Practice Lecture 5: Confidence intervals

Erlis Ruli (ruli@stat.unipd.it)

30 November 2021

In this handouts we illustrate examples considered in Lecture 5 in practice using `R`. In some examples we generate fictitious data from the model and in others we consider real-life data.

1 Exact pivots

1.1 Example 5.2 (Normal population with known variance)

Let $Y_1, \dots, Y_n$ be a random sample of size n from $N(\mu, 2)$ and our aim is to build a confidence interval for μ. In particular, suppose that the sample size is $n = 10$ and it is obtained from the above distribution with true parameter $\mu_0 = 0$.

So Nature generates the following data (or if you prefer to think more practically: suppose you are measuring the difference in diameter of bearing spheres with respect to a target value, and you have n such measurements)

```
> set.seed(5) # fix the random seed
> n <- 10 # set the sample size
> mu0 <- 0
> sigma2.0 <- 2
> yobs <- rnorm(n, mu0, sqrt(sigma2.0))
> yobs # measurements we are given by Nature
```

```
##  [1] -1.18914922  1.95777976 -1.77553362  0.09919685  2.42034289 -0.85264064
##  [7] -0.66774411 -0.89855073 -0.40414495  0.19531452
```

We saw in Lecture 5 the procedure for building a confidence interval for this problem, i.e. $\bar{Y} \pm z_{\alpha/2}\sigma/\sqrt{n}$. Let us consider a 0.95 confidence interval, i.e. $\alpha = 0.05$. Thus for the sample at hand we replace the random quantities with their observed counterparts to obtain

```
> alpha = 0.05
> (bar.y <- mean(yobs)) # sample mean
```

```
## [1] -0.1115129
```

```
> (se <- sqrt(sigma2.0/n)) # this is the standard error of the estimate
```

```
## [1] 0.4472136
```

```
> bar.y + c(-1,1)*qnorm(p = alpha/2, lower.tail =FALSE)*se
```

```
## [1] -0.9880355  0.7650096
```

```
> # note: we used lower.tail =FALSE in the quantile function of the
> # normal distribution in order to have the upper quantile as desire.
```

Thus our interval is $[-0.99, 0.77]$. At this point it is good to pause a bit and think about it. This interval is the observed (i.e. numerical) version of the random interval $\bar{Y} \pm z_{\alpha/2}\sigma/\sqrt{n}$. Thus, the probability that the interval $[-0.99, 0.77]$ contains μ_0 can only be either 1 or 0 depending on whether μ_0 is inside or not, respectively. By the way, in this particular case, the observed interval does contain μ_0 so the probability of coverage is 1.

Recall that, in practice, we do not know μ_0, so we will never know if μ_0 is inside the particular observed interval at hand or not. However, if we could compute in this way a large number of intervals, then 95% of them will contain μ_0. So we can only say that we are 95% **confident** that our observed interval $[-0.99, 0.77]$ contains μ_0.

