AdvStDaAn, Worksheet, Week 10

Michael Lappert

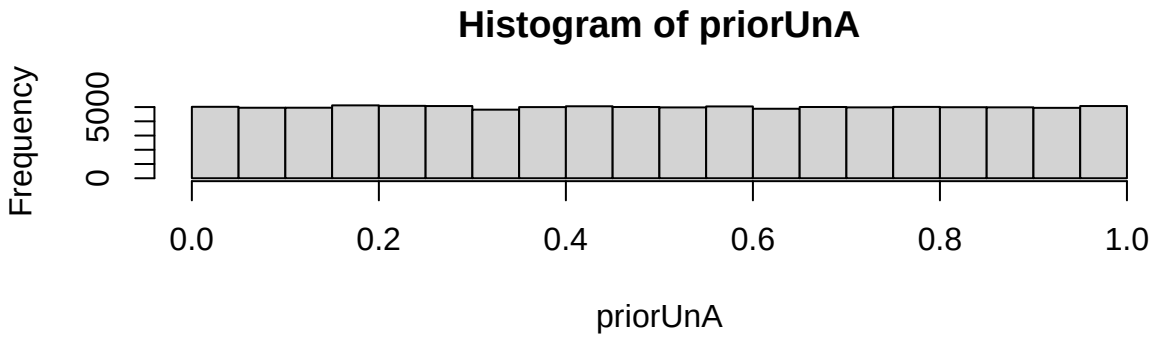
05 Mai, 2022

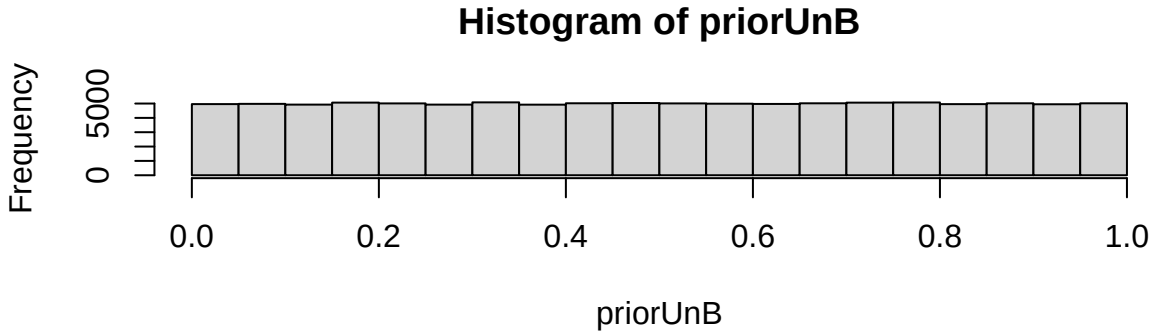
Contents

Task 1

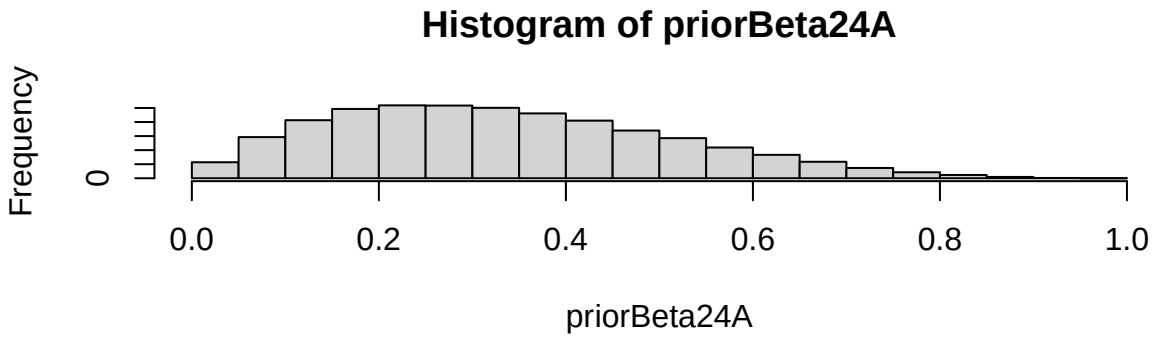
Study the influence of prior assumptions on the results of a Bayesian analysis. Remember Swedish Fish Inc.'s two advertising alternatives: - method A: 6 out of 16 signed up and - method B: 10 out of 16 signed up. Use an uninformative prior, a Beta(2, 4) prior and the more informative Beta(3, 25) prior for the signup rates θ_A and θ_B and compare the resulting marginal posterior distributions. ***

