Experiment - 3

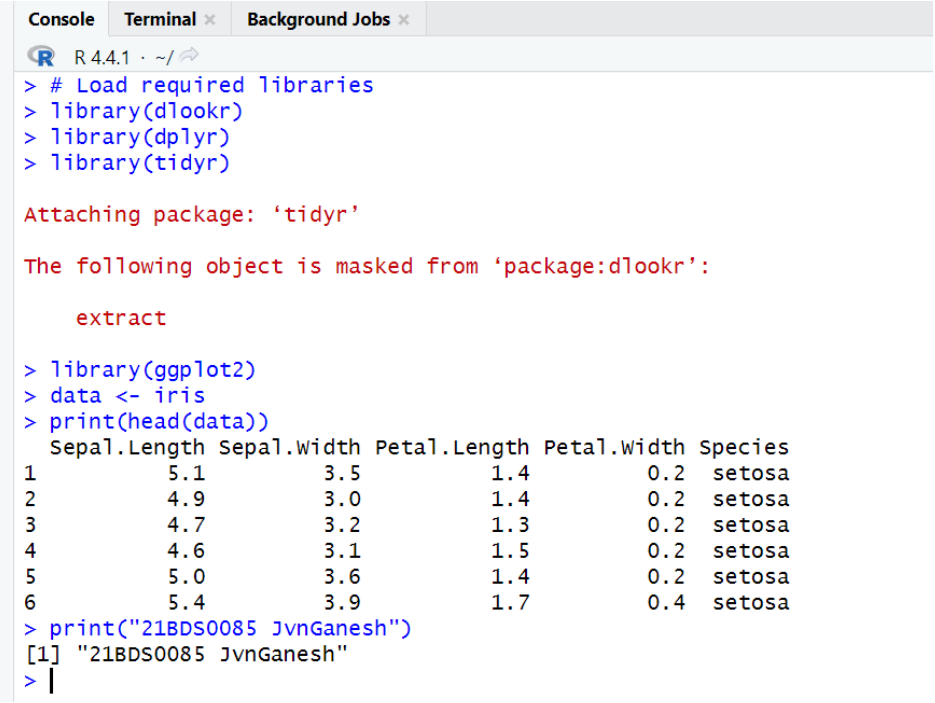
Exploratory Data Analysis

Name : JVN GANESH

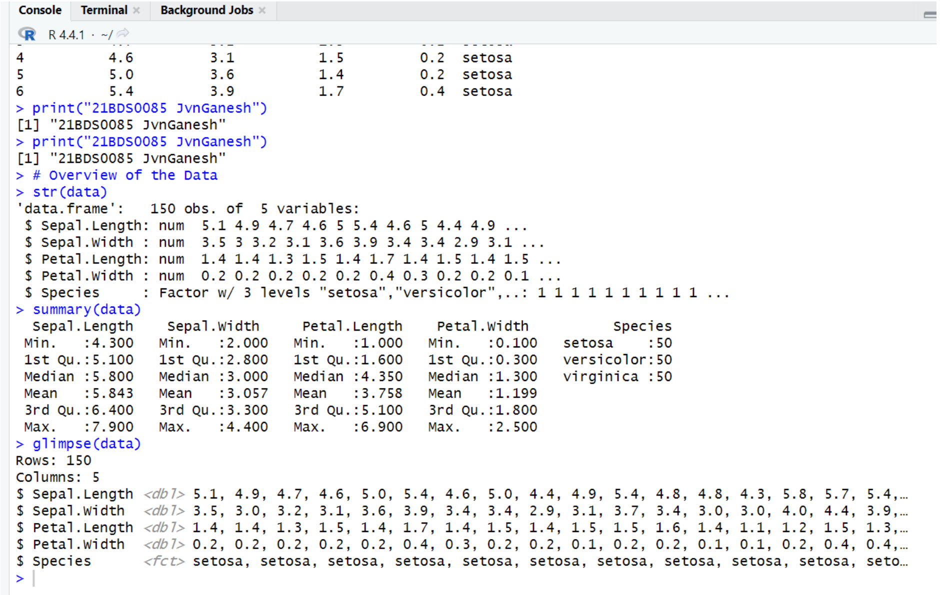
Roll N.O: 21BDS0085

EXP-3

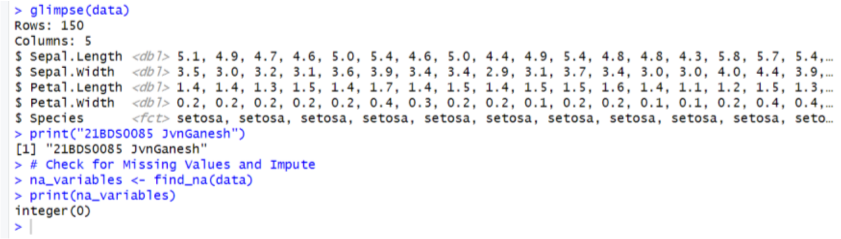
Importing data



```
> # Load required libraries
> library(dlookr)
> library(dplyr)
> library(tidyr)

Attaching package: ‘tidyr’

The following object is masked from ‘package:dlookr’:

    extract

> library(ggplot2)
> data <- iris
> print(head(data))
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> 
```



```
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Overview of the Data
> str(data)
'data.frame':	150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
> summary(data)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species  
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50  
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50  
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50  
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199                  
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800                  
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500                  
> glimpse(data)
Rows: 150
Columns: 5
$ Sepal.Length <dbl> 5.1, 4.9, 4.7, 4.6, 5.0, 5.4, 4.6, 5.0, 4.4, 4.9, 5.4, 4.8, 4.8, 4.3, 5.8, 5.7, 5.4,…
$ Sepal.Width  <dbl> 3.5, 3.0, 3.2, 3.1, 3.6, 3.9, 3.4, 3.4, 2.9, 3.1, 3.7, 3.4, 3.0, 3.0, 4.0, 4.4, 3.9,…
$ Petal.Length <dbl> 1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1.6, 1.4, 1.1, 1.2, 1.5, 1.3,…
$ Petal.Width  <dbl> 0.2, 0.2, 0.2, 0.2, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.2, 0.2, 0.1, 0.1, 0.2, 0.4, 0.4,…
$ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, seto…
> 
```

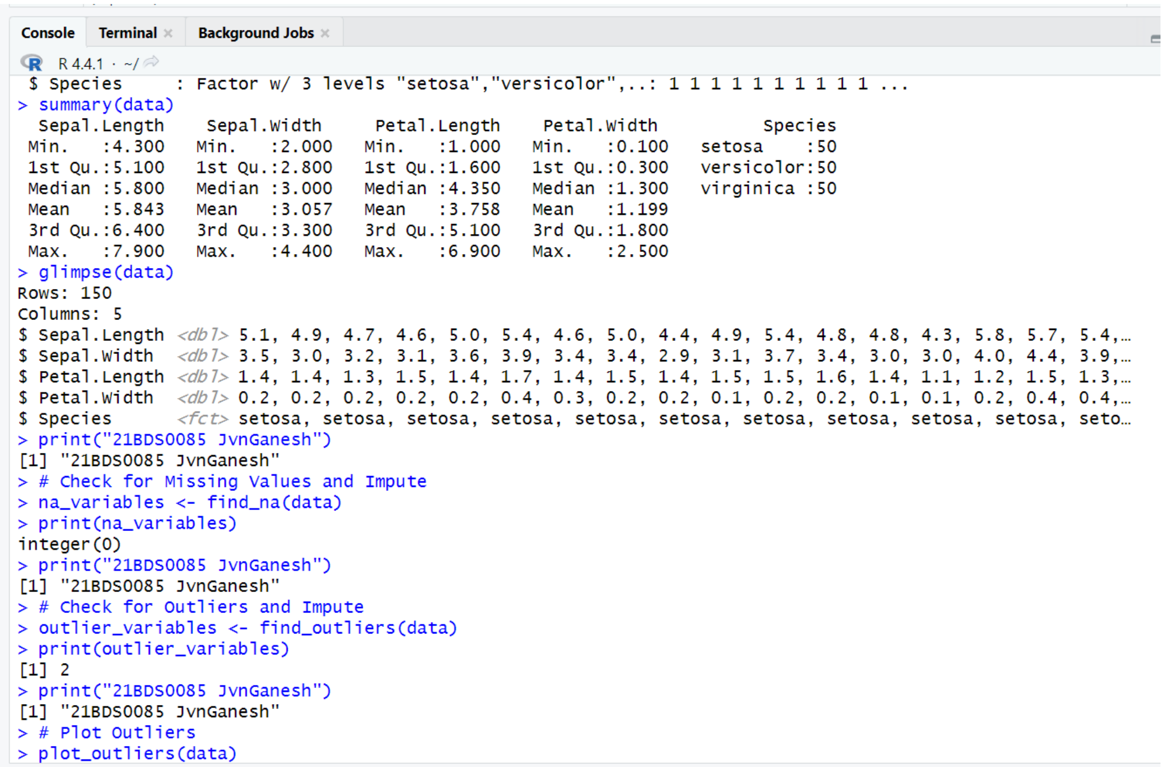


```
> glimpse(data)
Rows: 150
Columns: 5
$ Sepal.Length <dbl> 5.1, 4.9, 4.7, 4.6, 5.0, 5.4, 4.6, 5.0, 4.4, 4.9, 5.4, 4.8, 4.8, 4.3, 5.8, 5.7, 5.4,…
$ Sepal.Width  <dbl> 3.5, 3.0, 3.2, 3.1, 3.6, 3.9, 3.4, 3.4, 2.9, 3.1, 3.7, 3.4, 3.0, 3.0, 4.0, 4.4, 3.9,…
$ Petal.Length <dbl> 1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1.6, 1.4, 1.1, 1.2, 1.5, 1.3,…
$ Petal.Width  <dbl> 0.2, 0.2, 0.2, 0.2, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.2, 0.2, 0.1, 0.1, 0.2, 0.4, 0.4,…
$ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, seto…
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Check for Missing Values and Impute
> na_variables <- find_na(data)
> print(na_variables)
integer(0)
> 
```

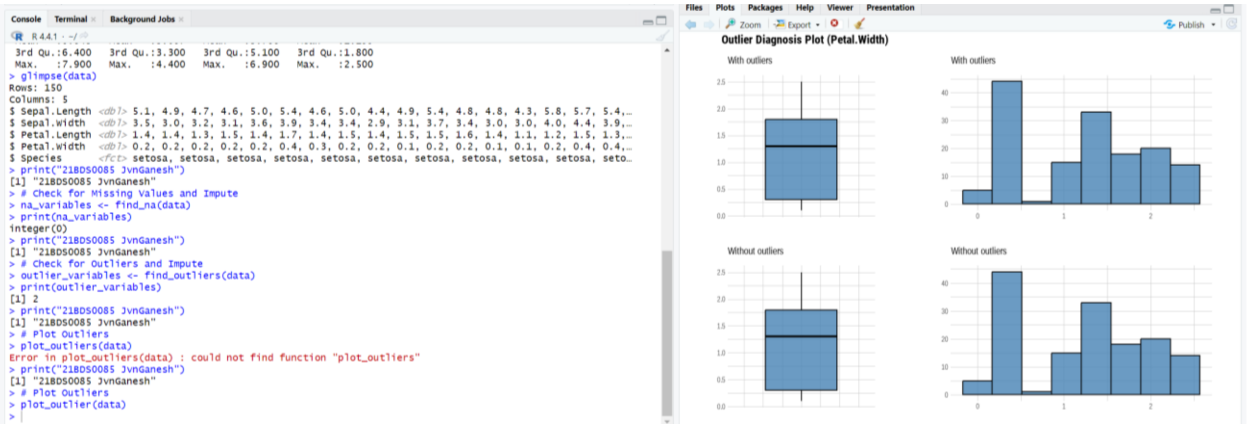
# Since the iris dataset does not contain missing values, no imputation is required here

# However, if there were missing values, the following line would impute them:

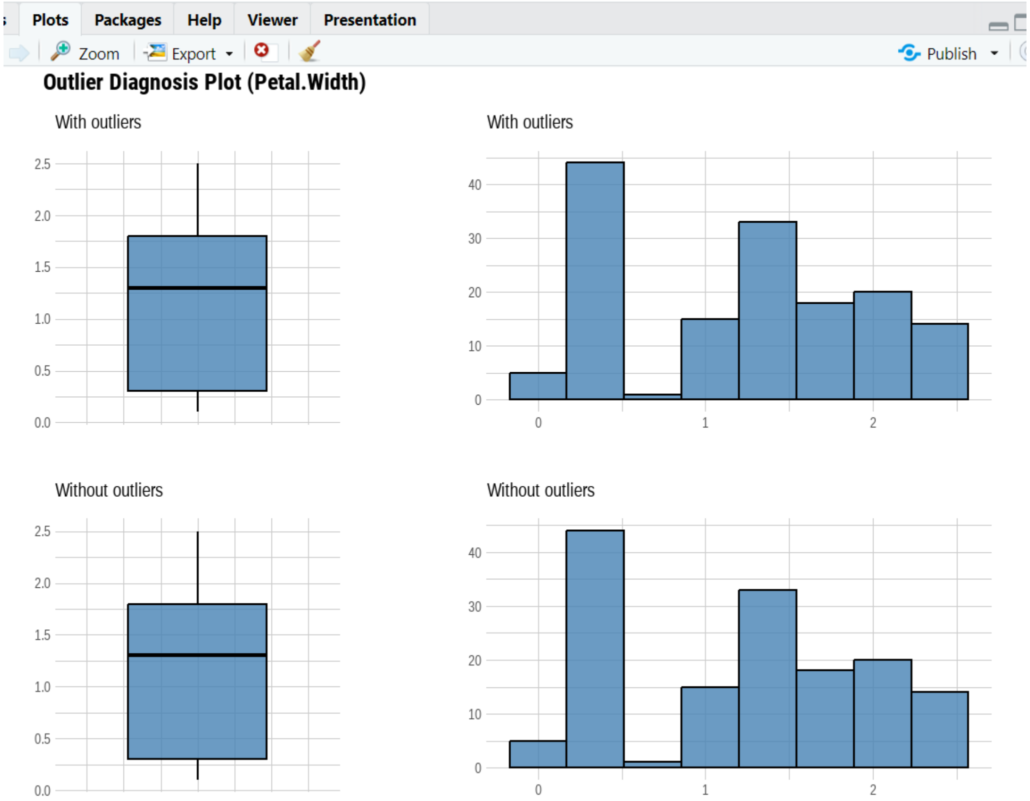
# data_imputed <- imputate_na(data)

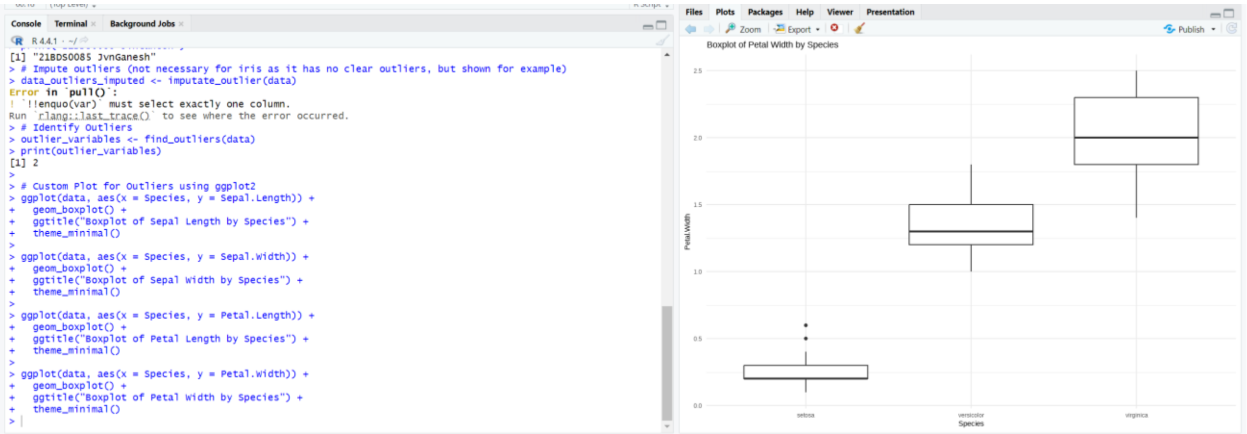


```
 $ Species      : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
> summary(data)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species  
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50  
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50  
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50  
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199                  
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800                  
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500                  
> glimpse(data)
Rows: 150
Columns: 5
$ Sepal.Length <dbl> 5.1, 4.9, 4.7, 4.6, 5.0, 5.4, 4.6, 5.0, 4.4, 4.9, 5.4, 4.8, 4.8, 4.3, 5.8, 5.7, 5.4,…
$ Sepal.Width  <dbl> 3.5, 3.0, 3.2, 3.1, 3.6, 3.9, 3.4, 3.4, 2.9, 3.1, 3.7, 3.4, 3.0, 3.0, 4.0, 4.4, 3.9,…
$ Petal.Length <dbl> 1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1.6, 1.4, 1.1, 1.2, 1.5, 1.3,…
$ Petal.Width  <dbl> 0.2, 0.2, 0.2, 0.2, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.2, 0.2, 0.1, 0.1, 0.2, 0.4, 0.4,…
$ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, seto…
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Check for Missing Values and Impute
> na_variables <- find_na(data)
> print(na_variables)
integer(0)
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Check for Outliers and Impute
> outlier_variables <- find_outliers(data)
> print(outlier_variables)
[1] 2
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Plot Outliers
> plot_outliers(data)
```

