Group Assignment

Reg.No	Name
D/DBA/22/0024	LPHL Lewangama
D/DBA/22/0025	GKP Sewmini

DBA – CM2062 Statistical computing with R

Lecturer: Mrs. ERC Sandamali

Semester 03

July 2023

Department of Computational Mathematics

Faculty of Computing

General Sir John Kotelawala Defence university

Contents

Table of figures

Introduction

Data analysis plays a crucial role in extracting insights and making informed decisions based on collected data. R, a powerful programming language and software environment for statistical computing and graphics, provides a wide range of tools and libraries specifically designed for data analysis. In this report, we will explore the process of data analysis using R and demonstrate its effectiveness in uncovering patterns, trends, and relationships within a given dataset.

The primary objective of this data analysis report is to present a comprehensive understanding of the dataset under investigation. We will begin by providing a clear overview of the data, including its source, format, and any preprocessing steps undertaken. This will ensure transparency and reproducibility in the analysis process.

Next, we will delve into the exploratory data analysis (EDA) phase. EDA involves examining the dataset's structure, identifying missing values or outliers, and visualizing the distribution of variables. By utilizing R's statistical functions and visualization libraries, we can gain valuable insights into the dataset's characteristics, identify potential data issues, and formulate initial hypotheses.

Following the EDA, we will focus on performing more advanced analyses and modeling techniques. R provides an extensive array of packages for regression, classification, clustering, and other statistical techniques. Depending on the objectives of the analysis, we will select appropriate methods to answer specific research questions or address business problems. We will detail the rationale behind the chosen analyses and describe their implementation using R code.

Throughout the report, we will emphasize the importance of data interpretation. Simply running analysis scripts is insufficient; understanding the results and their implications is vital for drawing meaningful conclusions. We will present the findings of each analysis method in a clear and concise manner, incorporating visualizations, summary statistics, and appropriate measures of uncertainty.

Finally, we will conclude the report by summarizing the key findings, highlighting the main insights obtained from the data analysis process. We will also discuss any limitations or caveats to be considered, providing recommendations for further research or areas for improvement.

In summary, this data analysis report using R aims to showcase the power of R as a tool for exploring, analyzing, and interpreting data. By leveraging R's vast ecosystem of packages and libraries, we can unlock valuable insights that can drive decision-making and inform future actions.

About Dataset

Content

Each row represents a customer, each column contains customer's attributes described on the column Metadata.

The data set includes information about:

- Customers who left within the last month – the column is called Churn
- Services that each customer has signed up for – phone, multiple lines, internet, online security, online backup, device protection, tech support, and streaming TV and movies
- Customer account information – how long they've been a customer, contract, payment method, paperless billing, monthly charges, and total charges
- Demographic info about customers – gender, age range, and if they have partners and dependents

Each row represents a customer, each column contains customer's attributes described on the column Metadata.

The raw data contains 7043 rows (customers) and 21 columns (features).

Variable Name	Variable description
Customer ID	Customer ID
Gender	Whether the customer is a male or a female
SeniorCitizen	Whether the customer is a senior citizen or not (1, 0)
Partner	Whether the customer has a partner or not (Yes, No)
Dependents	Whether the customer has dependents or not (Yes, No)
tenure	Number of months the customer has stayed with the company
PhoneService	Whether the customer has a phone service or not (Yes, No)
MultipleLines	Whether the customer has multiple lines or not (Yes, No, No phone service)
InternetService	Customer's internet service provider (DSL, Fiber optic, No)
OnlineSecurity	Whether the customer has online security or not (Yes, No, No internet service)

Examining the dataset

Structure of the dataset

```
> str(Customer_Data)
spc_tbl_ [7,043 × 22] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ customerID      : chr [1:7043] "7590-VHVEG" "5575-GNVDE" "3668-QPYBK" "779
5-CFOCW" ...
 $ gender          : chr [1:7043] "Female" "Male" "Male" "Male" ...
 $ SeniorCitizen   : num [1:7043] 0 0 0 0 0 0 0 0 0 0 ...
```

Variables in the dataset

```
> names(Customer_Data)
 [1] "customerID"       "gender"           "SeniorCitizen"
 [4] "Partner"          "Dependents"       "tenure"
 [7] "PhoneService"     "MultipleLines"    "InternetService"
[10] "OnlineSecurity"   "OnlineBackup"     "DeviceProtection"
[13] "TechSupport"      "StreamingTV"      "StreamingMovies"
[16] "Contract"         "PaperlessBilling" "PaymentMethod"
[19] "MonthlyCharges"   "TotalCharges"     "Churn"
```

selects the first four columns of the Customer_Data data frame and assigns it to a new data frame called ‘df’.

```
> df<-Customer_Data[1:4]
> df
   customerID gender SeniorCitizen Partner
1  7590-VHVEG Female             0     Yes
2  5575-GNVDE   Male             0      No
3  3668-QPYBK   Male             0      No
:      :          :                 :      :
```

dimensions of the df dataframe, which represents the number of rows and columns in the dataframe. In this case, the output is [1] 7043 4, indicating that the df dataframe has 7043 rows and 4 columns.

```
> dim(df)
[1] 7043    4
```

The df1 <- filter(Customer_Data, gender == "Male") command filters the Customer_Data dataframe to cr eate a new dataframe df1 that only contains rows where the gender column is equal to "Male".

