

# ASSIGNMENT 2 :SIT718

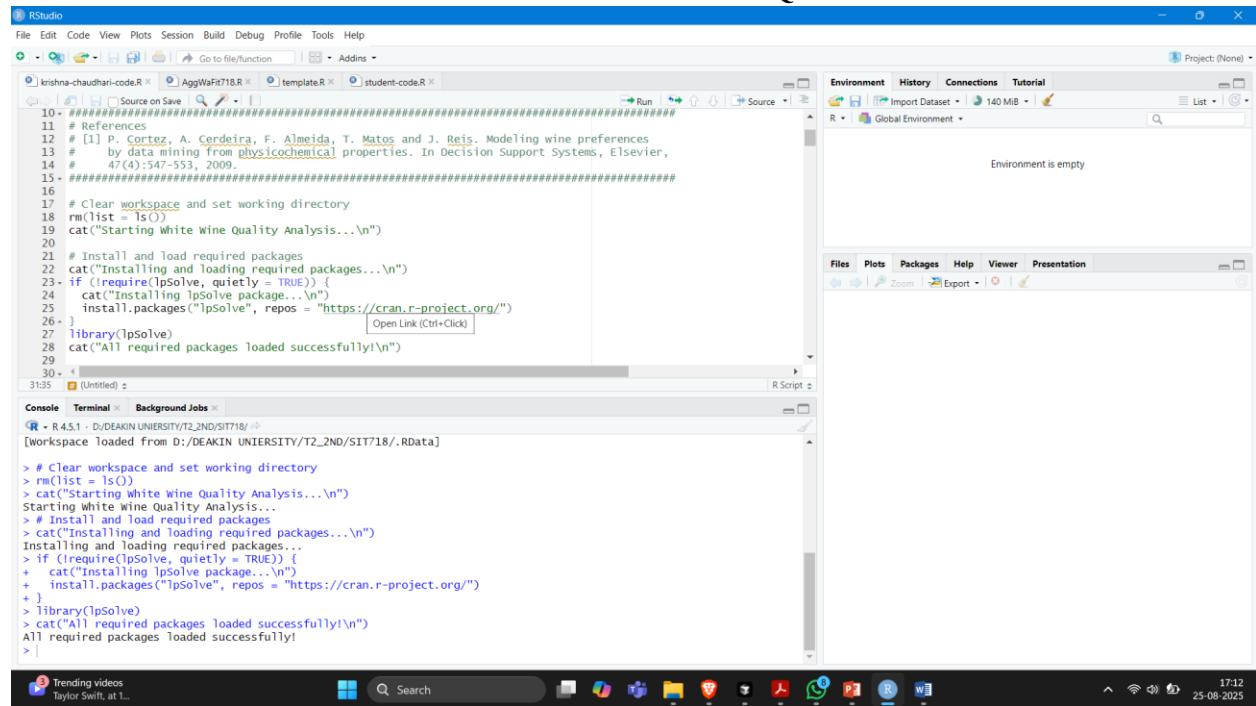
## REAL WORLD ANALYTICS

**GITHUB REPO LINK :-**

**YOUTUBE LINK :- <https://youtu.be/EP SnoEJpf5s>**

**CODE OUTPUT PROOF :-**

**1> IN THIS IMAGE WE ARE INSTALLING ALL THE REQUIRED PACKAGES :**



The screenshot shows the RStudio interface with the following details:

- File Menu:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Project:** Project: (None)
- Source Editor:** Shows a script named "student-code.R" with the following content:

```
10 - ##### References #####
11 # [1] P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modeling wine preferences
12 # by data mining from physicochemical properties. In Decision Support Systems, Elsevier,
13 # 47(4):547-553, 2009.
14 # #####
15 #
16 # Clear workspace and set working directory
17 rm(list = ls())
18 cat("Starting White Wine Quality Analysis...\n")
19
20 # Install and load required packages
21 cat("Installing and loading required packages...\n")
22 if (!require(lpSolve, quietly = TRUE)) {
23   cat("Installing lpSolve package...\n")
24   install.packages("lpSolve", repos = "https://cran.r-project.org/")
25 }
26 library(lpSolve)
27 cat("All required packages loaded successfully!\n")
28
29
30 - 
```
- Environment Tab:** Shows "Environment is empty".
- Plots Tab:** Shows "Files", "Plots", "Packages", "Help", "Viewer", "Presentation".
- Console:** Shows the R session output identical to the Source Editor.
- Taskbar:** Shows the R icon, a search bar, and various system icons.
- System Bar:** Shows the date and time (25-08-2025, 17:12).

**2> IN THIS SCREENSHOT QUESTION 1 SOLUTION HAS BEEN STARTED IN WHICH WE HAD DONE IMPORTING CSV. FILE , GETTING THE DIMENSIONS FOR THE DATASET AND PRINT THE ROWS OF DATA SET**

The screenshot shows the RStudio interface with the following details:

- File Menu:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Toolbar:** Source on Save, Run, Source.
- Code Editor:** Displays R code for reading and analyzing a wine dataset. The code includes importing the dataset, displaying basic information, and generating a subset of 1500 data points.
- Console:** Shows the R session output, including the command to import the dataset and the resulting data frame structure.
- Environment Tab:** Shows the global environment with a data.raw object of type num [1:4897, 1:7].
- Plots Tab:** Shows a histogram of the fixed.acidity variable.
- Help Tab:** Shows the help page for the hist function.
- Presentation Tab:** Shows a presentation slide with the title "Wine Quality Prediction" and a subtitle "Using R".
- Bottom Status Bar:** Shows the date (25-08-2025), time (17:15), and weather (18°C, Cloudy).

### **3. NOW WE ARE GENERATING THE SUBSET FOR THE 1500 SAMPLES .**

The screenshot shows the RStudio interface with the following details:

- File Menu:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Project:** Project (None) is selected.
- Code Editor:** The script editor contains R code for generating a subset of 1500 data points from a raw dataset. It includes comments for steps (i), (ii), and (iii). The code uses `set.seed(223751702)` and `sample(1:4897, 1500)` to ensure reproducibility.
- Environment Tab:** Shows the global environment with two objects:
  - `data.raw`: num [1:4897, 1:7] 7 6.3 8.1 7.2 7.2 8.1 ...
  - `data.subset`: num [1:1500, 1:7] 6.3 4.7 7.4 8 7.1 6.8 ...
- Console Tab:** Displays the R session history, including the creation of a subset from `Whitewine.csv` and the execution of the R code in the script editor.
- Plots Tab:** Available but currently empty.
- Packages Tab:** Available but currently empty.
- Help Tab:** Available but currently empty.
- Viewer Tab:** Available but currently empty.
- Presentation Tab:** Available but currently empty.
- Bottom Bar:** Includes icons for 18°C Cloudy weather, search, and various system status indicators.

