**The importance of graphs**

Data visualization is the presentation of data with graphics. It's a way to summarize your findings and display it in a form that facilitates interpretation and can help in identifying patterns or trends. Having great data visualizations will make our work more interesting and clear.

**The difference between R libraries**

The differences between the basic plot() library, that comes with R, and ggplot2 are many. Ggplot2 was created to attend design demands and was based on the book "The Grammar of Graphics", a book that describes the foundations for data plotting.

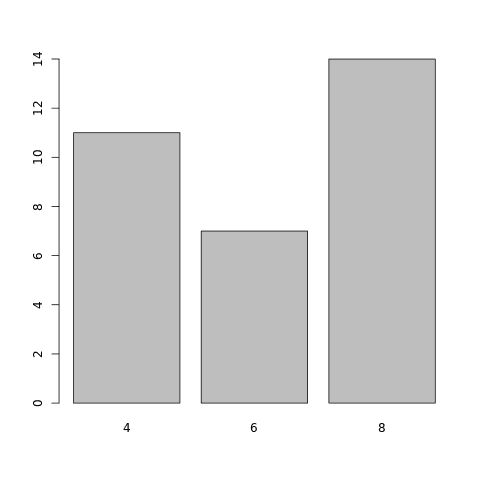
For example:

**plot**

The plot library is the default R library for plotting graphs. It's very simplistic in both syntax and aesthetics. To use it to create a bar plot, you use the barplot function, like so:

count **<-** table(mtcars**$**cyl)

barplot(count)

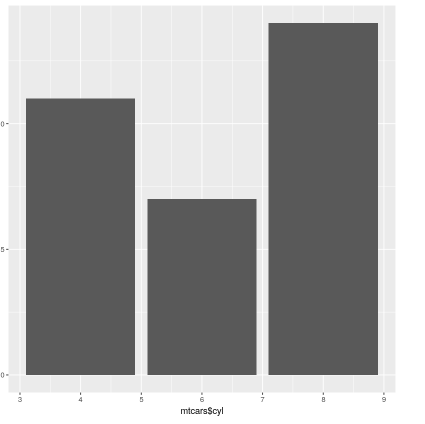


**ggplot2**

ggplot2, as mentioned above, is a specialized library made to create visually pleasing data visualizations. Before we can use ggplot2, we need to import it into the R environment. The code cell below will check if your system already has ggplot, as to not run install.packages for no reason. Then, using the library function, we can then import ggplot2.

**if**("ggplot2" **%in%** rownames(installed.packages()) **==** FALSE) {install.packages("ggplot2")} library(ggplot2)

qplot(mtcars**$**cyl, geom **=** "bar")



 ggplot2 offers us a nicer-looking graph, but has a slightly more complex syntax than the default plot library.

**ggplot2 has two principal functions, qplot() and ggplot().**

• qplot() offers a simpler syntax similar to the default plot function, but is limited in customization.

• ggplot() is the full-fledged function. It has far more possible customizations, but has a more complicated syntax than qplot().

**Making Bar Plots**

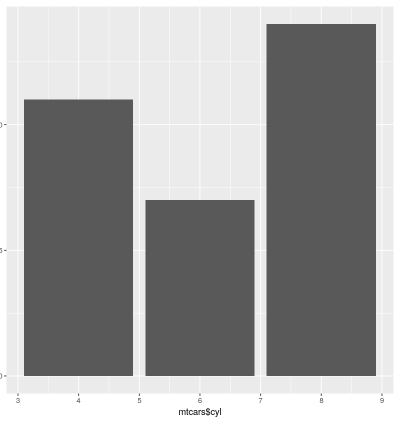
It's a plot format that shows your data using bars. Before actually creating any bar plot, let's import our plotting library, ggplot2.

**if**("ggplot2" **%in%** rownames(installed.packages()) **==** FALSE) {install.packages("ggplot2")}

library(ggplot2)

Now that we have loaded our libraries, let's start plotting. To plot easily using ggplot2, we can use the qplot function, which has simpler syntax, like so:

qplot(mtcars**$**cyl, geom **=** "bar")



As we can see, we plotted a bar plot, consisting of the count of every element with the same value. We can now exploit some of the possibilities of ggplot2's qplot function.

