

hw3prob2

Lisa He

2/25/2019

calcuP <- function(K){  
r<- 0.05  
sig <-.25  
  
S0<-100  
mT<-1/12  
d1 <- (log(S0/K)+(r+sig\*\*2/2)\*mT)/(sig\*sqrt(mT))  
d2<- d1 - sig\*sqrt(mT)  
P<- -S0\*pnorm(-d1)+K\*exp(-r\*mT)\*pnorm(-d2)  
  
return(P)}  
print(calcuP(60))

## [1] 3.6202e-13

print(calcuP(70))

## [1] 3.219925e-07

print(calcuP(80))

## [1] 0.001435591

print(calcuP(90))

## [1] 0.1922204

print(calcuP(100))

## [1] 2.669393

sensitosig <-function(sig){  
r<- 0.05  
K<-100  
  
S0<-100  
mT<-1/12  
d1 <- (log(S0/K)+(r+sig\*\*2/2)\*mT)/(sig\*sqrt(mT))  
d2<- d1 - sig\*sqrt(mT)  
P<- -S0\*pnorm(-d1)+K\*exp(-r\*mT)\*pnorm(-d2)  
  
return(P)  
}  
print(sensitosig(0.1))

## [1] 0.9532625

print(sensitosig(0.15))

## [1] 1.523818

print(sensitosig(0.2))

## [1] 2.096267

print(sensitosig(0.25))

## [1] 2.669393

print(sensitosig(0.3))

## [1] 3.242768

print(sensitosig(0.35))

## [1] 3.816191

print(sensitosig(0.4))

## [1] 4.389547

K = 60, P60 = 3.6202e-13 K = 70, P70 = 3.219925e-07 K = 80, P80 = 0.001435591 K = 90, P90 = 0.1922204 K = 100, P100 = 2.669393

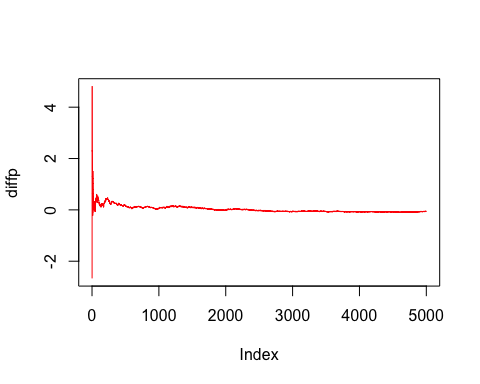
When sigman changes the put price changes a lot, so it is sensitive to sigma

1. [2.669393*0.99 ,2.669393*1.01]=[2.642699,2.696087]

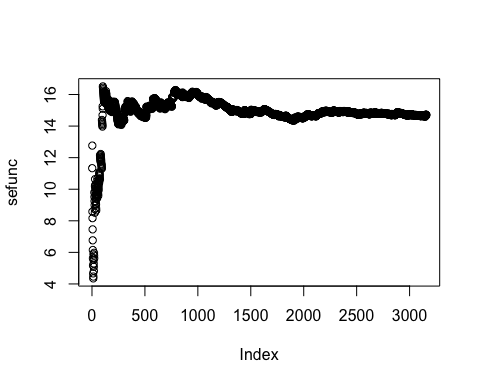
mT<-1/12  
delta <- 1/252  
totT<- mT/delta  
r<- 0.05  
sig <-.25  
  
S0<-100  
mT<-1/12  
K<-100  
diffp <-c()  
sefunc<- c()  
maefunc<-c()  
  
numSim<-5000  
Psim<-c()  
psum <-0  
for ( n in 1:numSim){  
simX <- c()  
simX[1]<- 100  
  
  
for (i in 2:totT){  
   
 simX[i]<- simX[i-1]+r\*simX[i-1]\*delta + sig\*simX[i-1]\*sqrt(delta)\*rnorm(1)  
}  
Psim[n] <- max(0,K-simX[totT])\*exp(-r\*mT)  
psum<- psum+ Psim[n]  
  
Pestimate<- ( psum/ n)   
diffp[n]<-Pestimate- 2.669393  
if(abs(diffp[n])<2.669393\*0.01){  
 print(n)  
}  
}

Need 3158 simulations at least

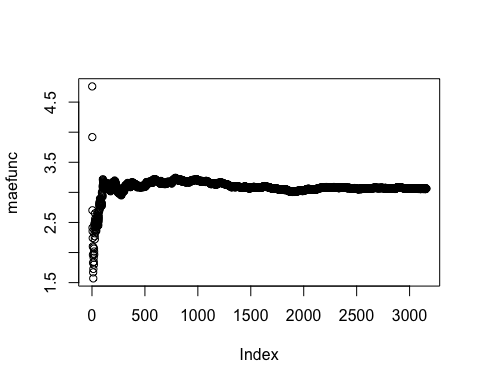
plot(diffp, type = "l",col ="red")



