Source Code

# Name – Vaibhav Karn

# Question 1. Which Store has Maximum Sales?

library(dplyr)

library(magrittr)

summary\_1 <- df %>%

group\_by(Store) %>%

summarize(mean = mean(Weekly\_Sales),

sum = sum(Weekly\_Sales),

max = max(Weekly\_Sales),

std = sd(Weekly\_Sales))

arrange <- arrange(summary\_1, desc(max))

arrange

arrange(summary\_1, desc(sum))

Question.2) Which Store has maximum standard deviation?

arrange <- arrange(summary\_1, desc(std))

arrange

summary\_1$cv= round(((summary\_1$std/summary\_1$mean)\*100), 2)

arrange <- arrange(summary\_1, desc(cv))

arrange

# Question 3. Which store/s has good quarterly growth rate in Q3’2012

# check data formats

sapply(df, class)

dtype <- sapply(df, class)

# change the date format

#Option 1

df$date\_v2 <- as.Date(df$Date, format = "%d-%m-%Y")

# option 2

df <- df %>%

mutate(date\_v3 = as.Date(Date, "%d-%m-%Y"))

# Check data format again

dtype <- sapply(df, class)

#Take year and month (option 1)

df$month\_1 <- substring(df$Date,4,5)

df$year\_1 <- substring(df$Date,7,10)

# Take year and month (option 2)

df$month\_2 <- format(as.Date(df$date\_v2, format="%d-%m-%Y"),"%m")

df$year\_2 <- format(as.Date(df$date\_v2, format="%d-%m-%Y"),"%Y")

sapply(df, class)

df$month\_3 <- as.numeric(df$month\_1)

attach(df)

df$quarter[month\_3 < 4] <- "Q1"

df$quarter[month\_3 > 3 & month\_3 <= 6] <- "Q2"

df$quarter[month\_3 > 6 & month\_3 <= 9] <- "Q3"

df$quarter[month\_3 > 9] <- "Q4"

df$year\_quarter <- paste(df$year\_1,df$quarter, sep = "-")

df2 <- aggregate(df$Weekly\_Sales, by=list( store=df$Store, y\_q=df$year\_quarter), FUN=sum)

df2$store\_y\_q <- paste(df2$store,df2$y\_q)

df2 <- arrange(df2, (store\_y\_q) )

attach(df2)

df3 <- select(df2, store, y\_q, x)

library(tidyverse)

df3 <-

df3 %>%

rename(quarter\_sale = x)

library(plyr)

df4 <- ddply(df3,"store",transform, growth=c(NA,exp(diff(log(x)))-1))

df4$growth = round((df4$growth\*100),1)

df5 <- filter(df4, y\_q == "2012-Q3")

df5 <- arrange(df5, desc(growth) )

View(df5)

# Question 4. Some holidays have a negative impact on sales. Find out holidays which have higher sales than the mean sales in non-holiday season for all stores together

df\_hd <- filter(df, Holiday\_Flag == "1")

df\_non\_hd <- filter(df, Holiday\_Flag != "1")

mean(df\_non\_hd$Weekly\_Sales) #1041256

df\_hd$sales <- as.numeric(df\_hd$Weekly\_Sales)

attach(df)

df\_hd$flag[df\_hd$sales > 1041256] <- "1"

df\_hd2 <- filter(df\_hd, flag == "1")

distinct(df\_hd2, Date)

# Question 5. Provide a monthly and semester view of sales in units and give insights

df$year\_month <- format(as.Date(df$date\_v2, format="%d-%m-%Y"),"%Y-%m")

df\_m1 <- aggregate(df$Weekly\_Sales, by=list( y\_m=df$year\_month), FUN=sum)

df\_m1 <- arrange(df\_m1, (y\_m) )

df\_m1$monthly\_sale <- round((df\_m1$x/1000000),2)

attach(df\_m1)

plot(monthly\_sale, type="l")

library(ggplot2)

ggplot(df\_m1, aes(x=y\_m, y=monthly\_sale , group=1)) + geom\_line(col="gray", linetype = "solid") +

geom\_point(col="red") +

theme(axis.text.x=element\_text(color = "black", size=11, angle=90, vjust=.8, hjust=0.8))

df$semester[month\_3 <= 6 ] <- "S1"

df$semester[month\_3 > 6] <- "S2"

attach(df)

table(semester)

df$year\_semester <- paste(df$year\_1,df$semester)

df\_s1 <- aggregate(df$Weekly\_Sales, by=list( y\_s=df$year\_semester), FUN=sum)

df\_s1 <- arrange(df\_s1, (y\_s) )

df\_s1$half\_year\_sale <- round((df\_s1$x/1000000),2)

ggplot(df\_s1, aes(x=y\_s, y=half\_year\_sale , group=1)) + geom\_line(col="red", linetype = "solid") +

geom\_point(col="gray") +

theme(axis.text.x=element\_text(color = "black", size=11, angle=90, vjust=.8, hjust=0.8))

6. Statistical Model :

df\_store\_1 <- filter(df, Store == "1")

df\_store\_1$date <- as.Date(df\_store\_1$Date, format = "%d-%m-%Y")

df\_store\_1 <- arrange(df\_m1, (date) )

df\_store\_1 <- cbind(date\_new = rownames(df\_store\_1), df\_store\_1)

rownames(df\_store\_1) <- 1:nrow(df\_store\_1)

df\_store\_1$sales <- as.numeric(df\_store\_1$Weekly\_Sales)

df\_store\_1$date\_new <- as.numeric(df\_store\_1$date\_new)

model\_obj = lm(sales ~ date\_new + CPI + Unemployment + Fuel\_Price , data=df\_store\_1)

)summary(model\_obj

AIC(model\_obj)

model\_obj2 = lm(sales ~ CPI + Unemployment + Fuel\_Price + month\_2 + year\_1, data=df\_store\_1)

summary(model\_obj2)

AIC(model\_obj2)

# 7. Change dates into days by creating new Variable

df$day <- format(as.Date(df$date\_v2, format="%d-%m-%Y"),"%d")