BioSys PhD | Earthsystems PhD

Statistics 1

Exploratory Data Analysis

Lisete Sousa
lmsousa@fc.ul.pt
Room: 6.4.25

Departamento de Estatística e Investigação Operacional
Faculdade de Ciências da Universidade de Lisboa

2017

Summary

1. Descriptive Statistics

2. Graphical Representations
   - Bar plot
   - Circular Diagram
   - Histogram
   - Box plot
   - Quantile-Quantile Plot

1. Descriptive Statistics

R, through package `stats`, allows the calculation of sample characteristics for one, or more, variables simultaneously.

- `mean()` → arithmetic mean
- `median()` → median
- `sd()` → standard deviation
- `var()` → variance
- `quantile()` → quantiles
- `min()` → minimum
- `max()` → maximum
- ...

Example:

```
> data<-c(2,5,3,7,1,4,2,5,2,7,1,3,2,3,5,6)
> mean(data)
[1] 3.625
> median(data)
[1] 3
> var(data)
[1] 3.983333
> sd(data)
[1] 1.995829
> quantile(data)

0%   25%   50%   75%  100%
1    2     3     5    7

> min(data)
[1] 1
> max(data)
[1] 7
```

Notes:

1. We may check the standard deviation by doing:

   ```
   > sqrt(var(data))
   [1] 1.995829
   ```

2. It is possible to calculate other quantiles by using function `quantile()`:

   ```
   quantile(data,c(0.2,0.7))
   20%   70%
   2     5
   ```

Let us calculate some of these characteristics for the database juul, available trough package ISwR, starting with assessing the variables contained therein.

```
>names(juul)
[1] "age" "menarche" "sex" "igf1" "tanner" "testvol"
```

To obtain the average ages we have to do:

```
>attach(juul)
>mean(age)
[1] NA
```

The average ages is not available (NA) because there are missing values

For these NA values to be ignored we have to include an extra argument:

```
> mean(age,na.rm=T)
[1] 15.09535

> colMeans(juul,na.rm=T)
       age   menarche       sex        igf1    tanner    testvol
 15.095352   1.475852  1.534483  340.167976  2.639672   7.895833
```

Note that the values indicated by the color red should not be considered because they are related to qualitative variables.

summary()

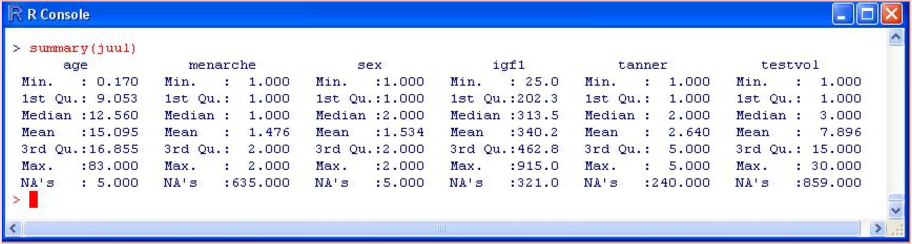
Function summary allows us to view a set of sample features concerning a certain variable.

```
> summary(age)
  Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's
 0.170    9.053   12.560   15.100   16.860   83.000    5.000
```

This function also returns the number of missing data.

This same function allows us to obtain a summary for all variables in the database:

```
> summary(juul)
```



```
> summary(juul)
      age           menarche            sex              igf1             tanner            testvol
 Min.   : 0.170   Min.   :  1.000   Min.   :1.000   Min.   : 25.0   Min.   :  1.000   Min.   :  1.000
 1st Qu.: 9.053   1st Qu.:  1.000   1st Qu.:1.000   1st Qu.:202.3   1st Qu.:  1.000   1st Qu.:  1.000
 Median :12.560   Median :  1.000   Median :2.000   Median :313.5   Median :  2.000   Median :  3.000
 Mean   :15.095   Mean   :  1.476   Mean   :1.534   Mean   :340.2   Mean   :  2.640   Mean   :  7.896
 3rd Qu.:16.855   3rd Qu.:  2.000   3rd Qu.:2.000   3rd Qu.:462.8   3rd Qu.:  5.000   3rd Qu.: 15.000
 Max.   :83.000   Max.   :  2.000   Max.   :2.000   Max.   :915.0   Max.   :  5.000   Max.   : 30.000
 NA's   : 5.000   NA's   :635.000   NA's   :5.000   NA's   :321.0   NA's   :240.000   NA's   :859.000
> 
```

NOTE: The qualitative variables (`sex, menarche` and `tanner`) are encoded as numerical (quantitative). This has to be changed as follows:

```
> juul$sex<-factor(juul$sex, labels=c("M","F"))
> juul$menarche<-factor(juul$menarche, labels=c("No","Yes"))
> juul$tanner<-factor(juul$tanner, labels=c("I","II","III",
+ "IV","V"))
```

Exercise:
Check now the new form of the summary table.

tapply()

Function tapply() calculates sample characteristics for the given variable for each level of a second variable, which may be qualitative (or categorical).

