DSA 0603 Data Handling and Visualization for Histogram Analysis

**List of Programs in R and Tableau**

1. **Monthly Sales Data**

|  |  |
| --- | --- |
| **Month** | **Sales (in $)** |
| January | 15000 |
| February | 18000 |
| March | 22000 |
| April | 20000 |
| May | 23000 |

1. Using R Create a line chart to visualize the monthly sales. Label the axes and tit the chart appropriately.
2. Using R Generate a bar chart to display the top-selling products for the year. Label the chart elements.
3. Using R Develop a scatter plot to explore the relationship between advertising budget and monthly sales. Explain the insights drawn from the scatter plot.

# Assuming you have a data frame named 'sales\_data' with columns: Month, Sales

# If not, you need to create a data frame first with your data.

# Sample Data

sales\_data <- data.frame(

Month = c("January", "February", "March", "April", "May"),

Sales = c(15000, 18000, 22000, 20000, 23000)

)

# a. Line chart for monthly sales

plot( sales\_data$Sales, type = "o", col = "blue", main = "Monthly Sales",xlab = "Month",ylab = "Sales (in $)")

# b. Bar chart for top-selling products

# In this case, we assume the data includes product names and their corresponding sales.

# Let's create some sample data for illustration.

product\_sales <- data.frame(

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

Sales = c(25000, 22000, 20000, 18000, 15000)

)

barplot(product\_sales$Sales, names.arg = product\_sales$Product, col = "green",

main = "Top-Selling Products",

xlab = "Product",

ylab = "Sales (in $)")

# c. Scatter plot for advertising budget vs. monthly sales

# Assuming you have data for advertising budget, let's create some sample data.

advertising\_data <- data.frame(

Month = c("January", "February", "March", "April", "May"),

Advertising\_Budget = c(5000, 6000, 7000, 5500, 8000)

)

# Combine the sales and advertising data

combined\_data <- merge(sales\_data, advertising\_data, by = "Month")

# Scatter plot

plot(combined\_data$Advertising\_Budget, combined\_data$Sales, col = "red",

main = "Scatter Plot: Advertising Budget vs. Monthly Sales",

xlab = "Advertising Budget (in $)",

ylab = "Sales (in $)")

# Add a trend line

abline(lm(combined\_data$Sales ~ combined\_data$Advertising\_Budget), col = "blue")

2.**Customer Feedback Analysis**

|  |  |  |
| --- | --- | --- |
| **Customer ID** | **Age** | **Satisfaction Score** |
| 1 | 25 | 4 |
| 2 | 30 | 5 |
| 3 | 35 | 3 |
| 4 | 28 | 4 |
| 5 | 40 | 5 |

1. Using R Create a histogram to represent the distribution of customer ages. Label the axes and title the chart.
2. Using R Generate a pie chart to display the overall distribution of customer satisfaction scores. Include labels.
3. Using R Build a stacked bar chart to visualize the distribution of customer satisfaction scores by age group

# Assuming you have a data frame named 'customer\_data' with columns: Customer\_ID, Age, Satisfaction\_Score

# If not, you need to create a data frame first with your data.

# Sample Data

customer\_data <- data.frame(

Customer\_ID = c(1, 2, 3, 4, 5),

Age = c(25, 30, 35, 28, 40),

Satisfaction\_Score = c(4, 5, 3, 4, 5)

)

# Load necessary libraries

library(ggplot2)

# a. Histogram for customer ages

ggplot(customer\_data, aes(x = Age)) +

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

labs(title = "Distribution of Customer Ages",

x = "Age",

y = "Frequency")

# b. Pie chart for distribution of overall customer satisfaction scores

satisfaction\_counts <- table(customer\_data$Satisfaction\_Score)

pie(satisfaction\_counts, labels = paste(names(satisfaction\_counts), ": ", satisfaction\_counts),

main = "Distribution of Customer Satisfaction Scores")

# c. Stacked bar chart for distribution of satisfaction scores by age group

# Create age groups

customer\_data <- customer\_data %>%

mutate(Age\_Group = cut(Age, breaks = c(20, 30, 40, 50), labels = c("20-30", "30-40", "40-50")))

# Create a stacked bar chart

ggplot(customer\_data, aes(x = Age\_Group, fill = factor(Satisfaction\_Score))) +

geom\_bar(position = "stack") +

labs(title = "Distribution of Customer Satisfaction Scores by Age Group",

x = "Age Group",

y = "Count") +

scale\_fill\_manual(values = c("3" = "red", "4" = "yellow", "5" = "green"))

1. **Employee Performance Evaluation**

|  |  |  |  |
| --- | --- | --- | --- |
| **Employee ID** | **Department** | **Years of Service** | **Performance Score** |
| 1 | Sales | 5 | 85 |
| 2 | HR | 3 | 92 |
| 3 | Marketing | 7 | 78 |
| 4 | Sales | 4 | 90 |
| 5 | HR | 2 | 76 |

1. Using R Create a line chart to visualize the performance trend of employees over time. Include a legend and labels.
2. Using R Generate a bar chart showing the distribution of employees across different departments. Label the chart elements.
3. Using R Build a scatter plot to analyse the correlation between years of service and performance scores. Explain any insights.

# Assuming you have a data frame named 'employee\_data' with columns: Employee\_ID, Department, Years\_of\_Service, Performance\_Score

# If not, you need to create a data frame first with your data.

# Sample Data

employee\_data <- data.frame(

Employee\_ID = c(1, 2, 3, 4, 5),

Department = c("Sales", "HR", "Marketing", "Sales", "HR"),

Years\_of\_Service = c(5, 3, 7, 4, 2),

Performance\_Score = c(85, 92, 78, 90, 76)

)

# Load necessary libraries

library(ggplot2)

# a. Line chart for performance trend over time

ggplot(employee\_data, aes(x = Years\_of\_Service, y = Performance\_Score, color = Department)) +

geom\_line() +

labs(title = "Performance Trend of Employees Over Time",

x = "Years of Service",

y = "Performance Score",

color = "Department") +

theme\_minimal()

# b. Bar chart for distribution of employees across different departments

ggplot(employee\_data, aes(x = Department, fill = Department)) +

geom\_bar() +

labs(title = "Distribution of Employees Across Departments",

x = "Department",

y = "Number of Employees") +

theme\_minimal()

# c. Scatter plot for correlation between years of service and performance scores

ggplot(employee\_data, aes(x = Years\_of\_Service, y = Performance\_Score)) +

geom\_point() +

labs(title = "Scatter Plot: Years of Service vs. Performance Score",

x = "Years of Service",

y = "Performance Score") +

theme\_minimal()

# Correlation analysis

correlation <- cor(employee\_data$Years\_of\_Service, employee\_data$Performance\_Score)

cat("Correlation between Years of Service and Performance Score:", correlation, "\n")

1. **Product Inventory Management**

|  |  |  |
| --- | --- | --- |
| **Product ID** | **Product Name** | **Quantity Available** |
| 1 | Product A | 250 |
| 2 | Product B | 175 |
| 3 | Product C | 300 |
| 4 | Product D | 200 |
| 5 | Product E | 220 |

1. Using R Create a bar chart to visualize the quantity of each product in the inventory. Label the axes and title the chart.
2. Using R Generate a stacked bar chart to show the quantity of each product within different product categories.
3. Using R Build a scatter plot to explore the relationship between product price and quantity available. Explain the findings.

**# Assuming you have a data frame named 'product\_data' with columns: Product\_ID, Product\_Name, Quantity\_Available**

**# If not, you need to create a data frame first with your data.**

**# Sample Data**

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

**Product\_ID = c(1, 2, 3, 4, 5),**

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

**Quantity\_Available = c(250, 175, 300, 200, 220)**

**)**

**# Load necessary libraries**

**library(ggplot2)**

**# a. Bar chart for quantity of each product in the inventory**

**ggplot(product\_data, aes(x = Product\_Name, y = Quantity\_Available, fill = Product\_Name)) +**

**geom\_bar(stat = "identity", color = "black") +**

**labs(title = "Quantity of Each Product in Inventory",**

**x = "Product Name",**

**y = "Quantity Available") +**

**theme\_minimal()**

**# b. Stacked bar chart for quantity of each product within different categories**

**# Assuming you have a column 'Product\_Category' in your data**

**product\_data$Product\_Category <- c("Category A", "Category B", "Category A", "Category B", "Category A")**

**ggplot(product\_data, aes(x = Product\_Category, y = Quantity\_Available, fill = Product\_Name)) +**

**geom\_bar(stat = "identity", color = "black") +**

**labs(title = "Quantity of Each Product within Categories",**

**x = "Product Category",**

**y = "Quantity Available") +**

**theme\_minimal()**

**# c. Scatter plot for relationship between product price and quantity available**

**# Assuming you have a column 'Product\_Price' in your data**

**product\_data$Product\_Price <- c(10, 15, 20, 18, 12)**

**ggplot(product\_data, aes(x = Product\_Price, y = Quantity\_Available)) +**

**geom\_point() +**

**labs(title = "Scatter Plot: Product Price vs. Quantity Available",**

**x = "Product Price",**

**y = "Quantity Available") +**

**theme\_minimal()**

**5. Website Analytics**

|  |  |  |
| --- | --- | --- |
| **Date** | **Page Views** | **Click-through Rate** |
| 2023-01-01 | 1500 | 2.3% |
| 2023-01-02 | 1600 | 2.7% |
| 2023-01-03 | 1400 | 2.0% |
| 2023-01-04 | 1650 | 2.4% |
| 2023-01-05 | 1800 | 2.6% |

1. Using R Create a line chart to visualize the trend in daily page views over time. Label the axes and title the chart.
2. Using R Generate a bar chart showing the top N days with the highest click-through rates. Label the chart elements.
3. Using R Develop a stacked area chart to display the distribution of user interactions (likes, shares, comments) on a website.

