Assignment 0

James Bond, group 007

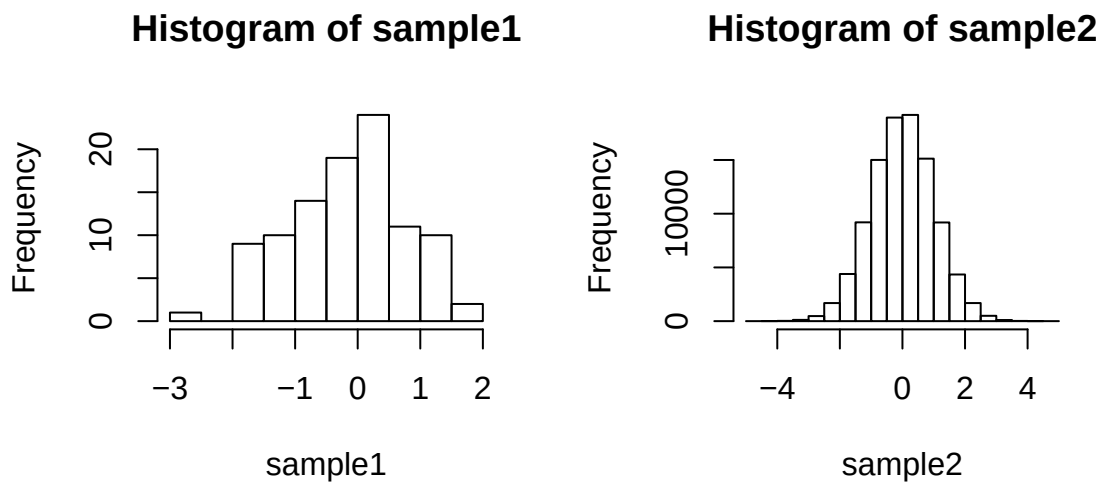
1 January 2033

In order not to be bothered with rounding the numbers, set `options(digits=3)` .

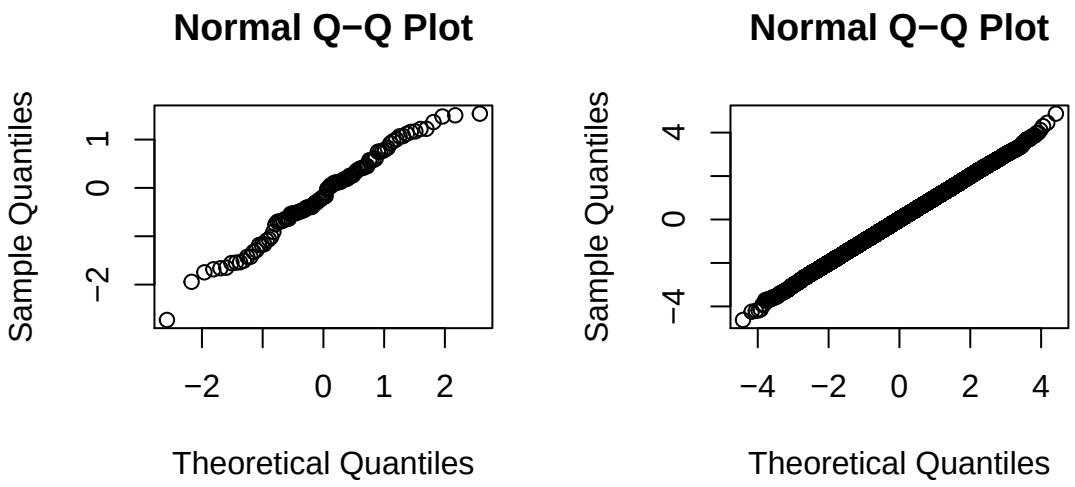
Exercise 1

a) We generate two samples of sizes 100 and 100000 from a standard normal distribution N(0,1) as follows: `sample1=rnorm(100)` , `sample2=rnorm(100000)`; then make histograms and QQ-plots for the both samples.

```
par(mfrow=c(1,2)); hist(sample1); hist(sample2) # two histograms next to each other
```




```
qqnorm(sample1); qqnorm(sample2) # two QQ-plots next to each other
```



For different samples, the figures are different. The quality of the histogram and QQ-plot depend on the sample size. If it is small, the histogram varies more and the QQ-plot varies more around a straight line whereas for large samples size the histogram is very stable and close to the true density, and the QQ-plot is straight in the middle with just some variation in the corners. The values of `mean` and `sd` only influence the scales on the axes, not the straightness of the line in the QQ-plot.