The View(df1) command opens a new window or tab in your R environment, displaying the contents of t he df1 dataframe in a spreadsheet-like format. This allows you to visually inspect the filtered data.

The table(Customer_Data$gender) command generates a frequency table of the gender column in the Cus tomer_Data dataframe. It counts the occurrences of each unique value in the column and displays the resu lts.

```
> df1<-filter(Customer_Data,gender=="Male")
```

```
> View(df1)
> table(Customer_Data$gender)
```

Female   Male

 3488   3555

This expression evaluates to TRUE for rows where the tenure column is greater than 5, and FALSE otherwise.

```
> table(Customer_Data$tenure>5)

FALSE  TRUE
 1371  5672
```

Dimension of ‘df1’ data frame

```
> dim(df1)
[1] 7043   21
```

Calculate the total number of missing values in the Customer_Data dataset. In this case, it returns a value of 11, indicating that there are 11 missing values in the dataset.

```
> sum(is.na(Customer_Data))
[1] 11
```

Calculate the number of missing values in each column of the Customer_Data dataset.

```
> colSums(is.na(Customer_Data))
      customerID           gender    SeniorCitizen          Partner
               0                0                0                0
      Dependents           tenure     PhoneService    MultipleLines
               0                0                0                0
 InternetService   OnlineSecurity     OnlineBackup DeviceProtection
               0                0                0                0
     TechSupport      StreamingTV  StreamingMovies         Contract
               0                0                0                0
PaperlessBilling    PaymentMethod   MonthlyCharges     TotalCharges
               0                0                0               11
           Churn
               0
```

All columns have 0 missing values except for the TotalCharges column, which has 11 missing values.

Filter the `Customer_Data` dataset to create a new dataframe `na_df` that contains only the rows where the `TotalCharges` column has missing values (`NA`).

```
> na_df<-filter(Customer_Data,is.na(TotalCharges))
```

The `table(Customer_Data$Churn)` command generates a frequency table for the `Churn` variable in the `Customer_Data` dataset. It shows the count of each unique value in the `Churn` column. In this case, the table indicates that there are 5,174 customers labeled as "No" and 1,869 customers labeled as "Yes" in terms of churn.

```
> table(Customer_Data$Churn)

  No  Yes 
5174 1869 
```

The command `Customer_Data$new_data <- ifelse(Customer_Data$Churn=="Yes", 1, 0)` creates a new variable called `new_data` in the `Customer_Data` dataset. It assigns a value of 1 to `new_data` if the corresponding `Churn` value is "Yes," and a value of 0 otherwise.

The subsequent command `table(Customer_Data$new_data)` generates a frequency table for the `new_data` variable. It shows the count of each unique value in the `new_data` column. In this case, the table indicates that there are 5,174 customers labeled as 0 (indicating churn is "No") and 1,869 customers labeled as 1 (indicating churn is "Yes").

```
> Customer_Data$new_data<-ifelse(Customer_Data$Churn=="Yes", 1, 0)
> table(Customer_Data$new_data)

   0    1 
5174 1869 
```

Descriptive statistics for the dataset

```
> dim(Customer_Data)
[1] 7043   22
> mean(Customer_Data$tenure)
[1] 32.37115
> sum(is.na(Customer_Data$TotalCharges))
[1] 11
> mean(Customer_Data$TotalCharges)
[1] NA
> mean(Customer_Data$TotalCharges,na.rm = T)
[1] 2283.3
> sum(is.na(Customer_Data$TotalCharges))
[1] 11
```

- The `Customer_Data` dataset has dimensions of 7043 rows and 22 columns.
- The mean tenure of customers in the `Customer_Data` dataset is approximately 32.37.
- There are 11 missing values in the `TotalCharges` column of the `Customer_Data` dataset.

- When calculating the mean of the `TotalCharges` column without removing the missing values, the result is `NA` (not available) due to the presence of missing values.

- However, if you use the `na.rm = TRUE` argument in the `mean()` function, it will calculate the mean while ignoring the missing values. In this case, the mean of the `TotalCharges` column is approximately 2283.3.

