Exercise 8 - Introduction to Bayesian Inference

Bosse Behrens, st.id 12347333

2024W

Task 1

part 1

As we know from the lecture, to get the unknown proportion θ with observed positive cases Y the likelihood is the kernel of a $Beta(\alpha, \beta)$ distribution with y the observed positive cases, n the total observations and $\alpha = y + 1$, $\beta = n - y + 1$. We add 1 for regularization. As we have for the German study 4 positive out of 4068 total cases, we obtain the Beta prior distribution $Beta(4 + 1, 4068 - 4 + 1) = Beta(5, 4065)$. We now as specified reweigh the parameters by $\frac{1}{10}$ and obtain

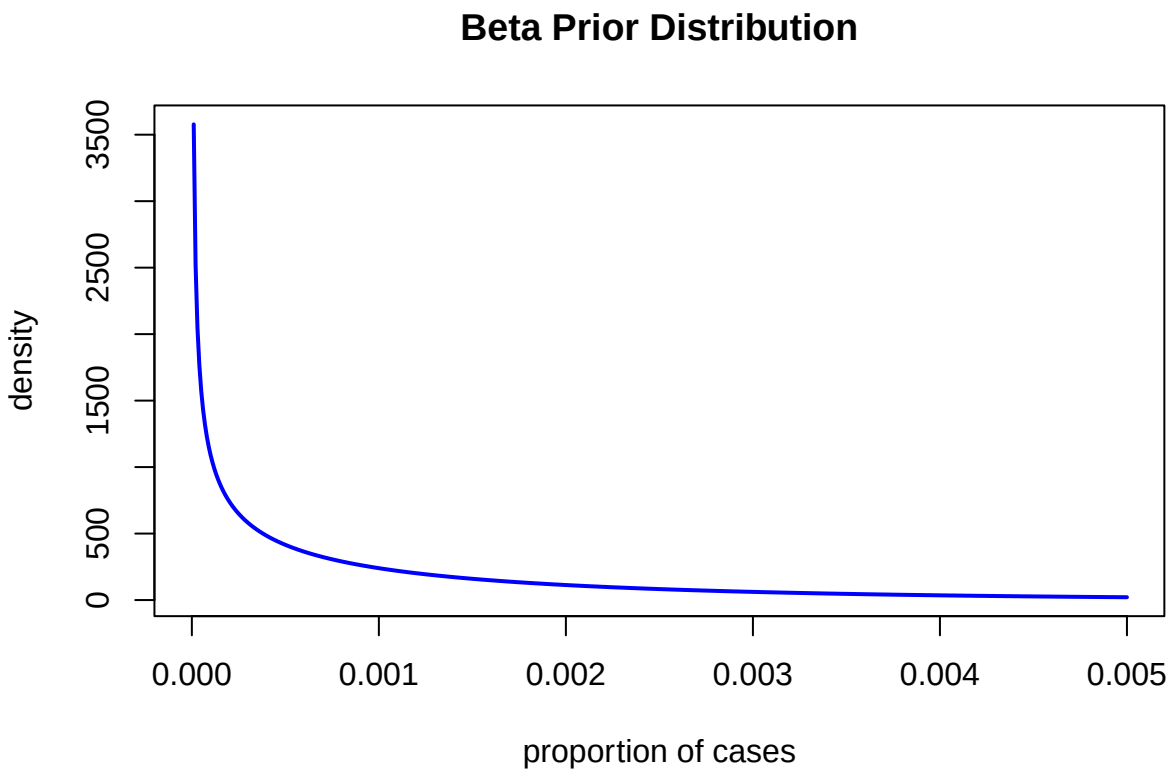
$$Beta(\frac{1}{2}, \frac{4065}{10})$$

We can also plot the density. For this we define a sequence of values.

```
sequence <- seq(0, 0.005, length.out = 500)

prior_d <- dbeta(sequence, 1/2, 4065/10)

plot(
  sequence, prior_d,
  type = "l",
  lwd = 2,
  col="blue",
  main = "Beta Prior Distribution",
  ylab = "density",
  xlab = "proportion of cases"
)
```