Now, we compute the means and standard deviations for the both samples, and summarize the results in the table.

Parameters	Est. for sample n=100	Est. for sample n=100000
mean=0	`mean(sample1)`=-0.164	`mean(sample2)`=8.605×10^{-5}
sd=1	`sd(sample1)`=0.894	`sd(sample2)`=1.001

The estimated mean and standard deviation are also clearly better for the second sample. This is not surprising as the second sample is of a much bigger size, i.e., containing much more data.

b) Given Z has a standard normal distribution, we need to compute the following probabilities: P(Z<2)=`pnorm(2)`=0.977, P(Z>-0.5)=`1-pnorm(-0.5)`=0.691, P(-1<Z<2)=`pnorm(2)-pnorm(-1)`=0.819.

c) For Z~N(0,1), the probabilities P(Z<2)=0.977, P(X>-0.5)=0.691 and P(-1 < Z < 2)=0.819 from b) can be estimated by using the data from a) as follows:

```
p1=sum(sample1<2)/length(sample1) # estimate of P(Z<2) for sample 1 with n=100
p2=sum(sample2<2)/length(sample2) # estimate of P(Z<2) for sample 2 with n=100000
p3=sum(sample1>-0.5)/length(sample1) # estimates of P(Z>-0.5) for sample 1
p4=sum(sample2>-0.5)/length(sample2) # estimates of P(Z>-0.5) for sample 2
p5=sum(sample1>-1&sample1<2)/length(sample1) # estimate of P(-1<Z<2) for sample 1
p6=sum(sample2>-1&sample2<2)/length(sample2) # estimate of P(-1<Z<2) for sample 2
c(p1,p2,p3,p4,p5,p6) # print all the estimates
## [1] 1.000 0.977 0.660 0.691 0.800 0.818
```

Summarize the results in the table. The 2nd and 3d columns in this table are the estimates of the corresponding theoretical probabilities from b).

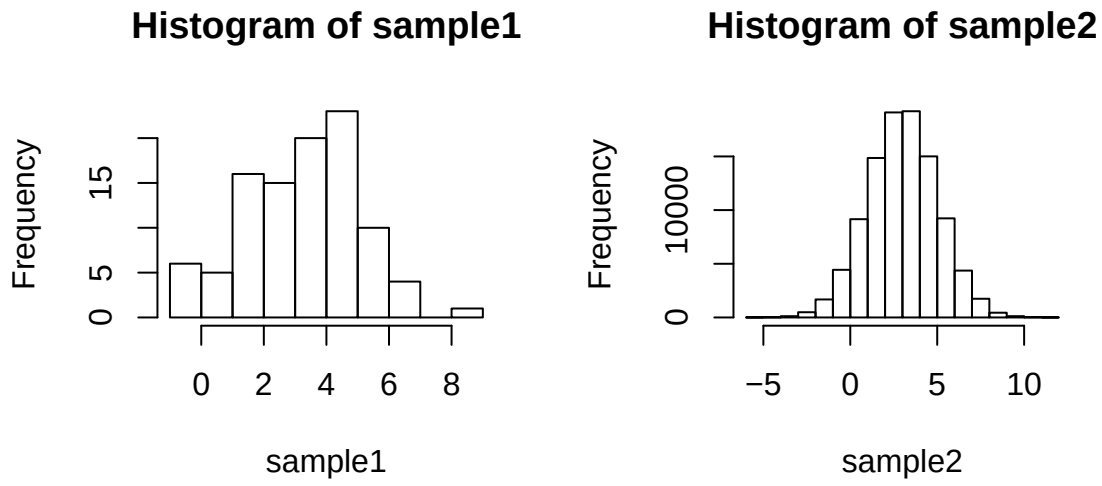
Probabilities from b)	Est. for sample n=100	Est. for sample n=100000
P(Z<2)=0.977	p1=1	p2=0.977
P(Z>-0.5)=0.691	p3=0.66	p4=0.691
P(-1<Z<2)=0.819	p5=0.8	p6=0.818

The estimates based on the second sample are clearly better, because the second sample is larger.

d) As in a), we first generate the samples `sample1=rnorm(100,mean=3,sd=2)`, `sample2=rnorm(100000,3,2)`. Next, we estimate the parameters `mean` and `sd` and construct histrograms for the both samples.

