Q4

February 26, 2025

0.1 Data Imputation

```
[1]: ### Preprocessing
```

```
[2]: library(tidyverse)
```

```
 Attaching core tidyverse packages
 tidyverse 2.0.0
 dplyr     1.1.4      readr     2.1.5
 forcats   1.0.0      stringr   1.5.1
 ggplot2   3.5.1      tibble    3.2.1
 lubridate 1.9.4      tidyr     1.3.1
 purrr     1.0.4
 Conflicts

tidyverse_conflicts()
 dplyr::filter() masks stats::filter()
 dplyr::lag()    masks stats::lag()
 Use the conflicted package
(<http://conflicted.r-lib.org/>) to force all conflicts to
become errors
```

```
[3]: setwd("/home/asus/content/Notes/Semester 4/FDN Lab/Experiments/Experiment 3")
```

```
[4]: df <- data.frame(
       ID = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10),
       Name = c("Alice", "Bob", NA, "David", "Emma", "Frank", NA, "Hannah", "Ian",␣
      ↪"Jack"),
       Age = c(25, NA, 30, 29, NA, 35, 40, NA, 50, 27),
       Salary = c(50000, 60000, 55000, NA, 70000, 75000, 80000, 65000, NA, 72000),
       Score = c(80, 90, NA, 85, 88, 92, NA, 77, 95, Inf)
     )
```

Convert NaN and Inf values to NA before applying imputation.

```
[5]: df <- df %>%
       mutate_all(~ ifelse(. == Inf | . == -Inf, NA, .)) %>%
       mutate_all(~ ifelse(is.nan(.), NA, .))
```

Remove rows with missing values using na.omit(df).

[6]:

```
df_no_na <- na.omit(df)  # Remove rows with any NA
```

Drop columns where more than 50% of data is missing.

[7]:

```
df <- df[, colSums(is.na(df)) < (0.5 * nrow(df))]
```

Replace all NA values with 0 for numerical columns.

[8]:

```
df[sapply(df, is.numeric)] <- lapply(df[sapply(df, is.numeric)], function(x) {␣
 ↪replace(x, is.na(x), 0) })
```

Replace missing values in Age with the mean.

[9]:

```
df$Age[is.na(df$Age)] <- mean(df$Age, na.rm = TRUE)
```

Replace missing values in Salary with the median.

[10]:

```
df$Salary[is.na(df$Salary)] <- median(df$Salary, na.rm = TRUE)
```

Replace missing Name values with the most frequent name (Mode)

[11]:

```
fill_mode <- function(x) {
  mode_value <- names(sort(table(x), decreasing = TRUE))[1]
  x[is.na(x)] <- mode_value
  return(x)
}

df$Name <- fill_mode(df$Name)  # Apply mode function to Name column
```

Summary

[12]:

```
summary(df)  # Check if missing values are handled
```

```
       ID            Name                Age            Salary     
 Min.   : 1.00   Length:10          Min.   : 0.00   Min.   :    0  
 1st Qu.: 3.25   Class :character   1st Qu.: 6.25   1st Qu.:51250  
 Median : 5.50   Mode  :character   Median :28.00   Median :62500  
 Mean   : 5.50                      Mean   :23.60   Mean   :52700  
 3rd Qu.: 7.75                      3rd Qu.:33.75   3rd Qu.:71500  
 Max.   :10.00                      Max.   :50.00   Max.   :80000  
     Score      
 Min.   : 0.00  
 1st Qu.:19.25  
 Median :82.50  
 Mean   :60.70  
 3rd Qu.:89.50  
 Max.   :95.00  
```