贝叶斯统计课程作业

赵越 17734316

2018/5/8

## 第四题

情景分析：构建MV模型，改变两个参数和的输入，观察投资组合的变化。 ###4.1 不变而改变 ####4.1.1 为对角阵

Sigma11 <- diag(rep(1,5))  
Mu11 <- rep(0,5)  
Mu12 <- rep(-0.02,5)  
Mu13 <- rep(0.02,5)  
Mu14 <- seq(-0.02,0.02,length.out = 5)  
cbind(Mu11,Mu12,Mu13,Mu14)

## Mu11 Mu12 Mu13 Mu14  
## [1,] 0 -0.02 0.02 -0.02  
## [2,] 0 -0.02 0.02 -0.01  
## [3,] 0 -0.02 0.02 0.00  
## [4,] 0 -0.02 0.02 0.01  
## [5,] 0 -0.02 0.02 0.02

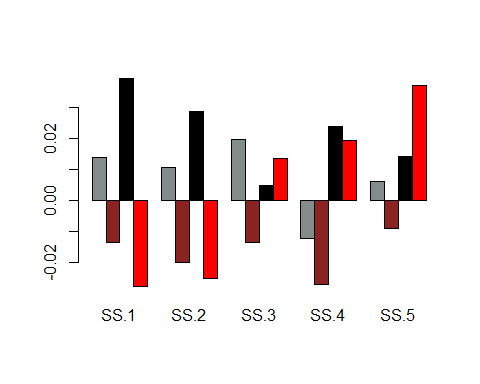
Sigma11

## [,1] [,2] [,3] [,4] [,5]  
## [1,] 1 0 0 0 0  
## [2,] 0 1 0 0 0  
## [3,] 0 0 1 0 0  
## [4,] 0 0 0 1 0  
## [5,] 0 0 0 0 1

R11 <- mvrnorm(n=10000, Mu11, Sigma11)  
R12 <- mvrnorm(n=10000, Mu12, Sigma11)  
R13 <- mvrnorm(n=10000, Mu13, Sigma11)  
R14 <- mvrnorm(n=10000, Mu14, Sigma11)  
Rt11 <- as.timeSeries(R11)  
Rt12 <- as.timeSeries(R12)  
Rt13 <- as.timeSeries(R13)  
Rt14 <- as.timeSeries(R14)  
u11 <- apply(Rt11,2,mean)  
u12 <- apply(Rt12,2,mean)  
u13 <- apply(Rt13,2,mean)  
u14 <- apply(Rt14,2,mean)  
ut1 <- data.frame(u1=u11,u2=u12,u3=u13,u4=u14)  
ut1 <- t(as.matrix(ut1))   
ut1

## SS.1 SS.2 SS.3 SS.4 SS.5  
## u1 0.01412115 0.01082598 0.01991624 -0.01213314 0.006287832  
## u2 -0.01351268 -0.02002276 -0.01365170 -0.02703536 -0.008823878  
## u3 0.03942572 0.02867697 0.00504540 0.02397066 0.014443338  
## u4 -0.02757868 -0.02503777 0.01380643 0.01947835 0.037147189

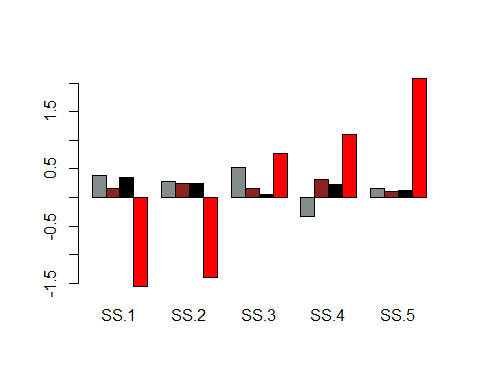
barplot(ut1,beside = TRUE,col = c("azure4","brown4","black","red"),axes = FALSE,ylim=c(min(ut1),max(ut1)))  
axis(2)



