TDDE07 - Lab 2

Arvid Edenheim - arved490
Sophie Lindberg - sopli268

2019-05-05

1. Linear and polynomial regression

1a)

Before looking at the data, we came to the conclusion that the intercept should be about -10 degrees, with a cyclic temperature throughout the year and a maximum at 7 months.

μ_0 = [-10, 100, -90] ω_0 =

$$\begin{pmatrix} 15 & 0 & 0 \\ 0 & 15 & 0 \\ 0 & 0 & 15 \end{pmatrix}$$

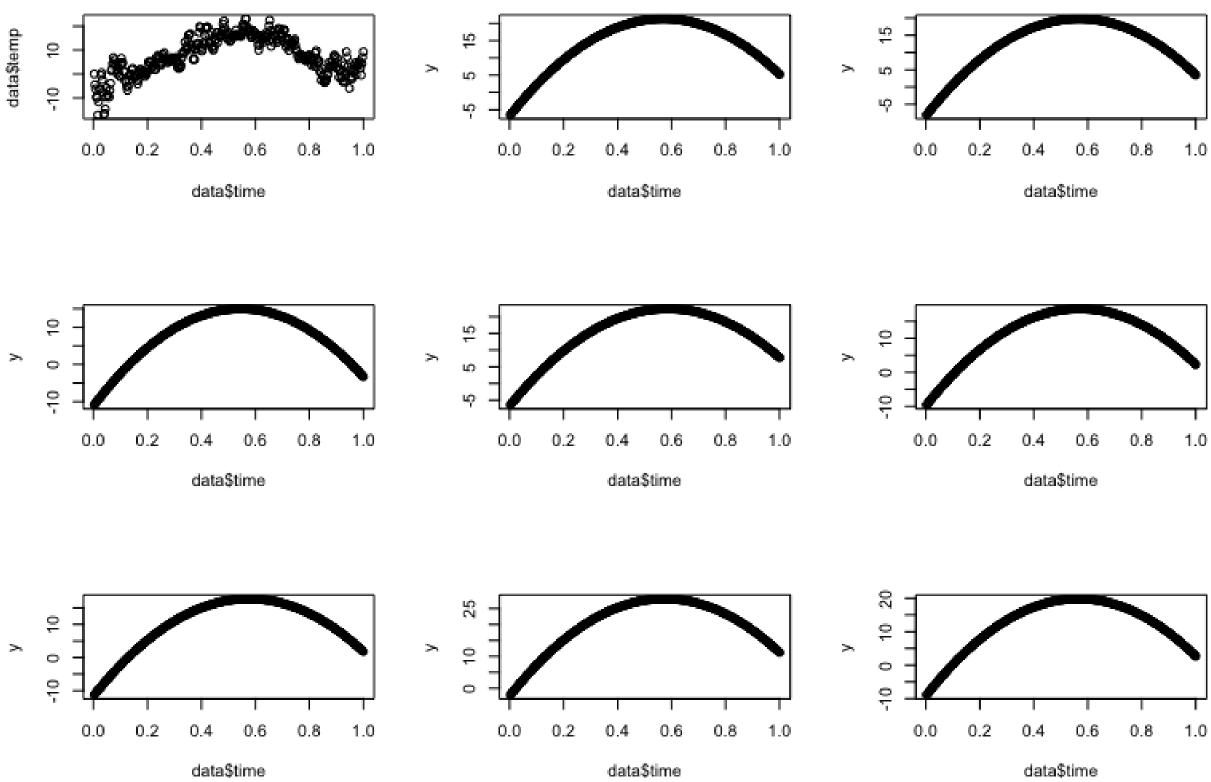

Figure 1: Plotting the chosen hyperparameters

1b)

Simulating from the joint posterior distribution of the parameters resulted in the following marginal posteriors.