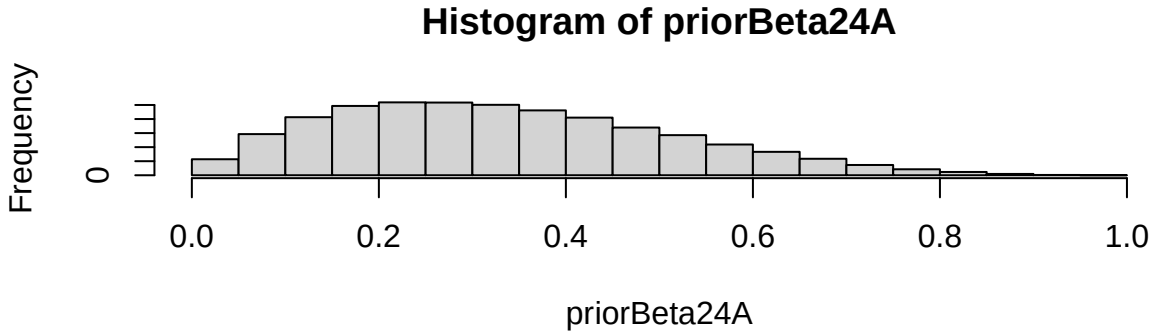
```
# Simulate n random draws from the different priors
n = 100000

par(mfrow=c(2,1))
# uninformative prior:
priorUnA <- runif(n)
hist(priorUnA) # Eyball the prior
priorUnB <- runif(n)
hist(priorUnB) # Eyball the prior
```



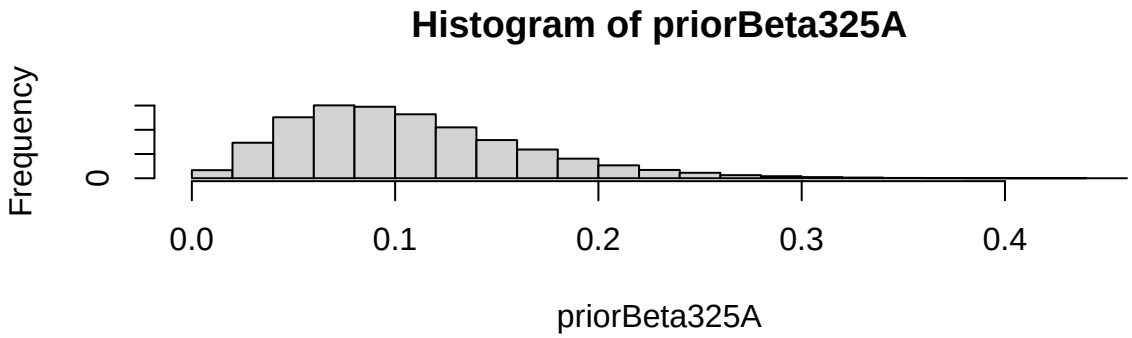


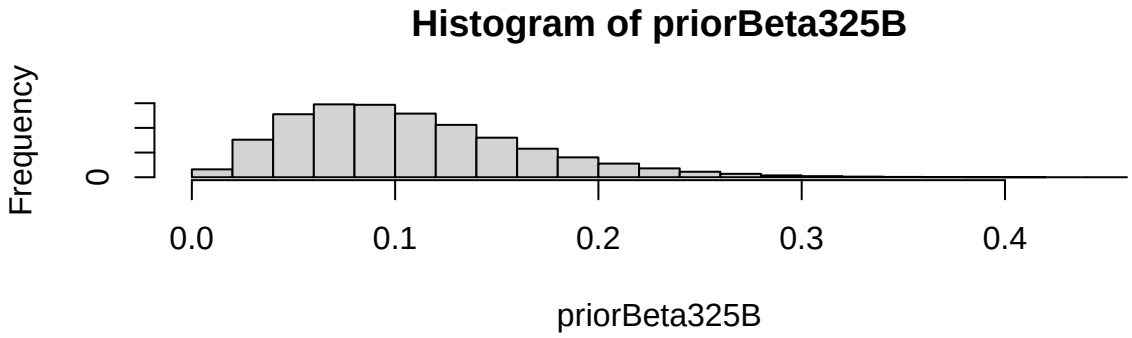

```
# beta(2, 4) prior
priorBeta24A <- rbeta(n, 2, 4)
hist(priorBeta24A)
priorBeta24B <- rbeta(n, 2, 4)
hist(priorBeta24A)
```






```
# beta(3, 25) prior
priorBeta325A <- rbeta(n, 3, 25)
hist(priorBeta325A)
priorBeta325B <- rbeta(n, 3, 25)
hist(priorBeta325B)
```