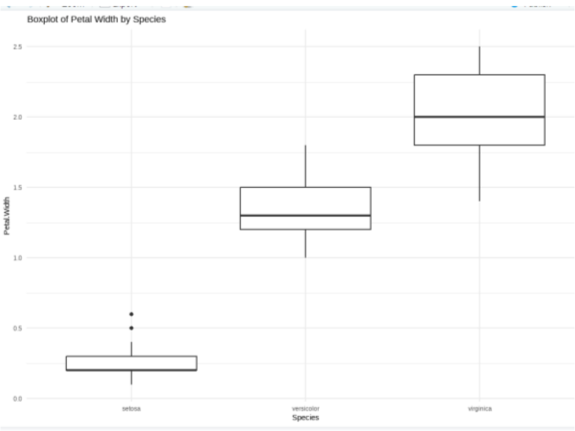


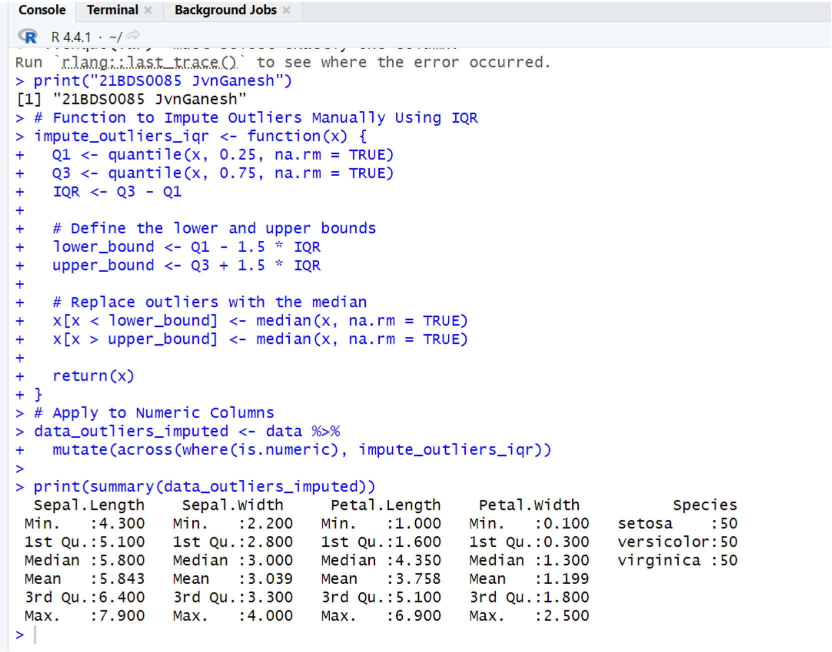
```
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800  
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500  
> glimpse(data)
Rows: 150
Columns: 5
$ Sepal.Length <dbl> 5.1, 4.9, 4.7, 4.6, 5.0, 5.4, 4.6, 5.0, 4.4, 4.9, 5.4, 4.8, 4.8, 4.3, 5.8, 5.7, 5.4,…
$ Sepal.Width  <dbl> 3.5, 3.0, 3.2, 3.1, 3.6, 3.9, 3.4, 3.4, 2.9, 3.1, 3.7, 3.4, 3.0, 3.0, 4.0, 4.4, 3.9,…
$ Petal.Length <dbl> 1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1.6, 1.4, 1.1, 1.2, 1.5, 1.3,…
$ Petal.Width  <dbl> 0.2, 0.2, 0.2, 0.2, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.2, 0.2, 0.1, 0.1, 0.2, 0.4, 0.4,…
$ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, setosa, seto…
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Check for Missing Values and Impute
> na_variables <- find_na(data)
> print(na_variables)
integer(0)
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Check for Outliers and Impute
> outlier_variables <- find_outliers(data)
> print(outlier_variables)
[1] 2
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Plot Outliers
> plot_outliers(data)
Error in plot_outliers(data) : could not find function "plot_outliers"
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Plot Outliers
> plot_outlier(data)
> 
```



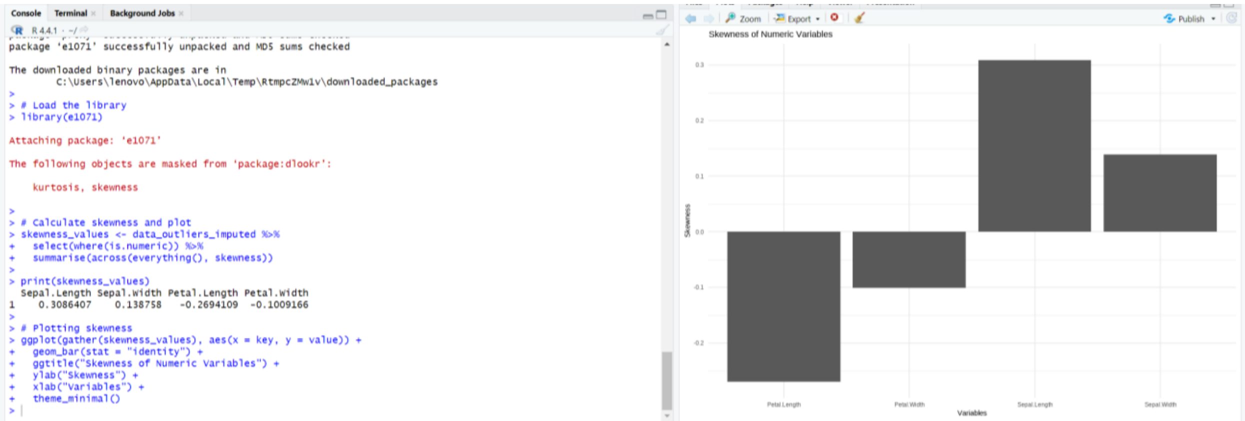


```
[1] "21BDS0085 JvnGanesh"
> # Impute outliers (not necessary for iris as it has no clear outliers, but shown for example)
> data_outliers_imputed <- imputate_outlier(data)
Error in `pull()`:
! `!!enquo(var)` must select exactly one column.
Run `rlang::last_trace()` to see where the error occurred.
> # Identify Outliers
> outlier_variables <- find_outliers(data)
> print(outlier_variables)
[1] 2
>
> # Custom Plot for Outliers using ggplot2
> ggplot(data, aes(x = Species, y = Sepal.Length)) +
+   geom_boxplot() +
+   ggtitle("Boxplot of Sepal Length by Species") +
+   theme_minimal()
>
> ggplot(data, aes(x = Species, y = Sepal.Width)) +
+   geom_boxplot() +
+   ggtitle("Boxplot of Sepal Width by Species") +
+   theme_minimal()
>
> ggplot(data, aes(x = Species, y = Petal.Length)) +
+   geom_boxplot() +
+   ggtitle("Boxplot of Petal Length by Species") +
+   theme_minimal()
>
> ggplot(data, aes(x = Species, y = Petal.Width)) +
+   geom_boxplot() +
+   ggtitle("Boxplot of Petal Width by Species") +
+   theme_minimal()
> 
```

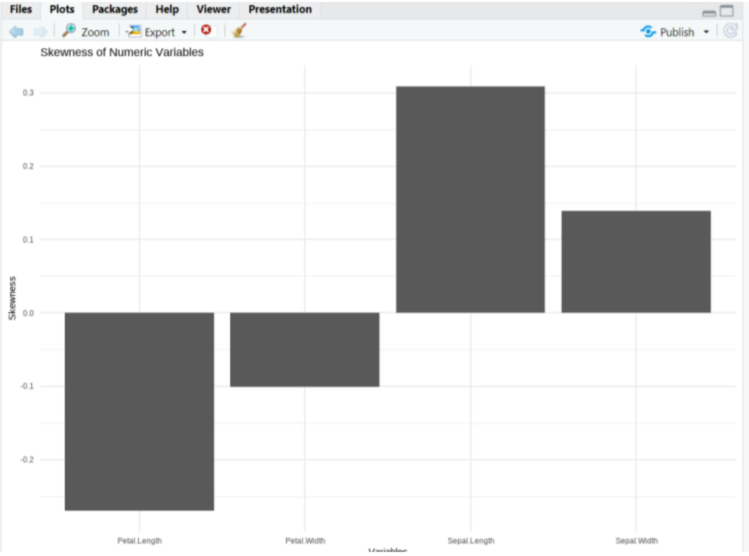


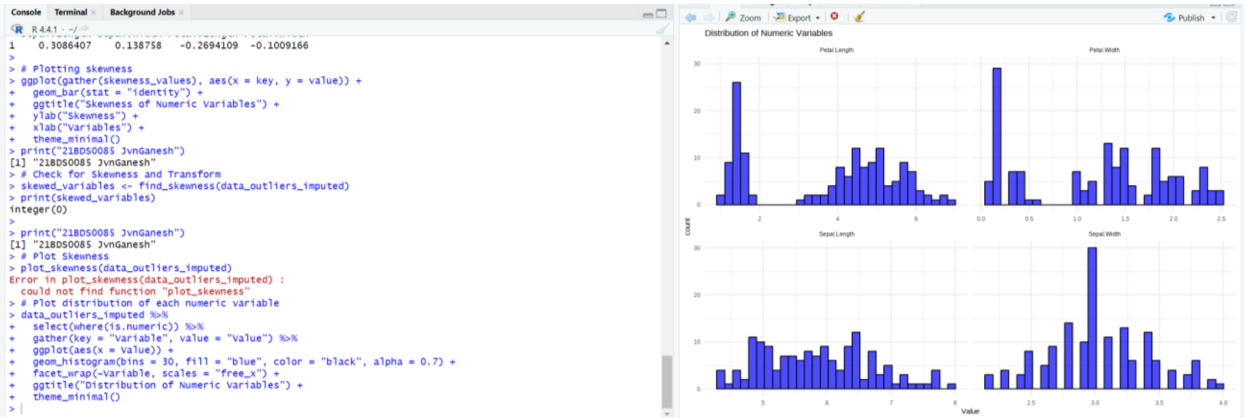


```
Run `rlang::last_trace()` to see where the error occurred.
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Function to Impute Outliers Manually Using IQR
> impute_outliers_iqr <- function(x) {
+   Q1 <- quantile(x, 0.25, na.rm = TRUE)
+   Q3 <- quantile(x, 0.75, na.rm = TRUE)
+   IQR <- Q3 - Q1
+
+   # Define the lower and upper bounds
+   lower_bound <- Q1 - 1.5 * IQR
+   upper_bound <- Q3 + 1.5 * IQR
+
+   # Replace outliers with the median
+   x[x < lower_bound] <- median(x, na.rm = TRUE)
+   x[x > upper_bound] <- median(x, na.rm = TRUE)
+
+   return(x)
+ }
> # Apply to Numeric Columns
> data_outliers_imputed <- data %>%
+   mutate(across(where(is.numeric), impute_outliers_iqr))
>
> print(summary(data_outliers_imputed))
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species
 Min.   :4.300   Min.   :2.200   Min.   :1.000   Min.   :0.100   setosa    :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50
 Mean   :5.843   Mean   :3.039   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.000   Max.   :6.900   Max.   :2.500
>
```

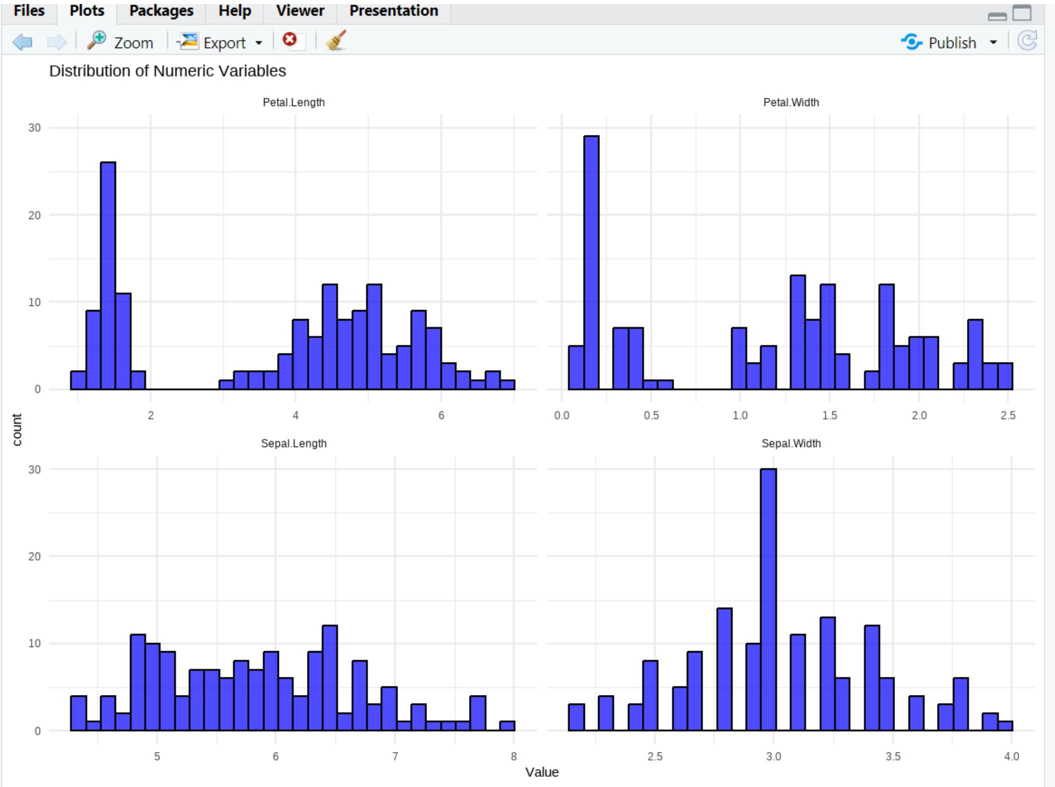


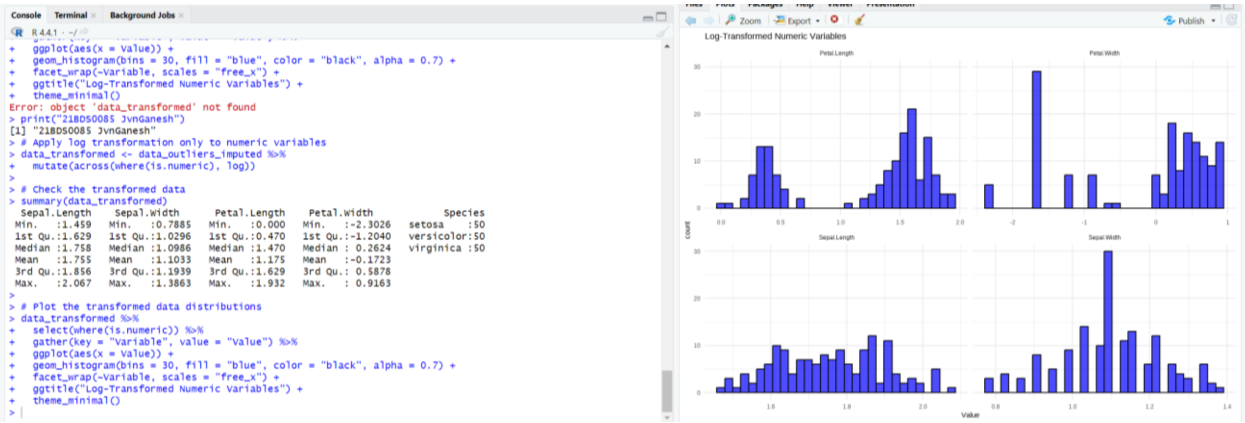
```
package ‘e1071’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
>
> # Load the library
> library(e1071)

Attaching package: ‘e1071’

The following objects are masked from ‘package:dlookr’:

    kurtosis, skewness

>
> # Calculate skewness and plot
> skewness_values <- data_outliers_imputed %>%
+   select(where(is.numeric)) %>%
+   summarise(across(everything(), skewness))
>
> print(skewness_values)
  Sepal.Length Sepal.Width Petal.Length Petal.Width
1    0.3086407    0.138758   -0.2694109  -0.1009166
>
> # Plotting skewness
> ggplot(gather(skewness_values), aes(x = key, y = value)) +
+   geom_bar(stat = "identity") +
+   ggtitle("Skewness of Numeric Variables") +
+   ylab("Skewness") +
+   xlab("Variables") +
+   theme_minimal()
>
```



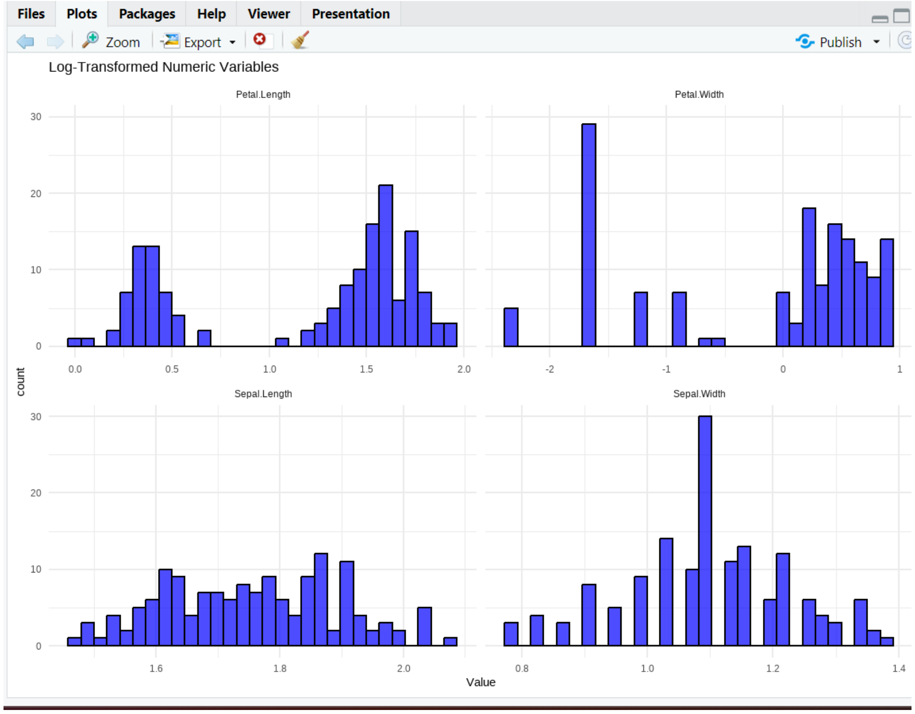


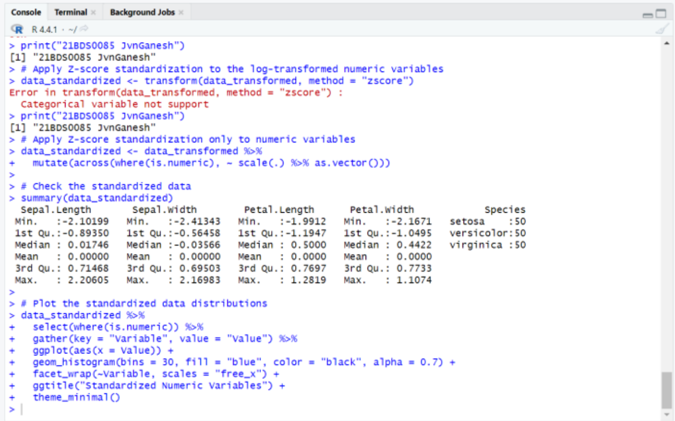
```
1     0.3086407    0.138758   -0.2694109  -0.1009166
>
> # Plotting skewness
> ggplot(gather(skewness_values), aes(x = key, y = value)) +
+   geom_bar(stat = "identity") +
+   ggtitle("Skewness of Numeric Variables") +
+   ylab("Skewness") +
+   xlab("Variables") +
+   theme_minimal()
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Check for Skewness and Transform
> skewed_variables <- find_skewness(data_outliers_imputed)
> print(skewed_variables)
integer(0)
>
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Plot Skewness
> plot_skewness(data_outliers_imputed)
Error in plot_skewness(data_outliers_imputed) :
  could not find function "plot_skewness"
> # Plot distribution of each numeric variable
> data_outliers_imputed %>%
+   select(where(is.numeric)) %>%
+   gather(key = "Variable", value = "Value") %>%
+   ggplot(aes(x = Value)) +
+   geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
+   facet_wrap(~Variable, scales = "free_x") +
+   ggtitle("Distribution of Numeric Variables") +
+   theme_minimal()
> |
```



