Statistical​ ​Methods​ ​Applications​ ​in​ ​Financial​ ​Engineering

Individual Report for Variance Reduction

Stat 428 Statistical Computing

Professor Uma Virendra Ravat

Group 8: Donghan Liu

Contribution on the Group Project:

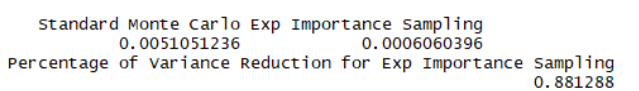
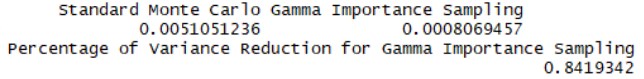
In our first meeting, which is on Nov, 12, Sunday afternoon, all of my teammates were present and we had detailed conversation about the general direction of our project, discussed about the project proposal, made sure everyone has the evenly distributed works. We also found some books that could directly or indirectly support our report, which was extremely helpful for my peers’ part. Through the two hours discussion, we finally figure out the purpose of this final project and be prepared to divide the works to each person.

During the meeting, I helped with the team leader and Matt to find the financial book that have relationship on application of technique that we learned from the class, then I worked with other members to demonstrate the feasibility of those methods. The procedure is time-consuming, so we generally spent most of our time on researching the potential useful statistical methods. Eventually, we realized that [this book](https://camjclub.wikispaces.com/file/view/Monte+Carlo+Methods+In+Financial+Engineering.pdf)1 might be helpful for the entire statistical technique implement. That book was named “Monte Carlo Methods in Financial Engineering” and written by Paul Glasserman is another major resource for our methods usage besides the textbooks.

After the proposal and the overall research direction finished, we decided to move on and choose the topics that individually intend to utilize. The choice process was not easy since none of us have experience with financial engineer before, so we have to make a fresh start. After reading several dozen of outside resource and textbooks, we ultimately determine to choose MCMC, optimization, and variance reduction to implement throughout the project.

I am responsible with variance reduction, so primarily I would like to discuss about my part in detail. From a bigger picture, the purpose of variance reduction part is principally to choose the best method to reduce the variance regarding this specific situation. The financial data that we used is Walmart stock price from November 29, 2000 to November 29, 2010. We firstly tried to check its normality, and it fails to meet the assumption. Then we read some notes from the book and outside resources, and realized that the stock price basically follow the standard lognormal distribution. Based on the relationship between normal distribution and lognormal distribution, we took the exponential of the data in order to convert them to normal distribution, and check the normality of the data that after conversion. The result of the Q-Q plot proves that the exponential of data follow the normal distribution, indicating that the original data follow the lognormal distribution. In accordance with this assumption, the variance reduction would use this trace to push on.

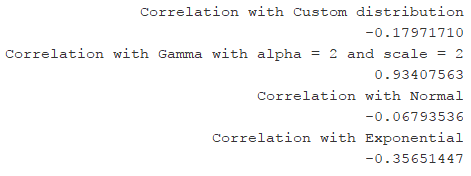
To start with, there are four methods that I learned from this class and aim to reduce the variance, which are *Importance Sampling, Stratified Importance Sampling, Control Variates and Antithetic Variates*. The purpose of my part is to find the best efficient technique to reduce the variance in respect of standard lognormal distribution.

Following the instruction provided by both textbooks and lecture notes, the first technique that I used is *importance sampling*. Because the importance function is uncertain, I have to attempt to use multiple distributions as importance function. For example, the first function that I used is Exponential Distribution with rate = 1, the generated result is that the variance was reduced by approximately 88%. Whereas, the second importance function is Gamma Distribution with rate = 1 and scale = 1, the percentage of variance reduction is slightly smaller than the case of Exponential Distribution with rate = 1 as importance function, which is 84%. In the third case, the inverse transform method was implemented, and the function similar with exp(sqrt(-2\*log(u))) 2. However, the inverse transform method is least efficient among the other two techniques, which only have around 52% variance could be reduced. https://lh3.googleusercontent.com/-B9RN0gVvomW39om3Zh4YSDNbQ3CDwvLGvG6g1VE14m8-OdiPFulwdP6nlh-z63d7OGp5g9syUsH9qNWnav4rzssDDkM-dKZctogKvCbaiNBiWSG1BKGT8xBMaEPl3IksrXWyJay

In addition, the reason why I choose these three distribution as importance function is that they all have the same domain with lognormal distribution, which is (0, inf). In conclusion, exponential distribution with rate = 1 has the best effects on reducing the targeted distribution variance within importance sampling.

Then, I process the *stratified importance sampling* to test if it is the best variance reduction approach regarding the standard lognormal distribution. Subsequently, the algorithm and method that were taught in the lecture note and textbook were utilized here. Along with 1000 replications, the percentage of variance reduction for stratified importance sampling is much smaller than the importance sampling with exponential distribution, which only has approximately 76% reduction rate.



