Starting with R

Lino AA Notarantonio (lino@tec.mx)

21 March 2021

Contents

Introduction

We present in this document topics prerequisite for an introductory course of Econometrics implemented in R providing the students with an introduction to the statistical language R.

To install R, RStudio in your computer, please watch the videos here; you can also find how to work with *projects* in RStudio.

This document is a work in progress, so feel free to suggest topics and/or improvements.

It is convenient to use different fonts and colors when using `R code`, `specific R libraries` or `name of a variable`, links.

When starting, R loads by default a certain number of libraries; others may be needed later on and will be specified explicitly.

Working with projects

A *project* is a working directory, where we can put all data, files, etc.

It is strongly recommended to work with projects because it makes the workflow much more orderly and streamlined, especially when handling several data sets.

Ideally, a work project (*e.g.*, a semester course) is contained in an RStudio project containing all the needed material.

Setting up a new project

Go to *File* in the menu bar; in the drop-down menu, select *New Project...*, a pop-up window appears; select *New Directory* and then *New Project*. In the field named *Directory name* put a meaningful name for your new project and then select the folder where you want to save your project.

You can also follow the instructions in the video here (last-but-one slide).

Descriptive statistics

Descriptive statistics is the process of summarizing the observed values of a sample for further study.

Also, a *descriptive statistic* (singular) is a number obtained by some formula, or algorithm, describing some features of a sample.

Grosso modo, we can divide descriptive statistics into

- measures of central tendency;
- measures of dispersion.

To illustrate the following concepts, it is convenient to use a sample; for reproducibility, we set the seed of the random generator. We siumulate observed values by a sampling procedure:

```
set.seed(1)
x <- sample(15, 20, replace = TRUE)
x <- c(x, NA)
x
```

```
##  [1]  9  4  7  1  2 13  7 11 14  2 11  3  1  5  5 10  6 14 10  7 NA
```

We also introduced a missing value, `NA`; it will come in handy below

Measures of tendency

Measures of tendency are

- sample mean (or arithmetic mean, depending on the context)
- median
- mode

For highly skewed data, the median is preferred.

Sample mean

The sample mean of a set of observations $x_1, \ldots, x_n$ is

$$\overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

The calculation of the sample mean is implemented by `mean()`; this function is sensitive to the presence of missing values:

```
mean(x)
```

```
## [1] NA
```

```
mean(x, na.rm = TRUE)
```

```
## [1] 7.1
```

See the method, `na.rm=TRUE`, because of the presence of the missing value in the observations.

The sample mean of `x` is $\overline{x} = \text{NA}$.

Median

The median, m, corresponds to the second quartile, that is, the value that leaves 50% of the observations below m and the remaining 50% above it.

The median is implemented by the function `median()`; the function `median()` is also sensitive to the presence of missing values:

```
median(x)
```

```
## [1] NA
```

```
median(x, na.rm = TRUE)
```

```
## [1] 7
```

Mode

The mode is the most frequent value observed.

R does not provide an implementation of the mode, but we can define a function to do that

```
getmode <- function(x){
  a = table(x)
  a[which.max(a)]
}
getmode(x)
```

```
## 7
## 3
```

The output consists of two values: the first one is the mode and the second one is the number of occurrences. In this case, the mode is 7, occurring three times.

If there is more than one mode, this function only returns the first one. We show how to find out whether there is more than one mode in the vector of observed values:

```
remove <- c(7,7,7)
getmode(x[! x %in% remove])
```

```
## 1
## 2
```

We see that there is only one mode in this vector.

If we add hte value 11,

```
y <- c(x, 11)
getmode(y)
```

```
## 7
## 3
```

then we have

```
remove <- c(7,7,7)
getmode(y[! y %in% remove])
```

```
## 11
##  3
```

a second mode, equal to 11.

```
remove2 <- c(7,7,7, 11,11,11)
getmode(y[! y %in% remove2])
```

```
## 1
## 2
```

We see that the vector `y` is *bimodal*, that is, it has two different modes.

Measures of dispersion

Typical measures of dispersions are

- range,
- interquartile range (IQR), and
- standard deviation, with the associated variance.

For ordinal data, the interquartile range is the preferred measure of dispersion.

Range

The range is the difference between the maximum and the minimum of the observed values.

The range is implemented in R with the function `range()`, which outputs the minimum and maximum, as a vector; it is sensitive to the presence of missing values:

```
range(x)
```

```
## [1] NA NA
```

```
range(x, na.rm = TRUE)
```

```
## [1]  1 14
```

The range is then

```
range(x, na.rm = TRUE)[2] - range(x, na.rm = TRUE)[1]
```

```
## [1] 13
```

Interquartile range (IQR)

The observed data is divided into *quartile*:

- Q_1: is the median between the minimum and the median of the full observed data set.
- Q_3: is the median between the median of the full observed data set and the maximum.

$$IQR = Q_3 - Q_1.$$

The interquartile range is implemented by `IQR()`; it is also sensitive to NA's:

```
IQR(x, na.rm = TRUE)
```

```
## [1] 6.5
```

Standard deviation

The (sample) standard deviation is defined as

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

The (sample) variance is the square of the standard deviation

$$var(x) = s^2 = \frac{\sum_{i=1}^{n}(x - \overline{x})^2}{n-1}.$$

The standard deviation is sometimes preferred over the variance because s has the same units of the observed data, while the variance has squared units.

The variance and standard deviation are implemented by `var)`, `sd()`, respectively:

```
sd(x, na.rm = TRUE)
```

```
## [1] 4.253791
```

```
var(x, na.rm = TRUE)
```

```
## [1] 18.09474
```

The `summary()` function

R implements a collection of the above measures of descriptive statistics in `summary()`

```
summary(x)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    1.00    3.75    7.00    7.10   10.25   14.00       1
```

This function does not provide variance or standard deviation, which have to be computed separately.

Plots

Plots can be useful to get a feeling of the observed data.

The most used plots for numerical data are

- histograms;
- boxplots;
- scatterplots;
- Q-Q plots.

It will be convenient now to consider observations taken from the `faithful` data set. THis data set contains data about the duration of eruptions, `eruptions` and the time between eruptions, `waiting` for the Old Faithful Geyser in Yellowston National Park (Wyoming).

For more information, copy and paste the following instruction in the console:

```
help(faithful)
```

The data set contains two variables, `eruptions`, eruption time, in minutes; `waiting`, waiting time to the next eruption (in minutes).

Histograms

A histogram can be used to identify the probability distribution that may follow the data under consideration.

The purpose of a histogram is to visualize the profile of the distribution of the data by means of rectangles whose area is proportional to the frequency of the observations and whose width is equal to the class interval: see, e.g., here.

The histogram can be thought of as an approximation of the data to the population distribution.

To draw a histogram we have to out the observations into "boxes" containing a certain proportion of the data. The implementation in R uses three different algorithms:

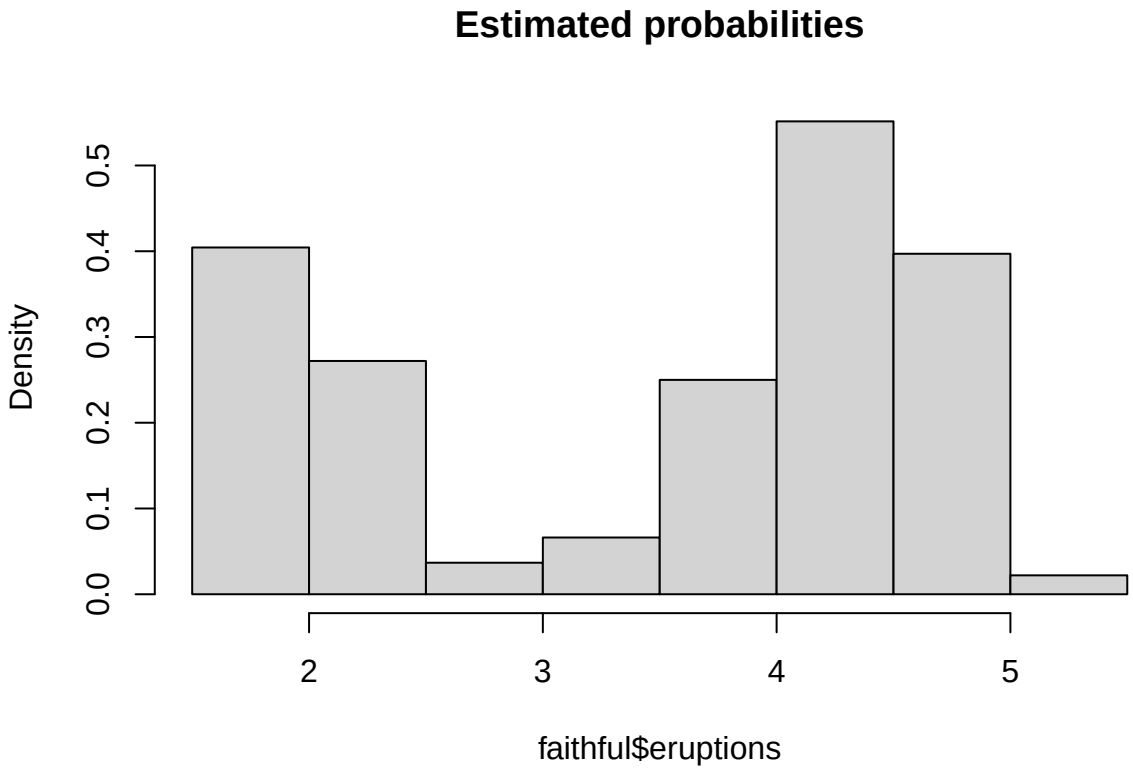
- Sturges's rule;
- Scott's rule, and
- Freedman & Diaconis's rule

See here for a discussion about the relative merits of these rules.

The default implementation in R is the Sturges's rule. To use one of the other two, we use the method `breaks = "Scott"` or `breaks = "FD"`, respectively.

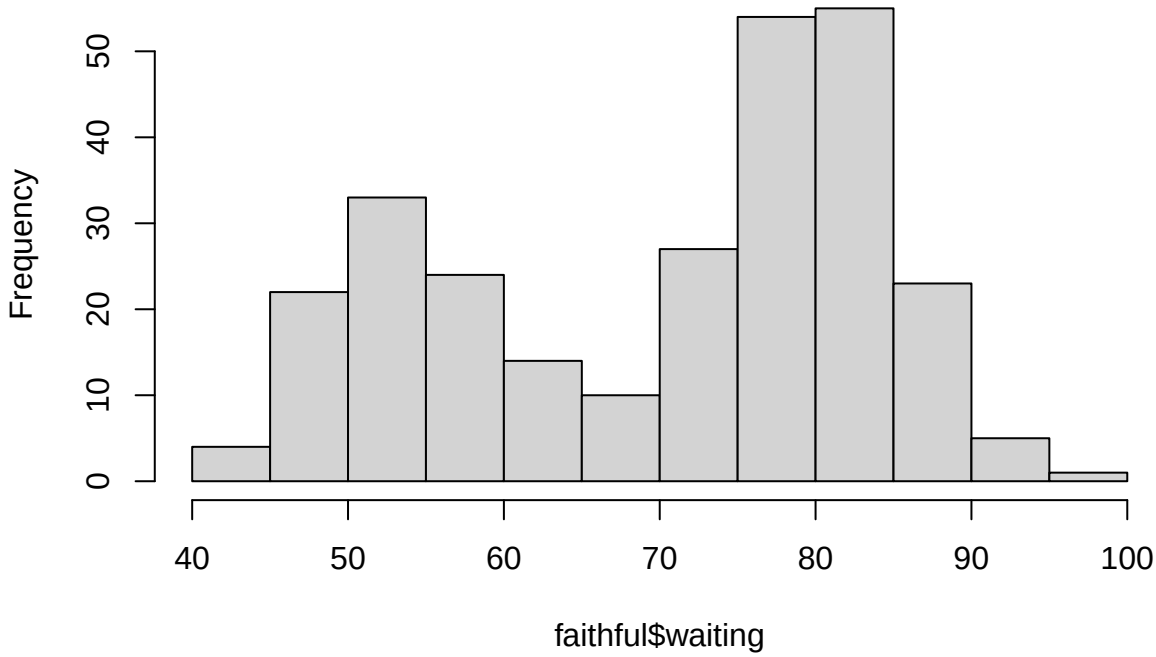
The histograms of the two variables in the data set `faithful` is displayed by **extraction** (direct referencing), using the `$` symbol between the name of the data set and the name of the variable. The method `probability = TRUE` shows the estimated probabilities, instead of the count frequency (the default `probability = FALSE`):

```
hist(faithful$eruptions, probability = TRUE,
     main = "Estimated probabilities")
```



We can see that the most frequent values are located at small values and large values of `eruptions`.

```
hist(faithful$waiting, main = "Observed frequency count")
```



Here, too, we can see that the most frequent values are located at small values and large values of `waiting`.

