Math 189: Homework 1

Alan Lui, Derek So, Xiangyu Wei

2021-01-15

Introduction

The dataset that is used in this report is the *babies.dat* dataset, which contains the following features:

- **bwt**: Baby's weight at birth, to the nearest ounce
- **gestation**: Duration of the pregnancy in days, calculated from the first day of the last normal menstrual period.
- **parity**: Indicator for whether the baby is the first born (1) or not (0).
- **age**: Mother's age at the time of conception, in years
- **height**: Height of the mother, in inches
- **weight**: Mother's prepregnancy weight, in pounds
- **smoking Indicator**: for whether the mother smokes (1) or not (0).

We will be exploring the babies dataset using the skills we learned in class. More specifically, we will display the data, calculate the transpose, inverse, trace, etc. of certain submatrix, and show whether certain matrix is positive definite or not.

Tasks & Analysis

1. & 2. (Download *babies.dat*) Load data and give proper data citation.

```
babies <- read.csv("Data/babies.dat", sep="")
```

We used *read.csv* to read in *babies.dat* as a dataframe.

The dataset is collected for each new mother in a *Child and Health Development Study.* It is found from http://www.stat.berkeley.edu/users/statlabs/labs.html, and it is presented by *Stat Labs*: Mathematical Statistics through Applications Springer-Verlag (2001) by Deborah Nolan and Terry Speed. We extracted the dataset from https://github.com/tuckermcelroy/ma189/blob/main/Data/babies.dat at 2021-01-12 20:07:02 PST.

3. Use the *head* command to examine the first few rows of the variables **bwt**, **age**, and **weight**

```
head(babies[,c(1,4,6)])
```

```
##   bwt age weight
## 1 120  27    100
## 2 113  33    135
## 3 128  28    115
## 4 123  36    190
## 5 108  23    125
## 6 136  25     93
```

head() command is then used on the 1st (**bwt**), 4th (**age**), and 6th (**weight**) columns of babies dataset to check and select out the desired columns. The output columns (**bwt**, **age**, and **weight**) are all in interger format.

4. Define and dispay a submatrix $\mathbf{X}$ corresponding to the last 5 records (babies), for the variables **bwt**, **age**, and **weight**

```
X = as.matrix(tail(babies[,c(1,4,6)], 5))
X
```

```
##      bwt age weight
## 1232 113  27    100
## 1233 128  24    120
## 1234 130  30    150
## 1235 125  21    110
## 1236 117  38    129
```

tail() is used to retrieve the last 5 records of the dataset for the 1st (**bwt**), 4th (**age**), and 6th (**weight**) columns with specific indexing and an additional argument signifying that we only want the last 5 records. The dataframe is converted to a matrix using *as.matrix()*, and then saved to the variable $\mathbf{X}$.

5. For the above $\mathbf{X}$, compute $\mathbf{A} = \mathbf{X}'\,\mathbf{X}$ in the notebook.

```
A = t(X) %*% X
A
```

```
##          bwt   age weight
## bwt    75367 17094  75003
## age    17094  4090  17292
## weight 75003 17292  75641
```

$\mathbf{X}'$ is calculated by applying *t()* to the matrix $\mathbf{X}$, meaning the traspose of the inputted matrix. The matrix multiplication ($\mathbf{X}'\,\mathbf{X}$) is done by using the operator *%*%*. The result is saved to variable $\mathbf{A}$.

As you can see from the above result, matrix $\mathbf{A}$ is symmetric.

6. Compute and display $\mathbf{A}^{-1}$.

```
inv_A = solve(A)
inv_A
```

```
##                  bwt           age       weight
## bwt     0.0010237933  0.0003898737 -0.001104286
## age     0.0003898737  0.0074508332 -0.002089892
## weight -0.0011042856 -0.0020898918  0.001585954
```

$\mathbf{A}^{-1}$, which is the inverse of the matrix $\mathbf{A}$, is calculated by using *solve()* on the inputted matrix. And the answer is saved to variable *inv_A*.

7. Compute and display the trace of $\mathbf{A}$.

```
trace = sum(diag(A))
trace
```

```
## [1] 155098
```

The trace of an $n \times n$ matrix is defined as the sum of its diagonal elements. Thus by applying the function *diag()* onto the inputted matrix, we can get its diagonal elements, and then by applying the function *sum()* we can get its trace.

As you can see from the above result, the trace of **A** is **155098**.

8. Prove whether **A** is positive definite or not.

```
is_pos_eig = eigen(A)[[1]] > 0
is_pos_eig
```

```
## [1] TRUE TRUE TRUE
```

A symmetric $n \times n$ dimensional matrix **A** is positive definite if $\underline{v}'\mathbf{A}\underline{v} > 0$ for all non-zero length n vectors $\underline{v}$. In other words, **A** is positive definite if and only if all of its eigenvalues are positive. Since we already know that our matrix **A** is symmetric from Q5 above, then the next step in order to prove whether matrix **A** is positive definite or not is to find all of its eigenvalues and see if they are all greater than 0.

By using the function *eigen()* on the inputted matrix and index the 1st element from the output list, we find all of its eigenvalues. By comparing the eigenvalues element-wise with 0, we can see if all of its values are larger than 0. We then save the results to *is_pos_eig*.

As you can see from the above result, all eigenvalues are larger than 0, which means that **A** is a positive definite matrix.

Discussion

By exploring the *babies.dat* dataset, we are able to apply some of the skills we learned in class into practice, which includes using *head()* and *tail()* to examine the dataframe, converting dataframe into matricies using *as.matrix()*, transposing the matrix using *t()*, mutiplying two matrices using *%*%*, calculating the inverse of the matrix using *solve()*, calculating the trace of matrix using *sum()* and *diag()*, and proving whether a matrix is positive definite by retrieving its eigenvalues using *eigen()*.

We will be exploring more of R's function in data analysis in furture assignemnts and reports.