INDIVIDUAL ASSIGNMENT

TECHNOLOGY PARK MALAYSIA

CT127-3-2-PFDA

PROGRAMMING FOR DATA ANALYSIS

APD2F2302CS(IS)

HAND OUT DATE: 29 MARCH 2023

HAND IN DATE: 22 MAY 2023

WEIGHTAGE: 50%

INSTRUCTIONS TO CANDIDATES:

1. Submit your assignment at the administrative counter.
2. Students are advised to underpin their answers with the use of references (cited using the American Psychological Association (APA) Referencing).
3. Late submissions will be awarded zero (0) unless Extenuating Circumstances (EC) are upheld.
4. Cases of plagiarism will be penalized.
5. The assignment should be bound in an appropriate style (comb bound or stapled). Where the assignment should be submitted in both hardcopy and softcopy, the softcopy of the written assignment and source code (where appropriate) should be on a CD in an envelope / CD cover and attached to the hardcopy.
6. You must obtain 50% overall to pass this module.

Table of Content

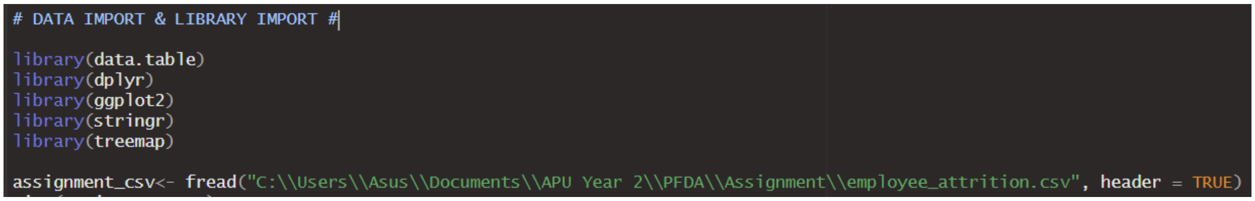
1.0 Introduction and Assumption

This report is written based on the analysis of the data set by utilizing RStudio to help with data analysis. The given dataset is about employee attrition. The data have several pieces of information that could be used for analysis, such as hire date, termination date, the current status of the employee, department, job title, etc. to perform relevant analysis.

The main purpose of this assignment is to find the relationship between 2 or more several columns and visualize the relationship between the column of a given employee attrition dataset with the graph and do some in-depth analysis about what is actually happening and help the Human Resources department to make a meaningful insight for decision making.

2.0 Data Import/ Data Cleaning/ Data Pre-processing/ Data Exploration

2.1 Data Import

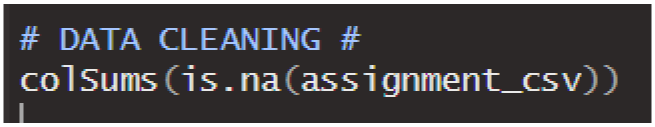


```
# DATA IMPORT & LIBRARY IMPORT #

library(data.table)
library(dplyr)
library(ggplot2)
library(stringr)
library(treemap)

assignment_csv<- fread("C:\\Users\\Asus\\Documents\\APU Year 2\\PFDA\\Assignment\\employee_attrition.csv", header = TRUE)
```

Figure 2.1.1 Library and dataset import

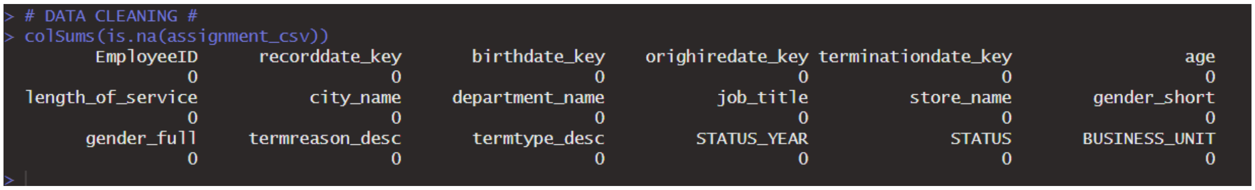
Before importing the dataset from the csv file, libraries and packages have to be loaded into Rstudio. The currently loaded libraries are “data.table”, “dplyr”, “ggplot2”, “stringr”, and “treemap”.

2.2 Data Cleaning



```
# DATA CLEANING #
colSums(is.na(assignment_csv))
```

Figure 2.2.1 Data cleaning

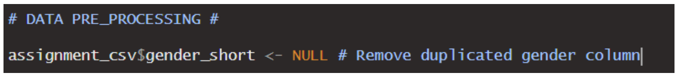


```
> # DATA CLEANING #
> colSums(is.na(assignment_csv))
       EmployeeID      recorddate_key       birthdate_key    orighiredate_key terminationdate_key                 age 
                0                   0                   0                   0                   0                   0 
length_of_service           city_name     department_name           job_title          store_name        gender_short 
                0                   0                   0                   0                   0                   0 
      gender_full     termreason_desc       termtype_desc         STATUS_YEAR              STATUS       BUSINESS_UNIT 
                0                   0                   0                   0                   0                   0 
>
```

Figure 2.2.2 Output of data cleaning

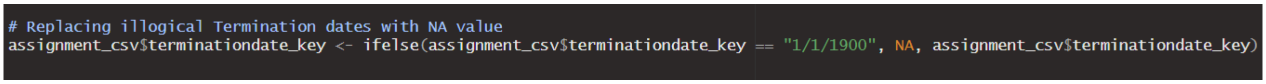
The code above is run to confirm that there are no missing, junk or “NA” values present in the data set that may hinder the analysis process later. According to the output produced by the code, there seem to be no missing values in the current data set.

2.3 Data Pre-processing



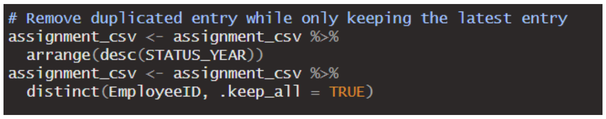
```
# DATA PRE_PROCESSING #

assignment_csv$gender_short <- NULL # Remove duplicated gender column
```

Figure 2.3.1 Removing duplicated columns



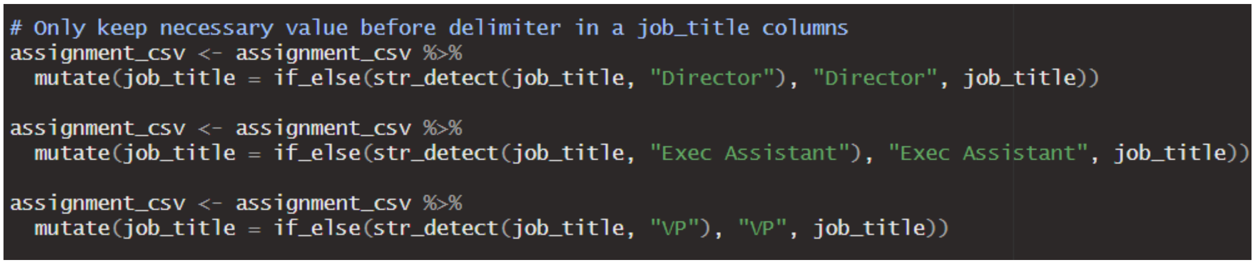
```
# Replacing illogical Termination dates with NA value
assignment_csv$terminationdate_key <- ifelse(assignment_csv$terminationdate_key == "1/1/1900", NA, assignment_csv$terminationdate_key)
```

Figure 2.3.2 Replacing illogical data with "NA"



```
# Remove duplicated entry while only keeping the latest entry
assignment_csv <- assignment_csv %>%
  arrange(desc(STATUS_YEAR))
assignment_csv <- assignment_csv %>%
  distinct(EmployeeID, .keep_all = TRUE)
```

Figure 2.3.3 Removing of redundant rows

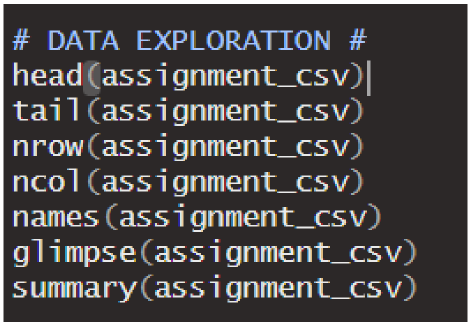


```
# Only keep necessary value before delimiter in a job_title columns
assignment_csv <- assignment_csv %>%
  mutate(job_title = if_else(str_detect(job_title, "Director"), "Director", job_title))

assignment_csv <- assignment_csv %>%
  mutate(job_title = if_else(str_detect(job_title, "Exec Assistant"), "Exec Assistant", job_title))

assignment_csv <- assignment_csv %>%
  mutate(job_title = if_else(str_detect(job_title, "VP"), "VP", job_title))
```

Figure 2.3.4 Remove and keep only values needed in the "job_title" column

Data pre-processing is when manipulation (e.g. changing data type and format) or dropping of data (e.g. removal of redundant rows or columns) is performed to ensure or enhance performance. While I explore the CSV file, I notice there is a duplicate column "gender", that column is dropped by executing the code shown in Figure 2.3.1. Moreover, there were anomaly data stored in the rows found in "terminationdate_key" whereby the default termination date assigned to the column is illogical. Therefore, those values were replaced with "NA" by executing the code shown in Figure 2.3.2. Additionally, I also notice that there are duplicated entries of the same employee at different years. I decided to keep the most recent record and drop the old redundant data from the dataset by executing the code in Figure 2.3.3. Last but not least, I also notice that some entities of job titles contain departments with a comma delimiter. To keep data consistent, codes shown in Figure 2.3.4 is executed to remove the redundancies.

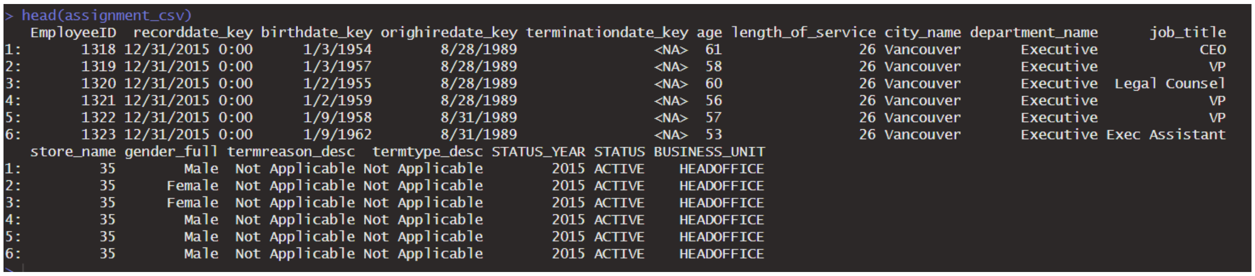
2.4 Data Exploration



```
# DATA EXPLORATION #
head(assignment_csv)
tail(assignment_csv)
nrow(assignment_csv)
ncol(assignment_csv)
names(assignment_csv)
glimpse(assignment_csv)
summary(assignment_csv)
```

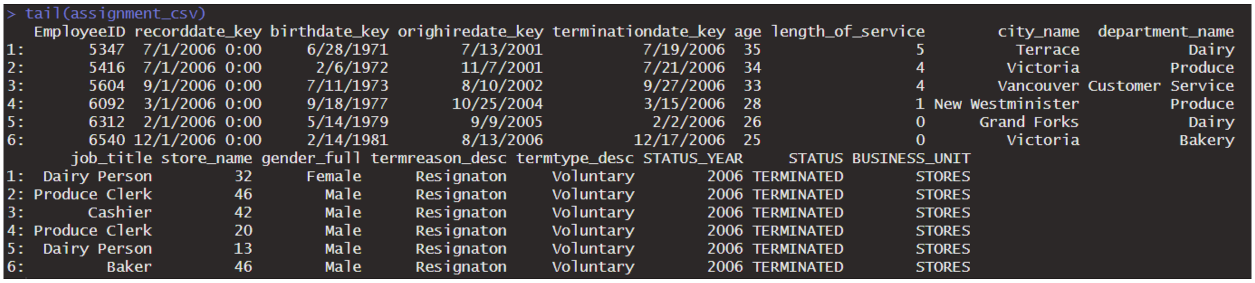
Figure 2.4.1 Data Exploration

Before we start analysing, we should know the nature and characteristics of the data set first to gain some insights. The code shown in Figure 2.4.1 will we executed to perform data exploration.



```
> head(assignment_csv)
   EmployeeID  recorddate_key birthdate_key orighiredate_key terminationdate_key age length_of_service city_name department_name      job_title
1:       1318 12/31/2015 0:00      1/3/1954        8/28/1989                <NA>  61                26 Vancouver       Executive            CEO
2:       1319 12/31/2015 0:00      1/3/1957        8/28/1989                <NA>  58                26 Vancouver       Executive             VP
3:       1320 12/31/2015 0:00      1/2/1955        8/28/1989                <NA>  60                26 Vancouver       Executive  Legal Counsel
4:       1321 12/31/2015 0:00      1/2/1959        8/28/1989                <NA>  56                26 Vancouver       Executive             VP
5:       1322 12/31/2015 0:00      1/9/1958        8/31/1989                <NA>  57                26 Vancouver       Executive             VP
6:       1323 12/31/2015 0:00      1/9/1962        8/31/1989                <NA>  53                26 Vancouver       Executive Exec Assistant
   store_name gender_full termreason_desc  termtype_desc STATUS_YEAR STATUS BUSINESS_UNIT
1:         35        Male  Not Applicable Not Applicable        2015 ACTIVE    HEADOFFICE
2:         35      Female  Not Applicable Not Applicable        2015 ACTIVE    HEADOFFICE
3:         35      Female  Not Applicable Not Applicable        2015 ACTIVE    HEADOFFICE
4:         35        Male  Not Applicable Not Applicable        2015 ACTIVE    HEADOFFICE
5:         35        Male  Not Applicable Not Applicable        2015 ACTIVE    HEADOFFICE
6:         35        Male  Not Applicable Not Applicable        2015 ACTIVE    HEADOFFICE
```

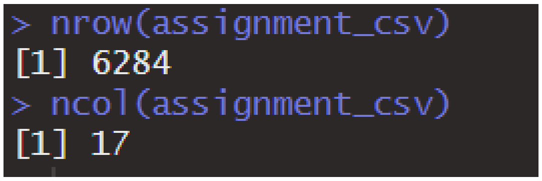
Figure 2.4.2 first few rows of dataset using head()



```
> tail(assignment_csv)
   EmployeeID recorddate_key birthdate_key orighiredate_key terminationdate_key age length_of_service       city_name  department_name
1:       5347  7/1/2006 0:00     6/28/1971        7/13/2001           7/19/2006  35                 5         Terrace            Dairy
2:       5416  7/1/2006 0:00      2/6/1972        11/7/2001           7/21/2006  34                 4        Victoria          Produce
3:       5604  9/1/2006 0:00     7/11/1973        8/10/2002           9/27/2006  33                 4       Vancouver Customer Service
4:       6092  3/1/2006 0:00     9/18/1977       10/25/2004           3/15/2006  28                 1 New Westminister          Produce
5:       6312  2/1/2006 0:00     5/14/1979         9/9/2005            2/2/2006  26                 0     Grand Forks            Dairy
6:       6540 12/1/2006 0:00     2/14/1981        8/13/2006          12/17/2006  25                 0        Victoria           Bakery
        job_title store_name gender_full termreason_desc termtype_desc STATUS_YEAR     STATUS BUSINESS_UNIT
1:  Dairy Person         32      Female      Resignaton     Voluntary        2006 TERMINATED        STORES
2: Produce Clerk         46        Male      Resignaton     Voluntary        2006 TERMINATED        STORES
3:       Cashier         42        Male      Resignaton     Voluntary        2006 TERMINATED        STORES
4: Produce Clerk         20        Male      Resignaton     Voluntary        2006 TERMINATED        STORES
5:  Dairy Person         13        Male      Resignaton     Voluntary        2006 TERMINATED        STORES
6:         Baker         46        Male      Resignaton     Voluntary        2006 TERMINATED        STORES
```

Figure 2.4.3 last few rows of dataset using tail()

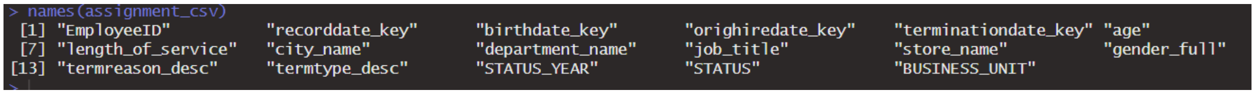
The head() and tail() functions will return the first 6 rows of our data set and the last 6 rows of our data set to gain brief understanding on the structure and content of data set we will be using later.



```
> nrow(assignment_csv)
[1] 6284
> ncol(assignment_csv)
[1] 17
```

Figure 2.4.4 Number of columns and row using nrow() and ncol()

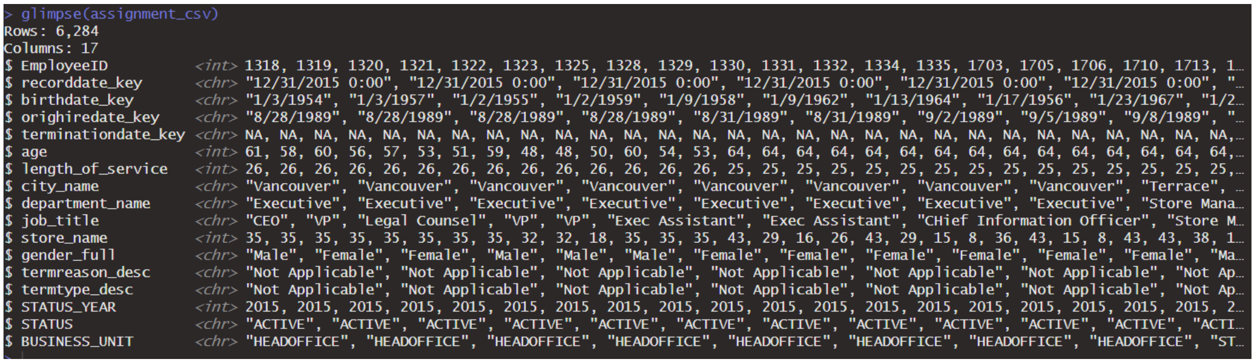
The nrow() will return the total number of rows in the data set whereas the ncol() will return the total number of columns in the data set. We can get a rough idea of how big our data set is.



```
> names(assignment_csv)
 [1] "EmployeeID"         "recorddate_key"      "birthdate_key"       "orighiredate_key"    "terminationdate_key" "age"
 [7] "length_of_service"  "city_name"           "department_name"     "job_title"           "store_name"          "gender_full"
[13] "termreason_desc"    "termtype_desc"       "STATUS_YEAR"         "STATUS"              "BUSINESS_UNIT"
```

Figure 2.4.5 Heading of dataset columns using names()

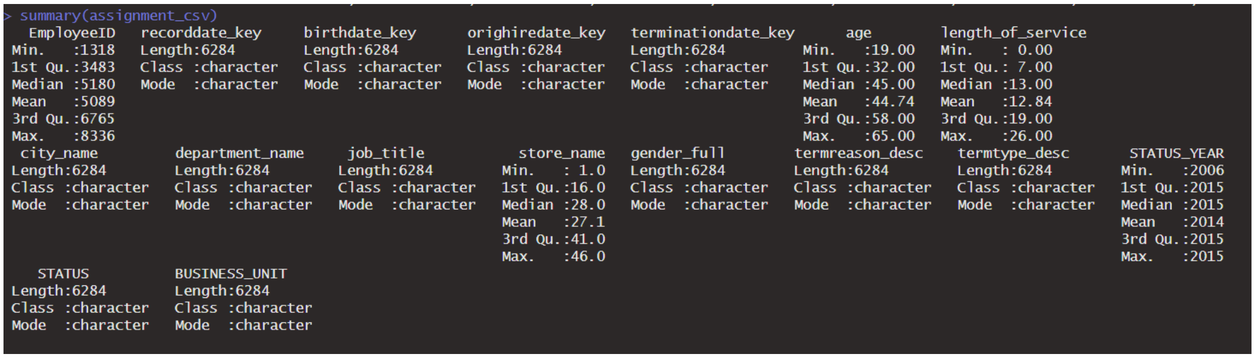
The name() will return the column heading of your dataset. So we have a rough idea of what kind of data we will be dealing with.



```
> glimpse(assignment_csv)
Rows: 6,284
Columns: 17
$ EmployeeID          <int> 1318, 1319, 1320, 1321, 1322, 1323, 1325, 1328, 1329, 1330, 1331, 1332, 1334, 1335, 1703, 1705, 1706, 1710, 1713, 1…
$ recorddate_key      <chr> "12/31/2015 0:00", "12/31/2015 0:00", "12/31/2015 0:00", "12/31/2015 0:00", "12/31/2015 0:00", "12/31/2015 0:00", "…
$ birthdate_key       <chr> "1/3/1954", "1/3/1957", "1/2/1955", "1/2/1959", "1/9/1958", "1/9/1962", "1/13/1964", "1/17/1956", "1/23/1967", "1/2…
$ orighiredate_key    <chr> "8/28/1989", "8/28/1989", "8/28/1989", "8/28/1989", "8/31/1989", "8/31/1989", "9/2/1989", "9/5/1989", "9/8/1989", "…
$ terminationdate_key <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,…
$ age                 <int> 61, 58, 60, 56, 57, 53, 51, 59, 48, 48, 50, 60, 54, 53, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64,…
$ length_of_service   <int> 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25,…
$ city_name           <chr> "Vancouver", "Vancouver", "Vancouver", "Vancouver", "Vancouver", "Vancouver", "Vancouver", "Vancouver", "Terrace", …
$ department_name     <chr> "Executive", "Executive", "Executive", "Executive", "Executive", "Executive", "Executive", "Executive", "Store Mana…
$ job_title           <chr> "CEO", "VP", "Legal Counsel", "VP", "VP", "Exec Assistant", "Exec Assistant", "CHief Information Officer", "Store M…
$ store_name          <int> 35, 35, 35, 35, 35, 35, 35, 35, 32, 32, 18, 35, 35, 35, 43, 29, 16, 26, 43, 29, 15, 8, 36, 43, 15, 8, 43, 43, 38, 1…
$ gender_full         <chr> "Male", "Female", "Female", "Male", "Male", "Male", "Female", "Female", "Female", "Female", "Female", "Female", "Ma…
$ termreason_desc     <chr> "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Ap…
$ termtype_desc       <chr> "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Ap…
$ STATUS_YEAR         <int> 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2…
$ STATUS              <chr> "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTI…
$ BUSINESS_UNIT       <chr> "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "ST…
```

Figure 2.4.6 Data exploration with glimpse()

You can use glimpse() to display a few rows of the data with a column displayed and read horizontally and a row being read and displayed vertically like an inverted Excel axis. Unlike the head() and tail() function that displays column vertically and row horizontally. The glimpse() function makes it more uncomplicatedto read.



```
> summary(assignment_csv)
   EmployeeID   recorddate_key     birthdate_key      orighiredate_key   terminationdate_key      age        length_of_service
 Min.   :1318   Length:6284        Length:6284        Length:6284        Length:6284         Min.   :19.00   Min.   : 0.00
 1st Qu.:3483   Class :character   Class :character   Class :character   Class :character    1st Qu.:32.00   1st Qu.: 7.00
 Median :5180   Mode  :character   Mode  :character   Mode  :character   Mode  :character    Median :45.00   Median :13.00
 Mean   :5089                                                                                Mean   :44.74   Mean   :12.84
 3rd Qu.:6765                                                                                3rd Qu.:58.00   3rd Qu.:19.00
 Max.   :8336                                                                                Max.   :65.00   Max.   :26.00
  city_name          department_name      job_title            store_name    gender_full        termreason_desc    termtype_desc       STATUS_YEAR
 Length:6284        Length:6284        Length:6284        Min.   : 1.0   Length:6284        Length:6284        Length:6284        Min.   :2006
 Class :character   Class :character   Class :character   1st Qu.:16.0   Class :character   Class :character   Class :character   1st Qu.:2015
 Mode  :character   Mode  :character   Mode  :character   Median :28.0   Mode  :character   Mode  :character   Mode  :character   Median :2015
                                                          Mean   :27.1                                                            Mean   :2014
                                                          3rd Qu.:41.0                                                            3rd Qu.:2015
                                                          Max.   :46.0                                                            Max.   :2015
    STATUS          BUSINESS_UNIT
 Length:6284        Length:6284
 Class :character   Class :character
 Mode  :character   Mode  :character
```

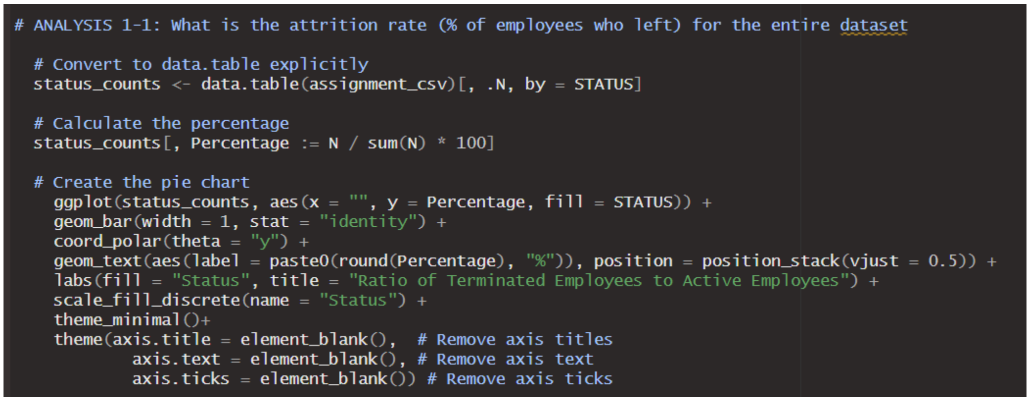
Figure 2.4.7 Data exploration with summary ()

Summary() is used to summarize the data set's data to determine what type of data each column contains and to calculate the mean, median, minimum and maximum. It is very helpful to have this function as it gives you the complete output of the function.

3.0 Question & Analysis

3.1 Question 1: Why do employees leave the organisation?

3.1.1 Analysis 1-1: What is the attrition rate from the entire dataset



```
# ANALYSIS 1-1: What is the attrition rate (% of employees who left) for the entire dataset

  # Convert to data.table explicitly
  status_counts <- data.table(assignment_csv)[, .N, by = STATUS]

  # Calculate the percentage
  status_counts[, Percentage := N / sum(N) * 100]

  # Create the pie chart
    ggplot(status_counts, aes(x = "", y = Percentage, fill = STATUS)) +
    geom_bar(width = 1, stat = "identity") +
    coord_polar(theta = "y") +
    geom_text(aes(label = paste0(round(Percentage), "%")), position = position_stack(vjust = 0.5)) +
    labs(fill = "Status", title = "Ratio of Terminated Employees to Active Employees") +
    scale_fill_discrete(name = "Status") +
    theme_minimal()+
    theme(axis.title = element_blank(),  # Remove axis titles
            axis.text = element_blank(), # Remove axis text
            axis.ticks = element_blank()) # Remove axis ticks
```

Figure 3.1.1: Attrition rate of entire dataset code

The approach of Analysis 1-1 is to explore a general knowledge of the organization's attrition rate. It provides an initial understanding of the scope of employee turnover and establishes a baseline for future investigation. A pie chart is plotted using “ggplot” based on the percentage calculated to visualise the proportion of Active and Terminated employees.

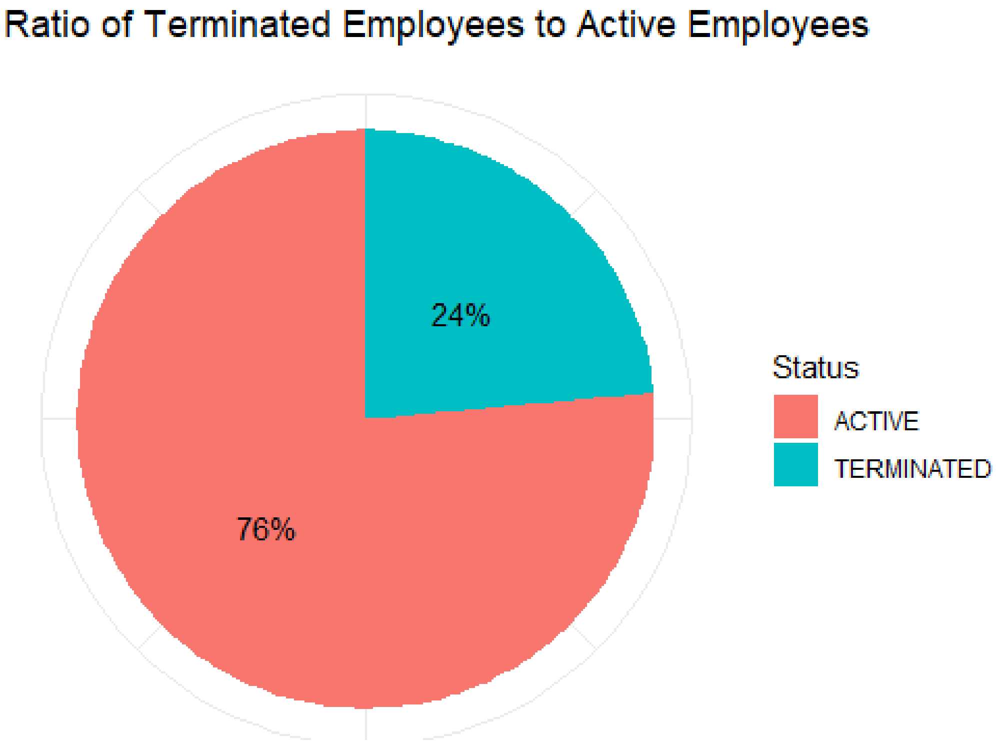
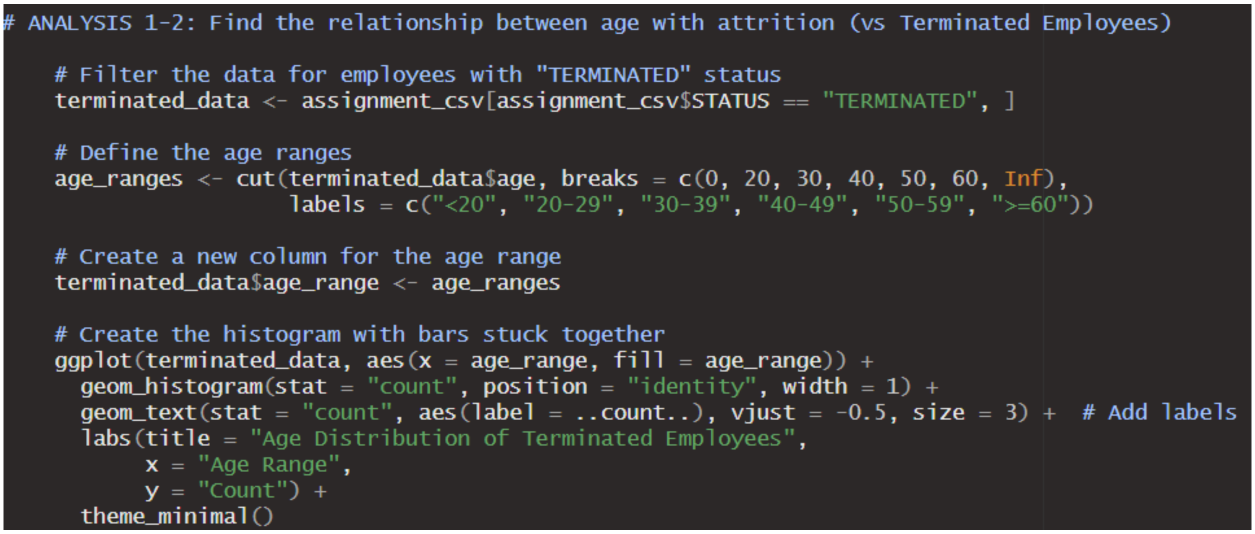


Figure 3.1.2: Ratio of terminated employees to active employees pie chart

The figure above shows the percentage of employees with “Terminated” and “Active” status from the entire dataset. The result shows that 76% of the employees recorded held “Active” status whereas employee with “Terminated status” accounts for 24%. This implies that the overall attrition rate is relatively low. However, we shall uncover the reasons that influence employees to leave the company.

3.1.3 Analysis 1-2: Find the relationship between age with attrition



```r
# ANALYSIS 1-2: Find the relationship between age with attrition (vs Terminated Employees)

    # Filter the data for employees with "TERMINATED" status
    terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]

    # Define the age ranges
    age_ranges <- cut(terminated_data$age, breaks = c(0, 20, 30, 40, 50, 60, Inf),
                      labels = c("<20", "20-29", "30-39", "40-49", "50-59", ">=60"))

    # Create a new column for the age range
    terminated_data$age_range <- age_ranges

    # Create the histogram with bars stuck together
    ggplot(terminated_data, aes(x = age_range, fill = age_range)) +
      geom_histogram(stat = "count", position = "identity", width = 1) +
      geom_text(stat = "count", aes(label = ..count..), vjust = -0.5, size = 3) +  # Add labels
      labs(title = "Age Distribution of Terminated Employees",
           x = "Age Range",
           y = "Count") +
      theme_minimal()
```

Figure 3.1.3: Relationship between age with attrition codes

Analysis 1-2 seeks to discover the association between age and attrition to see whether there are any patterns or trends in terms of age groups that are more likely to quit the organisation. The “geom_histogram()” function in the “ggplot” package is used to plot a histogram that illustrates the age distribution of the terminated employees, with the x-axis representing the “age ranges” and the y-axis showing the “ employee count”. The bars in the histogram are placed together in a way that there are no gaps between them with different colours of bad representing a distinct age range.

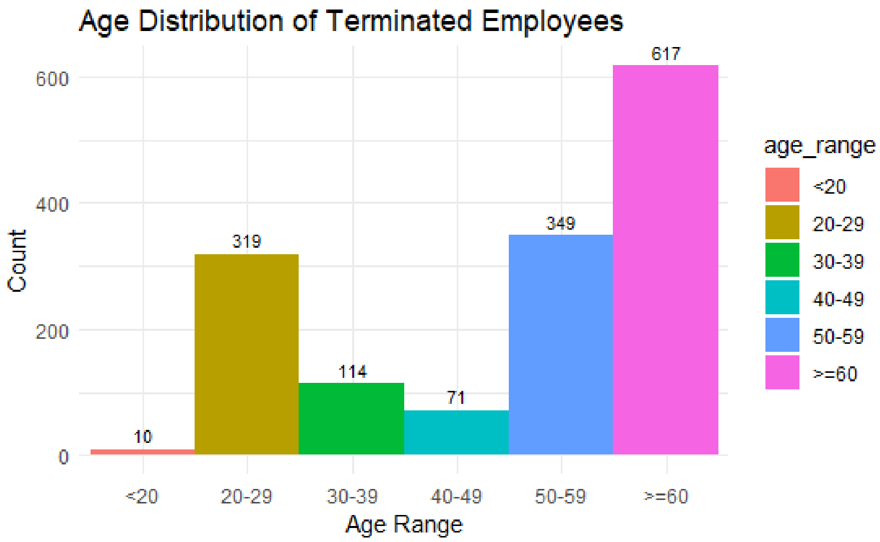


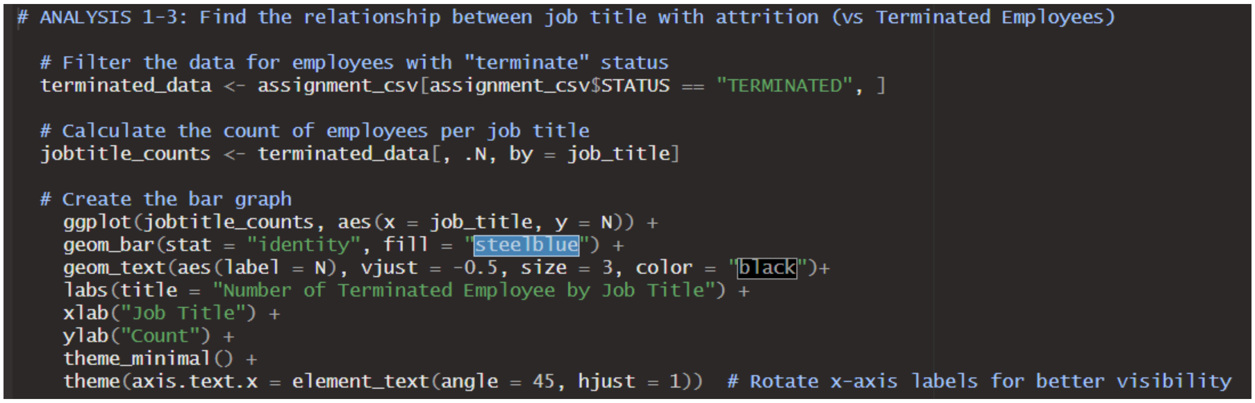
Figure 3.1.4: Relationship between age with attrition histogram

The histogram shows that the most significant number of terminated employees is among employees over 60 years old. After that, there are age categories "50-59", "20-29", "30-39", and "40-49", with age category <20 being the least significant.

The frequency count is significant for employees over 60, potentially due to retirement or health-related reasons that hinder them from attending work.

A relatively low number of terminated employees over the age of 20 may be due to a lack of experience and a desire to gain experience in the current organization.

3.1.3 Analysis 1-3: Find the relationship between job title with attrition



```
# ANALYSIS 1-3: Find the relationship between job title with attrition (vs Terminated Employees)

  # Filter the data for employees with "terminate" status
  terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]

  # Calculate the count of employees per job title
  jobtitle_counts <- terminated_data[, .N, by = job_title]

  # Create the bar graph
    ggplot(jobtitle_counts, aes(x = job_title, y = N)) +
    geom_bar(stat = "identity", fill = "steelblue") +
    geom_text(aes(label = N), vjust = -0.5, size = 3, color = "black")+
    labs(title = "Number of Terminated Employee by Job Title") +
    xlab("Job Title") +
    ylab("Count") +
    theme_minimal() +
    theme(axis.text.x = element_text(angle = 45, hjust = 1))  # Rotate x-axis labels for better visibility
```

Figure 3.1.5: Relationship between job title with attrition codes

Analysis 1-3 seeks to determine whether particular employment roles or positions have greater turnover rates by investigating the association between job title and attrition. It can assist evaluate whether certain job titles are more likely to result in employee termination. A bar graph is created with the geom_bar() function and the stat = "identity" parameter. This guarantees that a bar graph is generated in which the height of each bar matches directly with the number of employees for each job title. The x-axis displays job titles, while the y-axis displays the number of dismissed workers.