#### 4> AS AN OUTPUT WE GOT AN DATA.RAW

The screenshot shows the RStudio interface with the following details:

- Environment Pane:** Shows two variables: `data.raw` (a numeric vector of length 4897) and `data.subset` (a numeric vector of length 1500).
- Data View:** Displays the first 13 rows of the dataset, which includes columns: fixed.acidity, volatile.acidity, residual.sugar, free.sulfur.dioxide, total.sulfur.dioxide, alcohol, and quality.
- Console:** Displays the R code used to generate the subset, including importing the CSV file, setting a seed, and creating a sample of 1500 rows.
- System Tray:** Shows the date and time as 25-08-2025 at 17:19.

#### 5> AND ANOTHER ONE IS DATASUB.SET

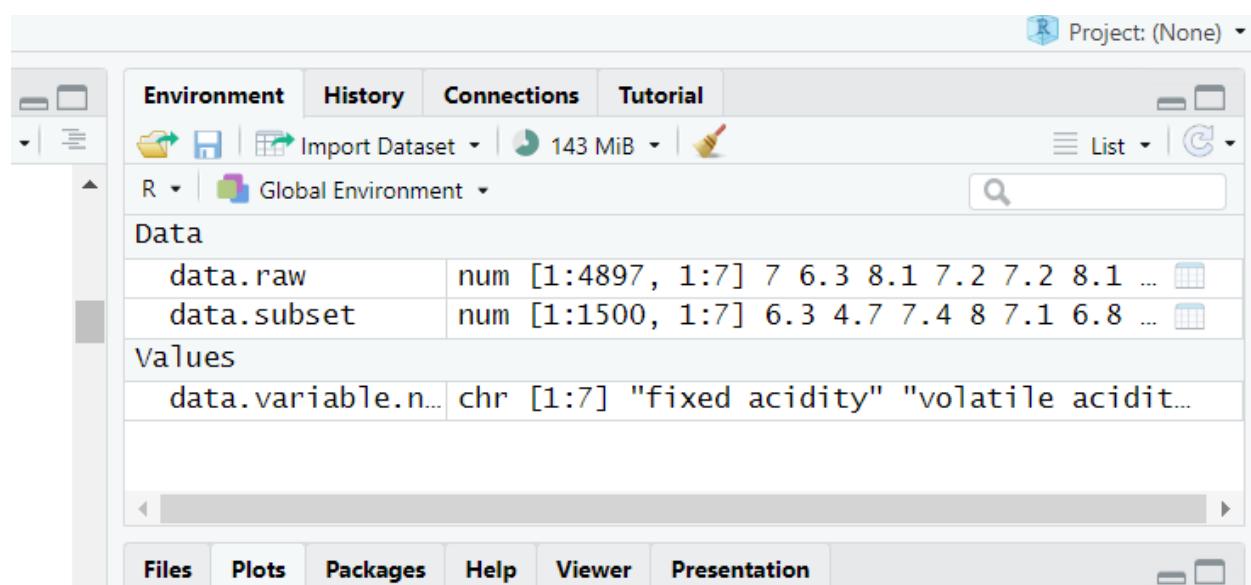
The screenshot shows the RStudio interface with the following details:

- Environment Pane:** Shows two variables: `data.raw` (a numeric vector of length 4897) and `data.subset` (a numeric vector of length 1500).
- Data View:** Displays the first 13 rows of the dataset, which includes columns: fixed.acidity, volatile.acidity, residual.sugar, free.sulfur.dioxide, total.sulfur.dioxide, alcohol, and quality.
- Console:** Displays the R code used to generate the subset, including importing the CSV file, setting a seed, and creating a sample of 1500 rows.
- System Tray:** Shows the date and time as 25-08-2025 at 17:20 and a weather notification for 'Severe weather...'.

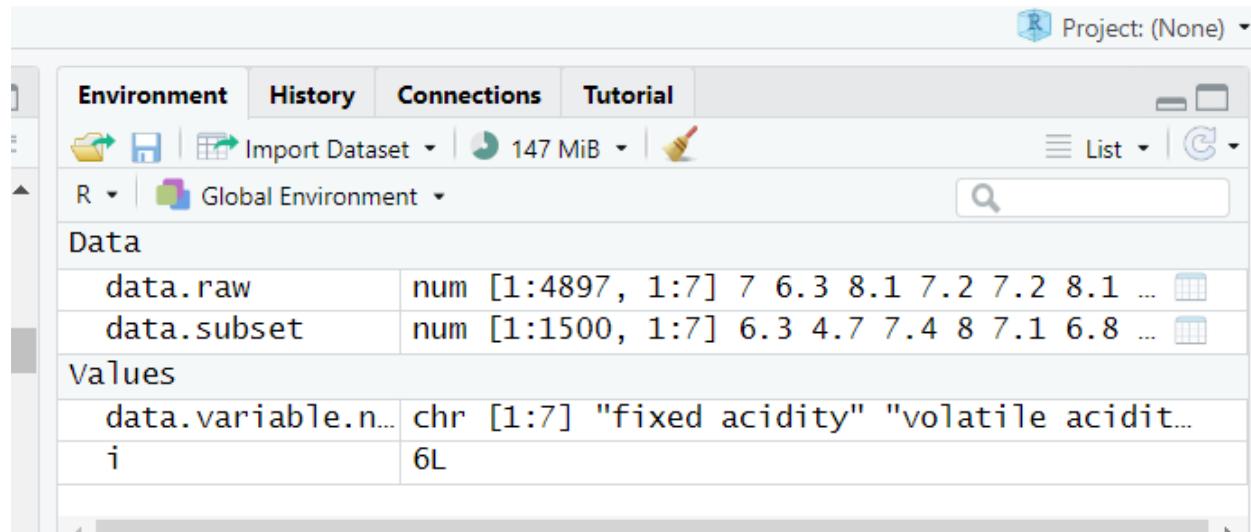
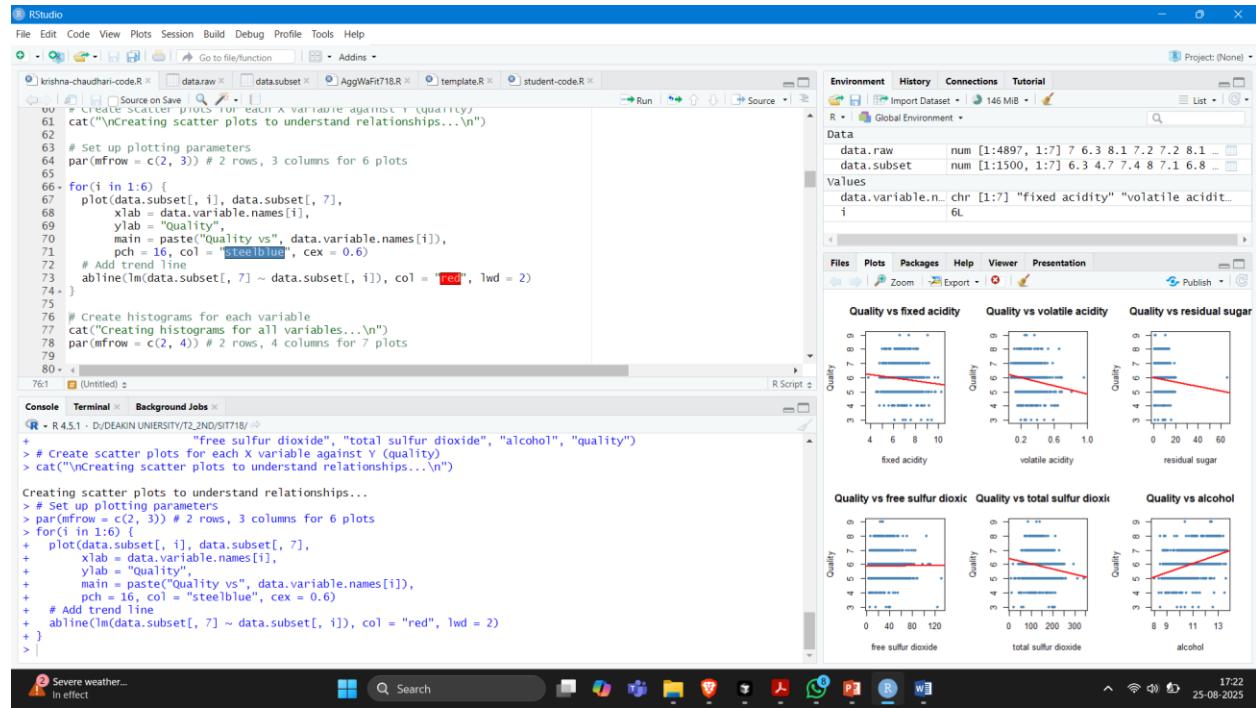
## 6> NOW WE CREATE THE VARIABLE NAMES OF OWN UNDERSTANDING

