Practice Lecture 4: Point estimation

Erlis Ruli (ruli@stat.unipd.it)

16 November 2021

1 Minimum least squares

We consider real data on bacterial counts taken from Ram et al. (2019)[1].

To load the data in R we use the `read.table` command:

```
> bacteria <- read.table("bacteria.csv", sep=";", dec=",", header = T)
>
> # quick look at the newly created object
> str(bacteria)
```

```
## 'data.frame':    67 obs. of  13 variables:
##  $ seconds: num  0 837 1674 2511 3348 ...
##  $ A1     : int  31906 30503 31362 30389 30886 31096 30445 30856 30571 30676 ...
##  $ A2     : int  33127 31315 31620 31923 32121 32158 32220 32315 32411 32357 ...
##  $ A3     : int  31601 31453 31833 32373 32430 32564 32575 32664 32589 32716 ...
##  $ A4     : int  32617 32689 33103 33269 33565 33431 33749 33639 33670 33760 ...
##  $ A5     : int  8329 8317 8395 8486 8405 8454 8458 8440 8528 8535 ...
##  $ A6     : int  8905 8877 8891 8975 9004 9006 9022 9037 9029 9014 ...
##  $ A7     : int  8603 8641 8646 8686 8781 8818 8786 8819 8829 8803 ...
##  $ A8     : int  9377 9285 9390 9476 9475 9515 9536 9475 9507 9462 ...
##  $ A9     : int  1620 1634 1658 1672 1701 1713 1713 1729 1733 1745 ...
##  $ A10    : int  1709 1711 1746 1764 1785 1798 1804 1822 1813 1836 ...
##  $ A11    : int  1679 1672 1696 1724 1744 1757 1769 1771 1785 1795 ...
##  $ A12    : int  1625 1632 1657 1681 1696 1713 1715 1730 1739 1749 ...
```

Se have 67 observations and 13 variables, i.e. columns. The first variable, `seconds`, gives the time points (in seconds) at which the measurements are taken. The other variables are bacteria counts under different different conditions. We focus here on `A12`.

We first transform the time in hours.

```
> bacteria$hours <- bacteria$seconds/60^2
> with(bacteria,
+      plot(y=A12, hours, ylab="Number of bacteria",xlab = "Time in hours"))
```

---

[1]Yoav Ram, Eynat Dellus-Gur,Maayan Bibi, Kedar Karkare, Uri Obolski, Marcus W. Feldman,Tim F. Cooper, Judith Berman, and Lilach Hadany (2019). "Predicting microbial growth in a mixed culture from growth curve data", *Proc. Natl. Acad. Sci. U.S.A.*, **116**, 14698–14707.
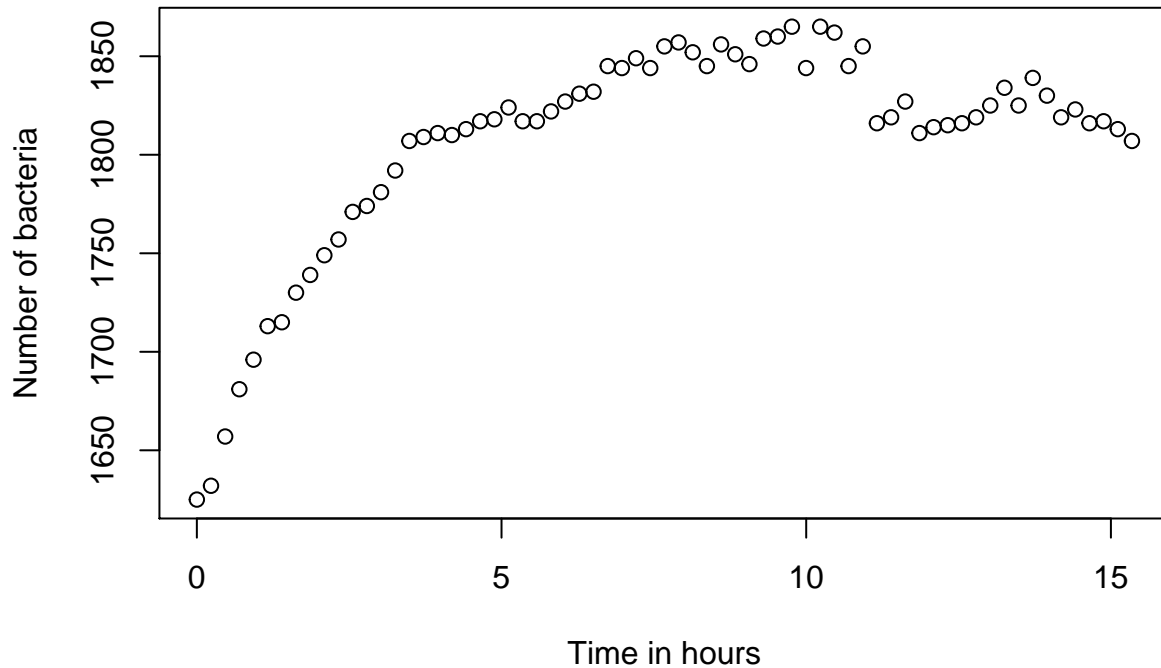
The plot seem to suggest that the observed bacteria counts $y_1, \ldots, y_n$ are quadratically related with time $t_1, \ldots, t_n$. Thus a model with $g_i(\theta) = \theta_0 + \theta_1 t_i + \theta_2 t_i^2$ seems reasonable.