Furthermore, the *control variate* is the next approach to test for efficiency. Because of the theorem of control variate, I have to find the distribution that has the largest correlation with the lognormal distribution in order to maximum the variance reduction, so I tried one custom function:  ((exp(-0.5))/(u\*sqrt(2\*pi)))3, gamma distribution with alpha = 2 and scale = 2, standard normal distribution, and exponential distribution with rate = 1. I tested them one by one, and eventually figured out that the correlation between gamma with alpha = 2 and scale = 2 and standard lognormal distribution is 0.934, which is a highest correlation among the four distributions.  I would therefore use gamma distribution with alpha = 2 and scale = 2 to process to next reduction step. By implementing the methods that introduced in the book, I generated the result as follows:

## Without Control Variate    With Control Variate

##             0.025398488             0.003292324

so the rate of reducing variance is 87%, which is slightly smaller than the importance sampling with exponential distribution. https://lh6.googleusercontent.com/8Ebj0NLSXzyIb35EnTaUxvQ42-bW00IEshYtkfK4-1uHP-8AgZpG0Hx92Q07m8n84cvDI6xyOyH2a_7l2o-txBkk5GoiZ73E7wl1ZurDu4CZQ8GjApCQbT8evVT9nfJkt6t6bRlp

Finally, after the determination of control variate, the *antithetic variate* is the last technique to achieve the purpose of decreasing variance. The lecture notes and textbook have clear and organized instructions, so they are extremely helpful for my part. The generated percentage is close to 90%, which has the highest rate on reducing variance among these four techniques.https://lh3.googleusercontent.com/LnjzdmN6IIjcbXvDTFZaveUNmuAWJ2JeFqFab3-wDiw8yDRLr0y2Zl2EtKIDzkArmM7g4oHNYLVYctKydXnIPDbcY6tn11R5CiNqRVRzeT7R-nmFqYyQszEhVmyViLSBZgmuPuNK

Overall, Antithetic variate has the highest efficiency on variance reduction regarding the standard lognormal distribution. However, this project is not intend to prove which technique is better, rather, it is trying to study the powerful of different methods on reducing variance in respect of financial data. Thus, the limitation of this project is that it only shows the specific financial data, and could not apply to the general cases.

Outcome of the Project:

According to results that generated from the different techniques, among the different importance functions, the best case in importance sampling methods is 88%, stratified importance sampling has the outcome of reduction percentage of 73%, control variate could reduce the variance by 87% and 90% variance is able to be decreased by antithetic variate. Thus, the rate is clear to see that antithetic variate is capable of reducing the highest variance in comparison with other techniques. The general conclusion would be, regarding the standard lognormal distribution, the antithetic variate is the relatively best approach to reduce variance while dealing with financial data.

Appendix:

1Financial book link:

<https://camjclub.wikispaces.com/file/view/Monte+Carlo+Methods+In+Financial+Engineering.pdf>

2Based on the inverse transform method and standard lognormal distribution

3Custom function in control variate was chosen based on the graph trend

4Code:  
**#**Importance sampling

#Exponential as importance

log\_normal = function(x){

 return(((exp(-(log(x)^2)/2))/(x\*sqrt(2\*pi))))

}

importance = function(){

 data = numeric(10000)

 data = rexp(length(data))

 return(mean(log\_normal(data) / dexp(data)))

}

#Importance Sampling for exp

log\_normal1 = numeric(1000)

for (i in 1:length(log\_normal1)){

 log\_normal1[i] = importance()

}

#Gamma as importance

impGamma = function(){

 data = numeric(10000)

 data = rgamma(length(data),1,1)

 return(mean(log\_normal(data) / dgamma(data,1,1)))

}

#Importance Sampling for gamma

log\_normal2 = numeric(1000)

for (i in 1:length(log\_normal2)){

 log\_normal2[i] = impGamma()

}

#Standard Monte Carlo estimator

n = 1000

estimates = numeric(n)

for(i in 1:n){

 estimates[i]=mean(log\_normal(runif(n)))

 }

se = sd(estimates)

#Percentage for exp importance sampling variance reduction

comp = c(se,sd(log\_normal1)/sqrt(1000))

names(comp) = c('Standard Monte Carlo','Exp Importance Sampling')

comp

perc = (se-sd(log\_normal1)/sqrt(1000))/se

names(perc) = c("Percentage of Variance Reduction for Exp Importance Sampling")

perc

#Percentage for gamma importance sampling variance reduction

comp = c(se,sd(log\_normal2)/sqrt(1000))

names(comp) = c('Standard Monte Carlo','Gamma Importance Sampling')

comp

perc = (se-sd(log\_normal2)/sqrt(1000))/se

names(perc) = c("Percentage of Variance Reduction for Gamma Importance Sampling")

perc

#Inverse transform method

u = runif(1000)

x = exp(sqrt(-2\*log(u)))

fg = log\_normal(x)/exp((-(log(x))^2)/2)

invi = sd(fg)/sqrt(1000)

