Solutions for Exercises for Applied Biostatistics I - HS 2018

1. If the elementary events have equal probability we can calculate the probability of event A as the number of elementary events in A divided by the total number of elementary events.

a) $\Omega = \{(1,1),(1,2),\dots,(1,6),(2,1),(2,2),\dots,(2,6),\dots,(6,6)\}$, $|\Omega| = 36$.

b) $P[\{\text{elementary event}\}] = \frac{1}{|\Omega|} = \frac{1}{36}$.

c) $E_1 = \{(1,6),(2,5),(3,4),(4,3),(5,2),(6,1)\}$;
Number of favourable cases: $|E_1| = 6$;
Number of possible cases: $|\Omega| = 36$;
$P[E_1] = \frac{|E_1|}{|\Omega|} = \frac{6}{36} = \frac{1}{6}$.

d) $E_2 = \{(1,1),(2,1),(1,2)\}$;
$P[E_2] = \frac{|E_2|}{|\Omega|} = \frac{3}{36} = \frac{1}{12}$.

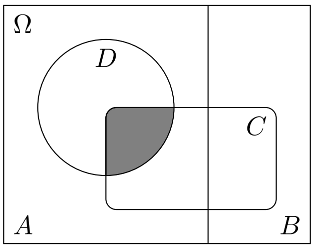
e) $E_3 = \{(1,1),(1,3),(1,5),(3,1),(3,3),(3,5),(5,1),(5,3),(5,5)\}$;
$P[E_3] = \frac{|E_3|}{|\Omega|} = \frac{9}{36} = \frac{1}{4}$.

f)

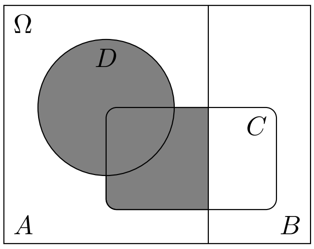
$$\begin{aligned} P[E_2 \cup E_3] &= P[E_2] + P[E_3] - P[E_2 \cap E_3] \\ &= P[E_2] + P[E_3] - P[\{(1,1)\}] \\ &= \frac{3}{36} + \frac{9}{36} - \frac{1}{36} = \frac{11}{36}. \end{aligned}$$

2. **a)** The operators union ($\cup$), intersection ($\cap$) and complement (^{c}) operate on sets and the addition ($+$) on numbers. We get that (i) and (ii) are meaningful and (iii) and (iv) are not.

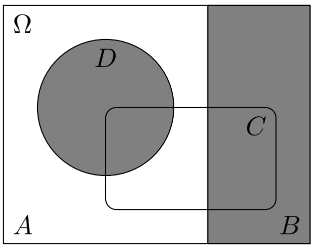
b) (i):



(ii):



(iii):



3. **a)** The event C occurs if either A or B occurs but not both. $C = (A \cup B) \setminus (A \cap B)$

b) $P[C] = P[(A \cup B) \setminus (A \cap B)] = P[A \cup B] - P[A \cap B] \overset{(*)}{=} P[A] + P[B] - 2P[A]P[B] = 0.1679$
Note that we made use of the fact that the events A and B are independent in the step $(*)$.

4. **a)** The solution is found dividing the number of smokers by the total number of participants:

$$P[\text{sm}] = \frac{1213}{6549} = 0.185.$$

b) We know that

$$P[A|B] = \frac{P[A \cap B]}{P[B]}.$$

In this case, A is the smoking fraction, while B is the high income fraction of the sample. We have

$$P[\text{hi}] = \frac{2115}{6549}.$$

while

$$P[\text{sm} \cap \text{hi}] = \frac{247}{6549}.$$

Then

$$P[\text{sm}|\text{hi}] = \frac{247}{6549} \cdot \frac{6549}{2115} = \frac{247}{2115} = 0.117.$$

c) To ask this question we have to answer the question if $P[\text{sm}] = P[\text{sm}|\text{hi}]$. If the conditional probability of being a smoker while having an high revenue is the same, then the two are independent. In this case we see from exercice a) and b) that these two qualities are dependent, *i.e.* the probability of smoking is lower for high revenue people than for the whole sample.

d) This question corresponds to asking if $P[\text{hsm}] = P[\text{hsm}|\text{wsm}]$, where "hsm" is the event of a husband smoking, "wsm" is the event of a wife smoking. We know that $P[\text{hsm}] = 0.3$ and $P[\text{wsm}] = 0.2$. Furthermore, we know that $P[\text{hsm} \cap \text{wsm}] = 0.08$. Then, we compute

$$P[\text{hsm}|\text{wsm}] = \frac{P[\text{hsm} \cap \text{wsm}]}{P[\text{wsm}]} = \frac{0.08}{0.2} = 0.4.$$

Hence the smoking status is dependent of that of the wife: if the wife smokes, the man has 40% of probability of smoking too.

5. **a)** Let X be the number of contaminated samples in one collective sample. The probability that a sample is contaminated is $\pi = 0.02$. Under the assumption that all samples are independent, X is binomially distributed: $X \sim \text{Bin}(n = 10, \pi = 0.02)$.
The probability not to find any contamination in the sample is given by

$$P[X = 0] = \binom{10}{0} \cdot 0.02^0 \cdot 0.98^{10} = 0.98^{10} = 0.8171.$$

In R, we can calculate $P[X = 0]$ as

```
> dbinom(0, size = 10, prob = 0.02)
[1] 0.8170728
```

Another possible solution: each sample is clean with a probability of 0.98, independently of the other samples. Therefore we have

$$P[\text{all samples are clean}] = \prod_{i=1}^{10} P[i\text{-th sample is clean}] = 0.98^{10} = 0.8171.$$

b) The random variable Y can only have the values 1 or 11, because:

1. if all samples are clean, we are done after one analysis: $Y = 1$.
2. if at least one sample is contaminated, then the collective sample is contaminated and we need to check all 10 samples separately: $Y = 11$

Hence

$$P[Y = 1] = P[\text{no sample is contaminated}] = 0.8171 \ ,$$
$$P[Y = 11] = 1 - P[Y = 1] = 0.1829 \ .$$

c) The average number of analyses for one collective sample is given through the expectation value of Y:

$$E[Y] = \sum_{k=0}^{\infty} kP[Y=k] = 1 \cdot P[Y=1] + 11 \cdot P[Y=11] = 1 \cdot 0.8171 + 11 \cdot 0.1829 = 2.8293 \ .$$

On average we save $10 - 2.8293 = 7.1707 \approx 7$ analyses.

6. We first prove the equality in case of discrete random variables. Then

$$\begin{aligned} \mathrm{Var}(X) &\overset{\text{def}}{=} \sum_{k=1}^{\infty} \big(x_k - E[X]\big)^2 p(x_k) = \sum_{k=1}^{\infty} \Big(x_k^2 + E[X]^2 - 2x_k E[X]\Big)\, p(x_k) \\ &= \underbrace{\sum_{k=1}^{\infty} x_k^2\, p(x_k)}_{E[X^2]} + E[X]^2 - 2E[X] \underbrace{\sum_{k=1}^{\infty} x_k\, p(x_k)}_{E[X]} = E[X^2] - E[X]^2. \end{aligned}$$

In case of a continuous random variable the proof is similar. We have

$$\begin{aligned} \mathrm{Var}(X) &\overset{\text{def}}{=} \int_{\mathbb{R}} \big(x - E[X]\big)^2 f(x)\, dx = \int_{\mathbb{R}} \Big(x^2 + E[X]^2 - 2xE[X]\Big)\, f(x)\, dx \\ &= \underbrace{\int_{\mathbb{R}} x^2\, f(x)\, dx}_{E[X^2]} + E[X]^2 - 2E[X] \underbrace{\int_{\mathbb{R}} x\, f(x)\, dx}_{E[X]} = E[X^2] - E[X]^2. \end{aligned}$$

7. The expectation value of a discrete random variable X can be calculated with the following formula:

$$E[X] = \sum_k k \cdot P[X=k].$$

As X is Poisson distributed with parameter λ, we know

$$P[X=k] = \frac{\lambda^k}{k!} e^{-\lambda}.$$

This gives us the equations:

$$\begin{aligned} E[X] = \sum_k k \cdot P[X=k] &= \sum_{k=0}^{\infty} k \cdot \frac{\lambda^k}{k!} e^{-\lambda} \\ &= e^{-\lambda} \sum_{k=1}^{\infty} k \cdot \frac{\lambda^k}{k!} \\ &= e^{-\lambda} \cdot \lambda \cdot \sum_{k=1}^{\infty} \frac{\lambda^{(k-1)}}{(k-1)!} \\ &= e^{-\lambda} \cdot \lambda \cdot \sum_{k=0}^{\infty} \frac{\lambda^k}{k!} \\ &= e^{-\lambda} \cdot \lambda \cdot e^{\lambda} \\ &= \lambda \end{aligned}$$

8. a) Haldane's model claims, that the number of crossovers is Poisson distributed with parameter $\lambda =$ "length of the chromosome in Morgan". Thus we have

$$P[X=0] = \frac{\lambda^0}{0!} \cdot e^{-\lambda} = e^{-3.05} = 0.047.$$

If we solve this problem with R, we get the following solution:

```
> lambda <- 3.05
> ppois(0, lambda)
[1] 0.04735892
```

b) With the same arguments as in a) we have

$$P[k \geq 2] = 1 - P[k = 1] - P[k = 0] = 1 - \lambda \cdot e^{-\lambda} - e^{-\lambda} = 0.808.$$

We can solve this exercise also with R:

```
> 1 - ppois(1, lambda)
[1] 0.8081964
```

c) As for the Poisson distribution with parameter λ the expectation value is exactly λ, we get $E[X] = \lambda = 3.05$.

d) As k can only be a natural number, we have

$$P[k \geq 3.05] = P[k \geq 4] = 1 - P[k = 3] - P[k = 2] - P[k = 1] - P[k = 0] = 0.364.$$

A solution to solve this exercise with R is:

```
> 1 - ppois(lambda, lambda)
[1] 0.3639687
```

9. **a)** A recombination (event R) between the two genes happens if and only if there is an *odd* number of crossovers between them. Formally:

$$R = \{X \text{ is odd}\} = \{X = 1\} \cup \{X = 3\} \cup \{X = 5\} \cup \ldots$$

b) By the consideration from task a), we have to calculate

$$P[R] = P[X = 1] + P[X = 3] + P[X = 5] + \ldots$$

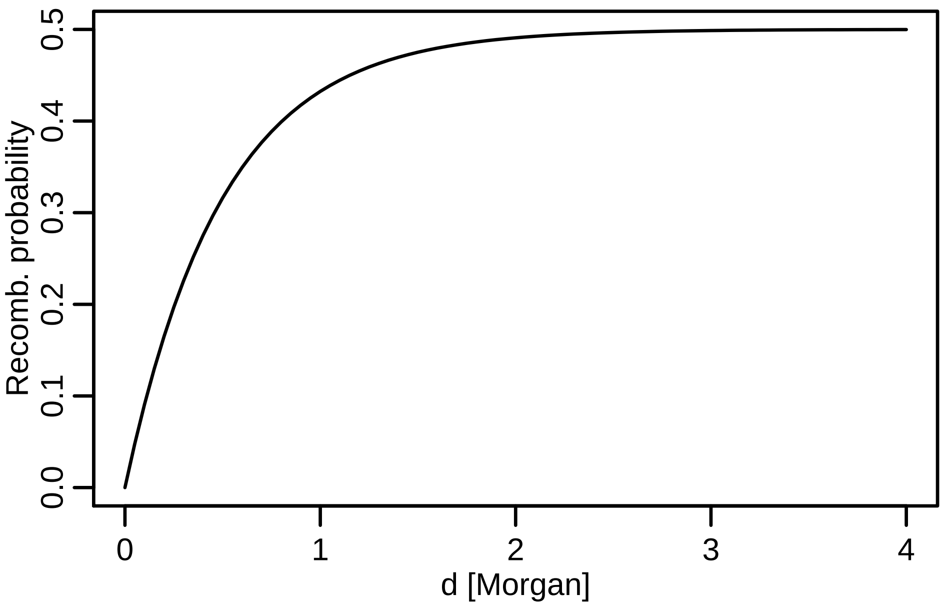
Since X is Poisson distributed with rate parameter $\lambda = d$ (Haldane's model!), we have $P[X = x] = e^{-d} \cdot \frac{d^x}{d!}$ and hence

$$P[R] = e^{-d} \cdot \left(d + \frac{d^3}{3!} + \frac{d^5}{5!} + \ldots \right) = e^{-d} \sum_{k=0}^{\infty} \frac{d^{2k+1}}{(2k+1)!}$$

