602Lab9

Andrea Aveni

2023-03-27

```
library(MASS)
library(ggplot2)
```

We now generate some bivariate data.

$$Y_i^a \overset{iid}{\sim} \mathcal{N}\left(\theta^a, \Sigma^a\right), \qquad Y_i^b \overset{iid}{\sim} \mathcal{N}\left(\theta^b, \Sigma^b\right)$$

```
n_a=50
aTruetheta=c(1/2,2)
aTrueSigma=matrix(c(1,0.75,0.75,1),nrow=2,ncol=2)
Ya=mvrnorm(n=n_a,mu=aTruetheta,Sigma=aTrueSigma)

n_b=30
bTruetheta=c(1,-1)
bTrueSigma=matrix(c(2,0.75,0.75,1),nrow=2,ncol=2)
Yb=mvrnorm(n=n_b,mu=bTruetheta,Sigma=bTrueSigma)
```

Now we forget about how these data were generated and we want to recover their parameters $\theta^a, \theta^b, \Sigma^a, \Sigma^b$ just looking at the observations.

We start with the following prior for both data sets

$$(\theta, \Sigma) \sim \mathcal{N}\mathrm{IW}\left(\theta_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \lambda_0 = 1, \mathbf{S}_0^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \nu_0 = 2\right)$$

This is the (multivariate) Normal Inverse Wishart. Sampling from this is equivalent to the following two step procedure

$$\Sigma^{-1} | \mathbf{S}_0^{-1}, \nu_0 \sim \mathrm{W}\left(\mathbf{S}_0^{-1}, \nu_0\right)$$

$$\theta | \theta_0, \lambda_0, \Sigma \sim \mathcal{N}\left(\theta_0, \frac{1}{\lambda_0}\Sigma\right)$$

Now it is possible to prove (see Wikipedia or prove it yourself, it's just time consuming) that the posterior is also given by a Normal Inverse Wishart distribution, with updated parameters

$$(\theta, \Sigma) | Y \sim \mathcal{N}\mathrm{IW}\left(\theta_n = \frac{\lambda_0 \theta_0 + n\bar{Y}}{\lambda_0 + n}, \lambda_n = \lambda_0 + n, \mathbf{S}_n^{-1} = \mathbf{S}_0^{-1} + \mathbf{S} + \frac{\lambda n}{\lambda + n}(\bar{Y} - \theta_0)(\bar{Y} - \theta_0)^T, \nu_n = \nu_0 + n\right)$$

Where

$$\mathbf{S} = \sum_{j=1}^{n} (Y_j - \bar{Y})(Y_j - \bar{Y})^T$$

Now let us see how to sample from the posterior in R

```
th_0=c(0,0)
la_0=1
S_0=matrix(c(1,0,0,1),nrow=2,ncol=2)
nu_0=2
```

Posterior for group a

```
n_a=dim(Ya)[1]
barYa=apply(Ya,2,mean)
Sa=matrix(c(0,0,0,0),ncol=2,nrow=2)
for(j in 1:n_a){
  Sa=Sa+(barYa-Ya[j,])%*%t(barYa-Ya[j,])
}

th_a=(la_0*th_0+n_a*barYa)*1/(la_0+n_a)
la_a=la_0+n_a
S_a=solve(solve(S_0)+Sa+la_0/(la_0+n_a)*(barYa-th_0)%*%t(barYa-th_0))
nu_a=nu_0+n_a
```

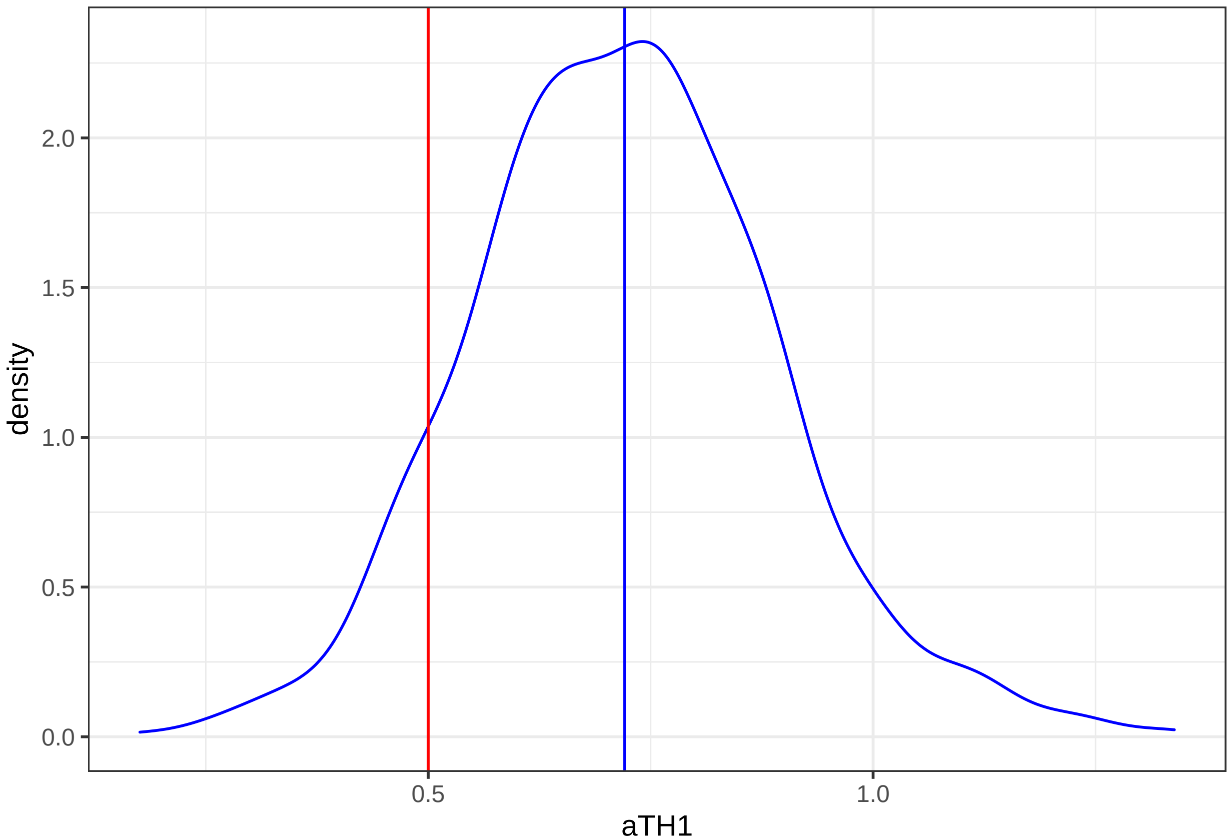
```
aSIGMA=aTHETA=list()
for(j in 1:1000){
  aSigma=solve(rWishart(n=1,nu_a,S_a)[,,1])
  atheta=mvrnorm(n=1,mu=th_a,Sigma=aSigma/la_a)

  aSIGMA[[j]]=aSigma
  aTHETA[[j]]=atheta
}
```