Boxplot

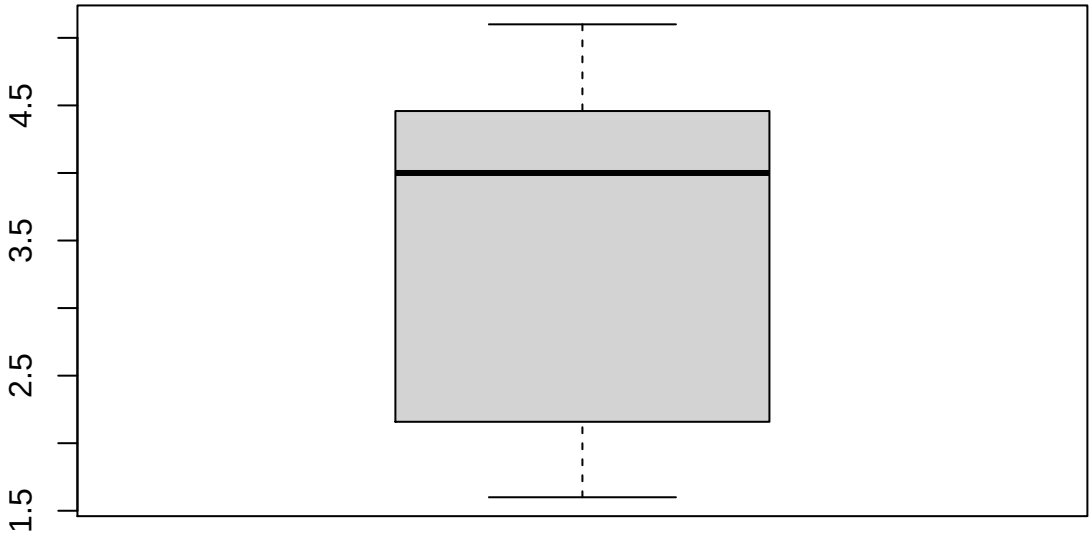
A boxplot (also known as *box and whisker diagram*) is a standardized way of displaying the distribution of data.

The box represent the IQR, while the whiskers correspond either to the maximum (resp., minimum) of the values of the observations.

An empirical definition of *outliers* are observed values with values greater, in absolute value, than $1.5IQR$.

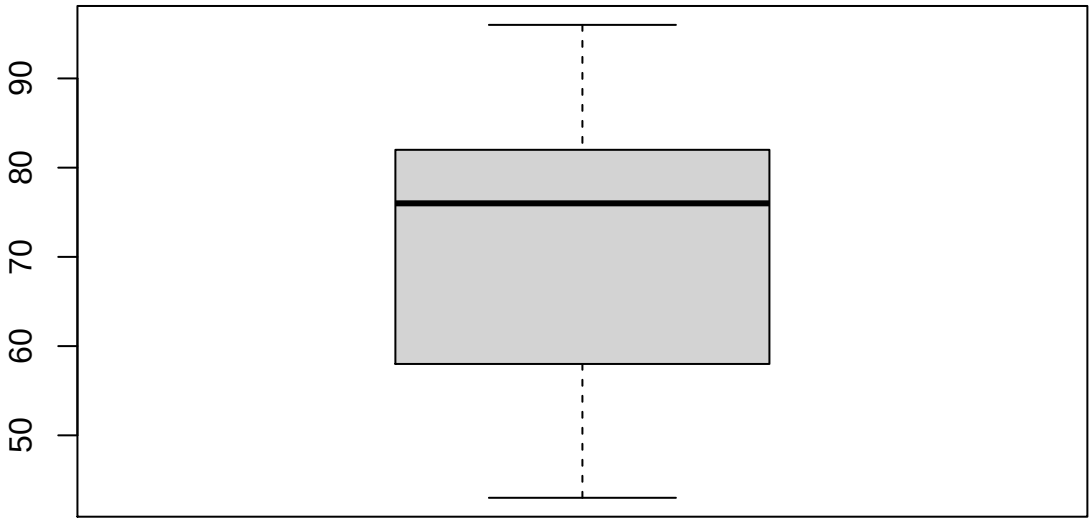
The outliers are presented in a boxplot with circles beyond the whiskers.

```
boxplot(faithful$eruptions)
```



The black solid line in the box represents the median of the variable `eruptions`. We can see that the there the variable is somehow not symmetric.

```
boxplot(faithful$waiting)
```



Neither variable contains outliers.

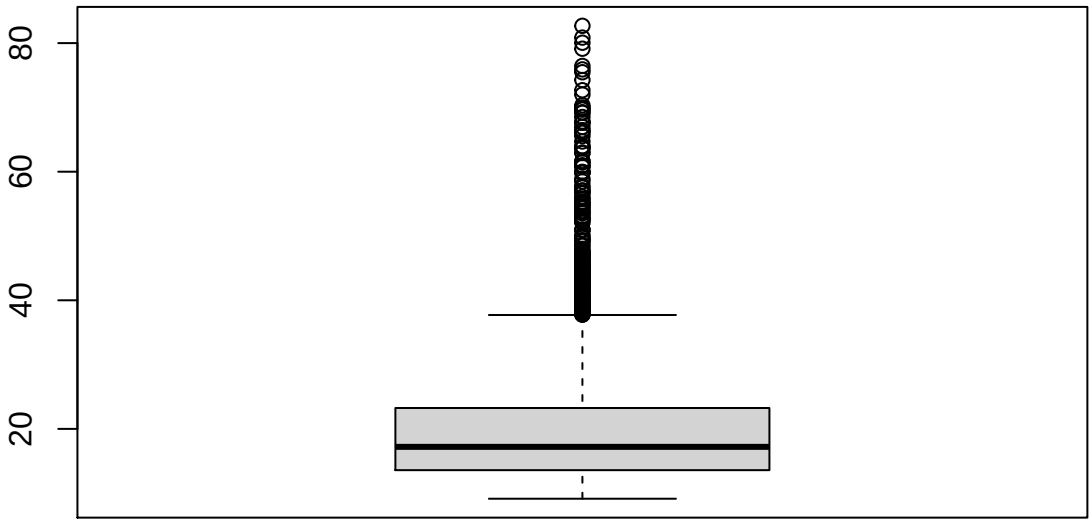
Boxplots with outliers

In the following example, we use the library `library(quantmod)` to download VIX data, extract the closing price and draw a boxplot

```
library(quantmod)
getSymbols("^VIX")
```

```
## [1] "^VIX"
```

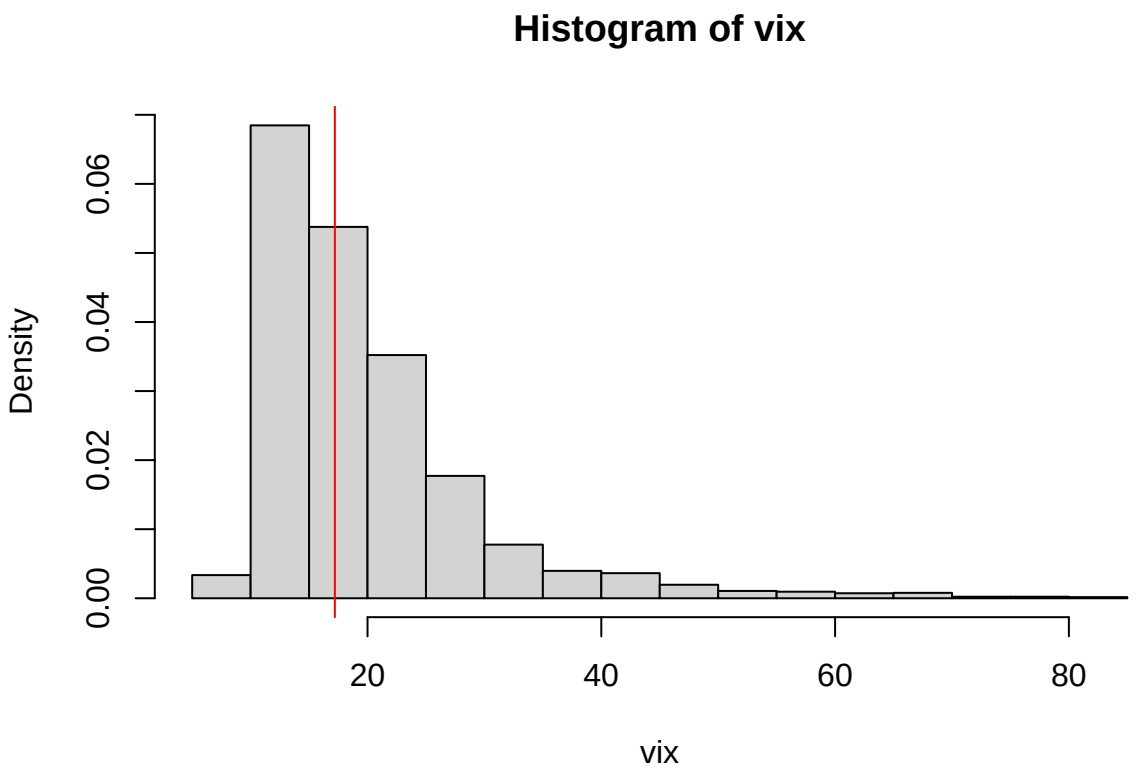
```
vix <- VIX$VIX.Close
boxplot(vix)
```



Here we can see that the closing price has a long tail to the right, but 50% of the observed values (corresponding to the median) are close to the minimum. Also, the long right tail has values beyond the threshold $1.5IQR$ that identifies empirically the outliers.

We can use also a histogram to check this out, drawing additionally a vertical red line corresponding to the median value:

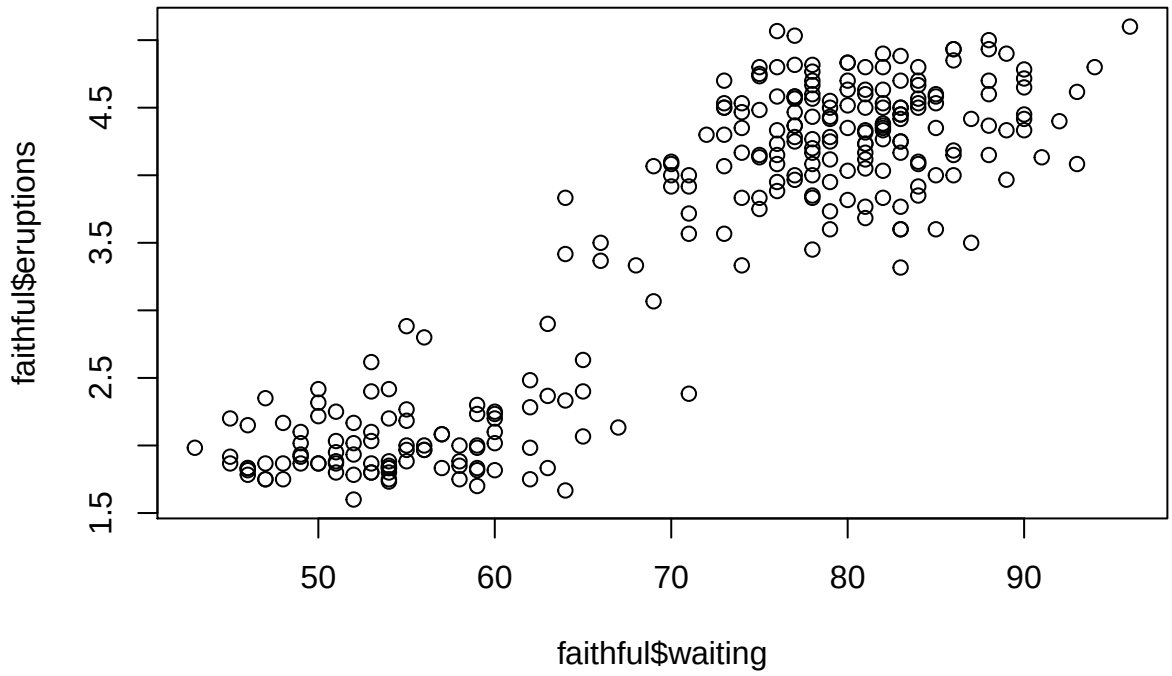
```
hist(vix, probability = TRUE)
abline(v = median(vix), col = "red")
```