# Assuming you have a data frame named 'web\_data' with columns: Date, Page\_Views, Click\_through\_Rate

# If not, you need to create a data frame first with your data.

# Sample Data

web\_data <- data.frame(

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

Page\_Views = c(1500, 1600, 1400, 1650, 1800),

Click\_through\_Rate = c(2.3, 2.7, 2.0, 2.4, 2.6)

)

# Load necessary libraries

library(ggplot2)

library(dplyr)

# a. Line chart for trend in daily page views over time

ggplot(web\_data, aes(x = Date, y = Page\_Views)) +

geom\_line(color = "blue") +

labs(title = "Trend in Daily Page Views Over Time",

x = "Date",

y = "Page Views") +

theme\_minimal()

# b. Bar chart for top N days with highest click-through rates

top\_n\_days <- 3 # Change this to the desired number of top days

top\_days <- web\_data %>% arrange(desc(Click\_through\_Rate)) %>% head(top\_n\_days)

ggplot(top\_days, aes(x = Date, y = Click\_through\_Rate)) +

geom\_bar(stat = "identity", fill = "green", color = "black") +

labs(title = paste("Top", top\_n\_days, "Days with Highest Click-through Rates"),

x = "Date",

y = "Click-through Rate") +

theme\_minimal()

# c. Stacked area chart for distribution of user interactions

# Assuming you have a data frame named 'interaction\_data' with columns: Date, Likes, Shares, Comments

# If not, you need to create a data frame first with your data.

# Sample Interaction Data

interaction\_data <- data.frame(

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

Likes = c(50, 60, 45, 70, 80),

Shares = c(30, 40, 25, 35, 50),

Comments = c(20, 25, 18, 30, 40)

)

ggplot(interaction\_data, aes(x = Date, y = Count, fill = Interaction\_Type)) +

geom\_area() +

labs(title = "Distribution of User Interactions",

x = "Date",

y = "Count") +

theme\_minimal()

**6. Product Sales Analysis**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Product ID** | **Product Name** | **January Sales** | **February Sales** | **March Sales** |
| 1 | Product A | 2000 | 2200 | 2400 |
| 2 | Product B | 1500 | 1800 | 1600 |
| 3 | Product C | 1200 | 1400 | 1100 |

1. Using R Create a grouped bar chart to visualize the sales of each product for the first quarter. Label the chart elements.
2. Using R Generate a stacked area chart to represent the overall sales trend for all products over the three months.
3. Using R Build a table to show the monthly sales data for each product. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

ProductID = c(1, 2, 3),

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

JanuarySales = c(2000, 1500, 1200),

FebruarySales = c(2200, 1800, 1400),

MarchSales = c(2400, 1600, 1100)

)

# Reshape data for ggplot

data\_long <- tidyr::gather(data, key = "Month", value = "Sales", -ProductID, -ProductName)

# Create grouped bar chart

ggplot(data\_long, aes(x = ProductName, y = Sales, fill = Month)) +

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

labs(title = "Quarterly Sales by Product",

x = "Product",

y = "Sales",

fill = "Month")

# Create a data frame with the provided data

data <- data.frame(

ProductID = c(1, 2, 3),

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

JanuarySales = c(2000, 1500, 1200),

FebruarySales = c(2200, 1800, 1400),

MarchSales = c(2400, 1600, 1100)

)

# Reshape data for ggplot

data\_long <- tidyr::gather(data, key = "Month", value = "Sales", -ProductID, -ProductName)

# Create stacked area chart

ggplot(data\_long, aes(x = ProductName, y = Sales, fill = Month)) +

geom\_area() +

labs(title = "Overall Sales Trend",

x = "Product",

y = "Sales",

fill = "Month")

# Create a data frame with the provided data

data <- data.frame(

ProductID = c(1, 2, 3),

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

JanuarySales = c(2000, 1500, 1200),

FebruarySales = c(2200, 1800, 1400),

MarchSales = c(2400, 1600, 1100)

)

# Display the table

print(data)

7. **Customer Demographics Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| **Customer ID** | **Age** | **Gender** | **Income (in $)** |
| 1 | 28 | Female | 50000 |
| 2 | 35 | Male | 60000 |
| 3 | 42 | Female | 75000 |

1. Using R Create a bar chart to visualize the distribution of customer ages. Label the axes and title the chart.
2. Using R Generate a pie chart to represent the distribution of customers by gender.
3. Using R Build a table to show the demographic information for each customer. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

CustomerID = c(1, 2, 3),

Age = c(28, 35, 42),

Gender = c("Female", "Male", "Female"),

Income = c(50000, 60000, 75000)

)

# Create bar chart for customer ages

ggplot(data, aes(x = factor(CustomerID), y = Age, fill = factor(CustomerID))) +

geom\_bar(stat = "identity") +

labs(title = "Distribution of Customer Ages",

x = "Customer ID",

y = "Age")

# Create a data frame with the provided data

data <- data.frame(

CustomerID = c(1, 2, 3),

Age = c(28, 35, 42),

Gender = c("Female", "Male", "Female"),

Income = c(50000, 60000, 75000)

)

# Create pie chart for distribution by gender

ggplot(data, aes(x = "", fill = Gender)) +

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

coord\_polar("y") +

labs(title = "Distribution of Customers by Gender",

fill = "Gender")

# Create a data frame with the provided data

data <- data.frame(

CustomerID = c(1, 2, 3),

Age = c(28, 35, 42),

Gender = c("Female", "Male", "Female"),

Income = c(50000, 60000, 75000)

)

# Display the table

print(data)

**8 .** **Employee Performance Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| **Employee ID** | **Department** | **Years of Service** | **Performance Score** |
| 1 | Sales | 5 | 85 |
| 2 | HR | 3 | 92 |
| 3 | Marketing | 7 | 78 |

1. Using R Create a line chart to visualize the performance trend of employees over time. Label the axes and title the chart.
2. Using R Generate a bar chart showing the distribution of employees across different departments. Label the chart elements.
3. Using R Build a table to display the performance data for each employee. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

EmployeeID = c(1, 2, 3),

Department = c("Sales", "HR", "Marketing"),

YearsOfService = c(5, 3, 7),

PerformanceScore = c(85, 92, 78)

)

# Create line chart for performance trend over time

ggplot(data, aes(x = YearsOfService, y = PerformanceScore, group = EmployeeID, color = factor(EmployeeID))) +

geom\_line() +

labs(title = "Performance Trend Over Time",

x = "Years of Service",

y = "Performance Score",

color = "Employee ID")

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

EmployeeID = c(1, 2, 3),

Department = c("Sales", "HR", "Marketing"),

YearsOfService = c(5, 3, 7),

PerformanceScore = c(85, 92, 78)

)

# Create bar chart for distribution across departments

ggplot(data, aes(x = Department, fill = Department)) +

geom\_bar() +

labs(title = "Distribution of Employees Across Departments",

x = "Department",

y = "Number of Employees",

fill = "Department")

# Create a data frame with the provided data

data <- data.frame(

EmployeeID = c(1, 2, 3),

Department = c("Sales", "HR", "Marketing"),

YearsOfService = c(5, 3, 7),

PerformanceScore = c(85, 92, 78)

)

# Display the table

print(data)

9. **Product Inventory Management**

|  |  |  |  |
| --- | --- | --- | --- |
| **Product ID** | **Product Name** | **Quantity Available** | **Price (in $)** |
| 1 | Product A | 250 | 20 |
| 2 | Product B | 175 | 15 |
| 3 | Product C | 300 | 18 |

1. Using R Create a bar chart to visualize the quantity of each product in the inventory. Label the axes and title the chart.
2. Using R Generate a stacked bar chart to show the quantity of each product within different product categories.
3. Using R Build a table to show the inventory data for each product. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

ProductID = c(1, 2, 3),

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

QuantityAvailable = c(250, 175, 300),

Price = c(20, 15, 18)

)

# Create bar chart for quantity of each product in the inventory

ggplot(data, aes(x = ProductName, y = QuantityAvailable, fill = ProductName)) +

geom\_bar(stat = "identity") +

labs(title = "Quantity of Each Product in Inventory",

x = "Product",

y = "Quantity Available")

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

ProductID = c(1, 2, 3),

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

QuantityAvailable = c(250, 175, 300),

Price = c(20, 15, 18)

)

# Create stacked bar chart for quantity within different product categories

ggplot(data, aes(x = "", y = QuantityAvailable, fill = ProductName)) +

geom\_bar(stat = "identity") +

labs(title = "Quantity Within Different Product Categories",

x = "Product Category",

y = "Total Quantity",

fill = "Product Name")

# Create a data frame with the provided data

data <- data.frame(

ProductID = c(1, 2, 3),

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

QuantityAvailable = c(250, 175, 300),

Price = c(20, 15, 18)

)

# Display the table

print(data)

10. **Survey Responses Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| **Survey ID** | **Question 1** | **Question 2** | **Question 3** |
| 1 | A | B | C |
| 2 | B | A | D |
| 3 | C | A | B |

1. Using R Create a grouped bar chart to visualize the distribution of answers for Question 1. Label the chart elements.
2. Using R Generate a stacked bar chart to represent the overall distribution of responses for all three questions.
3. Using R Build a table to show the survey response data for each survey. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

SurveyID = c(1, 2, 3),

Question1 = c("A", "B", "C"),

Question2 = c("B", "A", "A"),

Question3 = c("C", "D", "B")

)

# Reshape data for ggplot

data\_long <- tidyr::gather(data, key = "Question", value = "Answer", -SurveyID)

# Create grouped bar chart for distribution of answers for Question 1

ggplot(data\_long, aes(x = Answer, fill = factor(SurveyID))) +

geom\_bar(position = "dodge") +

labs(title = "Distribution of Answers for Question 1",

x = "Answer",

y = "Count",

fill = "Survey ID")

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

SurveyID = c(1, 2, 3),

Question1 = c("A", "B", "C"),

Question2 = c("B", "A", "A"),

Question3 = c("C", "D", "B")

)