part 2

From the lecture we now that with the obtained prior distribution parameters we can get the posterior distribution with y the observed positive cases and n the total number of cases for the Austrian study this time, meaning $y = 0,\ n = 1279$. The posterioir distribution is then $Beta(\alpha + y, \beta + n - y)$. This results in our case in:

$$\theta|y \sim Beta(\frac{1}{2}, \frac{16855}{10})$$

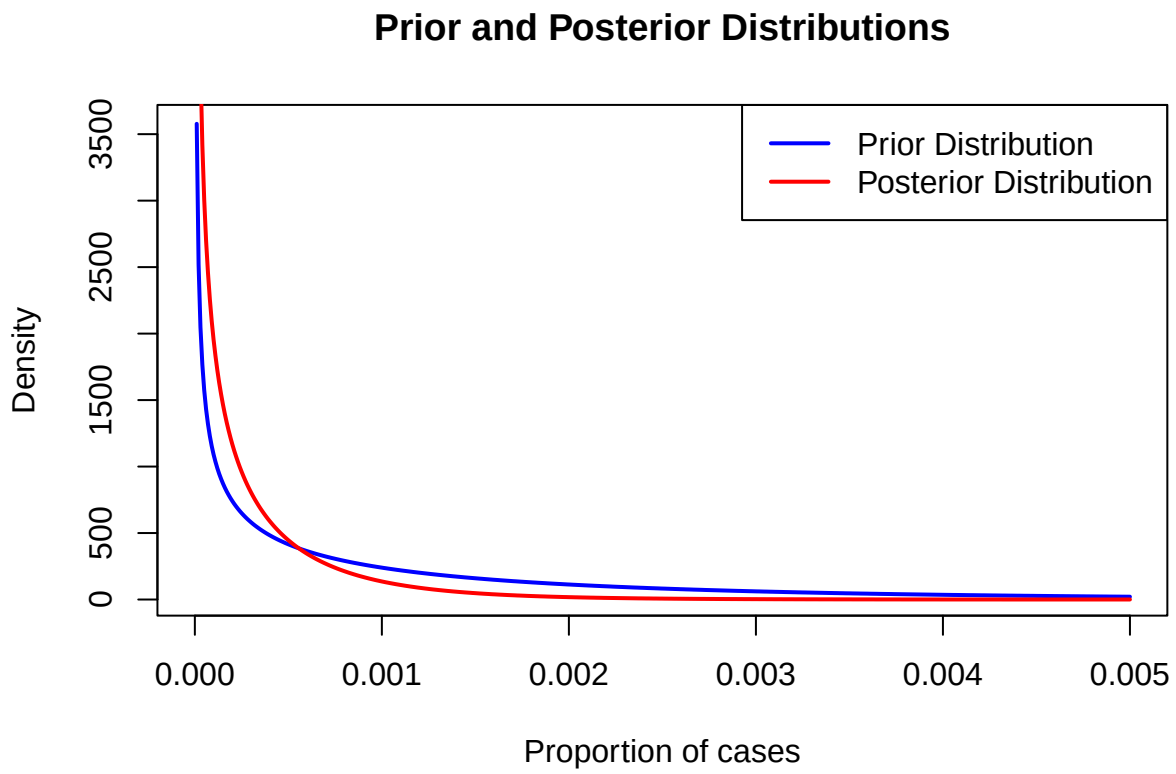
We can also plot this again togetehr with the prior distribution.

```
posterior_d <- dbeta(sequence, 1/2, 16855/10)
prior_d <- dbeta(sequence, 1/2, 4065/10)

plot(
  sequence, prior_d,
  type = "l",
  col = "blue",
  lwd = 2,
  main = "Prior and Posterior Distributions",
  xlab = "Proportion of cases",
  ylab = "Density"
)
lines(
  sequence, posterior_d,
  col = "red",
```

```
  lwd = 2
)

# Add a legend
legend(
  "topright",
  legend = c("Prior Distribution", "Posterior Distribution"),
  col = c("blue", "red"),
  lwd = 2
)
```



