Week13-IP

Sharon Maswai

3/6/2020

Definition of Question

Context

An entrepreneur from Kenya has created an online cause which she would like to advertise on her blog. Based on the data she collected from running the ads on her blog, ehe would like a data scientist to help her determine which individuals click on the ads.

Objective

Build a random forest and decison tree model to classify individuals that click and those that do not click on ads

Metric for success

Random forest and decision tree model with accuracies of over 90%

Data Appropriateness

This dataset is suitable for the analysis.

```r
#Importing libraries and loading them
install.packages("tidyverse")
```

```
## Installing package into '/home/sharon-maswai/R/x86_64-pc-linux-gnu-library/3.6'
## (as 'lib' is unspecified)
```

```r
install.packages("ggplot2")
```

```
## Installing package into '/home/sharon-maswai/R/x86_64-pc-linux-gnu-library/3.6'
## (as 'lib' is unspecified)
```

```r
library(tidyverse)
```

```
## -- Attaching packages ------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.0     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.4
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ---------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
```

Load Dataset

```
advertising <- read.csv("~/Downloads/advertising.csv")
head(advertising)
```

```
##   Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage
## 1                    68.95  35    61833.90               256.09
## 2                    80.23  31    68441.85               193.77
## 3                    69.47  26    59785.94               236.50
## 4                    74.15  29    54806.18               245.89
## 5                    68.37  35    73889.99               225.58
## 6                    59.99  23    59761.56               226.74
##                            Ad.Topic.Line           City Male    Country
## 1    Cloned 5thgeneration orchestration    Wrightburgh    0    Tunisia
## 2    Monitored national standardization      West Jodi    1      Nauru
## 3      Organic bottom-line service-desk       Davidton    0 San Marino
## 4 Triple-buffered reciprocal time-frame West Terrifurt    1      Italy
## 5         Robust logistical utilization   South Manuel    0    Iceland
## 6       Sharable client-driven software      Jamieberg    1     Norway
##             Timestamp Clicked.on.Ad
## 1 2016-03-27 00:53:11             0
## 2 2016-04-04 01:39:02             0
## 3 2016-03-13 20:35:42             0
## 4 2016-01-10 02:31:19             0
## 5 2016-06-03 03:36:18             0
## 6 2016-05-19 14:30:17             0
```

Data Understanding

Dataset summary

```
summary(advertising)
```

```
##  Daily.Time.Spent.on.Site      Age          Area.Income    Daily.Internet.Usage
##  Min.   :32.60            Min.   :19.00   Min.   :13996   Min.   :104.8
##  1st Qu.:51.36            1st Qu.:29.00   1st Qu.:47032   1st Qu.:138.8
##  Median :68.22            Median :35.00   Median :57012   Median :183.1
##  Mean   :65.00            Mean   :36.01   Mean   :55000   Mean   :180.0
```

```
##  3rd Qu.:78.55            3rd Qu.:42.00   3rd Qu.:65471   3rd Qu.:218.8
##  Max.   :91.43            Max.   :61.00   Max.   :79485   Max.   :270.0
##
##                                 Ad.Topic.Line              City
##  Adaptive 24hour Graphic Interface      :  1   Lisamouth      :  3
##  Adaptive asynchronous attitude         :  1   Williamsport   :  3
##  Adaptive context-sensitive application :  1   Benjaminchester:  2
##  Adaptive contextually-based methodology:  1   East John      :  2
##  Adaptive demand-driven knowledgebase   :  1   East Timothy   :  2
##  Adaptive uniform capability            :  1   Johnstad       :  2
##  (Other)                                :994   (Other)        :986
##       Male                 Country                   Timestamp   Clicked.on.Ad
##  Min.   :0.000   Czech Republic:  9   2016-01-01 02:52:10:  1   Min.   :0.0
##  1st Qu.:0.000   France        :  9   2016-01-01 03:35:35:  1   1st Qu.:0.0
##  Median :0.000   Afghanistan   :  8   2016-01-01 05:31:22:  1   Median :0.5
##  Mean   :0.481   Australia     :  8   2016-01-01 08:27:06:  1   Mean   :0.5
##  3rd Qu.:1.000   Cyprus        :  8   2016-01-01 15:14:24:  1   3rd Qu.:1.0
##  Max.   :1.000   Greece        :  8   2016-01-01 20:17:49:  1   Max.   :1.0
##                  (Other)       :950   (Other)            :994
```

Checking column names

```r
names(advertising)
```

```
##  [1] "Daily.Time.Spent.on.Site" "Age"
##  [3] "Area.Income"              "Daily.Internet.Usage"
##  [5] "Ad.Topic.Line"            "City"
##  [7] "Male"                     "Country"
##  [9] "Timestamp"                "Clicked.on.Ad"
```

Checking datatypes

```r
#obtaining the datatypes
sapply(data, class)
```

```
##       ...       list    package    lib.loc    verbose      envir overwrite
##    "name"     "call"     "NULL"     "NULL"     "call"     "name" "logical"       "{"
```

Checking for null values and duplicates

```r
colSums(is.na(advertising))
```

```
## Daily.Time.Spent.on.Site                      Age              Area.Income
##                        0                        0                        0
##     Daily.Internet.Usage            Ad.Topic.Line                     City
##                        0                        0                        0
##                     Male                  Country                Timestamp
##                        0                        0                        0
##            Clicked.on.Ad
##                        0
```

```r
is.double(advertising)
```

```
## [1] FALSE
```

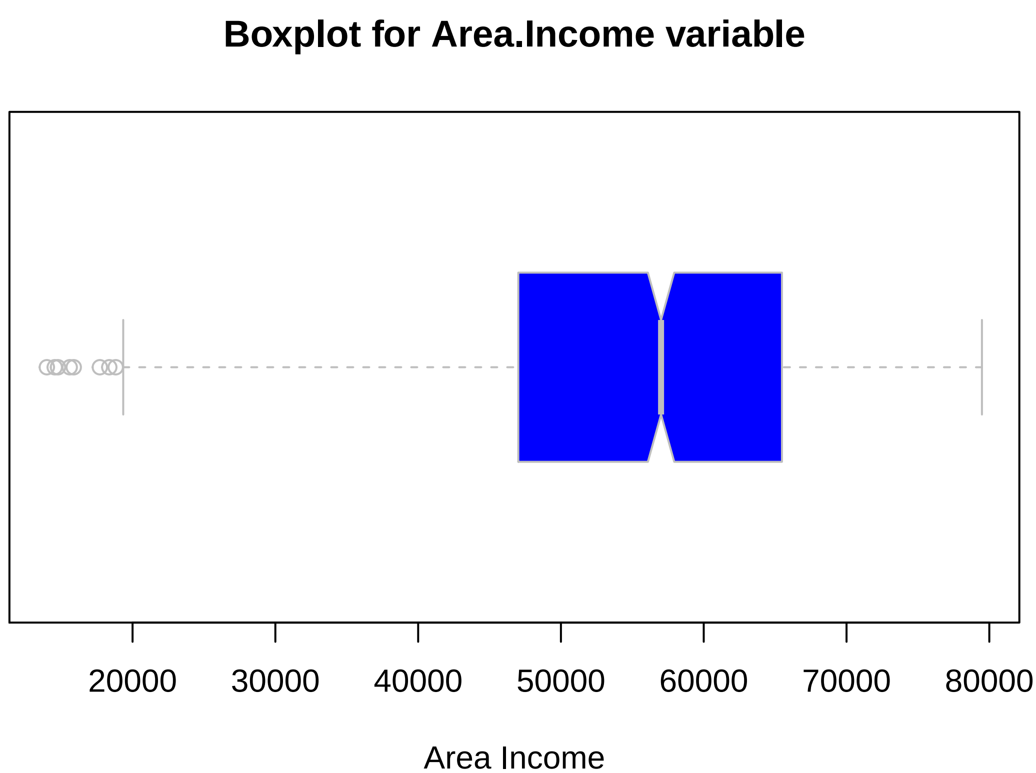
Conclusion

- There are no null values and duplicates in the dataset.

Univariate Analysis

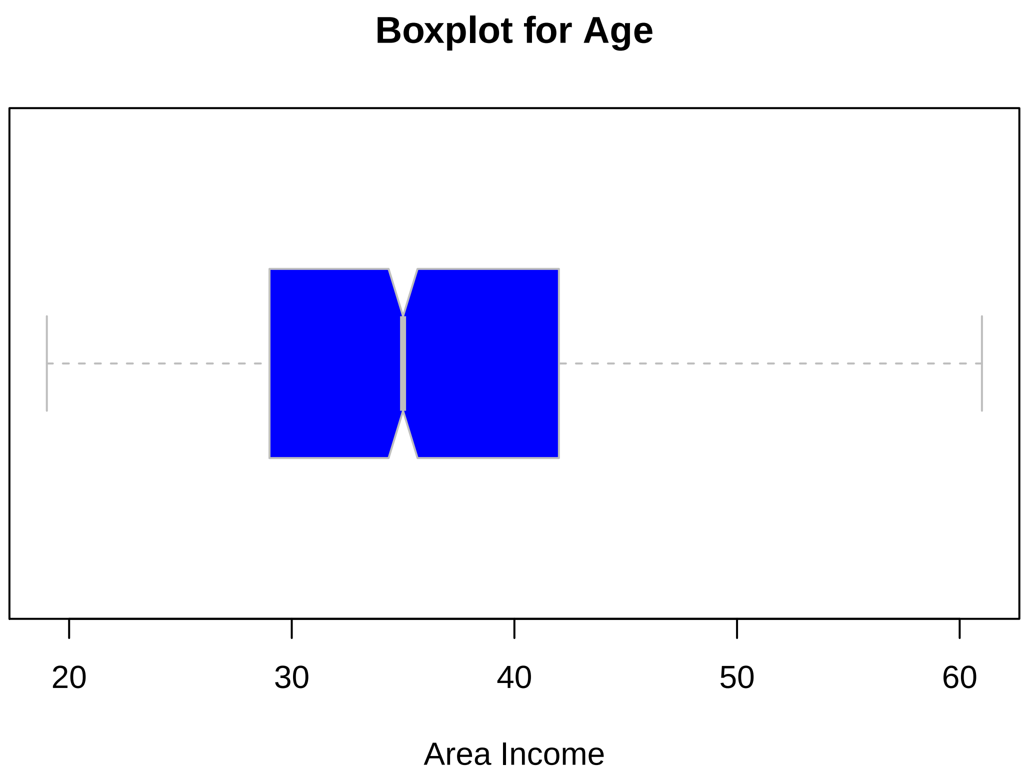
Checking for outliers

```r
bxplt_Area.Income = boxplot(advertising$Area.Income,
                    main = "Boxplot for Area.Income variable",
                    xlab = "Area Income",
                    col = "blue",
                    border = "grey",
                    horizontal = TRUE,
                    notch = TRUE)
```