# Reshape data for ggplot

data\_long <- tidyr::gather(data, key = "Question", value = "Answer", -SurveyID)

# Create stacked bar chart for overall distribution of responses for all three questions

ggplot(data\_long, aes(x = Answer, fill = Question)) +

geom\_bar() +

labs(title = "Overall Distribution of Responses for All Three Questions",

x = "Answer",

y = "Count",

fill = "Question")

# Create a data frame with the provided data

data <- data.frame(

SurveyID = c(1, 2, 3),

Question1 = c("A", "B", "C"),

Question2 = c("B", "A", "A"),

Question3 = c("C", "D", "B")

)

# Display the table

print(data)

**11.Product Category Analysis**

|  |  |
| --- | --- |
| **Category** | **Sales (in $)** |
| Electronics | 50000 |
| Clothing | 35000 |
| Appliances | 40000 |

1. Using R Create a pie chart to represent the distribution of sales across product categories. Include labels.
2. Using R Generate a funnel chart to analyze the sales conversion process for each product category. Label the stages and title the chart.
3. Using R Build a table to display the sales data for each product category. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

Category = c("Electronics", "Clothing", "Appliances"),

Sales = c(50000, 35000, 40000)

)

# Create pie chart for distribution of sales across product categories

ggplot(data, aes(x = "", y = Sales, fill = Category)) +

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

coord\_polar("y") +

labs(title = "Distribution of Sales Across Product Categories",

fill = "Category")

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

Category = c("Electronics", "Clothing", "Appliances"),

Sales = c(50000, 35000, 40000)

)

# Order the data for the funnel chart

data <- data[order(data$Sales, decreasing = TRUE), ]

# Create funnel chart for sales conversion process

ggplot(data, aes(y = Category, x = Sales, fill = Category)) +

geom\_bar(stat = "identity") +

labs(title = "Sales Conversion Process",

x = "Sales",

y = "Category",

fill = "Category")

# Create a data frame with the provided data

data <- data.frame(

Category = c("Electronics", "Clothing", "Appliances"),

Sales = c(50000, 35000, 40000)

)

# Display the table

print(data)

**12. Website Traffic**

|  |  |  |
| --- | --- | --- |
| **Date** | **Page Views** | **Click-through Rate** |
| 2023-01-01 | 1500 | 2.3% |
| 2023-01-02 | 1600 | 2.7% |
| 2023-01-03 | 1400 | 2.0% |

1. Using R Create a line chart to visualize the trend in daily page views over time. Label the axes and title the chart.
2. Using R Generate a bar chart showing the top N days with the highest click-through rates. Label the chart elements.
3. Using R Build a table to show the website traffic data for each day. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

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

PageViews = c(1500, 1600, 1400),

ClickThroughRate = c(2.3, 2.7, 2.0)

)

# Create line chart for trend in daily page views over time

ggplot(data, aes(x = Date, y = PageViews, group = 1)) +

geom\_line() +

labs(title = "Trend in Daily Page Views Over Time",

x = "Date",

y = "Page Views")

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

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

PageViews = c(1500, 1600, 1400),

ClickThroughRate = c(2.3, 2.7, 2.0)

)

# Order data by click-through rate in descending order

data <- data[order(data$ClickThroughRate, decreasing = TRUE), ]

# Select the top N days

top\_n\_days <- 2 # Change this to the desired number of top days

top\_data <- head(data, top\_n\_days)

# Create bar chart for top N days with highest click-through rates

ggplot(top\_data, aes(x = Date, y = ClickThroughRate, fill = as.factor(Date))) +

geom\_bar(stat = "identity") +

labs(title = paste("Top", top\_n\_days, "Days with Highest Click-through Rates"),

x = "Date",

y = "Click-through Rate",

fill = "Date")

# Create a data frame with the provided data

data <- data.frame(

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

PageViews = c(1500, 1600, 1400),

ClickThroughRate = c(2.3, 2.7, 2.0)

)

# Display the table

print(data)

**13. Geographic Data**

|  |  |  |  |
| --- | --- | --- | --- |
| **City** | **Population** | **Avg. Temperature** | **Elevation** |
| City A | 500000 | 75 | 1000 |
| City B | 700000 | 68 | 800 |
| City C | 600000 | 80 | 1200 |

1. Using R Create a map chart to visualize the distribution of cities on a geographic map. Label the map elements.
2. Using R Generate a scatter plot to explore the relationship between average temperature and population size. Explain any insights.
3. Using R Build a table to display the geographic data for each city. Label the table elements.

# Load required libraries

install.packages("leaflet")

library(leaflet)

# Create a data frame with the provided data

data <- data.frame(

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

Population = c(500000, 700000, 600000),

AvgTemperature = c(75, 68, 80),

Elevation = c(1000, 800, 1200)

)

# Create a map chart

leaflet(data) %>%

addTiles() %>%

addMarkers(~Elevation, ~Population, label = ~City)

# Load required libraries

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

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

Population = c(500000, 700000, 600000),

AvgTemperature = c(75, 68, 80),

Elevation = c(1000, 800, 1200)

)

# Create scatter plot

ggplot(data, aes(x = AvgTemperature, y = Population, label = City)) +

geom\_point() +

geom\_text(vjust = -0.5) +

labs(title = "Relationship Between Average Temperature and Population Size",

x = "Average Temperature",

y = "Population")

# Display the table

print(data)

**14. Survey Results**

|  |  |  |  |
| --- | --- | --- | --- |
| **Respondent** | **Question 1** | **Question 2** | **Question 3** |
| 1 | A | B | C |
| 2 | B | A | D |
| 3 | C | A | B |

1. Using R Create a stacked bar chart to display the distribution of answers for Question 1. Label the chart elements.
2. Using R Generate a radar chart to visualize the overall pattern of responses across all three questions.
3. Using R Build a table to show the survey response data for each respondent. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

data <- data.frame(

Respondent = c(1, 2, 3),

Question1 = c("A", "B", "C"),

Question2 = c("B", "A", "A"),

Question3 = c("C", "D", "B")

)

# Reshape data for ggplot

data\_long <- tidyr::gather(data, key = "Question", value = "Answer", -Respondent)

# Create stacked bar chart for distribution of answers for Question 1

ggplot(data\_long, aes(x = factor(Respondent), y = ..count.., fill = Answer)) +

geom\_bar(position = "stack") +

labs(title = "Distribution of Answers for Question 1",

x = "Respondent",

y = "Count",

fill = "Answer")

# Load required library

library(fmsb)

# Create a data frame with the provided data

data <- data.frame(

Respondent = c(1, 2, 3),

Question1 = c("A", "B", "C"),

Question2 = c("B", "A", "A"),

Question3 = c("C", "D", "B")

)

# Convert responses to numerical values

numeric\_data <- data

numeric\_data[, -1] <- apply(data[, -1], MARGIN = 2, function(x) as.numeric(factor(x)))

# Create radar chart

radarchart(numeric\_data, pcol = c("blue", "red", "green"))

# Create a data frame with the provided data

data <- data.frame(

Respondent = c(1, 2, 3),

Question1 = c("A", "B", "C"),

Question2 = c("B", "A", "A"),

Question3 = c("C", "D", "B")

)

# Display the table

print(data)

**15. Customer Satisfaction**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Employee ID** | **Name** | | **Department** | **Years of Service** | | **Performance Score** |
| 1 | John Smith | | Sales | 5 | | 85 |
| 2 | Jane Doe | | HR | 3 | | 92 |
| 3 | Robert Brown | | Marketing | 7 | | 78 |
| 4 | Sarah White | | Sales | 4 | | 90 |
| 5 | Michael Lee | | HR | 2 | | 76 |
| ... | | ... | | | ... | |

1. In R, create a histogram to visualize the distribution of customer ages. Label the axes and title the chart.
2. In R, generate a pie chart to represent the distribution of overall customer satisfaction scores. Include labels.
3. In R, perform sentiment analysis on open-ended customer feedback and create a word cloud visualization.

# Load required libraries

library(ggplot2)

# Create a data frame with the provided employee data

employee\_data <- data.frame(

EmployeeID = c(1, 2, 3, 4, 5),

Name = c("John Smith", "Jane Doe", "Robert Brown", "Sarah White", "Michael Lee"),

Department = c("Sales", "HR", "Marketing", "Sales", "HR"),

YearsOfService = c(5, 3, 7, 4, 2),

PerformanceScore = c(85, 92, 78, 90, 76)

)

# Create a histogram to visualize the distribution of years of service

ggplot(employee\_data, aes(x = YearsOfService)) +

geom\_histogram(binwidth = 1, fill = "blue", color = "black", alpha = 0.7) +

labs(title = "Distribution of Years of Service",

x = "Years of Service",

y = "Count")

# Create a pie chart to represent the distribution of overall performance scores

ggplot(employee\_data, aes(x = "", y = PerformanceScore, fill = factor(EmployeeID))) +

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

coord\_polar("y") +

labs(title = "Distribution of Performance Scores",

fill = "Employee ID")

**16. Product Sales Analysis**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Product ID** | **Product Name** | **January Sales** | **February Sales** | **March Sales** |
| 1 | Product A | 2000 | 2200 | 2400 |
| 2 | Product B | 1500 | 1800 | 1600 |
| 3 | Product C | 1200 | 1400 | 1100 |

* 1. Using R Create a grouped bar chart to visualize the sales of each product for the first quarter. Label the chart elements.
  2. Using R Generate a stacked area chart to represent the overall sales trend for all products over the three months.
  3. Using R Build a table to show the monthly sales data for each product. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

product\_data <- data.frame(

ProductID = c(1, 2, 3),

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

JanuarySales = c(2000, 1500, 1200),

FebruarySales = c(2200, 1800, 1400),

MarchSales = c(2400, 1600, 1100)

)

# Reshape data for ggplot

product\_data\_long <- tidyr::gather(product\_data, key = "Month", value = "Sales", -ProductID, -ProductName)

