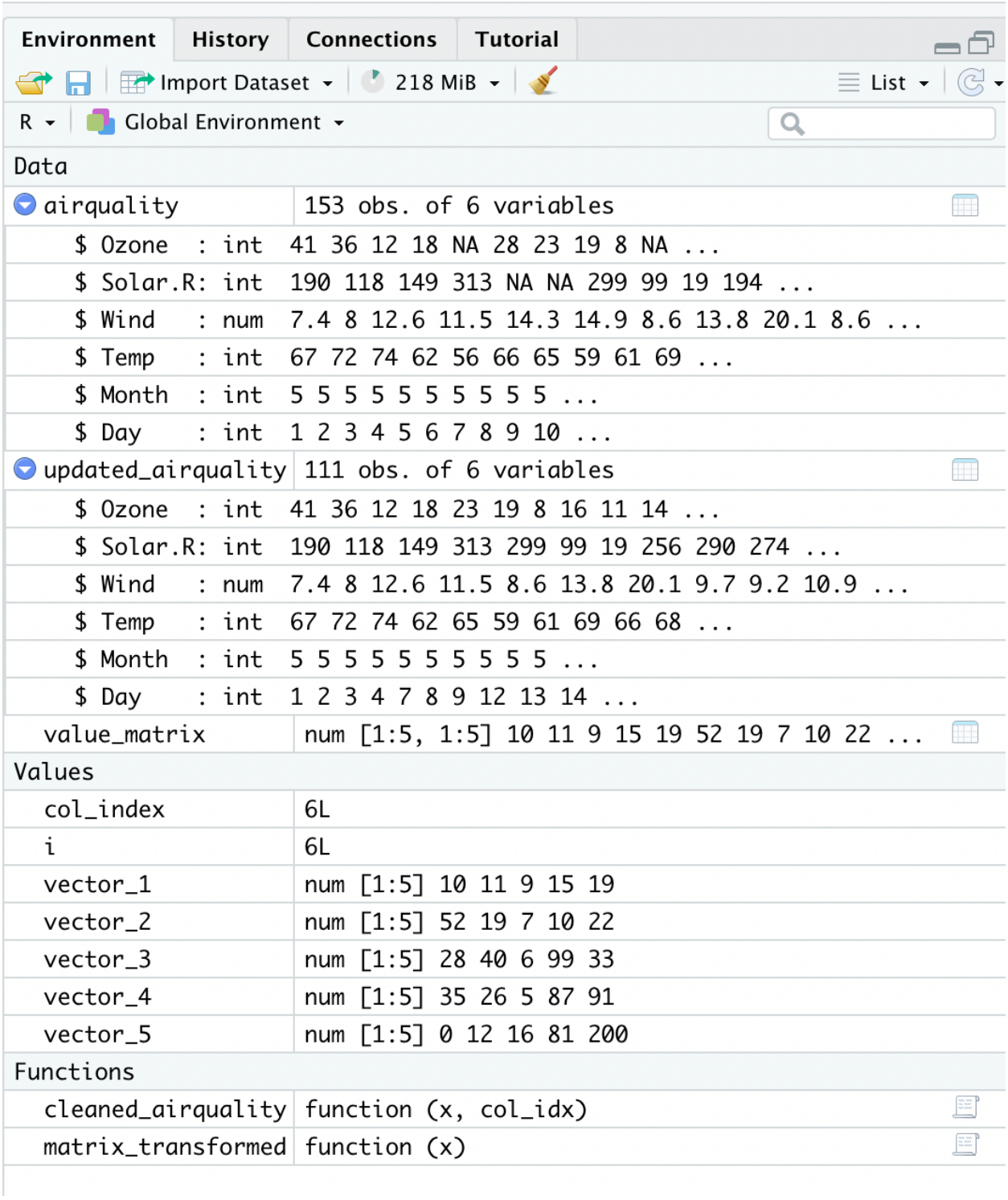
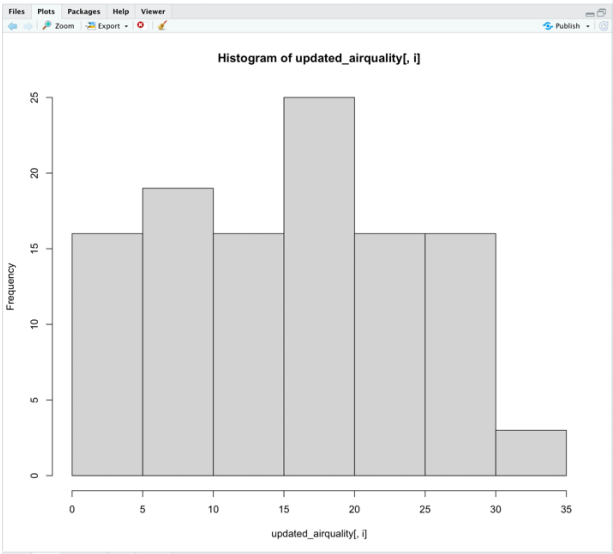
#Data Science - R Homework: Coding Questions #1

