**1. Visualize Scatter plot for given dataset.**

|  |  |
| --- | --- |
| Weight (kg) | Height (cm) |
| 60 | 170 |
| 65 | 175 |
| 70 | 168 |
| 72 | 180 |
| 75 | 178 |

**Sample Code:**

**# Load necessary library**

**library(ggplot2)**

**# Create a data frame**

**weight\_height\_data <- data.frame(**

**Weight = c(60, 65, 70, 72, 75),**

**Height = c(170, 175, 168, 180, 178)**

**)**

**# Plot scatter plot**

**ggplot(weight\_height\_data, aes(x = Weight, y = Height)) +**

**geom\_point(color = "blue", size = 3) +**

**labs(title = "Weight vs. Height",**

**x = "Weight (kg)",**

**y = "Height (cm)") +**

**theme\_minimal()**

**2. Visualize a Line Plot for the Given Population Data**

|  |  |
| --- | --- |
| **Year** | **Population (millions)** |
| 2010 | 1000 |
| 2011 | 1020 |
| 2012 | 1040 |
| 2013 | 1060 |
| 2014 | 1080 |

**# Create a data frame**

**year\_population\_data <- data.frame(**

**Year = 2010:2014,**

**Population = c(1000, 1020, 1040, 1060, 1080)**

**)**

**# Plot line plot**

**ggplot(year\_population\_data, aes(x = Year, y = Population)) +**

**geom\_line(color = "green", size = 1) +**

**geom\_point(color = "red", size = 3) +**

**labs(title = "Year vs. Population",**

**x = "Year",**

**y = "Population (millions)") +**

**theme\_minimal()**

**3. Bar Plot**

**Dataset: Sales by Product**

|  |  |
| --- | --- |
| Product | Sales |
| A | 300 |
| B | 450 |
| C | 500 |
| D | 350 |
| E | 400 |

**# Create a data frame**

**sales\_data <- data.frame(**

**Product = c("A", "B", "C", "D", "E"),**

**Sales = c(300, 450, 500, 350, 400)**

**)**

**# Plot bar plot**

**ggplot(sales\_data, aes(x = Product, y = Sales, fill = Product)) +**

**geom\_bar(stat = "identity") +**

**labs(title = "Sales by Product",**

**x = "Product",**

**y = "Sales") +**

**theme\_minimal()**

**4. Histogram**

**Dataset: Age Distribution**

|  |  |
| --- | --- |
| | **Age** | | --- | |
| |  | | --- | | 25 | |
| |  | | --- | | 30 | |
| |  | | --- | | 35 | |
| |  | | --- | | 40 | |
| |  | | --- | | 45 | |
| |  | | --- | | 50 | |
| |  | | --- | | 55 | |
| |  | | --- | | 60 | |
| |  | | --- | | 65 | |
| |  | | --- | | 70 | |

**# Create a data frame**

**age\_data <- data.frame(**

**Age = c(25, 30, 35, 40, 45, 50, 55, 60, 65, 70)**

**)**

**# Plot histogram**

**ggplot(age\_data, aes(x = Age)) +**

**geom\_histogram(binwidth = 5, fill = "purple", color = "black", alpha = 0.7) +**

**labs(title = "Age Distribution",**

**x = "Age",**

**y = "Frequency") +**

**theme\_minimal()**

**5. Pie Chart**

**Dataset: Market Share by Company**

|  |  |
| --- | --- |
| Company | Market Share |
| A | 20% |
| B | 30% |
| C | 25% |
| D | 25% |

**# Create a data frame**

**market\_share\_data <- data.frame(**

**Company = c("A", "B", "C", "D"),**

**Market\_Share = c(20, 30, 25, 25)**

**)**

**# Plot pie chart**

**ggplot(market\_share\_data, aes(x = "", y = Market\_Share, fill = Company)) +**

**geom\_bar(stat = "identity", width = 1) +**

**coord\_polar("y") +**

**labs(title = "Market Share by Company",**

**x = NULL,**

**y = NULL) +**

**theme\_void() +**

**theme(legend.position = "right")**

**6. Box Plot**

**Dataset: Exam Scores by Class**

|  |  |
| --- | --- |
| Class | Scores |
| A | 85 |
| A | 90 |
| A | 78 |
| A | 92 |
| B | 88 |
| B | 76 |
| B | 80 |
| B | 84 |

**# Create a data frame**

**scores\_data <- data.frame(**

**Class = c("A", "A", "A", "A", "B", "B", "B", "B"),**

**Scores = c(85, 90, 78, 92, 88, 76, 80, 84)**

**)**

**# Plot box plot**

**ggplot(scores\_data, aes(x = Class, y = Scores, fill = Class)) +**

**geom\_boxplot() +**

**labs(title = "Exam Scores by Class",**

**x = "Class",**

**y = "Scores") +**

**theme\_minimal()**

**7. Density Plot**

**Dataset: Weights of Individuals**

|  |  |
| --- | --- |
| | **Weight (kg)** | | --- | |
| |  | | --- | | 60 | |
| |  | | --- | | 65 | |
| |  | | --- | | 70 | |
| |  | | --- | | 75 | |
| |  | | --- | | 80 | |
| |  | | --- | | 85 | |
| |  | | --- | | 90 | |
| |  | | --- | | 95 | |
| |  | | --- | | 100 | |