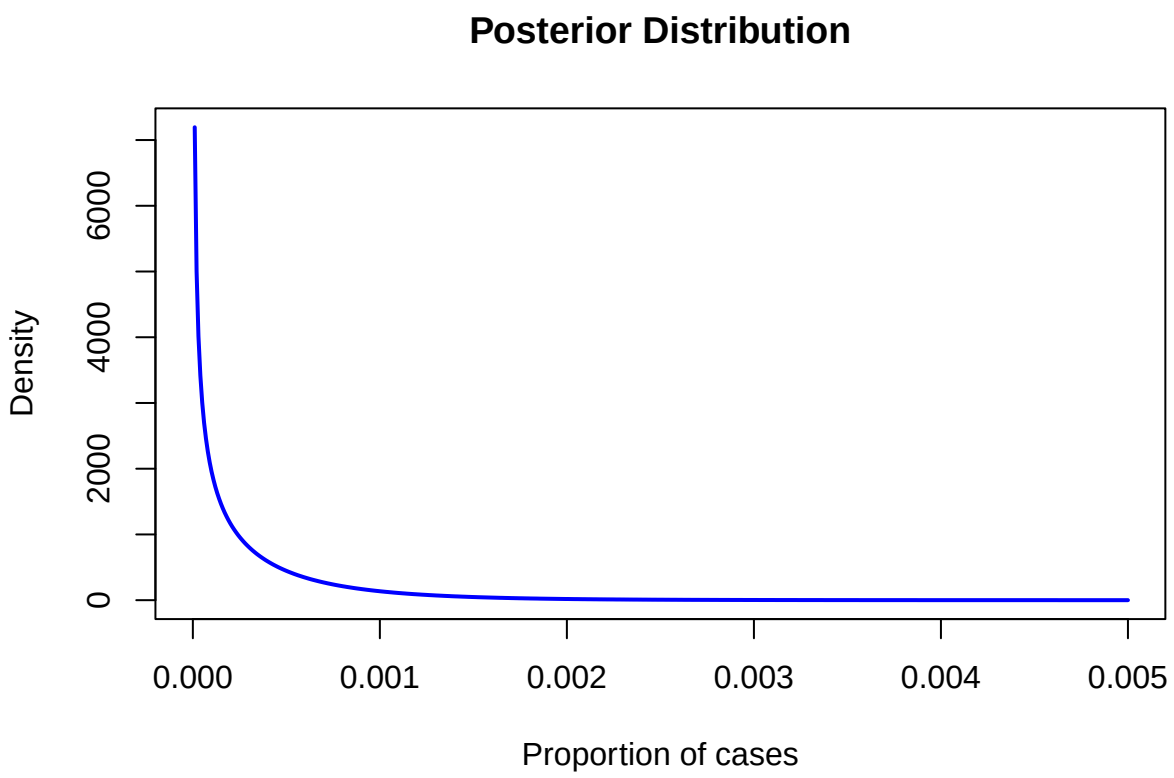
part 3

We now only plot the posterior density.

```
posterior_d <- dbeta(sequence, 1/2, 16855/10)

plot(
  sequence, posterior_d,
  type = "l",
  col = "blue",
  lwd = 2,
  main = "Posterior Distribution",
  xlab = "Proportion of cases",
```

```
  ylab = "Density"
)
```