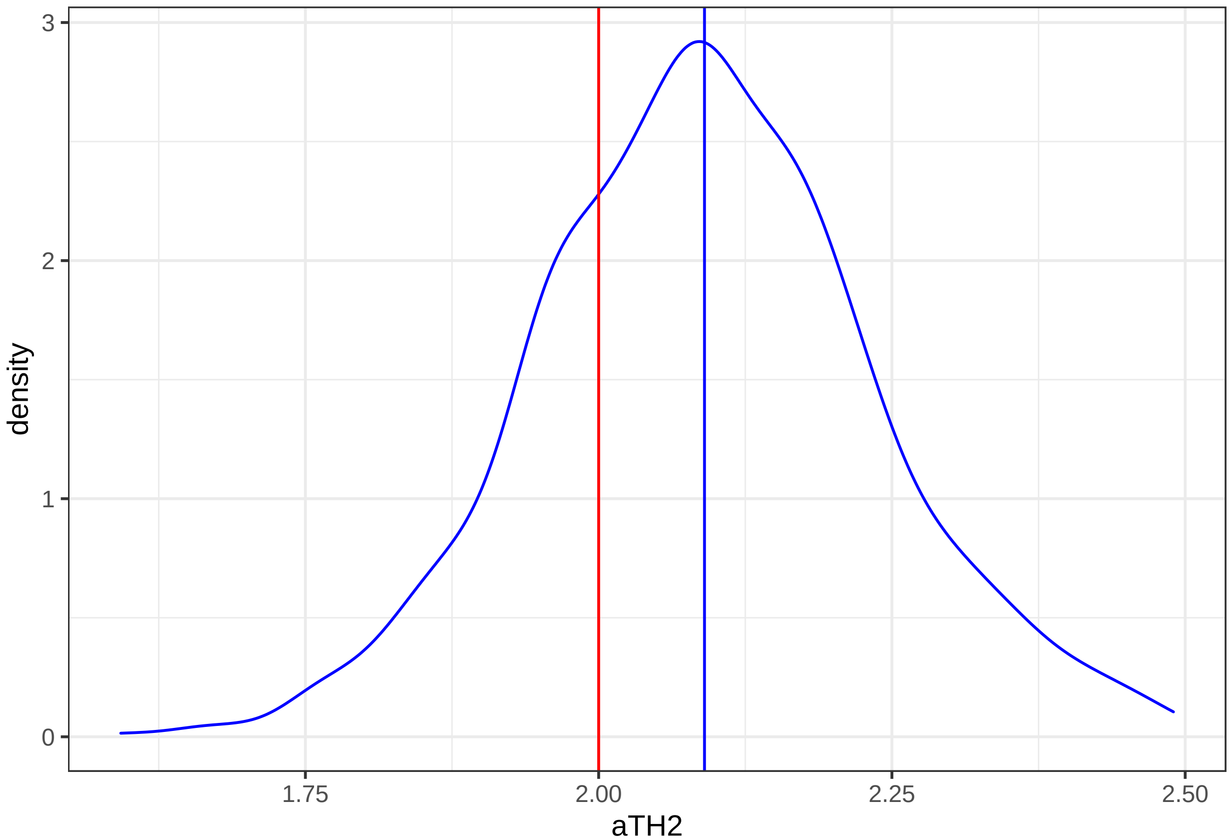
Plot the results

```
aTH1=aTH2=aSI11=aSI12=aSI22=c()
for(j in 1:1000){
  aTH1=c(aTH1,aTHETA[[j]][1])
  aTH2=c(aTH2,aTHETA[[j]][2])
  aSI11=c(aSI11,aSIGMA[[j]][1,1])
  aSI12=c(aSI12,aSIGMA[[j]][1,2])
  aSI22=c(aSI22,aSIGMA[[j]][2,2])
}
```

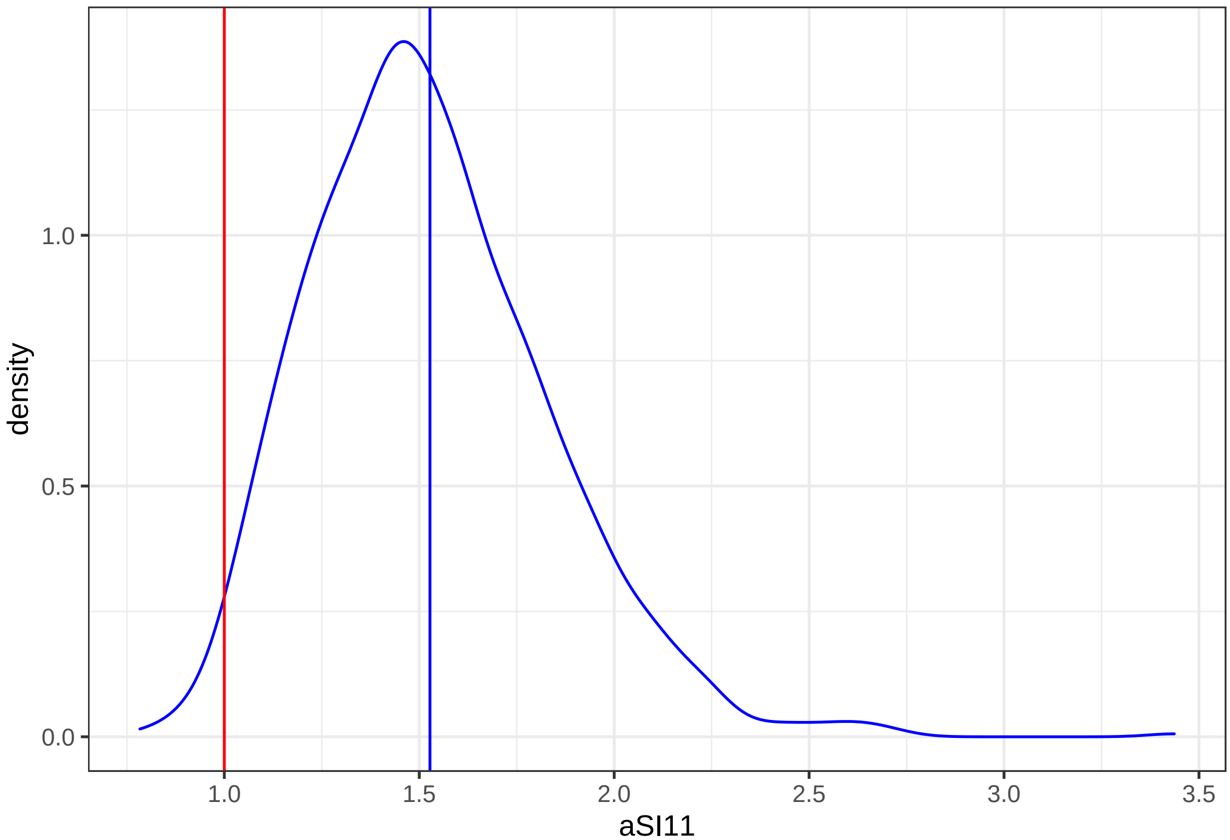
```
ggplot()+
  geom_density(aes(x=aTH1),col="blue")+
  geom_vline(xintercept=mean(aTH1),col="blue")+
  geom_vline(xintercept=aTruetheta[1],col="red")+
  theme_bw()
```