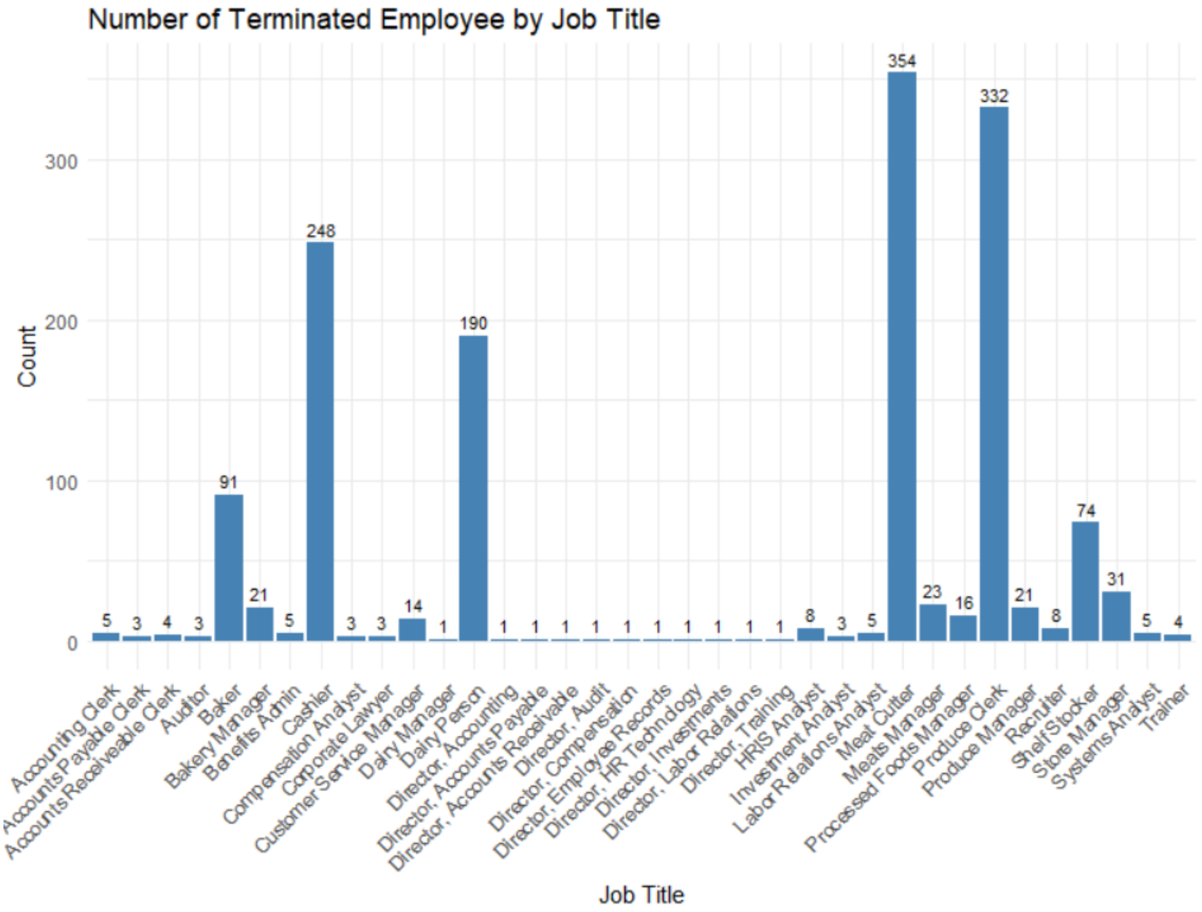
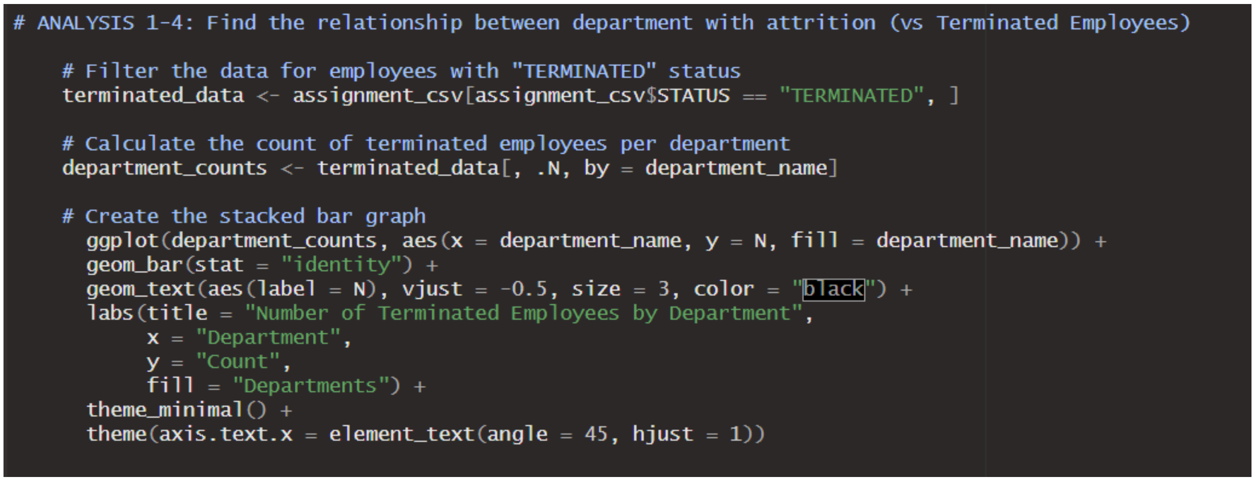


Figure 3.1.6: Relationship Between Job Title with Attrition Bar Chart

Based on the graph above, employee termination rates are more significant for Produce Clerks, Meat Cutters, Bakers, Cashiers, Dairy People, and Shelf stockers. This occurrence may be caused by the nature of the occupation: these occupations may require physically demanding or repetitive work, leading to greater turnover rates and termination rates; the working environment: Stress or discontent may increase employee turnover if the work environment is stressful or dissatisfying, such as working in a fast-paced retail setting or dealing with customer contacts.

3.1.4 Analysis1-4: Find the relationship between the department with attrition



```
# ANALYSIS 1-4: Find the relationship between department with attrition (vs Terminated Employees)

    # Filter the data for employees with "TERMINATED" status
    terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]

    # Calculate the count of terminated employees per department
    department_counts <- terminated_data[, .N, by = department_name]

    # Create the stacked bar graph
      ggplot(department_counts, aes(x = department_name, y = N, fill = department_name)) +
      geom_bar(stat = "identity") +
      geom_text(aes(label = N), vjust = -0.5, size = 3, color = "black") +
      labs(title = "Number of Terminated Employees by Department",
           x = "Department",
           y = "Count",
           fill = "Departments") +
      theme_minimal() +
      theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.1.7: Relationship Between department with Attrition codes

Analysis 1-4 seeks to investigate the connection between departments, and attrition might indicate if some departments have greater turnover rates. It assists in determining if particular job responsibilities are more vulnerable to employee termination. A stacked bar graph is then created by using the geom_bar() function in "ggplot" to visualise the number of dismissed workers by the department. The x-axis shows the division's names, while the y-axis shows the number of dismissed workers.

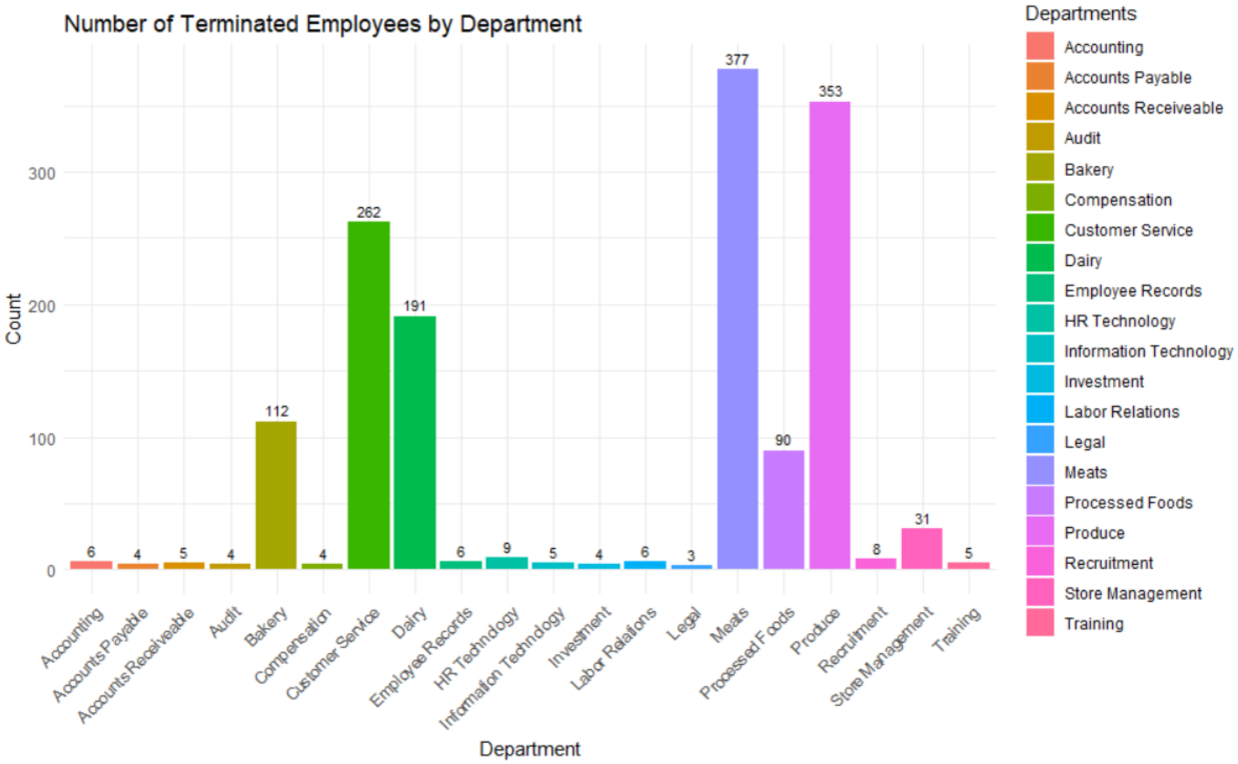
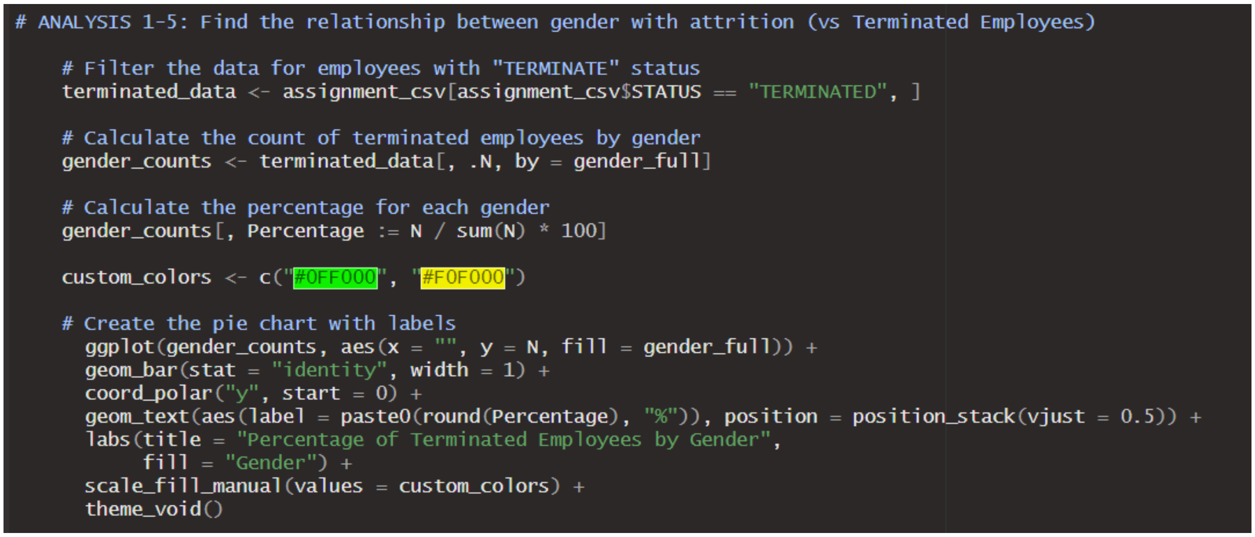


Figure 3.1.8: Relationship Between department with Attrition barchart

Based on the graph above, Bakery, Customer Service, Dairy, Meats, Processed Foods, and Produce have a high turnover rate. This occurs possibly due to variables such as rigorous workloads (e.g. resolving customer inquiries or complaints, strict quality control standards), restricted career growth prospects due to repetitive tasks, and possible job satisfaction problems (e.g. lack of recognition).

3.1.5 Analysis 1-5: Find the relationship between gender with attrition



```
# ANALYSIS 1-5: Find the relationship between gender with attrition (vs Terminated Employees)

    # Filter the data for employees with "TERMINATE" status
    terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]

    # Calculate the count of terminated employees by gender
    gender_counts <- terminated_data[, .N, by = gender_full]

    # Calculate the percentage for each gender
    gender_counts[, Percentage := N / sum(N) * 100]

    custom_colors <- c("#0FF000", "#F0F000")

    # Create the pie chart with labels
      ggplot(gender_counts, aes(x = "", y = N, fill = gender_full)) +
      geom_bar(stat = "identity", width = 1) +
      coord_polar("y", start = 0) +
      geom_text(aes(label = paste0(round(Percentage), "%")), position = position_stack(vjust = 0.5)) +
      labs(title = "Percentage of Terminated Employees by Gender",
           fill = "Gender") +
      scale_fill_manual(values = custom_colors) +
      theme_void()
```

Figure 3.1.9: Relationship Between gender with Attrition codes

Analysis 1-5 examines the link between gender and attrition to see whether there are any gender disparities in employee termination rates. It assists in determining if gender contributes to employee turnover. To visualise the percentage of dismissed workers by gender, a pie chart is plotted using geom_bar() in the ggplot package. The function coord_polar("y", start = 0) is used to transform a bar chart into a pie chart with a circular layout. The x-axis is kept blank, while the y-axis shows the number of dismissed workers.

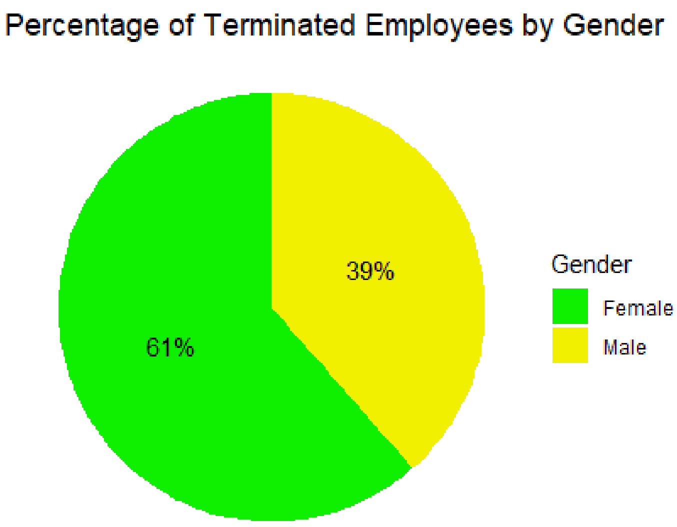
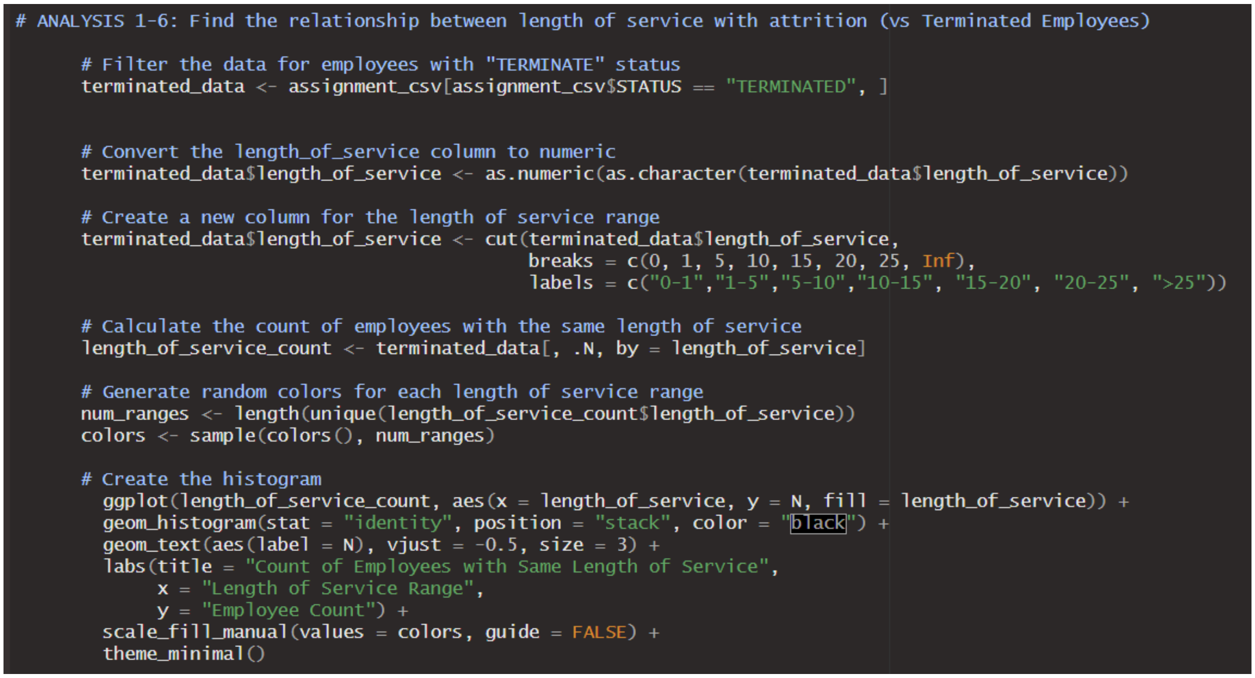


Figure 3.1.10: Relationship between gender with attrition pie chart

The pie chart above shows the percentage of female and male terminated employees. From the chart, female accounts for 61% of the overall termination whereas male accounts for 39%. The significant termination rate among female employees could be due to workplace culture and biases (e.g. gender discrimination, gender inequality, and gender stereotypes) or due to family factors (e.g. pregnancy).

3.1.6 Analysis 1-6: Determine if attrition rates differ based on tenure (length of service) categories.



```
# ANALYSIS 1-6: Find the relationship between length of service with attrition (vs Terminated Employees)

      # Filter the data for employees with "TERMINATE" status
      terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]


      # Convert the length_of_service column to numeric
      terminated_data$length_of_service <- as.numeric(as.character(terminated_data$length_of_service))

      # Create a new column for the length of service range
      terminated_data$length_of_service <- cut(terminated_data$length_of_service,
                                               breaks = c(0, 1, 5, 10, 15, 20, 25, Inf),
                                               labels = c("0-1","1-5","5-10","10-15", "15-20", "20-25", ">25"))

      # Calculate the count of employees with the same length of service
      length_of_service_count <- terminated_data[, .N, by = length_of_service]

      # Generate random colors for each length of service range
      num_ranges <- length(unique(length_of_service_count$length_of_service))
      colors <- sample(colors(), num_ranges)

      # Create the histogram
        ggplot(length_of_service_count, aes(x = length_of_service, y = N, fill = length_of_service)) +
        geom_histogram(stat = "identity", position = "stack", color = "black") +
        geom_text(aes(label = N), vjust = -0.5, size = 3) +
        labs(title = "Count of Employees with Same Length of Service",
             x = "Length of Service Range",
             y = "Employee Count") +
        scale_fill_manual(values = colors, guide = FALSE) +
        theme_minimal()
```

Figure 3.1.11: Relationship between employee count and tenure codes

Analysis 1-6 looks at attrition rates depending on tenure categories to see if workers with various periods of service have varied termination rates. It aids in determining whether there is some connection between length of service and attrition. The "geom_histogram()" function in the "ggplot" package is used to plot a histogram that illustrates the number of employees with the same length of service. The length of the service range is shown by the x-axis, while the number of employees is represented by the y-axis.

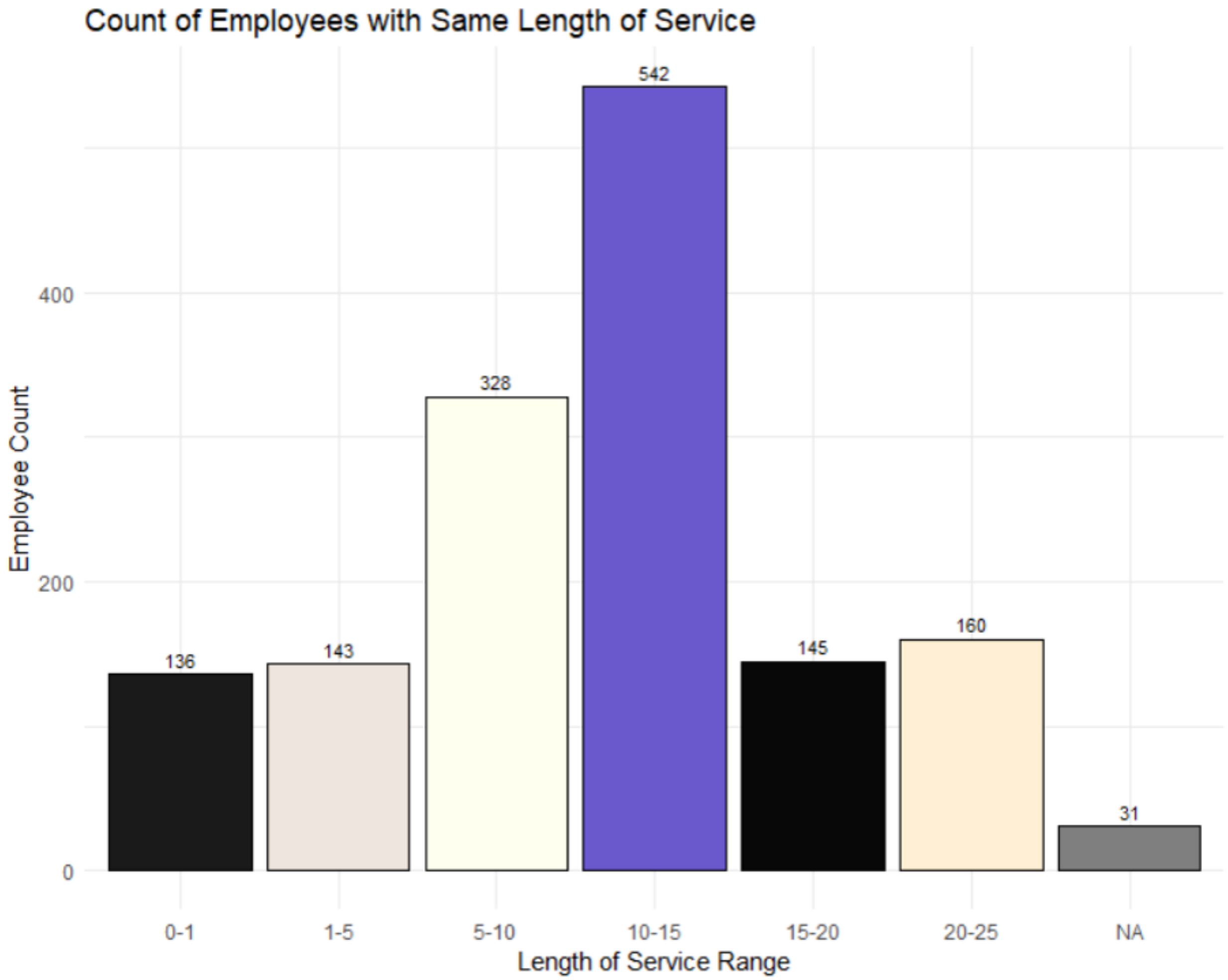
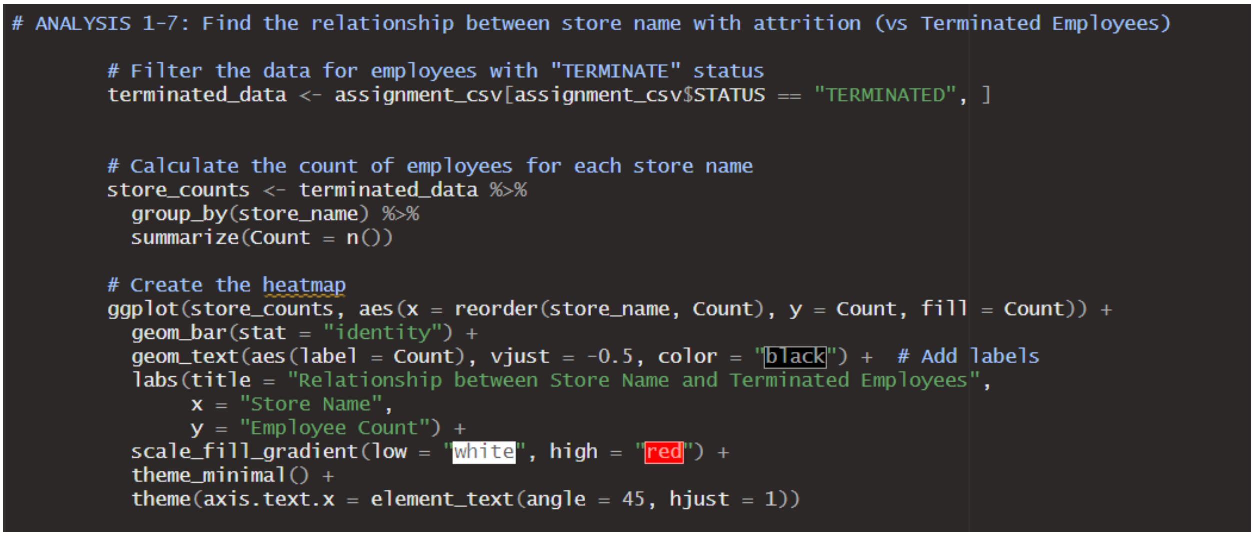


Figure 3.1.12: Relationship between attrition and employee count barchart

From the histogram above, employees with a length of service in the range of 10-15 years had the highest termination rate among the various lengths of service ranges. This data implies that employees who have been with the organisation for a long time, especially within this range, are more likely to be terminated possibly due to job dissatisfaction or misalignment of skills (e.g. inability to adapt to digital transformation).

3.1.7 Analysis 1-7: Find the relationship between store name with attrition



```
# ANALYSIS 1-7: Find the relationship between store name with attrition (vs Terminated Employees)

        # Filter the data for employees with "TERMINATE" status
        terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]


        # Calculate the count of employees for each store name
        store_counts <- terminated_data %>%
          group_by(store_name) %>%
          summarize(Count = n())

        # Create the heatmap
        ggplot(store_counts, aes(x = reorder(store_name, Count), y = Count, fill = Count)) +
          geom_bar(stat = "identity") +
          geom_text(aes(label = Count), vjust = -0.5, color = "black") +  # Add labels
          labs(title = "Relationship between Store Name and Terminated Employees",
               x = "Store Name",
               y = "Employee Count") +
          scale_fill_gradient(low = "white", high = "red") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.1.13: Relationship between store name and employee count codes

Analysis 1-7 intends to investigate the association between store names and attrition in order to determine whether there are certain stores with greater staff turnover rates. Heatmap is plotted using geom_bar() to illustrate this association, colour depth implies the rate of employee termination.

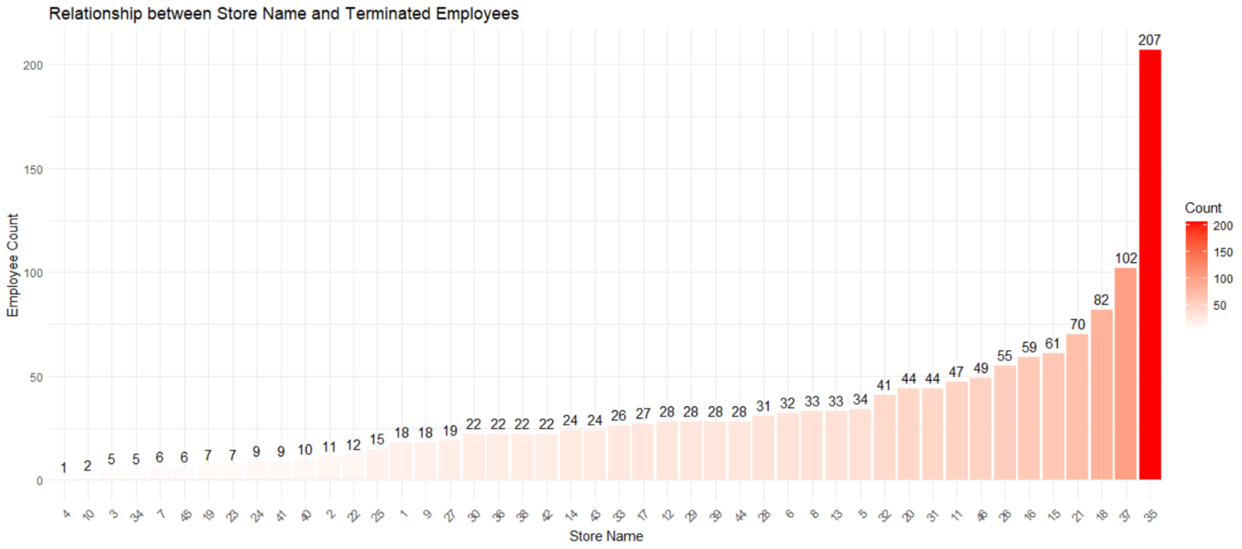
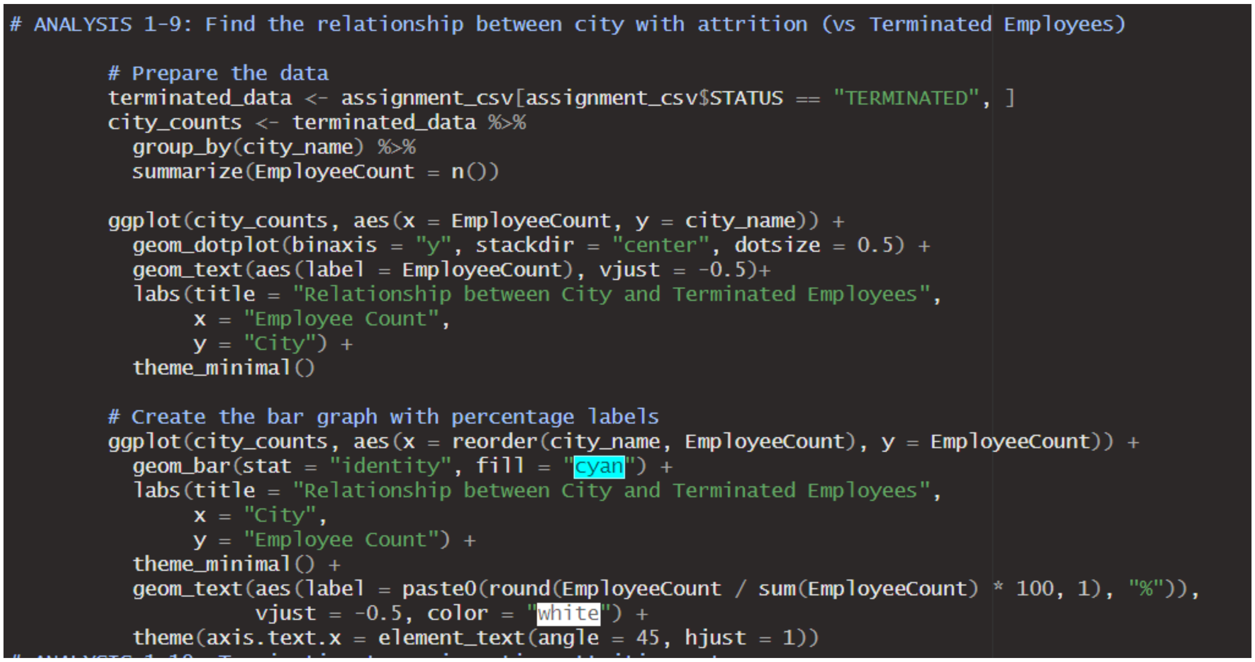


Figure 3.1.14: Relationship between store name and employee count barchart

From the heatmap above, it is apparent that store 35 has a high termination rate. This could be caused by a number of factors such as management issues (e.g. poor management, ineffective communication, lack of guidance), high workloads (e.g. high customer volume) or geographical factors: (e.g. store located in a remote area or competitive area).

3.1.8 Analysis 1-8: Find the relationship between the city with attrition



```
# ANALYSIS 1-9: Find the relationship between city with attrition (vs Terminated Employees)

        # Prepare the data
        terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]
        city_counts <- terminated_data %>%
          group_by(city_name) %>%
          summarize(EmployeeCount = n())

        ggplot(city_counts, aes(x = EmployeeCount, y = city_name)) +
          geom_dotplot(binaxis = "y", stackdir = "center", dotsize = 0.5) +
          geom_text(aes(label = EmployeeCount), vjust = -0.5)+
          labs(title = "Relationship between City and Terminated Employees",
               x = "Employee Count",
               y = "City") +
          theme_minimal()

        # Create the bar graph with percentage labels
        ggplot(city_counts, aes(x = reorder(city_name, EmployeeCount), y = EmployeeCount)) +
          geom_bar(stat = "identity", fill = "cyan") +
          labs(title = "Relationship between City and Terminated Employees",
               x = "City",
               y = "Employee Count") +
          theme_minimal() +
          geom_text(aes(label = paste0(round(EmployeeCount / sum(EmployeeCount) * 100, 1), "%")),
                    vjust = -0.5, color = "white") +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.1.15: Relationship between city and employee count codes

Analysis 1-8 strives to investigate the association between cities and attrition in order to establish whether there are certain geographical areas with greater employee turnover rates. It aids in determining if the city or area has an effect on termination rates. A bar chart is plotted using geom_bar() from “ggplot” package to illustrate this correlation.

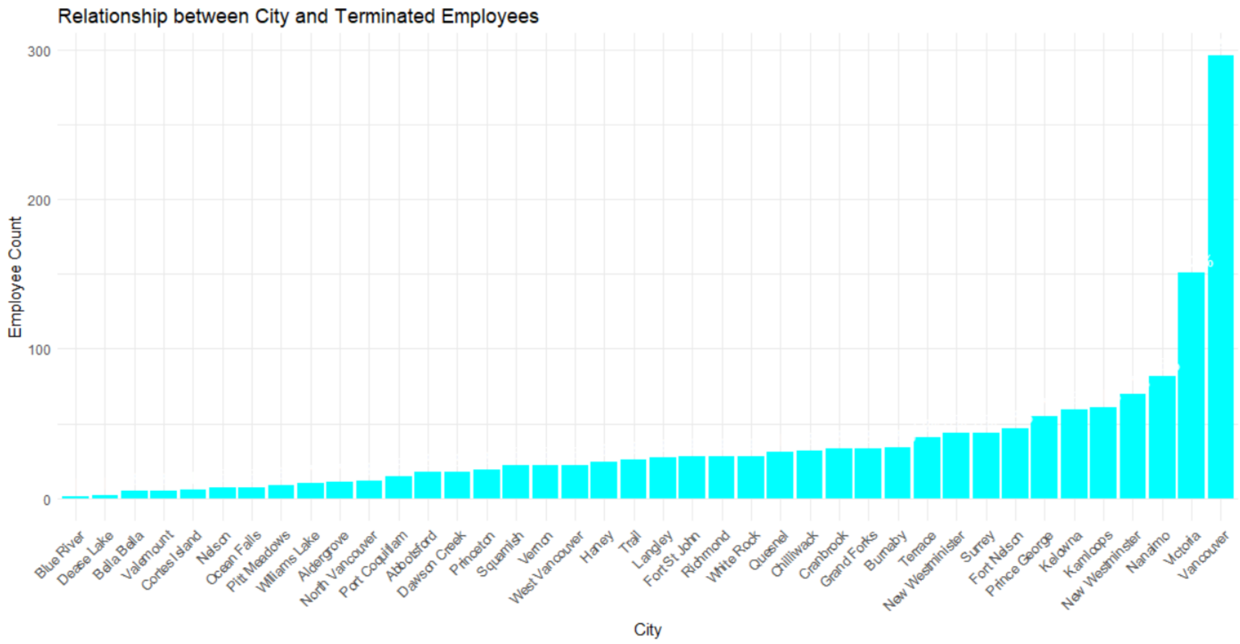
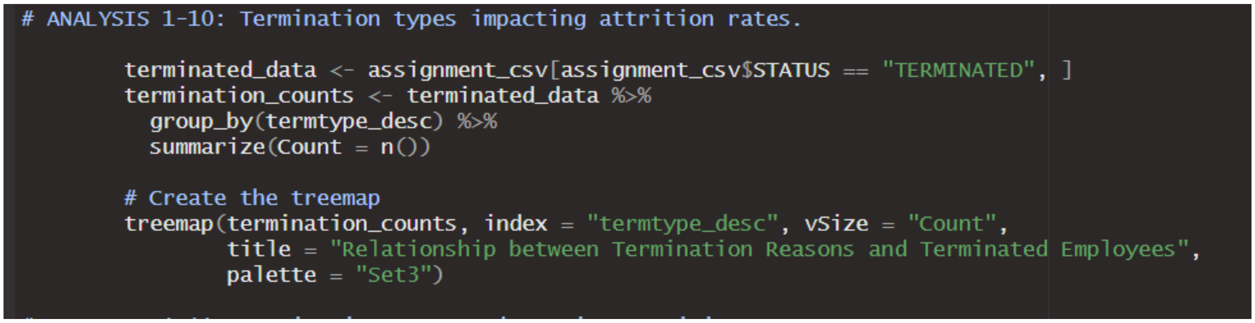


Figure 3.1.16: Relationship between city and employee count barchart

Based on the bar chart above, the Vancouver region has the highest termination rate among other cities. There are several factors that may contribute to the higher termination rate in the Vancouver compared to other branches across Canada, including geographic factors (e.g. location is too remote/busy) and economic factors (e.g. higher cost of living).

3.1.9 Analysis 1-9: Termination types impacting attrition rates.