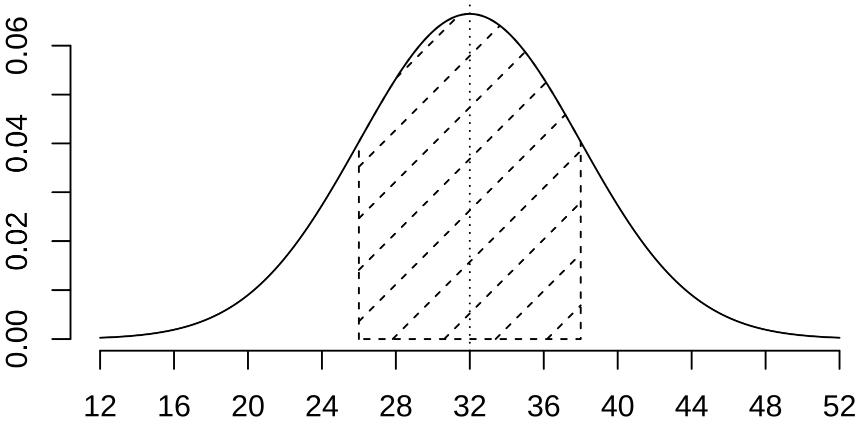
Using the hint, we find the recombination probability

$$P[R] = e^{-d} \sinh(d) \ .$$

A plot of this function shows that the recombination probability goes to $\frac{1}{2}$ as d grows, exactly as we expect:



10. a)



b) Let X be the lead content of the sample. It holds that

$$X \sim \mathcal{N}(\mu, \sigma^2) \qquad \text{with } \mu = 32 \text{ and } \sigma^2 = 6^2.$$

We solve the following problems with R. To calculate $P[X \leq 40]$ we use the following command:

```
> pnorm(40,mean = 32, sd = 6)
[1] 0.9087888
```

c) We calculate $P[X \leq 27]$.

```
> pnorm(27, mean = 32, sd = 6)
[1] 0.2023284
```

d) We chose c such that $P[X \leq c] = 0.975$.

```
> qnorm(0.975, mean = 32, sd = 6)
[1] 43.75978
```

e) This time we chose c such that $P[X \leq c] = 0.1$.

```
> qnorm(0.1, mean = 32, sd = 6)
[1] 24.31069
```

f) What is the probability of the area you draw in part a) of this exercise?

To calculate $P[26 \leq X \leq 38] = P[X \leq 38] - P[X < 26]$ we use the following command:

```
> pnorm(38, mean = 32, sd = 6) - pnorm(26, mean = 32, sd = 6)
[1] 0.6826895
```

11. a) If we calculate the sum of each row and each column, we see that the overall value of the probabilities is 1. We get the following table:

X/Y	1	2	3	$\sum$
1	0.05	0.08	0.12	0.25
2	0.14	0.19	0.09	0.42
3	0.22	0.08	0.03	0.33
$\sum$	0.41	0.35	0.24	1.00

The marignal distribution of X therefore is
$X = 1$ with probability 0.25
$X = 2$ with probability 0.42 and
$X = 3$ with probability 0.33.

The marignal distribution of Y therefore is
$Y = 1$ with probability 0.41
$Y = 2$ with probability 0.35 and
$Y = 3$ with probability 0.24.

b) The probability of X being on a low level is $P[X = 1] = 0.25$.
If we know the value of Y, the probability changes, as we now have a conditional probability. X can still take the same values (1,2 and 3), but as we already now that Y takes the value 2, only the middle column of the table is important to us. As we still need a total probability of 1, we need to adjust the probabilities, with which X takes its values. We do that with the formula

$$p_{X|Y=2}(i) = P[X = i|Y = 2] = \frac{P[X = i, Y = 2]}{P[Y = 2]} = \frac{p_{X,Y}(x, y)}{p_Y(2)} .$$

We have $p_{X|Y=2}(1) = P[X = 1|Y = 2] = \frac{0.08}{0.35} = 0.228571428571429$.

c) Gene X downregulates gene Y. Then for high X values, the higher Y values occur much less often than the smaller Y values. This is an indication of downregulation.

d) No, you can either show that by calculating the two values $p_{Y|X=1}(2)$ and $p_{Y|X=3}(2)$ or by argumentation.

In the cases of $p_{X,Y}(1, 2)$ and $p_{X,Y}(3, 2)$ we want to know the probabilities that Y takes the value 2 and X takes the value 1 (respectively 3). But we have just the probabilities what value X and Y are going to take.
In the case of $p_{Y|X=1}(2)$ and $p_{Y|X=3}(2)$ we already know the value of X. So we only need to know with which probability Y has the value 2, when X takes the value 1 (respectively 3).
As the random variables X and Y are dependent, there is a difference.

12. **a)** Importing the data set:

```
> d.pet <- read.table(”count.txt”, header = TRUE)
```

The argument `header = TRUE` tells R that the orignal dataset contains variable names on the first line.
The imported dataset `d.pet` is saved as a `data.frame`:

```
> class(d.pet)
[1] ”data.frame”
```

It has 5000 rows and 4 columns (variables):

```
> dim(d.pet)
[1] 5000 4
```

The function `str()` shows the internal structure of the R-object.

```
> str(d.pet)
’data.frame’: 5000 obs. of 4 variables:
$ a.v1: int 27 15 38 25 23 21 21 23 27 22 ...
$ a.v2: int 0 0 0 1 1 0 1 0 0 0 ...
$ b.v1: int 0 0 1 2 0 2 1 2 2 1 ...
$ c.v1: int 14 13 18 8 12 13 8 15 1 16 ...
```

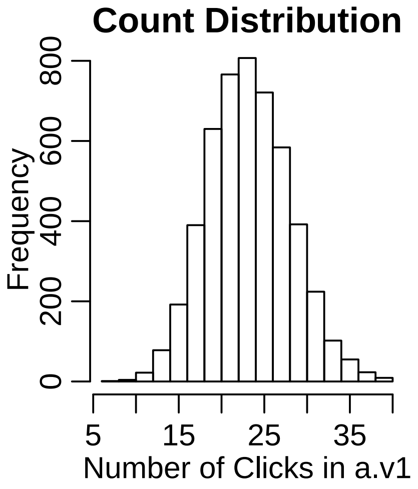
We can see the name of the variables, the saved type and a preview of the first entries. For example, the first variable is called `a.v1` and saved as an integer. In this dataset all variables of `d.pet` are saved as integers.

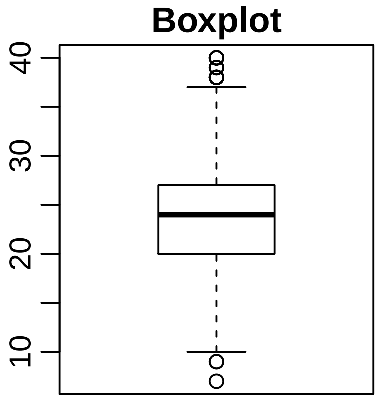
b) The characteristic numbers can be calculated as follows:

```
> mean(d.pet$a.v1) # Mean
[1] 23.708
> var(d.pet$a.v1) # Variance
[1] 23.13856
> summary(d.pet$a.v1) # Quantile, Minimum and Maximum
Min. 1st Qu. Median Mean 3rd Qu. Max.
7.00 20.00 24.00 23.71 27.00 40.00
```

c) Both the histogram and the boxplot represent the distribution of the observed click-count. However the histogram shows this distribution in its entirety by plotting the frequency of every possible number of clicks, whereas the boxplot might end up hiding some of the specific shape of the count distribution.

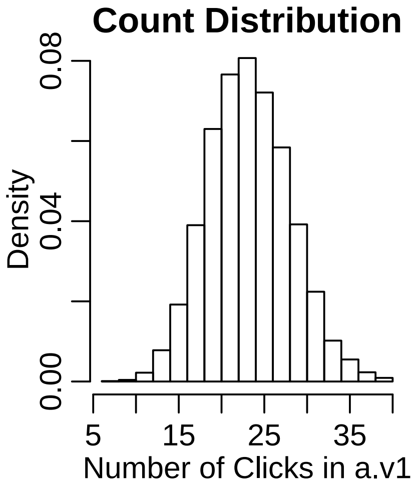
```
> d.pet <- read.table(”count.txt”, header = TRUE)
> par(mfrow = c(1,2)) # two plots in one window
> hist(d.pet$a.v1, main = ”Count Distribution”,xlab=”Number of Clicks in a.v1”)
> counts <- c(d.pet$a.v1)
> boxplot(counts, main = ”Boxplot”)
```





To obtain a scaled histogram we use the argument `probabilities = TRUE`.

```
> hist(d.pet$a.v1,probability = TRUE, main = ”Count Distribution”, xlab=”Number of Clicks
in a.v1”)
```



13. a) The number of test tubes of minor value X has a binomial distribution.

b) Here $X \sim \text{Bin}(n, \pi)$ with $n = 50$ and $\pi = 0.1$. So we get

$$P[X = 3] = \binom{50}{3} 0.1^3 \cdot 0.9^{47} = 0.139.$$

c) We have again $X \sim \text{Bin}(n, \pi)$ with $n = 50$ and $\pi = 0.1$. The probability that X is at most 3 is given by

$$P[X \leq 3] = \sum_{k=0}^{3} \binom{50}{k} (0.1)^k \cdot (0.9)^{50-k} = 0.25.$$

d) We use the central limit theorem (CLT). We approximate the cumulative distribution function (CDF) of X by the CDF of a normal distribution with mean $\mu = n\pi$ and variance $\sigma^2 = n\pi(1-\pi)$. So we get

$$P[X \leq 3] \approx 0.17.$$

We observe that the rule of thumb from the lecture notes is violated and so the approximation is rather imprecise. In this case it is better to use the real distribution.