**# Create a data frame**

**weights\_data <- data.frame(**

**Weight = c(60, 65, 70, 75, 80, 85, 90, 95, 100)**

**)**

**# Plot density plot**

**ggplot(weights\_data, aes(x = Weight)) +**

**geom\_density(fill = "lightblue") +**

**labs(title = "Density Plot of Weights",**

**x = "Weight (kg)",**

**y = "Density") +**

**theme\_minimal()**

**8. Violin Plot**

**Dataset: Scores by Group**

|  |  |
| --- | --- |
| Group | Score |
| X | 80 |
| X | 85 |
| X | 78 |
| X | 92 |
| Y | 88 |
| Y | 76 |
| Y | 80 |
| Y | 84 |

**# Create a data frame**

**scores\_group\_data <- data.frame(**

**Group = c("X", "X", "X", "X", "Y", "Y", "Y", "Y"),**

**Score = c(80, 85, 78, 92, 88, 76, 80, 84)**

**)**

**# Plot violin plot**

**ggplot(scores\_group\_data, aes(x = Group, y = Score, fill = Group)) +**

**geom\_violin() +**

**labs(title = "Scores by Group",**

**x = "Group",**

**y = "Score") +**

**theme\_minimal()**

**9. Heatmap**

**Dataset: Monthly Temperatures**

|  |  |  |  |
| --- | --- | --- | --- |
| Month | City A | City B | City C |
| Jan | 5 | 10 | 15 |
| Feb | 6 | 11 | 16 |
| Mar | 7 | 12 | 17 |
| Apr | 8 | 13 | 18 |
| May | 9 | 14 | 19 |

**# Create a data frame**

**temperature\_data <- data.frame(**

**Month = c("Jan", "Feb", "Mar", "Apr", "May"),**

**City\_A = c(5, 6, 7, 8, 9),**

**City\_B = c(10, 11, 12, 13, 14),**

**City\_C = c(15, 16, 17, 18, 19)**

**)**

**# Reshape data for heatmap**

**library(reshape2)**

**melted\_data <- melt(temperature\_data, id.vars = "Month")**

**# Plot heatmap**

**ggplot(melted\_data, aes(x = Month, y = variable, fill = value)) +**

**geom\_tile() +**

**scale\_fill\_gradient(low = "white", high = "red") +**

**labs(title = "Monthly Temperatures",**

**x = "Month",**

**y = "City") +**

**theme\_minimal()**

**10. Facet Plot**

**Dataset: Sales by Month and Region**

|  |  |  |
| --- | --- | --- |
| Month | Region | Sales |
| Jan | East | 200 |
| Jan | West | 150 |
| Feb | East | 220 |
| Feb | West | 170 |
| Mar | East | 210 |
| Mar | West | 160 |

**# Create a data frame**

**sales\_region\_data <- data.frame(**

**Month = c("Jan", "Jan", "Feb", "Feb", "Mar", "Mar"),**

**Region = c("East", "West", "East", "West", "East", "West"),**

**Sales = c(200, 150, 220, 170, 210, 160)**

**)**

**# Plot facet plot**

**ggplot(sales\_region\_data, aes(x = Month, y = Sales, fill = Region)) +**

**geom\_bar(stat = "identity", position = "dodge") +**

**facet\_wrap(~ Region) +**

**labs(title = "Sales by Month and Region",**

**x = "Month",**

**y = "Sales") +**

**theme\_minimal()**

**11. Area Plot**

**Dataset: Yearly Revenue**

|  |  |
| --- | --- |
| **Year** | **Revenue (in million)** |
| 2015 | 30 |
| 2016 | 35 |
| 2017 | 40 |
| 2018 | 45 |
| 2019 | 50 |

# Create a data frame

revenue\_data <- data.frame(

Year = 2015:2019,

Revenue = c(30, 35, 40, 45, 50)

)

# Plot area plot

ggplot(revenue\_data, aes(x = Year, y = Revenue)) +

geom\_area(fill = "skyblue", alpha = 0.5) +

labs(title = "Yearly Revenue",

x = "Year",

y = "Revenue (in million)") +

theme\_minimal()

**12. Step Plot**

**Dataset: Cumulative Sales**

|  |  |
| --- | --- |
| **Month** | **Cumulative Sales** |
| Jan | 100 |
| Feb | 200 |
| Mar | 300 |
| Apr | 400 |
| May | 500 |

# Load necessary library

library(ggplot2)

# Create a data frame

cumulative\_sales\_data <- data.frame(

Month = factor(c("Jan", "Feb", "Mar", "Apr", "May"), levels = c("Jan", "Feb", "Mar", "Apr", "May")),

Cumulative\_Sales = c(100, 200, 300, 400, 500)

)

# Create the step plot

ggplot(cumulative\_sales\_data, aes(x = Month, y = Cumulative\_Sales, group = 1)) +

geom\_step() +

labs(title = "Cumulative Sales Over Months",

x = "Month",

y = "Cumulative Sales") +

theme\_minimal()

**13. Ridgeline Plot**

**Dataset: Temperature Distributions**

|  |  |
| --- | --- |
| **City** | **Temperature** |
| City1 | 20 |
| City1 | 21 |
| City1 | 19 |
| City2 | 22 |
| City2 | 23 |
| City2 | 24 |

# Load necessary library

library(ggridges)

# Create a data frame

temperature\_data <- data.frame(

City = c("City1", "City1", "City1", "City2", "City2", "City2"),

Temperature = c(20, 21, 19, 22, 23, 24)