Let's fit this model by LS to this observed data. First we use the formula of Example 4.10.

```
> # y = (y1....,y_n)
> Y <- as.matrix(bacteria$A12)
>
> # X = [1_n | T | T^2]
> X <- as.matrix(cbind(1, bacteria$hours, bacteria$hours^2))
>
> # LS estimate
> hat.theta.LS <- solve(t(X) %*% X) %*% t(X) %*% Y
> row.names(hat.theta.LS) <- c("hat.theta0", "hat.theta1", "hat.theta2")
> hat.theta.LS

##                    [,1]
## hat.theta0 1664.655325
## hat.theta1   40.181716
## hat.theta2   -2.101778
```

Thus the estimated model is $y_i = 1664.65 + 40.18t_i - 2.10t_i^2$, $i = 1, \ldots, 67$. We can also plot this function over the data

```
> with(bacteria,{
+   plot(y=A12, hours, ylab="Number of bacteria",xlab = "Time in hours");
+   points(hours, 1664.65 + 40.18*hours -2.10* hours^2, type="l", lwd=2,
+          lty=2)
+ })
```
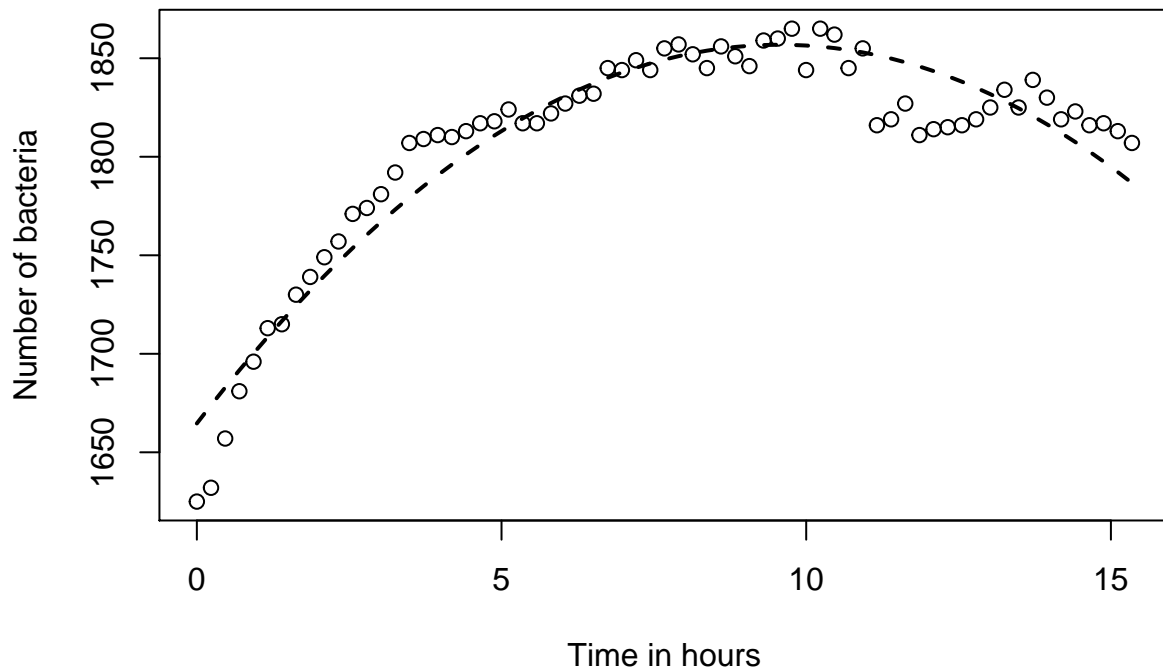
Recall that by the property of LS, the quadratic function shown here is the one which minimises the squared vertical distances of y_i from the curve. These distances are

```
> res <- bacteria$A12 - (1664.65 + 40.18*bacteria$hours -2.10* bacteria$hours^2)
> plot(bacteria$hours,res,type="h", ylab="Residuals (Observed - predicted)")
```
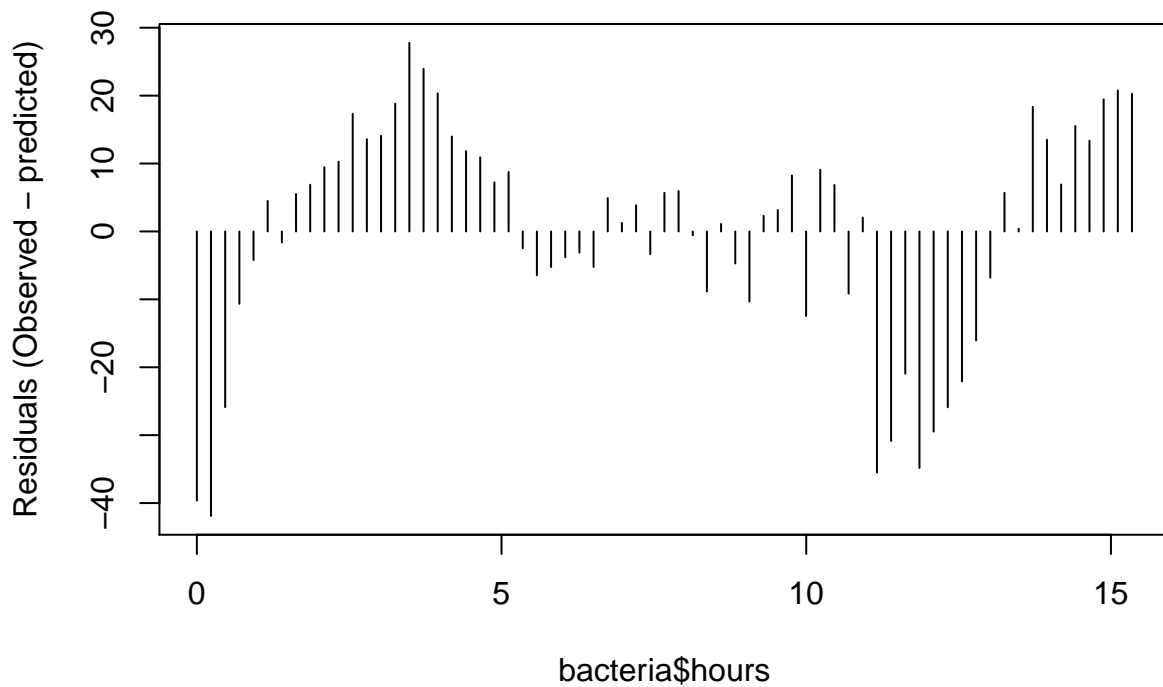