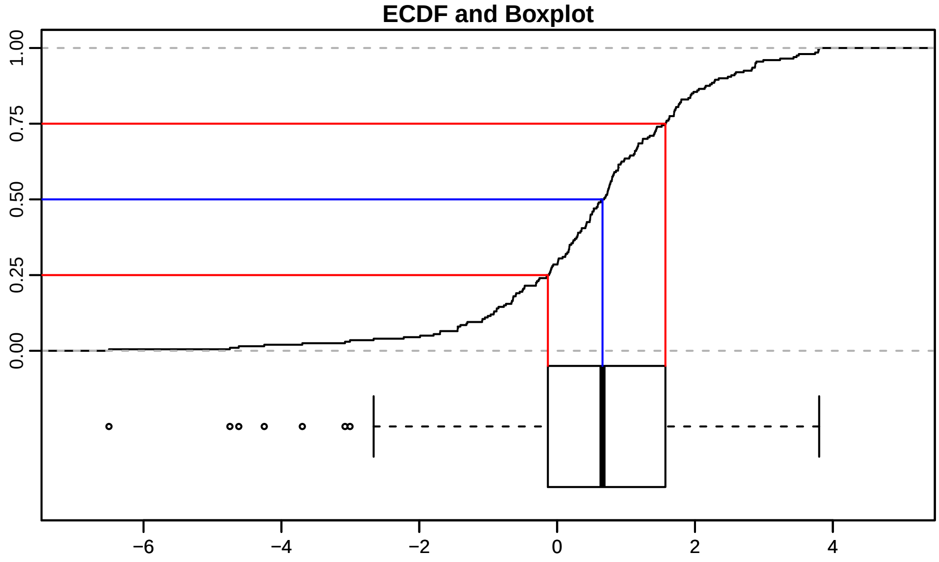
e) The manufacturer would like to define a critical bound K. If X lies below the bound the delivery is (with high probability) as desired and if X lies above K it is not. But as we saw in part (c), if he chooses $K = 3$ the probability that at most 3 test tubes are of minor value is only 25% even if the delivery is as desired. On the other hand, if he chooses K smaller than 3 the probability to reject the delivery becomes larger even if the delivery is as desired.

14. **a)**

$$a = 1 \qquad b = 3 \qquad c = 2 \qquad d = 4$$

b)

$$A = 2 \qquad B = 4 \qquad C = 3 \qquad D = 1$$



c)

15. **a)** The number of red ants X in the sample is binomially distributed with parameters $n = 5$ and $\pi = \frac{1}{10}$; hence we get

$$P[X = 3] = \binom{5}{3}\pi^3(1-\pi)^2 = 10 \cdot 0.1^3 \cdot 0.9^2 = 0.0081 \ .$$

c) The number Y of red ants is binomially distributed with parameters $n = 150$ and $\pi = 0.1$, hence we have $E[Y] = n\pi = 15$ and $Var(Y) = n\pi(1-\pi) = 13.5$.

d) sWe can approximate the distribution of Y by a normal one with the same mean and variance:

$$Y \approx \mathcal{N}\left(n\pi, n\pi(1-\pi)\right) = \mathcal{N}\left(\mu = 15, \sigma^2 = 13.5\right) \quad \Rightarrow \quad Z = \frac{Y - 15}{\sqrt{13.5}} \approx \mathcal{N}(0,1) \ .$$

Hence we can calculate the probability that Y is between 15 and 20 using the cumulative distribution function of the normal distribution:

$$P\left[15 \leq Y \leq 20\right] = P\left[0 \leq Z \leq \frac{20-15}{\sqrt{13.5}}\right] \approx \Phi(1.36) - \Phi(0)$$

With R, we get the probability with either of the following approaches:

```
> pnorm(1.36) - pnorm(0)
[1] 0.413085
> pnorm(20, 15, sqrt(13.5)) - pnorm(15, 15, sqrt(13.5))
[1] 0.4132159 (where the second term is 0.5 in both cases, of course. . . )
```

16. **a)** > B = 1000; n = 50; la = 2; exp.value = 0.5; count = 0; > set.seed(4672) # Set the random number generator to a starting point. > for (i in 1:B) # Simulate B-times... x = rexp(n,la) # ...n random variables from the exponential distribution... conf.int = c(mean(x) - qnorm(0.95)*sd(x)/sqrt(n), mean(x) + qnorm(0.95)*sd(x)/sqrt(n)) # ...and calculate the confidence interval using the formula: # estimate +/- 1.64*standard-error-of-estimate # (which applies because of the central limit theorem). if(exp.value > conf.int[1] & exp.value < conf.int[2]) # Check if the CI contains the true mean. count = count + 1 # If so set the count plus 1. > prob = count/B # Calculate the probability that in our simulations > # the CI contained the true mean, using > # (# CIs-incl-true-mean)/(total# CIs-created)

After simulating n exponentially distributed random variables for B times, we find that in 890 of the cases our confidence interval did contain the true mean. That's a probability of 0.89 which is near our confidence level of 0.9.

b) In this case we obtain in 764 of the cases a confidence interval including the true mean. The fact that this number is lower as in a) does not surprise us. For if we chose a sample size of only $n = 5$ we expect that our results (CIs) will vary a lot. This is because in our calculation of the confidence interval we use the normal approximation. However, for small sample sizes this is not really appropriate.

17. We start by reading in the data set and extracting the different variables:

```
> fracture <- read.table(”bone-fracture.csv”, sep = ”;”, header = TRUE)

> conc <- fracture$conc

> dif <- fracture$dif

> no.cells <- fracture$no.cells

> hit <- fracture$hit
```

We then load the packages needed for fitting and Q-Q plots, and define the significance level needed for the calculation of the confidence intervals:

```
> library(car)

> library(MASS)

> alpha <- 0.05
```

Variable `conc`

From a histogram (see below), we guess that the data is approximately normally distributed. Fitting a normal distribution yields:

```
> (fit.conc <- fitdistr(conc, ”normal”))

mean sd 3.41084484 0.38235027 (0.05407249) (0.03823503)
```

Note the numbers in brackets below the estimates: they denote the standard errors of the estimates. Hence we get lower and upper bounds of the 95% confidence intervals as follows:

```
> (conc.lower <- fit.conc$estimate - qnorm(1 - alpha/2)*fit.conc$sd)

mean sd
```

```
3.304865 0.307411
```

```
> (conc.upper <- fit.conc$estimate + qnorm(1 - alpha/2)*fit.conc$sd)
```

```
mean sd
```

```
3.5168250 0.4572895
```

In the above R-code sd is not the standard deviation but the estimated standard error.

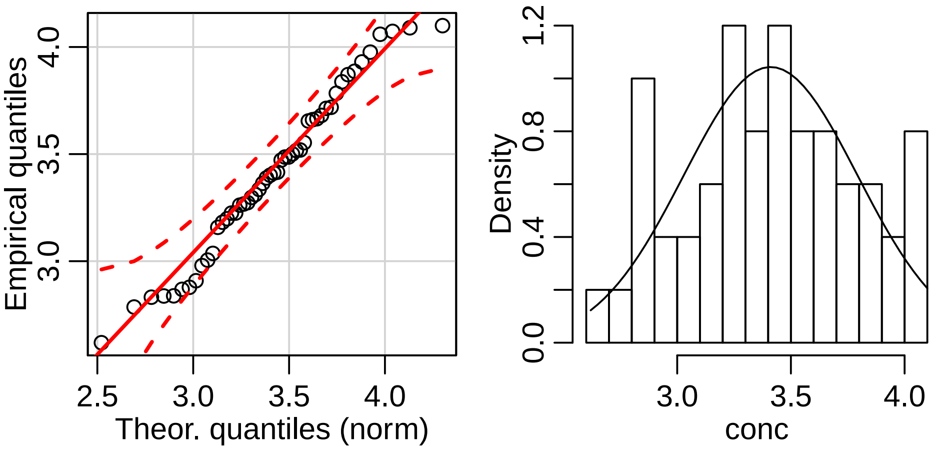
Checking the Q-Q plot and adding the estimated density into the histogram:

```
> par(mfrow = c(1, 2))
```

```
> qqPlot(conc, dist = ”norm”, mean = fit.conc$estimate[”mean”], sd = fit.conc$estimate[”sd”], xlab = ”Theor. quantiles (norm)”, ylab = ”Empirical quantiles”) > hist(conc, breaks = 20, freq = FALSE, main = ””)
```

```
> x.val <- seq(min(conc), max(conc), length = 50)
```

```
> lines(x.val, dnorm(x.val, mean = fit.conc$estimate[”mean”], sd = fit.conc$estimate[”sd”]))
```



Remark for experienced R users: to avoid copying, pasting and adapting the R code above for the next three variables, we write a function which generates the Q-Q plot and the histogram:

```
> plot.density <- function(x, estimate, dist, ...)  par(mfrow = c(1, 2)) do.call(qqPlot, c(list(x = x, dist = dist), as.list(estimate), xlab = sprintf(”Theor. quantiles (ylab = ”Empirical quantiles”)) hist(x, freq = FALSE, main = ””, ...) if (is.integer(x)) x.val <- seq(min(x), max(x)) else x.val <- seq(min(x), max(x), length.out = 50) lines(x.val, do.call(sprintf(”dc(list(x = x.val), as.list(estimate))))
```

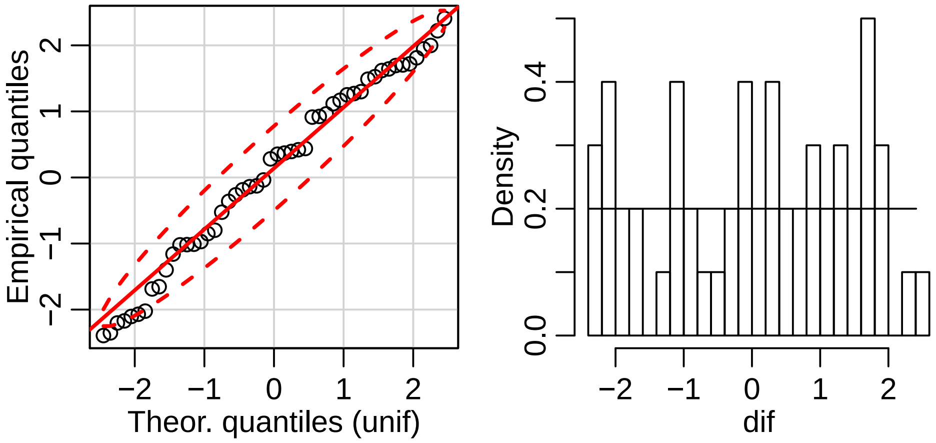
The plot above could then be generated with a single function call:

```
> plot.density(conc, fit.conc$estimate, ”norm”, breaks = 20, xlab = ”conc”)
```

Variable `dif`

Since no screw length in a 5 mm interval should be more likely than another one, we would expect the length differences to be approximately uniformly distributed in the interval $[-2.5\text{mm}, 2.5\text{mm}]$. There is no parameter to be estimated for the uniform distribution. We check our assumption with a Q-Q plot and plot the uniform density together with the histogram of the data:

```
> plot.density(dif, c(min = -2.5, max = 2.5), ”unif”, breaks = 20, xlab = ”dif”)
```

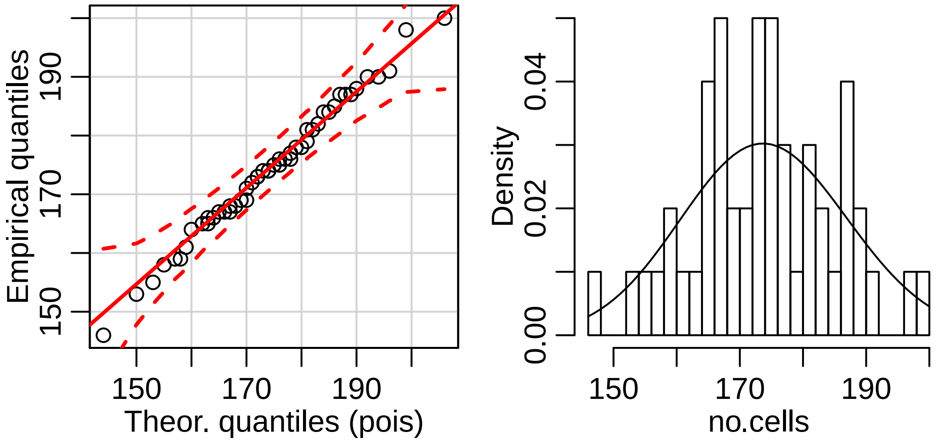


Variable `no.cells`
The distribution is concentrated around its mean. Since the data is discrete here, we fit a Poisson distribution and calculate the 95% confidence interval:

```
> (fit.cells <- fitdistr(no.cells, ”poisson”))

lambda

174.160000

( 1.866333)

> (cells.lower <- fit.cells$estimate - qnorm(1 - alpha/2)*fit.cells$sd)

lambda

170.5021

> (cells.upper <- fit.cells$estimate + qnorm(1 - alpha/2)*fit.cells$sd)

lambda

177.8179
```