)

# Plot ridgeline plot

ggplot(temperature\_data, aes(x = Temperature, y = City, fill = City)) +

geom\_density\_ridges() +

labs(title = "Temperature Distributions",

x = "Temperature",

y = "City") +

theme\_minimal()

**14. Dumbbell Plot**

**Dataset: Sales Before and After Campaign**

|  |  |  |
| --- | --- | --- |
| **Product** | **Before Campaign** | **After Campaign** |
| A | 200 | 250 |
| B | 300 | 350 |
| C | 400 | 450 |
| D | 500 | 550 |
| E | 600 | 650 |

# Load necessary library

library(ggalt)

# Create a data frame

sales\_campaign\_data <- data.frame(

Product = c("A", "B", "C"),

Before = c(200, 300, 400),

After = c(250, 350, 450)

)

# Plot dumbbell plot

ggplot(sales\_campaign\_data) +

geom\_dumbbell(aes(x = Before, xend = After, y = Product), size = 3, color = "gray", size\_x = 3, size\_xend = 3) +

labs(title = "Sales Before and After Campaign",

x = "Sales",

y = "Product") +

theme\_minimal()

**15. Lollipop Plot**

**Dataset: Sales by Region**

|  |  |
| --- | --- |
| **Region** | **Sales** |
| North | 150 |
| South | 200 |
| East | 180 |
| West | 210 |

# Create a data frame

sales\_region\_data <- data.frame(

Region = c("North", "South", "East", "West"),

Sales = c(150, 200, 180, 210)

)

# Plot lollipop plot

ggplot(sales\_region\_data, aes(x = Region, y = Sales)) +

geom\_segment(aes(x = Region, xend = Region, y = 0, yend = Sales), color = "gray") +

geom\_point(color = "blue", size = 5) +

labs(title = "Sales by Region",

x = "Region",

y = "Sales") +

theme\_minimal()

**16. Spaghetti Plot**

**Dataset: Sales over Time by Product**

|  |  |  |
| --- | --- | --- |
| **Month** | **Product** | **Sales** |
| Jan | A | 100 |
| Feb | A | 150 |
| Mar | A | 130 |
| Jan | B | 120 |
| Feb | B | 140 |
| Mar | B | 160 |

# Create a data frame

sales\_time\_data <- data.frame(

Month = c("Jan", "Feb", "Mar", "Jan", "Feb", "Mar"),

Product = c("A", "A", "A", "B", "B", "B"),

Sales = c(100, 150, 130, 120, 140, 160)

)

# Plot spaghetti plot

ggplot(sales\_time\_data, aes(x = Month, y = Sales, group = Product, color = Product)) +

geom\_line(size = 1) +

geom\_point(size = 3) +

labs(title = "Sales over Time by Product",

x = "Month",

y = "Sales") +

theme\_minimal()

**17. Waterfall Plot**

**Dataset: Profit and Loss by Month**

|  |  |
| --- | --- |
| **Month** | **Amount** |
| Jan | 100 |
| Feb | -20 |
| Mar | 50 |
| Apr | -10 |
| May | 80 |

# Load necessary library

library(waterfalls)

# Create a data frame with the appropriate structure

profit\_loss\_data <- data.frame(

labels = c("Jan", "Feb", "Mar", "Apr", "May"),

values = c(100, -20, 50, -10, 80)

)

# Plot waterfall plot

waterfall(profit\_loss\_data, fill\_by\_sign = TRUE)

**18. Hexbin Plot**

**Dataset: X and Y Coordinates**

|  |  |
| --- | --- |
| **X** | **Y** |
| 1 | 2 |
| 2 | 3 |
| 3 | 4 |
| 4 | 5 |
| 5 | 6 |
| 6 | 7 |
| 7 | 8 |
| 8 | 9 |

# Load necessary library

library(hexbin)

# Create a data frame

xy\_data <- data.frame(

X = c(1, 2, 3, 4, 5, 6, 7, 8),

Y = c(2, 3, 4, 5, 6, 7, 8, 9)

)

# Plot hexbin plot

ggplot(xy\_data, aes(x = X, y = Y)) +

geom\_hex() +

labs(title = "Hexbin Plot",

x = "X",

y = "Y") +

theme\_minimal()

**19. Chord Diagram**

**Dataset: Connections between Categories**

|  |  |  |
| --- | --- | --- |
| **From** | **To** | **Value** |
| A | B | 10 |
| A | C | 20 |
| B | C | 15 |
| B | D | 25 |
| C | D | 30 |

# Load necessary library

library(circlize)

# Create a data frame

chord\_data <- data.frame(

From = c("A", "A", "B", "B", "C"),

To = c("B", "C", "C", "D", "D"),

Value = c(10, 20, 15, 25, 30)

)

# Plot chord diagram

chordDiagram(chord\_data)

**20. Calendar Heatmap**

**Dataset: Daily Activity Count**

|  |  |
| --- | --- |
| **Date** | **Count** |
| 2023-01-01 | 10 |
| 2023-01-02 | 12 |
| 2023-01-03 | 15 |
| 2023-01-04 | 8 |
| 2023-01-05 | 20 |

# Load necessary library

library(lubridate)

library(ggplot2)

# Create a data frame

activity\_data <- data.frame(

Date = as.Date(c("2023-01-01", "2023-01-02", "2023-01-03", "2023-01-04", "2023-01-05")),

