19/20

8

DAY 3 – LAB ASSESSMENT

Reg No:
Name:

1. (i) Write a function in R programming to print generate Fibonacci sequence using Recursion in R

code:

```
fibonacci<-function(n)
{
 if (n<=0)
  stop("Invalid input.n should be a positive integer.")
 }
 if(n==1)
```

```
{
return (0)
}
else if(n==2)
{
return(1)
}
else
{
return(fibonacci(n-1)+fibonacci(n-2))
}
}
n<-10
for (i in 1:n)
{
fib_num<-fibonacci(i)
print(fib_num)
}
```

Output:

```
0
1
1
2
3
5
8
13
21
34.
```

(ii) Find sum of natural numbers up-to 10, without formula using loop statement.

Code:

```
sum<-0
for (i in 1:10)
{
sum<-sum+i
}
print (sum)
```

output:

```
55
```

(iii) create a vector 1:10 and Find a square of each number and store that in a separate list.

Code:

```
my_vector<-1:10
square_list<-list()
for(num in my_vector)
{
square<-num^2
square_list<-c(square_list, square)
}
print (square_list)
```

Output:

```
4
9
16
25
36
49
64
81
100
```

4. Performing Logistic regression on dataset to predict the cars Engine shape(vs) .
(i)Do the EDA analysis and find the features which is impact the Engine shape and use this for model.
(ii) Split the data set randomly with 80:20 ration to create train and test dataset and create logistic model
(iii)Create the Confusion matrix among prediction and test data.

Code:

```
data<-read.csv("cars.csv")
sum(is.na(data))
boxplot(data$mpg)
ggplot(data,aes(x=mpg,y=vs))+ geom_point()+ggtitle("Relationship between mpg and vs")
cor(data[,-9])
set.seed(123)
trainIndex<-createDataPartition(data$vs, p=0.8,list=FALSE)
train<-data[trainIndex,]
test<-data[-trainIndex,]
log_model<-glm(vs~mpg+wt+hp,data=train,family=binomial)
summary(log_model)
predictions<-predict(log_model,newdata=test,type="response")
predicted_classes<-ifelese(prediction>0.5,1,0)
confusionMatrix(predicted_classes, tes$vs)
```

O/P:

Actual	Predicted	
	0	1
0	25	5
1	6	14.

5. (i) Write R Program to create 15 x15 matrix filled with random numbers between -10 to 10, numbers can repeat. set random seed value to 328
(ii)Write R Program to display Lower Diagonal and upper Diagonal matrix
(iii)Write R Program to count 0's in the matrix and check the matrix is sparse matrix or not
(iv) Write R code to remove outliers. Here the outliers are negative numbers. replace the negative values with positive values
(v) Find the mean median and mode of the values corresponding to column
(vi)Find the mean median and mode of the values corresponding to row

code:

```
set.seed (328)
mat <- matrix (sample (-10:10, 15 * 15 , replace = TRUE), nrow = 15)
lower_tri <- lower.tri (mat)
upper_tri <- upper.tri (mat)
lower_mat <- mat * lower_tri
upper_mat <- mat * upper_tri
cat ("Lower Diagonal Matrix: \n")
print (lower_mat)
cat ("Upper Diagonal Matrix: \n")
print (upper_mat)
n_zeros <- sum(mat == 0)
n_elements <- length (mat)
density <- n_zeros / n_elements
if (density < 0.5)
{
cat ("The matrix is not a sparse
                     matrix\n")
}
else
{
cat ("The matrix is a spare matrix\n")
}
cat ("No. of zeros in the matrix:", n_zeros,
                     "\n")

mat [mat < 0] <- abs (mat [mat < 0])
col_means <- apply (mat, 2, mean)
col_median <- apply (mat, 2, median)
col_modes <- apply (mat, 2, mfv)
cat ("Column Means: \n")
print (col_means)
cat ("Column Modes: \n")
print (col_modes)

row_means <- apply (mat, 1, mean)
row_medians <- apply (mat, 1, median)
row_modes <- apply (mat, 1, mfv)
cat ("Row Means: \n")
print (row_means)
cat ("Row Medians: \n")
print (row_medians)
cat ("Row Modes: \n")
print (row_modes)
```

O/P:

Output

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]	[,12]	[,13]	[,14]	[,15]
[1,]	-6	-6	-1	1	-6	6	3	-5	2	8	-9	7	-8	-2	6
[2,]	0	9	3	4	4	0	-1	-2	5	-1	-8	0	9	7	4
[3,]	8	2	4	2	-2	5	-1	9	4	-7	4	0	4	7	-3
[4,]	-1	6	6	8	3	9	7	2	3	4	6	8	9	7	2
[5,]	6	-3	0	6	-6	5	-6	4	3	5	-6	7	8	-9	0
[6,]	7	-4	3	7	-7	6	-5	-4	5	1	-6	-4	-3	7	6
[7,]	-7	0	9	8	-8	4	4	3	9	7	5	7	6	5	0
[8,]	-1	-3	7	9	-9	3	3	4	-7	6	4	0	4	3	2
[9,]	-2	7	4	7	7	-2	7	-6	8	3	3	-1	0	-7	3
[10,]	-4	3	5	4	4	7	9	-7	6	9	-9	-5	0	9	6
[11,]	0	4	-3	3	3	-1	1	9	-5	4	7	4	3	1	2
[12,]	5	5	4	-2	2	-2	-2	8	4	5	5	0	3	4	0
[13,]	6	3	5	-7	0	9	6	2	-3	7	4	5	6	7	8
[14,]	9	7	3	-1	9	-6	9	3	2	8	3	4	3	-2	-1
[15,]															

DAY 4 – LAB ASSESSMENT

Reg No:

Name:

1.Randomly Sample the iris dataset such as 80% data for training and 20% for test and create Logistics regression with train data, use species as target and petals width and length as feature variables , Predict the probability of the model using test data, Create Confusion matrix for above test model

code:

```
data(iris)
set.seed(123)
train_index <-sample(1:nrow (iris),0.8*nrow(iris),replace=FALSE).
```