Assignment 5b

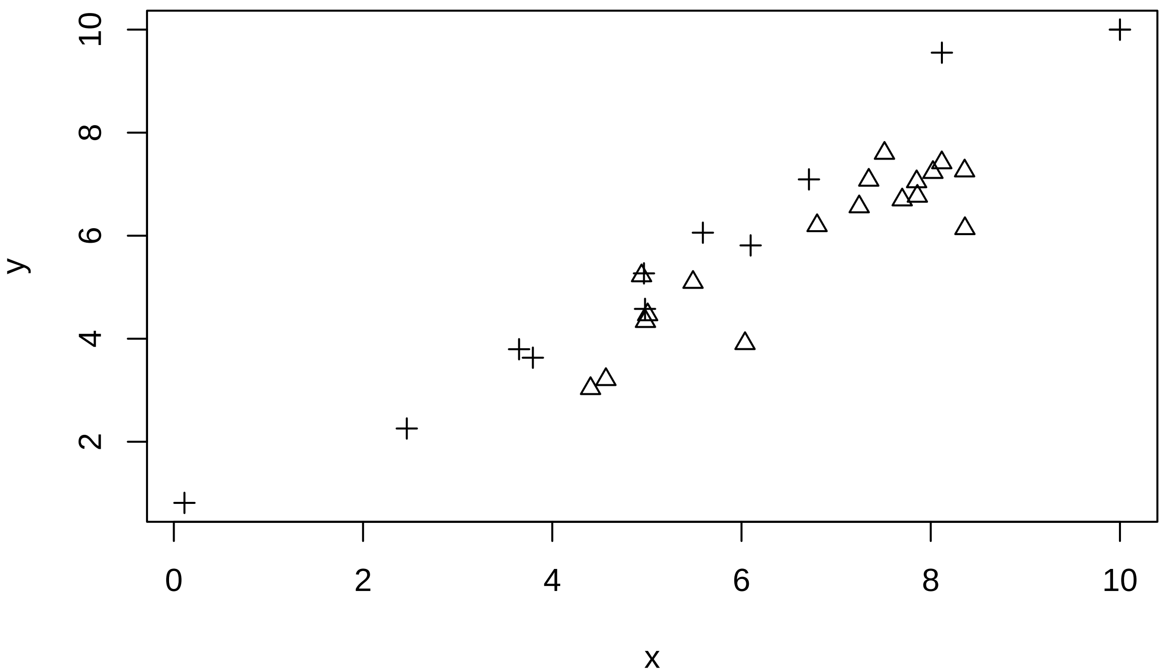
Satvik Saha

2024-10-08

Answer 1

```
library(rstanarm)

n <- 30
x <- rnorm(n, mean = 6, sd = 2)
x <- pmax(pmin(x, 10), 0)
z <- sample(c(0, 1), n, replace = TRUE)
y <- x + 1.5 * (z - 0.5) + 0.5 * rnorm(n)
y <- pmax(pmin(y, 10), 0)
df <- data.frame(x = x, y = y, z = z)
plot(x, y, pch = z + 2)
```




```
prior.x.sd <- 2.5 * sd(y) / sd(x)
```

(a)

```
fit.a <- stan_glm(
  y ~ x + z,
  data = df,
  refresh = 0
)
```

```r
fit.a
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      y ~ x + z
##  observations: 30
##  predictors:   3
## ------
##             Median MAD_SD
## (Intercept) -0.8    0.4  
## x            1.0    0.1  
## z            1.0    0.2  
## 
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 0.6    0.1   
## 
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```r
prior_summary(fit.a)
```

```
## Priors for model 'fit.a' 
## ------
## Intercept (after predictors centered)
##   Specified prior:
##     ~ normal(location = 5.8, scale = 2.5)
##   Adjusted prior:
##     ~ normal(location = 5.8, scale = 5.4)
## 
## Coefficients
##   Specified prior:
##     ~ normal(location = [0,0], scale = [2.5,2.5])
##   Adjusted prior:
##     ~ normal(location = [0,0], scale = [ 2.49,10.93])
## 
## Auxiliary (sigma)
##   Specified prior:
##     ~ exponential(rate = 1)
##   Adjusted prior:
##     ~ exponential(rate = 0.46)
## ------
## See help('prior_summary.stanreg') for more details
```

(b)

```r
fit.b <- stan_glm(
  y ~ x + z,
  data = df,
  prior = c(normal(c(0, 0), c(prior.x.sd, 10^3))),
  refresh = 0
)
fit.b
```

```
## stan_glm
```

```
##  family:       gaussian [identity]
##  formula:      y ~ x + z
##  observations: 30
##  predictors:   3
## ------
##             Median MAD_SD
## (Intercept) -0.8    0.4
## x            1.0    0.1
## z            1.0    0.2
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 0.6    0.1
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```r
prior_summary(fit.b)
```

```
## Priors for model 'fit.b'
## ------
## Intercept (after predictors centered)
##   Specified prior:
##     ~ normal(location = 5.8, scale = 2.5)
##   Adjusted prior:
##     ~ normal(location = 5.8, scale = 5.4)
##
## Coefficients
##  ~ normal(location = [0,0], scale = [   2.5,1000.0])
##
## Auxiliary (sigma)
##   Specified prior:
##     ~ exponential(rate = 1)
##   Adjusted prior:
##     ~ exponential(rate = 0.46)
## ------
## See help('prior_summary.stanreg') for more details
```