Count = c(10, 12, 15, 8, 20)

)

# Add weekday and week of year columns

activity\_data$Weekday <- wday(activity\_data$Date, label = TRUE)

activity\_data$Week <- week(activity\_data$Date)

# Plot calendar heatmap

ggplot(activity\_data, aes(x = Weekday, y = Week, fill = Count)) +

geom\_tile(color = "white") +

scale\_fill\_gradient(low = "white", high = "blue") +

labs(title = "Daily Activity Count",

x = "Weekday",

y = "Week",

fill = "Count") +

theme\_minimal()

**21. Dataset: Weekly Temperature Data**

|  |  |  |
| --- | --- | --- |
| **Week** | **City** | **Average Temperature (°C)** |
| **2023-01-01** | **New York** | **-2** |
| **2023-01-01** | **Los Angeles** | **15** |
| **2023-01-01** | **Chicago** | **-5** |
| **2023-01-08** | **New York** | **0** |
| **2023-01-08** | **Los Angeles** | **17** |
| **2023-01-08** | **Chicago** | **-3** |
| **2023-01-15** | **New York** | **2** |
| **2023-01-15** | **Los Angeles** | **18** |
| **2023-01-15** | **Chicago** | **-2** |

**Visualization: Line Plot with Multiple LinesGoal: To compare the weekly average temperatures of three cities.**

library(ggplot2)

data <- data.frame(

Week = as.Date(c("2023-01-01", "2023-01-01", "2023-01-01",

"2023-01-08", "2023-01-08", "2023-01-08",

"2023-01-15", "2023-01-15", "2023-01-15")),

City = c("New York", "Los Angeles", "Chicago",

"New York", "Los Angeles", "Chicago",

"New York", "Los Angeles", "Chicago"),

AvgTemp = c(-2, 15, -5, 0, 17, -3, 2, 18, -2)

)

ggplot(data, aes(x = Week, y = AvgTemp, color = City, group = City)) +

geom\_line() +

geom\_point() +

labs(title = "Weekly Average Temperatures",

x = "Week",

y = "Average Temperature (°C)") +

theme\_minimal()

**22: Social Media Engagement Analysis**

**Dataset: Daily Social Media Engagement**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Likes** | **Comments** | **Shares** |
| **2023-01-01** | **150** | **20** | **30** |
| **2023-01-02** | **200** | **25** | **35** |
| **2023-01-03** | **250** | **30** | **40** |
| **2023-01-04** | **300** | **35** | **45** |
| **2023-01-05** | **350** | **40** | **50** |
| **2023-01-06** | **400** | **45** | **55** |
| **2023-01-07** | **450** | **50** | **60** |

**Visualization: Hexbin Plot**

**Goal: To visualize the daily engagement on a social media platform.**

**library(ggplot2)**

**library(hexbin)**

**data <- data.frame(**

**Date = as.Date(c('2023-01-01', '2023-01-02', '2023-01-03', '2023-01-04', '2023-01-05', '2023-01-06', '2023-01-07')),**

**Likes = c(150, 200, 250, 300, 350, 400, 450),**

**Comments = c(20, 25, 30, 35, 40, 45, 50),**

**Shares = c(30, 35, 40, 45, 50, 55, 60)**

**)**

**library(reshape2)**

**data\_long <- melt(data, id.vars = "Date")**

**ggplot(data\_long, aes(x = Date, y = value)) +**

**geom\_hex(bins = 30) +**

**facet\_wrap(~ variable, scales = "free\_y") +**

**scale\_fill\_viridis\_c() +**

**theme\_minimal() +**

**labs(title = "Daily Social Media Engagement",**

**x = "Date",**

**y = "Engagement Count",**

**fill = "Count")**

**23. Sales Performance Visualization**

**Dataset: Monthly Sales Data**

|  |  |  |  |
| --- | --- | --- | --- |
| **Month** | **Product A Sales** | **Product B Sales** | **Product C Sales** |
| **Jan** | **500** | **300** | **200** |
| **Feb** | **600** | **320** | **250** |
| **Mar** | **700** | **350** | **300** |
| **Apr** | **800** | **400** | **350** |
| **May** | **750** | **420** | **400** |
| **Jun** | **850** | **450** | **420** |
| **Jul** | **900** | **480** | **450** |
| **Aug** | **950** | **500** | **470** |
| **Sep** | **1000** | **550** | **500** |
| **Oct** | **1050** | **580** | **530** |
| **Nov** | **1100** | **600** | **550** |

**Visualization: Violin Plot Goal: To visualize the sales performance of three products over a year**

**library(ggplot2)**

**library(reshape2)**

**sales\_data <- data.frame(**

**Month = c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov"),**

**Product\_A\_Sales = c(500, 600, 700, 800, 750, 850, 900, 950, 1000, 1050, 1100),**

**Product\_B\_Sales = c(300, 320, 350, 400, 420, 450, 480, 500, 550, 580, 600),**

**Product\_C\_Sales = c(200, 250, 300, 350, 400, 420, 450, 470, 500, 530, 550)**

**)**

**sales\_data\_long <- melt(sales\_data, id.vars = "Month",**

**variable.name = "Product", value.name = "Sales")**

**ggplot(sales\_data\_long, aes(x = Product, y = Sales, fill = Product)) +**

**geom\_violin(trim = FALSE) +**

**geom\_jitter(width = 0.2, size = 2, alpha = 0.5) +**

**labs(title = "Sales Performance of Products A, B, and C",**

**x = "Product", y = "Sales") +**

**theme\_minimal() +**

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

**theme(legend.position = "none")**

**24. Customer Satisfaction Survey Results**

**Dataset: Satisfaction Ratings**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Category** | **Very Dissatisfied** | **Dissatisfied** | **Neutral** | **Satisfied** | **Very Satisfied** |
| **Product Quality** | **5** | **15** | **25** | **35** | **20** |
| **Customer Service** | **10** | **20** | **30** | **25** | **15** |
| **Value for Money** | **8** | **12** | **28** | **32** | **20** |

