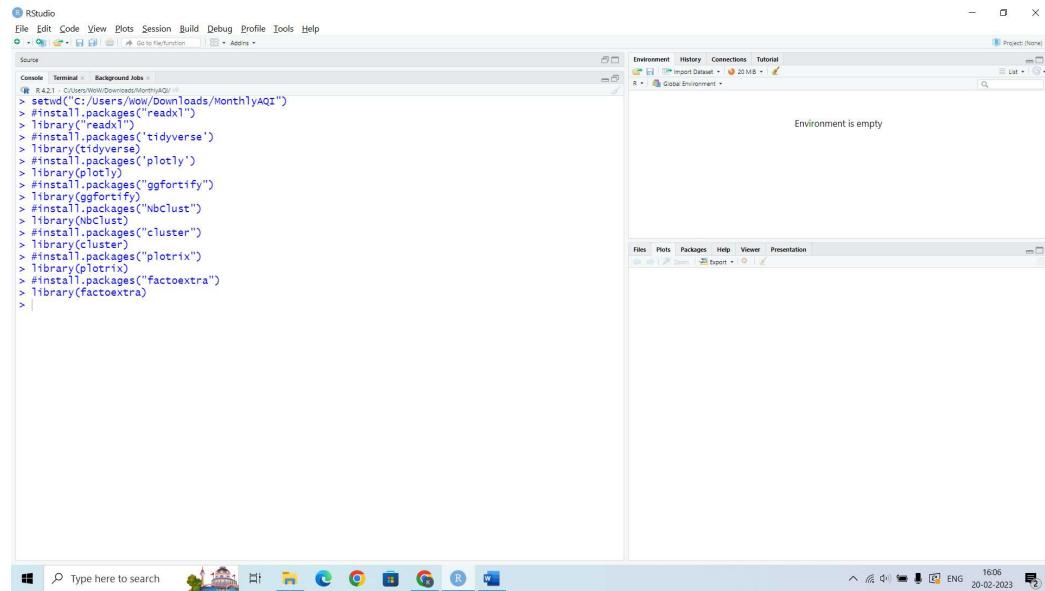


Fig 2.1: Screenshot of Setting the Directory and Installing, Loading Packages

First, the Working Directory was set to the required location. Then, the packages READXL, TIDYVERSE, PLOTLY, GGFORTIFY, NBCLUST, CLUSTER, PLOTRIX, FACTOEXTRA were loaded to the RStudio Environment. Plotly for Interactive Charts, Readxl for loading the excel sheets; Tidyverse, a collection all R packages for data science, GGPLOT for drawing the plots. Nbclust was used for finding the optimum number of clusters, Plotrix for 3d Pie Plots and Factoextra for the plotting the clusterplot.

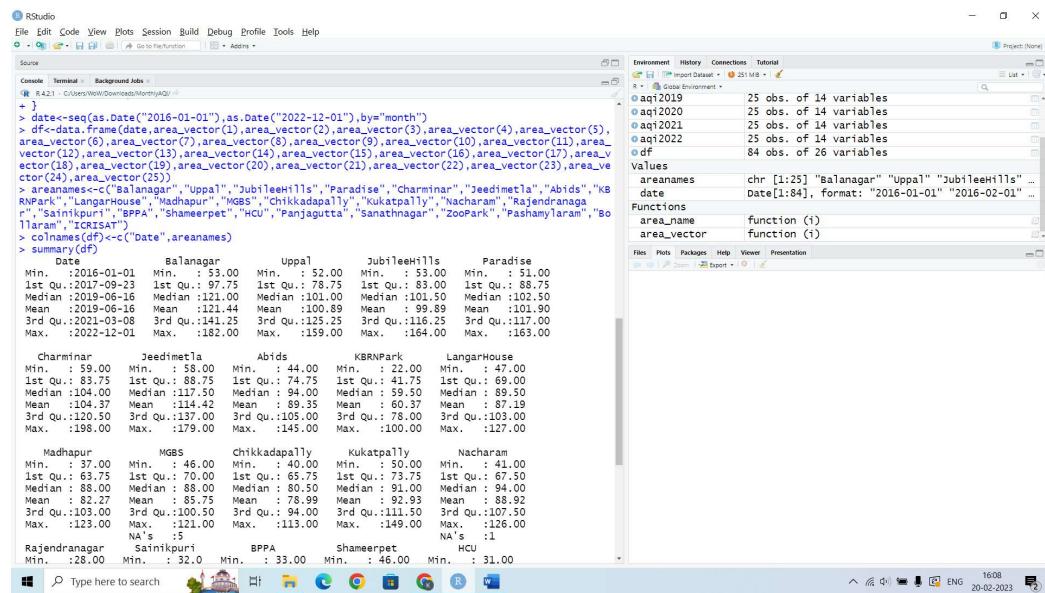
The screenshot shows the RStudio interface with the code editor containing the following R script:

```
> #Reading the excel file
> aq12016<-read_excel("AQI2016.xlsx")
> aq12017<-read_excel("aqi2017.xlsx")
> aq12018<-read_excel("aqi2018.xlsx")
> aq12019<-read_excel("aqi2019.xlsx")
> aq12020<-read_excel("aqi2020.xlsx")
> aq12021<-read_excel("aqi2021.xlsx")
> aq12022<-read_excel("aqi2022.xlsx")
> #Creating a function
> area<-function() {
+   aq<-as.data.frame(aq12016[,1])
+   area<-sLocation
+   return(area)
+ }
> #Creating a vector of AQ of the Area
> area_vector<-function() {
+   aq12016df<-as.data.frame(aq12016[,2:13])
+   aq12017df<-as.data.frame(aq12017[,2:13])
+   aq12018df<-as.data.frame(aq12018[,2:13])
+   aq12019df<-as.data.frame(aq12019[,2:13])
+   aq12020df<-as.data.frame(aq12020[,2:13])
+   aq12021df<-as.data.frame(aq12021[,2:13])
+   aq12022df<-as.data.frame(aq12022[,2:13])
+   aq12018vec<-c(aq12018$Jan,aq12018$Feb,aq12018$Mar,aq12018$Apr,aq12018$May,aq12018$Jun,aq12018$Jul,aq12018$Aug,aq12018$Sep,aq12018$Oct,aq12018$Nov,aq12018$Dec)
+   aq12019vec<-c(aq12019$Jan,aq12019$Feb,aq12019$Mar,aq12019$Apr,aq12019$May,aq12019$Jun,aq12019$Jul,aq12019$Aug,aq12019$Sep,aq12019$Oct,aq12019$Nov,aq12019$Dec)
+   aq12020vec<-c(aq12020$Jan,aq12020$Feb,aq12020$Mar,aq12020$Apr,aq12020$May,aq12020$Jun,aq12020$Jul,aq12020$Aug,aq12020$Sep,aq12020$Oct,aq12020$Nov,aq12020$Dec)
+   aq12021vec<-c(aq12021$Jan,aq12021$Feb,aq12021$Mar,aq12021$Apr,aq12021$May,aq12021$Jun,aq12021$Jul,aq12021$Aug,aq12021$Sep,aq12021$Oct,aq12021$Nov,aq12021$Dec)
+   aq12022vec<-c(aq12022$Jan,aq12022$Feb,aq12022$Mar,aq12022$Apr,aq12022$May,aq12022$Jun,aq12022$Jul,aq12022$Aug,aq12022$Sep,aq12022$Oct,aq12022$Nov,aq12022$Dec)
+   suppressWarnings(aqaq<-as.integer(cc(aq12018vec,aq12017vec,aq12018vec,aq12019vec,aq12020vec,aq12021vec,aq12022vec)))
+   return(aqaq)
}
```

In the top-right panel, under 'Environment', it shows the loaded datasets: Data (aq12016, aq12017, aq12018, aq12019, aq12020, aq12021, aq12022) and Functions (area_name, area_vector). At the bottom right of the interface, the status bar shows '16:07 20-02-2023'.