First we want to get the point estimators. Since $\alpha < 1$ it has no mode. We tehrefore only get the median and the mean (which is calculated as $\frac{\alpha}{\alpha+\beta}$). We also get the 95% highest posterior density interval. For the HPD Interval we use the package "HDInterval" since with normal R tools we can only get the equal tailed 95% quantile interval, which can differ in skewed distribution, which is what we have.

```
library(HDInterval)

alpha <- 1/2
beta <- 16855/10

post_mean <- alpha/(alpha+beta)
post_median <- qbeta(0.5,alpha,beta)

hdi_beta <- hdi(qbeta, shape1 = alpha, shape2 = beta, credMass=0.95)

cat("Mean:",post_mean,"\n")
```

```
## Mean: 0.0002965599
```

```
cat("Median:", post_median, "\n")
```

```
## Median: 0.0001349668
```

```r
cat("95% highest posterior density interval: ",hdi_beta ,"\n")
```

```
## 95% highest posterior density interval:  1.875777e-20 0.00113908
```

part 4

There are some reason Statistik Austria has probably chosen the Bayesian approach over a simualtionbases or frequentist inference one for obtaining intervals of the prevalence. First with the Bayesian inference prior knowledge can be integrated, meaning the results from the German study. With the prior distribution it helps stabilizing estimates and can provide more realistic intervals for the prevalence. In some frequentist methods due to there being no positive cases in the Austrian study it could not really have provided meaningful confidence intervals, only undefined or degenerated ones. Due to the same reason other frequentist methods might have yielded overly conservative or uninformative intervals and also might fail sunce the likelihood alone can not provide bounds for the interval (with zero-event data). Also frequentist confidence intervals only reflect long-run frequency properties for the estimates which is not always useful in one-time analyses. Simulation-based methods like bootstrapping can become very computationally expensive especially with rare events such as the positive cases. The analytical Bayesian methods on the other hand are computationally way more efficient. Opposed to that, Bayesian methods can provide credible intervals by using also external data prior.

Task 2

part 1

For readability and comprehension's sake we will assume there is no intercept β_0 and only one coefficient β, but it would work in the same way anyway with an intercept or more coefficients included. We now want to define conjugate priors independently for coefficient and varaince. since we assume a normal distribution for the data $y \sim N(x\beta, \sigma^2)$. From the lecture we know that for the parameters of a normal distributions the conjugate prior for the mean (coefficient in our case) is also a normal distribution, e.g.:

$$\beta \sim N(m, s^2)$$

Here we have to set the parameters m and s^2. To let the prior be uninformative we could choose $m = 0$ and $s^2 = 10^8$ (or another very large value). This would result in the density to be almost flat because of the large varaince s^2 so any β values in a reasonable range are nearly equally likely. These would be uninformative parameters for the prior because β is effectively being ignored and the likelihood dominates. Other choices would be foe example $s^2 = 1$ which would mean something akin to saying I believe typical values for β are on the order of 1. A evry small value like $s^2 = 0.001$ would mean the assumption that β is very near the mean (we used 0 here, but it can be set to any desired value), which means highly informative.\ For the prior variance σ^2 we know from th lecture the rior conjugate is the Inverse Gamma distribution, e.g.:

$$\sigma^2 \sim IG(a, b)$$

To get an uninformative prior we can set a and b to very small values, for example $a = b = 0.01$. This will result in a very wide distribution and again very flat density that again means σ^2 values in a reasonable range are nearly equally likely. For the IG distribution the mean is $\frac{\beta}{}\alpha - 1$ and the mode $\frac{\beta}{}\alpha + 1$. Therefore with the right choice of parameters this can be controlled. Also especially with the scale parameter b the weight of the tails can be influenced more so somehow similar to defining how much of the mass is close to some specific point, which would make the parameter way mroe informative.

part 2

We build for the corresponding model the regression inference. to do so we obtain the theoretical posterior disitrubtion for oth parameters assuming the other one is known respectively. First we have for $\beta = (\beta_0, ..., \beta_n)$ which includes the intercept that has the prior distribution $\beta \sim N(\mu_{prior}, \Sigma_{prior}$ where μ_{prior} is the mean parameter of the prior Normal distribution and Σ_{prior} the covariance matrix as in part 1 defined (expanded into to multiple β) and the observed predictors that include 1 for the intercept, i.e. $x_i = (1, x_{i,1}, ..., x_{i,n})$. The observations of the response are y_i and X the design matrix. The likelihood function of a normal distribution assuming σ^2 is known is also gaussian:

$$p(y|\beta, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{1}{2\sigma^2}(y_i - x_i^T\beta)^2)$$