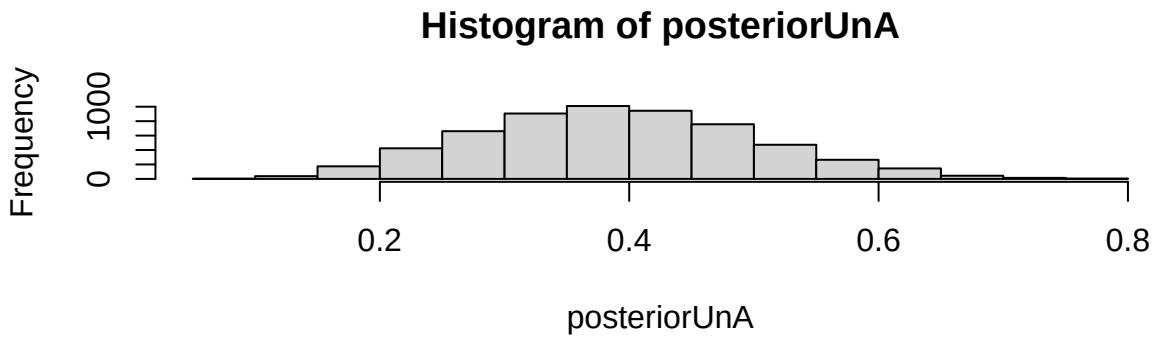


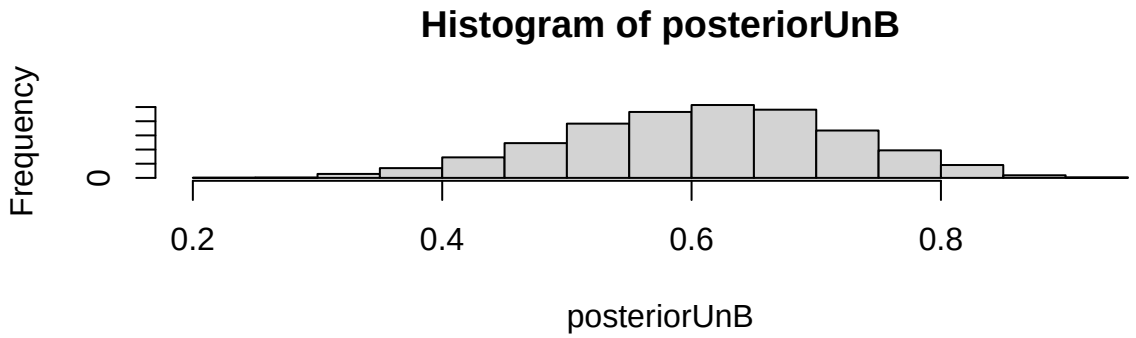



```r
# Define the generative model of uninformative prior:
generativemodelUnA <- function(theta) {
  rbinom(1, 16, theta)
}
generativemodelUnB <- function(theta) {
  rbinom(1, 16, theta)
}


# Simulate and store data from uninformative prior:
simdataUnA <- rep(NA, n)
for(i in 1:n) {
  simdataUnA[i] <- generativemodelUnA(priorUnA[i])
}
simdataUnB <- rep(NA, n)
for(i in 1:n) {
  simdataUnB[i] <- generativemodelUnA(priorUnB[i])
}

# Filter out all draws that do not match the data of the uniform prior:
posteriorUnA <- priorUnA[simdataUnA == 6]
hist(posteriorUnA)
length(posteriorUnA)
```

```
## [1] 5862
```

```
posteriorUnB <- priorUnB[simdataUnB == 10]
hist(posteriorUnB)
```