Fig 2.2: Reading the Excel Files and extracting required values to Vectors

On loading all the excel files are in the form of tibbles. Tibbles are converted to dataframe by using `as.data.frame()`. All the monthly AQI of each area can be converted into a vector by using the '\$.' Instead of loading all the area datafields separately, only the required Area AQI values are converted by creation of a Function called `area_vector()`. The area vector accepts an integer value as input and provides the data of the area which is present in that row of the Excel Files. The `area_name()` function returns the name of the area present in that row of the Excel File.



```

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Console Terminal Background Jobs
Source
File Edit Code View Plots Session Build Debug Profile Tools Help
Console Terminal Background Jobs
+ }
> date<-seq(as.Date("2016-01-01"),as.Date("2022-12-01"),by="month")
> df<-data.frame(area_vector(1),area_vector(2),area_vector(3),area_vector(4),area_vector(5),
  area_vector(6),area_vector(7),area_vector(8),area_vector(9),area_vector(10),area_vector(11),area_
vector(12),area_vector(13),area_vector(14),area_vector(15),area_vector(16),area_vector(17),area_v
ector(18),area_vector(19),area_vector(20),area_vector(21),area_vector(22),area_vector(23),area_v
ector(24),area_vector(25))
> areas<-c("Balanagar","Uppal","JubileeHills","Paradise","Jeedimetla","Abids","KBRNPark",
  "LangarHouse","Madhapur","MGBS","Chikkadapally","Kukatpally","Nacharam","Rajendranaga
r","Sainikpuri","BPPA","Shameerpet","HCU","Panjagutta","Sanathnagar","ZooPark","Pashamylaram","Bo
Tlaram","ICRISAT")
> colnames(df)<-c("Date",areas)
> summary(df)

      Date        Balanagar      Uppal       JubileeHills    Paradise
Min. :2016-01-01  Min. : 53.00  Min. : 52.00  Min. : 53.00  Min. : 51.00
1st Qu.:2017-09-23  1st Qu.: 97.75  1st Qu.: 78.75  1st Qu.: 83.00  1st Qu.: 88.75
Median :2019-06-16  Median :121.00  Median :101.00  Median :101.50  Median :102.50
Mean   :2019-06-16  Mean   :121.44  Mean   :100.89  Mean   : 99.89  Mean   :101.90
3rd Qu.:2021-03-08  3rd Qu.:141.25  3rd Qu.:125.25  3rd Qu.:116.25  3rd Qu.:117.00
Max.  :2022-12-01  Max.  :182.00  Max.  :159.00  Max.  :164.00  Max.  :163.00

      Jeedimetla     Abids      KBRNPark Langarhouse
Min. : 59.00  Min. : 58.00  Min. : 44.00  Min. : 22.00
1st Qu.: 83.75  1st Qu.: 88.75  1st Qu.: 74.75  1st Qu.: 41.75
Median :104.00  Median :117.50  Median : 94.00  Median : 59.50
Mean   :104.37  Mean   :114.42  Mean   : 89.35  Mean   : 87.19
3rd Qu.:120.50  3rd Qu.:137.00  3rd Qu.:105.00  3rd Qu.: 78.00
Max.  :198.00  Max.  :179.00  Max.  :145.00  Max.  :100.00
NA's   :5          NA's   :5          NA's   :1          NA's   :1

      Madhapur      MGBS Chikkadapally Kukatpally Nacharam
Min. : 37.00  Min. : 46.00  Min. : 40.00  Min. : 50.00  Min. : 41.00
1st Qu.: 63.75  1st Qu.: 70.00  1st Qu.: 65.75  1st Qu.: 73.75  1st Qu.: 67.50
Median : 82.00  Median : 88.00  Median : 80.50  Median : 91.00  Median : 90.00
Mean   : 82.37  Mean   : 88.00  Mean   : 80.50  Mean   : 91.33  Mean   : 88.92
3rd Qu.:103.00  3rd Qu.:100.50  3rd Qu.: 94.00  3rd Qu.:111.50  3rd Qu.:107.50
Max.  :123.00  Max.  :121.00  Max.  :113.00  Max.  :149.00  Max.  :126.00
NA's   :5          NA's   :5          NA's   :5          NA's   :1

      Rajendranagar Sainikpuri BPPA Shameerpet HCU
Min. :28.00  Min. : 32.00  Min. : 33.00  Min. : 46.00  Min. : 31.00

```

Fig 2.3: Screenshot of Function for creating a dataframe with all the areas AQI Values

All the data related to the area wise AQI are loaded into a dataframe named 'df'. The column names of the dataframe are set using `thecolnames()` function. The summary function returns the summary of the dataframe about the number of missing values, first quartile, median, mean and third quartile of the data and the minimum and maximum values.

The screenshot shows the RStudio interface with the console tab active. The code in the console is as follows:

```

R> #4.2.1: CleaningWIOpenSourceNormalQ()
> df$NGBS[which(is.na(df$NGBS))<-as.integer(mean(df$NGBS,na.rm=TRUE))]
> df$Nacharam[which(is.na(df$Nacharam))<-as.integer(mean(df$Nacharam,na.rm=TRUE))]
> df$Sainikpuri[which(is.na(df$Sainikpuri))<-as.integer(mean(df$Sainikpuri,na.rm=TRUE))]
> df$Shameerpet[which(is.na(df$Shameerpet))<-as.integer(mean(df$Shameerpet,na.rm=TRUE))]
> df$Panjagutta[which(is.na(df$Panjagutta))<-as.integer(mean(df$Panjagutta,na.rm=TRUE))]
> df$ZooPark[which(is.na(df$ZooPark))<-as.integer(mean(df$ZooPark,na.rm=TRUE))]
> df$Bollaram[which(is.na(df$Bollaram))<-as.integer(mean(df$Bollaram,na.rm=TRUE))]
> df$ICRISAT[which(is.na(df$ICRISAT))<-as.integer(mean(df$ICRISAT,na.rm=TRUE))]
> is.na(df)

```

The Data frame Balanagar uppal JubileeHills Paradise charminar Jeedimetla Abids KBRNPark has been modified to replace missing values with the mean of each area's AQI values.

Fig 2.4: Screenshot of the console during the handling of Missing Values

The missing values in each area are replaced with the mean of AQI Values of each area. After replacing, it is checked if there are still any missing values using the is.na() function.

The screenshot shows the RStudio interface with the Environment tab active, displaying the final cleaned DataFrame. The table has columns for date, area names, and their respective AQI values. The rows show data for various dates across different areas like Balanagar, Uppal, JubileeHills, etc.