This means both likelihood and prior distribution are normal, therefore the posterior will also be normal since multiplying them will yield a normal distribution. We are not including every step here, but as a sketch we would follow the same steps as the example in the lecture: We multiply the likelihood with the prior density nad collect terms and then with a good eye the new parameters can be read off. We are here using already established formula so we do not have to do all these steps: With the usual conjugate normal-normal process we can the update the precision and get the variance or covariance-matrix and then also the mean update. The posterior precision is the sum of the prior precision and data precision from which we can derive the posterior variance matrix:

$$\Sigma_{posterior}^{-1} = \Sigma_{prior}^{-1} + \frac{1}{\Sigma^2} X^T X$$

From this we get the varaince matrix:

$$\Sigma_{posterior} = (\Sigma_{prior}^{-1} + \frac{1}{\sigma^2} X^T X)^{-1}$$

With the formula for updating the mean we get:

$$\mu_{posterior} = \Sigma_{posterior}(\Sigma_{prior}^{-1}\mu_{prior} + \frac{1}{\sigma^2} X^T y)$$

In conclusion, assuming σ^2 is known, the theoretical posterior distribution for β is then:

$$\beta|y, \sigma^2 \sim N((\Sigma_{prior}^{-1} + \frac{1}{\sigma^2} X^T X)^{-1}(\Sigma_{prior}^{-1}\mu_{prior} + \frac{1}{\sigma^2} X^T y), (\Sigma_{prior}^{-1} + \frac{1}{\sigma^2} X^T X)^{-1})$$

Now we also want to get the posterior distribution for σ^2, assuming β is known. The likelihood is again the same:

$$p(y|\beta, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{1}{2\sigma^2}(y_i - x_i^T\beta)^2)$$

Using $\propto$ to ignore constant factors this is proportional to

$$\propto (\sigma^2)^{-\frac{n}{2}} exp(-\frac{1}{2\sigma^2}(y_i - x_i^T\beta)^2)$$

Now as explained in part 1 the prior distribution for σ^2 is an Inverse-gamma distribution $\sigma^2 \sim IG(a_{prior}, b_{prior})$. The density of the IG is:

$$p(\sigma^2) = \frac{b_{prior}^{a_{prior}}}{\Gamma(a_{prior})} (\sigma^2)^{-(a+1)} exp(-\frac{b_{prior}}{\sigma^2})$$

This is again proportional to

$$\propto (\sigma^2)^{-(a+1)} exp(-\frac{b_{prior}}{\sigma^2})$$

Now we multiply the likelihood and the prior density function, collect terms by adding up the exponents of σ^2 and in the *exp*. We then obtain:

$$p(\sigma^2|y,\beta) \propto (\sigma^2)^{-(a+\frac{n}{2}+1)} exp(-\frac{b_{prior} + \frac{1}{2}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2}{\sigma^2})$$

We can observe that this is again (proportional to) an Inverse-Gamma density with updated parameters:

$$a_{posterior} = a_{prior} + \frac{n}{2}$$

$$b_{posterior} = b_{prior} + \frac{1}{2}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2$$

In conclusion, assuming β is known, for σ^2 we get the theoretical posterior distribution

$$\sigma^2|y,\beta \sim IG(a_{prior} + \frac{n}{2}, b_{prior} + \frac{1}{2}\sum_{i=1}^{n}(y_i - x_i^T\beta)^2)$$

