HW2MuyangShi

Muyang Shi

2024-02-08

Note: the cpp source code to this document can be found on my Github, listed as `optimize.cpp`, here.

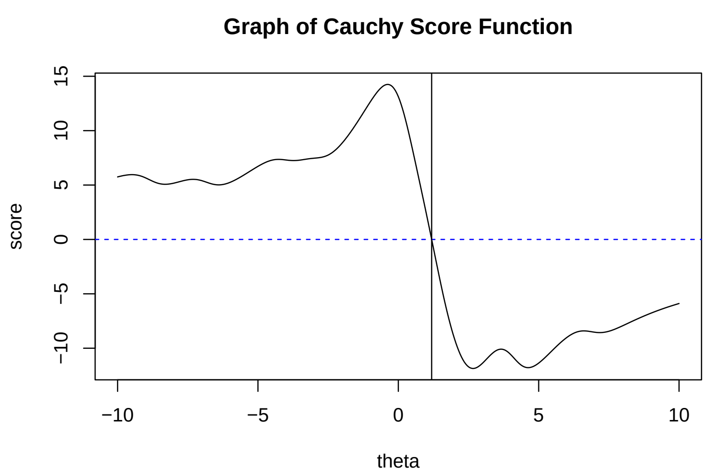
Problem 1

(a)

Using the density, we can derive that (with n observations):

$$l(\theta) = -n \log \pi - \sum_{i=1}^{n} \log(1 + (x_i - \theta)^2)$$
$$l'(\theta) = \sum_{i=1}^{n} \frac{2(x_i - \theta)}{1 + (x_i - \theta)^2}$$
$$l''(\theta) = \sum_{i=1}^{n} \frac{-2 + 2(x_i - \theta)^2}{(1 + (x_i - \theta)^2)^2}$$

Here is a plot of the first derivative of the log likelihood $l'(\theta)$: note that the vertical line is drawn where the derivative of the log likelihood equals zero, at $\hat{\theta} = 1.188$



(b)

i. Bisection

```r
Bisection_theta_hat <- Bisection_cauchy_cpp(a=0,b=3,dat=cauchy_data, eps=1e-8)
```

ii. Newton-Raphson

```r
Newton_theta_hat <- Newton_cauchy_cpp(x = 0, dat=cauchy_data, eps=1e-8)
```

iii. Fisher Scoring

```r
Fisher_theta_hat <- FisherScoring_cauchy_cpp(theta = 0, dat=cauchy_data, eps = 1e-8)
```

iv. Secant Method

```r
Secant_theta_hat <- Secant_cauchy_cpp(0, 3, dat=cauchy_data, eps = 1e-8)
```

(c)

Table 1: Results of Estimation

Method	theta_hat	Iters to Converge
Bisection	1.1879	28
Newton Raphson	1.1879	6
Fisher Scoring	1.1879	6
Secant	1.1879	7

(d)

I used the absolute convergence criteria with an $\epsilon = 1 \times 10^{-8}$, i.e. it mandates stopping when

$$\left|\hat{\theta}^{t+1} - \hat{\theta}^{t}\right| < \epsilon$$

(e)

```r
1/sqrt(-ddloglik_cauchy_cpp(Bisection_theta_hat, cauchy_data))
```

There is no "best" estimate of θ, as the four methods produce the same the point estimates $\hat{\theta} = 1.188$. The standard error of the estimate can be calculated using the fisher information evaluated at the estimate,

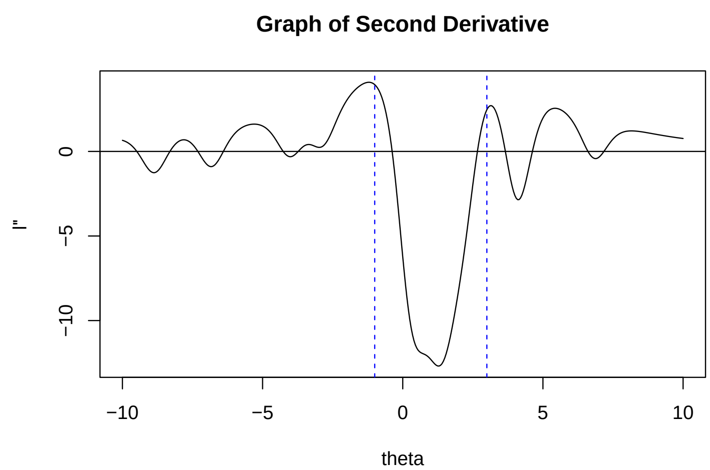
$$SE(\hat{\theta}) = \frac{1}{\sqrt{I(\hat{\theta})}} = \frac{1}{\sqrt{-l''(\hat{\theta})}} = 0.281$$