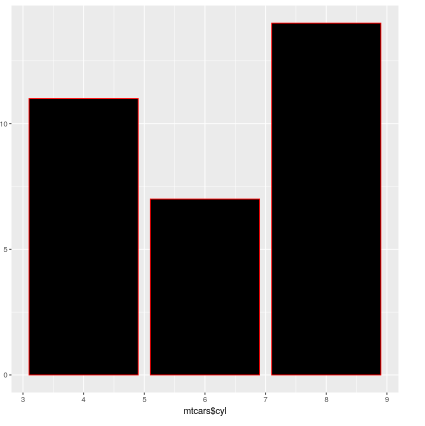
**Quick plot**

**Description**

qplot is a shortcut designed to be familiar if you're used to base plot(). It's a convenient wrapper for creating a number of different types of plots using a consistent calling scheme. It's great for allowing me to produce plots quickly.

This graph is plain as it stands right now. A plain graph is an excellent choice for academic papers, but for Internet content… simply being gray will not catch people’s attention. Let's give it some colour using the colour and the fill parameters. colour will modify the colour of the outline, while fill will change the colour of the bars.

qplot(mtcars**$**cyl, geom **=** "bar", fill **=** I("black"), colour **=** I("red"))

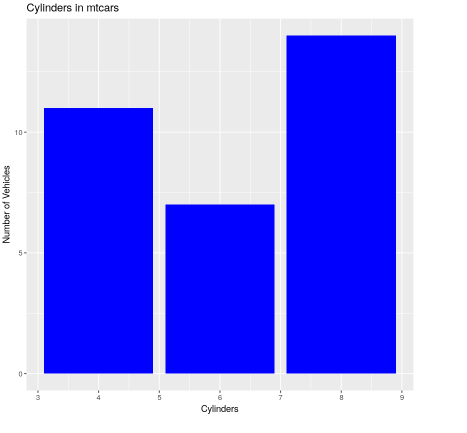


We can also change the name of our axes to make it more easily understandable by passing the xlab and ylab parameters (lab stands for "label"):

qplot(mtcars**$**cyl, geom **=** "bar", fill **=** I("blue"), xlab **=** "Cylinders", ylab **=** "Number of Vehicles")

To finish our bar plot, we can give a name for our graph. We can do this using the main parameter.

qplot(mtcars**$**cyl, geom **=** "bar", fill **=** I("blue"), xlab **=** "Cylinders", ylab **=** "Number of Vehicles", main **=** "Cylinders in mtcars")



**Pie charts**

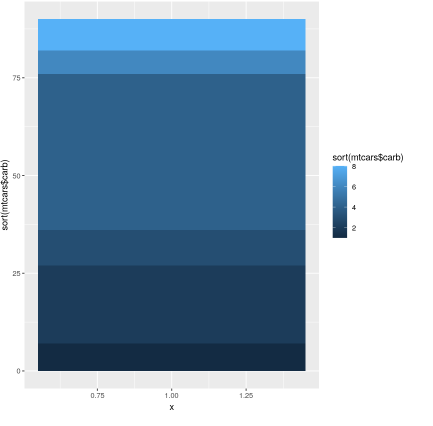
A pie chart is a circular graph, most commonly used in business. It shows the proportion that each part of data contributes to an overall total. This is usually done as a representation of the counts of qualitative data, like demographics.

A pie chart in ggplot2 is a transformed stacked bar plot. A stacked bar plot is a plot that stacks all the values on the vertical axis, instead of creating separate bars for each different data point.

barp **<-** ggplot(mtcars, aes(x**=**1, y**=**sort(mtcars**$**carb), fill**=**sort(mtcars**$**carb))) **+**

       geom\_bar(stat**=**"identity")

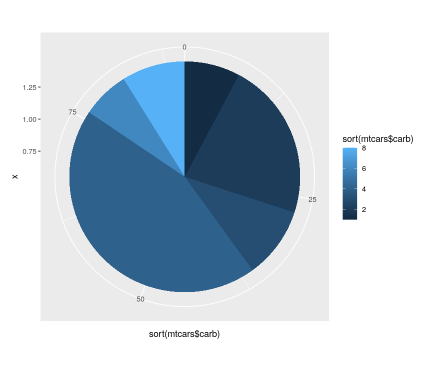
print(barp)



barp **<-** barp **+** coord\_polar(theta**=**'y')

​

print(barp)

  
  
  
Date: 23.09.2020

**Data Science Project-Regression Analysis**

**Problem Statement**: “FlightDelays.csv” is a dataset given here. It shows that there have been 2201 airplane flights in January 2004, flying from the Washington DC area into the NYC area.