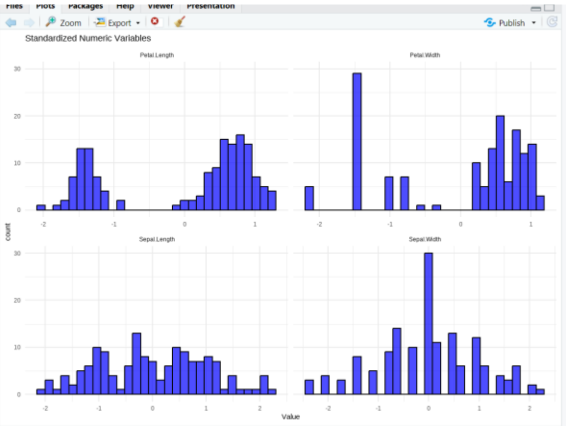


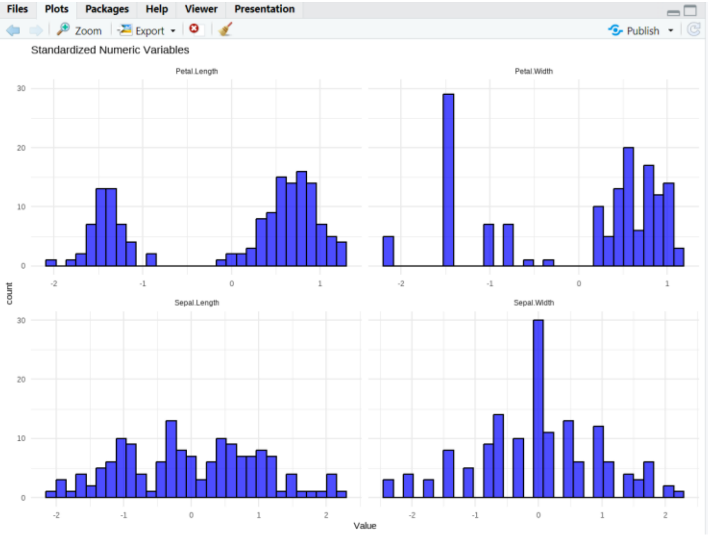
```
+   ggplot(aes(x = Value)) +
+   geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
+   facet_wrap(~Variable, scales = "free_x") +
+   ggtitle("Log-Transformed Numeric Variables") +
+   theme_minimal()
Error: object 'data_transformed' not found
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Apply log transformation only to numeric variables
> data_transformed <- data_outliers_imputed %>%
+   mutate(across(where(is.numeric), log))
>
> # Check the transformed data
> summary(data_transformed)
  Sepal.Length    Sepal.Width      Petal.Length    Petal.Width          Species  
 Min.   :1.459   Min.   :0.7885   Min.   :0.000   Min.   :-2.3026   setosa    :50  
 1st Qu.:1.629   1st Qu.:1.0296   1st Qu.:0.470   1st Qu.:-1.2040   versicolor:50  
 Median :1.758   Median :1.0986   Median :1.470   Median : 0.2624   virginica :50  
 Mean   :1.755   Mean   :1.1033   Mean   :1.175   Mean   :-0.1723                  
 3rd Qu.:1.856   3rd Qu.:1.1939   3rd Qu.:1.629   3rd Qu.: 0.5878                  
 Max.   :2.067   Max.   :1.3863   Max.   :1.932   Max.   : 0.9163                  
>
> # Plot the transformed data distributions
> data_transformed %>%
+   select(where(is.numeric)) %>%
+   gather(key = "Variable", value = "Value") %>%
+   ggplot(aes(x = Value)) +
+   geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
+   facet_wrap(~Variable, scales = "free_x") +
+   ggtitle("Log-Transformed Numeric Variables") +
+   theme_minimal()
> |
```

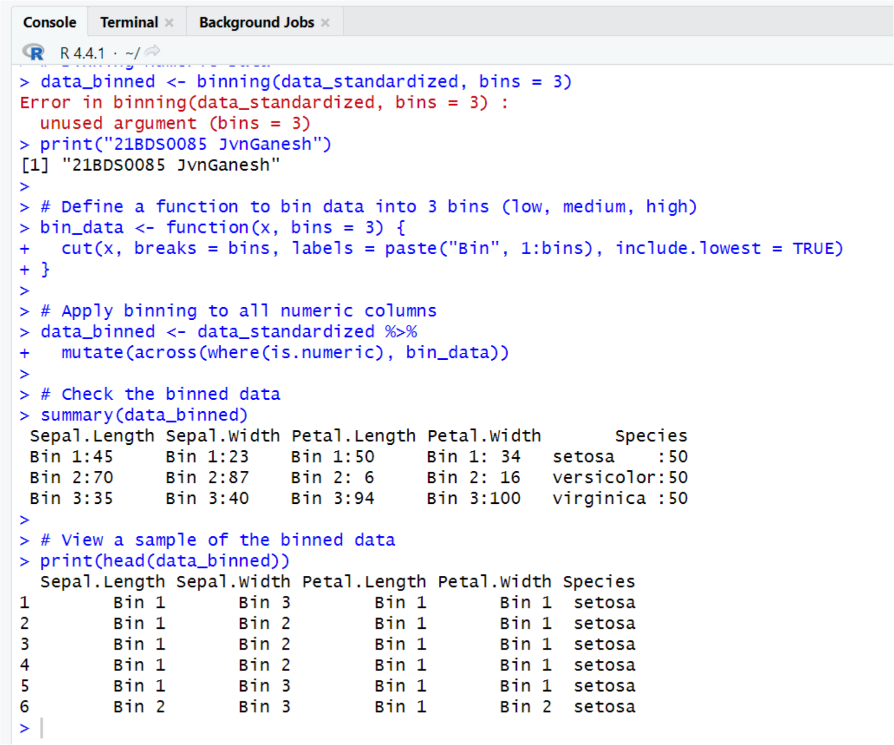




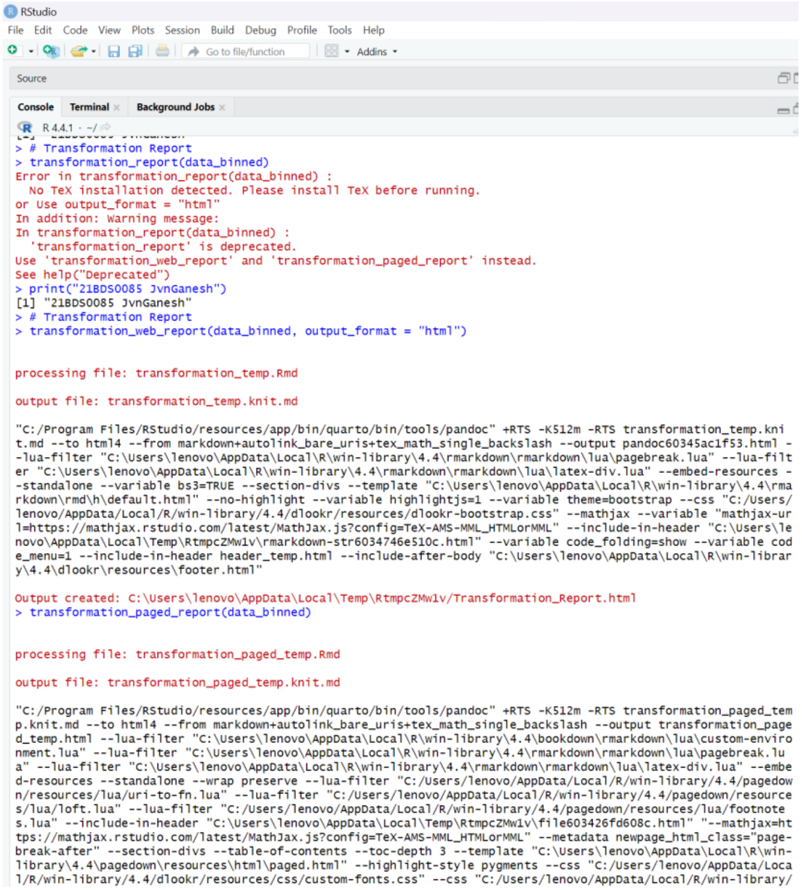
```
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Apply Z-score standardization to the log-transformed numeric variables
> data_standardized <- transform(data_transformed, method = "zscore")
Error in transform(data_transformed, method = "zscore") : 
  Categorical variable not support
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Apply Z-score standardization only to numeric variables
> data_standardized <- data_transformed %>%
+   mutate(across(where(is.numeric), ~ scale(.) %>% as.vector()))
> 
> # Check the standardized data
> summary(data_standardized)
  Sepal.Length       Sepal.Width        Petal.Length      Petal.Width            Species  
 Min.   :-2.10199   Min.   :-2.41343   Min.   :-1.9912   Min.   :-2.1671   setosa    :50  
 1st Qu.:-0.89350   1st Qu.:-0.56458   1st Qu.:-1.1947   1st Qu.:-1.0495   versicolor:50  
 Median : 0.01746   Median :-0.03566   Median : 0.5000   Median : 0.4422   virginica :50  
 Mean   : 0.00000   Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.0000                  
 3rd Qu.: 0.71468   3rd Qu.: 0.69503   3rd Qu.: 0.7697   3rd Qu.: 0.7733                  
 Max.   : 2.20605   Max.   : 2.16983   Max.   : 1.2819   Max.   : 1.1074                  
> 
> # Plot the standardized data distributions
> data_standardized %>%
+   select(where(is.numeric)) %>%
+   gather(key = "Variable", value = "Value") %>%
+   ggplot(aes(x = Value)) +
+   geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
+   facet_wrap(~Variable, scales = "free_x") +
+   ggtitle("Standardized Numeric Variables") +
+   theme_minimal()
> 
```



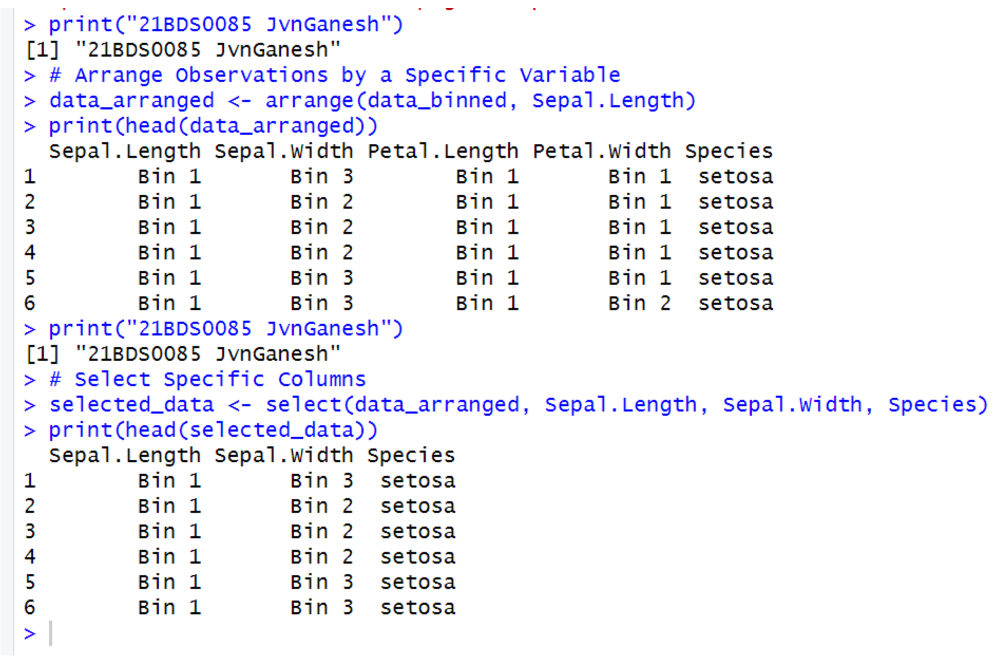




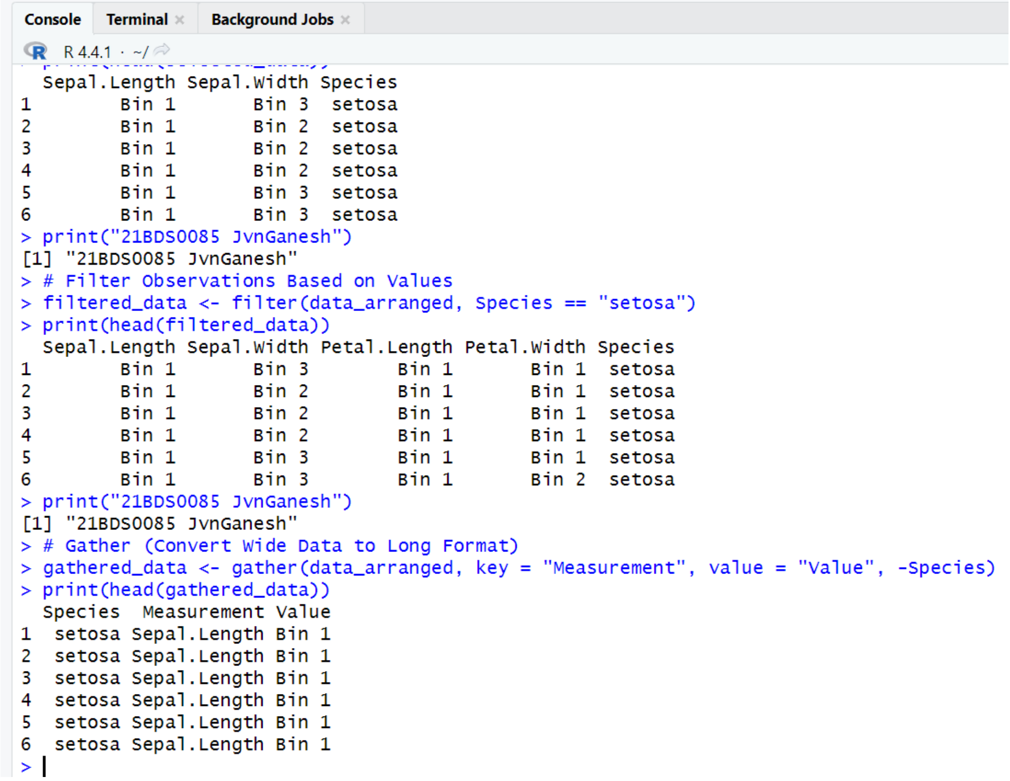
```
> data_binned <- binning(data_standardized, bins = 3)
Error in binning(data_standardized, bins = 3) : 
  unused argument (bins = 3)
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> 
> # Define a function to bin data into 3 bins (low, medium, high)
> bin_data <- function(x, bins = 3) {
+   cut(x, breaks = bins, labels = paste("Bin", 1:bins), include.lowest = TRUE)
+ }
> 
> # Apply binning to all numeric columns
> data_binned <- data_standardized %>%
+   mutate(across(where(is.numeric), bin_data))
> 
> # Check the binned data
> summary(data_binned)
 Sepal.Length Sepal.Width Petal.Length Petal.Width       Species  
 Bin 1:45     Bin 1:23    Bin 1:50     Bin 1: 34   setosa    :50  
 Bin 2:70     Bin 2:87    Bin 2: 6     Bin 2: 16   versicolor:50  
 Bin 3:35     Bin 3:40    Bin 3:94     Bin 3:100   virginica :50  
> 
> # View a sample of the binned data
> print(head(data_binned))
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1        Bin 1       Bin 3        Bin 1       Bin 1  setosa
2        Bin 1       Bin 2        Bin 1       Bin 1  setosa
3        Bin 1       Bin 2        Bin 1       Bin 1  setosa
4        Bin 1       Bin 2        Bin 1       Bin 1  setosa
5        Bin 1       Bin 3        Bin 1       Bin 1  setosa
6        Bin 2       Bin 3        Bin 1       Bin 2  setosa
> 
```



