Bootstrap Sampling Distribution and Confidence Intervals

MTH 541/643: Statistics Theory II

Instructor: Songfeng Zheng

For some simple cases, we have seen how we can obtain the sampling distribution and based on the sampling distribution, how we can obtain the confidence intervals for the unknown parameters. In all the examples, and in theory, only standard distributions allow us to perform these tasks by the introduced methods. However, in many applications, the underlying probability model is very complicated or even unknown, and this makes it very difficult or impossible to obtain sampling distribution. In some other cases, the sampling distribution can be derived in theory, but it is too complicated to apply. We know the sampling distribution and confidence interval are desired properties about the estimator. In these situations, we can use computer simulation to find the sampling distribution and approximate confidence intervals.

We will introduce the R programming for the simulation method via an example (we have already studied). The method introduced here is called Bootstrap method.

The Poisson distribution has been used by traffic engineers as a model for light traffic, based on the rationale that if the rate is approximately constant and the traffic is light (so the individual cars move independently of each other), the distribution of counts of cars in a given time interval or space area should be nearly Poisson (Gerlough and Schuhl 1955). The following table shows the number of right turns during 300 3-minute intervals at a specific intersection.

n	frequency
0	14
1	30
2	36
3	68
4	43
5	43
6	30
7	14
8	10
9	6
10	4
11	1
12	1
13+	0

If we suppose Poisson model might be a good model for this dataset, we still need to find out what Poisson, that is estimate the parameter λ in the Poisson model:

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!} \, .$$

Of course, we can use the formula to calculate MLE or MM of the parameter λ in the Poisson model as: $\hat{\lambda} = \bar{X}$ (please check this yourselves). The sampling distribution for $\hat{\lambda}$ can also be derived as a standard form. For the purpose of demonstrating, let us assume we don't know the standard distribution and use this as an example to show the computer simulation.

Bootstrap Sampling Distribution

Imagine for the moment that we knew the true value of the parameter λ_0. This means that the true distribution is known. So we could generate many, many samples of size 300 from the Poisson distribution with the parameter λ_0, and for each of these samples, we could calculate estimates of λ. A histogram of the values of the estimates of λ should then give us a good idea of the sampling distribution of $\hat{\lambda}$

The only problem with this idea is that it requires knowing the true parameter value, which we don't know in practice. We can substitute our estimated value of $\lambda = 3.8933$ for the true value. Suppose from this Poisson distribution, we can draw B samples of size n = 300, and using Method of Moment or MLE to estimate λ for each sample. Please see the following code:

read the data

y <- c(rep(0,14), rep(1,30), rep(2,36), rep(3,68), rep(4, 43), rep(5,43), rep(6, 30), rep(7,14), rep(8,10), rep(9, 6), rep(10,4), rep(11,1), rep(12,1))

n <- length(y)

NegLogLike<-function(p){

  NegLogLike <- -(mean(y*log(p))-p-mean(log(factorial(y))))

}

using MLE to find the estimated value

out<-nlm(NegLogLike, p=c(0.5), hessian = TRUE)

lambda_hat_MLE <- out$estimate

Next, we show how to do the sampling and redo estimation. Basically this will use a for loop.

B <- 2000   # we will sample 2000 samples and estimate 2000 times

we will save the bootstrap results in this array, and this array is initialized as 0 vector

lambda_hat_MLE_Bootstrap1 <- rep(0,B)

for(i in 1:B) # for loop, we will repeat the experiment B times

{

    # get a sample of size n, with parameter lambda_hat_MLE

    y <- rpois(n, lambda_hat_MLE)

    # for this new sample, find the MLE

    out<-nlm(NegLogLike, p=c(0.5), hessian=TRUE)

    # save the estimation result of the i-th iteration

    lambda_hat_MLE_Bootstrap1 [i]<- out$estimate

}

hist(lambda_hat_MLE_Bootstrap1, main="sampling distribution of Bootrapped MLE for Poisson", xlab = "estimated value", breaks=50, prob=TRUE)

curve(dnorm(x,   mean=mean(lambda_hat_MLE_Bootstrap1),   sd= sd(lambda_hat_MLE_Bootstrap1)),col='blue',add=TRUE)

The blue curve shows the normal curve, and we can see the sampling distribution is very close to normal.