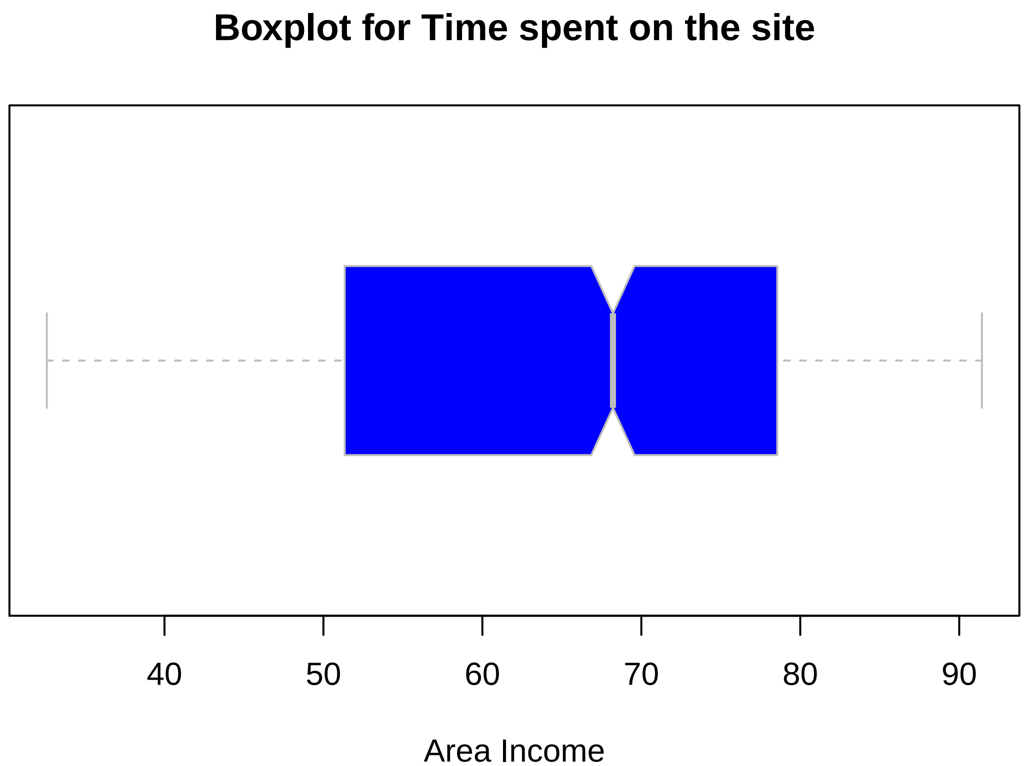



```r
bxplt_Area.Income = boxplot(advertising$Age,
                    main = "Boxplot for Age",
                    xlab = "Area Income",
```

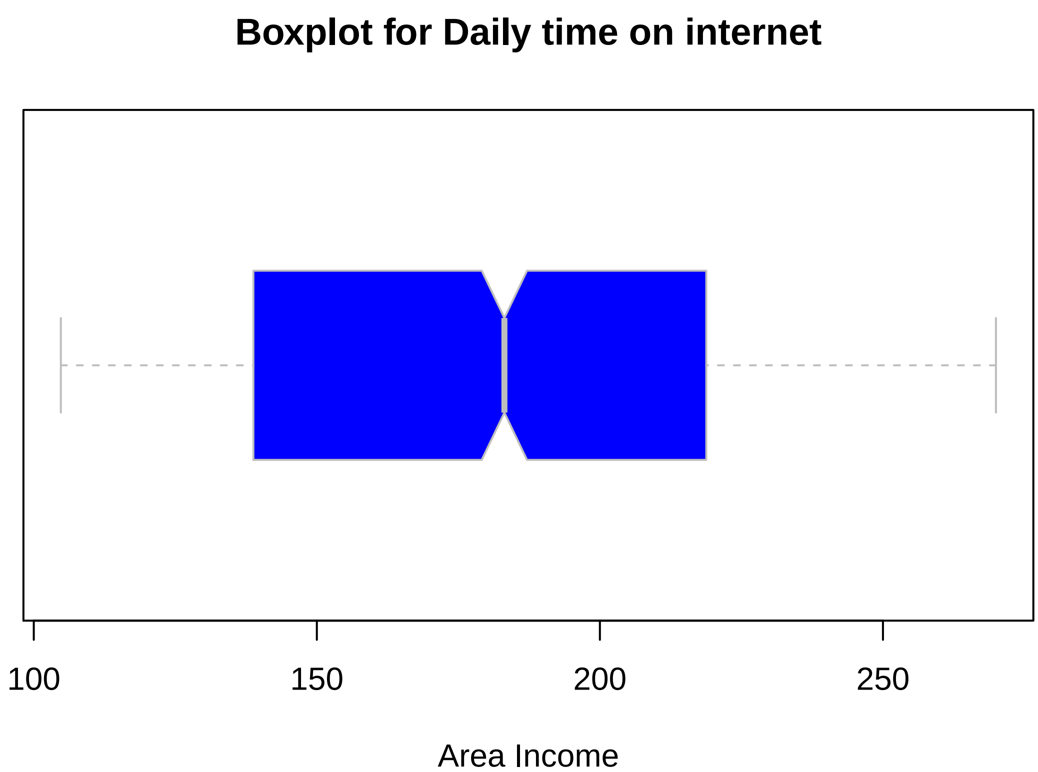
```
                         col = "blue",
                         border = "grey",
                         horizontal = TRUE,
                         notch = TRUE)
```




```
bxplt_Area.Income = boxplot(advertising$Daily.Time.Spent.on.Site,
                         main = "Boxplot for Time spent on the site",
                         xlab = "Area Income",
                         col = "blue",
                         border = "grey",
                         horizontal = TRUE,
                         notch = TRUE)
```



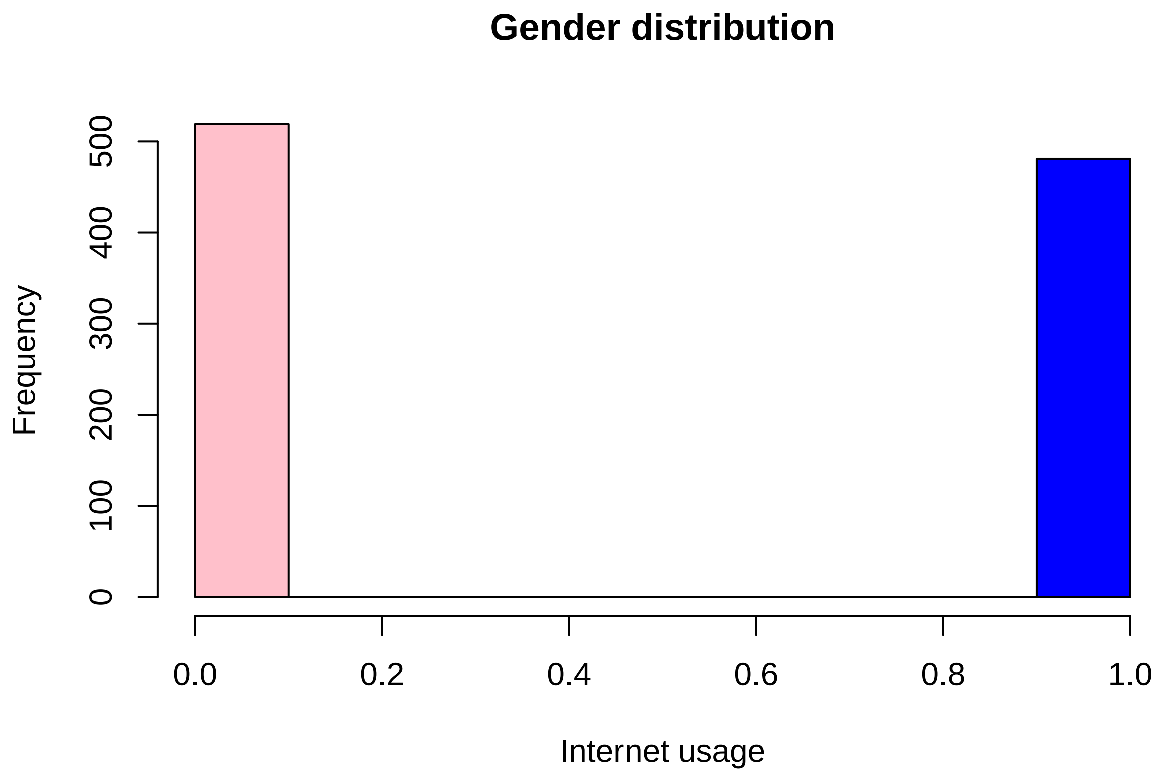

```r
bxplt_Area.Income = boxplot(advertising$Daily.Internet.Usage,
                    main = "Boxplot for Daily time on internet",
                    xlab = "Area Income",
                    col = "blue",
                    border = "grey",
                    horizontal = TRUE,
                    notch = TRUE)
```



Barcharts

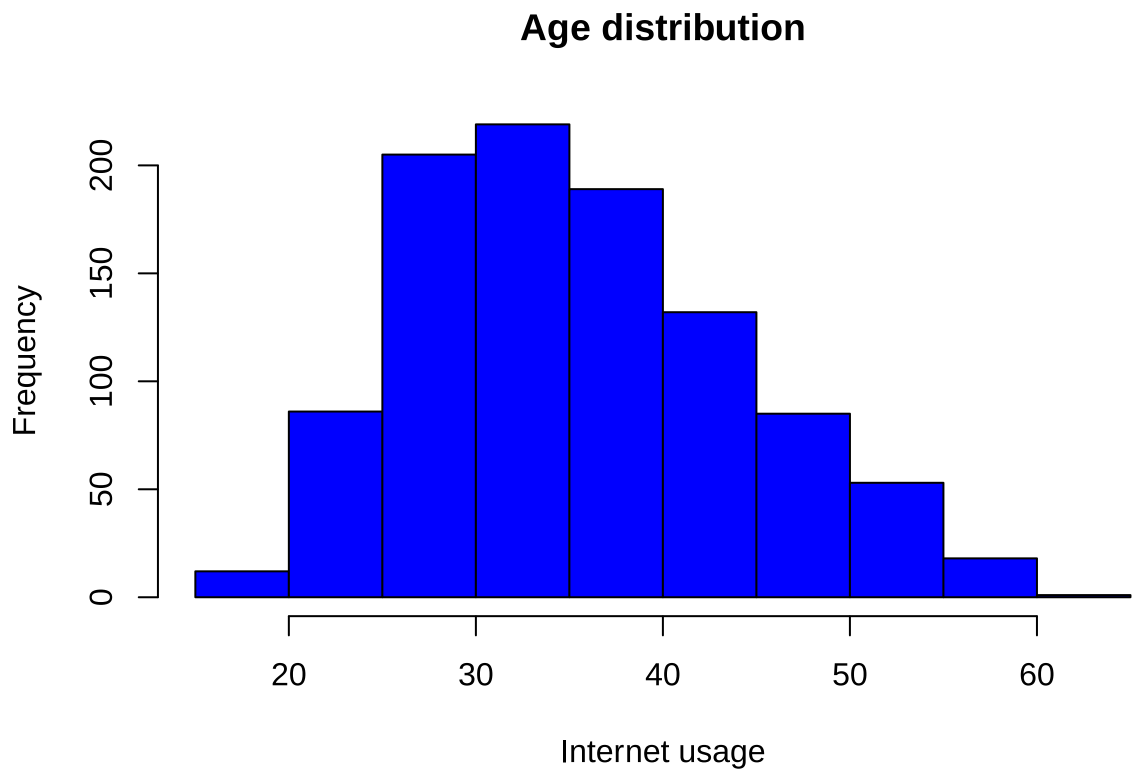
Gender

```
w = hist(advertising$Male,
         main = "Gender distribution",
         xlab = "Internet usage",
         col = c("Pink", "blue")
)
```