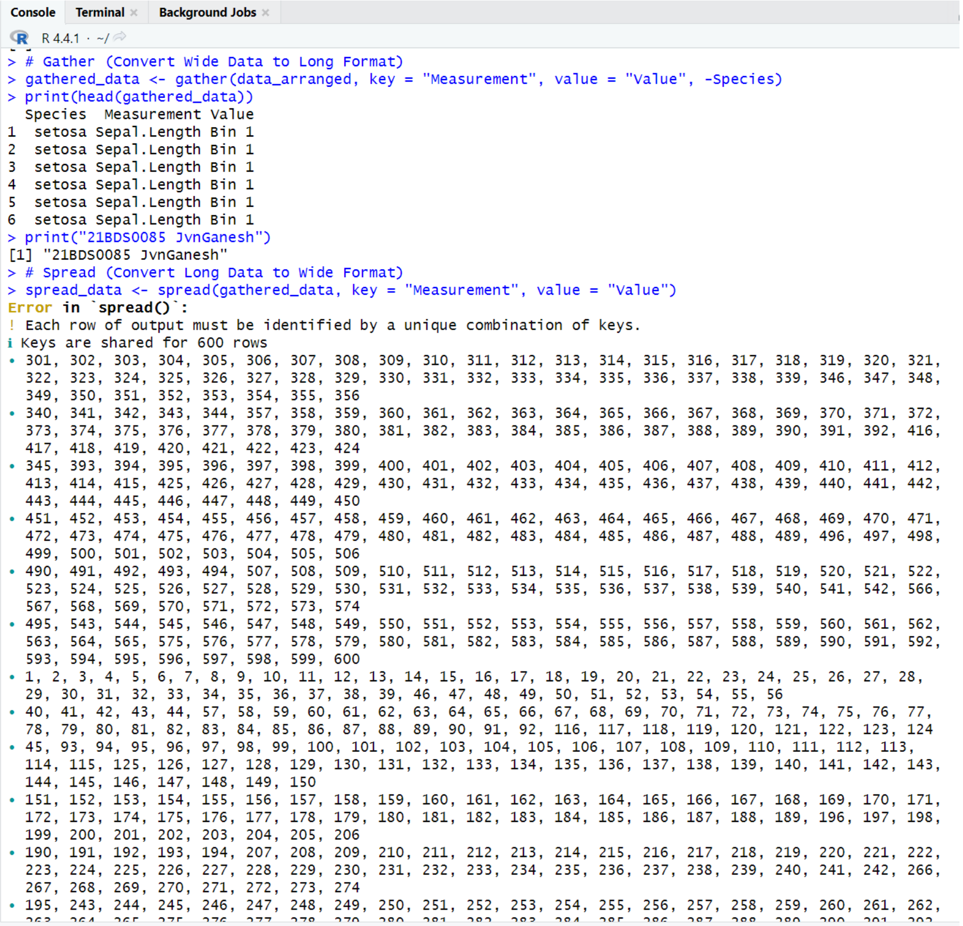
```
> # Transformation Report
> transformation_report(data_binned)
Error in transformation_report(data_binned) : 
  No TeX installation detected. Please install TeX before running.
or Use output_format = "html"
In addition: Warning message:
In transformation_report(data_binned) :
  'transformation_report' is deprecated.
Use 'transformation_web_report' and 'transformation_paged_report' instead.
See help("Deprecated")
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Transformation Report
> transformation_web_report(data_binned, output_format = "html")


processing file: transformation_temp.Rmd

output file: transformation_temp.knit.md

"C:/Program Files/RStudio/resources/app/bin/quarto/bin/tools/pandoc" +RTS -K512m -RTS transformation_temp.kni
t.md --to html4 --from markdown+autolink_bare_uris+tex_math_single_backslash --output pandoc60345ac1f53.html -
-lua-filter "C:\Users\lenovo\AppData\Local\R\win-library\4.4\rmarkdown\rmarkdown\lua\pagebreak.lua" --lua-filt
er "C:\Users\lenovo\AppData\Local\R\win-library\4.4\rmarkdown\rmarkdown\lua\latex-div.lua" --embed-resources -
-standalone --variable bs3=TRUE --section-divs --template "C:\Users\lenovo\AppData\Local\R\win-library\4.4\rma
rkdown\rmd\h\default.html" --no-highlight --variable highlightjs=1 --variable theme=bootstrap --css "C:/Users/
lenovo/AppData/Local/R/win-library/4.4/dlookr/resources/dlookr-bootstrap.css" --mathjax --variable "mathjax-ur
l=https://mathjax.rstudio.com/latest/MathJax.js?config=TeX-AMS-MML_HTMLorMML" --include-in-header "C:\Users\le
novo\AppData\Local\Temp\RtmpcZMw1v\rmarkdown-str6034746e510c.html" --variable code_folding=show --variable cod
e_menu=1 --include-in-header header_temp.html --include-after-body "C:\Users\lenovo\AppData\Local\R\win-librar
y\4.4\dlookr\resources\footer.html"

Output created: C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v/Transformation_Report.html
> transformation_paged_report(data_binned)


processing file: transformation_paged_temp.Rmd

output file: transformation_paged_temp.knit.md

"C:/Program Files/RStudio/resources/app/bin/quarto/bin/tools/pandoc" +RTS -K512m -RTS transformation_paged_tem
p.knit.md --to html4 --from markdown+autolink_bare_uris+tex_math_single_backslash --output transformation_page
d_temp.html --lua-filter "C:\Users\lenovo\AppData\Local\R\win-library\4.4\bookdown\rmarkdown\lua\custom-enviro
nment.lua" --lua-filter "C:\Users\lenovo\AppData\Local\R\win-library\4.4\rmarkdown\rmarkdown\lua\pagebreak.lu
a" --lua-filter "C:\Users\lenovo\AppData\Local\R\win-library\4.4\rmarkdown\rmarkdown\lua\latex-div.lua" --embe
d-resources --standalone --wrap preserve --lua-filter "C:/Users/lenovo/AppData/Local/R/win-library/4.4/pagedow
n/resources/lua/uri-to-fn.lua" --lua-filter "C:/Users/lenovo/AppData/Local/R/win-library/4.4/pagedown/resource
s/lua/loft.lua" --lua-filter "C:/Users/lenovo/AppData/Local/R/win-library/4.4/pagedown/resources/lua/footnote
s.lua" --include-in-header "C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\file603426fd608c.html" "--mathjax=ht
tps://mathjax.rstudio.com/latest/MathJax.js?config=TeX-AMS-MML_HTMLorMML" --metadata newpage_html_class="page-
break-after" --section-divs --table-of-contents --toc-depth 3 --template "C:\Users\lenovo\AppData\Local\R\win-
library\4.4\pagedown\resources\html\paged.html" --highlight-style pygments --css "C:/Users/lenovo/AppData/Loca
l/R/win-library/4.4/dlookr/resources/css/custom-fonts.css" --css "C:/Users/lenovo/AppData/Local/R/win-library/
```



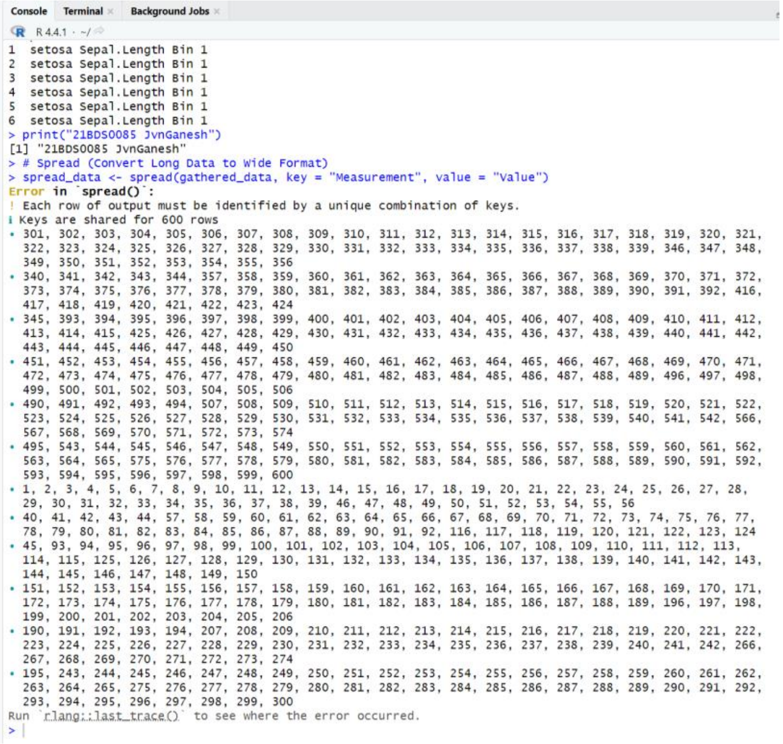
```
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Arrange Observations by a Specific Variable
> data_arranged <- arrange(data_binned, Sepal.Length)
> print(head(data_arranged))
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1        Bin 1       Bin 3        Bin 1       Bin 1  setosa
2        Bin 1       Bin 2        Bin 1       Bin 1  setosa
3        Bin 1       Bin 2        Bin 1       Bin 1  setosa
4        Bin 1       Bin 2        Bin 1       Bin 1  setosa
5        Bin 1       Bin 3        Bin 1       Bin 1  setosa
6        Bin 1       Bin 3        Bin 1       Bin 2  setosa
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Select Specific Columns
> selected_data <- select(data_arranged, Sepal.Length, Sepal.Width, Species)
> print(head(selected_data))
  Sepal.Length Sepal.Width Species
1        Bin 1       Bin 3  setosa
2        Bin 1       Bin 2  setosa
3        Bin 1       Bin 2  setosa
4        Bin 1       Bin 2  setosa
5        Bin 1       Bin 3  setosa
6        Bin 1       Bin 3  setosa
> 
```



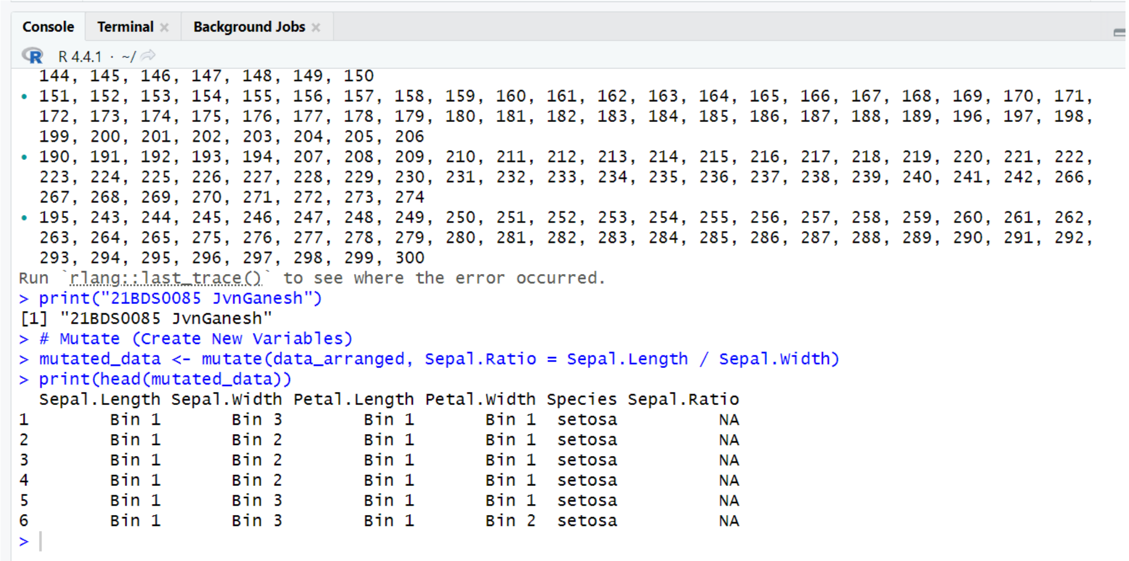
```
  Sepal.Length Sepal.Width Species
1        Bin 1       Bin 3  setosa
2        Bin 1       Bin 2  setosa
3        Bin 1       Bin 2  setosa
4        Bin 1       Bin 2  setosa
5        Bin 1       Bin 3  setosa
6        Bin 1       Bin 3  setosa
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Filter Observations Based on Values
> filtered_data <- filter(data_arranged, Species == "setosa")
> print(head(filtered_data))
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1        Bin 1       Bin 3        Bin 1       Bin 1  setosa
2        Bin 1       Bin 2        Bin 1       Bin 1  setosa
3        Bin 1       Bin 2        Bin 1       Bin 1  setosa
4        Bin 1       Bin 2        Bin 1       Bin 1  setosa
5        Bin 1       Bin 3        Bin 1       Bin 1  setosa
6        Bin 1       Bin 3        Bin 1       Bin 2  setosa
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Gather (Convert Wide Data to Long Format)
> gathered_data <- gather(data_arranged, key = "Measurement", value = "Value", -Species)
> print(head(gathered_data))
  Species  Measurement Value
1  setosa Sepal.Length Bin 1
2  setosa Sepal.Length Bin 1
3  setosa Sepal.Length Bin 1
4  setosa Sepal.Length Bin 1
5  setosa Sepal.Length Bin 1
6  setosa Sepal.Length Bin 1
> 
```



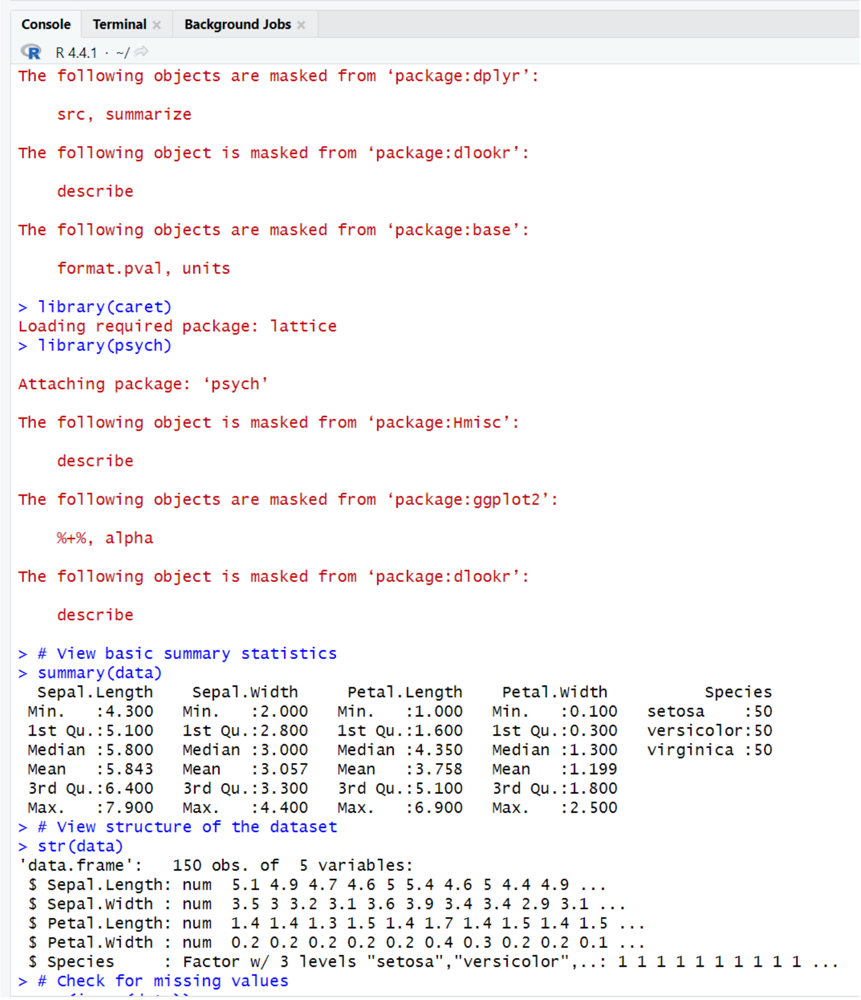
```
> # Gather (Convert Wide Data to Long Format)
> gathered_data <- gather(data_arranged, key = "Measurement", value = "Value", -Species)
> print(head(gathered_data))
  Species  Measurement Value
1  setosa Sepal.Length Bin 1
2  setosa Sepal.Length Bin 1
3  setosa Sepal.Length Bin 1
4  setosa Sepal.Length Bin 1
5  setosa Sepal.Length Bin 1
6  setosa Sepal.Length Bin 1
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Spread (Convert Long Data to Wide Format)
> spread_data <- spread(gathered_data, key = "Measurement", value = "Value")
Error in `spread()`:
! Each row of output must be identified by a unique combination of keys.
ℹ Keys are shared for 600 rows
• 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321,
  322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 346, 347, 348,
  349, 350, 351, 352, 353, 354, 355, 356
• 340, 341, 342, 343, 344, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372,
  373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 416,
  417, 418, 419, 420, 421, 422, 423, 424
• 345, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412,
  413, 414, 415, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442,
  443, 444, 445, 446, 447, 448, 449, 450
• 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471,
  472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 496, 497, 498,
  499, 500, 501, 502, 503, 504, 505, 506
• 490, 491, 492, 493, 494, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522,
  523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 566,
  567, 568, 569, 570, 571, 572, 573, 574
• 495, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562,
  563, 564, 565, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592,
  593, 594, 595, 596, 597, 598, 599, 600
• 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28,
  29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56
• 40, 41, 42, 43, 44, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77,
  78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 116, 117, 118, 119, 120, 121, 122, 123, 124
• 45, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113,
  114, 115, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143,
  144, 145, 146, 147, 148, 149, 150
• 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171,
  172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 196, 197, 198,
  199, 200, 201, 202, 203, 204, 205, 206
• 190, 191, 192, 193, 194, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222,
  223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 266,
  267, 268, 269, 270, 271, 272, 273, 274
• 195, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262,
```