```
# ANALYSIS 1-10: Termination types impacting attrition rates.

        terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]
        termination_counts <- terminated_data %>%
          group_by(termtype_desc) %>%
          summarize(Count = n())

        # Create the treemap
        treemap(termination_counts, index = "termtype_desc", vSize = "Count",
                title = "Relationship between Termination Reasons and Terminated Employees",
                palette = "Set3")
```

Figure 3.1.17: Proportions of termination types codes

Analysis 1-9 analyse termination type and their influence on attrition rates. Treemap is plotted to show the types of termination.

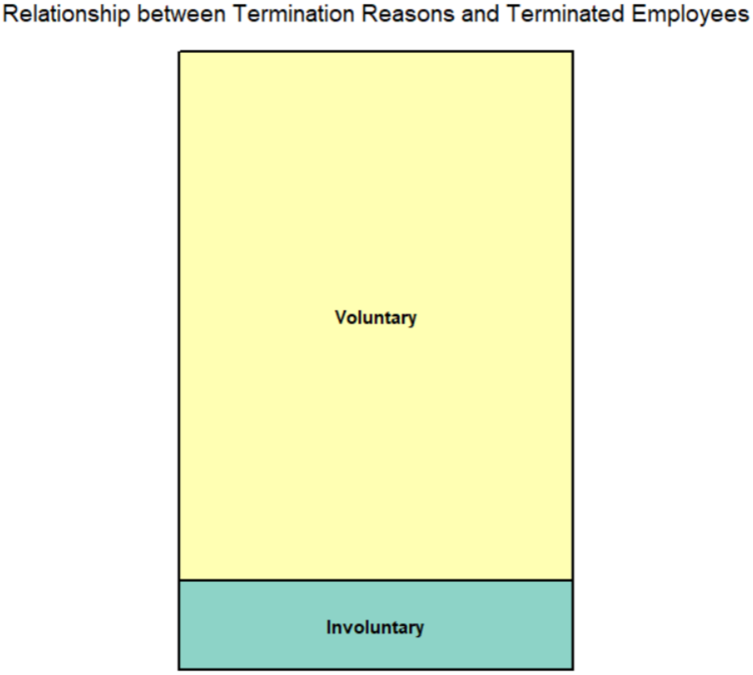
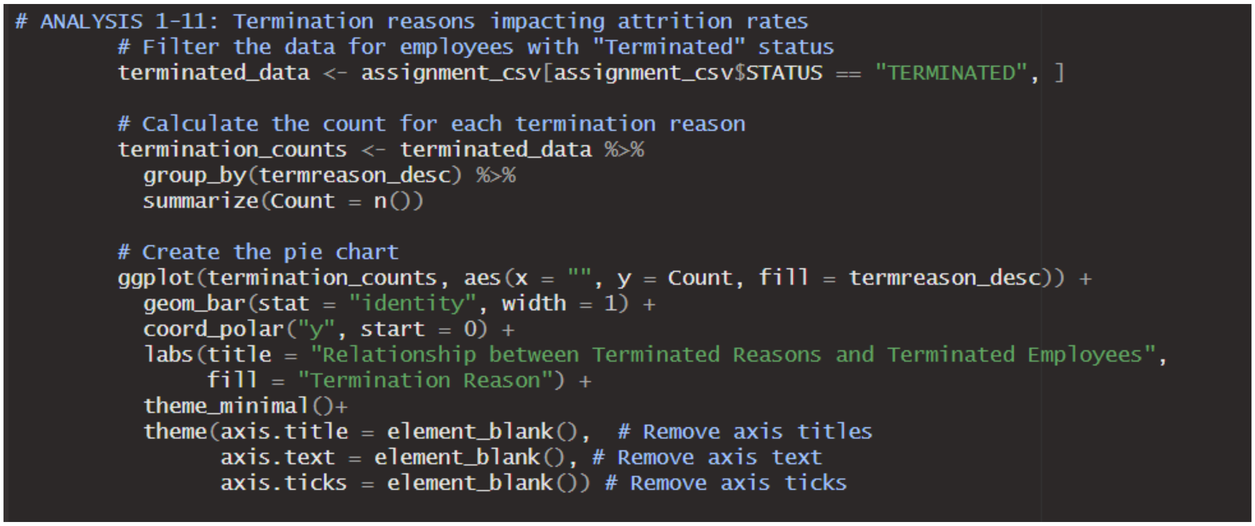


Figure 3.1.18: Proportions of termination types visual bar.

The treemap above indicates that the majority of the termination is Voluntary. Meaning the majority of employees leave the organisation due to personal factors (e.g. retirement, job shifting, family factors and many more).

3.1.10 Analysis 1-10: Termination reasons impacting attrition rates



```
# ANALYSIS 1-11: Termination reasons impacting attrition rates
        # Filter the data for employees with "Terminated" status
        terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]

        # Calculate the count for each termination reason
        termination_counts <- terminated_data %>%
          group_by(termreason_desc) %>%
          summarize(Count = n())

        # Create the pie chart
        ggplot(termination_counts, aes(x = "", y = Count, fill = termreason_desc)) +
          geom_bar(stat = "identity", width = 1) +
          coord_polar("y", start = 0) +
          labs(title = "Relationship between Terminated Reasons and Terminated Employees",
               fill = "Termination Reason") +
          theme_minimal()+
          theme(axis.title = element_blank(),  # Remove axis titles
                axis.text = element_blank(), # Remove axis text
                axis.ticks = element_blank()) # Remove axis ticks
```

Figure 3.1.19: Relationship between termination reason and attrition codes

Analysis 1-10 analyses termination causes and their influence on attrition rates in order to uncover distinct factors that contribute to employee terminations. To visualise the termination reasons of the employees, a pie chart is plotted using geom_bar() in the ggplot package. The function coord_polar("y", start = 0) is used to transform a bar chart into a pie chart with a circular layout.

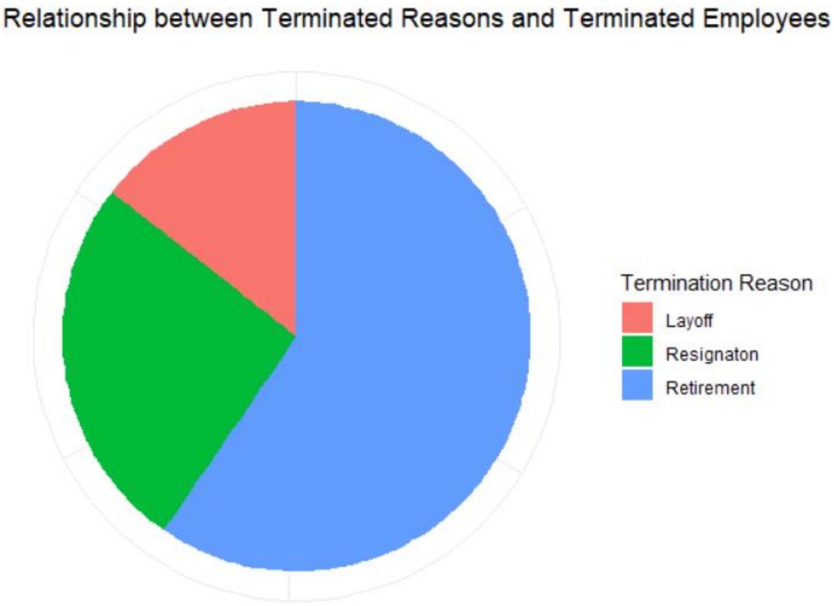
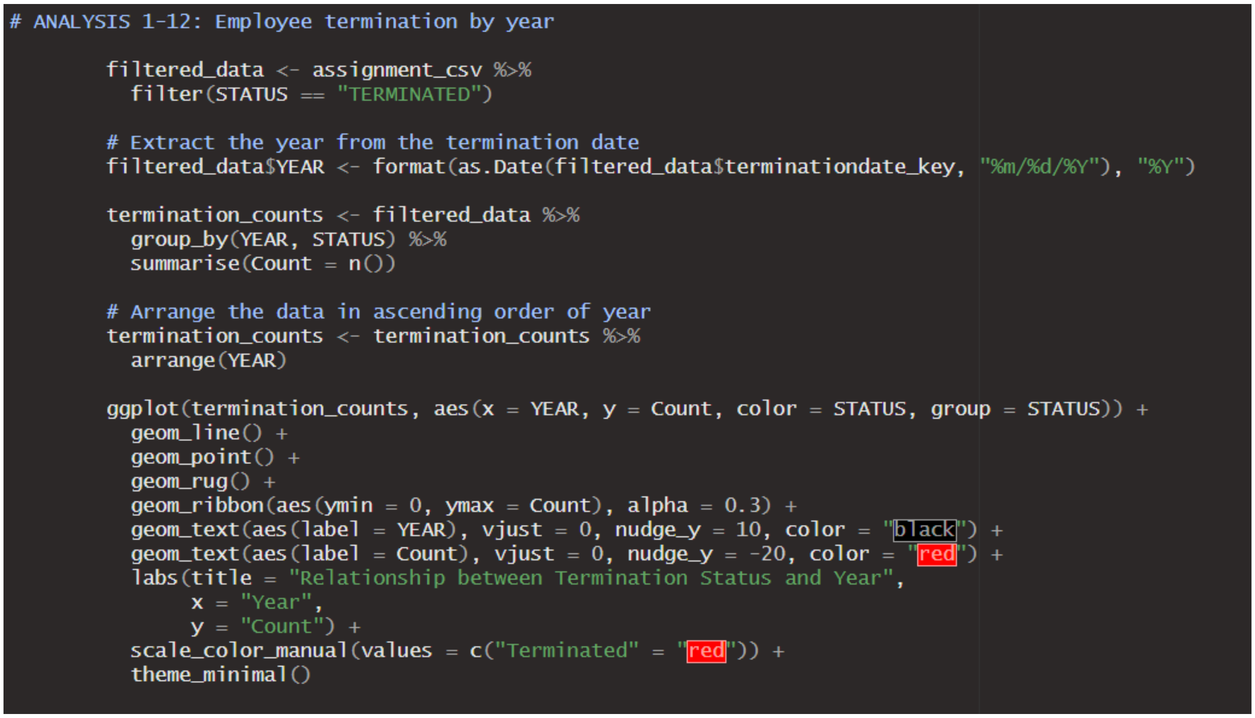


Figure 3.1.20: Proportions of termination reason pie chart

The pie chart above shows that the majority of termination reason of employees is due to retirement factor (e.g. ageing), followed by resignation (e.g. seek career growth) then Layoff (e.g. poor work performance).

3.1.11 Analysis 1-11: Employee termination by year



```
# ANALYSIS 1-12: Employee termination by year

        filtered_data <- assignment_csv %>%
          filter(STATUS == "TERMINATED")

        # Extract the year from the termination date
        filtered_data$YEAR <- format(as.Date(filtered_data$terminationdate_key, "%m/%d/%Y"), "%Y")

        termination_counts <- filtered_data %>%
          group_by(YEAR, STATUS) %>%
          summarise(Count = n())

        # Arrange the data in ascending order of year
        termination_counts <- termination_counts %>%
          arrange(YEAR)

        ggplot(termination_counts, aes(x = YEAR, y = Count, color = STATUS, group = STATUS)) +
          geom_line() +
          geom_point() +
          geom_rug() +
          geom_ribbon(aes(ymin = 0, ymax = Count), alpha = 0.3) +
          geom_text(aes(label = YEAR), vjust = 0, nudge_y = 10, color = "black") +
          geom_text(aes(label = Count), vjust = 0, nudge_y = -20, color = "red") +
          labs(title = "Relationship between Termination Status and Year",
               x = "Year",
               y = "Count") +
          scale_color_manual(values = c("Terminated" = "red")) +
          theme_minimal()
```

Figure 3.1.21: Employee termination by year codes

Analysis 1-11 intends to investigate employee terminations by year to offer a comprehensive perspective of termination trends and patterns across time. It obliges to determining whether there are any substantial changes in termination rates. A line graph is plotted using geom_line() in “ggplot” package to illustrate the trend.

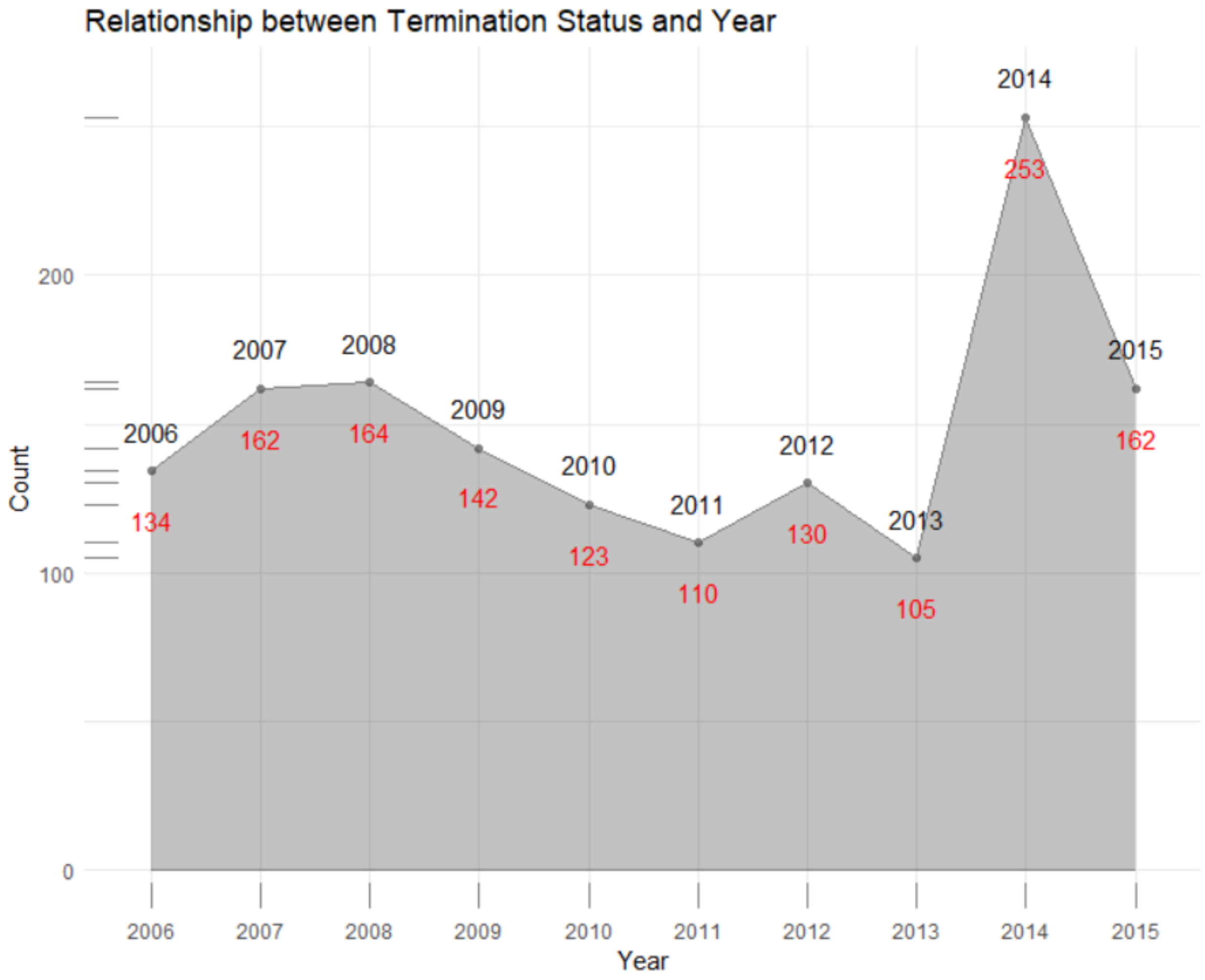
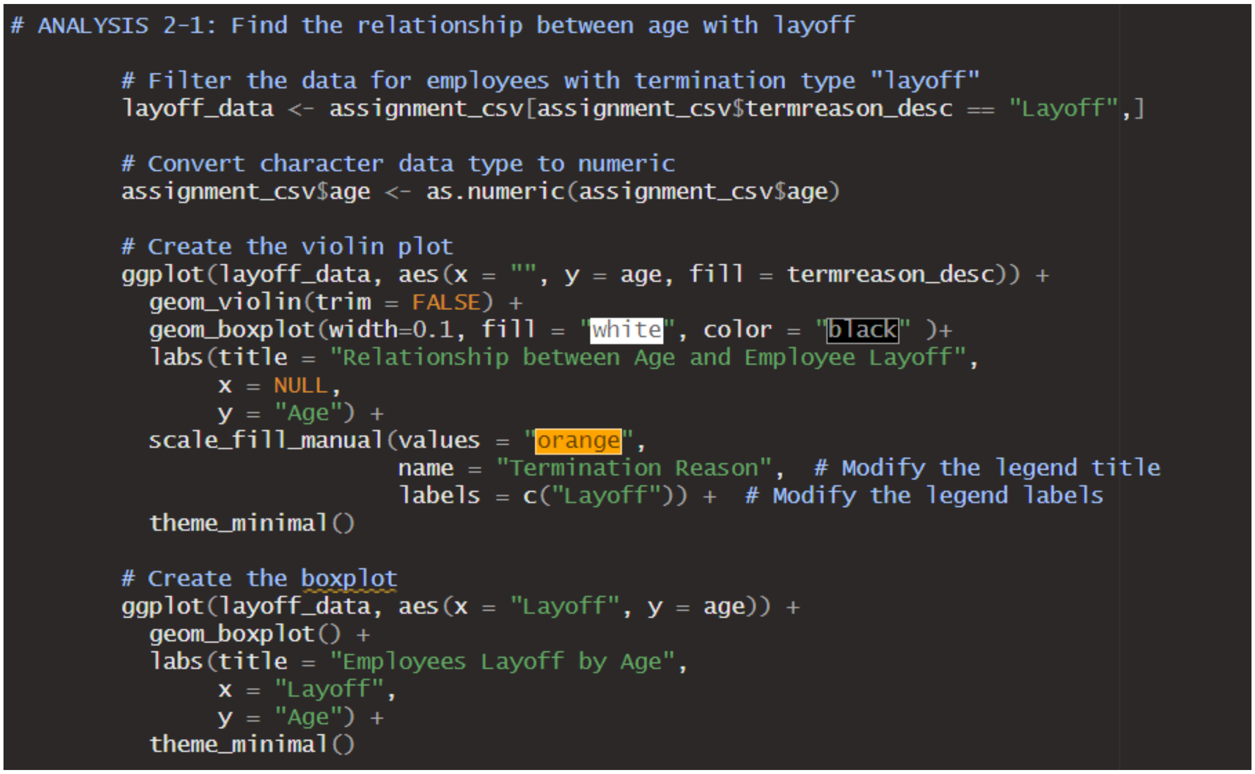


Figure 3.1.22: Employee termination by year graph

From the line graph above, the year 2013 has the least significant employee termination while the year 2014 has the most significant employee termination. There is a huge rise in employee termination within a year. Termination counts can vary over a year for a variety of reasons. For instance, changes in economic conditions, company policies, and digital transformation.

3.2 Question 2: What are the reasons for employee layoff?

3.2.1 Analysis 2-1: Find the relationship between age with employee layoff



```
# ANALYSIS 2-1: Find the relationship between age with layoff

        # Filter the data for employees with termination type "layoff"
        layoff_data <- assignment_csv[assignment_csv$termreason_desc == "Layoff",]

        # Convert character data type to numeric
        assignment_csv$age <- as.numeric(assignment_csv$age)

        # Create the violin plot
        ggplot(layoff_data, aes(x = "", y = age, fill = termreason_desc)) +
          geom_violin(trim = FALSE) +
          geom_boxplot(width=0.1, fill = "white", color = "black" )+
          labs(title = "Relationship between Age and Employee Layoff",
               x = NULL,
               y = "Age") +
          scale_fill_manual(values = "orange",
                            name = "Termination Reason",  # Modify the legend title
                            labels = c("Layoff")) +  # Modify the legend labels
          theme_minimal()

        # Create the boxplot
        ggplot(layoff_data, aes(x = "Layoff", y = age)) +
          geom_boxplot() +
          labs(title = "Employees Layoff by Age",
               x = "Layoff",
               y = "Age") +
          theme_minimal()
```

Figure 3.2.1: Relationship between age and employee layoff codes

Analysis 2-1 aims to discover the association between age and layoff to see whether there is a link between age groups and the chance of getting laid off. This will help determine whether age is a key issue in the layoff process. A violin plot is plotted using geom() in “ggplot” package to show this correlation. Geom_boxplot() is used to plot a boxplot graph within the violin graph.

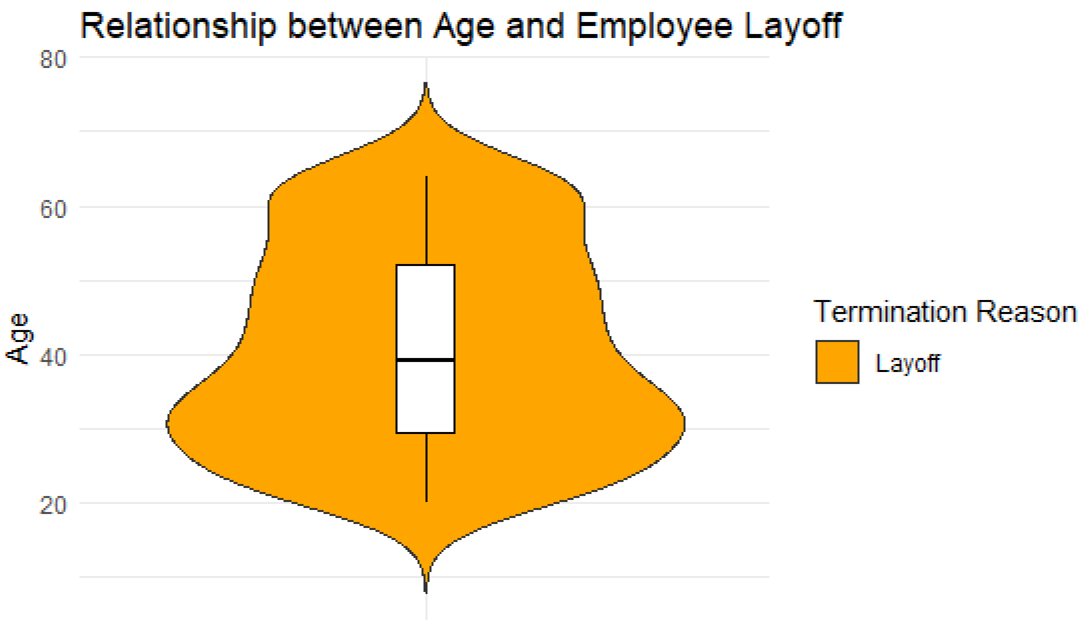
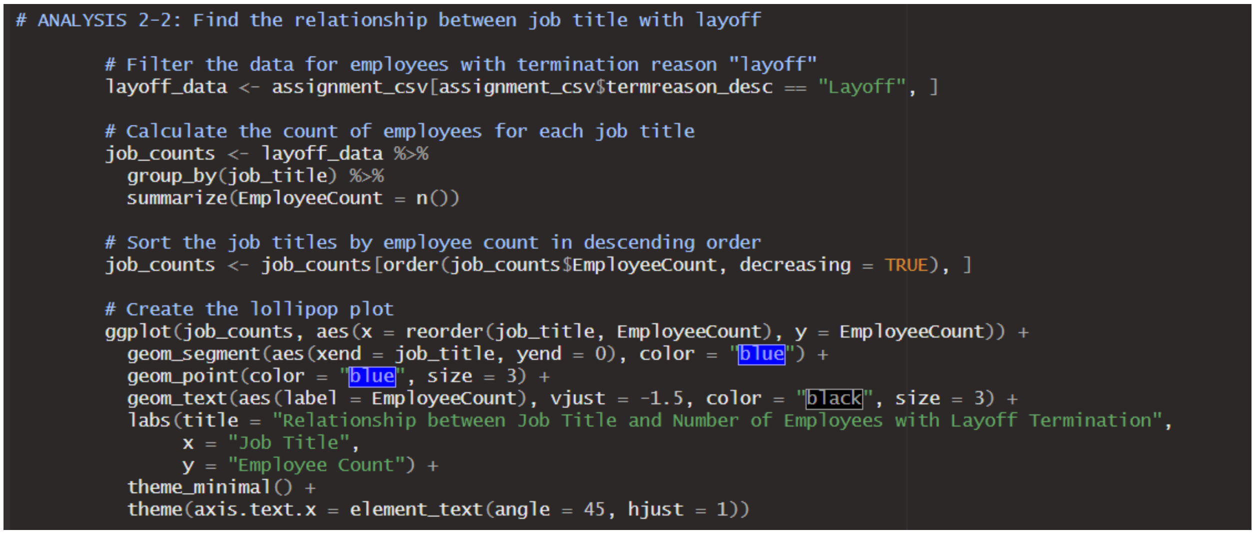


Figure 3.2.2: Relationship between age and employee layoff violin graph

The violin plot above shows that the wider sections between 20-40 are more likely to have high layoff rates for employees in this age range. Meanwhile, the skinnier sections below 20 and above 40 on the violin plot indicate low layoffs.

3.2.2 Analysis 2-2: Find the relationship between job title with employee layoff



```r
# ANALYSIS 2-2: Find the relationship between job title with layoff

        # Filter the data for employees with termination reason "layoff"
        layoff_data <- assignment_csv[assignment_csv$termreason_desc == "Layoff", ]

        # Calculate the count of employees for each job title
        job_counts <- layoff_data %>%
          group_by(job_title) %>%
          summarize(EmployeeCount = n())

        # Sort the job titles by employee count in descending order
        job_counts <- job_counts[order(job_counts$EmployeeCount, decreasing = TRUE), ]

        # Create the lollipop plot
        ggplot(job_counts, aes(x = reorder(job_title, EmployeeCount), y = EmployeeCount)) +
          geom_segment(aes(xend = job_title, yend = 0), color = "blue") +
          geom_point(color = "blue", size = 3) +
          geom_text(aes(label = EmployeeCount), vjust = -1.5, color = "black", size = 3) +
          labs(title = "Relationship between Job Title and Number of Employees with Layoff Termination",
               x = "Job Title",
               y = "Employee Count") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.2.3: Relationship between job title and employee layoff codes

Analysis 2-2 seeks to investigate the association between job titles and layoffs in order to determine if some occupations or roles are more vulnerable to layoffs. It assists in determining whether particular job titles are more likely to be laid off. A lollipop plot is plotted using geom_segment() and geom_point() in the “ggplot” package to visualise this relationship.

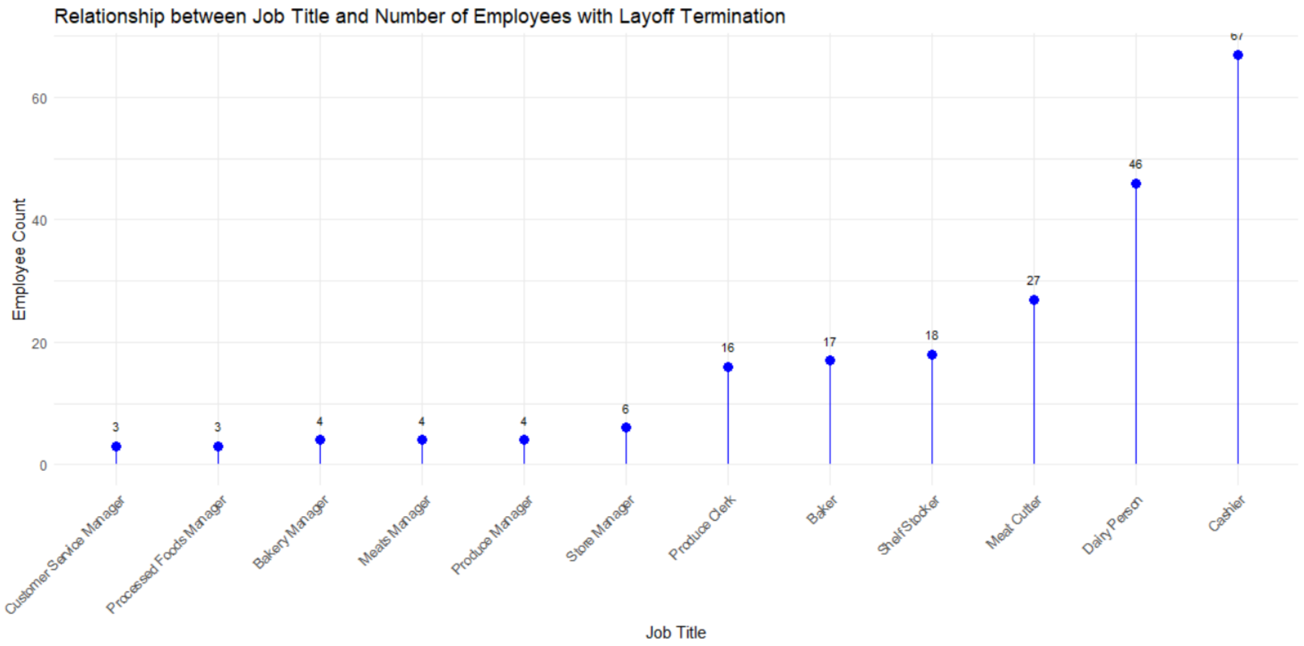
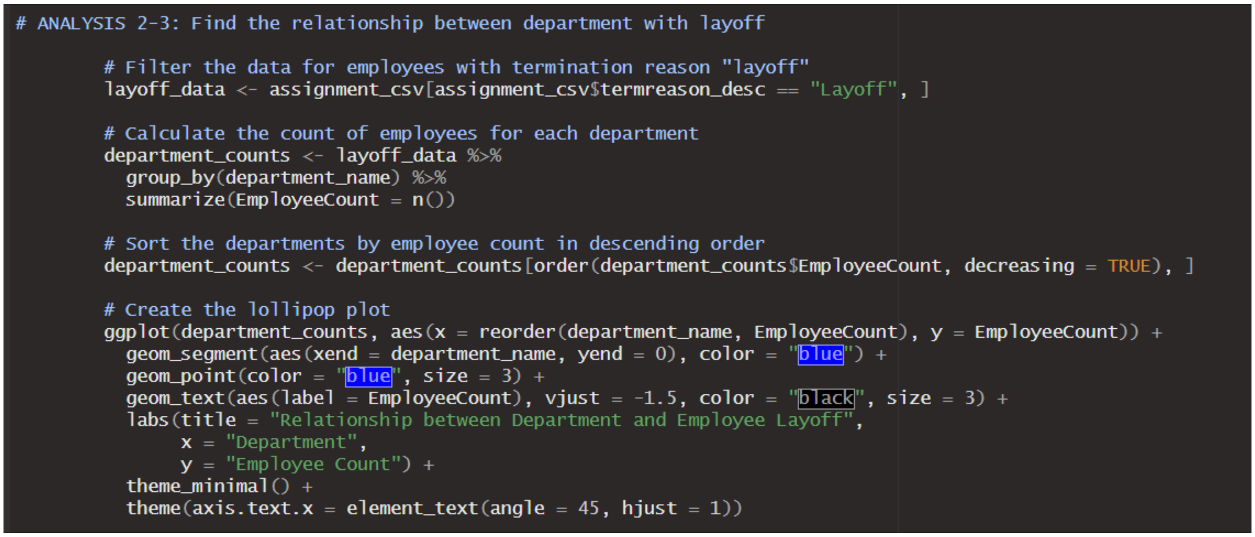


Figure 3.2.4: Relationship between job title and employee lollipop graph

Based on the lollipop plot, the cashier has the highest layoff count among other job titles. This could be due to poor employee ethics (e.g. caught stealing), poor working attitude (e.g. rude to customers), and poor work performance (e.g. slow).

3.2.3 Analysis 2-3: Find the relationship between the department with employee layoff



```
# ANALYSIS 2-3: Find the relationship between department with layoff

        # Filter the data for employees with termination reason "layoff"
        layoff_data <- assignment_csv[assignment_csv$termreason_desc == "Layoff", ]

        # Calculate the count of employees for each department
        department_counts <- layoff_data %>%
          group_by(department_name) %>%
          summarize(EmployeeCount = n())

        # Sort the departments by employee count in descending order
        department_counts <- department_counts[order(department_counts$EmployeeCount, decreasing = TRUE), ]

        # Create the lollipop plot
        ggplot(department_counts, aes(x = reorder(department_name, EmployeeCount), y = EmployeeCount)) +
          geom_segment(aes(xend = department_name, yend = 0), color = "blue") +
          geom_point(color = "blue", size = 3) +
          geom_text(aes(label = EmployeeCount), vjust = -1.5, color = "black", size = 3) +
          labs(title = "Relationship between Department and Employee Layoff",
               x = "Department",
               y = "Employee Count") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.2.5: Relationship between department and employee layoff codes

Analyses 2-3 attempt to investigate the association between departments and layoffs to determine whether there are certain departments that are more likely to experience layoffs. This assists in determining which sectors of the organisation are more vulnerable to layoffs. A lollipop plot is plotted using geom_segment() and geom_point() in the “ggplot” package to visualise this relationship.

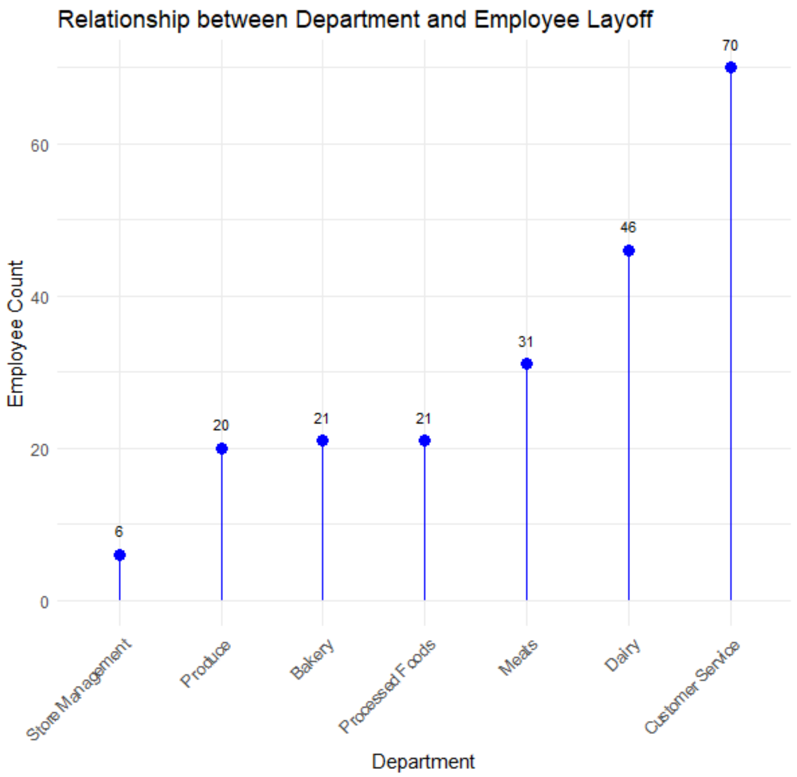
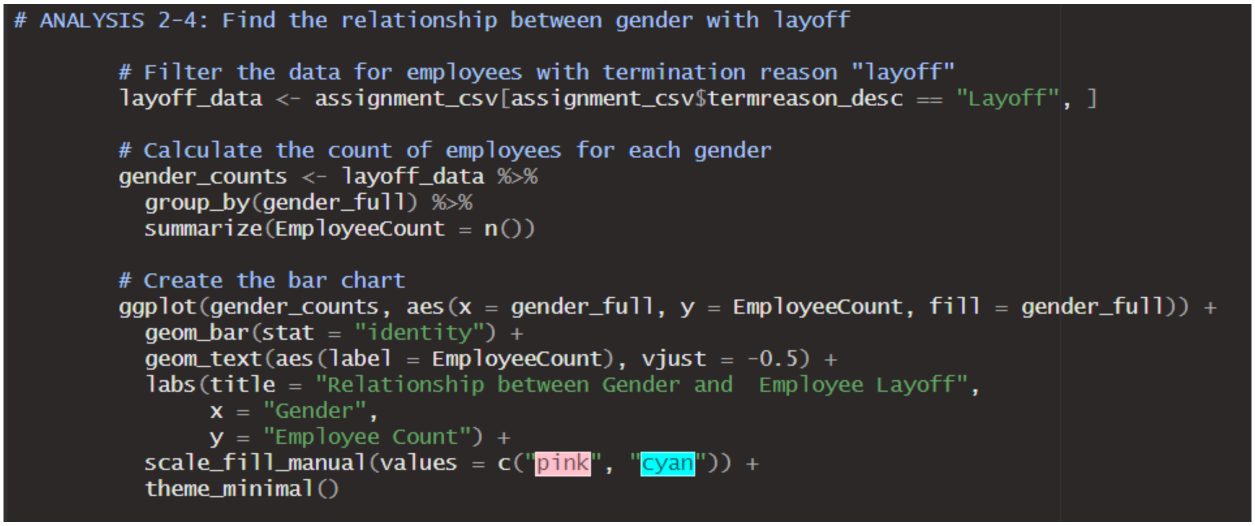


Figure 3.2.6 Relationship between department and employee layoff lollipop graph

Based on the lollipop plot, the Customer Service department has the highest layoff count among other departments. This could be majorly due to poor working attitude (e.g. rude to customers).

3.2.4 Analysis 2-4: Find the relationship between gender with employee layoff



```
# ANALYSIS 2-4: Find the relationship between gender with layoff

        # Filter the data for employees with termination reason "layoff"
        layoff_data <- assignment_csv[assignment_csv$termreason_desc == "Layoff", ]

        # Calculate the count of employees for each gender
        gender_counts <- layoff_data %>%
          group_by(gender_full) %>%
          summarize(EmployeeCount = n())