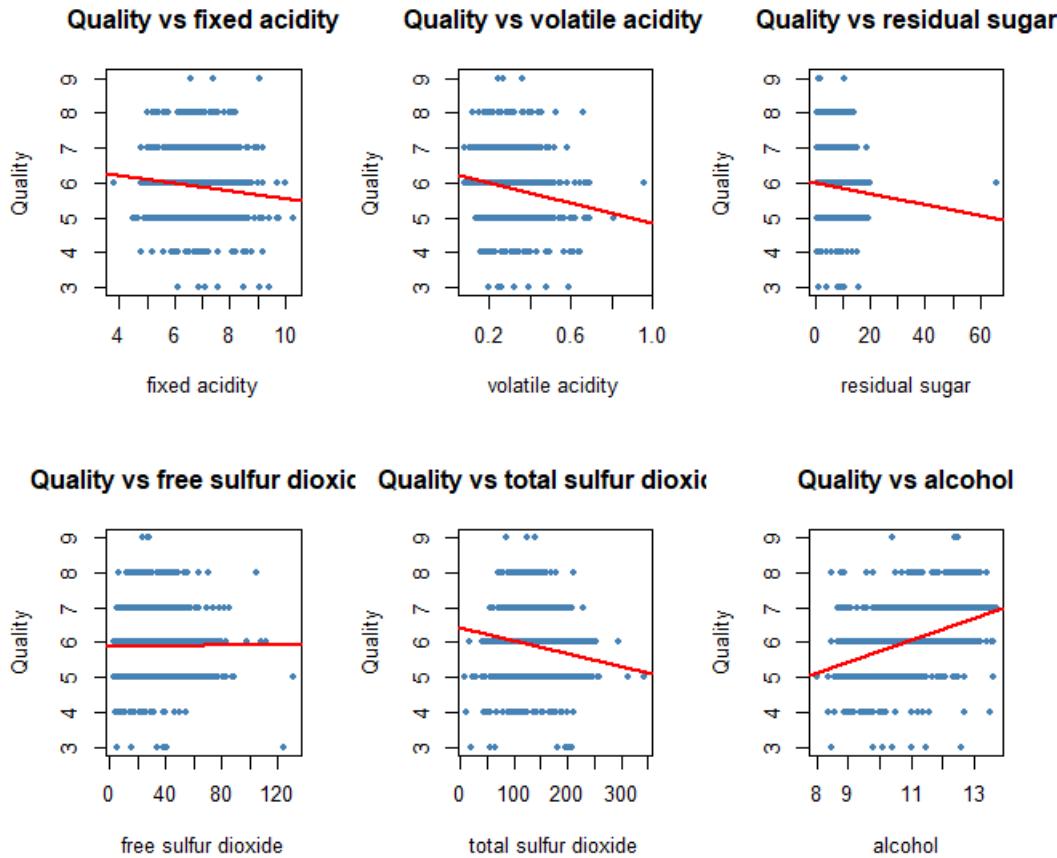
The screenshot shows the RStudio interface with the following details:

- File Bar:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Project Bar:** Project: (None).
- Source Editor:** Displays R script code. The code generates a subset of 1500 data points from a larger dataset, creates variable names for better understanding, and generates scatter plots.
- Console:** Shows the R session history, including commands like `set.seed`, `View`, and `cat` for generating subsets and variable names.
- Environment Tab:** Shows the global environment with objects `data.raw` and `data.subset`.
- Data View:** Shows the structure of `data.raw` and `data.subset`.
- Plots Tab:** Shows options for zooming and exporting plots.
- System Tray:** Shows a weather icon (Severe weather) and a date/time stamp (17:21, 25-08-2025).