- After calculating the mean, there are still 11 missing values in the `TotalCharges` column.

Visualization of Data

Data types of objects in the Customer_Data dataset.

```
> class(Customer_Data$Churn)
[1] "character"
> table(Customer_Data$Churn)

  No  Yes
5174 1869
> class(Customer_Data$Dependents)
[1] "character"
> class(Customer_Data$tenure)
[1] "numeric"
```

Barplot

```
> table(Customer_Data$Dependents)

  No  Yes
4933 2110
> barplot(table(Customer_Data$Dependents),col = "steelblue",xlab =
+          "Dependents", ylab = "Frequency", main =
+            "Dependants Distribution",ylim = c(0,5000))
```

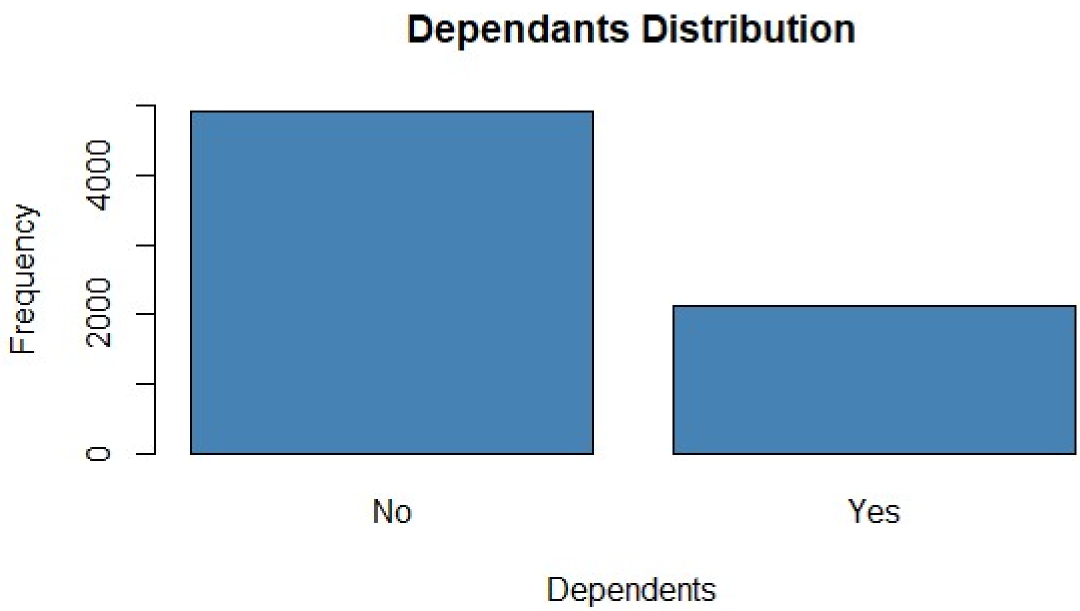

Figure 1: Dependents Distribution

This code will generate a bar plot with the x-axis representing the different levels of the Dependents variable, and the y-axis representing the frequency of each level. The ylim argument sets the range of the y-axis to ensure that all bars are visible within the plot.

```
> table(Customer_Data$PhoneService)

  No  Yes 
 682 6361 
> barplot(table(Customer_Data$PhoneService),col = "aquamarine",
+         xlab = "Phone Service",ylab = "Frequency"
+         ,main = "Distribution of Phone Service",ylim = c(0,7000))
> table(Customer_Data$Contract)
```