Fitting linear regression a models in R can be done with the command `lm`.

```
> ba.lm <- lm(A12~hours + I(hours^2), data=bacteria)
> summary(ba.lm)

##
## Call:
## lm(formula = A12 ~ hours + I(hours^2), data = bacteria)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -41.884  -6.476   3.298  10.613  27.772
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1664.6553     5.8238   285.8   <2e-16 ***
## hours         40.1817     1.7546    22.9   <2e-16 ***
## I(hours^2)    -2.1018     0.1106   -19.0   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.37 on 64 degrees of freedom
## Multiple R-squared:  0.9139, Adjusted R-squared:  0.9112
## F-statistic: 339.5 on 2 and 64 DF,  p-value: < 2.2e-16
```

The formula "A12~hours + I(hours^2)" says that the model must have `A12` as response and `hours` and `hours`^{2} as predictors. To make sure that the later is to be understood literally as the square of `hours` (and not as the name of some other variable) we use the function `I()`. This formula also includes the intercept θ_0 by default. The `lm` command thus fits the model, i.e. estimates the parameters θ. The output is saved in the object `ba.lm` and printed to the console by the command `summary`.

The LS estimates of the three parameters match with those we obtained manually. In addition, `lm` also provides the standard errors of the estimates, which can be used for building a confidence interval (Lecture 5) or for performing hypothesis testing (Lecture 6).

With this model we may address questions such as: what is the estimated time point of at which the maximum number of bacteria is observed?

In general the peak of the quadratic function $y = \theta_0 + \theta_1 t + \theta_2 t^2$ is at $-\theta_1/(2\theta_2)$, provided $\theta_2 \neq 0$. Thus the peak is estimated to be at hour $-\hat{\theta}_1/(2\hat{\theta}_2) = 9.5$.

Note that, if we assume that the observed sample $y_1, \ldots, y_n$ is obtained from the random sample $Y_i \overset{\text{iid}}{\sim}$ N$(g_i(\theta), \sigma^2)$, with $g_i(\theta) = \theta_0 + \theta_1 t_i + \theta_2 t_i^2$ and t_i fixed, then $\hat{\theta}_{\text{LS}}$ coincides with the MLE of θ.

1.1   Estimation of σ^2

Suppose now that $Y_i = \theta_0 + \theta_1 t_i + \theta_2 t_i^2 + \epsilon_i$, where ϵ_i are i.i.d. and $E(\epsilon_i) = 0$, var$(\epsilon_i) = \sigma^2$, $i = 1, \ldots, n$. Suppose $\theta_0, \theta_1, \theta_2$ are known and the aim is to estimate σ^2, the variance of ϵ_1. We can use the method of moments as follows.

Note that $\epsilon_i = Y_i - \theta_0 + \theta_1 t_i + \theta_2 t_i^2$, for all i, and given the observed sample $y_1, \ldots, y_n$ we can replace $\sigma^2 = E(\epsilon_1^2)$ by its sample-based version $\frac{1}{n} \sum_i (y_i - \theta_0 + \theta_1 t_i + \theta_2 t_i^2)^2 = \widehat{\sigma^2}$, which is thus MM estimator for σ^2. This is nothing more than the sample variance of the residuals.

