Approximation of the CDF of Normal Distribution

Yuance He

09/13/2018

Abstract

This presentation report is the whole analysis process of Monte Carlo method approximation of normal distribution, which contains comparison table and bias plots so that we can test the veracity of estimation.

First Approximation

The model I use is the Monte Carlo's model which is

$$\widehat{\Psi}(t) = \frac{1}{n}\sum_{i=1}^{n} I(X_i \leq t), \tag{1}$$

I start with n=100, t=0 as my first approximation by using the following code:

```
j <- 0
x <- rnorm(100)
#generating 100 N(0,1) numbers randomly, n=100
for (i in 1:100) {
  if (x[i]<=0){
#setting t=0
    j=j+1
  }else{
    j=j
  }

}
y <- j/100
```

Here my first output 0.46 is one approximated value of P(x<=0) in N(0,1) given n=100. Similarly, I get in total 27 approximated value for every combination of t and n by replacement of t and n. And the following table is the comparison between the true value of P(x<=t) and the approximated one.

```
#Collecting data in form of matrix
df <- matrix(nrow = 9, ncol = 5)
colnames(df) <- c("t", "n=100", "n=1000", "n=10000", "True Value")
t <- c(0.0,0.67,0.84,1.28,1.65,2.32,2.58,3.09,3.72)
df[,1] <- t
df[,2] <- c(0.46, 0.81, 0.84, 0.91, 0.97, 0.97, 0.99, 1, 1)
df[,3] <- c(0.52, 0.741, 0.801, 0.9, 0.948, 0.989, 0.995, 1, 1)
df[,4] <- c(0.5015, 0.75, 0.7957, 0.9026, 0.9505, 0.9885, 0.9934, 0.9995, 1)
df[,5] <- c(pnorm(t))
```

```
#Building table
knitr::kable(df, booktabs = TRUE,
             caption = 'Comparison')
```

Table 1: Comparison

t	n=100	n=1000	n=10000	True Value
0.00	0.46	0.520	0.5015	0.5000000
0.67	0.81	0.741	0.7500	0.7485711
0.84	0.84	0.801	0.7957	0.7995458
1.28	0.91	0.900	0.9026	0.8997274
1.65	0.97	0.948	0.9505	0.9505285
2.32	0.97	0.989	0.9885	0.9898296
2.58	0.99	0.995	0.9934	0.9950600
3.09	1.00	1.000	0.9995	0.9989992
3.72	1.00	1.000	1.0000	0.9999004