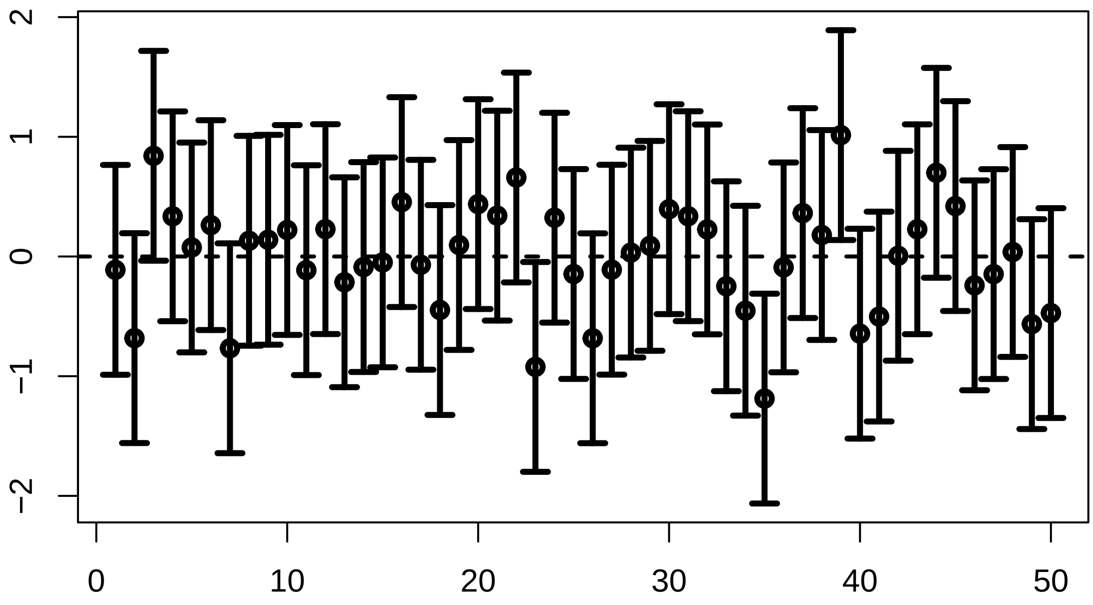
To see this better, we will perform a *simulation study* in which we compute a large number of such intervals, say $N = 10^6$ (i.e. we ask Nature to give us N datasets from the same model used to generate the first sample). The idea is the following. We generate $N = 10^6$ observed samples from the same model as above and for each observed sample, we compute the associated observed 95 percent interval. Here is the `R` code for this.

```
> set.seed(5)
> N = 1e+6 # the number of intervals we want to calculate
> my.N.CI <- matrix(NA, nrow = N, ncol = 2)# put the obs.intervals here
> for(i in 1:N){
+   yi <- rnorm(n, mu0, sqrt(sigma2.0))
+   bar.yi <- mean(yi)
+   my.N.CI[i,] <- bar.yi + c(-1,1)*qnorm(p = alpha/2, lower.tail =FALSE)*sqrt(sigma2.0/n)
+ }
> # here are the first 6 observed intervals out of N
> head(my.N.CI)
```

```
##             [,1]      [,2]
## [1,] -0.98803546 0.7650096
## [2,] -1.55854687 0.1944982
## [3,] -0.03504896 1.7179961
## [4,] -0.54076425 1.2122808
## [5,] -0.80106181 0.9519833
## [6,] -0.61434304 1.1387020
```

We plot below the first 50 observed confidence intervals along with the true parameter value

```
> # we use the plotrix library for doing this plot
> # if not already installed use
> # install.packages(plotrix)
> library(plotrix)
> plotCI(x= 1:50, y = apply(my.N.CI[1:50,], 1, mean),
+        li = my.N.CI[1:50,1],ui = my.N.CI[1:50,2],
+        xlab="Observed confidence intervals for mu",ylab=NA, lwd=3,)
> abline(h = 0, lwd=2, lty=2)
```



Observed confidence intervals for mu

Inspecting the first 50 observed intervals for μ, we note that there are three intervals which do not contain the true value (here denoted by the horizontal dashed line). How may of the N intervals do contain the true value μ_0?

```
> mu0.inside <- apply(my.N.CI, MARGIN = 1,
+                     function(x) ifelse(mu0 >= x[1]  & mu0 <= x[2],1,0))
> # m0.inside is a vector of 1 and 0
> head(mu0.inside)
```

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

```
> # how many ones are there relative to N?
> mean(mu0.inside)
```

```
## [1] 0.949681
```

Thus we conclude that the fraction of the intervals that contain μ_0 is essentially 0.95.

Remark: Obviously it is not exactly 0.95 because 0.949681 is only a sample average which targets the confidence level $(1-\alpha) = 0.95$. By the LLN, this sample average converges to the target as $N \to \infty$, but our lazy choice was $N = 10^6 < \infty$. Also, do not confuse n, the sample size with N the number of simulations we performed. The larger N the closer will be the sample average to the target value $(1-\alpha)$. On the other hand, the pivot we used to build the confidence intervals has an exact distribution, no matter what is n, thus the latter plays no role in this particular case.

Remark: The higher the confidence level, i.e. the lower α the wider is the confidence interval. Thus for $\alpha = 0$, our interval is simply $\mathbb{R}$, thus we are obviously certain that our interval will contain μ_0.

1.2 Example 5.3 (Normal population with known mean)

This time we have μ known and σ^2 is the unknown parameter of interest. Let us set $\mu = 0$, thus the model we are considering is $N(0, \sigma^2)$.