The above shows the histogram of the bootstrapped MLE for Poisson distribution, and this will give us a rough idea of the sampling distribution of $\hat{\lambda}$. The variability shown by the histogram can be summarized by calculating the standard deviation of the B estimates, thus providing estimated standard error of $\hat{\lambda}$. To be precise, if the B estimates of λ are denoted by $\lambda_i^*, i = 1, 2, \mathrm{L}, B$, then the standard error of $\hat{\lambda}$ can be estimated as

$$s_{\hat{\lambda}} = \sqrt{\frac{1}{B-1}\sum_{i=1}^{B}\left(\lambda_i^* - \bar{\lambda}\right)^2}$$

where $\overline{\lambda}$ is the mean of the B estimated values. The standard error can be calculated efficiently using a command in R by **sd(lambda_hat_MLE_Bootstrap1)**.

For the method of moment, we can do the same sampling process to obtain the approximate sampling distribution.

sampling distribution of lambda_hat_MLE_Bootstrap
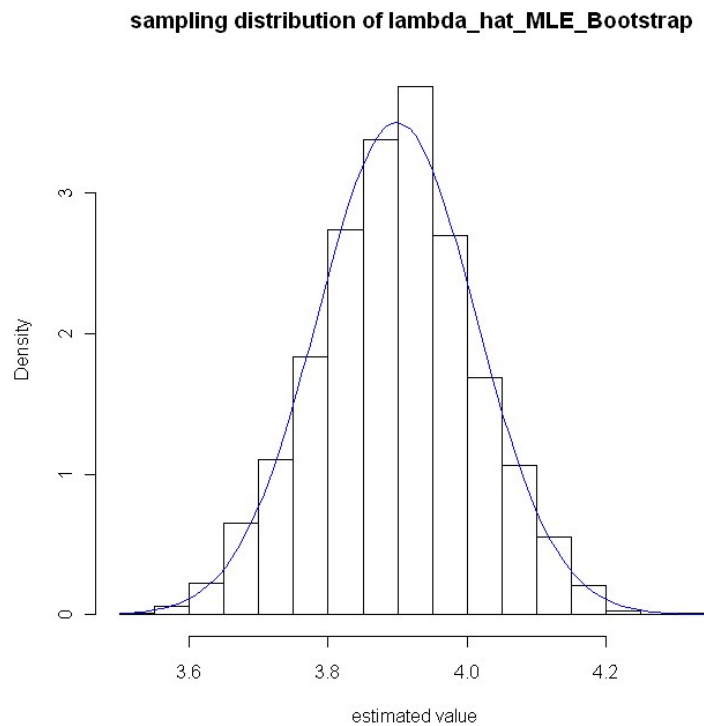
Note: The above method is from *Mathematical Statistics and Data Analysis, 3-rd Edition, by John A. Rice*. In this method, the Bootstrap is different from other literature, and another version of Bootstrap is introduced below

Another Version of Bootstrap

The following description for Bootstrap is widely accepted in literature, and it is available at http://en.wikipedia.org/wiki/Bootstrapping_(statistics)

The basic idea of bootstrapping is that the sample we have collected is often the best guess we have as to the shape of the population from which the sample was taken. For instance, a sample of observations with two peaks in its histogram would not be well approximated by a Gaussian or normal bell curve, which has only one peak. Therefore, instead of assuming a mathematical shape (like the normal curve or some other) for the population, we instead use the shape of the sample.

As an example, assume we are interested in the average (or mean) height of people worldwide. We cannot measure all the people in the global population, so instead we sample

only a tiny part of it, and measure that. Assume the sample is of size N; that is, we measure the heights of N individuals. From that single sample, only one value of the mean can be obtained. In order to reason about the population, we need some sense of the variability of the mean that we have computed.