# Create grouped bar chart

ggplot(product\_data\_long, aes(x = ProductName, y = Sales, fill = Month)) +

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

labs(title = "Quarterly Sales by Product",

x = "Product",

y = "Sales",

fill = "Month")

# Load required library

library(ggplot2)

# Create a data frame with the provided data

product\_data <- data.frame(

ProductID = c(1, 2, 3),

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

JanuarySales = c(2000, 1500, 1200),

FebruarySales = c(2200, 1800, 1400),

MarchSales = c(2400, 1600, 1100)

)

# Reshape data for ggplot

product\_data\_long <- tidyr::gather(product\_data, key = "Month", value = "Sales", -ProductID, -ProductName)

# Create stacked area chart

ggplot(product\_data\_long, aes(x = as.factor(ProductID), y = Sales, fill = Month)) +

geom\_area() +

labs(title = "Overall Sales Trend",

x = "Product",

y = "Sales",

fill = "Month")

# Create a data frame with the provided data

product\_data <- data.frame(

ProductID = c(1, 2, 3),

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

JanuarySales = c(2000, 1500, 1200),

FebruarySales = c(2200, 1800, 1400),

MarchSales = c(2400, 1600, 1100)

)

# Display the table

print(product\_data)

**17. Customer Demographics Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| **Customer ID** | **Age** | **Gender** | **Income (in $)** |
| 1 | 28 | Female | 50000 |
| 2 | 35 | Male | 60000 |
| 3 | 42 | Female | 75000 |

1. Using R Create a bar chart to visualize the distribution of customer ages. Label the axes and title the chart.
2. Using R Generate a pie chart to represent the distribution of customers by gender.
3. Using R Build a table to show the demographic information for each customer. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

customer\_data <- data.frame(

CustomerID = c(1, 2, 3),

Age = c(28, 35, 42),

Gender = c("Female", "Male", "Female"),

Income = c(50000, 60000, 75000)

)

# Create bar chart for distribution of customer ages

ggplot(customer\_data, aes(x = as.factor(CustomerID), y = Age, fill = Gender)) +

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

labs(title = "Distribution of Customer Ages",

x = "Customer ID",

y = "Age",

fill = "Gender")

# Create pie chart for distribution of customers by gender

ggplot(customer\_data, aes(x = "", fill = Gender)) +

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

coord\_polar("y") +

labs(title = "Distribution of Customers by Gender",

fill = "Gender")

# Display the table

print(customer\_data)

**18. Employee Performance Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| **Employee ID** | **Department** | **Years of Service** | **Performance Score** |
| 1 | Sales | 5 | 85 |
| 2 | HR | 3 | 92 |
| 3 | Marketing | 7 | 78 |

1. Using R Create a line chart to visualize the performance trend of employees over time. Label the axes and title the chart.
2. Using R Generate a bar chart showing the distribution of employees across different departments. Label the chart elements.
3. Using R Build a table to display the performance data for each employee. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

employee\_data <- data.frame(

EmployeeID = c(1, 2, 3),

Department = c("Sales", "HR", "Marketing"),

YearsOfService = c(5, 3, 7),

PerformanceScore = c(85, 92, 78)

)

# Create line chart for performance trend over time

ggplot(employee\_data, aes(x = EmployeeID, y = PerformanceScore, group = 1)) +

geom\_line() +

labs(title = "Performance Trend Over Time",

x = "Employee ID",

y = "Performance Score")

# Load required library

library(ggplot2)

# Create a data frame with the provided data

employee\_data <- data.frame(

EmployeeID = c(1, 2, 3),

Department = c("Sales", "HR", "Marketing"),

YearsOfService = c(5, 3, 7),

PerformanceScore = c(85, 92, 78)

)

# Create bar chart for distribution of employees across departments

ggplot(employee\_data, aes(x = Department, fill = Department)) +

geom\_bar() +

labs(title = "Distribution of Employees Across Departments",

x = "Department",

y = "Number of Employees",

fill = "Department")

# Display the table

print(employee\_data)

**19. Product Inventory Management**

|  |  |  |  |
| --- | --- | --- | --- |
| **Product ID** | **Product Name** | **Quantity Available** | **Price (in $)** |
| 1 | Product A | 250 | 20 |
| 2 | Product B | 175 | 15 |
| 3 | Product C | 300 | 18 |

1. Using R Create a bar chart to visualize the quantity of each product in the inventory. Label the axes and title the chart.
2. Using R Generate a stacked bar chart to show the quantity of each product within different product categories.
3. Using R Build a table to show the inventory data for each product. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

product\_data <- data.frame(

ProductID = c(1, 2, 3),

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

QuantityAvailable = c(250, 175, 300),

Price = c(20, 15, 18)

)

# Create bar chart for quantity of each product in the inventory

ggplot(product\_data, aes(x = ProductName, y = QuantityAvailable, fill = ProductName)) +

geom\_bar(stat = "identity") +

labs(title = "Quantity of Each Product in Inventory",

x = "Product",

y = "Quantity Available")

# Load required library

library(ggplot2)

# Create a data frame with the provided data

product\_data <- data.frame(

ProductID = c(1, 2, 3),

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

QuantityAvailable = c(250, 175, 300),

Price = c(20, 15, 18)

)

# Create stacked bar chart for quantity within different product categories

ggplot(product\_data, aes(x = "", y = QuantityAvailable, fill = ProductName)) +

geom\_bar(stat = "identity") +

labs(title = "Quantity Within Different Product Categories",

x = "Product Category",

y = "Total Quantity",

fill = "Product Name")

# Display the table

print(product\_data)

**20. Survey Responses Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| **Survey ID** | **Question 1** | **Question 2** | **Question 3** |
| 1 | A | B | C |
| 2 | B | A | D |
| 3 | C | A | B |

1. Using R Create a grouped bar chart to visualize the distribution of answers for Question 1. Label the chart elements.
2. Using R Generate a stacked bar chart to represent the overall distribution of responses for all three questions.
3. Using R Build a table to show the survey response data for each survey. Label the table elements.

# Load required library

library(ggplot2)

# Create a data frame with the provided data

survey\_data <- data.frame(

SurveyID = c(1, 2, 3),

Question1 = c("A", "B", "C"),

Question2 = c("B", "A", "A"),

Question3 = c("C", "D", "B")

)

# Reshape data for ggplot

survey\_data\_long <- tidyr::gather(survey\_data, key = "Question", value = "Answer", -SurveyID)

# Create grouped bar chart for distribution of answers for Question 1

ggplot(survey\_data\_long, aes(x = Answer, fill = factor(SurveyID))) +

geom\_bar(position = "dodge") +

labs(title = "Distribution of Answers for Question 1",

x = "Answer",

y = "Count",

fill = "Survey ID")

# Load required library

library(ggplot2)

# Create a data frame with the provided data

survey\_data <- data.frame(

SurveyID = c(1, 2, 3),

Question1 = c("A", "B", "C"),

Question2 = c("B", "A", "A"),

Question3 = c("C", "D", "B")

)

# Reshape data for ggplot

survey\_data\_long <- tidyr::gather(survey\_data, key = "Question", value = "Answer", -SurveyID)

# Create stacked bar chart for overall distribution of responses for all three questions

ggplot(survey\_data\_long, aes(x = Answer, fill = Question)) +

geom\_bar() +

labs(title = "Overall Distribution of Responses for All Three Questions",

x = "Answer",

y = "Count",

fill = "Question")

# Display the table

print(survey\_data)

**21. Stock Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Stock A** | **Stock B** | **Stock C** |
| 2023-01-01 | 100 | 150 | 120 |
| 2023-01-02 | 105 | 152 | 118 |
| 2023-01-03 | 110 | 148 | 122 |

1. Using R Create a line chart to visualize the stock prices of three companies (Stock A, Stock B, and Stock C) over a specific time period. Label the axes and title the chart.
2. Using R Generate a bar chart showing the daily percentage change in stock prices for Stock A. Label the chart elements.
3. Using R Build a table to display the stock price data for each company over the given period. Label the table elements.

# Install and load necessary packages

library(ggplot2)

# Your stock price data

data <- data.frame(

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

StockA = c(100, 105, 110),

StockB = c(150, 152, 148),

StockC = c(120, 118, 122)

)

# a. Line chart

ggplot(data, aes(x = Date)) +

geom\_line(aes(y = StockA, color = "Stock A"), size = 1.5) +

geom\_line(aes(y = StockB, color = "Stock B"), size = 1.5) +

geom\_line(aes(y = StockC, color = "Stock C"), size = 1.5) +

labs(title = "Stock Prices Over Time",

x = "Date",

y = "Stock Price") +

theme\_minimal()

# b. Bar chart of daily percentage change for Stock A

data$StockA\_PercentChange <- c(0, diff(data$StockA) / lag(data$StockA) \* 100)

ggplot(data, aes(x = Date, y = StockA\_PercentChange)) +

geom\_bar(stat = "identity", fill = "blue", alpha = 0.7) +

labs(title = "Daily Percentage Change in Stock A",

x = "Date",

y = "Percentage Change") +

theme\_minimal()

**22. Sales Data**

|  |  |  |
| --- | --- | --- |
| **Customer ID** | **Age** | **Satisfaction Score** |
| 1 | 28 | 4 |
| 2 | 35 | 5 |
| 3 | 42 | 3 |
| 4 | 30 | 4 |
| 5 | 45 | 5 |

1. a.In R, create a histogram to visualize the distribution of customer ages. Label the axes and title the chart.
2. In R, generate a pie chart to represent the distribution of overall customer satisfaction scores. Include labels.
3. In R, perform sentiment analysis on open-ended customer feedback and create a word cloud visualization.