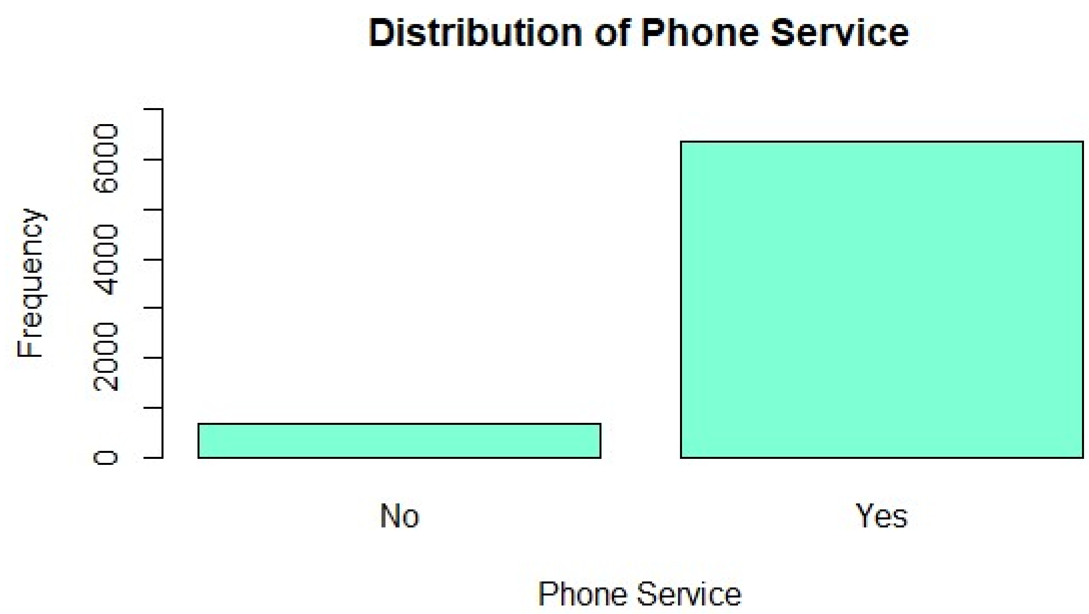

Figure 2: Distribution of Phone Service

This will generate a bar plot with the x-axis representing the two levels of the PhoneService variable, namely "No" and "Yes", and the y-axis representing the frequency of each level.

```
> table(Customer_Data$Contract)

Month-to-month       One year       Two year 
          3875           1473           1695 
> barplot(table(Customer_Data$Contract),col = "green3"
+         ,xlab = "Contract",ylab = "Frequency"
+         ,main = "Distribution of contract",ylim = c(0,4000))
```

x-axis representing the three levels of the Contract variable, namely "Month-to-month", "One year", and "Two year", and the y-axis representing the frequency of each level.

Figure 3: Distribution of Contract type

Stacked Bar Chart

```
> table(Customer_Data$gender)

Female   Male 
  3488   3555 
> df3<-Customer_Data[,c(2,18)]
> df3[df3$gender=="Male" & df3$PaymentMethod=="Bank transfer (automatic)",]
# A tibble: 756 × 2
   gender PaymentMethod            
   <chr>  <chr>                    
 1 Male   Bank transfer (automatic)
 2 Male   Bank transfer (automatic)
 3 Male   Bank transfer (automatic)
 4 Male   Bank transfer (automatic)
 5 Male   Bank transfer (automatic)
 6 Male   Bank transfer (automatic)
 7 Male   Bank transfer (automatic)
 8 Male   Bank transfer (automatic)
 9 Male   Bank transfer (automatic)
10 Male   Bank transfer (automatic)
# ℹ 746 more rows
# ℹ Use `print(n = ...)` to see more rows
> PM<-c("Bank transfer(Automatic)","Credit card(Automatic)",
+       "Electronic check","Mailed check")
> m1<-nrow(df3[df3$gender=="Male" & df3$PaymentMethod=="Bank transfer (automa
tic)",])
> m2<-nrow(df3[df3$gender=="Male" & df3$PaymentMethod=="Credit card (automati
c)",])
> m3<-nrow(df3[df3$gender=="Male" & df3$PaymentMethod=="Electronic check",])
> m4<-nrow(df3[df3$gender=="Male" & df3$PaymentMethod=="Mailed check",])
> f1<-nrow(df3[df3$gender=="Female" & df3$PaymentMethod=="Bank transfer (auto
matic)",])
```

```
> f2<-nrow(df3[df3$gender=="Female" & df3$PaymentMethod=="Credit card (automa
tic)",])
> f3<-nrow(df3[df3$gender=="Female" & df3$PaymentMethod=="Electronic check",]
)
> f4<-nrow(df3[df3$gender=="Female" & df3$PaymentMethod=="Mailed check",])
> Gender<-c("Male","Female")
> values<-matrix(c(m1,m2,m3,m4,f1,f2,f3,f4),nrow = 2,ncol = 4,byrow = TRUE)
> values
     [,1] [,2] [,3] [,4]
[1,]  756  770 1195  834
[2,]  788  752 1170  778
> colors=c("aquamarine","steelblue1")
> barplot(values,main = "Payment Methods chart",names.arg = PM,xlab =
+           "Payment Method",ylab = "Customers",col = colors)
> legend("topright",Gender,cex = 0.8,fill = colors)
```