To use the simplest bootstrap technique, we take our original data set of N heights, and, using a computer, make a new sample (called a bootstrap sample) that is also of size N. This new sample is taken from the original using sampling with replacement so it is not identical with the original "real" sample. We repeat this a lot (maybe 1000 or 10,000 times), and for each of these bootstrap samples we compute its mean (each of these are called bootstrap estimates). We now have a histogram of bootstrap means. This provides an estimate of the shape of the distribution of the mean from which we can answer questions about how much the mean varies. (The method here, described for the mean, can be applied to almost any other statistic or estimator.)

The key principle of the bootstrap is to provide a way to simulate repeated observations from an unknown population using the obtained sample as a basis. A great advantage of bootstrap is its simplicity. It is straightforward to derive estimates of standard errors and confidence intervals for complex estimators of complex parameters of the distribution, such as percentile points, proportions, odds ratio, and correlation coefficients. Moreover, it is an appropriate way to control and check the stability of the results.

For a concrete example, please refer to the following link:
https://en.wikipedia.org/wiki/Bootstrapping_(statistics)#Estimating_the_distribution_of_sample_mean

Let's study the following code for implementing this version of Bootstrap:

y_original is for the original data

y_original <- c(rep(0,14), rep(1,30), rep(2,36), rep(3,68), rep(4, 43), rep(5,43), rep(6, 30), rep(7,14), rep(8,10), rep(9, 6), rep(10,4), rep(11,1), rep(12,1))

this y will be changed in the iterations

y <- y_original

n <- length(y)

NegLogLike<-function(p){

  NegLogLike <- -(mean(y*log(p))-p-mean(log(factorial(y))))

}

B <- 2000   # we will sample 2000 samples and estimate 2000 times

we will save the bootstrap results in this array, and this array is initialized as 0 vector

lambda_hat_MLE_Bootstrap2 <- rep(0,B)

for(i in 1:B) # for loop, we will repeat the experiment B times

{

sample from the original data, using sampling with replacement!

y <- sample(y_original, n, replace=T)

for this new sample, find the MLE

out<-nlm(NegLogLike, p=c(0.5), hessian=TRUE)

save the estimation result of the i-th iteration

lambda_hat_MLE_Bootstrap2 [i]<- out$estimate

}

hist(lambda_hat_MLE_Bootstrap2, main="sampling distribution of Bootrapped MLE for Poisson", xlab = "estimated value", breaks=50, prob=TRUE)

curve(dnorm(x,   mean=mean(lambda_hat_MLE_Bootstrap2),   sd= sd(lambda_hat_MLE_Bootstrap2)), col='red', add=TRUE)

sampling distribution of Bootrapped MLE for Poisson
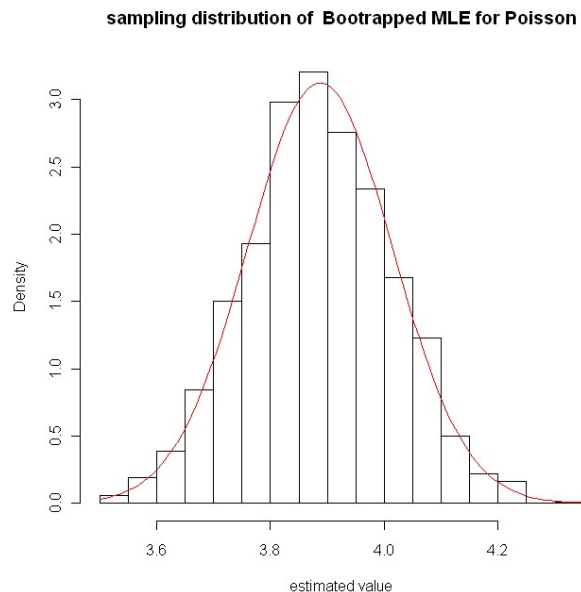
Comparing the results:

> mean(lambda_hat_MLE_Bootstrap2)

[1] 3.887253

> mean(lambda_hat_MLE_Bootstrap1)