Parameters	Est. for sample n=100	Est. for sample n=100000
mean=3	`mean(sample1)`=3.24	`mean(sample2)`=3.002
sd=2	`sd(sample1)`=1.842	`sd(sample2)`=2.004

```
par(mfrow=c(1,2)); hist(sample1); hist(sample2)
```



As before, the estimates and histrogram for the second sample are better as this sample is of a larger size.

For X~N(3,4), the probabilities are now found as follows: P(X<2)=`pnorm(2,mean=3,sd=2)`=0.309, P(X>-0.5)=`1-pnorm(-0.5,mean=3,sd=2)`=0.96, P(-1<X<2)=`pnorm(2,3,2)-pnorm(-1,3,2)`=0.286.

The value such that 95% of the outcomes is smaller than that value is nothing else but the 95%-quantile of the distribution N(3,4), which is `qnorm(0.95,mean=3,sd=2)`=6.29. Notice that it can also be found via the 95%-quantile `qnorm(0.95)` of the standard normal distribution as `3+2*qnorm(0.95)`=6.29.

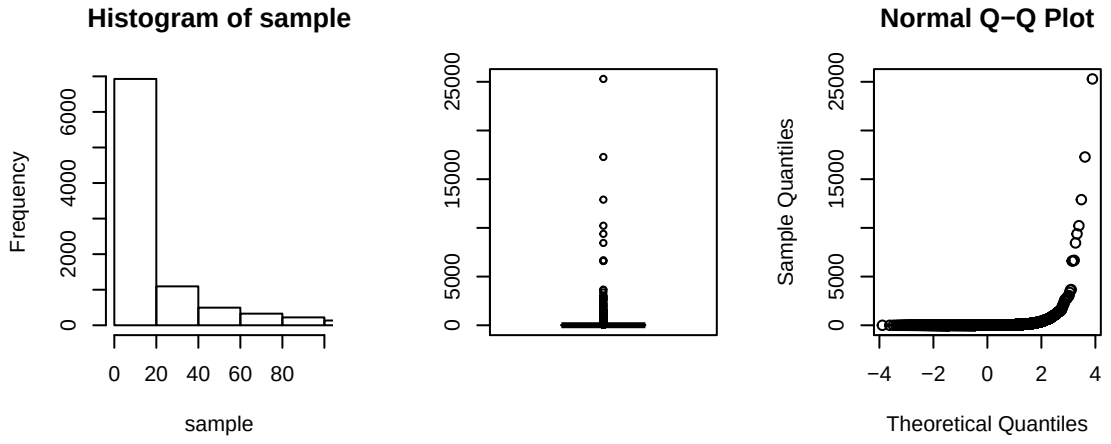
e) Any normal variable X~N(mu,sigma^2) can be generated from a standard normally distributed Z~N(0,1) as X=mu+sigma*Z. We generate in this way a sample of size 1000 from a normal distribution with `mean`=-10 and `sd`=5, and verify that the sample mean and sample standard deviation are close to the true values `mean`=-10 and `sd`=5.

```
sample=-10+5*rnorm(1000)
c(mean(sample),sd(sample)) # should be close to mean=-10 and sd=5
## [1] -9.90  4.88
```