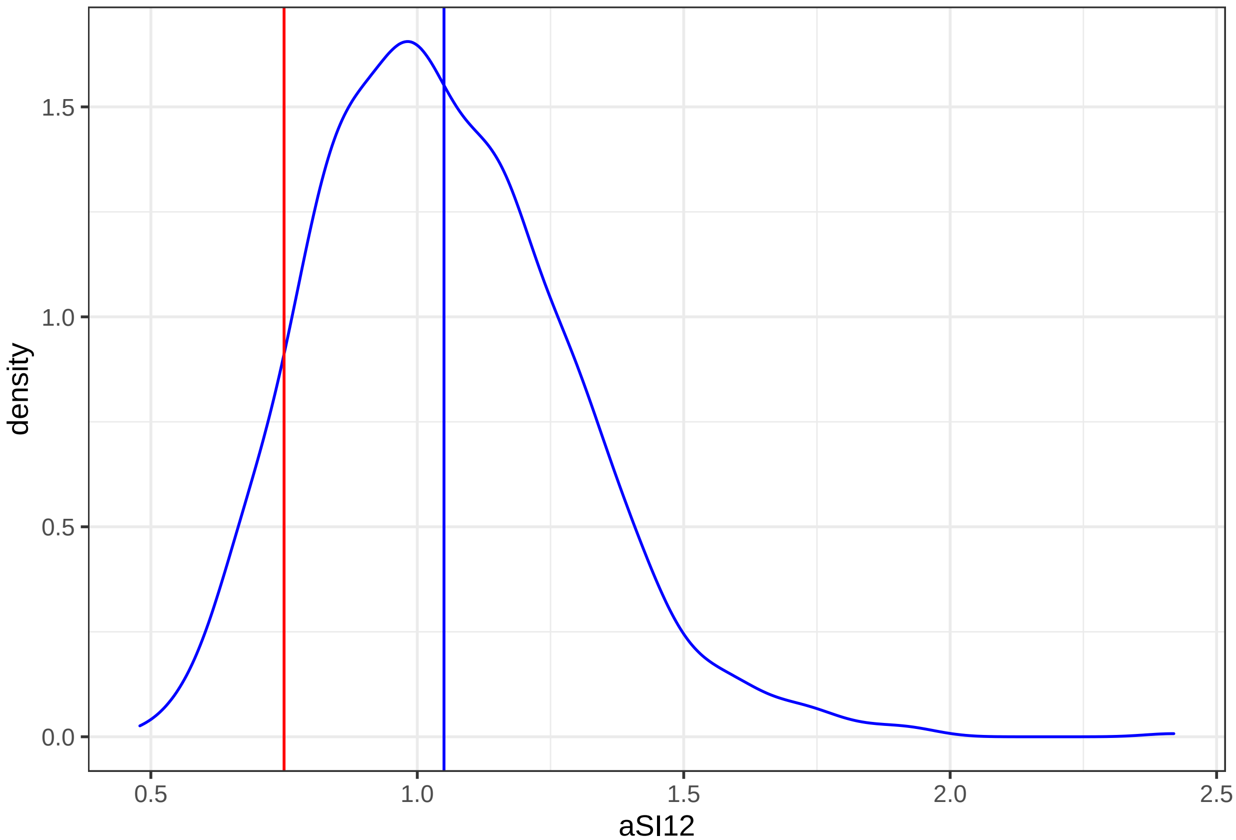

```
ggplot()+
  geom_density(aes(x=aTH2),col="blue")+
  geom_vline(xintercept=mean(aTH2),col="blue")+
  geom_vline(xintercept=aTruetheta[2],col="red")+
  theme_bw()
```



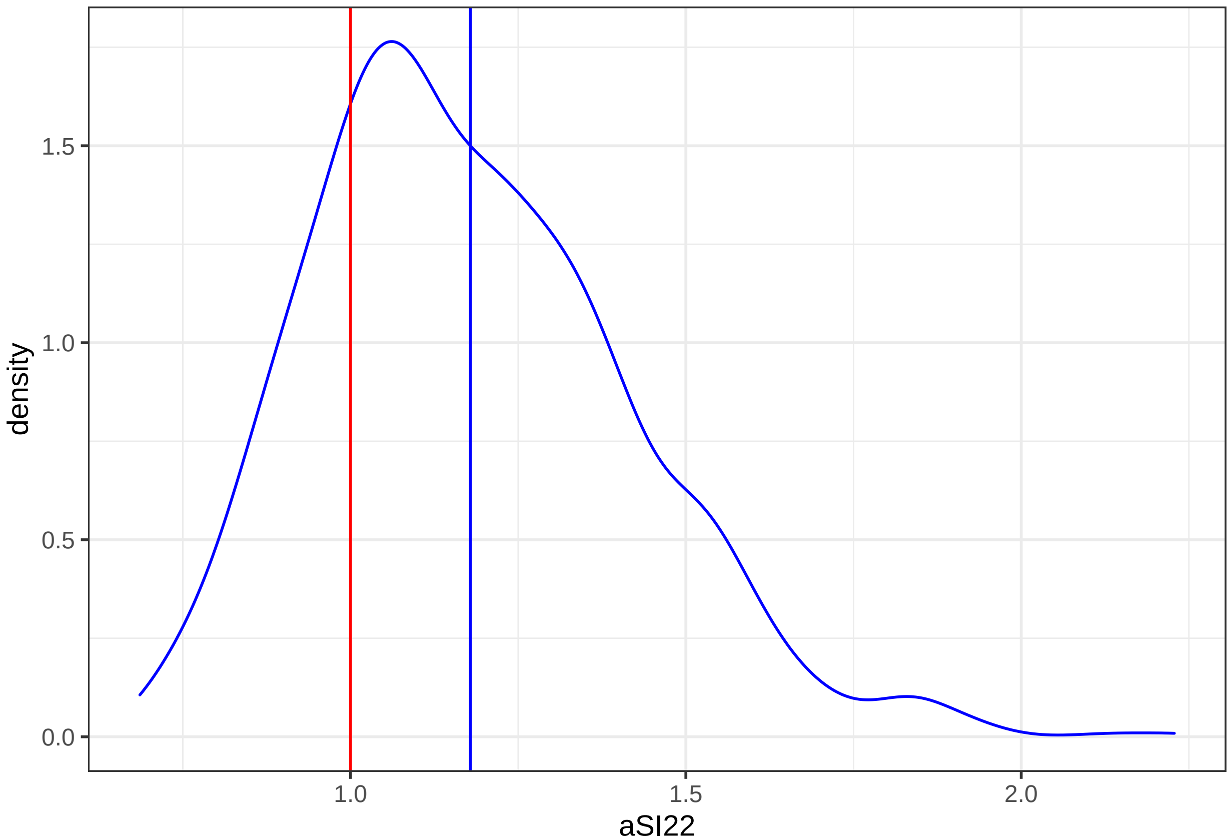

```
ggplot()+
  geom_density(aes(x=aSI11),col="blue")+
  geom_vline(xintercept=mean(aSI11),col="blue")+
  geom_vline(xintercept=aTrueSigma[1,1],col="red")+
  theme_bw()
```




```
ggplot()+
  geom_density(aes(x=aSI12),col="blue")+
  geom_vline(xintercept=mean(aSI12),col="blue")+
  geom_vline(xintercept=aTrueSigma[1,2],col="red")+
  theme_bw()
```




```
ggplot()+
  geom_density(aes(x=aSI22),col="blue")+
  geom_vline(xintercept=mean(aSI22),col="blue")+
  geom_vline(xintercept=aTrueSigma[2,2],col="red")+
  theme_bw()
```