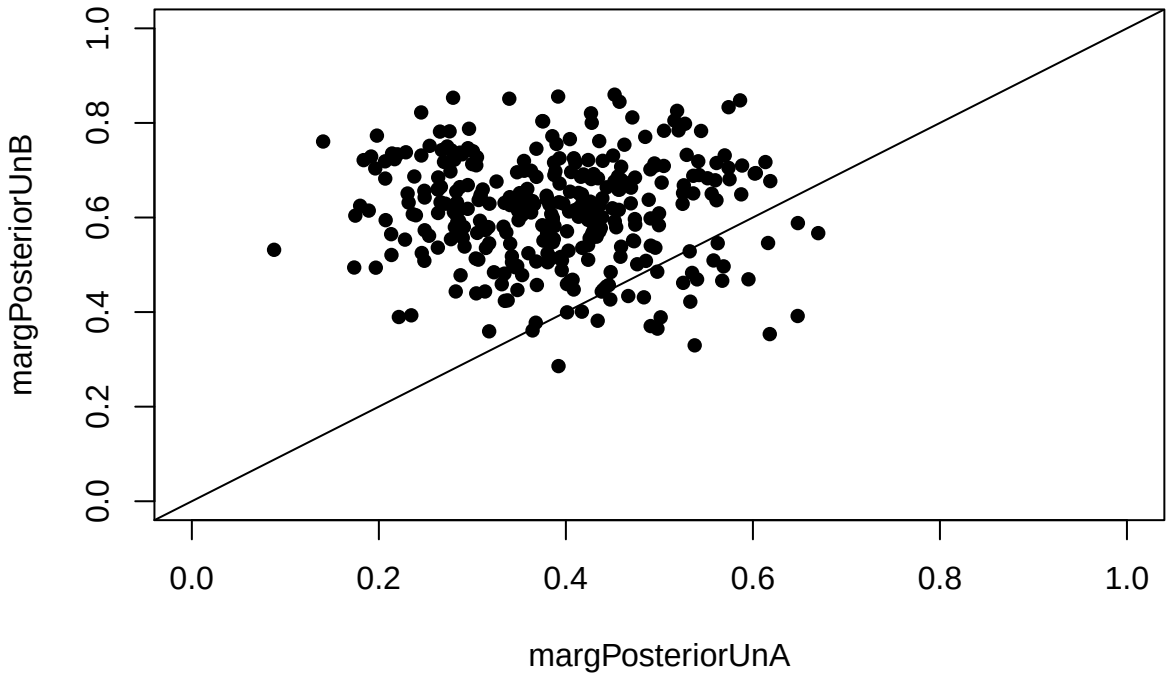

```
length(posteriorUnB)
```

```
## [1] 5956
```

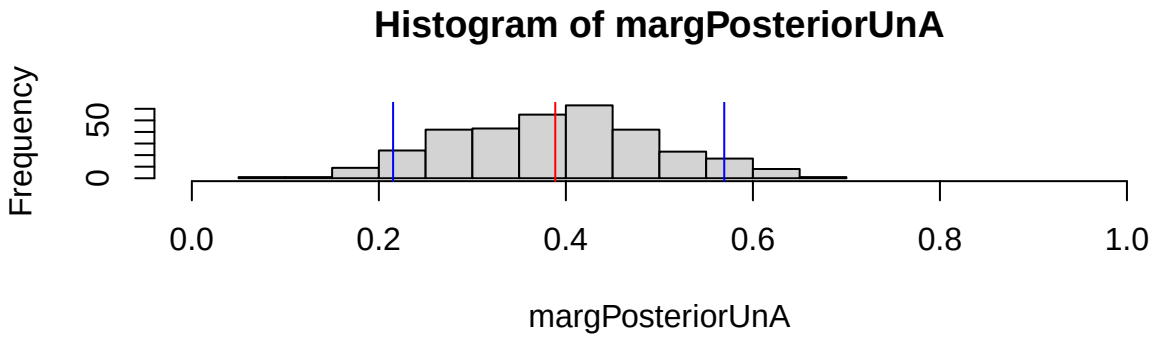
```
# Condition on observed data
ind = ( (simdataUnA==6) & (simdataUnB==10) )
ind[1:20]
```