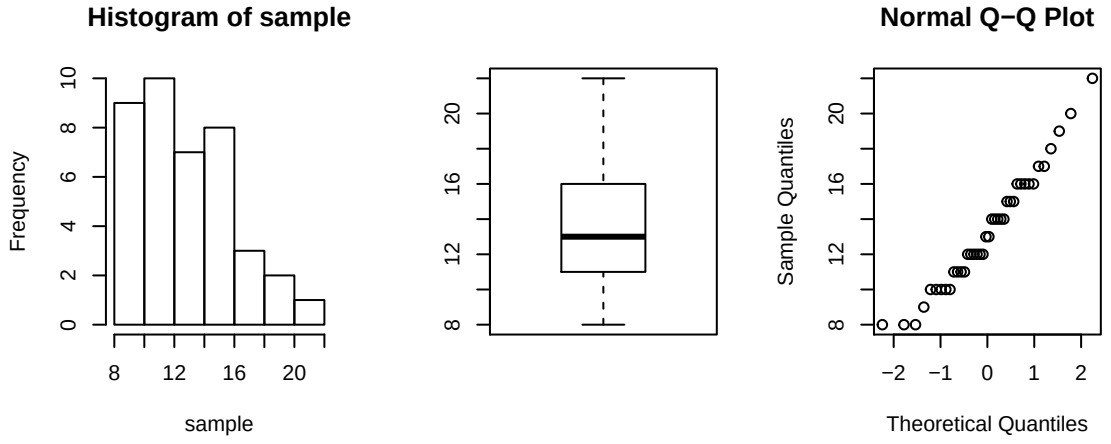
Exercise 2

We generate samples from the asked distributions and plot for each of the generated samples the histogram, boxplot and QQ-plot:

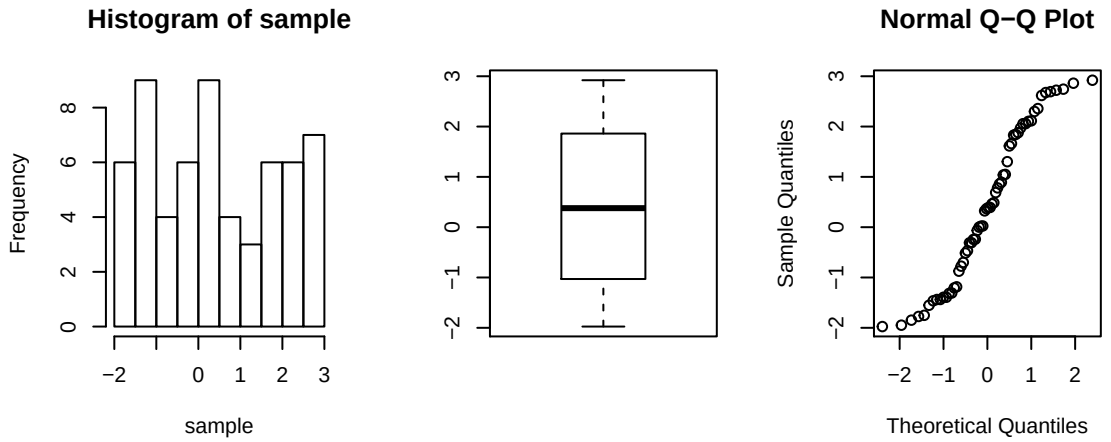
```
par(mfrow=c(1,3)) # two plots next each other
sample=rlnorm(10000,2,2) # from the lognormal distribution with mu=sigma=2
hist(sample,xlim=c(0,100),breaks=1000) # hist(sample) will not look good, why?
# to see the breaks: hist(sample,xlim=c(0,100),breaks=1000)$breaks
boxplot(sample) # a lot of outliers
qqnorm(sample) # of course, not normal
```



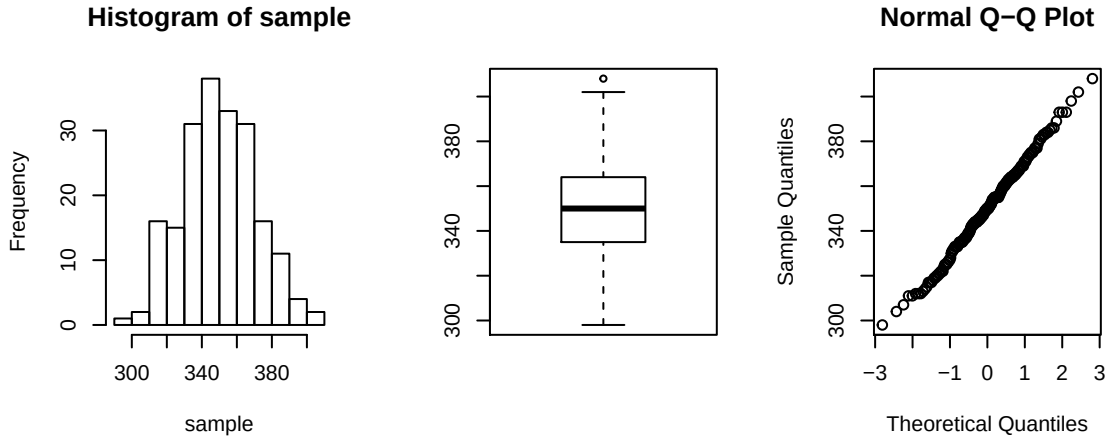

```
sample=rbinom(40,50,0.25) # from the binomial distribution with n=50 and p=0.25
hist(sample);boxplot(sample);qqnorm(sample) # looks like normal
```




```
sample=runif(60,-2,3) #from the uniform distribution on the interval [-2,3]
hist(sample);boxplot(sample);qqnorm(sample) # of course, not normal
```




```
sample=rpois(200,350) #from the Poisson distribution with lambda = 350
hist(sample);boxplot(sample);qqnorm(sample) # looks like normal
```