Posterior for group b

```
n_b=dim(Yb)[1]
barYb=apply(Yb,2,mean)
Sb=matrix(c(0,0,0,0),ncol=2,nrow=2)
for(j in 1:n_b){
  Sb=Sb+(barYb-Yb[j,])%*%t(barYb-Yb[j,])
}

th_b=(la_0*th_0+n_b*barYb)*1/(la_0+n_b)
la_b=la_0+n_b
S_b=solve(solve(S_0)+Sb+la_0/(la_0+n_b)*(barYb-th_0)%*%t(barYb-th_0))
nu_b=nu_0+n_b
```

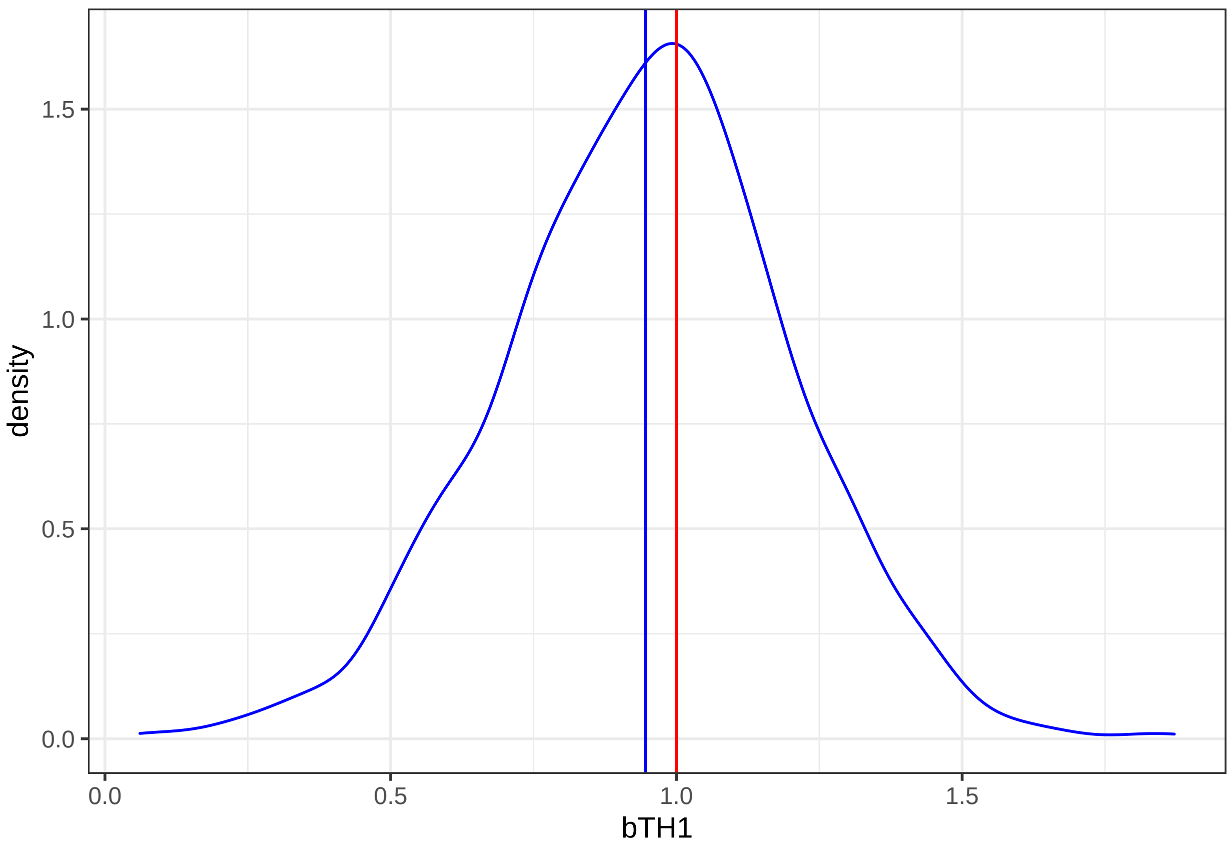
```
bSIGMA=bTHETA=list()
for(j in 1:1000){
  bSigma=solve(rWishart(n=1,nu_b,S_b)[,,1])
  btheta=mvrnorm(n=1,mu=th_b,Sigma=bSigma/la_b)

  bSIGMA[[j]]=bSigma
  bTHETA[[j]]=btheta
}
```

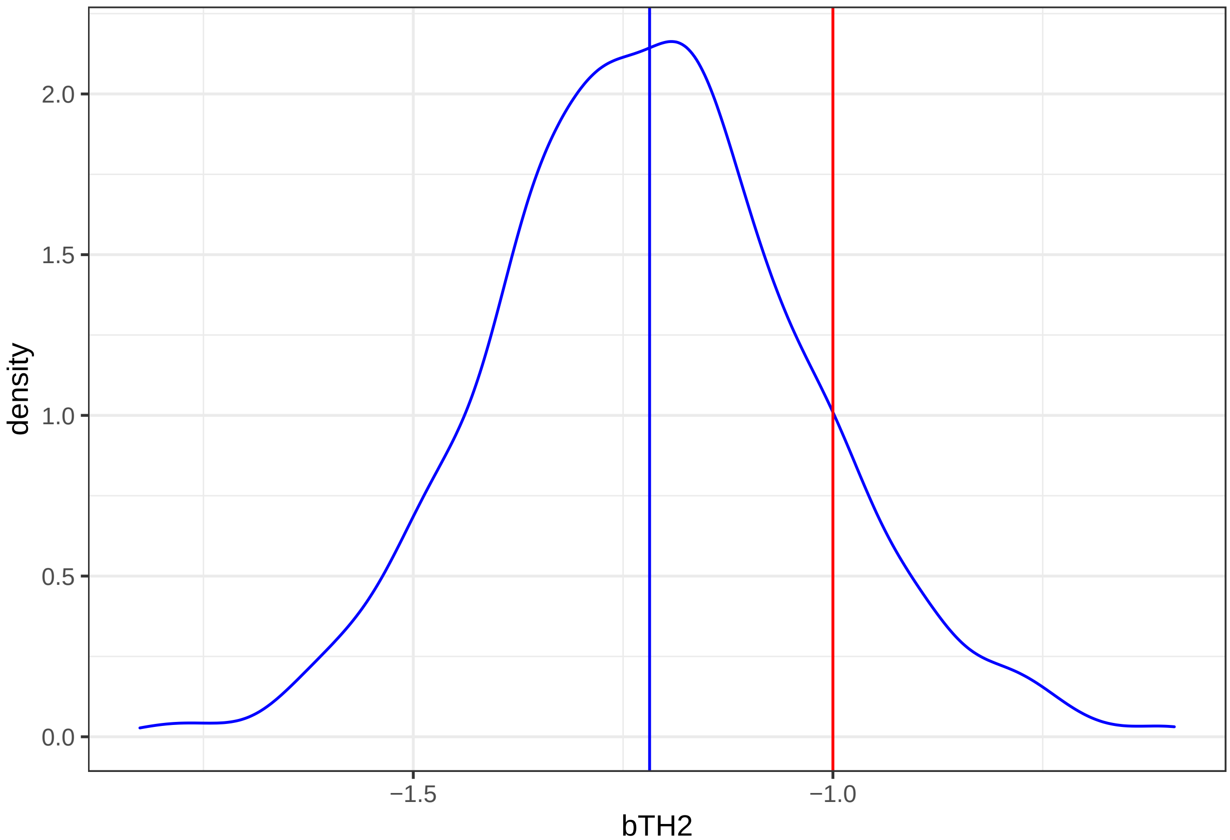
Plot the results

```
bTH1=bTH2=bSI11=bSI12=bSI22=c()
for(j in 1:1000){
  bTH1=c(bTH1,bTHETA[[j]][1])
  bTH2=c(bTH2,bTHETA[[j]][2])
  bSI11=c(bSI11,bSIGMA[[j]][1,1])
  bSI12=c(bSI12,bSIGMA[[j]][1,2])
  bSI22=c(bSI22,bSIGMA[[j]][2,2])
}
```

