Mini Project in R Programming (Student satisfaction survey)

Students satisfaction Survey

The dataset is regarding survey conducted at an autonomous college on college faculties performance and treatment towards students at the college. Questionnaire of 20 being rolled out to each department available at the college and randomly given to sampled list of students for the feedback.

1. Loading the Data

```
df=read_csv("./Datasets/Student_Satisfaction_Survey.csv")
```

```
## Rows: 580 Columns: 12
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (4): Questions, Average/ Percentage, Course Name, Basic Course
## dbl (8): SN, Total Feedback Given, Total Configured, Weightage 1, Weightage ...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
View(df)
```

The dataset contains information related to a survey conducted at an autonomous college to evaluate faculty performance. Here’s a breakdown of the features in the dataset:

SN : Likely a serial number or identifier for each entry in the survey.

Total Feedback Given: The number of feedback entries provided by students for a specific question.

Total Configured: Possibly refers to the total number of students or responses expected for each question.

Questions: The actual question asked in the survey, related to the performance and effectiveness of faculty members.

Weightage 1 Lowest grade on scale

Weightage 2 better than lowest but below average rating

Weightage 3 Average

Weightage 4 Above Average

Weightage 5 Best rating

Average/ Percentage: This could be the average score or percentage calculated from the responses, providing a quantitative measure of the feedback for each question.

Course Name: The specific course to which the feedback applies.

Basic Course: A more general category or type of course under which the specific course falls.

2. Understand Data Structure

Displays the structure of the dataset.

```
str(df)
```

```
## spc_tbl_ [580 × 12] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ SN                  : num [1:580] 1 2 3 4 5 6 7 8 9 10 ...
##  $ Total Feedback Given: num [1:580] 1 1 1 1 1 1 1 1 1 1 ...
##  $ Total Configured    : num [1:580] 12 12 12 12 12 12 12 12 12 12 ...
##  $ Questions           : chr [1:580] "How much of the syllabus was covered in the class?" "How well did the teachers prep
are for the classes?" "How well were the teachers able to communicate?" "The teacher\x92s approach to teaching can best be d
escribed as" ...
##  $ Weightage 1         : num [1:580] 0 0 0 0 0 0 0 0 0 0 ...
##  $ Weightage 2         : num [1:580] 0 0 0 0 0 0 0 1 0 0 ...
##  $ Weightage 3         : num [1:580] 1 0 0 1 0 0 1 0 0 1 ...
##  $ Weightage 4         : num [1:580] 0 0 0 0 1 1 0 0 1 0 ...
##  $ Weightage 5         : num [1:580] 0 1 1 0 0 0 0 0 0 0 ...
##  $ Average/ Percentage : chr [1:580] "3.00 / 60.00" "5.00 / 100.00" "5.00 / 100.00" "3.00 / 60.00" ...
##  $ Course Name         : chr [1:580] "FY B.VOC FOOD TECHNOLOGY" "FY B.VOC FOOD TECHNOLOGY" "FY B.VOC FOOD TECHNOLOGY" "FY
B.VOC FOOD TECHNOLOGY" ...
##  $ Basic Course        : chr [1:580] "B.VOC FOOD TECHNOLOGY" "B.VOC FOOD TECHNOLOGY" "B.VOC FOOD TECHNOLOGY" "B.VOC FOOD
TECHNOLOGY" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   SN = col_double(),
##   ..   `Total Feedback Given` = col_double(),
##   ..   `Total Configured` = col_double(),
##   ..   Questions = col_character(),
##   ..   `Weightage 1` = col_double(),
##   ..   `Weightage 2` = col_double(),
##   ..   `Weightage 3` = col_double(),
##   ..   `Weightage 4` = col_double(),
##   ..   `Weightage 5` = col_double(),
##   ..   `Average/ Percentage` = col_character(),
##   ..   `Course Name` = col_character(),
##   ..   `Basic Course` = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Return the dimention of our dataset.

```
dim(df)
```

```
## [1] 580  12
```

Shows the first few rows of the dataset.

```
head(df)
```

```
## # A tibble: 6 × 12
##      SN `Total Feedback Given` `Total Configured` Questions        `Weightage 1`
##   <dbl>                  <dbl>              <dbl> <chr>                      <dbl>
## 1     1                      1                 12 "How much of th…               0
## 2     2                      1                 12 "How well did t…               0
## 3     3                      1                 12 "How well were …               0
## 4     4                      1                 12 "The teacher\x9…               0
## 5     5                      1                 12 "Fairness of th…               0
## 6     6                      1                 12 "Was your perfo…               0
## # ℹ 7 more variables: `Weightage 2` <dbl>, `Weightage 3` <dbl>,
## #   `Weightage 4` <dbl>, `Weightage 5` <dbl>, `Average/ Percentage` <chr>,
## #   `Course Name` <chr>, `Basic Course` <chr>
```

Shows the last few rows of the dataset.

```
tail(df)
```

```
## # A tibble: 6 × 12
##      SN `Total Feedback Given` `Total Configured` Questions        `Weightage 1`
##   <dbl>                  <dbl>              <dbl> <chr>                      <dbl>
## 1    15                      9                170 The institution…               0
## 2    16                      9                170 The institute/ …               1
## 3    17                      9                170 Teachers encour…               0
## 4    18                      9                170 Efforts are mad…               0
## 5    19                      9                170 What percentage…               0
## 6    20                      9                170 The overall qua…               0
## # ℹ 7 more variables: `Weightage 2` <dbl>, `Weightage 3` <dbl>,
## #   `Weightage 4` <dbl>, `Weightage 5` <dbl>, `Average/ Percentage` <chr>,
## #   `Course Name` <chr>, `Basic Course` <chr>
```

Display row and column number

```
nrow(df)
```

```
## [1] 580
```

```
ncol(df)
```

```
## [1] 12
```

provide All the variables name

```
names(df)
```

```
##  [1] "SN"                  "Total Feedback Given" "Total Configured"
##  [4] "Questions"           "Weightage 1"          "Weightage 2"
##  [7] "Weightage 3"         "Weightage 4"          "Weightage 5"
## [10] "Average/ Percentage"  "Course Name"          "Basic Course"
```

Display the number of columns in our dataset.

```
length(df)
```

```
## [1] 12
```

3. Data Cleaning

```
df <- df %>%
  rename(avg_percentage = `Average/ Percentage`)
df$avg_percentage <- as.factor(df$avg_percentage)
str(df)
```

