lab2 TDDE07

alebo256 & danbi675

2021-04-28

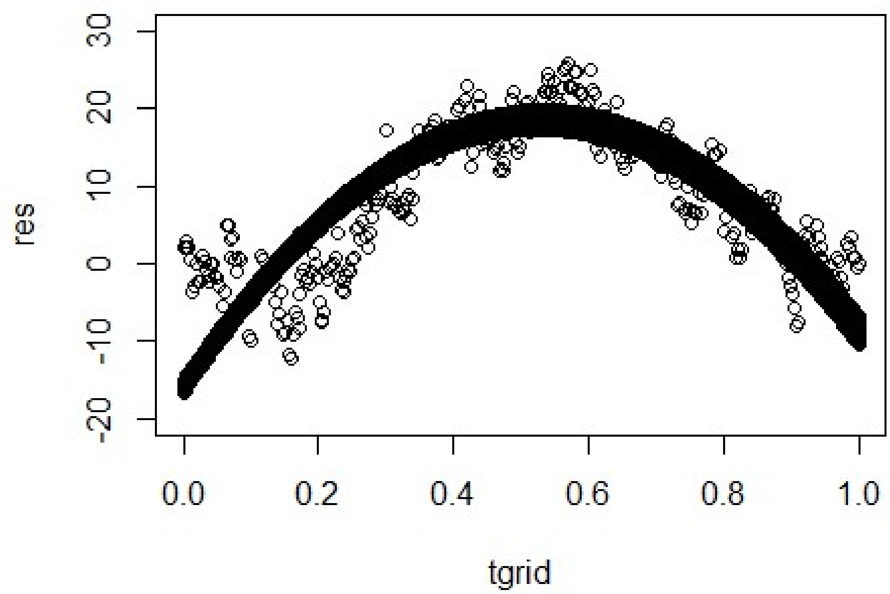
Task1

a)

We adjusted the prior to fit our expectations before making posterior draws.

```
templinkoping=read.table("TempLinkoping.txt", header = TRUE)
library(mvtnorm)

library(LaplacesDemon)

set.seed(12345)
tempfunc <- function(b , timevar, errorval) {
  predtemp <- b[1] + b[2]*timevar + b[3]*timevar**2 + errorval
  return(predtemp)
}
dinvchisq <- function(x, n, t) {
  a <- n/2
  b <- n*t/2
  toreturn <- (b^a)/gamma(a) * x^(-a-1) * exp(-b/x)
  if (is.nan(toreturn)) {
    return(0)
  }
  return(toreturn)
}
my0 <- c(-16, 130, -123)
omega0 <- 0.6*diag(3)
v0 <- 4
sigma0sqr <- 0.1

n <- 100
sigmasqr <- rinvchisq(1, v0, sigma0sqr)
draws <- rmvnorm(n, my0, sigmasqr*solve(omega0))
errortest <- rnorm(1,0,sigmasqr)


tgrid <- seq(0, 1, length.out = 365)
res <- rep(0, length(tgrid))
i <- 1
for (time in tgrid){
  res[i] <- tempfunc(draws[1,],time,errortest)
  i <- i+1
```