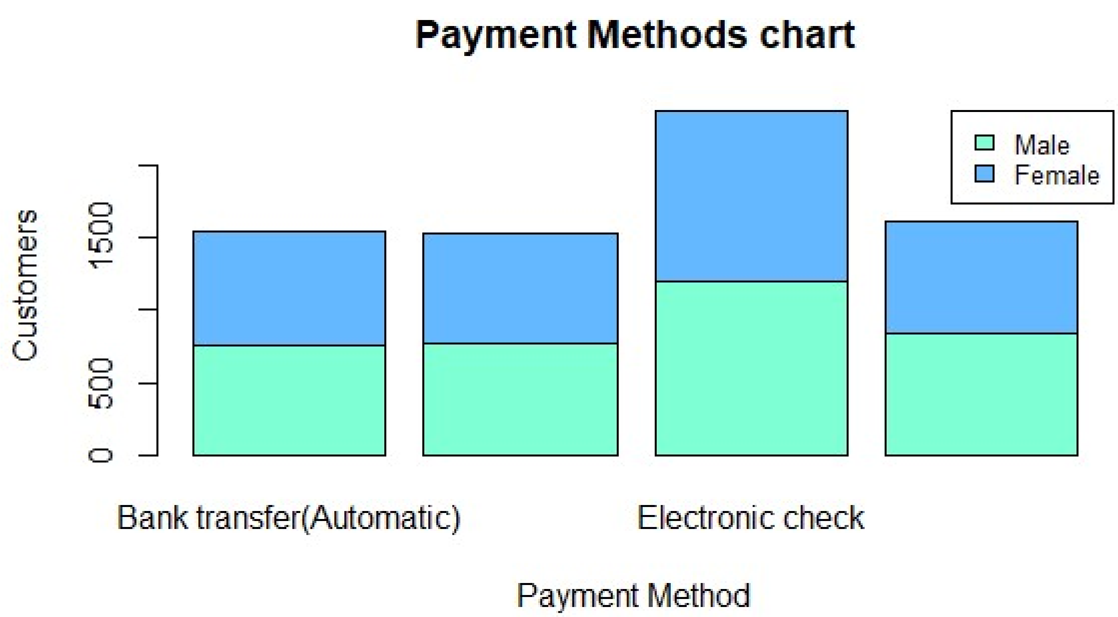

Figure 4: Payment methods chart1

```
> barplot(values,main = "Payment Methods chart",names.arg = PM,xlab =
+           "Payment Method",ylab = "Customers",col = colors,
+         beside = TRUE,ylim = c(0,1500))
> legend("topright",Gender,cex = 0.8,fill = colors)
```

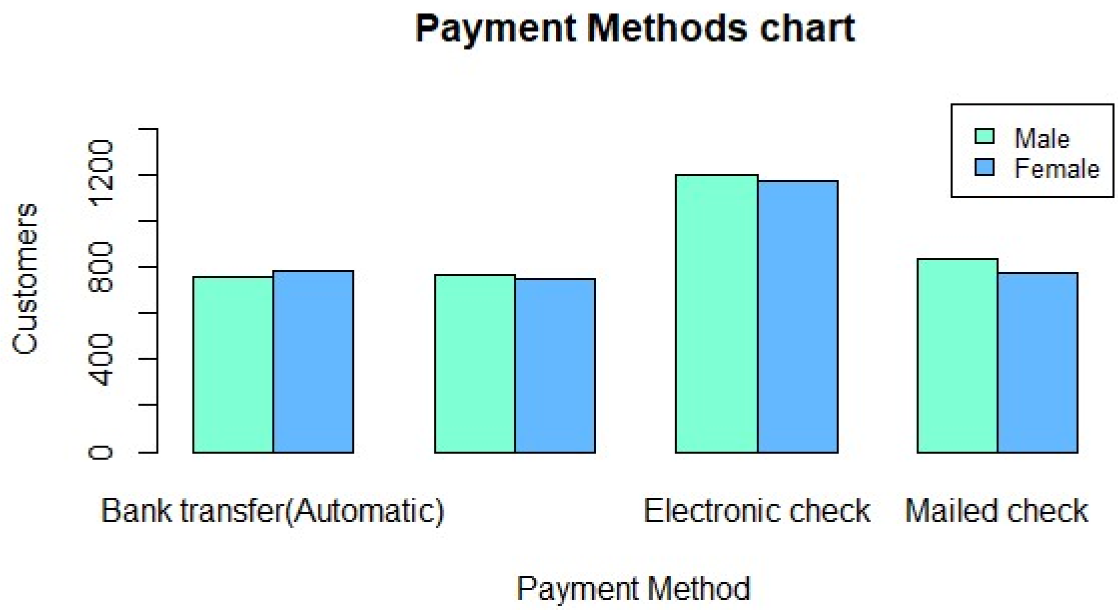

Figure 5: Payment methods chart2

Histogram

```
> hist(Customer_Data$tenure,col = "yellow",border = "black",
+      xlab = "Tenure",main = "Distribution of tenure",
+      breaks = 10,xlim = c(0,80))
```

x-axis representing the values of the tenure variable and the y-axis representing the frequency or count of those values. The breaks argument controls the number of bins or intervals in the histogram, and the xlim argument sets the range of the x-axis to ensure that all values are visible within the plot.

