CSDA304 BUSINESS MODELLING&APPLIEDANALYTICSUSINGR PROJECT-MODULEII

(Experiential Learning/ Component 1)

1. Create a bar plot to visualize the frequency of different car brands in the mtcars dataset

Code:

data(mtcars)

mtcars$brand <- sapply(strsplit(rownames(mtcars), " "), `[`, 1)

brand\_frequency <- table(mtcars$brand)

barplot(brand\_frequency,

main = "Frequency of Different Car Brands in mtcars Dataset",

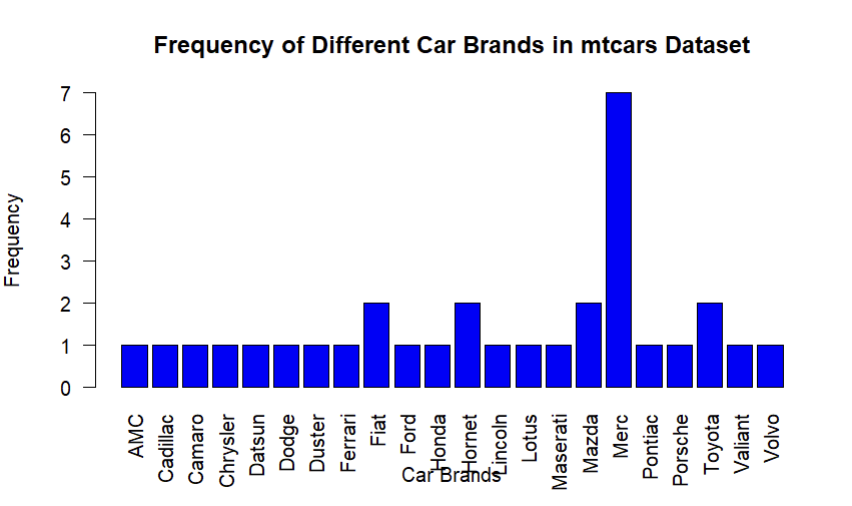
xlab = "Car Brands",

ylab = "Frequency",

col = "blue",

las = 2)

output:



1. Using the iris dataset, generate a scatter plot of sepal length vs. sepal width. Color the points by species

Code:

data(iris)

species\_colors <- c("setosa" = "red", "versicolor" = "green", "virginica" = "blue")

plot(iris$Sepal.Length, iris$Sepal.Width,

col = species\_colors[iris$Species],

pch = 19,

xlab = "Sepal Length",

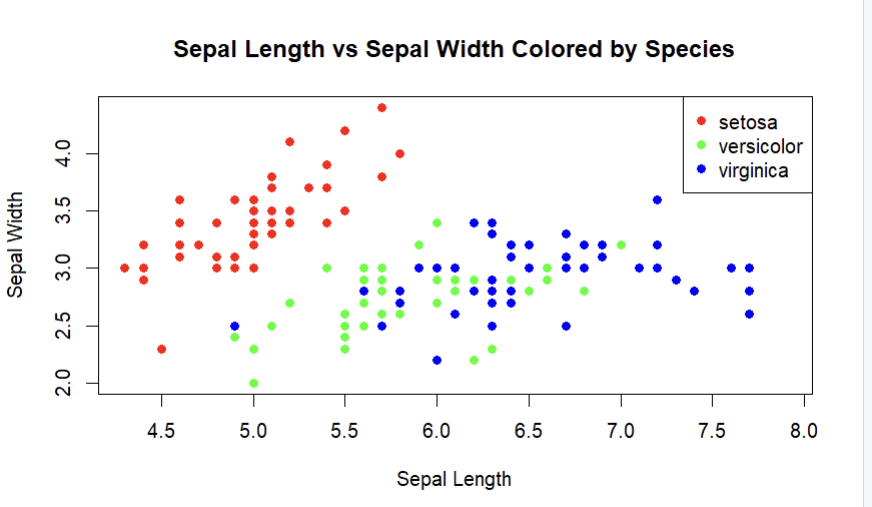
ylab = "Sepal Width",

main = "Sepal Length vs Sepal Width Colored by Species")

legend("topright", legend = levels(iris$Species),

col = species\_colors, pch = 19)

output:



1. Construct a histogram of the 'mpg' (miles per gallon) variable from the mtcars dataset. Experiment with different bin widths.