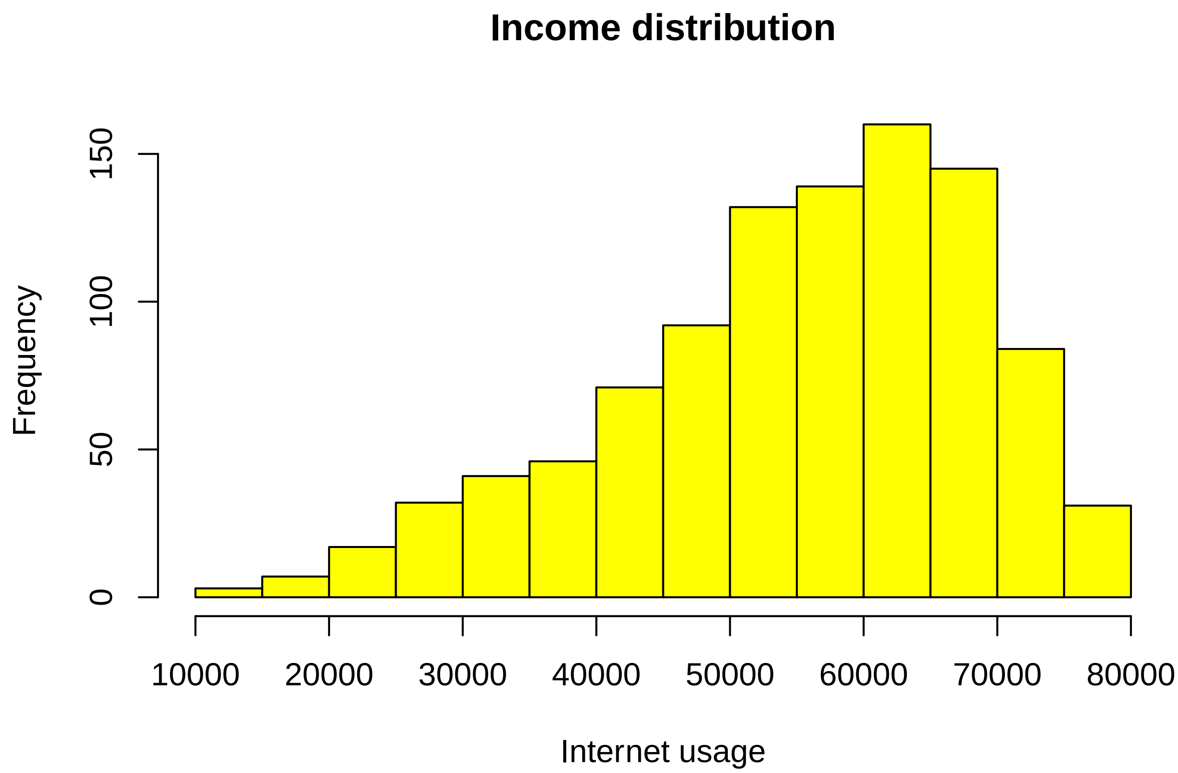
Age distribution

```
w = hist(advertising$Age,
         main = "Age distribution",
         xlab = "Internet usage",
         col = "Blue",
)
```



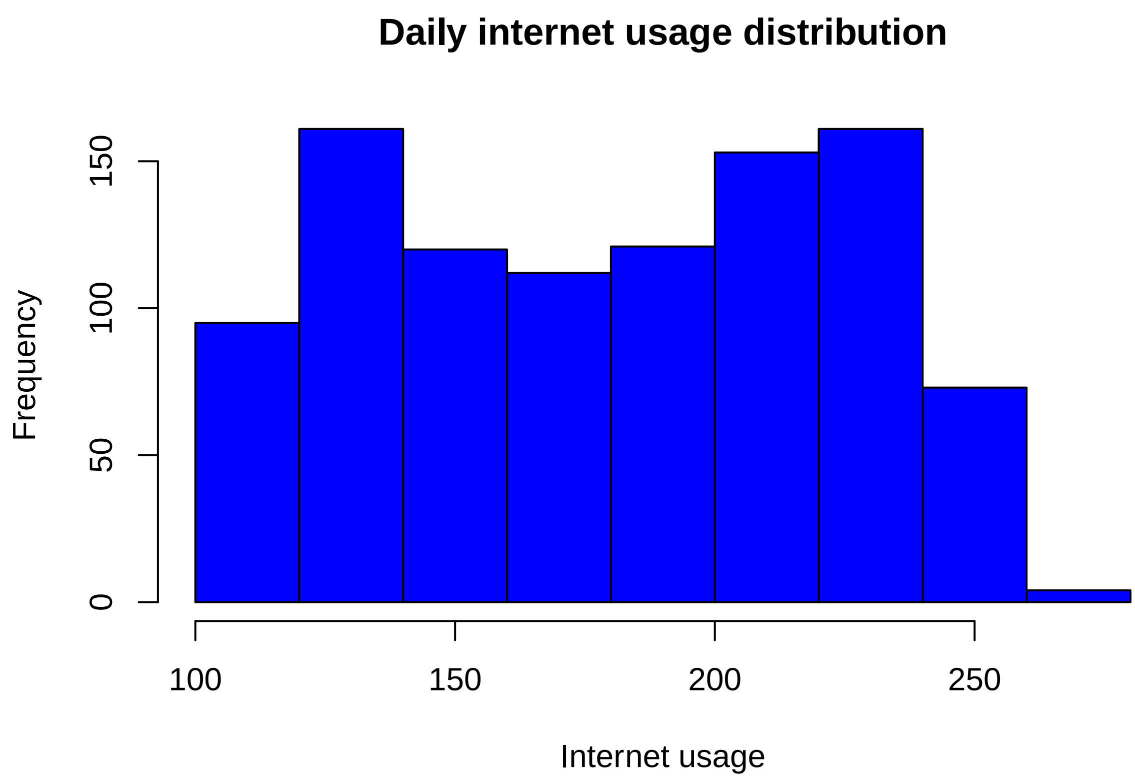
Income Distritution

```
w = hist(advertising$Area.Income,
         main = "Income distribution",
         xlab = "Internet usage",
         col = "Yellow",
)
```



Internet usage

```
w = hist(advertising$Daily.Internet.Usage,
         main = "Daily internet usage distribution",
         xlab = "Internet usage",
         col = "Blue",
)
```



Bivariate Analysis

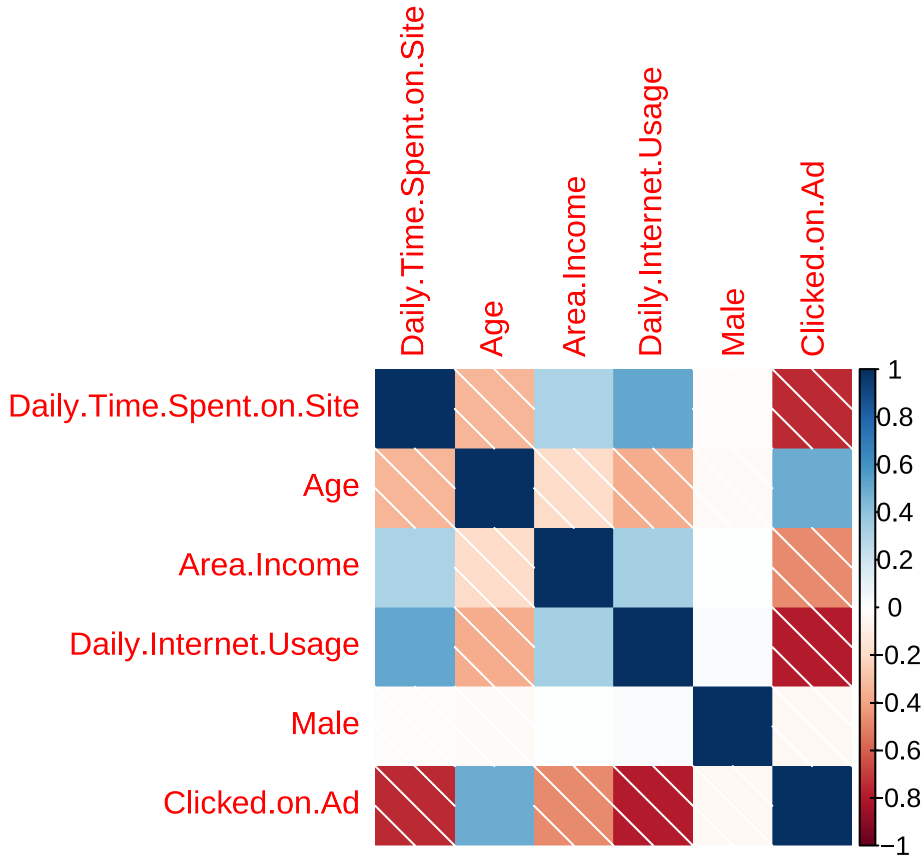
Correlation between variables

```r
#Accesing corrplot library
library(corrplot)
```

```
## corrplot 0.84 loaded
```

