2 Descriptive statistics with R

Before starting with basic concepts of data analysis, one should be aware of different types of data and ways to organize data in computer files.

2.1 Some basic terms

Population – an aggregate of subjects (creatures, things, cases and so on). For a given study, a *target population* has to be specified: on which subjects we'll generalize or use the results?

Sample – collection of subjects *in the study*. In general, a sample should be representative for the target population.

Observation – a study unit or *subject* or an individual. Often a human being, sometimes also an animal, plant or anything else.

Variable – quality or quantity, measured or recorded for each subject in the sample (age, sex, height, weight, smoking level, etc.).

Dataset – a set of values of all variables of interest for all individuals in the study. The numeric results obtained from the dataset will be used to draw conclusions about the target population.

2.2 Organization of data

A dataset is mostly organized (and stored as a computer file) in a form of a *data matrix*.

A data matrix representing sex (1-male; 0-female), age, no. of children, weight (kg), and height (cm) of 7 individuals:

NO	SEX	AGE	NO OF CHILDREN	WEIGHT	HEIGHT
1.	0	57	1	65	158
2.	1	70	3	100	175
3.	0	45	0	71	162
4.	0	38	2	58	164
5.	0	25	1	81	170
6.	1	50	4	68	172
7.	1	61	0	85	179

Each row of such a matrix represents one observation. All rows have the same length: the same data has been recorded for all individuals.
Each column represents one *variable*.

For instance, WEIGHT is the name of a variable, representing the body weight (in kg) of an individual.

2.3 Types of data

- **Numeric data**
  - Discrete data – the variable can take only integer values (0, 1, 2 etc.)
    examples: number of children, number of friends
  - Continuous data – any real-numbered values (often within a certain range) are possible
    examples: body weight, age
- **Qualitative (non-numeric, categorical) data**
  - Nominal data: unordered categories
    examples: blood group, eye color
  - Ordinal or ordered data: ordered categories
    examples: smoking level, attitudes (good-moderate-bad)

Numeric coding of nominal or ordered data does not make the data numeric!

2.4 Summarizing/presenting data

Continuous/discrete data

Summary *location* statistics: mean, median.

The sample *mean* is the arithmetic average of the data.

It can be calculated, by summing all of the data values and dividing the sum by the total sample size.

Example:

Data: 1 3 5 2 9
Mean: $(1 + 3 + 5 + 2 + 9)/5 = 20/5 = 4$

Mathematically: for a variable X, mean is often denoted as $\bar{x}$ and calculated as:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{x_1 + x_2 + \ldots + x_n}{n}$$

where $x_1, x_2, \ldots, x_n$ denote observations of a variable and n is the number of observations in the sample.

R:

```
> x <- c(1,3,5,2,9)
> mean(x)
> [1] 4
```

If there are missing values:

```
> x <- c(1,3,5,2,9,NA,7,10)
> mean(x)
[1] NA
> mean(x, na.rm=T)
[1] 5.285714
```

Median

Sometimes it is of interest to sort values of a variable in ascending or descending order. The order number of an observation in such a row is called as *rank*.

Median is the middle point of ordered data – either the middle observation (if the number of observations is odd) or the average of the two middle observations (if the number of observations is even).

Example:
Body heights of 11 individuals (in *cm*):

155, 160, 171, 182, 162, 153, 190, 167, 168, 165, 191.

Ordered data:

153 155 160 162 165 167 168 170 171 182 191

median: 167

R:

```
> x<-c(155, 160, 171, 182, 162, 153, 190, 167, 168, 165, 191)
> median(x)
[1] 167
> x<-c(155, 160, 171, 182, 162, 153, 190, 167, 168, 165, 191, 175)
> # added 1 observation
> median(x)
[1] 167.5
```

The advantage, but sometimes also the disadvantage of the median is, that it is not affected by extreme values in the data. It does not matter, how small or how big are the values that are larger or smaller than the median.

Neither the mean nor the median provides sufficient information about the data: one should also know about variability.