```
## spc_tbl_ [580 × 12] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ SN                 : num [1:580] 1 2 3 4 5 6 7 8 9 10 ...
##  $ Total Feedback Given: num [1:580] 1 1 1 1 1 1 1 1 1 1 ...
##  $ Total Configured    : num [1:580] 12 12 12 12 12 12 12 12 12 12 ...
##  $ Questions           : chr [1:580] "How much of the syllabus was covered in the class?" "How well did the teachers prep
are for the classes?" "How well were the teachers able to communicate?" "The teacher\x92s approach to teaching can best be d
escribed as" ...
##  $ Weightage 1         : num [1:580] 0 0 0 0 0 0 0 0 0 0 ...
##  $ Weightage 2         : num [1:580] 0 0 0 0 0 0 0 1 0 0 ...
##  $ Weightage 3         : num [1:580] 1 0 0 1 0 0 1 0 0 1 ...
##  $ Weightage 4         : num [1:580] 0 0 0 0 1 1 0 0 1 0 ...
##  $ Weightage 5         : num [1:580] 0 1 1 0 0 0 0 0 0 0 ...
##  $ avg_percentage      : Factor w/ 152 levels "1.33 / 26.67",..: 15 152 152 15 83 83 15 3 83 15 ...
##  $ Course Name         : chr [1:580] "FY B.VOC FOOD TECHNOLOGY" "FY B.VOC FOOD TECHNOLOGY" "FY B.VOC FOOD TECHNOLOGY" "FY
B.VOC FOOD TECHNOLOGY" ...
##  $ Basic Course        : chr [1:580] "B.VOC FOOD TECHNOLOGY" "B.VOC FOOD TECHNOLOGY" "B.VOC FOOD TECHNOLOGY" "B.VOC FOOD
TECHNOLOGY" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   SN = col_double(),
##   ..   `Total Feedback Given` = col_double(),
##   ..   `Total Configured` = col_double(),
##   ..   Questions = col_character(),
##   ..   `Weightage 1` = col_double(),
##   ..   `Weightage 2` = col_double(),
##   ..   `Weightage 3` = col_double(),
##   ..   `Weightage 4` = col_double(),
##   ..   `Weightage 5` = col_double(),
##   ..   `Average/ Percentage` = col_character(),
##   ..   `Course Name` = col_character(),
##   ..   `Basic Course` = col_character()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

check missing values in dataset

Identify missing values.

```
sum(is.na(df))
```

```
## [1] 0
```

Handle Missing Values.

```
df <- na.omit(df)
```

4. compute the Summary Statistics

Provides a summary of the dataset.

```
 summary(df)
```

```
##        SN        Total Feedback Given Total Configured  Questions
##  Min.   : 1.00   Min.   : 1.00        Min.   : 12.00   Length:580
##  1st Qu.: 5.75   1st Qu.: 3.00        1st Qu.: 20.00   Class :character
##  Median :10.50   Median : 7.00        Median : 42.00   Mode  :character
##  Mean   :10.50   Mean   :14.31        Mean   : 92.52
##  3rd Qu.:15.25   3rd Qu.:17.00        3rd Qu.:123.00
##  Max.   :20.00   Max.   :74.00        Max.   :559.00
##
##   Weightage 1       Weightage 2     Weightage 3      Weightage 4
##  Min.   : 0.0000   Min.   :0.000   Min.   : 0.000   Min.   : 0.000
##  1st Qu.: 0.0000   1st Qu.:0.000   1st Qu.: 0.000   1st Qu.: 1.000
##  Median : 0.0000   Median :0.000   Median : 1.000   Median : 3.000
##  Mean   : 0.5276   Mean   :1.188   Mean   : 2.538   Mean   : 5.083
##  3rd Qu.: 1.0000   3rd Qu.:2.000   3rd Qu.: 3.250   3rd Qu.: 6.000
##  Max.   :19.0000   Max.   :8.000   Max.   :26.000   Max.   :52.000
##
##   Weightage 5           avg_percentage Course Name        Basic Course
##  Min.   : 0.000   4.00 / 80.00 : 59    Length:580         Length:580
##  1st Qu.: 1.000   3.00 / 60.00 : 38    Class :character   Class :character
##  Median : 3.000   5.00 / 100.00: 34    Mode  :character   Mode  :character
##  Mean   : 4.974   4.50 / 90.00 : 22
##  3rd Qu.: 6.000   3.50 / 70.00 : 21
##  Max.   :38.000   3.67 / 73.33 : 21
##                   (Other)      :385
```

```
names(df)[names(df) == "Weightage 1"] <- "Weightage_1"

mean(df$Weightage_1)
```

```
## [1] 0.5275862
```

```
median(df$Weightage_1)
```

```
## [1] 0
```

```
sd(df$Weightage_1)
```

```
## [1] 1.305336
```

```
names(df)[names(df) == "Weightage 2"] <- "Weightage_2"

mean(df$Weightage_2)
```

```
## [1] 1.187931
```

```
median(df$Weightage_2)
```

```
## [1] 0
```

```
sd(df$Weightage_2)
```

```
## [1] 1.686334
```

```
names(df)[names(df) == "Weightage 3"] <- "Weightage_3"

mean(df$Weightage_3)
```

```
## [1] 2.537931
```

```
median(df$Weightage_3)
```

```
## [1] 1
```

```
sd(df$Weightage_3)
```

```
## [1] 3.193302
```

```
names(df)[names(df) == "Weightage 4"] <- "Weightage_4"

mean(df$Weightage_4)
```

```
## [1] 5.082759
```

```
median(df$Weightage_4)
```

```
## [1] 3
```

```
sd(df$Weightage_4)
```

```
## [1] 7.288293
```

```
names(df)[names(df) == "Weightage 5"] <- "Weightage_5"

mean(df$Weightage_5)
```

```
## [1] 4.974138
```

```
median(df$Weightage_5)
```

```
## [1] 3
```

```
sd(df$Weightage_5)
```

```
## [1] 6.494931
```

Checking For Unique Values

```
names(df)[names(df) == "Basic Course"] <- "Basic_Course"

unique_values <- unique(df$Basic_Course)
print(unique_values)
```

```
##  [1] "B.VOC FOOD TECHNOLOGY"
##  [2] "BACHELOR OF ARTS"
##  [3] "BACHELOR OF COMMERCE (ACCOUNTING AND FINANCE)"
##  [4] "BACHELOR OF COMMERCE (BANKING AND INSURANCE)"
##  [5] "BACHELOR OF MANAGEMENT STUDIES"
##  [6] "BACHELOR OF SCIENCE"
##  [7] "MA PSYCHOLOGY"
##  [8] "MSC COMPUTER SCIENCE"
##  [9] "MSC ANALYTICAL CHEMISTRY"
## [10] "MSC DATA SCIENCE"
## [11] "MSC INFORMATION TECHNOLOGY"
## [12] "MSC MICROBIOLOGY"
## [13] "MSC ORGANIC CHEMISTRY"
## [14] "MSC PHYSICS"
## [15] "BACHELOR OF COMMERCE"
## [16] "B.SC. COMPUTER SCIENCE"
```

```
unique_values <- unique(df$Weightage_1 )
print(unique_values)
```

```
##  [1]  0  1  2  3 19  6 10  4  9  5  7
```

```
unique_values <- unique(df$Weightage_2)
print(unique_values)
```

```
## [1] 0 1 2 5 3 6 4 7 8
```

```
unique_values <- unique(df$Weightage_3 )
print(unique_values)
```

```
##  [1]  1  0 13 14  7 17  9  8 22 12 11  6  5 10 26  2  3  4 16
```

```
unique_values <- unique(df$Weightage_4 )
print(unique_values)
```

```
##  [1]  0  1 27 52 28 35 21 10 40 43 32 30 34 46 44 18 36  4 12  3  8  9 11  2  6
## [26] 13  7  5 16 24 17 19 25 14 20 15
```

```
unique_values <- unique(df$Weightage_5)
print(unique_values)
```

```
##  [1]  0  1 31  6 36 17 38 35 34 19 30 27 33 24 15 32 23 25 14 10  9  5 12 13 11
## [26] 16  7  4  3  8  2 20 21 18 22
```

```
# Rename the column
names(df)[names(df) == "Course Name"] <- "Course_Name"

# Get unique values
unique_values <- unique(df$Course_Name)
print(unique_values)
```

```
##  [1] "FY B.VOC FOOD TECHNOLOGY"          "FYBA"
##  [3] "FY BCOM (ACCOUNTING & FINANCE)"    "FY BCOM (BANKING & INSURANCE)"
##  [5] "FYBMS"                             "FYBSC"
##  [7] "MA PSYCHOLOGY - 1"                 "MA PSYCHOLOGY - 3"
##  [9] "M.SC PART - 1 COMPUTER SCIENCE"    "MSC ANALYTICAL CHEMISTRY SEM I"
## [11] "MSC DATA SCIENCE - 1"              "MSC INFORMATION TECHNOLOGY - 1"
## [13] "MSC MICROBIOLOGY - 1"              "MSC ANALYTICAL CHEMISTRY SEM III"
## [15] "M.SC PART - 2 COMPUTER SCIENCE"    "MSC DATA SCIENCE - 3"
## [17] "MSC INFORMATION TECHNOLOGY - 3"    "MSC MICROBIOLOGY - 3"
## [19] "MSC ORGANIC CHEMISTRY - 3"         "MSC PHYSICS - 3"
## [21] "S.Y.B.A.F"                         "SYBCOM"
## [23] "SYBMS"                             "SYBSC"
## [25] "SY COMPUTER SCIENCE"               "TYBA"
## [27] "TYBCOM"                            "TYBMS"
## [29] "TYBSC"
```

```
# Rename the column
names(df)[names(df) == "Total Feedback Given"] <- "Total_Feedback_Given"

# Get unique values
unique_values <- unique(df$Total_Feedback_Given)
print(unique_values)
```

```
##  [1]  1 74 20 26 12  5  7  4  2  3 14  6 17 46 37 27 39  9
```

