**SAVEETHA SCHOOL OF ENGINEERING**

**ITA 0443 - STATISTICS WITH R PROGRAMMING FOR REAL TIME PROBLEM**

**DAY 2 – LAB ASSESSMENT**

**Reg No:192110623**

**Name: B.JASWANTH REDDY**

1**.Write a R program to create an array of two 3x3 matrices each with 3 rows and 3 columns from two given two vectors. Print the second row of the second matrix of the array and the element in the 3rd row and 3rd column of the 1st matrix**.   
**CODE:**

vec1 <- c(1,2,3,4,5,6,7,8,9)

vec2 <- c(10,20,30,40,50,60,70,80,90)

mat1 <- matrix(vec1, nrow=3, ncol=3)

mat2 <- matrix(vec2, nrow=3, ncol=3)

mat\_array <- array(c(mat1, mat2), dim=c(3,3,2))

print(mat\_array[2, , 2])

**OUTPUT**

[1] 40 50 60

**2.** **Write a R program to combine three arrays so that the first row of the first array is followed by the first row of the second array and then first row of the third array.**

**CODE:**

array1 <- matrix(1:6, ncol = 2)

array2 <- matrix(7:12, ncol = 2)

array3 <- matrix(13:18, ncol = 2)

result <- rbind(array1[1, ], array2[1, ], array3[1, ])

Print the result

print(result)

**OUTPUT:**

[,1] [,2]

[1,] 1 4

[2,] 7 10

[3,] 13 16

**3. Write a R program to create an array using four given columns, three given rows, and two given tables and display the content of the array.**

**CODE:**

table1 <- data.frame(col1 = c(1, 2, 3),

col2 = c(4, 5, 6))

table2 <- data.frame(col3 = c(7, 8, 9),

col4 = c(10, 11, 12))

result\_table <- cbind(table1, table2)

rows <- c(1, 2, 3)

result\_array <- result\_table[rows, c("col1", "col2", "col3", "col4")]

print(result\_array)

**OUTPUT:**

col1 col2 col3 col4

1 1 4 7 10

2 2 5 8 11

3 3 6 9 12

**4.** **Write a R program to create a two-dimensional 5x3 array of sequence of even integers greater than 50.**

**CODE:**

even\_integers <- seq(52, 62, by = 2)

result\_array <- matrix(even\_integers, nrow = 5, ncol = 3, byrow = TRUE)

print(result\_array)

**OUTPUT:** [,1] [,2] [,3]

[1,] 52 54 56

[2,] 58 60 62

[3,] 52 54 56

[4,] 58 60 62

[5,] 52 54 56

**Use Below Data frame from question 5 to 9**

exam\_data = data.frame(

name = c('Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'),

score = c(12.5, 9, 16.5, 12, 9, 20, 14.5, 13.5, 8, 19),

attempts = c(1, 3, 2, 3, 2, 3, 1, 1, 2, 1),

qualify = c('yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes')

)

5**. Write a R program to extract 3rd and 5th rows with 1st and 3rd columns from a given data frame**

**CODE:**

exam\_data <- data.frame(

name = c('Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'),

score = c(12.5, 9, 16.5, 12, 9, 20, 14.5, 13.5, 8, 19),

attempts = c(1, 3, 2, 3, 2, 3, 1, 1, 2, 1),

qualify = c('yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes')

)

result\_data <- exam\_data[c(3, 5), c(1, 3)]

print(result\_data)

**OUTPUT:**

name attempts

3 Katherine 2

5 Emily 2

6**. Write a R program to add a new column named country in a given data frame**

Country<-c("USA","USA","USA","USA","UK","USA","USA","India","USA","USA")

**CODE:**

exam\_data <- data.frame(

name = c('Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'),

score = c(12.5, 9, 16.5, 12, 9, 20, 14.5, 13.5, 8, 19),

attempts = c(1, 3, 2, 3, 2, 3, 1, 1, 2, 1),

qualify = c('yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes')

)

country <- c("USA", "USA", "USA", "USA", "UK", "USA", "USA", "India", "USA", "USA")

exam\_data$country <- country

print(exam\_data)

**OUTPUT:**

name score attempts qualify country

1 Anastasia 12.5 1 yes USA

2 Dima 9.0 3 no USA

3 Katherine 16.5 2 yes USA

4 James 12.0 3 no USA

5 Emily 9.0 2 no UK

6 Michael 20.0 3 yes USA

7 Matthew 14.5 1 yes USA

8 Laura 13.5 1 no India

9 Kevin 8.0 2 no USA

10 Jonas 19.0 1 yes USA

**7. Write a R program to add new row(s) to an existing data frame**

new\_exam\_data = data.frame(name = c('Robert', 'Sophia'),score = c(10.5, 9), attempts = c(1, 3),qualify = c('yes', 'no'))