Assume that the true variance is $\sigma_0^2 = 1$ and let $n = 10$. Under this assumption Nature generates the following data (this time we are interested in the variability of the differences in diameter of our bearing spheres).

```
> set.seed(5) # fix the random seed
> n <- 10 # set the sample size
> mu0 <-  0
> sigma2.0 <- 1
> yobs <- rnorm(n, mu0, sqrt(sigma2.0))
```

We saw that the random interval $\left[\frac{n\hat{\sigma}^2_\mu}{\chi^2_{n,\alpha/2}}, \frac{n\hat{\sigma}^2_\mu}{\chi_{n,1-\alpha/2}}\right]$ is a $(1-\alpha)$ confidence interval for σ^2. Let us consider a 0.95 confidence interval. Thus for the sample at hand we replace the random quantities with their observed counterparts to obtain

```
> (hat.sig2.mu <- sum((yobs-mu0)^2)/n)

## [1] 0.8224575
```

```
> # CI is
> c(n*hat.sig2.mu/qchisq(p=alpha/2, df=n, lower.tail = FALSE),
+   n*hat.sig2.mu/qchisq(p=1-alpha/2, df=n, lower.tail = FALSE))

## [1] 0.4015283 2.5329977
```

Again, this is an observed interval and may or may not contain the true value σ_0^2. All we can say is that we are 95% confident that it will.

Exercise. Compute a 0.95 confidence interval for $\log \sigma^2$.

1.3 Example 5.4 (Normal population)

This time both μ and σ^2 are unknown parameters, thus the model we are considering is $N(\mu, \sigma^2)$. We want to compute a confidence interval for each of the parameters.

This time we have two different pivotal quantities (see Lecture 5). Assume that the true mean is $\mu_0 = 0$ and the true variance is $\sigma_0^2 = 1$ and let $n = 10$. Under this assumption Nature generates the following data (this time we are interested both in the average and in the variability of the differences in diameter of our bearing spheres).

```
> set.seed(5) # fix the random seed
> n <- 10 # set the sample size
> mu0 <-  0
> sigma2.0 <- 1
> yobs <- rnorm(n, mu0, sqrt(sigma2.0))
```

Let us consider 0.95 confidence intervals. Thus for the sample at hand we replace the random quantities with their observed counterparts to obtain

```
> # recall that var divides by n-1!
> (hat.sig2 <- var(yobs))

## [1] 0.9069332
```

```
> (hat.mu <- mean(yobs))

## [1] -0.07885155
```

```
> # the standard error of hat.mu
> se <- sqrt(hat.sig2/n)
>
> # CI for mu
> c(hat.mu + c(-1,1)*qt(alpha/2, df=n-1, low=F)*se)
```

```
## [1] -0.7601077  0.6024046
```

```
> # CI for mu
> c(n*hat.sig2/qchisq(p=alpha/2, df=n-1, lower.tail = FALSE),
+   n*hat.sig2/qchisq(p=1-alpha/2, df=n-1, lower.tail = FALSE))
```

```
## [1] 0.476762 3.358527
```

Again, these are observed intervals, which may or may not contain their respective values. All we can say is that we are 95% confident that each interval will contain its true value.

Since R is a statistical software we do not need to do everything "by hand" as above. In this example we can instead use the built-in R command `t.test` which, besides other things, computes also a confidence interval for μ.

```
> # for the CI for mu
> t.test(yobs)
```

```
##
##  One Sample t-test
##
## data:  yobs
## t = -0.26183, df = 9, p-value = 0.7993
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.7601077  0.6024046
## sample estimates:
##   mean of x
## -0.07885155
```

Do not worry about the meaning of rest of the output, we will see it in the incoming lecture. Notice how much wider are the confidence intervals obtained in this example as compared to those of the previous two examples. The higher uncertainty due to wider intervals is the price we have to pay when the parameters are unknown and thus have to be estimated from data.

1.4 Example 5.5 (Difference of means for two normal samples)

To illustrate this example we consider real data on energy consumption of two type of WM's, which differ only on the type of motor they are equipped with. The data are not paired, in the sense that the WM's in the two groups have different ID's.