All but lognormal are symmetric (possibly not around zero), binomial and Poisson look like normal. Small sample sizes (10,40,60) show noise. Histograms are more stable and give better approximation of the true density for sufficiently large sample sizes.

Exercise 3

a) We read in the dataframe `mortality` and produce a couple of summaries of the two columns `mortality$teen` and `mortality$mort`.

```
# If necessary, you may need to set the working R-directory to be the one
# that contains the right data file: setwd("~/Documents/your R folder").
mortality=read.table("mortality.txt",header=TRUE) # read in the data
b=mortality$teen; summary(b)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    7.30    9.85   11.65   12.43   15.22   20.50
```

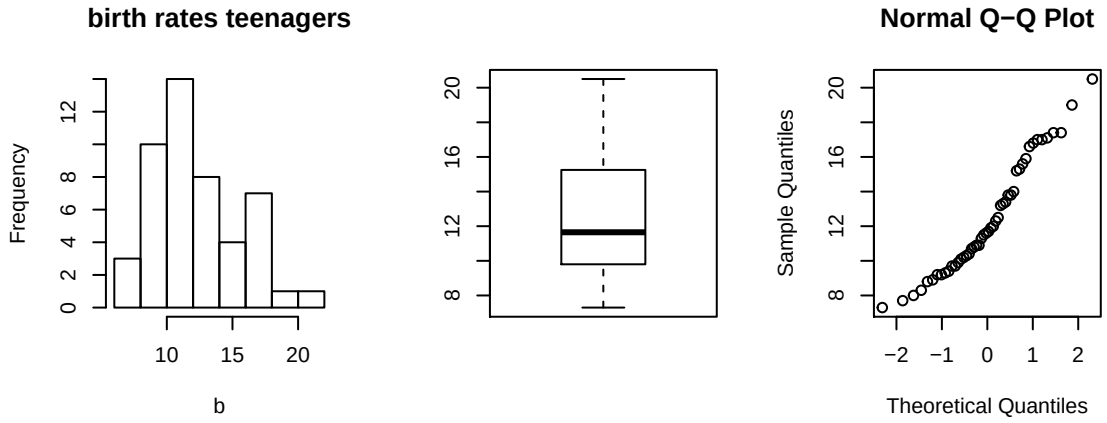
```
m=mortality$mort; summary(m)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

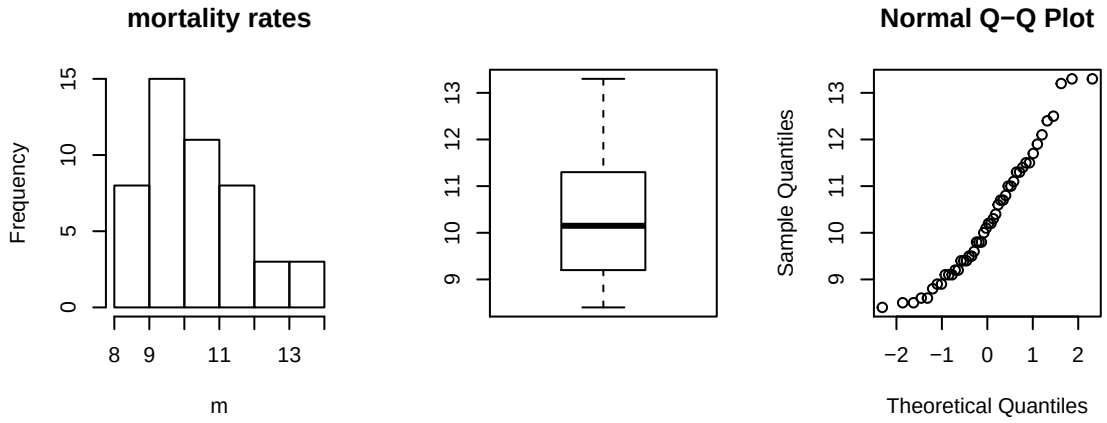
```
##      8.4      9.2     10.2     10.3     11.3     13.3
```

The other characteristics: `var(b)`=10.844, `sd(b)`=3.293, `range(b)`=(7.3,20.5) for the column `b=mortality$teen`, and `var(m)`=1.822, `sd(m)`=1.35, `range(m)`=(8.4,13.3) for the column `m=mortality$mort`.