In our case the estimate is found to be

```
> # hat.sigma
> (hat.sigma = sum(res^2)/length(res))
```

[1] 255.902

```
> # square root of hat sigma
> sqrt(hat.sigma)
```

[1] 15.99694

This MM estimator is biased, though asymptotically unbiased. The `lm` command also gives a similar estimate for σ^2 in "Residual standard error: 16.37", but it uses the slightly different estimator

$$\frac{1}{n-3} \sum_i (y_i - \theta_0 + \theta_1 t_i + \theta_2 t_i^2)^2,$$

Division by the quantity $n - 3$, called degrees of freedom, gives an unbiased estimator. Indeed

```
> # dividing by n-3 we get the unbiased estimate
> (R.hat.sigma = sum(res^2)/(length(res)-3))
```

[1] 267.8974

```
> sqrt(R.hat.sigma)
```

[1] 16.36757

2 Maximum likelihood for the Poisson model

In Example 4.14 of L4 we derived two possible distributions for the MLE. Let's compare these two distributions and see what happens as n increases.

Let the true parameter value, be $\lambda_0 = 3/2$ and we consider samples of size $n \in \{5, 10, 30, 50\}$. By L4 we know that the distribution of the MLE $\hat{\theta}_n$ is approximately $N(3/2, 3/2n)$. Furthermore, $n\hat{\theta}_n$ follows a $\text{Poi}(3n/2)$. Note that $\hat{\theta}_n \sim N(3/2, 3/(2n))$ implies $n\hat{\theta} \sim N(3n/2, 3n/2)$ by a property of the normal distribution.

The following figure compares the exact Poisson distribution with the asymptotic approximation (normal). We notice that, already for $n = 10$ the normal approximation is quite decent and as n increases the normal approximation and the exact distribution get closer and closer.

```
> par(mfrow=c(2,2))
>
> # sample size n=5
> n <- 5
> # exact distribution of n*hat.theta
> plot(x= 0:20,y = sapply(0:20, function(x) dpois(x,lambda=3*n/2)),
+      lwd=2,col=2, type="h",xlim=c(0,20), ylab=expression(f(n*hat(lambda)[n])),
+      xlab = expression(n*hat(lambda)[n]), main=paste("n =",n))
> # approximate dist. of n*hat.theta
> plot(function(x) dnorm(x, mean=n*3/2, sd=sqrt(3*n/2)),
```

```
+        lwd=2, add=TRUE, xlim = c(0,20))
> legend("topright", legend = c("exact", "asymptotic"), lwd=c(2,2), col=c(2,1),
+        lty=c(1,1))
>
> n <- 10
> plot(x= 0:35,y = sapply(0:35, function(x) dpois(x,lambda=3*n/2)),
+        lwd=2,col=2, type="h",xlim=c(0,35), ylab=expression(f(n*hat(lambda)[n])),
+        xlab = expression(n*hat(lambda)[n]),main=paste("n =",n))
> plot(function(x) dnorm(x, mean=n*3/2, sd=sqrt(3*n/2)),
+        lwd=2, add=TRUE, xlim = c(0,35))
>
>
> n <- 30
> plot(x= 20:80,y = sapply(20:80, function(x) dpois(x,lambda=3*n/2)),
+        lwd=2,col=2, type="h",xlim=c(20,80), ylab=expression(f(n*hat(lambda)[n])),
+        xlab = expression(n*hat(lambda)[n]),main=paste("n =",n))
> plot(function(x) dnorm(x, mean=n*3/2, sd=sqrt(3*n/2)),
+        lwd=2, add=TRUE, xlim = c(20,80))
>
> n <- 50
> plot(x= 40:110, y = sapply(40:110, function(x) dpois(x,lambda=3*n/2)),
+        lwd=2,col=2, type="h",xlim=c(40,110), ylab=expression(f(n*hat(lambda)[n])),
+        xlab = expression(n*hat(lambda)[n]),main=paste("n =",n))
> plot(function(x) dnorm(x, mean=n*3/2, sd=sqrt(3*n/2)),
+        lwd=2, add=TRUE, xlim = c(40,120))
```
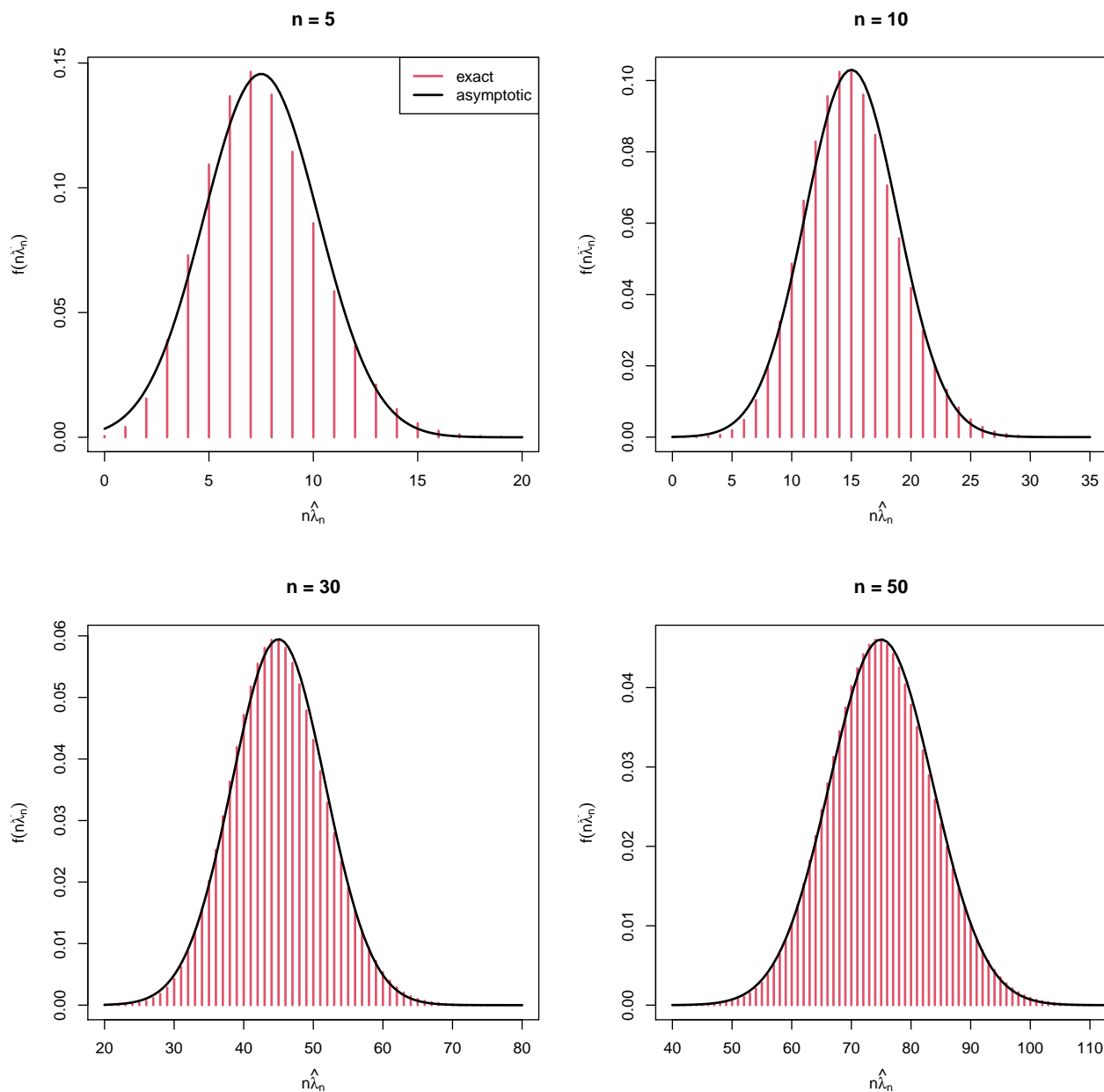
In Lecture 3 we said that the observed information increases with n. This holds also for the Fisher information and it can be immediately seen from the asymptotic distribution of the MLE. If n increases, the variance of the MLE decreases. In practice, this means that with increasing amounts of data, the likelihood function becomes more concentrated around the MLE and thus the distribution of the MLE becomes more concentrated around the true parameter value θ_0.