[1] 3.89412

> sd(lambda_hat_MLE_Bootstrap1)

[1] 0.1108861

> sd(lambda_hat_MLE_Bootstrap2)

[1] 0.1275645

Getting Fisher Information and the approximate CI from the MLE code

In the MLE code above, we are minimizing the negative log-likelihood function

$$g = -\frac{1}{n}\sum_{i=1}^{n} \log f(x_i \mid \lambda) = -\frac{1}{n}\sum_{i=1}^{n} l(x_i \mid \lambda)$$

then the Hessian (second order derivative) is

$$g'' = -\frac{1}{n}\sum_{i=1}^{n} l''(x_i \mid \lambda) \xrightarrow{LLN} -E\left(l''(X \mid \lambda)\right) = I(\lambda)$$

As such, we can get the approximated Fisher information from the output of the MLE code. As in this example, the output is

$minimum

[1] 2.223943

$estimate

[1] 3.893331

$gradient

[1] 1.140641e-09

$hessian

      [,1]

[1,] 0.2567982

$code

[1] 1

$iterations

[1] 9

Thus, from the code, we can get the Fisher Information as 0.2567982. This example is about a Poisson distribution, and we can calculate the Fisher Information as $I(\lambda) = 1/\lambda$

The the estimated Fisher Information is 1/3.893331= 0.2568495, which is very close to the Hessian.

With the estimated Fisher Information, we can get the approximate confidence interval for the parameter using the asymptotic distribution of MLE as

$$\hat{\lambda} \pm \frac{z(1-\alpha/2)}{\sqrt{nI(\hat{\lambda})}}$$

By the following command, we can get the confidence bounds as

FI <- out$hessian

alpha<- 0.05

Upper <- lambda_hat_MLE+qnorm(1-alpha/2)/sqrt(n*FI)

Lower <- lambda_hat_MLE-qnorm(1-alpha/2)/sqrt(n*FI)

The results are:

> Upper

   [,1]

[1,] 4.116633

> Lower

   [,1]

[1,] 3.67003

We will compare to the results in the next section.

Check **sd(lambda_hat_MLE_Bootstrap1)** and **sqrt(1/(n*FI))**, what do you observe? Why?

Bootstrap Confidence Interval

We now describe the use of the bootstrap for finding approximate confidence intervals. Suppose that $\hat{\theta}$ is an estimate of a parameter θ --- the true, unknown value of which is θ_0 --- and suppose for the moment that the distribution of $\Delta = \hat{\theta} - \theta_0$ is known. Denote the $\alpha/2$ and $1-\alpha/2$ quantiles of this distribution by $\underline{\delta}$ and $\bar{\delta}$, i.e.

$$P(\hat{\theta} - \theta_0 \leq \underline{\delta}) = \frac{\alpha}{2} \text{ and } P(\hat{\theta} - \theta_0 \leq \bar{\delta}) = 1 - \frac{\alpha}{2}$$

Then,

$$P(\underline{\delta} \leq \hat{\theta} - \theta_0 \leq \bar{\delta}) = 1 - \alpha$$

and manipulating this, we have the confidence interval:

$$P(\hat{\theta} - \bar{\delta} \leq \theta_0 \leq \hat{\theta} - \underline{\delta}) = 1 - \alpha.$$

The above process assumes that the distribution of $\Delta = \hat{\theta} - \theta_0$ is known, which is typically not the case. If θ_0 were known, this distribution can be approximated arbitrarily well by simulation: many, many samples of observations could be randomly generated on a computer with the true value θ_0; for each sample, we can estimate $\hat{\theta}$, and the difference $\hat{\theta} - \theta_0$ can be recorded; and the two quantiles $\underline{\delta}$ and $\bar{\delta}$ could be consequently determined as desired. Since θ_0 is not known, the bootstrap principle suggests using $\hat{\theta}$ to replace θ_0: generate many, many samples (say, B in total) from a distribution with value $\hat{\theta}$, and for each sample, construct an estimate of θ, say θ_j^*, $j = 1, 2, L, B$. The distribution of $\hat{\theta} - \theta_0$ is then approximated by that of $\theta^* - \hat{\theta}$, the quantiles of which are used to form an approximate confidence interval.