Date	Balanagar	Uppal	JubileeHills	Paradise	charminar	Jeedimetla	Abids	KBRNPark	Lengashouse	Muthuraj	MGITS	Chikkadpally	Kukatpally	Nacharam	Rajendranagar	Sainikpuri	BPPA	Shameerpet	HCU	Pangutta	Santacruz	ZooPark
2016-01-01	162	112	106	125	122	150	123	80	113	94	79	87	96	85	85	77	82	100	115	131		
2016-02-01	175	101	123	130	119	122	66	66	88	80	81	82	134	75	87	76	64	89	104	96		
2016-03-01	116	95	105	125	104	122	111	78	107	88	95	86	76	120	67	85	70	83	101	112	108	
2016-04-01	123	92	106	127	110	114	115	64	86	82	93	74	83	117	71	71	52	76	91	105	93	
2016-05-01	103	86	100	119	98	107	90	45	70	68	71	66	79	106	65	78	66	62	61	106	93	
2016-06-01	102	90	92	112	97	91	77	42	69	58	66	74	79	100	64	79	53	77	46	105	56	
2016-07-01	92	72	78	82	83	75	84	36	65	55	68	65	67	94	71	66	42	65	49	105	39	
2016-08-01	103	94	98	104	95	99	95	49	64	54	54	46	59	92	50	49	47	55	41	104	39	
2016-09-01	64	65	64	93	71	72	69	30	54	42	47	61	63	39	37	62	39	71	52	107	40	
2016-10-01	109	103	109	112	100	112	98	54	73	65	73	87	91	100	66	88	57	74	100	107	92	
2016-11-01	161	127	117	132	144	159	123	65	79	98	81	96	125	119	75	80	79	82	158	107	158	
2016-12-01	155	144	164	159	170	155	126	76	109	102	68	105	126	75	110	103	60	159	107	159		
2017-01-01	155	146	117	134	122	172	145	83	101	115	106	119	116	91	121	104	96	133	107	175		
2017-02-01	160	147	143	131	157	163	131	81	115	104	120	102	109	118	88	121	82	117	128	107	166	
2017-03-01	155	129	126	126	142	154	120	75	110	88	111	95	117	104	82	122	76	129	106	107	115	
2017-04-01	162	159	160	163	159	179	113	93	119	102	119	104	129	118	76	99	86	64	132	107	118	
2017-05-01	166	103	131	137	171	149	80	74	92	108	105	90	94	126	72	79	62	56	94	107	62	
2017-06-01	132	72	113	97	84	88	74	45	68	47	46	51	66	76	56	72	39	58	45	107	43	
2017-07-01	75	95	88	82	94	84	41	71	39	57	61	76	60	35	62	43	51	46	107	36		
2017-08-01	119	103	103	75	102	89	61	34	69	43	80	59	70	51	38	68	43	58	52	107		
2017-09-01	122	65	107	81	113	125	79	63	73	108	106	72	81	77	53	71	45	73	64	107	58	
2017-10-01	154	94	103	103	91	71	87	77	101	80	68	108	51	67	66	48	96	96	107	125		
2017-11-01	138	123	114	108	134	126	91	82	111	103	81	79	139	39	74	68	43	96	107	133		
2017-12-01	134	126	123	138	142	119	94	113	108	96	85	149	107	42	43	90	100	64	141	107	236	
2018-01-01	141	140	128	129	144	134	126	100	118	111	102	98	143	114	60	90	37	62	146	107	228	
2018-02-01	128	136	134	124	127	140	122	89	127	99	106	99	130	105	55	64	76	74	105	107	134	
2018-03-01	127	126	123	117	117	126	111	87	112	105	110	96	126	133	59	66	86	65	106	107	121	
2018-04-01	128	104	106	101	125	96	77	90	104	88	99	120	113	55	66	61	65	93	107	83		
2018-05-01	120	107	116	117	103	125	105	79	98	96	93	90	111	113	60	74	73	57	93	107	68	
2018-06-01	107	86	108	103	108	97	96	52	89	72	72	81	99	107	48	74	67	54	56	107	45	
2018-07-01	69	84	100	82	90	82	89	52	75	71	74	75	82	88	45	67	53	69	41	107	32	
2018-08-01	97	81	100	93	97	105	90	50	84	67	66	82	78	93	63	70	59	68	41	107	33	
2018-09-01	123	91	106	101	111	125	85	61	102	94	97	94	96	90	72	77	75	70	72	107	60	

Fig 2.5: Screenshot of the Dataframe created

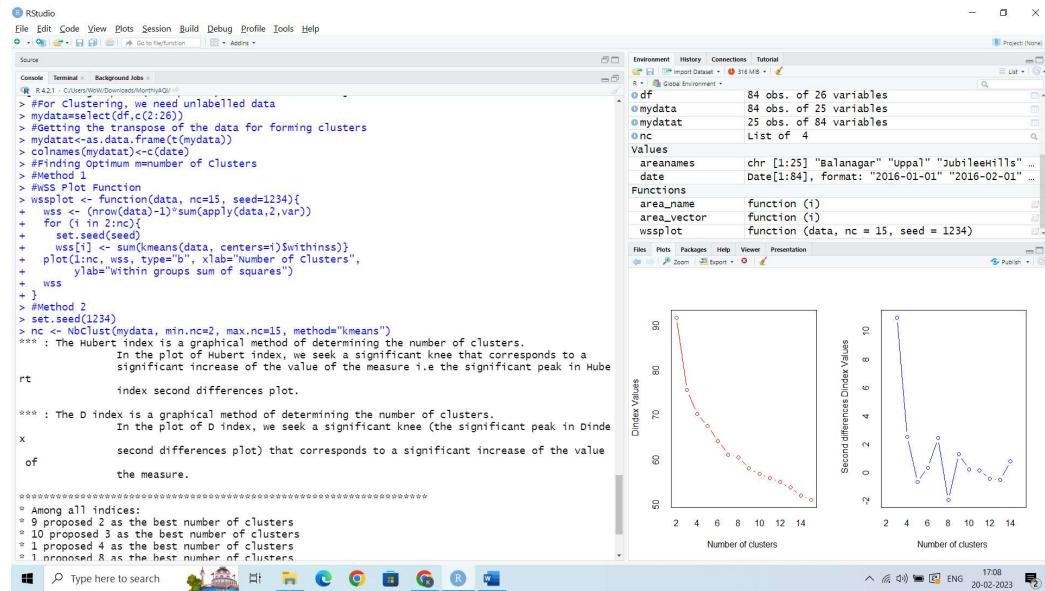


Fig 2.6: Screenshot of Function for finding the Optimum number of Clusters

The WSS Plot is used for checking the optimum number of clusters. The Nbclust function is also used for the same reason. The output was found to be 3 as a kink or an elbow shape appeared when the number of clusters was 3.

