Parameter estimation using R

M P Gururajan, Hina A Gokhale and Dayadeep Monder

Indian Institute of Technology Bombay, Mumbai

In this session, we are going to learn some R commands for parameter estimation. Specifically, we solve some of the problems from Ross using R.

1 River flooding

This is problem number 6 in Chapter 7 of (the fifth edition of) Ross.

Let us first consider the problem of estimating the value of a 100-year flood given the flood data from 1929 to 1965 of Blackstone River in Rhode Island, USA. We are asked to assume that the discharge follows lognormal distribution. The 100-year flood value v is defined as follows: $P(D \geq v) = 0.01$ where D is the discharge.

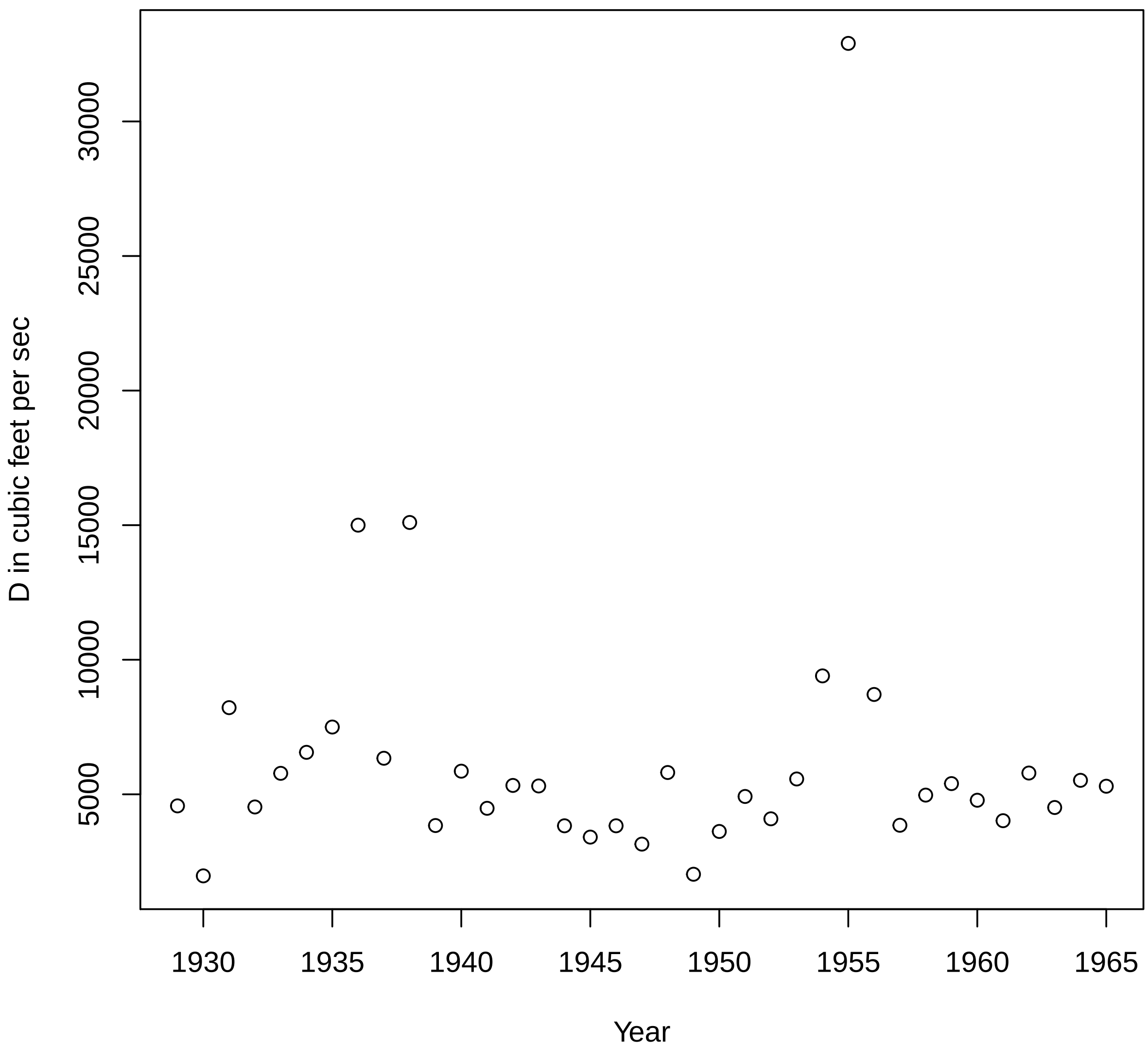
Let us first read the data, view it and plot it:

```
X <- read.csv("FloodData.csv")
X

##    Year Flood.discharge..cubic.ft.per.sec.
## 1  1929                               4570
## 2  1930                               1970
## 3  1931                               8220
## 4  1932                               4530
## 5  1933                               5780
## 6  1934                               6560
## 7  1935                               7500
## 8  1936                              15000
## 9  1937                               6340
## 10 1938                              15100
## 11 1939                               3840
## 12 1940                               5860
## 13 1941                               4480
## 14 1942                               5330
## 15 1943                               5310
## 16 1944                               3830
## 17 1945                               3410
## 18 1946                               3830
## 19 1947                               3150
## 20 1948                               5810
## 21 1949                               2030
## 22 1950                               3620
## 23 1951                               4920
```

```
## 24 1952                                 4090
## 25 1953                                 5570
## 26 1954                                 9400
## 27 1955                                32900
## 28 1956                                 8710
## 29 1957                                 3850
## 30 1958                                 4970
## 31 1959                                 5398
## 32 1960                                 4780
## 33 1961                                 4020
## 34 1962                                 5790
## 35 1963                                 4510
## 36 1964                                 5520
## 37 1965                                 5300

plot(X[,1],X[,2],xlab="Year",ylab="D in cubic feet per sec")
```