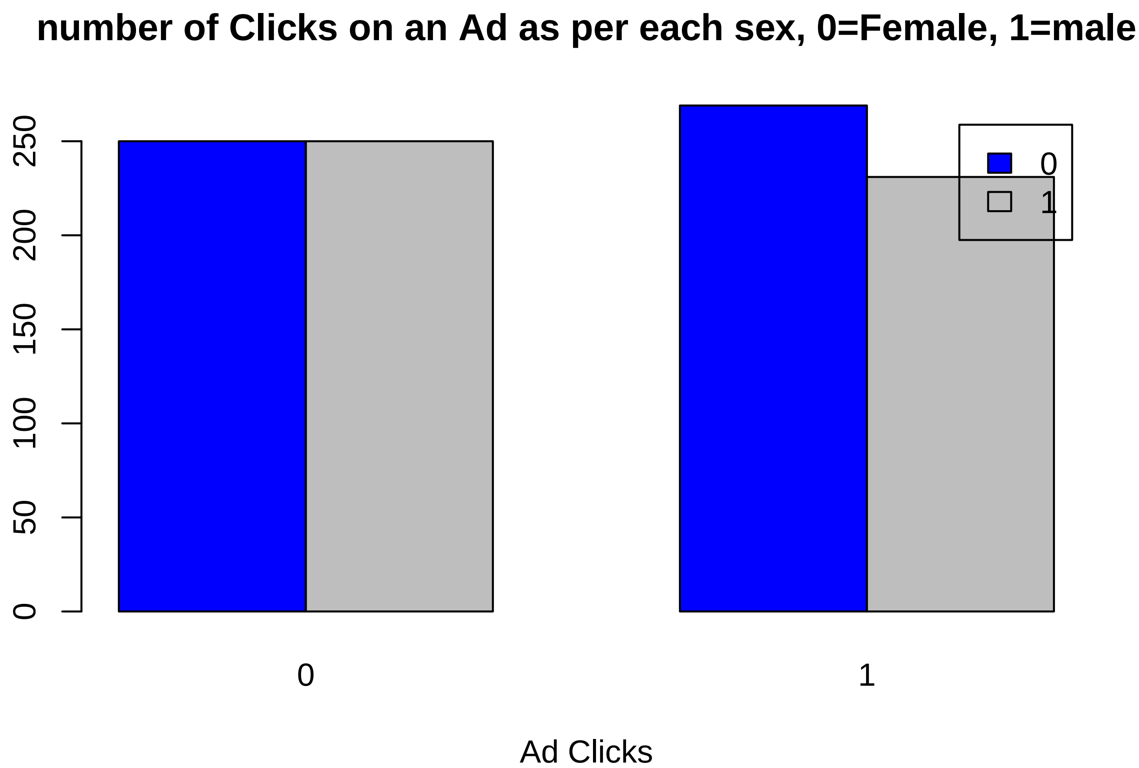
```r
#`getting the numeric values of our dataaset
data = advertising[, sapply(advertising, is.numeric)]

#plotting the numeric values.
corrplot(cor(data), method = 'shade')
```



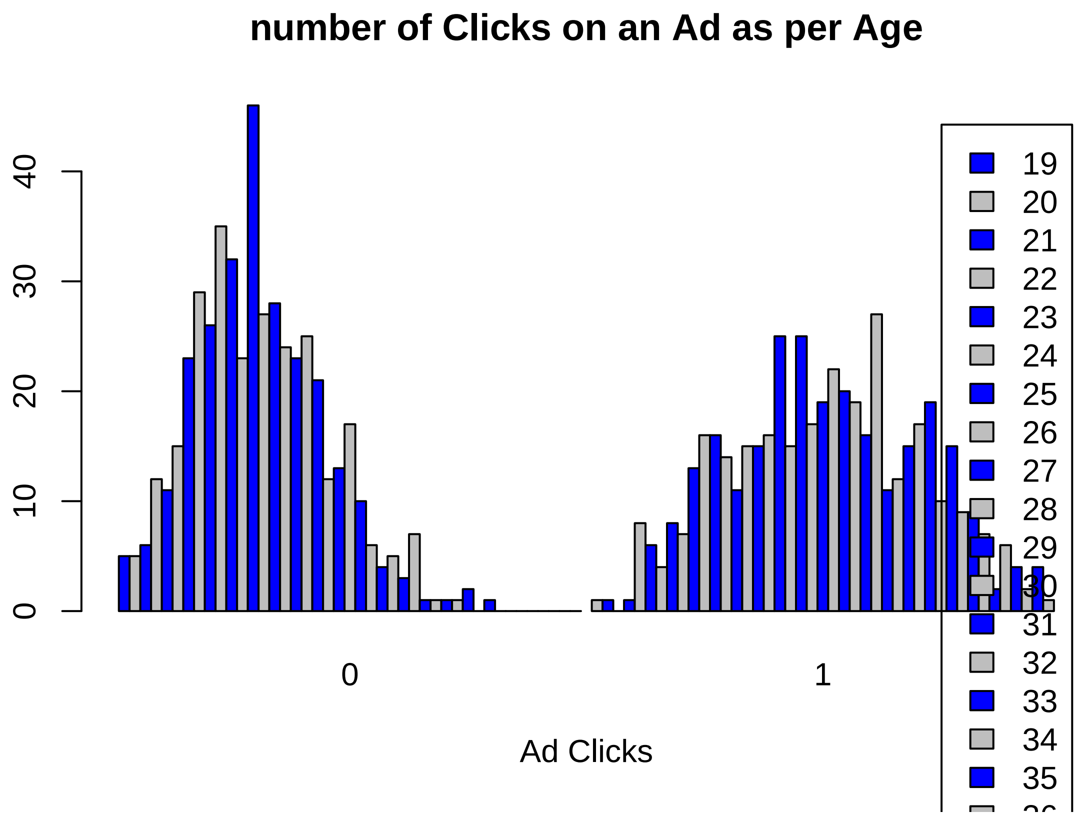
Sex versus Clicks

```
counts = table(advertising$Male, advertising$Clicked.on.Ad)
barplot(counts, main="number of Clicks on an Ad as per each sex, 0=Female, 1=male",
xlab="Ad Clicks", col=c("blue","grey"),
legend = rownames(counts), beside=TRUE)
```



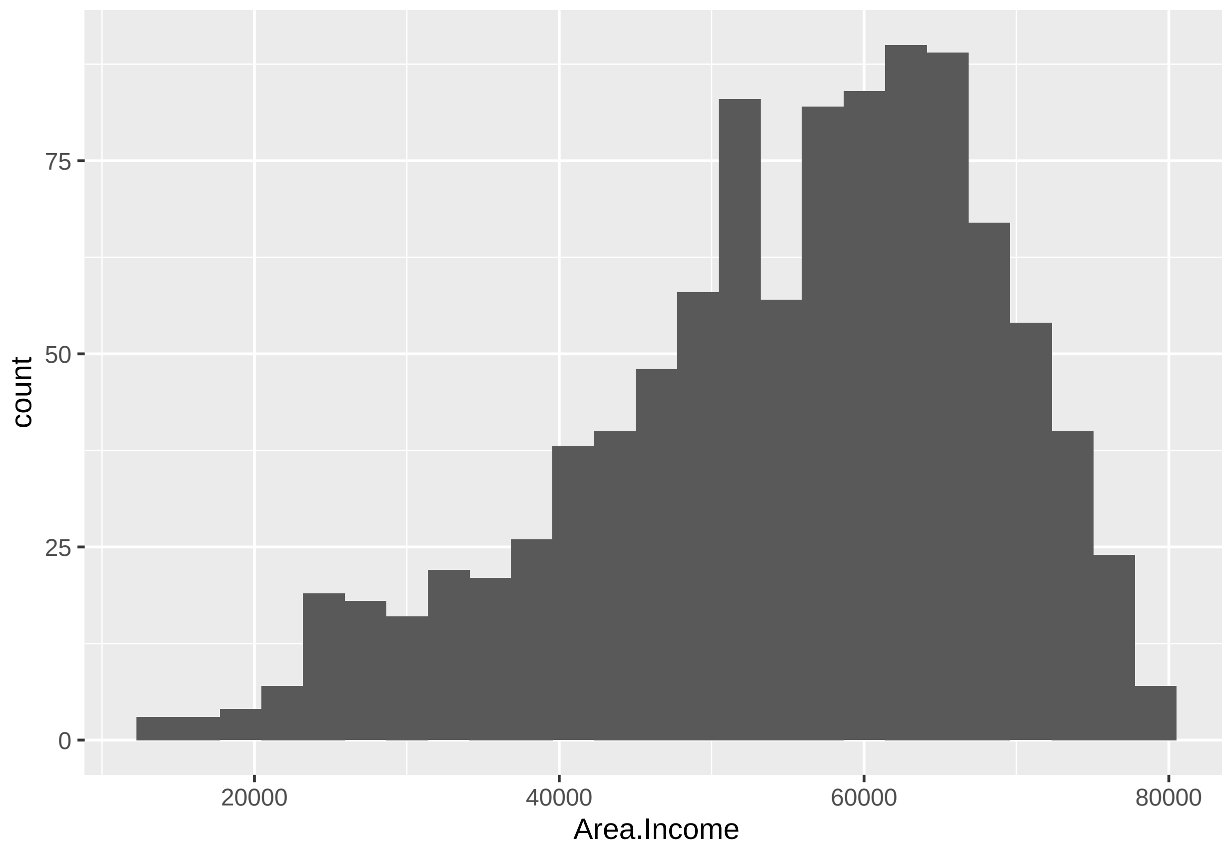
Age versus clicked

```
counts = table(advertising$Age, advertising$Clicked.on.Ad)
barplot(counts, main="number of Clicks on an Ad as per Age",
xlab="Ad Clicks", col=c("blue","grey"),
legend = rownames(counts), beside=TRUE)
```



Area Income versus clicks

```
area_plt = ggplot(data = advertising,col=c("blue","grey"), aes(x = Area.Income, fill = Clicked.on.Ad))+
geom_histogram(bins = 25)
area_plt
```



Modeling

Decision Trees

Subsetting dataset

```
#excluding columns categorical columns
advert= advertising[, c(1,2,3,4,7,10)]
head(advert)
```

```
##   Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage Male
## 1                    68.95  35    61833.90               256.09    0
## 2                    80.23  31    68441.85               193.77    1
## 3                    69.47  26    59785.94               236.50    0
## 4                    74.15  29    54806.18               245.89    1
## 5                    68.37  35    73889.99               225.58    0
## 6                    59.99  23    59761.56               226.74    1
##   Clicked.on.Ad
## 1             0
## 2             0
## 3             0
## 4             0
## 5             0
## 6             0
```