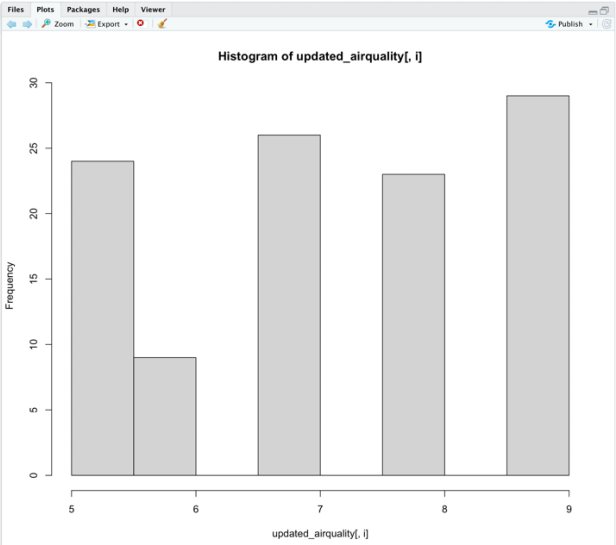
```
#Loading the required library and dataset to R
library(MASS)
airquality <- as.data.frame(airquality)

# Question 1 - Chapter 3.4 Exercise 2
# Create your own script header.
# Make sure you include all information about yourself
# so that it serves as a business card as well.
# Make sure it doesn't execute with the other R code using the "comment" option.

##########################################################################
# Created by - Mihir Sachdeva on 23.01.22
# Title - Homework: Coding questions #1
# MBAN - Cohort 1 (Morning batch)
# This script and analysis is made based on the requirements of questions from
# exercises 2, 3, 8 and 10 from Chapters 3.4 and 4.4.
# This write-up is in reference to the demands of question 2.
# This includes making the script header and net description for the assignment.
##########################################################################

# Question 2 - Chapter 3.4 Exercise 3
# Find the best library to build a neural network (type: perceptron).
# a) Install the Library
# b) Read the library documentation from CRAN
# c) Getting the R help documentation for the function that trains the model

#Installing key package - neuralnet
install.packages("neuralnet")

# Loading the library for neuralnet
library("neuralnet")

# Using help function - reading data for training of Neural Networks
?neuralnet

# Display the library as a function to find all theory that helps trains our model
library()

# Installing key package - caret (Classification and Regression Training)
install.packages("caret")

# Using help function - for information on caret
```

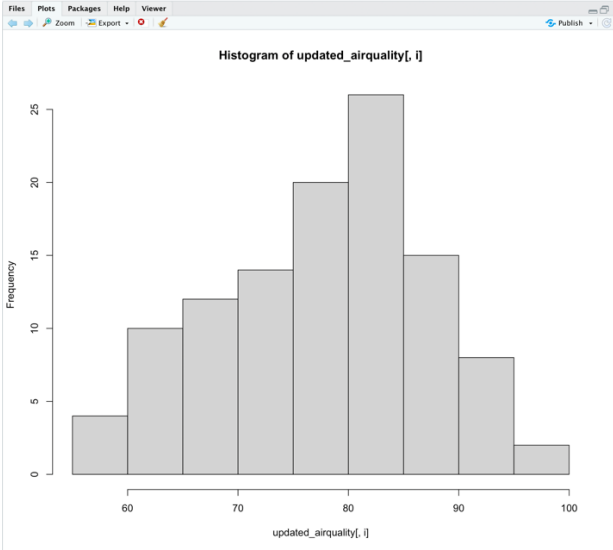
```
??caret

# Installing key package - to be able to fit neural networks
install.packages("nnet")
library(nnet)

# Using help function - on nnet library
?nnet

# Question 3: Create a user defined function named transformmatrix that
# takes the diagonal of a matrix and calculates a vector with two
# elements. Element one is the mean of the diagonal and element
# two is the median.
# a) Use the transformmatrix function to transform the matrix
# created in Exercise 7 from chapter 2.
# b) Use the transformmatrix function to transform the matrix that
# was given as an example in chapter 2.1

#Creating vectors from Exercise 7, Chapter 2 for the matrix
vector_1 <- c(10,11,9,15,19)
vector_2 <- c(52,19,7,10,22)
vector_3 <- c(28,40,6,99,33)
vector_4 <- c(35,26,5,87,91)
vector_5 <- c(0,12,16,81,200)

# Developing a value_matrix using vector variables from vector_1 upto vector_5
# for transformmatrix
value_matrix <- matrix(c(vector_1, vector_2, vector_3, vector_4, vector_5),
            nrow = 5,
            ncol = 5)

# Developing User Defined Function for the tranformation of value_matrix

matrix_transformed <- function(x){
  vector_output <- c(mean(diag(x)), median(diag(x)))
  return(vector_output)
} #Closing the transformation matrix - 'matrix_transformed'

# Assinging x as the new matrix to run the 'transformmatrix'

matrix_transformed(x = value_matrix)

#_Question 4: Chapter 4.4 Exercise 10_
# Create a user defined function that can clean up almost any data frame using
```