        # Create the bar chart
        ggplot(gender_counts, aes(x = gender_full, y = EmployeeCount, fill = gender_full)) +
          geom_bar(stat = "identity") +
          geom_text(aes(label = EmployeeCount), vjust = -0.5) +
          labs(title = "Relationship between Gender and  Employee Layoff",
               x = "Gender",
               y = "Employee Count") +
          scale_fill_manual(values = c("pink", "cyan")) +
          theme_minimal()
```

Figure 3.2.7: Relationship between gender and employee layoff codes

Analysis 2-4 seeks to investigate the association between gender and layoff in order to establish whether there are any gender-related factors in the chance of getting laid off. A bar graph is plotted using geom_bar() in “ggplot” package to illustrate this relationship.

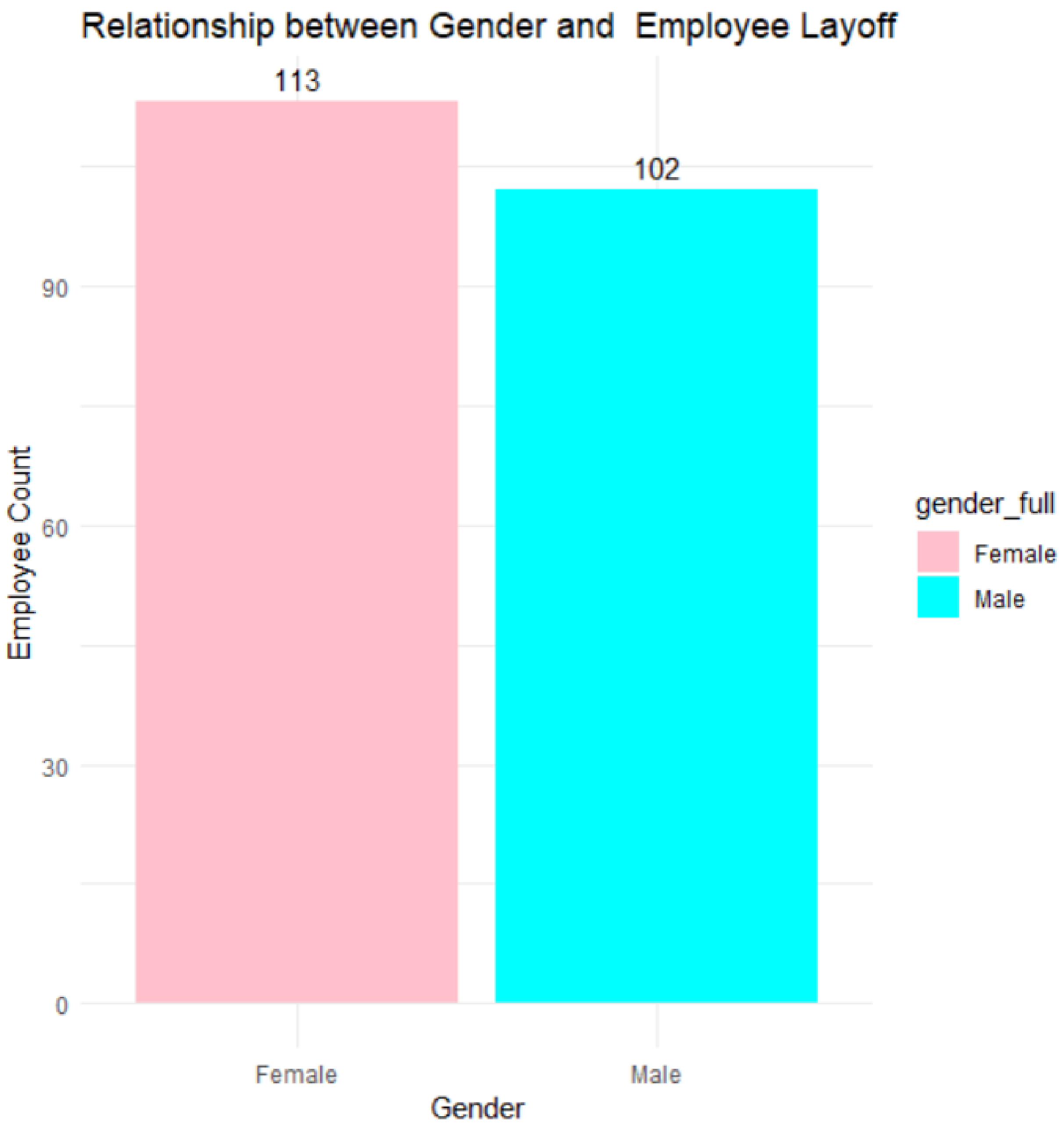
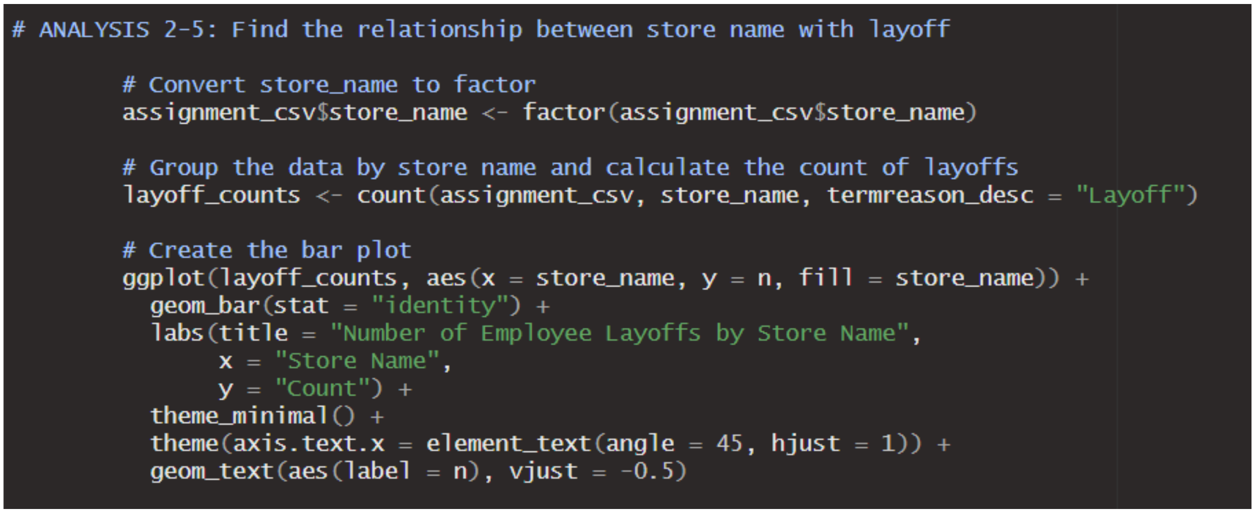


Figure 3.2.8: Relationship between gender and employee layoff bar graph.

Based on the bar chart above, males and females have close layoff counts. However, females still have a higher employee count that gets laid off than males. To be precise, women lose 11 more employees than men. An employer may want to terminate the female employee due to factors such as maternity leaves, pregnancy, and others.

3.2.5 Analysis 2-5: Find the relationship between store name with layoff



```
# ANALYSIS 2-5: Find the relationship between store name with layoff

        # Convert store_name to factor
        assignment_csv$store_name <- factor(assignment_csv$store_name)

        # Group the data by store name and calculate the count of layoffs
        layoff_counts <- count(assignment_csv, store_name, termreason_desc = "Layoff")

        # Create the bar plot
        ggplot(layoff_counts, aes(x = store_name, y = n, fill = store_name)) +
          geom_bar(stat = "identity") +
          labs(title = "Number of Employee Layoffs by Store Name",
               x = "Store Name",
               y = "Count") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
          geom_text(aes(label = n), vjust = -0.5)
```

Figure 3.2.9: Relationship between store name and employee layoff codes.

Analysis 2-5 seeks to investigate the association between store names and layoffs in order to determine whether there are certain stores that are more prone to layoffs. A bar graph is plotted using geom_bar() in “ggplot” package to illustrate this relationship.

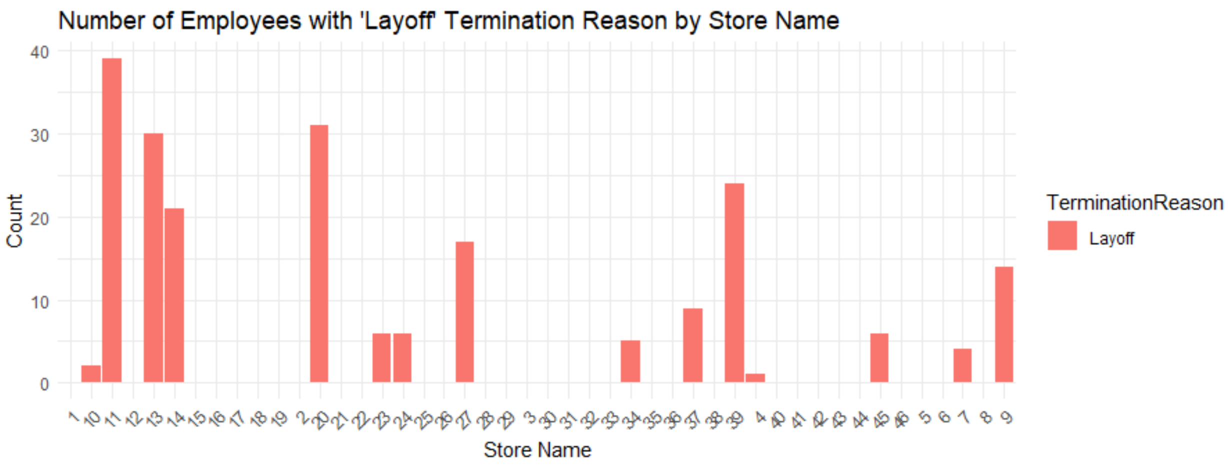
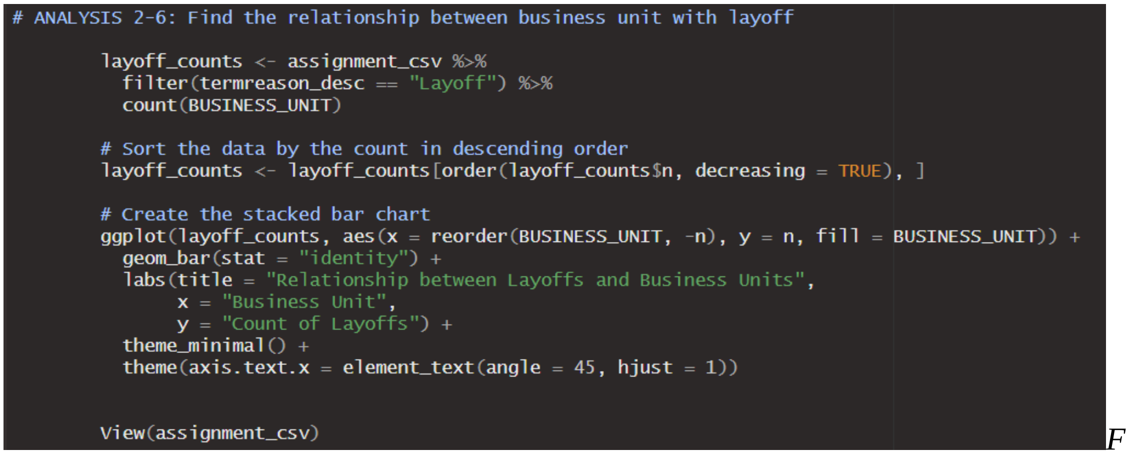


Figure 3.2.10: Relationship between store name and employee layoff bar graph.

From the bar chart above, certain stores have some and more employee layoffs while certain stores do not have any employee layoffs. This can occur due to factors like operational costs (e.g. certain branches' customer volume is smaller so fewer employees are needed) and performance and productivity (e.g. many employees are underperforming).

3.2.6 Analysis 2-6: Find the relationship between the business unit with employee layoff



```
# ANALYSIS 2-6: Find the relationship between business unit with layoff

        layoff_counts <- assignment_csv %>%
          filter(termreason_desc == "Layoff") %>%
          count(BUSINESS_UNIT)

        # Sort the data by the count in descending order
        layoff_counts <- layoff_counts[order(layoff_counts$n, decreasing = TRUE), ]

        # Create the stacked bar chart
        ggplot(layoff_counts, aes(x = reorder(BUSINESS_UNIT, -n), y = n, fill = BUSINESS_UNIT)) +
          geom_bar(stat = "identity") +
          labs(title = "Relationship between Layoffs and Business Units",
               x = "Business Unit",
               y = "Count of Layoffs") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))


        View(assignment_csv)
```

Figure 3.2.11: Relationship between the business unit and employee layoff codes.

Analysis 2-6 seek to examine the association between business units and layoffs in order to determine whether some business units or divisions are more sensitive to layoffs. A bar graph is plotted using geom_bar() in “ggplot” package to illustrate this relationship.

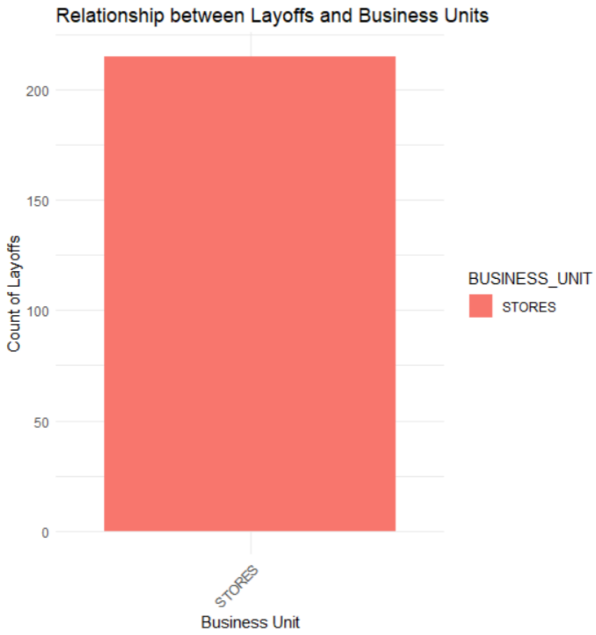
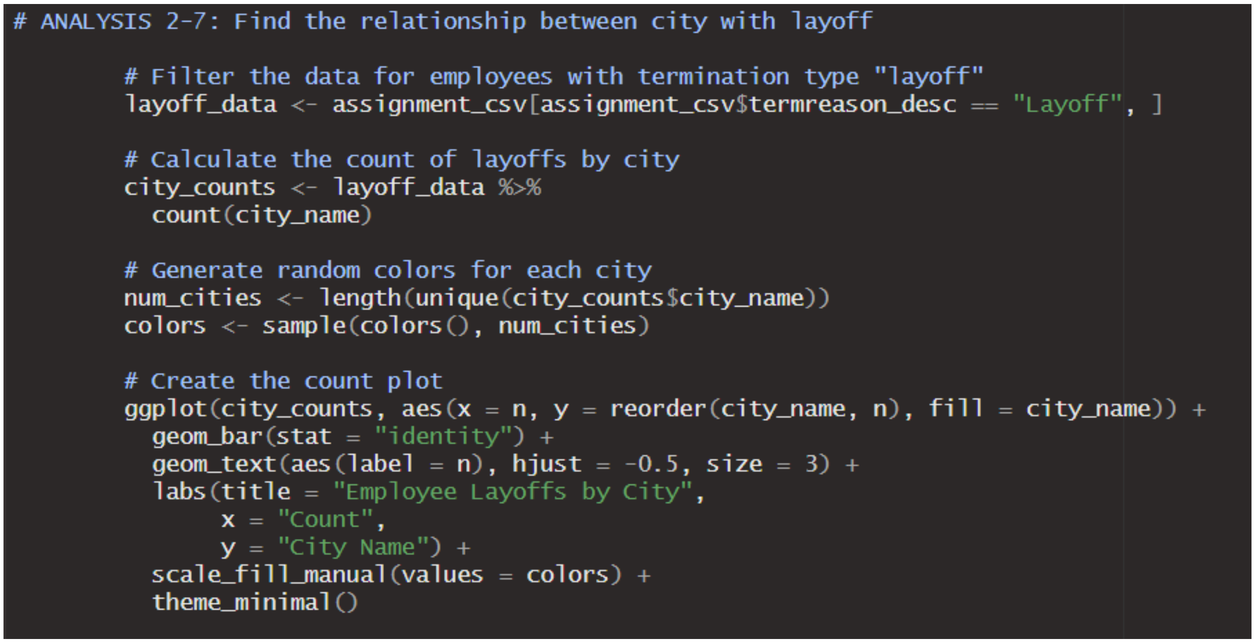


Figure 3.2.12: Relationship between the business unit and employee bar graph.

From the bar graph above, all employee layoffs occur in store branches only and none for head office. This implies that the employee is well-trained in the head office.

3.2.7 Analysis 2-7: Find the relationship between the city with employee layoff



```
# ANALYSIS 2-7: Find the relationship between city with layoff

        # Filter the data for employees with termination type "layoff"
        layoff_data <- assignment_csv[assignment_csv$termreason_desc == "Layoff", ]

        # Calculate the count of layoffs by city
        city_counts <- layoff_data %>%
          count(city_name)

        # Generate random colors for each city
        num_cities <- length(unique(city_counts$city_name))
        colors <- sample(colors(), num_cities)

        # Create the count plot
        ggplot(city_counts, aes(x = n, y = reorder(city_name, n), fill = city_name)) +
          geom_bar(stat = "identity") +
          geom_text(aes(label = n), hjust = -0.5, size = 3) +
          labs(title = "Employee Layoffs by City",
               x = "Count",
               y = "City Name") +
          scale_fill_manual(values = colors) +
          theme_minimal()
```

Figure 3.2.13: Relationship between city and employee layoff codes.

Analysis 2-7 seeks to investigate the association between cities and layoffs in order to establish whether there are distinct geographic areas with a higher chance of layoffs. A bar graph is plotted using geom_bar() in “ggplot” package to illustrate this relationship.

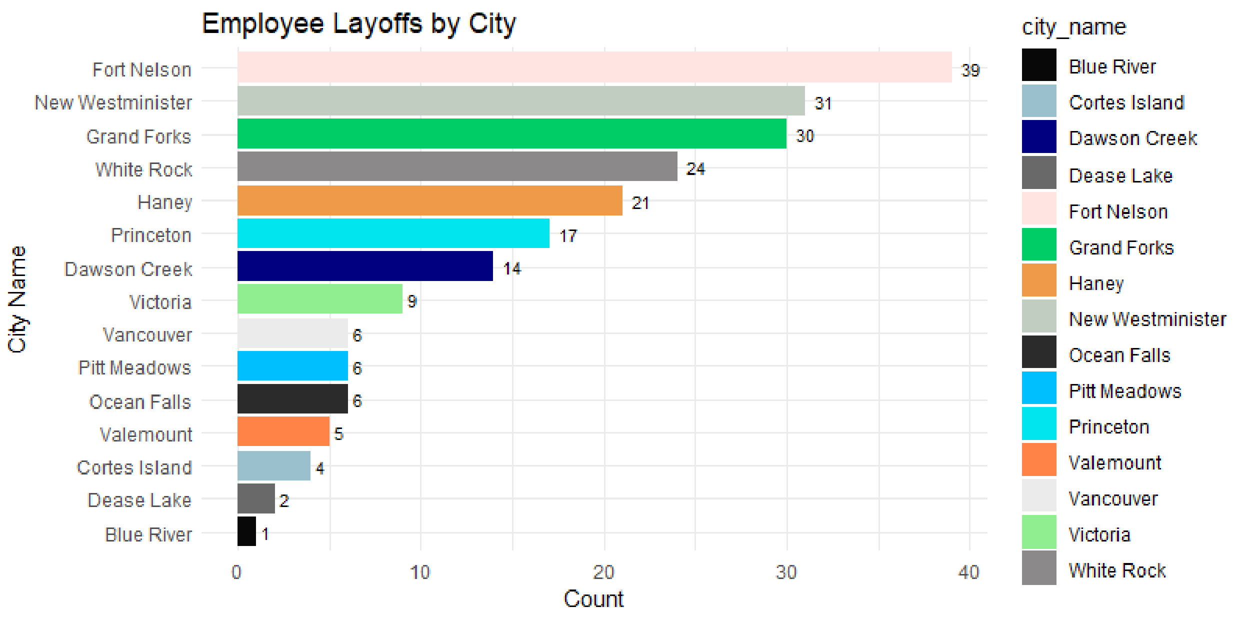
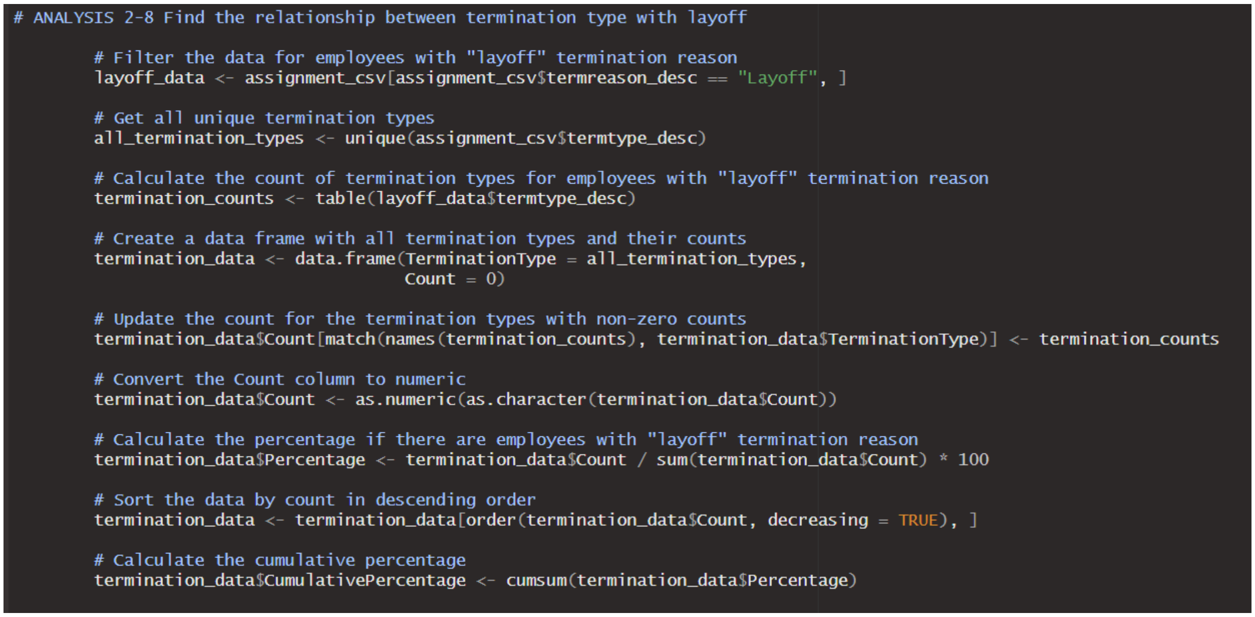


Figure 3.2.14: Relationship between city and employee layoff bar graph.

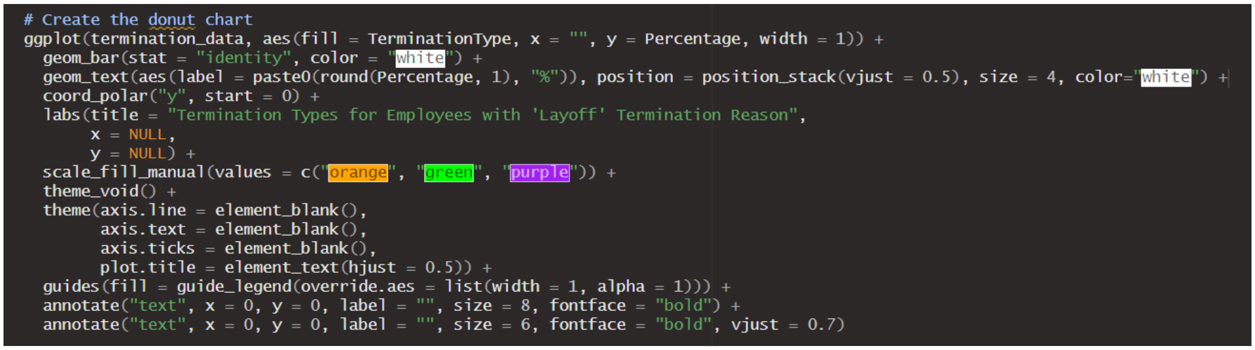
The results of the bar graph above show the relationship between termination type and employee layoff where the city of Fort Nelson has the highest number of employees getting laid off compared to the other 15 cities. Some cities have high employee layoffs while some cities have low employee layoffs. The economy could be a factor in why certain cities have higher employee layoffs. This occurs possibly due to an economic downturn in that region whereas some region is not affected.

3.2.8 Analysis 2-8: Find the relationship between termination type with employee layoff



```
# ANALYSIS 2-8 Find the relationship between termination type with layoff

        # Filter the data for employees with "layoff" termination reason
        layoff_data <- assignment_csv[assignment_csv$termreason_desc == "Layoff", ]

        # Get all unique termination types
        all_termination_types <- unique(assignment_csv$termtype_desc)

        # Calculate the count of termination types for employees with "layoff" termination reason
        termination_counts <- table(layoff_data$termtype_desc)

        # Create a data frame with all termination types and their counts
        termination_data <- data.frame(TerminationType = all_termination_types,
                                       Count = 0)

        # Update the count for the termination types with non-zero counts
        termination_data$Count[match(names(termination_counts), termination_data$TerminationType)] <- termination_counts

        # Convert the Count column to numeric
        termination_data$Count <- as.numeric(as.character(termination_data$Count))

        # Calculate the percentage if there are employees with "layoff" termination reason
        termination_data$Percentage <- termination_data$Count / sum(termination_data$Count) * 100

        # Sort the data by count in descending order
        termination_data <- termination_data[order(termination_data$Count, decreasing = TRUE), ]

        # Calculate the cumulative percentage
        termination_data$CumulativePercentage <- cumsum(termination_data$Percentage)
```

Figure 3.2.15: Relationship between termination type and employee layoff codes.



```
 # Create the donut chart
 ggplot(termination_data, aes(fill = TerminationType, x = "", y = Percentage, width = 1)) +
   geom_bar(stat = "identity", color = "white") +
   geom_text(aes(label = paste0(round(Percentage, 1), "%")), position = position_stack(vjust = 0.5), size = 4, color="white") +
   coord_polar("y", start = 0) +
   labs(title = "Termination Types for Employees with 'Layoff' Termination Reason",
        x = NULL,
        y = NULL) +
   scale_fill_manual(values = c("orange", "green", "purple")) +
   theme_void() +
   theme(axis.line = element_blank(),
         axis.text = element_blank(),
         axis.ticks = element_blank(),
         plot.title = element_text(hjust = 0.5)) +
   guides(fill = guide_legend(override.aes = list(width = 1, alpha = 1))) +
   annotate("text", x = 0, y = 0, label = "", size = 8, fontface = "bold") +
   annotate("text", x = 0, y = 0, label = "", size = 6, fontface = "bold", vjust = 0.7)
```

Figure 3.2.16: Relationship between termination type and employee layoff donut chart codes.

Analysis 2-7 seek to investigate the association between termination kinds and layoffs in order to find the most common reasons for layoffs. A donut chart is plotted using geom_bar() in “ggplot” package to illustrate this correlation. The function coord_polar("y", start = 0) is used to transform a bar chart into a donut chart with a circular layout.

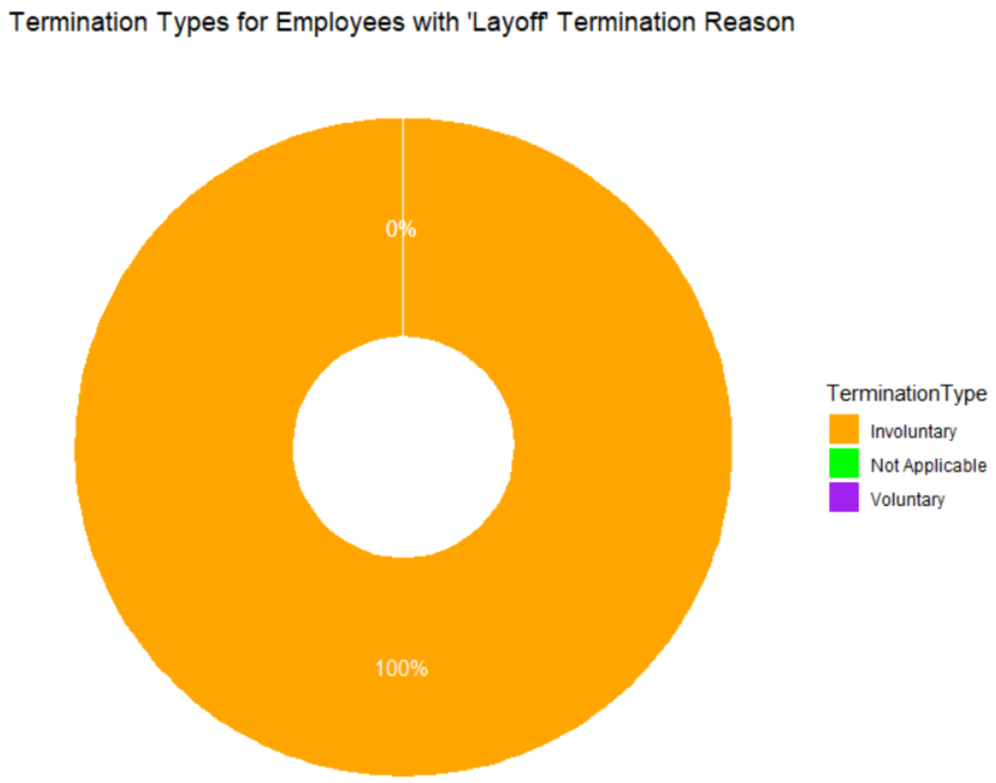
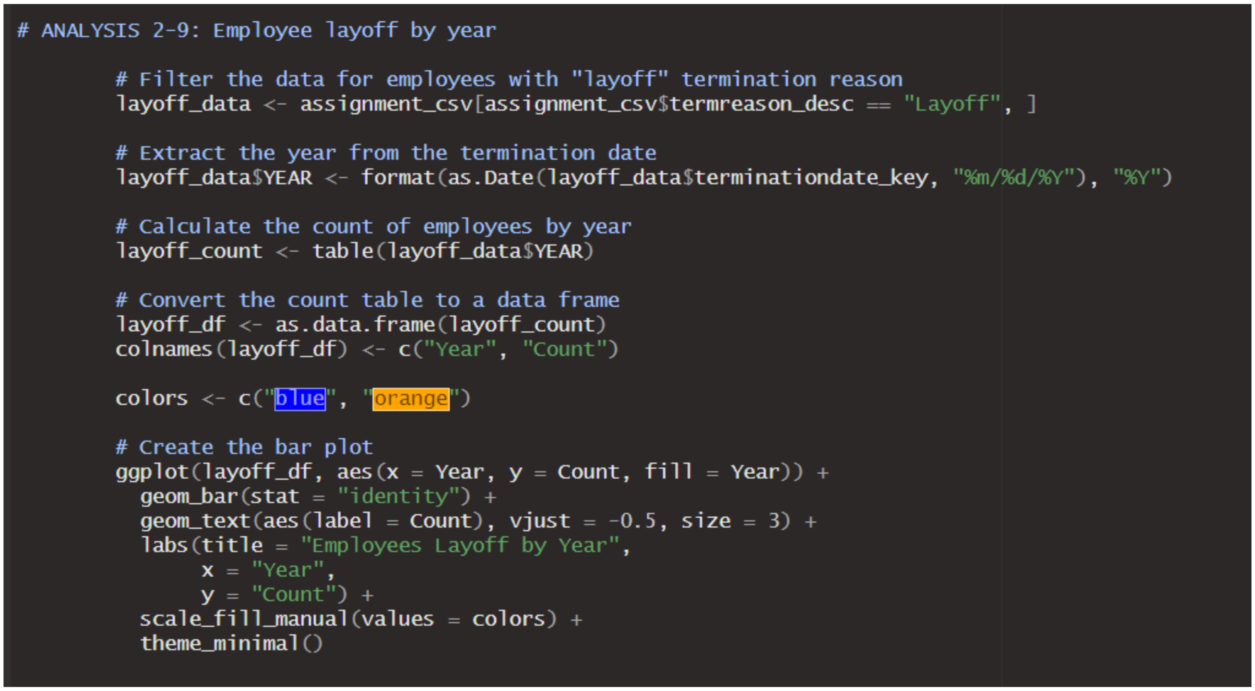


Figure 3.2.17: Relationship between termination type and employee layoff donut chart.

In the donut chart above, it shows that all the employees that got laid off were involuntarily terminated.

3.2.9 Analysis 2-9: Employee layoff by year



```
# ANALYSIS 2-9: Employee layoff by year

        # Filter the data for employees with "layoff" termination reason
        layoff_data <- assignment_csv[assignment_csv$termreason_desc == "Layoff", ]

        # Extract the year from the termination date
        layoff_data$YEAR <- format(as.Date(layoff_data$terminationdate_key, "%m/%d/%Y"), "%Y")

        # Calculate the count of employees by year
        layoff_count <- table(layoff_data$YEAR)

        # Convert the count table to a data frame
        layoff_df <- as.data.frame(layoff_count)
        colnames(layoff_df) <- c("Year", "Count")

        colors <- c("blue", "orange")

        # Create the bar plot
        ggplot(layoff_df, aes(x = Year, y = Count, fill = Year)) +
          geom_bar(stat = "identity") +
          geom_text(aes(label = Count), vjust = -0.5, size = 3) +
          labs(title = "Employees Layoff by Year",
               x = "Year",
               y = "Count") +
          scale_fill_manual(values = colors) +
          theme_minimal()
```

Figure 3.2.18: Employee layoff by year codes

Analysis 2-9 aims to analyse staff layoffs by year, providing a snapshot of layoff trends and patterns across time. A bar graph is plotted using geom_bar() in “ggplot” package to illustrate this relationship.

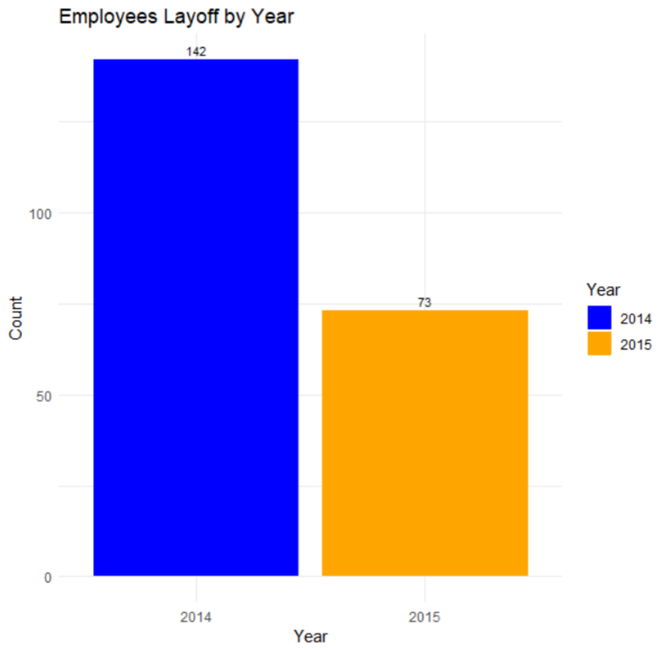
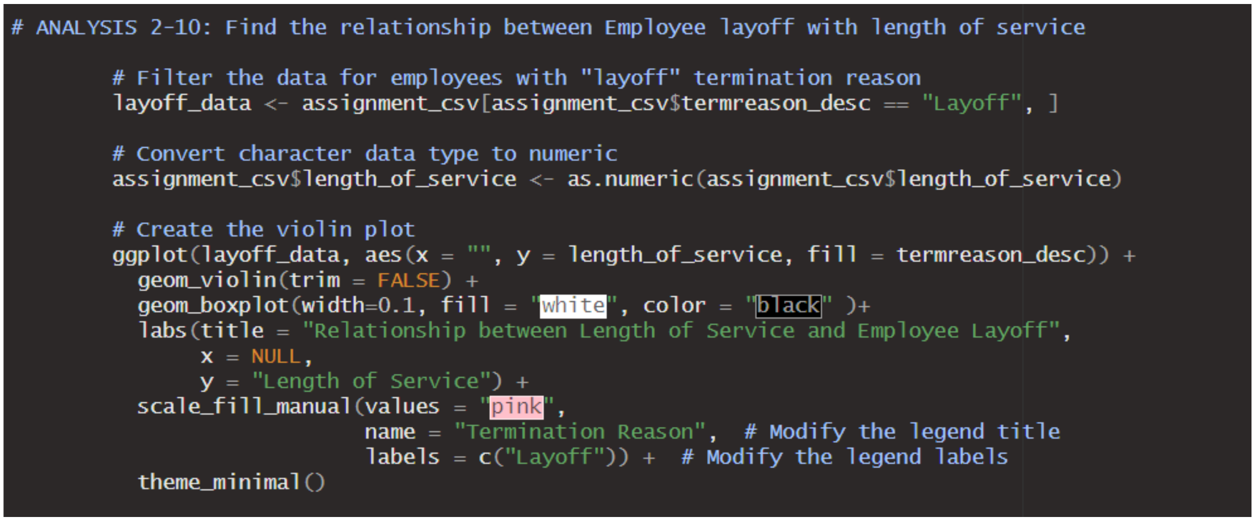


Figure 3.2.19: Employee layoff by year bar graph

The bar chart above shows that there were more layoffs in the year 2014 compared to the year 2015. To be exact, there were 69 more terminations in the year 2014 compared to the year 2015. This could be a result of the company re-evaluating employees' performance to determine which employee to stay or leave.

3.2.10 Analysis2-10: Find the relationship between Employee layoff with the length of service



```
# ANALYSIS 2-10: Find the relationship between Employee layoff with length of service

        # Filter the data for employees with "layoff" termination reason
        layoff_data <- assignment_csv[assignment_csv$termreason_desc == "Layoff", ]

        # Convert character data type to numeric
        assignment_csv$length_of_service <- as.numeric(assignment_csv$length_of_service)

        # Create the violin plot
        ggplot(layoff_data, aes(x = "", y = length_of_service, fill = termreason_desc)) +
          geom_violin(trim = FALSE) +
          geom_boxplot(width=0.1, fill = "white", color = "black" )+
          labs(title = "Relationship between Length of Service and Employee Layoff",
               x = NULL,
               y = "Length of Service") +
          scale_fill_manual(values = "pink",
                            name = "Termination Reason",  # Modify the legend title
                            labels = c("Layoff")) +  # Modify the legend labels
          theme_minimal()
```

Figure 3.2.20: Relationship between employee layoff and length of service codes

Analysis 2-10 seek to investigate the relationship between employee layoff and length of service in order to determine whether there is a link between the length of employment and the chance of being laid off. A violin plot is plotted using geom() in “ggplot” package to show this correlation. Geom_boxplot() is used to plot a boxplot graph within the violin graph.

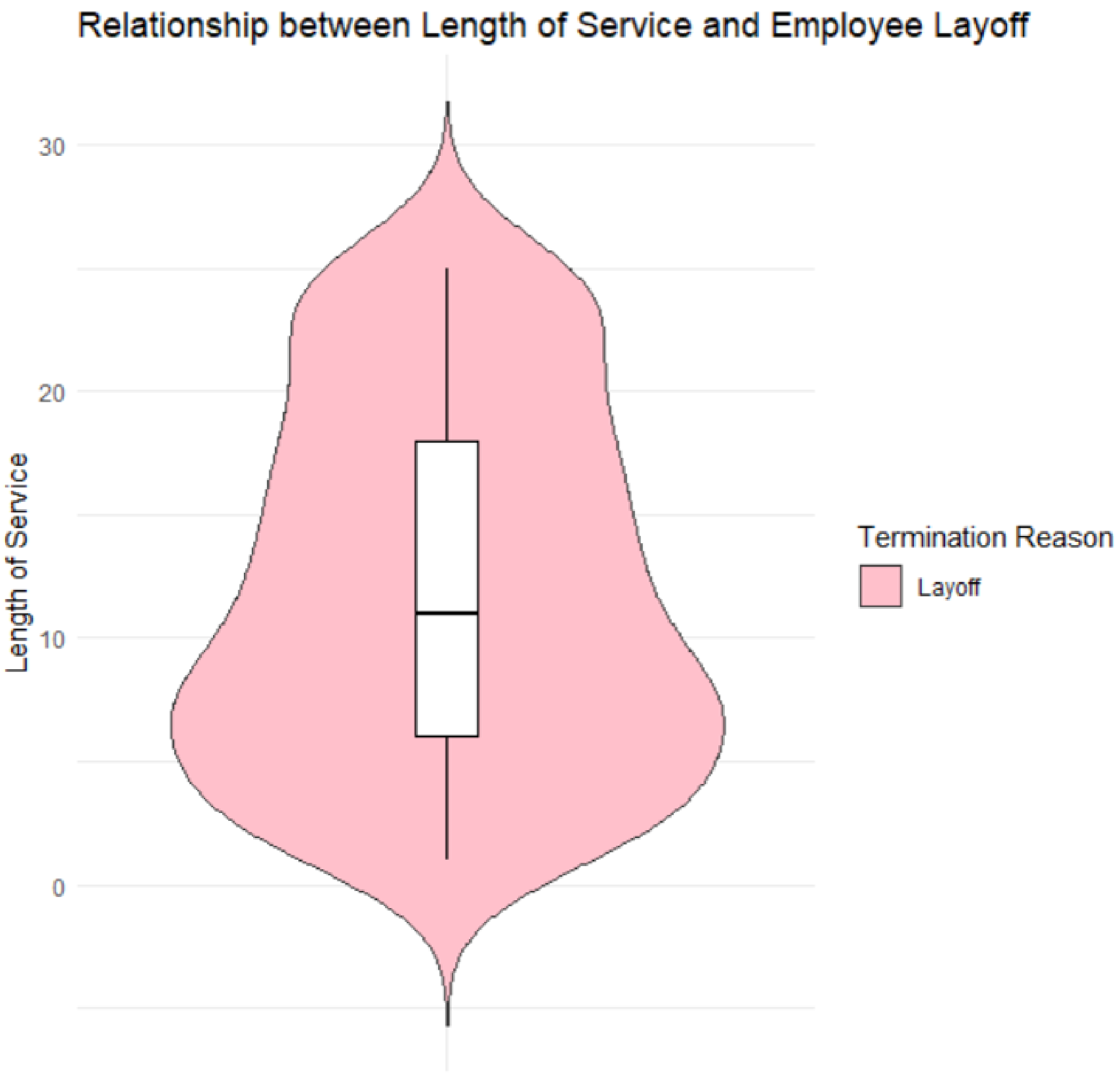
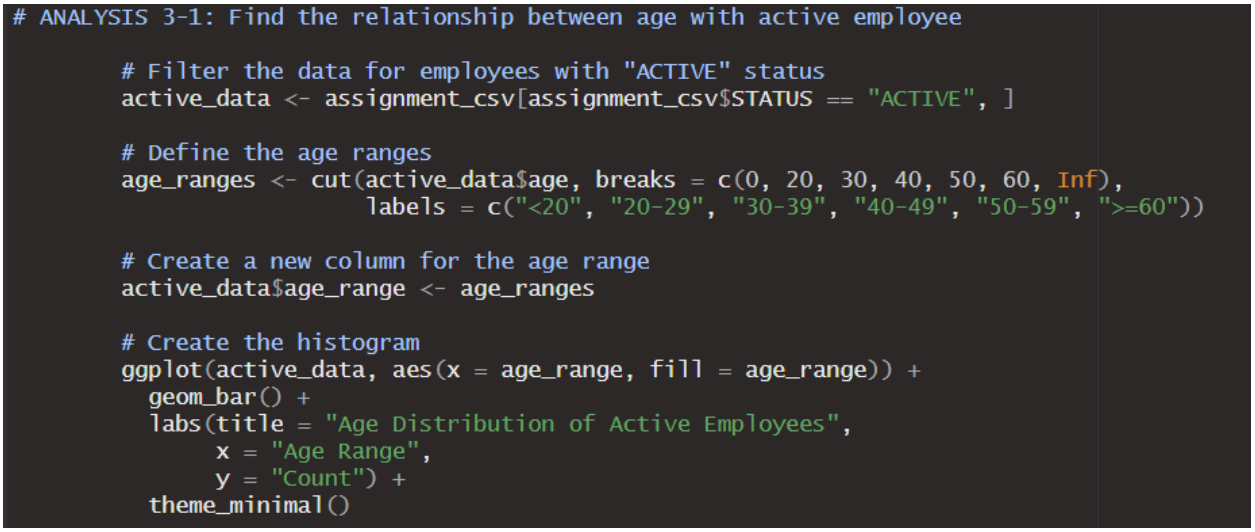


Figure 3.2.21: Relationship between employee layoff and length of service violin chart

The violin chart above shows wider sections between 0-10 years of length of service are more likely to have high layoff rates for employees in this age range. The skinnier sections greater than 10 years of length of service on the violin plot indicate low layoffs. The reason for the high probability of employee layoffs within this length of service range (0-10) is due to employees are inexperienced and the company would have to train them which increases the costs.

3.3 Question 3: Why do employees stay with the organization?

3.3.1 Analysis 3-1: Find the relationship between age with active employee