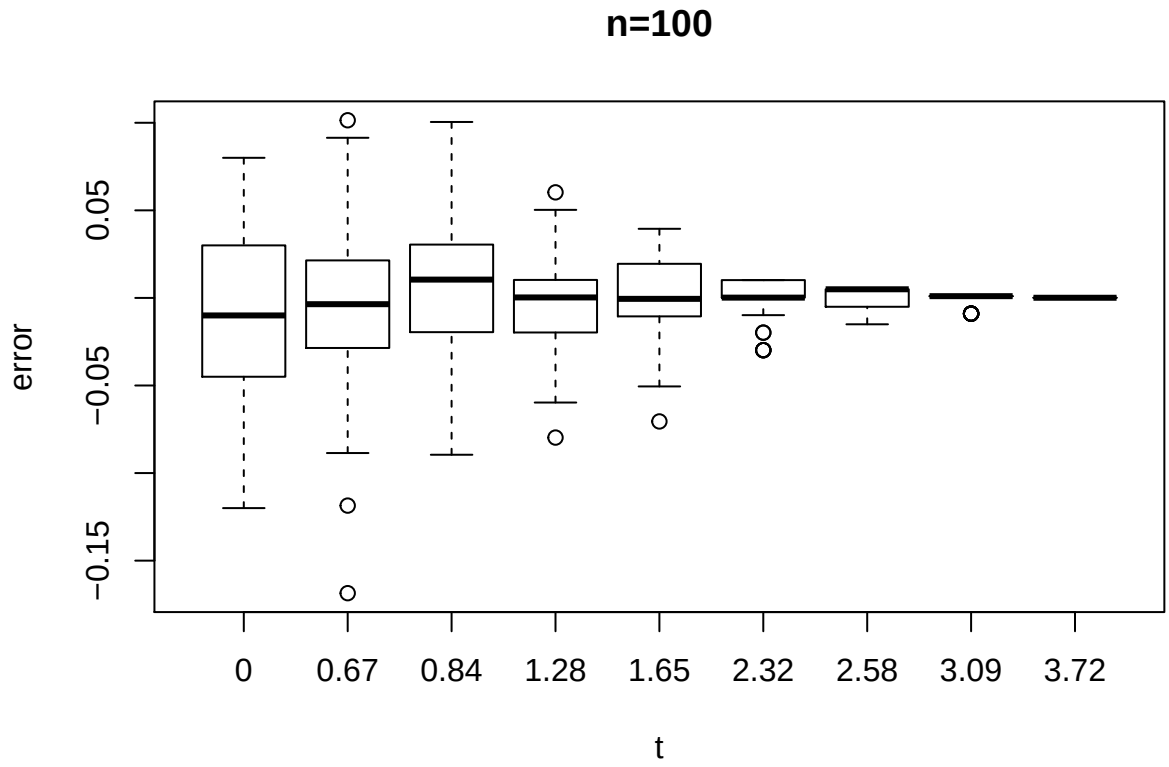
Repetition of the experiment

After taking single approximation, two questions come up to my mind: due to different n, which estimation model is the most precise one? Does t have influence on the veracity of experiment? Therefore, I repeat the whole experiment 100 times in order to analyze the bias of estimation starting at n=100,t=0 by using the following code:

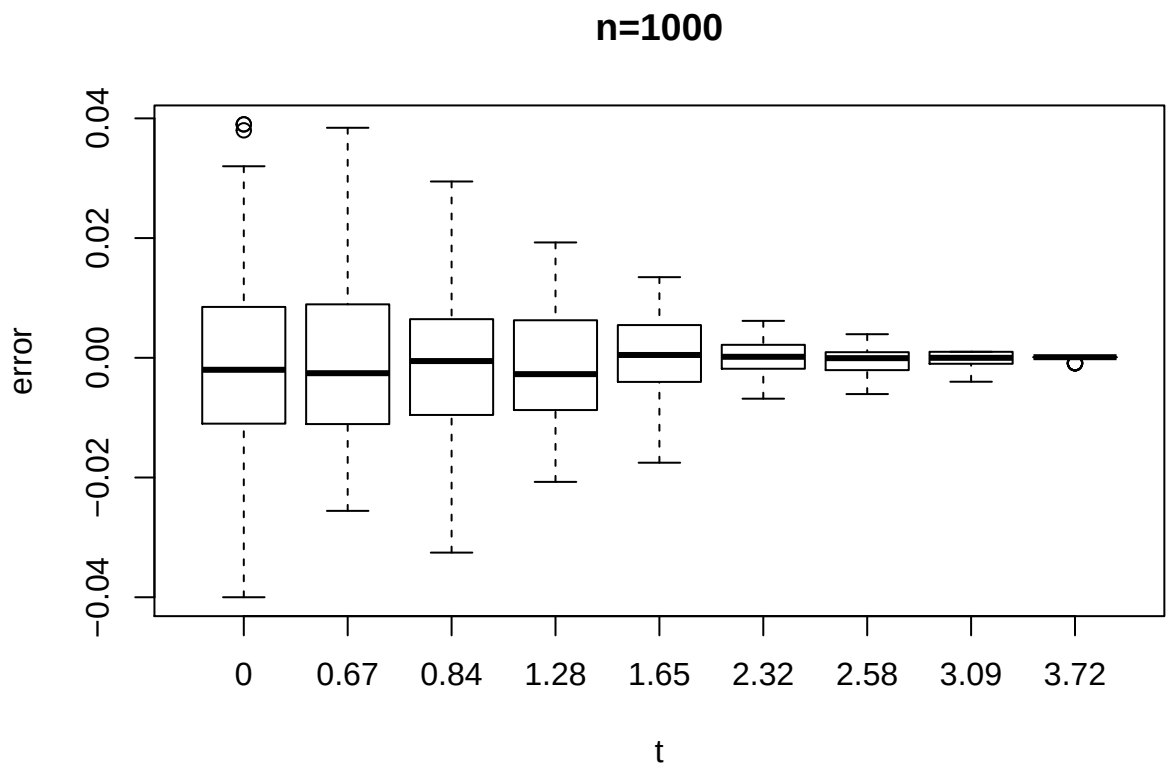
```
er <- function(n,t){
  x <- matrix(rnorm(100*n),nrow=100,ncol=n)
b <- array()
for (l in 1:100) {
  j <- 0
  for (m in 1:n) {
    if(x[l,m]<=t){
      j=j+1
    }else{
      j=j
    }
  }
  b[l] <- j
}
error <- b/n-pnorm(t)
return(error)
}
```