## 7> NOW WE PLOT THE SCATTER GRAPH FOR EACH VARIABLES , Y QUANTITY :





## 8> NOW WE ARE CREATING THE HISTROGRAMS FOR EACH VARIABLES :

The screenshot shows the RStudio interface with the code for creating histograms for each variable in the data subset. The code uses a loop to iterate through the variables, creating histograms with lightblue bars and black borders. It also includes a section to display summary statistics and a section to reset plotting parameters. The resulting histograms are displayed in the Plots pane, showing the distribution of fixed acidity, volatile acidity, residual sugar, free sulfur dioxide, total sulfur dioxide, and alcohol.

```

# Create histograms for each variable
cat("Creating histograms for all variables...\n")
par(mfrow = c(2, 4)) # 2 rows, 4 columns for 7 plots

for(i in 1:7) {
  hist(data.subset[, i],
       main = paste("Distribution of", data.variable.names[i]),
       xlab = data.variable.names[i],
       col = "lightblue", border = "black")
}

# Reset plotting parameters
par(mfrow = c(1, 1))

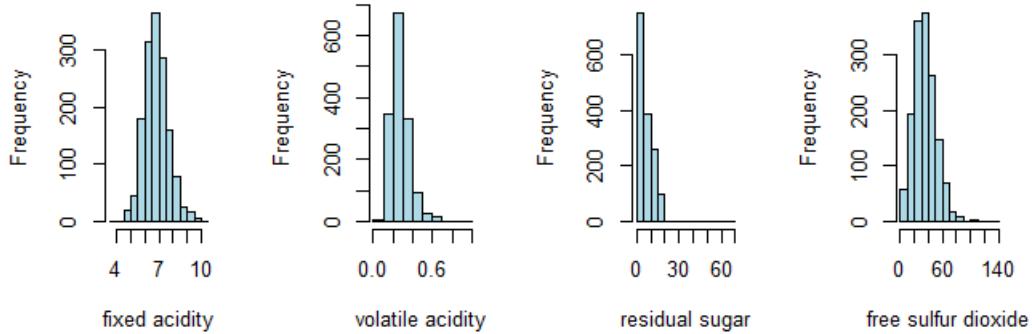
# Display summary statistics
cat("\nSummary statistics for the subset:\n")
summary.stats <- summary(data.subset)
print(summary.stats)

# Create histograms for each variable
cat("Creating histograms for all variables...\n")
Creating histograms for all variables...
par(mfrow = c(2, 4)) # 2 rows, 4 columns for 7 plots
for(i in 1:7) {
  hist(data.subset[, i],
       main = paste("Distribution of", data.variable.names[i]),
       xlab = data.variable.names[i],
       col = "lightblue", border = "black")
}

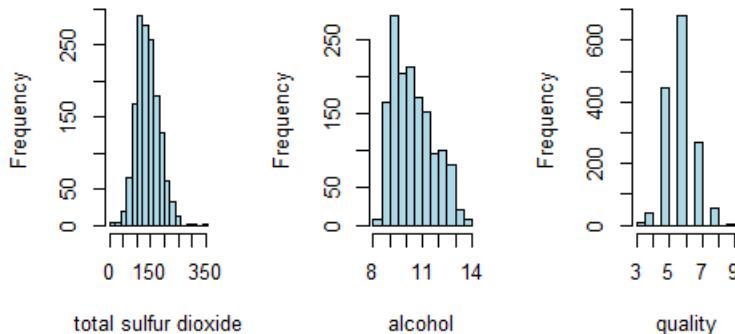
# Reset plotting parameters
par(mfrow = c(1, 1))

```

## Distribution of fixed acidity of volatile acidity of residual tribution of free sulfur



## Distribution of total sulfur      Distribution of alcohol      Distribution of quali



## 9> THEN WVE SUMMARISED THE SUBSET :

```

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Source on Save
77 cat("Creating histograms for all variables...\n")
78 par(mfrow = c(2, 4)) # 2 rows, 4 columns for 7 plots
79
80 for(i in 1:7) {
81   hist(data.subset[, i],
82       main = paste("Distribution of", data.variable.names[i]),
83       xlab = data.variable.names[i],
84       col = "lightblue", border = "black")
85 }
86
87 # Reset plotting parameters
88 par(mfrow = c(1, 1))
89
90 # Display summary statistics
91 cat("\nSummary statistics for the subset:\n")
92 summary.stats <- summary(data.subset)
93 print(summary.stats)
94
95 #####
96 # Question 2 - Transform the Data
97 +
98 (Untitled) 2
Console Terminal < Background Jobs <
R - R 4.5.1 - /DEAKIN UNIVERSITY/T2_2ND/SIT718/ >
> cat("Creating histograms for all variables...\n")
Creating histograms for all variables...
> par(mfrow = c(2, 4)) # 2 rows, 4 columns for 7 plots
> for(i in 1:7) {
+   hist(data.subset[, i],
+       main = paste("Distribution of", data.variable.names[i]),
+       xlab = data.variable.names[i],
+       col = "lightblue", border = "black")
+ }
> # Reset plotting parameters
> par(mfrow = c(1, 1))
> # Display summary statistics
> cat("\nSummary statistics for the subset:\n")
Summary statistics for the subset:
> summary.stats <- summary(data.subset)
> |
Environment History Connections Tutorial
Project: (None)
Data
data.raw      num [1:4897, 1:7] 7 6.3 8.1 7.2 7.2 8.1 ...
data.subset    num [1:1500, 1:7] 6.3 4.7 7.4 8 7.1 6.8 ...
Values
data.variable.names chr [1:7] "fixed acidity" "volatile acidit...
i             7L
summary.stats  "table" chr [1:6, 1:7] "Min. : 3.800 " ...
Files Plots Packages Help Viewer Presentation
Zoom Export
R Script
Distribution of fixed acidity of volatile acidity of residual tribution of free sulfur
fixed acidity      volatile acidity      residual sugar      free sulfur dioxide
Distribution of total sulfur      Distribution of alcohol      Distribution of quali
total sulfur dioxide      alcohol      quality

```

95:1 # (Untitled) ◊

Console Terminal × Background Jobs ×

R - R 4.5.1 · D:/DEAKIN UNIVERSITY/T2\_2ND/SIT718/

```
> summary_stats <- summary(data_subset)
> print(summary_stats)
```

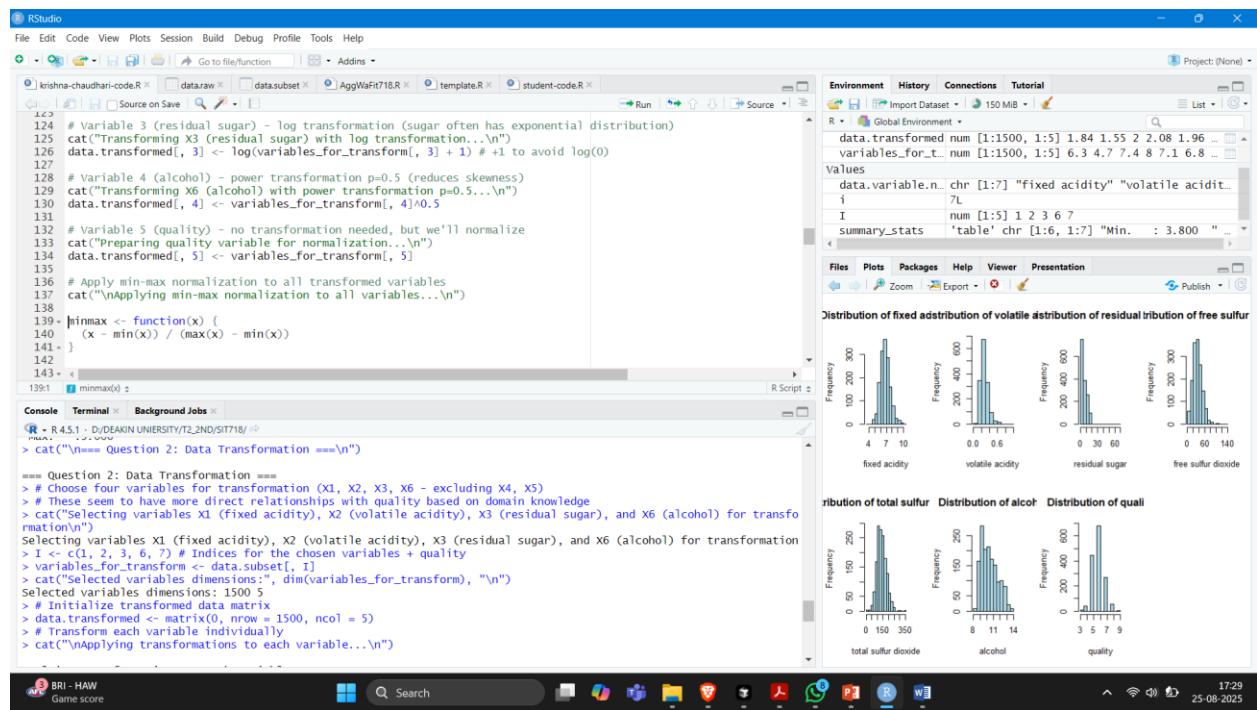
	fixed.acidity	volatile.acidity	residual.sugar	free.sulfur.dioxide	total.sulfur.dioxide	alcohol
Min.	3.800	0.0800	0.600	3.00	9.0	8.00
1st Qu.	6.300	0.2100	1.700	24.00	110.0	9.40
Median	6.800	0.2600	5.000	34.00	134.0	10.30
Mean	6.851	0.2775	6.295	35.46	138.1	10.48
3rd Qu.	7.400	0.3300	9.800	45.00	165.0	11.30
Max.	10.300	0.9650	65.800	131.00	344.0	13.70