List of total strength of the batch in Different Courses.

```
List1 <- df %>%
  group_by(`Total Configured`, `Basic_Course`) %>%
  summarize(Count = n()) %>%
  ungroup()
```

```
## `summarise()` has grouped output by 'Total Configured'. You can override using
## the `.groups` argument.
```

```
# Group and count based on 'Basic Course' only
df_grouped <- df %>%
  group_by(`Basic_Course`) %>%
  summarize(Count = n()) %>%
  ungroup()

# Print List1
print(List1)
```

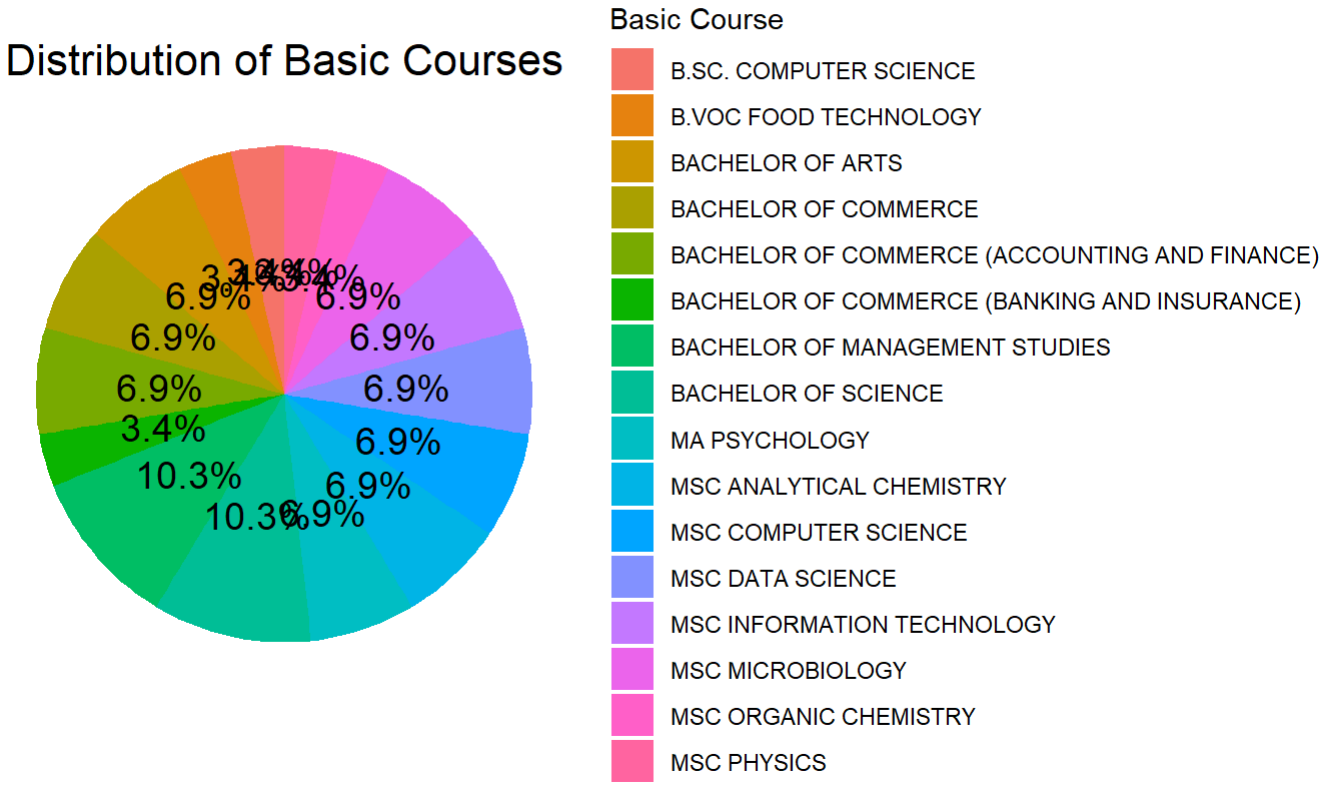
```
## # A tibble: 28 × 3
##    `Total Configured` Basic_Course               Count
##                 <dbl> <chr>                      <int>
##  1                 12 B.VOC FOOD TECHNOLOGY         20
##  2                 14 MSC DATA SCIENCE              20
##  3                 15 MA PSYCHOLOGY                 20
##  4                 18 MSC PHYSICS                   20
##  5                 20 MA PSYCHOLOGY                 20
##  6                 20 MSC COMPUTER SCIENCE          20
##  7                 20 MSC DATA SCIENCE              20
##  8                 20 MSC INFORMATION TECHNOLOGY    20
##  9                 21 MSC INFORMATION TECHNOLOGY    20
## 10                 22 MSC ORGANIC CHEMISTRY         20
## # ℹ 18 more rows
```

5. Data visualization

List of total strength of the batch in Different Courses.