##计算权重  
a<-matrix(c(1,1,1,1,1),ncol=1)  
w11<-(solve(cov(Rt11))%\*%u11)/rep(t(a)%\*%solve(cov(Rt11))%\*%u11,5)  
w12<-(solve(cov(Rt12))%\*%u12)/rep(t(a)%\*%solve(cov(Rt12))%\*%u12,5)  
w13<-(solve(cov(Rt13))%\*%u13)/rep(t(a)%\*%solve(cov(Rt13))%\*%u13,5)  
w14<-(solve(cov(Rt14))%\*%u14)/rep(t(a)%\*%solve(cov(Rt14))%\*%u14,5)  
wt1 <- data.frame(w1=w11,w2=w12,w3=w13,w4=w14)  
wt1 <- t(as.matrix(wt1))   
wt1

## SS.1 SS.2 SS.3 SS.4 SS.5  
## w1 0.3768973 0.2824971 0.52712516 -0.3382376 0.1517180  
## w2 0.1610395 0.2477732 0.16081510 0.3218706 0.1085016  
## w3 0.3536663 0.2494558 0.04798165 0.2189322 0.1299641  
## w4 -1.5497192 -1.3971897 0.77447693 1.0949169 2.0775150

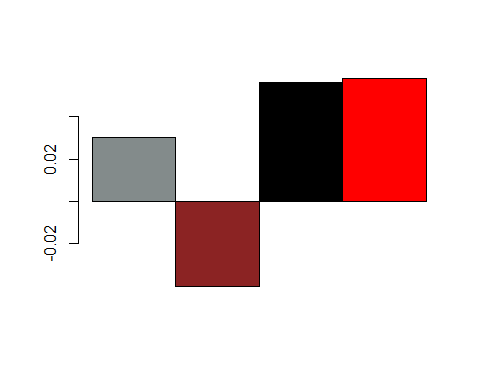
barplot(wt1,beside = TRUE,col = c("azure4","brown4","black","red"),axes = FALSE,ylim=c(min(wt1),max(wt1)))  
axis(2)



##计算夏普比  
SR11<-t(w11)%\*%u11/(sqrt(t(w11)%\*%cov(Rt11)%\*%w11))  
SR12<-t(w12)%\*%u12/(sqrt(t(w12)%\*%cov(Rt12)%\*%w12))  
SR13<-t(w13)%\*%u13/(sqrt(t(w13)%\*%cov(Rt13)%\*%w13))  
SR14<-t(w14)%\*%u14/(sqrt(t(w14)%\*%cov(Rt14)%\*%w14))  
SRt1 <- data.frame(SR1=SR11,SR2=SR12,SR3=SR13,SR4=SR14)  
SRt1 <- t(as.matrix(SRt1))   
SRt1

## [,1]  
## SR1 0.02997011  
## SR2 -0.03990488  
## SR3 0.05625766  
## SR4 0.05789956

barplot(SRt1,beside = TRUE,col = c("azure4","brown4","black","red"),axes = FALSE,ylim=c(min(SRt1),max(SRt1)))  
axis(2)



#### 4.1.2 为正相关矩阵

Sigma31 <- matrix(c(1,0.5,0.6,0.7,0.8,0.5,1,0.6,0.7,0.8,0.6,0.6,1,0.7,0.8,0.7,0.7,0.7,1,0.8,0.8,0.8,0.8,0.8,1),5,5)  
Mu31 <- rep(0,5)  
Mu32 <- rep(-0.02,5)  
Mu33 <- rep(0.02,5)  
Mu34 <- seq(-0.02,0.02,length.out = 5)  
cbind(Mu31,Mu32,Mu33,Mu34)

## Mu31 Mu32 Mu33 Mu34  
## [1,] 0 -0.02 0.02 -0.02  
## [2,] 0 -0.02 0.02 -0.01  
## [3,] 0 -0.02 0.02 0.00  
## [4,] 0 -0.02 0.02 0.01  
## [5,] 0 -0.02 0.02 0.02