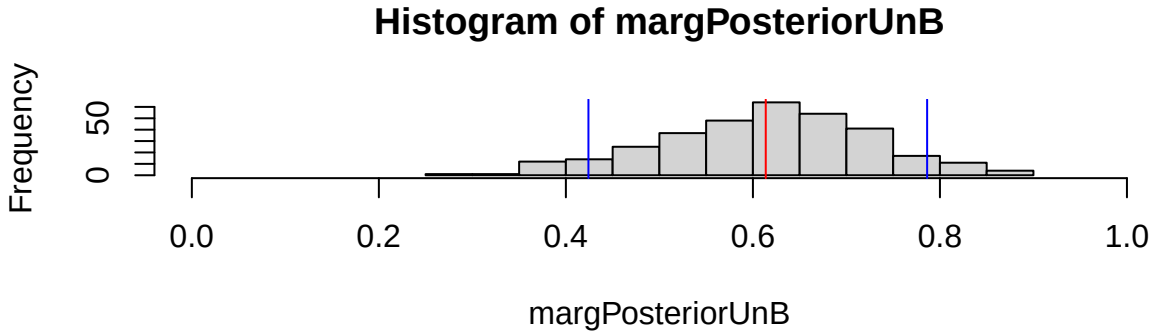
```
##  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

```
margPosteriorUnA <- priorUnA[ind]
margPosteriorUnB <- priorUnB[ind]

# evaluate results
par(mfrow=c(1,1))
plot(margPosteriorUnA, margPosteriorUnB, cex=1, pch=16, xlim=c(0,1), ylim=c(0,1))
abline(0,1)
```




```
par(mfcol=c(2,1))
hist(margPosteriorUnA,xlim=0:1)
abline(v=mean(margPosteriorUnA),col="red")
abline(v=quantile(margPosteriorUnA,c(0.05,0.95)),col="blue")
hist(margPosteriorUnB,xlim=0:1)
abline(v=mean(margPosteriorUnB),col="red")
abline(v=quantile(margPosteriorUnB,c(0.05,0.95)),col="blue")
```






```
par(mfcol=c(1,1))
```

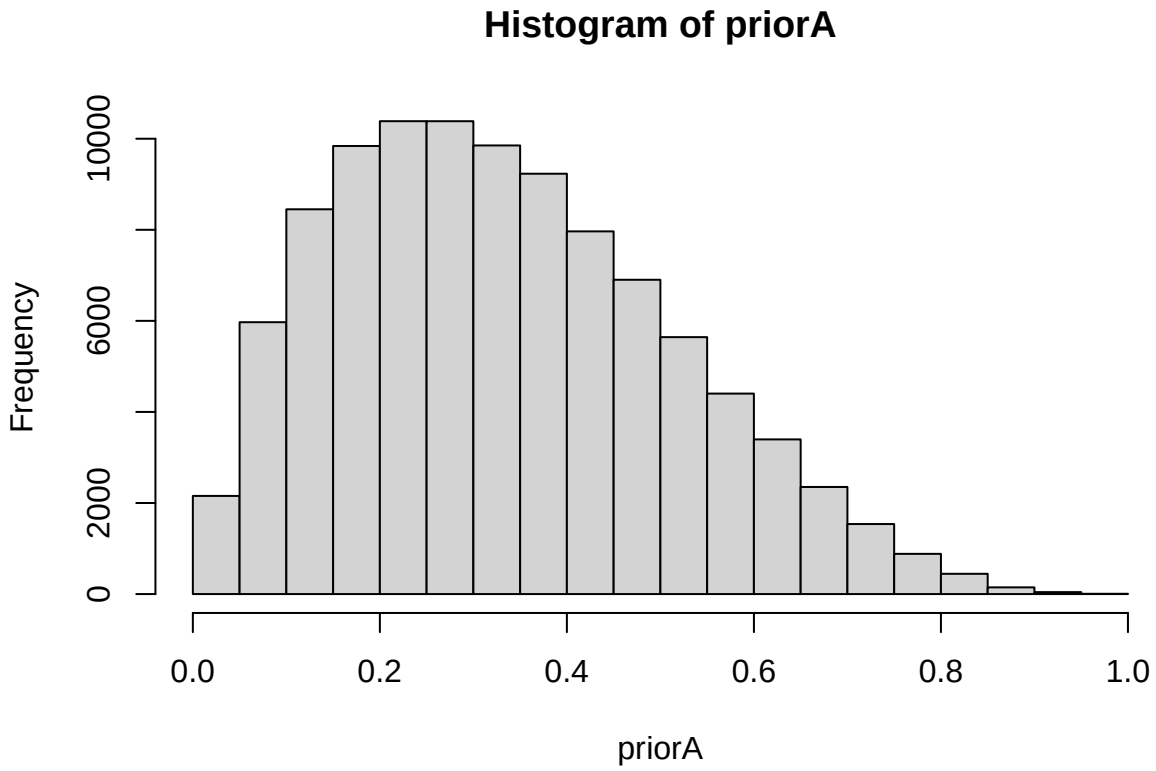
This would be the approach per prior. But in the solution was a way easier approach to do so. This one is used beneath.

Question 1, Task 1

> Why are there these warning in the model data simulation process in this kind of approach? I adjusted n for the binomial sampling in rbinom to the singnups. Which is apparently wrong.