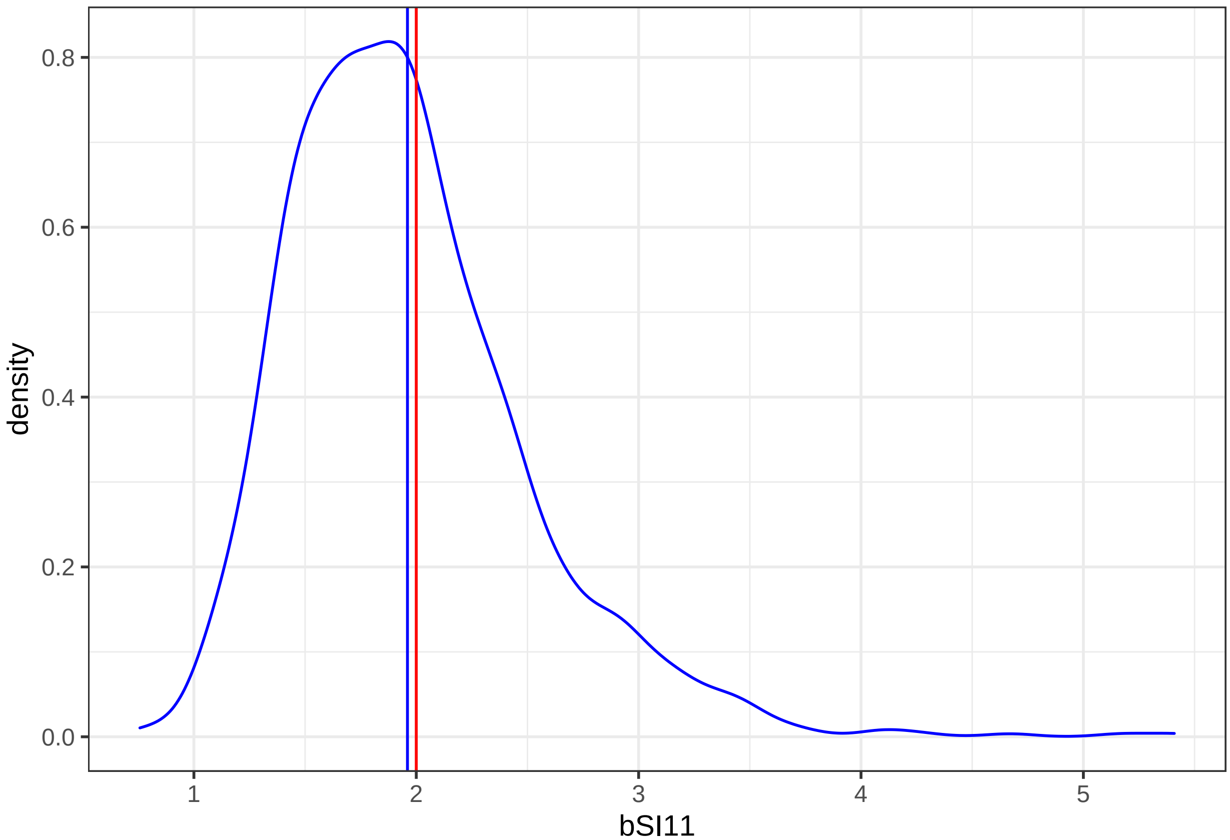
```
ggplot()+
  geom_density(aes(x=bTH1),col="blue")+
  geom_vline(xintercept=mean(bTH1),col="blue")+
  geom_vline(xintercept=bTruetheta[1],col="red")+
  theme_bw()
```



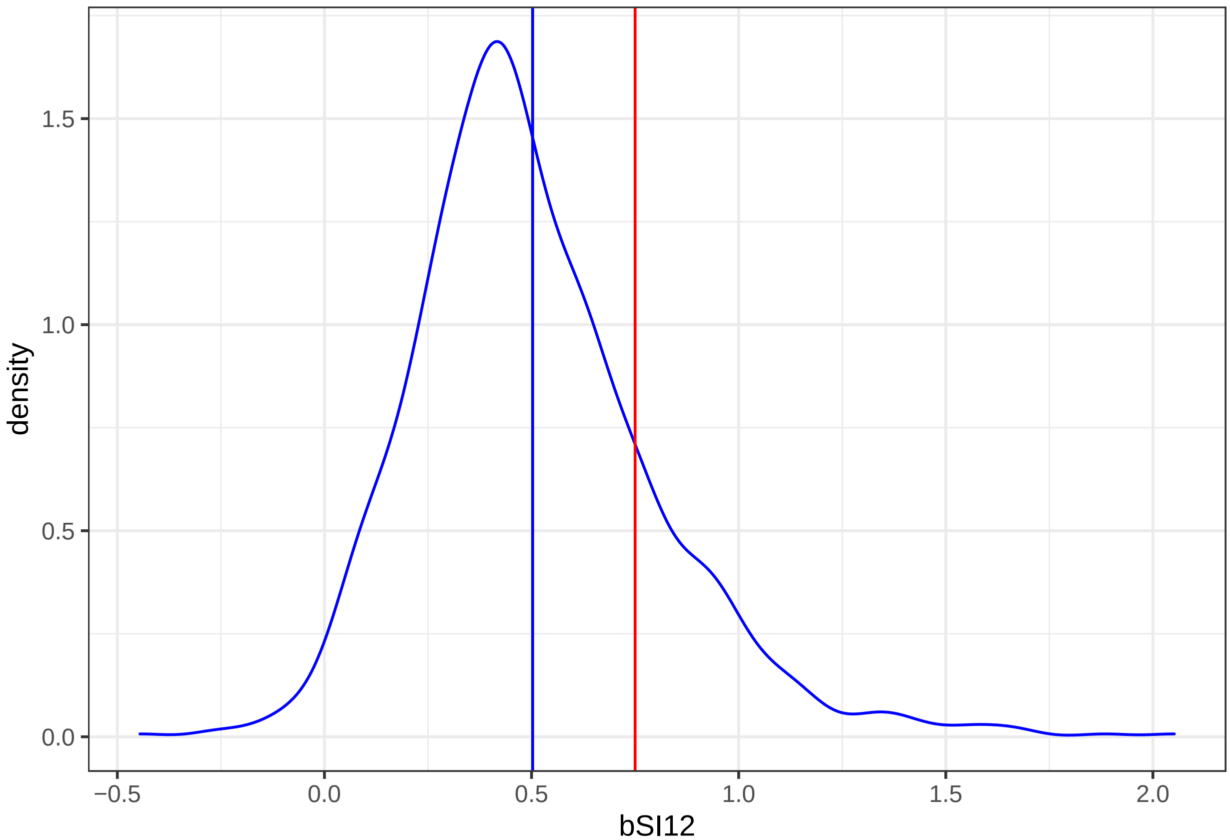

```
ggplot()+
  geom_density(aes(x=bTH2),col="blue")+
  geom_vline(xintercept=mean(bTH2),col="blue")+
  geom_vline(xintercept=bTruetheta[2],col="red")+
  theme_bw()
```