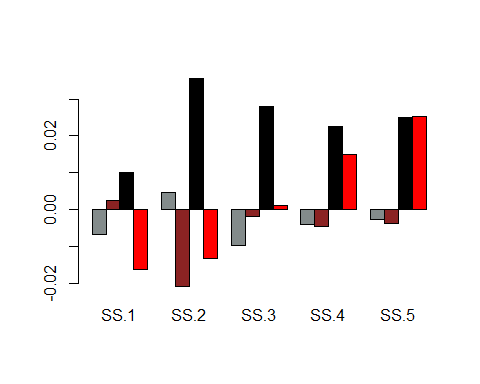
Sigma31

## [,1] [,2] [,3] [,4] [,5]  
## [1,] 1.0 0.5 0.6 0.7 0.8  
## [2,] 0.5 1.0 0.6 0.7 0.8  
## [3,] 0.6 0.6 1.0 0.7 0.8  
## [4,] 0.7 0.7 0.7 1.0 0.8  
## [5,] 0.8 0.8 0.8 0.8 1.0

R31 <- mvrnorm(n=10000, Mu31, Sigma31)  
R32 <- mvrnorm(n=10000, Mu32, Sigma31)  
R33 <- mvrnorm(n=10000, Mu33, Sigma31)  
R34 <- mvrnorm(n=10000, Mu34, Sigma31)  
Rt31 <- as.timeSeries(R31)  
Rt32 <- as.timeSeries(R32)  
Rt33 <- as.timeSeries(R33)  
Rt34 <- as.timeSeries(R34)  
u31 <- apply(Rt31,2,mean)  
u32 <- apply(Rt32,2,mean)  
u33 <- apply(Rt33,2,mean)  
u34 <- apply(Rt34,2,mean)  
ut3 <- data.frame(u1=u31,u2=u32,u3=u33,u4=u34)  
ut3 <- t(as.matrix(ut3))   
ut3

## SS.1 SS.2 SS.3 SS.4 SS.5  
## u1 -0.006501832 0.004602711 -0.009507319 -0.003990821 -0.002662582  
## u2 0.002586528 -0.020583114 -0.001820297 -0.004561890 -0.003591117  
## u3 0.010017744 0.035488294 0.027941640 0.022694629 0.025014440  
## u4 -0.016117513 -0.013200506 0.001122652 0.015008142 0.025191107

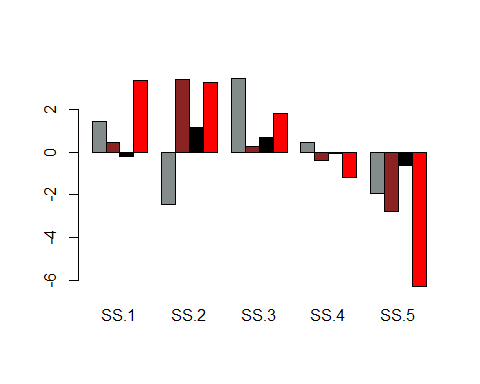
barplot(ut3,beside = TRUE,col = c("azure4","brown4","black","red"),axes = FALSE,ylim=c(min(ut3),max(ut3)))  
axis(2)



##计算权重  
a<-matrix(c(1,1,1,1,1),ncol=1)  
w31<-(solve(cov(Rt31))%\*%u31)/rep(t(a)%\*%solve(cov(Rt31))%\*%u31,5)  
w32<-(solve(cov(Rt32))%\*%u32)/rep(t(a)%\*%solve(cov(Rt32))%\*%u32,5)  
w33<-(solve(cov(Rt33))%\*%u33)/rep(t(a)%\*%solve(cov(Rt33))%\*%u33,5)  
w34<-(solve(cov(Rt34))%\*%u34)/rep(t(a)%\*%solve(cov(Rt34))%\*%u34,5)  
wt3 <- data.frame(w1=w31,w2=w32,w3=w33,w4=w34)  
wt3 <- t(as.matrix(wt3))   
wt3

## SS.1 SS.2 SS.3 SS.4 SS.5  
## w1 1.4585361 -2.462495 3.4547275 0.47927825 -1.9300467  
## w2 0.4656783 3.419620 0.2869677 -0.40370026 -2.7685655  
## w3 -0.1965427 1.172453 0.7111067 -0.06638938 -0.6206278  
## w4 3.3558824 3.281339 1.8104986 -1.16852525 -6.2791953