# Assuming your data is in a data frame called 'customer\_data'

# Install and load necessary packages

library(ggplot2)

# Create a histogram

ggplot(customer\_data, aes(x = Age)) +

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

labs(title = "Distribution of Customer Ages",

x = "Age",

y = "Frequency") +

theme\_minimal()

# Assuming your data is in a data frame called 'customer\_data'

# Install and load necessary packages

install.packages("ggplot2")

library(ggplot2)

# Create a pie chart

satisfaction\_counts <- table(customer\_data$Satisfaction\_Score)

satisfaction\_labels <- c("1", "2", "3", "4", "5") # Assuming satisfaction scores range from 1 to 5

ggplot(data = NULL, aes(x = factor(1), y = satisfaction\_counts, fill = satisfaction\_labels)) +

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

coord\_polar("y") +

labs(title = "Distribution of Customer Satisfaction Scores",

fill = "Satisfaction Score") +

theme\_minimal() +

theme(legend.position = "bottom")

**23. Time Series Analysis**

|  |  |
| --- | --- |
| **Month** | **Sales (in $)** |
| January | 15000 |
| February | 18000 |
| March | 22000 |
| April | 20000 |
| May | 23000 |

1. In R, create a time series line chart to visualize the trend in monthly sales. Label the axes and title the chart.
2. In R, generate a scatter plot to analyse the relationship between advertising budget and monthly sales. Explain any insights.
3. In R, create an autocorrelation plot to identify seasonality in the time series data.

# Assuming your data is in a data frame called 'sales\_data'

# Install and load necessary packages

library(ggplot2)

# Create a time series line chart

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

geom\_line(color = "blue", size = 1.5) +

labs(title = "Monthly Sales Trend",

x = "Month",

y = "Sales (in $)") +

theme\_minimal()

# Assuming your data is in a data frame called 'sales\_data' with columns 'Month', 'Sales', and 'Advertising\_Budget'

# Install and load necessary packages

install.packages("ggplot2")

library(ggplot2)

# Create a scatter plot

ggplot(sales\_data, aes(x = Advertising\_Budget, y = Sales)) +

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

geom\_smooth(method = "lm", se = FALSE, color = "blue") + # Add a linear regression line

labs(title = "Relationship between Advertising Budget and Monthly Sales",

x = "Advertising Budget",

y = "Sales (in $)") +

theme\_minimal()

# Assuming your data is in a data frame called 'sales\_data' with columns 'Month' and 'Sales'

# Install and load necessary packages

install.packages("forecast")

library(forecast)

# Create a time series object

sales\_ts <- ts(sales\_data$Sales, frequency = 12) # Assuming monthly data

# Create an autocorrelation plot

acf(sales\_ts, lag.max = 12, main = "Autocorrelation Plot for Monthly Sales")

**24. Employee Performance Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| **Employee ID** | **Department** | **Years of Service** | **Performance Score** |
| 1 | Sales | 5 | 85 |
| 2 | HR | 3 | 92 |
| 3 | Marketing | 7 | 78 |
| 4 | Sales | 4 | 90 |
| 5 | HR | 2 | 76 |

1. In R, create a bar chart to visualize the distribution of employees across different departments. Label the chart elements.
2. In R, generate a line chart to visualize the performance trend of employees over time. Label the axes and title the chart.
3. In R, develop a table showing the performance data for each employee. Label the table elements.

# Assuming your data is in a data frame called 'employee\_data'

# Install and load necessary packages

library(ggplot2)

# Create a bar chart

ggplot(employee\_data, aes(x = Department)) +

geom\_bar(fill = "skyblue", alpha = 0.7) +

labs(title = "Employee Distribution Across Departments",

x = "Department",

y = "Number of Employees") +

theme\_minimal()

# Assuming your data is in a data frame called 'employee\_data'

# Install and load necessary packages

install.packages("ggplot2")

library(ggplot2)

# Create a line chart

ggplot(employee\_data, aes(x = Years\_of\_Service, y = Performance\_Score, group = 1)) +

geom\_line(color = "green", size = 1.5) +

labs(title = "Performance Trend Over Time",

x = "Years of Service",

y = "Performance Score") +

theme\_minimal()

# Assuming your data is in a data frame called 'employee\_data'

**25. Product Inventory Management**

|  |  |  |
| --- | --- | --- |
| **Product ID** | **Product Name** | **Quantity Available** |
| 1 | Product A | 250 |
| 2 | Product B | 175 |
| 3 | Product C | 300 |
| 4 | Product D | 200 |
| 5 | Product E | 220 |

1. In R, create a bar chart to visualize the quantity of each product in the inventory. Label the axes and title the chart.
2. In R, generate a stacked bar chart to show the quantity of each product within different product categories.
3. In R, create a scatter plot to explore the relationship between product price and quantity available. Explain any insights.

# Assuming your data is in a data frame called 'product\_data'

# Install and load necessary packages

install.packages("ggplot2")

library(ggplot2)

# Create a bar chart

ggplot(product\_data, aes(x = Product\_Name, y = Quantity\_Available, fill = Product\_Name)) +

geom\_bar(stat = "identity", alpha = 0.7) +

labs(title = "Quantity of Each Product in Inventory",

x = "Product Name",

y = "Quantity Available") +

theme\_minimal() +

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

# Assuming your data is in a data frame called 'product\_data' with a column 'Product\_Category'

# Install and load necessary packages

install.packages("ggplot2")

library(ggplot2)

# Create a stacked bar chart

ggplot(product\_data, aes(x = Product\_Category, y = Quantity\_Available, fill = Product\_Name)) +

geom\_bar(stat = "identity", position = "stack", alpha = 0.7) +

labs(title = "Product Quantity by Category",

x = "Product Category",

y = "Quantity Available") +

theme\_minimal() +

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

# Assuming your data is in a data frame called 'product\_data' with columns 'Product\_Name', 'Quantity\_Available', and 'Product\_Price'

# Install and load necessary packages

install.packages("ggplot2")

library(ggplot2)

# Create a scatter plot

ggplot(product\_data, aes(x = Product\_Price, y = Quantity\_Available, label = Product\_Name)) +

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

labs(title = "Relationship between Product Price and Quantity Available",

x = "Product Price",

y = "Quantity Available") +

geom\_text(nudge\_x = 1, nudge\_y = 5, check\_overlap = TRUE) +

theme\_minimal()

**26. Website Traffic Analysis**

|  |  |  |
| --- | --- | --- |
| **Date** | **Page Views** | **Click-through Rate** |
| 2023-01-01 | 1500 | 2.3% |
| 2023-01-02 | 1600 | 2.7% |
| 2023-01-03 | 1400 | 2.0% |
| 2023-01-04 | 1650 | 2.4% |
| 2023-01-05 | 1800 | 2.6% |

1. In Tableau, create a line chart to visualize the trend in page views over time. Label the axes and title the chart.
2. In R, generate a bar chart to show the top N days with the highest click-through rates. Label the chart elements.
3. In R, build a stacked area chart to display the distribution of user interactions (likes, shares, comments) on the website.
4. In Tableau, create a dashboard with an interactive map showing traffic sources and a bar chart displaying page views by source.

# Install and load required libraries

install.packages(c("tidyverse", "ggplot2"))

library(tidyverse)

# Create the data frame

data <- data.frame(

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

Page\_Views = c(1500, 1600, 1400, 1650, 1800),

Click\_Through\_Rate = c(2.3, 2.7, 2.0, 2.4, 2.6)

)

# a. Line chart to visualize trend in page views over time

ggplot(data, aes(x = Date, y = Page\_Views)) +

geom\_line(color = "blue") +

labs(title = "Page Views Over Time", x = "Date", y = "Page Views")

# b. Bar chart for top N days with highest click-through rates

N <- 3 # Change N as needed

top\_days <- data %>% top\_n(N, Click\_Through\_Rate)

ggplot(top\_days, aes(x = Date, y = Click\_Through\_Rate, fill = Date)) +

geom\_bar(stat = "identity") +

labs(title = paste("Top", N, "Days with Highest Click-through Rates"), x = "Date", y = "Click-through Rate")

# c. Stacked area chart for user interactions

# Assuming you have data for user interactions (likes, shares, comments)

# Replace the values in the data frame accordingly

interaction\_data <- data.frame(

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

Likes = c(100, 120, 90, 110, 130),

Shares = c(50, 60, 45, 55, 65),

Comments = c(30, 40, 35, 25, 50)

)

interaction\_data\_long <- interaction\_data %>%

pivot\_longer(cols = c(Likes, Shares, Comments), names\_to = "Interaction", values\_to = "Count")

ggplot(interaction\_data\_long, aes(x = Date, y = Count, fill = Interaction)) +

geom\_area() +

labs(title = "Distribution of User Interactions", x = "Date", y = "Count")

# Tableau visualization code for the remaining parts

# d. Create a dashboard with an interactive map and a bar chart

# You can use Tableau Desktop for this part.

# Connect Tableau to your data source and create the visualizations as described.

# Combine them into a dashboard with interactive features.

**27. Employee Sales Data**

|  |  |
| --- | --- |
| Employee | Sales (in dollars) |
| John | 5000 |
| Alice | 6200 |
| Bob | 4500 |
| Sarah | 7400 |

* 1. Using R Create a line chart to visualize the monthly sales. Label the axes and tit the chart appropriately.
  2. Using R Generate a bar chart to display the top-selling products for the year. Label the chart elements.
  3. Using R Develop a scatter plot to explore the relationship between advertising budget and monthly sales. Explain the insights drawn from the scatter plot.
  4. Using Tableau Build an interactive dashboard combining the line chart and bar chart to allow users to explore sales data interactively.