Maximum number and summary statistics

```
> max(Customer_Data$tenure)
[1] 72
> summary(Customer_Data$tenure)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   0.00    9.00   29.00   32.37   55.00   72.00 
```

gplot

```
> ggplot(data = Customer_Data,aes(x=tenure))+
```

```
+   geom_histogram(bins = 20,fill = "blue4",col = "white")+
+   xlab("Tenure")+ggtitle("Distribution of Tenure of Customers")
```

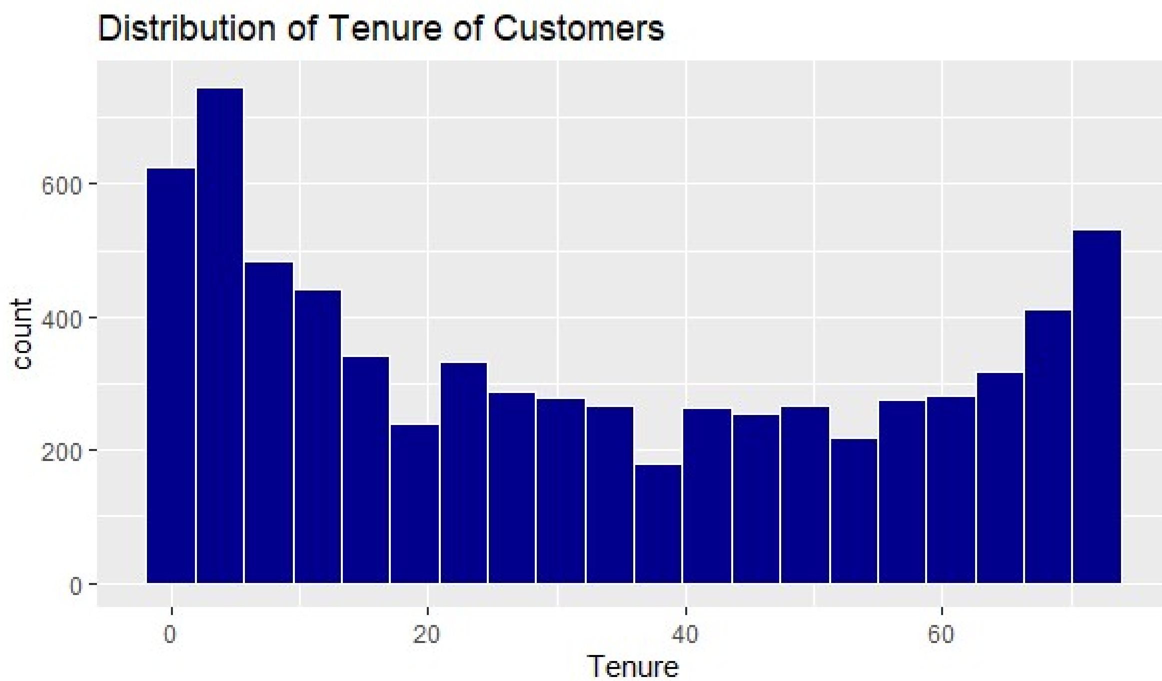

Figure 6: Distribution of tenure of customers

```
> ggplot(data = Customer_Data,aes(x=tenure))+
+   geom_histogram(bins = 20,fill = "blue4",col = "white",alpha=0.5)+
+   stat_bin(bins = 20,geom = "text",color="black",aes(label=..count..),
+   vjust = -0.5)+labs(title = "Tenure Distribution",x="Tenure",y="Frequency"
)+
+   theme(plot.title = element_text(hjust = 0.5,face = "bold"))
```

