R handout: Monte Carlo Method and Application

Yuhui Yao

# Methodology

1. Motivation

* In practice, we may meet some hard integrations which is really hard to solve. For example, in one of my working papers, we have this integrand
* where , and are constants. is the c.d.f. of the standard normal distribution. is the quantile function the standard normal distribution. is the quantile function of with degrees of freedom.
* The analytical solution probably is desperate, but we can use other methods to obtain the numerical solution. This handout will show a way to solve integrations numerically.

1. Example. where . Find out
   1. Method 1: Direct Integration (By hand)
   * As we know, the p.d.f. of
   * where is the p.d.f. of the standard normal distribution. The expectation
   1. Method 2: Direct Integration (In R)
      * R code
      * # define the p.d.f. of Y  
         Y.pdf <- function(y) 2 \* dnorm(y)  
           
         # define the integrand of the expecation of Y  
         E.Y <- function(y) y \* Y.pdf(y)  
           
         # Integrate the integrand  
         integrate(E.Y, lower = 0, upper = Inf)  
         # Result: 0.7978846
   2. Monte Carlo Method
   * Suppose is a random variable with a p.d.f. and a c.d.f. over a domain . Also, we have a sample from with observations.
   * Besides, because of Central Limit Theorem, asymptotically follows where is the mean of and is the variance of . So, when , $\bar{X} \xrightarrow{p} E(X)$. Hence, intuitively, the result is more precise with a larger sample size.
     1. Method 3: Monte Carlo Method with Transformation 1
        + Motivation
        + Our target distribution may be hard to be found, but the original distribution may be not. Hence, we simulate a sample from the original distribution and transform it to our target, and then, calculate the expectation.
        + The analytical part
        + Because
        + where is the number of simulation and is a sample from .
        + R code
        + # set seed to make this process repeatable  
           set.seed(12345)  
             
           # set the number of simulation  
           n <- 1000  
             
           # simulate a sample from the standard   
           # normal distribution  
           X.s <- rnorm(n)  
             
           # calculate the mean of the absoluate values  
           mean(abs(X.s))  
           # Result: 0.7944
     2. Method 4: Monte Carlo Method with Transformation 2
        + Motivation
        + Both of our target distribution and the original distribution may be hard to be found, so we use other relatively simple distributions which have same domains as the ones of our target distribution or the original distribution, to simulate our target, and then, calculate the expectation.
        + The analytical part
        + Because has a same domain as ,
        + where is the number of simulation and is a sample from .
        + R code
        + # set seed to make this process repeatable  
           set.seed(12345)  
             
           # set the number of simulation  
           n <- 1000  
             
           # define the p.d.f. of Y  
           Y.pdf <- function(y) 2 \* dnorm(y)  
             
           # simulate a sample from the standard   
           # normal distribution  
           Y.s <- rexp(n)  
             
           # calculate the mean of the absoluate values  
           mean(Y.s \* Y.pdf(Y.s) / dexp(Y.s))  
           # Result: 0.7729
2. Summary
   1. Comparison of these methods

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| * + Methods | * + Method 1 | * + Method 2 | * + Method 3\* | * + Method 4\* |
| * + Results | * + 0.7979 | * + 0.7979 | * + 0.7944 | * + 0.7729 |

* + \* Method 3 and Method 4 have 1000 simulations.
  1. Steps of Monte Carlo Method
     1. Check the domain
     2. Simulate an appropriate sample over the domain
     3. Calculate the mean via an appropriate transformation

1. Practice: suppose , where . Use Monte Carlo Method to find .

* Hint: You need to show the analytical part and the coding part in R, separately. Also, you may need to use and in R

9

Kaggle, *Crime in Vancouver*, Link: https://www.kaggle.com/wosaku/crime-in-vancouver 2017.