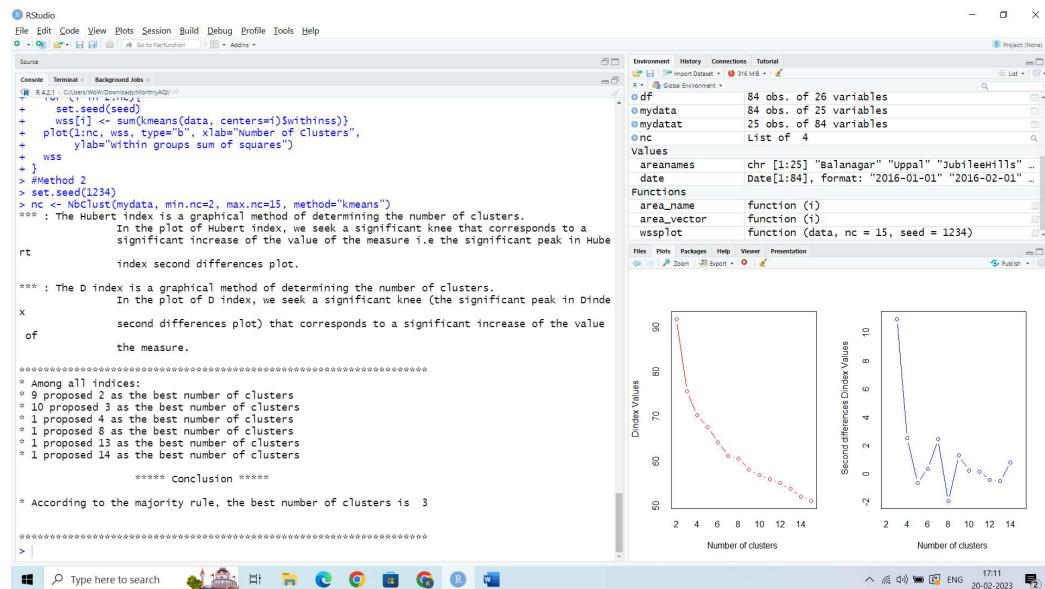


Fig 2.7: Screenshot of Output of NBCLUST Function



Fig 2.8: Screenshot of Representation of Three Clusters

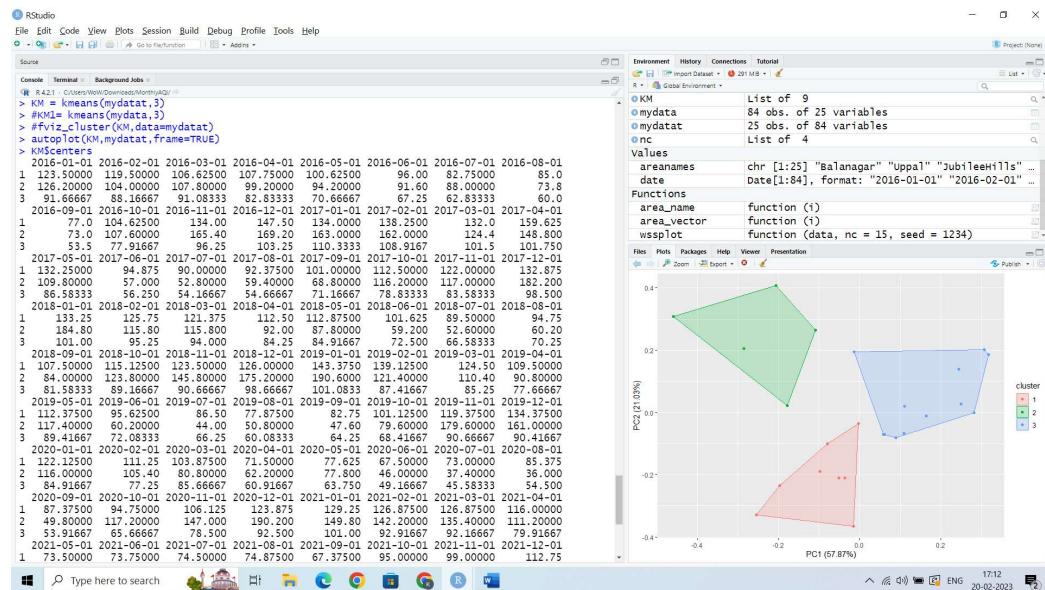


Fig 2.9: Screenshot of K Mean Cluster Centers

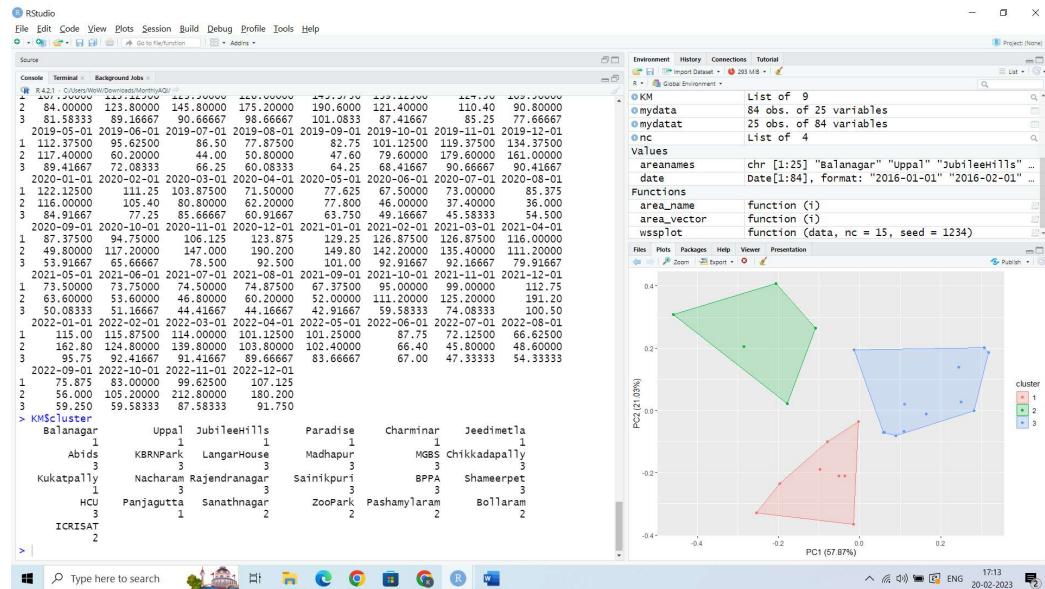


Fig 2.10: Screenshot of Division of areas into Clusters

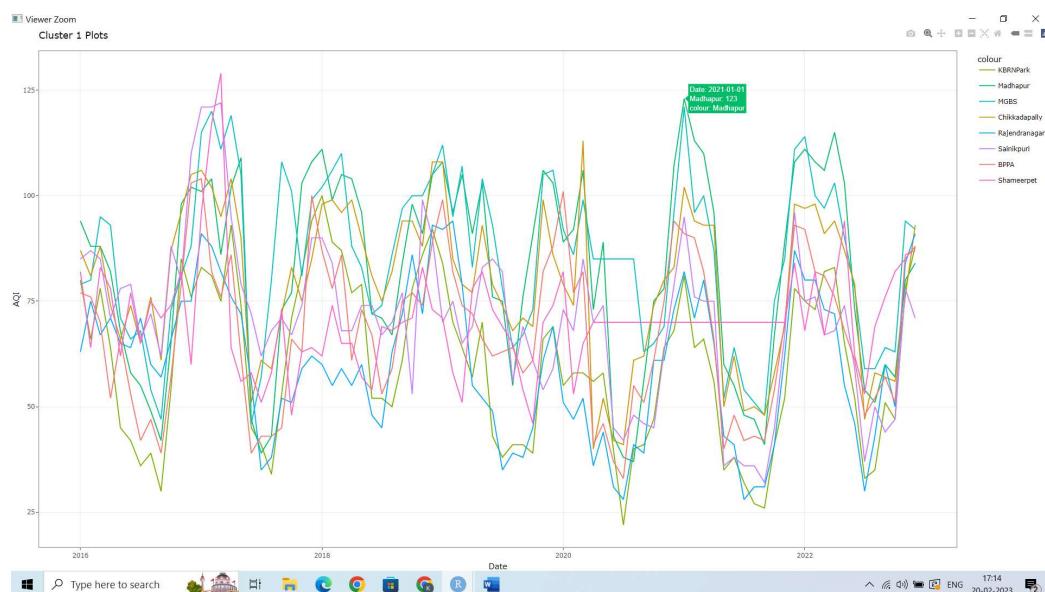


Fig 2.10: Screenshot of Interactive Plots of Areas under Cluster 1

The AQI Values v/s Date were plotted for all the areas that fall under the cluster 1. Similarly for cluster 2 and cluster 3, the plots are drawn. With these plots, it can be understood that the areas are rightly categorized into the particular clusters as the trends of these areas over years was almost similar.