```
# number of samples
nSamples = 100000

# Data
asked    = 16 # number of asked/invited people
SingnupA =  6 # number of signups with method A
SingnupB = 10 # number of signups with method B

# Simulate prior
# Case 1
```

```
# priorA = runif(nSamples,0,1)
# priorB = runif(nSamples,0,1)
# Case 2
priorA = rbeta(nSamples,2,4)
priorB = rbeta(nSamples,2,4)
# Case 3
# priorA = rbeta(nSamples,3,25)
# priorB = rbeta(nSamples,3,25)
hist(priorA)
```

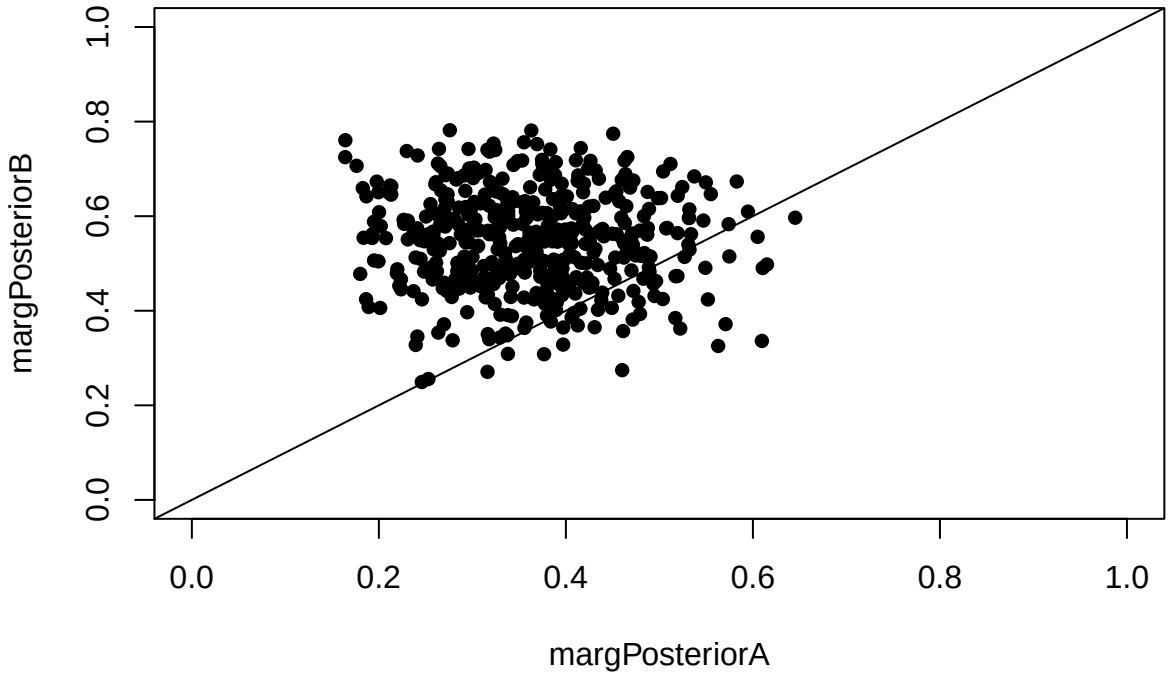