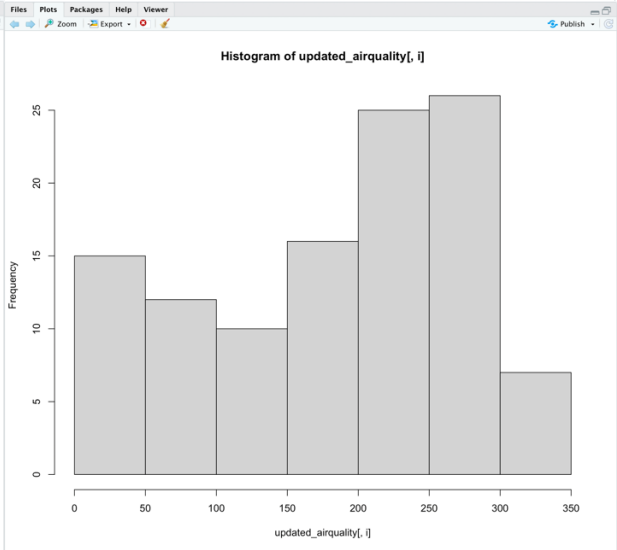
```
# a loop. The function inputs are the dataset name and the column indexes that
# we want to clean up. An example of the function call might look like this:
# function_name(x=mydata, col_idx=c(1,2,3))
# The loop inside the function will take column indexes (from the call inputs)
# and remove observations with empty values. Hint: inside the loop, use:
# new_df <- x[-which(is.na(x[ ,col_idx[i]]))]


# Developing col_index to store the number of variables in our data frame
col_index <- ncol(airquality)

#Creating and utilizing the - User defined function  (UDF)
cleaned_airquality <- function(x, col_idx){
  #staring  the for loop - for 'cleaned_airquality'
  for(i in 1:length(col_idx)){
    x <- x[-which(is.na(x[,col_idx[i]])), ] #removing all the null value cells

  } #closing the [i] - loop
  return(x)
} #closing the user defined function for 'cleaned_airquality'

#checking the UDF with our original airquality data frame from the provided set
updated_airquality <- cleaned_airquality(airquality, c(1,2))

#Running a for loop to check the descriptive statistic values for -
# min, mean, and max

for(i in 1:6){
  print(min(updated_airquality[, i], na.rm = TRUE))
  print(mean(updated_airquality[, i], na.rm = TRUE))
  print(max(updated_airquality[, i], na.rm = TRUE))
} # Closing the [i] - loop

# for loop to output the histograms for all current variables in the updated set
for (i in 1:ncol(updated_airquality)) {
  try(hist(updated_airquality[,i]))
} # Closing the [i] - loop
```