Shuffling the rows

```
rows <- sample(nrow(advert))

# Shuffle
advert <- advert[rows, ]

head(advert,10)
```

```
##     Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage Male
## 255                    56.70  48    62784.85               123.13    0
## 99                     35.61  46    51868.85               158.22    0
## 777                    56.46  26    66187.58               151.63    0
## 884                    41.53  42    67575.12               158.81    0
## 870                    82.41  36    65882.81               222.08    0
## 631                    63.43  29    66504.16               236.75    1
## 270                    79.15  26    62312.23               203.23    0
## 57                     65.19  36    75254.88               150.61    0
## 651                    83.66  38    68877.02               175.14    0
## 962                    78.67  26    63319.99               195.56    0
##     Clicked.on.Ad
## 255             1
## 99              1
## 777             1
## 884             1
## 870             0
## 631             0
## 270             0
## 57              1
## 651             0
## 962             0
```

Spliting the data

```
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(skimr)
options(warn = -1)
#Set the seed to 100
set.seed(100)

#80% training and 20% testing
```

```r
train_set = createDataPartition(advert$Clicked.on.Ad, p=0.80, list=FALSE)

#train dataset
train = advert[train_set,]

#Create the test dataset
test = advert[-train_set,]

head(train,5)
```

```
##     Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage Male
## 255                    56.70  48    62784.85               123.13    0
## 884                    41.53  42    67575.12               158.81    0
## 870                    82.41  36    65882.81               222.08    0
## 631                    63.43  29    66504.16               236.75    1
## 270                    79.15  26    62312.23               203.23    0
##     Clicked.on.Ad
## 255             1
## 884             1
## 870             0
## 631             0
## 270             0
```

```r
head(test,5)
```

```
##     Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage Male
## 99                     35.61  46    51868.85               158.22    0
## 777                    56.46  26    66187.58               151.63    0
## 57                     65.19  36    75254.88               150.61    0
## 962                    78.67  26    63319.99               195.56    0
## 4                      74.15  29    54806.18               245.89    1
##     Clicked.on.Ad
## 99              1
## 777             1
## 57              1
## 962             0
## 4               0
```

Setting the variables

```r
x = train
y = train$Clicked.on.Ad
#Checking the x and y head to confirm setting of dependent and independent variables
head(x,5)
```

```
##     Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage Male
## 255                    56.70  48    62784.85               123.13    0
## 884                    41.53  42    67575.12               158.81    0
## 870                    82.41  36    65882.81               222.08    0
## 631                    63.43  29    66504.16               236.75    1
## 270                    79.15  26    62312.23               203.23    0
```

```
##     Clicked.on.Ad
## 255             1
## 884             1
## 870             0
## 631             0
## 270             0
```

```r
head(y,5)
```

```
## [1] 1 1 0 0 0
```

###Training the model: Random Forest

```r
install.packages("randomForest")
```

```
## Installing package into '/home/sharon-maswai/R/x86_64-pc-linux-gnu-library/3.6'
## (as 'lib' is unspecified)
```

```r
library(randomForest)
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

Fitting the model

```r
model = randomForest(Clicked.on.Ad ~ ., data = train, importance = TRUE)

model
```

```
##
## Call:
##  randomForest(formula = Clicked.on.Ad ~ ., data = train, importance = TRUE)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 1
##
##           Mean of squared residuals: 0.03563158
##                     % Var explained: 85.75
```

```r
#Checking prediction
pred = predict(model, train, type = "class")

#Checking the accuracy of classification
mean(pred == train$Clicked.on.Ad)
```

```
## [1] 0
```

```r
accuracy = table(pred, train$Clicked.on.Ad)
accuracy
```

```
##
## pred                 0 1
##   0.0485770969939074 1 0
##   0.0490200252095296 1 0
##   0.0490623152449157 1 0
##   0.0492888050362683 1 0
##   0.0494692315731404 1 0
##   0.0495443212484512 1 0
##   0.0495883936955844 1 0
##   0.0496273184781104 1 0
##   0.049856042415068  1 0
##   0.049925810566639  1 0
##   0.0499696983245519 1 0
##   0.0500437242985778 1 0
##   0.050389474565137  1 0
##   0.0506536171626351 1 0
##   0.0506700002950513 1 0
##   0.0509446510295149 1 0
##   0.0510045281037194 1 0
##   0.0510576334056256 1 0
##   0.05109735010755   1 0
##   0.0513811179374879 1 0
##   0.0514062038514816 1 0
##   0.0514596720004368 1 0
##   0.0516285180192527 1 0
##   0.0518811217107681 1 0
##   0.0518865362690847 1 0
##   0.05201097441258   1 0
##   0.0520953800952689 1 0
##   0.0522880248859493 1 0
##   0.0523114021834048 1 0
##   0.0524346245124916 1 0
##   0.052493591838536  1 0
##   0.0525279688352154 1 0
##   0.0525381244057371 1 0
##   0.052585313379757  1 0
##   0.0526668432701185 1 0
##   0.052955967299623  1 0
##   0.0530346390713471 1 0
##   0.053393784998124  1 0
##   0.0535146929634766 1 0
```

```
##    0.0535257230342282 1 0
##    0.0535858603115724 1 0
##    0.0536632451019246 1 0
##    0.0540074195376335 1 0
##    0.0542426357535059 1 0
##    0.0543726733264422 1 0
##    0.0548883982813119 1 0
##    0.0550720755119942 1 0
##    0.0553165027139031 1 0
##    0.0553855633954352 1 0
##    0.0556151827197285 1 0
##    0.0556719823762158 1 0
##    0.0558504275351261 1 0
##    0.0559424429711376 1 0
##    0.0559759883232876 1 0
##    0.0559803256529531 1 0
##    0.0559842520874245 1 0
##    0.0561646450039446 1 0
##    0.0562288038401588 1 0
##    0.0567092626347172 1 0
##    0.0569230660832829 1 0
##    0.0572802881005286 1 0
##    0.0577371689572258 1 0
##    0.0579545608430575 1 0
##    0.0580108799507184 1 0
##    0.0580294359439894 1 0
##    0.0581728977331146 1 0
##    0.0582739874541646 1 0
##    0.0583760806205545 1 0
##    0.0583825459401326 1 0
##    0.0584390887506358 1 0
##    0.0584593908509055 1 0
##    0.0586570557440541 1 0
##    0.0586604064878374 1 0
##    0.0594227436887635 1 0
##    0.0596592829112196 1 0
##    0.0599062051230446 1 0
##    0.0599947626108485 1 0
##    0.0600092725377364 1 0
##    0.0600292223828581 1 0
##    0.0600668903683118 1 0
##    0.0601944474746855 1 0
##    0.0602183673662406 1 0
##    0.0603435081263523 1 0
##    0.0603973264268371 1 0
##    0.0604033809390325 1 0
##    0.0604636798228383 1 0
##    0.0608723060762415 1 0
##    0.0608816086890361 1 0
##    0.0609018722935853 1 0
##    0.0609922811103219 1 0
##    0.0611417663937019 1 0
##    0.0612376546231991 1 0
##    0.0613914403439336 1 0
```

```
##    0.0614402477627804 1 0
##    0.0614730256912508 1 0
##    0.0614804184220713 1 0
##    0.0615840647120963 1 0
##    0.061723363896382  1 0
##    0.0617418190719001 1 0
##    0.0621353768861645 1 0
##    0.0623512164935696 1 0
##    0.0626410361199676 1 0
##    0.0626570169636192 1 0
##    0.0626747309820056 1 0
##    0.062715230157349  1 0
##    0.0627984732636657 2 0
##    0.062801475758611  1 0
##    0.0628325655678597 1 0
##    0.0628881732139966 1 0
##    0.062931181769098  1 0
##    0.0629351982715631 1 0
##    0.0629721208331692 1 0
##    0.0629987607605803 1 0
##    0.0630284518378698 1 0
##    0.0630884289238194 1 0
##    0.0631111689433936 1 0
##    0.0631910361199676 1 0
##    0.063252721749798  1 0
##    0.0632935296133794 1 0
##    0.063425296366909  1 0
##    0.0634813560344925 1 0
##    0.0635106778137541 1 0
##    0.0636448660891675 1 0
##    0.0637630148370024 1 0
##    0.0638386862475038 1 0
##    0.0642467919426897 1 0
##    0.0643723877025738 1 0
##    0.0644084374389856 1 0
##    0.0646587065411394 1 0
##    0.0647436002603611 1 0
##    0.0648758652252574 1 0
##    0.0648798817356921 1 0
##    0.0649258721319304 1 0
##    0.0649314065251228 1 0
##    0.0649830824058631 1 0
##    0.0654296934071954 1 0
##    0.0655287389524037 1 0
##    0.0655451923671414 1 0
##    0.0655847664483864 1 0
##    0.0658241919702699 1 0
##    0.0658869929194419 1 0
##    0.0660015683588491 1 0
##    0.0660207554663964 1 0
##    0.0660288581630274 1 0
##    0.0660465949999374 1 0
##    0.0661147484298646 1 0
##    0.0662377706001401 1 0
```

```
##   0.066306975766316  1 0
##   0.0664492579876494 1 0
##   0.0664695540269869 1 0
##   0.0665094687613994 1 0
##   0.0666702180048814 1 0
##   0.0668679560650209 1 0
##   0.0668953141115383 1 0
##   0.0670205711782062 1 0
##   0.0671675200977881 1 0
##   0.0674397154423407 1 0
##   0.0676097281082699 1 0
##   0.0676280259056843 1 0
##   0.0679418029894878 1 0
##   0.0683043180138962 1 0
##   0.0683159952470006 1 0
##   0.0683257563157975 1 0
##   0.0683298382608601 1 0
##   0.068338902248618  1 0
##   0.0683399336666324 1 0
##   0.0687705207094366 1 0
##   0.0690271340876619 1 0
##   0.0691934014436206 1 0
##   0.0692153141115384 1 0
##   0.0692561509932543 1 0
##   0.0697955341032442 1 0
##   0.0698003051418033 1 0
##   0.0698744317190159 1 0
##   0.0699245410780411 1 0
##   0.0699976897699839 1 0
##   0.0700414605693741 1 0
##   0.0700746180628282 1 0
##   0.0703407603972871 1 0
##   0.0703811740796207 1 0
##   0.0705040921933086 1 0
##   0.0705339174624386 1 0
##   0.0707223281350163 1 0
##   0.0707969220782252 1 0
##   0.0708689088143718 1 0
##   0.0710230631043356 1 0
##   0.071154129655522  1 0
##   0.0712874656823373 1 0
##   0.0712882719397359 1 0
##   0.0716358606590242 1 0
##   0.0717707634798926 1 0
##   0.0717970288936087 1 0
##   0.0718152342051416 1 0
##   0.0719695957654957 1 0
##   0.0721745410780411 1 0
##   0.0722118349139295 1 0
##   0.0723018962169264 1 0
##   0.0723231309266083 1 0
##   0.0725863592390176 1 0
##   0.0730882503192118 1 0
##   0.0731894200929004 1 0
```