Code:

data(mtcars)

hist(mtcars$mpg,

breaks=7, main="Histogram of mpg in mtcars - 7 Bins",

xlab="mpg", col="lightblue", border="black")

hist(mtcars$mpg,

breaks=10, main="Histogram of mpg in mtcars - 10 Bins",

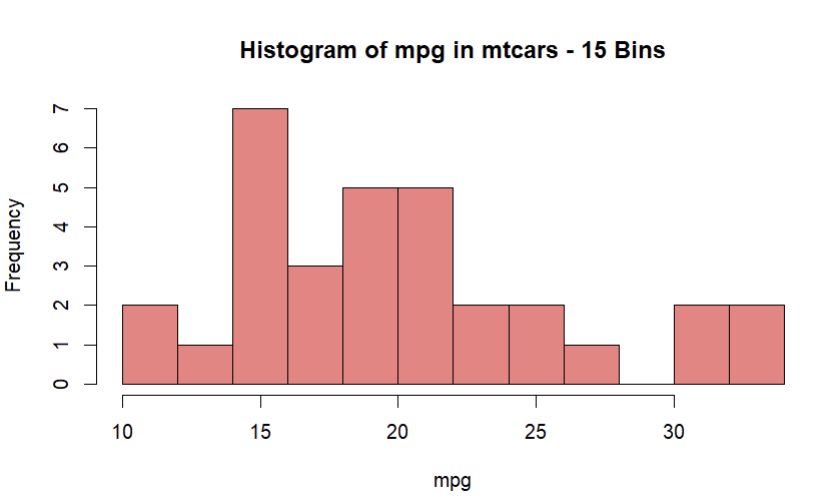
xlab="mpg", col="lightgreen", border="black")

