EXPLORATORY DATA ANALYSIS

23CSEG28

```
library(dplyr)

## Warning: package 'dplyr' was built under R version 4.3.3

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(lubridate)

## Warning: package 'lubridate' was built under R version 4.3.3

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.3.3

library(plotly)

library(reshape2)

## Warning: package 'reshape2' was built under R version 4.3.3

df <- read.csv("C:/Users/ADMIN/Downloads/Walmart.csv")

# Assuming 'df' is your data frame name, change it accordingly if it's differ
ent
# Change date type
data <- df %>% mutate(Date = dmy(Date))
sprintf("The data type of the Date variable is: %s", class(data$Date))

## [1] "The data type of the Date variable is: Date"

str(data)
```

```
## 'data.frame':	6435 obs. of  8 variables:
##  $ Store       : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Date        : Date, format: "2010-02-05" "2010-02-12" ...
##  $ Weekly_Sales: num  1643691 1641957 1611968 1409728 1554807 ...
##  $ Holiday_Flag: int  0 1 0 0 0 0 0 0 0 0 ...
##  $ Temperature : num  42.3 38.5 39.9 46.6 46.5 ...
##  $ Fuel_Price  : num  2.57 2.55 2.51 2.56 2.62 ...
##  $ CPI         : num  211 211 211 211 211 ...
##  $ Unemployment: num  8.11 8.11 8.11 8.11 8.11 ...

# Filtering month, date, and year
Weekday <- day(data$Date)
Months <- month(data$Date)
Year <- year(data$Date)

# Add Day column
data <- data %>% mutate(Weekday = Weekday)
# Add year column
data <- data %>% mutate(Year = Year)

## Classifying fuel prices
data <- data %>% mutate(Sts_Fuel_Price = ifelse(Fuel_Price < mean(data$Fuel_P
rice), "Low", "High"))
```

```r
# Univariate Analysis
# Distribution of Weekly sales
ggplot(data = data, aes(x = Weekly_Sales)) +
  geom_histogram(bins = 20, color = 'purple', fill = 'white', aes(y = after_stat(density))) +geom_density(alpha = 0.5) +
labs(title = "Distribution of Weekly sales", x = "Weekly Sales", y = "Frequency") +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  theme(plot.title = element_text(hjust = 0.5))
```