```
# ANALYSIS 3-1: Find the relationship between age with active employee

    # Filter the data for employees with "ACTIVE" status
    active_data <- assignment_csv[assignment_csv$STATUS == "ACTIVE", ]

    # Define the age ranges
    age_ranges <- cut(active_data$age, breaks = c(0, 20, 30, 40, 50, 60, Inf),
             labels = c("<20", "20-29", "30-39", "40-49", "50-59", ">=60"))

    # Create a new column for the age range
    active_data$age_range <- age_ranges

    # Create the histogram
    ggplot(active_data, aes(x = age_range, fill = age_range)) +
     geom_bar() +
     labs(title = "Age Distribution of Active Employees",
        x = "Age Range",
        y = "Count") +
     theme_minimal()
```

Figure 3.3.1: Relationship between age and active employee codes

Analysis 3-1 seeks to investigate the relationship between age and active workers in order to determine whether there is a link between age groups and the chance of employees remaining with the organisation. A bar graph is plotted using geom_bar() in “ggplot” package to illustrate this relationship.

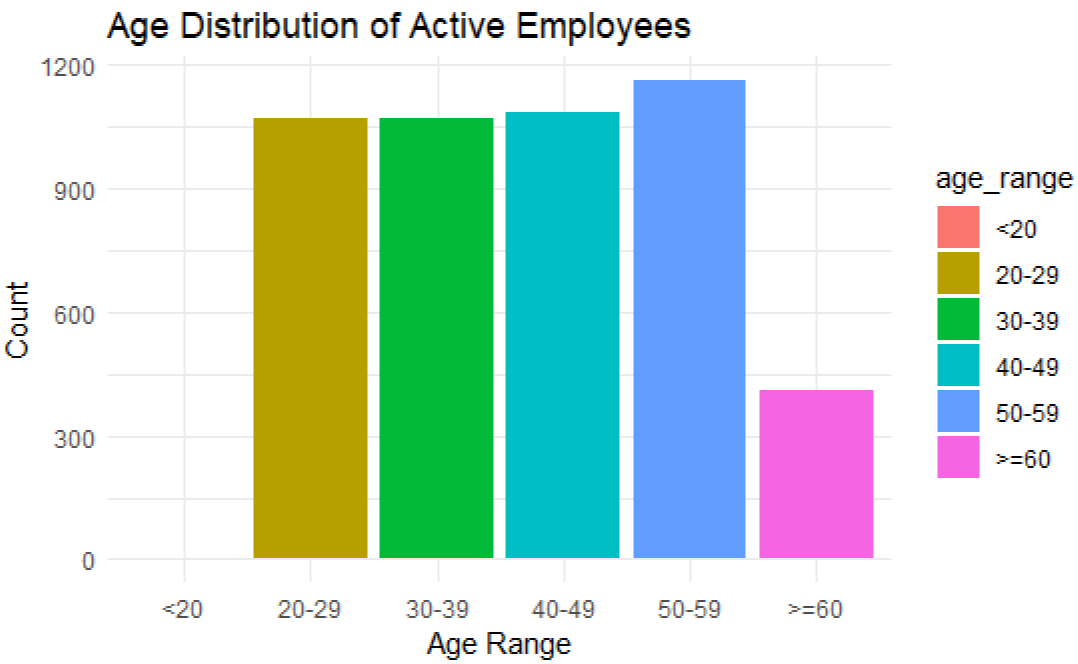
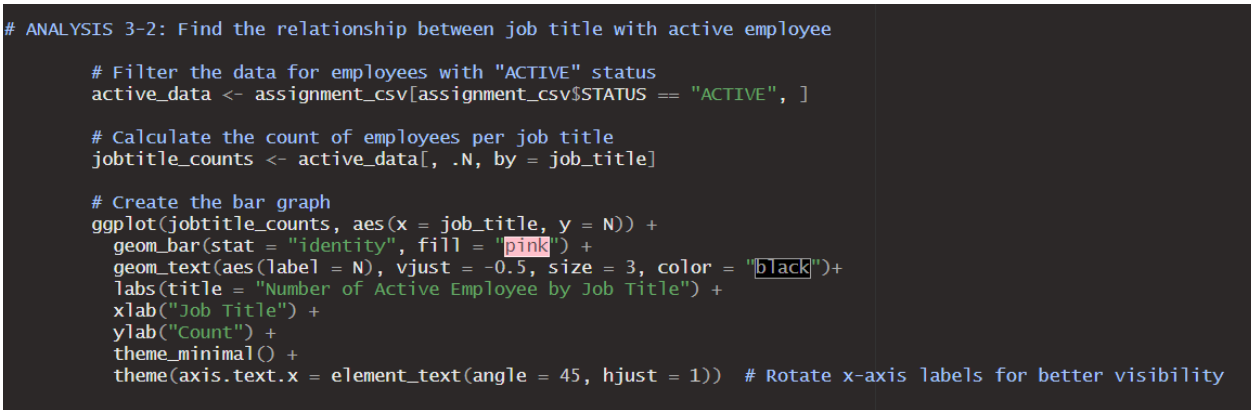


Figure 3.3.2: Relationship between age and active employee count histogram.

Based on the histogram above, it is evident that the highest number of active employees can be found within the age range of 50-59, followed by the age ranges of 40-49, 20-29, 30-39, and those above 60 years old. This also shows that the number of active employees across the age group 20-29, 30-39, 40-49 and 50-59 are quite evenly distributed. A low active employee count for ages above 60 could be mainly caused by retirement.

3.3.2 Analysis 3-2: Find the relationship between job title with active employee



```
# ANALYSIS 3-2: Find the relationship between job title with active employee

        # Filter the data for employees with "ACTIVE" status
        active_data <- assignment_csv[assignment_csv$STATUS == "ACTIVE", ]

        # Calculate the count of employees per job title
        jobtitle_counts <- active_data[, .N, by = job_title]

        # Create the bar graph
        ggplot(jobtitle_counts, aes(x = job_title, y = N)) +
          geom_bar(stat = "identity", fill = "pink") +
          geom_text(aes(label = N), vjust = -0.5, size = 3, color = "black")+
          labs(title = "Number of Active Employee by Job Title") +
          xlab("Job Title") +
          ylab("Count") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))  # Rotate x-axis labels for better visibility
```

Figure 3.3.3: Relationship between job title and active employee count codes.

Analysis 3-2 seeks to investigate the association between job titles and active personnel in order to determine whether particular positions or roles have greater retention rates. A bar graph is plotted using geom_bar() in “ggplot” package to illustrate this relationship.

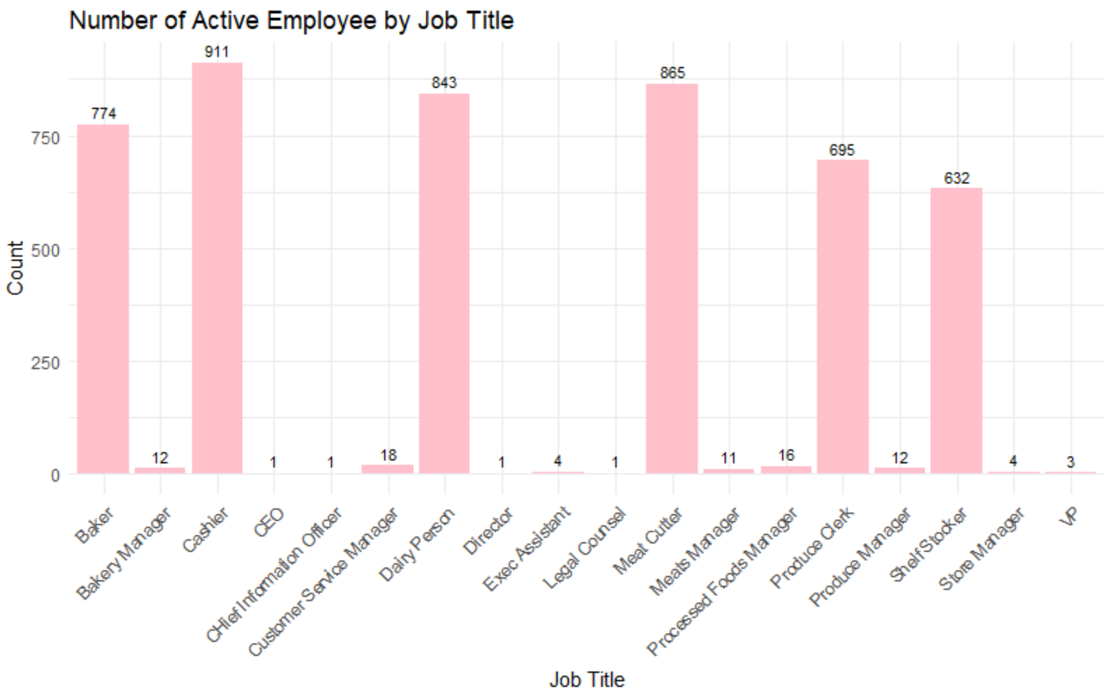
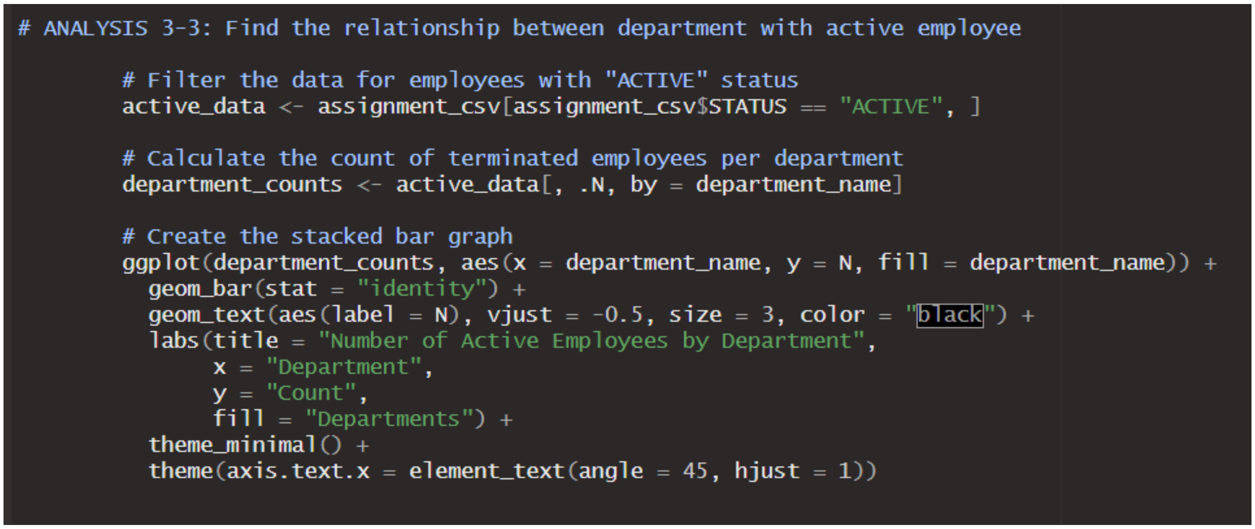


Figure 3.3.4: Relationship between job title and active employee count bar chart.

From the bar graph above, Baker, Cashier, Dairy Person, Meat Cutter, Produce Clerk and Shelf Stocker are job titles with a high number of active employees. The reason for this occurrence could be the organisation requires sufficient employees of these job titles to be stationed across the store branches to handle the business operation of each store branch.

3.3.3 Analysis 3-3: Find the relationship between the department with active employee



```
# ANALYSIS 3-3: Find the relationship between department with active employee

        # Filter the data for employees with "ACTIVE" status
        active_data <- assignment_csv[assignment_csv$STATUS == "ACTIVE", ]

        # Calculate the count of terminated employees per department
        department_counts <- active_data[, .N, by = department_name]

        # Create the stacked bar graph
        ggplot(department_counts, aes(x = department_name, y = N, fill = department_name)) +
          geom_bar(stat = "identity") +
          geom_text(aes(label = N), vjust = -0.5, size = 3, color = "black") +
          labs(title = "Number of Active Employees by Department",
               x = "Department",
               y = "Count",
               fill = "Departments") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.3.5: Relationship between the department and active employee count codes.

Analysis 3-3 seeks to examine the link between departments and active workers in order to determine whether some departments or divisions have greater percentages of staff remaining. A bar graph is plotted using geom_bar() in “ggplot” package to illustrate this relationship.

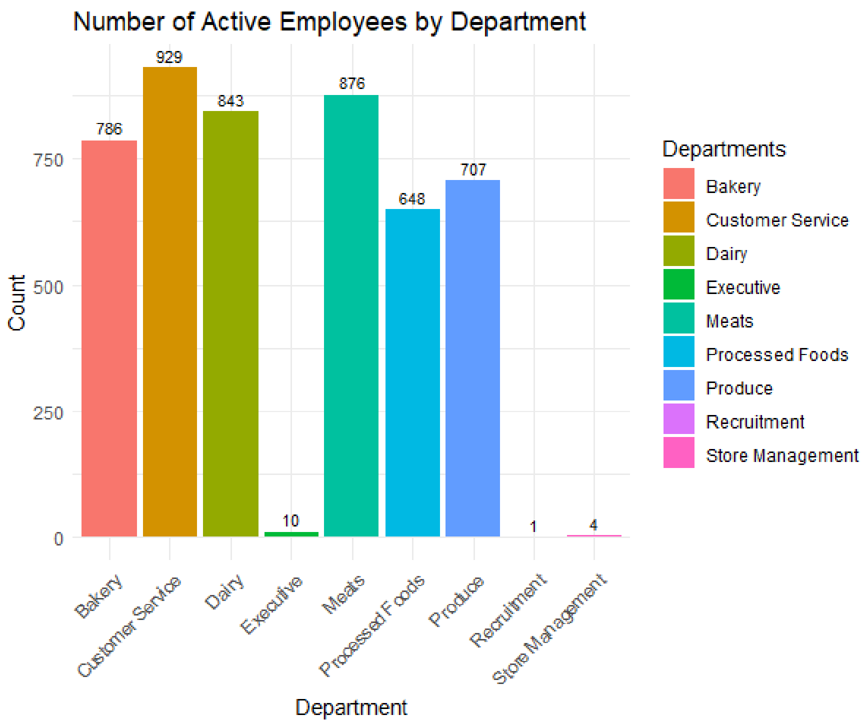
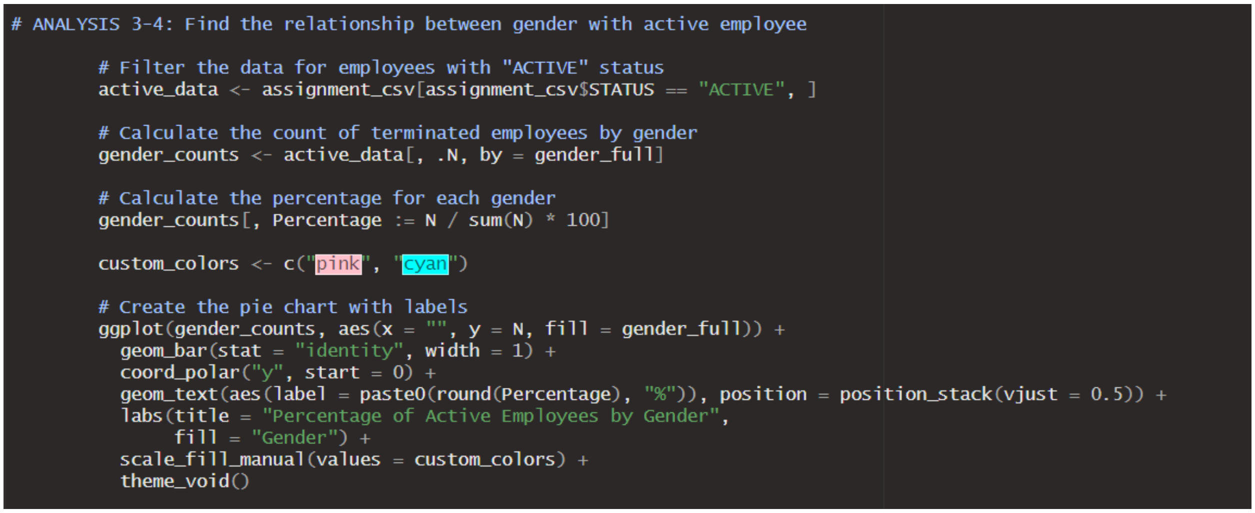


Figure 3.3.6: Relationship between the department and active employee count bar chart.

From the bar graph above, Bakery, Customer Service, Dairy, Meats, Processed Foods and Produce are departments with a high number of active employees. The reason for this occurrence could be the organisation requires sufficient employees under these departments to handle the business operation of each store branch.

3.3.4 Analysis 3-4: Find the relationship between gender with active employee



```
# ANALYSIS 3-4: Find the relationship between gender with active employee

        # Filter the data for employees with "ACTIVE" status
        active_data <- assignment_csv[assignment_csv$STATUS == "ACTIVE", ]

        # Calculate the count of terminated employees by gender
        gender_counts <- active_data[, .N, by = gender_full]

        # Calculate the percentage for each gender
        gender_counts[, Percentage := N / sum(N) * 100]

        custom_colors <- c("pink", "cyan")

        # Create the pie chart with labels
        ggplot(gender_counts, aes(x = "", y = N, fill = gender_full)) +
          geom_bar(stat = "identity", width = 1) +
          coord_polar("y", start = 0) +
          geom_text(aes(label = paste0(round(Percentage), "%")), position = position_stack(vjust = 0.5)) +
          labs(title = "Percentage of Active Employees by Gender",
               fill = "Gender") +
          scale_fill_manual(values = custom_colors) +
          theme_void()
```

Figure 3.3.7: Relationship between gender and active employee count codes.

Analysis 3-4 seek to investigate the link between gender and active workers and might give insight into if there is any gender disparity in the retention of employees. A pie chart is plotted using “ggplot” based on the percentage calculated to visualise the relationship.

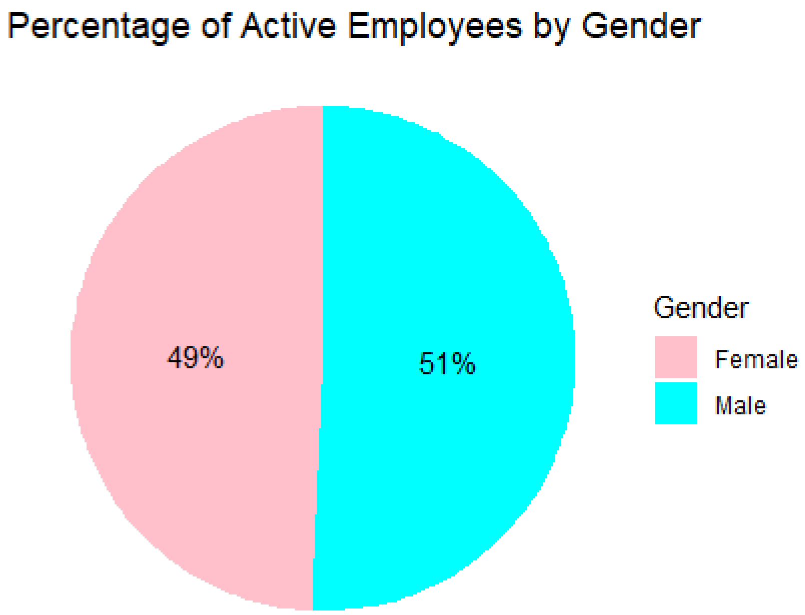
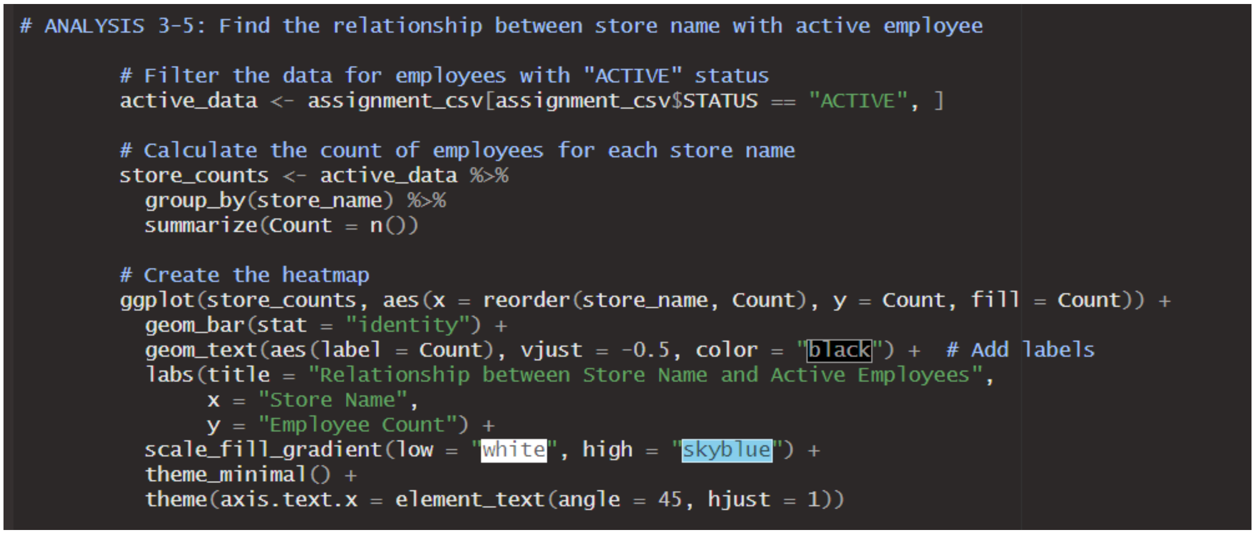


Figure 3.3.8: Relationship between gender and active employee count pie chart.

The pie chart presented above illustrates the distribution of active employees within the company, revealing a higher representation of females than males. Specifically, females constitute 52% of the total employee count, whereas males account for 48%. Based on these findings, it can be inferred that there is a balanced and healthy gender ratio in the organisation.

3.3.5 Analysis 3-5: Find the relationship between store name with active employee



```
# ANALYSIS 3-5: Find the relationship between store name with active employee

        # Filter the data for employees with "ACTIVE" status
        active_data <- assignment_csv[assignment_csv$STATUS == "ACTIVE", ]

        # Calculate the count of employees for each store name
        store_counts <- active_data %>%
          group_by(store_name) %>%
          summarize(Count = n())

        # Create the heatmap
        ggplot(store_counts, aes(x = reorder(store_name, Count), y = Count, fill = Count)) +
          geom_bar(stat = "identity") +
          geom_text(aes(label = Count), vjust = -0.5, color = "black") +  # Add labels
          labs(title = "Relationship between Store Name and Active Employees",
               x = "Store Name",
               y = "Employee Count") +
          scale_fill_gradient(low = "white", high = "skyblue") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.3.9: Relationship between stores and active employee count codes.

Analysis 3-5 aims to investigate the association between shop names and active workers in order to determine whether there are certain store locations with greater staff retention rates. Heatmap is plotted using geom_bar() to illustrate this association, colour depth implies the number of active employee.

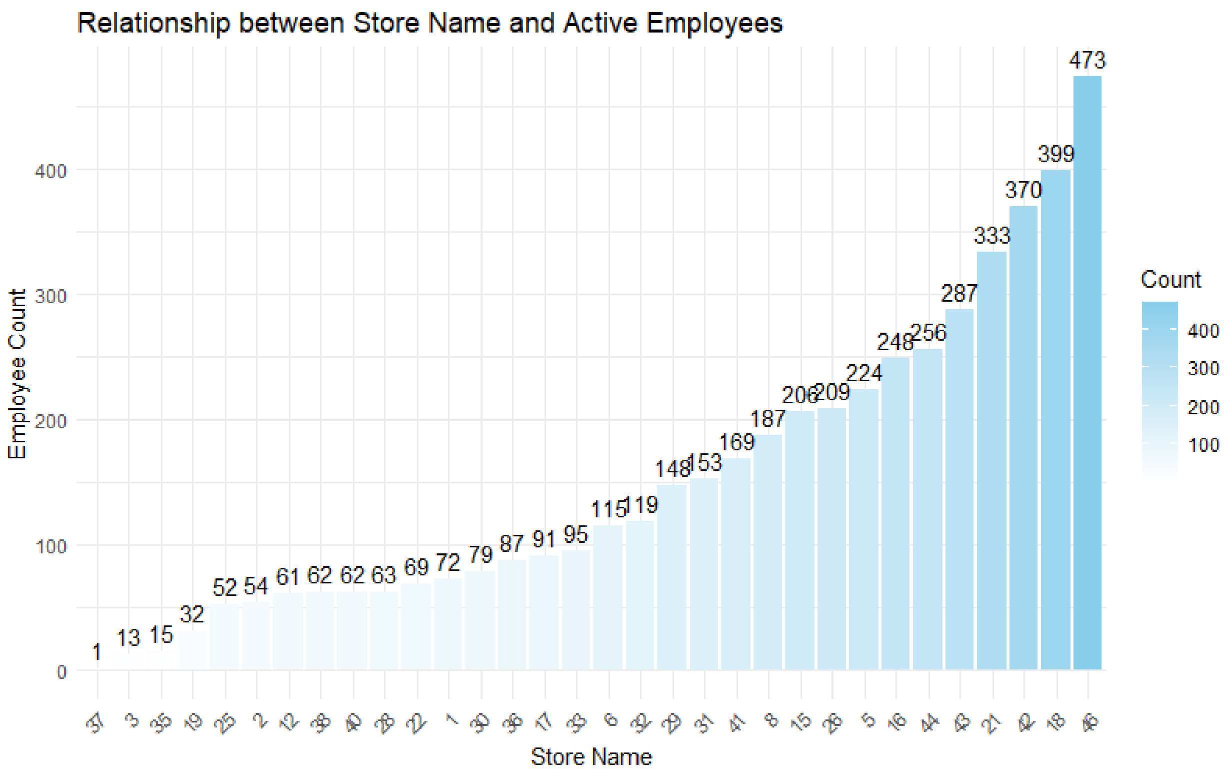
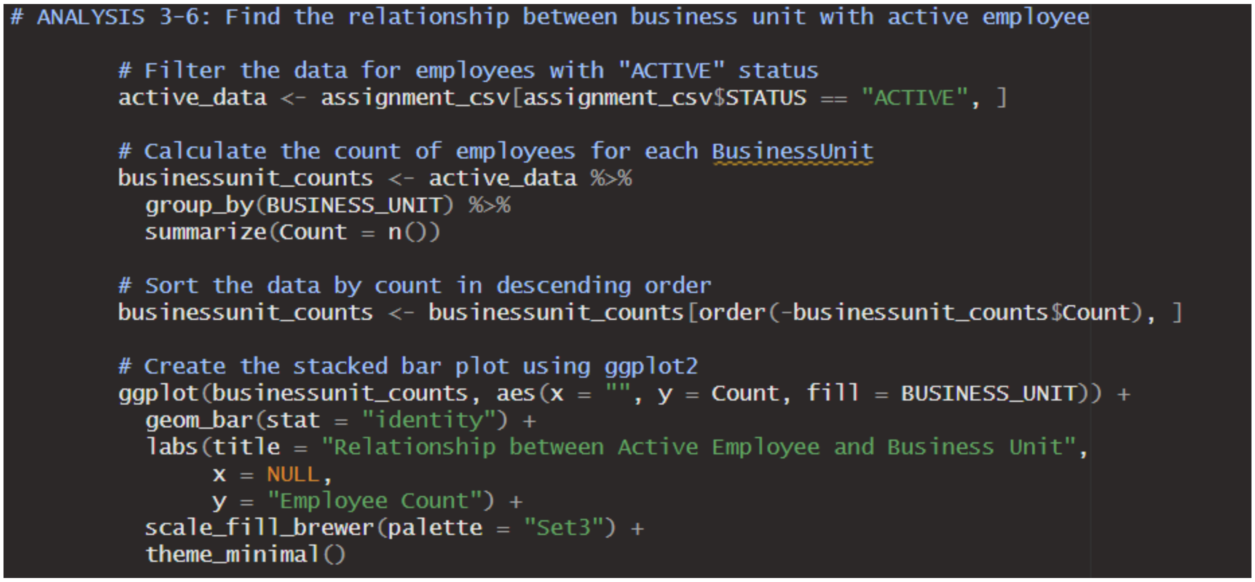


Figure 3.3.10: Relationship between stores and active employee count heatmap.

3.3.6 Analysis 3-6: Find the relationship between the business unit with active employee



```
# ANALYSIS 3-6: Find the relationship between business unit with active employee

        # Filter the data for employees with "ACTIVE" status
        active_data <- assignment_csv[assignment_csv$STATUS == "ACTIVE", ]

        # Calculate the count of employees for each BusinessUnit
        businessunit_counts <- active_data %>%
          group_by(BUSINESS_UNIT) %>%
          summarize(Count = n())

        # Sort the data by count in descending order
        businessunit_counts <- businessunit_counts[order(-businessunit_counts$Count), ]

        # Create the stacked bar plot using ggplot2
        ggplot(businessunit_counts, aes(x = "", y = Count, fill = BUSINESS_UNIT)) +
          geom_bar(stat = "identity") +
          labs(title = "Relationship between Active Employee and Business Unit",
               x = NULL,
               y = "Employee Count") +
          scale_fill_brewer(palette = "Set3") +
          theme_minimal()
```

Figure 3.3.11: Relationship between business unit and active employee count codes.

Analysis 3-6 seeks to investigate the link between business units and active workers in order to determine whether there are certain business units or divisions with higher employee retention rates. A stacked bar graph is plotted using geom_bar() in “ggplot” package to illustrate this relationship.

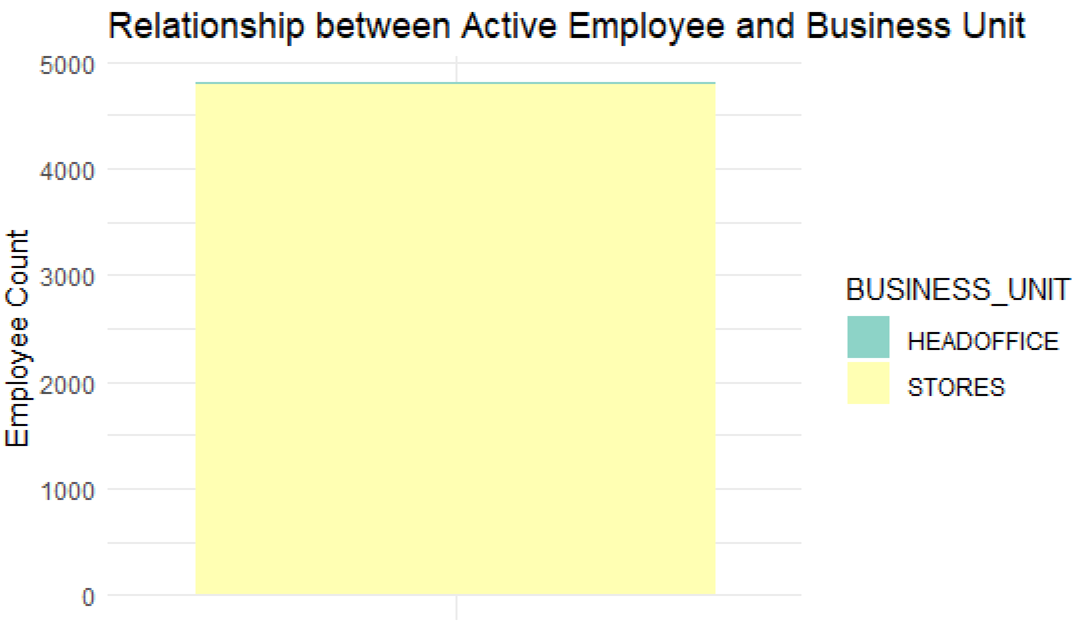
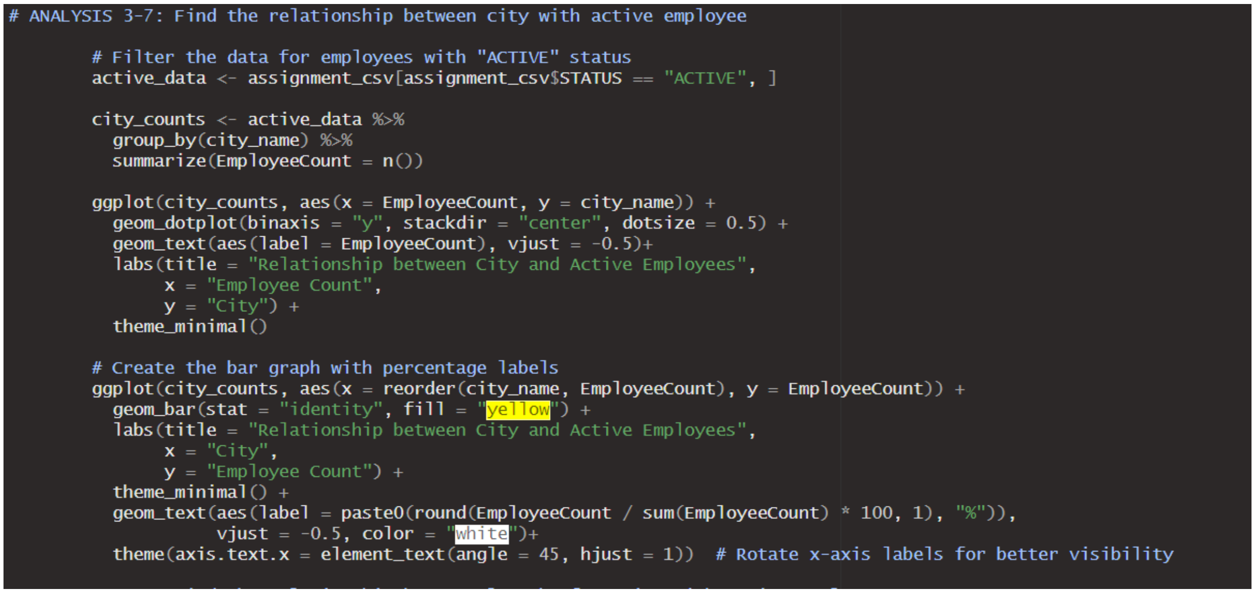


Figure 3.3.12: Relationship between the business unit and active employee count bar chart.

The stacked bar plot presented above depicts a notable disparity in the count of active employees between the stores and the head office. Specifically, the stores exhibit a significantly higher count, surpassing 4500 employees.

3.3.7 Analysis 3-7: Find the relationship between the city with active employee



```
# ANALYSIS 3-7: Find the relationship between city with active employee

        # Filter the data for employees with "ACTIVE" status
        active_data <- assignment_csv[assignment_csv$STATUS == "ACTIVE", ]

        city_counts <- active_data %>%
          group_by(city_name) %>%
          summarize(EmployeeCount = n())

        ggplot(city_counts, aes(x = EmployeeCount, y = city_name)) +
          geom_dotplot(binaxis = "y", stackdir = "center", dotsize = 0.5) +
          geom_text(aes(label = EmployeeCount), vjust = -0.5)+
          labs(title = "Relationship between City and Active Employees",
               x = "Employee Count",
               y = "City") +
          theme_minimal()

