Assignment 4b

Satvik Saha

2024-10-01

Answer 1

(a)

```
library(rstanarm)

gen.fit.linear <- function(n, a, b, sigma, x.lo, x.hi) {
  x <- runif(n, min = x.lo, max = x.hi)
  y <- a + b*x + sigma * rnorm(n)
  df <- data.frame(x, y)

  fit.y <- stan_glm(y ~ x, data = df, refresh = 0)
  a.hat <- coef(fit.y)[1]
  b.hat <- coef(fit.y)[2]
  a.se <- se(fit.y)[1]
  b.se <- se(fit.y)[2]

  return(list(
    fit = fit.y,
    slope.1.se = abs(b.hat - b) < b.se
  ))
}
```

(b)

```
gen.fit.linear(
  n = 50,
  a = 10,
  b = -20,
  sigma = 30,
  x.lo = 0,
  x.hi = 1
)
```

```
## $fit
## stan_glm
##  family:       gaussian [identity]
##  formula:      y ~ x
##  observations: 50
##  predictors:   2
## ------
##             Median MAD_SD
## (Intercept)  20.6    8.0
## x           -28.2   15.6
```

```
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 28.7    2.9
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
##
## $slope.1.se
##    x
## TRUE
```

Answer 2

(a) The regression model $y_i = a + \epsilon_i$ involves minimizing the residual sum of squares $\text{RSS} = \sum_{i=1}^{n}(y_i - a)^2$, which is solved by $\hat{a} = \frac{1}{n}\sum_{i=1}^{n} y_i = \bar{y}$. This is easily seen via

$$\text{RSS} = \sum_{i=1}^{n}(y_i - \bar{y})^2 + n(\bar{y} - a)^2.$$

```
fit.scores <- function(scores) {
  fit.y <- stan_glm(test_scores ~ 1, data = scores, refresh = 0)
  a.hat <- coef(fit.y)[1]

  return(list(
    fit = fit.y,
    a.hat = a.hat
  ))
}
```

```
ugrad <- data.frame(
  level = 1,
  test_scores = 50 + 10 * rnorm(10)
)
grad <- data.frame(
  level = 2,
  test_scores = 60 + 15 * rnorm(10)
)
scores <- rbind(ugrad, grad)
head(scores[sample(nrow(scores)), ])
```

```
##    level test_scores
## 18     2    60.97827
## 6      1    43.77190
## 7      1    42.12011
## 19     2    60.57420
## 8      1    42.86010
## 4      1    37.45056
```

```
mean(scores$test_scores)
```

```
## [1] 51.49435
```

```
fit.scores(scores)
```

```
## $fit
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      test_scores ~ 1
##  observations: 20
##  predictors:   1
## ------
##             Median MAD_SD
## (Intercept) 51.5    3.1
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 14.3    2.3
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
##
## $a.hat
## (Intercept)
##    51.52437
```

Of course, `stan_glm` goes beyond simple linear regression, with priors and regularization, but the point stands.