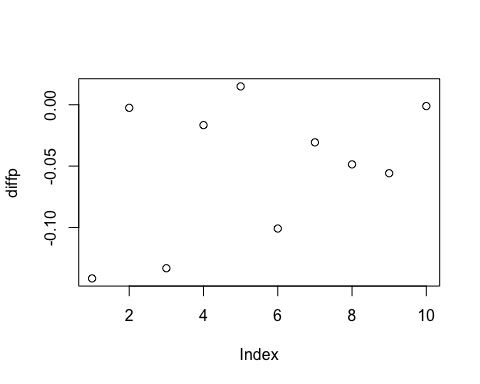
delta <- 1/252  
totT<- mT/delta  
r<- 0.05  
sig <-.25  
  
S0<-100  
mT<-1/12  
K<-100  
diffp <-c()  
sefunc<-c()  
maefunc<-c()  
  
numSim<-3158  
Psim<-c()  
psum <-0  
for ( n in 1:numSim){  
simX <- c()  
simX[1]<- 100  
  
  
for (i in 2:totT){  
   
 simX[i]<- simX[i-1]+r\*simX[i-1]\*delta + sig\*simX[i-1]\*sqrt(delta)\*rnorm(1)  
}  
Psim[n] <- max(0,K-simX[totT])\*exp(-r\*mT)  
psum<- psum+ Psim[n]  
  
Pestimate<- ( psum/ n)   
diffp[n]<-Pestimate- 2.669393  
  
SEsum <-0  
MAEsum<-0  
for ( j in 1:n){  
 SEsum<- SEsum+(Psim[j]-Pestimate)^2  
 MAEsum<- MAEsum+ abs(Psim[j]-Pestimate)  
}  
sefunc[n] <- SEsum/(n-1)  
  
maefunc[n] <-MAEsum / (n -1)  
  
}  
  
plot(sefunc)



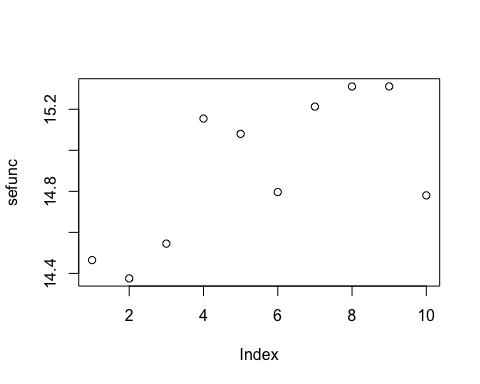
plot(maefunc)



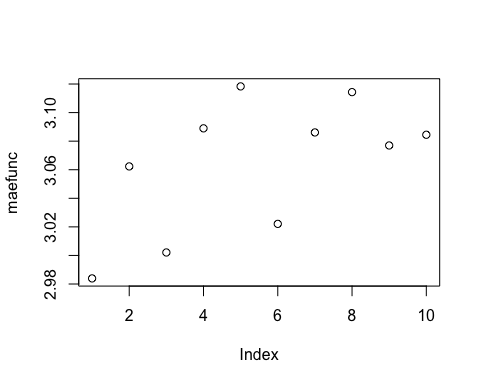
r<- 0.05  
sig <-.25  
  
S0<-100  
mT<-1/12  
K<-100  
diffp <-c()  
sefunc<-c()  
maefunc<-c()  
  
numSim<-3158  
  
for (k in 1:10){  
delta <- 1/252/k  
totT<- mT/delta  
  
Psim<-c()  
psum <-0  
for ( n in 1:numSim){  
simX <- c()  
simX[1]<- 100  
  
  
for (i in 2:totT){  
   
 simX[i]<- simX[i-1]+r\*simX[i-1]\*delta + sig\*simX[i-1]\*sqrt(delta)\*rnorm(1)  
}  
Psim[n] <- max(0,K-simX[totT])\*exp(-r\*mT)  
psum<- psum+ Psim[n]  
}  
  
  
  
Pestimate<- ( psum/ numSim)   
diffp[k]<-Pestimate- 2.669393  
  
  
SEsum <-0  
MAEsum<-0  
for ( j in 1:n){  
 SEsum<- SEsum+(Psim[j]-Pestimate)^2  
 MAEsum<- MAEsum+ abs(Psim[j]-Pestimate)  
}  
sefunc[k] <- SEsum/(numSim-1)  
  
maefunc[k] <-MAEsum / (numSim -1)  
  
  
}  
plot(diffp)



plot(sefunc)



plot(maefunc)

 The diffp is not converging after we reduced delta to 1/2520, so number of simulations are more important. It's better to run 2 times as many simulations than to reduce delta in half