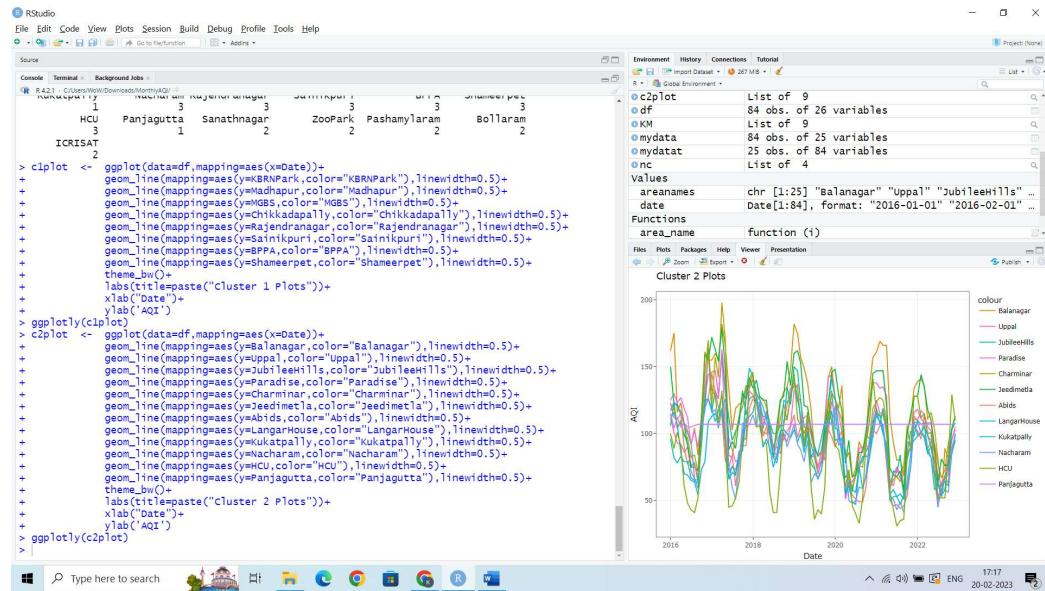


Fig 2.10: Screenshot of Console while plotting the AQI Plots of each cluster

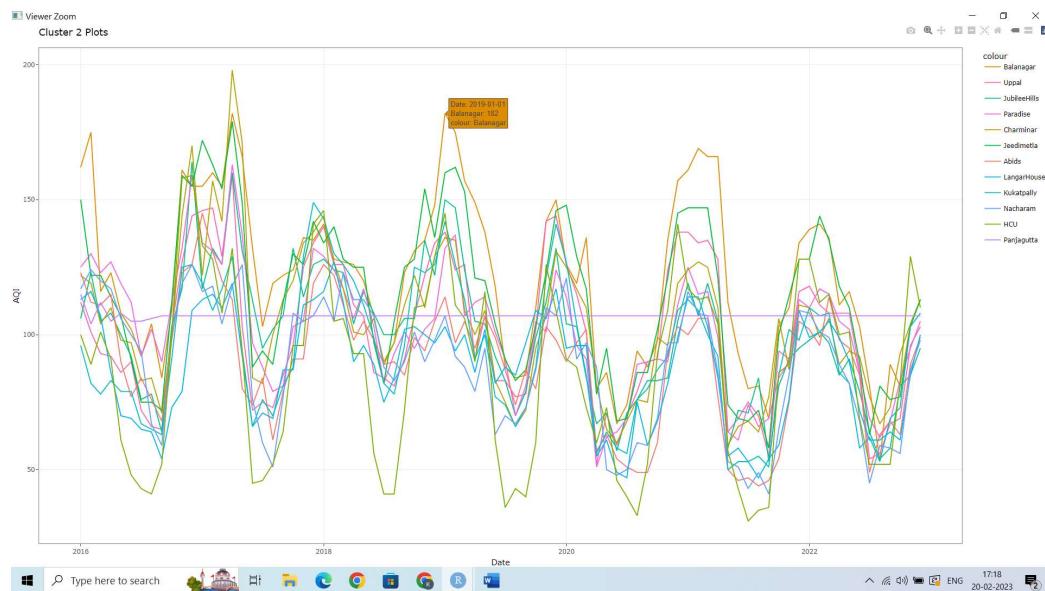


Fig 2.11: Screenshot of the Interactive Plots of Areas under Cluster 2

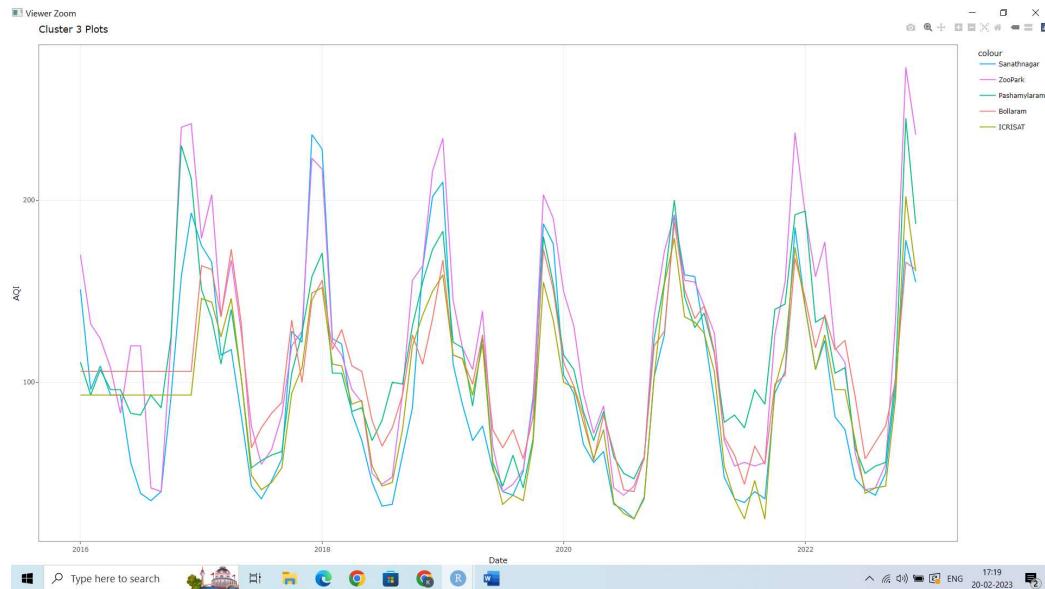


Fig 2.12: Screenshot of the Interactive Plots of Areas under Cluster 3

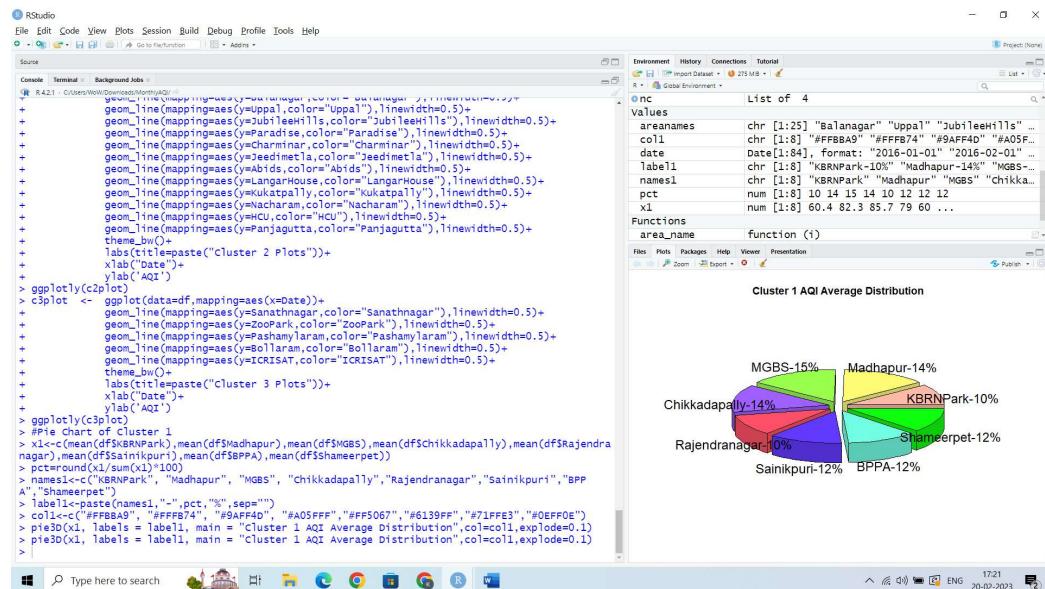
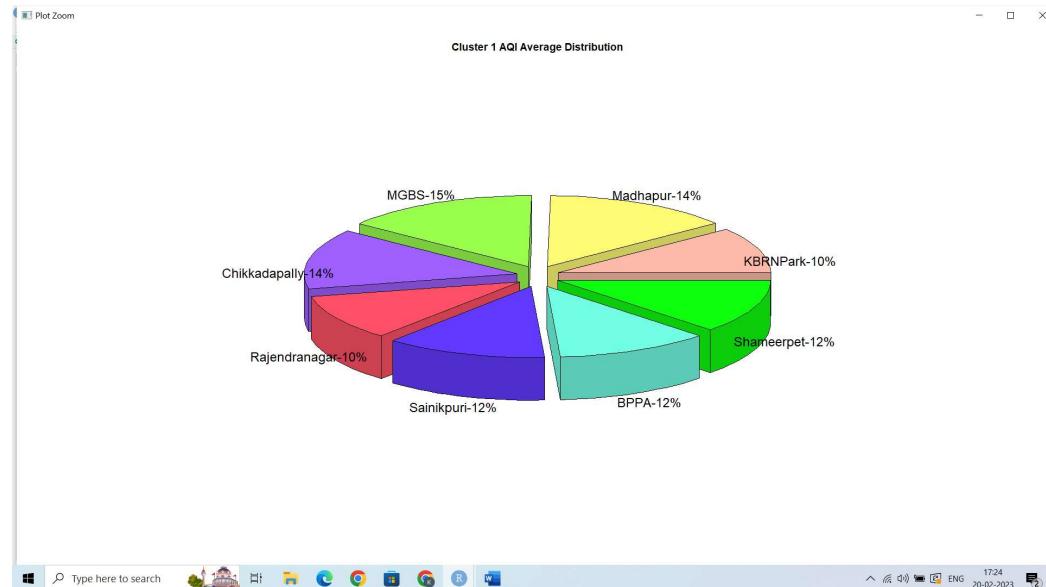


Fig 2.13: Screenshot of the Console while plotting the 3D Pie graphs

After plotting the Interactive AQI Values of the clusters, the distribution of each cluster were checked by plotting 3D Pie Plots.



. Fig 2.14: Screenshot of the Cluster 1 AQI Average Distribution in 3D Pie graphs