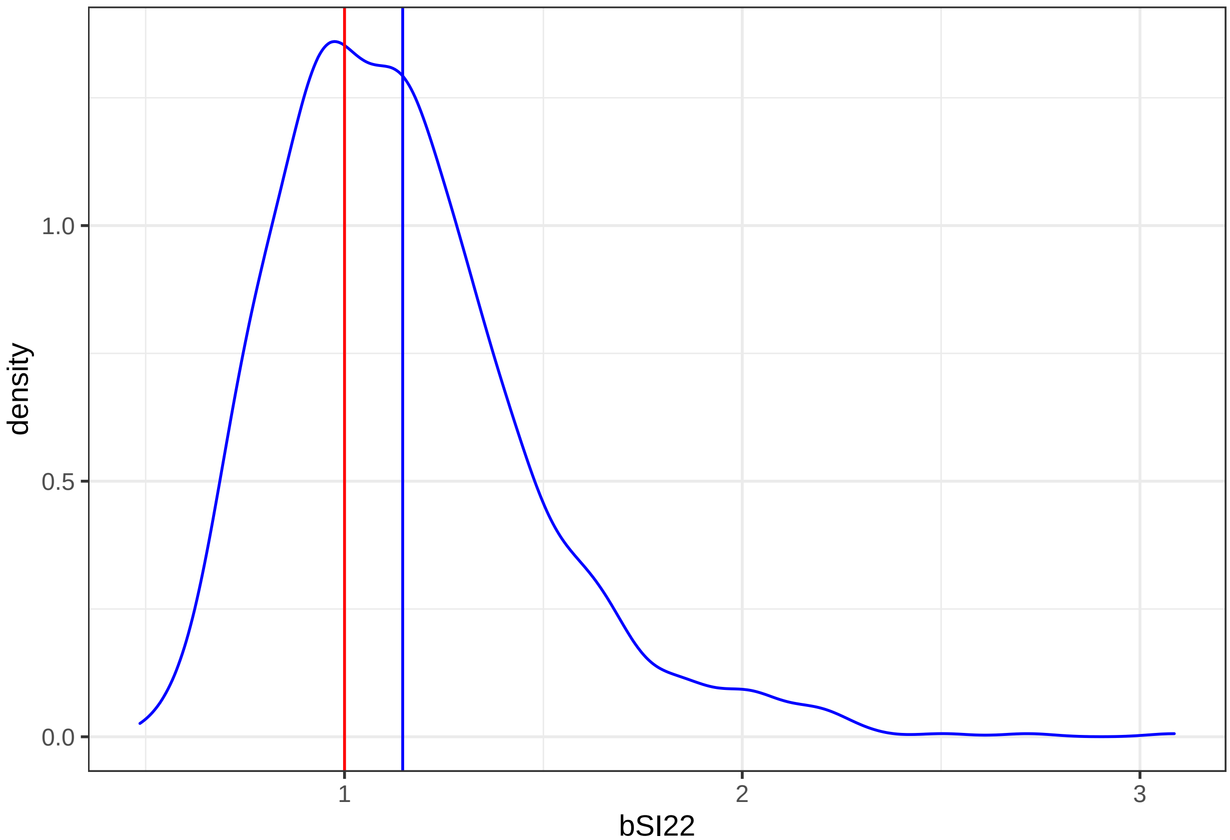

```
ggplot()+
  geom_density(aes(x=bSI11),col="blue")+
  geom_vline(xintercept=mean(bSI11),col="blue")+
  geom_vline(xintercept=bTrueSigma[1,1],col="red")+
  theme_bw()
```




```
ggplot()+
  geom_density(aes(x=bSI12),col="blue")+
  geom_vline(xintercept=mean(bSI12),col="blue")+
  geom_vline(xintercept=bTrueSigma[1,2],col="red")+
  theme_bw()
```




```
ggplot()+
  geom_density(aes(x=bSI22),col="blue")+
  geom_vline(xintercept=mean(bSI22),col="blue")+
  geom_vline(xintercept=bTrueSigma[2,2],col="red")+
  theme_bw()
```