What’s interesting to note is that the main characteristic of interest (the response) is if a flight has been delayed by more than 15 min (coded as 0 for no delay, and 1 for delay) or not. The explanatory variables include three different arrival airports—John F Kennedy, Newark, and LaGuardia. The three different departure airports are Reagan, Dulles, and Baltimore, along with eight carriers, and a categorical variable for 16 different hours of departure (6 am to 10 pm). The other variables include weather conditions (0 = good/1 = bad) and day of the week (1 for Sunday and Monday and 0 for all other days).

Apply logistic regression analysis to identify the flights that are likely to be delayed. This can be done by selecting 60% of the cases of data set (1320 cases) for the fitting (training) data set, wherein the remaining 40% of the cases (881 cases) become the evaluation data set.

For classification, the logistic regression model can be used that includes all explanatory variables as covariates. On the basis of the available training data set, the logistic regression model can help you estimate and classify a new case as a success in case its predicted probability of success is greater than 0.5. Do not forget to provide the R-script file with all the necessary codes.

Code and analysis used in Project:

> set.seed(1)

> ## read and print the data

> del <- read.csv(file.choose(),header=T)

> del[1:3,]

schedtime carrier deptime dest distance date flightnumber origin weather

1 1455 OH 1455 JFK 184 1/1/2004 5935 BWI 0

2 1640 DH 1640 JFK 213 1/1/2004 6155 DCA 0

3 1245 DH 1245 LGA 229 1/1/2004 7208 IAD 0

dayweek daymonth tailnu delay

1 4 1 N940CA ontime

2 4 1 N405FJ ontime

3 4 1 N695BR ontime

> table(del$sched)

600 630 640 645 700 730 735 759 800 830 840 845 850 900 925 930

26 57 22 21 92 24 17 2 40 26 62 3 31 77 3 28

1000 1030 1039 1040 1100 1130 1200 1230 1240 1245 1300 1315 1330 1359 1400 1430

23 56 6 15 48 20 22 28 31 61 109 4 19 25 46 52

1455 1500 1515 1520 1525 1530 1600 1605 1610 1630 1640 1645 1700 1710 1715 1720

138 77 5 1 21 50 45 1 24 51 27 30 74 28 61 27

1725 1730 1800 1830 1900 1930 2000 2030 2100 2120 2130

1 50 27 58 99 20 22 31 45 90 2

> ## define hours of departure

> del$sched=factor(floor(del$schedtime/100))

> table(del$carrier)

CO DH DL MQ OH RU UA US

94 551 388 295 30 408 31 404

> table(del$origin)

BWI DCA IAD

145 1370 686

> table(del$dest)

EWR JFK LGA

665 386 1150

> table(del$dayweek)

1 2 3 4 5 6 7

308 307 320 372 391 250 253

> table(del$weather)

0 1

2169 32

> table(del$delay)

delayed ontime

428 1773

> table(del$daymonth)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26

50 62 51 63 78 85 83 85 84 50 68 84 85 84 69 80 49 55 81 85 85 86 81 50 67 65

27 28 29 30 31

52 68 82 84 50

> del$delay=as.numeric(levels(del$delay)[del$delay])

Warning message:

NAs introduced by coercion

> del$delay=recode(del$delay,"'delayed'=1;else=0")

Error in recode(del$delay, "'delayed'=1;else=0") :

could not find function "recode"

> table(del$delay)

< table of extent 0 >

> ## Delay: 1=Monday; 2=Tuesday; 3=Wednesday; 4=Thursday;