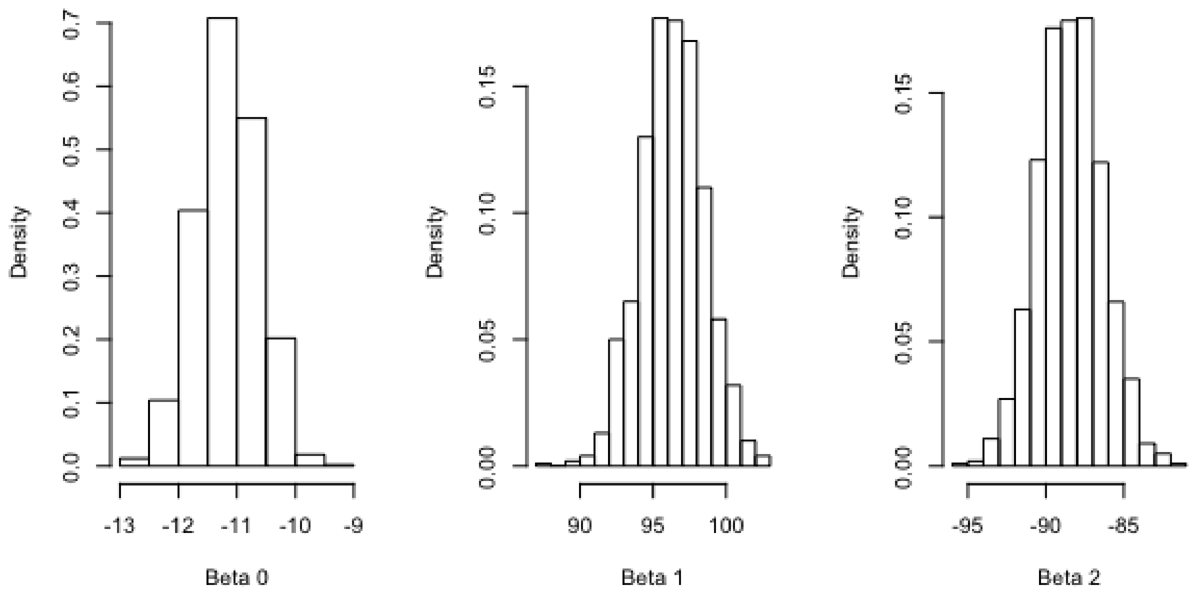

Figure 2: Plotting the marginal posteriors

The figure below shows the plotted data along with the curve for the posterior median of the regression function. Furthermore, a 95 % credible interval has been calculated. The interval bands doesn't contain most od the data points due to the high credibility.

