# Advertising

##1. Defining the question ### Specifying the question To identify the trends in the dataset.

### Defining the metric for success

We will consider identifying trends as a success ### Understanding the context Companies pay money for a their brands to be advertised on certain websites . How often people visit a websites contributes to whether or not they will view and click on the advertisment. Analysing how often people visit a website will help us understand whether or not the people will atleast see the advertisment. ### Recording the experimental design We will perform our work in the following order: 1. Read the data 2. Check the data 3. Clean the data 4. Univariate Analysis 5. Bivariate Analysis 6. The conclusion

### Understanding the data

There are 1000 rows and 10 columns. In this dataset the columns are namely : Daily time spent on site Age Area Income Daily Internet Usage Ad Topic Line City Male Country Timestamp Clicked on Ad

##2. Reading the data

library(readr)  
advertising <- read\_csv("C:/Users/mutho/Downloads/advertising.csv")

##   
## -- Column specification --------------------------------------------------------------------------------------  
## cols(  
## `Daily Time Spent on Site` = col\_double(),  
## Age = col\_double(),  
## `Area Income` = col\_double(),  
## `Daily Internet Usage` = col\_double(),  
## `Ad Topic Line` = col\_character(),  
## City = col\_character(),  
## Male = col\_double(),  
## Country = col\_character(),  
## Timestamp = col\_datetime(format = ""),  
## `Clicked on Ad` = col\_double()  
## )

##3. Checking the data

# Checking the top of the dataset   
head(advertising)

## Warning: `...` is not empty.  
##   
## We detected these problematic arguments:  
## \* `needs\_dots`  
##   
## These dots only exist to allow future extensions and should be empty.  
## Did you misspecify an argument?

## # A tibble: 6 x 10  
## `Daily Time Spe~ Age `Area Income` `Daily Internet~ `Ad Topic Line` City   
## <dbl> <dbl> <dbl> <dbl> <chr> <chr>  
## 1 69.0 35 61834. 256. Cloned 5thgene~ Wrig~  
## 2 80.2 31 68442. 194. Monitored nati~ West~  
## 3 69.5 26 59786. 236. Organic bottom~ Davi~  
## 4 74.2 29 54806. 246. Triple-buffere~ West~  
## 5 68.4 35 73890. 226. Robust logisti~ Sout~  
## 6 60.0 23 59762. 227. Sharable clien~ Jami~  
## # ... with 4 more variables: Male <dbl>, Country <chr>, Timestamp <dttm>,  
## # `Clicked on Ad` <dbl>

# Checking the datatypes of the dataset   
str(advertising)

## tibble [1,000 x 10] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ Daily Time Spent on Site: num [1:1000] 69 80.2 69.5 74.2 68.4 ...  
## $ Age : num [1:1000] 35 31 26 29 35 23 33 48 30 20 ...  
## $ Area Income : num [1:1000] 61834 68442 59786 54806 73890 ...  
## $ Daily Internet Usage : num [1:1000] 256 194 236 246 226 ...  
## $ Ad Topic Line : chr [1:1000] "Cloned 5thgeneration orchestration" "Monitored national standardization" "Organic bottom-line service-desk" "Triple-buffered reciprocal time-frame" ...  
## $ City : chr [1:1000] "Wrightburgh" "West Jodi" "Davidton" "West Terrifurt" ...  
## $ Male : num [1:1000] 0 1 0 1 0 1 0 1 1 1 ...  
## $ Country : chr [1:1000] "Tunisia" "Nauru" "San Marino" "Italy" ...  
## $ Timestamp : POSIXct[1:1000], format: "2016-03-27 00:53:11" "2016-04-04 01:39:02" ...  
## $ Clicked on Ad : num [1:1000] 0 0 0 0 0 0 0 1 0 0 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. `Daily Time Spent on Site` = col\_double(),  
## .. Age = col\_double(),  
## .. `Area Income` = col\_double(),  
## .. `Daily Internet Usage` = col\_double(),  
## .. `Ad Topic Line` = col\_character(),  
## .. City = col\_character(),  
## .. Male = col\_double(),  
## .. Country = col\_character(),  
## .. Timestamp = col\_datetime(format = ""),  
## .. `Clicked on Ad` = col\_double()  
## .. )