Scatterplots

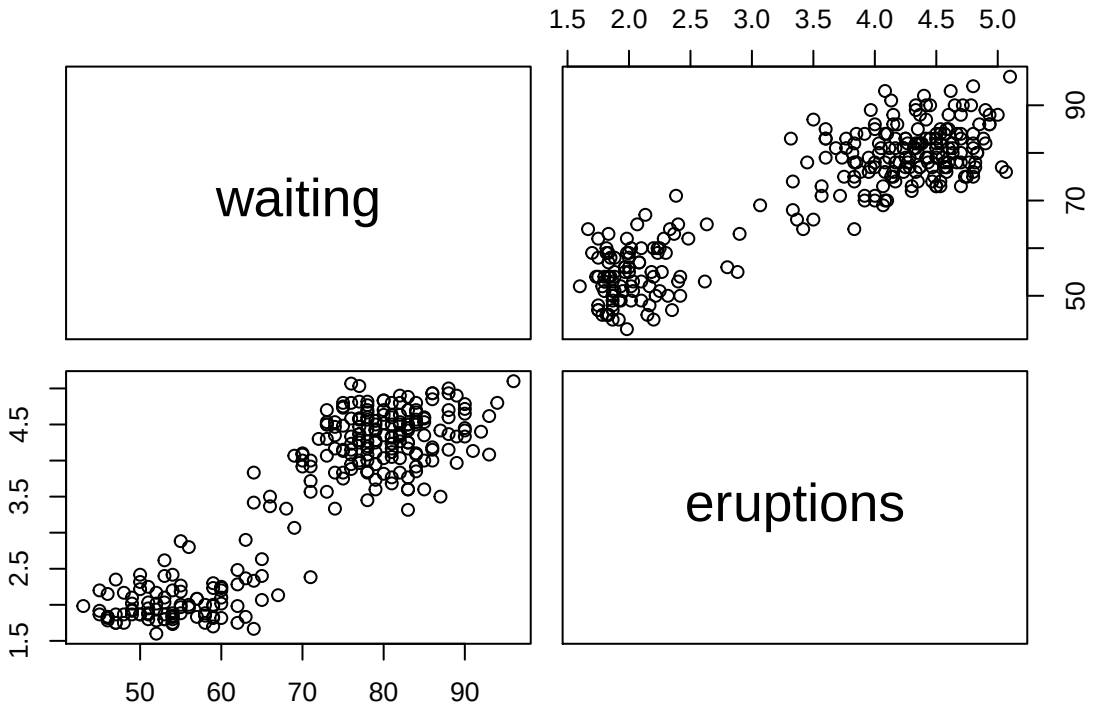
A scatterplot allows to sketch one variable against another one. It is implemented in R by the `plot()` function

```
plot(faithful$waiting, faithful$eruptions)
```



We can also sketch a scatterplot matrix, using the `pairs()` function

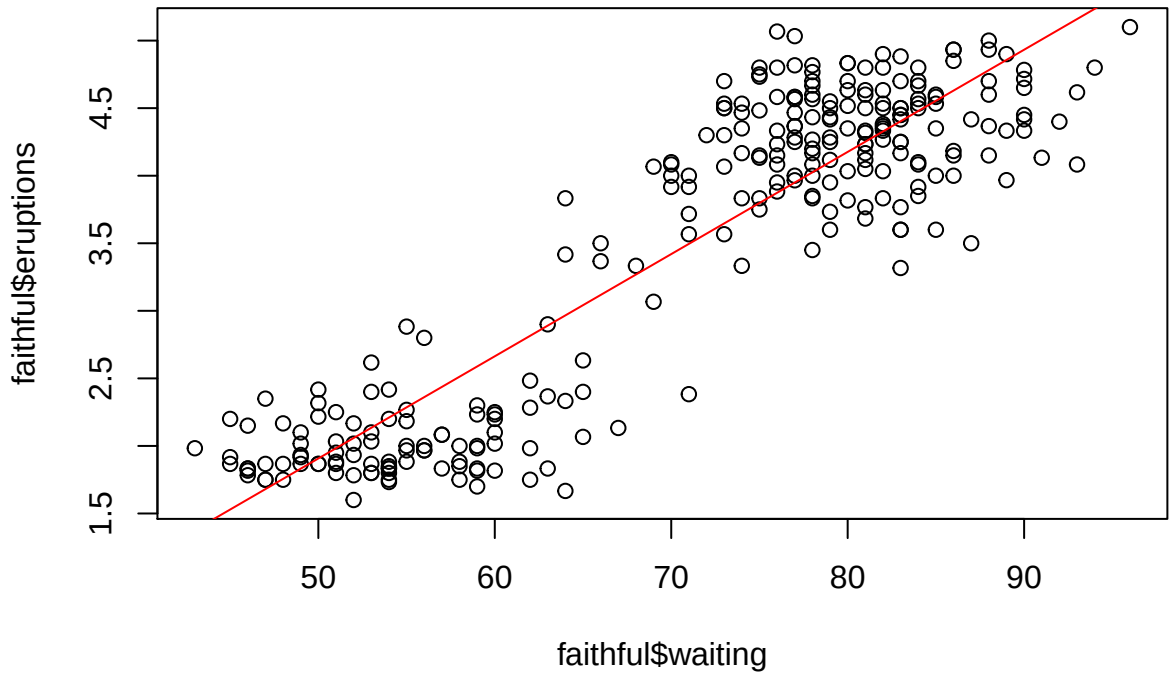
```
pairs(~ waiting + eruptions, data=faithful)
```



Either way, we see easily that there is a certain trend between the two variables.

We can also add the function `abline()` to `plot()` to include a ternd line, using the `lm()` function

```
plot(faithful$waiting, faithful$eruptions)
abline(lm(eruptions ~ waiting, data = faithful),
       col = "red")
```



There are many more options of scatterplots with additional libraries.

Q-Q plots

A Q-Q plot (quantile-quantile plot) allows us to assess graphically whether two sets of values have the same (probability) distribution.

We plot the quantiles of the two data sets against one another. If the plotted quantiles align along a straight line, then we can conclude that the two data sets have the same distribution.

In the applications, we draw the quantiles of an empirical distribution against the quantile of a theoretical distribution (*e.g.*, a normal distribution).

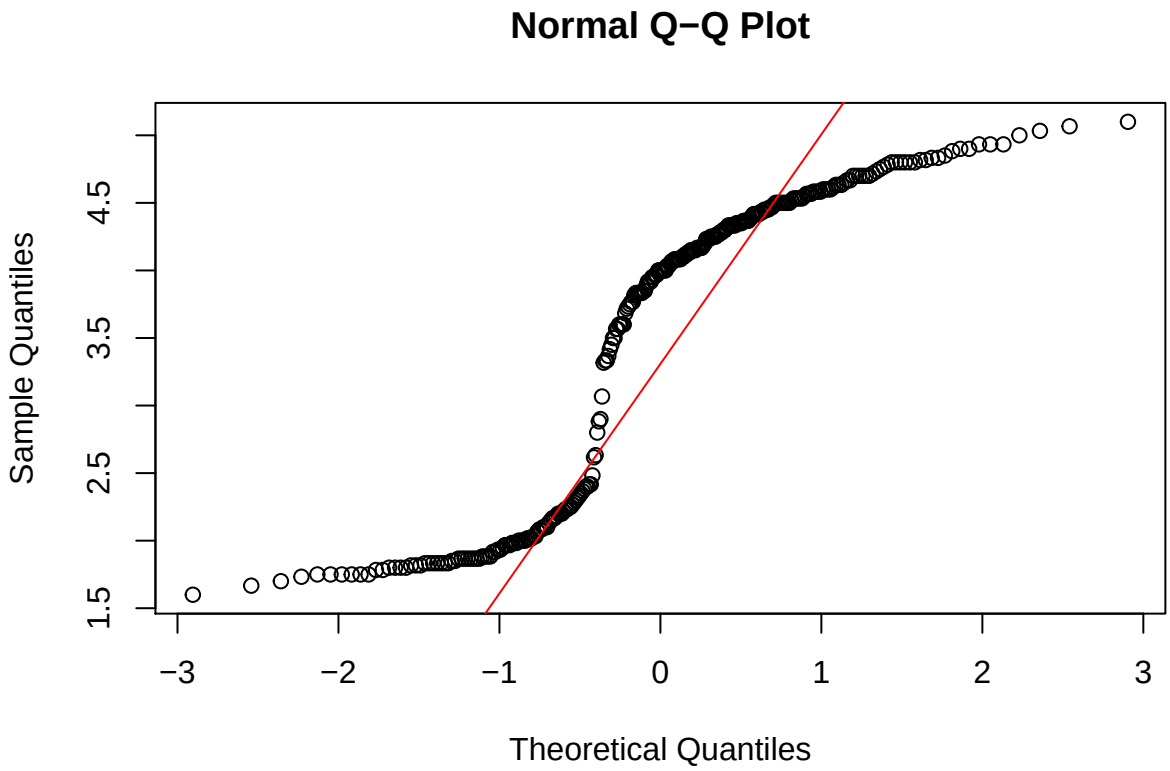
The Q-Q plot is a visual tool, not a formal proof, but it can show us whether the assumption that the observed data follow a theoretical distribution is plausible.

In R there are two different functions that implement a Q-Q plot:

- `qqnorm()` that checks whether the empirical distribution is a normal standard distribution;
- `qqplot()` for other theoretical distributions.

For example, to check whether the empirical distribution of the variable `eruptions` follows a standard normal distribution; we add `qqline()` to identify graphically where the lack of normality may occur:

```
qqnorm(faithful$eruptions)
qqline(faithful$eruptions, col = "red")
```

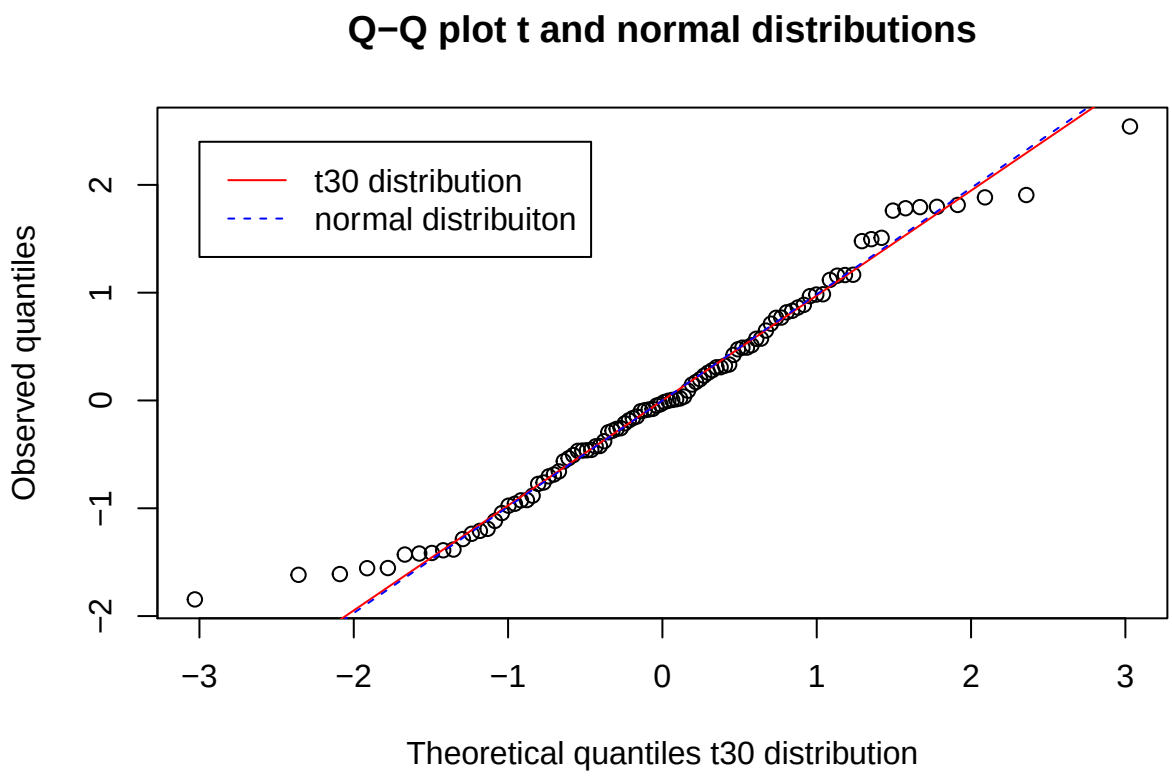


This gives enough visual evidence to conclude that the empirical distribution of `eruptions` is not normal.

As a second example, we simulate 100 points drawn from the t_{10} distribution with $df = 10$ degrees of freedom and draw a Q-Q plot against a t_{30} distribution ($df = 30$). We also add a normal line, `qqline()` to compare the theoretical t_{30} distirbution with the normal distribution:

```
t.obs <- rt(100, df = 10)
qqplot(qt(ppoints(200), df=30), t.obs,
       main = "Q-Q plot t and normal distributions",
```

```
       xlab = "Theoretical quantiles t30 distribution",
       ylab = "Observed quantiles")
qqline(t.obs,
       distribution = function(p) qt(p, df = 30),
       col = "red")
qqline(t.obs, col = "blue", lty = 2)
legend(-3,2.4, legend = c("t30 distribution", "normal distribuiton"),
       col = c("red", "blue"), lty = 1:2)
```



Here we can see that the two t distributions differ at the tails. The two lines, on the other hand, show us that the normal and t distributions do not differ much, provided that df is large (typically, $df \geq 30$).