Standard deviation (SD, s) is a quantity that reflects the variability of the sample. One could interpret SD as the approximate mean distance from the mean.

More precisely, SD is defined as the square root of the **variance** (s^2) (sum of squared differences from the mean divided by sample size minus 1 (the latter called as the sample variance, s^2).

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

$$s = \sqrt{s^2} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}.$$

Similarly to the mean, standard deviation is sensitive to the extremes in the data.

R:

```
> x<-c(1,4,5,7,8,11)
> mean(x)
[1] 6
> median(x)
[1] 6
> var(x)   # variance
[1] 12
> sd(x)  # standard deviation
[1] 3.464102
> x<-c(1,4,5,7,8,110)  # change the last observation  from 11 to 110
> mean(x)
[1] 22.5
> median(x)
[1] 6
> var(x)
[1] 1843.5
> sd(x)
[1] 42.936
```

A more robust approach is to divide the distribution of the (ordered) data into four, and find the points below which are 25%, 50% and 75% of the distribution. These are known as **quartiles** (the median is the second quartile).

Example:
A sample:

```
6  9  9 10  9 10  3 12  7  6  6  4  8  8  3  8  6  4 11 11
```

The ordered sample

```
3  3  4  4  6  6  6  6  7  8  8  8  9  9  9  9 10 11 11 12
```

The ordered sample divided to 4 parts:

```
3  3  4  4  6 | 6  6  6  7  8 | 8  8  9  9  9 | 9 10 11 11 12
```

Quartiles: the cutpoints: 6, 8 (the median) and 9.

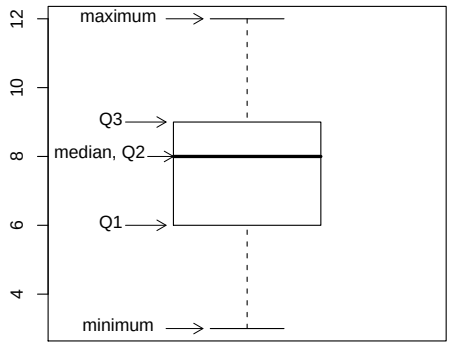
R:
In R, you can use function `quantile` to get median and quartiles, or you can also use te function `summary`, to get also the mean:

```
> z<-scan()
1: 3  3  4  4  6  6  6  6  7  8  8  8  9  9  9  9 10 11 11 12
21:
Read 20 items
> summary(z)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   3.00    6.00    8.00    7.45    9.00   12.00
```

The variation of the data can be summarized in the **interquartile range (IQR)**, the distance between the first and third quartile (here: IQR$= 9-6 = 3$).

In general pth **percentile** is the p%- cutpoint of ordered data (from smallest to largest). Sometimes in official statistics, deciles are used – 10th, 20th, etc percentiles.

A **boxplot** is a graphical representation of median and quartiles:

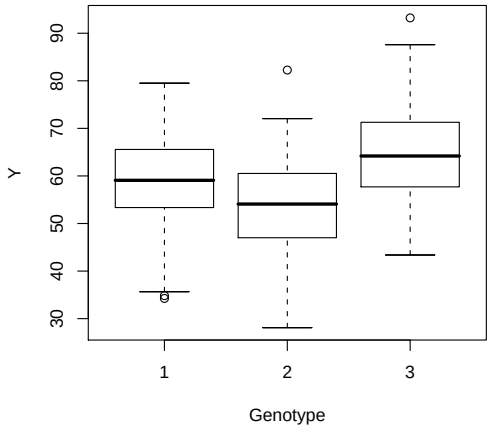


R:

```
> boxplot(x)
```

Boxplot gives an overview of the *distribution* of the data. It is often used to compare data across different groups.