Checking the Q-Q plot and adding the estimated density into the histogram:

```
> plot.density(no.cells, fit.cells$estimate, ”pois”, breaks = 20, xlab = ”no.cells”)
```



Variable `hit`
`hit` is a Bernoulli variable; we only have to estimate its probability of being 1:

```
> (p <- mean(hit))
```

```
[1] 0.68
```

The standard error of the arithmetic mean is given by $\sigma/\sqrt{n}$, where σ denotes the standard deviation of the samples and n the sample size. Hence we can calculate the 95% confidence level for p as follows:

```
> (p.lower <- p - qnorm(1 - alpha/2)*sd(hit)/sqrt(length(hit)))
```

```
[1] 0.5493891
```

```
> (p.upper <- p + qnorm(1 - alpha/2)*sd(hit)/sqrt(length(hit)))
```

```
[1] 0.8106109
```

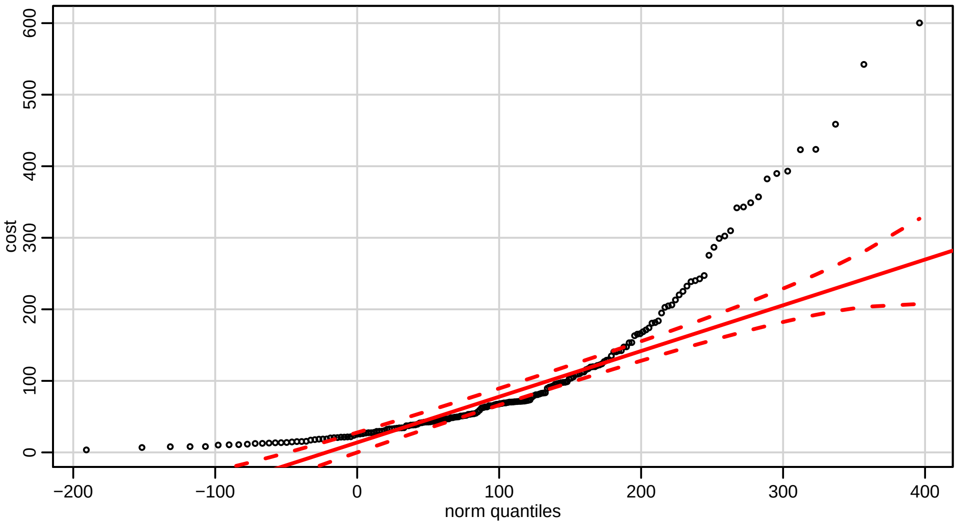
For Bernoulli variables, we can of course fit the empirical distribution exactly; a histogram for comparison is superfluous.

18. **a)** By looking at the shape of the histogram, we might guess that the data follow a poisson distribution. But this can not be since cost in CHF is a continuous variable which is rounded. Yet for a poisson distribution we need a variable which counts and only takes integers as values. So we try two other distributions (for continuous variables) we know: the exponential distribution and the normal distribution. We start with an automatic estimation of the parameters using the function `fitdistr()` from the package `MASS`:

```
> library(MASS)
> norm.fit <- fitdistr(cost, ”normal”)
```

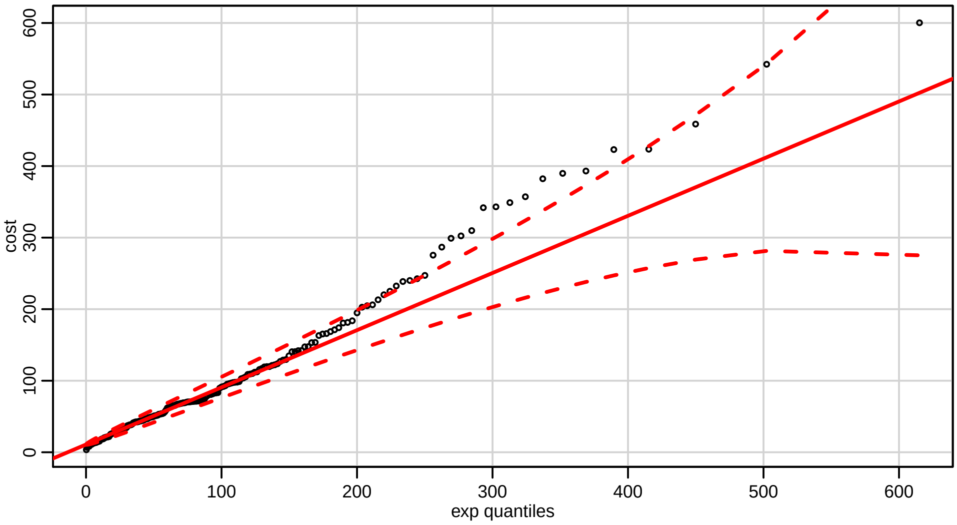
We start with testing the normal distribution. The Q-Q plot of the fitted distribution doesn't look good; this indicates that the choice of the normal distribution for this variable is not appropriate.

```
> library(car)
> qqPlot(cost, dist = ”norm”,
mean = norm.fit$estimate[”mean”],
sd = norm.fit$estimate[”sd”])
```



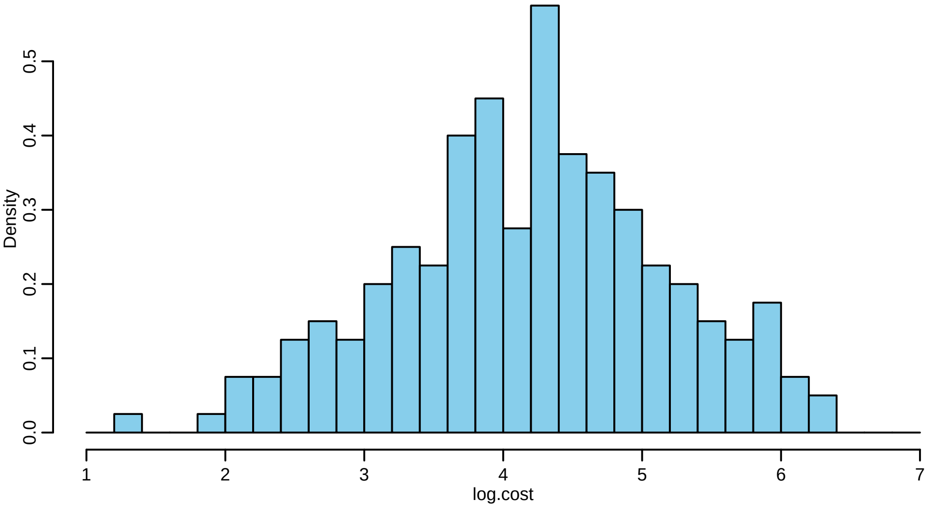
If we do the same for an exponential distribution we get:

```
> exp.fit <- fitdistr(cost, ”exponential”)
> qqPlot(cost, ”exp”, rate = exp.fit$estimate[”rate”])
```



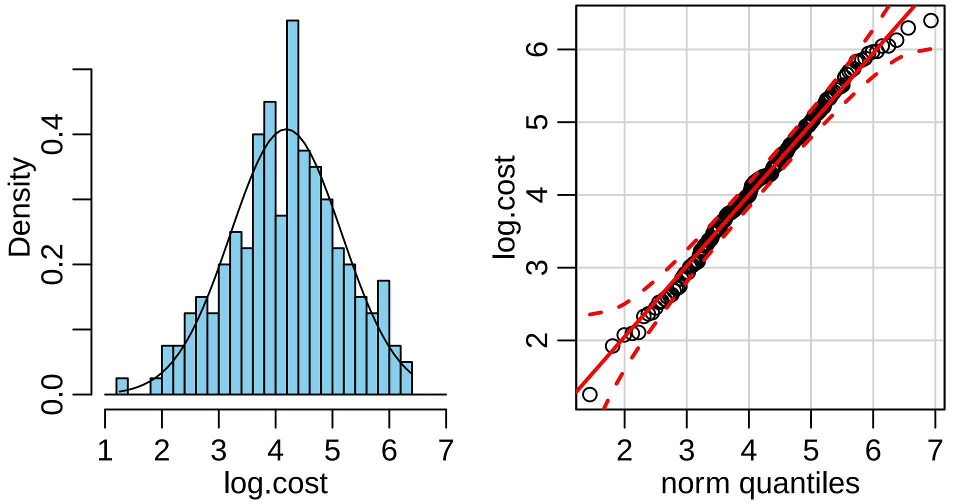
We see that the exponential distribution does not fit particularly well either.

b) After the log-transformation, the data looks like this:



We fit a normal distribution to the transformed data set and check its Q-Q plot. Since the log-normal distribution fits the initial data well, it is not surprising that the normal distribution fits the log-transformed data well, as can be seen from both plots:

> par(mfrow = c(1, 2))
> norm.fit <- fitdistr(log.cost, ”normal”)
> hist(log.cost, freq = FALSE, breaks = seq(1, 7, by = 0.2), col = ”skyblue”, main = ””)
> x.val <- seq(min(log.cost), max(log.cost), length = 50)
> lines(x.val, dnorm(x.val, mean = norm.fit$estimate[”mean”], sd = norm.fit$estimate[”sd”]))
> qqPlot(log.cost, dist = ”norm”, mean = norm.fit$estimate[”mean”], sd = norm.fit$estimate[”sd”])



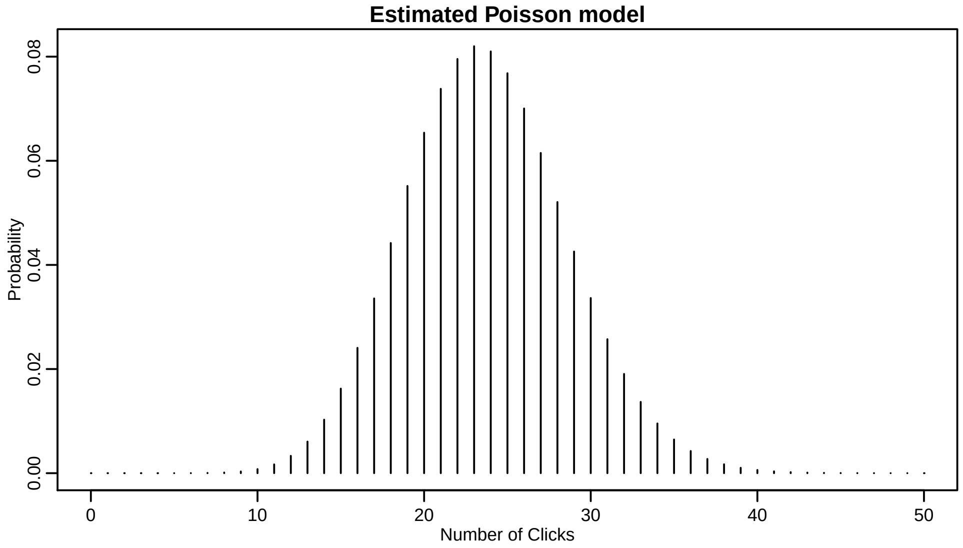
c) We can read off the parameters from the list `norm.fit`:

```
> norm.fit
mean sd
4.18541817 0.97764830
(0.06913017) (0.04888241)
```

Hence we have a normal distribution $\mathcal{N}(\hat{\mu}, \hat{\sigma}^2)$ with mean $\hat{\mu} = 4.185$ and standard deviation $\hat{\sigma} = 0.978$.

