---

title: "SOA Challenge"

author: "Lai Wei"

date: "2023-03-05"

output: html\_document

---

```{r setup, include=FALSE}

knitr::opts\_chunk$set(echo = TRUE)

```

```{r}

rm(list = ls())

```

# Load Packages

```{r load\_packages, warning=FALSE, message=FALSE}

library(dplyr)

library(tidyverse)

library(ggplot2)

library(stats)

library(factoextra)

library(fitdistrplus)

library(glmnet)

```

# Import & View data

```{r , results='hide'}

raw\_data <- read.csv("hazard.csv")

# to store the original data as data to avoid any mistakes in future

data <- raw\_data

#assume 2020

fata\_amount <- 4760884

Injury\_amount <- 203910

```

```{r Data Cleaning}

#convert both injuries and fatalities into amounts

#data[8:9] <- sapply(data[8:9],as.numeric)

data <- data %>% mutate(Fatalities\_amt = fata\_amount\*Fatalities,

Injury\_amt = Injury\_amount\*Injuries,

total\_cost = fata\_amount\*Fatalities + Injury\_amount\*Injuries + OnLevel.PD)

#removed data before 1970

data <- data %>% filter(Year > 1969) %>%

mutate(Fatalities\_scale = scale(Fatalities\_amt),

Injuries\_scale = scale(Injury\_amt),

OnLevel.PD\_scale = scale(OnLevel.PD)) %>%

arrange(desc(OnLevel.PD))

#remove the first 2 outliers

data\_ <- data[-c(1:2),]

# Compute the correlation matrix

cor\_matrix <- cor(data\_[,c("Fatalities", "Injuries", "OnLevel.PD")])

# Print the correlation matrix

title <- "Correlation Matrix of Fatality, Injury, and onlevelled Property Damage"

cat("\n", title, "\n")

print(cor\_matrix)

```

```{r K-means to group data into 3 categories}

k\_mean\_region <- function(data, r, k = 3, damage){

if (r == 0) {

d <- data %>% dplyr::select(Fatalities\_scale ,Injuries\_scale, damage)

} else {

d <- data %>% filter(Region == r) %>%

dplyr::select(Fatalities\_scale ,Injuries\_scale, damage)

}

set.seed(1000)

hazard\_kmeans <- kmeans(d, centers = k)

f <- fviz\_cluster(hazard\_kmeans, data = d)

hazard\_clustered <- data %>% mutate(Cluster = factor(hazard\_kmeans$cluster))

return(list(visual = f, outp = hazard\_clustered))

}

```

```{r K-means to find}

#visualize original property damage

#visualize on-levelled property damage

out = k\_mean\_region(data\_, r=0, damage = "OnLevel.PD\_scale")

data\_o = out$outp

print(out$visual)

print(summary(out$outp))

data\_o <- data\_o %>% mutate(Type = ifelse(Cluster == 1, "Minor", ifelse(Cluster == 2, "Medium", "Major"))) %>%

dplyr::select(-Cluster)

print(table(data\_o$Type))

#write\_csv(as.data.frame(data\_o[,-c(11:12,14:16)]), "~/R output - DataWithType.csv")

```

```{r empirical\_distribution}

PD\_epdf <- function(data, r){

data <- data %>% filter(total\_cost > 0)

# print(data)

if (r == 0) {

d <- data

} else {

d <- data %>% filter(Region == r)}

print(d)

g = ggplot(d, aes(x = total\_cost)) +

geom\_density() +

scale\_x\_continuous(limits = c(0, 1000000))+

theme(plot.title = element\_text(hjust = 0.5, face = "bold"))

return(g)

}

print(PD\_epdf(data = data\_o, r = 0))

```

```{r fit GLM}

data\_glm <- data\_o %>% dplyr::select(total\_cost, Region, Quarter, Year, Hazard.Group, Type) %>%

filter(total\_cost > 0)

data\_glm[c(2:3,5:6)] <- sapply(data\_glm[c(2:3,5:6)], factor)

# fit gamma with lasso component

x <- model.matrix(total\_cost~., data = data\_glm)

y <- data\_glm$total\_cost

glm\_elastic <- cv.glmnet(x,log(y), data=data\_glm, family=Gamma(link = "log"), alpha =1, nfolds = 5)

r2 <- coef(glm\_elastic, s = glm\_elastic$lambda.min)

#write\_csv(output, "/Users/Lai\_Pro\_2021/Desktop/R output - glm.csv")

```

```{r project future inflation using ARIMA}

inflation <- read.csv("inflation.csv")

ts.inf <- ts(inflation$Inflation, start=c(1960,1), frequency=1)

m<-matrix(2,2,data=c(1,2,1,3))

layout(m)

ts.plot(inflation$Inflation,main="Inflation from 1960 to 2021",ylab="Inflation rate")

acf(inflation$Inflation,main="")

pacf(inflation$Inflation ,main="")

#Diagnostic checking

tsdiag(inf\_ar1)

#prediction

inf\_ar1 <- arima(inflation$Inflation,order=c(1,0,0))

project\_inf <- predict(inf\_ar1,n.ahead=129)

project\_inf <- ts(project\_inf$pred, start=c(2022,1), frequency=1)

#forecasting

ts.plot(ts.inf, xlim = c(1960,2150), main="Inflation from 1960 to 2021 and future projection",ylab="Inflation rate")

lines(project\_inf, col = "red")

future\_inflation <- data.frame(Year = seq(2022,2021+length(project\_inf)),

Inflation = as.array(project\_inf))

#write.csv(future\_inflation, "~/R output - future inflation projection.csv")

```