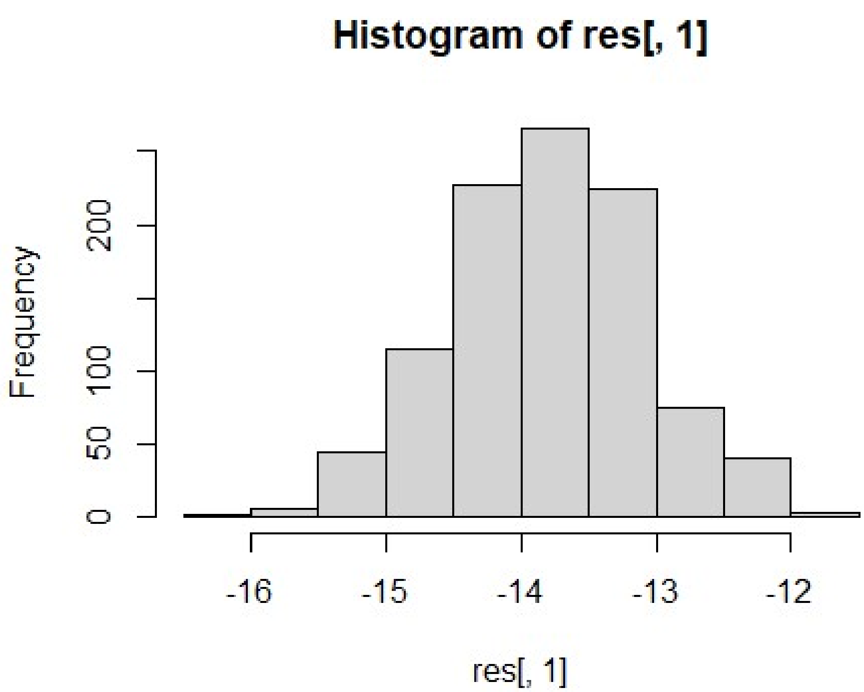
```
}
plot(tgrid,res, ylim = c(-20,30))

drawnr <- seq(2, n, 1)
for (drawn in drawnr) {
  sigmasqr <- rinvchisq(1, v0, sigma0sqr)
  tgrid <- seq(0, 1, length.out = 365)
  res <- rep(0, length(tgrid))
  i <- 1
  for (time in tgrid){
    res[i] <- tempfunc(draws[drawn,],time,errortest)
    i <- i+1
  }
  points(tgrid,res)


}
points(tgrid,templinkoping[,2])
```



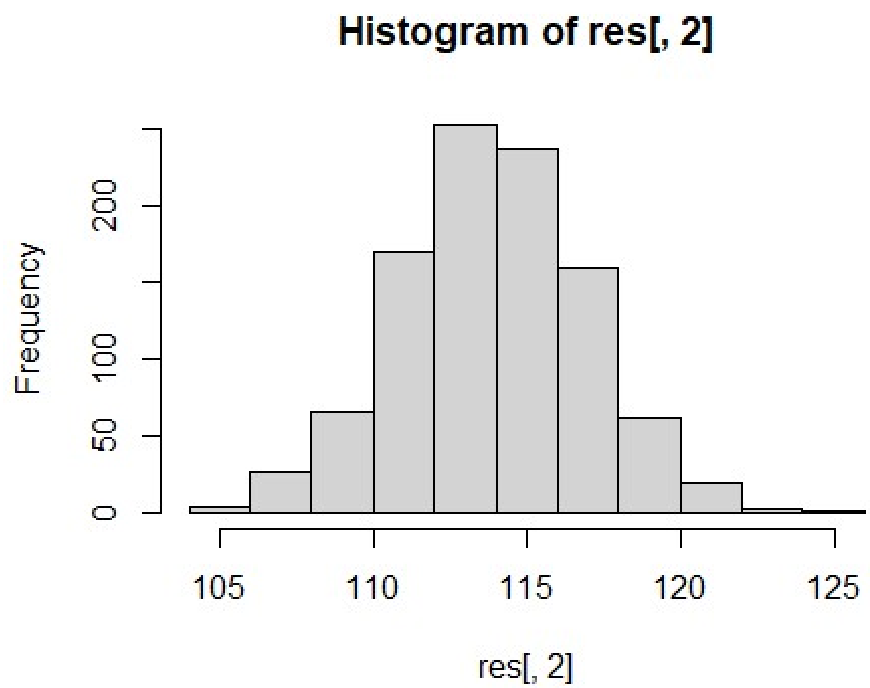
b)

```
postdraw <- function(data, sigsqr) {
  ones <- rep(1,365)
  X <- cbind(ones, data[,1], data[,1]**2)
  y <- templinkoping[,2]
  betahat <- solve(t(X)%*%X)%*%t(X)%*%data$temp
  myn <- solve(t(X)%*%X+omega0)%*%(t(X)%*%X%*%betahat+omega0%*%my0)
```