```
# Simulate generative model (likelihood)
simSingnupA = rbinom(nSamples,asked,priorA)
simSingnupB = rbinom(nSamples,asked,priorB)

# Condition on observed data
ind = ( (simSingnupA==SingnupA) & (simSingnupB==SingnupB) )
ind[1:20]
```

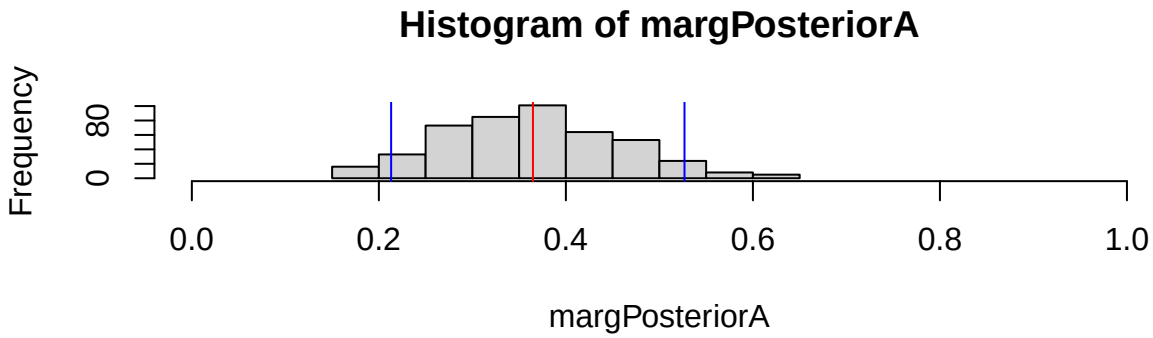
```
##  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

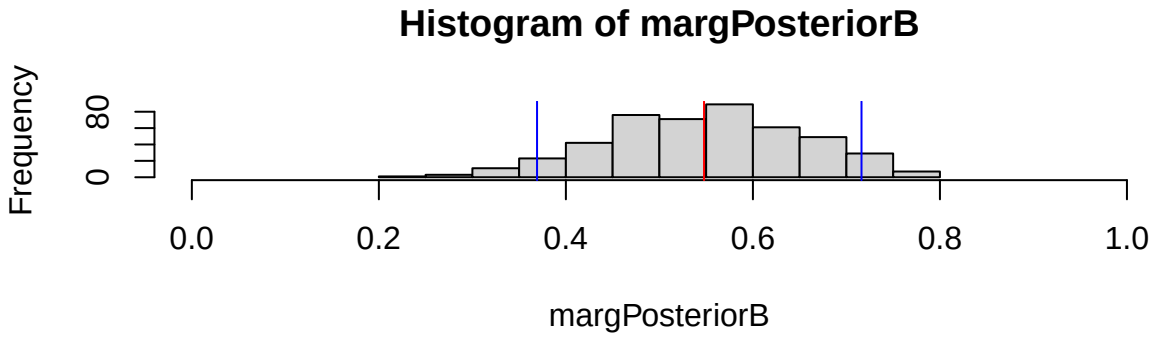
```
margPosteriorA = priorA[ind]
margPosteriorB = priorB[ind]
```

```r
# evaluate results
par(mfrow=c(1,1))
plot(margPosteriorA,margPosteriorB,cex=1,pch=16,xlim=c(0,1),ylim=c(0,1))
abline(0,1)
```




```r
par(mfcol=c(2,1))
hist(margPosteriorA,xlim=0:1)
abline(v=mean(margPosteriorA),col="red")
abline(v=quantile(margPosteriorA,c(0.05,0.95)),col="blue")
hist(margPosteriorB,xlim=0:1)
abline(v=mean(margPosteriorB),col="red")
abline(v=quantile(margPosteriorB,c(0.05,0.95)),col="blue")
```






```
par(mfcol=c(1,1))
```

Question 2, Task 1

If i use a prior of beta(3, 25) i do not get any marginal posteriors. Why is that?