Example:

The average of *igf*1 per gender:

```
> tapply(igf1,sex,mean,na.rm=T)
       M          F
 310.8866   368.1006
```

NOTE: The argument na.rm=T must be inserted due to the presence of missing data.

table()

A simple way to describe qualitative data (and also grouped data) consists of building tables.

For univariate table construction the procedure is simple:

```
> table(sex)
sex
   M     F
 621  3713
```

For two variables a double-entry table is implemented:

```
> table(sex,tanner)
     tanner
 sex    I  II  III  IV    V
   M  291  55    34  41  124
   F  224  48    38  40  204
```

Is is also possible to build a triple-entry table:

```
> table(menarche,sex,tanner)
```

Let us see now, how to use R, trough the package graphics, to do graphical representations.

NOTE: There are other packages with graphical functions in R, such as, gplots, lattice, misc3d, etc.

2. Graphical Representations

Bar Plot

Bar Plot

The bar plot (diagram of bars) is built using the function `barplot()`.

If we want to represent variable `tanner` through a bar plot (absolute frequencies), we do:

```
> barplot(table(tanner))
```

If we want to represent `tanner` in a bar plot, with relative frequencies, we do:

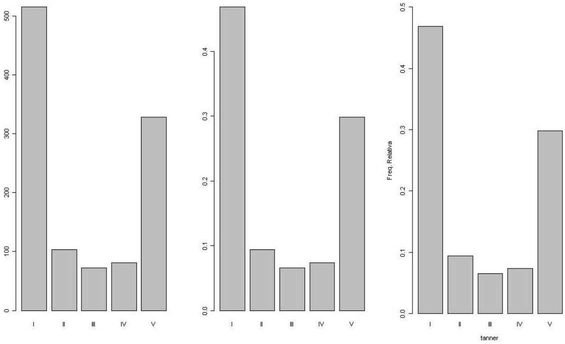
```
> table(tanner)/sum(!is.na(tanner))
```

NOTE:The relative frequency must be calculated based on the total number of non-missing observations. We run the function `sum(!is.na(tanner))`, or `length(tanner)-sum(is.na(tanner))`, to calculate the number of non-missing values.

The graphic may be improved if we include some more arguments:

```
> barplot(table(tanner)/(length(tanner)-240),xlab="tanner",
+ ylab="Relative Freq.",ylim=c(0,0.5))
```

```
> par(mfrow=c(1,3))
> barplot(table(tanner))
> sum(is.na(tanner))
> table(tanner)/(length(tanner)-240)
> barplot(table(tanner)/(length(tanner)-240))
> barplot(table(tanner)/(length(tanner)-240),xlab="tanner",
+ ylab="Relative Freq.",ylim=c(0,0.5))
```



Circular Diagram

Circular Diagram (Pie Chart)

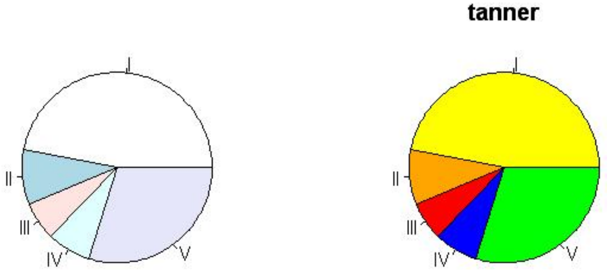
The circular diagram is built using the function `pie()`.

This diagram is valid for qualitative data and grouped data.

```
> pie(table(tanner))
```

We can change the colors and insert a title:

```
> slices<-c("yellow","orange","red","blue","green")
> pie(table(tanner),col=slices,main="tanner")
> par(mfrow=c(1,2),mex=0.3,mar=c(1,0))
> pie(table(tanner))
> slices<-c("yellow","orange","red","blue","green")
> pie(table(tanner),col=slices,main="tanner")
```



Histogram

Histogram

The histogram is built using the function `hist()`.

```
> hist(igf1)
```

By default, this function uses the rule of Sturges to build classes and represents the absolute frequencies.