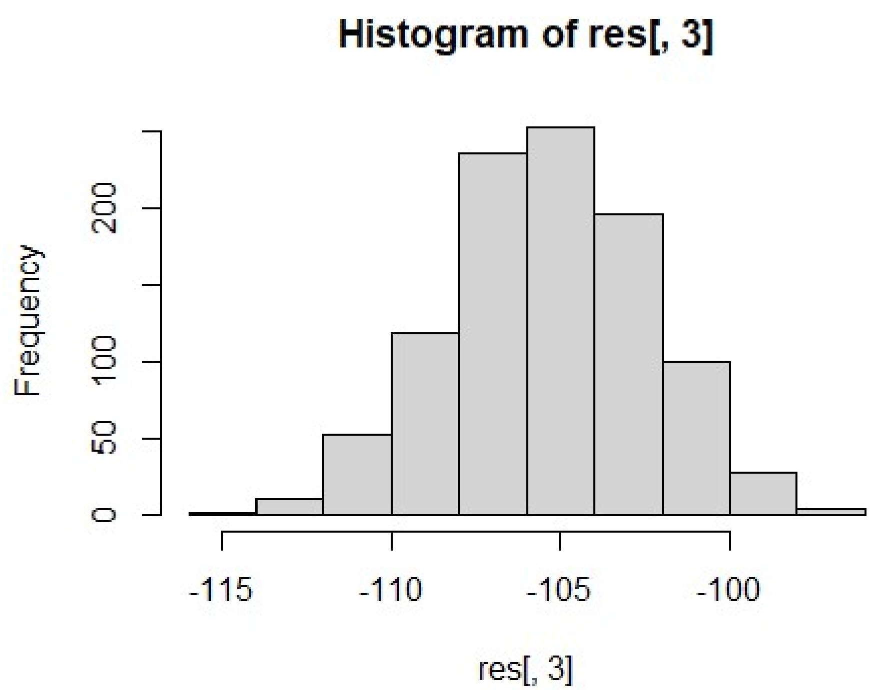
```r
  omegan <- t(X)%*%X+omega0
  vn <- v0+365
  vnsigmansqr <- v0*sigma0sqr + t(y)%*%y + t(my0)%*%omega0%*%my0 -
t(myn)%*%omegan%*%myn
  sigmasqrdraw <- rinvchisq(1, vn, vnsigmansqr/vn)
  betadraw <- rmvnorm(1,myn,as.numeric(sigmasqrdraw)*solve(omegan))
  return(cbind(betadraw,sigmasqrdraw))
}

n <- 1000
nloop <- seq(1,1000,1)
res <- matrix(, nrow=n, ncol=4)
for (i in nloop) {
  res[i,] <- postdraw(templinkoping,sigmasqr)
}
hist(res[,1])
```



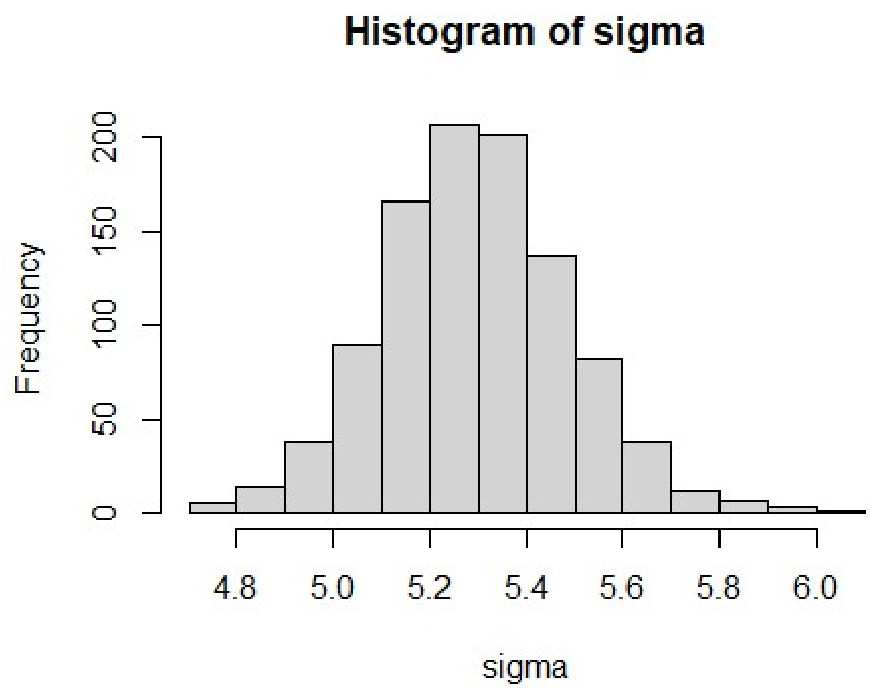

```r
hist(res[,2])
```


Histogram of res[, 2]
Frequency
0
50
100
200
105
110
115
120
125
res[, 2]

```
hist(res[,3])
```