Since it is given that the data follows lognormal, let us take logarithm, and use the t-distribution to identify the value of the discharge for which the probability cdf is 0.99 (because we only know the sample mean and sample standard deviation. Note that the identified discharge value has to be transformed back from logarithmic space to real space to report the answer – namely, greater than about 19083 ft^3/sec.

```
Y <- log(X[,2])
Z <- mean(Y) + qt(0.99,36)*sd(Y)
exp(Z)

## [1] 19082.96
```

2 Uniform random variables

This is the 29th problem in Chapter 7 of (the fifth edition of) Ross.

Let us consider the problem of 95% confidence interval estimate of random variables which have the same distribution as the number of uniform (0,1) random variables that need to be summed to exceed 1. We are asked to use random numbers to generate 36 random variables and use the data to solve the problem.

In the following, we first generate 36 random variates. Using these, we calculate the sample mean and sample standard deviation, to evaluate the confidence interval for the expectation value, namely, the mean:

$$E[N] = \left(\bar{x} - t_{\alpha/2,n-1}\frac{s}{\sqrt{N}}, \bar{x} + t_{\alpha/2,n-1}\frac{s}{\sqrt{N}}\right) \tag{1}$$

```
Y <- c(36,1)
for(k in 1:36){
  N=0
  sum=0
  while(sum<1){
    sum = sum + runif(1)
    N = N+1}
  Y[k] = N
}
lower <- mean(Y) - qt(0.975,35)*sd(Y)/6
upper <- mean(Y) + qt(0.975,35)*sd(Y)/6
lower

## [1] 2.592406

mean(Y)

## [1] 2.944444

upper

## [1] 3.296483
```

From these two limits, it is clear that the expectation value could be e. We can check this by generating more data and hence reducing the spread.

```
M = 100000
Y <- c(M,1)
for(k in 1:M){
  N=0
  sum=0
  while(sum<1){
    sum = sum + runif(1)
    N = N+1}
  Y[k] = N
}
lower <- mean(Y) - qt(0.975,M-1)*sd(Y)/sqrt(M)
upper <- mean(Y) + qt(0.975,M-1)*sd(Y)/sqrt(M)
lower
```

```
## [1] 2.717634

mean(Y)

## [1] 2.72307

upper

## [1] 2.728506
```

It indeed is converging to e.

