Sheth L.U.J & Sir M.V College Of Science

Subject :- Data Analysis with SAS / SPSS /R

Module 2 Practical no 1

Aim:- Generating descriptive statistics using PROC MEANS (SAS), Descriptive Statistics (SPSS), and summary() or describe() (R).

Output :-

```
> library("dplyr")

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union

> library("psych")
> df <- read_csv("elevator_traffic_dataset (1).csv")
Rows: 300 Columns: 7
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr  (2): direction, peak_hour
dbl  (4): floor_requested, wait_time_seconds, people_count, load_percent
dttm (1): timestamp

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
> print("First 6 rows of the dataset")
[1] "First 6 rows of the dataset"
> head(df)
# A tibble: 6 × 7
  timestamp           floor_requested wait_time_seconds direction people_count peak_hour load_percent
  <dttm>                        <dbl>             <dbl> <chr>              <dbl> <chr>            <dbl>
1 2025-01-01 11:31:00               5              16.5 Down                   5 No                  53
2 2025-01-01 10:22:00              12              19.0 Down                   3 Yes                 20
3 2025-01-01 16:25:00               4              19.4 Down                   1 No                  10
4 2025-01-01 10:37:00               8               3.22 Up                    4 Yes                 31
5 2025-01-01 07:31:00               6              17.3 Up                     4 No                  50
6 2025-01-01 11:48:00               1              13.0 Down                   5 No                  57
> print("Last 6 rows of the dataset")
[1] "Last 6 rows of the dataset"
> tail(df)
# A tibble: 6 × 7
  timestamp           floor_requested wait_time_seconds direction people_count peak_hour load_percent
  <dttm>                        <dbl>             <dbl> <chr>              <dbl> <chr>            <dbl>
1 2025-01-01 12:52:00              12              15.2 Up                     3 No                  43
2 2025-01-01 06:50:00              11               8.89 Up                    4 No                  19
3 2025-01-01 11:03:00               2              19.3 Up                     6 No                  59
4 2025-01-01 11:00:00               7              12.6 Down                   1 No                  10
5 2025-01-01 15:54:00               4              16.6 Up                     6 No                  69
6 2025-01-01 14:14:00               5               4.41 Up                    3 No                  20
> print("Structure of dataset:")
[1] "Structure of dataset:"
> str(df)
spc_tbl_ [300 × 7] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ timestamp       : POSIXct[1:300], format: "2025-01-01 11:31:00" "2025-01-01 10:22:00" "2025-01-01 16:25:00" "2025-01-01 10:37:00" ...
 $ floor_requested : num [1:300] 5 12 4 8 6 1 11 10 11 8 ...
 $ wait_time_seconds: num [1:300] 16.53 18.98 19.35 3.22 17.32 ...
 $ direction       : chr [1:300] "Down" "Down" "Down" "Up" ...
 $ people_count    : num [1:300] 5 3 1 4 4 5 1 4 4 1 ...
 $ peak_hour       : chr [1:300] "No" "Yes" "No" "Yes" ...
 $ load_percent    : num [1:300] 53 20 10 31 50 57 14 49 46 10 ...
 - attr(*, "spec")=
  .. cols(
  ..   timestamp = col_datetime(format = ""),
  ..   floor_requested = col_double(),
  ..   wait_time_seconds = col_double(),
  ..   direction = col_character(),
  ..   people_count = col_double(),
  ..   peak_hour = col_character(),
  ..   load_percent = col_double()
  .. )
 - attr(*, "problems")=<externalptr>
> print("Column names:")
[1] "Column names:"
> colnames(df)
[1] "timestamp"        "floor_requested"  "wait_time_seconds" "direction"        "people_count"     "peak_hour"        "load_percent"
> print("Dataset dimensions (Rows, Columns):")
[1] "Dataset dimensions (Rows, Columns):"
> dim(df)
[1] 300   7
> df$timestamp <- as.POSIXct(df$timestamp)
> df$direction  <- as.factor(df$direction)
> df$peak_hour  <- as.factor(df$peak_hour)
```