Let us use the above Poisson distribution as an example for estimating an approximate 95% confidence interval. See the following code:

the difference of the bootstrapped estimations and the original MLE

diff <-lambda_hat_MLE_Bootstrap1 - lambda_hat_MLE

find the 2.5% and 97.5% quantiles

delta <- quantile(diff, probs=c(0.025,0.975), names=FALSE)

find the upper and lower quantiles

delta_up <- delta [2]

delta_down<- delta [1]

hist(diff, main="sampling distribution of boot_lambda_hat_MLE1 - lambda_hat_MLE", xlab = "difference", breaks=50, prob=TRUE)

abline(v= delta_down, col = "blue")

abline(v= delta_up, col = "red")

calculate the upper and lower confidence bound for the CI

L <- lambda_hat_MLE - delta_up

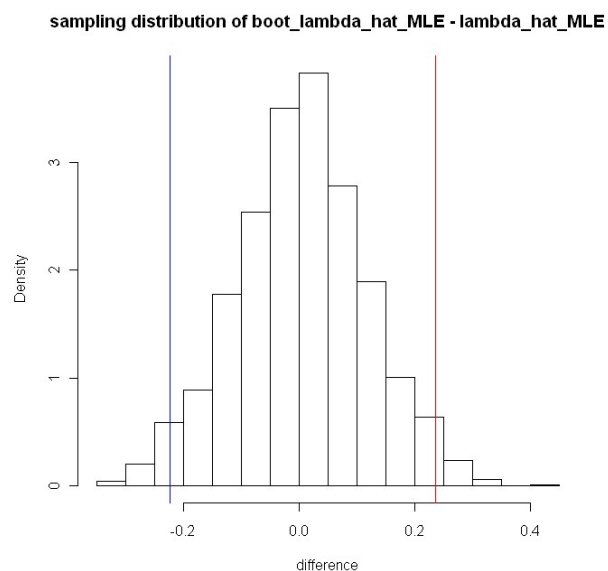U <- lambda_hat_MLE - delta_down

the output:

> L

[1] 3.663331

> U

[1] 4.106665

So the desired confidence interval is [3.663331, 4.106665], which is close to the CI obtained from asymptotic distribution.

The histogram of the difference:

sampling distribution of boot_lambda_hat_MLE - lambda_hat_MLE

Exercise

The file *gamma-arrivals.txt* contains another set of gamma-ray data, this one consisting of the times between arrivals (inter-arrival times) of 3935 photons (units are seconds). Assume the Gamma distribution is a good model for the data:

$$f(x \mid \alpha, \beta) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}, \quad \text{for } x \geq 0$$

where both alpha and beta are unknown.

  a. For maximum likelihood method, using the bootstrap method to simulate the sampling distributions of the estimated parameters. Plot out the histograms, and find the standard errors. Bootstrap 2000 estimations. Please do this using both versions of Bootstrapping for both alpha and beta (that is, you will have to draw four histograms).
  b. For each histogram, please also plot the normal curve, is the histogram close to the normal curve? Please also plot the qqnorm plot for each histogram, is the qq plot close to a straight line?
  c. For the method of moments, repeat what you are asked to do in problem a). How do the estimated standard errors of the two methods compare? Which estimator is better and why? Also, please redo part b) for the results from method of moments.
  d. For the results of maximum likelihood, use the Fisher information method to form approximate 95% confidence intervals for both the parameters.
  e. For the results of maximum likelihood, use the bootstrap (that is, the methods on page 9) to form approximate 95% confidence intervals for both the parameters. What are the lengths of the confidence intervals? For this part, you can either use Bootstrap version 1 or 2.
  f. For the method of moments, repeat what you are asked to do prob e). How do the confidence intervals for the two methods compare? Which estimator is better and why?

*Notes: to sample from a Gamma distribution, please google or help the function "**rgamma**" in R.*