(c) One appropriate prior for z is the $N(0, 1.5)$.

```r
fit.c <- stan_glm(
  y ~ x + z,
  data = df,
  prior = c(normal(c(0, 0), c(prior.x.sd, 1.5))),
  refresh = 0
)
fit.c
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      y ~ x + z
##  observations: 30
##  predictors:   3
## ------
##             Median MAD_SD
```

```
## (Intercept) -0.8    0.4
## x            1.0    0.1
## z            1.0    0.2
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 0.6    0.1
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```r
prior_summary(fit.c)
```

```
## Priors for model 'fit.c'
## ------
## Intercept (after predictors centered)
##   Specified prior:
##     ~ normal(location = 5.8, scale = 2.5)
##   Adjusted prior:
##     ~ normal(location = 5.8, scale = 5.4)
##
## Coefficients
##   ~ normal(location = [0,0], scale = [2.5,1.5])
##
## Auxiliary (sigma)
##   Specified prior:
##     ~ exponential(rate = 1)
##   Adjusted prior:
##     ~ exponential(rate = 0.46)
## ------
## See help('prior_summary.stanreg') for more details
```

Answer 2

The point on the extreme right (on the x scale) has most influence, being the one furthest away from the mean of the x data.

Research homework assignment

```r
fit.normal <- function(n, M, S) {
  y <- rnorm(n, mean = M, sd = S)
  stan_glm(
    y ~ 1,
    prior_intercept = normal(0, 1),
    prior_aux = NULL,
    refresh = 0
  )
}
```

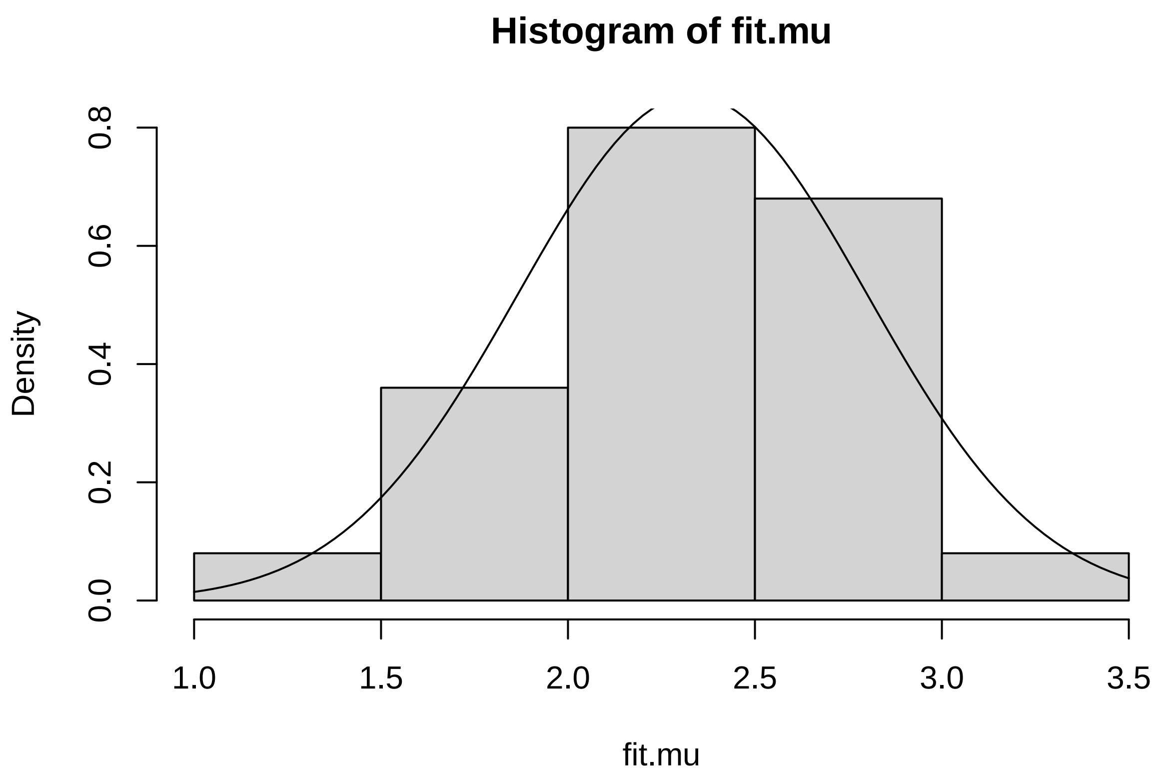
For $M = 5$, $S = 10$, note that the prior and data will have similar information about μ when $S^2/n \approx \sigma^2_{\text{prior}}$, i.e. we set $n \approx 100$.

```r
fit.normal(100, 5, 10)
```

```
## stan_glm
```

```
##  family:       gaussian [identity]
##  formula:      y ~ 1
##  observations: 100
##  predictors:   1
## ------
##             Median MAD_SD
## (Intercept) 2.9    0.8
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 10.4    0.8
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```r
fit.mu <- replicate(50, coef(fit.normal(100, 5, 10)))
hist(fit.mu, probability = TRUE)
curve(dnorm(x, mean = mean(fit.mu), sd = sd(fit.mu)), add = TRUE)
```




```r
mean(fit.mu)
```

```
## [1] 2.332914
```

```r
sd(fit.mu)
```

```
## [1] 0.4670078
```