We illustrate this fact numerically by plotting the asymptotic distribution of the MLE in the case of the Poisson model, with samples of increasing size drawn from the model with $\theta_0 = 3/2$.

```
> # sample of size 5
> n1 = 5
> plot(function(x) dnorm(x, mean=3/2, sd=sqrt(3/(2*n1))),
+      lwd=2, xlim=c(0,3.5),n=500, ylim=c(0,2.5),
```

```
+       ylab=expression(f(hat(lambda)[n])),
+       xlab = expression(hat(lambda)[n]),
+       main = "Distribution of the MLE")
> abline(v=3/2, lwd=2, lty=2, col="lightgray")
> # samples of sizes 10, 30, 50,
> sapply(c(10, 30, 50),
+        function(y) plot(function(x) dnorm(x, mean=3/2, sd=sqrt(3/(2*y))),
+        lwd=2, xlim=c(0,10),col=round(1+y/10),lty=round(1+y/10), n=500, add=TRUE))

##   [,1]        [,2]        [,3]
## x Numeric,500 Numeric,500 Numeric,500
## y Numeric,500 Numeric,500 Numeric,500

> legend("topright", legend = c("n=5", "n=10", "n=30", "n=50"),
+        lwd=c(2,2,2,2), col=c(1,2,4,6), lty=c(1,2,4,6))
```