19. a) The parameter λ of the Poisson distribution can be estimated by the mean of the counts of a.v1, because the MLE is exactly the mean of $a.v1$ (shown below). This gives $\lambda = 23.708$. The Poisson distribution is stored in the function `dpois()`. To plot the expected values we use the function `dpois()` with the estimated λ and calculate the probabilities of each number of clicks between 0 and 50.

```
> la <- mean(d.pet$a.v1)
> x <- 0:50
> expected <- dpois(x,lambda = la)
> plot(x, expected ,type = ”h”, ylab = ”Probability”, xlab = ”Number of Clicks”, main = ”Esti-
mated Poisson model”)
```



In addition, let us show that the maximum likelihood estimator for the parameter λ is indeed the sample mean $\bar{x}$. The log-likelihood $l(\lambda)$ is given by

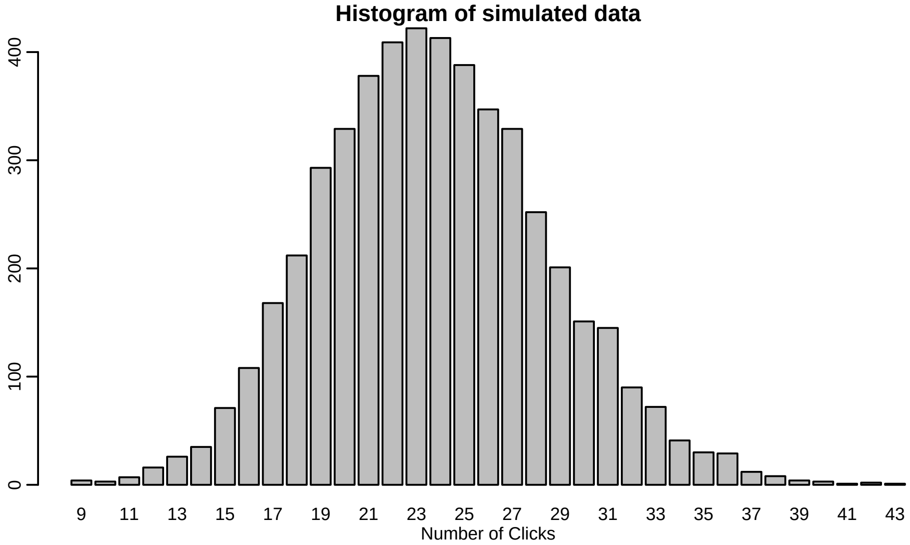
$$\begin{aligned} l(\lambda) &= \log \prod_{i=1}^{n} p(x_i;\lambda) = \sum_{i=1}^{n} \log(p(x_i;\lambda) \\ &= \sum_{i=1}^{n} \log(e^{-\lambda}\frac{\lambda^{x_i}}{x_i!}) \\ &= \sum_{i=1}^{n} \left[x_i \log(\lambda) - \lambda - \log(x_i!)\right] \\ &= \log(\lambda)n\bar{x} - \lambda n - \sum_{i=1}^{n} \log(x_i!). \end{aligned}$$

The MLE $\hat{\lambda} = \arg\max_{\lambda}(l(\lambda))$ can now be calculated by

$$\begin{aligned} l'(\lambda) &= \frac{n\bar{x}}{\lambda} - n \overset{!}{=} 0 \\ &\Rightarrow \hat{\lambda} = \bar{x}. \end{aligned}$$

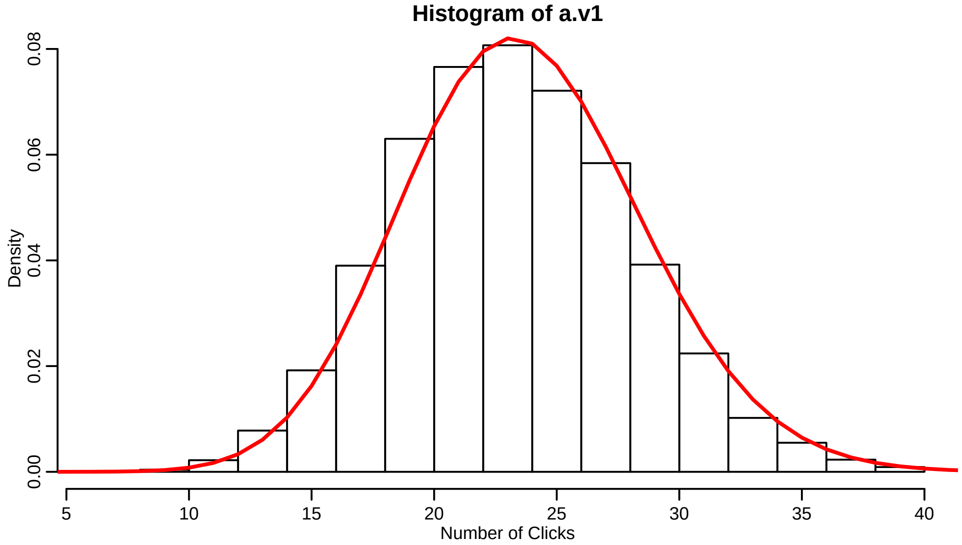
b) The function `rpois()` generats random numbers from a Poisson distribution. The estimated λ and the length of the series can be set as arguments.

```
> n <- nrow(d.pet) # Number of observations
> set.seed(4892) # Set seed for the random number generator.
> sim <- rpois(n, lambda = la) # Simulating a new series of counts
> barplot(table(sim), xlab=”Number of Clicks”, main=”Histogram of simulated data”)
```



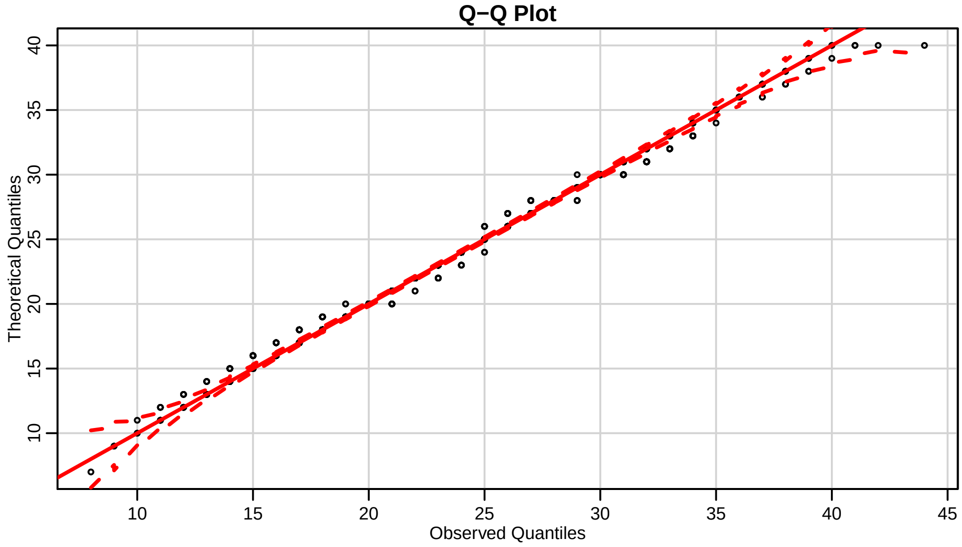
c) First we draw a histogram of the observed counts of `av.1`. The histogram can be scaled to probabilities with the additional option `probability = TRUE`. Afterwards we can add the curve of estimated counts with the function `lines()` .

```
> hist(d.pet$a.v1,probability = TRUE, main = ”Histogram of a.v1”, xlab = ”Number of Clicks”)
> x <- 0:50
> lines(x,dpois(x,lambda = la),col = ”red”, lwd = 2)
```



The red curve fits the histogram very well. Hence, the Poisson model describes the data well. A second possibility for checking the goodness of fit is the quantile-quantile plot (QQ-Plot). In this plot the quantiles of the observed counts are plotted against the quantiles of the counts from the Poisson model. If model and data fit well, we should see a straight line.

> library(”car”)
> sim.dist <- rpois(n, lambda= la)
> qqPlot(d.pet$a.v1,dist = ”pois”,lambda = la,main=”Q-Q Plot”, ylab = ”Theoretical Quantiles”, xlab = ”Observed Quantiles”)



The points fall along the line and into the dotted confidence region. This indicates a good fit too.

20. **a)** Note that the number of cured patients is a binomially distributed random variable, $X \sim \mathrm{Bin}(n = 10, \pi = 0.3)$. Hence:

$$P(X = k) = \binom{10}{k} 0.3^k 0.7^{n-k}$$

$$P(X \leq 2) = P(X = 0) + P(X = 1) + P(X = 2) = 0.7^{10} + \binom{10}{1} 0.3^1 0.7^9 + \binom{10}{2} 0.3^2 0.7^8 = 0.38$$

The solution can be derived in R in the following way:
> n <- 10
> pi <- 0.3
> dbinom(0,n,pi) + dbinom(1,n,pi) + dbinom(2,n,pi)

```
[1] 0.3827828
> # Or directly using:
> pbinom(2, n, pi)
[1] 0.3827828
```

b) 1. **Model:** X is the number of patients which where successfully treated, $X \sim \mathrm{Bin}(10, \pi)$.

2. **Null hypothesis:** $H_0 : \pi = 0.3$
   Alternative hypothesis: $H_\mathrm{A} : \pi > 0.3$

3. **Test statistic:** X: number of cured patients.
   Distribution under H_0: $X \sim \mathrm{Bin}(10, 0.3)$

4. **Significance level:** $\alpha = 0.01$

5. **Region of rejection** (note: one-sided test):
   We look for set $K = \{...\}$ such that $P_{H_0}[X \in K] \leq \alpha$.

	$x = 4$	$x = 5$	$x = 6$	$x = 7$	$x = 8$	$x = 9$
$P(X \geq x)$	0.1503	0.0473	0.0106	0.0016	0.0001	5.9×10^{-6}

Therefore the rejection range is $K = \{7, 8, 9, 10\}$.
The probabilities listed in the table can be calculated in R in the following way:

```
> n=10
> pi=0.3
> 1-pbinom(4:9,n,pi)
[1] 0.1502683326 0.0473489874 0.0105920784 0.0015903864
[5] 0.0001436859 0.0000059049
```

6. **Test decision:** Since $4 \notin K$, H_0 can't be rejected. Therefore we can't proof that the success rate of the new drug is better.