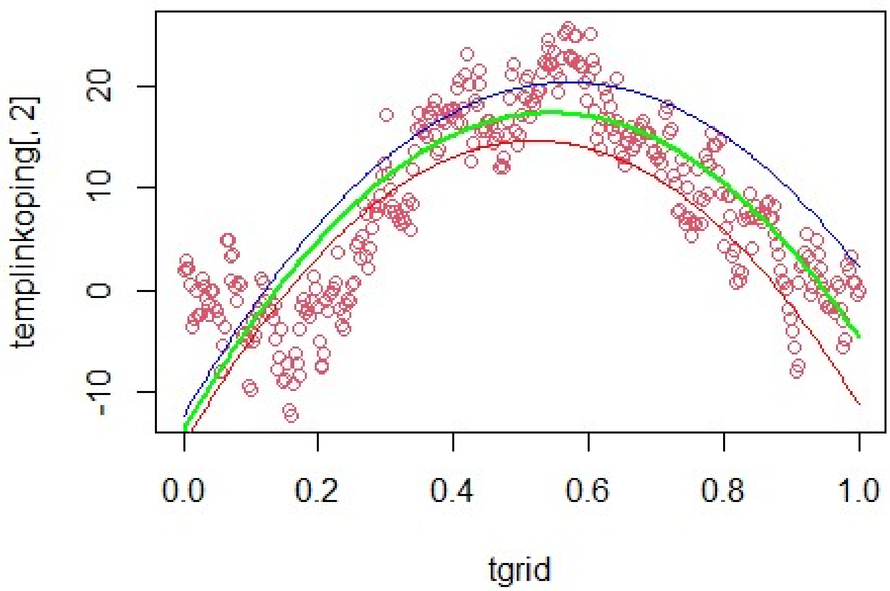

Histogram of res[, 3]
Frequency
0
50
100
200
-115
-110
-105
-100
res[, 3]

```
sigma <- sqrt(res[,4])
res <- res[,-4]
hist(sigma)
```




```
tgrid <- seq(0, 1, length.out = 365)
toplot <- rep(0, length(tgrid))
quantiles <- matrix(,nrow=length(tgrid),ncol=2)
i <- 1
for (time in tgrid){
  tempdraws <- rep(0, length(n))
  for (ii in 1:n) {
    tempdraws[ii] <- res[ii,1] + res[11,2]*time + res[ii,3]*time**2 +
errortest
  }
  quantiles[i,] <- quantile(tempdraws, probs=c(0.025,0.975))
  toplot[i] <- median(tempdraws)
  i <- i+1
}



top <- loess(toplot~tgrid)
plot(tgrid,templinkoping[,2],col=2)
lines(tgrid, predict(top), col='green', lwd=2)

top <- loess(quantiles[,1]~tgrid)
lines(tgrid, predict(top),col= "red")
```

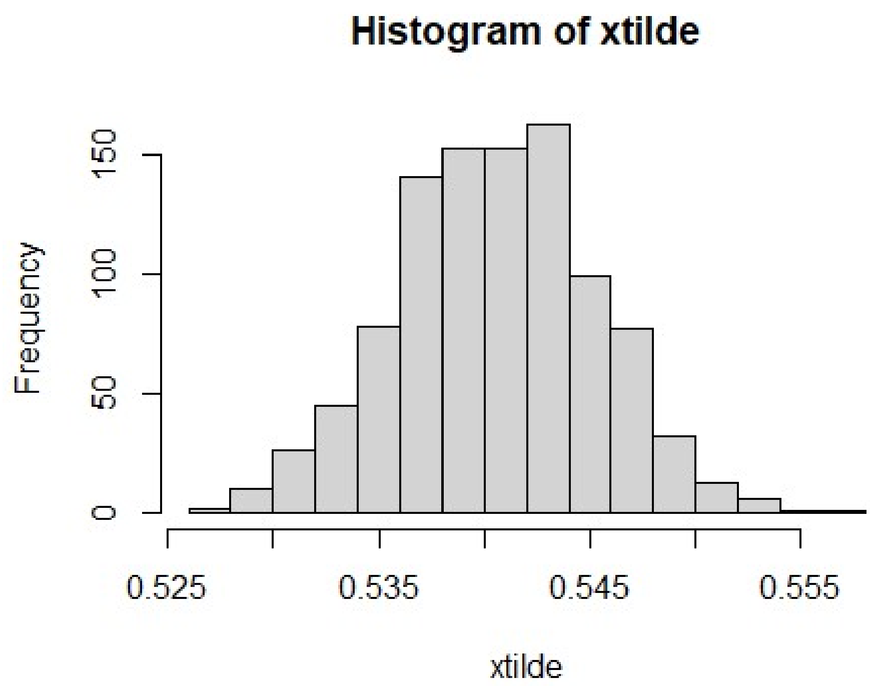
```
top <- loess(quantiles[,2]~tgrid)
lines(tgrid, predict(top),col= "blue")
```