# Install and load required libraries

install.packages(c("tidyverse", "ggplot2"))

library(tidyverse)

# Create the data frame

sales\_data <- data.frame(

Employee = c("John", "Alice", "Bob", "Sarah"),

Sales = c(5000, 6200, 4500, 7400)

)

# a. Line chart to visualize monthly sales

ggplot(sales\_data, aes(x = Employee, y = Sales, group = 1)) +

geom\_line(color = "blue") +

labs(title = "Monthly Sales", x = "Employee", y = "Sales (in dollars)")

# b. Bar chart for top-selling products

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

geom\_bar(stat = "identity") +

labs(title = "Top-Selling Products", x = "Employee", y = "Sales (in dollars)")

# c. Scatter plot for relationship between advertising budget and monthly sales

# Assuming you have data for advertising budget, replace the values accordingly

advertising\_data <- data.frame(

Employee = c("John", "Alice", "Bob", "Sarah"),

Advertising\_Budget = c(2000, 2500, 1800, 3000)

)

scatter\_data <- merge(sales\_data, advertising\_data, by = "Employee")

ggplot(scatter\_data, aes(x = Advertising\_Budget, y = Sales, label = Employee)) +

geom\_point() +

geom\_text(vjust = -0.5) +

labs(title = "Relationship between Advertising Budget and Monthly Sales", x = "Advertising Budget", y = "Sales (in dollars)")

# Tableau visualization code for part d

# Connect Tableau to your data source and create the visualizations as described.

# Combine them into a dashboard with interactive features.

**28. Weather Data**

|  |  |  |
| --- | --- | --- |
| Date | Temperature (°C) | Precipitation (mm) |
| 2023-01-01 | 12 | 0.5 |
| 2023-01-02 | 9 | 2.0 |
| 2023-01-03 | 15 | 0.0 |
| 2023-01-04 | 8 | 5.0 |

1. Using R Create a map chart to visualize the distribution of temperature on a geographic map. Label the map elements.
2. Using R Generate a scatter plot to explore the relationship between average temperature and precipitation. Explain any insights.
3. Using R Build a table to display the weather data for each date. Label the table elements.
4. Develop a Tableau dashboard combining the map chart, scatter plot, and the table for interactive exploration of weather data.

# Install and load required libraries

install.packages("tidyverse")

library(tidyverse)

# Create the data frames

weather\_data <- data.frame(

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

Temperature = c(12, 9, 15, 8),

Precipitation = c(0.5, 2.0, 0.0, 5.0)

)

# a. Map chart to visualize the distribution of temperature

# Note: For a geographic map, you need latitude and longitude coordinates.

# Replace the values in the data frame accordingly.

map\_data <- data.frame(

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

Latitude = c(XX, YY, ZZ, AA), # Replace with actual latitude values

Longitude = c(YY, ZZ, AA, BB) # Replace with actual longitude values

)

ggplot() +

geom\_point(data = map\_data, aes(x = Longitude, y = Latitude, size = Temperature), color = "blue") +

labs(title = "Distribution of Temperature on Geographic Map", size = "Temperature (°C)")

# b. Scatter plot for relationship between average temperature and precipitation

average\_weather <- summarize(weather\_data, Avg\_Temperature = mean(Temperature), Total\_Precipitation = sum(Precipitation))

ggplot(average\_weather, aes(x = Avg\_Temperature, y = Total\_Precipitation)) +

geom\_point() +

labs(title = "Relationship between Average Temperature and Precipitation", x = "Average Temperature (°C)", y = "Total Precipitation (mm)")

# c. Table to display weather data for each date

# Using the kableExtra package for a formatted table

**29. Product Inventory**

|  |  |  |
| --- | --- | --- |
| Product | Quantity | Price (USD) |
| Widget A | 100 | 10.00 |
| Widget B | 75 | 15.00 |
| Widget C | 120 | 8.50 |
| Widget D | 50 | 20.00 |

1. Using R Create a bar chart to visualize the quantity of each product in the inventory. Label the axes and title the chart.
2. Using R Generate a stacked bar chart to show the quantity of each product within different product categories.
3. Using R Build a table to show the inventory data for each product. Label the table elements.
4. Develop a Tableau dashboard combining the bar chart, stacked bar chart, and the table for interactive exploration of inventory data.

# Install and load required libraries

library(ggplot2)

# Create the data frame

inventory\_data <- data.frame(

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

Quantity = c(100, 75, 120, 50),

Price = c(10.00, 15.00, 8.50, 20.00)

)

# a. Bar chart to visualize the quantity of each product

ggplot(inventory\_data, aes(x = Product, y = Quantity, fill = Product)) +

geom\_bar(stat = "identity") +

labs(title = "Quantity of Each Product in the Inventory", x = "Product", y = "Quantity")

# b. Stacked bar chart to show quantity within different product categories

# Assuming you have product categories, replace the values accordingly

categories\_data <- data.frame(

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

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

)

inventory\_data <- merge(inventory\_data, categories\_data, by = "Product")

ggplot(inventory\_data, aes(x = Category, y = Quantity, fill = Product)) +

geom\_bar(stat = "identity") +

labs(title = "Quantity of Each Product within Different Categories", x = "Category", y = "Quantity")

**30. Student Exam Scores**

|  |  |  |  |
| --- | --- | --- | --- |
| Student | Math | Science | History |
| Student A | 85 | 92 | 78 |
| Student B | 76 | 88 | 89 |
| Student C | 92 | 79 | 87 |
| Student D | 88 | 95 | 82 |

1. Using R Create a bar chart to visualize the marks of each student.
2. Using R Generate a stacked bar chart to show different subject status.
3. Using R Build a scatter plot to explore the relationship between maths subject and history subject
4. Develop a Tableau dashboard with the pie chart to visualize the distribution of students by grade level.

# Assuming you have a data frame named 'students'

students <- data.frame(

Student = c("A", "B", "C", "D"),

Math = c(85, 76, 92, 88),

Science = c(92, 88, 79, 95),

History = c(78, 89, 87, 82)

)

# Load the necessary library for plotting (if not already installed)

# install.packages("ggplot2")

library(ggplot2)

# a. Bar chart for each student

bar\_chart <- ggplot(students, aes(x = Student)) +

geom\_bar(aes(y = Math), stat = "identity", fill = "skyblue", position = "dodge") +

geom\_bar(aes(y = Science), stat = "identity", fill = "lightgreen", position = "dodge") +

geom\_bar(aes(y = History), stat = "identity", fill = "coral", position = "dodge") +

labs(title = "Student Exam Scores",

x = "Student",

y = "Marks") +

scale\_fill\_manual(values = c("skyblue", "lightgreen", "coral"))

# b. Stacked bar chart for different subject status

stacked\_bar\_chart <- ggplot(students, aes(x = Student, y = c(Math, Science, History), fill = factor(variable))) +

geom\_bar(stat = "identity") +

labs(title = "Subject Status",

x = "Student",

y = "Marks",

fill = "Subject")

# c. Scatter plot for the relationship between Math and History

scatter\_plot <- ggplot(students, aes(x = Math, y = History)) +

geom\_point() +

labs(title = "Math vs. History",

x = "Math",

y = "History")

# Show the plots

print(bar\_chart)

print(stacked\_bar\_chart)

print(scatter\_plot)

**31. Customer Feedback**

|  |  |  |
| --- | --- | --- |
| Customer | Rating | Feedback |
| Customer 1 | 4 | "Great service!" |
| Customer 2 | 3 | "Decent but could improve." |
| Customer 3 | 5 | "Excellent experience!" |
| Customer 4 | 2 | "Not satisfied at all." |

1. Using R Create a histogram to represent the distribution of customer ratings. Label the axes and title the chart.
2. Using R Generate a pie chart to display the overall distribution of customer rating. Include labels.
3. Using R Build a stacked bar chart to visualize the distribution of feedback by customer rating
4. In Tableau, develop a word cloud from open-ended customer feedback to identify prevalent customer sentiments.

Install ggplot2 package if you haven't already

# install.packages("ggplot2")

# Load necessary library

library(ggplot2)

# Sample data

customer\_data <- data.frame(

Customer = c("Customer 1", "Customer 2", "Customer 3", "Customer 4"),

Rating = c(4, 3, 5, 2),

Feedback = c("Great service!", "Decent but could improve.", "Excellent experience!", "Not satisfied at all.")

)

# a. Histogram of customer ratings

histogram <- ggplot(customer\_data, aes(x = Rating)) +

geom\_histogram(binwidth = 1, fill = "skyblue", color = "black") +

labs(title = "Distribution of Customer Ratings",

x = "Ratings",

y = "Frequency")

print(histogram)

# b. Pie chart of overall distribution of customer ratings

pie\_chart <- ggplot(customer\_data, aes(x = factor(1), fill = factor(Rating))) +

geom\_bar(width = 1) +

coord\_polar("y", start = 0) +

labs(title = "Overall Distribution of Customer Ratings") +

scale\_fill\_gradient(low = "red", high = "green", guide = "legend", labels = c("1", "2", "3", "4", "5"))

print(pie\_chart)

# c. Stacked bar chart of feedback by customer rating

stacked\_bar <- ggplot(customer\_data, aes(x = factor(Rating), fill = Feedback)) +

geom\_bar(position = "stack") +

labs(title = "Distribution of Feedback by Customer Rating",

x = "Rating",

y = "Count") +

scale\_fill\_discrete(name = "Feedback")

print(stacked\_bar)

**32. Product Inventory**

|  |  |  |  |
| --- | --- | --- | --- |
| Product | Category | Quantity | Price (USD) |
| Laptop A | Electronics | 50 | 800.00 |
| Phone B | Electronics | 75 | 500.00 |
| Chair C | Furniture | 120 | 50.00 |
| Book D | Books | 200 | 10.00 |

1. Using R Create a bar chart to visualize the quantity of each product in the inventory. Label the axes and title the chart.
2. Using R Generate a stacked bar chart to show the quantity of each product within different product categories.
3. Using R Build a scatter plot to explore the relationship between product price and quantity available. Explain the findings.
4. Develop a Tableau dashboard with the bar chart and stacked bar chart to allow users to interact with the data.