```
par(mfrow=c(1,3))
hist(b,main="birth rates teenagers");boxplot(b);qqnorm(b)
```



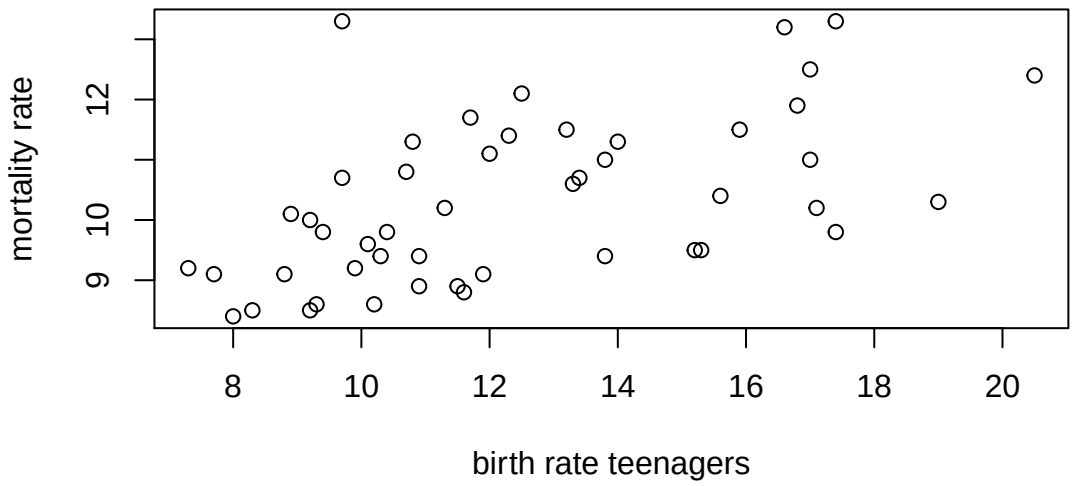

```
hist(m,main="mortality rates");boxplot(m);qqnorm(m)
```



Both look rather symmetric, unimodal, and somewhat deviating from normal.

b) Now look at the correlation and the scatter plot between the two rates.

```
cor(b,m)
## [1] 0.549
par(mfrow=c(1,1)) # make also a scatter plot one against the other
plot(b,m,xlab="birth rate teenagers",ylab="mortality rate")
```



The correlation between the birth and mortality rates `cor(b,m)`$=0.549$ is positive. Also in the plot we see some positive slope, but there is quite some variation: there may be more (unknown) influence factors.

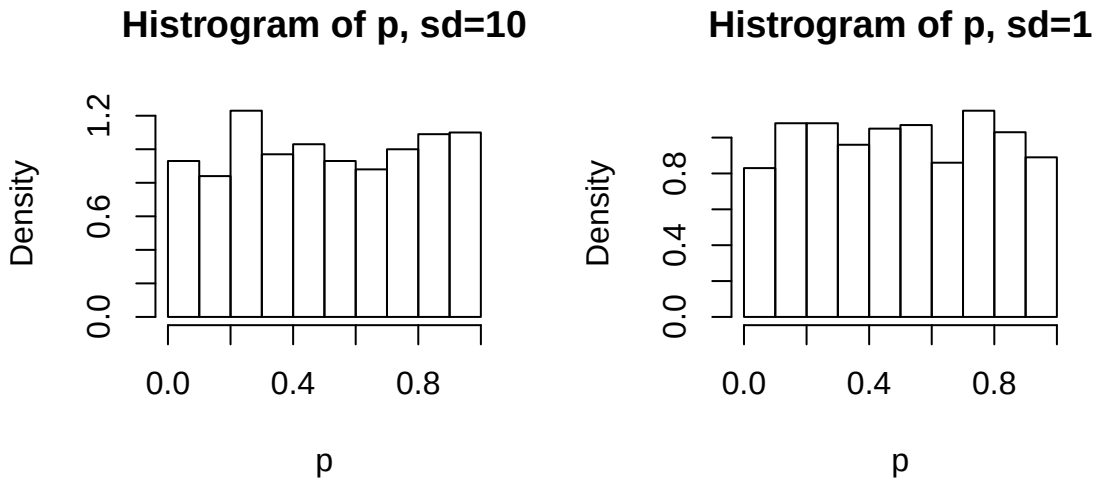
Exercise 4

The following function outputs an array of p-values of the t-test (two independent samples, equal variances).