Most of the data dows not fit inside the posterior probabilty interval. We don't think it should since the model is not only fitted according to our data, but also depend on out prior.

c)

```
highesttempday <- function(beta) {
  toret <- optim(1,tempfunc, b = beta, errorval = 0, method = 'Brent', lower
= 0, upper = 1,control=list(fnscale=-1))
  return(toret$par)
}
xtilde <- apply(res, 1, highesttempday)
hist(xtilde)
```



d)

We assign my0 to 0 for higher order terms since we do not expect them to have an impact. We assign omega0 to a very large value to not allow flexibility in the model so to combat overfitting by not allowing the model to vary that much.

Task 2

a)

```
library("mvtnorm")
set.seed(12345)
data <- read.table("WomenWork.dat", header = TRUE)

X <- as.matrix(data[,2:9])
Y <- as.vector(data[,1])

muPrior <- as.matrix(rep(0,8))
SigmaPrior <- 100*diag(8)
B <- c(0,0,0,0,0,0,0,0)

LogReg <- function(betas, y, x, mu, Sigma){
  pred <- x%*%betas
  lik <- sum(pred*y - log(1 + exp(pred)))
  prior <- dmvnorm(betas, mu, Sigma, log=TRUE)
  return(lik + prior)
}
```

```
res <- optim(B, LogReg, y = Y, x = X, mu = muPrior, Sigma = SigmaPrior,
control=list(fnscale=-1), hessian = TRUE)
print("Optimized parameters:")

## [1] "Optimized parameters:"

res$par

## [1]  0.90530134 -0.02102790  0.17601860  0.15948434 -0.11959660 -
0.08521259
## [7] -1.38196346 -0.04416287

print("Inverse Hessian after optimization:")

## [1] "Inverse Hessian after optimization:"

inv.hess <- -solve(res$hessian)

betas <- rmvnorm(1000, res$par, inv.hess)
quantile(betas[,7],probs = c(0.025,0.975))

##       2.5%      97.5%
## -2.1532803 -0.5942022

glm.model = glm(Work~0+., data = data, family = binomial)

print("Parameter values times standard deviation from inverse hessian:")

## [1] "Parameter values times standard deviation from inverse hessian:"

sqrt(diag(inv.hess))*res$par

## [1]  1.3645921167 -0.0003351424  0.0138579691  0.0105409767 -0.0284245269
## [6] -0.0022915605 -0.5400877628 -0.0062419094
```