Statistical inference

Statistical inference is the dataa-driven process that allows to draw inference (information) about population parameters and distributions from sample from that same population.

One of the basis of statistical inference is the concept of *random sample*.

If we take one sample from a given population, that sample represents a certain measurement with a certain degree of randomness. We represent mathematically that measurement with a random variable.

If we sample n times from that population in a random fashion, we have then a collection of n independent, identically distributed random variables (iid random variables).

So we can say that a collection of iid random variables is a random sample from a population; the statistical properties of that population are represented by the properties of the underlying probability distribution.

Sampling process

Let us explain the mechanics of a sampling procedure with a simple experiment.

Consider the roll of a fair, standard, die. Let X be the number of dots that show up in the upper face.

Being the die fair, we can suppose that probability distribution of X is uniform on the possible values $\{1, \ldots, 6\}$.

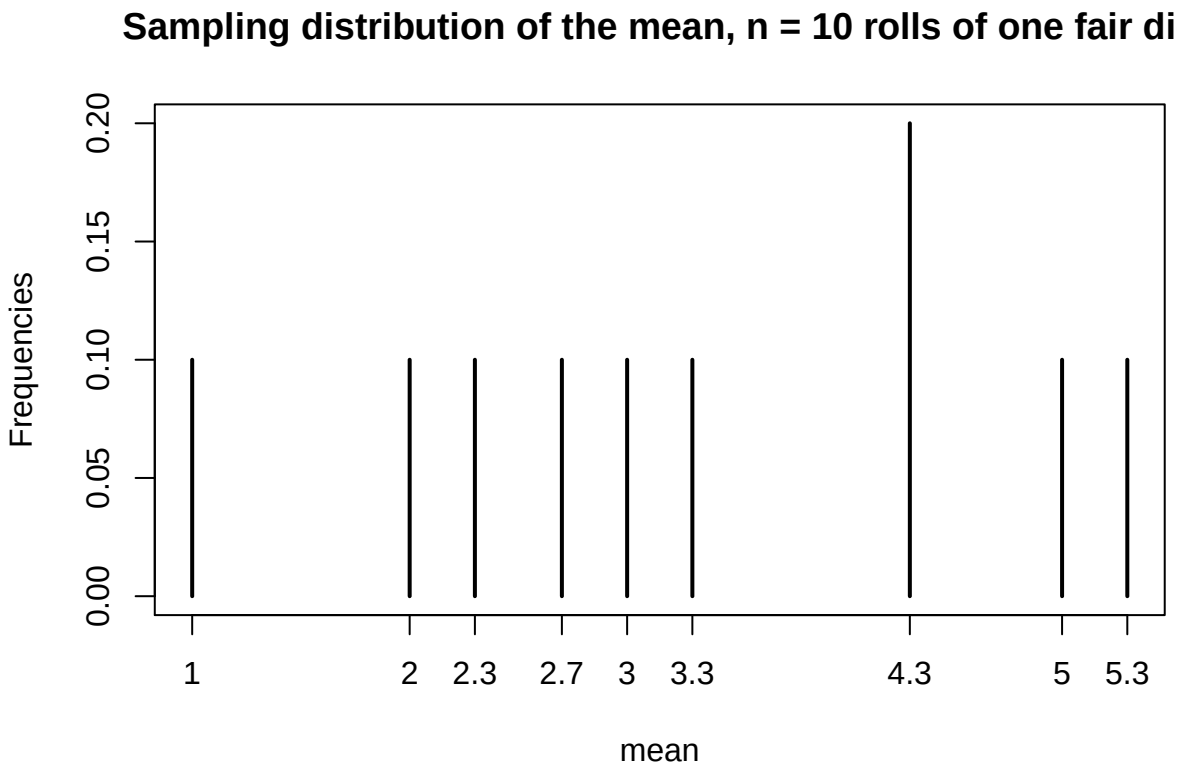
If we roll the die 5 times, we obtain 5 of the possible values, not necessarily with a specific pattern:

```
x <- replicate(5, sample(1:6, size = 1, replace = T))
x
```

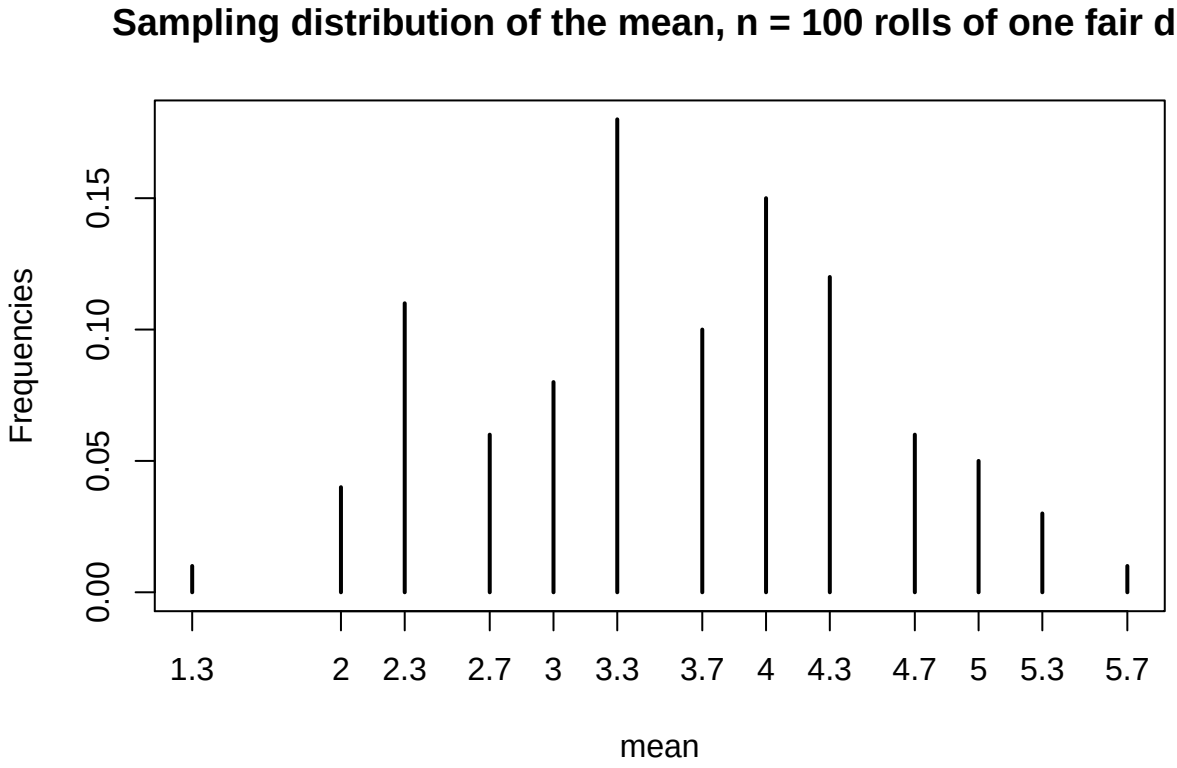
```
## [1] 3 1 5 3 5
```

If we roll the same die repeatedly, say 10, 100, 1000 times, a certain pattern starts to emerge about the *sample mean*:

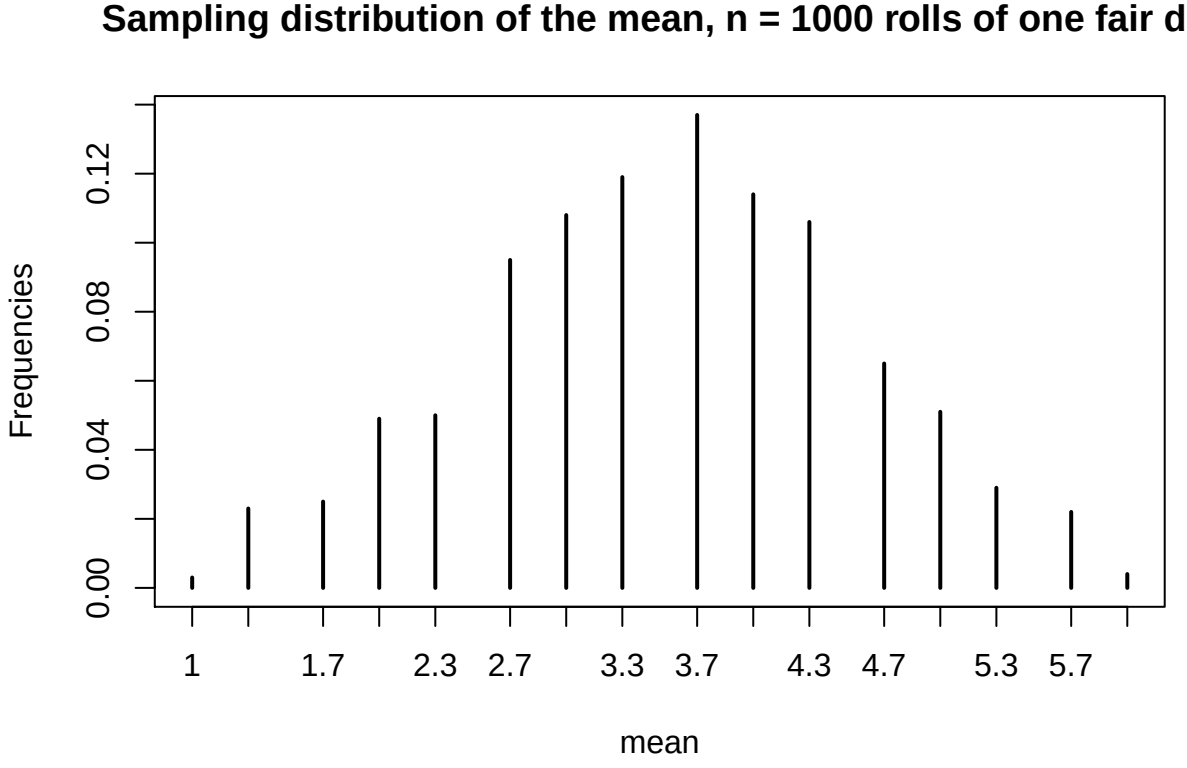
```
die.mean <- function(){
  rolls <- sample(1:6, size = 3, replace=TRUE)
  return(mean(rolls))
}
mc10 <- round(replicate(10, die.mean()), 1)
plot(table(mc10)/length(mc10), xlab = "mean", ylab = "Frequencies",
     main = "Sampling distribution of the mean, n = 10 rolls of one fair die")
```




```
mc100 <- round(replicate(100, die.mean()), 1)
plot(table(mc100)/length(mc100), xlab = "mean", ylab = "Frequencies",
     main = "Sampling distribution of the mean, n = 100 rolls of one fair die")
```




```
mc1000 <- round(replicate(1000, die.mean()), 1)
plot(table(mc1000)/length(mc1000), xlab = "mean", ylab = "Frequencies",
     main = "Sampling distribution of the mean, n = 1000 rolls of one fair die")
```



Sampling distribution

A *statistic* is a function of the observable random variables in a sample (and known constants).

A statistic may be the mean of the repeated rolls of a die, for example. Being a function of the sample, a statistic is a random variable. Its probability distribution is called the *sampling distribution*.

If $f(x)$ is the probability density function (probability mass function, for discrete random variables) of the population, the sampling distribution of a random sample $X_1, \ldots, X_n$, can be written as the multivariate probability distribution

$$f(x_1, \ldots, x_n) = f(x_1) \cdots f(x_n).$$

For a given, finite, sample size n, the properties of the sampling distribution can be fairly complicated.

Consider, for instance, one roll of *three* fair dice. Its sample space is

$$\{1, 2, 3, 4, 5, 6\}^3$$

having with uniform distribution. Even though it is a finite space, with $6^3 = 216$ points (triplets), the complete description of the associated probability distribution is doable, but lengthy.

Sample mean and its properties

If $X_1, \ldots, X_n$ is a random sample from a population with mean μ and variance σ^2, then the sample mean

$$\overline{X} = \frac{1}{n} \sum_{i=1}^{n} X_i.$$

Being a function of the random sample, $\overline{X}$ is a random variable and we can compute its mean and variance:

$$E[\overline{X}] = \mu$$
$$var(\overline{X}) = \frac{\sigma^2}{n}$$

If we let the sample size $n \to \infty$, on the other hand, we can have a fairly simple description of the sampling distribution of the sample mean using the Central Limit Theorem.

Central Limit Theorem (CLT)

Let X_1, ..., X_n be independent and identically distributed random variables, each of which with mean μ and variance σ^2. Let, also,

$$Z_n = \frac{\overline{X} - \mu}{\sigma/\sqrt{n}};$$

then the limiting distribution of Z_n, as $n \to n$, is standard normal distribution[1].

