HW4

Nathan Bick

2022-10-04

```
options(scipen=999)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(stringr)
```

Problem 1

Download the datafile `economic_data.txt`. See the datafile for details regarding the column values and meaning, note that the datafile has six covariates ($A_1, A_2, ..., A_6$). Consider the multiple linear model, and find the best fit for this model.

Response

Economic and unemployment data were recorded.

There are 16 rows of data. The data include:

- I, the index;
- A1, the percentage price deflation;
- A2, the GNP in millions of dollars;
- A3, the number of unemployed;
- A4, the number of people employed by the military;
- A5, the number of people over 14;
- A6, the year
- Y, the number of people employed.

A3-A5 are in units of thousands

We seek a model of the form:

$$Y = \beta_0 + A_1\beta_1 + A_2\beta_2 + A_3\beta_3 + A_4\beta_4 + A_5\beta_5 + A_6\beta_6$$

.

First we read in the data and do some data cleaning. This is followed by two portions

```r
# read in the data
economic_data <- read.csv(
  "~/Documents/Graduate School/MATH504 - Numerical Methods/Homework/HW4/economic_data.txt",
  sep = "\t",
  skip=35,
  header = FALSE)

# we see that there are some issues with the white spaces, so we do data cleaning.
# this outputs a dataframe which can be used for the R native linear model.
economic_data_clean <- economic_data %>%
  mutate(V1 = gsub("[[:space:]]+", ",", trimws(V1))) %>%
  tidyr::separate(
    sep = ",",
    col =  V1,
    into =  c("I","A1","A2","A3","A4","A5","A6","Y")
  ) %>% mutate_all(as.numeric)
```

we need matrices for our manual solution, so we get those here.

```r
X = economic_data_clean %>% select(-I,-Y) %>% as.matrix()
A = economic_data_clean %>%
  mutate(one = 1)%>% select(one,A1,A2,A3,A4,A5,A6,-I,-Y) %>% as.matrix()
y = economic_data_clean %>% select(Y) %>% as.matrix()
```

We have two ways to solve this. First, we can use what we learned in the lecture to directly solve via the following minimization problem:

$$min||A\beta - y||^2$$

over β. This corresponds to the following formula.

$$\beta^* = (A^T A)^{-1} A^T y$$

We solve this formula with the following R code. This gives us the set of coefficients β vector.

```r
solve((t(A) %*% A), tol = 1e-21) %*% t(A) %*% y
```

```
##                       Y
## one -3475440.84060007
## A1        14.78948545
## A2        -0.03574762
## A3        -2.02019513
## A4        -1.03276578
## A5        -0.04911940
## A6      1825.54366146
```

We can check our answer using the internal `lm()` function that does what we just did by hand. This gives us the same β vector. We see the results are the same.

```r
lm(
  data = economic_data_clean,
  formula = Y ~ A1 + A2 + A3 + A4 + A5 + A6)
```

```
##
## Call:
## lm(formula = Y ~ A1 + A2 + A3 + A4 + A5 + A6, data = economic_data_clean)
```

```
##
## Coefficients:
##    (Intercept)             A1             A2             A3             A4
## -3475440.82413       14.78948       -0.03575       -2.02020       -1.03277
##             A5             A6
##       -0.04912     1825.54365
```

We now see that these coefficients are the same in both methods.