**CODE:**

exam\_data = data.frame(

name = c('Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'),

score = c(12.5, 9, 16.5, 12, 9, 20, 14.5, 13.5, 8, 19),

attempts = c(1, 3, 2, 3, 2, 3, 1, 1, 2, 1),

qualify = c('yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', 'no', 'no', 'yes')

)

new\_exam\_data = data.frame(name = c('Robert', 'Sophia'),score = c(10.5, 9), attempts = c(1, 3),qualify = c('yes', 'no'))

exam\_data = rbind(exam\_data, new\_exam\_data)

**OUTPUT:**

> exam\_data

name score attempts qualify

1 Anastasia 12.5 1 yes

2 Dima 9.0 3 no

3 Katherine 16.5 2 yes

4 James 12.0 3 no

5 Emily 9.0 2 no

6 Michael 20.0 3 yes

7 Matthew 14.5 1 yes

8 Laura 13.5 1 no

9 Kevin 8.0 2 no

10 Jonas 19.0 1 yes

11 Robert 10.5 1 yes

12 Sophia 9.0 3 no

**8. Write a R program to sort a given data frame by name and score**

**CODE:**

exam\_data[order(exam\_data$name, exam\_data$score),]

**OUTPUT:**

name score attempts qualify

7 Laura 13.5 1 no

10 Jonas 19.0 1 yes

1 Anastasia 12.5 1 yes

4 James 12.0 3 no

6 Michael 20.0 3 yes

9 Kevin 8.0 2 no

8 Kevin 19.0 1 yes

3 Katherine 16.5 2 yes

5 Emily 9.0 2 no

2 Dima 9.0 3 no

**9.  Write a R program to save the information of a data frame in a file and display the information of the file.**

**CODE:**

write.csv(exam\_data, "exam\_data.csv")

read\_data <- read.csv("exam\_data.csv")

str(read\_data)

**OUTPUT:**

**10. Write a R program to call the (built-in) dataset airquality. Check whether it is a data frame or not? Order the entire data frame by the first and second column. remove the variables 'Solar.R' and 'Wind' and display the data frame.**

**CODE:**

data("airquality")

is.data.frame(airquality) # Returns "TRUE"

airquality <- airquality[order(airquality[,1], airquality[,2]),]

airquality <- airquality[,-c(3,6)]

head(airquality)

**OUTPUT;**

**Ozone Month Day**

**5 37 5 5**

**6 20 5 6**

**7 23 5 7**

**8 19 5 8**

**9 8 5 9**

**10 NA 5 10**

**11. Write a R program to create a factor corresponding to height of women data set , which inbuild in R, contains height and weights for a sample of women.**

**CODE:**

data("women")women$height\_factor <- cut(women$height, breaks = c(58, 63, 68, 73), labels = c("Short", "Medium", "Tall"), right = FALSE)

head(women)

**OUTPUT:**

height weight height\_factor

1 58 115 Short

2 59 117 Short

3 60 120 Short

4 61 123 Short

5 62 126 Short

6 63 129 Medium

**12. Write a R program to extract the five of the levels of factor created from a random sample from the LETTERS (Part of the base R distribution**.)

**CODE:**

L = sample(LETTERS,size=50,replace=TRUE)

print("Original data:")

print(L)

f = factor(L)

print("Original factors:")

print(f)

print("Only five of the levels")

print(table(L[1:5]))

**OUTPUT:**

[1] "Original data:"

[1] "H" "N" "O" "D" "L" "E" "H" "U" "W" "W" "S" "Q" "A" "O" "I" "G" "G" "W" "T"

[20] "Z" "I" "S" "B" "P" "I" "F" "L" "B" "X" "A" "J" "V" "X" "C" "U" "A" "C" "W"

[39] "D" "J" "X" "C" "U" "O" "F" "V" "Y" "Z" "W" "Z"

[1] "Original factors:"

[1] H N O D L E H U W W S Q A O I G G W T Z I S B P I F L B X A J V X C U A C W

[39] D J X C U O F V Y Z W Z

Levels: A B C D E F G H I J L N O P Q S T U V W X Y Z

[1] "Only five of the levels"

D H L N O

1 1 1 1 1

13. **Iris dataset is a very famous dataset in almost all data mining, machine learning courses, and it has been an R build-in dataset. The dataset consists of 50 samples from each of three species of Iris flowers (Iris setosa, Iris virginica and Iris versicolor). Four features(variables) were measured from each sample, they are the length and the width of sepal and petal, in centimetres. Perform the following EDA steps .**