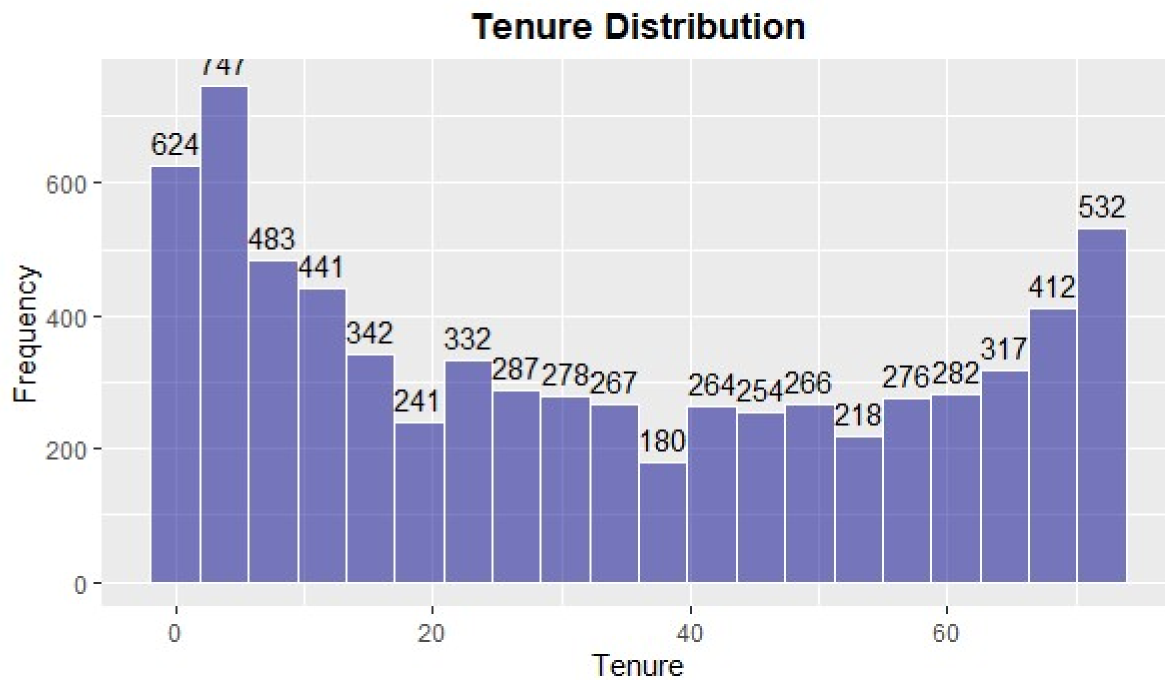

Figure 7: Distribution of tenure of customers

```
> ggplot(data = Customer_Data,aes(y=tenure,x=Partner))+
+   geom_boxplot(fill="violet",col = "black")
```