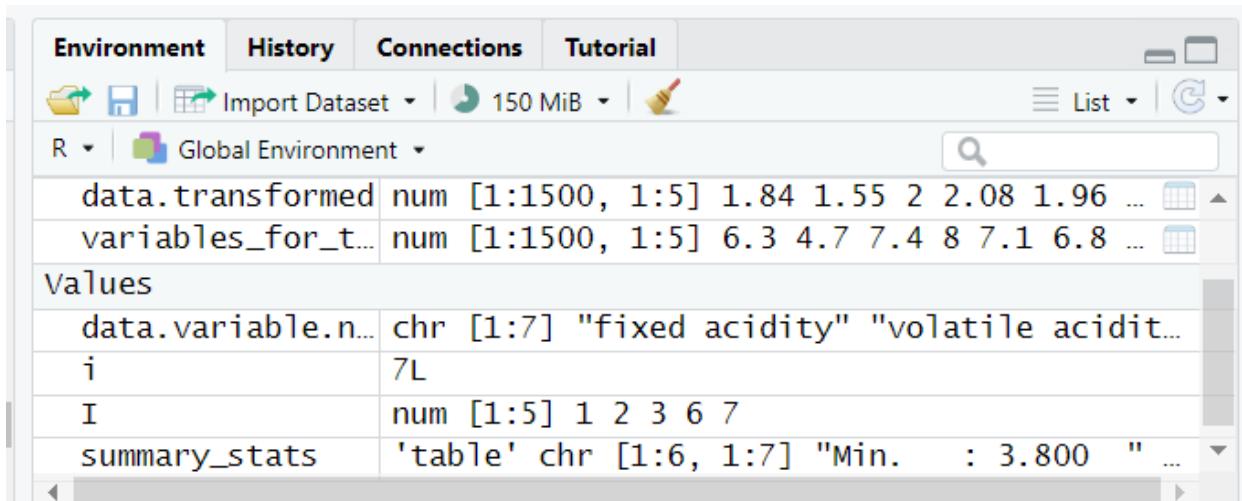
quality

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Min.	3.000	5.000	6.000	5.897	6.000	9.000

## QUESTION 2 :

**10 > HERE WE HAD APPLIED TRANSFORMATION TO EACH VARIABLE INDIVIDUALLY**





## 11> NOW APPLYING THE MIN-MAX NORMALIZATION TO ALL THE TRANSFORMED VARIABLES

The screenshot shows the RStudio interface with the following details:

- File Menu:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Project:** Project: (None)
- Code Editor:** The code editor displays R code for transforming and normalizing a dataset. It includes comments explaining the steps: "data transformation required, but we'll normalize", "Preparing quality variable for normalization...", and "# Apply min-max normalization to all transformed variables". The code uses the `minmax` function to calculate normalized ranges for each variable.
- Console:** The console shows the execution of the script, saving the transformed data to a file named "krishna-chaudhari-transformed.txt".
- Environment:** The environment pane shows the global environment with objects like `data.variable.n` (fixed acidity), `I` (volatile acidity), and `summary\_stats` (table).
- Plots:** Six histograms are displayed in the plots pane, showing the distribution of fixed acidity, volatile acidity, residual sugar, free sulfur dioxide, total sulfur dioxide, alcohol, and quality.

**12>HERE ONCE THE TRANSFORMATION HAD BEEN DONE SO CHANGES HAS BEEN SAVE IN THE FILE NAMED : KRISHNA-CHAUDHARI-TRANSFORMED.TXT**

The screenshot shows the RStudio interface with the following details:

- Code Editor:** Displays an R script named "krishna-chaudhari-transformed.R". The script contains code to calculate minmax values, normalize data, and save the transformed data to a file named "krishna-chaudhari-transformed.txt". It also includes comments for Question 3.
- Console:** Shows the output of the R script, including messages about normalized ranges and the saved file.
- Plots:** A grid of histograms showing the distribution of various wine quality variables. The variables include fixed acidity, volatile acidity, residual sugar, total sulfur dioxide, alcohol, and quality.
- Environment:** Shows the global environment with objects like "data.variable.n\_1" and "summary.stats".
- File Explorer:** Shows files like "krishna-chaudhari-code.R", "minmax.R", "data.raw", "data.subset", "AggWaFr718.R", "template.R", and "student-code.R".
- Taskbar:** Shows system icons for weather (17°C), search, task manager, and file explorer.

Rplot 1	25-08-2025 17:24	PNG File	11 KB
Rplot 2	25-08-2025 17:25	PNG File	11 KB
krishna-chaudhari-transformed	25-08-2025 17:32	Text Document	124 KB

SIT718

Dmain Untitled H! I'm Which Additi krishna-chak krishna-chak krishna-chak krishr

File Edit View H! B A D:\DEAKIN UNIVERSITY\T2\_2ND\SIT718\krishna-chaudhari-transformed.txt

```

0.506997144455218 0.250771265528877 0.119593527949665 0.400518945190083 0.5
0.21317050679365 0.765820556061023 0.0597967639748323 0.984496563267642 0.333333333333333
0.66838634432837 0.16748684665381 0.594325425582758 0.138785414069726 0.6666666666666667
0.746573563651793 0.310445812494945 0.08537454228224 0.272091068717412 0.333333333333333
0.626884834796655 0.365508579001293 0.45682888305648 0.50798363885634 0.5
0.583588959189736 0.352118725894512 0.130103675850148 0.436388724784221 0.5
0.612659555742853 0.234982914516966 0.38376721724886 0.64465859378999 0.5
0.568731422875326 0.363508579001293 0.340451465123551 0.364292225989803 0.5
0.681850116527924 0.250771265528877 0.119593527949665 0.610729170366146 0.5
0.365508579001293 0.4312881928088 0.593651902470015 0.833333333333333
0.538339237694775 0.159366292893715 0.72821288567522 0.6666666666666667
0.598229589402226 0.18508589958136 0.130103675850148 0.627731040831311 0.333333333333333
0.49950956397627 0.16748684665381 0.0728712799711503 0.559267103253344 0.5
0.733958738728722 0.42923783604579 0.413993519965718 0.810043941343944 0.5
0.783493022000575 0.453404778336079 0.617033907217216 0.119309971283651 0.333333333333333
0.522790626108973 0.16748684665381 0.085337454228224 0.610729170366146 0.6666666666666667
0.708242833363562 0.281258098210371 0.533157945468127 0.298723914769864 0.5
0.406626910927879 0.56524586367018 0.108654277601926 0.158149246976831 0.333333333333333
0.406911152366159 0.716762866079546 0.108654277601926 0.327697694546226 0.1666666666666667
0.522790626108973 0.533095613290302 0.108654277601926 0.627731040831311 0.333333333333333