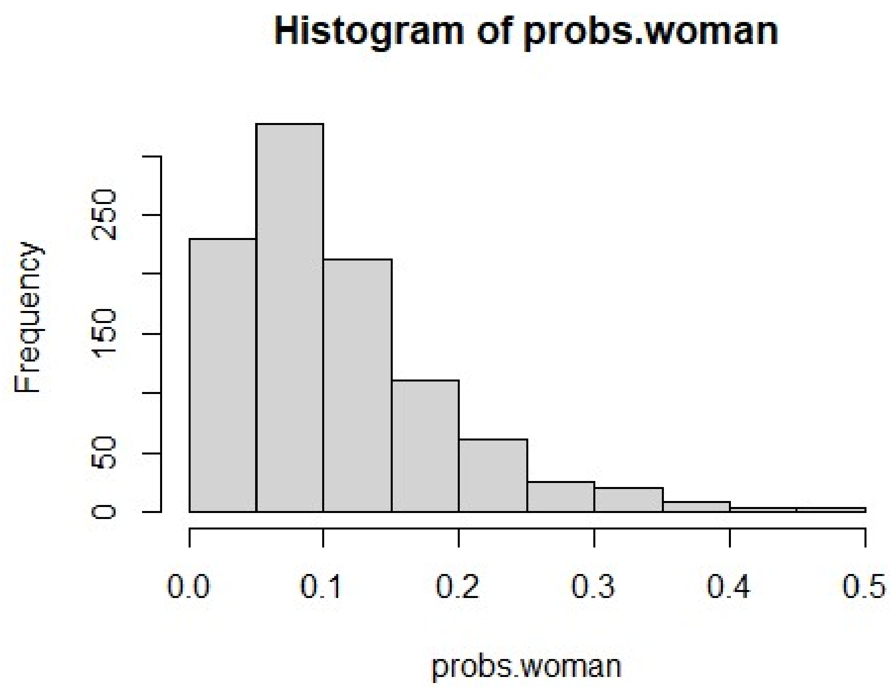
As can be seen for index 7 the feature seems to be important based on the value of it's parameter and the standard deviation, pointing to a great relevance compared to the other values.

b)

As can be seen in the histogram the probabilities for this sample woman is deemed very low by many draws.

```
woman <- c(1,13,8,11,1.21,37,2,0)

probFunc <- function(x, n){
  be <- rmvnorm(n, res$par, inv.hess)
  work = exp(x %*% t(be))/(1+exp(x %*% t(be)))

  return(work)
```

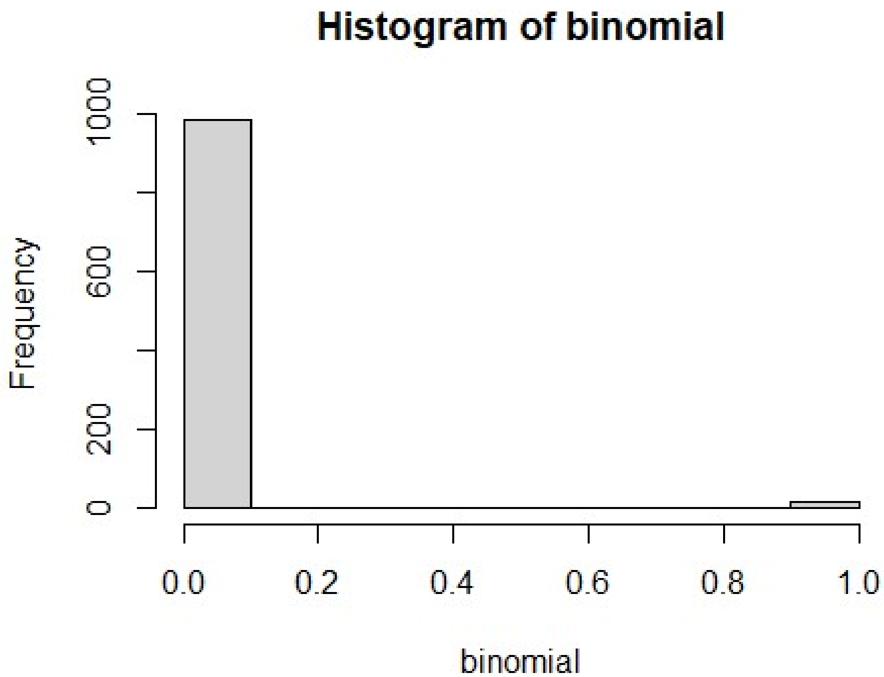
```
}

probs.woman <- probFunc(woman, 1000)
hist(probs.woman)
```



c)

As can be seen in the histogram very few women were classified as working.

```
binomial = 0

for (i in 1:8){
  pred = ifelse(probFunc(woman, 1000)>0.5, 1, 0)
  binomial = binomial + pred
}
hist(binomial)
```



Histogram of binomial
Frequency
0
200
600
1000
0.0
0.2
0.4
0.6
0.8
1.0
binomial