        # Create the bar graph with percentage labels
        ggplot(city_counts, aes(x = reorder(city_name, EmployeeCount), y = EmployeeCount)) +
          geom_bar(stat = "identity", fill = "yellow") +
          labs(title = "Relationship between City and Active Employees",
               x = "City",
               y = "Employee Count") +
          theme_minimal() +
          geom_text(aes(label = paste0(round(EmployeeCount / sum(EmployeeCount) * 100, 1), "%")),
                    vjust = -0.5, color = "white")+
          theme(axis.text.x = element_text(angle = 45, hjust = 1))  # Rotate x-axis labels for better visibility
```

Figure 3.3.13: Relationship between the city and active employee count codes.

Analysis 3-7 seek to investigate the association between cities and active workers in order to see whether there are distinct geographic areas with higher staff retention rates. A bar graph is plotted using geom_bar() in “ggplot” package to illustrate this relationship.

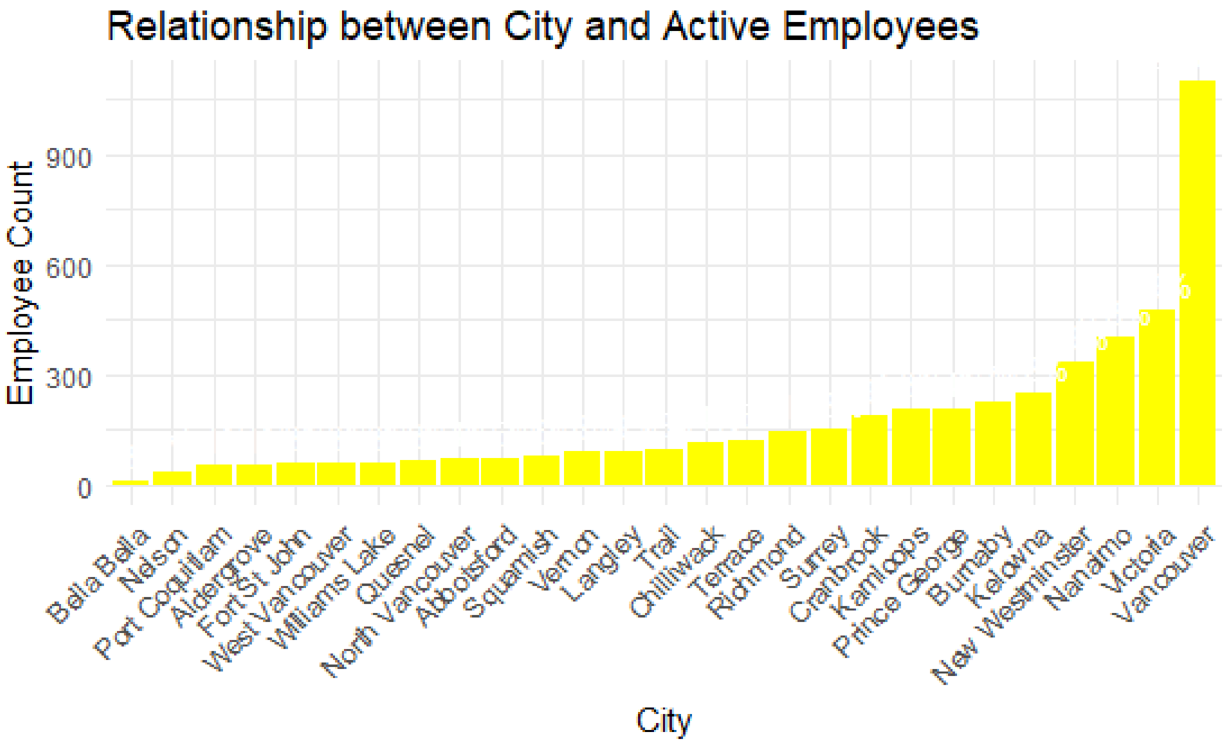
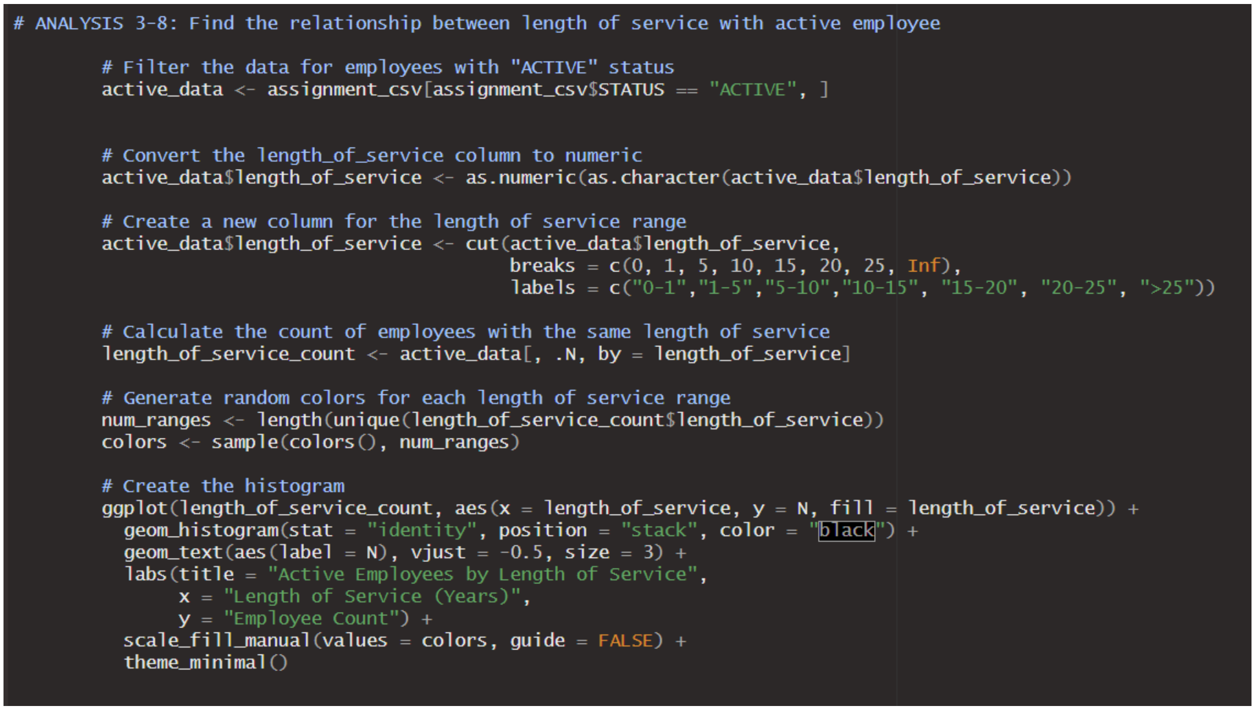


Figure 3.3.14: Relationship between the city and active employee count bar chart.

The bar graph presented above illustrates the correlation between different cities and their respective active employee counts. Vancouver emerges as the city with the highest number of active employees, exceeding 900 individuals. Other cities such as Victoria, Nanaimo, and New Westminster exhibit active employee counts reaching approximately 300. On the other hand, cities like Kelowna, Burnaby, Kamloops, and others display fewer than 300 active employees. It is worth noting that the cities of Blue River, Dease Lake, Valemount, Cortes Island, Pitt Meadows, and Ocean Falls exhibit similarly low employee counts, which are the lowest among the cities represented in the graph.

3.3.8 Analysis 3-8: Find the relationship between the length of service with active employee



```
# ANALYSIS 3-8: Find the relationship between length of service with active employee

        # Filter the data for employees with "ACTIVE" status
        active_data <- assignment_csv[assignment_csv$STATUS == "ACTIVE", ]


        # Convert the length_of_service column to numeric
        active_data$length_of_service <- as.numeric(as.character(active_data$length_of_service))

        # Create a new column for the length of service range
        active_data$length_of_service <- cut(active_data$length_of_service,
                                             breaks = c(0, 1, 5, 10, 15, 20, 25, Inf),
                                             labels = c("0-1","1-5","5-10","10-15", "15-20", "20-25", ">25"))

        # Calculate the count of employees with the same length of service
        length_of_service_count <- active_data[, .N, by = length_of_service]

        # Generate random colors for each length of service range
        num_ranges <- length(unique(length_of_service_count$length_of_service))
        colors <- sample(colors(), num_ranges)

        # Create the histogram
        ggplot(length_of_service_count, aes(x = length_of_service, y = N, fill = length_of_service)) +
          geom_histogram(stat = "identity", position = "stack", color = "black") +
          geom_text(aes(label = N), vjust = -0.5, size = 3) +
          labs(title = "Active Employees by Length of Service",
               x = "Length of Service (Years)",
               y = "Employee Count") +
          scale_fill_manual(values = colors, guide = FALSE) +
          theme_minimal()
```

Figure 3.3.15: Relationship between length of service and active employee count codes.

Analysis 3-8 seeks to investigate the relationship between the length of service and active workers in order to determine whether there is a link between the length of service and employee retention. The “geom_histogram()” function in the “ggplot” package is used to plot a histogram that illustrates the age distribution of the active employees, with the x-axis representing the “length of service ranges” and the y-axis showing the “ employee count”.

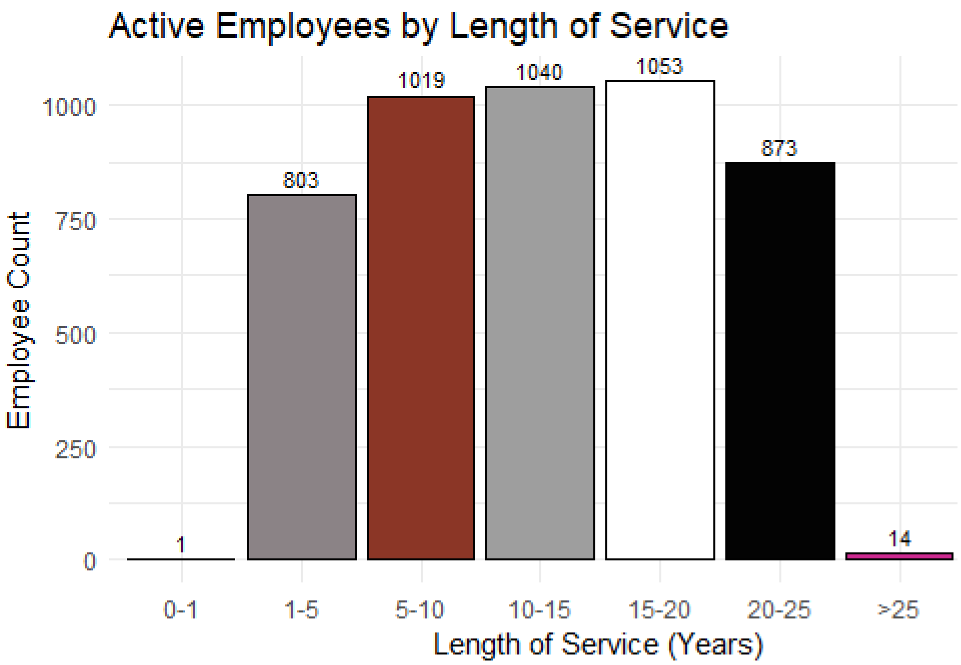
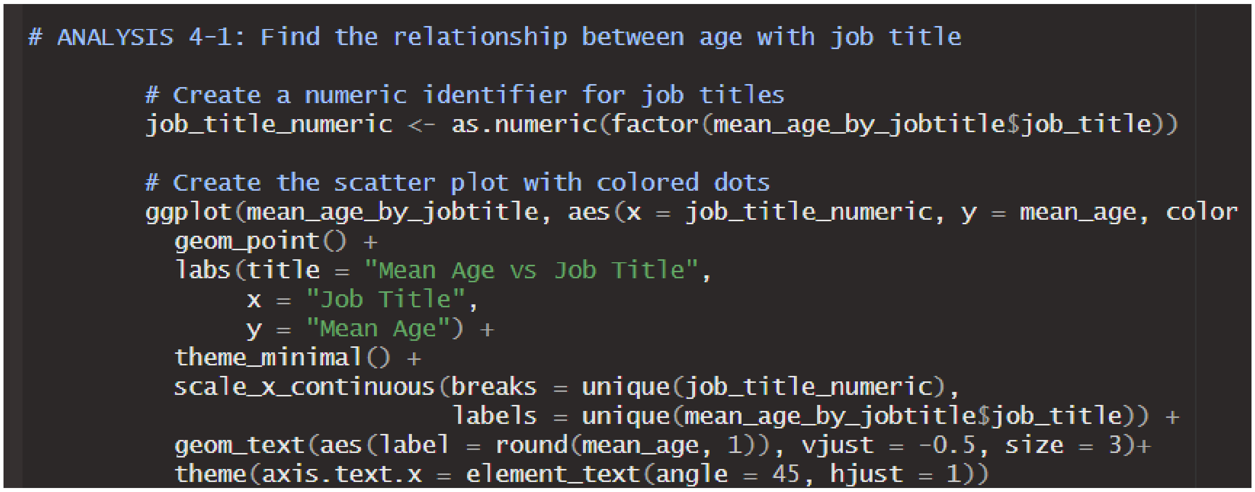


Figure 3.3.16: Relationship between length of service and active employee histogram.

In the bar chart above, it shows that the count of active employees with 0-1 year length of service is the lowest, with only 1 employee. While, the count of active employees with 15-20 years of experience is the highest, with 1053 employees. The number of active employees in the remaining range of length of service is also evenly distributed. This implies that the active employees in the company are the majority of senior employees of the organisation. The company is assumed to provide employees with good employee benefits which motivates active employees to continue working here.

3.4 Question 4: What are the factors influencing Employee promotion opportunities?

3.4.1 Analysis 4-1: Find the relationship between age with job title



```
# ANALYSIS 4-1: Find the relationship between age with job title

        # Create a numeric identifier for job titles
        job_title_numeric <- as.numeric(factor(mean_age_by_jobtitle$job_title))

        # Create the scatter plot with colored dots
        ggplot(mean_age_by_jobtitle, aes(x = job_title_numeric, y = mean_age, color
          geom_point() +
          labs(title = "Mean Age vs Job Title",
               x = "Job Title",
               y = "Mean Age") +
          theme_minimal() +
          scale_x_continuous(breaks = unique(job_title_numeric),
                             labels = unique(mean_age_by_jobtitle$job_title)) +
          geom_text(aes(label = round(mean_age, 1)), vjust = -0.5, size = 3)+
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.1.1: Relationship between age and job title codes.

Analysis 4-1 examines the relationship between age and job titles to see if there is a correlation between age groups and job positions. This contributes to determining if age plays a role in deciding job promotion possibilities. The scatter plot is plotted using geom_point() in “ggplot” package to illustrate the relationship.

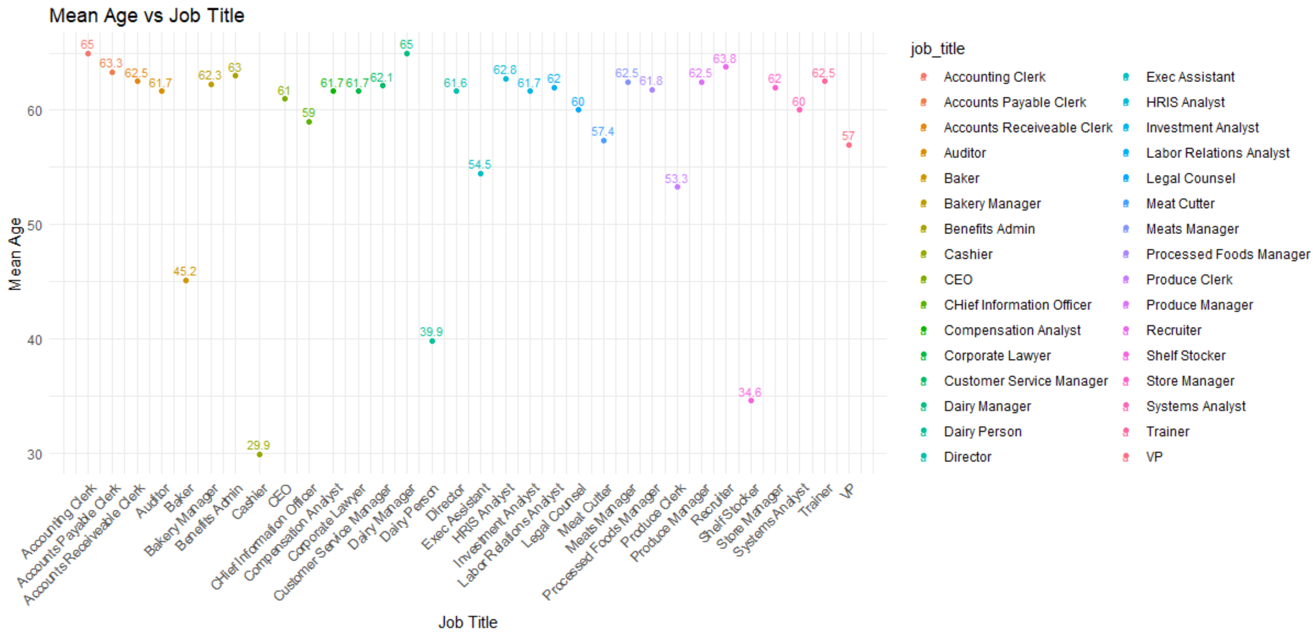
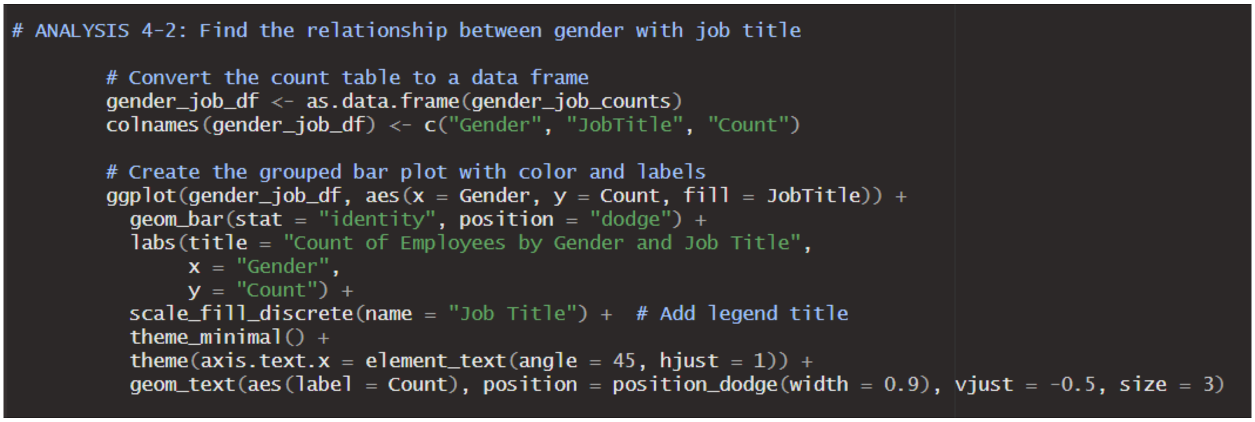


Figure 3.4.2: Relationship between age and job title scatter plot.

Based on the scatter graph above the mean age of higher job positions is higher compared to the entry-level job positions such as Cashier, Shelf Stocker, Dairy Person and Baker. Age could be a contributing factor in job promotion where younger employees are more likely to start in entry-level roles. However, they will eventually advance in their careers as they mature.

3.4.2 Analysis 4-2: Find the relationship between gender with job title



```
# ANALYSIS 4-2: Find the relationship between gender with job title

        # Convert the count table to a data frame
        gender_job_df <- as.data.frame(gender_job_counts)
        colnames(gender_job_df) <- c("Gender", "JobTitle", "Count")

        # Create the grouped bar plot with color and labels
        ggplot(gender_job_df, aes(x = Gender, y = Count, fill = JobTitle)) +
          geom_bar(stat = "identity", position = "dodge") +
          labs(title = "Count of Employees by Gender and Job Title",
               x = "Gender",
               y = "Count") +
          scale_fill_discrete(name = "Job Title") +  # Add legend title
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
          geom_text(aes(label = Count), position = position_dodge(width = 0.9), vjust = -0.5, size = 3)
```

Figure 3.4.3: Relationship between gender and job title codes.

Analysis 4-2 analyses the link between gender and job titles to see if there are any disparities due to gender in the distribution of employment roles. A bar graph is plotted using geom_bar() to illustrate this association.

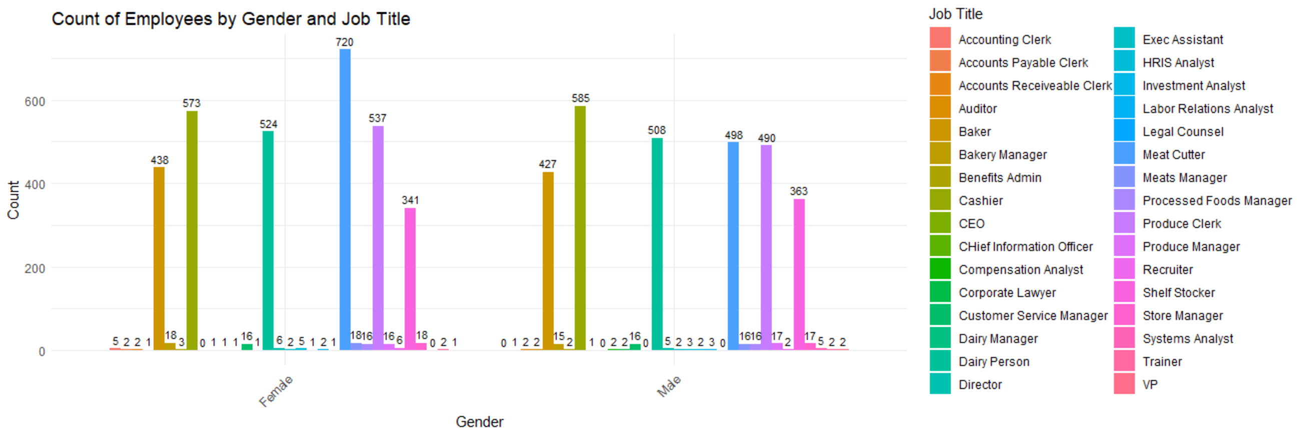
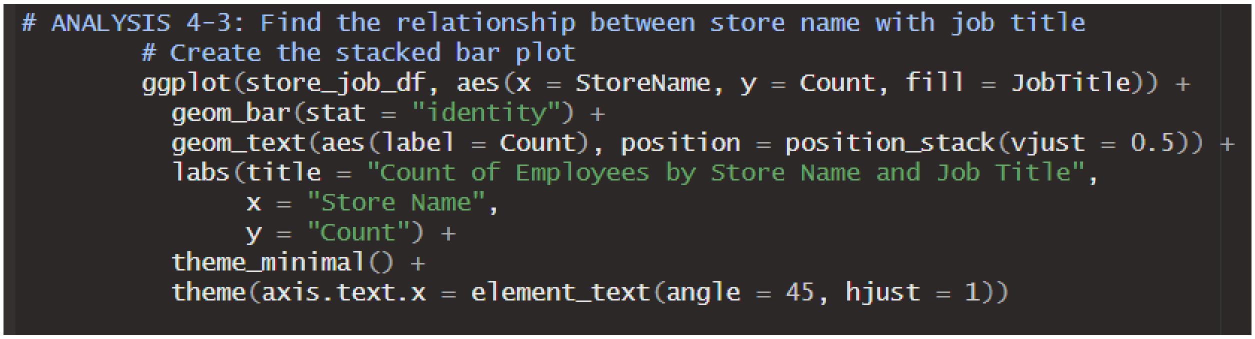


Figure 3.4.:4 Relationship between gender and job title bar graph.

The bar graph above shows the number of females and males in each department. From the graph, the ratio between male and female employees is quite balanced. This indicates that the job positions available in the organisation do not judge job promotion candidates based on gender, both genders have equal opportunities for a job promotion.

3.4.3 Analysis 4-3: Find the relationship between store name with job title



```
# ANALYSIS 4-3: Find the relationship between store name with job title
        # Create the stacked bar plot
        ggplot(store_job_df, aes(x = StoreName, y = Count, fill = JobTitle)) +
          geom_bar(stat = "identity") +
          geom_text(aes(label = Count), position = position_stack(vjust = 0.5)) +
          labs(title = "Count of Employees by Store Name and Job Title",
               x = "Store Name",
               y = "Count") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.4.5: Relationship between store and job title codes.

Analysis 4-3 analyses the association between shop names and job titles to see whether there are distinct retail locations with employees in higher-level roles. It helps in determining which stores have greater promotional chances. A bar graph is plotted using geom_bar() to illustrate this association.

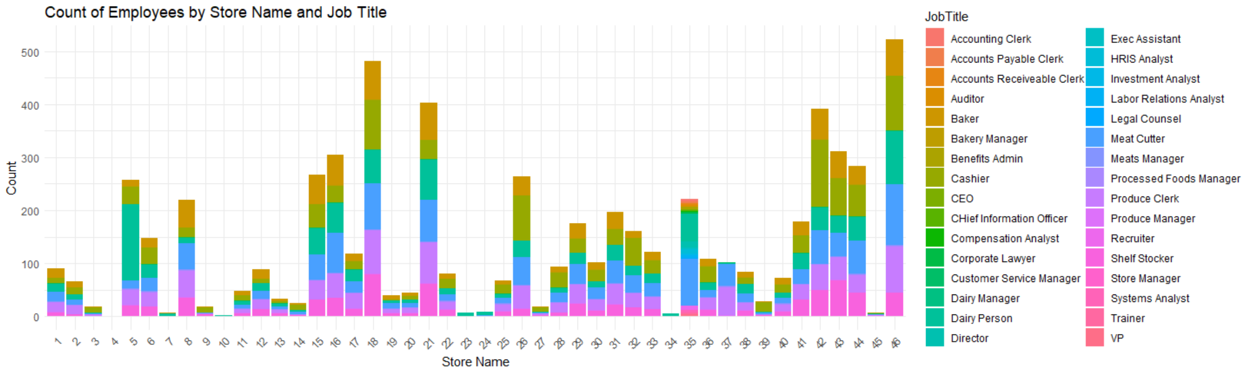
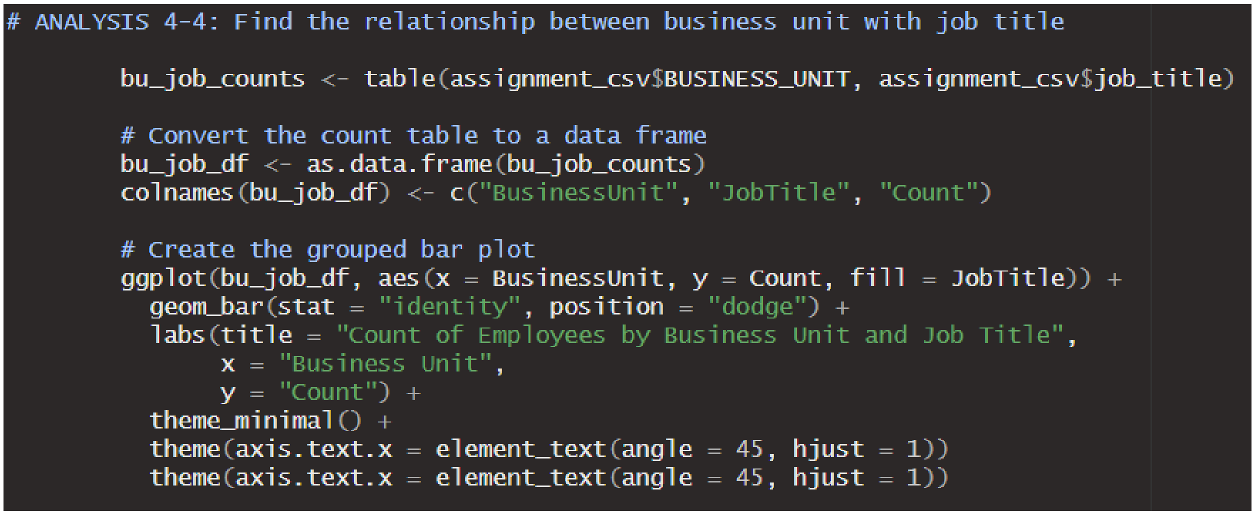


Figure 3.4.6: Relationship between store name and job title bar graph.

From the bar chart above, certain stores like stores 48, 42, 21 and 18 have a high count of employees overall among other stores. This indicates that these stores have high competition among employees, thus, chances for job promotion are lower. Vice versa, chances for job promotion may be higher among stores with less employee count overall. However, it could also indicate that there are more positions available in stores with more employee count overall.

3.4.4 Analysis 4-4: Find the relationship between the business unit and job title



```
# ANALYSIS 4-4: Find the relationship between business unit with job title

        bu_job_counts <- table(assignment_csv$BUSINESS_UNIT, assignment_csv$job_title)

        # Convert the count table to a data frame
        bu_job_df <- as.data.frame(bu_job_counts)
        colnames(bu_job_df) <- c("BusinessUnit", "JobTitle", "Count")

        # Create the grouped bar plot
        ggplot(bu_job_df, aes(x = BusinessUnit, y = Count, fill = JobTitle)) +
          geom_bar(stat = "identity", position = "dodge") +
          labs(title = "Count of Employees by Business Unit and Job Title",
               x = "Business Unit",
               y = "Count") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.4.7: Relationship between the business unit and job title codes.

Analysis 4-4 tries to find the link between business units and job titles, which can reveal whether certain units or departments have personnel in higher-level roles. A bar graph is plotted using geom_bar() to illustrate this association.

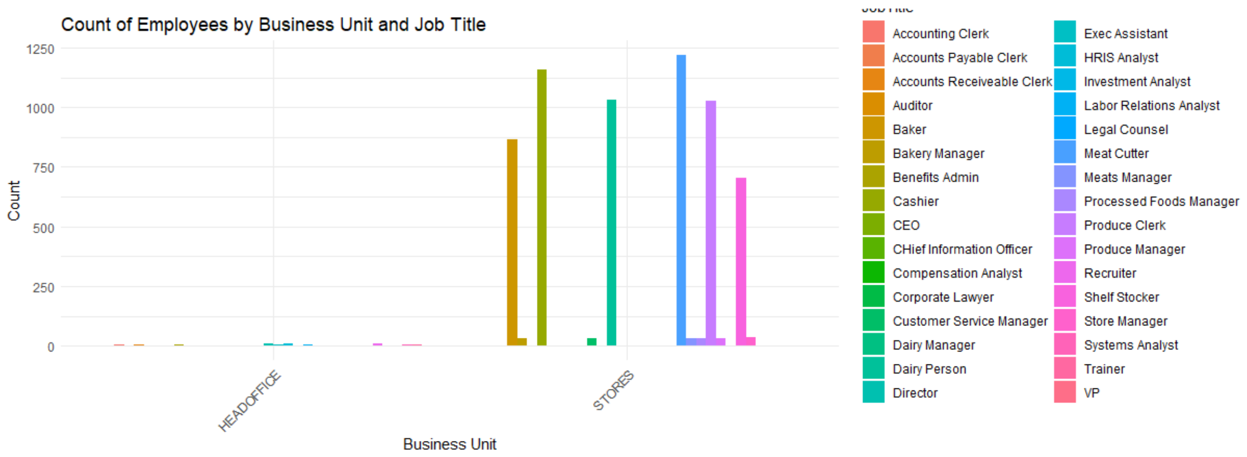
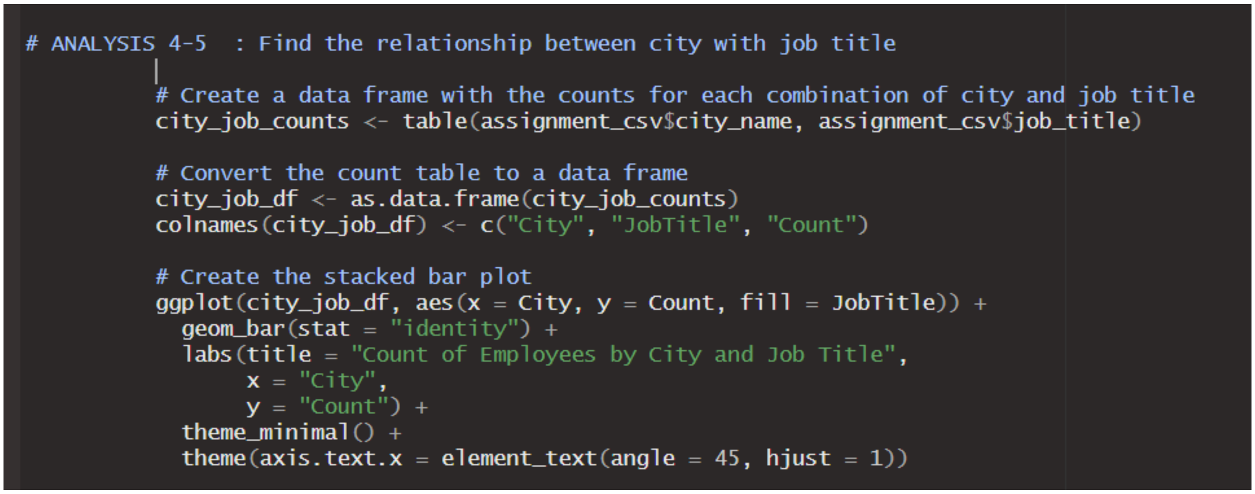


Figure 3.4.8: Relationship between the business unit and job title bar chart.

From the graph above, the employee count grouped by job title in the head office is relatively low. It shows that the majority of employees work in stores branches than in the head office. This indicates that chance for promotion can be limited in the head office. Whereas, the stores have more job positions available but competition may be high.

3.4.5 Analysis 4-5: Find the relationship between the city and job title



```
# ANALYSIS 4-5  : Find the relationship between city with job title

          # Create a data frame with the counts for each combination of city and job title
          city_job_counts <- table(assignment_csv$city_name, assignment_csv$job_title)

          # Convert the count table to a data frame
          city_job_df <- as.data.frame(city_job_counts)
          colnames(city_job_df) <- c("City", "JobTitle", "Count")

          # Create the stacked bar plot
          ggplot(city_job_df, aes(x = City, y = Count, fill = JobTitle)) +
            geom_bar(stat = "identity") +
            labs(title = "Count of Employees by City and Job Title",
                 x = "City",
                 y = "Count") +
            theme_minimal() +
            theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.4.9: Relationship between city and job title codes.

Analysis 4-5 aims to explore the association between cities and job titles and might assist detect whether there are distinct geographic areas where individuals have more prospects for advancement. A bar graph is plotted using geom_bar() to illustrate this association.

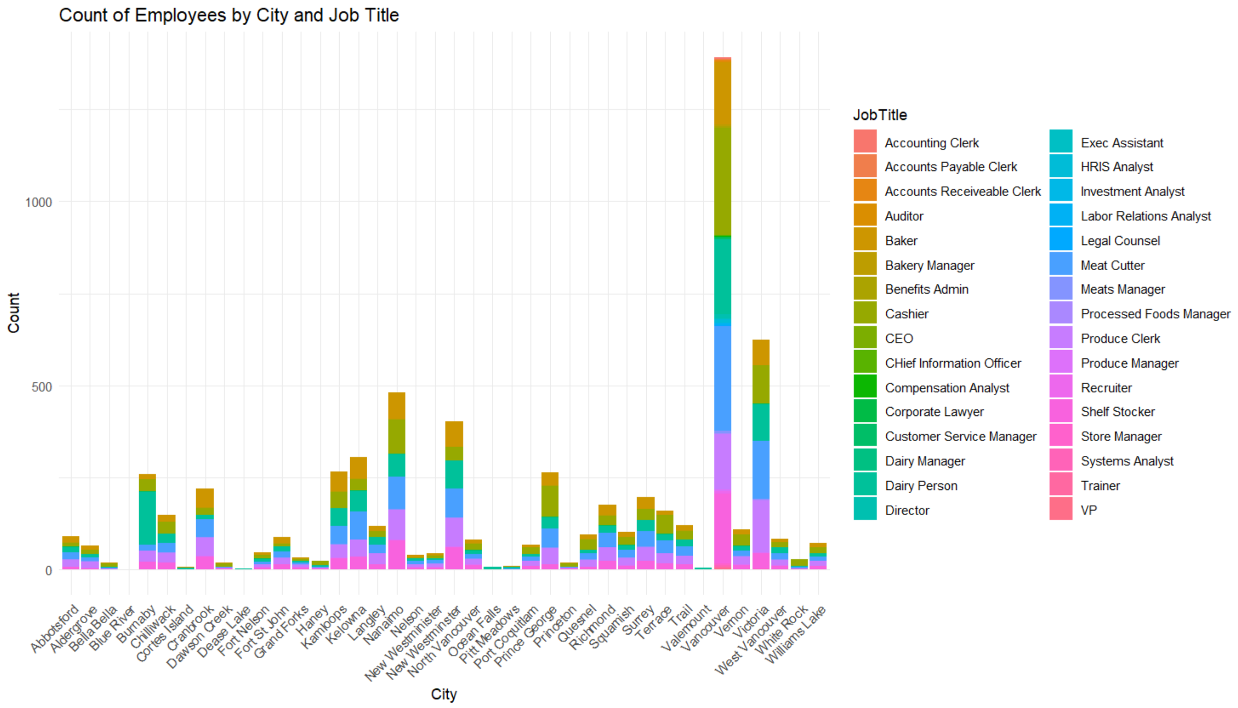
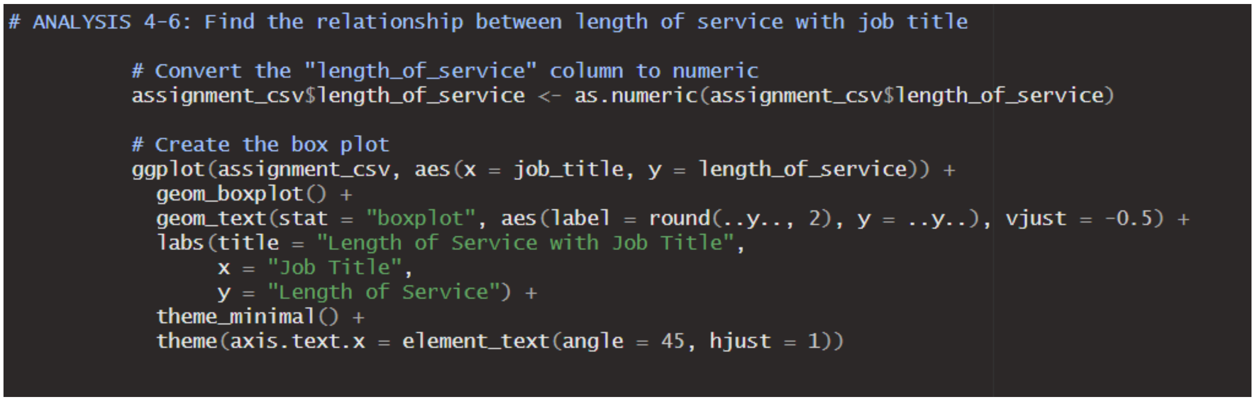


Figure 3.4.10: Relationship between the city and job title bar chart.

The graph above shows the employee count group by job title in different cities. Among other cities, Vancouver has the most significant number of employees. This indicates that most store branches are located in Vancouver. Chances for promotion may be higher there but competition can be high.

3.4.6 Analysis 4-6: Find the relationship between the length of service with the job title



```
# ANALYSIS 4-6: Find the relationship between length of service with job title

          # Convert the "length_of_service" column to numeric
          assignment_csv$length_of_service <- as.numeric(assignment_csv$length_of_service)

          # Create the box plot
          ggplot(assignment_csv, aes(x = job_title, y = length_of_service)) +
            geom_boxplot() +
            geom_text(stat = "boxplot", aes(label = round(..y.., 2), y = ..y..), vjust = -0.5) +
            labs(title = "Length of Service with Job Title",
                 x = "Job Title",
                 y = "Length of Service") +
            theme_minimal() +
            theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.4.11: Relationship between length of service and job title codes.

The objectives of Analysis 4-6 Finding the link between the length of service and job titles can give insight into whether or not there is a relationship between the length of employment and the degree of work positions held. A box plot graph is plotted using geom_boxplot() to illustrate this association.