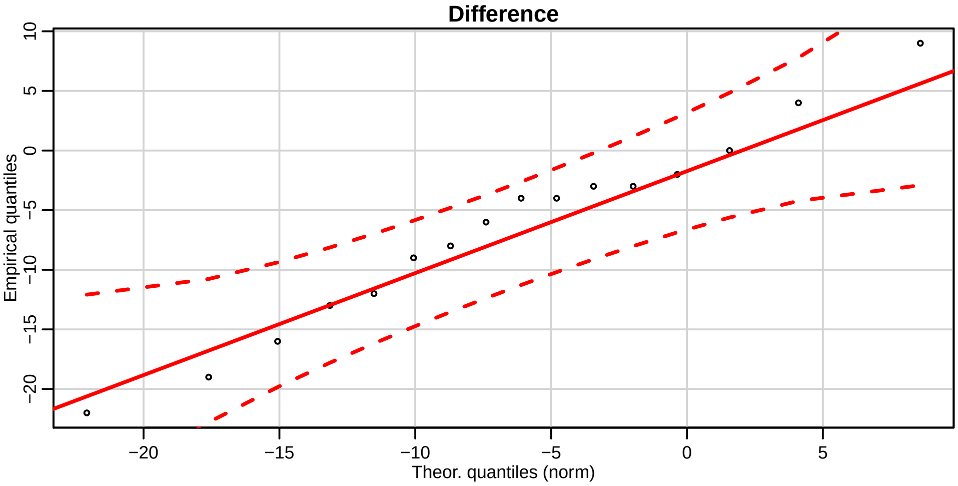
c) The power of the test is the probability that the null hypothesis is rejected, if the alternative hypothesis is true: $P_{H_A}(X \in K)$.
(Alternatively: power $= 1 - P(\text{type II error}) = 1 - P_{H_A}(X \notin K)$)
In our case: we reject H_0 if $X \in \{8, 9, 10\}$ and if H_A is true, then $X \sim \mathrm{Bin}(10, 0.6)$. So:

$$\text{power} = \binom{10}{8} 0.6^8 0.4^2 + \binom{10}{9} 0.6^9 0.4 + \binom{10}{10} 0.6^{10} 0.4^0 = 0.1672898$$

The code in R could be the following:

```
> n=10
> pi=0.6
> dbinom(8,n,pi) + dbinom(9,n,pi) + dbinom(10,n,pi)
[1] 0.1672898
> 1-pbinom(7,n,pi)
[1] 0.1672898
```

21. Our variable of interest (voi) constitutes the difference in time during which arm elevation is below 30 degrees. Each subject is tested once before and once after the change in working conditions, so we have to perform a *paired-sample* (or *one-sample*) test on these differences. Since we want to know whether the time during which arm elevation is below 30 degrees changes into either direction, we have to perform a *two-sided* test. Our expectation is that there is no difference, and that our voi is a normally distributed variable. We check this assumption by considering the Q-Q Plot. In R, this is a one-liner after initialization of the data: > before <- neckshoulder$before > after < -neckshoulder$after > dif <- after - before > library(car) > qqPlot(dif, dist = "norm", mean = mean(dif), sd = sd(dif), xlab = "Theor. quantiles (norm)", ylab = "Empirical quantiles", main =



”Difference”)

As all points are within the boundaries, we can now perform our paired-sample, two-sided t-test: > t.test(after, before, alternative = ”two.sided”, paired = TRUE, conf.level = 0.9) Paired t-test

data: after and before t = -3.2791, df = 15, p-value = 0.005072 alternative hypothesis: true difference in means is not equal to 0 90 percent confidence interval: -10.358687 -3.141313 sample estimates: mean of the differences -6.75 Our result, $t = -3.28$ has a p-value below 0.05, suggesting that there is a significant lessening of time during which arm elevation is below 30 degrees under the new working conditions.
For a better understanding of the test statistic, it may be helpful to perform the test “manually”, i.e. without using the function `t.test`. The test statistic is calculated as follows:

> (t.stat <- sqrt(length(dif))*mean(dif)/sd(dif))

[1] -3.279057 (Note that we get the same value as in the automatic calculation.)

For a paired test on a significance level of 10%, the region of rejection lies below the 5% quantile and above the 95% quantile of the null distribution of the test statistic. In our case, the null hypothesis is rejected if $|T| > t_{n-1,0.95}$; this is indeed the case:

> (qt(0.95, length(dif) - 1))

[1] 1.75305

> (abs(t.stat) > qt(0.95, length(dif) - 1))

[1] TRUE

22. **a)** 1. **Model:** X: Number of test tubes of lower quality in a sample of 50 tubes. $X \sim \text{Bin}(50, \pi)$.

2. **Null hypotheses:** $H_0 : \pi = 0.1$
Alternative hypotheses: $H_A : \pi < 0.1$
The manufacturer wants to make sure that he does *not* violate his promise of at most 10% low quality tubes. Hence he must set up null and alternative hypothesis such that he can *hope* to reject the null hypothesis, i.e. such that the alternative hypothesis is to his benefit. By choosing the alternative hypothesis $\pi < 0.1$, he knows that if he rejects the null hypothesis, he will do this *falsely* with a probability of only 5% ($= \alpha$); so he can control the probability of “falsely feeling safe”.

3. **Test statistic:** T: Number of test tubes of lower quality in a sample of 50 tubes.
Distribution of T under H_0: $T \sim \text{Bin}(50, 0.1)$

4. **Significance level:** $\alpha = 0.05$

5. **Range of rejection:** If H_0 holds, we have:

$$\begin{array}{lcl@{\qquad\qquad}lcl}
P(X = 0) & = & 0.0052 & P(X \leq 0) & = & 0.0052 \\
P(X = 1) & = & 0.0286 & P(X \leq 1) & = & 0.0338 \\
P(X = 2) & = & 0.0779 & P(X \leq 2) & = & 0.1117
\end{array}$$

The range of rejection K at the significance level $\alpha = 0.05$ should be such that $P[X \leq \max(K)] \leq \alpha = 0.05$ holds. Therefore, $K = \{0, 1\}$.

6. **Test decision:** The observed value was $x = 3$. This does not lie in the range of rejection and therefore we cannot reject H_0 in favour of H_A on the significance level $\alpha = 0.05$.

b) We need to calculate $P[X \leq 1]$ in case of $\pi = 0.075$.

```
> pbinom(1, 50, 0.075)
[1] 0.1025006
```

Even if the delivery contains 7.5% tubes of lower quality, only in 10% of the cases can we proof that fact with our test with sample size $n = 50$.

c) We want to know what happens in our model for increasing sample size n, to obtain the required power. For our model we have $\pi = 0.075$ and $X(n)$: the number of tubes of lower quality in a sample of n, distributed according to $\text{Bin}(n, 0.075)$. Furthermore, our test statistic $T(n) \sim \text{Bin}(n, 0.1)$, because under H_0: $\pi = 0.1$. The range of rejection $K(n) = \{1, 2, \dots, k(n)\}$ depends on n as well. So when we increase our sample size, critical value $k(n)$ increases too. In fact, $k(n)$ is the 5% quantile of a binomial distribution with size n and probability 0.1.

Now we want to chose n as small as possible but giving a power of (at least) 50% when $\pi = 0.075$. Mathematically, this is equivalent to chosing the smallest n for which

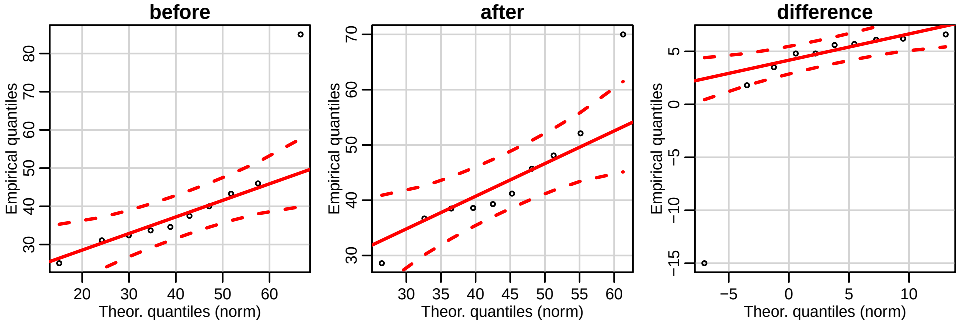
$$P[X(n) \leq k(n)] = \sum_{i=0}^{k(n)} \binom{n}{i} 0.075^{n-i}(1 - 0.075)^i \geq 0.5$$

holds. We solve this inequality numerically and use the fact that the probability is growing in n.

```
> n = 50
> while(pbinom(qbinom(0.05,n,0.1),n,0.075)<0.5)
n = n + 1
if(n == 10000) break
> n
[1] 275
```

23. **a)** We use a Q-Q plot for all variables:

```
> library(car)
> par(mfrow = c(1, 3), cex = 0.6)
> qqPlot(training$before, dist = ”norm”,
mean = mean(training$before), sd = sd(training$before),
xlab = ”Theor. quantiles (norm)”, ylab = ”Empirical quantiles”, main = ”before”)
> qqPlot(training$after, dist = ”norm”,
mean = mean(training$after), sd = sd(training$after),
xlab = ”Theor. quantiles (norm)”, ylab = ”Empirical quantiles”, main = ”after”)
> dif <- trainingafter − trainingbefore
> qqPlot(dif, dist = ”norm”,
mean = mean(dif), sd = sd(dif),
xlab = ”Theor. quantiles (norm)”, ylab = ”Empirical quantiles”,
main = ”difference”)
```



Apart from one clear outlier (which is already visible in the table), the data can be assumed to be normally distributed.

b) We can not use the t-test since we have an outlier in the Q-Q plot of the differences. We can not use the Wilcoxon test either, because our data is not symmetric around the median. So we use the sign-test. This can be done in the following way:

1. **Model:** $X_1, \ldots, X_n$: muscle activation of pilot i *before* the training
   $Y_1, \ldots, Y_n$: muscle activation of the same pilot *after* the training
   $D_i := Y_i - X_i$: difference in muscle activation
   Model assumption: $D_1, \ldots, D_n$ arbitrarily distributed with expectation value μ.
2. **Null hypothesis:** $H_0 : \mu = \mu_0 = 0$
   Alternative hypothesis: $H_\mathrm{A} : \mu > \mu_0$
3. **Test statistic:** $V = \#\{i | D_i > 0\}$: number of values larger 0.
   Distribution of V under H_0: $V \sim Bin(n, 0.5)$
4. **Significance level:** $\alpha = 0.05$
5. **Region of rejection** (note: one-sided test): Define $K = [n-c, n]$ such that $P_{H_0}[V \in K] \leq \alpha$. Due to the discrete nature of the sign test, c is determined by the binomial distribution: $P_{H_0}[V \in K] = P_{H_0}[V \geq n - c] = 1 - P_{H_0}[V \leq n - c - 1]$. Here $n - c - 1$ takes the value 8, therefore $K = [9, 10]$. This can be checked with R in the following way:
   ```
   > n <- 10
   > pi <- 0.5
   > pbinom(5:10,n,pi,lower.tail=FALSE) # = 1 - pbinom(5:10, n, pi)
   [1] 0.3769531250 0.1718750000 0.0546875000 0.0107421875
   [5] 0.0009765625 0.0000000000
   ># In the results, check for the value smaller than or equal to alpha: this is n-c-1
   ```
6. **Test decision:** Reject H_0 if $V \in K$, otherwise keep it.
   We have $V = 9$, which is in K, so we reject H_0.
   We can also let R do the work for us in the following way:
   ```
   > n <- 10
   > V <- sum(trainingafter > trainingbefore)
   > binom.test(V, n, p = 0.5, alternative = ”greater”, conf.level = 0.95)
   Exact binomial test
   data: V and n
   number of successes = 9, number of trials = 10,
   p-value = 0.01074
   alternative hypothesis: true probability of success is greater than 0.5
   95 percent confidence interval:
   0.6058367 1.0000000
   sample estimates:
   probability of success
   0.9
   ```
   So we see that our alternative hypothesis is true with certainty 0.95.

c) The published report is misleading. Performing a test at a significance level of 5% means that there is a chance of a type I error (“false positive”) of 5%; this means that in 20 independent tests where the null hypothesis is true, we have to expect one false refection of the null hypothesis. Hence we should not believe the one positive test outcome in 20 tests on the 5% level. Instead, the only training programme that showed a significant effect should be tested again, for example with the rest of the pilots.