```
##    0.0733478861065562 1 0
##    0.0734590687057413 1 0
##    0.0735114434165721 1 0
##    0.0738051876522124 1 0
##    0.0739862258704939 1 0
##    0.0740939856369533 1 0
##    0.0744098360544353 1 0
##    0.0744532269612909 1 0
##    0.0744572478462073 1 0
##    0.0744948801046401 1 0
##    0.0744998758587394 1 0
##    0.0745790652401718 1 0
##    0.0750681337836004 1 0
##    0.0751602847272096 1 0
##    0.0753651100816421 1 0
##    0.0754847953540519 1 0
##    0.0755925702971162 1 0
##    0.0758166681303253 1 0
##    0.0758294151972904 1 0
##    0.0758310397272038 1 0
##    0.0762425978852516 1 0
##    0.076538937980973  1 0
##    0.076630204355657  1 0
##    0.076802408558431  1 0
##    0.076962373608821  1 0
##    0.0778980908722351 1 0
##    0.0780268982320425 1 0
##    0.0785134223561373 1 0
##    0.0787680364346605 1 0
##    0.0788074270639537 1 0
##    0.079217271824089  1 0
##    0.0795626411512723 1 0
##    0.0796838764421333 1 0
##    0.0796917728063765 1 0
##    0.0799655223653654 1 0
##    0.0800400066518242 1 0
##    0.0800438385351947 1 0
##    0.080665895723312  1 0
##    0.0814083035849305 1 0
##    0.0817980155023885 1 0
##    0.0818383882826322 1 0
##    0.081995778833171  1 0
##    0.0827066057444523 1 0
##    0.0828375419764166 1 0
##    0.0831101392924234 1 0
##    0.0834516087099679 1 0
##    0.0839112673350434 1 0
##    0.0840193308737885 1 0
##    0.0840731002375962 1 0
##    0.0854454936412195 1 0
##    0.0855733572126914 1 0
##    0.0856949202642036 1 0
##    0.0860096416113422 1 0
##    0.0864295929028437 1 0
```

```
##    0.0873366075008313 1 0
##    0.0874766742505495 1 0
##    0.0886676256866537 1 0
##    0.0886856233593784 1 0
##    0.0893072707324216 1 0
##    0.0895357452112721 1 0
##    0.0898001184196439 1 0
##    0.0898961054419184 1 0
##    0.0905070324782009 1 0
##    0.0907097052076359 1 0
##    0.0907402961785683 1 0
##    0.0908051462258243 1 0
##    0.0908217842358944 1 0
##    0.0908232316745513 1 0
##    0.0910589741019909 1 0
##    0.0922194384932385 1 0
##    0.0924502650026052 1 0
##    0.0931052210156522 1 0
##    0.0934955239481898 1 0
##    0.0938080222933678 1 0
##    0.0939996822074591 1 0
##    0.0940700669496496 1 0
##    0.0949836028819591 1 0
##    0.0951348978792715 1 0
##    0.0967137109768811 1 0
##    0.0973672520054064 1 0
##    0.0981794837752886 1 0
##    0.098594169476353  1 0
##    0.0998180128617249 1 0
##    0.100158771241245  1 0
##    0.100762496323157  1 0
##    0.100796291308833  1 0
##    0.102086076335885  1 0
##    0.103172982341003  1 0
##    0.104320127118978  1 0
##    0.10491752027225   1 0
##    0.105188671369916  1 0
##    0.106458380893918  1 0
##    0.108117509474948  1 0
##    0.109053583662553  1 0
##    0.109141620981687  1 0
##    0.110055566113561  1 0
##    0.110427247155494  1 0
##    0.110762990145826  1 0
##    0.110872012473671  1 0
##    0.11096289197929   1 0
##    0.111485488562586  1 0
##    0.112437352599663  1 0
##    0.112453680103998  1 0
##    0.113766822565618  1 0
##    0.113836226179011  1 0
##    0.115994369586618  1 0
##    0.117735594097946  1 0
##    0.118185333920012  1 0
```

```
## 	0.120469675304712	1 0
## 	0.121490943725133	1 0
## 	0.121760043974335	1 0
## 	0.122810203499617	1 0
## 	0.123184910053119	1 0
## 	0.125350112254201	1 0
## 	0.125543346449235	1 0
## 	0.128255137686571	1 0
## 	0.128643720452812	1 0
## 	0.128844954409463	1 0
## 	0.128866249158323	1 0
## 	0.129032914331643	1 0
## 	0.129712839094319	1 0
## 	0.130779694765102	1 0
## 	0.130854659459458	1 0
## 	0.134426601945198	1 0
## 	0.137521736133298	1 0
## 	0.138544758663692	1 0
## 	0.13973928151802	1 0
## 	0.143013496209144	1 0
## 	0.143375993122107	1 0
## 	0.147431057470561	1 0
## 	0.147961574025015	1 0
## 	0.14831160201383	1 0
## 	0.15408539526662	1 0
## 	0.155166132161506	1 0
## 	0.155960284476755	1 0
## 	0.157939414235597	1 0
## 	0.160001089511049	1 0
## 	0.160720080752397	1 0
## 	0.162039510721528	1 0
## 	0.162479057285642	1 0
## 	0.163833126885656	1 0
## 	0.16740108218768	1 0
## 	0.168259224524898	1 0
## 	0.168849681670876	1 0
## 	0.169000586289972	1 0
## 	0.170679304971034	1 0
## 	0.17184104009851	1 0
## 	0.173878353701427	1 0
## 	0.177684732664408	1 0
## 	0.184979748087772	1 0
## 	0.185134580801622	1 0
## 	0.186587019926628	1 0
## 	0.188900207417503	1 0
## 	0.192493222781561	1 0
## 	0.1943993866297	1 0
## 	0.195451661850467	1 0
## 	0.197331283628324	1 0
## 	0.199993144270042	1 0
## 	0.200072309521753	1 0
## 	0.20022970702686	1 0
## 	0.201867133455467	1 0
## 	0.202709086556083	1 0
```

```
##    0.202726228885656  1 0
##    0.20782344203053   1 0
##    0.215589065190642  1 0
##    0.217625453258301  1 0
##    0.220055059382038  1 0
##    0.220509209441346  1 0
##    0.230438409506919  1 0
##    0.231390840227947  1 0
##    0.23385986795163   1 0
##    0.239502222623509  1 0
##    0.241810126457014  1 0
##    0.243540226215655  1 0
##    0.244776141291585  1 0
##    0.248306934131015  1 0
##    0.252523471168177  1 0
##    0.256861811693794  1 0
##    0.258669874749158  1 0
##    0.261100434858554  1 0
##    0.265398156433482  1 0
##    0.268340446790125  1 0
##    0.272644307930509  1 0
##    0.282927801526681  1 0
##    0.283931656184062  1 0
##    0.284882183080329  1 0
##    0.28907655924758   1 0
##    0.295454024707689  0 1
##    0.304545779482438  1 0
##    0.326093093026061  1 0
##    0.328940716718473  1 0
##    0.33061766244406   1 0
##    0.333142567707461  1 0
##    0.340274832004548  1 0
##    0.342650681044991  0 1
##    0.356067765429106  0 1
##    0.362212440658287  0 1
##    0.375289358428947  0 1
##    0.380879242772181  1 0
##    0.383647865407326  1 0
##    0.423682180040877  1 0
##    0.45167569748555   1 0
##    0.504456349628979  1 0
##    0.540166785432263  0 1
##    0.543293169628549  0 1
##    0.544601701004387  0 1
##    0.549560847335325  0 1
##    0.609159283007331  0 1
##    0.611647479906026  0 1
##    0.618964919192653  0 1
##    0.625310496646236  0 1
##    0.671411946986042  0 1
##    0.672110273857054  0 1
##    0.674967988852092  0 1
##    0.685799600924101  0 1
##    0.689261094824096  0 1
```

```
## 0.695263302197518 0 1
## 0.696793923035187 0 1
## 0.710210873398813 0 1
## 0.715383516188635 0 1
## 0.720029332557508 0 1
## 0.725513065419441 0 1
## 0.734632949319791 0 1
## 0.735639872644098 0 1
## 0.73964926599143  0 1
## 0.740888033318443 0 1
## 0.747181794891402 0 1
## 0.750366276231324 0 1
## 0.754431895798604 0 1
## 0.755859862018458 0 1
## 0.75652162568405  0 1
## 0.756793640379228 0 1
## 0.756916315504735 0 1
## 0.763901778659517 0 1
## 0.764896455000196 0 1
## 0.766590173243481 0 1
## 0.769729374385695 0 1
## 0.770288125296733 0 1
## 0.770841859637494 0 1
## 0.775054324777934 0 1
## 0.775725452998851 0 1
## 0.779831004140333 0 1
## 0.784962504782083 0 1
## 0.790724093486248 0 1
## 0.795305511027981 0 1
## 0.797988357837341 0 1
## 0.799078349820666 0 1
## 0.805595977701031 0 1
## 0.808234034420696 0 1
## 0.808870406450025 0 1
## 0.809270027958836 0 1
## 0.811407250639491 0 1
## 0.813126958496738 0 1
## 0.814712552460296 0 1
## 0.81713349195168  0 1
## 0.817591671837837 0 1
## 0.817914788802102 0 1
## 0.825047209125026 0 1
## 0.82683929947956  0 1
## 0.826889187320551 0 1
## 0.838114105959785 0 1
## 0.838805256597349 0 1
## 0.839321010573624 0 1
## 0.840549791601136 0 1
## 0.841173563011496 0 1
## 0.843304426315925 0 1
## 0.844056781712331 0 1
## 0.845461710425604 0 1
## 0.846892694019646 0 1
## 0.847477837204377 0 1
```

```
## 0.849945692442587 0 1
## 0.850894737308256 0 1
## 0.851768777866335 0 1
## 0.852316573355253 0 1
## 0.856144933438419 0 1
## 0.856779093866216 0 1
## 0.859221727461609 0 1
## 0.861810488512652 0 1
## 0.861864820229704 0 1
## 0.864880372254416 0 1
## 0.866536010780601 0 1
## 0.867091468047089 0 1
## 0.870661280713275 0 1
## 0.870695945736237 0 1
## 0.872046641770486 0 1
## 0.872055866456037 0 1
## 0.872730357987085 0 1
## 0.875039715443928 0 1
## 0.875357040740584 0 1
## 0.878750194024899 0 1
## 0.879950044128469 0 1
## 0.881443427361302 0 1
## 0.881462278939079 0 1
## 0.881795383340205 0 1
## 0.885377159719583 0 1
## 0.886319993279113 0 1
## 0.88647356432041  0 1
## 0.886515018310725 0 1
## 0.886577553541568 0 1
## 0.887881521523237 0 1
## 0.889252228409368 0 1
## 0.889581312508256 0 1
## 0.890625410231633 0 1
## 0.892409281858446 0 1
## 0.893715954387373 0 1
## 0.894816176752205 0 1
## 0.894886725648975 0 1
## 0.895928774165521 0 1
## 0.896221826891265 0 1
## 0.897729936130632 0 1
## 0.898166647216833 0 1
## 0.899921765967825 0 1
## 0.901852203437447 0 1
## 0.902701465766833 0 1
## 0.903032580754197 0 1
## 0.905644460897671 0 1
## 0.907602710372276 0 1
## 0.907847911277813 0 1
## 0.908432213928579 0 1
## 0.908683235230936 0 1
## 0.909494965412614 0 1
## 0.909826614484786 0 1
## 0.909840278231475 0 1
## 0.910393083141661 0 1
```