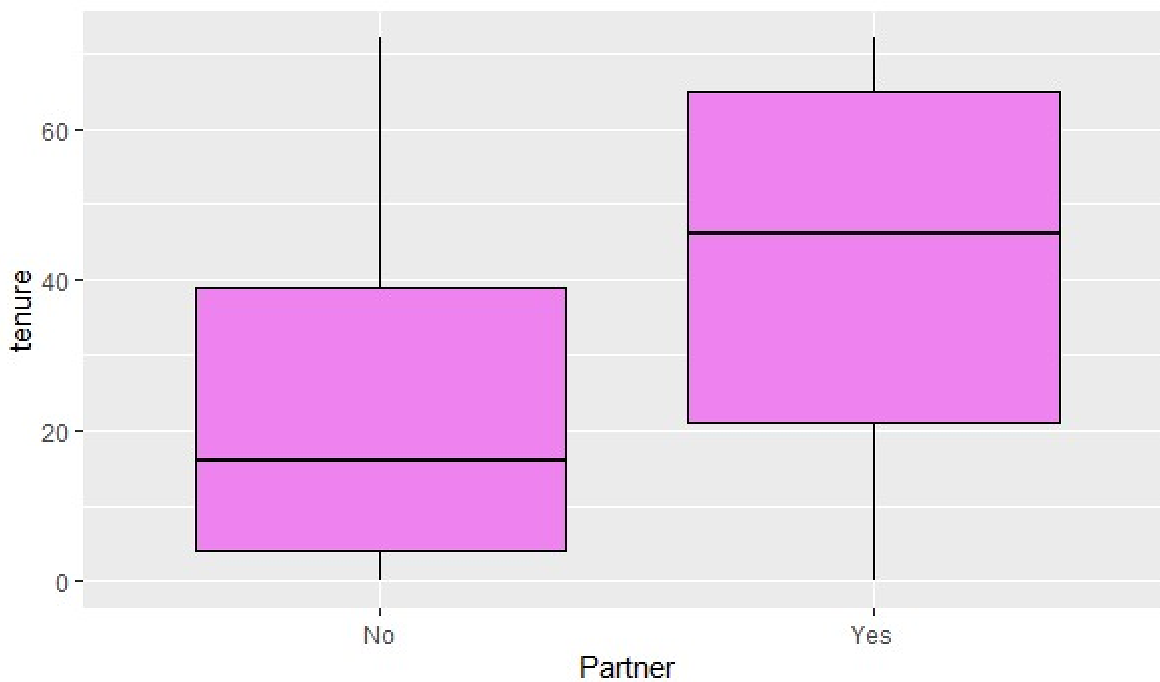

Figure 8: Boxplot

Pie Chart

```
> table(Customer_Data$PaymentMethod)

Bank transfer (automatic)   Credit card (automatic)          Electronic check 
                     1544                      1522                      2365 
             Mailed check 
                     1612 
> x<-c(1544,1522,2365,1612)
> labels1<-c("Bank transfer(Automatic)","Credit card(Automatic)",
+            "Electronic check","Mailed check")
> pie(x,labels1,main = "Payment Methods",col = c("darkblue","darkviolet",
+                                                "cyan","cornsilk"))
> 
> piepercent<-round(100*x/sum(x),3)
> lbls<-paste(piepercent,"%")
> 
> pie(x,labels=lbls,main = "Payment Methods",col = c("darkblue","darkviolet",
+                                                    "cyan","cornsilk"))
> legend("topleft",c("Bank transfer(Automatic)","Credit card(Automatic)",
+                    "Electronic check","Mailed check"),cex = 0.8,
+        fill = c("darkblue","darkviolet",
+                 "cyan","cornsilk"))
```

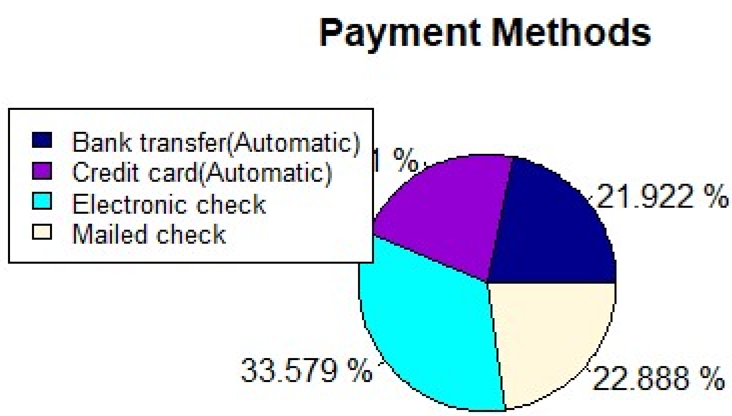

Figure 9: Payment methods pie chart

```
> table(Customer_Data$Contract)

Month-to-month       One year       Two year 
          3875           1473           1695 
> y<-c(3875,1473,1695)
> labels2<-c("Month to month","One year","Two year")
> pie(y,labels,main = "Contract Type",col = grey(seq(0.4,1.0,length = 3)),
+     clockwise = TRUE)
> piepercent<-round(100*x/sum(x),3)
```

```
> lbls<-paste(piepercent,"%")
> pie(y,labels = lbls,main = "Contract Type",col = grey(seq(0.4,1.0,length =
3)),
+     clockwise = TRUE)
> legend("topright",c("Month to month","One year","Two year"),cex = 0.8,
+        fill = grey(seq(0.4,1.0,length = 3)))
```

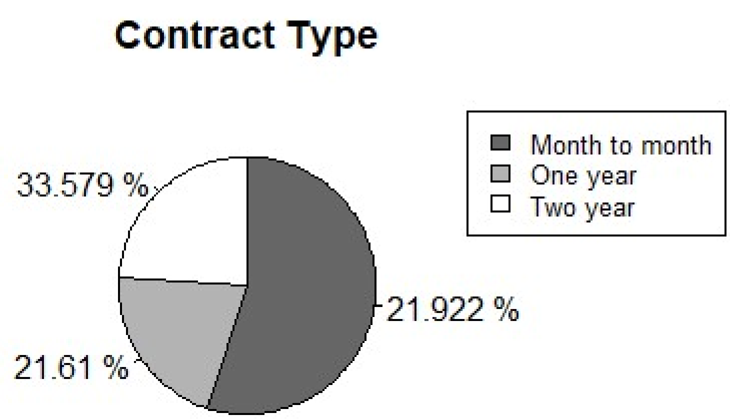

Figure 10: Contract type pie chart

Conclusion

In conclusion, the analysis of the dataset has provided valuable insights into the Customer_Data. Through a thorough exploration of the data and the application of various analytical techniques, several key findings have emerged. The dataset contains customers' data with 7043 variables, including gender, payment type, ID etc.

However, it is important to acknowledge the limitations of the dataset, such as limited sample size, missing data, etc. These limitations should be taken into consideration when interpreting the results.

Despite these limitations, the analysis has provided valuable information that can guide decision-making and inform future actions. Moving forward, further research and improvements in data collection and analysis methods would contribute to a deeper understanding of customer behavior and preferences in data analysis.

Overall, this analysis serves as a foundation for further exploration and underscores the importance of data-driven insights in understanding customers and making informed business decisions in the customer churn.

References:

https://www.kaggle.com/datasets/blastchar/telco-customer-churn