Remarks

- Z_n is the *standardization* of $\overline{X}$.
- The conclusion of CLT is sometimes replaced in the applications by

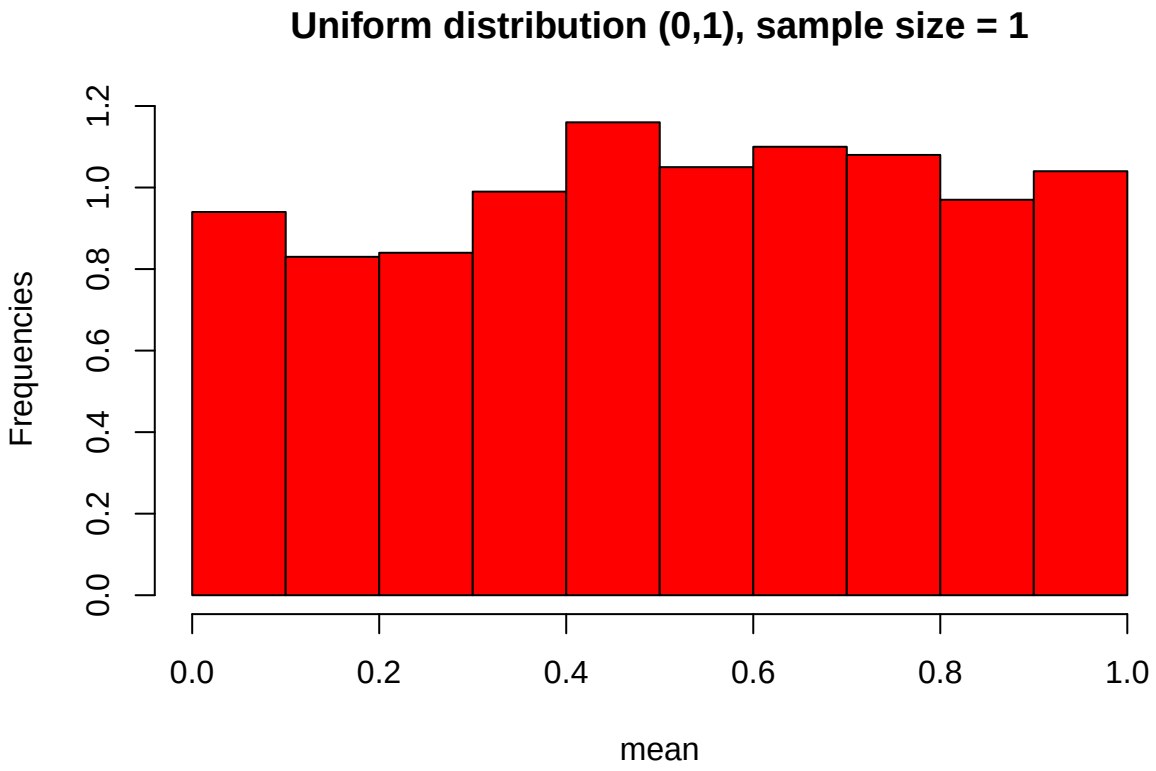
  $\overline{X}$ is asymptotically normally distributed, with mean μ and variance σ^2/n.
- The CLT can be applied to a random sample X_1, ..., X_n from any distribution as long as $E[X_i] = \mu$, $var(X_i) = \sigma^2$ are both finite and n is large.
- We present simulations of the conclusion of CLT, with samples drawn from a variety of distributions. The visualization is done by using histograms.

[1] More precisely, the distribution function of Z_n converges to the standard normal distribution function, as $n \to \infty$.

Simulations of CLT

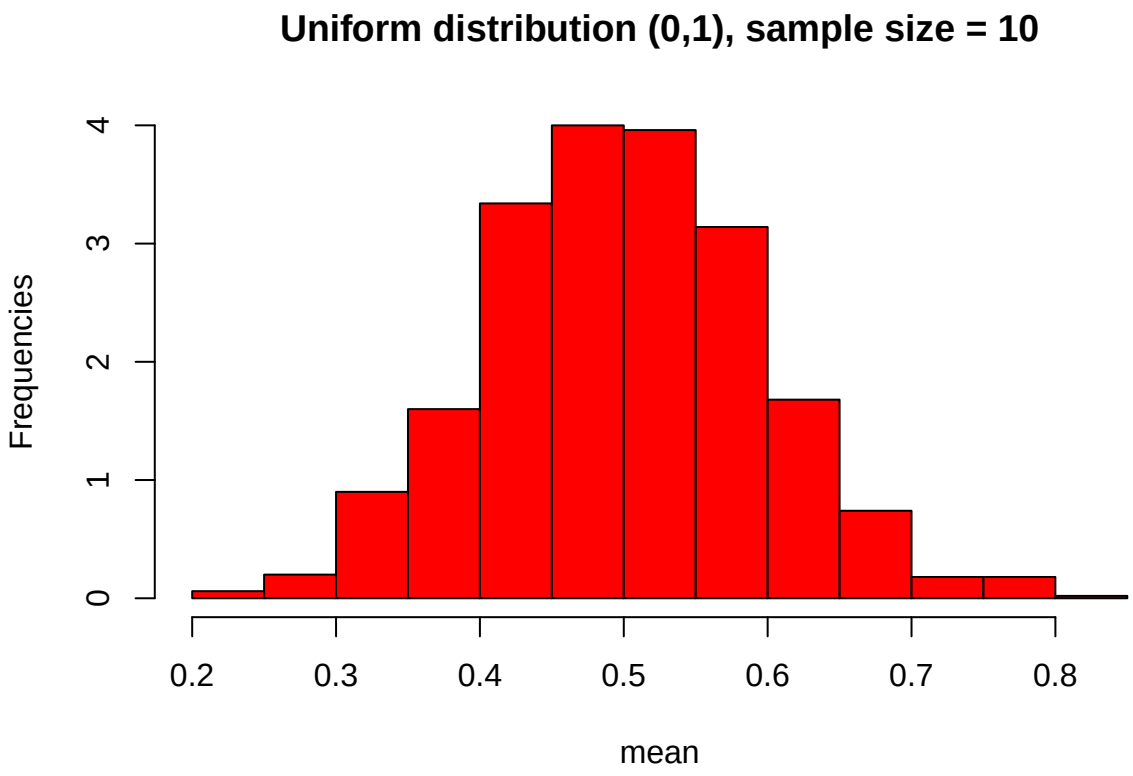
Uniform distribution: $n = 1$

```
clt1=numeric(1000)
for (i in 1:1000){x=runif(1);clt1[i]=mean(x)}
hist(clt1,col="red", xlab = "mean", ylab = "Frequencies",
     main="Uniform distribution (0,1), sample size = 1",
     freq = FALSE)
```



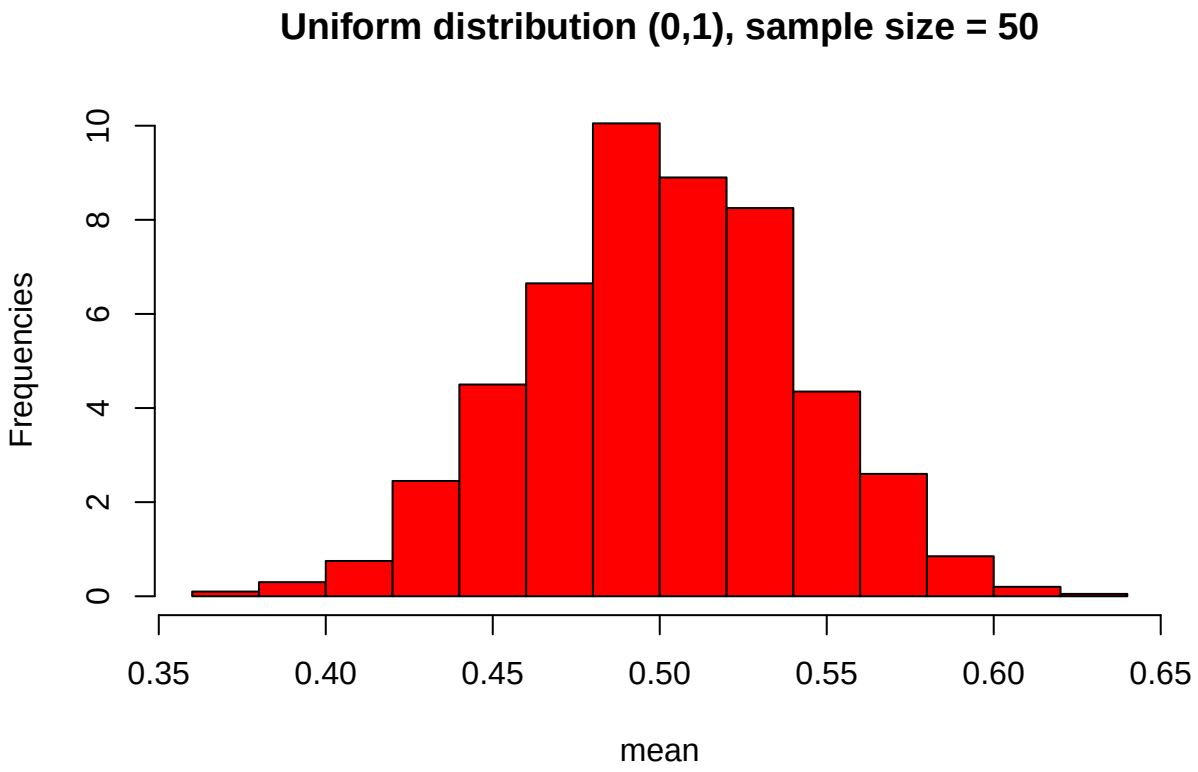
Uniform distribution: $n = 10$

```
clt10=numeric(1000)
for (i in 1:1000){x=runif(10);clt10[i]=mean(x)}
hist(clt10,col="red", xlab = "mean", ylab = "Frequencies",
     main="Uniform distribution (0,1), sample size = 10",
     freq = FALSE)
```



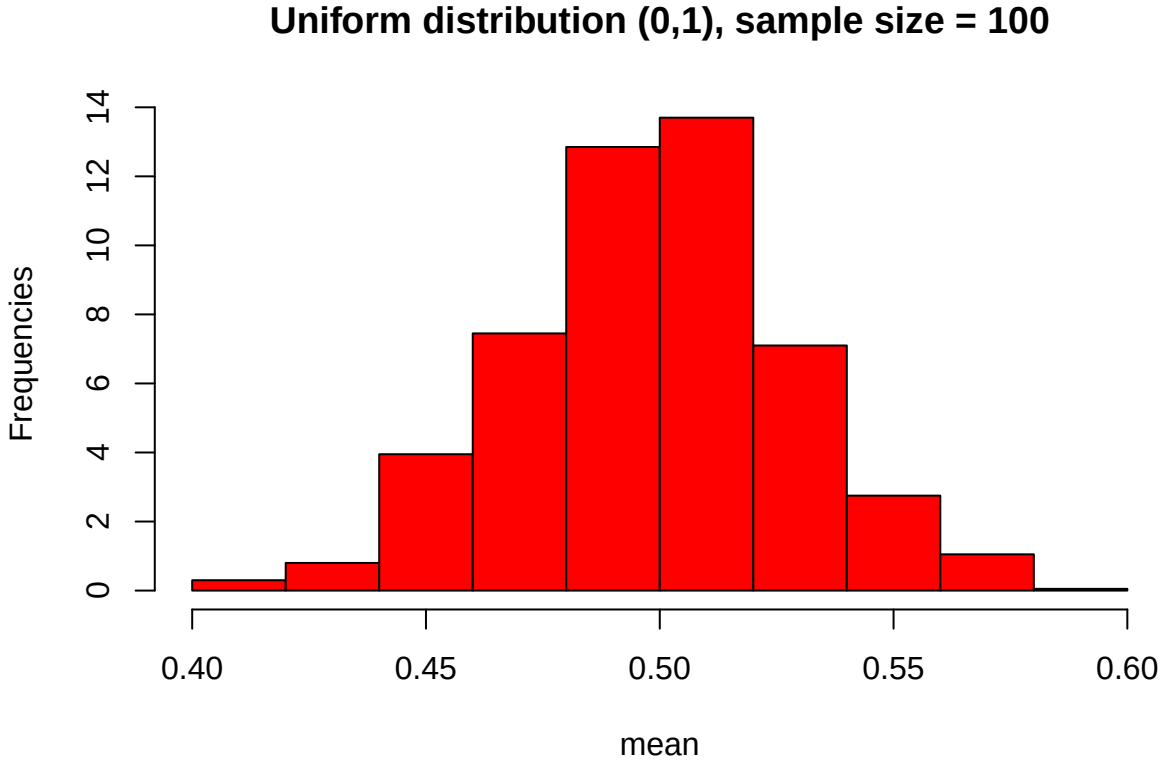
Uniform distribution: $n = 50$

```
clt100=numeric(1000)
for (i in 1:1000){x=runif(50);clt100[i]=mean(x)}
hist(clt100,col="red", xlab = "mean", ylab = "Frequencies",
     main="Uniform distribution (0,1), sample size = 50",
     freq = FALSE)
```