3 Log-normal distribution

This is the solved problem 7.2f in Chapter 7 of (the fifth edition of) Ross.

The data given is the lengths (in mm) of 10 grains of metallic sand from a pile; it is given that by Kolmogorov's law of fragmentation these sand particles follow log-normal distribution. We are asked now to estimate the percentage of particles in the pile with lengths between 2 and 3 mm.

```
l <- c(2.2, 3.4, 1.6, 0.8, 2.7, 3.3, 1.6, 2.8, 2.5, 1.9)
x <- log(l)
xbar <- mean(x)
s <- sd(x)
xbar

## [1] 0.7504035

s

## [1] 0.435063

n1 <- (log(3)-xbar)/s
n2 <- (log(2)-xbar)/s
n1

## [1] 0.800364

n2

## [1] -0.1316046

n <- pnorm(n1) - pnorm(n2)
n

## [1] 0.3406015
```

So, the answer is about 34%.

4 Confidence interval for mean

This is the solved problem 7.3a and 7.3e of Chapter 7 of Ross (fifth edition). In this problem, a number is sent 9 times; it is received with an error with mean zero and variance 4. Given the numbers that are received, we have to construct the 95% confidence interval.

```
signal <- c(5,8.5,12,15,7,9,7.5,6.5,10.5)
xbar <- mean(signal)
s <- sd(signal)
lowerlimit <- xbar - qnorm(0.975)*sqrt(4)/sqrt(9)
upperlimit <- xbar + qnorm(0.975)*sqrt(4)/sqrt(9)
lowerlimit

## [1] 7.693357

upperlimit

## [1] 10.30664
```

In case the standard deviation is not known, we can use the sample standard deviation and the t-distribution to estimate the limits.

```
signal <- c(5,8.5,12,15,7,9,7.5,6.5,10.5)
xbar <- mean(signal)
s <- sd(signal)
lowerlimit <- xbar - qt(0.025,df=8)*s/sqrt(9)
upperlimit <- xbar + qt(0.025,df=8)*s/sqrt(9)
lowerlimit

## [1] 11.36919

upperlimit

## [1] 6.630806
```

Note that not knowing the standard deviation and using the t-distribution increases the uncertainty range.

5 One sided confidence interval

This is solved example 7.3b of Ross. It is a follow-up on 7.3a that we solved above – of the 95% confidence interval for a signal that was sent 9 times and with known variance. In this case, we want to calculate the upper and lower confidence intervals. Of course, the calculation is rather straight-forward.

```
signal <- c(5,8.5,12,15,7,9,7.5,6.5,10.5)
xbar <- mean(signal)
s <- sd(signal)
onesided_upper <- xbar - qnorm(0.95)*sqrt(4)/sqrt(9)
onesided_lower <- xbar + qnorm(0.95)*sqrt(4)/sqrt(9)
onesided_lower

## [1] 10.09657
```

```
onesided_upper

## [1] 7.903431
```

Thus, the one sided confidence interval on the upper side is $(7.903, +\infty)$ and on the lower side is $(-\infty, 10.097)$

6 Estimating the sample size for a given confidence interval

This is the solved problem 7.3d in Ross. Here we are given the standard deviation (0.3 pounds) and are asked to identify the sample size that would help us identify the mean to within ± 0.1 with 95% confidence. We first calculate z value for 95% confidence:

```
qnorm(0.975)

## [1] 1.959964
```

Given this is 1.96 and the known standard deviation is 0.3, for the mean to be within ± 0.1, we know that $\frac{1.96 \times 0.3}{\sqrt{n}} \leq 0.1$. Hence, the same size should be greater than $(0.588/0.1)^2 = 34.57$ or, 35.