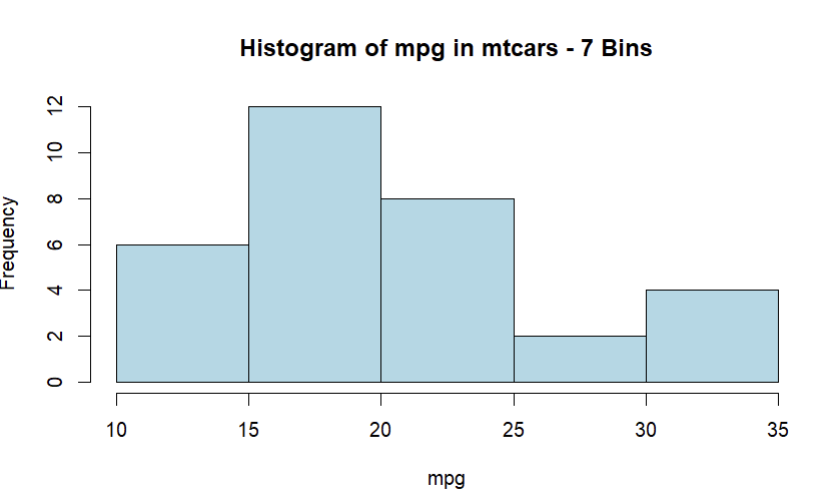
hist(mtcars$mpg,

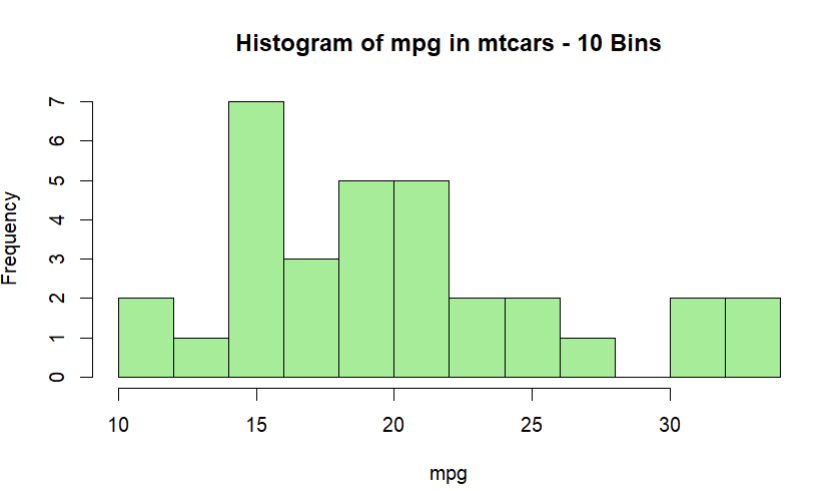
breaks=15, main="Histogram of mpg in mtcars - 15 Bins",

xlab="mpg", col="lightcoral", border="black")

output:







1. Create two pie charts side by side using the Titanic dataset: one showing the proportion of survivors vs. non-survivors, and another showing the proportion of passengers in each class.

library(datasets)

data(Titanic)

titanic\_df <- as.data.frame(Titanic)

survival\_summary <- aggregate(Freq ~ Survived, data = titanic\_df, sum)

class\_summary <- aggregate(Freq ~ Class, data = titanic\_df, sum)

par(mfrow = c(1, 2))

pie(survival\_summary$Freq, labels = survival\_summary$Survived,

main = "Proportion of Survivors vs. Non-Survivors",

col = c("red", "green"))

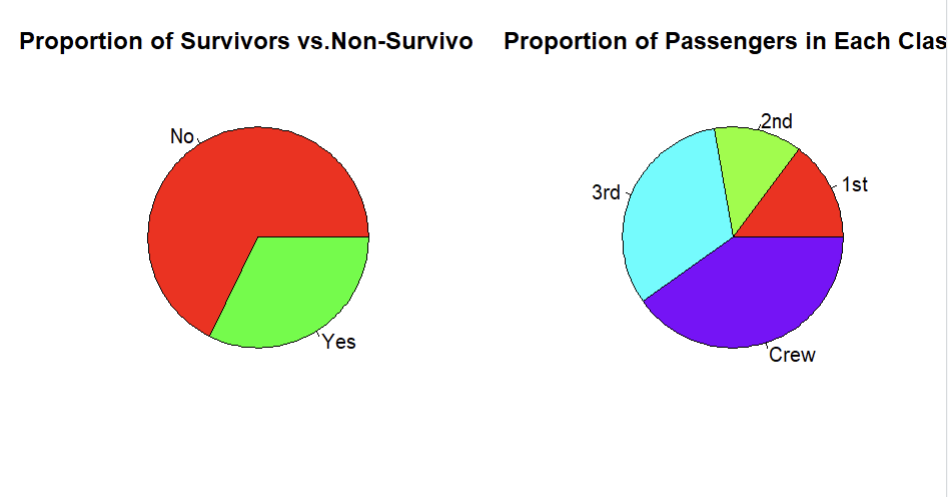
pie(class\_summary$Freq, labels = class\_summary$Class,

main = "Proportion of Passengers in Each Class",

col = rainbow(length(class\_summary$Class)))

par(mfrow = c(1, 1))

output:



1. Using the diamonds dataset from ggplot2, create a box plot of price grouped by cut quality. Add individual data points as a jitter plot over the box plot. (Use ggplot2 package only for loading data