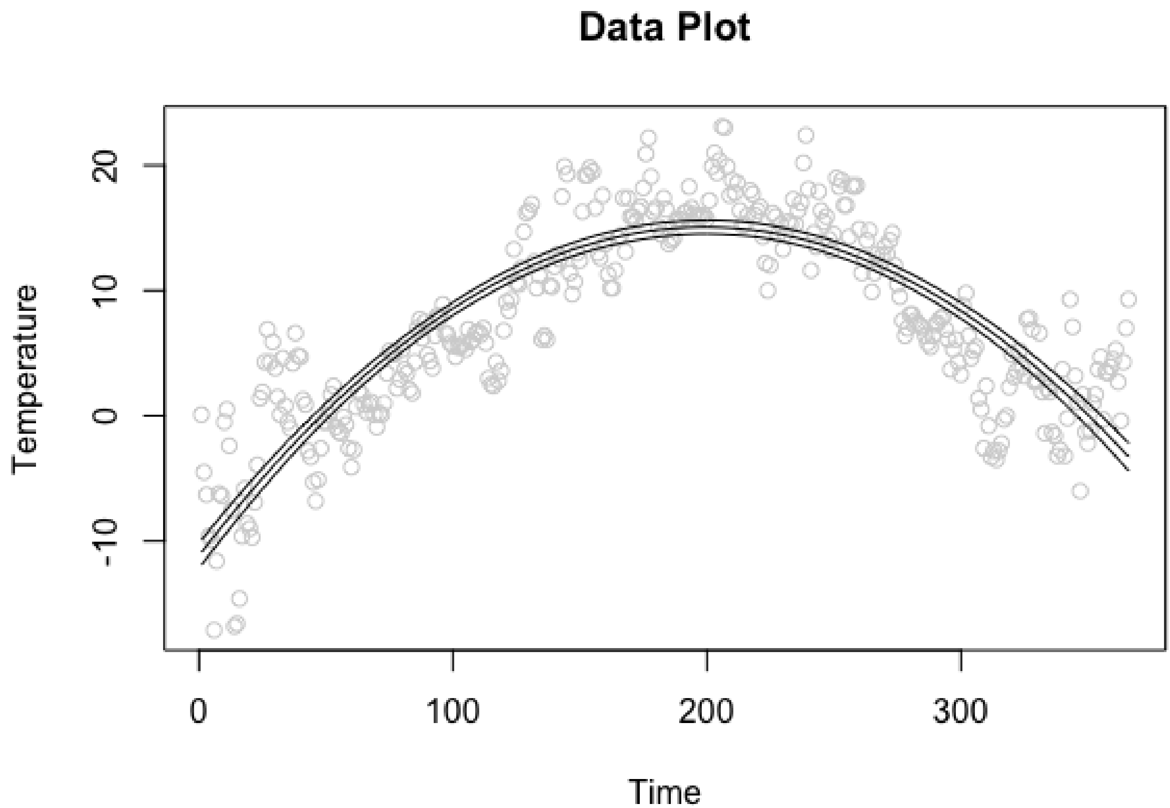

Figure 3: Plotting the marginal posteriors

1c)

Finding the maximum value gave the following result.

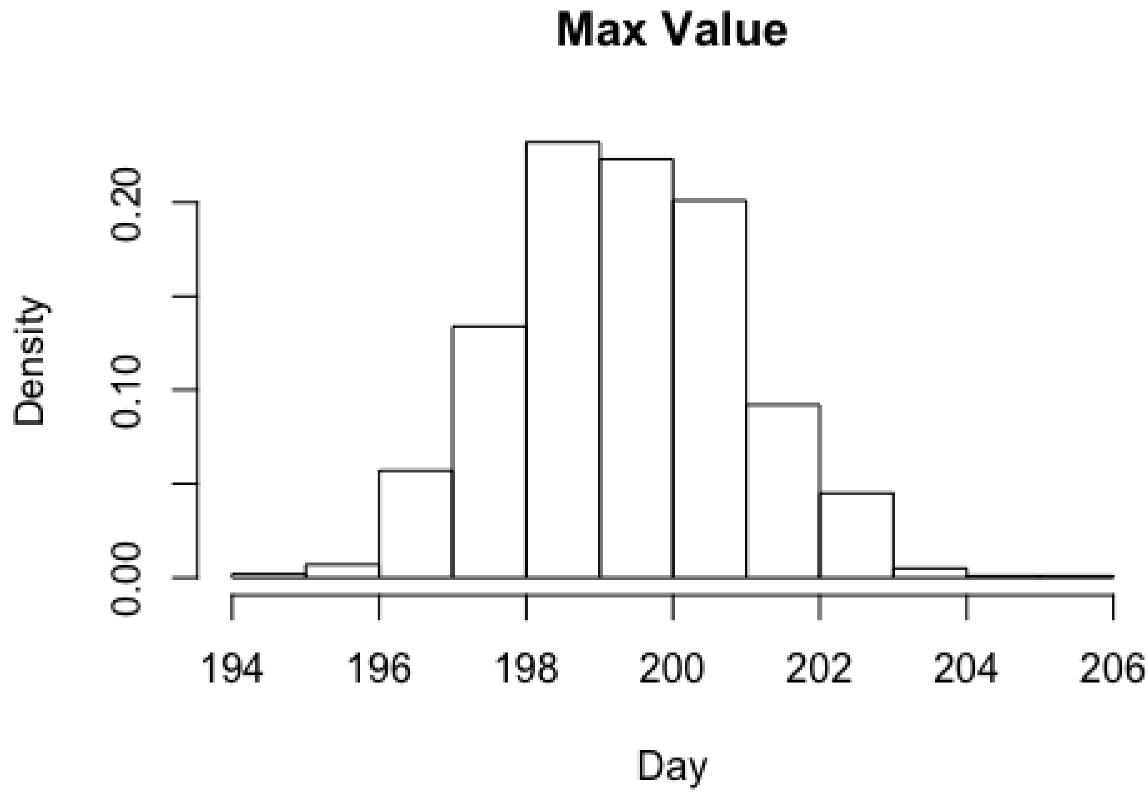

Figure 4: Plotting the marginal posteriors

1d)