Name :- Vinod Mali

Roll No :- S092

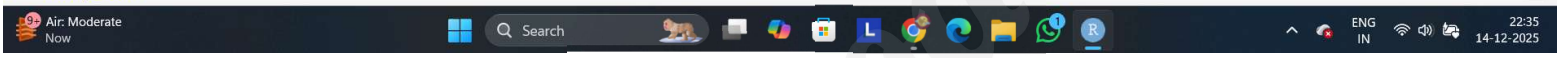
Sheth L.U.J & Sir M.V College Of Science

Subject :- Data Analysis with SAS / SPSS /R

Module 2 Practical no 1

```
> df$direction  <- as.factor(df$direction)
> df$peak_hour  <- as.factor(df$peak_hour)
> print("Missing values per column:")
[1] "Missing values per column:"
> colSums(is.na(df))
       timestamp   floor_requested wait_time_seconds         direction      people_count         peak_hour      load_percent
               0                0                0                 0                 0                 0                 0
> df <- na.omit(df)
> print("Dataset dimensions after cleaning:")
[1] "Dataset dimensions after cleaning:"
> dim(df)
[1] 300   7
> df$Load_Category <- ifelse(df$load_percent > 75,
+                            "High Load",
+                            "Normal Load")
> df$Load_Category <- as.factor(df$Load_Category)
> print("Load Category distribution:")
[1] "Load Category distribution:"
> table(df$Load_Category)

  High Load Normal Load
          6         294
> print("------------------------------------------------------------")
[1] "------------------------------------------------------------"
> print("DESCRIPTIVE STATISTICS USING summary()")
[1] "DESCRIPTIVE STATISTICS USING summary()"
> print("------------------------------------------------------------")
[1] "------------------------------------------------------------"
> summary(df)
   timestamp                   floor_requested  wait_time_seconds direction   people_count    peak_hour  load_percent       Load_Category
 Min.   :2025-01-01 06:00:00   Min.   : 1.000   Min.   : 0.160    Down:149   Min.   : 1.000   No :185   Min.   : 10.00   High Load  :  6
 1st Qu.:2025-01-01 08:52:15   1st Qu.: 4.000   1st Qu.: 9.055    Up  :151   1st Qu.: 2.000   Yes:115   1st Qu.: 21.00   Normal Load:294
 Median :2025-01-01 12:04:30   Median : 7.000   Median :12.490               Median : 4.000             Median : 35.00
 Mean   :2025-01-01 12:38:54   Mean   : 6.683   Mean   :12.227               Mean   : 3.713             Mean   : 36.25
 3rd Qu.:2025-01-01 16:37:30   3rd Qu.:10.000   3rd Qu.:15.238               3rd Qu.: 5.000             3rd Qu.: 50.00
 Max.   :2025-01-01 20:00:00   Max.   :12.000   Max.   :28.130               Max.   :10.000             Max.   :100.00
> numeric_data <- df %>% select(
+   floor_requested,
+   wait_time_seconds,
+   people_count,
+   load_percent
```




```
+ )
> summary(numeric_data)
 floor_requested  wait_time_seconds  people_count     load_percent
 Min.   : 1.000   Min.   : 0.160    Min.   : 1.000   Min.   : 10.00
 1st Qu.: 4.000   1st Qu.: 9.055    1st Qu.: 2.000   1st Qu.: 21.00
 Median : 7.000   Median :12.490    Median : 4.000   Median : 35.00
 Mean   : 6.683   Mean   :12.227    Mean   : 3.713   Mean   : 36.25
 3rd Qu.:10.000   3rd Qu.:15.238    3rd Qu.: 5.000   3rd Qu.: 50.00
 Max.   :12.000   Max.   :28.130    Max.   :10.000   Max.   :100.00
> print("------------------------------------------------------------")
[1] "------------------------------------------------------------"
> print("DESCRIPTIVE STATISTICS USING describe()")
[1] "DESCRIPTIVE STATISTICS USING describe()"
> print("------------------------------------------------------------")
[1] "------------------------------------------------------------"
> describe(numeric_data)
                vars   n  mean    sd median trimmed   mad   min    max range  skew kurtosis   se
floor_requested    1 300  6.68  3.46   7.00    6.72  4.45  1.00  12.00 11.00 -0.09    -1.17 0.20
wait_time_seconds  2 300 12.23  4.52  12.49   12.25  4.28  0.16  28.13 27.97 -0.01     0.07 0.26
people_count       3 300  3.71  1.89   4.00    3.62  1.48  1.00  10.00  9.00  0.42    -0.12 0.11
load_percent       4 300 36.25 19.45  35.00   35.05 20.76 10.00 100.00 90.00  0.45    -0.28 1.12
> print("------------------------------------------------------------")
[1] "------------------------------------------------------------"
> print("GROUP-WISE STATISTICS BY LOAD CATEGORY")
[1] "GROUP-WISE STATISTICS BY LOAD CATEGORY"
> print("------------------------------------------------------------")
[1] "------------------------------------------------------------"
> group_means <- df %>%
+   group_by(Load_Category) %>%
+   summarise(
+     Avg_Wait_Time   = mean(wait_time_seconds),
+     Avg_People      = mean(people_count),
+     Avg_Load        = mean(load_percent),
+     Avg_Floor       = mean(floor_requested)
+   )
> print(group_means)
# A tibble: 2 × 5
  Load_Category Avg_Wait_Time Avg_People Avg_Load Avg_Floor
  <fct>                <db1>      <db1>    <db1>     <db1>
1 High Load             13.8       8.83     89.7      6.17
```