```

Ln 1. Col 1 1,24,756 characters Plain text

krishna-chaudhari-wam-output 25-08-2025 17:35 Text Document 150 KB

krishna-chaudhari-wam-stats 25-08-2025 17:35 Text Document 1 KB

17 items | 1 item selected 123 KB |

17°C Cloudy 17:36 25-08-2025

### QUESTION 3 :

#### 13> NOW WE WILL BE BUILDING THE MODEL AND INVESTIGATE IT .

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

krishna-chaudhari-code.R minmax data.raw data.subset AggWaFit18.R template.R student-code.R

```

161 cat("Loading AggWaFit18.R...\n")
162 source("AggWaFit18.R")
163
164 # Read the saved transformed data
165 cat("Reading transformed data..\n")
166 data.transformed_copy <- as.matrix(read.table("krishna-chaudhari-transformed.txt"))
167
168 # Verify data structure
169 cat("Transformed data dimensions:", dim(data.transformed_copy), "\n")
170
171 # (i) Get weights for weighted Arithmetic Mean (WAM)
172 cat("Fitting weighted Arithmetic Mean (WAM)...\n")
173 wam_result <- fit.QAM(data.transformed_copy, "krishna-chaudhari-wam-output.txt", "krishna-chaudhari-wam-stats.txt")
174 cat("WAM fitting completed. Check output files for results.\n")
175
176 # (ii) Get weights for Power Mean p=0.5
177 cat("Fitting Power Mean p=0.5...\n")
178 pm05_result <- fit.QAM(data.transformed_copy, "krishna-chaudhari-pm05-output.txt", "krishna-chaudhari-pm05-stats.txt")
179 cat("Power Mean p=0.5 fitting completed.\n")
180
1761 (Untitled) z

```

Console Terminal Background Jobs

R 4.5.1 - D:\DEAKIN UNIVERSITY\T2\_2ND\SIT718\ Loading AggWaFit18.R...
> source("AggWaFit18.R")
> # Read the saved transformed data
> cat("Reading transformed data...\n")
Reading transformed data...
> data.transformed\_copy <- as.matrix(read.table("krishna-chaudhari-transformed.txt"))
> # Verify data structure
> cat("Transformed data dimensions:", dim(data.transformed\_copy), "\n")
Transformed data dimensions: 1500 5
> # (i) Get weights for weighted Arithmetic Mean (WAM)
> cat("Fitting weighted Arithmetic Mean (WAM)...\n")
--- Fitting weighted Arithmetic Mean (WAM) ---
> wam\_result <- fit.QAM(data.transformed\_copy, "krishna-chaudhari-wam-output.txt", "krishna-chaudhari-wam-stats.txt")
> cat("WAM fitting completed. Check output files for results.\n")
WAM fitting completed. Check output files for results.

Environment History Connections Tutorial

I num [1:5] 1 2 3 6 7
summary\_stats "table" chr [1:6, 1:7] "Min. : 3.800 "
wam\_result NULL

Functions

AM function (x)
choquet function (x, v)
f.plot3d function (f, x.dom = c(0, 1), y.dom = c(0, 1), z.dom = c(0, 1))

Files Plots Packages Help Viewer Presentation

Distribution of fixed acidity Distribution of volatile acidity Distribution of residual sugar Distribution of free sulfur dioxide

Distribution of total sulfur Distribution of alcohol Distribution of quality

17:37 25-08-2025

Krishna-chaudhari-transformed	25-08-2025 17:32	Text Document	124 KB
krishna-chaudhari-wam-output	25-08-2025 17:35	Text Document	150 KB
krishna-chaudhari-wam-stats	25-08-2025 17:35	Text Document	1 KB

SIT718

```

RMSE 0.141098658198263
Av. abs error 0.110080573560074
Pearson correlation 0.399952485936333
Spearman correlation 0.411174523981722
1 w_1
1 0.281129919399362
2 0
3 0.218178650692157
4 0.500691429908481

```

Ln 1, Col 1 203 characters Plain text 100% Windows (CRLF) UTF-8

Data (D) template 24-08-2025 20:49 R File 4 KB

data WhiteWine 24-08-2025 20:49 Microsoft Excel Co... 130 KB

DEAKIN UNI demo documents

17 items | 1 item selected 212 bytes |

17°C Cloudy 17:38 25-08-2025

SIT718

```

0.506997144455218 0.259771265528877 0.119593527949665 0.400518945190083 0.5 0.3691616173298654
0.213170506579305 0.765820550601023 0.0597967639748323 0.984496563207642 0.333333333333333 0.56590397661328
0.66838634432837 0.1674060465381 0.50435425583758 0.138785414060726 0.666666666666667 0.387061829772253
0.746573563653794 0.310445812494045 0.08533745428224 0.272091068717412 0.333333333333333 0.364736642654151
0.626884834796556 0.3659085759081293 0.45682888305648 0.50798363885634 0.5 0.529806197299684
0.583588959189736 0.352118725894512 0.13010367580148 0.436388724784221 0.5 0.410946256114593
0.612659555742853 0.234962914516966 0.383767212724868 0.44665850378999 0.5 0.578741732242715
0.568731422873326 0.3659085759081293 0.340451465123551 0.364292225989803 0.5 0.416564655895027
0.681850116527924 0.250771265528877 0.119593527949665 0.610729170366164 0.5 0.5235688084458959
0.72118321107832 0.3659085759081293 0.431288128803088 0.593651902470015 0.833333333333333 0.5948080473883322
0.538333923769475 0.159366492893715 0.77821288567522 0.6580723540258666
0.598229589402226 0.185085899581336 0.13010367580148 0.627731040831311 0.333333333333333 0.510865633129804
0.496950956397627 0.16748684665381 0.0728712799711503 0.55926710325334 0.5 0.433940205968111
0.733958738728722 0.42923783604576 0.413993519965718 0.810043941343944 0.5 0.702244367922881
0.783493022800575 0.4534047783336679 0.617033907217216 0.119309971283651 0.333333333333333 0.414624435557297
0.522790626108973 0.16748684665381 0.088337454228224 0.610729170366146 0.666666666666667 0.471377758795221
0.708242833363562 0.281258986210371 0.533157945468127 0.29672391476986 0.5 0.46999490401427
0.522790626108973 0.310445812494045 0.046051473107497 0.52456826804979 0.666666666666667 0.4196666371060207
0.388876618198931 0.441407630914837 0.486750527130771 0.661512527692536 0.666666666666667 0.529282786900774
0.598229589402226 0.352118725894512 0.49375139092577 0.576498225871703 0.333333333333333 0.564553969151766
0.598229589402226 0.310445812494045 0.310645839948788 0.272091068717412 0.5 0.372100192715567
0.598229589402226 0.30445812494045 0.38245086983804 0.333333333333333 0.488831258947032
0.746573563653794 0.465236420479798 0.344713533359428 0.333333333333333 0.657035099592439
0.406626910927879 0.56524586367018 0.108654277601926 0.158149246976831 0.33333333333333 0.217205086981664
0.640911152366159 0.71672866079546 0.188654277601926 0.327697694546228 0.166666666666667 0.367960771546032
0.522790626108973 0.533095613290302 0.108654277601926 0.627731040831311 0.333333333333333 0.484977682691645