Distribution of the MLE
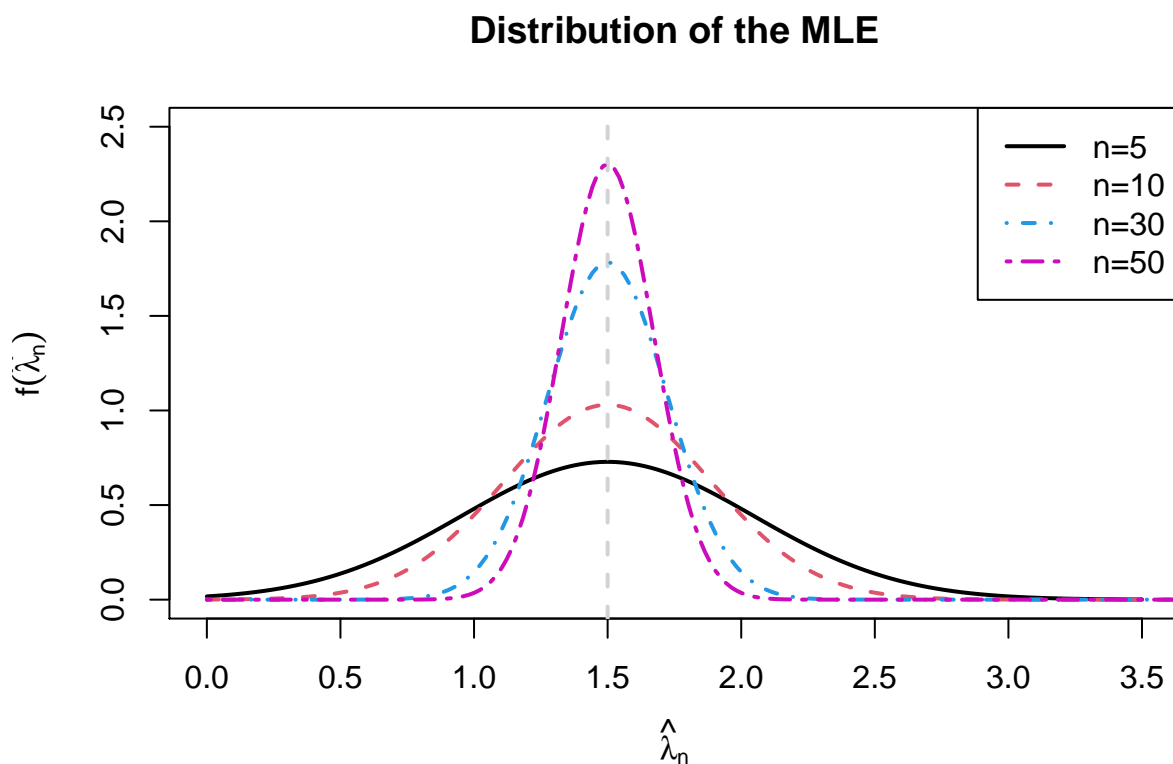
3 MLE for a logistic regression

As a last application consider Example 2.8 in Lecture 2 applied to a different problem, the space shuttle Challenger[2].

On January 28, 1986, a routine launch was anticipated for the space shuttle named Challenger. Seventy-three seconds into the flight, a disaster happened: the shuttle broke apart, killing all seven crew members on board. An investigation into the cause of the disaster focused on a critical seal of the solid rocket boosters, called O-ring, and it is believed that damage to these O-rings during a shuttle launch may be related to the ambient temperature during the launch. Below we summarize observational data on O-rings from 23 pre-Challenger

---

[2]Dalal, Fowlkes and Hoadley (1989) Risk Analysis of the Space Shuttle: Pre-Challenger Prediction of Failure. Journal of the American Statistical Association Vol. 84, No. 408, pp.945-957. See also here https://www.space.com/18084-space-shuttle-challenger.html

shuttle missions, where the mission order is based on the temperature at the time of the launch. Also reported are the number of O-rings damaged after the launch (`orings`).

First we load and plot the data

```
> challenger <- read.table("challenger2.dat", header = TRUE)
> plot(challenger)
> points(x=53, y=5, col=2, pch="*", cex=2)
```
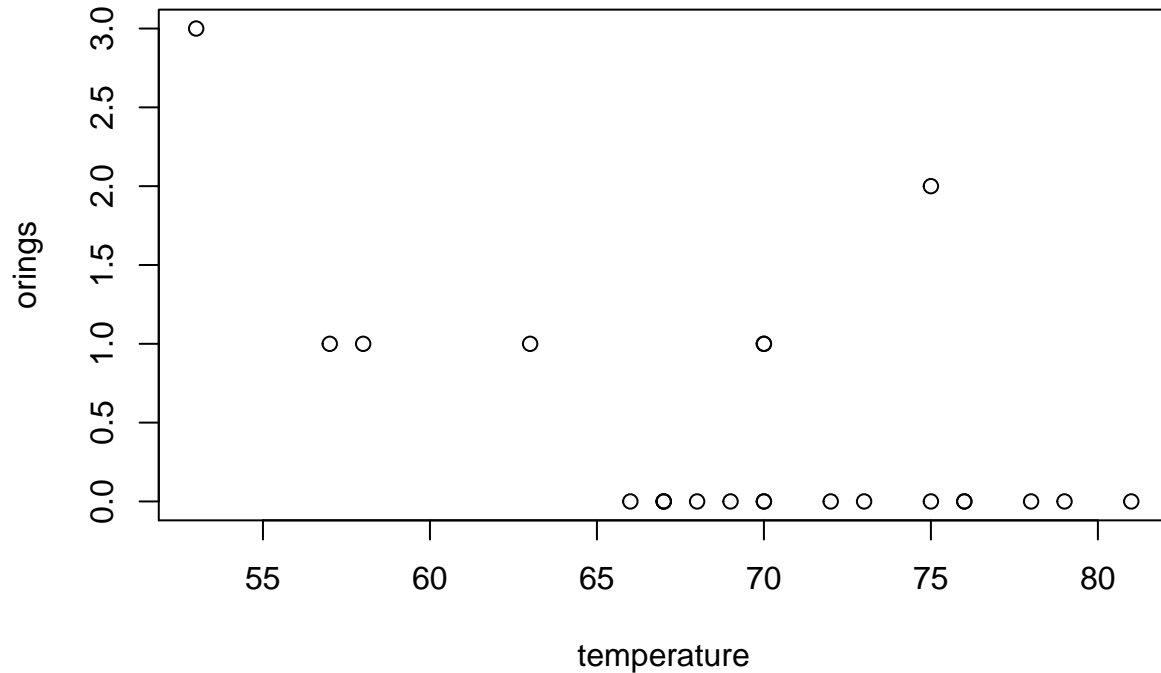