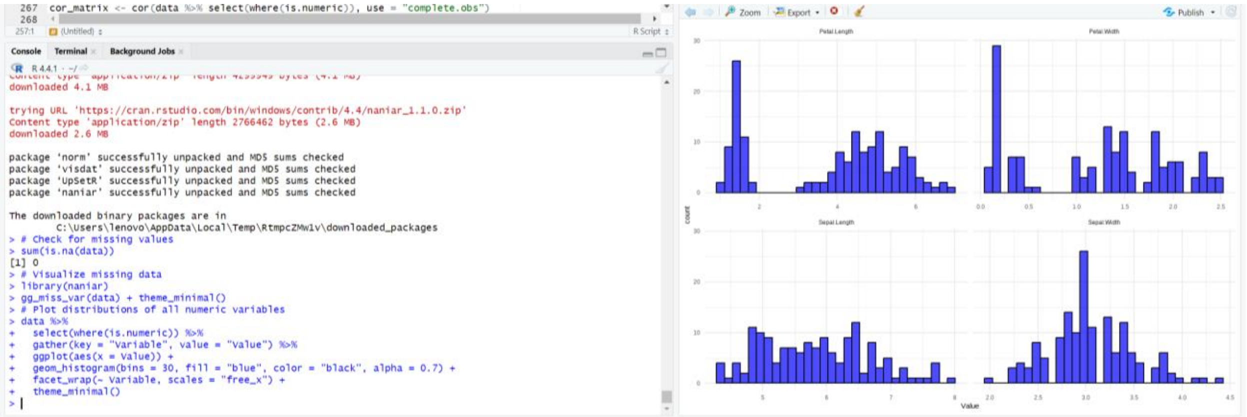
```
1  setosa Sepal.Length Bin 1
2  setosa Sepal.Length Bin 1
3  setosa Sepal.Length Bin 1
4  setosa Sepal.Length Bin 1
5  setosa Sepal.Length Bin 1
6  setosa Sepal.Length Bin 1
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Spread (Convert Long Data to Wide Format)
> spread_data <- spread(gathered_data, key = "Measurement", value = "Value")
Error in `spread()`:
! Each row of output must be identified by a unique combination of keys.
ℹ Keys are shared for 600 rows
• 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321,
  322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 346, 347, 348,
  349, 350, 351, 352, 353, 354, 355, 356
• 340, 341, 342, 343, 344, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372,
  373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 416,
  417, 418, 419, 420, 421, 422, 423, 424
• 345, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412,
  413, 414, 415, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442,
  443, 444, 445, 446, 447, 448, 449, 450
• 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471,
  472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 496, 497, 498,
  499, 500, 501, 502, 503, 504, 505, 506
• 490, 491, 492, 493, 494, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522,
  523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 566,
  567, 568, 569, 570, 571, 572, 573, 574
• 495, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562,
  563, 564, 565, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592,
  593, 594, 595, 596, 597, 598, 599, 600
• 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28,
  29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56
• 40, 41, 42, 43, 44, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77,
  78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 116, 117, 118, 119, 120, 121, 122, 123, 124
• 45, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113,
  114, 115, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143,
  144, 145, 146, 147, 148, 149, 150
• 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171,
  172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 196, 197, 198,
  199, 200, 201, 202, 203, 204, 205, 206
• 190, 191, 192, 193, 194, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222,
  223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 266,
  267, 268, 269, 270, 271, 272, 273, 274
• 195, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262,
  263, 264, 265, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292,
  293, 294, 295, 296, 297, 298, 299, 300
Run `rlang::last_trace()` to see where the error occurred.
>
```



```
  144, 145, 146, 147, 148, 149, 150
• 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171,
  172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 196, 197, 198,
  199, 200, 201, 202, 203, 204, 205, 206
• 190, 191, 192, 193, 194, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222,
  223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 266,
  267, 268, 269, 270, 271, 272, 273, 274
• 195, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262,
  263, 264, 265, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292,
  293, 294, 295, 296, 297, 298, 299, 300
Run `rlang::last_trace()` to see where the error occurred.
> print("21BDS0085 JvnGanesh")
[1] "21BDS0085 JvnGanesh"
> # Mutate (Create New Variables)
> mutated_data <- mutate(data_arranged, Sepal.Ratio = Sepal.Length / Sepal.Width)
> print(head(mutated_data))
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species Sepal.Ratio
1        Bin 1       Bin 3        Bin 1       Bin 1  setosa          NA
2        Bin 1       Bin 2        Bin 1       Bin 1  setosa          NA
3        Bin 1       Bin 2        Bin 1       Bin 1  setosa          NA
4        Bin 1       Bin 2        Bin 1       Bin 1  setosa          NA
5        Bin 1       Bin 3        Bin 1       Bin 1  setosa          NA
6        Bin 1       Bin 3        Bin 1       Bin 2  setosa          NA
> 
```

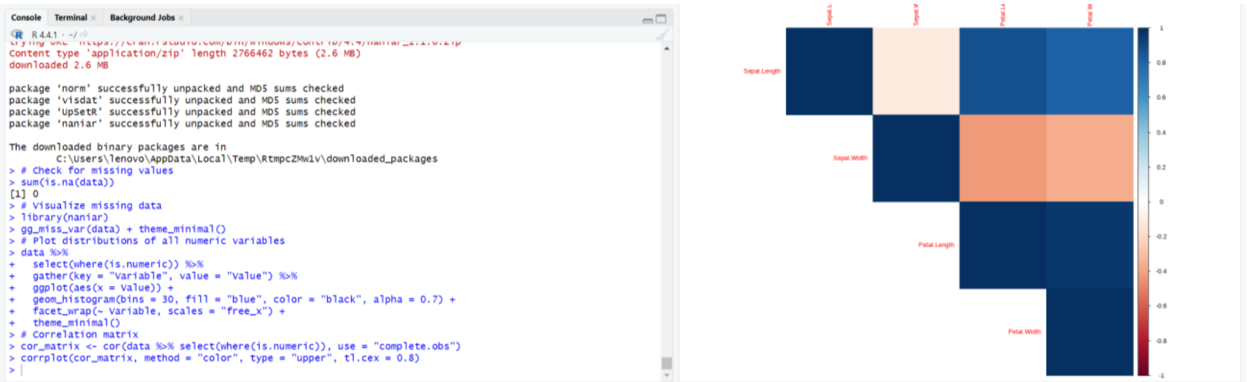


```
The following objects are masked from ‘package:dplyr’:

    src, summarize

The following object is masked from ‘package:dlookr’:

    describe

The following objects are masked from ‘package:base’:

    format.pval, units

> library(caret)
Loading required package: lattice
> library(psych)

Attaching package: ‘psych’

The following object is masked from ‘package:Hmisc’:

    describe

The following objects are masked from ‘package:ggplot2’:

    %+%, alpha

The following object is masked from ‘package:dlookr’:

    describe

> # View basic summary statistics
> summary(data)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species  
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50  
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50  
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50  
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199                  
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800                  
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500                  
> # View structure of the dataset
> str(data)
'data.frame':   150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
> # Check for missing values
```

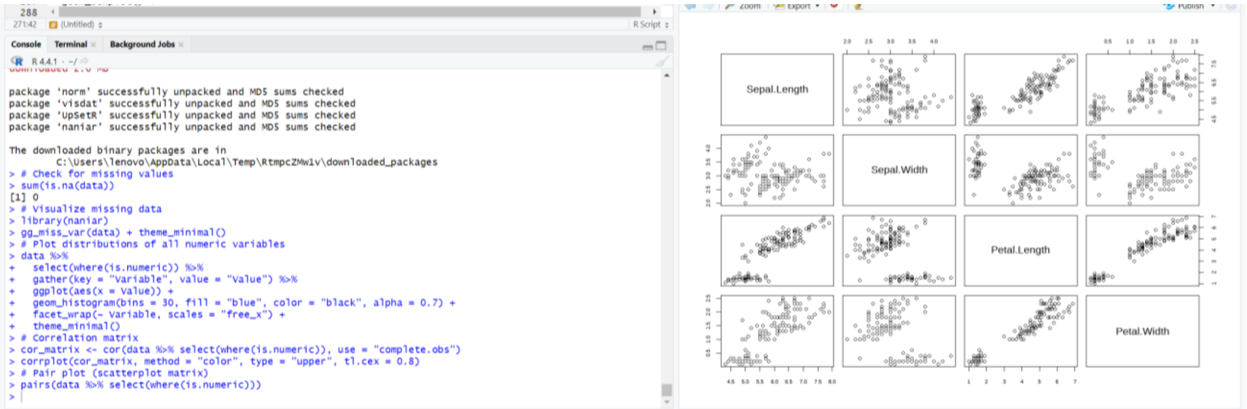


```
267  cor_matrix <- cor(data %>% select(where(is.numeric)), use = "complete.obs")
268
```

```
Content type 'application/zip' length 4299949 bytes (4.1 MB)
downloaded 4.1 MB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/naniar_1.1.0.zip'
Content type 'application/zip' length 2766462 bytes (2.6 MB)
downloaded 2.6 MB

package 'norm' successfully unpacked and MD5 sums checked
package 'visdat' successfully unpacked and MD5 sums checked
package 'UpSetR' successfully unpacked and MD5 sums checked
package 'naniar' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> # Check for missing values
> sum(is.na(data))
[1] 0
> # Visualize missing data
> library(naniar)
> gg_miss_var(data) + theme_minimal()
> # Plot distributions of all numeric variables
> data %>%
+   select(where(is.numeric)) %>%
+   gather(key = "Variable", value = "Value") %>%
+   ggplot(aes(x = Value)) +
+   geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
+   facet_wrap(~ Variable, scales = "free_x") +
+   theme_minimal()
> 
```

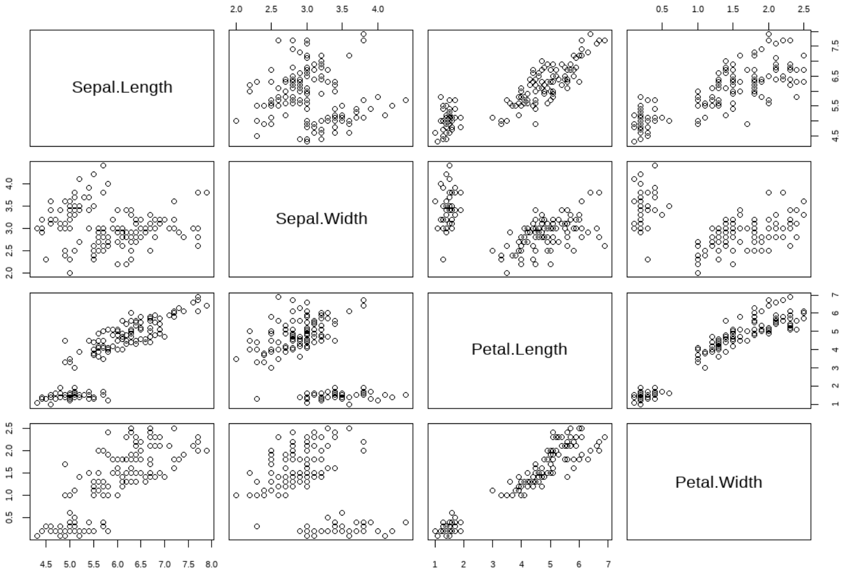


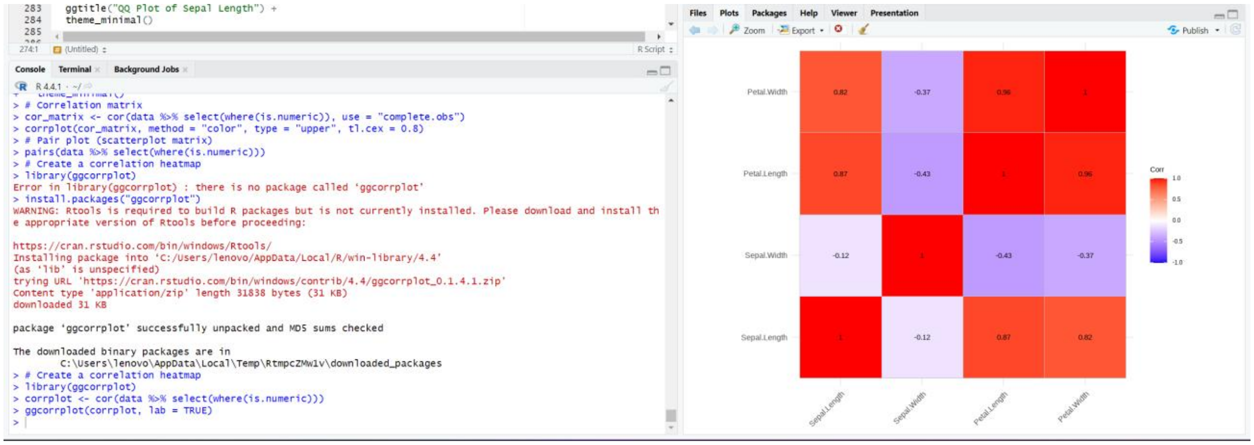
```
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/naniar_1.1.0.zip'
Content type 'application/zip' length 2766462 bytes (2.6 MB)
downloaded 2.6 MB

package 'norm' successfully unpacked and MD5 sums checked
package 'visdat' successfully unpacked and MD5 sums checked
package 'UpSetR' successfully unpacked and MD5 sums checked
package 'naniar' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> # Check for missing values
> sum(is.na(data))
[1] 0
> # Visualize missing data
> library(naniar)
> gg_miss_var(data) + theme_minimal()
> # Plot distributions of all numeric variables
> data %>%
+   select(where(is.numeric)) %>%
+   gather(key = "Variable", value = "Value") %>%
+   ggplot(aes(x = Value)) +
+   geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
+   facet_wrap(~ Variable, scales = "free_x") +
+   theme_minimal()
> # Correlation matrix
> cor_matrix <- cor(data %>% select(where(is.numeric)), use = "complete.obs")
> corrplot(cor_matrix, method = "color", type = "upper", tl.cex = 0.8)
> 
```



```
288
```

```
downloaded 2.6 MB

package 'norm' successfully unpacked and MD5 sums checked
package 'visdat' successfully unpacked and MD5 sums checked
package 'UpSetR' successfully unpacked and MD5 sums checked
package 'naniar' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> # Check for missing values
> sum(is.na(data))
[1] 0
> # Visualize missing data
> library(naniar)
> gg_miss_var(data) + theme_minimal()
> # Plot distributions of all numeric variables
> data %>%
+   select(where(is.numeric)) %>%
+   gather(key = "Variable", value = "Value") %>%
+   ggplot(aes(x = Value)) +
+   geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
+   facet_wrap(~ Variable, scales = "free_x") +
+   theme_minimal()
> # Correlation matrix
> cor_matrix <- cor(data %>% select(where(is.numeric)), use = "complete.obs")
> corrplot(cor_matrix, method = "color", type = "upper", tl.cex = 0.8)
> # Pair plot (scatterplot matrix)
> pairs(data %>% select(where(is.numeric)))
> 
```

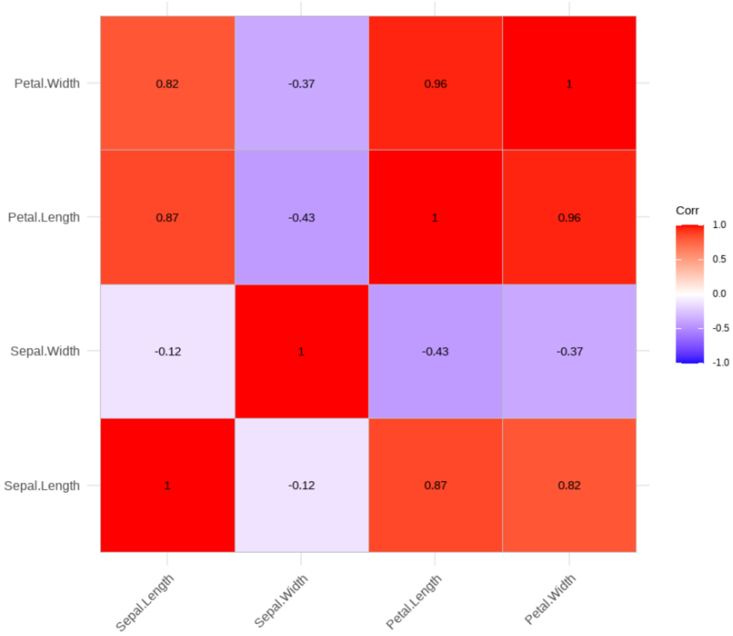


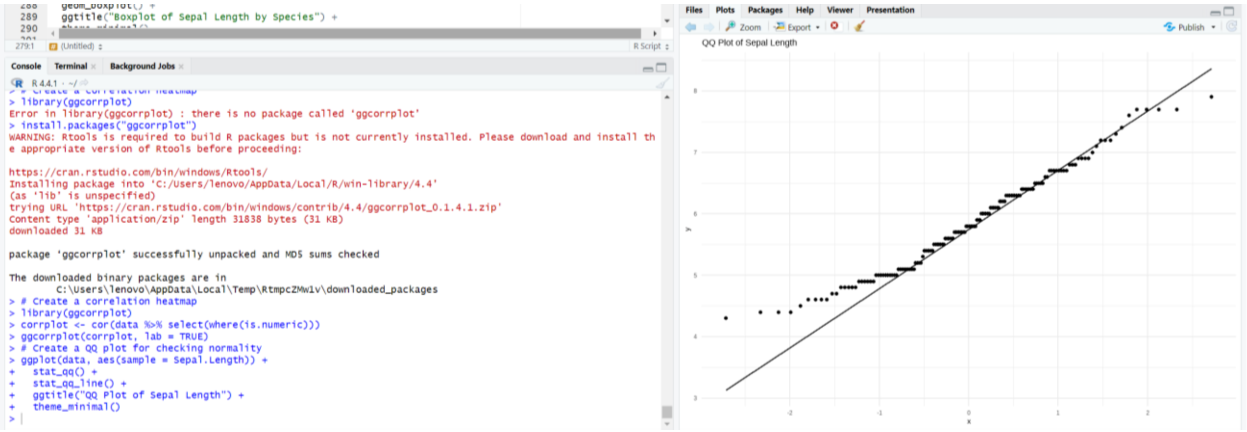


```
ggtitle("QQ Plot of Sepal Length") +
theme_minimal()
```

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
Petal.Width	0.82	-0.37	0.96	1
Petal.Length	0.87	-0.43	1	0.96
Sepal.Width	-0.12	1	-0.43	-0.37
Sepal.Length	1	-0.12	0.87	0.82