**Visualization: Stacked Bar Plot.Goal: To visualize the distribution of customer satisfaction ratings across different categories.**

**# Load necessary library**

**library(ggplot2)**

**satisfaction\_data <- data.frame(**

**Category = c("Product Quality", "Customer Service", "Value for Money"),**

**Very\_Dissatisfied = c(5, 10, 8),**

**Dissatisfied = c(15, 20, 12),**

**Neutral = c(25, 30, 28),**

**Satisfied = c(35, 25, 32),**

**Very\_Satisfied = c(20, 15, 20)**

**)**

**satisfaction\_long <- reshape2::melt(satisfaction\_data, id.vars = "Category",**

**variable.name = "Satisfaction\_Level",**

**value.name = "Count")**

**ggplot(satisfaction\_long, aes(x = Category, y = Count, fill = Satisfaction\_Level)) +**

**geom\_bar(stat = "identity") +**

**labs(title = "Customer Satisfaction Survey Results",**

**x = "Category",**

**y = "Count",**

**fill = "Satisfaction Level") +**

**theme\_minimal() +**

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

**25. Employee Performance Evaluation**

**Dataset: Monthly Performance Scores**

|  |  |  |
| --- | --- | --- |
| **Month** | **Employee** | **Score** |
| **Jan** | **A** | **85** |
| **Jan** | **B** | **90** |
| **Jan** | **C** | **78** |
| **Feb** | **A** | **88** |
| **Feb** | **B** | **92** |
| **Feb** | **C** | **80** |
| **Mar** | **A** | **90** |
| **Mar** | **B** | **94** |
| **Mar** | **C** | **82** |

**Visualization: Histogram Plot.Goal: To compare the monthly performance scores of three employees.**

**library(ggplot2)**

**data <- data.frame(**

**Month = rep(c("Jan", "Feb", "Mar"), each = 3),**

**Employee = rep(c("A", "B", "C"), 3),**

**Score = c(85, 90, 80, 88, 92, 90, 90, 80, 82)**

**)**

**ggplot(data, aes(x = Score, fill = Employee)) +**

**geom\_histogram(binwidth = 2, color = "black") +**

**labs(title = "Monthly Performance Scores of Employees",**

**x = "Score",**

**y = "Count") +**

**theme\_minimal()**

**26. Weather Data**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Temperature (°C)** | **Humidity (%)** | **Wind Speed (km/h)** |
| **2023-01-01** | **10** | **75** | **15** |
| **2023-01-02** | **12** | **70** | **12** |
| **2023-01-03** | **8** | **80** | **18** |
| **2023-01-04** | **15** | **65** | **20** |
| **2023-01-05** | **14** | **72** | **16** |

1. **How does temperature vary with humidity and wind speed over the recorded dates?**
2. **Can we visualize the relationship between wind speed and humidity, considering temperature as the third dimension?**
3. **Is there any discernible pattern between temperature, humidity, and wind speed throughout the dataset?**
4. **Create a 3D surface plot to show how temperature changes with varying levels of humidity and wind speed.**
5. **Compare the 3D plots of temperature against both humidity and wind speed separately. Are there any notable differences in trends?**