7 Confidence interval for standard deviation

This is solved problem 7.3i of Ross. In this problem, we are given the thickness of 10 washers (in inches) and we are asked to calculate the 90% confidence interval for the standard deviation. Given the $100(1-\alpha)$ confidence interval for σ^2 of a sample with a sample variance of s^2 is given by

$$\left(\frac{(n-1)s^2}{\chi^2_{\alpha/2,n-1}}, \frac{(n-1)s^2}{\chi^2_{1-\alpha/2,n-1}} \right) \tag{2}$$

the following script accomplishes the task:

```
t <- c(0.123,0.133,0.124,0.125,0.126,0.128,0.120,0.124,0.130,0.126)
s <- var(t)
lowerlimit <- 9*s/qchisq(0.05,9)
upperlimit <- 9*s/qchisq(0.95,9)
sqrt(lowerlimit)

## [1] 0.006079568

sqrt(upperlimit)

## [1] 0.002695187
```

Notice the small difference between the results from our code and what is given in the textbook. What do you think is the reason for this discrepancy?

8 Difference in means

This is solved problem 7.4a of Ross. Two data sets are given and their variances are known. We are asked to calculate the two sided and one sided confidence interval for the difference in means of these two normal populations. The following code accomplishes that:

```
A <- c(36,54,44,52,41,37,53,51,38,44,36,35,34,44)
B <- c(52,60,64,44,38,48,68,46,66,70,52,62)
varA <- 40
varB <- 100
lowerlimit <-  mean(A) - mean(B) - qnorm(0.025)*sqrt(varA/14 + varB/12)
upperlimit <-  mean(A) - mean(B) + qnorm(0.025)*sqrt(varA/14 + varB/12)
lowerlimit

## [1] -6.491114

upperlimit

## [1] -19.60412

onesided_lower <- mean(A) - mean(B) - qnorm(0.05)*sqrt(varA/14 + varB/12)
onesided_lower

## [1] -7.545227
```

9 Confidence interval for difference in means

This is solved problem 7.4b of Ross. In this problem, the capacities (in Ampere hours) of batteries produced using two different techniques are given – for 12 and 14 random samples produced using techniques I and II respectively. We are asked to calculate the two sided and one-sided confidence interval for the difference in the means. The following script accomplishes the task:

```
I <- c(140,132,136,142,138,150,150,154,152,136,144,142)
II <- c(144,134,132,130,136,146,140,128,128,131,150,137,130,135)
xbar <- mean(I)
ybar <- mean(II)
n <- 12
m <- 14
s1 <- var(I)
s2 <- var(II)
sp <- sqrt(((n-1)*s1+(m-1)*s2)/(n+m-2))
a <- 0.1
xbar-ybar-qt(0.5*a,n+m-2)*sp*sqrt((1/m)+(1/n))

## [1] 11.93041

xbar-ybar+qt(0.5*a,n+m-2)*sp*sqrt((1/m)+(1/n))

## [1] 2.498164

xbar-ybar+qt(0.5*a,n+m-2)*sp*sqrt((1/m)+(1/n))

## [1] 2.498164
```

10 Home work

- Write an R script to solve the following problem (solved problem 7.3f of Ross).

  You are given the random selection of resting pulse of 15 members of a health club: 54,63,58,72,49,92,70,73,69,104,48,66, You have to determine 95% confidence interval of the resting pulse of the members of the club and the 95% lower confidence interval.

- Using Monte Carlo method, carry out the following integration:

$$f = \int_0^1 \sqrt{1-x^2}dx \tag{3}$$

  Note that R can do the integration numerically:

```
f <- function(x) sqrt(1-x^2)
integrate(f,0,1)$value

## [1] 0.7853983
```

  You can compare the numerical solution; from the standard deviation of the data, calculate the 95% confidence interval and check that the solution from numerical integration agrees with the numerical integration value above.

- Peruse the help file for `integrate`. Notice that integration limits can be $\pm\infty$. R can handle these limits!!