Name :- Vinod Mali

Roll No :- S092

Sheth L.U.J & Sir M.V College Of Science

Subject :- Data Analysis with SAS / SPSS /R

Module 2 Practical no 1

```
2 Normal Load            12.2       3.61     35.2      6.69
> print("------------------------------------------------------------")
[1] "------------------------------------------------------------"
> print("ADDITIONAL STATISTICAL MEASURES")
[1] "ADDITIONAL STATISTICAL MEASURES"
> print("------------------------------------------------------------")
[1] "------------------------------------------------------------"
> # Variance
> variance_values <- sapply(numeric_data, var)
> # Standard deviation
> sd_values <- sapply(numeric_data, sd)
> print("Variance of numeric variables:")
[1] "Variance of numeric variables:"
> print(variance_values)
  floor_requested wait_time_seconds      people_count      load_percent
        11.942865         20.445000         3.589788       378.362040
> print("Standard deviation of numeric variables:")
[1] "Standard deviation of numeric variables:"
> print(sd_values)
  floor_requested wait_time_seconds      people_count      load_percent
         3.455845          4.521615          1.894674        19.451531
> print("------------------------------------------------------------")
[1] "------------------------------------------------------------"
> print("FREQUENCY DISTRIBUTION")
[1] "FREQUENCY DISTRIBUTION"
> print("------------------------------------------------------------")
[1] "------------------------------------------------------------"
> direction_freq <- table(df$direction)
> print(direction_freq)

Down   Up
 149  151
> peak_hour_freq <- table(df$peak_hour)
> print(peak_hour_freq)

 No Yes
185 115
> peak_hour_prop <- prop.table(peak_hour_freq)
> print(peak_hour_prop)
```


```
> peak_hour_prop <- prop.table(peak_hour_freq)
> print(peak_hour_prop)

       No       Yes
0.6166667 0.3833333
> write.csv(
+   describe(numeric_data),
+   "elevator_descriptive_statistics.csv",
+   row.names = TRUE
+ )
> # Save group-wise statistics
> write.csv(
+   group_means,
+   "elevator_groupwise_statistics.csv",
+   row.names = FALSE
+ )
> print("Elevator traffic data analysis completed successfully.")
[1] "Elevator traffic data analysis completed successfully."
>
>
>
>
>
>
> |
```

Name :- Vinod Mali

Roll No :- S092