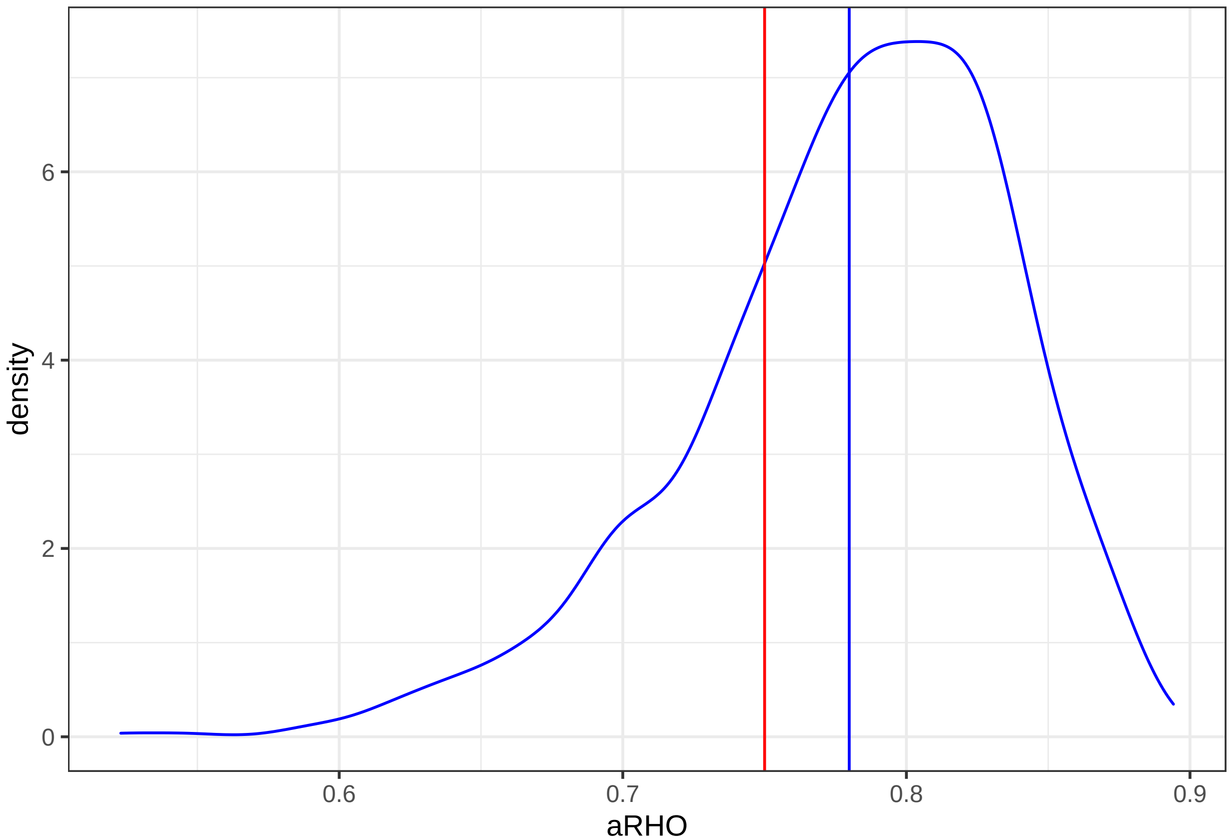


Comparing the two populations

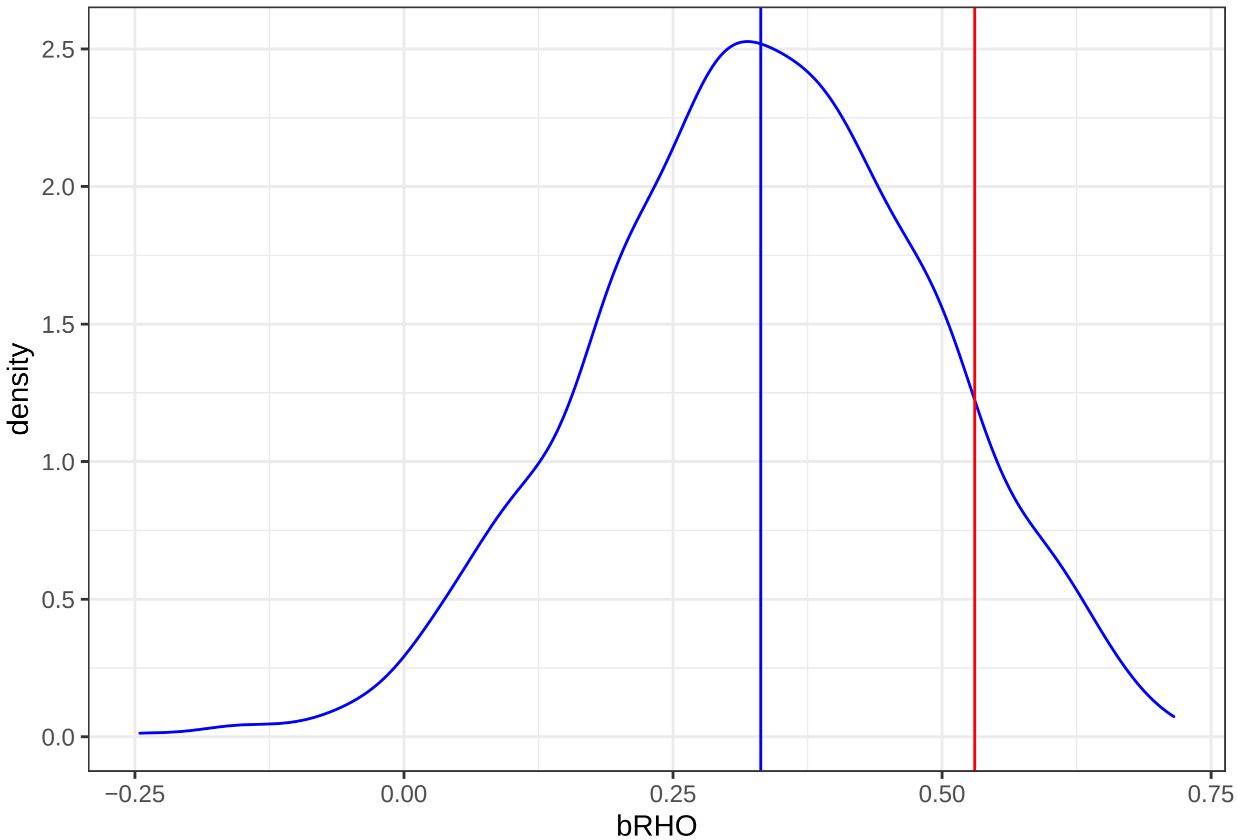
We now want to compare the correlation among the two populations

```
aRHO=bRHO=c()
for(j in 1:1000){
  aRHO=c(aRHO,aSI12[j]/sqrt(aSI11[j]*aSI22[j]))
  bRHO=c(bRHO,bSI12[j]/sqrt(bSI11[j]*bSI22[j]))
}
aTrueRho=aTrueSigma[1,2]/sqrt(aTrueSigma[1,1]*aTrueSigma[2,2])
bTrueRho=bTrueSigma[1,2]/sqrt(bTrueSigma[1,1]*bTrueSigma[2,2])
```

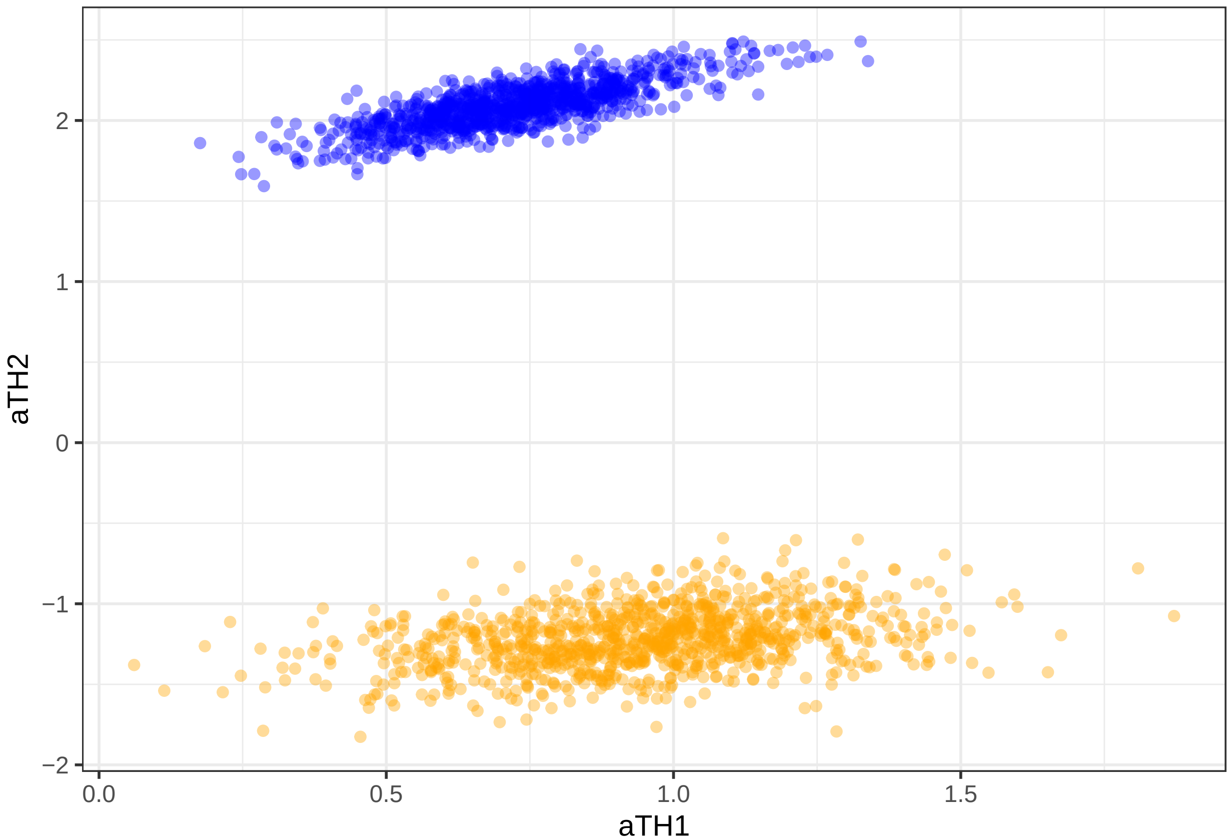
```
ggplot()+
  geom_density(aes(x=aRHO),col="blue")+
  geom_vline(xintercept=mean(aRHO),col="blue")+
  geom_vline(xintercept=aTrueRho,col="red")+
  theme_bw()
```