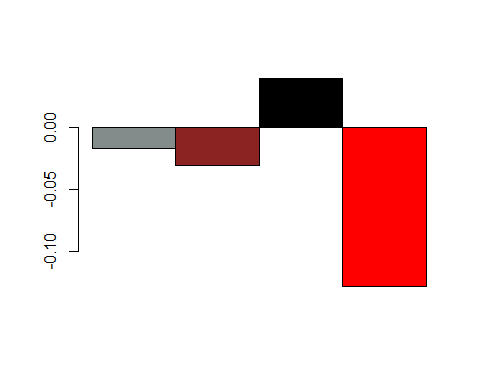
barplot(wt3,beside = TRUE,col = c("azure4","brown4","black","red"),axes = FALSE,ylim=c(min(wt3),max(wt3)))  
axis(2)



##计算夏普比  
SR31<-t(w31)%\*%u31/(sqrt(t(w31)%\*%cov(Rt31)%\*%w31))  
SR32<-t(w32)%\*%u32/(sqrt(t(w32)%\*%cov(Rt32)%\*%w32))  
SR33<-t(w33)%\*%u33/(sqrt(t(w33)%\*%cov(Rt33)%\*%w33))  
SR34<-t(w34)%\*%u34/(sqrt(t(w34)%\*%cov(Rt34)%\*%w34))  
SRt3 <- data.frame(SR1=SR31,SR2=SR32,SR3=SR33,SR4=SR34)  
SRt3 <- t(as.matrix(SRt3))   
SRt3

## [,1]  
## SR1 -0.01694363  
## SR2 -0.03025088  
## SR3 0.03974784  
## SR4 -0.12784552

barplot(SRt3,beside = TRUE,col = c("azure4","brown4","black","red"),axes = FALSE,ylim=c(min(SRt3),max(SRt3)))  
axis(2)



#### 4.1.4 为负相关矩阵

Sigma31 <- matrix(c(1,0.5,0.6,0.7,-0.8,0.5,1,0.6,0.7,-0.8,0.6,0.6,1,0.7,-0.8,0.7,0.7,0.7,1,-0.8,-0.8,-0.8,-0.8,-0.8,1),5,5)  
Mu31 <- rep(0,5)  
Mu32 <- rep(-0.02,5)  
Mu33 <- rep(0.02,5)  
Mu34 <- seq(-0.02,0.02,length.out = 5)  
cbind(Mu31,Mu32,Mu33,Mu34)

## Mu31 Mu32 Mu33 Mu34  
## [1,] 0 -0.02 0.02 -0.02  
## [2,] 0 -0.02 0.02 -0.01  
## [3,] 0 -0.02 0.02 0.00  
## [4,] 0 -0.02 0.02 0.01  
## [5,] 0 -0.02 0.02 0.02

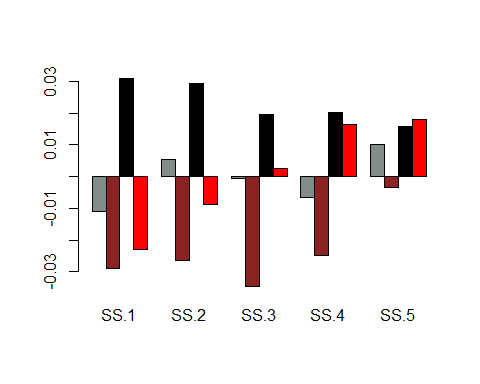
Sigma31

## [,1] [,2] [,3] [,4] [,5]  
## [1,] 1.0 0.5 0.6 0.7 -0.8  
## [2,] 0.5 1.0 0.6 0.7 -0.8  
## [3,] 0.6 0.6 1.0 0.7 -0.8  
## [4,] 0.7 0.7 0.7 1.0 -0.8  
## [5,] -0.8 -0.8 -0.8 -0.8 1.0