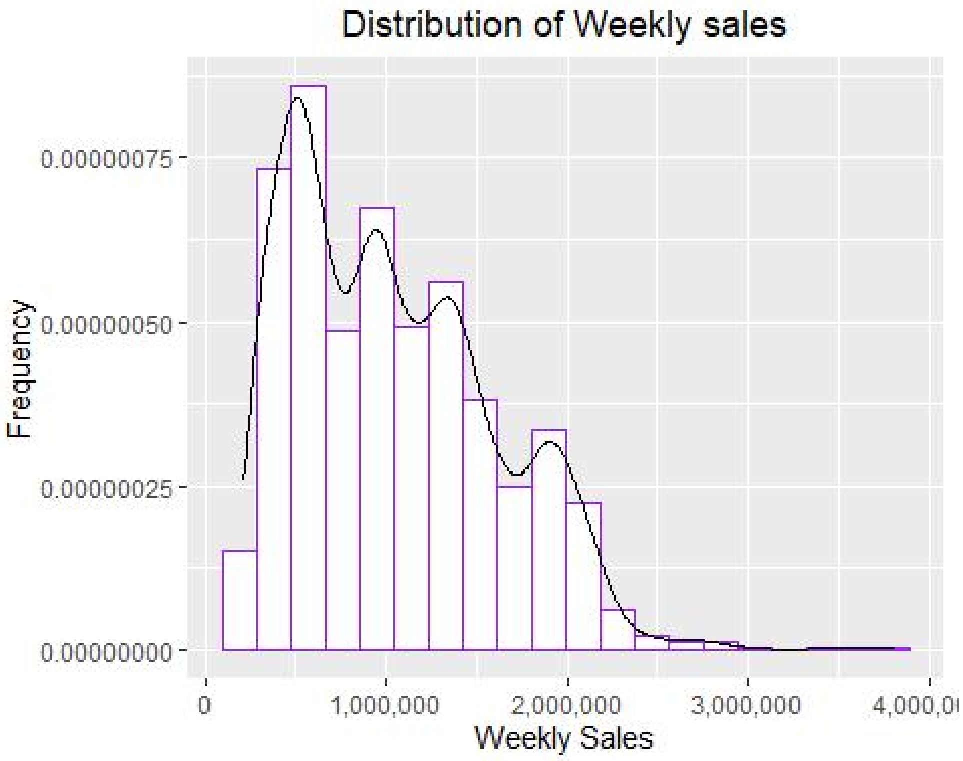


Fig 1.1

```r
# Distribution of Consumer Price Index
ggplot(data = data, aes(x = CPI)) +geom_histogram(bins = 25, color = 'green',
fill = 'white', aes(y = after_stat(density))) +geom_density(alpha = 0.5) +
labs(title = "Distribution of Consumer Price Index", x = "Consumer Price Inde
x", y = "Frequency") +scale_x_continuous(labels = scales::comma)+scale_y_cont
inuous(labels = scales::comma) +theme(plot.title = element_text(hjust = 0.5))
```

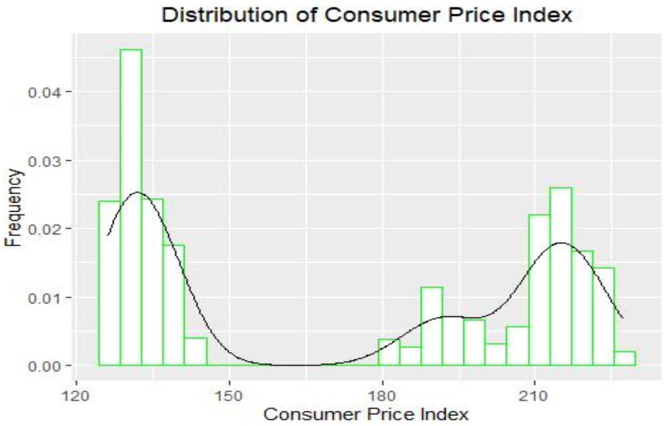


Fig 1.2

```r
# Distribution of Weekdays of the sales
ggplot(data = data, aes(x = Weekday)) +geom_histogram(bins = 30, color = 'red
',fill = 'white', aes(y = after_stat(density)))+geom_density(alpha = 0.5) +
 labs(title = "Distribution of Weekdays in store", x = "Weekdays", y = "Frequ
ency") +scale_x_continuous(labels = scales::comma) +scale_y_continuous(labels
 = scales::comma) +theme(plot.title = element_text(hjust = 0.5))
```