**# Load necessary libraries**

**library(ggplot2)**

**library(plotly)**

**library(reshape2)**

**# Data**

**data <- data.frame(**

**Date = as.Date(c('2023-01-01', '2023-01-02', '2023-01-03', '2023-01-04', '2023-01-05')),**

**Temperature = c(10, 12, 8, 15, 14),**

**Humidity = c(75, 70, 80, 65, 72),**

**WindSpeed = c(15, 12, 18, 20, 16)**

**)**

**# 1. How does temperature vary with humidity and wind speed over the recorded dates?**

**correlation\_matrix <- cor(data[, c("Temperature", "Humidity", "WindSpeed")])**

**print(correlation\_matrix)**

**# 2. Visualize the relationship between wind speed and humidity, considering temperature as the third dimension**

**ggplot(data, aes(x = WindSpeed, y = Humidity, color = Temperature)) +**

**geom\_point(size = 4) +**

**scale\_color\_gradient(low = "blue", high = "red") +**

**labs(title = "Wind Speed vs Humidity (Temperature as Color)",**

**x = "Wind Speed (km/h)", y = "Humidity (%)", color = "Temperature (°C)")**

**# 3. Discernible pattern between temperature, humidity, and wind speed**

**# This can be observed from the correlation matrix printed earlier**

**# 4. Create a 3D surface plot to show how temperature changes with varying levels of humidity and wind speed**

**# Reshape the data for 3D plotting**

**humidity\_values <- unique(data$Humidity)**

**wind\_speed\_values <- unique(data$WindSpeed)**

**temperature\_matrix <- matrix(data$Temperature, nrow = length(humidity\_values), ncol = length(wind\_speed\_values))**

**fig <- plot\_ly(x = humidity\_values, y = wind\_speed\_values, z = temperature\_matrix, type = 'surface')**

**fig <- fig %>%**

**layout(scene = list(xaxis = list(title = 'Humidity (%)'),**

**yaxis = list(title = 'Wind Speed (km/h)'),**

**zaxis = list(title = 'Temperature (°C)')),**

**title = '3D Surface Plot: Temperature vs Humidity and Wind Speed')**

**fig**

**# 5. Compare the 3D plots of temperature against both humidity and wind speed separately**

**# 3D plot: Temperature vs Humidity**

**fig1 <- plot\_ly(data, x = ~Humidity, y = ~Temperature, z = ~WindSpeed, type = 'scatter3d', mode = 'markers',**

**marker = list(size = 5, color = ~Temperature, colorscale = 'Viridis'))**

**fig1 <- fig1 %>%**

**layout(scene = list(xaxis = list(title = 'Humidity (%)'),**

**yaxis = list(title = 'Temperature (°C)'),**

**zaxis = list(title = 'Wind Speed (km/h)')),**

**title = '3D Plot: Temperature vs Humidity')**

**fig1**

**# 3D plot: Temperature vs Wind Speed**

**fig2 <- plot\_ly(data, x = ~WindSpeed, y = ~Temperature, z = ~Humidity, type = 'scatter3d', mode = 'markers',**

**marker = list(size = 5, color = ~Temperature, colorscale = 'Viridis'))**

**fig2 <- fig2 %>%**

**layout(scene = list(xaxis = list(title = 'Wind Speed (km/h)'),**

**yaxis = list(title = 'Temperature (°C)'),**

**zaxis = list(title = 'Humidity (%)')),**

**title = '3D Plot: Temperature vs Wind Speed')**

**fig2**

**27. Financial Market Data**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Stock Price ($)** | **Volume Traded (millions)** | **Market Cap ($)** |
| **2023-01-01** | **100** | **2.5** | **500** |
| **2023-01-02** | **102** | **3.0** | **510** |
| **2023-01-03** | **98** | **2.2** | **490** |
| **2023-01-04** | **105** | **2.8** | **525** |
| **2023-01-05** | **108** | **3.5** | **540** |

1. **How does stock price relate to both volume traded and market capitalization?**
2. **Create a 3D scatter plot to visualize the relationship between volume traded, market cap, and stock price.**
3. **Can we see any clustering or outliers in the 3D plot of stock price, volume traded, and market cap?**
4. **Generate a 3D surface plot to illustrate how market capitalization changes with variations in stock price and volume traded.**
5. **Compare the 3D plots of stock price against both volume traded and market cap separately. Are there any noticeable patterns?**