Create a pie chart

```
ggplot(df_grouped, aes(x = "", y = Count, fill = `Basic_Course`)) +
  geom_bar(width = 1, stat = "identity") +
  coord_polar(theta = "y") +
  geom_text(aes(label = paste0(round(Count / sum(Count) * 100, 1), "%")),
            position = position_stack(vjust = 0.5),
            size = 5,
            color = "black") +  # Increase the text size for better visibility
  labs(title = "Distribution of Basic Courses", x = "", y = "") +
  theme_void() +
  theme(
    plot.title = element_text(hjust = 0.5, vjust = 1, size = 16),  # Center the title
    plot.margin = margin(30, 0, 0, 0)  # Add margin at the top to make space for the title
  ) +
  guides(fill = guide_legend(title = "Basic Course"))
```



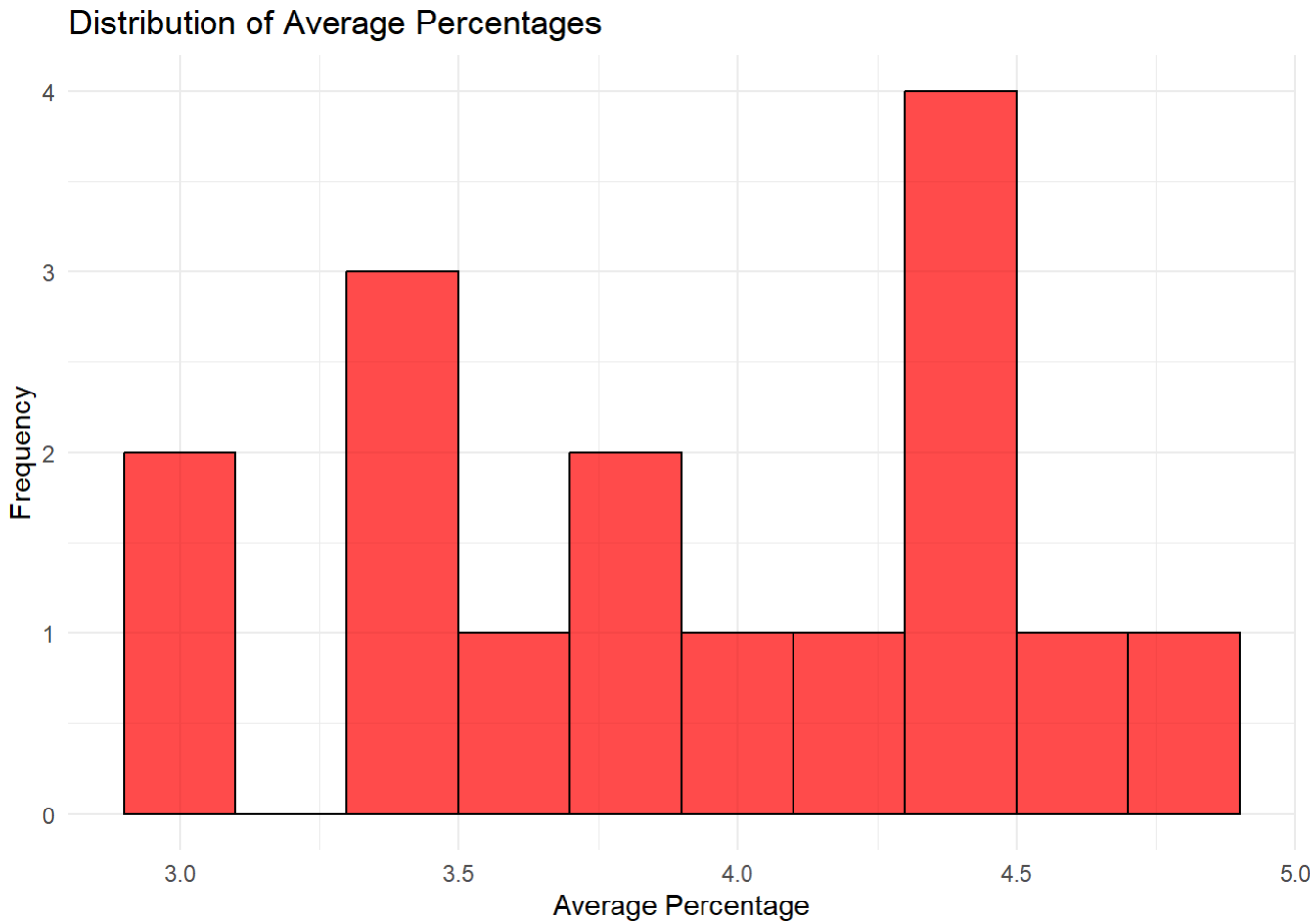
Understanding Student Experience: Insights from Satisfaction Survey

1. **How much of the syllabus was covered in the class ?**

```
# Ensure 'avg_percentage' is a character vector
df <- df %>%
  mutate(`avg_percentage` = as.character(`avg_percentage`)) %>%
  mutate(`avg_percentage` = as.numeric(sapply(strsplit(`avg_percentage`, " / "), `[`, 1)))

# Filter data for the specific question
question_one_df <- df %>%
  filter(Questions == "How much of the syllabus was covered in the class?")

# Calculate the average percentage grouped by 'Basic Course'
average_percentage1 <- question_one_df %>%
  group_by(`Basic_Course`) %>%
  summarize(avg_percentage = mean(`avg_percentage`, na.rm = TRUE))

# Print the result
print(average_percentage1)
```

```
## # A tibble: 16 × 2
##    Basic_Course                                avg_percentage
##    <chr>                                               <dbl>
##  1 B.SC. COMPUTER SCIENCE                               3.56
##  2 B.VOC FOOD TECHNOLOGY                                3
##  3 BACHELOR OF ARTS                                     4.32
##  4 BACHELOR OF COMMERCE                                 3.76
##  5 BACHELOR OF COMMERCE (ACCOUNTING AND FINANCE)        4.35
##  6 BACHELOR OF COMMERCE (BANKING AND INSURANCE)         4.7
##  7 BACHELOR OF MANAGEMENT STUDIES                       3.84
##  8 BACHELOR OF SCIENCE                                  4.05
##  9 MA PSYCHOLOGY                                        3.37
## 10 MSC ANALYTICAL CHEMISTRY                             4.43
## 11 MSC COMPUTER SCIENCE                                 4.41
## 12 MSC DATA SCIENCE                                     3.34
## 13 MSC INFORMATION TECHNOLOGY                           4.84
## 14 MSC MICROBIOLOGY                                     3.48
## 15 MSC ORGANIC CHEMISTRY                                4.12
## 16 MSC PHYSICS                                          3
```

```
ggplot(average_percentage1, aes(x = avg_percentage)) +
  geom_histogram(binwidth = 0.2, fill = "red", color = "black", alpha = 0.7) +
  labs(x = "Average Percentage", y = "Frequency",
       title = "Distribution of Average Percentages") +
  theme_minimal()
```



Summary statistics (min and max)

```
# Calculate min and max of average percentages
min_avg <- min(average_percentage1$avg_percentage, na.rm = TRUE)
max_avg <- max(average_percentage1$avg_percentage, na.rm = TRUE)

# Print min and max
cat("Min Average Percentage:", min_avg, "\n")
```

```
## Min Average Percentage: 3
```

```
cat("Max Average Percentage:", max_avg, "\n")
```

```
## Max Average Percentage: 4.835
```

Observation

Here are some observations you can make about the data for question 1: “How much of the syllabus was covered in the class?”

Overall Coverage:

It’s difficult to determine a single “coverage” value for the entire dataset. The provided values indicate a range of coverage from 3.00 (lowest) to 4.84 (highest). Course Types:

Bachelor Degrees: Bachelor’s degrees seem to have an average coverage between 3.33 and 4.35 (excluding outliers). This suggests a moderate to good portion of the syllabus was covered in these classes. Master’s Degrees: Master’s degrees generally show higher coverage, ranging from 3.00 to 4.84. This implies a potentially more comprehensive syllabus coverage in these classes. However, the sample size for Master’s degrees is smaller (5) compared to Bachelor’s (8), so the observation might not be statistically significant.