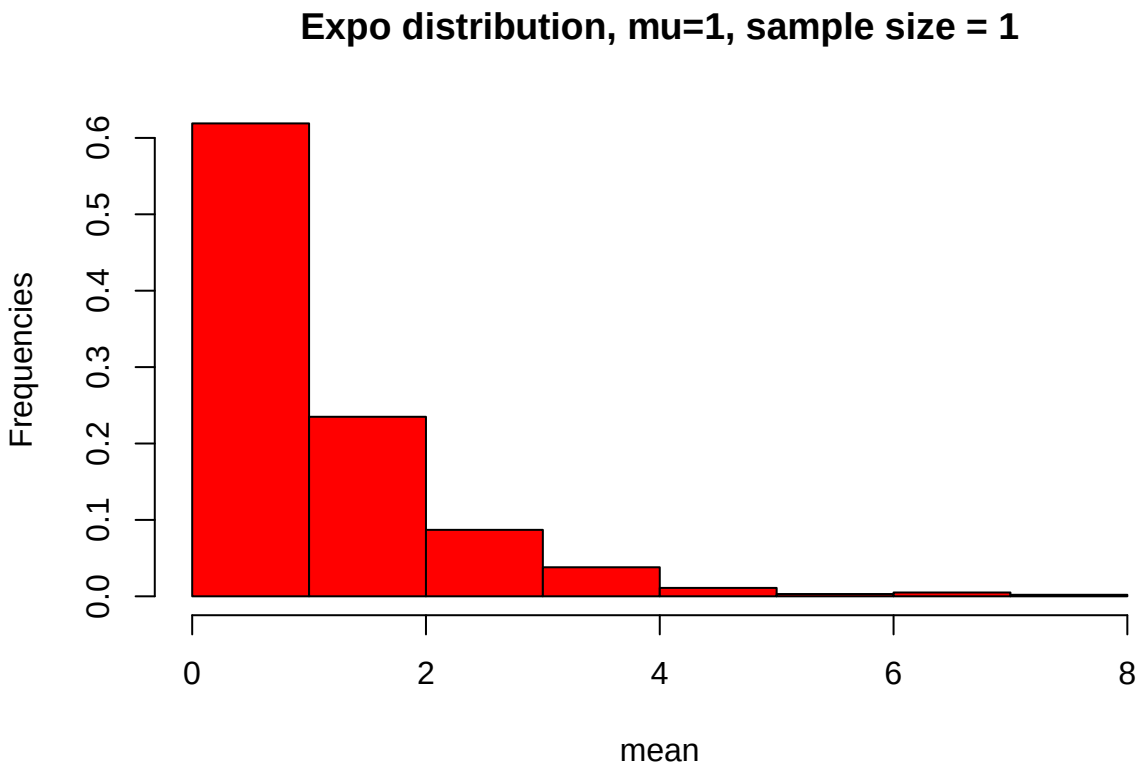
Uniform distribution: $n = 100$

```
clt1000=numeric(1000)
for (i in 1:1000){x=runif(100);clt1000[i]=mean(x)}
hist(clt1000,col="red", xlab = "mean", ylab = "Frequencies",
     main="Uniform distribution (0,1), sample size = 100", freq = FALSE)
```



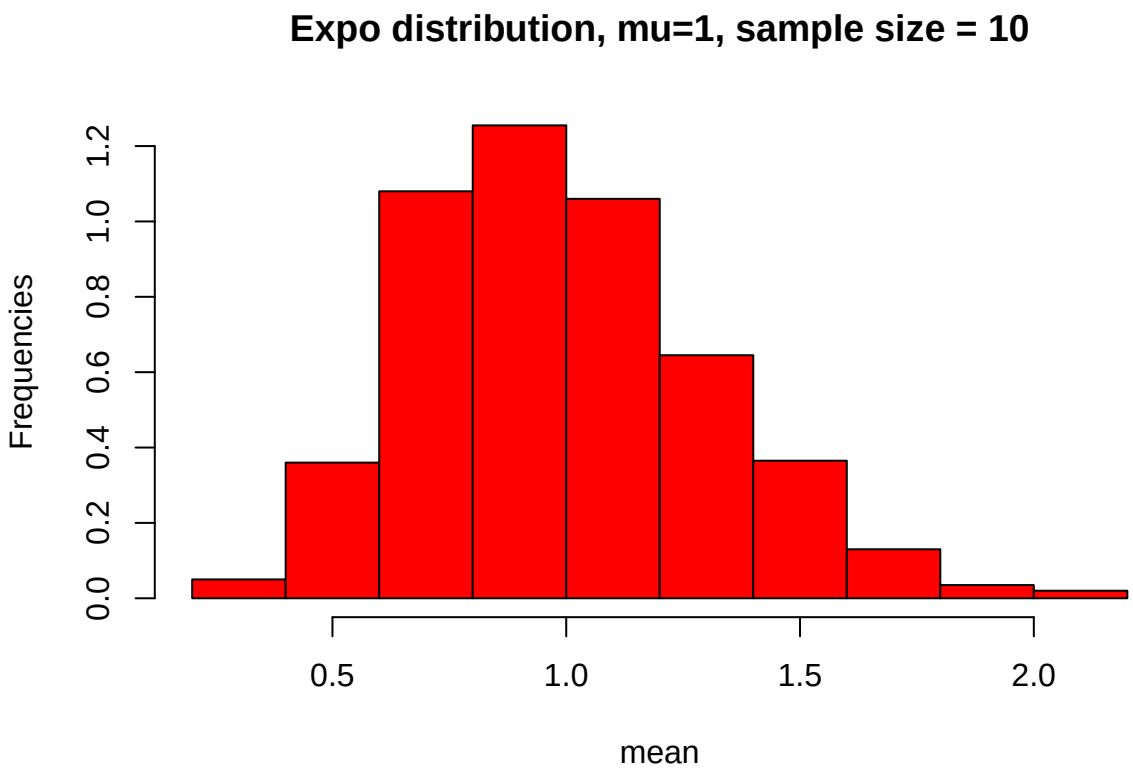
Exponential distribution: $n = 1$

```
clt1=numeric(1000)
for (i in 1:1000){x=rexp(1);clt1[i]=mean(x)}
hist(clt1,col="red", xlab = "mean", ylab = "Frequencies",
     main="Expo distribution, mu=1, sample size = 1", freq = FALSE)
```



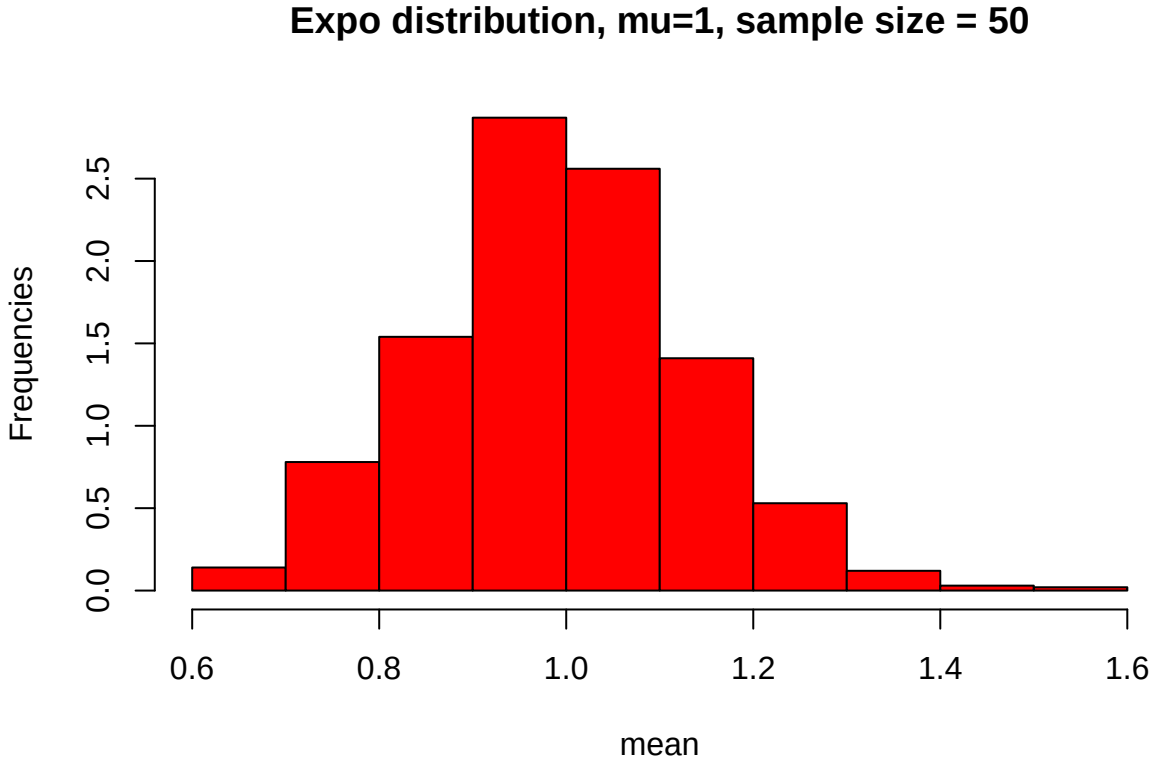
Exponential distribution: $n = 10$

```
clt10=numeric(1000)
for (i in 1:1000){x=rexp(10);clt10[i]=mean(x)}
hist(clt10,col="red", xlab = "mean", ylab = "Frequencies",
     main="Expo distribution, mu=1, sample size = 10", freq = FALSE)
```



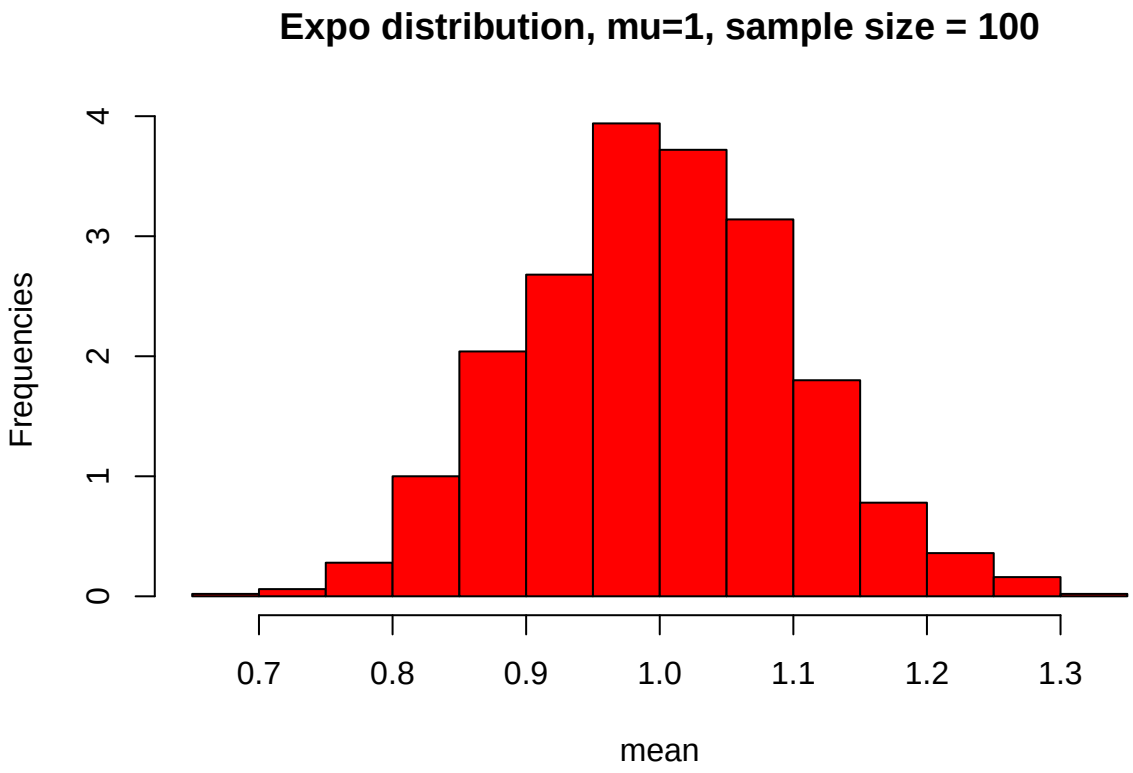
Exponential distribution: $n = 50$

```
clt100=numeric(1000)
for (i in 1:1000){x=rexp(50);clt100[i]=mean(x)}
hist(clt100,col="red", xlab = "mean", ylab = "Frequencies",
     main="Expo distribution, mu=1, sample size = 50", freq = FALSE)
```



Exponential distribution: $n = 100$

```
clt1000=numeric(1000)
for (i in 1:1000){x=rexp(100);clt1000[i]=mean(x)}
hist(clt1000,col="red", xlab = "mean", ylab = "Frequencies",
     main="Expo distribution, mu=1, sample size = 100", freq = FALSE)
```



Binomial distribution

The case of this distribution is specia, as we show below.