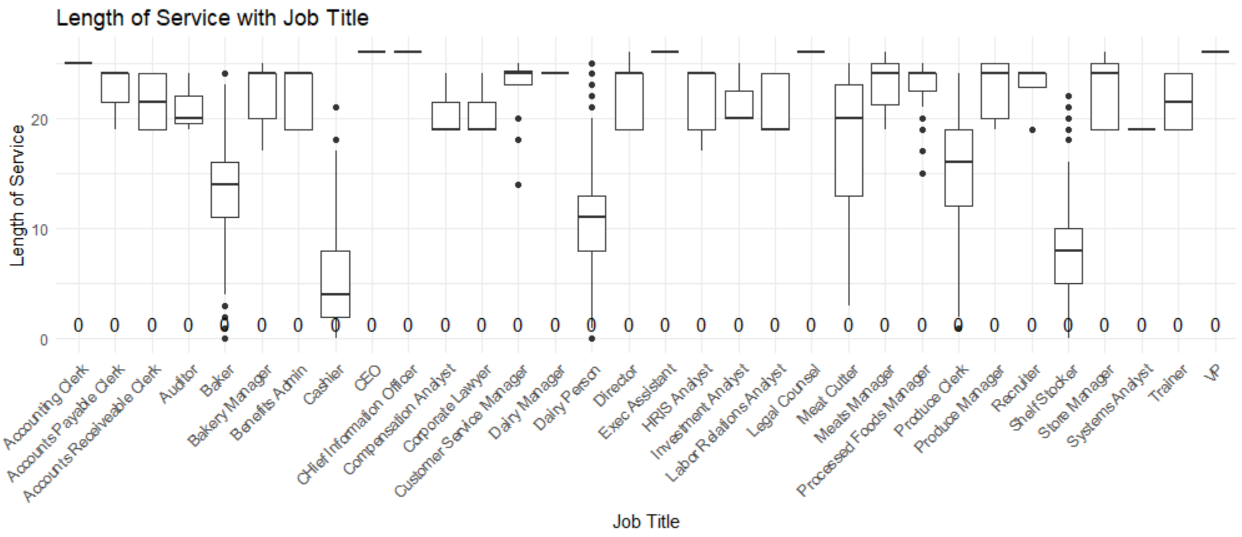
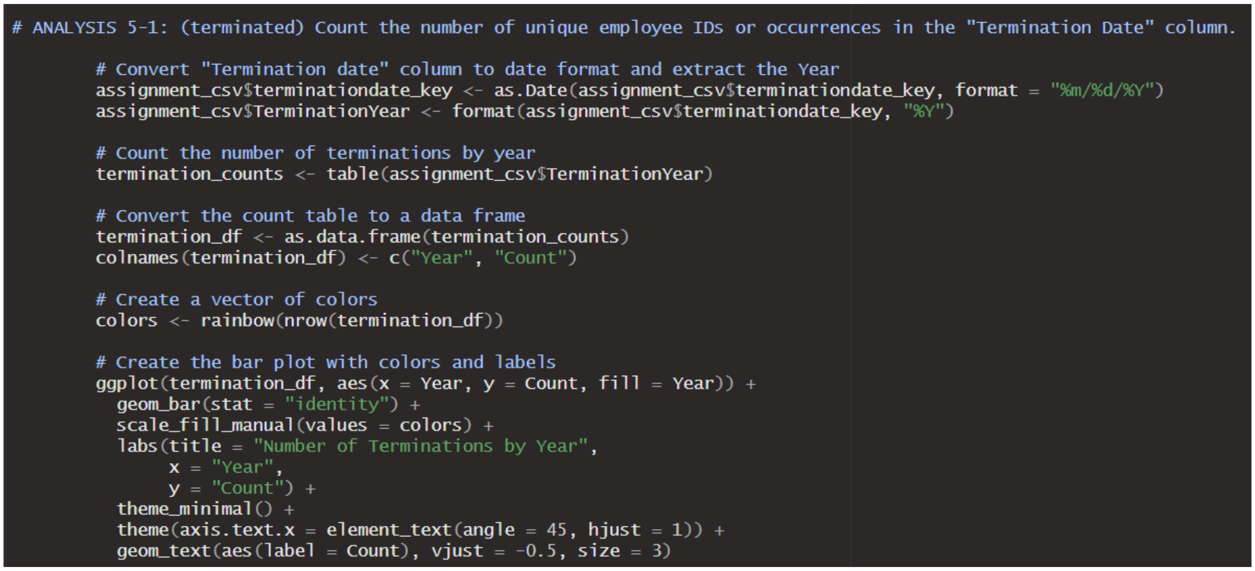


Figure 3.4.12: Relationship between length of service and job title box plot.

The box plot above shows the employee count group by job title based on length of service. The boxplots above show lower employee length of service in entry-level job positions such as Cashier, Shelf Stocker, Dairy Person and Baker. Whereas, the more administrative, technical jobs show higher show lower employee length of service. This could be because employees who have been in technical roles need longer periods of time to master their foundation of knowledge and competence slowly nurturing them to be more suitable for promotions to higher-level technical positions. Hence, the length of service for those jobs is higher.

3.5 Question 5: Are the number of terminated employees replaced by an equal number of new hires?

3.5.1 Analysis 5-1: Calculate the number of terminated employees by year



```
# ANALYSIS 5-1: (terminated) Count the number of unique employee IDs or occurrences in the "Termination Date" column.

        # Convert "Termination date" column to date format and extract the Year
        assignment_csv$terminationdate_key <- as.Date(assignment_csv$terminationdate_key, format = "%m/%d/%Y")
        assignment_csv$TerminationYear <- format(assignment_csv$terminationdate_key, "%Y")

        # Count the number of terminations by year
        termination_counts <- table(assignment_csv$TerminationYear)

        # Convert the count table to a data frame
        termination_df <- as.data.frame(termination_counts)
        colnames(termination_df) <- c("Year", "Count")

        # Create a vector of colors
        colors <- rainbow(nrow(termination_df))

        # Create the bar plot with colors and labels
        ggplot(termination_df, aes(x = Year, y = Count, fill = Year)) +
          geom_bar(stat = "identity") +
          scale_fill_manual(values = colors) +
          labs(title = "Number of Terminations by Year",
               x = "Year",
               y = "Count") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
          geom_text(aes(label = Count), vjust = -0.5, size = 3)
```

Figure 3.5.1: Number of termination by year codes

Analysis 5-1 seeks to identify the number of terminated workers for each year, hence offering insight into the employee turnover rate. A box plot graph is plotted using geom_boxplot() to illustrate this association.

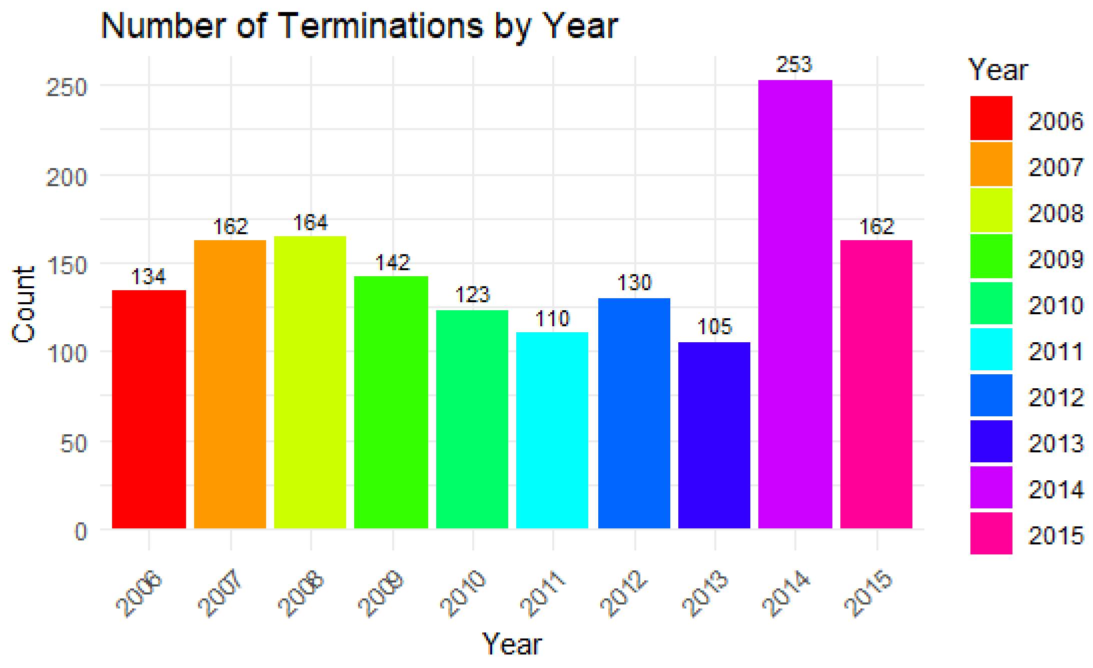
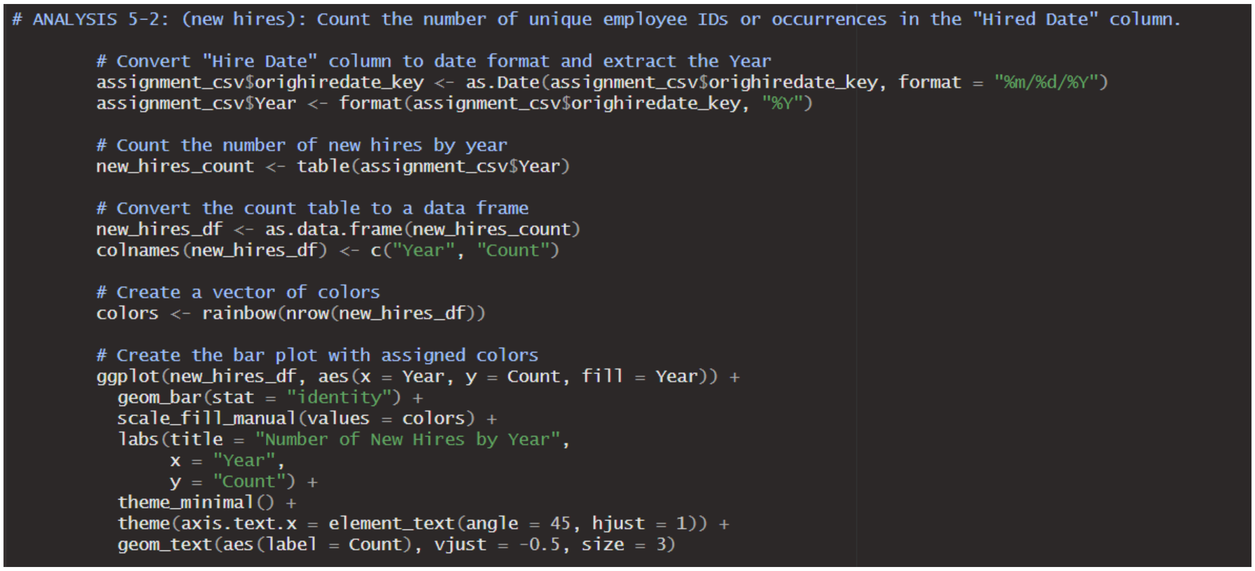


Figure 3.5.2: Number of terminations by the year bar chart

The bar graph above shows the number of employee turnovers by year. From the bar graph above, the year 2013 has the least significant employee termination while the year 2014 has the most significant employee termination. There is a huge rise in employee termination within a year. Termination counts can vary over a year for a variety of reasons. For instance, changes in economic conditions, company policies, and digital transformation. This graph provides insights into the employee turnover rate over the years. Later on, we'll uncover if turnover rate is covered by the employee hire rate to determine if the organisation has a healthy ratio.

3.5.2 Analysis 5-2: Calculate the number of hires by year



```
# ANALYSIS 5-2: (new hires): Count the number of unique employee IDs or occurrences in the "Hired Date" column.

        # Convert "Hire Date" column to date format and extract the Year
        assignment_csv$orighiredate_key <- as.Date(assignment_csv$orighiredate_key, format = "%m/%d/%Y")
        assignment_csv$Year <- format(assignment_csv$orighiredate_key, "%Y")

        # Count the number of new hires by year
        new_hires_count <- table(assignment_csv$Year)

        # Convert the count table to a data frame
        new_hires_df <- as.data.frame(new_hires_count)
        colnames(new_hires_df) <- c("Year", "Count")

        # Create a vector of colors
        colors <- rainbow(nrow(new_hires_df))

        # Create the bar plot with assigned colors
        ggplot(new_hires_df, aes(x = Year, y = Count, fill = Year)) +
          geom_bar(stat = "identity") +
          scale_fill_manual(values = colors) +
          labs(title = "Number of New Hires by Year",
               x = "Year",
               y = "Count") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
          geom_text(aes(label = Count), vjust = -0.5, size = 3)
```

Figure 3.5.3: Number of newly hired by year codes.

Analysis 5-2 seeks to identify the number of new recruits for each year, indicating the organization's recruiting and hiring patterns.

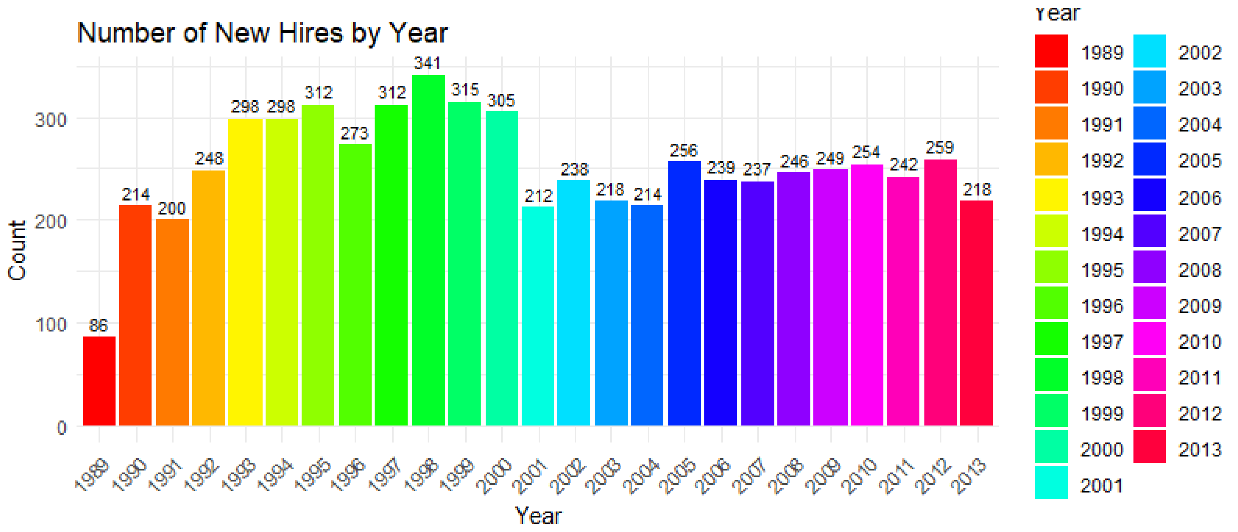
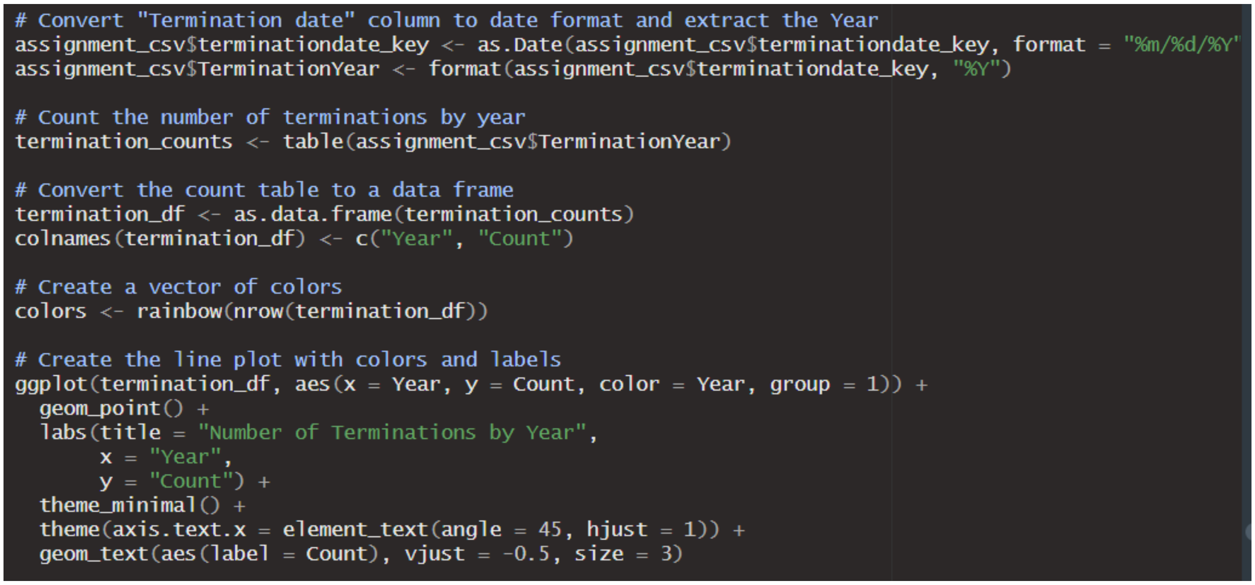


Figure 3.5.4: Number of newly hired by the year bar chart.

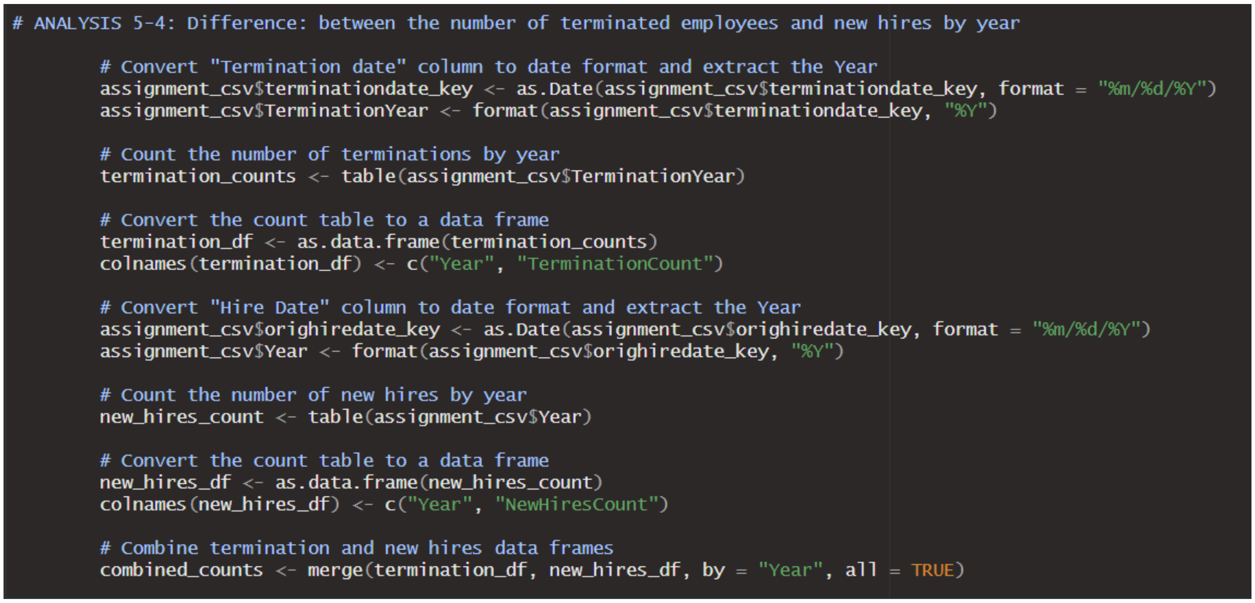
The bar graph above visualise the number of employees hired by year. From the line graph above, the year 1989 has the least employee hired. This could be because the organisation's size is still small, and not many employees are needed to handle the business. Meanwhile, the year 1998 has the most employee hired. This is because the business is more stable now, the owner is planning to expand the business, therefore, more employees are hired.

3.5.3 Analysis 5-3: Compare the count of terminated employees with the count of new hires



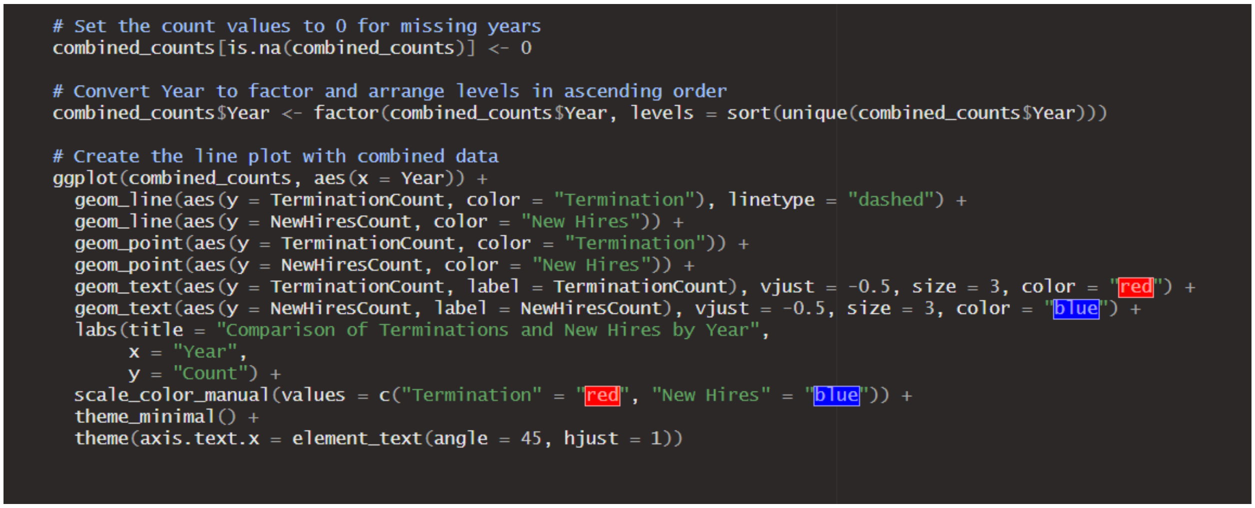
```r
# Convert "Termination date" column to date format and extract the Year
assignment_csv$terminationdate_key <- as.Date(assignment_csv$terminationdate_key, format = "%m/%d/%Y"
assignment_csv$TerminationYear <- format(assignment_csv$terminationdate_key, "%Y")

# Count the number of terminations by year
termination_counts <- table(assignment_csv$TerminationYear)

# Convert the count table to a data frame
termination_df <- as.data.frame(termination_counts)
colnames(termination_df) <- c("Year", "Count")

# Create a vector of colors
colors <- rainbow(nrow(termination_df))

# Create the line plot with colors and labels
ggplot(termination_df, aes(x = Year, y = Count, color = Year, group = 1)) +
  geom_point() +
  labs(title = "Number of Terminations by Year",
       x = "Year",
       y = "Count") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  geom_text(aes(label = Count), vjust = -0.5, size = 3)
```

Figure 3.5.5 Difference between terminated and newly hired employees by year codes (part 1)



```r
# ANALYSIS 5-4: Difference: between the number of terminated employees and new hires by year

        # Convert "Termination date" column to date format and extract the Year
        assignment_csv$terminationdate_key <- as.Date(assignment_csv$terminationdate_key, format = "%m/%d/%Y")
        assignment_csv$TerminationYear <- format(assignment_csv$terminationdate_key, "%Y")

        # Count the number of terminations by year
        termination_counts <- table(assignment_csv$TerminationYear)

        # Convert the count table to a data frame
        termination_df <- as.data.frame(termination_counts)
        colnames(termination_df) <- c("Year", "TerminationCount")

        # Convert "Hire Date" column to date format and extract the Year
        assignment_csv$orighiredate_key <- as.Date(assignment_csv$orighiredate_key, format = "%m/%d/%Y")
        assignment_csv$Year <- format(assignment_csv$orighiredate_key, "%Y")

        # Count the number of new hires by year
        new_hires_count <- table(assignment_csv$Year)

        # Convert the count table to a data frame
        new_hires_df <- as.data.frame(new_hires_count)
        colnames(new_hires_df) <- c("Year", "NewHiresCount")

        # Combine termination and new hires data frames
        combined_counts <- merge(termination_df, new_hires_df, by = "Year", all = TRUE)
```

Figure 3.5.6: Difference between terminated and newly hired employees by year codes (part 2)



```
# Set the count values to 0 for missing years
combined_counts[is.na(combined_counts)] <- 0

# Convert Year to factor and arrange levels in ascending order
combined_counts$Year <- factor(combined_counts$Year, levels = sort(unique(combined_counts$Year)))

# Create the line plot with combined data
ggplot(combined_counts, aes(x = Year)) +
  geom_line(aes(y = TerminationCount, color = "Termination"), linetype = "dashed") +
  geom_line(aes(y = NewHiresCount, color = "New Hires")) +
  geom_point(aes(y = TerminationCount, color = "Termination")) +
  geom_point(aes(y = NewHiresCount, color = "New Hires")) +
  geom_text(aes(y = TerminationCount, label = TerminationCount), vjust = -0.5, size = 3, color = "red") +
  geom_text(aes(y = NewHiresCount, label = NewHiresCount), vjust = -0.5, size = 3, color = "blue") +
  labs(title = "Comparison of Terminations and New Hires by Year",
       x = "Year",
       y = "Count") +
  scale_color_manual(values = c("Termination" = "red", "New Hires" = "blue")) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 3.5.7: Difference between terminated and newly hired employees by year codes (part 3)

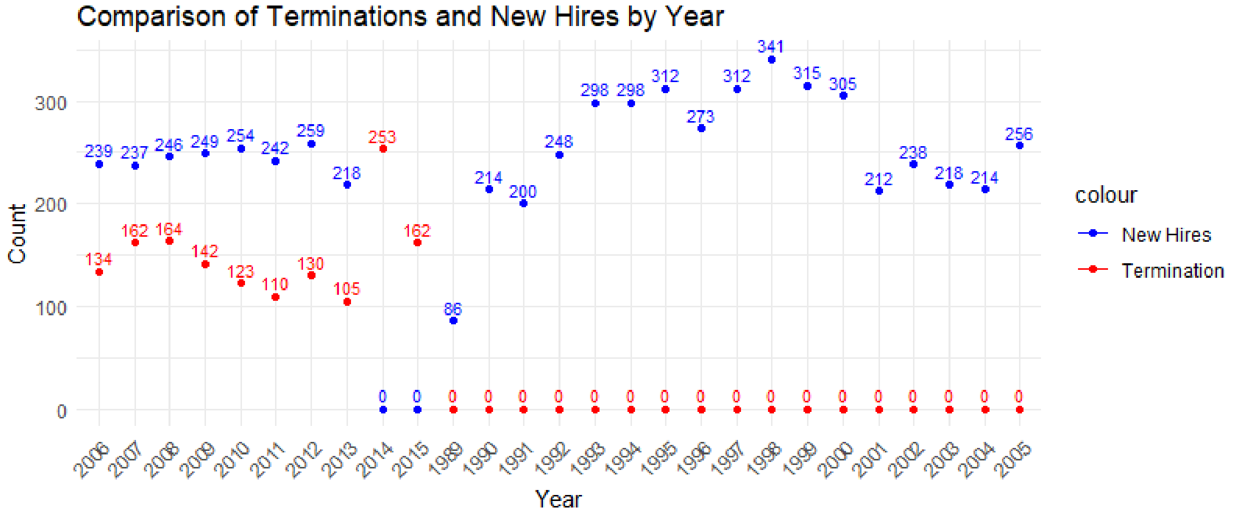


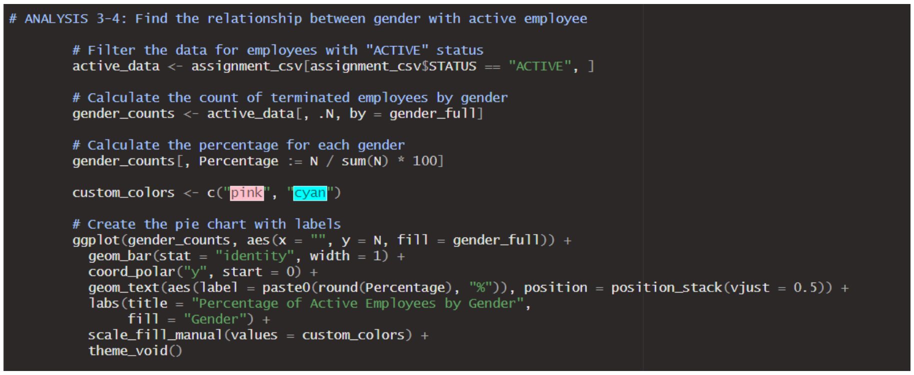
Figure 3.5.8: Difference between terminated and newly hired employees by year point graph

The scatter graph above compares the number of employee turnover and hired by year. The graph above shows that the employee turnover and hiring ratio has always been positive. This indicates that the company have a strong Human Resource team and knows how to maintain employee retention. However, in the years 2014 and 2015 the ratio is negative. This can happen due to a couple of reasons, for instance, some stores closed down due to low customer volume. Therefore, employees leave the organisation.

4.0 Extra/Additional Features

4.1 Graphs and Charts

4.1.1 Pie chart



```
# ANALYSIS 3-4: Find the relationship between gender with active employee

        # Filter the data for employees with "ACTIVE" status
        active_data <- assignment_csv[assignment_csv$STATUS == "ACTIVE", ]

        # Calculate the count of terminated employees by gender
        gender_counts <- active_data[, .N, by = gender_full]

        # Calculate the percentage for each gender
        gender_counts[, Percentage := N / sum(N) * 100]

        custom_colors <- c("pink", "cyan")

        # Create the pie chart with labels
        ggplot(gender_counts, aes(x = "", y = N, fill = gender_full)) +
          geom_bar(stat = "identity", width = 1) +
          coord_polar("y", start = 0) +
          geom_text(aes(label = paste0(round(Percentage), "%")), position = position_stack(vjust = 0.5)) +
          labs(title = "Percentage of Active Employees by Gender",
               fill = "Gender") +
          scale_fill_manual(values = custom_colors) +
          theme_void()
```

Figure 4.1.1 Pie Chart Creation code

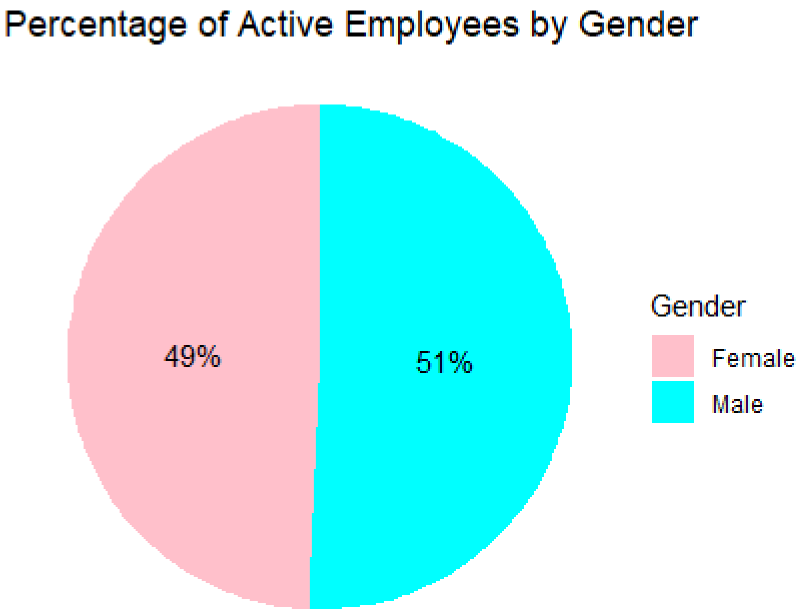
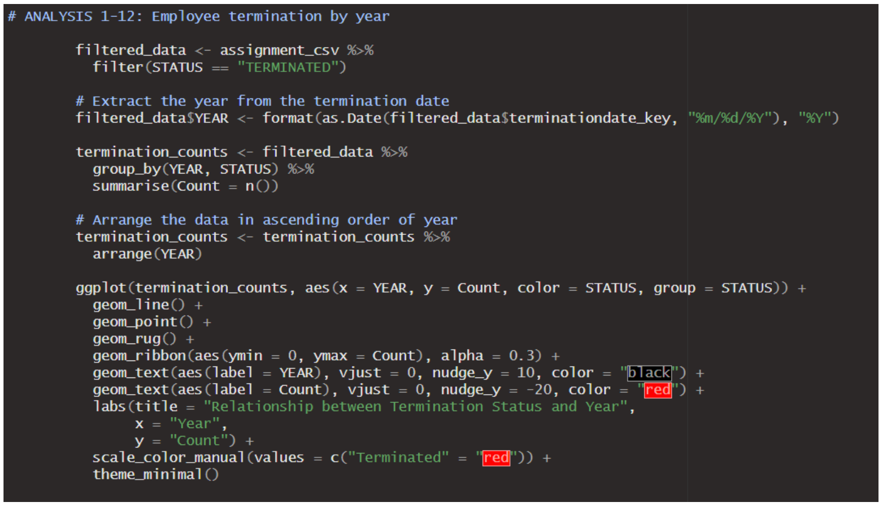


Figure 4.1.2 Pie Chart example

4.1.2 Ribbon graph



```r
# ANALYSIS 1-12: Employee termination by year

        filtered_data <- assignment_csv %>%
          filter(STATUS == "TERMINATED")

        # Extract the year from the termination date
        filtered_data$YEAR <- format(as.Date(filtered_data$terminationdate_key, "%m/%d/%Y"), "%Y")

        termination_counts <- filtered_data %>%
          group_by(YEAR, STATUS) %>%
          summarise(Count = n())

        # Arrange the data in ascending order of year
        termination_counts <- termination_counts %>%
          arrange(YEAR)

        ggplot(termination_counts, aes(x = YEAR, y = Count, color = STATUS, group = STATUS)) +
          geom_line() +
          geom_point() +
          geom_rug() +
          geom_ribbon(aes(ymin = 0, ymax = Count), alpha = 0.3) +
          geom_text(aes(label = YEAR), vjust = 0, nudge_y = 10, color = "black") +
          geom_text(aes(label = Count), vjust = 0, nudge_y = -20, color = "red") +
          labs(title = "Relationship between Termination Status and Year",
               x = "Year",
               y = "Count") +
          scale_color_manual(values = c("Terminated" = "red")) +
          theme_minimal()
```

Figure 4.1.3 Ribbon graph creation code

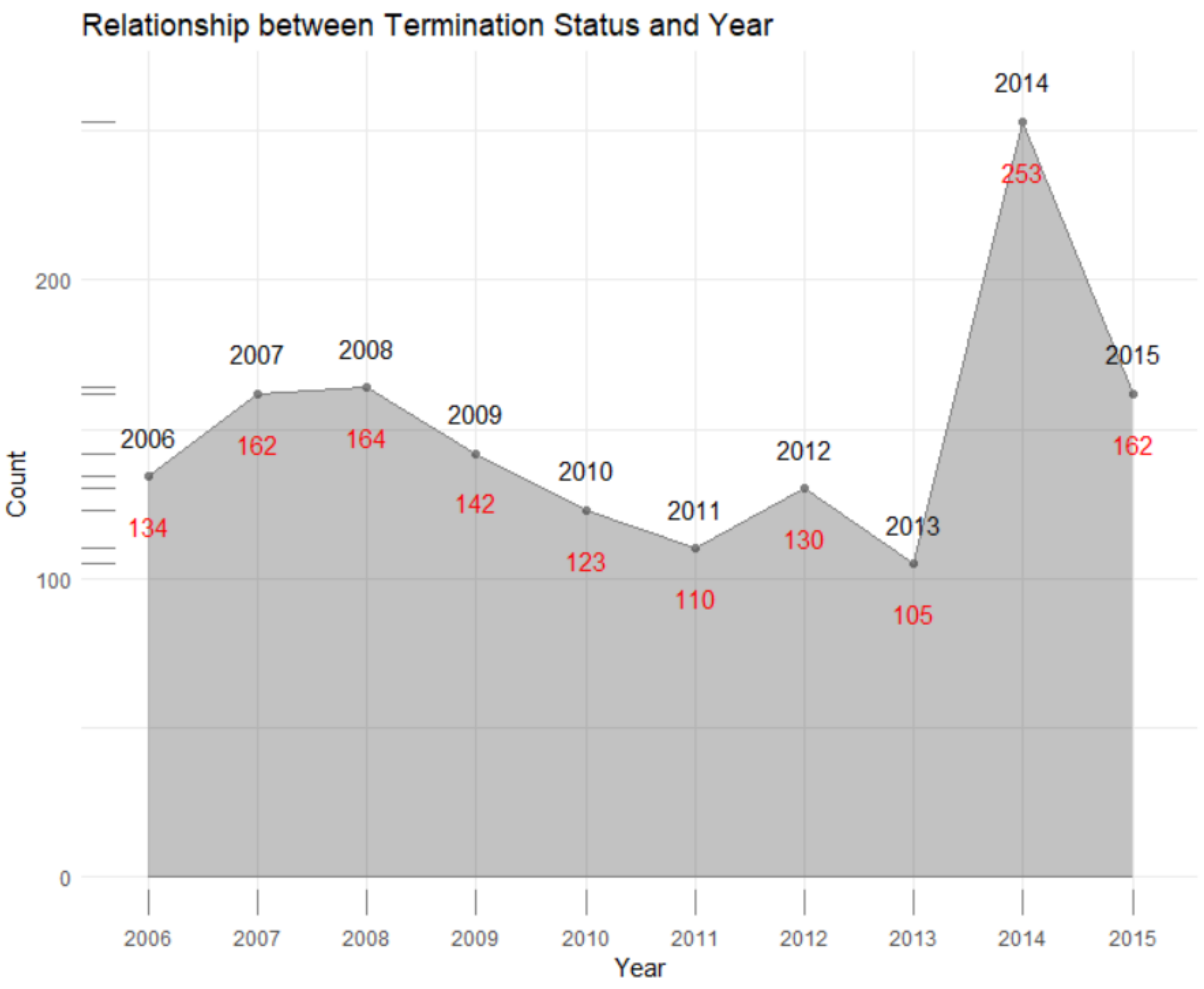
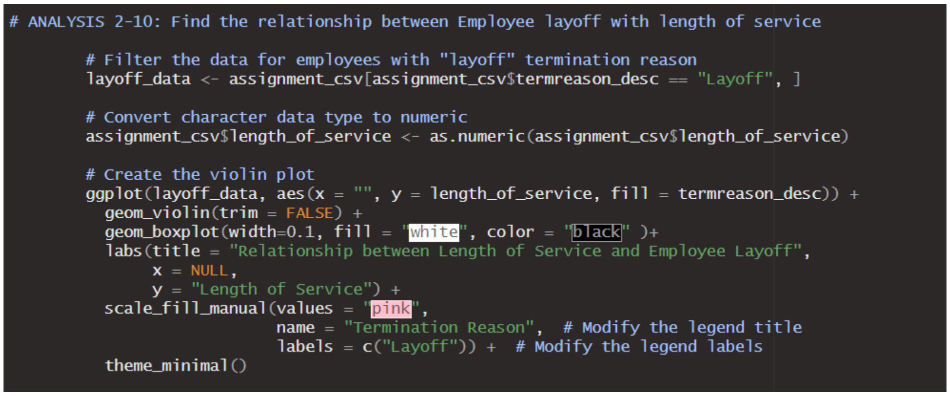


Figure 4.1.4 Ribbon graph example

Ribbon graph is for showing y intervals that have the value of y max and y min for each x value in the graph. In an instance, in the graph above the ribbon graph is the gray shading below the geom_line(), and it is the interval from 0 to each y for each x value.

4.1.3 Violin chart



```
# ANALYSIS 2-10: Find the relationship between Employee layoff with length of service

        # Filter the data for employees with "layoff" termination reason
        layoff_data <- assignment_csv[assignment_csv$termreason_desc == "Layoff", ]

        # Convert character data type to numeric
        assignment_csv$length_of_service <- as.numeric(assignment_csv$length_of_service)