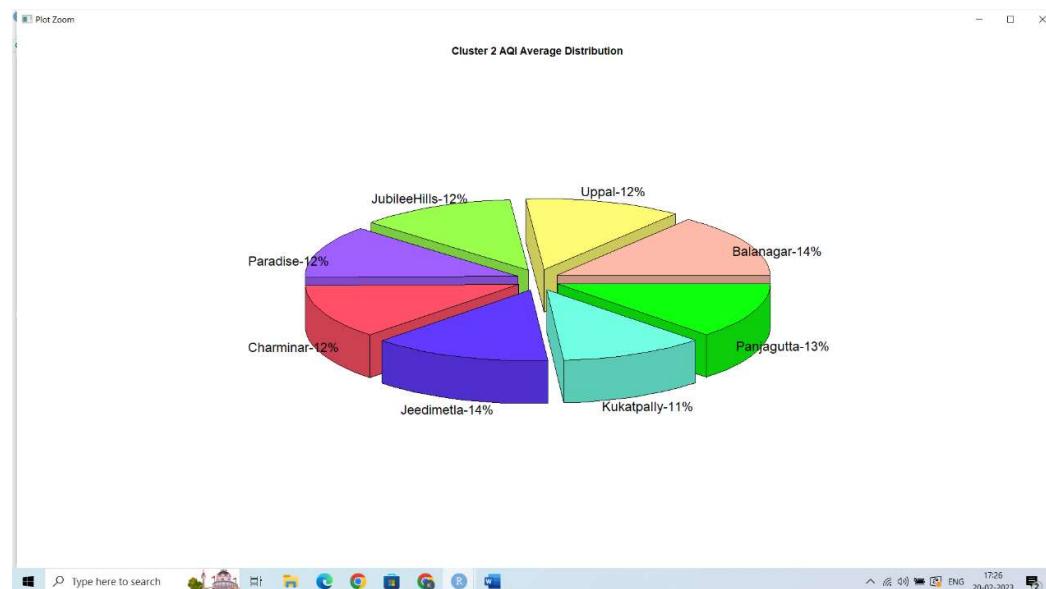


Fig 2.15: Screenshot of the Cluster 2 AQI Average Distribution in 3D Pie graphs

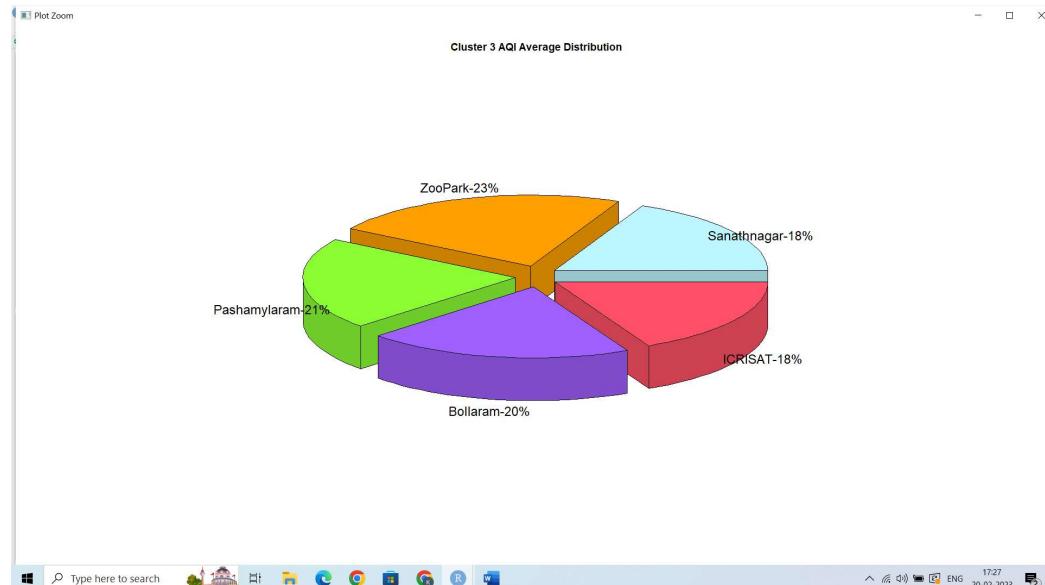


Fig 2.16: Screenshot of the Cluster 3 AQI Average Distribution in 3D Pie graphs

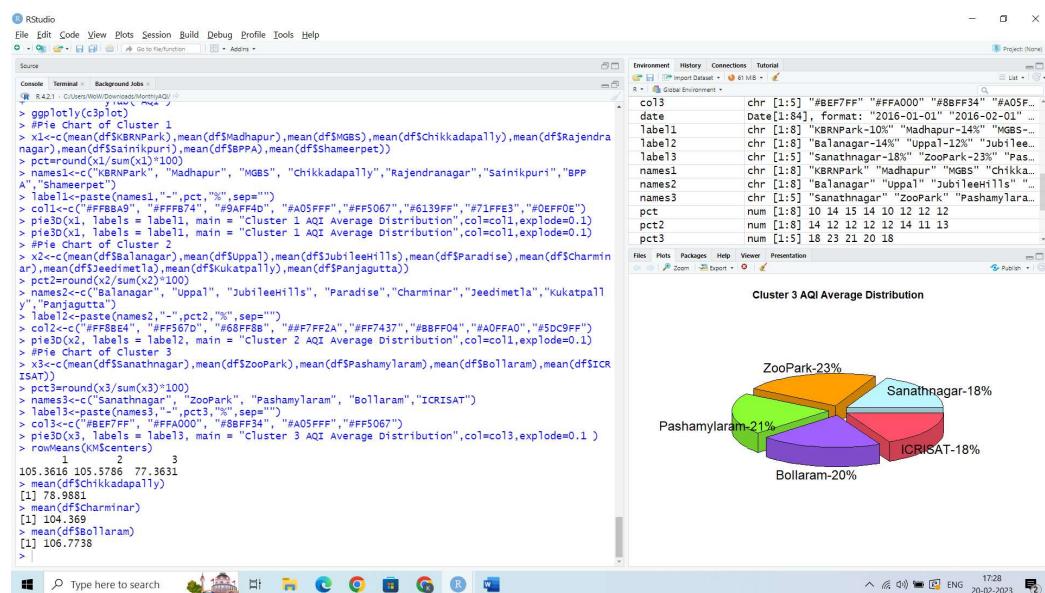


Fig 2.17: Screenshot of the after plotting the Cluster wise AQI Average Distribution in 3D Pie graphs

After plotting the 3D Pie Plots, the mean of each row of the cluster was calculated and on checking the nearest value of the areawise mean of AQI, the nearest value was chosen and it represents all the areas in that category.

