Submitted by,

Tushar Chandrashekhar Jakhalekar

##Internet Project

##installing the packages

library(readxl) ## To import the .xlsx file

library(dplyr) ## For data manipulation

library(caret) ## For checking accuracy of the model

df=read_excel(file.choose()) ## Selecting the file from data

head(df) ## Displaying first 6 rows of the data

str(df) ## Checking the structure of the data

dim(df) ## Checking dimensions of the data

glimpse(df) ## Again checking structure of the data

##Problem no.1

summary(df) ## Analysing the variables of the data thorough Summeriszation

sum(is.na(df)) ## Checking for Null or NA values

library(ggplot2) ## For data Visualisation

##Problem no.2

ggplot(data=df,aes(x=Uniquepageviews,y=Visits),)+ ## Visualising the relation between Uniquepagereviews and Visits

```
 geom_point()+
 geom_smooth(method=lm,se=FALSE)

plot(x=df$Uniquepageviews,y=df$Visits,xlab="Uniquepageviews",ylab="Visits",main="Uniquepageviews vs Visits plot")
abline(lm(df$Uniquepageviews~df$Visits))

## Building Linear regression Model

model1=lm(Uniquepageviews~Visits,data=df)
print(model1)

model1$coefficients ## Checking model Coefficients
print(summary(model1))

##Here p-value is less than 0.05 so we reject the null hypothesis

##So to conclude that Visits play some important part in determining the value of unique page reviews
##but Unique page reviews value does not totally depends on Visits to the website
df[10,]

pred=model1$coefficients[1]+model1$coefficients[2]*1
pred

all_pred=predict(model1,select(df,Visits)) ## Predicting all the remaining values
all_pred

RMSE(df$Uniquepageviews,all_pred) ## Checking the accuracy of the Model1

##Problem no.3
```

```
head(df) ## Displaying first 6 rows of the data

summary(df)

colnames(df) ## Checking for column names from the data

newdf=df%>%select(-c(Continent,Sourcegroup)) ## Removing the columns which have non-numeric values

head(newdf)

## Visualizing the data and checking the relation between variables

plot(x=newdf$Exits,y=newdf$Timeinpage,xlab="Exits",ylab="Timeinpage",main="Exit vs Timeinpage")

plot(x=newdf$Exits,y=newdf$Uniquepageviews,xlab="Exits",ylab="Uniquepagereview",main="Exit vs Uniquepagereviews")

abline(lm(newdf$Exits~newdf$Uniquepageviews))

newdf=as.data.frame(newdf) ## Taking newdf as Dataframe

library(caTools) ## Loading library for smaple.split function

## For cross validation splitting the data into 25:75 ratio

sampledata=sample.split(newdf,.75)

trainset=newdf[sampledata,] ## train data set

testset=newdf[-sampledata,] ## test data set

head(trainset)

head(testset)
```

```
model2=glm(Exits~.,data=trainset,family="gaussian") ## Building the Logistic regression model

print(summary(model2)) ##Printing the summary of model2


all_pred2=predict(model2,testset) ## Predicting the remaining values

all_pred2


RMSE(testset$Exits,all_pred2) ## Checking the accuracy of the Model2


##Problem no.4


head(newdf)


## Visualising the relationship between the variables


plot(x=newdf$Timeinpage,y=newdf$Uniquepageviews,xlab="Timeonpage",ylab="Uniquepageviews"
,main="Timeonpage vs Uniquepageview")

abline(glm(newdf$Timeinpage~newdf$Uniquepageviews))


model3=glm(Timeinpage~.,data=trainset,family="gaussian") ## Training the Logistic regression
model

print(summary(model3))


all_pred3=predict(model3,testset) ## Predicting the remaining values

all_pred3


RMSE(testset$Timeinpage,all_pred3) ## Checking the accuracy of the Model3


##Problem no.5


head(newdf)


## Visualising the relationship between the variables
```

```
plot(x=newdf$Bounces,y=newdf$Uniquepageviews,xlab="Bounces",ylab="Uniquepageviews",main=
"Bounces vs Uniquepageviews")

abline(lm(newdf$Bounces~newdf$Uniquepageviews))


model4=glm(Bounces~.,data=trainset,family="gaussian") ## training Logistic Regression Model

print(summary(model4))


all_pred4=predict(model4,testset) ## Predicting the remaining values

all_pred4


RMSE(testset$Bounces,all_pred4) ## Checking the accuracy of the Model4
```