You can also view the results in the form of text:

```
> hist(igf1,plot=FALSE)
```

You may be interested in:

- color the bars and represent the *densities* (relative frequency divided by the range of each class)

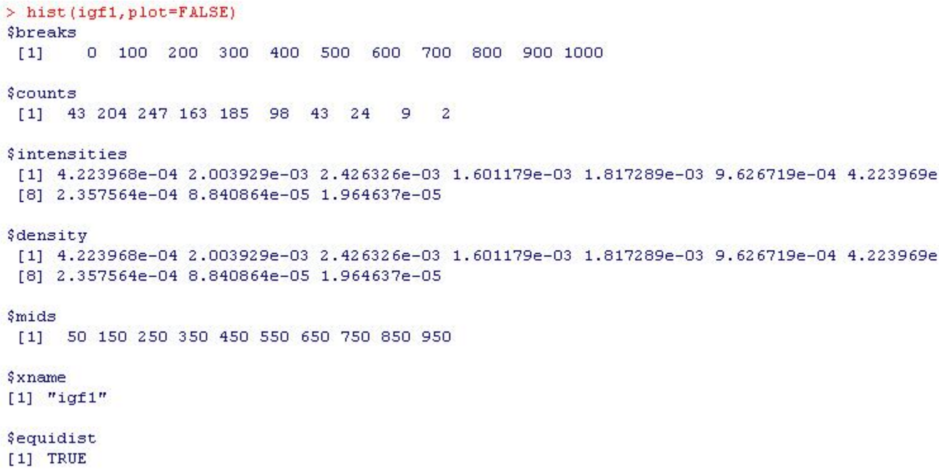
```
> hist(igf1,probability=TRUE,col="blue")
```

- define other classes

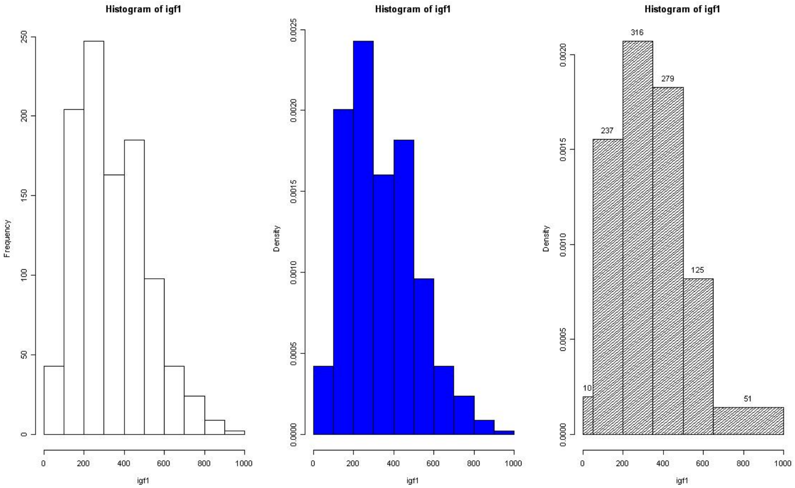
```
> class<-c(0,50,200,350,500,650,1000)
```

- put the absolute frequency at the top of each bar and shade them

```
> h<-hist(igf1,breaks=class,angle=45,density=40)
> h
> text(h$mids,h$density,h$counts,adj=c(0.5,-1))
```



```
> hist(igf1,plot=FALSE)
$breaks
 [1]    0  100  200  300  400  500  600  700  800  900 1000

$counts
 [1]  43 204 247 163 185  98  43  24   9   2

$intensities
 [1] 4.223968e-04 2.003929e-03 2.426326e-03 1.601179e-03 1.817289e-03 9.626719e-04 4.223969e
 [8] 2.357564e-04 8.840864e-05 1.964637e-05

$density
 [1] 4.223968e-04 2.003929e-03 2.426326e-03 1.601179e-03 1.817289e-03 9.626719e-04 4.223969e
 [8] 2.357564e-04 8.840864e-05 1.964637e-05

$mids
 [1]  50 150 250 350 450 550 650 750 850 950

$xname
[1] "igf1"

$equidist
[1] TRUE
```


Histogram of igf1
Frequency
0
50
100
150
200
250
0
200
400
600
800
1000
igf1
Histogram of igf1
Density
0.0000
0.0005
0.0010
0.0015
0.0020
0.0025
0
200
400
600
800
1000
igf1
Histogram of igf1
Density
0.0000
0.0005
0.0010
0.0015
0.0020
10
237
316
279
125
51
0
200
400
600
800
1000
igf1

Box Plot

Box plot

The box plot is built using the function `boxplot()`.