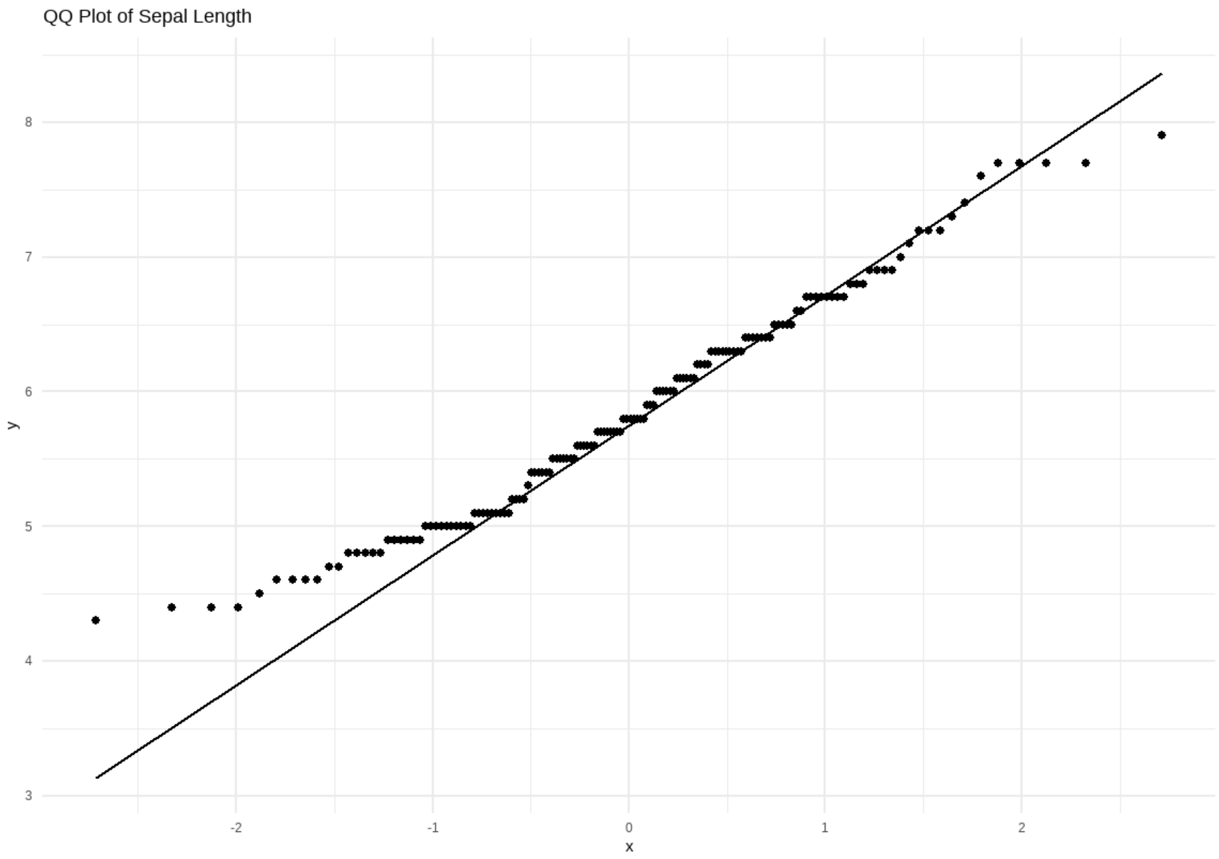
```
+   theme_minimal()
> # Correlation matrix
> cor_matrix <- cor(data %>% select(where(is.numeric)), use = "complete.obs")
> corrplot(cor_matrix, method = "color", type = "upper", tl.cex = 0.8)
> # Pair plot (scatterplot matrix)
> pairs(data %>% select(where(is.numeric)))
> # Create a correlation heatmap
> library(ggcorrplot)
Error in library(ggcorrplot) : there is no package called ‘ggcorrplot’
> install.packages("ggcorrplot")
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install th
e appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into ‘C:/Users/lenovo/AppData/Local/R/win-library/4.4’
(as ‘lib’ is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/ggcorrplot_0.1.4.1.zip'
Content type 'application/zip' length 31838 bytes (31 KB)
downloaded 31 KB

package ‘ggcorrplot’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> # Create a correlation heatmap
> library(ggcorrplot)
> corrplot <- cor(data %>% select(where(is.numeric)))
> ggcorrplot(corrplot, lab = TRUE)
> 
```

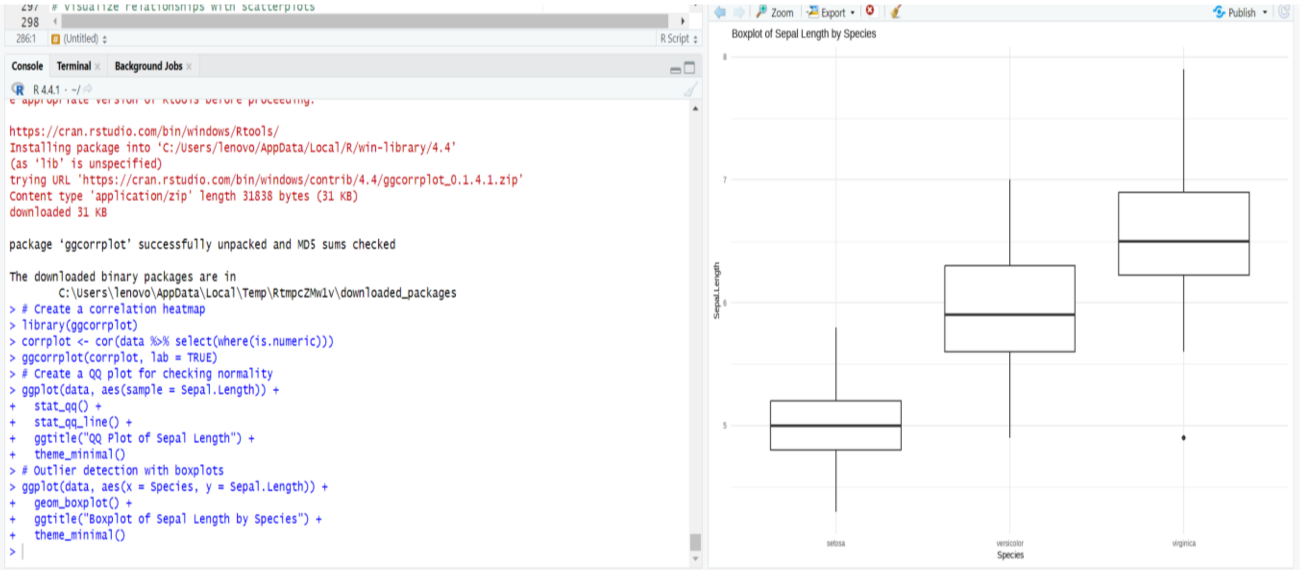




```
  geom_boxplot() +
  ggtitle("Boxplot of Sepal Length by Species") +
  theme_minimal()
```

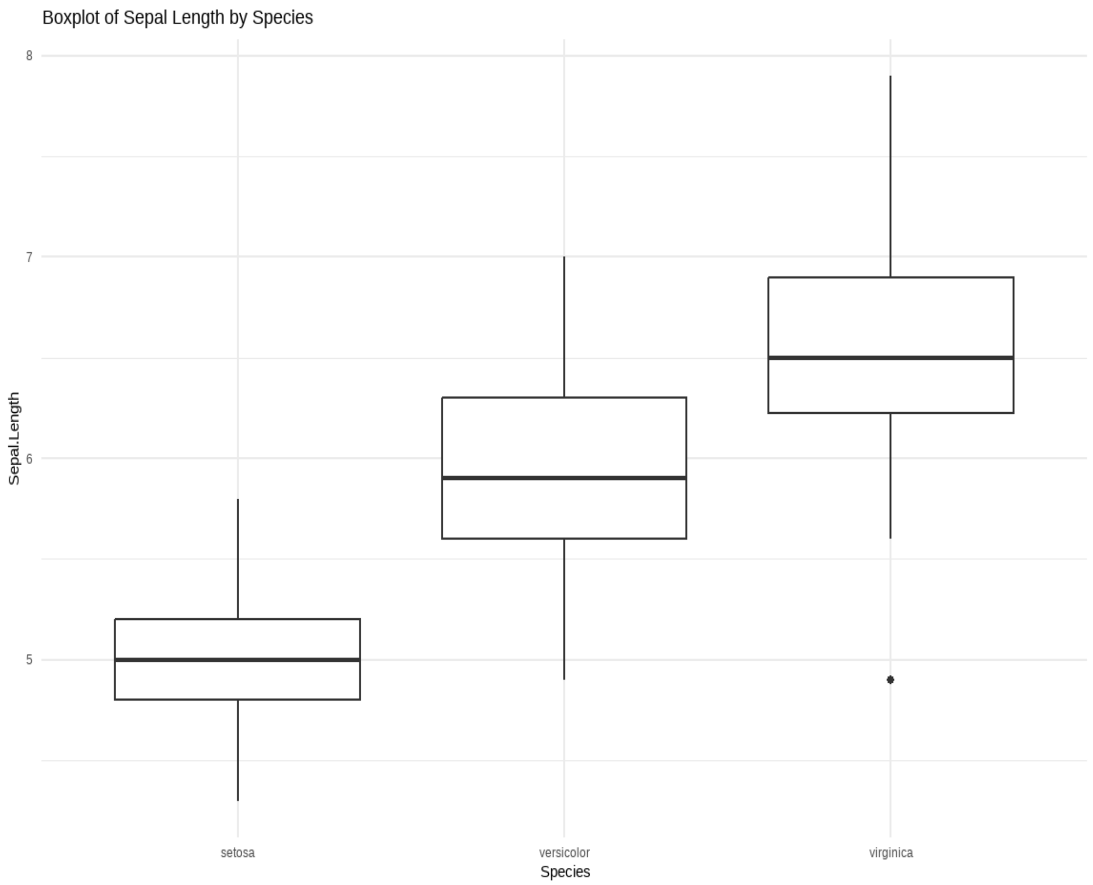
```
> # Create a correlation heatmap
> library(ggcorrplot)
Error in library(ggcorrplot) : there is no package called 'ggcorrplot'
> install.packages("ggcorrplot")
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install th
e appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/lenovo/AppData/Local/R/win-library/4.4'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/ggcorrplot_0.1.4.1.zip'
Content type 'application/zip' length 31838 bytes (31 KB)
downloaded 31 KB

package 'ggcorrplot' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> # Create a correlation heatmap
> library(ggcorrplot)
> corrplot <- cor(data %>% select(where(is.numeric)))
> ggcorrplot(corrplot, lab = TRUE)
> # Create a QQ plot for checking normality
> ggplot(data, aes(sample = Sepal.Length)) +
+   stat_qq() +
+   stat_qq_line() +
+   ggtitle("QQ Plot of Sepal Length") +
+   theme_minimal()
> 
```

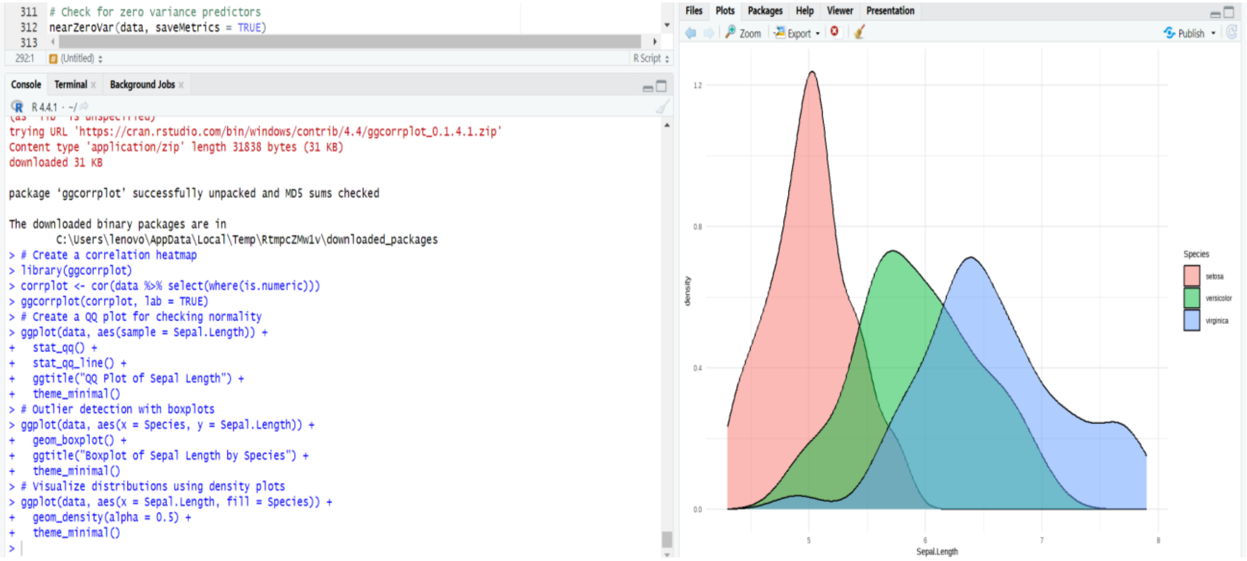




```
# Visualize relationships with scatterplots
```

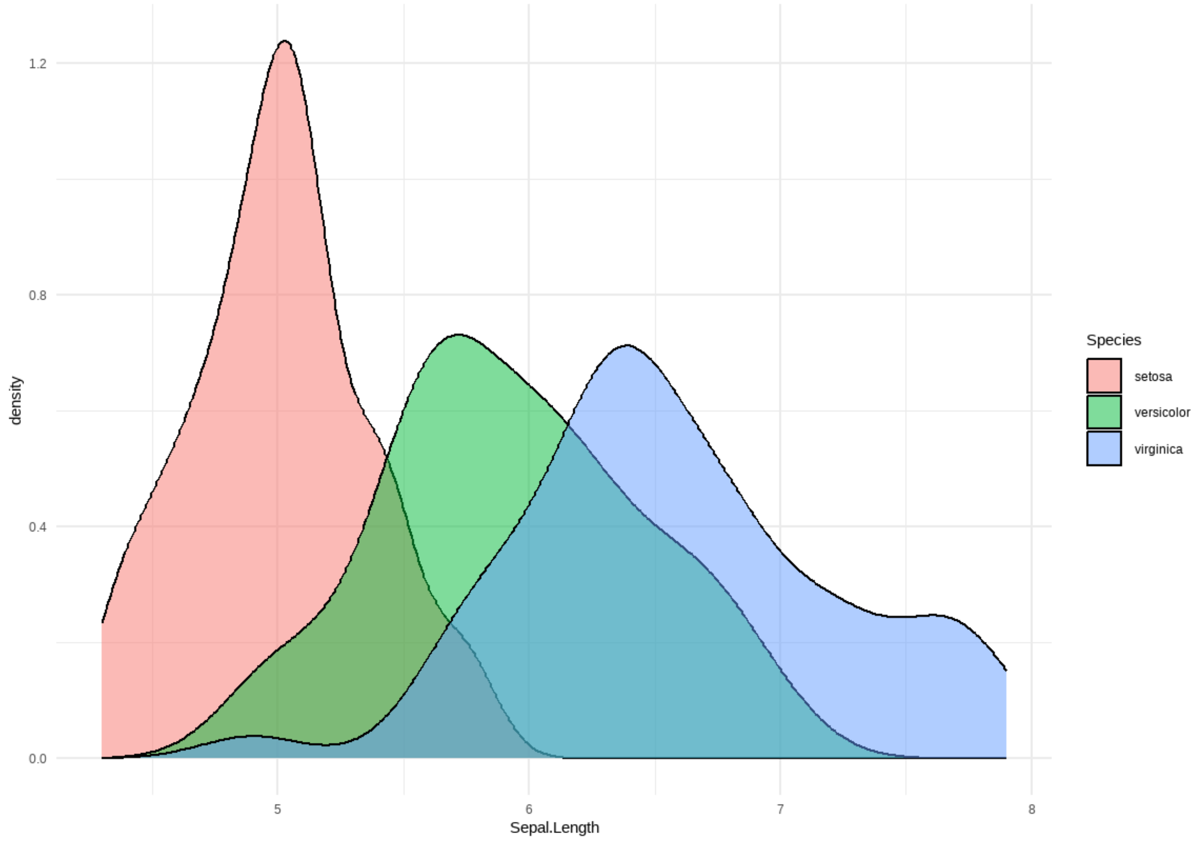
```
e appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/lenovo/AppData/Local/R/win-library/4.4'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/ggcorrplot_0.1.4.1.zip'
Content type 'application/zip' length 31838 bytes (31 KB)
downloaded 31 KB

package 'ggcorrplot' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> # Create a correlation heatmap
> library(ggcorrplot)
> corrplot <- cor(data %>% select(where(is.numeric)))
> ggcorrplot(corrplot, lab = TRUE)
> # Create a QQ plot for checking normality
> ggplot(data, aes(sample = Sepal.Length)) +
+   stat_qq() +
+   stat_qq_line() +
+   ggtitle("QQ Plot of Sepal Length") +
+   theme_minimal()
> # Outlier detection with boxplots
> ggplot(data, aes(x = Species, y = Sepal.Length)) +
+   geom_boxplot() +
+   ggtitle("Boxplot of Sepal Length by Species") +
+   theme_minimal()
> |
```

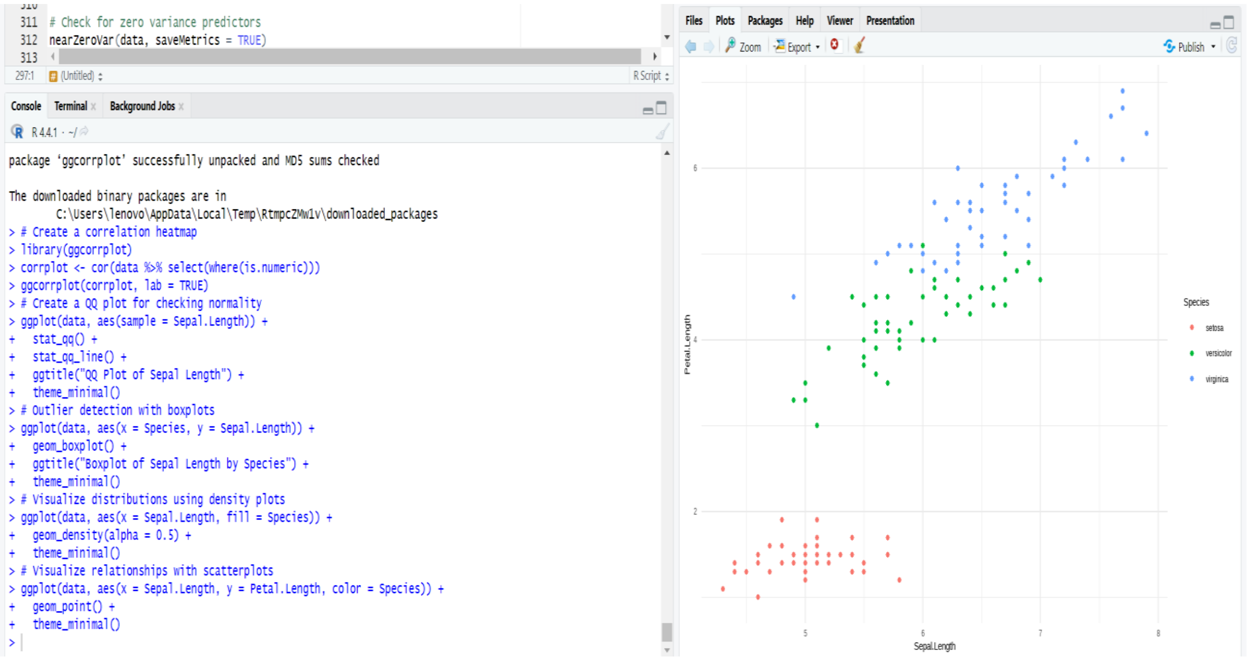




```
# Check for zero variance predictors
nearZeroVar(data, saveMetrics = TRUE)
```