Outliers:

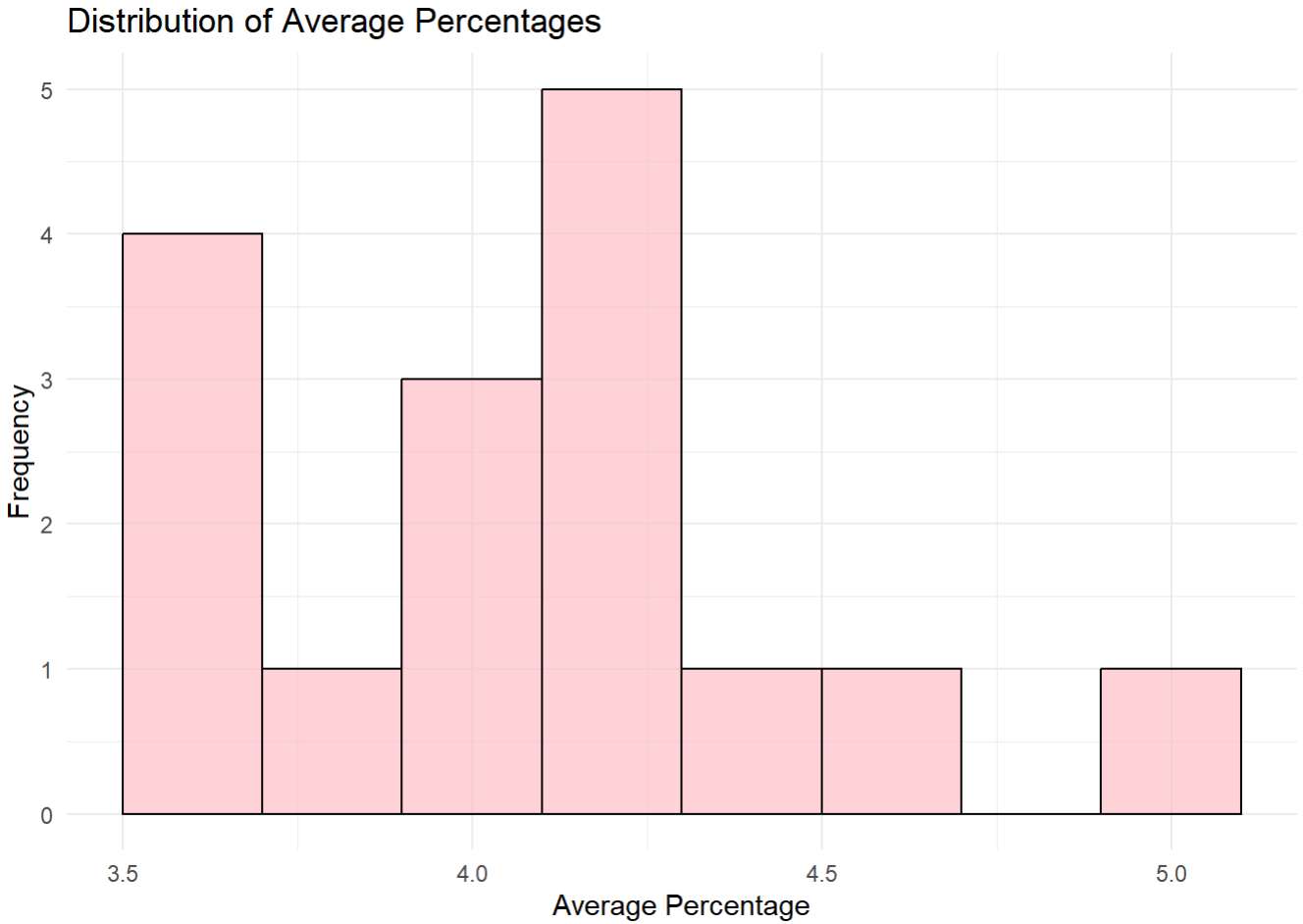
BACHELOR OF COMMERCE (BANKING AND INSURANCE): This course has the highest reported coverage (4.70). It might be worth investigating if this high coverage is accurate or represents an anomaly in the data collection process. MSC INFORMATION TECHNOLOGY: This course has the highest reported coverage (4.84) among Master’s degrees. Similar to the previous outlier, a closer look might be beneficial to confirm the accuracy

2. **How well did the teachers prepare for the classes ?**

```
# Convert 'avg_percentage' to numeric
df$`avg_percentage` <- as.numeric(df$`avg_percentage`)

# Filter data for the specific question
question_two_df <- df %>%
  filter(Questions == "How well did the teachers prepare for the classes?")

# Calculate the average percentage grouped by 'Basic Course'
average_percentage2 <- question_two_df %>%
  group_by(`Basic_Course`) %>%
  summarize(Average_Percentage = mean(`avg_percentage`, na.rm = TRUE))

# Print the result
print(average_percentage2)
```

```
## # A tibble: 16 × 2
##    Basic_Course                                Average_Percentage
##    <chr>                                                  <dbl>
##  1 B.SC. COMPUTER SCIENCE                                  3.56
##  2 B.VOC FOOD TECHNOLOGY                                   5
##  3 BACHELOR OF ARTS                                        4.68
##  4 BACHELOR OF COMMERCE                                    3.82
##  5 BACHELOR OF COMMERCE (ACCOUNTING AND FINANCE)           3.94
##  6 BACHELOR OF COMMERCE (BANKING AND INSURANCE)            4.05
##  7 BACHELOR OF MANAGEMENT STUDIES                          4.23
##  8 BACHELOR OF SCIENCE                                     4.21
##  9 MA PSYCHOLOGY                                           3.64
## 10 MSC ANALYTICAL CHEMISTRY                                4.28
## 11 MSC COMPUTER SCIENCE                                    4.02
## 12 MSC DATA SCIENCE                                        3.67
## 13 MSC INFORMATION TECHNOLOGY                              4.25
## 14 MSC MICROBIOLOGY                                        3.69
## 15 MSC ORGANIC CHEMISTRY                                   4.18
## 16 MSC PHYSICS                                             4.5
```

```
# Plotting with ggplot2
ggplot(average_percentage2, aes(x = Average_Percentage)) +
  geom_histogram(binwidth = 0.2, fill = "pink", color = "black", alpha = 0.7) +
  labs(x = "Average Percentage", y = "Frequency",
       title = "Distribution of Average Percentages") +
  theme_minimal()
```



```
# Calculate min and max of average percentages
min_avg <- min(average_percentage2$Average_Percentage, na.rm = TRUE)
max_avg <- max(average_percentage2$Average_Percentage, na.rm = TRUE)

# Print min and max
cat("Min Average Percentage:", min_avg, "\n")
```

```
## Min Average Percentage: 3.56
```

```
cat("Max Average Percentage:", max_avg, "\n")
```

```
## Max Average Percentage: 5
```

Observation

Here are some observations you can make about the data for question 2: “How well did the teachers prepare for the classes?”

Teacher Preparation:

The data suggests a range of perceived teacher preparedness, from 3.56 (lowest) to 5.00 (highest). This indicates that students have varying opinions about how well teachers prepared for classes. Course Types:

Bachelor Degrees: The average perceived teacher preparedness seems to be between 3.8 and 4.6 (excluding outliers).

Master’s Degrees: The average perceived teacher preparedness seems to be slightly higher (around 4.0) for Master’s degrees compared to Bachelor’s degrees. However, the sample size for Master’s degrees is smaller (5) compared to Bachelor’s (8), so the observation might not be statistically significant. Outliers:

B.VOC FOOD TECHNOLOGY: This course has the highest reported teacher preparedness (5.00). It’s worth considering if this exceptional rating reflects a unique situation or represents the general perception for this course. There are no clear outliers on the low end.