> ## 5=Friday; 6=Saturday; 7=Sunday

> ## 7=Sunday and 1=Monday coded as 1

> del$dayweek=recode(del$dayweek,"c(1,7)=1;else=0")

Error in recode(del$dayweek, "c(1,7)=1;else=0") :

could not find function "recode"

> ## omit unused variables

> del=del[,c(-1,-3,-5,-6,-7,-11,-12)]

> table(del$dayweek)

1 2 3 4 5 6 7

308 307 320 372 391 250 253

> del[1:3,]

carrier dest origin weather dayweek delay sched

1 OH JFK BWI 0 4 NA 14

2 DH JFK DCA 0 4 NA 16

3 DH LGA IAD 0 4 NA 12

> n=length(del$delay)

> n1=floor(n\*(0.6))

> n

[1] 2201

> n1

[1] 1320

> n2

[1] 881

> n2=n-n1

> train=sample(1:n,n1)

> Xdel[1:3,]

Error in Xdel[1:3, ] : subscript out of bounds

> ## estimation of the logistic regression model

> ## explanatory variables: carrier, destination, origin, weather, day of week

> ## (weekday/weekend), scheduled hour of departure

> ## create design matrix; indicators for categorical variables (factors)

> Xdel <- model.matrix(delay~.,data=del)[,-1]

> xtrain <- Xdel[train,]

Error in Xdel[train, ] : subscript out of bounds

> xnew <- Xdel[-train,]

> ytrain <- del$delay[train]

> ynew <- del$delay[-train]

> m1=glm(delay~.,family=binomial,data=data.frame(delay=ytrain,xtrain))

Error in data.frame(delay = ytrain, xtrain) : object 'xtrain' not found

> summary(m1)

Error in summary(m1) : object 'm1' not found

> ## prediction: predicted default probabilities for cases in test set

> ptest <- predict(m1,newdata=data.frame(xnew),type="response")

Error in predict(m1, newdata = data.frame(xnew), type = "response") :

object 'm1' not found

> data.frame(ynew,ptest)[1:10,]

Error in data.frame(ynew, ptest) : object 'ptest' not found

> ## first column in list represents the case number of the test element

> plot(ynew~ptest)

Error in eval(predvars, data, env) : object 'ptest' not found

> 26

[1] 26

> ## coding as 1 if probability 0.5 or larger

> gg1=floor(ptest+0.5) ## floor function; see help command

Error: object 'ptest' not found

> ttt=table(ynew,gg1)

Error in table(ynew, gg1) : object 'gg1' not found

> ttt

Error: object 'ttt' not found

> error

Error: object 'error' not found

> ## coding as 1 if probability 0.3 or larger

> gg2=floor(ptest+0.7)

Error: object 'ptest' not found

> ttt

Error: object 'ttt' not found

> error=(ttt[1,2]+ttt[2,1])/n2

Error: object 'ttt' not found

> error

Error: object 'error' not found

> ttt=table(ynew,gg2)

Error in table(ynew, gg2) : object 'gg2' not found

> error=(ttt[1,2]+ttt[2,1])/n2

Error: object 'ttt' not found

> bb=cbind(ptest,ynew)

Error in cbind(ptest, ynew) : object 'ptest' not found

> bb

Error: object 'bb' not found

> bb1=bb[order(ptest,decreasing=TRUE),]

Error: object 'bb' not found

> bb1

Error: object 'bb1' not found

> xbar

[1] NA

> ## order cases in test set according to their success prob

> ## actual outcome shown next to it

> ## overall success (delay) prob in the evaluation data set

> xbar=mean(ynew)

> ## calculating the lift

> ## cumulative 1’s sorted by predicted values

> ## cumulative 1’s using the average success prob from evaluation set

> axis=dim(n2)

> ax=dim(n2)

> ay=dim(n2)

> axis[1]=1

> ax[1]=xbar

> ay[1]=bb1[1,2]

Error: object 'bb1' not found

> aaa=cbind(bb1[,1],bb1[,2],ay,ax)

Error in cbind(bb1[, 1], bb1[, 2], ay, ax) : object 'bb1' not found

> for (i in 2:n2) {

+ axis[i]=i

+ ax[i]=xbar\*i

+ ay[i]=ay[i-1]+bb1[i,2]