This graphical representation is suitable for discrete and continuous quantitative data.

The central line of the rectangle (box) represents the median of the observations. The lower and upper extremes of the box represent the first and third quartiles, respectively.

The dashes at the end of the vertical lines can represent:

1) the minimum and maximum of the sample

```
> boxplot(igf1,range=0)
```

2) the lowest and the highest values of the sample, which are not considered outliers

```
> boxplot(igf1)
```

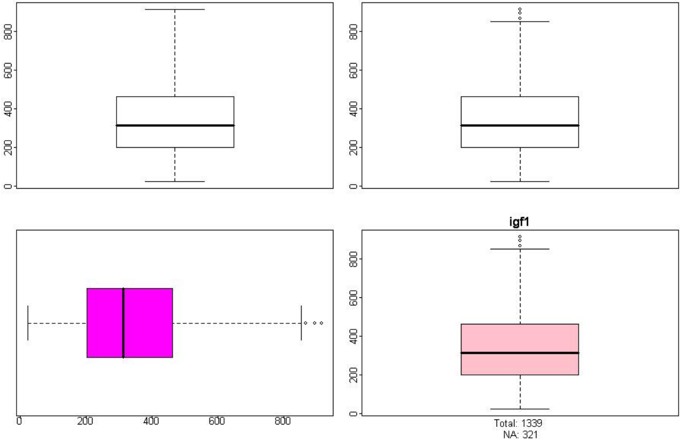
There are some arguments available for this representation:

- color the box make an horizontal representation

```
> boxplot(igf1,horizontal=T,col="magenta")
```

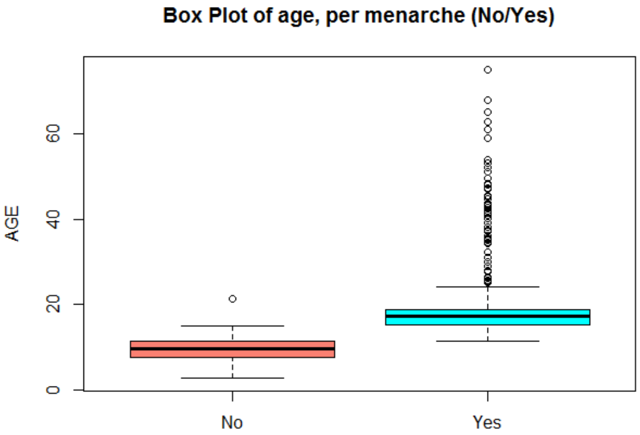
- insert title and subtitle

```
> boxplot(igf1,col="pink",main="IGF-1",sub=paste("Total:",
+ length(igf1),"\n","NA:",sum(is.na(igf1))))
```


0
200
400
600
800
igf1
Total: 1339
NA: 321

Box plots can also be made by groups:

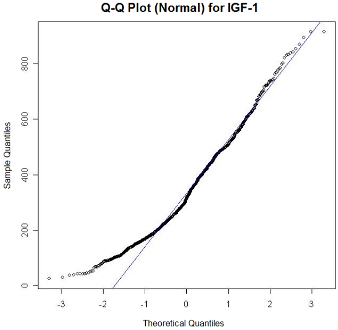
```
> boxplot(age∼menarche,col="orange",main="Box Plot of age,
+ per menarche (No/Yes)",ylab="AGE",names=("No","Yes"))
```



Quantile-Quantile Plot

Through the function qqnorm() we can represent the empirical quantiles (sample) *vs.* the theoretical quantiles according to the normal distribution. If the observations 'follow' a normal distribution, the points shall be provided on a line.

```
> qqnorm(igf1,main="Q-QPlot (Normal) for IGF-1")
> qqline(igf1,col="blue")
```



Acknowledgements:

Carina Silva-Fortes (ESTeSL – IPL), for allowing the use of some material produced by both of us in previous courses.

Bibliography:

Keen, K. J. (2010). *Graphics for Statistics and Data Analysis with R*. Chapman & Hall.

Maindonald, J. (2010). *Data Analysis and Graphics Using R: an Example-Based Approach*. Cambridge University Press.