```

Ln 1, Col 1 1,51601 characters Plain text 100% Windows (CRLF) UTF-8

Data (D) template 24-08-2025 20:49 R File 4 KB

data WhiteWine 24-08-2025 20:49 Microsoft Excel Co... 130 KB

DEAKIN UNI demo documents

17 items | 1 item selected 149 KB |

17°C Cloudy 17:38 25-08-2025

**14> NOW WILL CHECK THE FITTING POWER MEAN P=0.5 , P=2 , FITTING ORDERED WEIGHTED AVERAGE (OWA), ALL THE MODELS HAS BEEN FITTED AND CHECKED**

## PROPERLY .

The screenshot shows the RStudio interface with several windows open:

- Code Editor:** Displays R script code for fitting various models (WAM, Power Mean p=0.5, Power Mean p=2, OWA) and printing their results to text files.
- Environment:** Shows the global environment with functions like `choquet`, `f.plot3d`, `fit.choquet`, etc.
- Plots:** Four histograms showing the distribution of different wine quality parameters: fixed acidity, volatile acidity, residual sugar, and free sulfur dioxide.
- Console:** Shows the command history and output of the R session.
- File Explorer:** Shows the file structure, including the generated output files.
- Task View:** Shows background jobs.
- System Tray:** Shows weather (17°C), search bar, and system icons.

	krishna-chaudhari-pm05-output	25-08-2025 17:38	Text Document	150 KB
	krishna-chaudhari-pm05-stats	25-08-2025 17:38	Text Document	1 KB
	krishna-chaudhari-pm2-output	25-08-2025 17:39	Text Document	150 KB
	krishna-chaudhari-pm2-stats	25-08-2025 17:39	Text Document	1 KB
	krishna-chaudhari-owa-output	25-08-2025 17:39	Text Document	150 KB
	krishna-chaudhari-owa-stats	25-08-2025 17:39	Text Document	1 KB

LIKE FITTED P=0.5 EACH MODEL HAS CREATED THE OUTPUT FILES .

SIT718

File Edit View H1 B A D:\DEAKIN UNIVERSITY\T2\_2ND\SIT718\krishna-chaudhari-pm05-output.txt

```

0.506997144455218 0.250771265528877 0.119593527949665 0.400518945190083 0.5 0.382252765704103
0.21317050679395 0.765820556061023 0.0597967639748323 0.984496563267642 0.3333333333333333 0.471142639620665
0.66838634432837 0.16748684665381 0.594325425582758 0.138785414069726 0.666666666666667 0.353424523923698
0.746573563651793 0.310445812494945 0.0853745428224 0.272091068717412 0.3333333333333333 0.373817334608426
0.626884834796655 0.365508575001293 0.45682888305648 0.50798363885634 0.5 0.541088813034473
0.583588959189736 0.352118725894512 0.130103675850148 0.436387827478221 0.5 0.42619590516548
0.612659555742853 0.234982914516966 0.38376721724886 0.64465859378999 0.5 0.58999137816962
0.5687314228735 0.363508575001293 0.340451465123551 0.364292252598980 0.5 0.429364977807883
0.681850116527924 0.250771265528877 0.119593527949665 0.610729170366143 0.5 0.53544162705874
0.721183211077832 0.365508575001293 0.4312881928088 0.593651902479013 0.8333333333333333 0.611392566708395
0.538339237694775 0.159366298987315 0.728212885675522 0.666666666666667 0.549434382684931
0.598229589402226 0.18508589958136 0.130103675850148 0.627731040831313 0.3333333333333333 0.518723443705374
0.490950956397627 0.16748684665381 0.0728712979711503 0.55926710325344 0.5 0.5593968756286
0.733958738728722 0.42923783604579 0.413993519965718 0.810043941343944 0.5 0.71487873392646
0.783493022000579 0.4534047783336079 0.617033907217216 0.11930997128365 0.3333333333333333 0.369791036638872
0.522790626108973 0.16748684665381 0.627731040831313 0.515292215917979
0.708242833363562 0.281258098210371 0.533157945468127 0.298723914769864 0.5 0.459630352283491
0.522790626108973 0.310445812494945 0.046051473107497 0.524568266804979 0.6666666666666667 0.418355095190015
0.388876618198031 0.414407630914837 0.406758527130771 0.661512527692534 0.6666666666666667 0.515292215917979
0.598229589402226 0.352118725894512 0.493751390927507 0.576498225071703 0.3333333333333333 0.51734705558181
0.598229589402226 0.310445812494945 0.3106458339948788 0.272091068717412 0.5 0.382614638122317
0.66838634432837 0.324661539756261 0.501959104678403 0.38245086938046 0.3333333333333333 0.496144793389918
0.746573563651794 0.465236424077908 0.3040451465123551 0.744713533359428 0.3333333333333333 0.674245000181046
0.406626910927879 0.56524586367018 0.108654277601926 0.158149246976831 0.3333333333333333 0.225272569890462
0.640911152366159 0.716762866079546 0.108654277601926 0.327697694546226 0.1666666666666667 0.38275473759993
0.522790626108973 0.533095613290302 0.108654277601926 0.627731040831311 0.3333333333333333 0.486096118827391

```

Ln 1. Col 1 1,51612 characters Plain text 100% Windows (CRLF) UTF-8

krisha-chaudhari-wam-output 25-08-2025 17:35 Text Document 150 kB  
krisha-chaudhari-wam-stats 25-08-2025 17:35 Text Document 1 kB  
DEAKIN UNI 24-08-2025 23:29 Markdown Source ... 6 kB  
data 25-08-2025 17:24 PNG File 11 kB  
demo 25-08-2025 17:25 PNG File 11 kB  
documents 25-08-2025 17:06 R File 11 kB

17°C Cloudy 17:43 25-08-2025

## QUESTION 4 :

### 15 > NOW WILL MAKING PREDICTIONS :

RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

krisha-chaudhari-code.R\* minmax\* data.raw\* data.subset\* AggWaFit718.R template.R\* student-code.R