R31 <- mvrnorm(n=10000, Mu31, Sigma31)  
R32 <- mvrnorm(n=10000, Mu32, Sigma31)  
R33 <- mvrnorm(n=10000, Mu33, Sigma31)  
R34 <- mvrnorm(n=10000, Mu34, Sigma31)  
Rt31 <- as.timeSeries(R31)  
Rt32 <- as.timeSeries(R32)  
Rt33 <- as.timeSeries(R33)  
Rt34 <- as.timeSeries(R34)  
u31 <- apply(Rt31,2,mean)  
u32 <- apply(Rt32,2,mean)  
u33 <- apply(Rt33,2,mean)  
u34 <- apply(Rt34,2,mean)  
ut3 <- data.frame(u1=u31,u2=u32,u3=u33,u4=u34)  
ut3 <- t(as.matrix(ut3))   
ut3

## SS.1 SS.2 SS.3 SS.4 SS.5  
## u1 -0.01083749 0.005535315 -0.000523682 -0.006434231 0.010277444  
## u2 -0.02876641 -0.026321679 -0.034460107 -0.024717471 -0.003388652  
## u3 0.03085069 0.029425193 0.019733540 0.020166747 0.015930696  
## u4 -0.02295154 -0.008821377 0.002675823 0.016346694 0.018163635

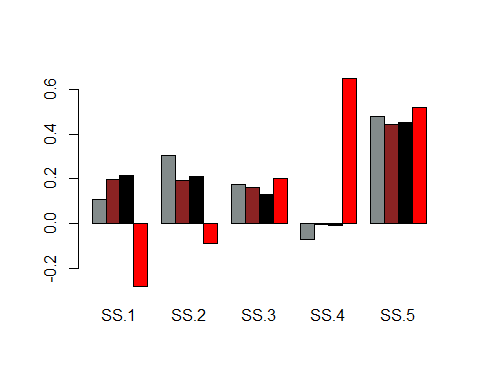
barplot(ut3,beside = TRUE,col = c("azure4","brown4","black","red"),axes = FALSE,ylim=c(min(ut3),max(ut3)))  
axis(2)



##计算权重  
a<-matrix(c(1,1,1,1,1),ncol=1)  
w31<-(solve(cov(Rt31))%\*%u31)/rep(t(a)%\*%solve(cov(Rt31))%\*%u31,5)  
w32<-(solve(cov(Rt32))%\*%u32)/rep(t(a)%\*%solve(cov(Rt32))%\*%u32,5)  
w33<-(solve(cov(Rt33))%\*%u33)/rep(t(a)%\*%solve(cov(Rt33))%\*%u33,5)  
w34<-(solve(cov(Rt34))%\*%u34)/rep(t(a)%\*%solve(cov(Rt34))%\*%u34,5)  
wt3 <- data.frame(w1=w31,w2=w32,w3=w33,w4=w34)  
wt3 <- t(as.matrix(wt3))   
wt3

## SS.1 SS.2 SS.3 SS.4 SS.5  
## w1 0.1085701 0.30361223 0.1766085 -0.068689448 0.4798986  
## w2 0.1995410 0.19294445 0.1642083 -0.001803974 0.4451102  
## w3 0.2158935 0.20898032 0.1307986 -0.008182310 0.4525099  
## w4 -0.2779357 -0.08691977 0.2004470 0.647526011 0.5168825

barplot(wt3,beside = TRUE,col = c("azure4","brown4","black","red"),axes = FALSE,ylim=c(min(wt3),max(wt3)))  
axis(2)



##计算夏普比  
SR31<-t(w31)%\*%u31/(sqrt(t(w31)%\*%cov(Rt31)%\*%w31))  
SR32<-t(w32)%\*%u32/(sqrt(t(w32)%\*%cov(Rt32)%\*%w32))  
SR33<-t(w33)%\*%u33/(sqrt(t(w33)%\*%cov(Rt33)%\*%w33))  
SR34<-t(w34)%\*%u34/(sqrt(t(w34)%\*%cov(Rt34)%\*%w34))  
SRt3 <- data.frame(SR1=SR31,SR2=SR32,SR3=SR33,SR4=SR34)  
SRt3 <- t(as.matrix(SRt3))   
SRt3