**library(scatterplot3d)**

**library(plotly)**

**# Create data frame**

**data <- data.frame(**

**Date = as.Date(c("2023-01-01", "2023-01-02", "2023-01-03", "2023-01-04", "2023-01-05")),**

**StockPrice = c(100, 102, 98, 105, 108),**

**VolumeTraded = c(2.5, 3.0, 2.2, 2.8, 3.5),**

**MarketCap = c(500, 510, 490, 525, 540)**

**)**

**# Correlation analysis**

**cor(data$StockPrice, data$VolumeTraded)**

**cor(data$StockPrice, data$MarketCap)**

**# 3D scatter plot**

**scatterplot3d(data$VolumeTraded, data$MarketCap, data$StockPrice, pch = 19,**

**xlab = "Volume Traded (millions)", ylab = "Market Cap ($)", zlab = "Stock Price ($)",**

**main = "3D Scatter Plot")**

**# 3D scatter plot using plotly for interactive visualization**

**plot\_ly(data, x = ~VolumeTraded, y = ~MarketCap, z = ~StockPrice, type = 'scatter3d', mode = 'markers') %>%**

**layout(scene = list(xaxis = list(title = 'Volume Traded (millions)'),**

**yaxis = list(title = 'Market Cap ($)'),**

**zaxis = list(title = 'Stock Price ($)'),**

**title = '3D Scatter Plot'))**

**# 3D surface plot**

**surface\_data <- expand.grid(VolumeTraded = seq(min(data$VolumeTraded), max(data$VolumeTraded), length.out = 50),**

**StockPrice = seq(min(data$StockPrice), max(data$StockPrice), length.out = 50))**

**surface\_data$MarketCap <- 500 \* surface\_data$StockPrice / 100**

**plot\_ly(surface\_data, x = ~VolumeTraded, y = ~StockPrice, z = ~MarketCap, type = 'surface') %>%**

**layout(scene = list(xaxis = list(title = 'Volume Traded (millions)'),**

**yaxis = list(title = 'Stock Price ($)'),**

**zaxis = list(title = 'Market Cap ($)'),**

**title = '3D Surface Plot'))**

**# 3D scatter plots for Stock Price vs Volume Traded and Market Cap separately**

**plot\_ly(data, x = ~VolumeTraded, y = ~StockPrice, z = ~MarketCap, type = 'scatter3d', mode = 'markers') %>%**

**layout(scene = list(xaxis = list(title = 'Volume Traded (millions)'),**

**yaxis = list(title = 'Stock Price ($)'),**

**zaxis = list(title = 'Market Cap ($)'),**

**title = 'Stock Price vs Volume Traded'))**

**plot\_ly(data, x = ~MarketCap, y = ~StockPrice, z = ~VolumeTraded, type = 'scatter3d', mode = 'markers') %>%**

**layout(scene = list(xaxis = list(title = 'Market Cap ($)'),**

**yaxis = list(title = 'Stock Price ($)'),**

**zaxis = list(title = 'Volume Traded (millions)'),**

**title = 'Stock Price vs Market Cap'))**

**28. Consumer Survey Data**

|  |  |  |  |
| --- | --- | --- | --- |
| **Product** | **Price ($)** | **Rating (out of 5)** | **Age Group** |
| **A** | **50** | **4.2** | **25-35** |
| **B** | **70** | **3.8** | **35-45** |
| **C** | **60** | **4.0** | **18-25** |
| **D** | **45** | **4.5** | **45-55** |
| **E** | **55** | **3.9** | **25-35** |

1. **How do product ratings vary with both price and age group?**
2. **Create a 3D scatter plot to visualize the relationship between price, rating, and age group.**
3. **Is there a correlation between age group, product price, and consumer ratings based on the 3D plot?**
4. **Generate a 3D surface plot to show how product ratings change with variations in both price and age group.**
5. **Compare the 3D plots of product ratings against both price and age group separately. Are there any significant insights?**

**library(plotly)**

**library(akima)**

**# Create the data frame**

**data <- data.frame(**

**Product = c("A", "B", "C", "D", "E"),**

**Price = c(50, 70, 60, 45, 55),**

**Rating = c(4.2, 3.8, 4.0, 4.5, 3.9),**

**AgeGroup = c("25-35", "35-45", "18-25", "45-55", "25-35")**

**)**

**# Map age groups to numerical values for plotting**

**data$AgeGroupNum <- as.numeric(factor(data$AgeGroup, levels = unique(data$AgeGroup)))**

**# Create 3D scatter plot**

**scatter\_plot <- plot\_ly(data, x = ~Price, y = ~Rating, z = ~AgeGroupNum, type = "scatter3d", mode = "markers",**

**marker = list(size = 5, color = ~Rating, colorscale = "Viridis", showscale = TRUE)) %>%**

**layout(scene = list(xaxis = list(title = "Price ($)"),**

**yaxis = list(title = "Rating"),**

**zaxis = list(title = "Age Group")))**

**scatter\_plot**

**# Check correlation**

**correlation\_matrix <- cor(data[, c("Price", "Rating", "AgeGroupNum")])**

**print(correlation\_matrix)**

**# Example data for surface plot, you might need to adjust this for your actual data**

**# Create grid for interpolation**

**interp\_data <- with(data, interp(x = Price, y = AgeGroupNum, z = Rating, duplicate = "mean"))**

**# Create 3D surface plot**

**surface\_plot <- plot\_ly(z = ~interp\_data$z, x = ~interp\_data$x, y = ~interp\_data$y, type = "surface") %>%**

**layout(scene = list(xaxis = list(title = "Price ($)"),**

**yaxis = list(title = "Age Group"),**

**zaxis = list(title = "Rating")))**

**surface\_plot**

**# 3D scatter plot: Rating vs Price vs Age Group**

**scatter\_plot\_price\_age <- plot\_ly(data, x = ~Price, y = ~Rating, z = ~AgeGroupNum, type = "scatter3d", mode = "markers",**

**marker = list(size = 5, color = ~Rating, colorscale = "Viridis", showscale = TRUE)) %>%**

**layout(scene = list(xaxis = list(title = "Price ($)"),**

**yaxis = list(title = "Rating"),**

**zaxis = list(title = "Age Group")))**

**scatter\_plot\_price\_age**

**# 3D scatter plot: Rating vs Age Group vs Price**

**scatter\_plot\_age\_price <- plot\_ly(data, x = ~AgeGroupNum, y = ~Rating, z = ~Price, type = "scatter3d", mode = "markers",**

**marker = list(size = 5, color = ~Rating, colorscale = "Viridis", showscale = TRUE)) %>%**

**layout(scene = list(xaxis = list(title = "Age Group"),**

**yaxis = list(title = "Rating"),**

**zaxis = list(title = "Price ($)")))**

**scatter\_plot\_age\_price**

**29. Environmental Data**

|  |  |  |  |
| --- | --- | --- | --- |
| **Location** | **Temperature (°C)** | **Humidity (%)** | **CO2 Levels (ppm)** |
| **A** | **15** | **65** | **400** |
| **B** | **20** | **70** | **450** |
| **C** | **18** | **68** | **420** |
| **D** | **12** | **60** | **380** |
| **E** | **17** | **72** | **430** |

1. **How do CO2 levels vary with both temperature and humidity across different locations?**
2. **Create a 3D scatter plot to visualize the relationship between temperature, humidity, and CO2 levels.**
3. **Is there a spatial pattern in the 3D plot of temperature, humidity, and CO2 levels among the locations?**
4. **Generate a 3D surface plot to illustrate how CO2 levels change with variations in both temperature and humidity.**
5. **Compare the 3D plots of CO2 levels against both temperature and humidity separately. Are there any noticeable trends?**