# Getting the shape of the dataset   
dim(advertising)

## [1] 1000 10

# Finding out the class of the dataset   
class(advertising)

## [1] "spec\_tbl\_df" "tbl\_df" "tbl" "data.frame"

##4. Cleaning the data

# Checking the number of missing values in the columns   
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

# Checking for outliers   
library(tidyverse)

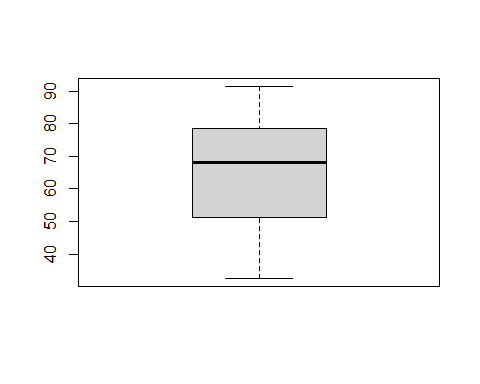
## -- Attaching packages --------------------------------------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3 v dplyr 1.0.5  
## v tibble 3.0.2 v stringr 1.4.0  
## v tidyr 1.1.3 v forcats 0.5.1  
## v purrr 0.3.4

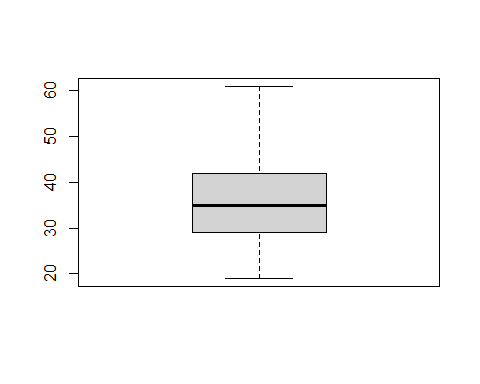
## -- Conflicts ------------------------------------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

time <- advertising$`Daily Time Spent on Site`  
age <- advertising$Age  
income <- advertising$`Area Income`  
internet <- advertising$`Daily Internet Usage`  
df<- data.frame(time,age,income,internet)

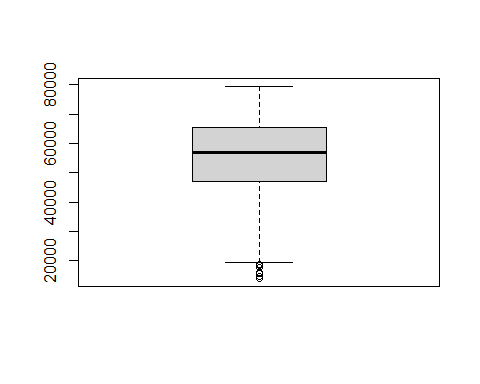
boxplot(time)



boxplot(age)



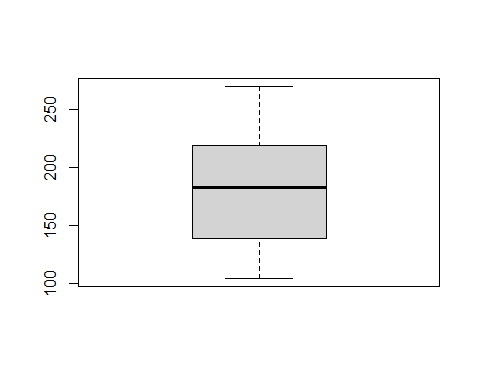
boxplot(income)