#Percentage for inverse sampling variance reduction

comp = c(se,invi)

names(comp) = c('Standard Monte Carlo','Inverse Importance Sampling')

comp

perc = (se-invi)/se

names(perc) = c("Percentage of Variance Reduction For Inverse Transform")

perc

##Stratified Importance Sampling

log\_normal = function(x){

 return(((exp(-(log(x)^2)/2))/(x\*sqrt(2\*pi)))\*(x>0))

}

m=1000 # number of replicates

k=10 # number of strata

n=1000 # number of experiments

T2= numeric(k)

estimates=matrix(0,n,2)

for(i in 1:n){ #Standard Monte Carlo estimator

 estimates[i,1]=mean(log\_normal(runif(m)))

for(j in 1:k) #Stratified Importance Sampling

 T2[j]=(1/k)\*mean(log\_normal(runif(m/k,(j-1)/k,j/k)))

 estimates[i,2]=sum(T2)

}

comp = apply(estimates,2,sd)

perc = (comp[1]-comp[2])/comp[1]

names(perc) = c("Percentage of Variance Reduction")

perc

names(comp) = c('Standard Monte Carlo','Stratified Importance Sampling')

comp

##Control Variates

#First try ((exp(-0.5))/(u\*sqrt(2\*pi)))

u = runif(10000)

f1 = function(u){

 ((exp(-0.5))/(u\*sqrt(2\*pi)))\*(u>0)

}

g1 = function(u){

 ((exp(-(log(u)^2)/2))/(u\*sqrt(2\*pi)))\*(u>0)

}

cov1 = cor(f1(u),g1(u))

#Then try gamma distribution with alpha = 2 and scale = 2

u = runif(10000)

f2 = function(u){

 dgamma(u,2,2)

 }

g2 = function(u){

 ((exp(-(log(u)^2)/2))/(u\*sqrt(2\*pi)))\*(u>0)

}

cov2 = cor(f2(u),g2(u))

#Third, try normal distribution

u = runif(10000)

f3 = function(u){

 dnorm(u)

 }

g3 = function(u){

 ((exp(-(log(u)^2)/2))/(u\*sqrt(2\*pi)))\*(u>0)

}

cov3 = cor(f3(u),g3(u))

#Finally try exponential

u = runif(10000)

f4 = function(u){

 dexp(u)

}

g4 = function(u){

 ((exp(-(log(u)^2)/2))/(u\*sqrt(2\*pi)))\*(u>0)

}

cov4 = cor(f4(u),g4(u))

table = (c(cov1,cov2,cov3,cov4))

names(table) = c("Correlation with Custom distribution","Correlation with Gamma with alpha = 2 and scale = 2","Correlation with Normal","Correlation with Exponential")

table

###Percentage of reduction

u = runif(10000)

epct = 2-3/sqrt(exp(1)) #Expected value for Gamma with (2,2)

a = -cov(g2(u),f2(u))/var(f2(u))

orig = g2(u)

revi = orig + a\*(f2(u) - epct)

corr = c(var(orig),var(revi))

names(corr) = c("Without Control Variate","With Control Variate")

corr

perc = (var(orig) - var(revi))/var(orig)

names(perc) = c("Percentage of Control Variates Variance Reduction")

perc

##Antithetic Variates

anti = function(x,R = 10000, anti = TRUE){

 u = runif(R/2)

 if(!anti){

   v = runif(R/2)

 }

 else{

   v = 1-u

 }

 u = c(u,v)

 cdf = numeric(length(x))

 for (i in 1:length(x)){

   g = log(x[i])\*exp(-(u \* log(x[i]))^2 / 2)

   cdf[i] = mean(g) / sqrt(2\*pi) + 0.5

 }

 cdf

}

#Check the difference of CDF value

x = seq(.1, 2.5, length=5)

woanti = anti(x,anti = FALSE)

wanti = anti(x)

cdftable = matrix(0,5,2)

cdftable[,1] = woanti

cdftable[,2] = wanti

colnames(cdftable) = c("Without Antithetic Variate","With Antithetic Variate")

rownames(cdftable) = x

cdftable

#Check the difference of variance between with antithetic variate and without antithetic variate.

m = 1000

antiF = antiT = numeric(m)

x = 1.95

for (i in 1:m){

 antiF[i] = anti(x,R = 1000,anti = FALSE)

 antiT[i] = anti(x,R = 1000)

}

antitable = c(var(antiF),var(antiT))

names(antitable) = c("Variance without antithetic variate","Variance with antithetic variate")

antitable

perc = (var(antiF) - var(antiT))/var(antiF)

names(perc) = c("Percentage of Antithetic Variates Variance Reduction")

perc