In this problem we are interested in the difference in of the two population means $\mu_1 - \mu_2$. We first load the data in R by

```
> # read the file, specifying header=TRUE since the first row contains the names of the variables.
> motors <- read.table("wm_motors.txt", header = TRUE)
> head(motors)
```

```
##      energy motor
## 1 0.7423580  GEN1
## 2 0.6971112  GEN1
## 3 0.7268793  GEN1
## 4 0.7188058  GEN1
## 5 0.7218520  GEN1
## 6 0.7777559  GEN1
```

```
> # another quick view for data frames is
> str(motors)
```

```
## 'data.frame':    30 obs. of  2 variables:
```

```
##  $ energy: num  0.742 0.697 0.727 0.719 0.722 ...
##  $ motor : chr  "GEN1" "GEN1" "GEN1" "GEN1" ...
```

The command `read.table` reads the external file and loads the data into an `R` object, here called `motors`, which is of type: `data.frame`. The latter generalises matrices, as created by the command `matrix`, since they are able to store numbers as well as strings, i.e. in a matrix you can only store things of the same type.

First we perform some reshaping of the data in order to have the two variables separated.

```
> # energy consumption with motor GEN1
> y <- motors$energy[motors$motor == "GEN1"]
>
> # energy consumption with motor GEN2
> x <- motors$energy[motors$motor == "GEN2"]
```

Now we compute the confidence interval for $\mu_1 - \mu_2$ by assuming that the two samples come from two normal distributions, resp. $Y_i \overset{iid}{\sim} N(\mu_1, \sigma_1^2)$ and $X_i \overset{iid}{\sim} N(\mu_2, \sigma_2^2)$, assuming equal variances, i.e. $\sigma_1^2 = \sigma_2^2 = \sigma$.

```
> # sample averages of energy consumption under GEN1 and GEN2
> bar.y <- mean(y)
> bar.x <- mean(x)
>
> # compute the size of the two samples
> n <- length(y)
> m <- length(x)
>
> # sample variances
> s2.y <- var(y)
> s2.x <- var(x)
>
> # pooled variance
> pooled.s2 <- ((n-1)*s2.y + (m-1)*s2.x)/(n+m-2)
>
> # the standard error is thus
> se = sqrt(pooled.s2*(1/n+1/m))
>
> # the confidence interval is then
> (bar.y-bar.x) + c(-1,1)*qt(alpha/2, df=n+m-2, low=F)*se
```

```
## [1] -0.044354504 -0.008654067
```

We see that both limits of the observed confidence interval are negative. To interpret this result, suppose that the true value of μ_1, say $\mu_{1,0}$ and the true value of μ_2, $\mu_{2,0}$, are equal, i.e. $\mu_{1,0} = \mu_{2,0}$ so then $\mu_{1,0} - \mu_{2,0} = 0$. Actually, this was what the engineers wanted to check in the study. Indeed, they suspected that the two motors do not consume on average the same energy, other things held equal. We see that zero is not inside the confidence interval. Thus, what we can conclude from this study is that we are 95% confident that the means of energy consumption in the two groups of WM's are not equal, i.e. with 95% confidence the two motors consume on average different amounts of energy.

Again, there is a quicker way to compute this interval: the `t.test` command.

```
> t.test(y,x, var.equal = T)
```

```
##
##  Two Sample t-test
##
## data:  y and x
## t = -3.0415, df = 28, p-value = 0.005068
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.044354504 -0.008654067
## sample estimates:
## mean of x mean of y
## 0.7139261 0.7404304
```

1.5 Example 5.6 (Ratio of variances for two normal samples)

We consider the same data of the example above but now we assume different variances between the samples and we are interested on the ratio of the variances. We compute the 0.95 confidence interval for the ratio σ_1^2/σ_2^2, both manually, by applying its mathematical definition, and by using the R command var.test.

```
> c(s2.y/s2.x * qf(alpha/2, df1=n-1, df2=m-1),
+   s2.y/s2.x * qf(1- alpha/2, df1=n-1, df2=m-1))
```

```
## [1]  8.632077 76.583642
```

```
> # the same done with the built-in R command
> var.test(y,x)
```

```
##
##  F test to compare two variances
##
## data:  y and x
## F = 25.711, num df = 14, denom df = 14, p-value = 2.897e-07
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##   8.632077 76.583642
## sample estimates:
## ratio of variances
##            25.7114
```

Thus we see that with 95% of confidence the ratio of the variances σ_1^2/σ_2^2 varies from 8.6 to 76.6.

Exercise If the engineers claim that the true variances $\sigma_{1,0}^2$ and $\sigma_{2,0}^2$ are equal. Do the data support this claim?