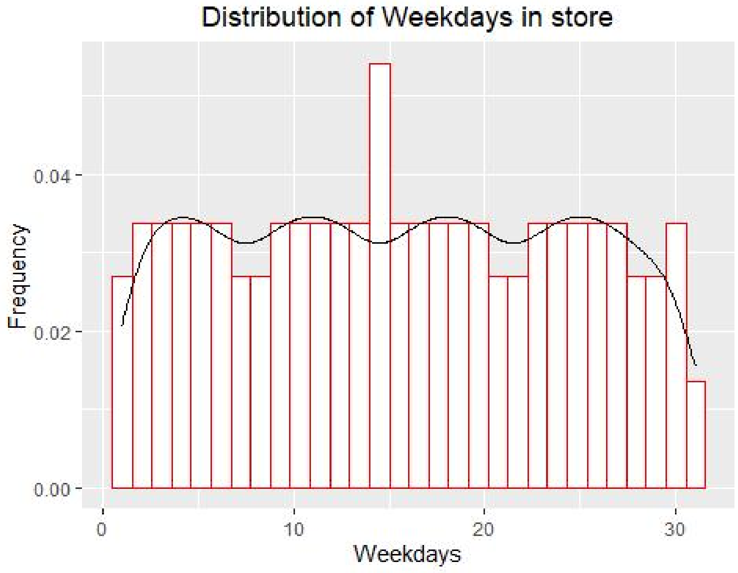


Fig 1.3

```
# Distribution of Temperature
ggplot(data = data, aes(x = Temperature)) +
  geom_histogram(aes(y = ..density..), position = "identity", bins = 30, colo
ur = 'blue', fill = "white") +
  geom_density(alpha = 0.2, fill = "white") +
  labs(title = "Distribution of Temperature", x = "Temperature")
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.
4.0.
## ℹ Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

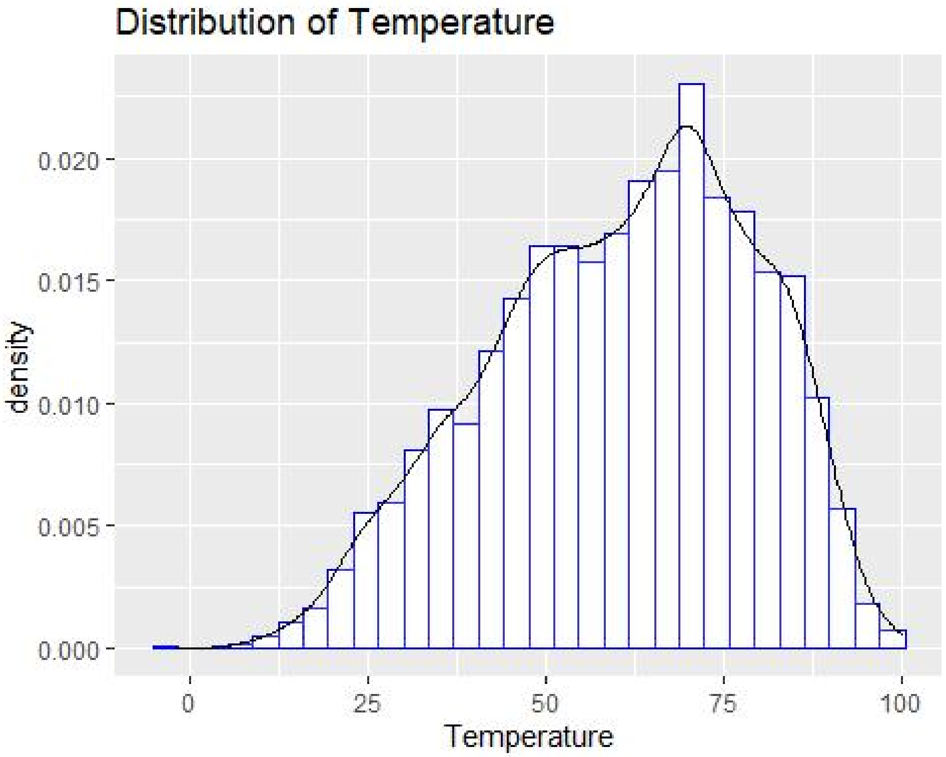


Fig 1.4

```
# Distribution of fuel prices
ggplot(data = data, aes(x = Fuel_Price)) +
  geom_histogram(aes(y = ..density..), position = "identity", binwidth = 0.10,
 colour = 'black', fill = "white") +
  geom_density(alpha = 0.2, fill = "white") +
  labs(title = "Distribution of Fuel prices", x = "Fuel Price")
```

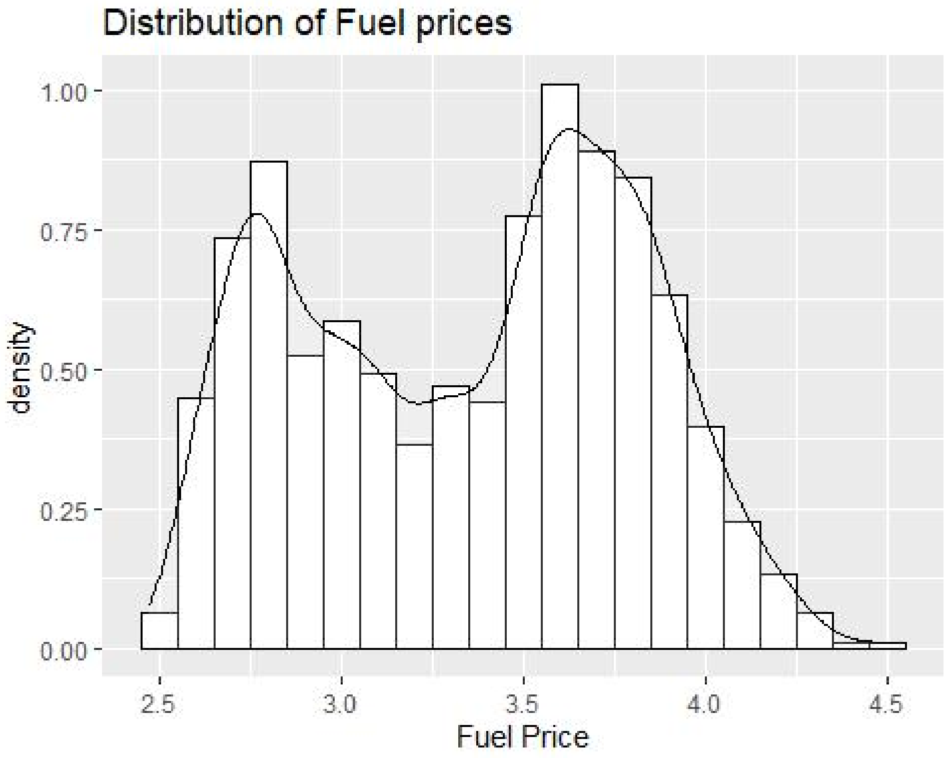


Fig 1.5