# Load necessary library

library(ggplot2)

# Sample data

product\_data <- data.frame(

Product = c("Laptop A", "Phone B", "Chair C", "Book D"),

Category = c("Electronics", "Electronics", "Furniture", "Books"),

Quantity = c(50, 75, 120, 200),

Price = c(800.00, 500.00, 50.00, 10.00)

)

# a. Bar chart of quantity for each product

bar\_chart\_quantity <- ggplot(product\_data, aes(x = Product, y = Quantity, fill = Product)) +

geom\_bar(stat = "identity", width = 0.5) +

labs(title = "Quantity of Each Product in Inventory",

x = "Product",

y = "Quantity") +

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

print(bar\_chart\_quantity)

# b. Stacked bar chart of quantity within different product categories

stacked\_bar\_category <- ggplot(product\_data, aes(x = Category, y = Quantity, fill = Product)) +

geom\_bar(stat = "identity", position = "stack", width = 0.7) +

labs(title = "Quantity of Each Product in Different Categories",

x = "Category",

y = "Quantity") +

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

print(stacked\_bar\_category)

# c. Scatter plot to explore relationship between product price and quantity available

scatter\_plot <- ggplot(product\_data, aes(x = Price, y = Quantity, color = Category, size = Quantity)) +

geom\_point() +

labs(title = "Relationship between Product Price and Quantity Available",

x = "Price (USD)",

y = "Quantity") +

scale\_color\_discrete(name = "Category") +

theme(legend.position = "bottom")

print(scatter\_plot)

**33. Employee Demographics**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Employee | Age | Gender | Department | Salary (USD) |
| John | 32 | Male | HR | 60000 |
| Alice | 28 | Female | Sales | 70000 |
| Bob | 35 | Male | IT | 75000 |
| Sarah | 29 | Female | Finance | 68000 |

1. Using R Create a bar chart to visualize the distribution of customer ages. Label the axes and title the chart.
2. Using R Generate a pie chart to represent the distribution of customers by gender.
3. Using R Build a table to show the demographic information for each customer. Label the table elements.
4. Develop a Tableau dashboard combining the bar chart, pie chart, and the table for interactive exploration of customer demographics.

# Load necessary library

library(ggplot2)

library(dplyr)

# Sample data

employee\_data <- data.frame(

Employee = c("John", "Alice", "Bob", "Sarah"),

Age = c(32, 28, 35, 29),

Gender = c("Male", "Female", "Male", "Female"),

Department = c("HR", "Sales", "IT", "Finance"),

Salary = c(60000, 70000, 75000, 68000)

)

# a. Bar chart of distribution of customer ages

bar\_chart\_ages <- ggplot(employee\_data, aes(x = Employee, y = Age, fill = Employee)) +

geom\_bar(stat = "identity", width = 0.5) +

labs(title = "Distribution of Employee Ages",

x = "Employee",

y = "Age") +

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

print(bar\_chart\_ages)

# b. Pie chart of distribution of customers by gender

pie\_chart\_gender <- ggplot(employee\_data, aes(x = factor(1), fill = Gender)) +

geom\_bar(width = 1) +

coord\_polar("y", start = 0) +

labs(title = "Distribution of Employees by Gender") +

scale\_fill\_manual(values = c("blue", "pink"), labels = c("Male", "Female"))

print(pie\_chart\_gender)

# c. Table showing demographic information for each customer

table\_data <- employee\_data %>%

select(Employee, Age, Gender, Department, Salary)

print(table\_data)

**34. Customer Orders**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Order ID | Customer Name | Product | Quantity | Price |
| 2001 | John | Widget X | 10 | $5 |
| 2002 | Alice | Widget Y | 5 | $8 |
| 2003 | Bob | Widget X | 8 | $5 |
| 2004 | Sarah | Widget Z | 12 | $7 |

1. In R, create a histogram to visualize the distribution of quantity of products. Label the axes and title the chart.
2. In R, generate a pie chart to represent the distribution of overall quantity and price Include labels.
3. In Tableau, build a stacked bar chart to visualize the distribution of quantity by price.
4. In R, perform sentiment analysis on open-ended customer feedback and create a word cloud visualization.

# Load necessary library

library(ggplot2)

# Sample data

order\_data <- data.frame(

Order\_ID = c(2001, 2002, 2003, 2004),

Customer\_Name = c("John", "Alice", "Bob", "Sarah"),

Product = c("Widget X", "Widget Y", "Widget X", "Widget Z"),

Quantity = c(10, 5, 8, 12),

Price = c(5, 8, 5, 7)

)

# Convert Price column to numeric (remove "$" and convert to numeric)

order\_data$Price <- as.numeric(gsub("\\$", "", order\_data$Price))

# a. Histogram to visualize the distribution of quantity of products

histogram\_quantity <- ggplot(order\_data, aes(x = Quantity)) +

geom\_histogram(binwidth = 1, fill = "skyblue", color = "black") +

labs(title = "Distribution of Product Quantity",

x = "Quantity",

y = "Frequency")

print(histogram\_quantity)

# b. Pie chart to represent the distribution of overall quantity and price

total\_quantity <- sum(order\_data$Quantity)

total\_price <- sum(order\_data$Quantity \* order\_data$Price)

pie\_chart <- ggplot(order\_data, aes(x = factor(1), y = Quantity, fill = factor(Price))) +

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

coord\_polar("y", start = 0) +

labs(title = "Distribution of Overall Quantity and Price",

fill = "Price",

x = "",

y = "") +

scale\_fill\_gradient(low = "red", high = "green", labels = c("$5", "$7", "$8")) +

annotate("text", x = 0, y = 0, label = paste("Total Quantity:", total\_quantity), size = 3, vjust = -0.5) +

annotate("text", x = 0, y = 0, label = paste("Total Price:", "$", total\_price), size = 3, vjust = 0.5)

print(pie\_chart)

**35. Student Attendance**

|  |  |  |
| --- | --- | --- |
| Student | Class Date | Attendance |
| Student A | 2023-01-01 | Present |
| Student B | 2023-01-01 | Absent |
| Student C | 2023-01-02 | Present |
| Student D | 2023-01-02 | Present |

1. Using R Create a bar chart to visualize the attendance of each student.
2. Using R Generate a stacked bar chart to show most recent attendance status.
3. Using R Build a scatter plot to explore the relationship between class date and attendance
4. Develop a Tableau dashboard with the pie chart to visualize the distribution of students by grade level.

# Load necessary library

library(ggplot2)

# Sample data

attendance\_data <- data.frame(

Student = c("Student A", "Student B", "Student C", "Student D"),

Class\_Date = c("2023-01-01", "2023-01-01", "2023-01-02", "2023-01-02"),

Attendance = c("Present", "Absent", "Present", "Present")

)

# Bar chart to visualize the attendance of each student

bar\_chart\_attendance <- ggplot(attendance\_data, aes(x = Student, fill = Attendance)) +

geom\_bar(stat = "count", position = "stack") +

labs(title = "Attendance of Each Student",

x = "Student",

y = "Count") +

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

print(bar\_chart\_attendance)

# Convert Class\_Date to Date format

attendance\_data$Class\_Date <- as.Date(attendance\_data$Class\_Date)

# Sort data by Class\_Date in descending order to get the most recent attendance

attendance\_data <- attendance\_data[order(attendance\_data$Class\_Date, decreasing = TRUE), ]

# Stacked bar chart for most recent attendance status

stacked\_bar\_recent\_attendance <- ggplot(attendance\_data[1, ], aes(x = factor(1), fill = Attendance)) +

geom\_bar(width = 1) +

coord\_polar("y", start = 0) +

labs(title = "Most Recent Attendance Status",

x = "",

y = "") +

theme\_void() +

scale\_fill\_manual(values = c("green", "red"), labels = c("Present", "Absent"))

print(stacked\_bar\_recent\_attendance)

# Scatter plot to explore relationship between class date and attendance

scatter\_plot\_attendance <- ggplot(attendance\_data, aes(x = Class\_Date, y = Student, color = Attendance)) +

geom\_point(size = 3) +

labs(title = "Relationship between Class Date and Attendance",

x = "Class Date",

y = "Student") +

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

print(scatter\_plot\_attendance)

**36. Sales by Region**

|  |  |  |
| --- | --- | --- |
| Region | Month | Sales (USD) |
| North | Jan | 5000 |
| North | Feb | 6200 |
| South | Jan | 4800 |
| South | Feb | 5600 |

1. Using R Create a grouped bar chart to visualize the sales of each product for the first quarter. Label the chart elements.
2. Using R Generate a stacked area chart to represent the overall sales trend for all products over the three months.
3. Using R Build a table to show the monthly sales data for each product. Label the table elements.
4. Develop a Tableau dashboard combining the grouped bar chart, stacked area chart, and the table for interactive exploration of sales data.

# Load necessary library

library(ggplot2)

# Sample data

sales\_data <- data.frame(

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

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

Sales = c(5000, 6200, 4800, 5600)

)

# Grouped bar chart for sales of each region per month

grouped\_bar\_chart <- ggplot(sales\_data, aes(x = Month, y = Sales, fill = Region)) +

geom\_bar(stat = "identity", position = "dodge", width = 0.7) +

labs(title = "Sales of Each Region for First Quarter",

x = "Month",

y = "Sales (USD)") +

scale\_fill\_manual(values = c("North" = "skyblue", "South" = "salmon")) +

theme\_minimal()

print(grouped\_bar\_chart)

# Summing up sales per month for the overall trend

overall\_sales <- aggregate(Sales ~ Month, data = sales\_data, sum)

# Stacked area chart for overall sales trend

stacked\_area\_chart <- ggplot(sales\_data, aes(x = Month, y = Sales, fill = Region)) +

geom\_area(position = "stack") +

labs(title = "Overall Sales Trend for All Regions",

x = "Month",

y = "Total Sales (USD)") +

scale\_fill\_manual(values = c("North" = "skyblue", "South" = "salmon")) +

theme\_minimal()

print(stacked\_area\_chart)

# Reshaping the data to wide format for tabulation

sales\_table <- reshape(sales\_data, idvar = "Month", timevar = "Region", direction = "wide")

# Displaying the table

print(sales\_table)

**37. Customer Purchases**

|  |  |  |  |
| --- | --- | --- | --- |
| Customer | Product | Quantity | Price (USD) |
| Customer A | Widget X | 5 | 10.00 |
| Customer B | Widget Y | 3 | 15.00 |
| Customer A | Widget Z | 2 | 8.50 |
| Customer C | Widget X | 4 | 9.00 |

1. In R, create a histogram to visualize the distribution of Quantity. Label the axes and title the chart.
2. In R, generate a pie chart to represent the distribution of overall quantity and price. Include labels.
3. In Tableau, build a stacked bar chart to visualize the distribution overall quantity and price
4. In R, perform sentiment analysis on open-ended customer feedback and create a word cloud visualization.