        # Create the violin plot
        ggplot(layoff_data, aes(x = "", y = length_of_service, fill = termreason_desc)) +
          geom_violin(trim = FALSE) +
          geom_boxplot(width=0.1, fill = "white", color = "black" )+
          labs(title = "Relationship between Length of Service and Employee Layoff",
               x = NULL,
               y = "Length of Service") +
          scale_fill_manual(values = "pink",
                            name = "Termination Reason",  # Modify the legend title
                            labels = c("Layoff")) +  # Modify the legend labels
          theme_minimal()
```

Figure 4.1.5 Viiolin graph creation code

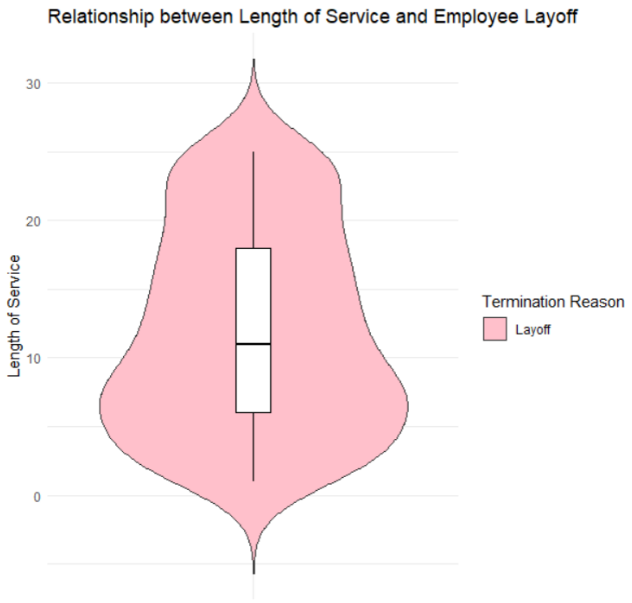
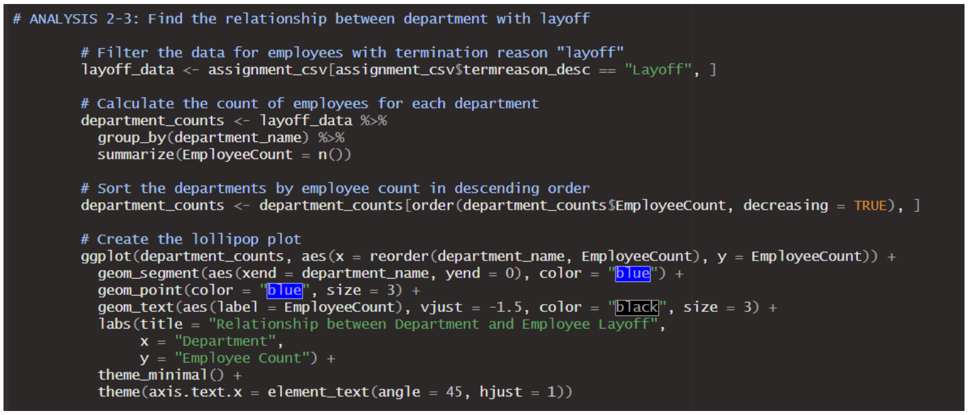


Figure 4.1.6 Violin graph creation example

Violin is a brief display about continuous distribution.

4.1.4 Lollipop chart



```
# ANALYSIS 2-3: Find the relationship between department with layoff

        # Filter the data for employees with termination reason "layoff"
        layoff_data <- assignment_csv[assignment_csv$termreason_desc == "Layoff", ]

        # Calculate the count of employees for each department
        department_counts <- layoff_data %>%
          group_by(department_name) %>%
          summarize(EmployeeCount = n())

        # Sort the departments by employee count in descending order
        department_counts <- department_counts[order(department_counts$EmployeeCount, decreasing = TRUE), ]

        # Create the lollipop plot
        ggplot(department_counts, aes(x = reorder(department_name, EmployeeCount), y = EmployeeCount)) +
          geom_segment(aes(xend = department_name, yend = 0), color = "blue") +
          geom_point(color = "blue", size = 3) +
          geom_text(aes(label = EmployeeCount), vjust = -1.5, color = "black", size = 3) +
          labs(title = "Relationship between Department and Employee Layoff",
               x = "Department",
               y = "Employee Count") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 4.1.7 Lolipop graph creation code

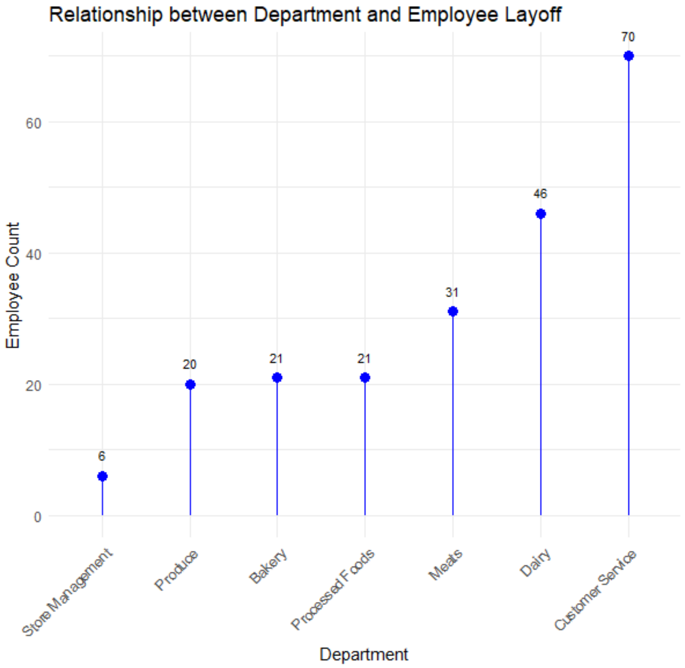
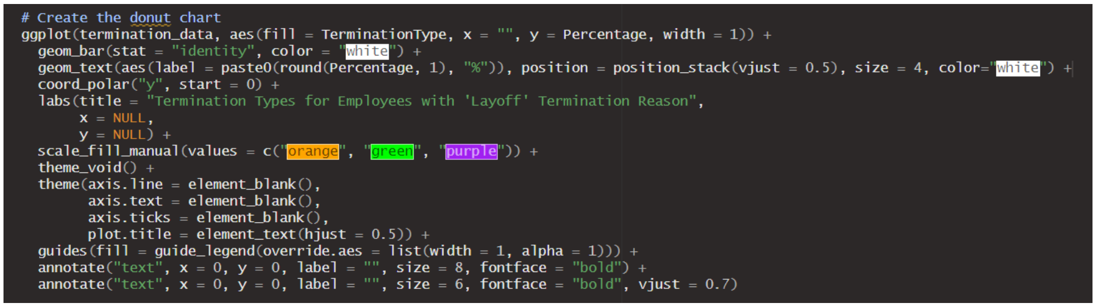


Figure 4.1.8 Lolipop graph creation example

A lollipop plot is basically a barplot, where the bar is transformed in a line and a dot . It shows the relationship between a numeric and a categorical variable.

4.1.5 Donut chart



```
# Create the donut chart
ggplot(termination_data, aes(fill = TerminationType, x = "", y = Percentage, width = 1)) +
  geom_bar(stat = "identity", color = "white") +
  geom_text(aes(label = paste0(round(Percentage, 1), "%")), position = position_stack(vjust = 0.5), size = 4, color="white") +
  coord_polar("y", start = 0) +
  labs(title = "Termination Types for Employees with 'Layoff' Termination Reason",
       x = NULL,
       y = NULL) +
  scale_fill_manual(values = c("orange", "green", "purple")) +
  theme_void() +
  theme(axis.line = element_blank(),
        axis.text = element_blank(),
        axis.ticks = element_blank(),
        plot.title = element_text(hjust = 0.5)) +
  guides(fill = guide_legend(override.aes = list(width = 1, alpha = 1))) +
  annotate("text", x = 0, y = 0, label = "", size = 8, fontface = "bold") +
  annotate("text", x = 0, y = 0, label = "", size = 6, fontface = "bold", vjust = 0.7)
```

Figure 4.1.9 Donut chart creation code

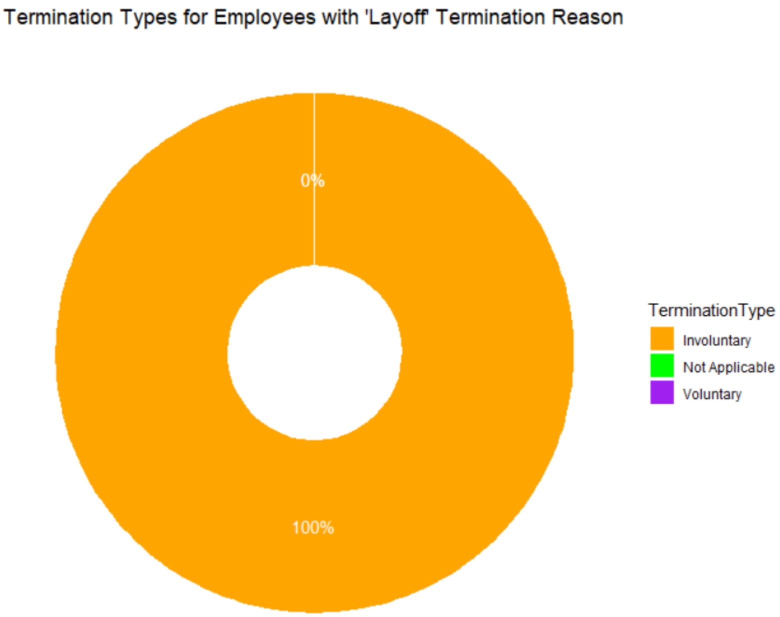
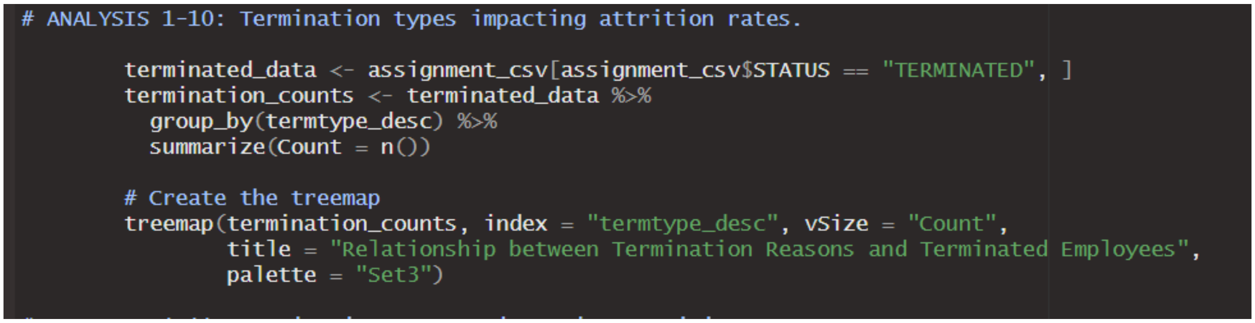


Figure 4.1.10 Donut chart example

Similar to pie chart. Doughnut charts can include many data series and are used to illustrate how parts relate to the total. A ring is added to a doughnut chart for each data series that is plotted.

4.1.6 Treemap



```
# ANALYSIS 1-10: Termination types impacting attrition rates.

        terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]
        termination_counts <- terminated_data %>%
          group_by(termtype_desc) %>%
          summarize(Count = n())

        # Create the treemap
        treemap(termination_counts, index = "termtype_desc", vSize = "Count",
                title = "Relationship between Termination Reasons and Terminated Employees",
                palette = "Set3")
```

Figure 4.1.11: Treemap creation codes

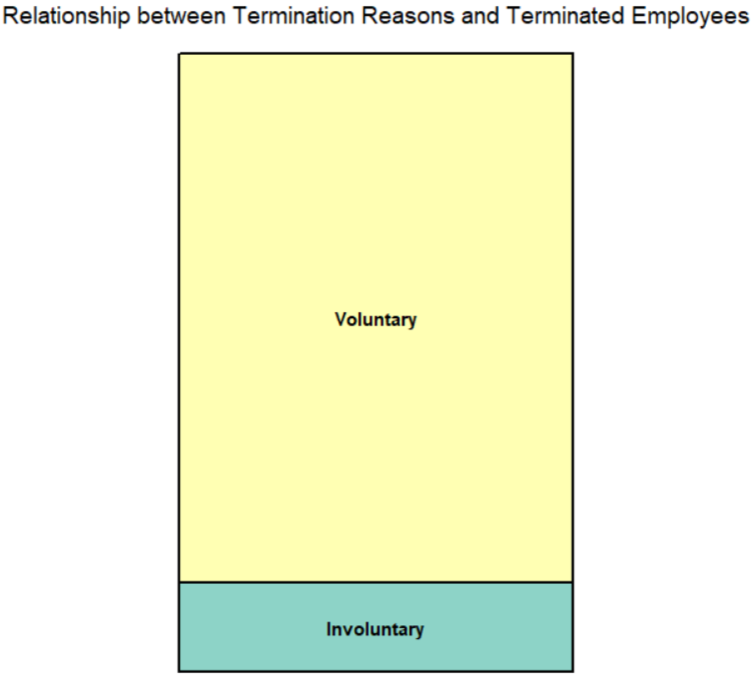
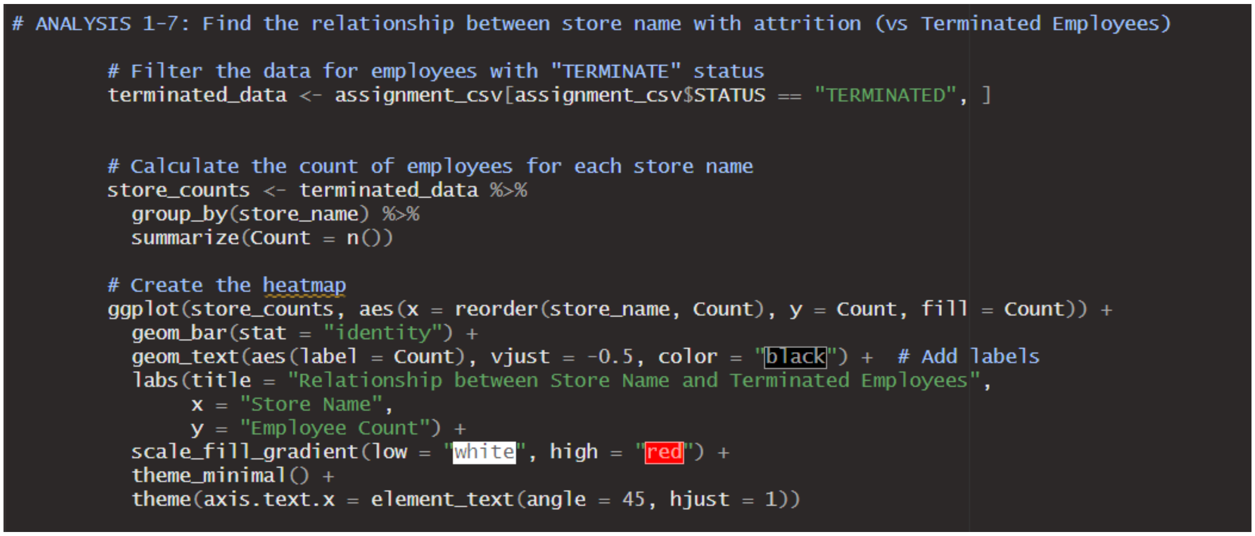


Figure 4.1.12 Treemap example

The treemap acts as a rectangle-nested visualisation. These rectangles are arranged in a hierarchy, or "tree," to represent certain categories within a chosen dimension.

4.1.7 Heatmaps



```
# ANALYSIS 1-7: Find the relationship between store name with attrition (vs Terminated Employees)

        # Filter the data for employees with "TERMINATE" status
        terminated_data <- assignment_csv[assignment_csv$STATUS == "TERMINATED", ]


        # Calculate the count of employees for each store name
        store_counts <- terminated_data %>%
          group_by(store_name) %>%
          summarize(Count = n())

        # Create the heatmap
        ggplot(store_counts, aes(x = reorder(store_name, Count), y = Count, fill = Count)) +
          geom_bar(stat = "identity") +
          geom_text(aes(label = Count), vjust = -0.5, color = "black") +  # Add labels
          labs(title = "Relationship between Store Name and Terminated Employees",
               x = "Store Name",
               y = "Employee Count") +
          scale_fill_gradient(low = "white", high = "red") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 4.1.13: Heatmapcreation codes

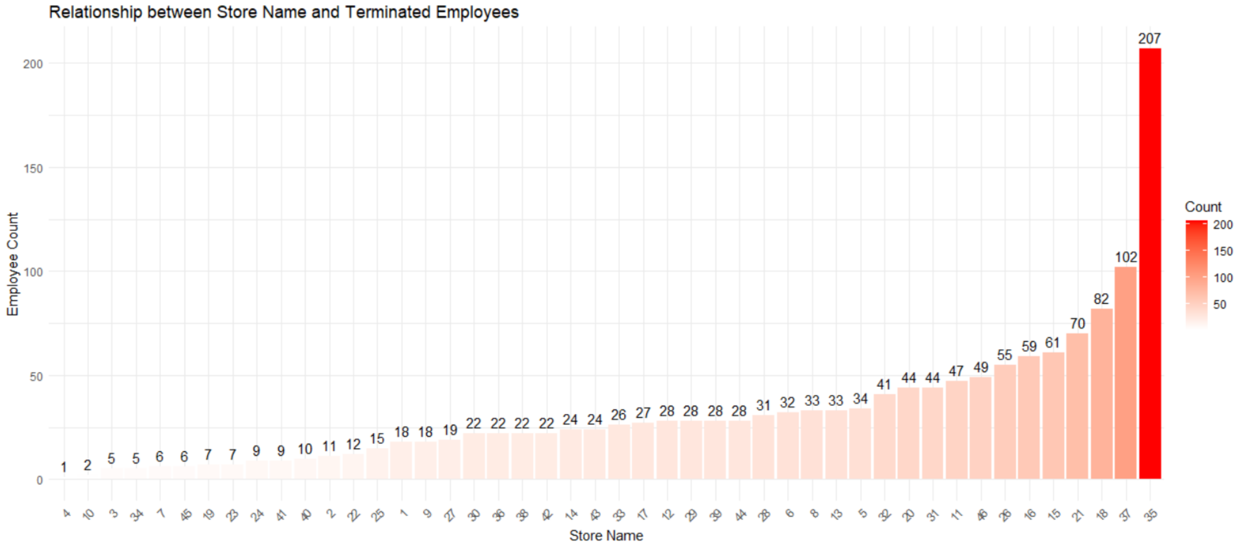
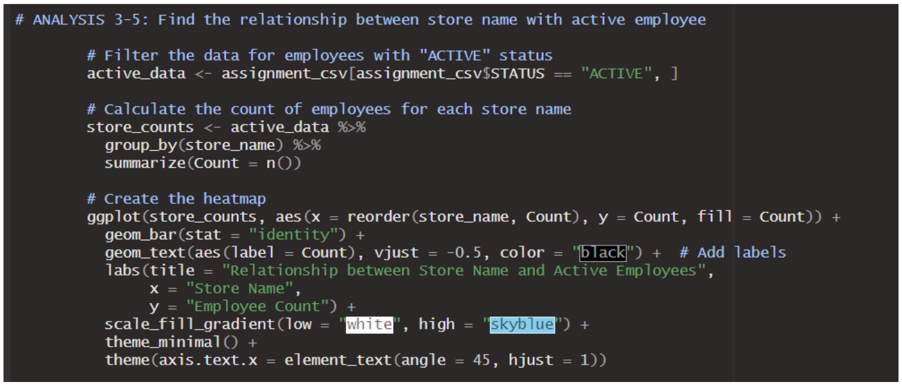


Figure 4.1.14 Heatmap example

A heat map is a 2D data visualisation method that uses colours to indicate specific values in a data collection.

4.2 Colorings and text

4.2.1 Scale gradient



```
# ANALYSIS 3-5: Find the relationship between store name with active employee

        # Filter the data for employees with "ACTIVE" status
        active_data <- assignment_csv[assignment_csv$STATUS == "ACTIVE", ]

        # Calculate the count of employees for each store name
        store_counts <- active_data %>%
          group_by(store_name) %>%
          summarize(Count = n())

        # Create the heatmap
        ggplot(store_counts, aes(x = reorder(store_name, Count), y = Count, fill = Count)) +
          geom_bar(stat = "identity") +
          geom_text(aes(label = Count), vjust = -0.5, color = "black") +  # Add labels
          labs(title = "Relationship between Store Name and Active Employees",
               x = "Store Name",
               y = "Employee Count") +
          scale_fill_gradient(low = "white", high = "skyblue") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Figure 4.2.1 Scale_gradient usage code

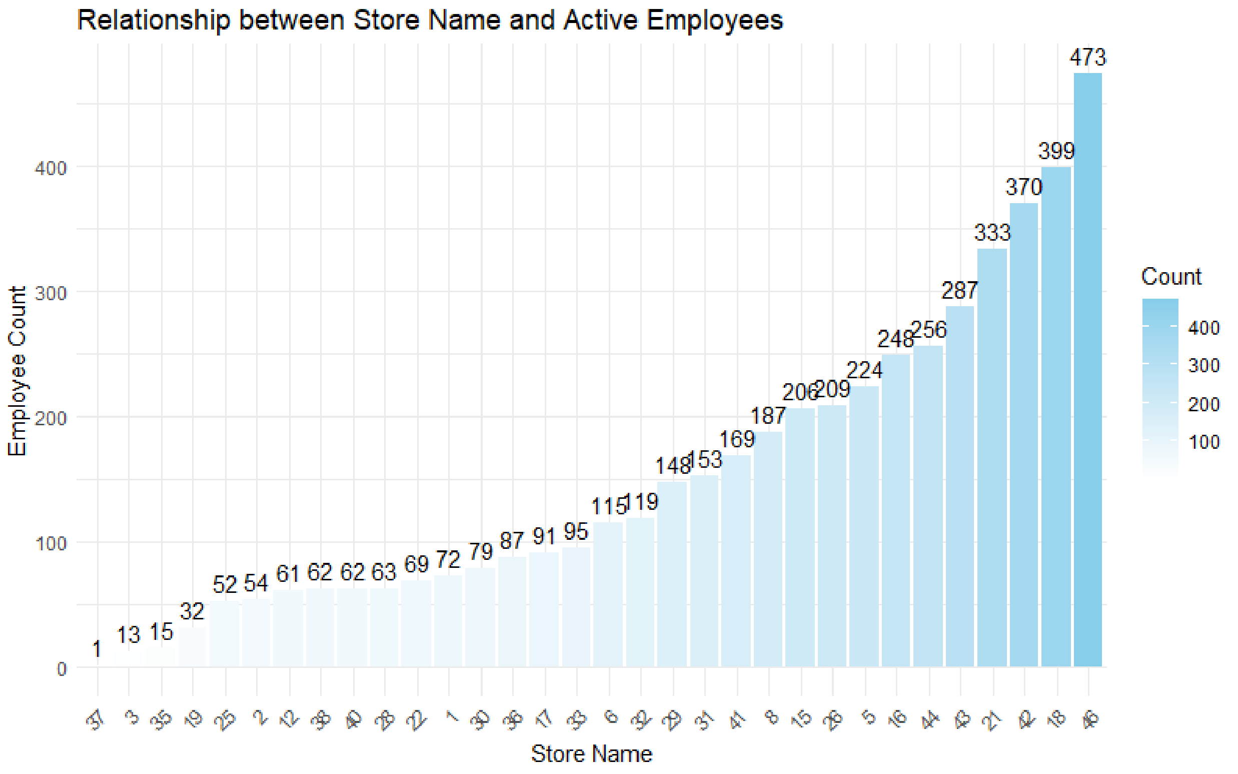
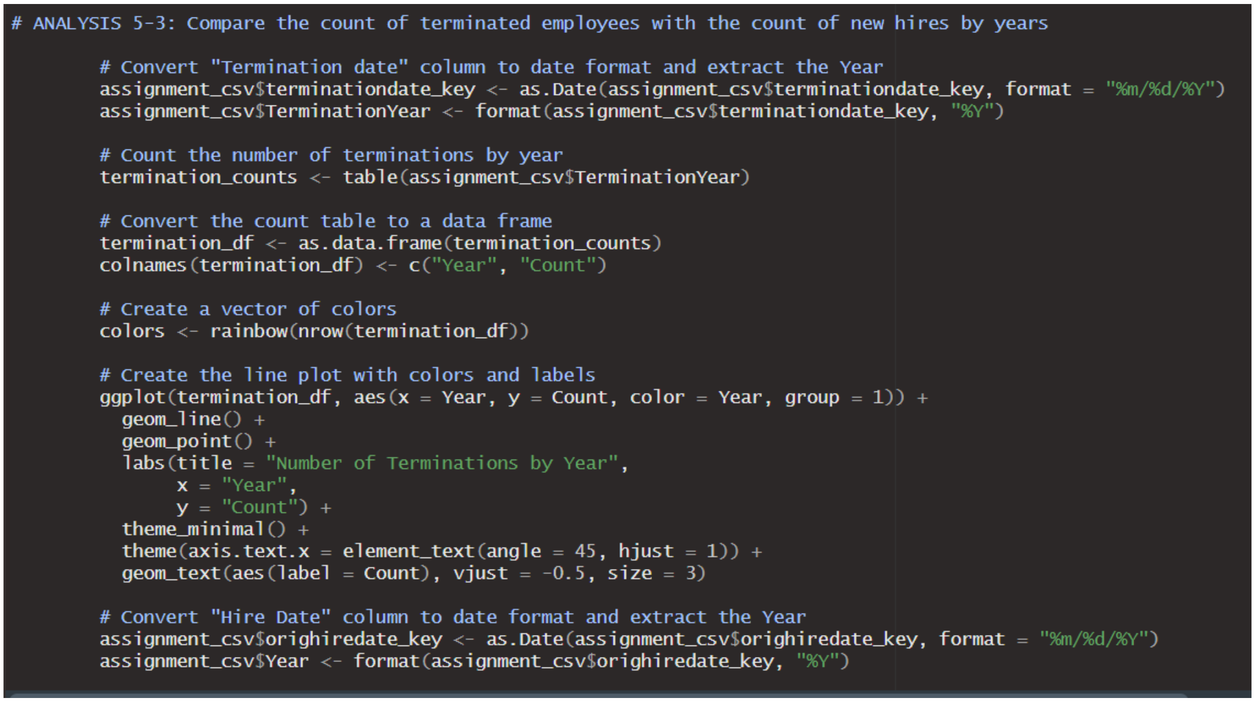


Figure 4.2.2 Scale_gradient usage example

4.2.2 theme_minimal()



```
# ANALYSIS 5-3: Compare the count of terminated employees with the count of new hires by years

        # Convert "Termination date" column to date format and extract the Year
        assignment_csv$terminationdate_key <- as.Date(assignment_csv$terminationdate_key, format = "%m/%d/%Y")
        assignment_csv$TerminationYear <- format(assignment_csv$terminationdate_key, "%Y")

        # Count the number of terminations by year
        termination_counts <- table(assignment_csv$TerminationYear)

        # Convert the count table to a data frame
        termination_df <- as.data.frame(termination_counts)
        colnames(termination_df) <- c("Year", "Count")

        # Create a vector of colors
        colors <- rainbow(nrow(termination_df))

        # Create the line plot with colors and labels
        ggplot(termination_df, aes(x = Year, y = Count, color = Year, group = 1)) +
          geom_line() +
          geom_point() +
          labs(title = "Number of Terminations by Year",
               x = "Year",
               y = "Count") +
          theme_minimal() +
          theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
          geom_text(aes(label = Count), vjust = -0.5, size = 3)

        # Convert "Hire Date" column to date format and extract the Year
        assignment_csv$orighiredate_key <- as.Date(assignment_csv$orighiredate_key, format = "%m/%d/%Y")
        assignment_csv$Year <- format(assignment_csv$orighiredate_key, "%Y")
```

Figure 4.2.3: Theme_minimal() + Axis.text.x + Angle + HJust usage code.

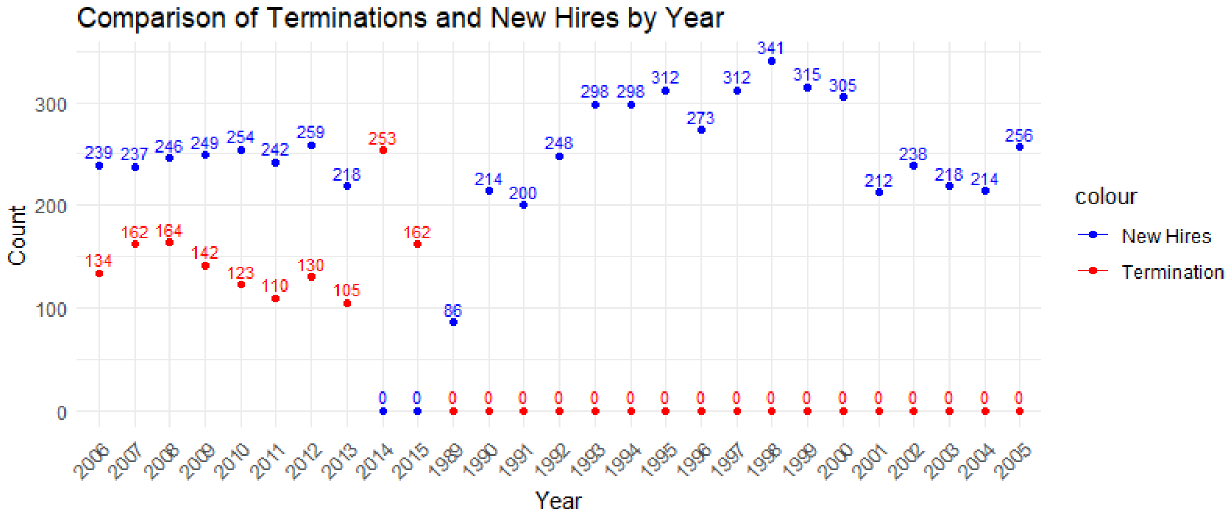
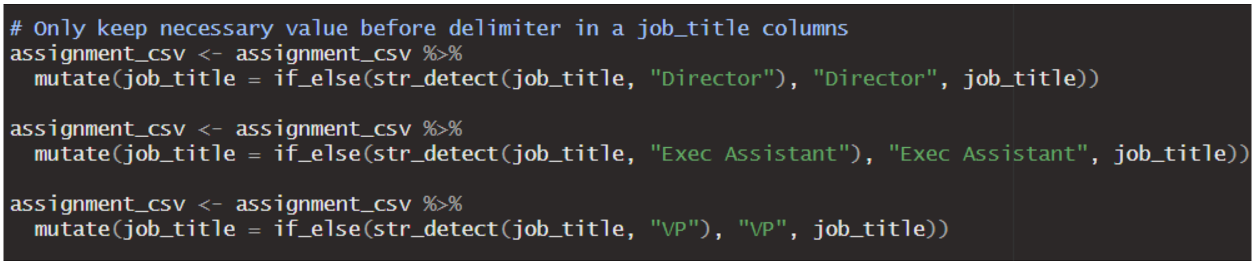


Figure 4.2.3: Theme_minimal() + Axis.text.x + Angle + HJust example.

4.3 Data Pre-prosessing

4.3.1 str_detect()

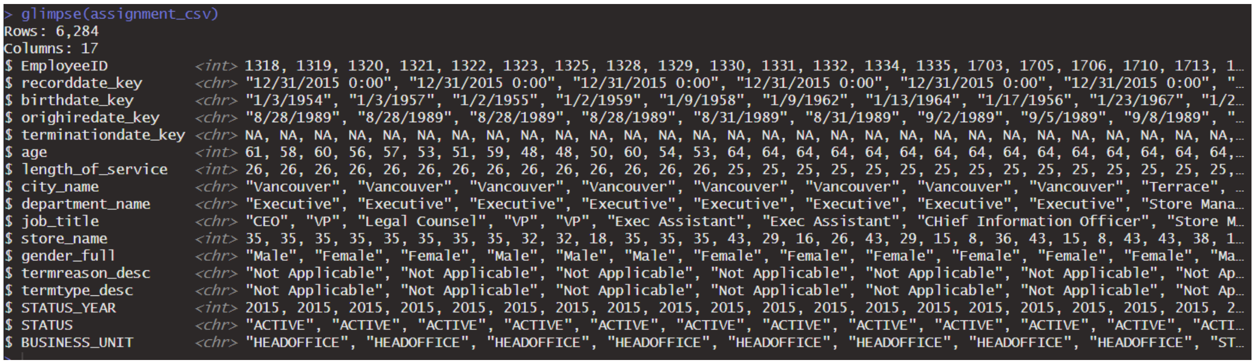


```
# Only keep necessary value before delimiter in a job_title columns
assignment_csv <- assignment_csv %>%
  mutate(job_title = if_else(str_detect(job_title, "Director"), "Director", job_title))

assignment_csv <- assignment_csv %>%
  mutate(job_title = if_else(str_detect(job_title, "Exec Assistant"), "Exec Assistant", job_title))

assignment_csv <- assignment_csv %>%
  mutate(job_title = if_else(str_detect(job_title, "VP"), "VP", job_title))
```

\4.3.1 str_detect usage example

4.4 Data Exploration

4.4.1 glimpse()



```
> glimpse(assignment_csv)
Rows: 6,284
Columns: 17
$ EmployeeID          <int> 1318, 1319, 1320, 1321, 1322, 1323, 1325, 1328, 1329, 1330, 1331, 1332, 1334, 1335, 1703, 1705, 1706, 1710, 1713, 1…
$ recorddate_key      <chr> "12/31/2015 0:00", "12/31/2015 0:00", "12/31/2015 0:00", "12/31/2015 0:00", "12/31/2015 0:00", "12/31/2015 0:00", "…
$ birthdate_key       <chr> "1/3/1954", "1/3/1957", "1/2/1955", "1/2/1959", "1/9/1958", "1/9/1962", "1/13/1964", "1/17/1956", "1/23/1967", "1/2…
$ orighiredate_key    <chr> "8/28/1989", "8/28/1989", "8/28/1989", "8/28/1989", "8/31/1989", "8/31/1989", "9/2/1989", "9/5/1989", "9/8/1989", "…
$ terminationdate_key <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,…
$ age                 <int> 61, 58, 60, 56, 57, 53, 51, 59, 48, 48, 50, 60, 54, 53, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64, 64,…
$ length_of_service   <int> 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 26, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25, 25,…
$ city_name           <chr> "Vancouver", "Vancouver", "Vancouver", "Vancouver", "Vancouver", "Vancouver", "Vancouver", "Vancouver", "Terrace", …
$ department_name     <chr> "Executive", "Executive", "Executive", "Executive", "Executive", "Executive", "Executive", "Executive", "Store Mana…
$ job_title           <chr> "CEO", "VP", "Legal Counsel", "VP", "VP", "Exec Assistant", "Exec Assistant", "CHief Information Officer", "Store M…
$ store_name          <int> 35, 35, 35, 35, 35, 35, 35, 35, 32, 32, 18, 35, 35, 35, 43, 29, 16, 26, 43, 29, 15, 8, 36, 43, 15, 8, 43, 43, 38, 1…
$ gender_full         <chr> "Male", "Female", "Female", "Male", "Male", "Male", "Female", "Female", "Female", "Female", "Female", "Female", "Ma…
$ termreason_desc     <chr> "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Ap…
$ termtype_desc       <chr> "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Applicable", "Not Ap…
$ STATUS_YEAR         <int> 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2…
$ STATUS              <chr> "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTIVE", "ACTI…
$ BUSINESS_UNIT       <chr> "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "HEADOFFICE", "ST…
```

4.4.1 glimpse() usage example

5.0 Conclusion

To conclude, this assignment has given me insight into using the R programming language on the RStudio platform to program for data analysis purposes. I demonstrated my learning outcome in this assignment by exploring the given data set on employee attrition. As a result, 5 questions along with 38 analyses were produced to explore the relationship between columns in the dataset.

Question 1: What are the reasons for employee termination?

The majority of employee terminations are voluntary, mostly due to retirement followed by resignation and then layoff. My analysis shows that many factors could be attributed to an employee's desire to leave the company. Some employees seek challenges by switching companies. Additionally, employees may be terminated due to performance issues or disciplinary reasons, such as misconduct or negligence.

Question 2: What are the reasons for employee layoffs?

In the related analysis, it shows that there could be many reasons for employee layoffs, including poor employee performance, ethics, reduced operation costs, implementation of technology or automation, gender (e.g. pregnancy or maternity leave) and age (e.g. fresh graduates with no experience) for instance.

Question 3: Why do employees stay with the organization?

Based on the analysis, it shows that employees may stay with an organization for different reasons, such as job stability as most employees have been with the organization for many years. Job satisfaction is an important factor that can influence employee retention. Employees that do not feel valued or satisfied with their job may be more likely to leave the organization. Employee recognition and appreciation can be a key factor in retaining employees and creating a positive work environment.

Question 4: What factors influence employee promotion opportunities?

With the findings from my analysis, it shows that employee promotion opportunities depend on a variety of factors, such as work experience. Additionally, some organizations consider seniority.

Finally, the number of open positions and the level of competition among applicants can also affect an individual's chances of promotion.

Question 5: Are terminated employees replaced by equal numbers of newly hired hires?

My analysis shows the company has a positive and healthy turnover and hire rate. However, this is not always the case. In some of these years, companies may replace terminated employees with fewer new hires and rely more on existing staff to fill the gap. Additionally, existing staff may have to take on more responsibilities or hours to fill the gap.

6.0 References

Carron, J. (2021, December 13). *Violin Plots 101: Visualizing Distribution and Probability Density | Mode*. Mode.com. https://mode.com/blog/violin-plot-examples/

David. (2022, September 2). *How to Make a Donut Chart in ggplot*. Rfortherestofus.com. https://rfortherestofus.com/2022/09/how-to-make-a-donut-chart-in-ggplot/

ggplot2. (n.d.-a). *Histograms and frequency polygons — geom_freqpoly*. Ggplot2.Tidyverse.org. Retrieved May 1, 2023, from https://ggplot2.tidyverse.org/reference/geom_histogram.html#:~:text=Histograms%20(%20geom_histogram()%20)%20display%20the

ggplot2. (n.d.-b). *Ribbons and area plots — geom_ribbon*. Ggplot2.Tidyverse.org. Retrieved May 11, 2023, from https://ggplot2.tidyverse.org/reference/geom_ribbon.html

Holtz, Y. (2018). *Lollipop | the R Graph Gallery*. R-Graph-Gallery.com. https://r-graph-gallery.com/lollipop-plot.html#:~:text=A%20lollipop%20plot%20is%20basically

Schork, J. (n.d.). *R Change Colors of Ranges in ggplot2 Heatmap | Gradient & Categories*. Statistics Globe. Retrieved May 8, 2023, from https://statisticsglobe.com/change-colors-of-ranges-in-ggplot2-heatmap-r

stringr. (n.d.). *Simple, Consistent Wrappers for Common String Operations*. Stringr.tidyverse.org. Retrieved May 11, 2023, from https://stringr.tidyverse.org/

tutorialspoint. (2019). *R - Pie Charts - Tutorialspoint*. Tutorialspoint.com. https://www.tutorialspoint.com/r/r_pie_charts.htm