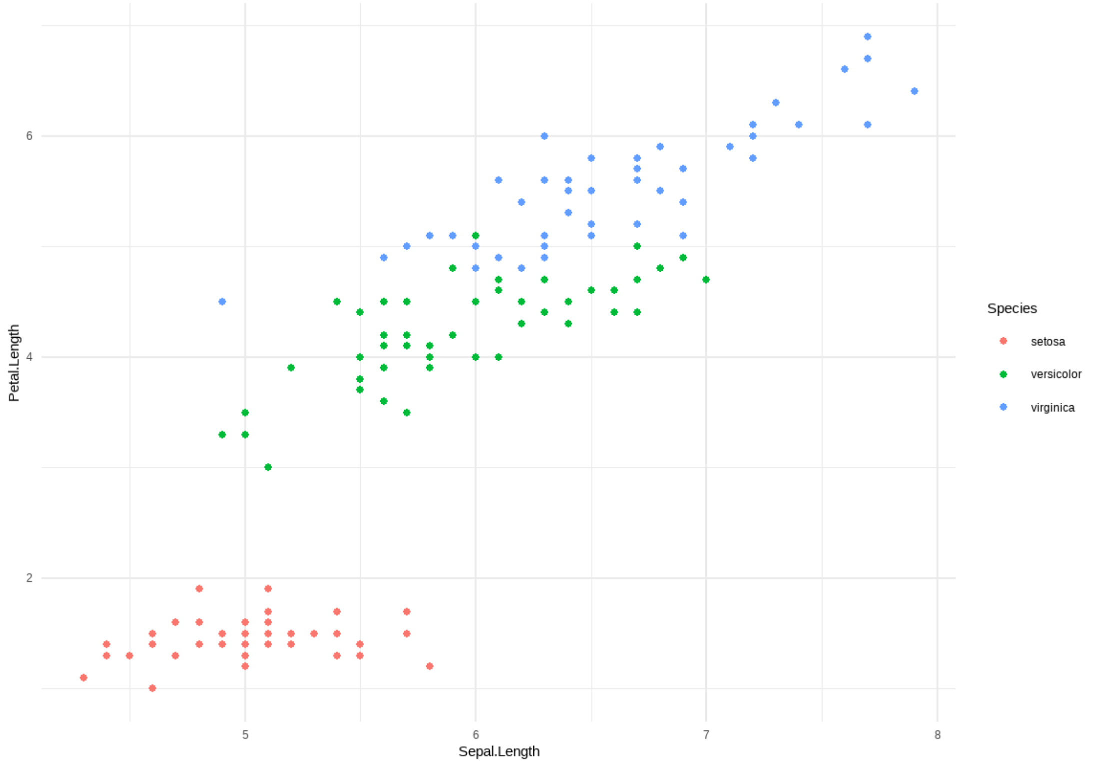
```
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/ggcorrplot_0.1.4.1.zip'
Content type 'application/zip' length 31838 bytes (31 KB)
downloaded 31 KB

package 'ggcorrplot' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> # Create a correlation heatmap
> library(ggcorrplot)
> corrplot <- cor(data %>% select(where(is.numeric)))
> ggcorrplot(corrplot, lab = TRUE)
> # Create a QQ plot for checking normality
> ggplot(data, aes(sample = Sepal.Length)) +
+   stat_qq() +
+   stat_qq_line() +
+   ggtitle("QQ Plot of Sepal Length") +
+   theme_minimal()
> # Outlier detection with boxplots
> ggplot(data, aes(x = Species, y = Sepal.Length)) +
+   geom_boxplot() +
+   ggtitle("Boxplot of Sepal Length by Species") +
+   theme_minimal()
> # Visualize distributions using density plots
> ggplot(data, aes(x = Sepal.Length, fill = Species)) +
+   geom_density(alpha = 0.5) +
+   theme_minimal()
> 
```

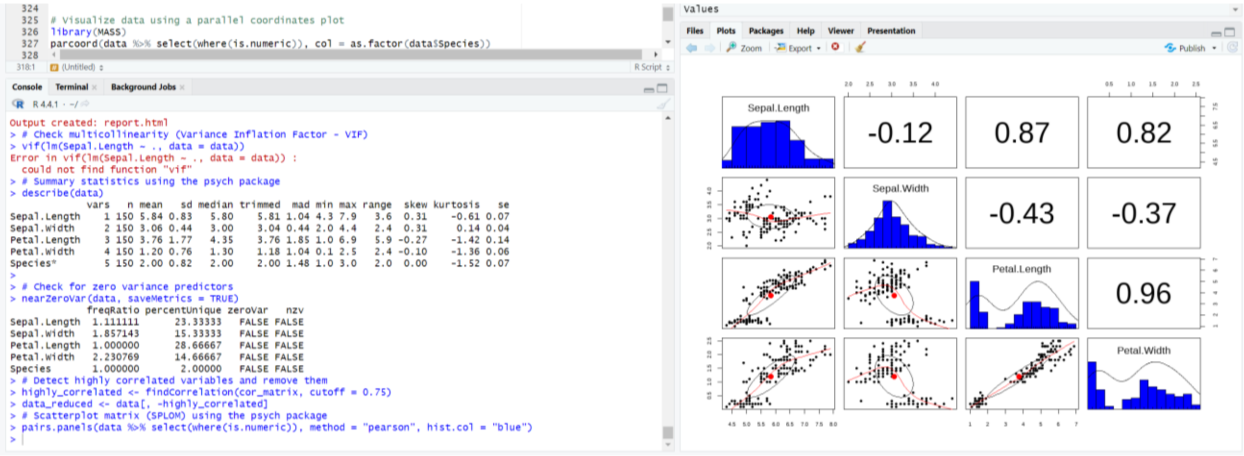




```
# Check for zero variance predictors
nearZeroVar(data, saveMetrics = TRUE)
```

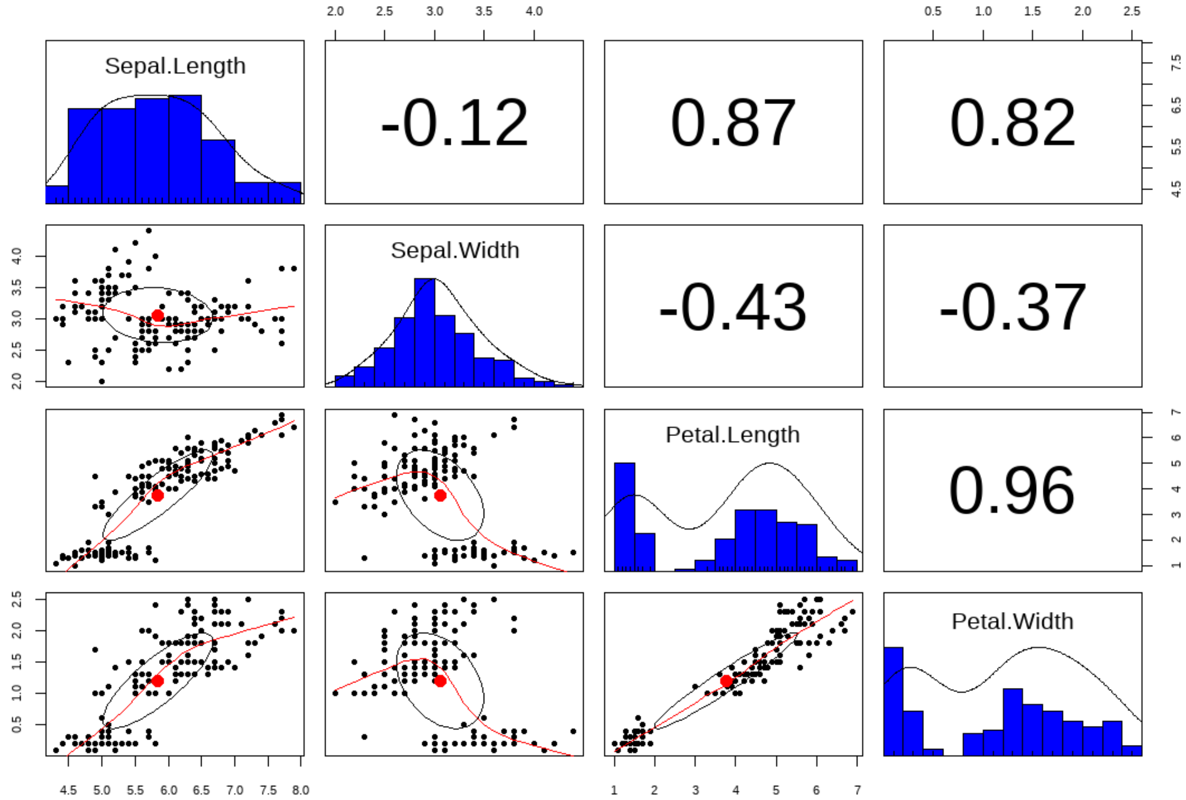
```
package ‘ggcorrplot’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> # Create a correlation heatmap
> library(ggcorrplot)
> corrplot <- cor(data %>% select(where(is.numeric)))
> ggcorrplot(corrplot, lab = TRUE)
> # Create a QQ plot for checking normality
> ggplot(data, aes(sample = Sepal.Length)) +
+   stat_qq() +
+   stat_qq_line() +
+   ggtitle("QQ Plot of Sepal Length") +
+   theme_minimal()
> # Outlier detection with boxplots
> ggplot(data, aes(x = Species, y = Sepal.Length)) +
+   geom_boxplot() +
+   ggtitle("Boxplot of Sepal Length by Species") +
+   theme_minimal()
> # Visualize distributions using density plots
> ggplot(data, aes(x = Sepal.Length, fill = Species)) +
+   geom_density(alpha = 0.5) +
+   theme_minimal()
> # Visualize relationships with scatterplots
> ggplot(data, aes(x = Sepal.Length, y = Petal.Length, color = Species)) +
+   geom_point() +
+   theme_minimal()
>
```

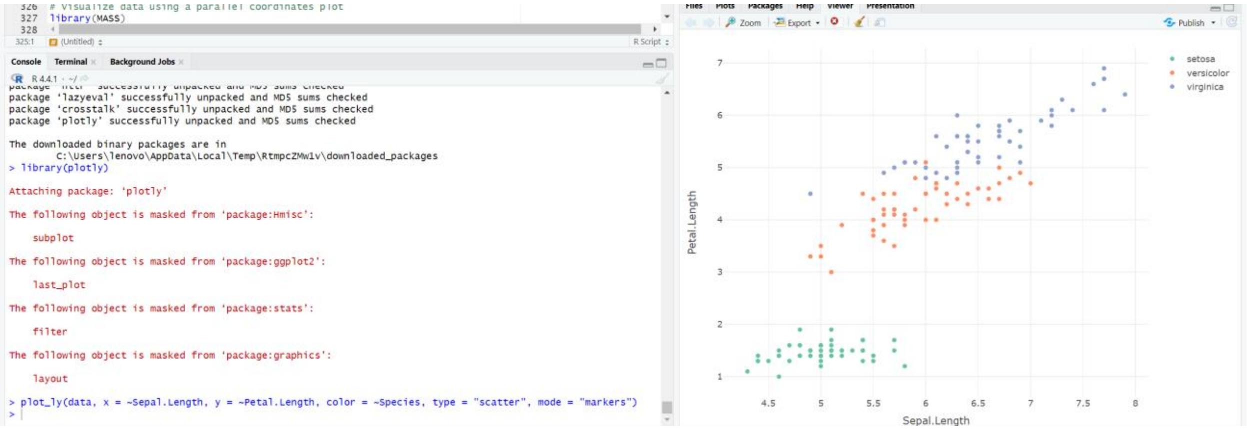




```

# Visualize data using a parallel coordinates plot
library(MASS)
parcoord(data %>% select(where(is.numeric)), col = as.factor(data$Species))

```

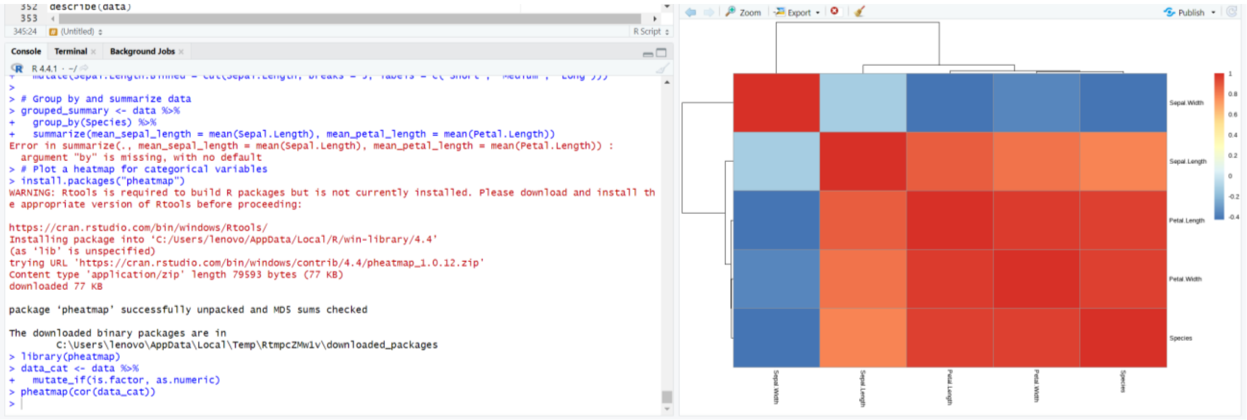
```
Output created: report.html
> # Check multicollinearity (Variance Inflation Factor - VIF)
> vif(lm(Sepal.Length ~ ., data = data))
Error in vif(lm(Sepal.Length ~ ., data = data)) :
  could not find function "vif"
> # Summary statistics using the psych package
> describe(data)
             vars   n mean   sd median trimmed  mad min max range  skew kurtosis   se
Sepal.Length    1 150 5.84 0.83   5.80    5.81 1.04 4.3 7.9   3.6  0.31    -0.61 0.07
Sepal.Width     2 150 3.06 0.44   3.00    3.04 0.44 2.0 4.4   2.4  0.31     0.14 0.04
Petal.Length    3 150 3.76 1.77   4.35    3.76 1.85 1.0 6.9   5.9 -0.27    -1.42 0.14
Petal.Width     4 150 1.20 0.76   1.30    1.18 1.04 0.1 2.5   2.4 -0.10    -1.36 0.06
Species*        5 150 2.00 0.82   2.00    2.00 1.48 1.0 3.0   2.0  0.00    -1.52 0.07
>
> # Check for zero variance predictors
> nearZeroVar(data, saveMetrics = TRUE)
             freqRatio percentUnique zeroVar   nzv
Sepal.Length  1.111111      23.33333   FALSE FALSE
Sepal.Width   1.857143      15.33333   FALSE FALSE
Petal.Length  1.000000      28.66667   FALSE FALSE
Petal.Width   2.230769      14.66667   FALSE FALSE
Species       1.000000       2.00000   FALSE FALSE
> # Detect highly correlated variables and remove them
> highly_correlated <- findCorrelation(cor_matrix, cutoff = 0.75)
> data_reduced <- data[, -highly_correlated]
> # Scatterplot matrix (SPLOM) using the psych package
> pairs.panels(data %>% select(where(is.numeric)), method = "pearson", hist.col = "blue")
>
```





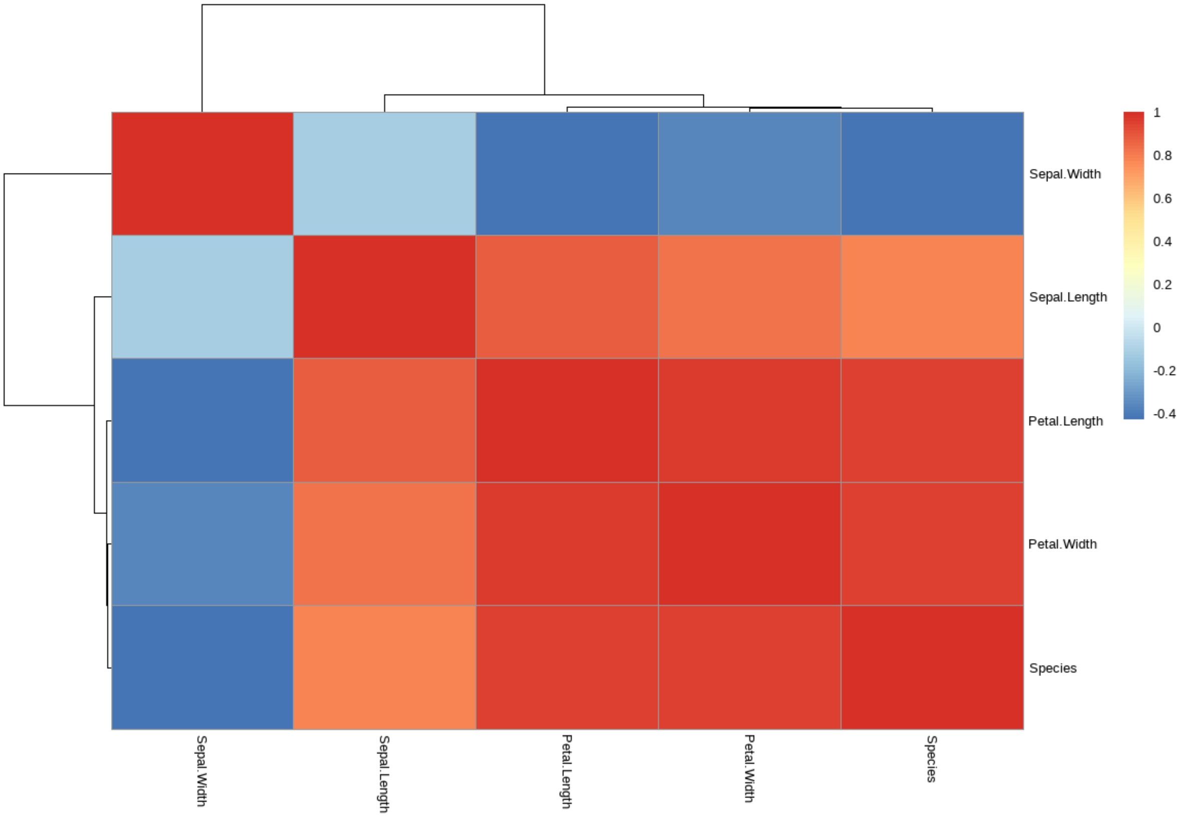
```
# Visualize data using a parallel coordinates plot
library(MASS)
```