```r
# Distribution of Unemployment ratio
ggplot(data = data, aes(x = Unemployment)) +
  geom_histogram(bins = 15, color = 'red', fill = 'white', aes(y = after_stat
(density))) +
  geom_density(alpha = 0.5) +
  labs(title = "Distribution of Unemployment ratio", x = "Unemployment ratio",
 y = "Frequency") +
  scale_x_continuous(labels = scales::comma) +
  scale_y_continuous(labels = scales::comma) +
  theme(plot.title = element_text(hjust = 0.5))
```

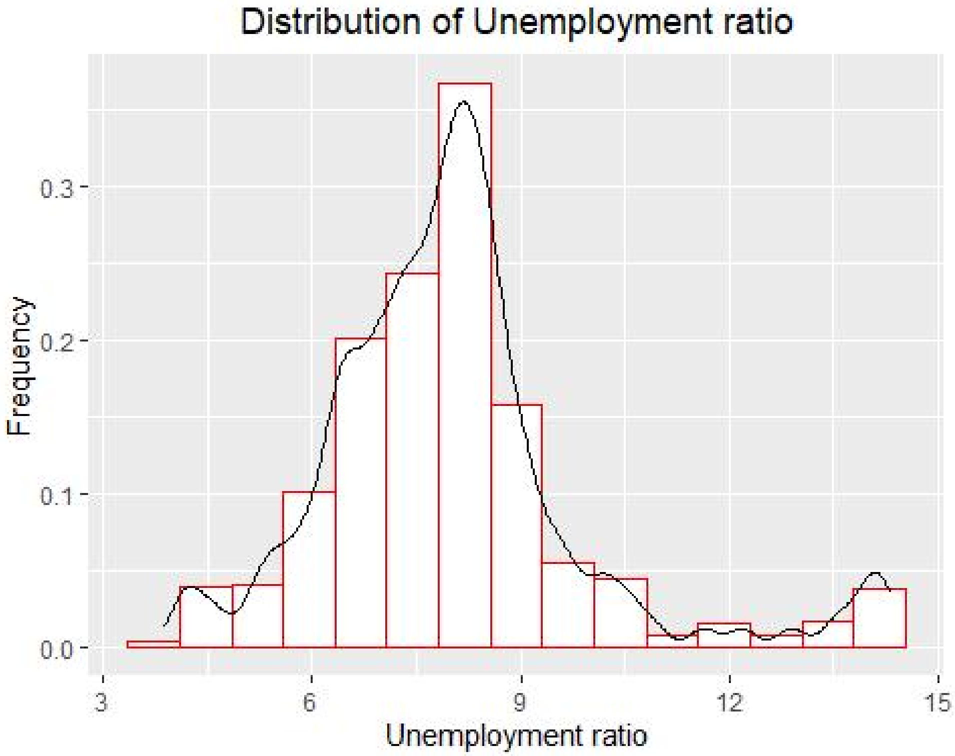


Fig 1.6

```r
# Holiday flags classes
holiday_flag_plot <- data %>% count(Holiday_Flag) %>%
  plot_ly(x = ~Holiday_Flag, y = ~n, type = 'bar', text = ~n) %>%
  layout(title = "Holiday Flag Classes",
         xaxis = list(title = "Holiday Flag"),
         yaxis = list(title = "Frequency"))

