732A91 Lab 2

Fanny Karelius (fanka300), Milda Poceviciute (milpo192)

18 april 2018

1. Linear and polynomial regression

The data set TempLinkoping contains daily temperatures (in Celsius) at Malmslätt (Linköping) during 2016. The response variable is $temp$ and $time = \frac{\text{number of days since beginning of year}}{366}$ is the covariate. A Bayesian analysis of a quadratic regression $temp = \beta_0 + \beta_1 \cdot time + \beta_2 \cdot time^2 + \epsilon$, $\epsilon \overset{iid}{\sim} N(0, \sigma^2)$ is performed.

a)

Conjugate priors:

$$\beta|\sigma^2 \sim N(\mu_0, \sigma^2\Omega_0^{-1})\sigma^2 \sim Inv - \chi^2(\nu_0, \sigma_0^2)$$

The hyperparameters were chosen to be

$$\mu_0 = (-5, 90, -80)\nu_0 = 5\sigma_0^2 = 9\Omega_0 = I_3$$

The prior was based on data from NOAA, the weather of the current week, and results from the lm fit.

```
temps<-read.table("TempLinkoping.dat",header=TRUE)
#response: temp, covariate: time
temp <- temps$temp
time <- temps$time
temps_lm <-lm(temp ~ time + I(time^2))

#hyperparameters
mu0<-c(-5,90,-80) #based on lm
omega0<-diag(3)
inv_omega0 <- solve(omega0)
nu0<-5
sigma20<-9
```

b)

50 simulations from the prior were plotted and compared to the graph from above the prior seemed sensible.

```
library(mvtnorm)
n1 <- 50
sim_X <- rchisq(n1,nu0)
sigma2 <- nu0*sigma20/sim_X

# betas <- matrix(nrow=n1, ncol=3)
# for(i in 1:n1){
#   betas[i,] <- rmvnorm(n = 1, mean=mu0, sigma=sigma2[i]*inv_omega0)
# }
betas_mat1 <- t(apply(as.matrix(sigma2),1,function(x){rmvnorm(n=1,mean=mu0,sigma=x*inv_omega0)}))
betas <- as.data.frame(betas_mat1)
colnames(betas)<-c("beta0", "beta1", "beta2")
```

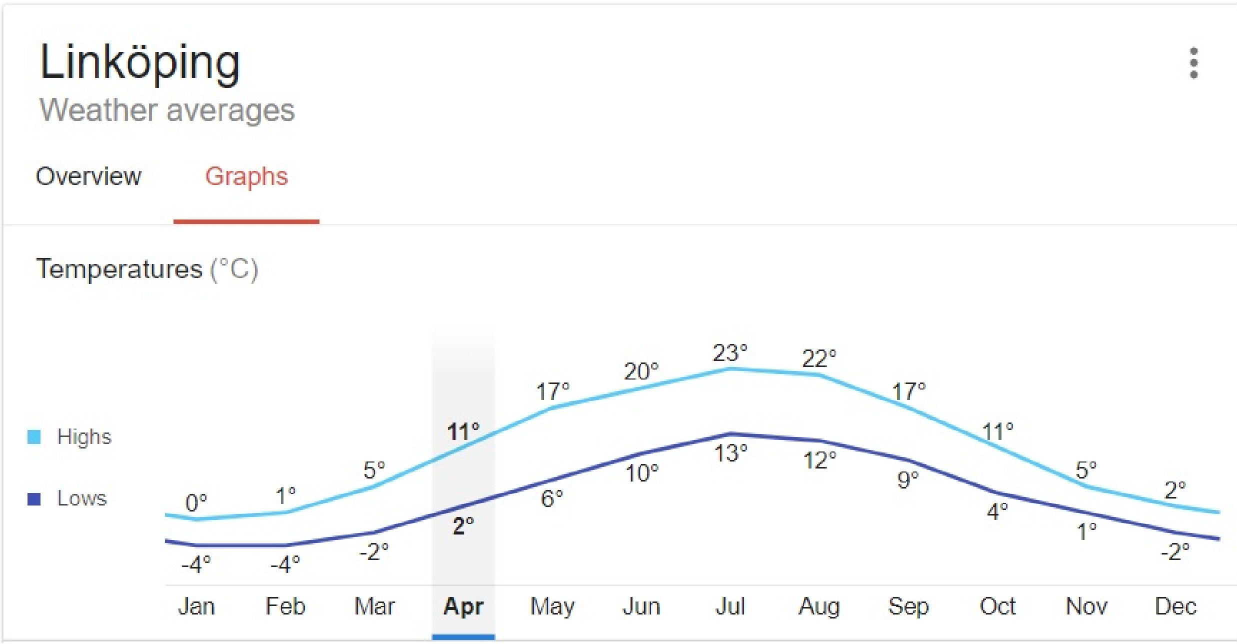
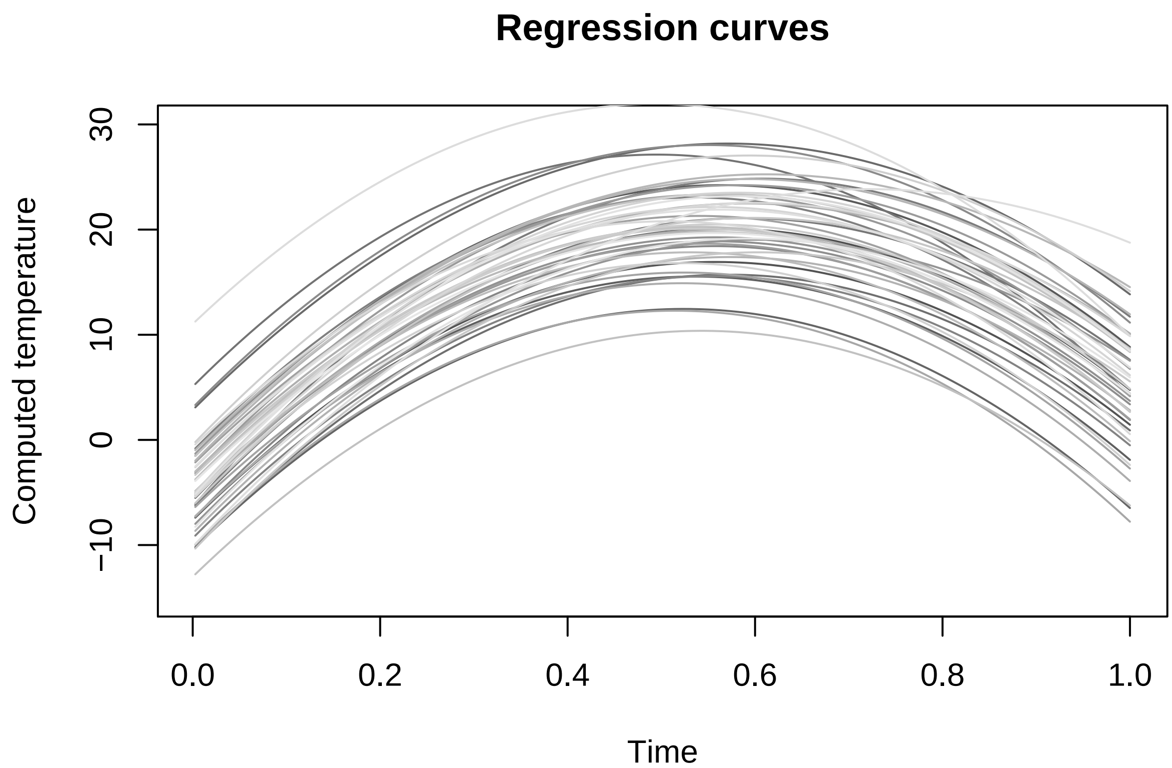


Figure 1: Weather averages in Linköping. Source: NOAA

```r
X <- as.matrix(data.frame(intercept=rep(1, length(time)), time, time^2))
y <- apply(betas_mat1,1,function(a){X%*%a})

cl <- grey.colors(n1)
plot(x=time, y=y[,1], col=cl[1], type="l", ylim=c(-15, 30),
     ylab="Computed temperature", xlab="Time", main="Regression curves")
for(i in 2:ncol(y)){
 lines(time, y[,i], col=cl[i])
}
```

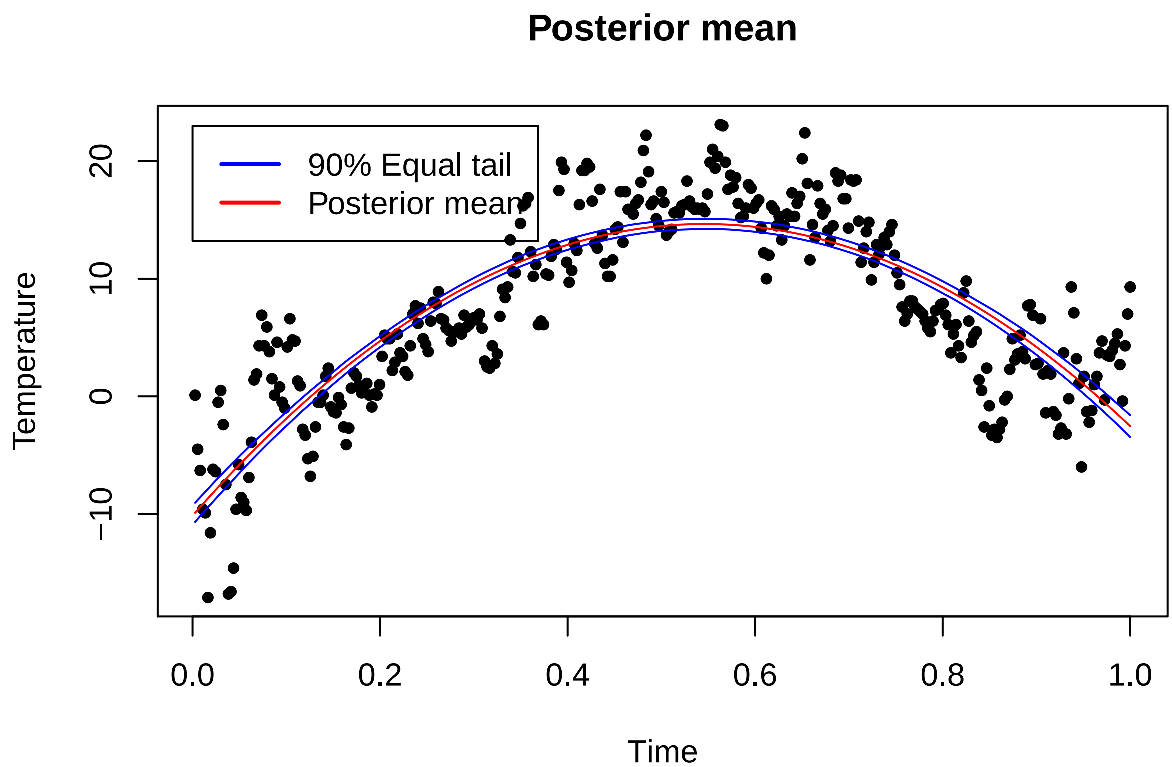


c)

The plot below shows the scatter plot of the temperatures and the posterior mean of the regression function $f(time)$ computed for every value of $time$. It also shows the 95% posterior credible interval for $f(time)$. The interval is narrow and includes only a few data points because the interval shows how certain we are about the posterior mean of $f(time)$ (the trend in the data).

```
n<-200
beta_hat<-solve(t(X)%*%X)%*%t(X)%*%temp #same as found with lm
mun <- solve(t(X)%*%X+omega0)%*%(t(X)%*%X%*%beta_hat+omega0%*%mu0)
omegan <-t(X)%*%X+omega0
inv_omegan <- solve(omegan)
nun <- nu0+nrow(temps)
sigma2n <- (nu0*sigma20+(t(temp)%*%temp+t(mu0)%*%omega0%*%mu0-t(mun)%*%omegan%*%mun))/nun

sim_X2 <- rchisq(n,nun)
sigma2_post <- nun*as.vector(sigma2n)/sim_X2

betas_mat<-t(apply(as.matrix(sigma2_post), 1, function(x){rmvnorm(n=1, mean=mun, sigma=x*inv_omegan)}))
betas_post <- as.data.frame(betas_mat)
colnames(betas_post)<-c("beta0", "beta1", "beta2")

y_post <- apply(betas_mat,1,function(a){X%*%a})
y_mean <- rowMeans(y_post)
```

```
perc = 0.05*n
low <- apply(y_post,1,function(x){x[order(x, decreasing = FALSE)[perc+1]]})
upp <- apply(y_post,1,function(x){x[order(x, decreasing = FALSE)[n-perc-1]]})

plot(x=time, y=temp, pch=20, ylab="Temperature", xlab="Time", main="Posterior mean")
lines(time, y_mean, col="red")
lines(time, low, col="blue")
lines(time, upp, col="blue")
legend(x = 0, y=23, c("90% Equal tail", "Posterior mean"), col=c("blue", "red"), lwd = 2)
```

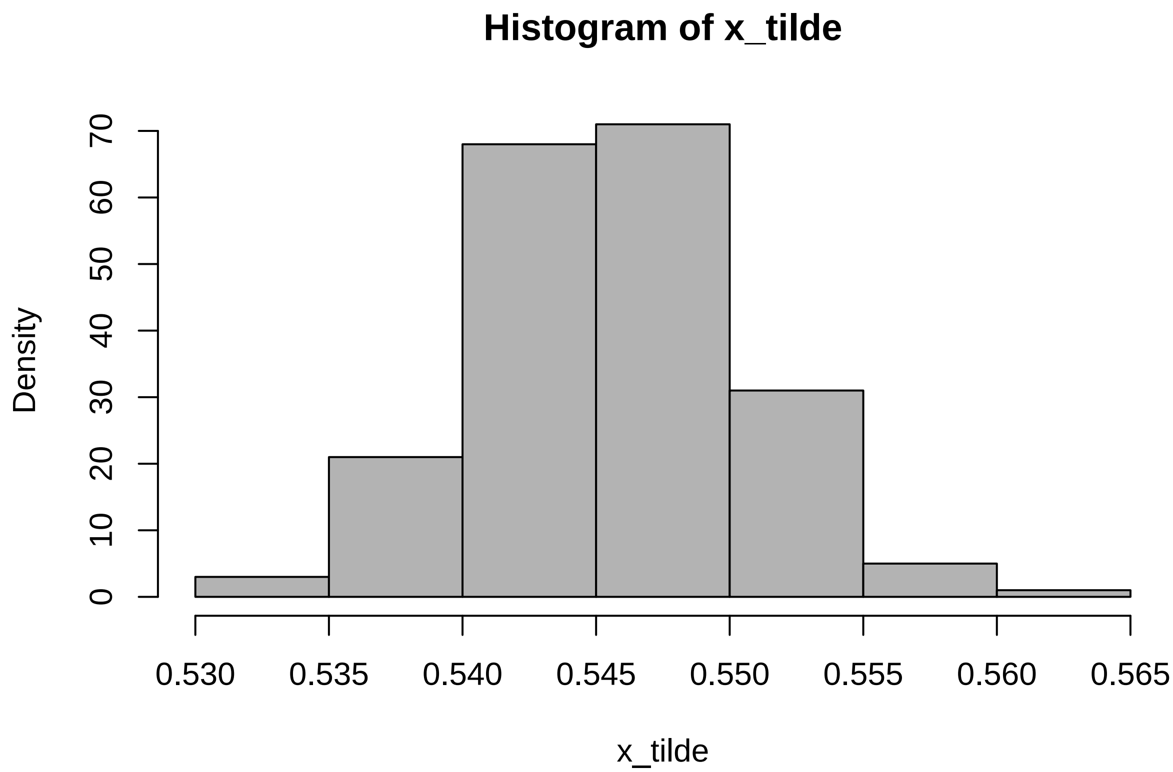


d)

$$f'(time) = \beta_1 + 2\beta_2 \cdot time$$

Setting the derivative to 0 and solving for $time$ we get that the $time$ that maximizes $f(time)$ is $\tilde{x} = -\frac{\beta_1}{2\beta_2}$. From the simulations in c) $\tilde{x}$ was found for each set of βs. The histogram below shows the distribution of $\tilde{x}$.

```
x_tilde <- apply(betas_mat,1,function(x){-x[2]/(2*x[3])})
hist(x_tilde, freq=FALSE, col="grey70")
```



e)

By results from Lecture 5, $\mu_0 = 0$ and $\Omega_o = \lambda I$, where λ would be found either using cross-validation (frequentist way) or using a prior on λ (Bayesian way). The prior would be

$$\beta|\sigma^2 \overset{iid}{\sim} N(0, \sigma^2(\lambda I)^{-1})$$

The larger the λ the more β values tend to 0 and in this way the risk of overfitting is decreased (but the risk of underfitting increases).

2. Posterior approximation for classification with logistic regression

a)

A logistic regression is used for classifying women as working ($y = 1$) and not working ($y = 0$):

$$Pr(y = 1|x) = \frac{exp(x^T\beta)}{1 + exp(x^T\beta)}$$

where x is a 8-dimensional vector containing features (including intercept).

A logistic regression was done using maximum likelihood estimation:

```r
work<-read.table("WomenWork.dat",header=TRUE)
glmModel <- glm(Work ~ 0 + ., data = work, family = binomial)
glmModel
```

```
##
## Call:  glm(formula = Work ~ 0 + ., family = binomial, data = work)
##
## Coefficients:
##    Constant   HusbandInc    EducYears     ExpYears    ExpYears2
##     0.64430     -0.01977      0.17988      0.16751     -0.14436
##         Age  NSmallChild    NBigChild
##    -0.08234     -1.36250     -0.02543
##
## Degrees of Freedom: 200 Total (i.e. Null);  192 Residual
## Null Deviance:       277.3
## Residual Deviance: 222.7     AIC: 238.7
```

b)

The posterior distribution of β is approximated by a multivariate normal distribution:

$$\beta|y, X \sim N(\tilde{\beta}, J_y^{-1}(\tilde{\beta}))$$

where $\tilde{\beta}$ is the posterior mode and $J(\tilde{\beta})$ is the observed Hessian evaluated at the posterior mode. The prior is $\beta \sim N(0, \tau^2 I)$, where $\tau = 10$.

```r
y_work <- work$Work
X_work <- work[,-work$Work]
nparam <- ncol(X_work)

#Prior
mu <- as.vector(rep(0, nparam))
tau2<-100
sigma2 <- tau2*diag(nparam)

logpost_logistic <- function(betas, y, X, mu, sigma2){
  npara <- length(betas)
  X<-as.matrix(X)
  lin_pred <- X%*%betas
  loglik <- sum(lin_pred*y-log(1+exp(lin_pred)))
  if(abs(loglik)==Inf){
    loglik <- -20000
  }
  log_prior <- dmvnorm(betas, mean=mu, sigma2, log=TRUE)
  log_post<-loglik+log_prior
  return(log_post)
}

betas_init <- as.vector(rep(0,nparam))
opt_results <- optim(betas_init, logpost_logistic, gr=NULL, y_work,
                     X_work, mu, sigma2, method=c("BFGS"),control=list(fnscale=-1), hessian=TRUE)

beta_tilde <- opt_results$par
beta_hessian <- -1*opt_results$hessian
```

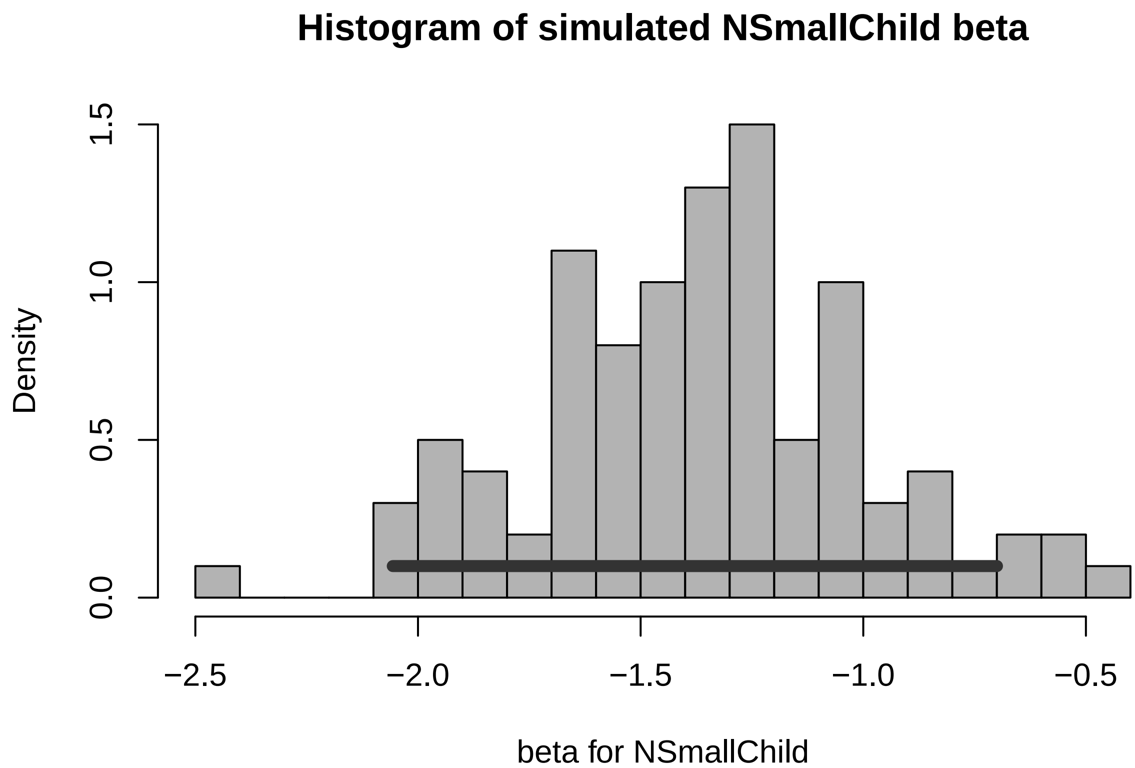
```r
inv_hessian <- solve(beta_hessian) # Posterior covariance matrix is -inv(Hessian)

print(beta_tilde)
```

```
## [1]  0.62672884 -0.01979113  0.18021897  0.16756670 -0.14459669 -0.08206561
## [7] -1.35913317 -0.02468351
```

```r
print(inv_hessian)
```

```
##              [,1]          [,2]          [,3]          [,4]          [,5]
## [1,]  2.266022568  3.338861e-03 -6.545121e-02 -1.179140e-02  0.0457807243
## [2,]  0.003338861  2.528045e-04 -5.610225e-04 -3.125413e-05  0.0001414915
## [3,] -0.065451206 -5.610225e-04  6.218199e-03 -3.558209e-04  0.0018962893
## [4,] -0.011791404 -3.125413e-05 -3.558209e-04  4.351716e-03 -0.0142490853
## [5,]  0.045780724  1.414915e-04  1.896289e-03 -1.424909e-02  0.0555786706
## [6,] -0.030293450 -3.588562e-05 -3.240448e-06 -1.340888e-04 -0.0003299398
## [7,] -0.188748354  5.066847e-04 -6.134564e-03 -1.468951e-03  0.0032082535
## [8,] -0.098023929 -1.444223e-04  1.752732e-03  5.437105e-04  0.0005120144
##               [,6]          [,7]          [,8]
## [1,] -3.029345e-02 -0.1887483542 -0.0980239285
## [2,] -3.588562e-05  0.0005066847 -0.0001444223
## [3,] -3.240448e-06 -0.0061345645  0.0017527317
## [4,] -1.340888e-04 -0.0014689508  0.0005437105
## [5,] -3.299398e-04  0.0032082535  0.0005120144
## [6,]  7.184611e-04  0.0051841611  0.0010952903
## [7,]  5.184161e-03  0.1512621814  0.0067688739
## [8,]  1.095290e-03  0.0067688739  0.0199722657
```

```r
sim_beta<-rmvnorm(n=100, mean=beta_tilde, sigma=inv_hessian)
colnames(sim_beta) <- colnames(X_work)
sim_NSmallChild <- sim_beta[,"NSmallChild"]
perc2 = 0.025*length(sim_NSmallChild)
lower <- sim_NSmallChild[order(sim_NSmallChild, decreasing = FALSE)[perc2+1]]
upper <- sim_NSmallChild[order(sim_NSmallChild, decreasing = FALSE)[length(sim_NSmallChild)-perc2-1]]

hist(sim_NSmallChild, freq = FALSE, breaks = 20, col = "grey70",
     xlab="beta for NSmallChild", main="Histogram of simulated NSmallChild beta")
lines(c(lower, upper), c(0.1,0.1), col="grey20", lwd=6)
```

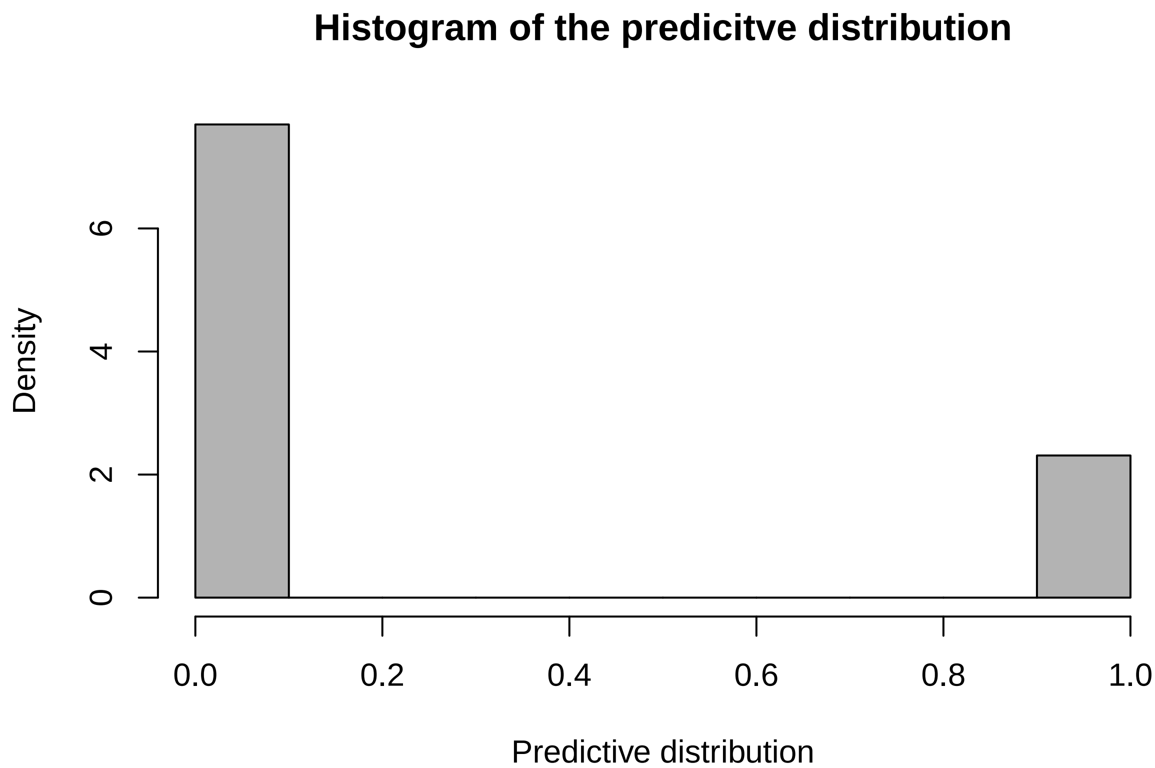


Since the 95% credible interval (black bar) for the simulated β values corresponding to the variable NSmallChild does not include 0, we conclude that the feature is an important determinant of the probability that a woman works.

c)

Using the normal approximation from b), the predictive distribution for the Work variable was simulated when a woman is 40 years old, has one small and one big child, 8 years of education, 10 years of experience and a husband with income of 10.

```
log_pred <- function(husband, edu, exper, age, smallchild, bigchild, n){
  sim_b<-rmvnorm(n=n, mean=beta_tilde, sigma=inv_hessian)
  x <- c(1, husband, edu, exper, (exper/10)^2, age, smallchild, bigchild)
  prob_pred <- exp(sim_b%*%x)/(1+exp(sim_b%*%x))
  u <- runif(n = n)
  y_pred <- ifelse(prob_pred>u, 1, 0)
  y_pred
}

woman <- log_pred(10, 8, 10, 40, 1, 1, 1000)
hist(woman, freq=FALSE, col = "grey70", xlab="Predictive distribution",
     main="Histogram of the predicitve distribution")
```



From the predictive distribution we conclude that it is more likely that the woman does not work.

Appendix

```
temps<-read.table("TempLinkoping.dat",header=TRUE)
#response: temp, covariate: time
temp <- temps$temp
time <- temps$time
temps_lm <-lm(temp ~ time + I(time^2))

#hyperparameters
mu0<-c(-5,90,-80) #based on lm
omega0<-diag(3)
inv_omega0 <- solve(omega0)
nu0<-5
sigma20<-9
library(mvtnorm)
n1 <- 50
sim_X <- rchisq(n1,nu0)
sigma2 <- nu0*sigma20/sim_X

# betas <- matrix(nrow=n1, ncol=3)
# for(i in 1:n1){
#   betas[i,] <- rmvnorm(n = 1, mean=mu0, sigma=sigma2[i]*inv_omega0)
```

```r
# }
betas_mat1 <- t(apply(as.matrix(sigma2),1,function(x){rmvnorm(n=1,mean=mu0,sigma=x*inv_omega0)}))
betas <- as.data.frame(betas_mat1)
colnames(betas)<-c("beta0", "beta1", "beta2")

X <- as.matrix(data.frame(intercept=rep(1, length(time)), time, time^2))
y <- apply(betas_mat1,1,function(a){X%*%a})

cl <- grey.colors(n1)
plot(x=time, y=y[,1], col=cl[1], type="l", ylim=c(-15, 30),
     ylab="Computed temperature", xlab="Time", main="Regression curves")
for(i in 2:ncol(y)){
 lines(time, y[,i], col=cl[i])
}
n<-200
beta_hat<-solve(t(X)%*%X)%*%t(X)%*%temp #same as found with lm
mun <- solve(t(X)%*%X+omega0)%*%(t(X)%*%X%*%beta_hat+omega0%*%mu0)
omegan <-t(X)%*%X+omega0
inv_omegan <- solve(omegan)
nun <- nu0+nrow(temps)
sigma2n <- (nu0*sigma20+(t(temp)%*%temp+t(mu0)%*%omega0%*%mu0-t(mun)%*%omegan%*%mun))/nun

sim_X2 <- rchisq(n,nun)
sigma2_post <- nun*as.vector(sigma2n)/sim_X2

betas_mat<-t(apply(as.matrix(sigma2_post), 1, function(x){rmvnorm(n=1, mean=mun, sigma=x*inv_omegan)}))
betas_post <- as.data.frame(betas_mat)
colnames(betas_post)<-c("beta0", "beta1", "beta2")

y_post <- apply(betas_mat,1,function(a){X%*%a})
y_mean <- rowMeans(y_post)

perc = 0.05*n
low <- apply(y_post,1,function(x){x[order(x, decreasing = FALSE)[perc+1]]})
upp <- apply(y_post,1,function(x){x[order(x, decreasing = FALSE)[n-perc-1]]})

plot(x=time, y=temp, pch=20, ylab="Temperature", xlab="Time", main="Posterior mean")
lines(time, y_mean, col="red")
lines(time, low, col="blue")
lines(time, upp, col="blue")
legend(x = 0, y=23, c("90% Equal tail", "Posterior mean"), col=c("blue", "red"), lwd = 2)
x_tilde <- apply(betas_mat,1,function(x){-x[2]/(2*x[3])})
hist(x_tilde, freq=FALSE, col="grey70")
work<-read.table("WomenWork.dat",header=TRUE)
glmModel <- glm(Work ~ 0 + ., data = work, family = binomial)
glmModel
y_work <- work$Work
X_work <- work[,-work$Work]
nparam <- ncol(X_work)

#Prior
mu <- as.vector(rep(0, nparam))
tau2<-100
```

```r
sigma2 <- tau2*diag(nparam)

logpost_logistic <- function(betas, y, X, mu, sigma2){
  npara <- length(betas)
  X<-as.matrix(X)
  lin_pred <- X%*%betas
  loglik <- sum(lin_pred*y-log(1+exp(lin_pred)))
  if(abs(loglik)==Inf){
    loglik <- -20000
  }
  log_prior <- dmvnorm(betas, mean=mu, sigma2, log=TRUE)
  log_post<-loglik+log_prior
  return(log_post)
}

betas_init <- as.vector(rep(0,nparam))
opt_results <- optim(betas_init, logpost_logistic, gr=NULL, y_work,
                     X_work, mu, sigma2, method=c("BFGS"),control=list(fnscale=-1), hessian=TRUE)

beta_tilde <- opt_results$par
beta_hessian <- -1*opt_results$hessian
inv_hessian <- solve(beta_hessian) # Posterior covariance matrix is -inv(Hessian)

print(beta_tilde)
print(inv_hessian)

sim_beta<-rmvnorm(n=100, mean=beta_tilde, sigma=inv_hessian)
colnames(sim_beta) <- colnames(X_work)
sim_NSmallChild <- sim_beta[,"NSmallChild"]
perc2 = 0.025*length(sim_NSmallChild)
lower <- sim_NSmallChild[order(sim_NSmallChild, decreasing = FALSE)[perc2+1]]
upper <- sim_NSmallChild[order(sim_NSmallChild, decreasing = FALSE)[length(sim_NSmallChild)-perc2-1]]

hist(sim_NSmallChild, freq = FALSE, breaks = 20, col = "grey70",
     xlab="beta for NSmallChild", main="Histogram of simulated NSmallChild beta")
lines(c(lower, upper), c(0.1,0.1), col="grey20", lwd=6)
log_pred <- function(husband, edu, exper, age, smallchild, bigchild, n){
  sim_b<-rmvnorm(n=n, mean=beta_tilde, sigma=inv_hessian)
  x <- c(1, husband, edu, exper, (exper/10)^2, age, smallchild, bigchild)
  prob_pred <- exp(sim_b%*%x)/(1+exp(sim_b%*%x))
  u <- runif(n = n)
  y_pred <- ifelse(prob_pred>u, 1, 0)
  y_pred
}

woman <- log_pred(10, 8, 10, 40, 1, 1, 1000)
hist(woman, freq=FALSE, col = "grey70", xlab="Predictive distribution",
     main="Histogram of the predicitve distribution")
```