24. **a)** The simplest way to read in the file is to remove the string “`10:`” at the beginning; we assume that `palindromes-preprocessed.txt` is the file preprocessed that way. This file can then be read into R with the function `scan`:

```
> positions <- scan("palindromes-preprocessed.txt", sep = ",")
```

Note: the preprocessing can be done manually in an editor, or with a single bash line in Linux:

```
$ sed -e s/^10: // palindromes.txt > palindromes-preprocessed.txt
```

b) $X_i \overset{\text{i.i.d.}}{\sim} \text{Pois}(\lambda)$: A palindrome in a bin can be seen as a rare “event” in a given region (of DNA); this is modeled by a Poisson distribution. As the number of palindromes in one bin is not influenced by number of palindromes in another bin, the count numbers in different bins are independent. Finally, all bins should on average have the same number of palindromes. Altogether, we get i.i.d. Poisson-distributed random variables.

c) There are many ways to solve this problem in R. Here two possible solutions shall be presented.

The easier way (but less efficient to run in R) to solve the problem is the following:
Define `dna.length` as the number of base pairs. Create the bins by dividing `dna.length` into intervals of length 4000. Use a for-loop to select the palindrome centers that fall within each bin, and sum their number to obtain the count (i.e. our potentially Poisson distributed random variable).

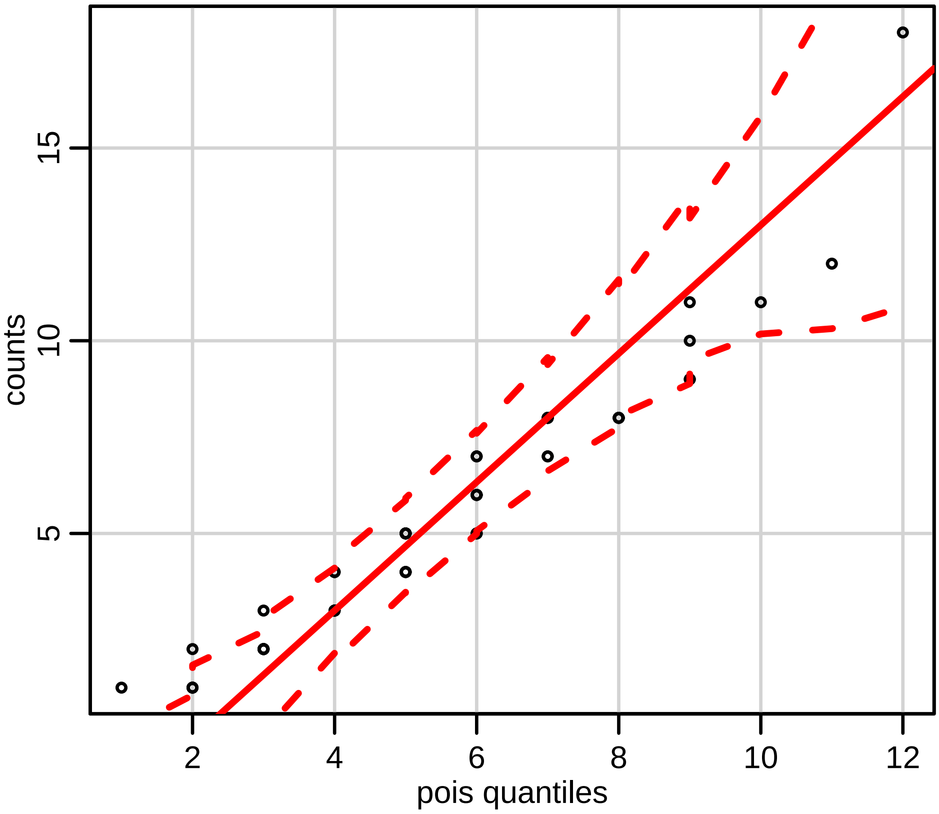
```
> dna.length <- 235727
> n.bins <- ceiling(dna.length/4000)
> x <- 4000
> counts <- rep(NA, n.bins)
> for (i in 1:n.bins) {
    counts[i] <- sum(positions > x*(i-1) & positions <= x*i)
 }
```

Another way would be to use a function to count the number of palindroms in each sequence. In R this might look like this:

```
> dna.length <- 235727
> n.bins <- ceiling(dna.length/4000)
> breaks <- seq(0, dna.length, by = 4000)
> counts <- sapply(breaks,
    function(a) sum(positions > a & positions <= a + 4000))
```

To fit the Poisson distribution to our counts, we estimate λ by $\hat{\lambda}_{\text{ML}} = \overline{X} =$ `mean(counts) = 5.576`. Looking at the Q-Q plot, the Poisson distribution appears appropriate for our data.

```
> library(car)
> qqPlot(counts, dist = "pois", lambda = mean(counts))
```



d) We want to find regions with more palindromes than could reasonably be expected, so we want to perform a one-sided hypothesis test. Be aware that the test outlined below is only for one bin, and has to be repeated for every bin. However only step 6 differs. We use again $\hat{\lambda}_{\text{ML}} = \overline{X} =$ `mean(counts)`

1. **Model:** X: the number of palindromes in one bin. $X \sim \text{Pois}(\lambda)$.
2. **Null hypotheses:** $H_0 : \lambda = \lambda_0 = 5.576$
   The value of λ_0 under the null hypothesis equals the maximum likelihood estimator for λ given the counts in the 59 bins:

```
> (lambda0 <- mean(counts))
[1] 5.576271
```

Alternative hypotheses: $H_A : \lambda > \lambda_0$

3. **Test statistic:** X: Number of palindromes in one bin.
   Distribution of X under H_0: $X \sim \text{Pois}(\lambda_0)$.
4. **Significance level:** $\alpha = 0.0005 = 0.05\%$
5. **Range of rejection:** If H_0 holds, we have:

	$x = 13$	$x = 14$	$x = 15$	**x=16**	$x = 17$
$P(X \geq x)$	$4.9724 \cdot 10^{-3}$	$1.9073 \cdot 10^{-3}$	$6.8648 \cdot 10^{-4}$	$2.3263 \cdot 10^{-4}$	$7.4455 \cdot 10^{-5}$

   In R, we can calculate these numbers as follows:

```
> 1 - ppois(12:16, lambda = mean(counts))
[1] 4.972446e-03 1.907333e-03 6.864835e-04 2.326309e-04
[5] 7.445554e-05
```

   The range of rejection K at the significance level $\alpha = 0.0005$ should be such, that $P(X \geq min(K)) \leq 0.0005$ holds. Therefore, $K = \{16, 17, 18, ...\}$. So any bin having 16 or more palindromes would be significant.

6. **Test decision:** The observed values are stored in our variable `counts`. Check if in any bin there is a number of 16 or higher:

```
> which(counts >= 16)
[1] 24
```

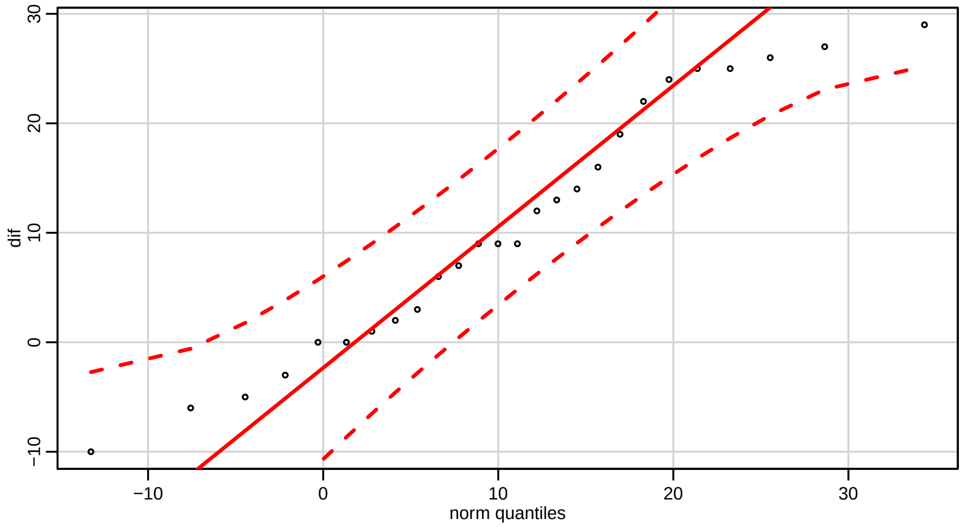
We find that there are too many palindromes in bin 24 (there are 18 palindromes). So for this bin we reject H_0 in favour of H_A on the significance level $\alpha = 0.0005$. For all the other bins we cannot reject H_0.

e) It would be a good idea for the biologist to have a look at the sequence in bin 24: under the assumption that the palindromes occur just by chance (and are thus uniformly distributed over the genome), the probability to find the 18 palindromes of bin 24 is below 0.05%. This is a strong indication that the cluster of palindromes there does not just occur by chance, but has a biological meaning.

f) If we chose a significance level of 5%, the expected number of type I errors when testing 59 bins would be 2.95. So just by set-up alone, we might well encounter several false positives. However, with a level of significance of 0.05%, the expected number of type I errors is only 0.0295. So when we now reject the null hypothesis for a bin, we are far more certain that for this bin the alternative hypothesis is truly more appropriate, rather than this outcome being a mere by-product of doing so many tests.

25. a) We have to use a *paired* test to compare the pulse of the 26 astronauts before and after their flight. If the pulse differences are normally distributed, we can use a t-test; so let's check this assumption first:

> library(car)
> dif <- astro$after - astro$before
> qqPlot(dif, dist = ”norm”, mean = mean(dif), sd = sd(dif))



The assumption of the normal distribution seems to hold, hence we can perform a t-test.

1. **Model:** $X_1, \dots, X_n$: pulse of astronaut i *before* the flight
   $Y_1, \dots, Y_n$: pulse of the same astronaut *after* the flight
   $D_i := Y_i - X_i$: pulse differences

   Model assumption: $D_1, \dots, D_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mu, \sigma^2)$.
2. **Null hypothesis:** $H_0 : \mu = \mu_0 = 0$
   Alternative hypothesis: $H_A : \mu \neq \mu_0$
3. **Test statistic:** $T = \dfrac{\sqrt{n}(\overline{D} - \mu_0)}{\hat{\sigma}}$
   Distribution of the test statistic under H_0: $T \sim t_{n-1}$
4. **Significance level:** $\alpha = 0.05$
5. **Region of rejection** (note: two-sided test):

$$K = (-\infty, -t_{n-1;1-\alpha/2}] \cup [t_{n-1;1-\alpha/2}, \infty) = (-\infty, -2.06] \cup [2.06, \infty)$$

6. **Test decision:**

```
> t.test(astro$after,
astro$before,
paired = TRUE,
alternative = ”two.sided”)
Paired t-test
data: astro$after and astro$before
t = 4.6725, df = 25, p-value = 8.707e-05
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
5.893362 15.183561
sample estimates:
mean of the differences
10.53846
```

The pulse is significantly different before and after the flight.

b) *Different* astronauts have simulated the flight with or without salt; hence we must use an *unpaired* test this time. We choose an unpaired t-test.