```
##    0.911558465162151  0 1
##    0.912545619938294  0 1
##    0.913222798321475  0 1
##    0.914495405025167  0 1
##    0.915245528139236  0 1
##    0.915586445660767  0 1
##    0.916120884628095  0 1
##    0.916302300207296  0 1
##    0.916824321434039  0 1
##    0.917474831164229  0 1
##    0.917678962951514  0 1
##    0.918576763553236  0 1
##    0.918830058433281  0 1
##    0.919054594930568  0 1
##    0.91943141355162   0 1
##    0.919714990091909  0 1
##    0.919921412373833  0 1
##    0.920233998818382  0 1
##    0.920568982490577  0 1
##    0.920575315279457  0 1
##    0.921794105206618  0 1
##    0.921819178328577  0 1
##    0.922059277422036  0 1
##    0.922135816207678  0 1
##    0.92213953980885   0 1
##    0.922416573304437  0 1
##    0.922599813265585  0 1
##    0.922994240652742  0 1
##    0.922998404107969  0 1
##    0.923115219401555  0 1
##    0.923486017893844  0 1
##    0.924409510937859  0 1
##    0.924661565433278  0 1
##    0.925170939101818  0 1
##    0.925422283218627  0 1
##    0.925430500450593  0 1
##    0.925464118494957  0 1
##    0.926026839021141  0 1
##    0.926239498961812  0 1
##    0.926317981218882  0 1
##    0.926997240586432  0 1
##    0.927207559744212  0 1
##    0.927349265518627  0 1
##    0.927946675706862  0 1
##    0.928004272435151  0 1
##    0.9284556836544    0 1
##    0.928910918459568  0 1
##    0.929359998821903  0 1
##    0.929838102819268  0 1
##    0.930564638460195  0 1
##    0.930690745138351  0 1
##    0.930700896113692  0 1
##    0.931320021699766  0 1
##    0.93207009579602   0 1
```

```
##    0.93228312814087    0 1
##    0.932316770145589  0 1
##    0.932777099162008  0 1
##    0.933055052833471  0 1
##    0.933060392373169  0 1
##    0.933473064018573  0 1
##    0.93357196766895    0 1
##    0.933587662248399  0 1
##    0.933632528488505  0 1
##    0.933731650682247  0 1
##    0.934682045784645  0 1
##    0.935152973191957  0 1
##    0.935519330124311  0 1
##    0.935583848683804  0 1
##    0.935982491503786  0 1
##    0.936157185935983  0 1
##    0.936433297356666  0 1
##    0.936470343136089  0 1
##    0.937134923333388  0 1
##    0.937387454433286  0 1
##    0.937784239436159  0 1
##    0.938164473451438  0 1
##    0.93859295571692    0 1
##    0.93867946032974    0 1
##    0.938954237160038  0 1
##    0.939027237083074  0 1
##    0.939099231588392  0 1
##    0.939212266620719  0 1
##    0.939239764892744  0 1
##    0.939518243915607  0 1
##    0.939972466971668  0 1
##    0.940001194611235  0 1
##    0.940251930145723  0 1
##    0.940994821447499  0 1
##    0.941060981704438  0 1
##    0.941164353194583  0 1
##    0.941164667509383  0 1
##    0.941331571459232  0 1
##    0.941466391660082  0 1
##    0.941566277992983  0 1
##    0.942615742609897  0 1
##    0.942697829247483  0 1
##    0.942718243915607  0 1
##    0.942901469957846  0 1
##    0.942958253873542  0 1
##    0.943032926039197  0 1
##    0.943275661638237  0 1
##    0.943415113522761  0 1
##    0.943516446274755  0 1
##    0.943662301527613  0 1
##    0.944013660824523  0 1
##    0.944129326821482  0 1
##    0.944282762572165  0 1
##    0.944556565424353  0 1
```

```
## 0.945182072544942 0 1
## 0.945187703221595 0 1
## 0.945400580013962 0 1
## 0.945824893384393 0 1
## 0.946252446903917 0 1
## 0.946315578284708 0 1
## 0.94648146272911  0 1
## 0.946692859839334 0 1
## 0.947135705042697 0 1
## 0.947275615680197 0 1
## 0.947543141098311 0 1
## 0.947614683139097 0 1
## 0.947673105005944 0 1
## 0.947714994229596 0 1
## 0.947750974295289 0 1
## 0.947763250520446 0 1
## 0.948052031374856 0 1
## 0.948142664652211 0 1
## 0.948311043439979 0 1
## 0.948564628230331 0 1
## 0.948595974671663 0 1
## 0.948798228096449 0 1
## 0.948830058108523 0 1
## 0.948854212582082 0 1
## 0.948907747852785 0 1
## 0.949049138820498 0 1
## 0.9491311876006   0 1
## 0.949252697955847 0 1
## 0.949346499637923 0 1
## 0.949369232754321 0 1
## 0.949521201124317 0 1
## 0.949548436121931 0 1
## 0.949579966346822 0 1
## 0.949879883537093 0 1
## 0.949957976319115 0 1
## 0.950060791574415 0 1
## 0.950247974950635 0 1
## 0.950279889510501 0 1
## 0.950318189419395 0 1
## 0.950514877995071 0 1
## 0.950544365449665 0 1
## 0.950731286739764 0 1
## 0.950872081122068 0 1
## 0.951080617124092 0 1
## 0.951148386203403 0 1
## 0.951350316464923 0 1
## 0.951408693048276 0 1
## 0.951483713641765 0 1
## 0.951555952818939 0 1
## 0.951685602026864 0 1
## 0.952506963187122 0 1
## 0.952532802588991 0 1
## 0.952553698427116 0 1
## 0.952675704269471 0 1
```

```
##    0.952739456595758  0 1
##    0.952803217355922  0 1
##    0.953303217355922  0 1
##    0.95366132072912   0 1
##    0.953766233551381  0 1
##    0.954279929157036  0 1
##    0.954510193266769  0 1
##    0.954636819249409  0 1
##    0.954707841378284  0 1
##    0.955000812927359  0 1
##    0.955254488214997  0 1
##    0.95539749723897   0 1
##    0.955406391572438  0 1
##    0.955412657881913  0 1
##    0.95554834683929   0 1
##    0.955599623540235  0 1
##    0.955661277867933  0 1
##    0.955774741252743  0 1
##    0.955904545630461  0 1
##    0.956248424048442  0 1
##    0.956474867727068  0 1
##    0.95665529037987   0 1
##    0.956669918498496  0 1
##    0.956897301472193  0 1
##    0.957066535585456  0 1
##    0.957515610817961  0 1
##    0.957622290639573  0 1
##    0.957660451410436  0 1
##    0.957661517907403  0 1
##    0.957861680655056  0 1
##    0.958202598992422  0 1
##    0.95821096693783   0 1
##    0.958247456638088  0 1
##    0.958286146391575  0 1
##    0.958314798370702  0 1
##    0.958338215904481  0 1
##    0.958349929862612  0 1
##    0.958485892909991  0 1
##    0.958652468962765  0 1
##    0.958748297434666  0 1
##    0.958762547004221  0 1
##    0.958852318355569  0 1
##    0.958903521737363  0 1
##    0.959196099293606  0 1
##    0.959379412604265  0 1
##    0.959411192239644  0 1
##    0.959472983194275  0 1
##    0.959562087355782  0 1
##    0.959665657377849  0 1
##    0.959748161895451  0 1
##    0.959939487940056  0 1
##    0.960099952067601  0 1
##    0.960173787314299  0 1
##    0.960200287836907  0 1
```

```
##    0.960345418077137  0 1
##    0.960375475245251  0 1
##    0.960379952067601  0 1
##    0.960543973362465  0 1
##    0.960555050636637  0 1
##    0.960818329638398  0 1
##    0.960871917000392  0 1
##    0.960883213791322  0 1
##    0.960899952067601  0 1
##    0.961022284879565  0 1
##    0.961400191856309  0 1
##    0.961440360872469  0 1
##    0.961793132261136  0 1
##    0.961874747377923  0 1
##    0.961985535447711  0 1
##    0.962154840237012  0 1
##    0.96234653288798   0 1
##    0.962617040820438  0 1
##    0.9626928185866    0 1
##    0.962866502131508  0 1
##    0.962870845683544  0 1
##    0.96353189904637   0 1
##    0.964311828125912  0 1
##    0.964542078340274  0 1
##    0.964607955784501  0 1
##    0.964750812927359  0 1
##    0.965253546660553  0 1
##    0.965426374233445  0 1
##    0.966593106729541  0 1
##    0.96686383571683   0 1
##    0.966882296905277  0 1
##    0.967445744059823  0 1
##    0.96794716972615   0 1
##    0.968275988040538  0 1
##    0.968451005528151  0 1
##    0.968570898406226  0 1
##    0.968585181896366  0 1
##    0.968616639113162  0 1
##    0.968666315317616  0 1
##    0.968942410045534  0 1
##    0.968975411673607  0 1
##    0.969095133631552  0 1
##    0.969298841909246  0 1
##    0.969529646806118  0 1
##    0.970145721063146  0 1
##    0.970279651736902  0 1
##    0.970329831570484  0 1
##    0.97058109102238   0 1
##    0.970669727777778  0 1
##    0.970675988040538  0 1
##    0.970707828964087  0 1
##    0.970903792965972  0 1
##    0.971205232251449  0 1
##    0.971243873937632  0 1
```

```
##   0.971377646044553  0 1
##   0.971387318767474  0 1
##   0.971593627632213  0 1
##   0.971727399739134  0 1
```

```r
#Prediction on the test set
pred_test <- predict(model, test, type = "class")