## [,1]  
## SR1 0.03110900  
## SR2 -0.12211050  
## SR3 0.15289125  
## SR4 0.06144947

barplot(SRt3,beside = TRUE,col = c("azure4","brown4","black","red"),axes = FALSE,ylim=c(min(SRt3),max(SRt3)))  
axis(2)

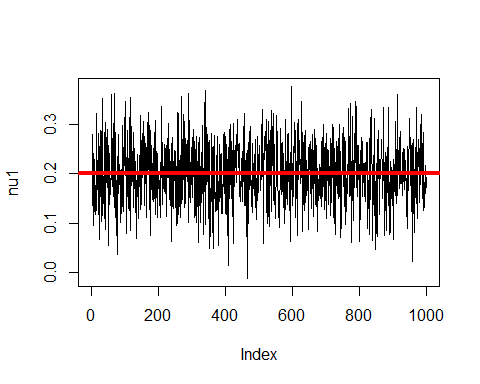
## 

## 第三题

### 3.1 第一种情景

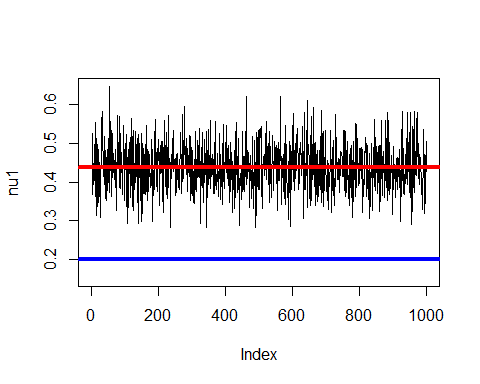
DGP为,用回归 ####3.1.1 与独立

nu1 <- vector()  
for(i in 1:1000){  
 x1 <- rnorm(100,0,1)  
 x2 <- rnorm(100,0,1)  
 beta0 <- 0.8  
 beta1 <- 0.2  
 beta2 <- 0.4  
 epsilon <- rnorm(100,0,0.5)  
 y <- beta0+beta1\*x1+beta2\*x2+epsilon  
 temp <- lm(y~x1)  
 nu1 <- c(nu1,as.numeric(temp$coefficients[2]))  
}  
plot(nu1,type="l")  
abline(h=beta1,lwd=4,col="blue")  
abline(h=mean(nu1),lwd=4,col="red")

 可以看出，与没有明显差异

#### 3.1.2 与多元正态，相关系数为0.6

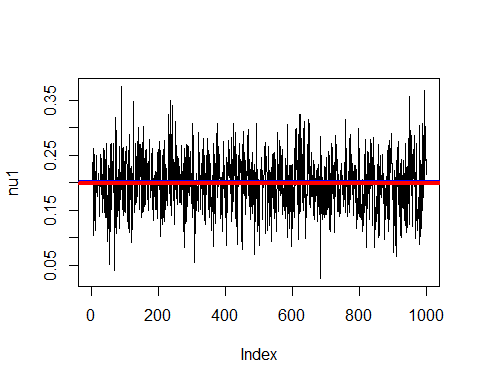
nu1 <- vector()  
for(i in 1:1000){  
 Sigma <- matrix(c(1,0.6,0.6,1),2,2)  
 X <- mvrnorm(n=100, rep(0,2), Sigma)  
 x1 <- as.numeric(X[,1])  
 x2 <- as.numeric(X[,2])  
 beta0 <- 0.8  
 beta1 <- 0.2  
 beta2 <- 0.4  
 epsilon <- rnorm(100,0,0.5)  
 y <- beta0+beta1\*x1+beta2\*x2+epsilon  
 temp <- lm(y~x1)  
 nu1 <- c(nu1,as.numeric(temp$coefficients[2]))  
}  
plot(nu1,type="l",ylim=c(0.15,max(nu1)))  
abline(h=beta1,lwd=4,col="blue")  
abline(h=mean(nu1),lwd=4,col="red")