(f)

From the visual examination (i.e. "eye-balling") of the plot of the score function, we see that it crosses zero once and only once somewhere between $\theta \in (0, 3)$. Therefore,

- we initialized the Bisection solver with the two endpoints being 0 and 3. The result "should" not be sensitive to where we chose the two endpoints because the score function crosses zero only once, as long as that $\hat{\theta} = 1.188$ is within the search range between the two endpoints;

- for the other three Newton-like methods (Newton-Raphson, Fisher Scoring, and the Secant methods), calculation for the second derivative could potentially lead to trouble especially for the Newton-Raphson and Fisher Scoring methods. As illustrated in the example below, when we feed the algorithms and initial values (e.g. $\hat{\theta} = 3$ or $\hat{\theta} = -1$) that are near regions of $l''(\hat{\theta}) = 0$, the algorithm will run into non-convergence as the second derivative is on the denominator, and when the denominator turns zero it causes trouble – see the two `tryCatch` error below. As for the Secant method the above rationale stays the same as we are approximating the derivative each time only; this means that we can certainly run into the same issue and it would not converge.




```r
tryCatch(Newton_cauchy_cpp(x = 3, dat=cauchy_data, eps=1e-8),
         error = print)
```

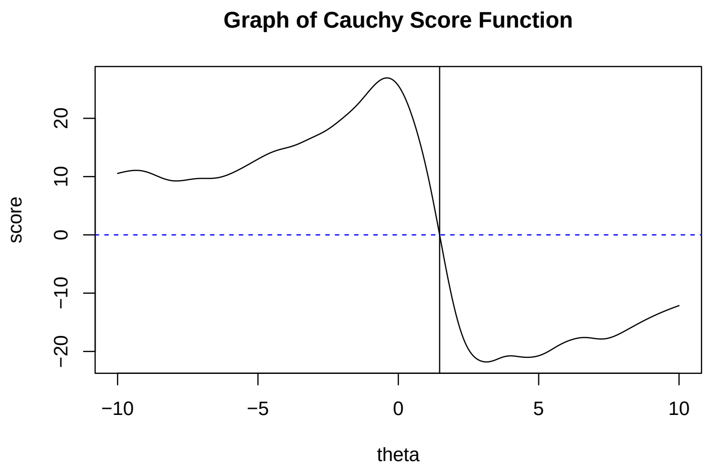
```
## <Rcpp::exception in eval(expr, envir, enclos): l''(theta_hat) equals 0!>
```

```r
tryCatch(FisherScoring_cauchy_cpp(theta = -1, dat=cauchy_data, eps=1e-8),
         error = print)
```

```
## <Rcpp::exception in eval(expr, envir, enclos): l''(theta_hat) equals 0!>
```

(g)

Below is a graph of the new score function with the new data added:



Using the four methods would actually still give the same estimates of $\hat{\theta}$ as long as we feed them the appropriate starting values:

Table 2: Results of Estimation

Method	theta_hat	Iters to Converge
Bisection	1.4713	28
Newton Raphson	1.4713	4
Fisher Scoring	1.4713	5
Secant	1.4713	6

Hence, our best estimate of $\hat{\theta}$ is 1.471, with a standard error of 0.197.

```
Bisection_theta_hat2
```

```
## [1] 1.471299
```

```
1/sqrt(-ddloglik_cauchy_cpp(Bisection_theta_hat2, cauchy_data_full))
```

```
## [1] 0.1970398
```

Problem 2

From the course slides, we know that Newton's method has quadratic convergence order $\beta = 2$, i.e.