Code:

install.packages("ggplot2")

library(ggplot2)

data("diamonds")

ggplot(diamonds, aes(x = cut, y = price)) +

geom\_boxplot(outlier.colour = NA, fill = "lightblue") + # Create the box plot

geom\_jitter(width = 0.2, alpha = 0.3, color = "darkblue") + # Add jitter plot

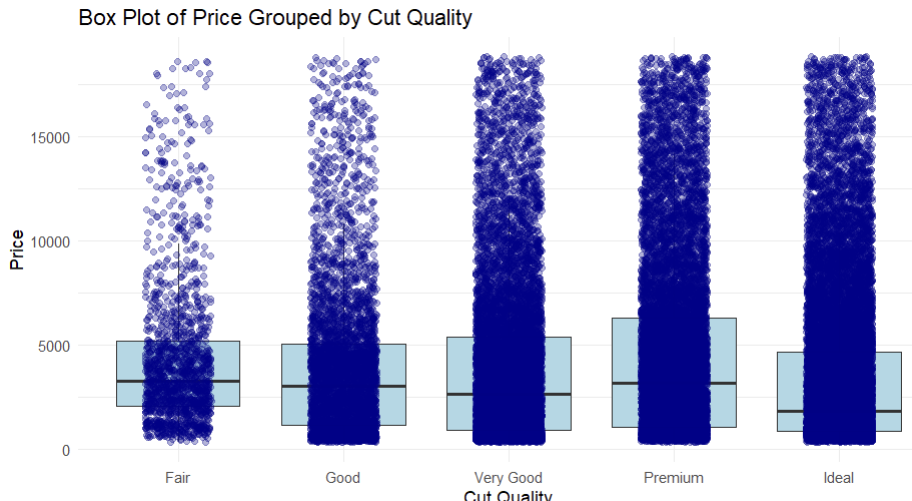
labs(title = "Box Plot of Price Grouped by Cut Quality",

x = "Cut Quality",

y = "Price") +

theme\_minimal()

output:

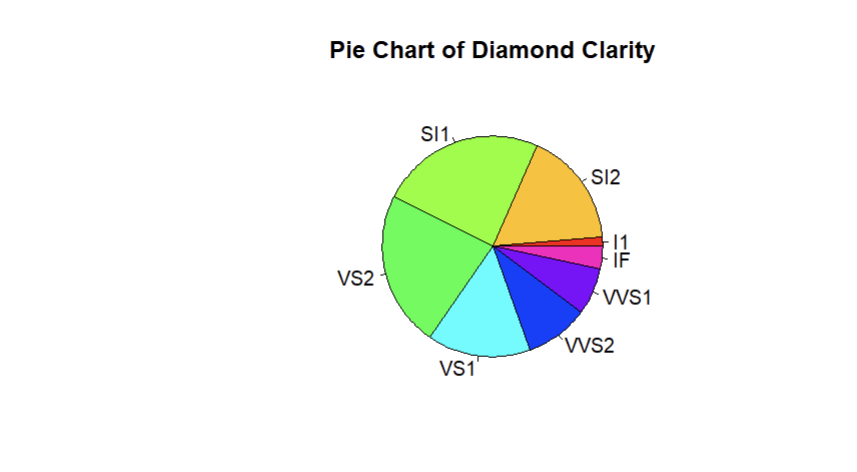


6.Using the diamonds dataset from ggplot2, create a pie chart of diamond clarity categories. Experiment with different color palettes. (Use ggplot2 package only for loading data

Code:

data(diamonds, package="ggplot2")  
  
clarity <- table(diamonds$clarity)  
  
pie(clarity, main="Pie Chart of Diamond Clarity", col=rainbow(length(clarity)))

output:



7.Using the mpg dataset, create a grouped and stacked bar chart showing the count of cars by manufacturer, with each bar segment representing different classes of vehicles, and the groups representing transmission type (automatic/manual)