# Print the plot
print(holiday_flag_plot)
```

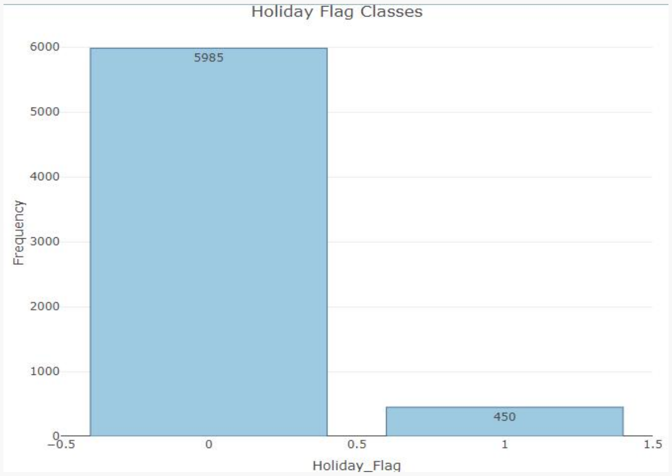


Fig 1.7

```
# Bivariate analysis
# Store of Unemployment by year attribute
ggplot(data = data, aes(y = Unemployment, x = Weekly_Sales)) +
  geom_boxplot(aes(fill = factor(Year))) +
  coord_flip() +
labs(title = "Store of Unemployment by year", x = "Unemployment", y = "Store")
```

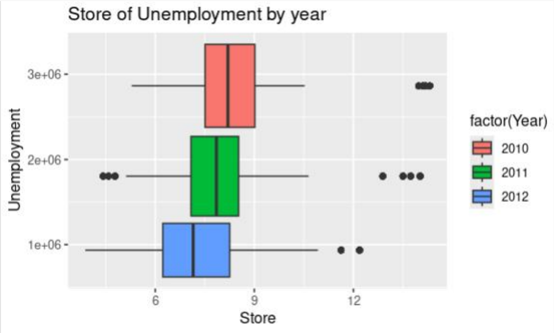


Fig 1.8

```
# Annual income of all the year
Annual_Sales <- aggregate(Weekly_Sales ~ Year, data, sum)
Annual_Sales <- Annual_Sales[order(-Annual_Sales$Weekly_Sales), ]
print(Annual_Sales)

##   Year Weekly_Sales
## 2 2011   2448200007
## 1 2010   2288886120
## 3 2012   2000132859

sales <- Annual_Sales$Weekly_Sales
labels <- c('2010', '2011', '2012')
porcent <- round(sales / sum(sales) * 100)
labels <- paste(labels, porcent, "%", sep = " ")
pie(sales,labels = labels,col = rainbow(length(labels)),main = "Annual Sales")
legend("topright", c("2010", "2011", "2012"),cex = 0.8,
 fill = rainbow(length(labels)))
```

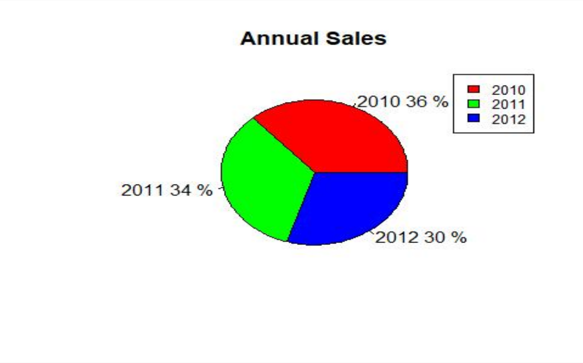


Fig 1.9

```
# Month wise Weekly Sales by Year
ggplot(data = data, aes(x = Months, y = Weekly_Sales)) +geom_point(aes(color
= factor(Year))) + labs(title ='Month wise Weekly Sales by Year', x = 'Month
of the sales', y = 'Weekly sales') +scale_y_continuous(labels = scales::comma)
 +theme(plot.title = element_text(hjust = 0.5))
```

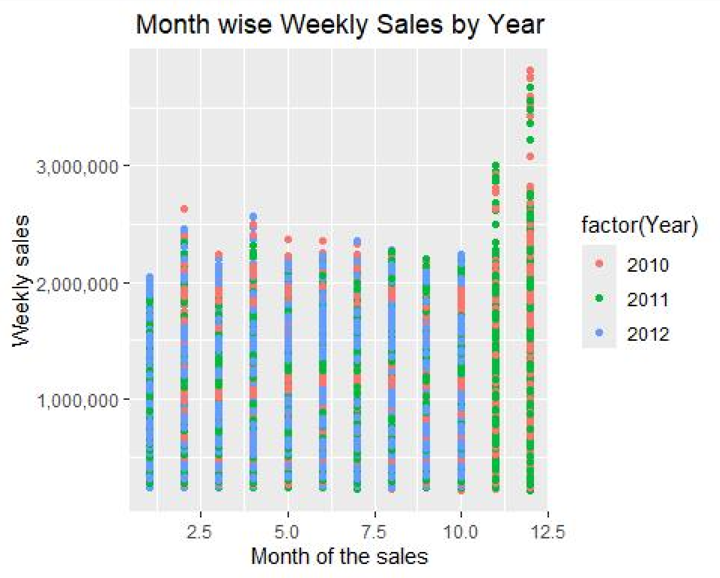


Fig 1.10