3. **Average Rating Given by Different Departments Across 20 Questions ?**

```
df$`avg_percentage` <- as.numeric(df$`avg_percentage`)

# Group by 'Basic Course' and calculate the mean of 'avg_percentage'
df_grouped <- df %>%
  group_by(`Basic_Course`) %>%
  summarize(Average_Percentage = mean(`avg_percentage`, na.rm = TRUE))

# Calculate the overall mean of the grouped averages
average_average_rating <- mean(df_grouped$avg_percentage, na.rm = TRUE)
```

```
## Warning: Unknown or uninitialised column: `avg_percentage`.
```

```
## Warning in mean.default(df_grouped$avg_percentage, na.rm = TRUE): argument is
## not numeric or logical: returning NA
```

```
# Print the results
cat("Average average rating by course (sum of average ratings across 20 questions):\n")
```

```
## Average average rating by course (sum of average ratings across 20 questions):
```

```
print(df_grouped)
```

```
## # A tibble: 16 × 2
##    Basic_Course                                 Average_Percentage
##    <chr>                                                     <dbl>
##  1 B.SC. COMPUTER SCIENCE                                     3.35
##  2 B.VOC FOOD TECHNOLOGY                                      3.4
##  3 BACHELOR OF ARTS                                           4.34
##  4 BACHELOR OF COMMERCE                                       3.72
##  5 BACHELOR OF COMMERCE (ACCOUNTING AND FINANCE)              4.08
##  6 BACHELOR OF COMMERCE (BANKING AND INSURANCE)               4.35
##  7 BACHELOR OF MANAGEMENT STUDIES                             4.09
##  8 BACHELOR OF SCIENCE                                        3.85
##  9 MA PSYCHOLOGY                                              3.60
## 10 MSC ANALYTICAL CHEMISTRY                                   4.03
## 11 MSC COMPUTER SCIENCE                                       3.85
## 12 MSC DATA SCIENCE                                           3.05
## 13 MSC INFORMATION TECHNOLOGY                                 4.35
## 14 MSC MICROBIOLOGY                                           3.54
## 15 MSC ORGANIC CHEMISTRY                                      3.97
## 16 MSC PHYSICS                                                3.42
```

```
cat("\nOverall average average rating across all courses:\n")
```

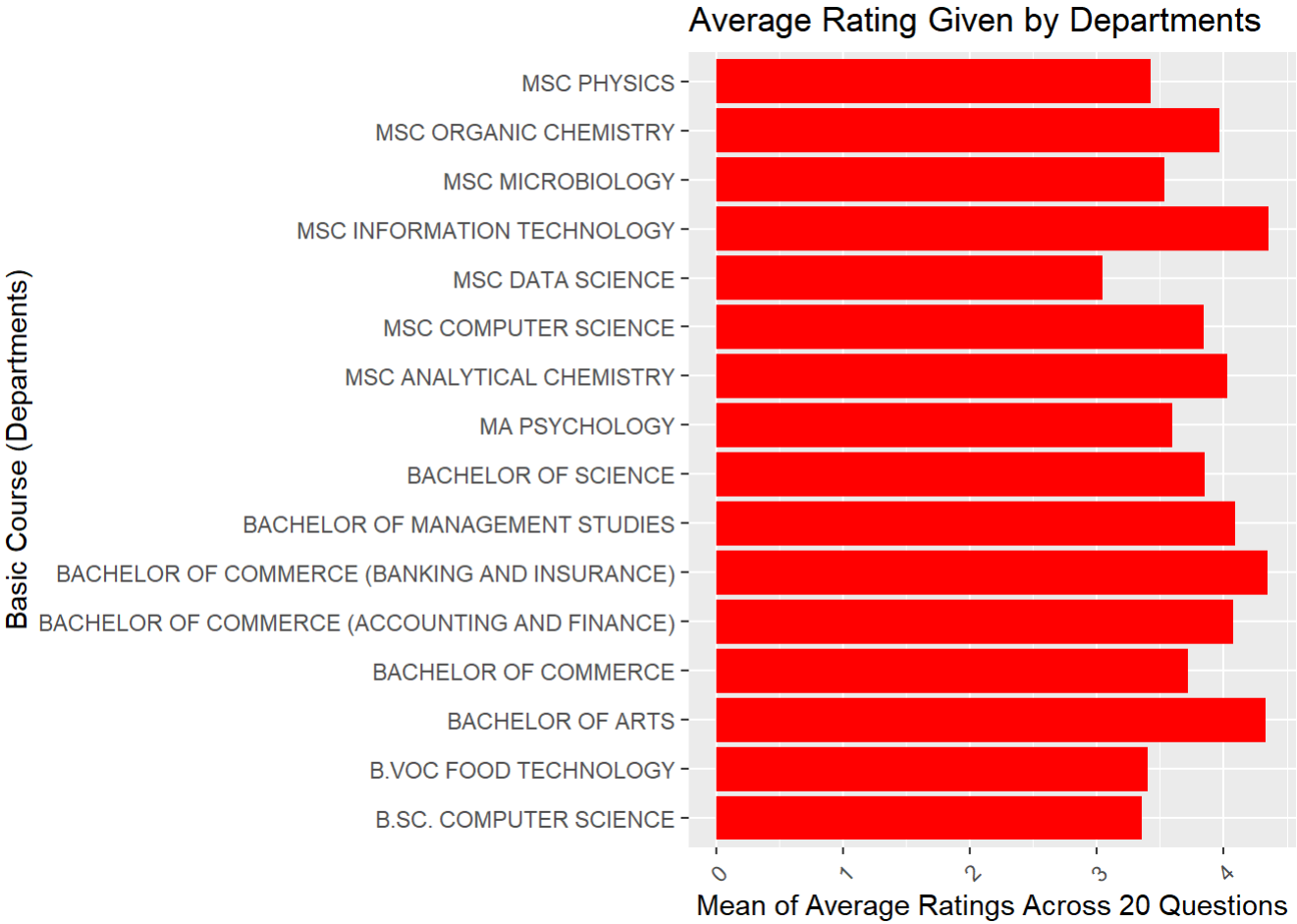
```
##
## Overall average average rating across all courses:
```

```
print(average_average_rating)
```

```
## [1] NA
```

Plotting with ggplot2

```
ggplot(df_grouped, aes(x = Basic_Course, y = Average_Percentage)) +
  geom_bar(stat = "identity", fill = "red") +
  labs(x = "Basic Course (Departments)", y = "Mean of Average Ratings Across 20 Questions",
       title = "Average Rating Given by Departments") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  coord_flip()
```



Observations on Average Average Rating by Course:

The data shows a variation in student perceptions (3.050500 to 4.354500) of the overall teaching quality across courses, likely based on the sum of average ratings provided for 20 questions. Here’s a breakdown of the findings:

Higher Average Ratings:

BACHELOR OF ARTS (4.335000) & BACHELOR OF COMMERCE (BANKING AND INSURANCE) (4.350000) - Students perceive the highest average quality in these programs. BACHELOR OF COMMERCE (ACCOUNTING AND FINANCE) (4.075500), BACHELOR OF MANAGEMENT STUDIES (4.092167), MSC INFORMATION TECHNOLOGY (4.354500) - These courses also show above average ratings.