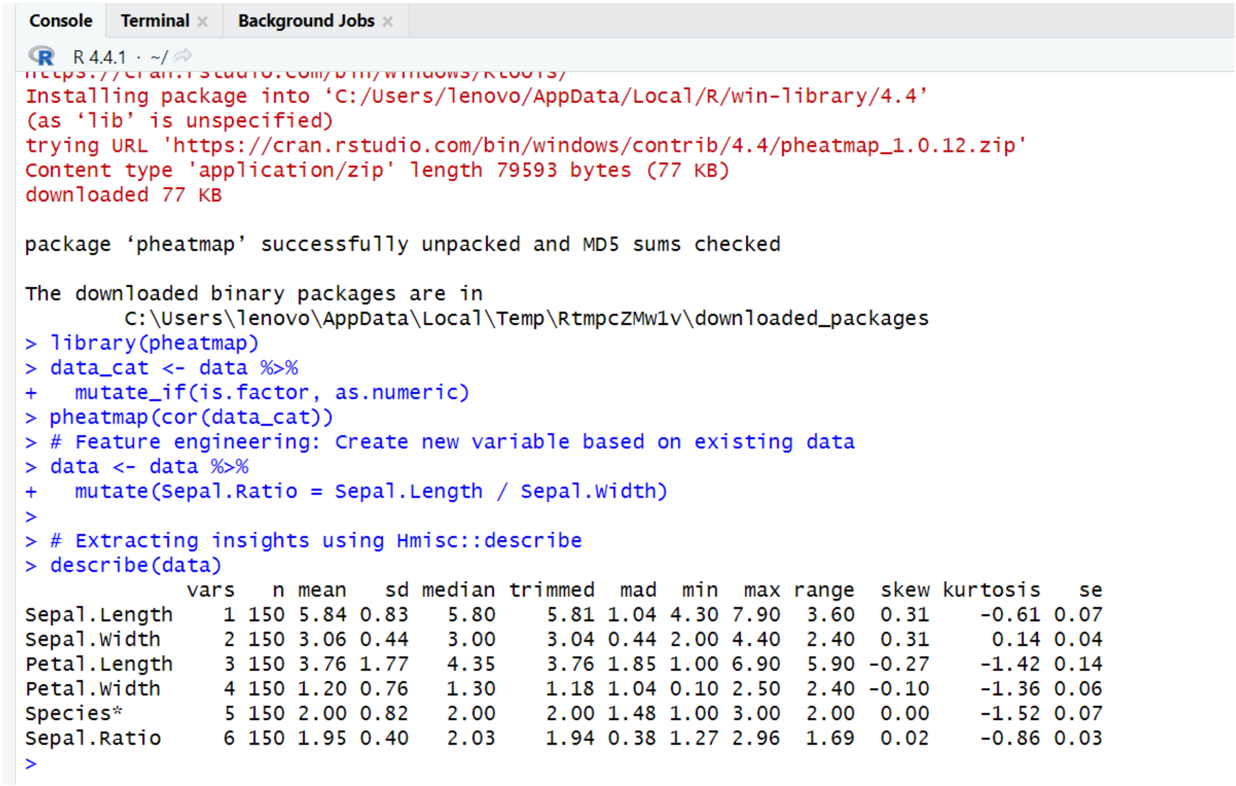
```
package 'httr' successfully unpacked and MD5 sums checked
package 'lazyeval' successfully unpacked and MD5 sums checked
package 'crosstalk' successfully unpacked and MD5 sums checked
package 'plotly' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> library(plotly)

Attaching package: 'plotly'

The following object is masked from 'package:Hmisc':

    subplot

The following object is masked from 'package:ggplot2':

    last_plot

The following object is masked from 'package:stats':

    filter

The following object is masked from 'package:graphics':

    layout

> plot_ly(data, x = ~Sepal.Length, y = ~Petal.Length, color = ~Species, type = "scatter", mode = "markers")
> 
```



```
describe(data)
```

```
+   mutate(Sepal.Length.Binned = cut(Sepal.Length, breaks = 3, labels = c("Short", "Medium", "Long")))
>
> # Group by and summarize data
> grouped_summary <- data %>%
+   group_by(Species) %>%
+   summarize(mean_sepal_length = mean(Sepal.Length), mean_petal_length = mean(Petal.Length))
Error in summarize(., mean_sepal_length = mean(Sepal.Length), mean_petal_length = mean(Petal.Length)) :
  argument "by" is missing, with no default
> # Plot a heatmap for categorical variables
> install.packages("pheatmap")
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install th
e appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/lenovo/AppData/Local/R/win-library/4.4'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/pheatmap_1.0.12.zip'
Content type 'application/zip' length 79593 bytes (77 KB)
downloaded 77 KB

package 'pheatmap' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> library(pheatmap)
> data_cat <- data %>%
+   mutate_if(is.factor, as.numeric)
> pheatmap(cor(data_cat))
>
```





```
https://cran.rstudio.com/bin/windows/Rtools/
Installing package into ‘C:/Users/lenovo/AppData/Local/R/win-library/4.4’
(as ‘lib’ is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/pheatmap_1.0.12.zip'
Content type 'application/zip' length 79593 bytes (77 KB)
downloaded 77 KB

package ‘pheatmap’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\lenovo\AppData\Local\Temp\RtmpcZMw1v\downloaded_packages
> library(pheatmap)
> data_cat <- data %>%
+   mutate_if(is.factor, as.numeric)
> pheatmap(cor(data_cat))
> # Feature engineering: Create new variable based on existing data
> data <- data %>%
+   mutate(Sepal.Ratio = Sepal.Length / Sepal.Width)
>
> # Extracting insights using Hmisc::describe
> describe(data)
             vars   n mean   sd median trimmed  mad  min  max range  skew kurtosis   se
Sepal.Length    1 150 5.84 0.83   5.80    5.81 1.04 4.30 7.90  3.60  0.31    -0.61 0.07
Sepal.Width     2 150 3.06 0.44   3.00    3.04 0.44 2.00 4.40  2.40  0.31     0.14 0.04
Petal.Length    3 150 3.76 1.77   4.35    3.76 1.85 1.00 6.90  5.90 -0.27    -1.42 0.14
Petal.Width     4 150 1.20 0.76   1.30    1.18 1.04 0.10 2.50  2.40 -0.10    -1.36 0.06
Species*        5 150 2.00 0.82   2.00    2.00 1.48 1.00 3.00  2.00  0.00    -1.52 0.07
Sepal.Ratio     6 150 1.95 0.40   2.03    1.94 0.38 1.27 2.96  1.69  0.02    -0.86 0.03
>
```

Code

```
# Load required libraries

library(dlookr)

library(dplyr)

library(tidyr)

library(ggplot2)


print("21BDS0085 JvnGanesh")

# Load the iris dataset

data <- iris

print(head(data))


print("21BDS0085 JvnGanesh")
```

```
# Overview of the Data

str(data)

summary(data)

glimpse(data)


print("21BDS0085 JvnGanesh")

# Check for Missing Values and Impute

na_variables <- find_na(data)

print(na_variables)


print("21BDS0085 JvnGanesh")

# Since the iris dataset does not contain missing values, no imputation is required here

# However, if there were missing values, the following line would impute them:

# data_imputed <- imputate_na(data)


print("21BDS0085 JvnGanesh")

# Check for Outliers and Impute

outlier_variables <- find_outliers(data)

print(outlier_variables)


print("21BDS0085 JvnGanesh")

# Plot Outliers

plot_outlier(data)


# Identify Outliers

outlier_variables <- find_outliers(data)
```

```
print(outlier_variables)


# Custom Plot for Outliers using ggplot2
ggplot(data, aes(x = Species, y = Sepal.Length)) +
  geom_boxplot() +
  ggtitle("Boxplot of Sepal Length by Species") +
  theme_minimal()

ggplot(data, aes(x = Species, y = Sepal.Width)) +
  geom_boxplot() +
  ggtitle("Boxplot of Sepal Width by Species") +
  theme_minimal()

ggplot(data, aes(x = Species, y = Petal.Length)) +
  geom_boxplot() +
  ggtitle("Boxplot of Petal Length by Species") +
  theme_minimal()

ggplot(data, aes(x = Species, y = Petal.Width)) +
  geom_boxplot() +
  ggtitle("Boxplot of Petal Width by Species") +
  theme_minimal()
###########

# Install the e1071 package if you haven't
install.packages("e1071")
```

```
# Load the library
library(e1071)

# Calculate skewness and plot
skewness_values <- data_outliers_imputed %>%
  select(where(is.numeric)) %>%
  summarise(across(everything(), skewness))

print(skewness_values)

# Plotting skewness
ggplot(gather(skewness_values), aes(x = key, y = value)) +
  geom_bar(stat = "identity") +
  ggtitle("Skewness of Numeric Variables") +
  ylab("Skewness") +
  xlab("Variables") +
  theme_minimal()

print("21BDS0085 JvnGanesh")
# Function to Impute Outliers Manually Using IQR
impute_outliers_iqr <- function(x) {
  Q1 <- quantile(x, 0.25, na.rm = TRUE)
  Q3 <- quantile(x, 0.75, na.rm = TRUE)
  IQR <- Q3 - Q1
  # Define the lower and upper bounds
  lower_bound <- Q1 - 1.5 * IQR
  upper_bound <- Q3 + 1.5 * IQR

  # Replace outliers with the median
  x[x < lower_bound] <- median(x, na.rm = TRUE)
  x[x > upper_bound] <- median(x, na.rm = TRUE)

  return(x)
}

# Apply to Numeric Columns
data_outliers_imputed <- data %>%
  mutate(across(where(is.numeric), impute_outliers_iqr))

print(summary(data_outliers_imputed))
##########
# Plot distribution of each numeric variable
data_outliers_imputed %>%
  select(where(is.numeric)) %>%
  gather(key = "Variable", value = "Value") %>%
  ggplot(aes(x = Value)) +
  geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
  facet_wrap(~Variable, scales = "free_x") +
  ggtitle("Distribution of Numeric Variables") +
  theme_minimal()
```

```
#########
print("21BDS0085 JvnGanesh")
# Check for Skewness and Transform
skewed_variables <- find_skewness(data_outliers_imputed)
print(skewed_variables)

print("21BDS0085 JvnGanesh")
# Plot Skewness
plot_skewness(data_outliers_imputed)

print("21BDS0085 JvnGanesh")
# Apply log transformation only to numeric variables
data_transformed <- data_outliers_imputed %>%
  mutate(across(where(is.numeric), log))

# Check the transformed data
summary(data_transformed)

# Plot the transformed data distributions
data_transformed %>%
  select(where(is.numeric)) %>%
  gather(key = "Variable", value = "Value") %>%
  ggplot(aes(x = Value)) +
  geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
  facet_wrap(~Variable, scales = "free_x") +
```

```r
  ggtitle("Log-Transformed Numeric Variables") +
  theme_minimal()



print("21BDS0085 JvnGanesh")


print("21BDS0085 JvnGanesh")
# Apply Z-score standardization only to numeric variables
data_standardized <- data_transformed %>%
  mutate(across(where(is.numeric), ~ scale(.) %>% as.vector()))

# Check the standardized data
summary(data_standardized)

# Plot the standardized data distributions
data_standardized %>%
  select(where(is.numeric)) %>%
  gather(key = "Variable", value = "Value") %>%
  ggplot(aes(x = Value)) +
  geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
  facet_wrap(~Variable, scales = "free_x") +
  ggtitle("Standardized Numeric Variables") +
  theme_minimal()



print("21BDS0085 JvnGanesh")
```

```
# Define a function to bin data into 3 bins (low, medium, high)
bin_data <- function(x, bins = 3) {
  cut(x, breaks = bins, labels = paste("Bin", 1:bins), include.lowest = TRUE)
}

# Apply binning to all numeric columns
data_binned <- data_standardized %>%
  mutate(across(where(is.numeric), bin_data))

# Check the binned data
summary(data_binned)

# View a sample of the binned data
print(head(data_binned))



print("21BDS0085 JvnGanesh")
# Transformation Report
transformation_web_report(data_binned, output_format = "html")
transformation_paged_report(data_binned)

print("21BDS0085 JvnGanesh")
# Arrange Observations by a Specific Variable
data_arranged <- arrange(data_binned, Sepal.Length)
print(head(data_arranged))
print("21BDS0085 JvnGanesh")
# Select Specific Columns
selected_data <- select(data_arranged, Sepal.Length, Sepal.Width, Species)
print(head(selected_data))

print("21BDS0085 JvnGanesh")
# Filter Observations Based on Values
filtered_data <- filter(data_arranged, Species == "setosa")
print(head(filtered_data))

print("21BDS0085 JvnGanesh")
# Gather (Convert Wide Data to Long Format)
gathered_data <- gather(data_arranged, key = "Measurement", value = "Value", -Species)
print(head(gathered_data))

print("21BDS0085 JvnGanesh")
# Spread (Convert Long Data to Wide Format)
spread_data <- spread(gathered_data, key = "Measurement", value = "Value")
print(head(spread_data))

print("21BDS0085 JvnGanesh")
# Group Data by a Variable and Summarize
grouped_data <- data_arranged %>%
  group_by(Species) %>%
  summarize(mean_sepal_length = mean(Sepal.Length), mean_sepal_width = mean(Sepal.Width))

print(grouped_data)


print("21BDS0085 JvnGanesh")

# Mutate (Create New Variables)

mutated_data <- mutate(data_arranged, Sepal.Ratio = Sepal.Length / Sepal.Width)

print(head(mutated_data))


#####################################################################################################

# Load necessary libraries

install.packages("corrplot")

install.packages("naniar")

install.packages("DataExplorer")

install.packages("Hmisc")

install.packages("caret")

install.packages("ggcorrplot")

install.packages("psych")

library(dplyr)

library(ggplot2)

library(corrplot)

library(DataExplorer)

library(Hmisc)

library(caret)

library(psych)
```

```r
# View basic summary statistics
summary(data)

# View structure of the dataset
str(data)

# Check for missing values
sum(is.na(data))

# Visualize missing data
library(naniar)
gg_miss_var(data) + theme_minimal()

# Plot distributions of all numeric variables
data %>%
  select(where(is.numeric)) %>%
  gather(key = "Variable", value = "Value") %>%
  ggplot(aes(x = Value)) +
  geom_histogram(bins = 30, fill = "blue", color = "black", alpha = 0.7) +
  facet_wrap(~ Variable, scales = "free_x") +
  theme_minimal()

# Correlation matrix
cor_matrix <- cor(data %>% select(where(is.numeric)), use = "complete.obs")
corrplot(cor_matrix, method = "color", type = "upper", tl.cex = 0.8)
```

```
# Pair plot (scatterplot matrix)

pairs(data %>% select(where(is.numeric)))


# Create a correlation heatmap

library(ggcorrplot)

corrplot <- cor(data %>% select(where(is.numeric)))

ggcorrplot(corrplot, lab = TRUE)


# Create a QQ plot for checking normality

ggplot(data, aes(sample = Sepal.Length)) +

  stat_qq() +

  stat_qq_line() +

  ggtitle("QQ Plot of Sepal Length") +

  theme_minimal()


# Outlier detection with boxplots

ggplot(data, aes(x = Species, y = Sepal.Length)) +

  geom_boxplot() +

  ggtitle("Boxplot of Sepal Length by Species") +

  theme_minimal()


# Visualize distributions using density plots

ggplot(data, aes(x = Sepal.Length, fill = Species)) +

  geom_density(alpha = 0.5) +

  theme_minimal()
```

```
# Visualize relationships with scatterplots
ggplot(data, aes(x = Sepal.Length, y = Petal.Length, color = Species)) +
  geom_point() +
  theme_minimal()

# Generate a report using DataExplorer
create_report(data)

# Check multicollinearity (Variance Inflation Factor - VIF)
vif(lm(Sepal.Length ~ ., data = data))

# Summary statistics using the psych package
describe(data)

# Check for zero variance predictors
nearZeroVar(data, saveMetrics = TRUE)

# Detect highly correlated variables and remove them
highly_correlated <- findCorrelation(cor_matrix, cutoff = 0.75)
data_reduced <- data[, -highly_correlated]

# Scatterplot matrix (SPLOM) using the psych package
pairs.panels(data %>% select(where(is.numeric)), method = "pearson", hist.col = "blue")

# Visualize the distribution of all variables using plotly
install.packages("plotly")
```

```
library(plotly)

plot_ly(data, x = ~Sepal.Length, y = ~Petal.Length, color = ~Species, type = "scatter", mode = "markers")


# Visualize data using a parallel coordinates plot

install.packages("MASS")

library(MASS)

parcoord(data %>% select(where(is.numeric)), col = as.factor(data$Species))


# Binning continuous variables

data_binned <- data %>%

  mutate(Sepal.Length.Binned = cut(Sepal.Length, breaks = 3, labels = c("Short", "Medium", "Long")))


# Group by and summarize data

grouped_summary <- data %>%

  group_by(Species) %>%

  summarize(mean_sepal_length = mean(Sepal.Length), mean_petal_length = mean(Petal.Length))


# Plot a heatmap for categorical variables

install.packages("pheatmap")

library(pheatmap)

data_cat <- data %>%

  mutate_if(is.factor, as.numeric)

pheatmap(cor(data_cat))
```

```
# Feature engineering: Create new variable based on existing data
data <- data %>%
  mutate(Sepal.Ratio = Sepal.Length / Sepal.Width)

# Extracting insights using Hmisc::describe
describe(data)
```