```
# Relationship of Weekly sales by fuel price
ggplot(data = data, aes(x = Fuel_Price, y = Weekly_Sales)) +geom_point(alpha=
0.1, colour = 'blue') +labs(title = 'Relationship of Weekly sales by fuel pri
ce', y = 'Weekly sales', x = 'Fuel Price')
```

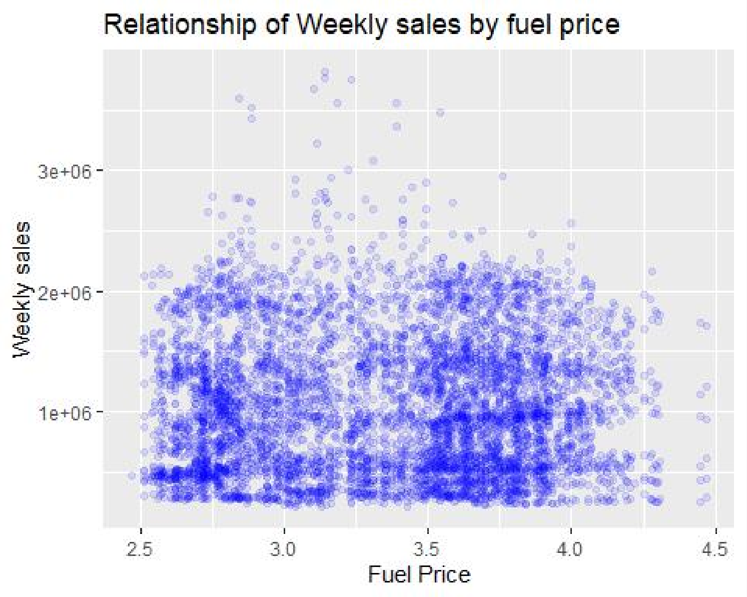


Fig 1.11

```
# Relationship of Weekly Sales by CPI
ggplot(data = data, aes(x = CPI, y = Weekly_Sales)) +
  geom_point(alpha = 0.1,  colour = 'blue') +
  labs(title = 'Relationship of Weekly sales by CPI', y = 'Weekly sales', x =
 'CPI')
```

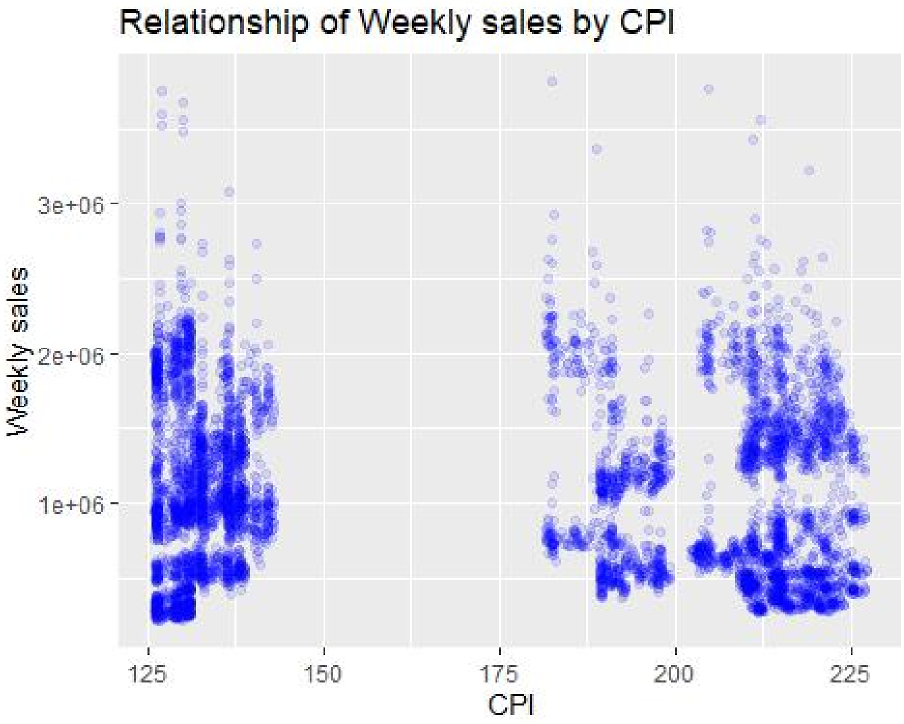


Fig 1.12