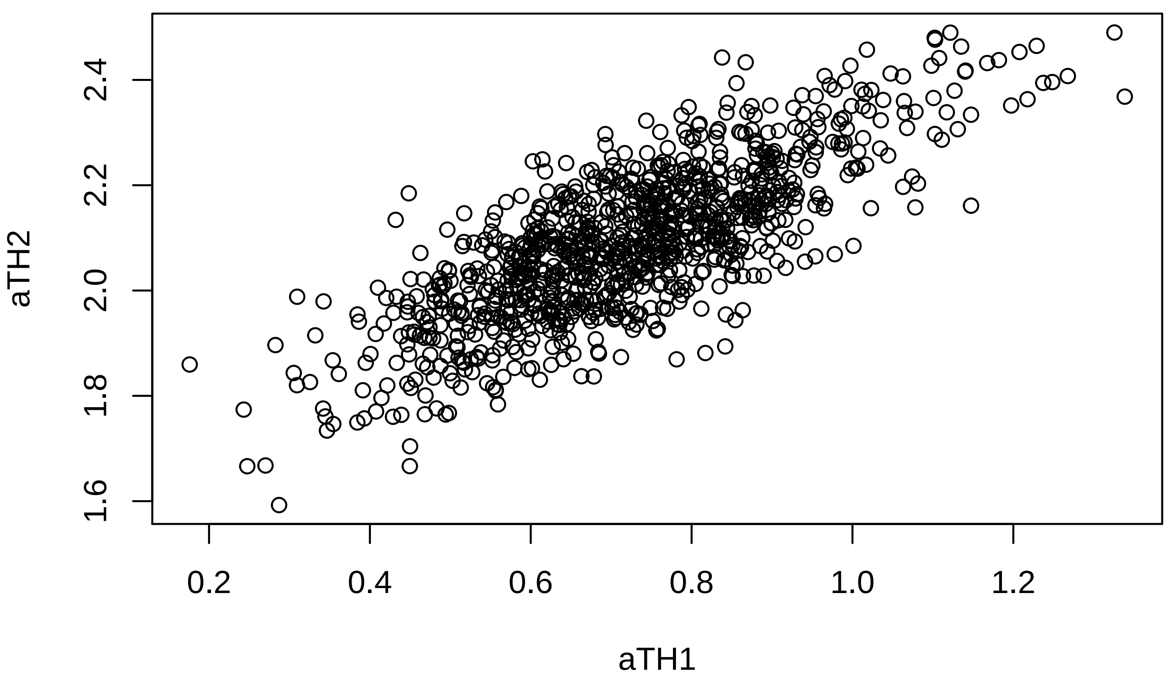

```
ggplot()+
  geom_density(aes(x=bRHO),col="blue")+
  geom_vline(xintercept=mean(bRHO),col="blue")+
  geom_vline(xintercept=bTrueRho,col="red")+
  theme_bw()
```



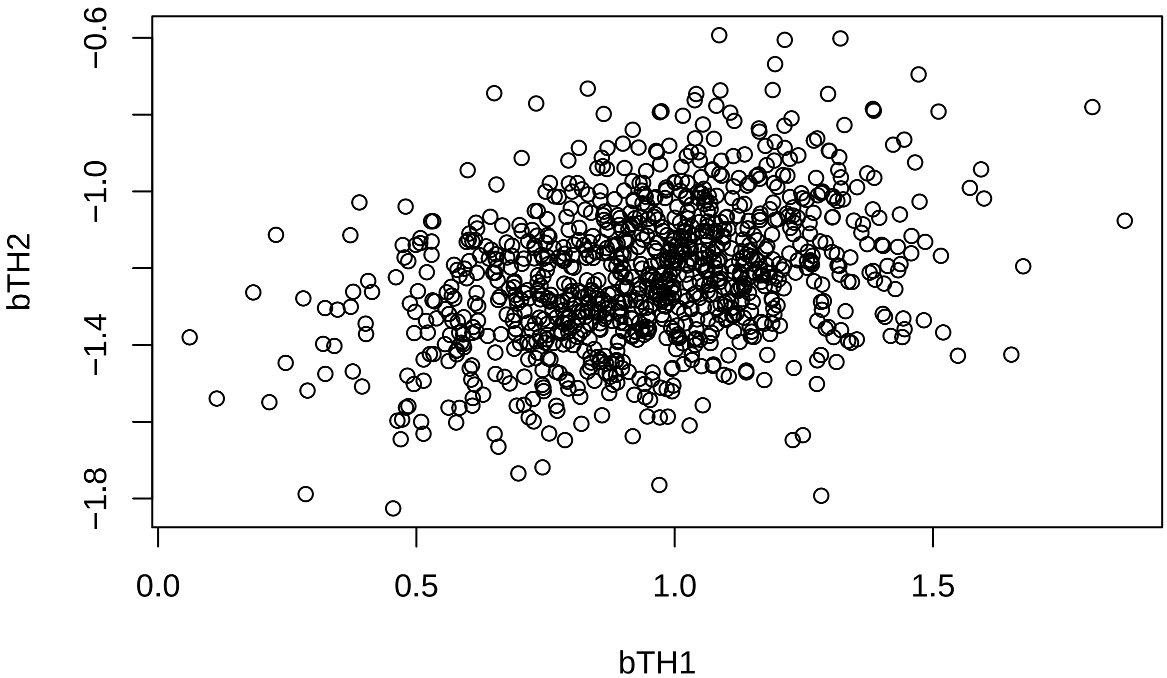

```
ggplot()+
  geom_point(data=NULL,aes(x=aTH1,y=aTH2),col="blue",alpha=0.4)+
  geom_point(data=NULL,aes(x=bTH1,y=bTH2),col="orange",alpha=0.4)+
  theme_bw()
```




```
plot(x=aTH1,y=aTH2)
```




```
plot(x=bTH1,y=bTH2)
```



We can easily compute the posterior probability that $\rho^a > \rho^b$ with

$$\mathbb{P}\left[\rho^a > \rho^b | Y^a - Y^b\right] \approx \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \mathbf{1}_{\rho_i^a > \rho_j^b}$$

$$\mathbb{P}\left[\rho^a > \rho^b | Y^a - Y^b\right] \approx \frac{1}{N} \sum_{j=1}^{N} \mathbf{1}_{\rho_j^a > \rho_j^b}$$

```
prob=0
for(i in 1:1000){
  for(j in 1:1000){
    if(aRHO[i]>bRHO[j]){prob=prob+1}
    else{prob=prob+0}
  }
}
prob=prob/1000^2
prob
```

```
## [1] 0.998968
```

```
prob=0
for(j in 1:1000){
  if(aRHO[j]>bRHO[j]){prob=prob+1}
  else{prob=prob+0}
}
prob=prob/1000
prob
```

```
## [1] 0.999
```