Now after taking 100 approximation errors at t=0, n=100, similarly I get all the errors for every combination of t and n by replacement of t and n. And the following is the plots of errors of all t at different n:

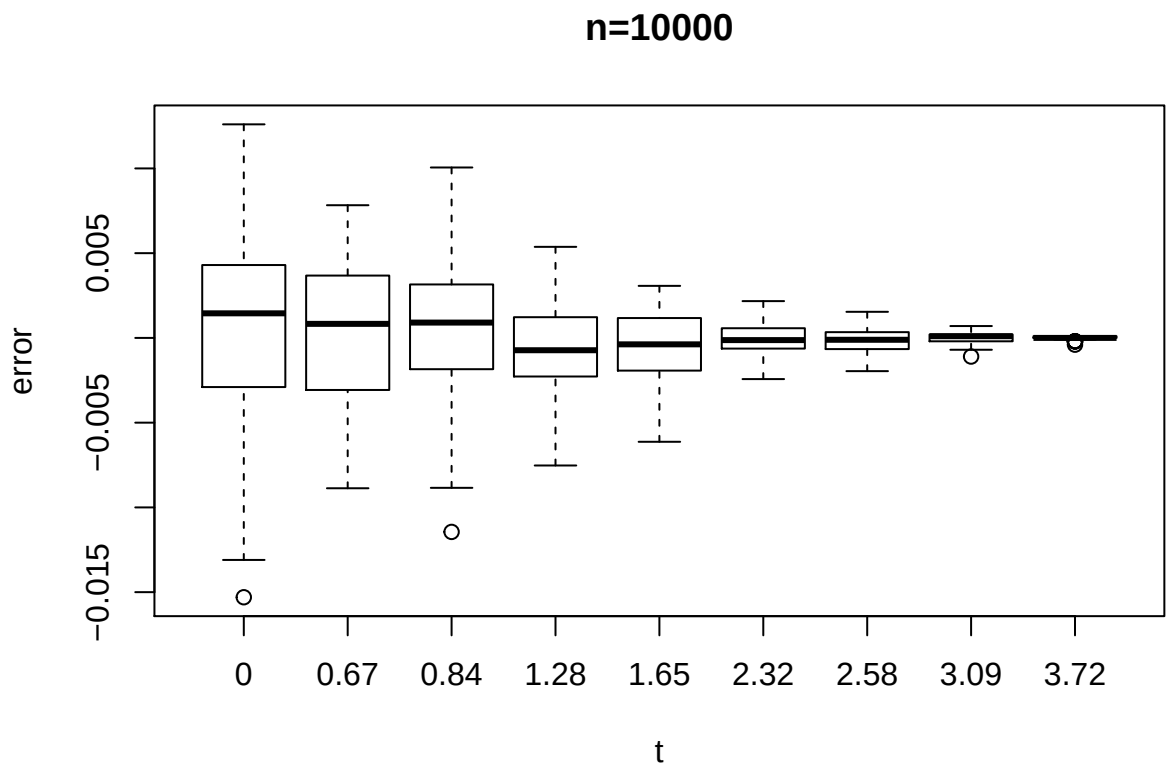
```
t <- c(0.0,0.67,0.84,1.28,1.65,2.32,2.58,3.09,3.72)
list1 <- c(er(100,t[1]), er(100,t[2]), er(100,t[3]), er(100,t[4]),
        er(100,t[5]), er(100,t[6]), er(100,t[7]), er(100,t[8]), er(100,t[9]))
list2 <- c(er(1000,t[1]), er(1000,t[2]), er(1000,t[3]), er(1000,t[4]),
      er(1000,t[5]), er(1000,t[6]), er(1000,t[7]), er(1000,t[8]), er(1000,t[9]))
list3 <-c(er(10000,t[1]), er(10000,t[2]), er(10000,t[3]), er(10000,t[4]),
  er(10000,t[5]), er(10000,t[6]), er(10000,t[7]), er(10000,t[8]), er(10000,t[9]))
df1 <- data.frame(rep(t,each=100), list1 )
df2 <- data.frame(rep(t,each=100), list2 )
df3 <- data.frame(rep(t,each=100), list3 )
boxplot(list1~rep(t,each=100),xlab="t", ylab="error", main="n=100")
```




```
boxplot(list2~rep(t,each=100),xlab="t", ylab="error", main="n=1000")
```




```
boxplot(list3~rep(t,each=100),xlab="t", ylab="error", main="n=10000")
```



Result

From the box plots of errors and all t under different n, I can conclude that the model's veracity is increasing when data size n is increasing, and the more extreme t value we take, the less scale of error is.