**Lab-9**

**DS-670**

Neha Kumari

1. Complete reference to the IEEE or ACM paper you are trying to outperform.

<https://www.ijircce.com/upload/2015/sacaim/10_403.pdf>

1. Describe the results of the paper you have chosen.

The article uses K-means clustering to find the healthiest area in the city. It used 10 clusters to group the data and find the place where the ozone level concentrations are minimum. The results are summarized below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Ozone** | **Particulate matter** | **Carbon monoxide** | **Sulfur dioxide** | **Nitrogen dioxide** |
| **Cluster1** | 108.50388 | 130.46822 | 111.04294 | 113.57209 | 131.13068 |
| **Cluster2** | 114.71191 | 102.95795 | 112.14154 | 95.15947 | 93.84657 |
| **Cluster3** | 129.12724 | 118.28160 | 117.18714 | 97.55066 | 113.80135 |
| **Cluster4** | 115.90357 | 111.63613 | 138.91796 | 122.28279 | 112.76867 |
| **Cluster5** | 100.02959 | 107.48233 | 92.11272 | 116.41592 | 114.40066 |
| **Cluster6** | 102.93000 | 102.95874 | 119.27416 | 105.36514 | 122.55327 |
| **Cluster7** | **89.66607** | 118.30523 | 118.43308 | 105.96982 | 99.67516 |
| **Cluster8** | 119.45119 | 97.08002 | 109.04265 | 124.00820 | 121.91692 |
| **Cluster9** | 127.47206 | 114.86291 | 97.06541 | 119.05108 | 102.96825 |
| **Cluster10** | 109.20916 | 124.17216 | 118.14208 | 123.34767 | 102.08294 |

For cluster 7 the minimum ozone is found which is **89.66607**

1. Describe your results so far, and your next objective in terms of data analysis.

I did K-means clustering and Time series analysis using ARIMA model.

K-means:

K=10

avg\_ozone particulate monoxide sulfur nitrogen

1 106.93403 98.82974 102.8302 116.12671 118.26981

2 **85.06180** 125.69522 111.1593 100.80956 97.05194

3 116.35451 107.57061 115.8282 98.64193 96.15474

4 115.52903 107.40190 103.9809 117.67564 102.26439

5 113.37976 121.60003 105.7415 113.11510 116.78425

6 136.89854 120.92265 115.2964 108.97840 102.40764

7 119.59255 107.59357 123.5841 117.58886 116.02013

8 98.68995 117.36842 113.6513 121.99025 122.17094

9 120.76275 109.99191 107.9365 96.68356 117.24993

10 100.62492 110.05658 121.4600 110.97163 105.65457

For cluster 2 the minimum ozone is found which is 85.06180

Time series Analysis: I used ARIMA(1,1,0) model to forecast ozone levels for next day.

|  |  |
| --- | --- |
| Coefficients | -0.0176 |
| Standard Error | 0.0075 |
| AIC | 90854.27 |
| ME | -0.00231218 |
| RMSE | 3.211875 |
| MAE | 2.769806 |
| MPE | -0.1918289 |
| MAPE | 4.21961 |
| MASE | 1.0013 |
| Log likelihood | -45425.13 |

1. Describe how will you use the concepts of aggregation and group operations

In my data I have 449 files and each file captures the pollution levels for a particular location. We grouped the data for each location (by latitude and longitude)

The pollution data was aggregated and mean levels of each pollutant were taken, that mean level now represents the pollutant level at that place.

1. Create a table showing the code for every operation in the left column and the time measurements of every operation in the right column. You need at least 10 operations. Highlight in bold the lines of code where you use aggregation or group operations concepts.

|  |  |
| --- | --- |
| Operations | Time |
| **DF = sqlContext.sql("select avg(ozone) as avg\_ozone, avg(particullate\_matter) as particulate, avg(carbon\_monoxide) as monoxide, avg(sulfure\_dioxide) as sulfur, avg(nitrogen\_dioxide) as nitrogen from data2 group by longitude,latitude")** | 22 seconds |
| temp = list.files(pattern="\*.csv")  tbl = lapply(temp, read\_csv) %>% bind\_rows() | 26 seconds |
| Result10<-kmeans(x,10)  Result10$centers | 2seconds |
| mapgilbert <- get\_map(location = c(lon = mean(df1$lon), lat = mean(df1$lat)), zoom = 12, maptype = "terrain", scale = 1) | 3seconds |
| ggmap(mapgilbert) + geom\_point(data = df1, aes(x = lon, y = lat, fill = "red", alpha = 0.1), size = 5, shape = 21) + guides(fill=FALSE, alpha=FALSE, size=FALSE) | 4seconds |
| dat1$dat <- sapply(strsplit(as.character(dat1$timestamp), " "), "[", 1) | 4seconds |
| dat1$tim <- sapply(strsplit(as.character(dat1$timestamp), " "), "[", 2) | 4seconds |
| ft<-arima(ts1, order = c(1,1,0)) | 2seconds |
| plot(forecast(ft,h=500))  points(1:length(ts1),fitted(ft),type="l",col="green") | 5seconds |
| newdata <- subset(df, avg\_ozone== min\_ozone\_level,  select=c(longitude, latitude)) | 4seconds |