+ }

Error in bb1 : object 'bb1' not found

> aaa[1:100,]

Error: object 'aaa' not found

> points(axis,ax,type="l")

Error in plot.xy(xy.coords(x, y), type = type, ...) :

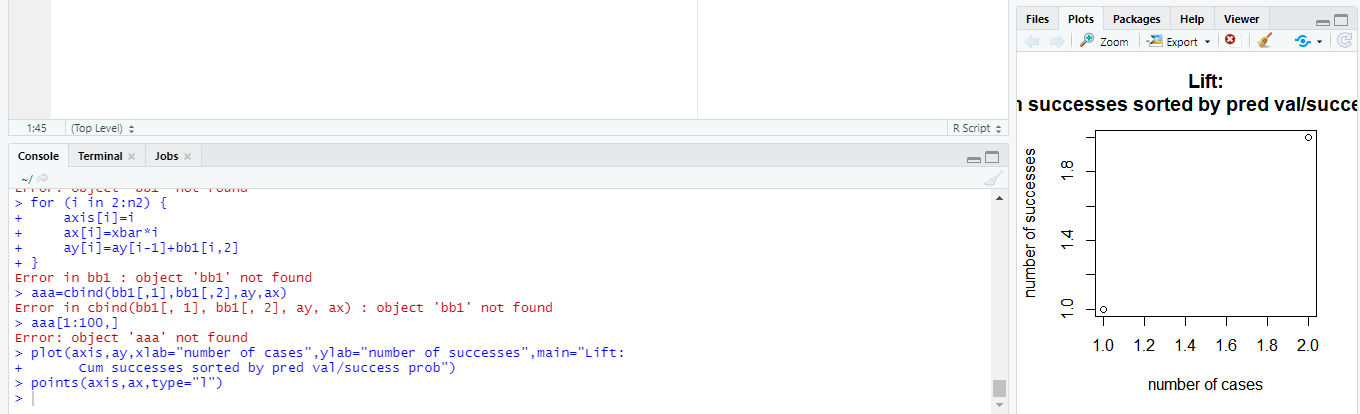
plot.new has not been called yet

> plot(axis,ay,xlab="number of cases",ylab="number of successes",main="Lift:

+ Cum successes sorted by pred val/success prob")

Output: NA

Output Snap:



Conclusion:

Date: 14.09.2020

**Project on Hypothesis Test- 01(Non Parametric Test)**

**Case:**

Consider a table that is a result of a survey conducted among students about their smoking habits, with the significance level of 0.05, check if the smoking habit of students are affecting their Exercise performance or not.

**Note:** (Case has used a inbuilt Data Set ‘’Survey’’)

**Code used:**

library(MASS)

head(survey)

tail(survey)

tbl<-table(survey%Smoke, survey$Exer)

tbl<-table(survey$Smoke,survey$Exer)

tbl

chisq.test(tbl)

**Process:**

> library(MASS)

> head(survey)

Sex Wr.Hnd NW.Hnd W.Hnd Fold Pulse Clap Exer Smoke Height M.I Age

1 Female 18.5 18.0 Right R on L 92 Left Some Never 173.00 Metric 18.250

2 Male 19.5 20.5 Left R on L 104 Left None Regul 177.80 Imperial 17.583

3 Male 18.0 13.3 Right L on R 87 Neither None Occas NA <NA> 16.917

4 Male 18.8 18.9 Right R on L NA Neither None Never 160.00 Metric 20.333

5 Male 20.0 20.0 Right Neither 35 Right Some Never 165.00 Metric 23.667

6 Female 18.0 17.7 Right L on R 64 Right Some Never 172.72 Imperial 21.000

> tail(survey)

Sex Wr.Hnd NW.Hnd W.Hnd Fold Pulse Clap Exer Smoke Height M.I Age

232 Male 18.0 16.0 Right R on L NA Right Some Never 180.34 Imperial 20.750

233 Female 18.0 18.0 Right L on R 85 Right Some Never 165.10 Imperial 17.667

234 Female 18.5 18.0 Right L on R 88 Right Some Never 160.00 Metric 16.917

235 Female 17.5 16.5 Right R on L NA Right Some Never 170.00 Metric 18.583

236 Male 21.0 21.5 Right R on L 90 Right Some Never 183.00 Metric 17.167

237 Female 17.6 17.3 Right R on L 85 Right Freq Never 168.50 Metric 17.750

> tbl<-table(survey%Smoke, survey$Exer)