**library(plotly)**

**library(akima)**

**# Data**

**location <- c("A", "B", "C", "D", "E")**

**temperature <- c(15, 20, 18, 12, 17)**

**humidity <- c(65, 70, 68, 60, 72)**

**CO2\_levels <- c(400, 450, 420, 380, 430)**

**# Create a data frame**

**data <- data.frame(location, temperature, humidity, CO2\_levels)**

**# 3D Scatter Plot (Temperature, Humidity, CO2 Levels)**

**fig\_scatter <- plot\_ly(data, x = ~temperature, y = ~humidity, z = ~CO2\_levels, type = 'scatter3d', mode = 'markers', text = ~location)**

**fig\_scatter <- fig\_scatter %>% layout(scene = list(xaxis = list(title = 'Temperature (°C)'),**

**yaxis = list(title = 'Humidity (%)'),**

**zaxis = list(title = 'CO2 Levels (ppm)')))**

**fig\_scatter**

**# Interpolate data for 3D Surface Plot**

**interp\_data <- with(data, interp(x = temperature, y = humidity, z = CO2\_levels, duplicate = "mean"))**

**# 3D Surface Plot**

**fig\_surface <- plot\_ly(x = interp\_data$x, y = interp\_data$y, z = interp\_data$z, type = "surface")**

**fig\_surface <- fig\_surface %>% layout(scene = list(xaxis = list(title = 'Temperature (°C)'),**

**yaxis = list(title = 'Humidity (%)'),**

**zaxis = list(title = 'CO2 Levels (ppm)')))**

**fig\_surface**

**# 3D Scatter Plot (CO2 Levels vs Temperature)**

**fig\_temp <- plot\_ly(data, x = ~temperature, y = ~CO2\_levels, z = ~humidity, type = 'scatter3d', mode = 'markers', text = ~location)**

**fig\_temp <- fig\_temp %>% layout(scene = list(xaxis = list(title = 'Temperature (°C)'),**

**yaxis = list(title = 'CO2 Levels (ppm)'),**

**zaxis = list(title = 'Humidity (%)')))**

**fig\_temp**

**# 3D Scatter Plot (CO2 Levels vs Humidity)**

**fig\_humidity <- plot\_ly(data, x = ~humidity, y = ~CO2\_levels, z = ~temperature, type = 'scatter3d', mode = 'markers', text = ~location)**

**fig\_humidity <- fig\_humidity %>% layout(scene = list(xaxis = list(title = 'Humidity (%)'),**

**yaxis = list(title = 'CO2 Levels (ppm)'),**

**zaxis = list(title = 'Temperature (°C)')))**

**fig\_humidity**

**30. Academic Performance Data**

|  |  |  |  |
| --- | --- | --- | --- |
| **Student** | **Math Score** | **Science Score** | **Attendance (%)** |
| **A** | **85** | **78** | **95** |
| **B** | **72** | **85** | **92** |
| **C** | **90** | **80** | **98** |
| **D** | **78** | **75** | **85** |
| **E** | **88** | **82** | **93** |

1. **How do science scores vary with both math scores and attendance percentage?**
2. **Create a 3D scatter plot to visualize the relationship between math score, science score, and attendance.**
3. **Is there a correlation between attendance, math scores, and science scores based on the 3D plot?**
4. **Generate a 3D surface plot to show how science scores change with variations in both math scores and attendance.**
5. **Compare the 3D plots of science scores against both math scores and attendance separately. Are there any significant patterns or outliers?**

**library(plotly)**

**library(scatterplot3d)**

**# Create the data frame**

**students\_data <- data.frame(**

**Student = c('A', 'B', 'C', 'D', 'E'),**

**Math\_Score = c(85, 72, 90, 78, 88),**

**Science\_Score = c(78, 85, 80, 75, 82),**

**Attendance = c(95, 92, 98, 85, 93)**

**)**

**# 3D Scatter plot**

**scatter\_plot <- plot\_ly(students\_data, x = ~Math\_Score, y = ~Science\_Score, z = ~Attendance,**

**type = 'scatter3d', mode = 'markers') %>%**

**layout(scene = list(xaxis = list(title = 'Math Score'),**

**yaxis = list(title = 'Science Score'),**

**zaxis = list(title = 'Attendance (%)')))**

**# Display the scatter plot**

**scatter\_plot**

**# Calculate correlation**

**correlation\_matrix <- cor(students\_data[, c("Math\_Score", "Science\_Score", "Attendance")])**

**print(correlation\_matrix)**

**# 3D Surface plot**

**surface\_plot <- plot\_ly(students\_data, x = ~Math\_Score, y = ~Attendance, z = ~Science\_Score,**

**type = 'mesh3d') %>%**

**layout(scene = list(xaxis = list(title = 'Math Score'),**

**yaxis = list(title = 'Attendance (%)'),**

**zaxis = list(title = 'Science Score')))**

**# Display the surface plot**

**surface\_plot**

**# Compare Science Score vs Math Score and Attendance**

**# Science Score vs Math Score**

**scatterplot3d(students\_data$Math\_Score, students\_data$Science\_Score, angle = 45, pch = 16, color = "blue",**

**xlab = "Math Score", ylab = "Science Score", zlab = "Attendance (%)")**

**# Science Score vs Attendance**

**scatterplot3d(students\_data$Attendance, students\_data$Science\_Score, angle = 45, pch = 16, color = "red",**

**xlab = "Attendance (%)", ylab = "Science Score", zlab = "Math Score")**