Due to suspicion that the introduced variables might not be needed, we set the prior to

μ_0 = [-10, 100, -90, 0, 0, 0, 0]

For the omega prior we want low variance (high bias to avoid overfitting), thus high diagonal values in omega_0. We set the same prior values for the first three parameters as before. The rest of the values are set to zero since we can't say anything about the covariance between the new variables.

$\omega_0 =$

$$\begin{pmatrix} 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 100 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 100 \end{pmatrix}$$

1 - Code

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

######## A ########
I <- diag(3)
mu <- c(-10, 100, -90)
omega <- 10 * I
v <- 4
s2 <- 8
x <- seq(0, 10, by=0.01)


# Inverse chisquare from lab1
draw_sigma <- function(v, s2) {
  X_draw <- rchisq(1, v)
  return (v * s2 / X_draw)
}

# Given temperature regression function
temp <- function(beta, time) {
  return (beta[1] + beta[2] * time + beta[3] * (time ^ 2)) + rnorm(0,1)
}

par(mfrow=c(3,3))

plot(data$time, data$temp)

for (i in 1:8) {
  sigma_sq = draw_sigma(v, s2)
  beta = rmvnorm(1, mu, sigma_sq*solve(omega))
  y = temp(beta[1,], data$time)
  plot(data$time, y)
}

dev.off()


######## B ########
n_draws <- 1000
ones <- c(rep(1, length(data$time)))
x2 <- data$time^2
X <- cbind(ones, data$time, x2)
beta_hat <- solve(t(X)%*%X)%*%t(X)%*%data$temp
omega_n <- t(X) %*% X + omega
mu_n <- solve(omega_n) %*% (t(X) %*% X %*% beta_hat + omega %*% mu)

v_n <- v + length(data$time)
s2_n <- (v*s2 + (t(data$temp)%*%data$temp + t(mu)%*%omega%*%mu - t(mu_n)%*%omega_n%*%mu_n))/v_n

sigma_post = draw_sigma(v_n, s2_n)
```

```r
beta_post = rmvnorm(n_draws, mu_n, solve(omega_n)*sigma_post[1])

Y = matrix(nrow=n_draws, ncol=366)
for (i in 1:n_draws) {
  Y[i,] = temp(beta_post[i,], data$time)
}

y_med = c()
y_up = c()
y_low = c()

for (i in 1:366) {
  y_med = c(y_med, median(Y[,i]))
  y_low = c(y_low, quantile(Y[,i], 0.025))
  y_up = c(y_up, quantile(Y[,i], 0.975))
}

par(mfrow=c(1,3))
hist(beta_post[,1], freq = FALSE, xlab = "Beta 0")
hist(beta_post[,2], freq = FALSE, xlab = "Beta 1")
hist(beta_post[,3], freq = FALSE, xlab = "Beta 2")
dev.off()

plot(data$temp, type='p', col='lightgray')
lines(y_med, type='l')
lines(y_low, type='l')
lines(y_up, type='l')

######## C ########

x_max = - 366 * beta_post[,2]/(2 * beta_post[,3])
hist(x_max)

######## D ########

# Prior
# Mu_0 = [...mu, 0, 0, 0, 0] due to suspicion that the introduced variables might not be needed
# Omega_0, want low variance (high bias to avoid overfitting) => High diagonal values in omega_0for the

new_mu = c(mu_n, rep(0, 5))
new_omega <- 100 * diag(8)
new_omega[1:3,1:3] = omega_n
```

2 - Code