part 3

For *beta* the point estimators, e.g. mean and mode are easy to get. The posterior distribution is a normal distribution so both are the same and obviously $posterior\ mean = posterior\ mode = \mu_{posterior}$. Due to the same reason (normal distribution, which is symmetrical) the 95% HPD Interval is the same as the 95% symmetric equal-tailed interval, so using the usual 1.96 SE rule to get the 95% interval for each coefficient β_j the interval is $[\mu_{posterior,j} - 1.96\sqrt{(\Sigma_{posterior})_{j,j}}, \mu_{posterior,j} + 1.96\sqrt{(\Sigma_{posterior})_{j,j}}]$. For σ^2 which has an Inverse-Gamma $IG(a_{posterior}, b_{posterior})$ posterior distribution the mean is $\frac{b_{posterior}}{a_{posterior}-1}$ and the mode $\frac{b_{posterior}}{a_{posterior}+1}$. The 95% HPD Interval is for this distribution family not really possible to be analytically obtained. We can use a package as in Task 1 in R to compute it numerically, or if an analytical solution is desired simply take the equal-tailed 95% interval $[quantile_{IG(a_{posterior},b_{posterior})}(0.025), quantile_{IG(a_{posterior},b_{posterior})}(0.975)]$ again, though this might be inaccurate depending on the parameters of the IG distribution.

part 4

Now we test our results on the Auto data from the ISLR package, similar to what we already did with a frequentist approach in exercise 6. First we load the data and get the frequentist mdoel as in exercise 6 again and obtain the estimates and confidence intervals.

```
library(ISLR)
data("Auto")

freq_fit <- lm(mpg ~ horsepower, data = Auto)
summary(freq_fit)
```

```
##
## Call:
## lm(formula = mpg ~ horsepower, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.5710  -3.2592  -0.3435   2.7630  16.9240
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 39.935861   0.717499   55.66   <2e-16 ***
## horsepower  -0.157845   0.006446  -24.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.906 on 390 degrees of freedom
## Multiple R-squared:  0.6059, Adjusted R-squared:  0.6049
## F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

```
confint(freq_fit)
```

```
##                 2.5 %     97.5 %
## (Intercept) 38.525212 41.3465103
## horsepower  -0.170517 -0.1451725
```

Now we fit a Bayesian linear model. For this we us the package MCMCpack which uses exactly what we did, e.g. a normal prior for the coefficient and an Inverse-Gamma for the covariance matrix. As specified we use uninformative parameters for the priors. In the function we will call we therefore choose b0=c(0,0) for the mean for the normal prior, B0=diag(0.0001,2) for the precision, which translates to a huge covariance when inversed, then co=d0=0.001 which are the very small shape and scale parameters for the Inverse-gamma prior so it is also very flat.

```
library(MCMCpack)
```

```
## Lade nötiges Paket: coda

## Lade nötiges Paket: MASS

## ##
## ## Markov Chain Monte Carlo Package (MCMCpack)

## ## Copyright (C) 2003-2025 Andrew D. Martin, Kevin M. Quinn, and Jong Hee Park

## ##
## ## Support provided by the U.S. National Science Foundation

## ## (Grants SES-0350646 and SES-0350613)
## ##
```

```
mcmc_fit <- MCMCregress(
  mpg ~ horsepower,
  data = Auto,
  b0   = c(0, 0),
  B0   = diag(0.000001, 2),
  c0   = 0.001,
  d0   = 0.001,
  burnin  = 1000,
  mcmc    = 5000,
  thin    = 5,
  verbose = 0
)

summary(mcmc_fit)
```

```
##
## Iterations = 1001:5996
## Thinning interval = 5
## Number of chains = 1
## Sample size per chain = 1000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##                Mean       SD  Naive SE Time-series SE
## (Intercept) 39.9247 0.711833 0.0225101      0.0237659
## horsepower  -0.1577 0.006422 0.0002031      0.0002186
## sigma2      24.0905 1.723069 0.0544882      0.0578835
##
## 2. Quantiles for each variable:
##
##                2.5%     25%     50%     75%  97.5%
## (Intercept) 38.5434 39.4483 39.9120 40.3906 41.333
## horsepower  -0.1701 -0.1621 -0.1578 -0.1535 -0.145
## sigma2      20.8914 22.9287 24.0555 25.1159 27.784
```

Now we aggregate the results for the mean/estimate of the coefficients and the HDP Intervals. Here for the HDP Intervals ita gain uses the normal 95% credible interval with the quantiles, but it should be a good enough approxiamtion. We can only get the results for the bcoefficients since in the frequentist approach ther exists no distribution for the $sigma^2$.