2 Asymptotic pivots

In the case of exact pivots (as in Section 1), the only way the sample size n affects the results is by leading to narrower confidence intervals.

In the case of asymptotic pivots, not only n does affect the width of the interval, in the same way as above, but it does affect also the coverage probability of the interval itself. Thus, the coverage of confidence intervals from asymptotic pivots will not be exactly equal to $1-\alpha$ but it will converge to $(1-\alpha)$ as $n \to \infty$.

In practice our dataset has a fixed sample size n and this is result is not very useful since it cannot tell us how accurate our intervals will be in that particular sample size. However, we can check the coverage of our intervals for a given sample size by simulation. We will do this in the next example.

2.1 Example 5.8 (Poisson population)

Consider the same setting as in Lecture 5 and the aim is to check the coverage probability of the approximate $(1-\alpha)$ Wald confidence intervals, i.e. we want to compute

$$P_\lambda\left(\bar{Y}+z_{\alpha/2}\sqrt{\frac{\bar{Y}}{n}}\leq\lambda\leq\bar{Y}+z_{\alpha/2}\sqrt{\frac{\bar{Y}}{n}}\right),$$

for different values of n. The aim is thus to see if the coverage probability goes towards $(1-\alpha)$ as n increases.

This probability can be computed analytically, but here we will approximate it by simulation, i.e. we will use the Monte Carlo method (see P1). The Monte Carlo method can be used to approximate any feature of a probability distribution that is analytically difficult.

We set the true parameter $\lambda_0 = 3/2$ and take sample sizes $n = 5, 50$. We will approximate such the coverage probability by its empirical average in a large Monte Carlo sample N. By the LLN we know that as $N \to \infty$, the sample average will converge to the true probability coverage of the interval.

```
> lambda0 <- 3/2
> N <- 1e+6
>
> # we first build a function which takes as input an observed sample and
> # outputs a confidence interval
> ci.poisson <- function(ysamp, alpha){
+   n = length(ysamp)
+   bar.y = mean(ysamp)
+   se = sqrt(bar.y/n)
+   oo = c(bar.y + c(-1,1)*qnorm(alpha/2, low=F)*se)
+   return(oo)
+ }
```

Now we generate datasets from the Poison model and compute the confidence intervals by our newly defined function `ci.poisson`.

With sample of size $n = 5$ we have

```
> set.seed(5)
> n <- 5
> CI.5 <- matrix(NA, nrow = N, ncol = 2)# put the obs.intervals here
> for(i in 1:N) {
+   y5 <- rpois(n, lambda0)
+   CI.5[i,] <- ci.poisson(y5,alpha = 0.05)
+ }
> lambda0.inside5 <- apply(CI.5, MARGIN = 1,
+                      function(x) ifelse(lambda0 >= x[1]  & lambda0 <= x[2],1,0))
> # how many ones are there?
> mean(lambda0.inside5)
```

```
## [1] 0.936302
```

With sample of size $n = 50$ we have

```
> set.seed(5)
> n <- 50
> CI.50 <- matrix(NA, nrow = N, ncol = 2)# put the obs.intervals here
> for(i in 1:N) {
+   y50 <- rpois(n, lambda0)
+   CI.50[i,] <- ci.poisson(y50,alpha = 0.05)
+ }
> lambda0.inside50 <- apply(CI.50, MARGIN = 1,
+                      function(x) ifelse(lambda0 >= x[1]  & lambda0 <= x[2],1,0))
```

```
> # how many ones are there?
> mean(lambda0.inside50)
```

```
## [1] 0.952314
```

Thus we see that with $n = 5$ and when $\lambda_0 = 3/2$ the probability coverage of the interval is approximately equal to 0.9363 whereas with $n = 50$ such a probability is 0.9523, much closer to the nominal value 0.95.

Exercise Use the fact that $n\bar{X} \sim \text{Poi}(n\lambda_0)$, to compute the coverage of the above intervals exactly.

2.2 Example 5.9 (Is Mendel's theory supported?)

In one of his experiments with pea-plants Mendel crossed a certain number of green pod plants with yellow pod plants. The first generation (G1) he got only green plants (green is the dominant trait colour). Successively G1 plants were let to self-pollinate leading to second generation (G2) plants. The G2 plants Mendel got where 39 green and 17 were yellow. Mendel wanted to know if the proportion of the green plants was equal to 3/4.