The data we see were analysed by NASA's engineers the day before the Challenger disaster. There was some perplexity about launching the shuttle on that day, due to very low temperature. Some of the engineers plotted a truncated version of the data, discarding all the zeros, as in the figure below.

```
> plot(challenger[challenger$orings>0,],
+      ylab = "Orings(>0)")
> points(x=53, y=5, col=2, pch="*", cex=2)
```
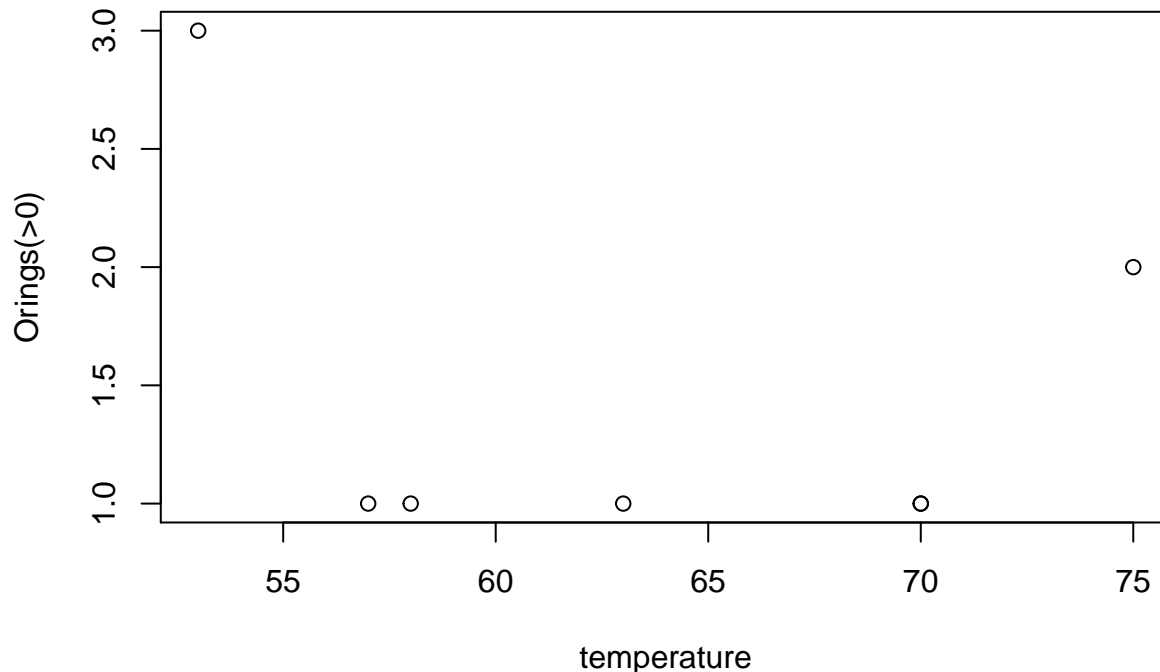
The conclusion was that there is no clear relationship between the number of broken O-rings and temperature, so they decided to launch the shuttle...

To formulate a suitable statistical model, we assume that the launches are independent of each other. Furthermore, each shuttle rocket booster was equipped with 6 O-rings. The r.v. Y is the number of broken O-rings, which thus ranges form 0 to 6.

Let $Y_1, \ldots, Y_n$ be a random sample of broken O-orings. Letting θ denote the probability of failure of a single O-ring, a reasonable distribution is $Y_i \sim \text{Bin}(\theta_i)$, where θ_i depends on characteristics of the ith launch, such as temperature (t_i).

The model is thus

$$
\begin{aligned}
Y_i &\sim \text{Bin}(6, \theta_i), \quad \text{with } Y_i \text{ independent from } Y_j, \text{ for all } i \neq j \\
\text{logit}(\theta_i) &= \alpha + \beta \text{t}_i, \quad i, j = 1, \ldots, n.
\end{aligned}
$$

The parameter α is an intercept and β concerns the relationship between the number of broken O-rings and the temperature; $\beta < 0 \implies$ negative relation.

Obviously the parameters are unknown and we will estimate them from the data.