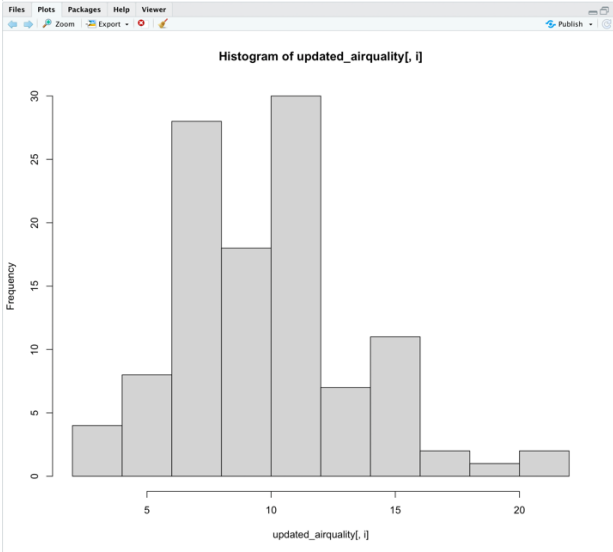

Environment
History
Connections
Tutorial
Import Dataset
218 MiB
List
R
Global Environment
Data
airquality
153 obs. of 6 variables
$ Ozone : int 41 36 12 18 NA 28 23 19 8 NA ...
$ Solar.R: int 190 118 149 313 NA NA 299 99 19 194 ...
$ Wind : num 7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...
$ Temp : int 67 72 74 62 56 66 65 59 61 69 ...
$ Month : int 5 5 5 5 5 5 5 5 5 5 ...
$ Day : int 1 2 3 4 5 6 7 8 9 10 ...
updated_airquality
111 obs. of 6 variables
$ Ozone : int 41 36 12 18 23 19 8 16 11 14 ...
$ Solar.R: int 190 118 149 313 299 99 19 256 290 274 ...
$ Wind : num 7.4 8 12.6 11.5 8.6 13.8 20.1 9.7 9.2 10.9 ...
$ Temp : int 67 72 74 62 65 59 61 69 66 68 ...
$ Month : int 5 5 5 5 5 5 5 5 5 5 ...
$ Day : int 1 2 3 4 7 8 9 12 13 14 ...
value_matrix
num [1:5, 1:5] 10 11 9 15 19 52 19 7 10 22 ...
Values
col_index
6L
i
6L
vector_1
num [1:5] 10 11 9 15 19
vector_2
num [1:5] 52 19 7 10 22
vector_3
num [1:5] 28 40 6 99 33
vector_4
num [1:5] 35 26 5 87 91
vector_5
num [1:5] 0 12 16 81 200
Functions
cleaned_airquality
function (x, col_idx)
matrix_transformed
function (x)

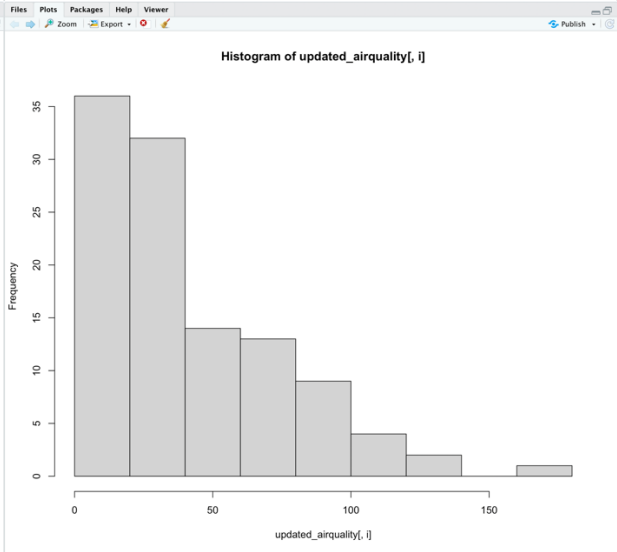

Files Plots Packages Help Viewer
Zoom Export Publish
Histogram of updated_airquality[, i]
Frequency
0 5 10 15 20 25
0 5 10 15 20 25 30 35
updated_airquality[, i]


Files Plots Packages Help Viewer
Zoom Export Publish
Histogram of updated_airquality[, i]
Frequency
0 5 10 15 20 25 30
5 6 7 8 9
updated_airquality[, i]


Files Plots Packages Help Viewer
Zoom Export Publish
Histogram of updated_airquality[, i]
Frequency
0 5 10 15 20 25
60 70 80 90 100
updated_airquality[, i]


Files Plots Packages Help Viewer
Zoom Export Publish
Histogram of updated_airquality[, i]
Frequency
0 5 10 15 20 25
0 50 100 150 200 250 300 350
updated_airquality[, i]


Files Plots Packages Help Viewer
Zoom Export Publish
Histogram of updated_airquality[, i]
Frequency
0 5 10 15 20 25 30
5 10 15 20
updated_airquality[, i]


Files Plots Packages Help Viewer
Zoom Export Publish
Histogram of updated_airquality[, i]
Frequency
0 5 10 15 20 25 30 35
0 50 100 150
updated_airquality[, i]

Conclusion –

Some key takeaways from case study and analysis:
It very well may be seen from the outcomes that the normal ozone layer, sunlight-based radiation, wind, temperature, month, day from the factors are showing the potential a dangerous atmospheric deviation that is conceivably influencing the air quality. This dataset and its examination can assist with tracking down the nature of air and anticipate the estimate for next couple of months. It very well may be seen from the hist plots that the dissemination of our factors are outstanding, slanted, and unbalanced. It comprehends the conduct of our factors.

Extra factors for reference of air quality impact:
Ozone - Ozone at ground level is undesirable and is a part of exhaust cloud.
Solar.R - Researchers have observed that the air contamination assimilates and scatters daylight and accordingly lessens the sum that arrives at the Earth's surface.
Wind - Air temperature influences the development of air, and in this manner the development of air contamination.
Temp - As the ground warms up during the day, the air by and large turns out to be more tempestuous, making air poisons scatter in the air.