```
p.value=function(n,m,mu,nu,sd,B=1000){
  p=numeric(B) # p will be an array of realized p-values
  for (b in 1:B) {x=rnorm(n,mu,sd); y=rnorm(m,nu,sd)
                  p[b]=t.test(x,y,var.equal=TRUE)[[3]]}
 return(p)}
```

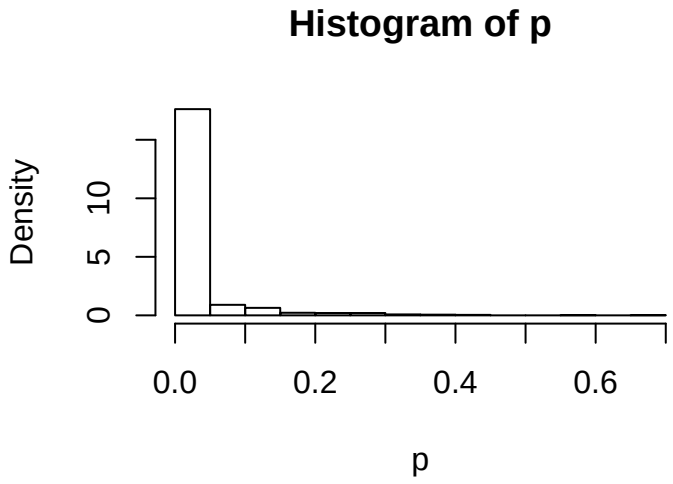
a)-b) In these both cases the null hypothesis H0 holds because `mu=nu=180`.

```
par(mfrow=c(1,2))
## a)
n=m=30; mu=nu=180; sd=10; # set the parameters for the case a)
p=p.value(n,m,mu,nu,sd) # an array of p-values
mean(p<0.05) # fraction p-values smaller than 5%, should be appr. 0.05
## [1] 0.048
mean(p<0.1) # fraction p-values smaller than 10%, should be approx. 0.1
## [1] 0.093
hist(p,freq=F,main="Histrogram of p, sd=10") # should be approx uniform on [0,1]
## b)
n=m=30; mu=nu=180; sd=1 # set the parameters for the case b)
p=p.value(n,m,mu,nu,sd) # an array of p-values
mean(p<0.05) # fraction p-values smaller than 5%, should be appr. 0.05
## [1] 0.042
mean(p<0.1) # fraction p-values smaller than 10%, should be approx. 0.1
## [1] 0.083
hist(p,freq=F,main="Histrogram of p, sd=1") # should be approx uniform on [0,1]
```



c) Now the null hypothesis H0 does not hold because `mu=180`, `nu=175`.

```
n=m=30; mu=180; nu=175; sd=6
p=p.value(n,m,mu,nu,sd) # an array of p-values
mean(p<0.05) # should not be close to 0.05
## [1] 0.881
mean(p<0.1) # should not be close to 0.1
## [1] 0.926
hist(p,freq=F) # should not be uniform on [0,1]
```



d) The null hypothesis H0 holds in a) and b) as `mu=nu`. Under H0, p-values are distributed uniformly on [0,1]. Hence the events {p<0.05} and {p<0.1} should occur approximately in 5% and 10% of cases respectively, and histograms of p-values should be close to uniform on [0,1]. In b) the approximations should be better because the variance is smaller.

In c) H0 does not hold (because `mu>nu`), so p-values are not uniformly distributed and `mean(p<0.05)` gives approximately the values of the power function at point `mu-nu=180-175=5`, which should approach 1 for a good test.

All these claims are confirmed by the simulations results in a), b) and c).