1. **Model:** $X_1, \dots, X_n$: pulses of astronauts who did *not* take salt
   $Y_1, \dots, Y_m$: pulses of astronauts who did take salt

   Model assumption: $X_1, \dots, X_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mu_X, \sigma_X^2), Y_1, \dots, Y_m \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mu_Y, \sigma_Y^2)$.
2. **Null hypothesis:** $H_0 : \mu_X = \mu_Y$
   Alternative hypothesis: $H_A : \mu_X \neq \mu_Y$
3. **Test statistic:** $T = \dfrac{\overline{X} - \overline{Y}}{s_{pool}\sqrt{1/n + 1/m}}$ for $s_{pool}^2 = \dfrac{(n-1)s_X^2 + (m-1)s_Y^2}{n+m-2}$
   Distribution of the test statistic under H_0: $T \sim t_{n+m-2}$

4. **Significance level:** $\alpha = 0.05$
5. **Region of rejection** (note: two-sided test):

$$K = (-\infty, -t_{n+m-2;1-\alpha/2}] \cup [t_{n+m-2;1-\alpha/2}, \infty) = (-\infty, -2.064] \cup [2.064, \infty)$$

6. **Test decision:**

```
> t.test(astro$after[astro$salt == 1],
astro$after[astro$salt == 0],
paired = FALSE,
alternative = ”two.sided”) Welch Two Sample t-test
data: astro$after[astro$salt == 1] and astro$after[astro$salt == 0]
t = -2.2528, df = 12.045, p-value = 0.0437
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-21.4417422 -0.3621793
sample estimates:
mean of x mean of y
63.76471 74.66667
```

The pulse is significantly different with and without salt.

26. **a)** The sample is **unpaired**, because to each male does not belong a particular female. Also, the number of male and female jackals need not necessarily be equal.

b) We perform a **two-sided, unpaired** ***t*****-test:**

1. **Model:** $X_1, \ldots, X_{10}$ jaw lenght of male jackals,
   $Y_1, \ldots, Y_{10}$ jaw lenght of female jackals,
   with distributions:
   $X_i \overset{\text{i.i.d}}{\sim} \mathcal{N}(\mu_x, \sigma_x^2)$,
   $Y_j \overset{\text{i.i.d}}{\sim} \mathcal{N}(\mu_y, \sigma_y^2)$,
   X_i, Y_j independent.
2. **Null hypothesis:** H_0: $\mu_x = \mu_y$
   Alternative hypothesis: H_A: $\mu_x \neq \mu_y$
3. **Test statistic:**

$$\begin{aligned} s_{pool} &= \sqrt{\frac{1}{n+m-2}\Big((n-1)s_x^2 + (m-1)s_y^2\Big)} \\ &= \sqrt{\frac{1}{18}\left(9 \times 13.82 + 9 \times 5.16\right)} = 3.08 \\ T &= \frac{\bar{X} - \bar{Y}}{s_{pool}\sqrt{1/n + 1/m}} \\ &= \frac{113.4 - 108.6}{3.08 \times \sqrt{2/10}} = 3.48 \end{aligned}$$

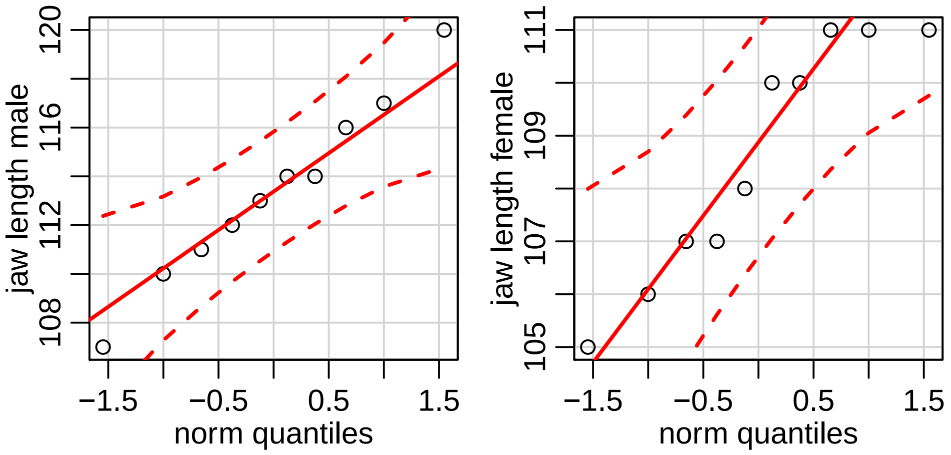
   Under H_0: $T \sim t_{n+m-2}$, so here $T \sim t_{18}$.
4. **Significance level:** $\alpha = 5\%$
5. **Range of rejection:** $t_{18,0.975} = 2.1$, so
   $K = \{|T| > 2.1\}$.
6. **Test decision:** Because $T = 3.48 > 2.1$, the value of the test statistic lies inside K. So we reject the null hypothesis with 95% confidence.

c) We get the **R**-output:

```
> jackals <- read.table(”jackals.dat”, header=TRUE)
> t.test(jackals$m, jackals$f)
Welch Two Sample t-test
data: jackals$m and jackals$f
t = 3.4843, df = 14.894, p-value = 0.00336
alternative hypothesis: true difference in means is not equal to 0
```

```
95 percent confidence interval:
1.861895 7.738105
sample estimates:
mean of x mean of y
113.4 108.6
```

The p-value is $0.0034 < 0.05$, so the null hypotheses can (again) be rejected.



27. **a)** Create a vector with male and one with female values:

```
> jackals <- read.table(”jackals.dat”, header=TRUE)
> jackals.all <- c(jackals$m, jackals$f)
```

The follwing function calculates the value of the test statistic assuming the jackals in `group1` are the males:

```
> mean.diff <- function(group1) mean(jackals.all[group1]) - mean(jackals.all[-group1])
> n <- length(jackals$m) # number of values from male jackals.
> m <- length(jackals$f) # number of values from female jackals.
```

Calculate the value of the test statistic for the real data (i.e. using the true sex of the jackals):

```
> (D <- mean.diff(1:n))
[1] 4.8
```

Finally, calculate mean samples by randomly assigning sexes to jackals:

```
> set.seed(42)
> N <- 9999 # Set the number of permutations.
> # Calculate N test statistic values under the null hypothesis (permutations)
> D.sample <- replicate(N,mean.diff(sample.int(n+m,n)))
> (p.value <- (sum(D <= D.sample) + 1)/(N+1)) # Calculate the p- value.
[1] 0.0011
```

Note: since the p-value is so small, you might get a p-value of 0 if you choose N too small: in that case you did not find any value that exceeded the actual one by chance.

b)

```
> library(”perm”) # Load the package ”perm” after the installation.
> permTS(jackals$m, jackals$f, alternative=”greater”, method = ”exact.mc”)
Exact Permutation Test Estimated by Monte Carlo
data: jackalsmandjackalsf
p-value = 0.001
alternative hypothesis: true mean jackalsm − meanjackalsf is greater than 0
sample estimates:
mean jackals$m - mean jackals$f 4.8
p-value estimated from 999 Monte Carlo replications
99 percent confidence interval on p-value:
0.000000000 0.005289582
```

Both methods need not necessarily give exactly the same p-value due to the randomness in the generated permutations. By setting the argument `method = "exact.mc"`, we make sure that `permTS` uses the same methods as we implemented in a).

c)

```
> t.test(jackals$m,jackals$f,alternative=”greater”)
Welch Two Sample t-test
data: jackals$m and jackals$f
t = 3.4843, df = 14.894, p-value = 0.00168
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
2.38387 Inf
sample estimates:
mean of x mean of y
113.4 108.6
```

All three p-values are reasonable small and do not differ much.

d) The t-test should be preferred in this case because the data can be assumed to be normal distributed as we can see from the Q-Q plots (for male and female jackals separately). If it is not possible to check the normal assumption adequately one should use a permutation test, despite the test having lesser power.

28. a)

```
> counts <- scan(”counts.txt”)
> lambda0 <- mean(counts)
> p.val <- 1 - ppois(counts - 1, lambda = lambda0)
```

b) We consider all bins as significantly enriched with palindromes that have a p-value less than or equal to 0.05.

```
> which(p.val <= 0.05)
[1] 24 42 48 51
```

c) With the Bonferroni method only one p-value remains significant:

```
> p.adj.bonf <- p.adjust(p.val, method = ”bonferroni”)
> head(sort(p.adj.bonf))
[1] 0.001331719 0.714971653 1.000000000 1.000000000 [5] 1.000000000 1.000000000
> which(p.adj.bonf <= 0.05)
[1] 24
```

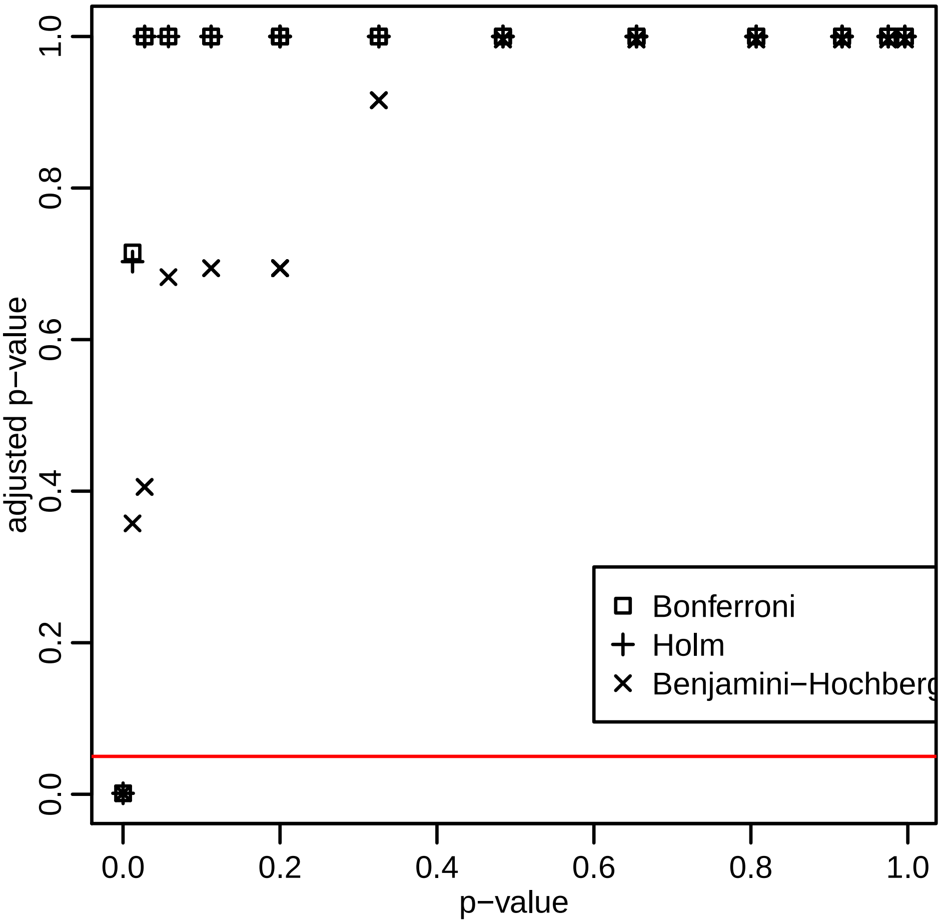
d) With the Holm method also just one bin returns a significant p-value:

```
> p.adj.holm <- p.adjust(p.val, method = ”holm”)
> head(sort(p.adj.holm))
[1] 0.001331719 0.702853490 1.000000000 1.000000000 [5] 1.000000000 1.000000000
> which(p.adj.holm <= 0.05)
[1] 24
```

e)

```
> plot(p.val, p.adj.bonf, pch = 0, xlab =”p-value”, ylab=”adjusted p-value”)
> points(p.val, p.adj.holm, pch = 3)
> points(p.val, p.adj.fdr, pch = 4)
> abline(h=0.05,col=”red”)
```

> legend(x = 0.6, y = 0.3, legend = c(”Bonferroni”,”Holm”,”Benjamini-Hochberg”), pch = c(0,3,4))



In this example all three methods give the same significant bins. However, one should chose between the Holm or Benjamini-Hochberg method, because the Bonferroni method is similar to Holm but has less power. Here we would chose the Holm method. If the Holm method would not detect any significant values, we would chose the Benjamini-Hochberg method.