- In the case of the binomial distribution, the normal approximation works well as long as p is not close to zero or one.
- A usefult rule of thumb, is that the normal approximation is appropriate when

$$0 < p - 3\sqrt{p(1-p)/n} < p + 3\sqrt{p(1-p)/n} < 1$$

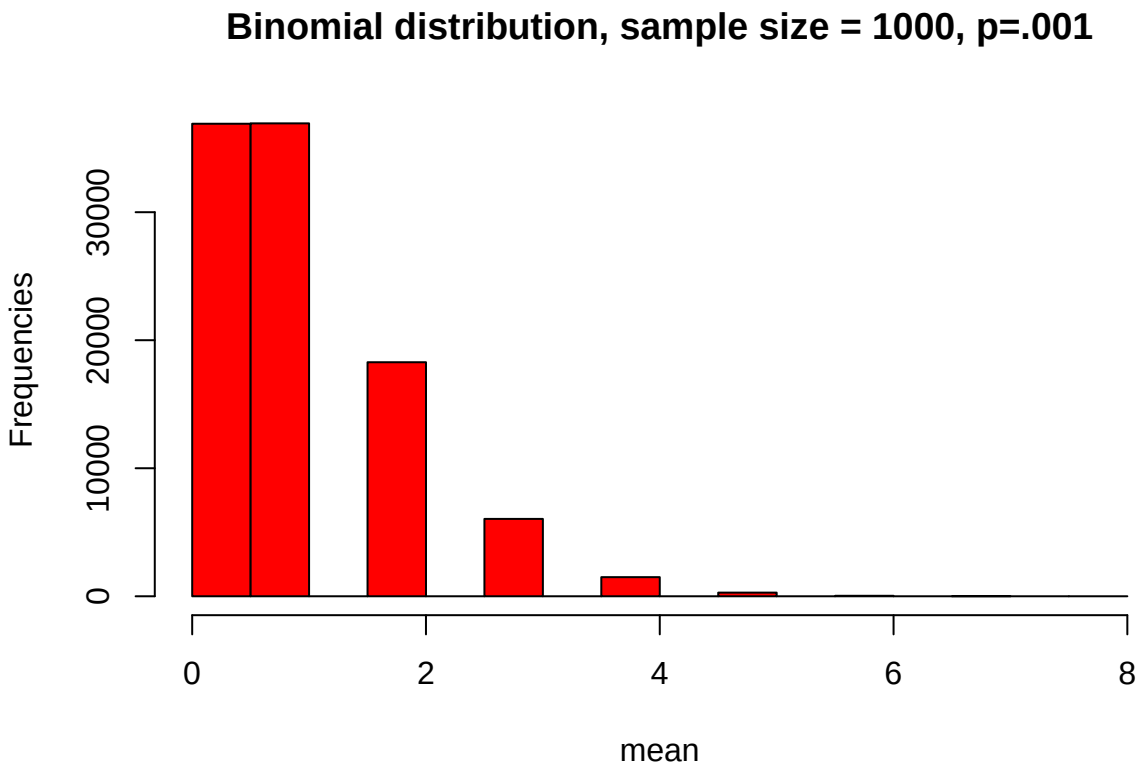
- In terms of sample size n, the normal approximation is appropriate when

$$n > 9\left(\frac{\text{larger of } p \text{ and } (1-p)}{\text{smaller of } p \text{ and } (1-p)}\right)$$

Here we show the failure of the normal approximation, when $n = 1000$, $p = .001$, $1 - p = .999$, so that

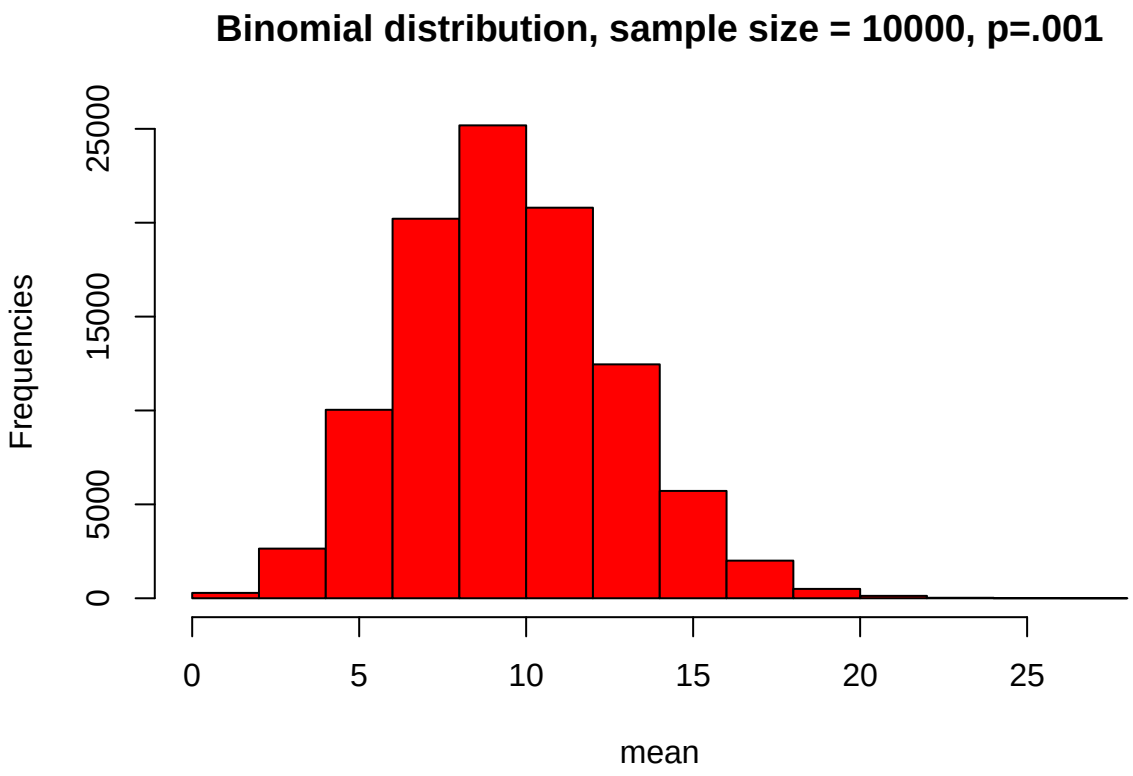
$$n < 9\left(\frac{\text{larger of } p \text{ and } (1-p)}{\text{smaller of } p \text{ and } (1-p)}\right) = 9(999) = 8,991$$

```
cltbinom1 <- numeric(100000)
for (i in 1:100000){x=rbinom(1, 1000, 0.001);cltbinom1[i]=mean(x)}
hist(cltbinom1,col = "red" , xlab = "mean", ylab = "Frequencies",
     main="Binomial distribution, sample size = 1000, p=.001")
```



If, on the other hand, the sample size $n > 8991$, then the normal approximation works fine. Below we use $n = 10,000$:

```
cltbinom1 <- numeric(100000)
for (i in 1:100000){x=rbinom(1, 10000, 0.001);cltbinom1[i]=mean(x)}
hist(cltbinom1,col = "red" , xlab = "mean", ylab = "Frequencies",
     main="Binomial distribution, sample size = 10000, p=.001")
```



Hypothesis test of a mean

It is useful to know how to prove a statistical hypothesis about a mean μ in the applications.

The function to use is `t.test()` that can also be used to prove hypotheses about difference of means.

Suppose we want to prove that the eruptions at Old Faithful Geyser lasts more than 3 minutes.

We write the hypothesis test

$$H_0 : \mu = 3$$
$$H_1 : \mu > 3$$

Using $\alpha = .05$, the test is implemented as

```
t.test(faithful$eruptions, mu = 3, alternative = "greater")
```

```
##
##  One Sample t-test
##
## data:  faithful$eruptions
## t = 7.0483, df = 271, p-value = 7.509e-12
## alternative hypothesis: true mean is greater than 3
## 95 percent confidence interval:
##  3.373559      Inf
## sample estimates:
## mean of x
##  3.487783
```

We reject H_0 and conclude that, on average, the durations of the eruptions are greater than 3 minutes.

Power of the test

Type I error

The level of significance is the probability of committing a Type I error

$$\text{Type I error } = \text{Reject } H_0 \text{ when } H_0 \text{ is true.}$$

with

$$\alpha = \Pr(\text{Type I error .})$$

A Type I error is also known as *false positive*. If $\alpha = .05$, we are willing to have 5% of false positives in our tests or, in other words, that we are willing to accept a 5% chance to be wrong when H_0 is true.

Type II error

We can reduce the false positives by reducing the level of significance, say, from .05 to .01. However, this reduction may cause an increase of a Type II error (also known as *false negative*:

$$\text{Type II error } = \text{Accept } H_0 \text{ when } H_0 \text{ is false.}$$

The probability of a type II error

$$\beta = \Pr(\text{Type II error }).$$

The *power* of the test is

$$\text{Power of the test } = 1 - \beta = \Pr(\text{Reject } H_0 \text{ when } H_0 \text{ is false}).$$

To compute the power of a test, we need to consider a specific alternative. For example, to compute the power of the test on the mean time of eruptions at Old Faithful, we consider

$$\mu_1 = 3.7$$

An implementation of the power test is given by `power.t.test()`. We have to specify

- `delta`, the difference between the value of $\mu = 3$ (under H_0) and $\mu_1 = 3.7$;
- the standard deviation;
- the sample size;
- the significance level of the test;
- the rejection area (one sided or two sided).

```
n <- length(faithful$eruptions)
s <- sd(faithful$eruptions)
power.t.test(delta=.7, n, sd = s,
             sig.level = .05, alternative = "one.sided")
```

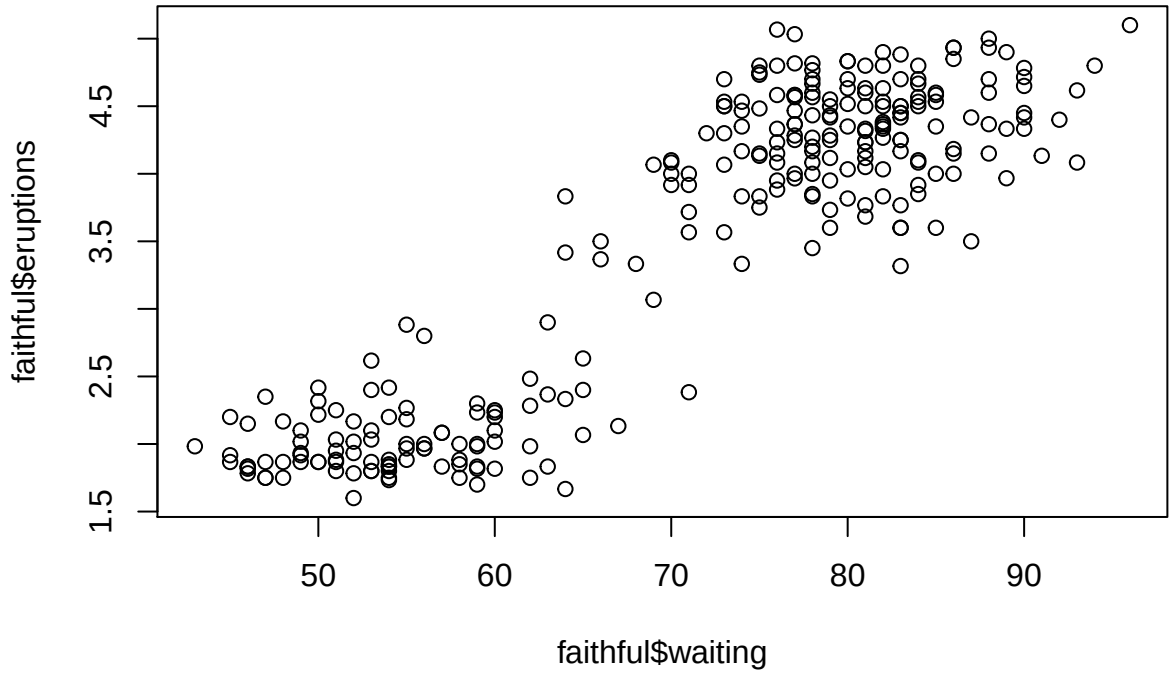
```
##
##      Two-sample t test power calculation
##
##               n = 272
##           delta = 0.7
##              sd = 1.141371
##       sig.level = 0.05
##           power = 1
##     alternative = one.sided
##
## NOTE: n is number in *each* group
```

With $\mu_1 = 3.7$, we are certain to reject H_0 when it is false.