#Checking the classification accuracy
mean(pred_test == test$Clicked.on.Ad)
```

```
## [1] 0
```

```r
table(pred_test,test$Clicked.on.Ad)
```

```
##
## pred_test            0 1
##   0.0483619597629037 1 0
##   0.0493154353405164 1 0
##   0.0502611582999974 1 0
##   0.0502831334474604 1 0
##   0.0504644764648881 1 0
##   0.0506234520269898 1 0
##   0.0513588088045242 1 0
##   0.0527745895817915 1 0
##   0.0534341870710557 1 0
##   0.0544801648156761 1 0
##   0.055288357452805  1 0
##   0.0567067814569856 1 0
##   0.0571649443053114 1 0
##   0.0579254554422101 1 0
##   0.0593596765825113 1 0
##   0.0606082235748213 1 0
##   0.0608660720330581 1 0
##   0.0610930034495709 1 0
##   0.0624611058754586 1 0
##   0.0630501817154585 1 0
##   0.0633915449284206 1 0
##   0.0650279859504942 1 0
##   0.067011810714304  1 0
##   0.0672276605582407 1 0
##   0.0673362055694944 1 0
##   0.0673509586571575 1 0
##   0.0677708819380632 1 0
##   0.0677727458760926 1 0
##   0.0684316759722753 1 0
##   0.0684401804763839 1 0
##   0.0689828818450924 1 0
##   0.0691055586925582 1 0
##   0.0701151013829042 1 0
##   0.0705501153133618 1 0
##   0.0721416830722832 1 0
##   0.0725706004595641 1 0
```

```
##    0.0728064041554851 1 0
##    0.0773672166686711 1 0
##    0.0774399953303281 0 1
##    0.0777678824521477 1 0
##    0.0780408666918706 1 0
##    0.0787615977046684 1 0
##    0.0798115871844262 1 0
##    0.0800154526512489 1 0
##    0.0802795752552861 1 0
##    0.0807479272670774 1 0
##    0.0813272442073483 1 0
##    0.0816883770385136 1 0
##    0.082102709771679  1 0
##    0.0822080707853388 1 0
##    0.0825827653473646 1 0
##    0.0827570652166633 1 0
##    0.0837494320140313 1 0
##    0.086419364708648  1 0
##    0.0888652875802441 1 0
##    0.0889544929248034 1 0
##    0.0911625770405231 1 0
##    0.0928897918421695 1 0
##    0.0953040465152102 1 0
##    0.0972068912523461 1 0
##    0.104113342979076  1 0
##    0.106493135099975  1 0
##    0.108218287405168  1 0
##    0.110783722602667  1 0
##    0.11563369947235   1 0
##    0.116145227531459  1 0
##    0.124495142052497  1 0
##    0.133416759067439  1 0
##    0.138936159534157  1 0
##    0.142028902242328  0 1
##    0.142280279296969  1 0
##    0.142932793855034  1 0
##    0.145125288012455  1 0
##    0.148022006115544  1 0
##    0.14808599133438   1 0
##    0.15307345905487   1 0
##    0.162470074246052  1 0
##    0.166771605713013  1 0
##    0.169311673160765  1 0
##    0.180092495586733  1 0
##    0.194958201762258  1 0
##    0.220881016405284  1 0
##    0.225619892010995  1 0
##    0.228898306878585  1 0
##    0.253795265099112  1 0
##    0.261481172230072  1 0
##    0.262721527759683  1 0
##    0.267495057610163  1 0
##    0.289218060578016  1 0
##    0.305797367723734  1 0
```

```
##    0.306661324409709  1 0
##    0.307045575513781  0 1
##    0.320328064672456  1 0
##    0.322251114037025  1 0
##    0.327472503217805  1 0
##    0.347393653901184  1 0
##    0.360929454925042  1 0
##    0.427447488830841  1 0
##    0.487396874793465  1 0
##    0.48901578004134   0 1
##    0.490777302715652  0 1
##    0.565462531134996  1 0
##    0.577806778148543  0 1
##    0.59448976880012   0 1
##    0.600431868345866  0 1
##    0.602339860021551  1 0
##    0.615839378098772  0 1
##    0.646512186442816  0 1
##    0.648964666960944  1 0
##    0.653251619701279  1 0
##    0.684942746279167  0 1
##    0.697251008005658  0 1
##    0.712084119945178  0 1
##    0.730376150409675  0 1
##    0.738186688211467  0 1
##    0.742847522842982  0 1
##    0.74596563599806   0 1
##    0.773456477842798  0 1
##    0.779572641531276  0 1
##    0.785199854140646  0 1
##    0.789777959805151  0 1
##    0.794205150215739  0 1
##    0.79936514361198   0 1
##    0.809880771329182  0 1
##    0.810042860673864  0 1
##    0.823812800006614  0 1
##    0.825672443337915  0 1
##    0.842888388938487  0 1
##    0.846376265367824  0 1
##    0.852240111503049  0 1
##    0.857248396999325  0 1
##    0.863098276698611  0 1
##    0.86394835434202   0 1
##    0.86766019792435   0 1
##    0.882733511832158  0 1
##    0.891868843637183  0 1
##    0.898217525367053  0 1
##    0.898600763495523  0 1
##    0.899196885595763  0 1
##    0.90248368325995   0 1
##    0.903792643796113  0 1
##    0.908306149767478  0 1
##    0.912758780143107  0 1
##    0.913824060857289  0 1
```

```
##    0.915460628647817  0 1
##    0.915960628647817  0 1
##    0.916871989280941  0 1
##    0.924348484969796  0 1
##    0.92599167289399   0 1
##    0.926297904443545  0 1
##    0.927804275214415  0 1
##    0.928514219764597  0 1
##    0.92886169875917   0 1
##    0.932600549261253  0 1
##    0.933611822292701  0 1
##    0.935239414781609  0 1
##    0.936459363979429  0 1
##    0.936642807943276  0 1
##    0.937240630260931  0 1
##    0.939058481029427  0 1
##    0.939840452421823  0 1
##    0.940630395052571  0 1
##    0.942081570615996  0 1
##    0.942239892560604  0 1
##    0.943183072099696  0 1
##    0.943626805610536  0 1
##    0.944164839078951  0 1
##    0.945585586226429  0 1
##    0.946783564979481  0 1
##    0.947374145461317  0 1
##    0.947760680054959  0 1
##    0.948801115727052  0 1
##    0.949053278859157  0 1
##    0.949103194537827  0 1
##    0.94915086847298   0 1
##    0.949234412281177  0 1
##    0.950417046975099  0 1
##    0.950736736568168  0 1
##    0.950865012953499  0 1
##    0.951176549988461  0 1
##    0.951738162910543  0 1
##    0.952821957046537  0 1
##    0.952986172056803  0 1
##    0.953719948000012  0 1
##    0.955002122269504  0 1
##    0.955692020754074  0 1
##    0.956604852234059  0 1
##    0.956820747273105  0 1
##    0.958180527369838  0 1
##    0.958821295093023  0 1
##    0.959131274387513  0 1
##    0.960310062887188  0 1
##    0.961059408491516  0 1
##    0.961844330793495  0 1
##    0.963196774725783  0 1
##    0.965534427655788  0 1
##    0.965616639113162  0 1
##    0.968854460863013  0 1
```

```
##    0.968975411673607  0 1
##    0.97058109102238   0 1
```

Training: Decision Trees

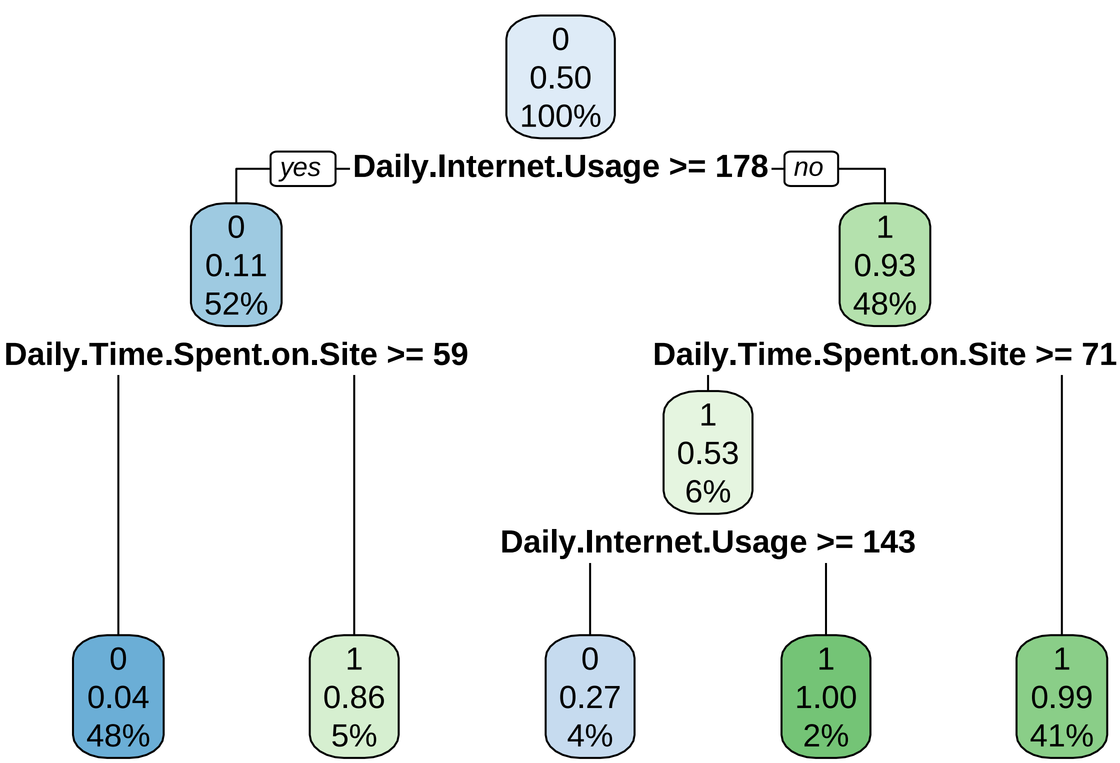
```r
# Importing library rpart
#
library(rpart)
#install.packages('rpart.plot')
library(rpart.plot)
```

Fitting the model

```r
# Training the model
#
model<- rpart(Clicked.on.Ad ~ ., data = train ,
           method = "class")
```

Ploting the rsults

```r
# Plotting findings
#
rpart.plot(model)
```



Predictions

```r
# Prediction on train dataset
pred <- predict(model, train, type = "class")
table(pred, train$Clicked.on.Ad)
```

```
##
## pred   0   1
##    0 392  24
##    1   8 376
```

```r
# Prediction on test dataset
pred <- predict(model, test, type = "class")
table(pred, test$Clicked.on.Ad)
```

```
##
## pred  0  1
##    0 95 10
##    1  5 90
```

```r
mean(test$Clicked.on.Ad == pred)
```

```
## [1] 0.925
```

Conclusion

- Both the decisn tree and random forest models have accuracies of 96% which is good.