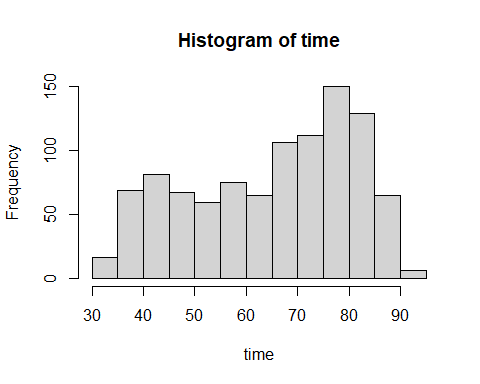
# Getting rid of the outliers   
Q <- quantile(advertising$`Area Income`, probs=c(.25, .75), na.rm = FALSE)  
iqr <- IQR(advertising$`Area Income`)  
up <- Q[2]+1.5\*iqr # Upper Range   
low<- Q[1]-1.5\*iqr # Lower Range

# Removing the outliers   
eliminated<- subset(advertising, advertising$`Area Income` > (Q[1] - 1.5\*iqr) & advertising$`Area Income` < (Q[2]+1.5\*iqr))

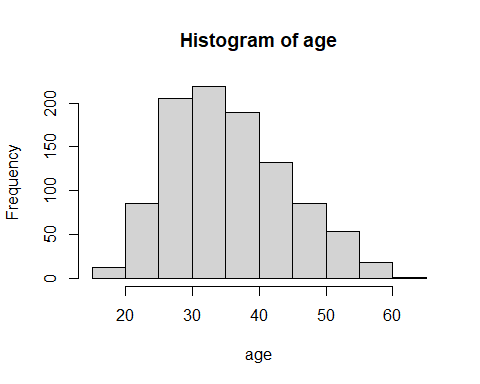
boxplot(internet)

 ##5. Univariate Analysis

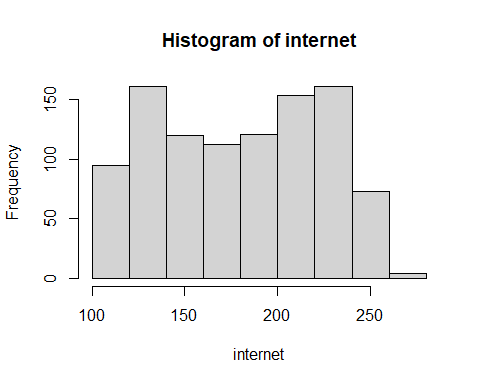
hist(time)



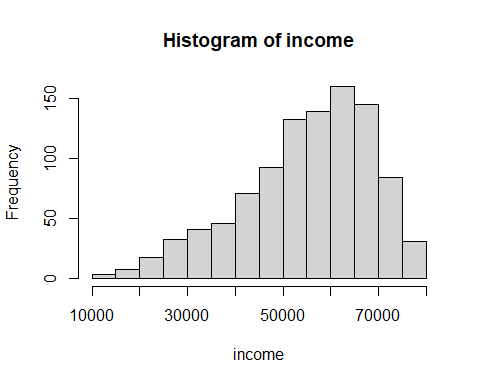
hist(age)



hist(internet)



hist(income)



# Getting the mean of the numeric columns   
mean.age <- mean(age)  
mean.age

## [1] 36.009

mean.internet <- mean(internet)  
mean.internet

## [1] 180.0001

mean.income <- mean(income)  
mean.income

## [1] 55000

mean.time <- mean(time)  
mean.time

## [1] 65.0002

# Getting the standard deviation of the numeric columns   
std.age <- sd(age)  
std.age

## [1] 8.785562

std.internet <- sd(internet)  
std.internet

## [1] 43.90234

std.income <- sd(income)  
std.income

## [1] 13414.63

std.time <- sd(time)  
std.time

## [1] 15.85361

# Getting the median of all the values   
median.age <- median(age)  
median.age

## [1] 35

median.internet <- median(internet)  
median.internet

## [1] 183.13

median.income <- median(income)  
median.income

## [1] 57012.3

median.time <- median(time)  
median.time

## [1] 68.215

# Getting the mode of the four columns   
getmode <- function(v) {  
 uniqv <- unique(v)  
 uniqv[which.max(tabulate(match(v, uniqv)))]  
}  
age.mode <- getmode(advertising$Age)  
age.mode