Linear regression

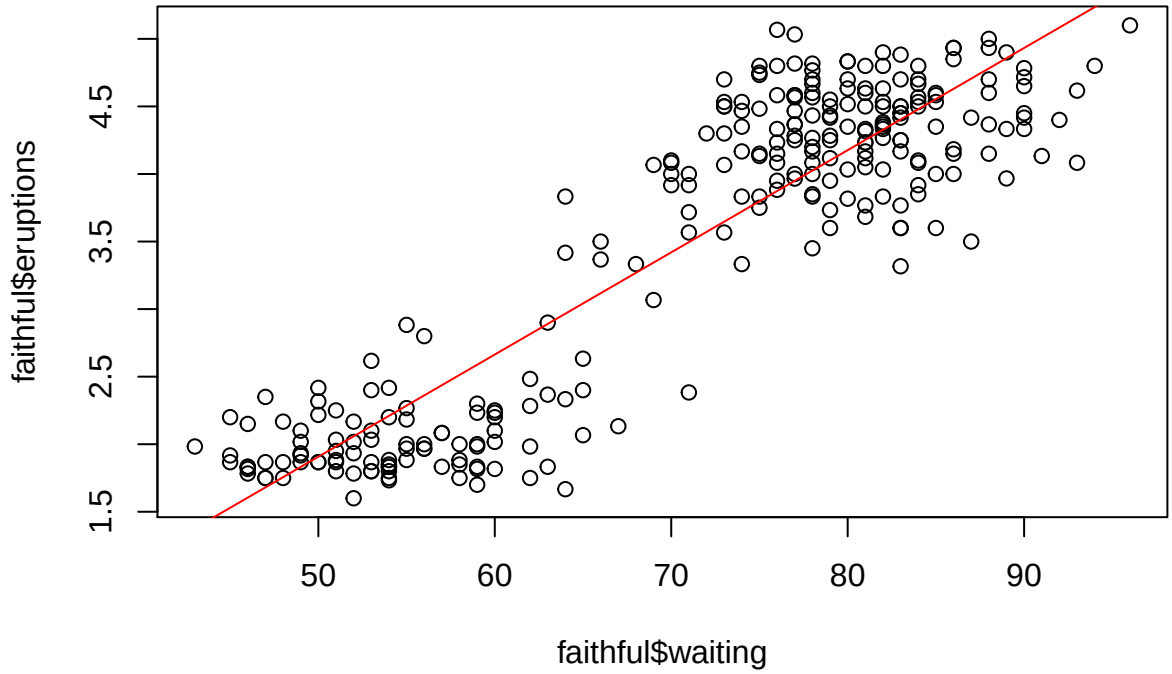
If we plot the eruption time against the waiting time until the next eruption of the Old Faithful Geyser in Yellowstone National Park,

```
plot(faithful$waiting, faithful$eruptions)
```



we can see that we might fit a linear trend there

```
plot(faithful$waiting, faithful$eruptions)
abline(lm(eruptions ~ waiting, data = faithful), col = "red")
```



The most used function to estimate a linear regression is `lm()`.

Consider the linear regression model

$$y = \beta_0 + \beta_1 x + u.$$

For example, if we want to estimate

$$eruptions = \beta_0 + \beta_1 waiting + u,$$

where `eruptions`, `waiting` are variables in the data set `faithful`, then the estimate (by OLS) of the regression model is done by

```
lm( eruptions ~ waiting, data = faithful)
```

```
##
## Call:
## lm(formula = eruptions ~ waiting, data = faithful)
##
## Coefficients:
## (Intercept)      waiting
##    -1.87402      0.07563
```

We can see that calling the function this way results only in the estimate of the intercept and slope.

It is more convenient to assign the result of the `lm()` to an object:

```
m <- lm( eruptions ~ waiting, data = faithful)
```

The output of the estimation can be seen with the `summary()` function:

```
summary(m)
```

```
##
## Call:
## lm(formula = eruptions ~ waiting, data = faithful)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.29917 -0.37689  0.03508  0.34909  1.19329
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.874016   0.160143  -11.70   <2e-16 ***
## waiting      0.075628   0.002219   34.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4965 on 270 degrees of freedom
## Multiple R-squared:  0.8115, Adjusted R-squared:  0.8108
## F-statistic:  1162 on 1 and 270 DF,  p-value: < 2.2e-16
```

We can also assign the output of the summary to another object

```
m.s <- summary(m)
```

and extract parts of it, when convenient; for example, if only the coefficients are of interest,

```
m.s$coefficients
```

```
##                Estimate  Std. Error   t value      Pr(>|t|)
## (Intercept) -1.87401599 0.160143302 -11.70212  7.359171e-26
## waiting      0.07562795 0.002218541  34.08904 8.129959e-100
```

If fewer decimal digits, say 4, are preferable,

```
round(m.s$coefficients, 4)
```

```
##             Estimate Std. Error  t value Pr(>|t|)
## (Intercept)  -1.8740     0.1601 -11.7021        0
## waiting       0.0756     0.0022  34.0890        0
```

Or, perhaps, we want to find out the sample mean of the residuals

```
mean(m.s$residuals)
```

```
## [1] 2.529938e-18
```

or just extract the value of R^2 of the estimate

```
m.s$r.squared
```

```
## [1] 0.8114608
```

Using the `attach()` function

If we are using a single data set in the estimation of a linear regression, the functions `attach()`, `detach()` can be useful. The function `attach()` attaches the data set to the R search path, so that when we want to use a variable in that dataset, we can reference to it directly, without specifying the name of the data set it belongs to. For example,

```
attach(faithful)
m1 <- lm(eruptions ~ waiting)
summary(m1)
```

```
##
## Call:
## lm(formula = eruptions ~ waiting)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.29917 -0.37689  0.03508  0.34909  1.19329
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.874016   0.160143  -11.70   <2e-16 ***
## waiting      0.075628   0.002219   34.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4965 on 270 degrees of freedom
## Multiple R-squared:  0.8115, Adjusted R-squared:  0.8108
## F-statistic:  1162 on 1 and 270 DF,  p-value: < 2.2e-16
```

However, the convenience to reference directly to a variable's name may turn into a liability, because often different data sets may share the same name for different variables. For example, the name `wage` can appear in different econometric data sets. To avoid this confusion, it is strongly recommended that, once we are done with using a specific data set, we remove that data set from R search path by means of `detach()`:

```
detach(faithful)
```

See here for a discussion about attaching or not; alternatives to `attach()` are also considered, among them is the use of `with()`

```
m2 <- with(faithful, lm(eruptions ~ waiting))
summary(m2)
```

```
##
## Call:
## lm(formula = eruptions ~ waiting)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.29917 -0.37689  0.03508  0.34909  1.19329
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.874016   0.160143  -11.70   <2e-16 ***
## waiting      0.075628   0.002219   34.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4965 on 270 degrees of freedom
## Multiple R-squared:  0.8115, Adjusted R-squared:  0.8108
## F-statistic:  1162 on 1 and 270 DF,  p-value: < 2.2e-16
```

Distributions related to the normal distribution

Introduction

In this appendix we present general results that are used in Statistics and Econometrics.

Specifically, we consider distributions derived from the normal distribution such as

- χ^2 distribution
- t(-Student) distribution
- F distribution

Proofs are found, *e.g.*, in the book "Mathematical Statistics and Applications", by Dennis D. Wackerly, William Mendenhall III, Richard Scheaffer, Thomson Books/Cole, 2008.

Sampling distribution of a normal sample

Theorem 1

Let $X_1, \ldots, X_n$ be a random sample of size n from a normal distribution with mean μ and variance σ^2. Then

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

is normally distributed with mean $\mu_{\overline{X}} = \mu$ and variance $\sigma^2_{\overline{X}} = \sigma^2/n$.

Remarks

- Theorem 1 provides the basis of inferential procedures about the (unknown) mean μ of a normal population with known variance σ^2.

- In many applications, especially in Econometrics, the variance is not known. In this case, the population variance σ^2 is estimated by the sample variance

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2$$

and the inferential analysis needs the t distribution, which we consider below.

χ^2 distribution

Let $X_1, \ldots, X_n$ be a random sample of size n from a normal distribution with mean μ and variance σ^2. Then the standardized variables

$$Z_i = \frac{X_i - \mu}{\sigma}$$

are independent standard normal random variables, $i = 1, \ldots, n$, and

$$\sum_{i=1}^{n} Z_i^2 = \sum_{i=1}^{n}\left(\frac{X_i - \mu}{\sigma}\right)^2$$

has χ^2 distribution with n degrees of freedom (df).

Distribution of the sample variance

Let

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2$$

be the sample variance of a random sample.

Theorem 2

Let $X_1, \ldots, X_n$ be a random sample of size n from a normal distribution with mean μ and variance σ^2. Then the sample variance

$$\frac{(n-1)S^2}{\sigma^2} = \frac{1}{\sigma^2}\sum_{i=1}^{n}(X_i - \overline{X})^2$$

has χ^2 distribution with $(n-1)$ degrees of freedom.

Also, $\overline{X}$, S^2 are independent random variables.

t distribution

Theorem 3

Let Z be a standard normal random variable and let W be a χ^2-distributed variable with ν^2 degrees of freedom.

Then, if Z, W are independent,

$$T = \frac{Z}{\sqrt{W/\nu}}$$

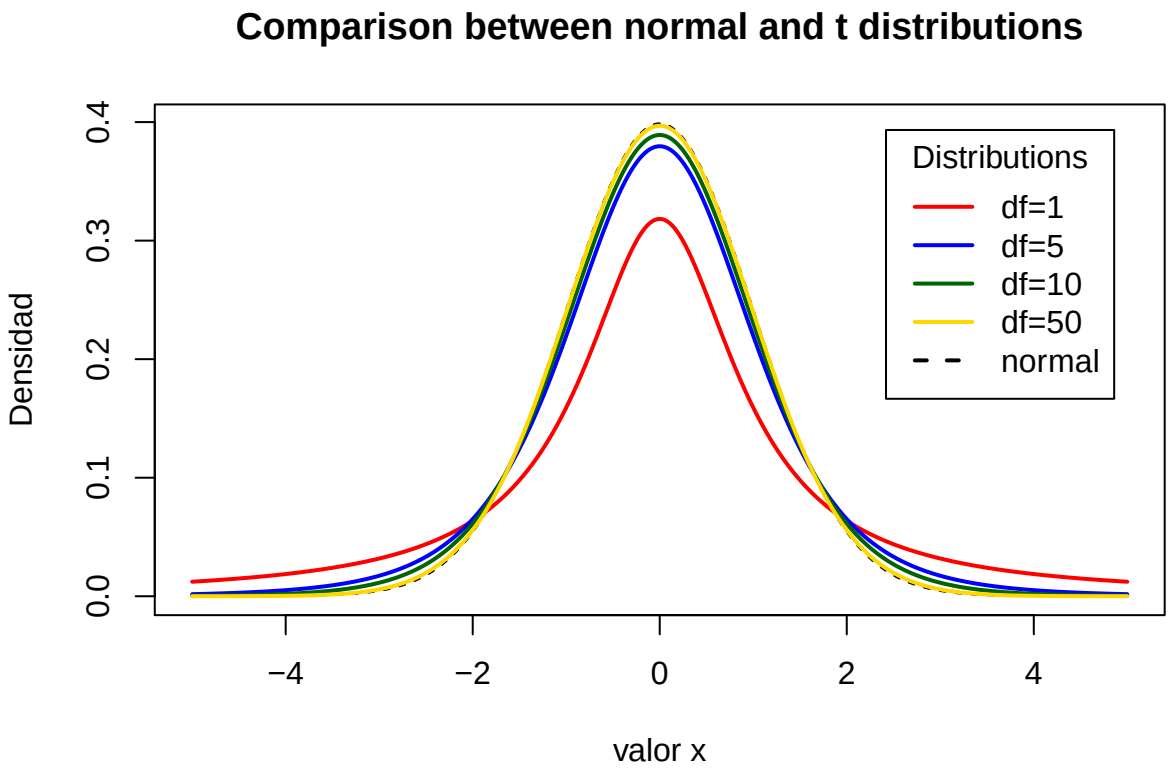
is said to have a t distribution with ν degrees of freedom.

Remark

For large values of ν (degrees of freedom), the normal and t distributions are very close, as shown below.

Comparison between normal and t distributions

[2] ν is the Greek letter "nu".



F distribution

Definition

Let W_1, W_2 be *independent* χ^2-distributed random variables with ν_1, ν_2 df, respectively. Then,

$$F = \frac{W_1/\nu_1}{W_2/\nu_2}$$

is said to have an F distribution with ν_1 numerator degrees of freedom and ν_2 denominator degrees of freedom.