 发现，明显大于

### 3.2 第二种情景

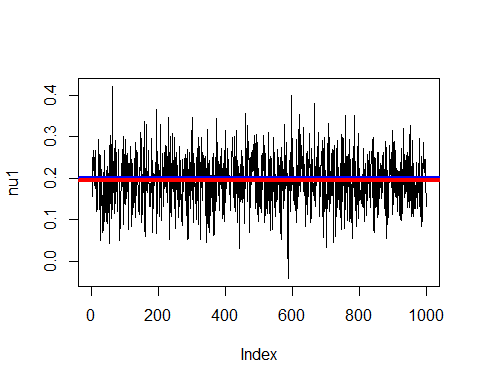
DGP为,用回归 ####3.1.1 与独立

nu1 <- vector()  
for(i in 1:1000){  
 x1 <- rnorm(100,0,1)  
 x2 <- rnorm(100,0,1)  
 beta0 <- 0.8  
 beta1 <- 0.2  
 epsilon <- rnorm(100,0,0.5)  
 y <- beta0+beta1\*x1+epsilon  
 temp <- lm(y~x1+x2)  
 nu1 <- c(nu1,as.numeric(temp$coefficients[2]))  
}  
plot(nu1,type="l")  
abline(h=beta1,lwd=4,col="blue")  
abline(h=mean(nu1),lwd=4,col="red")

 可以看出，与没有明显差异

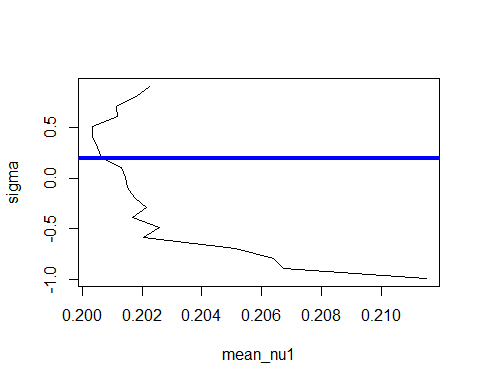
#### 3.1.2 与多元正态，相关系数为0.6

nu1 <- vector()  
for(i in 1:1000){  
 Sigma <- matrix(c(1,0.6,0.6,1),2,2)  
 X <- mvrnorm(n=100, rep(0,2), Sigma)  
 x1 <- as.numeric(X[,1])  
 x2 <- as.numeric(X[,2])  
 beta0 <- 0.8  
 beta1 <- 0.2  
 epsilon <- rnorm(100,0,0.5)  
 y <- beta0+beta1\*x1+epsilon  
 temp <- lm(y~x1+x2)  
 nu1 <- c(nu1,as.numeric(temp$coefficients[2]))  
}  
plot(nu1,type="l")  
abline(h=beta1,lwd=4,col="blue")  
abline(h=mean(nu1),lwd=4,col="red")

 发现，小于

#### 3.1.3 与多元正态，相关系数从负到正

nu1 <- vector()  
mean\_nu1 <- vector()  
sigma <- seq(-0.99,0.99,0.1)  
for(j in 1:length(sigma)){  
 for(i in 1:100){  
 Sigma <- matrix(c(1,sigma[j],sigma[j],1),2,2)  
 X <- mvrnorm(n=100, rep(0,2), Sigma)  
 x1 <- as.numeric(X[,1])  
 x2 <- as.numeric(X[,2])  
 beta0 <- 0.8  
 beta1 <- 0.2  
 epsilon <- rnorm(100,0,0.5)  
 y <- beta0+beta1\*x1+epsilon  
 temp <- lm(y~x1+x2)  
 nu1 <- c(nu1,as.numeric(temp$coefficients[2]))  
 }  
 mean\_nu1[j] <- mean(nu1)  
}  
plot(mean\_nu1,sigma,type="l")  
abline(h=beta1,lwd=4,col="blue")

 可以看到，相关系数为负时会放大，相关系数为正时会稀释