## [1] 31

getmode <- function(v) {  
 uniqv <- unique(v)  
 uniqv[which.max(tabulate(match(v, uniqv)))]  
}  
time.mode <- getmode(advertising$`Daily Time Spent on Site`)  
time.mode

## [1] 62.26

getmode <- function(v) {  
 uniqv <- unique(v)  
 uniqv[which.max(tabulate(match(v, uniqv)))]  
}  
income.mode <- getmode(advertising$`Area Income`)  
income.mode

## [1] 61833.9

getmode <- function(v) {  
 uniqv <- unique(v)  
 uniqv[which.max(tabulate(match(v, uniqv)))]  
}  
internet.mode <- getmode(advertising$`Daily Internet Usage`)  
internet.mode

## [1] 167.22

# Getting the values of the quantiles   
quantiles.age <- quantile(advertising$Age)  
quantiles.age

## 0% 25% 50% 75% 100%   
## 19 29 35 42 61

quantiles.internet <- quantile(advertising$`Daily Internet Usage`)  
quantiles.internet

## 0% 25% 50% 75% 100%   
## 104.7800 138.8300 183.1300 218.7925 269.9600

quantiles.income <- quantile(advertising$`Area Income`)  
quantiles.income

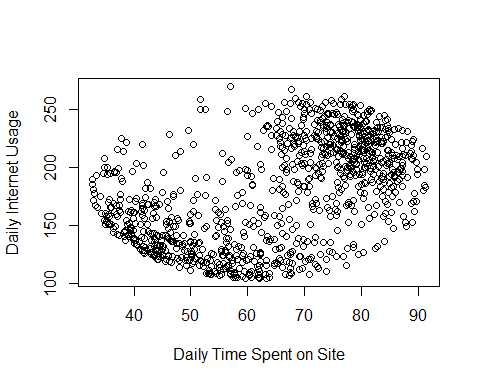
## 0% 25% 50% 75% 100%   
## 13996.50 47031.80 57012.30 65470.64 79484.80

quantiles.time <- quantile(advertising$`Daily Time Spent on Site`)  
quantiles.time

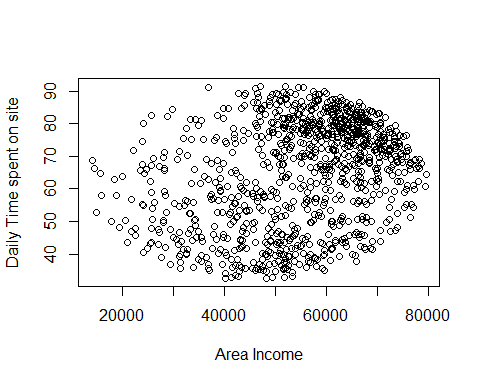
## 0% 25% 50% 75% 100%   
## 32.6000 51.3600 68.2150 78.5475 91.4300

##6. Bivariate Analysis

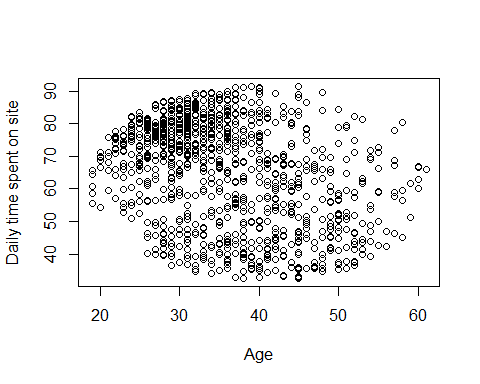
plot(time, internet, xlab="Daily Time Spent on Site", ylab="Daily Internet Usage")



plot(income,time, xlab="Area Income", ylab="Daily Time spent on site")



plot(age, time, xlab= "Age", ylab="Daily time spent on site")