Lower Average Ratings:

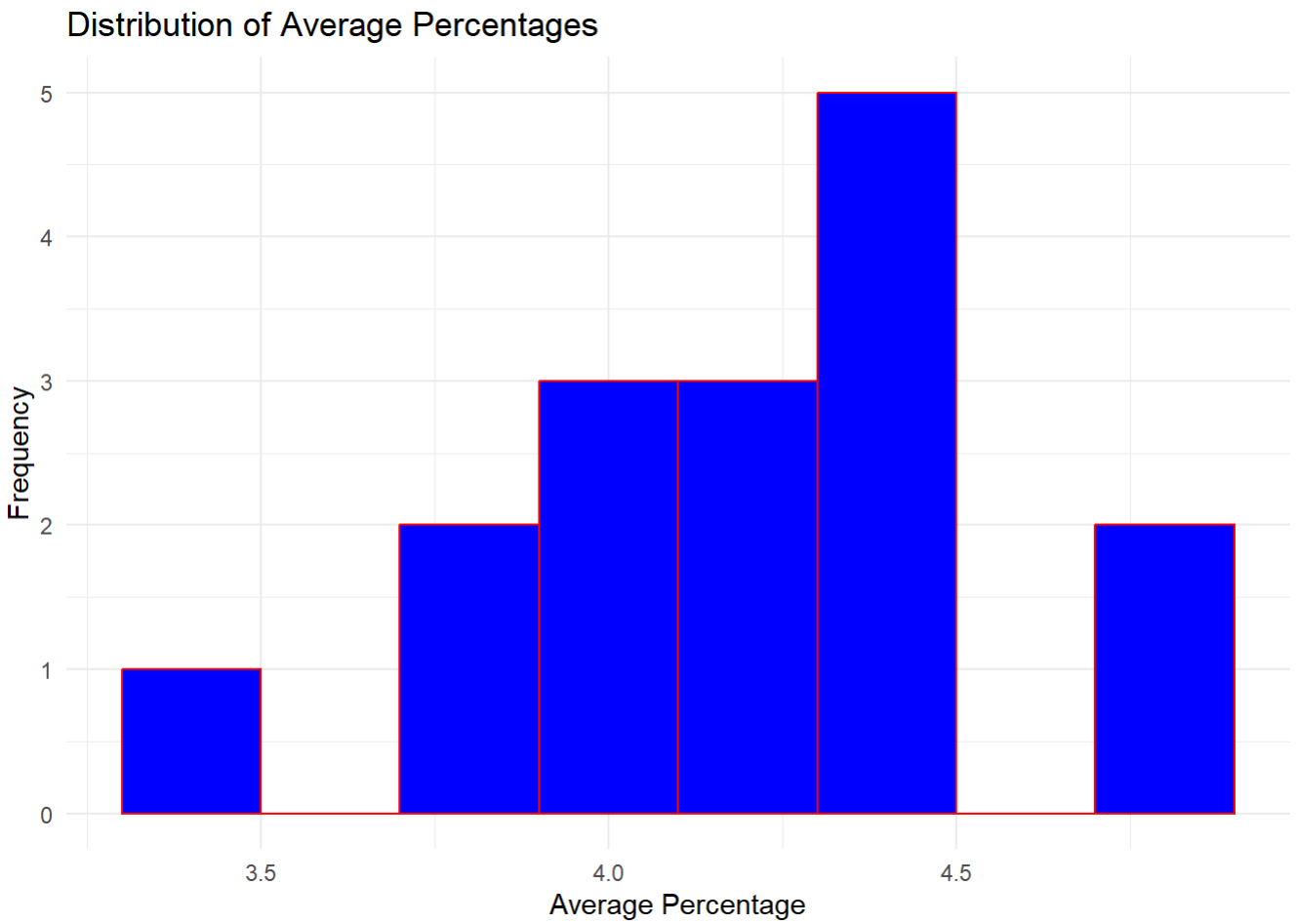
MSC DATA SCIENCE (3.050500) - Students perceive the lowest average quality in this program. B.VOC FOOD TECHNOLOGY (3.400000), MSC PHYSICS (3.425000) - These courses along with others fall below the overall average (average is not provided but can be calculated).

4. **Fairness of the internal evaluation process by the teachers.**

```
question_five_df <- df %>%
  filter(Questions == "Fairness of the internal evaluation process by the teachers.")

# Calculate the average percentage grouped by 'Basic Course'
average_percentage5 <- question_five_df %>%
  group_by(`Basic_Course`) %>%
  summarize(Average_Percentage = mean(`avg_percentage`, na.rm = TRUE))

# Print the result
print(average_percentage5)
```

```
## # A tibble: 16 × 2
##    Basic_Course                                  Average_Percentage
##    <chr>                                                      <dbl>
##  1 B.SC. COMPUTER SCIENCE                                      3.41
##  2 B.VOC FOOD TECHNOLOGY                                       4
##  3 BACHELOR OF ARTS                                            4.77
##  4 BACHELOR OF COMMERCE                                        3.90
##  5 BACHELOR OF COMMERCE (ACCOUNTING AND FINANCE)               4.28
##  6 BACHELOR OF COMMERCE (BANKING AND INSURANCE)                4.45
##  7 BACHELOR OF MANAGEMENT STUDIES                              4.17
##  8 BACHELOR OF SCIENCE                                         4.03
##  9 MA PSYCHOLOGY                                               4.42
## 10 MSC ANALYTICAL CHEMISTRY                                    4.28
## 11 MSC COMPUTER SCIENCE                                        4.30
## 12 MSC DATA SCIENCE                                            3.84
## 13 MSC INFORMATION TECHNOLOGY                                  4.75
## 14 MSC MICROBIOLOGY                                            3.90
## 15 MSC ORGANIC CHEMISTRY                                       4.41
## 16 MSC PHYSICS                                                 4.5
```

```
# Plotting the histogram
ggplot(average_percentage5, aes(x = Average_Percentage)) +
  geom_histogram(binwidth = 0.2, fill = "blue", color = "red") +
  labs(x = "Average Percentage", y = "Frequency",
       title = "Distribution of Average Percentages") +
  theme_minimal()
```




```
# Calculate min and max of average percentages
min_avg <- min(average_percentage5$Average_Percentage, na.rm = TRUE)
max_avg <- max(average_percentage5$Average_Percentage, na.rm = TRUE)

# Print min and max
cat("Min Average Percentage:", min_avg, "\n")
```

```
## Min Average Percentage: 3.41
```

```
cat("Max Average Percentage:", max_avg, "\n")
```

```
## Max Average Percentage: 4.765
```

Observation : Here's a summary of the data for question 5: "Fairness of the internal evaluation process by the teachers."