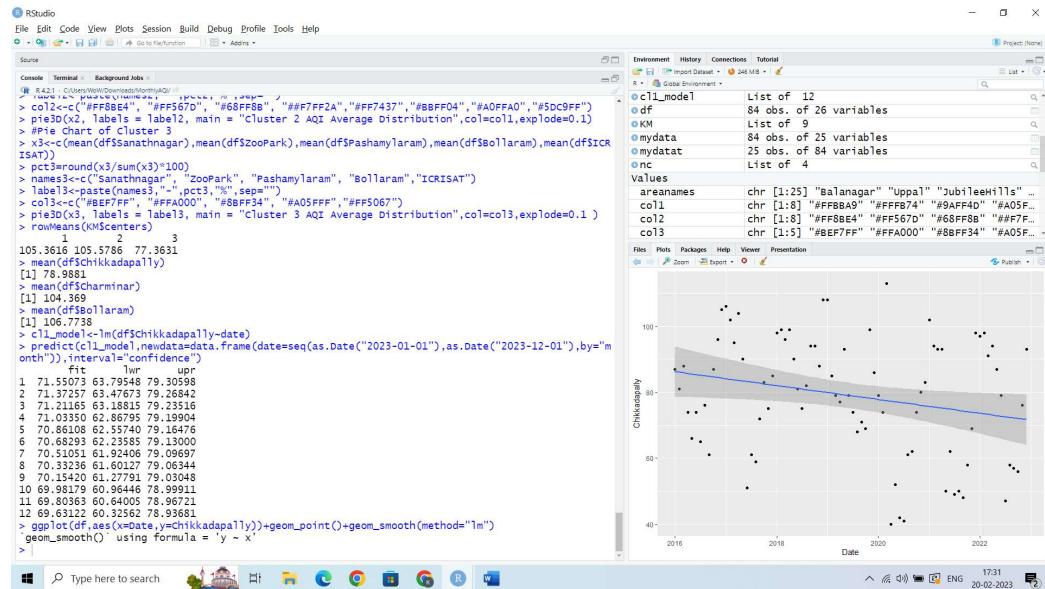


Fig 2.18: Screenshot of the Linear Model and Predicted Values and the Linear Model Plot

In the first Cluster, Chikkadapally was used to represent all the areas under the first cluster category. The linear model was created using the AQI Values and corresponding dates. Since, Linear Regression was beginner friendly and simple to use, a linear model was chosen to predict the future AQI values of the places in this area.

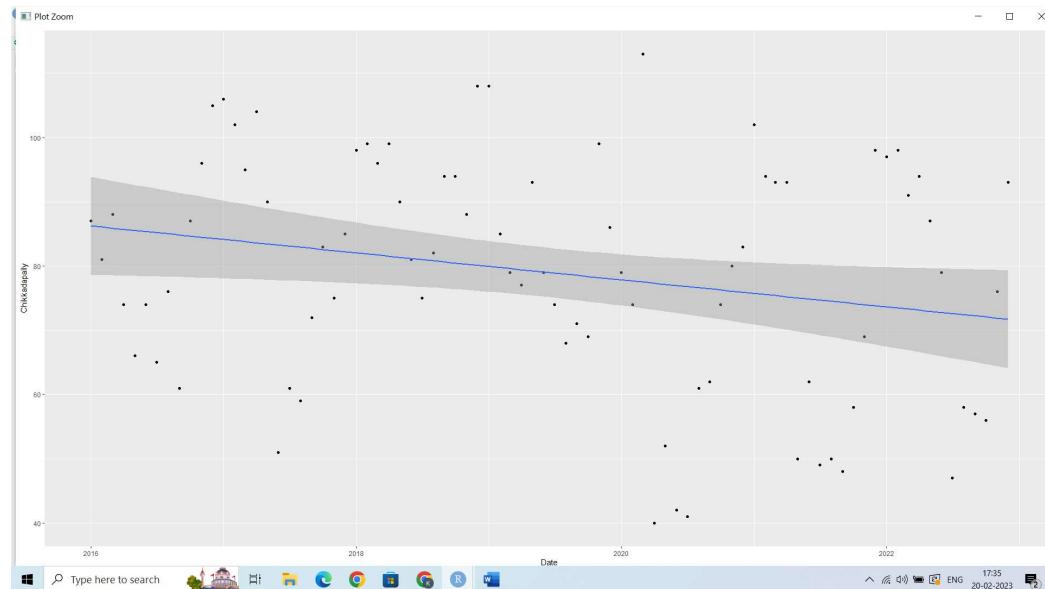


Fig 2.19: Screenshot of the Linear Model Plot of Chikkadapally Area Representing the areas in Cluster 1