242  
243 cat("Normalized transformed values:", normalized\_new\_input, "\n")  
244  
245 # For prediction, we'll use the best model from Question 3  
246 # Based on typical results, WAM often performs well, so we'll use that  
247 cat("\nUsing WAM model for prediction...\n")  
248  
249 # Read the WAM weights from the output file  
250 #wam\_weights <- read.table("krisha-chaudhari-02-wam-output.txt", header = FALSE)  
251 # Convert to numeric vector for easier handling  
252 #wam\_weights <- as.numeric(wam\_weights[,1])  
253 #cat("WAM weights loaded:", wam\_weights, "\n")  
254  
255 wam\_weights <- read.table("krisha-chaudhari-wam-output.txt", header = FALSE)  
256 wam\_weights <- as.numeric(wam\_weights[,1]) # Convert to numeric vector  
257 cat("WAM weights loaded:", wam\_weights, "\n") # Works as | simple vector  
258 prediction <- sum(normalized\_new\_input \* wam\_weights[1:4])  
259  
260 # Make prediction using WAM  
261  
262 #wam\_weights <- read.table("krisha-chaudhari-02-wam-output.txt", header = FALSE)  
263 wam\_weights <- read.table("krisha-chaudhari-wam-output.txt", header = FALSE)  
264 cat("WAM weights loaded:", wam\_weights, "\n") # Works as | simple vector  
WAM weights loaded: 0.5069971 0.2131706 0.6663886 0.7465736 0.6268848 0.535899 0.6126596 0.5687314 0.6818501 0.7211832  
0.5383392 0.5982296 0.499951 0.7339587 0.783493 0.5227906 0.7082428 0.5227906 0.388766 0.5982296 0.6683886  
0.7465736 0.4066269 0.6409112 0.5227906 0.5687314 0.6951333 0.5536505 0.4580672 0.35227906 0.626596  
0.6126596 0.5536505 0.5336505 0.6683886 0.2905826 0.5227906 0.5069971 0.6683886 0.5383392 0.5536505 0.6951333  
0.6409112 0.1916027 0.583589 0.783493 0.583589 0.8646941 0.5069971 0.6818501 0.5687314 0.7211832 0.4412119 0.7590317  
0.5227906 0.583589 0.5687314 0.583589 0.583589 0.5687314 0.7713369 0.7339587 0.5383392 0.499951 0.2950826 0.5069971 0.626  
8848 0.7082428 0.5687314 0.583589 0.4240685 0.6126596 0.4580672 0.4240685 0.5882296 0.6126596 0.388766 0.5383392 0.640  
9112 0.6126596 0.5687314 0.7339587 0.5356505 0.499951 0.8073719 0.5069971 0.3524048 0.5383392 0.5069971 0.499951 0.4746  
438 0.6951333 0.583589 0.5982296 0.868736 0.5353605 0.4240685 0.499951 0.4240685 0.6683886 0.6951333 0.7465736 0.553650  
5 0.3708065 0.6268848 0.654744 0.5982296 0.6268848 0.7590317 0.7465736 0.654744 0.3524048 0.4580672 0.1755  
Severe weather... In effect 17:43 25-08-2025

Environment History Connections Tutorial Project: (None)

denorm\_predict... -1.0609572850728 final\_prediction -1 i 4L I num [1:5] 1 2 3 6 7 maxs Named num [1:4] 10.3 0.965 65.8 13.7 mins Named num [1:4] 3.8 0.08 0.6 8 new\_input num [1:6] 6.7 0.18 4.7 57 161 10.5 new\_input\_for\_t num [1:5] 6.7 0.18 4.7 10.5

Distribution of fixed acidity Distribution of volatile acidity Distribution of residual sugar Distribution of free sulfur dioxide

fixed acidity volatile acidity residual sugar free sulfur dioxide

Distribution of total sulfur Distribution of alcohol Distribution of quality

total sulfur alcohol quality

17:43 25-08-2025

The screenshot shows an RStudio interface with several tabs open in the top navigation bar: File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help. The main workspace contains R code for reading data, calculating normalized values, and making predictions using a WAM model. The console output shows the prediction result as -0.6768263. The environment pane shows global variables like `normalized_new_input`, `wam_weights`, and `final_prediction`. The right sidebar provides links to R Resources, Posit Support, Manuals, Reference, and Miscellaneous Material.

```
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins

krishna-chaudhari-code.R minmax.R data.raw.R data.subset.R AggWaFit718.R template.R student-code.R
Run Source Environment History Connections Tutorial
Import Dataset 206 MB Global Environment
denorm_predicti... 1.06095752850728 final_prediction <-
i 4L
I num [1:5] 1 2 3 6 7
maxs Named num [1:4] 10.3 0.965 65.8 13.7
mins Named num [1:4] 3.8 0.08 0.6 8
new_input num [1:6] 6.7 0.18 4.7 57 161 10.5
< new_input_for_t num [1:4] 6.7 0.18 4.7 10.5
Files Plots Packages Help Viewer Presentation
Home Find in Topic

R Resources
Learning R Online
CRAN Task Views
R on StackOverflow
Getting Help with R

Posit Support
Posit Community Forum for the RStudio IDE
Posit Cheat Sheets
Posit Packages
Posit Products

Manuals
An Introduction to R
Writing R Extensions
R Data Import/Export

Reference
Packages
Search Engine & Keywords

Miscellaneous Material
```

```
242
243 cat("Normalized transformed values:", normalized_new_input, "\n")
244
245 # For prediction, we'll use the best model from Question 3
246 # Based on typical results, WAM often performs well, so we'll use that
247 cat("\nUsing WAM model for prediction...\n")
248
249 # Read the WAM weights from the output file
250 wam_weights <- read.table("krishna-chaudhari-02-wam-output.txt", header = FALSE)
251 # Convert to numeric vector for easier handling
252 wam_weights <- as.numeric(wam_weights[,1])
253 cat("WAM weights loaded:", wam_weights, "\n")
254
255 wam_weights <- read.table("krishna-chaudhari-wam-output.txt", header = FALSE)
256 wam_weights <- as.numeric(wam_weights[,1]) # Convert to numeric vector
257 cat("WAM weights loaded:", wam_weights, "\n") # Works: simple vector
258 prediction <- sum(normalized_new_input * wam_weights[1:4])
259
260 # Make prediction using WAM
261
262
263
264
265
266
267
268
269
270
271
```

R - R 4.5.1 - Di/DEAKIN UNIVERSITY/T2\_2ND/SIT718/

```
Console Terminal Background Jobs
Predicted normalized quality: -0.6768263
> # Reverse transformation to get quality in original scale
> # We need to denormalize
> denorm_prediction <- prediction * (max(data_subset[, 7]) - min(data_subset[, 7])) + min(data_subset[, 7])
> # Round to nearest integer (quality scores are integers)
> final_prediction <- round(denorm_prediction)
> cat("\n--- Final Results ---\n")

--- Final Results ---
> cat("Predicted wine quality:", final_prediction, "/10\n")
Predicted wine quality: -1 /10
> cat("This prediction is based on the WAM model using the transformed physicochemical properties.\n")
This prediction is based on the WAM model using the transformed physicochemical properties.
> cat("\n==== Analysis Complete ====\n")

==== Analysis Complete ====
>
```