First let us code the log-likelihood ourselves and then compute the MLE.

```
>    # log-likelihood function for a logistic regression with a single covariate
> logLBin <- function(theta, data){
+
+      n   = nrow(data)
+      alpha. = theta[1]
+      beta. = theta[2]
+
+      # compute the success prob. for each launch (observation), i.e. e^(a + b*t)/(1+e^(a+b*t))
+      thetai = plogis(alpha. + beta.*data$temperature)
+
+      # log-likelihood for each launch (observation)
```

10

```
+      oo = dbinom(x = data$orings, # observed broken o-rings
+                  size = 6, # maximum number of o-rings
+                  prob = thetai,
+                  log=TRUE # TRUE gives the log of the density
+      )
+
+      # sum all the log-likelihood contributions
+      return(sum(oo))
+   }
```

To maximise the log-likelihood function we take as starting value $(\alpha = 10, \beta = 0)$.

```
>   # actually optim minimises the negative log-likelihood
>   start0 <- c(10, 0)
>   (oo <- optim(par = start0,
+               fn = function(x) -logLBin(x, data = challenger),
+               hessian = TRUE)
+   )
```

```
## $par
## [1]  6.7601709 -0.1398558
##
## $value
## [1] 16.37869
##
## $counts
## function gradient
##       77       NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##            [,1]      [,2]
## [1,]    8.402383   535.328
## [2,] 535.328034 34572.101
```

We see that the numerical optimisation routine converged and yelded:

- the MLE of $\theta = (\alpha, \beta)$ given by $\hat{\theta} \doteq (6.76, -0.14$ and $-\log L(\hat{\theta}) \doteq 16.38$
- the observed information matrix

$$J_n \doteq \begin{pmatrix} 8.4 & 535.33 \\ 535.33 & 3.45721 \times 10^4 \end{pmatrix}.$$

The estimated relation between expected number of broken O-rings and temperature is negative, thus the lowest the temperature the higher is the expected number of broken O-rings.

At temperature $t = 31°F$ the estimated probability of failure of a single O-ring is

$$\hat{\theta}_t = \frac{e^{6.76 - 0.14 \cdot 31}}{1 + e^{6.76 - 0.14 \cdot 31}},$$

and the estimated distribution for the failures of O-rings under that temperature is $\text{Bin}(6, \hat{\theta}_t)$. In R

```
> # estimated prob. of failure of single O-ring
> (theta.t <- plogis(oo$par[1] + oo$par[2]*31))
```
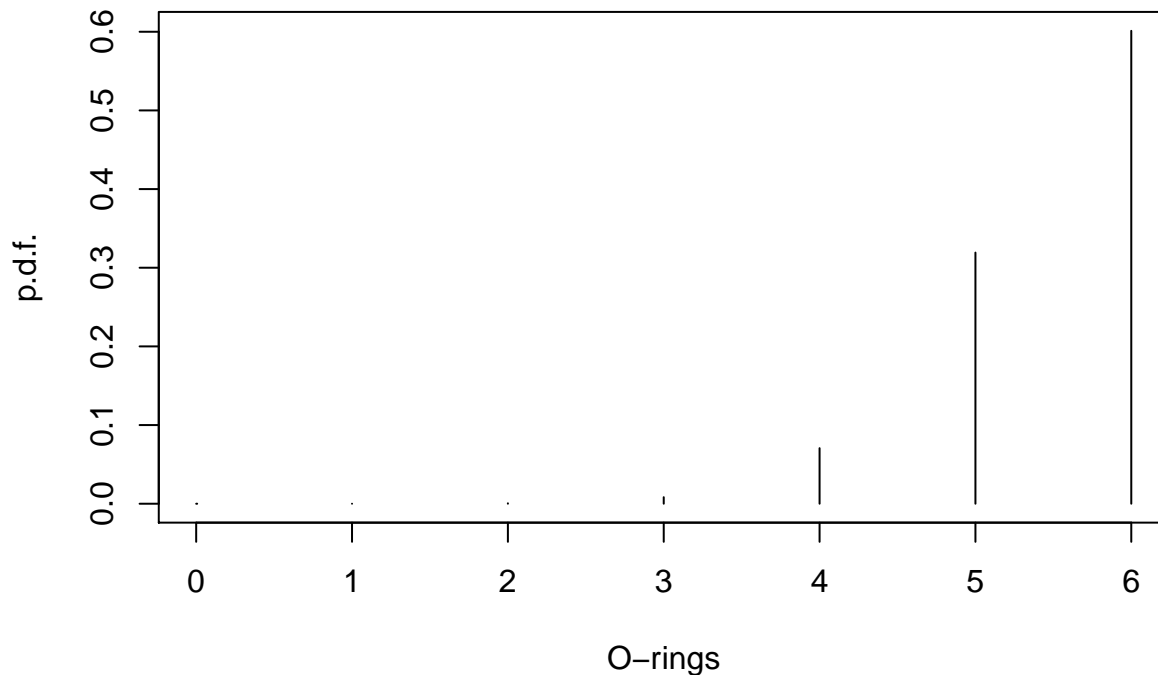
```
## [1] 0.9186872
```

```
> # estimated prob. distribution of nr. of failed O-rings at temp = 31F
> plot(x = 0:6, dbinom(0:6, 6, theta.t), type="h",
+      xlab="O-rings", ylab="p.d.f.")
```


With this simple model we can predict the probability of failure of the space shuttle given the launch temperature. The space shuttle can tolerate up to 3 broken O-rings. Thus estimated estimated probability for the failure of the Challenger space shuttle, launched at temperature $t = 31F$, is the sum of the probability of observing 4, or 5 or 5 failed O-rings:

```
> sum(dbinom(4:6,6,theta.t))
```

```
## [1] 0.9910897
```

```
> # or also
> 1-pbinom(3,6, prob = theta.t,)
```

```
## [1] 0.9910897
```

Which is sadly close to 1.