

PRACTICAL NO 15

AIM: Generating basic summaries using str() or summary() (R)

The image displays two screenshots of the RStudio interface, showing the execution of R code to generate summaries of a dataset.

Top Screenshot: The Source pane shows the following code:

```
R - R452 - ~/
> # Movie data with mixed data types
> movies_df <- data.frame(
+   MovieID = 1:6,
+   Genre = c("Action", "Drama", "Action", "Comedy", "Drama", "Horror"),
+   WatchTime = c(120, 95, 150, NA, 110, 80), # NA included
+   Is_Premium = c(TRUE, FALSE, TRUE, FALSE, TRUE, FALSE),
+   User_Rating = c(4.7, 3.9, 4.8, 4.0, 3.5, 4.1)
+ )
> print("---- Movies Dataset Loaded ----")
[1] "---- Movies Dataset Loaded ----"
> print(movies_df)
  MovieID Genre WatchTime Is_Premium User_Rating
1      1 Action      120      TRUE      4.7
2      2 Drama       95     FALSE      3.9
3      3 Action      150      TRUE      4.8
4      4 Comedy      NA     FALSE      4.0
5      5 Drama      110      TRUE      3.5
6      6 Horror       80     FALSE      4.1
> print("---- Output of str() ----")
[1] "---- Output of str() ----"
> str(movies_df)
'data.frame':   6 obs. of  5 variables:
 $ MovieID   : int  1 2 3 4 5 6
 $ Genre     : chr  "Action" "Drama" "Action" "Comedy" ...
 $ WatchTime : num  120 95 150 NA 110 80
 $ Is_Premium: logi  TRUE FALSE TRUE FALSE TRUE FALSE
 $ User_Rating: num  4.7 3.9 4.8 4.0 3.5 4.1
> print("---- Output of summary() (Before Factor Conversion) ----")
[1] "---- Output of summary() (Before Factor Conversion) ----"
> summary(movies_df)
  MovieID      Genre      WatchTime      Is_Premium      User_Rating
Min.   :1.00      Length:6      Min.   : 80 Mode :logical Min.   :3.500
1st Qu.:2.25      Class :character 1st Qu.: 95 FALSE:3 1st Qu.:3.925
Median :3.50      Mode  :character Median :110 TRUE :3  Median :4.050
Mean   :3.50      Mean   :111 Mean   :4.167
3rd Qu.:4.75      3rd Qu.:120 3rd Qu.:4.550
Max.   :6.00      Max.   :150 Max.   :4.800
NA's   :1
> movies_df$Genre <- as.factor(movies_df$Genre)
> print("---- Output of summary() After Converting Genre to Factor ----")
[1] "---- Output of summary() After Converting Genre to Factor ----"
> summary(movies_df)
  MovieID      Genre      WatchTime      Is_Premium      User_Rating
Min.   :1.00      Length:6      Min.   : 80 Mode :logical Min.   :3.500
1st Qu.:2.25      Class :character 1st Qu.: 95 FALSE:3 1st Qu.:3.925
Median :3.50      Mode  :character Median :110 TRUE :3  Median :4.050
Mean   :3.50      Mean   :111 Mean   :4.167
3rd Qu.:4.75      3rd Qu.:120 3rd Qu.:4.550
Max.   :6.00      Max.   :150 Max.   :4.800
NA's   :1
> avg_rating <- mean(movies_df$User_Rating)
> max_watch <- max(movies_df$WatchTime, na.rm = TRUE) # Ignoring NA
> print(paste("Average User Rating:", avg_rating))
[1] "Average User Rating: 4.16666666666667"
> print(paste("Maximum Watch Time:", max_watch))
[1] "Maximum Watch Time: 150"
> |
```

The Environment pane on the right shows the loaded objects:

- movies_df: 6 obs. of 5 variables
- new_employees: 2 obs. of 3 variables
- normal_or_overweight: 166 obs. of 5 variables
- obese_class1: 20 obs. of 5 variables
- obese_people: 137 obs. of 5 variables
- older_unhealthy: 44 obs. of 5 variables
- orders_df: 5 obs. of 2 variables
- processed_orders: 5 obs. of 11 variables
- range_cols: 244 obs. of 7 variables
- selected_cols: 244 obs. of 3 variables

Bottom Screenshot: This screenshot shows the same code being executed again, with the Environment pane updated to reflect the new state of the workspace. The code is identical to the top screenshot, and the Environment pane shows the same objects as before.