```
# Distribution Weekly_sale by Temperature

walmart_data %>% ggplot(aes(x = Temperature, y = Weekly_Sales)) +geom_point(a
lpha = 0.1, colour = 'blue') +labs(title = 'Distribution Weekly_sale by Tempe
rature',y='Weekly sales',x='Temperature')
```

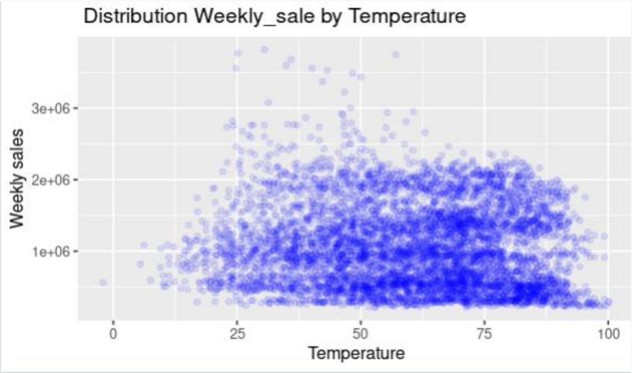


Fig 1.13

```
    #Relationship of Weekly sales by Holiday Flag
ggplot(data =  data, aes(x = Holiday_Flag, y = Weekly_Sales)) +
  geom_point(alpha = 0.1,  colour = 'blue') +
  labs(title = 'Relationship of Weekly sales by Holiday Flag', y = 'Weekly sa
les', x = 'Holiday Flag')
```

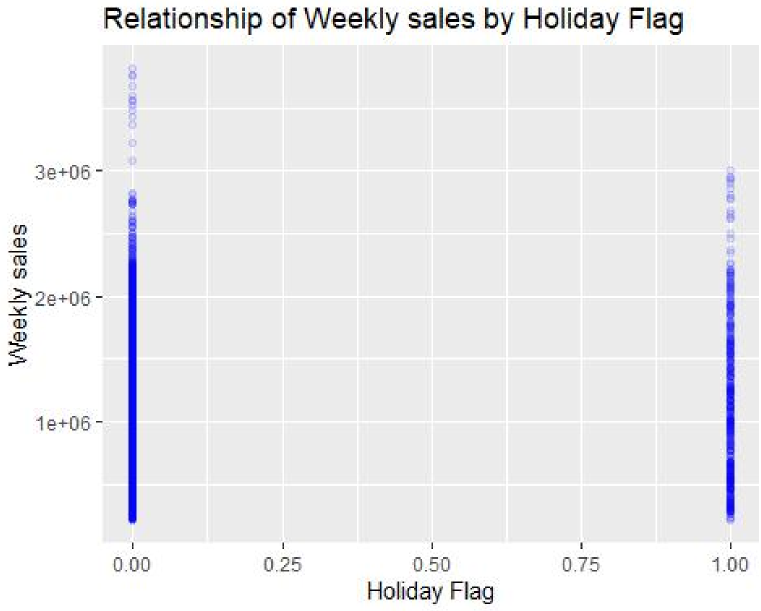


Fig 1.16

```
# Relationship of  Weekly Sales by Unemployment

walmart_data %>% ggplot(aes(x = Unemployment,y = Weekly_Sales)) +geom_point(a
lpha = 0.1, colour = 'blue') +labs(title = 'Relationship of  Weekly Sales by
Unemployment',y='Weekly sales',x='Unemployment')
```

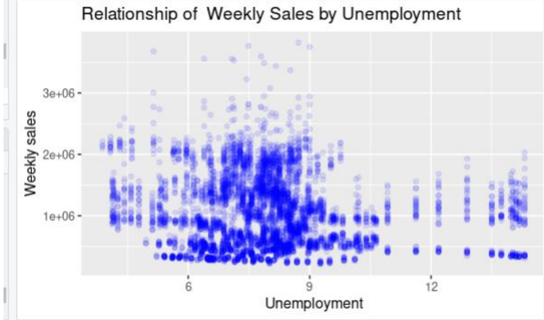


Fig 1.15