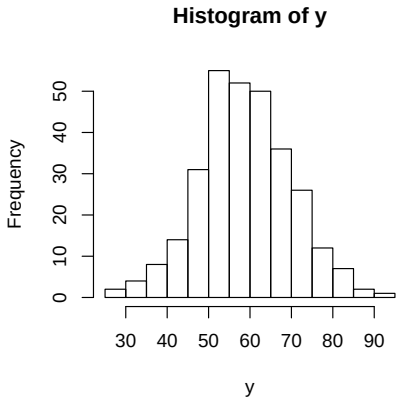
```
> boxplot(Y~g) # g is a categorical variable with values 1, 2 and 3 (genotype)
```



Another way to look at the distribution, is to plot a **histogram** of the data. To obtain a histogram, the scale of the variable is divided to consecutive intervals of equal length and the number of observations in each interval is counted.

R:

```
> hist(Y)
```



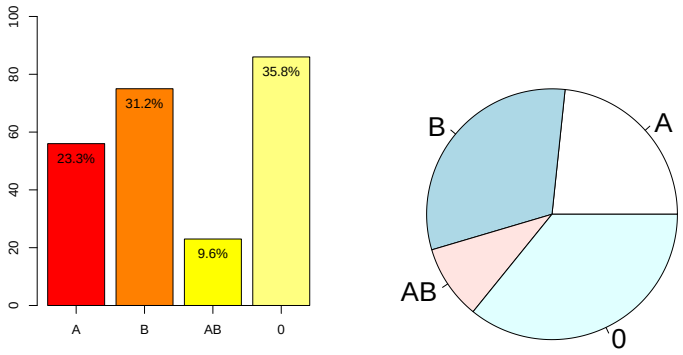
For nominal data, mean and standard deviation do not make much sense; neither do median and percentiles. To see the distribution of the data, look at the frequency table.

Example:

Blood groups of 240 individuals:

Blood group	n	%
A	56	23.3%
B	75	31.2%
AB	23	9.6%
0	86	35.8%

. . . and its graphical representation – bar chart or pie chart:



Note: pie charts are misleading and should be avoided!

R:

```
> eyecol<-c(1,2,1,2,2,2,3,3,1,4,2,2,2,3,1,4,3,2,1,1,1) # reading in data
> table(eyecol)   # a simple frequency table
eyecol
1 2 3 4
8 7 4 2
> eyecol<-factor(eyecol, labels=c("blue","grey","brown","green"))
>                                          # created a factor variable
> table(eyecol)                       # a more meaningful table
eyecol
 blue  grey brown green
    8     7     4     2
> prop.table(table(eyecol))   # relative frequencies
eyecol
     blue      grey     brown     green
0.3809524 0.3333333 0.1904762 0.0952381
> round(100*prop.table(table(eyecol)),1)
>                  # percentages, rounded to 1 decimal place
eyecol
 blue  grey brown green
 38.1  33.3  19.0   9.5
> barplot(table(eyecol))  # a simple bar chart
> barplot(table(eyecol),col=c("blue","grey","brown","green"),main="Eye color")
>                                                    # a nicer one
```

Some notes on statistical graphics:

- A graph should communicate useful information more efficiently than any other (numeric) summaries
- A good graph should have a good *information to ink ratio* – avoid fancy details, that do not add information, but make the graph more complicated (larger, more colorful).
- Pay attention to the scale of the graph!

2.5 R: Descriptive statistics by groups, 2-dimensional tables

Suppose you would like to compare means or other descriptive statistics in different subgroups of your sample. In R, you can use the function `tapply` for that. This function takes 3 arguments: the numeric variable, a categorical grouping variable and the function to apply.

To understand, how it works, try the following examples:

```
weight <- c(56, 67, 65, 78, 49, 87, 55, 63, 70, 72, 79, 52, 60, 78, 90)
sex <- c(1,1,1,2,1,2,1,1,1,2,1,1,1,2,2)
tapply(weight,sex,mean)
tapply(weight,sex,summary)
```

If you have 2 categorical variables, you might be interested in a 2-dimensional contingency table

```
eyecol<-c(1,2,1,2,2,2,3,3,1,4,2,2,2,3,1)
table(sex,eyecol)
prop.table(table(sex,eyecol),1) # row percentages
prop.table(table(sex,eyecol),2) # column percentages
```