**(i)Find dimension, Structure, Summary statistics, Standard Deviation of all features.**

**(ii)Find mean and standard deviation of features groped by three species of Iris flowers (Iris setosa, Iris virginica and Iris versicolor)**

**(iii)Find quantile value of sepal width and length**

**(iV)create new data frame named iris1 which have a new column name Sepal.Length.Cate that categorizes “Sepal.Length” by quantile**

**(V) Average value of numerical varialbes by two categorical variables: Species and Sepal.Length.Cate:**

**(vi) Average mean value of numerical varialbes by Species and Sepal.Length.Cate**

**(vii)Create Pivot Table based on Species and Sepal.Length.Cate.**

# Load the Iris dataset

data(iris)

# i) Dimension, Structure, Summary Statistics, Standard Deviation of all features

dim(iris) # Dimension of the data frame

str(iris) # Structure of the data frame

summary(iris) # Summary statistics of all features

apply(iris[,1:4], 2, sd) # Standard deviation of all features

# ii) Mean and Standard Deviation of features grouped by three species of Iris flowers (Iris setosa, Iris virginica and Iris versicolor)

by\_species <- aggregate(iris[,1:4], by=list(Species=iris$Species), FUN=mean)

by\_species\_sd <- aggregate(iris[,1:4], by=list(Species=iris$Species), FUN=sd)

# iii) Quantile value of sepal width and length

quantile(iris$Sepal.Width)

quantile(iris$Sepal.Length)

# iv) Create new data frame named iris1 which have a new column name Sepal.Length.Cate that categorizes “Sepal.Length” by quantile

iris1 <- iris

iris1$Sepal.Length.Cate <- cut(iris1$Sepal.Length, quantile(iris1$Sepal.Length))

# v) Average value of numerical variables by two categorical variables: Species and Sepal.Length.Cate

aggregate(iris[,1:4], by=list(Species=iris1$Species, Sepal.Length.Cate=iris1$Sepal.Length.Cate), FUN=mean)

# vi) Average mean value of numerical variables by Species and Sepal.Length.Cate

aggregate(iris[,1:4], by=list(Species=iris1$Species, Sepal.Length.Cate=iris1$Sepal.Length.Cate), mean)

# vii) Create Pivot Table based on Species and Sepal.Length.Cate

library(reshape2)

pivot\_table <- dcast(iris1, Species ~ Sepal.Length.Cate, value.var="Sepal.Length")

**OUTPUT:**

# i) Dimension, Structure, Summary Statistics, Standard Deviation of all features

> dim(iris)

[1] 150 5

> str(iris)

'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(iris)

Sepal.Length Sepal.Width Petal.Length Petal.Width

Min. :4.300 Min. :2.000 Min. :1.000 Min. :0.100

1st Qu.:5.100 1st Qu.:2.800 1st Qu.:1.600 1st Qu.:0.300

Median :5.800 Median :3.000 Median :4.350 Median :1.300

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

> apply(iris[,1:4], 2, sd)

Sepal.Length Sepal.Width Petal.Length Petal.Width

0.8280661 0.4358663 1.7652982 0.7622377

# ii) Mean and Standard Deviation of features grouped by three species of Iris flowers (Iris setosa, Iris virginica and Iris versicolor)

> by\_species

Species Sepal.Length Sepal.Width Petal.Length Petal.Width

1 setosa 5.006 3.428 1.462 0.246

2 versicolor 5.936 2.770 4.260 1.326

3 virginica 6.588 2.974 5.552 2.026

> by\_species\_sd

Species Sepal.Length Sepal.Width Petal.Length Petal.Width

1 setosa 0.3524896 0.3790644 0.1736644 0.1053856

2 versicolor 0.5161711 0.3137983 0.4699110 0.1977527

3 virginica 0.6358796 0.3224966 0.5518947 0.2746501

**14.Titanic Casualties – Use the standard ‘Titanic’ dataset which is part of R Base to answer  
the following questions.  
(i). Use an appropriate apply function to get the sum of males vs females aboard.**

CODE:

library(datasets)

data("Titanic")

apply(Titanic, 2, sum)["Sex"]

**Output:**

Male Female

843466  
**(ii). Get a table with the sum of survivors vs sex.**

CODE:

table(Titanic$Survived, Titanic$Sex)

**OUTPUT:**

Female Male

No 81 468

Yes 233 109

**(iii). Get a table with the sum of passengers by sex vs age**

CODE:

with(Titanic, table(Sex, Age))

**OUTPUT:**

Age

Sex 0-5 years 6-18 years adult senior

Female 6 33 233 94

Male 62 30 550 201