Lecture 7 Exercises

Isabel Fulcher

8/14/2018

Install packages

```
library(matrixStats)
library(knitr)
library(tidyverse)
library(reshape2)
library(MASS)
```

Load in the infants dataset from Lecture 5. We are again interested in the relationship between birthweigh Y, smoking X_1, and mother's weight X_2.

```
load("infants.dat")
```

Exercise 1

Recall, the likelihood for a linear model where we assume $\epsilon_i \sim N(0, \sigma^2)$ and observe $X_1, ...X_n$ is,

$$\mathcal{L}(\beta_0, \beta_1, \beta_2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} exp(\frac{-1}{2\sigma^2}(Y_i - (\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2}))^2)$$

The log-likelihood can then be written as,

$$\ell(\beta_0, \beta_1, \beta_2) = \sum_{i=1}^{n} -\log(\sqrt{2\pi}\sigma) - \frac{1}{2\sigma^2}(Y_i - (\beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2}))^2$$

1. Write a function for that calculates the negative log-likelihood and takes in values for Y, X_1, and X_2, which are all vectors of length n, and a vector for the unknown parameters, i.e. $\{\beta_0, \beta_1, \beta_2, \sigma^2\}$.

```
loglik <- function(par,Y,X){
  beta <- par[1:3]
  sigma2 <- par[4]
  l <- sum(log(sqrt(2*pi*sigma2)) + (1/(2*sigma2))*(Y-X%*%beta)^2)
  return(l)
}
```

2. Use the optim() function to find the the MLE of $\boldsymbol{\beta}$ when the outcome Y is birthweight, $X1$ is smoking, and $X2$ mother's weight. NOTE: you would not typically do this in practice because there is a closed-form solution (recall OLS estimates!). This is just for illustration.

```
X <- cbind(1,infants$smoker,infants$gained)
Y <- infants$weight

fit.optim <- optim(runif(4,0,1),loglik,Y=Y,X=X,method="BFGS")

beta.optim <- fit.optim$par[1:3]
beta.optim
```

```
## [1]  3.04725674 -0.23443554  0.00849865
```

3. Calculate the OLS estimate for $\boldsymbol{\beta}$ using R and the analytical expression give in Lecture 5. How does this compare to the above?

```
#Using lm() function
fit.ols1 <- lm(weight ~ smoker + gained, data=infants)
beta.ols1 <- coefficients(fit.ols1)

#Using actual values
beta.ols2 <- t(solve(crossprod(X))%*%t(X)%*%Y)

#Compare all methods
beta.ols1 ; beta.ols2 ; beta.optim
```

```
##  (Intercept)       smoker       gained
##  3.047136188 -0.234406015  0.008500909

##          [,1]      [,2]        [,3]
## [1,] 3.047136 -0.234406 0.008500909

## [1]  3.04725674 -0.23443554  0.00849865
```

Exercise 2

A logistic regression model is given by,

$$\text{logit}(Pr(Y = 1|X_1, X_2)) = \alpha_0 + \alpha_1 X_1 + \alpha_2 X_2 \implies Pr(Y = 1|X_1, X_2) = \text{expit}(\alpha_0 + \alpha_1 X_1 + \alpha_2 X_2)$$

The likelihood for a logistic model where we observe $X_1, ...X_n$ is given by,

$$\mathcal{L}(\alpha_0, \alpha_1, \alpha_2) = \prod_{i=1}^{n} Pr(Y_i = 1|X_{i1}, X_{i2})^{Y_i}(1 - Pr(Y_i = 1|X_{i1}, X_{i2}))^{1-Y_i}$$

The log-likelihood can be written as,

$$\ell(\alpha_0, \alpha_1, \alpha_2) = \sum_{i=1}^{n} Y_i(\alpha_0 + \alpha_1 X_{i1} + \alpha_2 X_{i2}) - \log[1 + \exp(\alpha_0 + \alpha_1 X_{i1} + \alpha_2 X_{i2})]$$

1. Write a function for that calculates the negative log-likelihood and takes in values for Y, X_1, and X_2, which are all vectors of length n, and $\boldsymbol{\alpha}$.

```
# thanks Jemar!
negloglik = function(alpha, X, Y) {
  return(-sum(
    Y*(X%*%alpha)
    -log(1 + exp(X %*% alpha))
  )
  )
}
```

2. Use the optim() function to find the the MLE of $\boldsymbol{\alpha}$ in this dataset.

```r
X <- cbind(1,infants$smoker,infants$gained)
infants %>% mutate(weight.binary = ifelse(weight <= 2.5,1,0)) -> infants
Y <- infants$weight.binary

fit.optim <- optim(runif(3,0,1),negloglik,Y=Y,X=X,method="BFGS")
beta.optim <- fit.optim$par
beta.optim
```

```
## [1] -1.70856498  0.62727086 -0.02521087
```

3. Check your answer using the built-in R function for logistic regression (and estimation of parameters in GLMs in general).

```r
fit.logit <- glm(weight.binary ~ smoker + gained, data=infants, family=binomial)

coefficients(fit.logit) ; beta.optim
```

```
## (Intercept)      smoker      gained
## -1.70928460  0.62722972 -0.02518894
```

```
## [1] -1.70856498  0.62727086 -0.02521087
```

The parameter values differ slightly. This is because the optimization function used in the glm() function is different than the one used in optim(). Specifically, the glm() function uses Fisher scoring, which you will likely learn more about in your Methods course.