Code:

library(ggplot2)

library(dplyr)

data("mpg")

mpg\_summary <- mpg %>%

group\_by(manufacturer, class) %>%

summarise(count = n()) %>%

ungroup()

ggplot(mpg\_summary, aes(x = manufacturer, y = count, fill = class)) +

geom\_bar(stat = "identity") +

labs(title = "Count of Cars by Manufacturer and Class",

x = "Manufacturer",

y = "Count",

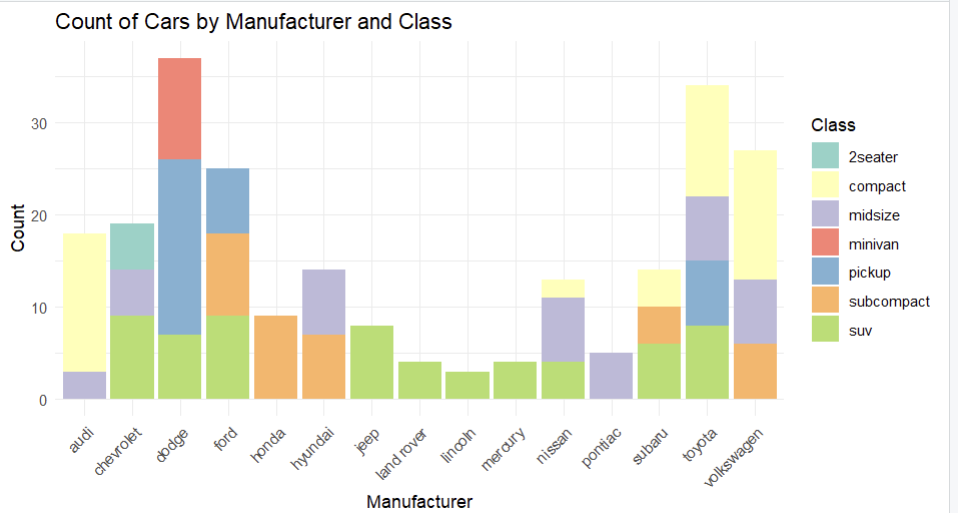
fill = "Class") +

theme\_minimal() +

theme(axis.text.x = element\_text(angle = 45, hjust = 1)) +

scale\_fill\_brewer(palette = "Set3")

output:



8. Create a comparative histogram (two overlaid histograms with transparency) of highway and city mpg from the mpg dataset. (Use ggplot2 package only for loading data)

Code:

# Load necessary library

library(ggplot2)

# Load the mpg dataset

data(mpg)

# Create the comparative histogram

ggplot(mpg) +

geom\_histogram(aes(x = hwy, fill = "Highway MPG"), alpha = 0.5, binwidth = 2, color = "black") +

geom\_histogram(aes(x = cty, fill = "City MPG"), alpha = 0.5, binwidth = 2, color = "black") +

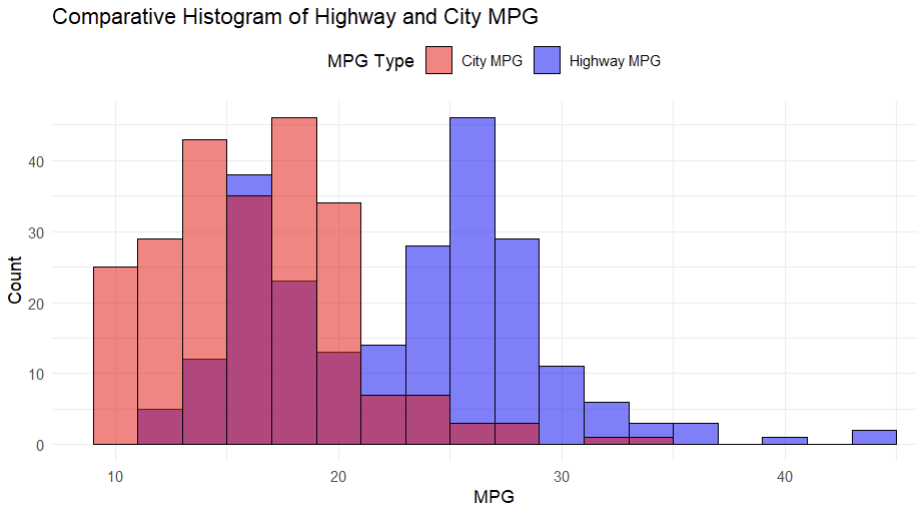
scale\_fill\_manual(values = c("Highway MPG" = "blue", "City MPG" = "red")) +

theme\_minimal() +

labs(title = "Comparative Histogram of Highway and City MPG",x = "MPG",y = "Count",fill = "MPG Type") +

theme(legend.position = "top")

output:



9. Using the airquality dataset, create a box plot of Ozone levels grouped by Month. Color the boxes by temperature ranges (you'll need to create these categories).