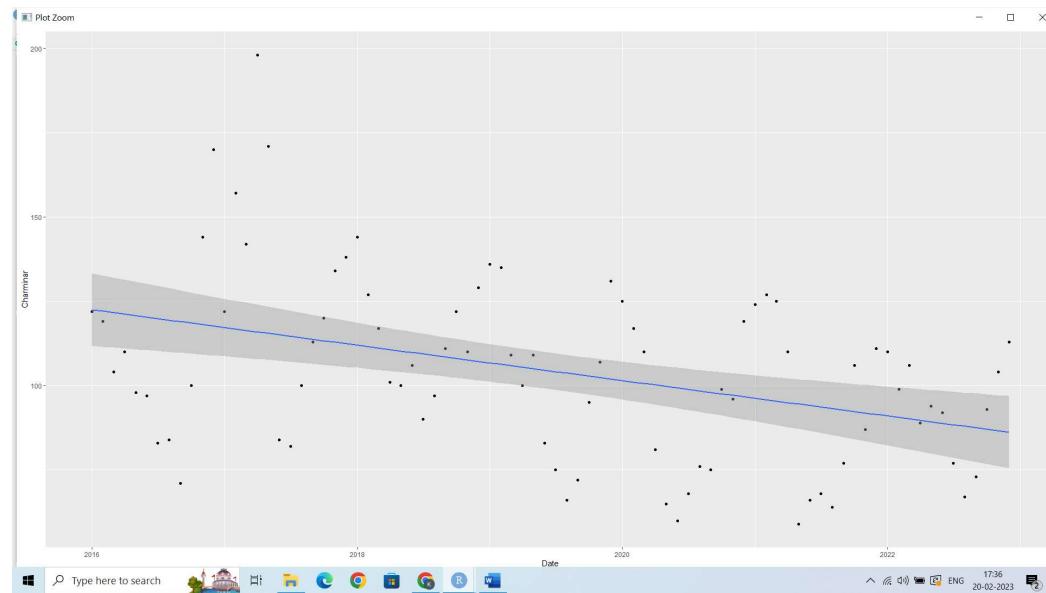


Fig 2.20: Screenshot of the Linear Model Plot of Charminar Area Representing the areas in Cluster 2

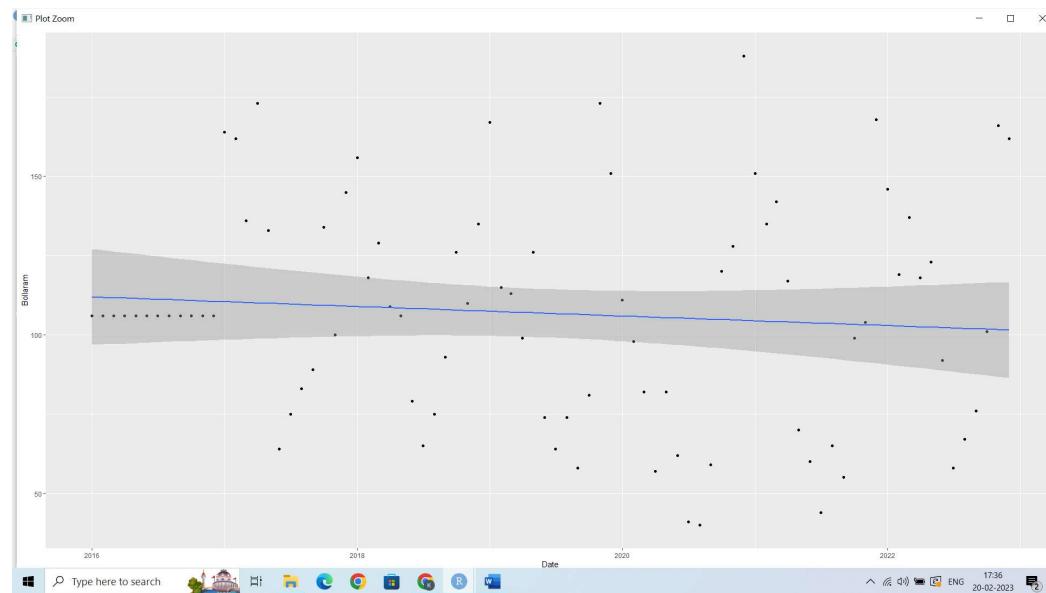


Fig 2.21: Screenshot of the Linear Model Plot of Bollaram Area Representing Cluster 3

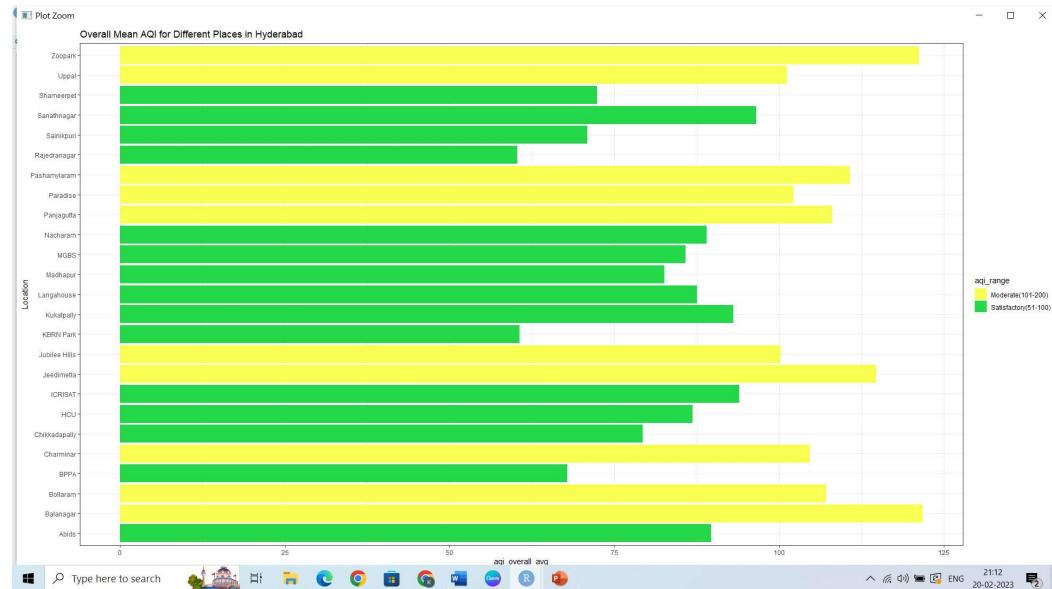


Fig 2.22: Screenshot of the Overall Mean AQI for Different Places in Hyderabad

CONCLUSION

The entire project has been developed to Analyse the Trends in the Air Quality Index of Hyderabad from 2016-2022. From the individual graphs, the values of all the areas during the year 2020 has shown a great fall in the Air Quality Index Values. This might be due to the prevalence Covid-19 and the lockdown imposed in the city. After 2020, during 2021 and 2022, the values are increasing as the lockdowns have been removed. But the values are seemingly less when compared to the data in the years 2016-2019.

Individual plots are drawn for the Linear Models which are generated for each area. By comparing the graphs, the areas Pashamylaram and Shameerpet have an increasing Linear Model which means the Air Quality Index Values would increase and the Air Quality in these areas would be decreasing. These areas are majorly industrial areas where the core reason for the pollution would be the emissions from the industries. Shameerpet has many pharmaceutical industries and the same was the case with Pashamylaram. Government must take actions regarding this, such as controlling the emissions from the industries, checking the Air Quality near these industries. Like the Effluent Treatment Plants, Industrial Air Treatment Plants must be setup in these areas.

Also in the remaining areas, measures must be taken to continue the Lower AQI readings. This could be done by providing awareness among people on the public transportation systems. If the commutators in the city travel by the means of Public Transportation instead of their using their own vehicles, the number of vehicles on road would reduce. Also, the Public Transportation System is to be able to capacitate the needs of all the civilians. Awareness must be provided to people that during the traffic signals, continuous ignition would release smokes from the vehicles which affects the Air Quality. Usage of Electric Vehicles must be encouraged by the government so that there would be an improvement in the Air Quality in the public areas.

Since, many missing values were found in the data, building efficient Linear Models was not possible. The Pollution Control Board must try to get all the readings effectively, so that Linear Models could be generated in all the areas and the graphs can be compared.

FUTURE SCOPE

In future, we can update this project by using efficient Regression Models for the comparison of Time Series Data. Furthermore, in the case of missing data fields, the linear model can be generated by just using the available values. These would also help in the comparison of area-wise Air Quality Data.

REFERENCES

1. Telangana State Pollution Control Board for the Data