Note that the fitted values of μ seem to follow a normal distribution with a mean in the middle of the prior and model means, i.e. between 0 and 5.

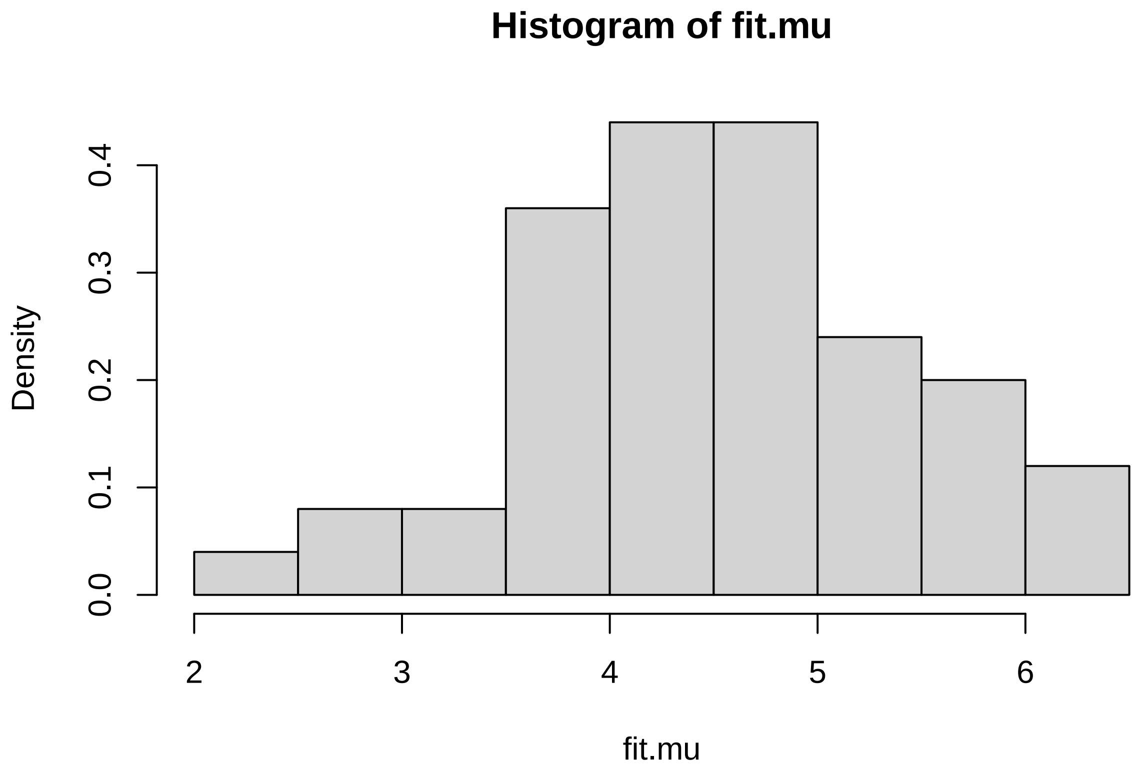
We repeat the same for the second model; we keep the approximation $n = 100$.

```
fit.t <- function(n, M, S) {
  y <- rnorm(n, mean = M, sd = S)
  stan_glm(
    y ~ 1,
    prior_intercept = student_t(df = 1, location = 0, scale = 1),
    prior_aux = NULL,
    refresh = 0
  )
}
```

```
fit.t(100, 5, 10)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      y ~ 1
##  observations: 100
##  predictors:   1
## ------
##             Median MAD_SD
## (Intercept) 4.8    1.0
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 10.1    0.7
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```
fit.mu <- replicate(50, coef(fit.t(100, 5, 10)))
hist(fit.mu, probability = TRUE)
```



Here, the fitted values of μ are much more unstable. This may be explained by the fact that the $t_1(0,1)$ prior on μ has fairly heavy tails, making prior draws of μ erratic. This, the shrinking effect of the prior on the estimate of μ seen in the first model is not so apparent in this one.