Code:

# Load necessary libraries

library(ggplot2)

library(dplyr)

data(airquality)

# Remove rows with missing Ozone or Temp values

airquality <- na.omit(airquality)

# Create temperature categories

airquality <- airquality %>%

mutate(temp\_range = cut(Temp,

breaks = c(-Inf, 60, 70, 80, 90, Inf),

labels = c("<=60°F", "61-70°F", "71-80°F", "81-90°F", ">90°F")))

# Create the box plot

ggplot(airquality, aes(x = factor(Month, levels = 5:9, labels = c("May", "June", "July", "August", "September")),

y = Ozone, fill = temp\_range)) +

geom\_boxplot() +

theme\_minimal() +

labs(title = "Box Plot of Ozone Levels by Month",

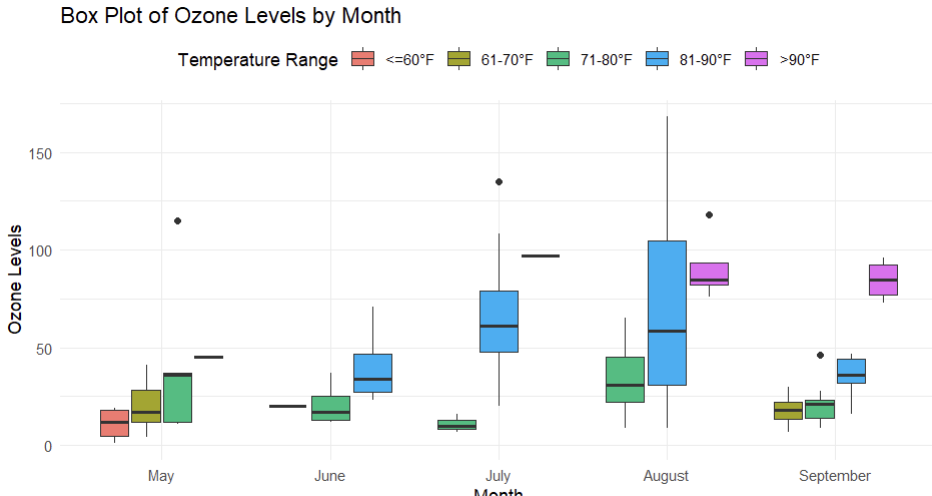
x = "Month",

y = "Ozone Levels",

fill = "Temperature Range") +

theme(legend.position = "top")

output:



10.Create an animated scatter plot using the gapminder dataset, showing how the relationship between GDP per capita and life expectancy changes over time. Use continents for color and population for point size

Code:

install.packages("gapminder")

library(gganimate)

library(ggplot2)

library(gapminder)

# Load the gapminder dataset

data(gapminder)

head(gapminder)

# Create the ggplot object

p <- ggplot(gapminder, aes(x = gdpPercap, y = lifeExp, size = pop, color = continent, frame = year)) +

geom\_point(alpha = 0.7, show.legend = FALSE) +

scale\_x\_log10() + # Use log scale for GDP per capita

scale\_size(range = c(2, 12)) + # Set the range for point sizes

labs(title = 'Year: {frame\_time}', x = 'GDP per Capita', y = 'Life Expectancy') +

theme\_minimal() +

theme(plot.title = element\_text(hjust = 0.5)) +

transition\_time(year) +

ease\_aes('linear')

# Animate and save the plot

animate(p, duration = 20, fps = 10, width = 800, height = 600, renderer = gifski\_renderer("gdp\_lifeExp\_animation.gif"))

output:

