Computer Lab2 - 732A73

Helena Llorens Lluís (hllor282), Yi Yang (yiyan338)

1. Linear and polynomial regression

We aim to model daily average temperatures in Linköping using the following quadratic Bayesian regression model:

$$temp = \beta_0 + \beta_1 \cdot time + \beta_2 \cdot time^2 + \epsilon,\ \epsilon \sim N(0, \sigma^2)$$

Here, *time* is defined as the number of days since the start of the observation period divided by 365, i.e., a scaled time variable ranging from 0 to 1.

We place a conjugate prior on the parameters β and the error variance σ^2:

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

We began with the following prior hyperparameters: $\mu_0 = (0, -100, 100)^T$, $\Omega_0 = 0.01 \cdot I_3$, $\nu_0 = 1$ and $\sigma_0^2 = 1$.

However, when sampling from this prior and visualizing the resulting regression curves, we found that most curves did not align with our prior belief that temperature should vary smoothly within a realistic range (-20°C to +30°C) over the course of the year. Many curves were too extreme due to overly vague prior variances.

To better reflect plausible seasonal temperature patterns, we adjusted the priors to: $\mu_0 = (20, -100, 100)$ (reflecting a mean intercept of 20°C) and $\Omega_0 = 50 \cdot I_3$ (wider spread but still regularizing). These new priors produce more reasonable prior predictive regression curves.

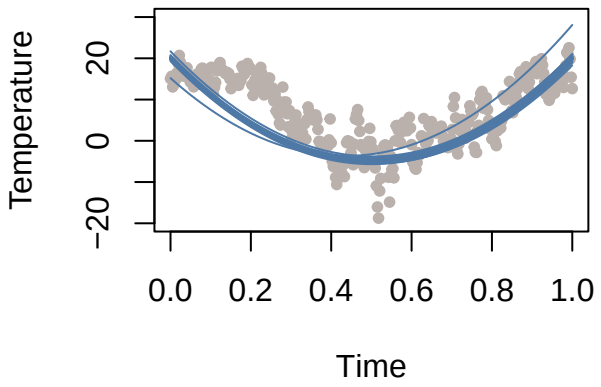

Figure 1: Prior Predictive Regression Curves

The prior predictive curves reflect a wide but plausible range of behaviors for seasonal temperature trends in Linköping. Most curves follow a quadratic seasonal pattern, dipping around the center of the time range, consistent with expected annual temperature fluctuations. This confirms that the revised priors are now better aligned with prior beliefs.

We now aim to simulate draws from the joint posterior distribution of the model parameters $\beta_0, \beta_1, \beta_2$ and σ^2. Based on conjugate Bayesian linear regression theory, the posterior distributions are given by:

$$\beta|\sigma^2 \sim N(\mu_n, \sigma^2\Omega_n^{-1}), \quad \sigma^2|y \sim \text{Inv} - \chi^2(\nu_n, \sigma_n^2)$$

with updated hyperparameters:

$$\mu_n = (X'X + \Omega_0)^{-1}(X'X\hat{\beta} + \Omega_0\mu_0)$$
$$\Omega_n = X'X + \Omega_0$$
$$\nu_n = \nu_0 + n$$
$$\sigma_n^2 = \frac{1}{\nu_n}(\nu_0\sigma_0^2 + (y'y + \mu_0'\Omega_0\mu_0 + \mu_n'\Omega_n\mu_n))$$

We simulate from the posterior and visualize the marginal distributions of each parameter:

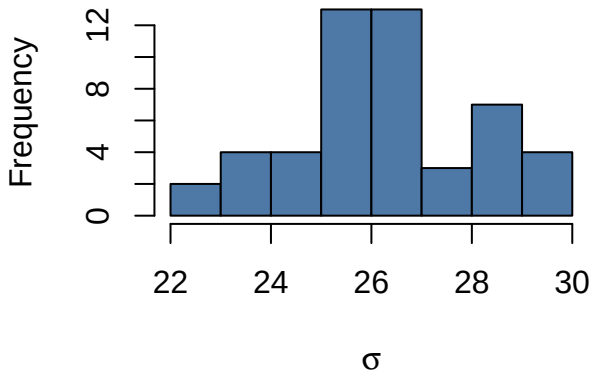

Figure 2: Marginal posterior of σ

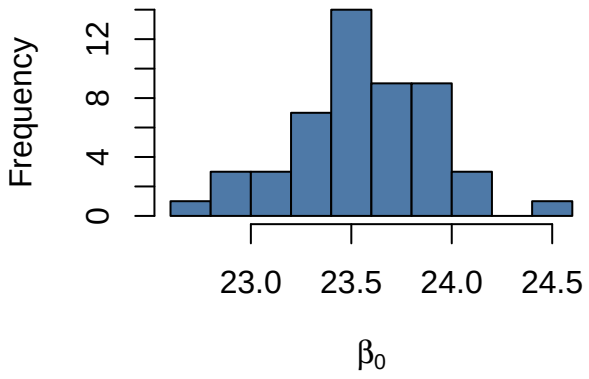

Figure 3: Marginal posterior of β_0

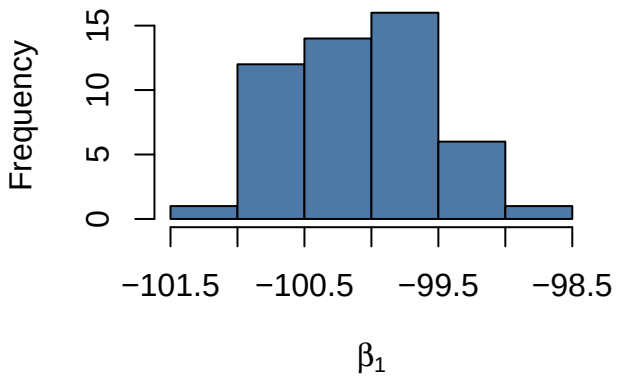

Figure 4: Marginal posterior of β_1

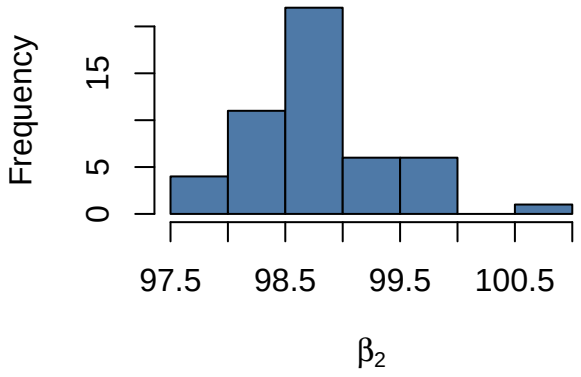

Figure 5: Marginal posterior of β_2

To summarize the uncertainty in the regression function $f(time) = E[temp|time] = \beta_0 + \beta_1 \cdot time + \beta_2 \cdot time^2$, we compute the posterior median curve for $f(time)$ in blue and the 90% equal-tail credible interval in red.

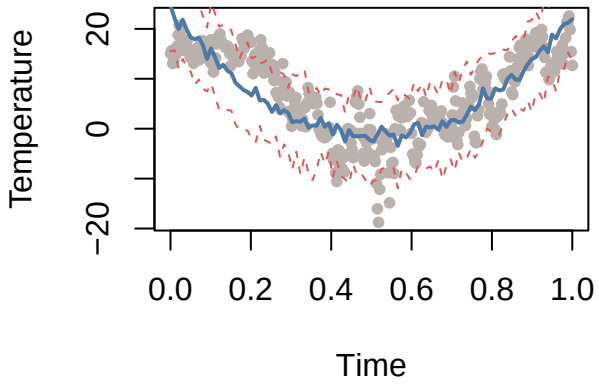

Figure 6: Posterior Regression Curve with 90% Credible Intervals

The posterior median curve closely follows the seasonal temperature pattern suggested by the data. The 90% credible intervals cover most observed data points, as expected. Since the regression model accounts for observation noise ($\epsilon \sim N(0, \sigma^2)$) we expect a large proportion of data points to fall within the interval—but not necessarily all. Roughly 90% of them should, under the assumption that the model is well-calibrated.

We are interested in identifying the time $\tilde{x}$ at which the expected temperature reaches its minimum. Since the regression function is a quadratic polynomial, the minimum occurs at

$$\tilde{x} = -\frac{\beta_1}{2\beta_2}$$

Using the posterior draws of β_1 and β_2 obtained previously, we simulate the posterior distribution of $\tilde{x}$.

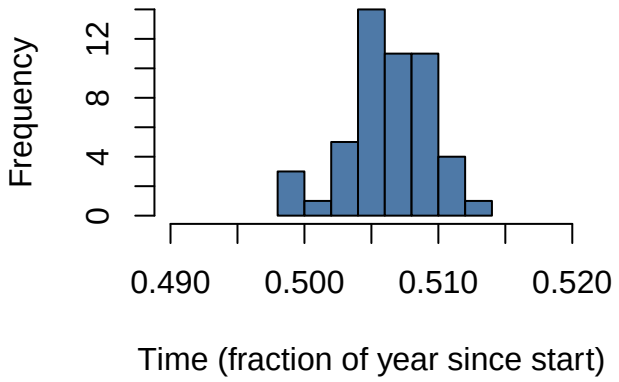

Figure 7: Posterior Distribution of $\tilde{x}$

As shown in the histogram, the posterior distribution of $\tilde{x}$ is concentrated around the middle of the winter period (January), which aligns well with prior expectations for the coldest time of year in Linköping. The narrow distribution suggests high certainty in the estimate, given the data.

We now estimate a polynomial regression of order 10. However, since higher-order polynomial terms may not contribute significantly to explaining the data—and could even lead to overfitting—we choose a shrinkage prior to regularize the model and mitigate this risk.

We again use the conjugate prior:

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

To reflect our belief that only the lower-order terms (e.g., constant, linear, quadratic) are likely to be influential, we set: $\mu_0 = (20, -100, 100, 0, 0, 0, 0, 0, 0, 0, 0)^T$, $\Omega_0 = 50 \cdot I_1 1$, $\nu_0 = 1$ and $\sigma_0^2 = 1$.

This prior centers the first three coefficients on meaningful values while shrinking the remaining higher-order terms toward zero, discouraging unnecessary model complexity.

We now simulate from the joint prior to generate prior predictive regression curves:

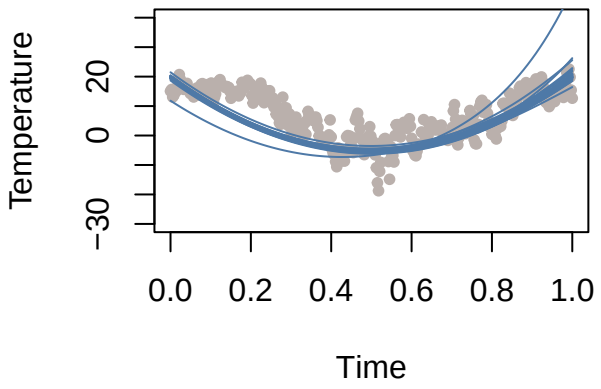

Figure 8: Prior Predictive Regression Curves

As shown in the plot, the shrinkage prior successfully constrains the model's flexibility. Most curves resemble smooth, low-degree polynomials, and avoid the erratic behavior typically associated with unregularized high-order models. This behavior is consistent with our prior belief that only the lower-degree terms are essential for capturing the seasonal temperature trend.

2. Posterior approximation for classification with logistic regression

We aim to use a normal approximation to the posterior distribution in a Bayesian logistic regression model. The logistic regression model takes the form:

$$P(y_i = 1 \mid x_i, \beta) = \frac{1}{1 + \exp(-x_i^\top \beta)}$$

where x_i represents the covariates (including an intercept) and β are the regression coefficients.Each observation contains six variables related to a particular disease: five features (Age, Gender, Duration of Symptoms, Dyspnoea, and White Blood Count) and one intercept term.We first standardize the continuous variables — Age, Duration of Symptoms, and White Blood Count — using the formula:

$$x_{\text{standardized}} = \frac{x - \mu}{\sigma}$$

Next, we construct the log-posterior distribution of the parameters:

$$\log p(\beta \mid y, X) = \log p(y \mid \beta) + \log p(\beta)$$

where $logp(\beta)$ is the log-prior and $\log p(y \mid \beta)$ is the log-likelihood. Assuming a multivariate normal prior:

$$\beta \sim N(0, \tau^2 I)$$

where $\tau^2 = 2$.

Then we use `optim` function to find the mode of the posterior distribution (i.e., the MAP estimate) and to compute the `observed information matrix`,which is the negative of the Hessian at the mode. The approximate normal posterior is :

$$\beta \mid \mathbf{y}, \mathbf{x} \sim N\left(\tilde{\beta}, J_{\mathbf{y}}^{-1}(\tilde{\beta})\right)$$

The numeric values of $\tilde{\beta}$ and $J_y^{-1}(\tilde{\beta})$ are:

Table 1: Mode of beta

Intercept	age	gender	duration_of_symptoms	dyspnoea	white_blood
-0.3882422	-0.2661479	-0.6423896	0.19313	-0.1375241	-0.0449894

Table 2: Observed Information Matrix (Inverse Negative Hessian)

Intercept	age	gender	duration_of_symptoms	dyspnoea	white_blood
0.0977472	-0.0005446	-0.0346230	-0.0029001	-0.0797577	0.0015042
-0.0005446	0.0163897	0.0024843	-0.0001409	0.0009300	-0.0010567
-0.0346230	0.0024843	0.0631838	0.0028489	0.0027927	-0.0013382
-0.0029001	-0.0001409	0.0028489	0.0149913	0.0006546	-0.0006456
-0.0797577	0.0009300	0.0027927	0.0006546	0.0975459	-0.0006531
0.0015042	-0.0010567	-0.0013382	-0.0006456	-0.0006531	0.0163251

We then compute an approximate 95% equal-tail posterior credible interval for the variable `Age`.

```
## 95% approximate credibility intervals for Age is [ -0.5170715 -0.01522428 ].
```

Since this interval does not include `0`, we can say with 95% posterior probability (under the normal approximation) that the coefficient for Age is not zero. This implies that Age has `a statistically significant effect` on the probability of having the disease. Using `glm` function,we compare the posterior means to the maximum likelihood estimates (MLEs) obtained from the logistic regression model.

```
## [1] "Maximum Likelihood Estimates (MLE) for variable Age:"
```

```
##        age
## -0.2676216
```

```
## [1] "Posterior Means for variable Age:"
```

```
## [1] -0.2661479
```

After comparing, we observe that the posterior mean values closely match the MLEs from the glm Model, thereby confirming the validity and stability of our estimation procedure.

We then use the normal approximation to the posterior distribution of the regression coefficients obtained in (a), to simulate draws from the posterior predictive distribution of $Pr(y = 1 \mid x)$ for the specific individual. After standardizing the covariates, We then draw samples from the posterior distribution.For each draw of $\beta^{(s)}$,we compute:

$$Pr(y = 1 \mid x_{\text{new}}, \beta^{(s)}) = \frac{1}{1 + \exp(-x_{\text{new}}^{\top}\beta^{(s)})}$$

Finally, we visualize the distribution of predictive probabilities using a histogram.

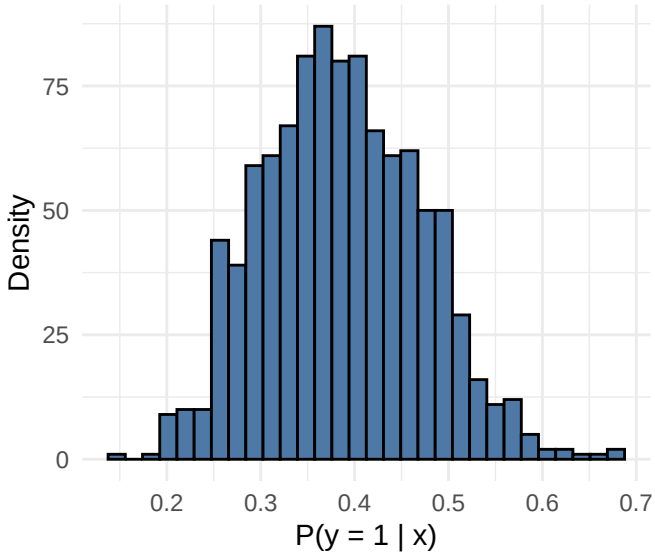

Figure 9: Posterior Predictive Distribution

Appendix

Assignment 1

```
# load data
data <- read.csv("D:\\liu\\BL\\Bayesian-Learning\\lab2\\temp_linkoping.csv")
n <- nrow(data)
y <- data$temp
x <- data$time

# (a)
# prior hyperparameters
mu0 <- t(c(20, -100, 100))
omega0 <- 50 * diag(3)
nu0 <- 1
sigma0 <- 1

# scale inverse chi function
scale_inv_chi <- function(nu0, sigma0, N) {
   sigma0 <- as.numeric(sigma0)
   samples <- (nu0 * sigma0) / rchisq(N, df = nu0)
   return(samples)
}

# prior distributions
N <- 50
set.seed(123)
prior_sigma <- scale_inv_chi(nu0, sigma0, N)
prior_beta <- matrix(NA, nrow = N, ncol = 3)
for(i in 1:N){
```

```r
  prior_beta[i, ] <- mvtnorm::rmvnorm(n = 1, mean = mu0, sigma = prior_sigma[i] * solve(omega0))
}

# Plot prior regression curves
plot(data$time, y, col = "#BA0AC", pch = 20,
     xlab = "Time", ylab = "Temperature", ylim = c(-20, 30))
for (i in 1:N) {
  curve <- prior_beta[i, 1] + prior_beta[i, 2] * data$time + prior_beta[i, 3] * data$time^2
  lines(data$time, curve, col = "#4E79A7")
}


# (b)
# posterior distribution function
X <- matrix(c(rep(1, n), data$time, data$time^2), ncol = 3)
Y <- y
posterior <- function(mu0, omega0, nu0, sigma0, X, Y, N, p){
  beta <- solve(t(X) %*% X) %*% t(X) %*% Y
  mu_n <- solve(t(X) %*% X + omega0) %*% (t(X) %*% X %*% beta + omega0 %*% matrix(mu0, ncol = 1))
  omega_n <- t(X) %*% X + omega0
  nu_n <- nu0 + nrow(X)
  sigma_n <- (nu0 * sigma0 +
                (t(Y) %*% Y + t(matrix(mu0, ncol = 1)) %*% omega0 %*% matrix(mu0, ncol = 1) -
                   t(mu_n) %*% omega_n %*% mu_n))/nu_n
  post_sigma <- scale_inv_chi(nu_n, sigma_n, N)
  post_beta <- matrix(NA, nrow = N, ncol = p + 1)
  for(i in 1:N){
    post_beta[i, ] <- mvtnorm::rmvnorm(n = 1, mean = mu_n, sigma = post_sigma[i] * solve(omega_n))
  }

  return(list(sigma = post_sigma, beta = post_beta))
}

set.seed(123)
post <- posterior(mu0, omega0, nu0, sigma0, X, Y, N, 2)

# (i) Histograms of the magrinal posterior for each parameter
hist(post$sigma, col = "#4E79A7", xlab = expression(sigma), main = "")
hist(post$beta[, 1], col = "#4E79A7", xlab = expression(beta[0]), main = "")
hist(post$beta[, 2], col = "#4E79A7", xlab = expression(beta[1]), main = "")
hist(post$beta[, 3], col = "#4E79A7", xlab = expression(beta[2]), main = "")

# (ii)
# draws from the function
time_grid <- seq(min(data$time), max(data$time), length.out = 100)
f_pred_draws <- matrix(NA, nrow = N, ncol = length(time_grid))
for (i in 1:N) {
  f_mean <- post$beta[i, 1] + post$beta[i, 2] * time_grid + post$beta[i, 3] * time_grid^2
  f_pred_draws[i, ] <- rnorm(length(time_grid), mean = f_mean, sd = sqrt(post$sigma[i]))
}

# compute median and CI
pred_median <- apply(f_pred_draws, 2, median)
```

```r
pred_lb <- apply(f_pred_draws, 2, quantile, probs = 0.05)
pred_ub <- apply(f_pred_draws, 2, quantile, probs = 0.95)

# Plot data
plot(x, y, col = "#BAB0AC", pch = 20,
     main = "",
     xlab = "Time", ylab = "Temperature")

# Add posterior median curve
lines(time_grid, pred_median, col = "#4E79A7", lwd = 2)

# Add 90% posterior credible interval
lines(time_grid, pred_lb, col = "#E15759", lty = 2)
lines(time_grid, pred_ub, col = "#E15759", lty = 2)


# (c)
x_tilde <- -post$beta[, 2] / (2*post$beta[, 3])

# Plot histogram
hist(x_tilde, col = "#4E79A7",
     main = "",
     xlab = "Time (fraction of year since start)", xlim = c(0.49, 0.52))


# (d)
# hyperparameters for the 10th-order polynomial regression with shrinkage prior
p <- 10
mu0_poly <- c(20, -100, 100, rep(0, p - 2))
lambda <- 50
omega0_poly <- lambda * diag(p + 1)

# Plot prior predictive regression curves for poly
plot(data$time, y, col = "#BAB0AC", pch = 20, main = "",
     xlab = "Time", ylab = "Temperature", ylim = c(-30, 40))
for (i in 1:N) {
  beta_i <- mvtnorm::rmvnorm(1, mu0_poly, prior_sigma[i] * solve(omega0_poly))
  y_pred <- sapply(time_grid, function(t) sum(beta_i * t^(0:p)))
  lines(time_grid, y_pred, col = "#4E79A7")
}
```

Assignment2

```r
#assignment2
library("mvtnorm")
#(a)
data <- read.csv("Disease.csv")
data <- as.data.frame(data)
#normalize age,symtoms and white blood counts
age_mean <- mean(data$age)
age_sd <- sd(data$age)
duration_mean <- mean(data$duration_of_symptoms)
```

```r
duration_sd <- sd(data$duration_of_symptoms)
white_mean <- mean(data$white_blood)
white_sd <- sd(data$white_blood)

data[,2] <- (data[,2]-age_mean)/age_sd
data[,4] <- (data[,4]-duration_mean)/duration_sd
data[,6] <- (data[,6]-white_mean)/white_sd

y <- data[,7] #response
X <- as.matrix(data[,1:6]) #Covariates
Xnames <- colnames(X)

#log posterior distribution
LogPostLogistic <- function(betas,y,X,mu,Sigma){
  linPred <- X%*%betas;
  logLik <- sum( linPred*y - log(1 + exp(linPred)) );
  #log prior
  logPrior <- dmvnorm(betas, mu, Sigma, log=TRUE);

  return(logLik + logPrior)
}

#set the prior
mu <- rep(0,ncol(X))
sigma <- 2*diag(ncol(X))
#initial beta
initVal <- rep(0,ncol(X))

#using optim function fine the mode of beta(mean of posterior dist) and observed information at the mod
OptimRes <- optim(initVal,LogPostLogistic,gr=NULL,y,X,mu,sigma,
                  method=c("BFGS"),control=list(fnscale=-1),hessian=TRUE)

#find the mu of approx posterior (mode of beta)
approxPostMode <-matrix(OptimRes$par,1,ncol(X))
colnames(approxPostMode) <- Xnames
#find the sd of approx posterior
approxPostStd <- sqrt(diag(solve(-OptimRes$hessian))) # Computing approximate standard deviations.
approxPostStd <- matrix(approxPostStd,1,ncol(X))
colnames(approxPostStd) <- Xnames

#find the 95% equal tail posterior probability interval for coefficient Age
Cred_int <- matrix(0,2,ncol(X)) # Create 95 % approximate credibility intervals for each coefficient
Cred_int[1,] <- approxPostMode - 1.96*approxPostStd
Cred_int[2,] <- approxPostMode + 1.96*approxPostStd
colnames(Cred_int) <- Xnames

cat("95% approximate credibility intervals for Age is [",Cred_int[1,2],Cred_int[2,2],"].","\n")

#Comparison with glm
glmModel<- glm(class_of_diagnosis ~ 0 + ., data = data, family = binomial)
print("Maximum Likelihood Estimates (MLE) for variable Age:")
coef(glmModel)[2]
print("Posterior Means for variable Age:")
```

```r
approxPostMode[2]

#(b)
#normalizing the sample
xnew <- c(1,38,1,10,0,11000)
xnew_normalized <- c(1,(38-age_mean)/age_sd,1,(10-duration_mean)/duration_sd,
                     0,(11000-white_mean)/white_sd)

#sampling beta from the posterior distribution
set.seed(12345)
beta_sample <- rmvnorm(1000,mean =approxPostMode,sigma=solve(-OptimRes$hessian))

#compute the predictive probability for different beta samples
predict_probs <- numeric(nrow(beta_sample))
for (i in 1:nrow(beta_sample)) {
  beta <- beta_sample[i,]
  linPred <- sum(xnew*beta)
  prob <- 1/(1+exp(-linPred))
  predict_probs[i] <- prob
}


hist(predict_probs,breaks = 30,probability = T,main = "Posterior Predictive Distribution",xlab = "P(y=1
```