```
intercept_res <- c(39.935861, 39.9247, 38.525212, 38.5434, 41.3465103, 41.333)
horse_res <- c(-0.157845, -0.1577, -0.170517, -0.1701, -0.1451725, -0.145)

results <-rbind(intercept_res, horse_res)
rownames(results) <- c("intercept", "horsepower coef")
colnames(results) <- c("OLS estimate", "Bayesian post mean", "OLS lower 95%",
                       "Bay. low 95% HDPI", "OLS upper 95%", "Bay. upper 95% HDPI")
results
```

```
##                 OLS estimate Bayesian post mean OLS lower 95% Bay. low 95% HDPI
## intercept          39.935861            39.9247     38.525212           38.5434
## horsepower coef    -0.157845            -0.1577     -0.170517           -0.1701
##                 OLS upper 95% Bay. upper 95% HDPI
## intercept          41.3465103              41.333
## horsepower coef    -0.1451725              -0.145
```

We also want to do the same again but use the OLS results to create way more informative priors. To do so we choose the parameters for the priors so that the values such that the mean is more similar to the OLS results and the varaince is small.

```
mcmc_fit_informative <- MCMCregress(
  mpg ~ horsepower,
  data = Auto,
  b0   = c(40, -0.15),
  B0   = diag(c(10, 10)),
  c0   = 5,
  d0   = 10,
```

```
  burnin  = 1000,
  mcmc    = 5000,
  thin    = 5,
  verbose = 0
)

summary(mcmc_fit_informative)
```

```
##
## Iterations = 1001:5996
## Thinning interval = 5
## Number of chains = 1
## Sample size per chain = 1000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##                Mean      SD  Naive SE Time-series SE
## (Intercept) 39.9817 0.27566 0.0087170      0.0087170
## horsepower  -0.1583 0.00317 0.0001002      0.0001002
## sigma2      23.8540 1.67325 0.0529128      0.0554085
##
## 2. Quantiles for each variable:
##
##                2.5%     25%     50%     75%   97.5%
## (Intercept) 39.4501 39.7764 39.9783 40.1730 40.5190
## horsepower  -0.1646 -0.1603 -0.1583 -0.1562 -0.1518
## sigma2      20.8451 22.6828 23.8009 24.8815 27.4324
```

```
intercept_res <- c(39.935861, 39.9817, 38.525212, 39.4501, 41.3465103, 40.5190)
horse_res <- c(-0.157845, -0.1583, -0.170517, -0.1646, -0.1451725, -0.1518)

results <-rbind(intercept_res, horse_res)
rownames(results) <- c("intercept", "horsepower coef")
colnames(results) <- c("OLS estimate", "Bayesian post mean", "OLS lower 95%",
                       "Bay. low 95% HDPI", "OLS upper 95%", "Bay. upper 95% HDPI")
results
```

```
##                 OLS estimate Bayesian post mean OLS lower 95% Bay. low 95% HDPI
## intercept          39.935861            39.9817     38.525212           39.4501
## horsepower coef    -0.157845            -0.1583     -0.170517           -0.1646
##                 OLS upper 95% Bay. upper 95% HDPI
## intercept          41.3465103             40.5190
## horsepower coef    -0.1451725             -0.1518
```

As we can observe for the informative Bayes approach the values are very similar to the OLS regression. This might also be due to the not too small data size but mainly since we chose parameters as uninformative it makes sense that it "lets the data speak for itself" and by that aligns more with the OLS results. Using priors with more informative parameters the posterior is pulled more to the prior mean as we can also observe and the HDP Intervals are narrower and mroe precise. Of course if we would have chosen priors that do not match with the data instead of close to the OLS estimates, the posterior estimates would have been off and shifted more towards the priors. For large data both frequentist and Bayesian intervals often align anyway, but as we have seen in Task 1 this might not always be the case and Bayesian methods can provide more flexibility by adding external knowledge to the data.