# Load necessary libraries

library(ggplot2)

library(tm)

library(wordcloud)

# Sample data

customer\_data <- data.frame(

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

Product = c("Widget X", "Widget Y", "Widget Z", "Widget X"),

Quantity = c(5, 3, 2, 4),

Price = c(10.00, 15.00, 8.50, 9.00)

)

# a. Histogram to visualize the distribution of Quantity

histogram\_quantity <- ggplot(customer\_data, aes(x = Quantity)) +

geom\_histogram(binwidth = 1, fill = "skyblue", color = "black") +

labs(title = "Distribution of Quantity",

x = "Quantity",

y = "Frequency")

print(histogram\_quantity)

# b. Pie chart to represent the distribution of overall quantity and price

total\_quantity <- sum(customer\_data$Quantity)

total\_price <- sum(customer\_data$Quantity \* customer\_data$Price)

pie\_chart <- ggplot(data.frame(Values = c(total\_quantity, total\_price)),

aes(x = factor(1), y = Values, fill = factor(Values))) +

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

coord\_polar("y", start = 0) +

labs(title = "Distribution of Overall Quantity and Price",

fill = "Values",

x = "",

y = "") +

scale\_fill\_manual(values = c("skyblue", "salmon"), labels = c("Total Quantity", "Total Price")) +

annotate("text", x = 0, y = 0, label = paste("Total Quantity:", total\_quantity), size = 3, vjust = -0.5) +

annotate("text", x = 0, y = 0, label = paste("Total Price:", "$", total\_price), size = 3, vjust = 0.5)

print(pie\_chart)

# Sample customer feedback data

customer\_feedback <- c(

"Great product and excellent service!",

"Not satisfied with the product quality.",

"Really happy with my purchase.",

"Customer service needs improvement."

)

# Perform sentiment analysis

corpus <- Corpus(VectorSource(customer\_feedback))

corpus <- tm\_map(corpus, content\_transformer(tolower))

corpus <- tm\_map(corpus, removePunctuation)

corpus <- tm\_map(corpus, removeNumbers)

corpus <- tm\_map(corpus, removeWords, stopwords("english"))

# Create Document Term Matrix (DTM)

dtm <- DocumentTermMatrix(corpus)

# Convert to matrix

m <- as.matrix(dtm)

# Calculate word frequencies

word\_freq <- sort(rowSums(m), decreasing = TRUE)

# Create word cloud

wordcloud(words = names(word\_freq), freq = word\_freq, min.freq = 1,

max.words = 50, random.order = FALSE, colors = brewer.pal(8, "Dark2"))

**38. Student Enrollment**

|  |  |  |
| --- | --- | --- |
| Student | Grade Level | Enrollment Date |
| Student A | 9th | 2023-08-15 |
| Student B | 10th | 2023-09-02 |
| Student C | 11th | 2023-07-20 |
| Student D | 9th | 2023-08-30 |

1. Using R Create a bar chart to visualize the average grade level of the students.
2. Using R Generate a stacked bar chart to show most recent enrollment date.
3. Using R Build a scatter plot to explore the relationship between grade level and enrolment date.
4. Develop a Tableau dashboard with the pie chart to visualize the distribution of students by grade level.

# Load necessary libraries

library(ggplot2)

library(dplyr)

# Sample data

student\_data <- data.frame(

Student = c("Student A", "Student B", "Student C", "Student D"),

Grade\_Level = c("9th", "10th", "11th", "9th"),

Enrollment\_Date = as.Date(c("2023-08-15", "2023-09-02", "2023-07-20", "2023-08-30"))

)

# Calculate average grade level

avg\_grade <- student\_data %>%

mutate(Grade\_Level = factor(Grade\_Level, levels = c("9th", "10th", "11th"))) %>%

group\_by(Grade\_Level) %>%

summarise(Avg\_Enrollment = mean(as.numeric(Grade\_Level)))

# Bar chart for average grade level of students

bar\_chart\_grade <- ggplot(avg\_grade, aes(x = Grade\_Level, y = Avg\_Enrollment, fill = Grade\_Level)) +

geom\_bar(stat = "identity", width = 0.5) +

labs(title = "Average Grade Level of Students",

x = "Grade Level",

y = "Average Grade Level") +

theme\_minimal()

print(bar\_chart\_grade)

# Sort data by Enrollment\_Date in descending order to get the most recent enrollment

student\_data <- student\_data[order(student\_data$Enrollment\_Date, decreasing = TRUE), ]

# Stacked bar chart for most recent enrollment date

stacked\_bar\_recent\_enrollment <- ggplot(student\_data[1, ], aes(x = factor(1), fill = Grade\_Level)) +

geom\_bar(width = 1) +

coord\_polar("y", start = 0) +

labs(title = "Most Recent Enrollment Date",

x = "",

y = "") +

theme\_void()

print(stacked\_bar\_recent\_enrollment)

# Scatter plot to explore relationship between grade level and enrollment date

scatter\_plot <- ggplot(student\_data, aes(x = Enrollment\_Date, y = as.numeric(Grade\_Level), color = Grade\_Level)) +

geom\_point(size = 3) +

labs(title = "Relationship between Grade Level and Enrollment Date",

x = "Enrollment Date",

y = "Grade Level") +

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

print(scatter\_plot)

**39. Product Sales by Store**

|  |  |  |  |
| --- | --- | --- | --- |
| Store | Product | Quantity Sold | Revenue (USD) |
| Store A | Widget X | 100 | 500.00 |
| Store B | Widget Y | 75 | 1125.00 |
| Store A | Widget Y | 50 | 750.00 |
| Store C | Widget X | 80 | 400.00 |

1. Using R Create a bar chart to visualize the quantity of each product in the inventory. Label the axes and title the chart.
2. Using R Generate a stacked bar chart to show the quantity of each product within different product categories.
3. Using R Build a scatter plot to explore the relationship between Revenue and quantity available. Explain the findings.
4. Develop a Tableau dashboard with the bar chart and stacked bar chart to allow users to interact with the data.

# Load necessary libraries

library(ggplot2)

# Sample data

store\_data <- data.frame(

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

Product = c("Widget X", "Widget Y", "Widget Y", "Widget X"),

Quantity\_Sold = c(100, 75, 50, 80),

Revenue = c(500.00, 1125.00, 750.00, 400.00)

)

# Bar chart to visualize the quantity of each product in the inventory

bar\_chart\_quantity <- ggplot(store\_data, aes(x = Product, y = Quantity\_Sold, fill = Product)) +

geom\_bar(stat = "identity", width = 0.5) +

labs(title = "Quantity of Each Product in Inventory",

x = "Product",

y = "Quantity") +

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

print(bar\_chart\_quantity)

# Stacked bar chart to show the quantity of each product within different stores

stacked\_bar\_category <- ggplot(store\_data, aes(x = Store, y = Quantity\_Sold, fill = Product)) +

geom\_bar(stat = "identity", position = "stack", width = 0.7) +

labs(title = "Quantity of Each Product in Different Stores",

x = "Store",

y = "Quantity") +

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

print(stacked\_bar\_category)

# Scatter plot to explore relationship between Revenue and quantity available

scatter\_plot <- ggplot(store\_data, aes(x = Quantity\_Sold, y = Revenue)) +

geom\_point(size = 3) +

labs(title = "Relationship between Revenue and Quantity Sold",

x = "Quantity Sold",

y = "Revenue") +

theme\_minimal()

print(scatter\_plot)

**40. Employee Training Hours**

|  |  |  |
| --- | --- | --- |
| Employee | Department | Training Hours |
| John | HR | 10 |
| Alice | Sales | 8 |
| Bob | IT | 12 |
| Sarah | Finance | 6 |

1. Using R Create a histogram to represent the distribution of Training Hours. Label the axes and title the chart.
2. Using R Generate a pie chart to display the overall employee Training hours. Include labels.
3. Using R Build a stacked bar chart to visualize the distribution of training hours by Employee
4. In Tableau, develop a word cloud from open-ended customer feedback to identify prevalent customer sentiments.

# Load necessary libraries

library(ggplot2)

# Sample data

store\_data <- data.frame(

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

Product = c("Widget X", "Widget Y", "Widget Y", "Widget X"),

Quantity\_Sold = c(100, 75, 50, 80),

Revenue = c(500.00, 1125.00, 750.00, 400.00)

)

# Bar chart to visualize the quantity of each product in the inventory

bar\_chart\_quantity <- ggplot(store\_data, aes(x = Product, y = Quantity\_Sold, fill = Product)) +

geom\_bar(stat = "identity", width = 0.5) +

labs(title = "Quantity of Each Product in Inventory",

x = "Product",

y = "Quantity") +

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

print(bar\_chart\_quantity)

# Stacked bar chart to show the quantity of each product within different stores

stacked\_bar\_category <- ggplot(store\_data, aes(x = Store, y = Quantity\_Sold, fill = Product)) +

geom\_bar(stat = "identity", position = "stack", width = 0.7) +

labs(title = "Quantity of Each Product in Different Stores",

x = "Store",

y = "Quantity") +

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

print(stacked\_bar\_category)

# Scatter plot to explore relationship between Revenue and quantity available

scatter\_plot <- ggplot(store\_data, aes(x = Quantity\_Sold, y = Revenue)) +

geom\_point(size = 3) +

labs(title = "Relationship between Revenue and Quantity Sold",

x = "Quantity Sold",

y = "Revenue") +

theme\_minimal()

print(scatter\_plot)

Top of Form