Error: unexpected input in "tbl<-table(survey%Smoke, survey$Exer)"

> tbl<-table(survey$Smoke,survey$Exer)

> tbl

Freq None Some

Heavy 7 1 3

Never 87 18 84

Occas 12 3 4

Regul 9 1 7

> chisq.test(tbl)

Pearson's Chi-squared test

data: tbl

X-squared = 5.4885, df = 6, p-value = 0.4828

Warning message:

In chisq.test(tbl) : Chi-squared approximation may be incorrect

**Output:**

Pearson's Chi-squared test

data: tbl

X-squared = 5.4885, df = 6, p-value = 0.4828

**Conclusion:**

Since value (0.4828) is greater than significance Value (0.05) we cannot reject the null hypothesis and hence we can say Smoking habit of students is independent of their Exercise.

**Project on Hypothesis Test- 02 (Parametric Test)**

**Case:**

A Fast-food chain wants to test and market three of its new menu items. To analyze if they are equally popular, considering 18 random restaurants for the study, 6 of the restaurants to test market the first menu item, another 6 the second one, and the remaining 6 for the last one.

The table shows the sales figures of the menu items in the last 18 restaurants At. 0.05 level of significance, test whether the mean sales volumes for these menu items are equal.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | salary | | | |
| Service | Low | Medium | High | Total |
| Excellent | 9 | 10 | 7 | 26 |
| Good | 11 | 9 | 31 | 51 |
| Poor | 12 | 8 | 3 | 23 |
| Total | 32 | 27 | 41 | 100 |
|  |  |  |  |  |

**Code used:**

del <- read.csv(file.choose(),header=T)

del[1:3,]

del[1:6,]

r<-c(t(as.matrix(del)))

r

f<-c("item.1","item.2","item.3")

f

k<-3

n<-6

tm<-gl(k,1,n\*k,factor(f))

tm

tm<-gl(k,1,n\*k,factor(f))

tm

av<-aov(r~tm)

summary(av)

**Process:**

> del <- read.csv(file.choose(),header=T)

> del[1:3,]

item.1 item.2 item.3

1 22 52 16

2 42 33 24

3 44 8 19

> del[1:6,]

item.1 item.2 item.3

1 22 52 16

2 42 33 24

3 44 8 19

4 52 47 18

5 45 43 34

6 37 32 39

> r<-c(t(as.matrix(df1)))

Error in as.matrix(df1) : object 'df1' not found

> r<-c(t(as.matrix(del)))

> r

[1] 22 52 16 42 33 24 44 8 19 52 47 18 45 43 34 37 32 39

> f<-c("item.1","item.2","item.3")

> f

[1] "item.1" "item.2" "item.3"

> k<-3

> n<-6

> tm<-gl(k,1,n\*k,factor(f))

> tm

[1] item.1 item.2 item.3 item.1 item.2 item.3 item.1 item.2 item.3 item.1 item.2 item.3 item.1 item.2 item.3 item.1

[17] item.2 item.3

Levels: item.1 item.2 item.3

> tm<-gl(k,1,n\*k,factor(f))

> tm

[1] item.1 item.2 item.3 item.1 item.2 item.3 item.1 item.2 item.3 item.1 item.2 item.3 item.1 item.2 item.3 item.1

[17] item.2 item.3

Levels: item.1 item.2 item.3

> av<-aov(r~tm)

> summary(av)

Df Sum Sq Mean Sq F value Pr(>F)

tm 2 745.4 372.7 2.541 0.112

Residuals 15 2200.2 146.7

**Output:**

F value: 2.541, pValue: 0.11

**Conclusion:**

Since value (2.541) is greater than significance Value (0.05) we cannot reject the null hypothesis and hence we can say that the mean sales volume of the new items is all equal.