Formalising this problem statistically, let Y_i denote a binary r.v. which takes value 1 if the ith G2 plant is green and takes 0 otherwise, let $\theta = P(Y_i = 1)$, i.e. the probability of having a green plant among the G2 plants. It is reasonable to assume that the plants of the G2 are between them independent, thus we have that $Y_1, \ldots, Y_n$, with $Y_i \sim \text{Ber}(\theta)$, and in this case $n = 56$. The aim is then to compute a confidence interval for θ.

With the data above we have that

```
> # 36 green out of 56
> bar.y = 39/(39+17)
> se = sqrt((bar.y)*(1-bar.y)/56)
>
> # the 0.95 CI is
> bar.y + c(-1,1)*qnorm(alpha/2, low=F)*se
```

```
## [1] 0.5760019 0.8168553
```

According to Mendel's theory, since the green colour is dominant, it must appear in 75% of the plants. In other words, $\theta_0 = 0.75$. With 0.95 confidence we can say that Mendel's theory is supported.

A confidence interval for the probability of success θ can also be obtained by the `R` command `prop.test` as follows

```
> prop.test(x=39, n=39+17, correct = F)
```

```
##
##  1-sample proportions test without continuity correction
##
## data:  39 out of 39 + 17, null probability 0.5
## X-squared = 8.6429, df = 1, p-value = 0.003283
## alternative hypothesis: true p is not equal to 0.5
## 95 percent confidence interval:
##  0.5666413 0.8009967
## sample estimates:
##         p
## 0.6964286
```

Note the 0.95 confidence interval here does not exactly coincide with the Wald-type confidence interval we computed above. This is because `prop.test` uses another approximate pivot which is based on the score function. We will not illustrate it here but it suffices to say that in large samples the two pivots give very similar results. By this command we can build also a confidence interval for a two sample problem as in Example 5.10.

2.3 Example 5.11 (Two normal populations with unequal variances)

This example can be handled similarly as in Example 5.3, where this time the option `var.equal = FALSE` in the command `t.test` must be specified.

2.4 The Challenger disaster (again!)

We revisit space shuttle data and this time we want to address the if there is a relationship between the number of broken o-rings and temperature.

Using the same statistical model as in P4, we have

$$\begin{aligned} \text{orings}_i &\sim \text{Bin}(6, \theta_i), \quad \text{with oring}_i \text{ independent from oring}_j, \text{ for all } i \neq j = 1, \dots, 23, \\ \text{logit}(\theta_i) &= \alpha + \beta\text{temperature}_i, \quad i = 1, \dots, 23, \end{aligned}$$

The parameter β concerns the relationship between the number of broken o-rings and the temperature; $\beta < 0$ implies that there is a negative relation.

Will estimate the parameters (α, β) from the observed data using the MLE. Since there is no exact pivotal quantity for this problem we will appeal to Theorem 4.9 for building a confidence interval for β.

To estimate the parameters we use the built-in `R` command `glm` as follows.

```
> challenger <- read.table("challenger2.dat", header = TRUE)
> oo2 <- glm(cbind(orings, 6-orings)~temperature, data=challenger,
+            family = binomial())
> summary(oo2)
```

```
##
## Call:
## glm(formula = cbind(orings, 6 - orings) ~ temperature, family = binomial(),
##     data = challenger)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9876  -0.7798  -0.4987  -0.2975   2.7483
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  6.75183    2.97989   2.266  0.02346 *
## temperature -0.13971    0.04647  -3.007  0.00264 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 28.761  on 22  degrees of freedom
## Residual deviance: 19.093  on 21  degrees of freedom
## AIC: 36.757
##
## Number of Fisher Scoring iterations: 5
```

The description of this command is lengthy and outside the scope of this course. The rest of the output need not concern us for the moment. We will see what the columns Z and Pr(>|Z|) are useful for in Lecture 6.

The output however provides us the standard errors of the estimators. For instance, we see that $se(\hat{\beta})\doteq$0.04647. We can thus build an asymptotic Wald confidence interval for β:

$$\left[\hat{\beta} \pm z_{\alpha/2} se(\hat{\beta})\right],$$

which for large n has confidence level $1-\alpha$. For $\alpha = 0.05$ we have the interval $[-0.231, -0.049]$.

We conclude that we are 95% confident that the relation between temperature and expected number of failed o-rings is negative.