#Getting the covariance of the numerical columns   
cov(time,internet)

## [1] 360.9919

cov(age,internet)

## [1] -141.6348

cov(income,internet)

## [1] 198762.5

# Getting the correlations   
cor(time,internet)

## [1] 0.5186585

cor(age,internet)

## [1] -0.3672086

cor(income,internet)

## [1] 0.3374955

We are now going to perform knn on our advertising dataset to predict who is going to click on the advertisment.

# We first remove the categorical columns   
advertising <- subset(advertising, select = -c(`Ad Topic Line`,City,Country,Timestamp))

set.seed(1234)  
  
# Randomizing the rows, creates a uniform distribution of 150  
random <- runif(1000)  
advertising\_random <- advertising[order(random),]  
  
# Selecting the first 6 rows from iris\_random  
head(advertising\_random)

## Warning: `...` is not empty.  
##   
## We detected these problematic arguments:  
## \* `needs\_dots`  
##   
## These dots only exist to allow future extensions and should be empty.  
## Did you misspecify an argument?

## # A tibble: 6 x 6  
## `Daily Time Spent~ Age `Area Income` `Daily Internet ~ Male `Clicked on Ad`  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 80.5 29 56909. 231. 0 0  
## 2 78.4 24 55015. 207. 0 0  
## 3 58.0 50 62466. 125. 0 1  
## 4 73.0 30 71385. 209. 1 1  
## 5 77.7 29 67081. 168. 0 0  
## 6 38.9 33 56370. 151. 1 1

normal <- function(x) (  
 return( ((x - min(x)) /(max(x)-min(x))) )  
)  
normal(1:5)

## [1] 0.00 0.25 0.50 0.75 1.00

advertising\_new <- as.data.frame(lapply(advertising\_random[,-5], normal))  
summary(advertising\_new)

## Daily.Time.Spent.on.Site Age Area.Income   
## Min. :0.0000 Min. :0.0000 Min. :0.0000   
## 1st Qu.:0.3189 1st Qu.:0.2381 1st Qu.:0.5044   
## Median :0.6054 Median :0.3810 Median :0.6568   
## Mean :0.5507 Mean :0.4050 Mean :0.6261   
## 3rd Qu.:0.7810 3rd Qu.:0.5476 3rd Qu.:0.7860   
## Max. :1.0000 Max. :1.0000 Max. :1.0000   
## Daily.Internet.Usage Clicked.on.Ad  
## Min. :0.0000 Min. :0.0   
## 1st Qu.:0.2061 1st Qu.:0.0   
## Median :0.4743 Median :0.5   
## Mean :0.4554 Mean :0.5   
## 3rd Qu.:0.6902 3rd Qu.:1.0   
## Max. :1.0000 Max. :1.0

# Now creating the train and the test datasets   
train <- advertising\_new[0:700,-5]  
test <- advertising\_new[701:1000,-5]  
train\_sp <- advertising\_new[0:700,5]  
test\_sp <- advertising\_new[701:1000,5]

library(class)   
require(class)  
model <- knn(train= train,test=test, ,cl= train\_sp,k=13)  
table(factor(model))

##   
## 0 1   
## 159 141

table(test\_sp,model)

## model  
## test\_sp 0 1  
## 0 151 3  
## 1 8 138

## The recommendation

We have seen that most of the time spent on the internet is 80 mins. We have also observed that the other features have no correlation with the time spent on site variable. We have also seen that most of the numeric data is skewed and does not have a normal distribution.We can also see the number of people predicted to click on the advert is less than the number of people predicted to not click on the advert. The advertising company should find better ways or different websites to place their adverts in order to get more views on their adverts.