```
# Weekly sales for year
data %>% group_by(Year) %>%
  summarise(Weekly_Sales = mean(Weekly_Sales, na.rm = T)) %>%
  ggplot(aes(Year, Weekly_Sales)) +
  geom_point(aes(color = Weekly_Sales > 1200000), show.legend = F) +
  geom_line(color = 'grey') +
  labs(title = 'Weekly Sales for year', y = 'Weekly sales', x = 'Year') +
  theme_bw()
```

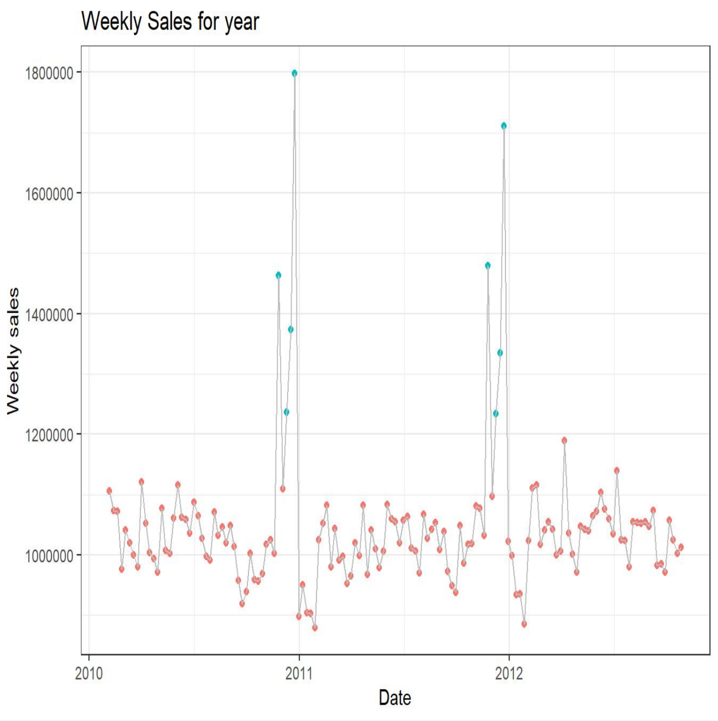
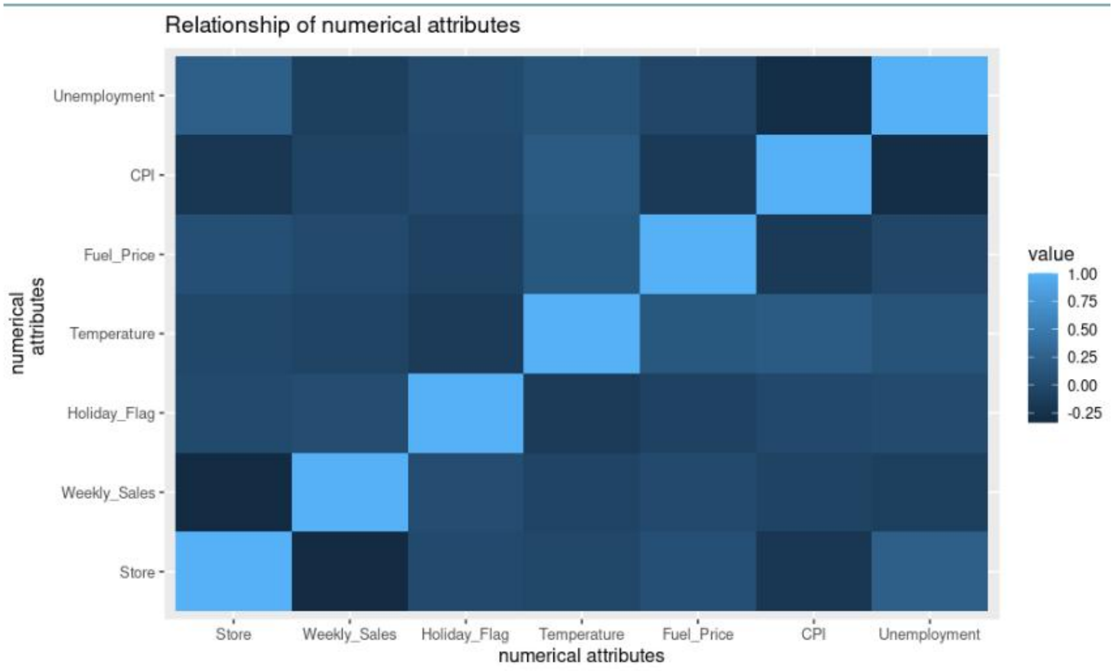


Fig 1.15

```r
# Relationship of numerical attributes in the walmart dataset
data_cor <- cor(data[sapply(data, is.numeric)])
data_melted <- melt(data_cor)

ggplot(data_melted, aes(x = Var1, y = Var2, fill = value)) +
  geom_tile() +
  labs(title = "Relationship of numerical attributes", x = "Numerical Attribu
tes", y = "Numerical Attributes")
```

Fig 1.17


Relationship of numerical attributes
Unemployment
CPI
Fuel_Price
Temperature
Holiday_Flag
Weekly_Sales
Store
numerical attributes
Store
Weekly_Sales
Holiday_Flag
Temperature
Fuel_Price
CPI
Unemployment
numerical attributes
value