Perceived Fairness:

The data shows a range of perceptions regarding fairness in the internal evaluation process, from 3.41 (lowest) to 4.765 (highest). This indicates that students have varying opinions on how fair the evaluation process was in different courses. Course Types:

Bachelor Degrees: The average perceived fairness for Bachelor's degrees seems to be around 4.0 and 4.5 (excluding outliers). Master's Degrees: Master's degrees generally show slightly higher perceived fairness, with an average around 4.3. However, the sample size for Master's degrees is smaller (5) compared to Bachelor's (8), so the observation might not be statistically significant. Outliers:

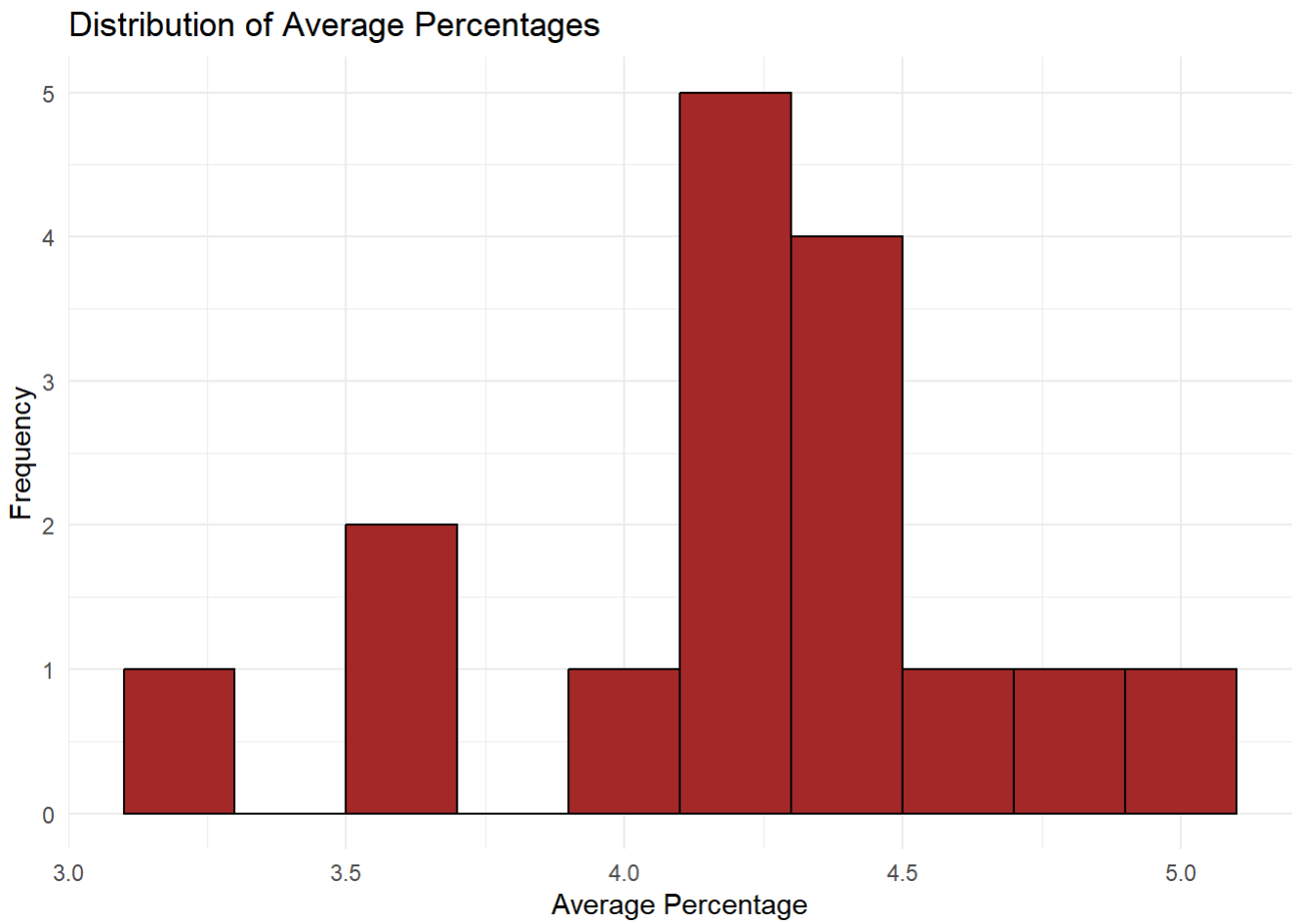
BACHELOR OF ARTS: This course has the highest reported fairness rating (4.765). It might be worth investigating if this exceptional rating reflects a unique situation or represents the general perception for this course. B.SC. COMPUTER SCIENCE: This course has the lowest reported fairness rating (3.41). It might be valuable to explore why students felt the evaluation process in this course was less fair.

5.**How well were the teachers able to communicate ?**

```
# Filter data for the specific question
question_three_df <- df %>%
  filter(Questions == "How well were the teachers able to communicate?")

# Calculate the average percentage grouped by 'Basic Course'
average_percentage3 <- question_three_df %>%
  group_by(`Basic_Course`) %>%
  summarize(Average_Percentage = mean(`avg_percentage`, na.rm = TRUE))

# Print the result
print(average_percentage3)
```

```
## # A tibble: 16 × 2
##    Basic_Course                                 Average_Percentage
##    <chr>                                                     <dbl>
##  1 B.SC. COMPUTER SCIENCE                                     3.63
##  2 B.VOC FOOD TECHNOLOGY                                      5
##  3 BACHELOR OF ARTS                                           4.73
##  4 BACHELOR OF COMMERCE                                       4.14
##  5 BACHELOR OF COMMERCE (ACCOUNTING AND FINANCE)              4.36
##  6 BACHELOR OF COMMERCE (BANKING AND INSURANCE)               4.5
##  7 BACHELOR OF MANAGEMENT STUDIES                             4.17
##  8 BACHELOR OF SCIENCE                                        4.32
##  9 MA PSYCHOLOGY                                              3.68
## 10 MSC ANALYTICAL CHEMISTRY                                   4.22
## 11 MSC COMPUTER SCIENCE                                       4.23
## 12 MSC DATA SCIENCE                                           3.16
## 13 MSC INFORMATION TECHNOLOGY                                 4.58
## 14 MSC MICROBIOLOGY                                           4.16
## 15 MSC ORGANIC CHEMISTRY                                      4.06
## 16 MSC PHYSICS                                                4.5
```

```
ggplot(average_percentage3, aes(x = Average_Percentage)) +
  geom_histogram(binwidth = 0.2, fill = "brown", color = "black") +
  labs(x = "Average Percentage", y = "Frequency",
       title = "Distribution of Average Percentages") +
  theme_minimal()
```




```
# Calculate min and max of average percentages
min_avg <- min(average_percentage3$Average_Percentage, na.rm = TRUE)
max_avg <- max(average_percentage3$Average_Percentage, na.rm = TRUE)

# Print min and max
cat("Min Average Percentage:", min_avg, "\n")
```

```
## Min Average Percentage: 3.165
```

```
cat("Max Average Percentage:", max_avg, "\n")
```

```
## Max Average Percentage: 5
```

Observation : Here are some observations you can make about the data for question 3: “How well were the teachers able to communicate?”

Teacher Communication:

The data suggests a range of perceived teacher communication skills, from 3.165 (lowest) to 5.00 (highest). This indicates that students have varying opinions about how effectively teachers communicated the course material. Course Types:

Bachelor Degrees: The average perceived teacher communication seems to be between 4.1 and 4.7 (excluding outliers). Master’s Degrees: The average perceived teacher communication seems to be slightly higher (around 4.2) for Master’s degrees compared to Bachelor’s degrees. However, the sample size for Master’s degrees is smaller (5) compared to Bachelor’s (8), so the observation might not be statistically significant. Outliers:

B.VOC FOOD TECHNOLOGY: This course has the highest reported teacher communication rating (5.00). It’s worth considering if this exceptional rating reflects a unique situation or represents the general perception for this course. MSC DATA SCIENCE: This course has the lowest reported teacher communication rating (3.165). It might be valuable to investigate why students felt the communication in this course was less effective.