$$\lim_{t\to\infty} \frac{|\epsilon^{(t+1)}|}{|\epsilon^{(t)}|^2} = c$$

As for the *1-step* Secant method, from the textbook equation 2.27, we have that as $t \to \infty$

$$\epsilon^{(t+1)} \approx d^{(t)} \epsilon^{(t)} \epsilon^{(t-1)}$$

, where

$$d^{(t)} \to \frac{g'''(x^*)}{2g''(x^*)} = d$$

Next, to find the β such that

$$\lim_{t\to\infty}\frac{|\epsilon^{(t+1)}|}{|\epsilon^{(t)}|^{\beta}}=c$$

we use this relationship to replace $\epsilon^{(t-1)}$ and $\epsilon^{(t+1)}$ in the equation above, we will get as $t\to\infty$,

$$c|\epsilon^{(t)}|^{\beta}=d|\epsilon^{(t)}|\frac{|\epsilon^{(t)}|}{c}^{1/\beta}$$

with rearrangement we have

$$\lim_{t\to\infty}|\epsilon^{(t)}|^{1-\beta+1/\beta}=\frac{c^{1+1/\beta}}{d}=c^*$$

where c^* is just some constant, i.e. $1-\beta+1/\beta=0$. Then, solving for β yields

$$\beta=(1+\sqrt{5})/2\approx 1.62$$

. Call the convergence rate for the *1-step* Secant method β_1, we have $\beta_1<2$. Hence, the Newton's method enjoys a faster convergence rate than the *1-step* Secant method.

Similarly, we can derive the convergence rate for the **2-step** Secant method, i.e. we need to find the β_2 such that

$$\lim_{t\to\infty}\frac{|\epsilon^{(t+2)}|}{|\epsilon^{(t)}|^{\beta_2}}=c_2$$

Using equation (2.27) from the textbook again, we want to solve for β_2 in

$$\lim_{t\to\infty}\frac{|d^{(t+1)}\epsilon^{(t+1)}\epsilon^{(t)}|}{|\epsilon^{(t)}|^{\beta_2}}=c_2$$

And as with before, 1. using the relation that

$$\lim_{t\to\infty}d^{(t+1)}\to\frac{g'''(x^*)}{2g''(x^*)}=d$$

and 2. using the *1-step* Secant convergence rate we have

$$\lim_{t\to\infty}|\epsilon^{(t+1)}|=c_1|\epsilon^{t}|^{\beta_1}$$

, we can re-write the convergence rate equation for the **2-step** Secant method as

$$\lim_{t\to\infty}\frac{dc_1|\epsilon^{(t)}|^{\beta_1}|\epsilon^{(t)}|}{|\epsilon^{t}|^{\beta_2}}=c_2$$

, i.e.,

$$\lim_{t\to\infty}|\epsilon^{(t)}|^{\beta_1+1-\beta_2}=\frac{c_2}{c_1}d=c^*\ \text{(some constant)}$$

Thus,

$$\beta_2=1+\beta_1\approx 1+1.62>2$$

, the **2-step** Secant method enjoys a faster convergence rate than the Newton's method.

Problem 3

(a)

Denote $\boldsymbol{X_i\beta} = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}$, we can write the likelihood for this problem as (treated as binomials):

$$
\begin{aligned}
L(\boldsymbol{\beta};\boldsymbol{X}) &= \prod_{i=1}^{n} \left(\frac{\exp(\boldsymbol{X_i\beta})}{1+\exp(\boldsymbol{X_i\beta})}\right)^{y_i} \left(1-\frac{\exp(\boldsymbol{X_i\beta})}{1+\exp(\boldsymbol{X_i\beta})}\right)^{1-y_i} \\
&= \prod_{i=1}^{n} \left(\frac{\exp(\boldsymbol{X_i\beta})}{1+\exp(\boldsymbol{X_i\beta})}\right)^{y_i} \left(\frac{1}{1+\exp(\boldsymbol{X_i\beta})}\right)^{1-y_i} \\
&= \prod_{i=1}^{n} \frac{(\exp(\boldsymbol{X_i\beta}))^{y_i}}{1+\exp(\boldsymbol{X_i\beta})}
\end{aligned}
$$

Hence the log likelihood is:

$$
\begin{aligned}
l(\boldsymbol{\beta};\boldsymbol{X}) &= \sum_{i=1}^{n} y_i * \log(\exp(\boldsymbol{X_i\beta})) - (1+\exp(\boldsymbol{X_i\beta})) \\
&= \sum_{i=1}^{n} y_i \boldsymbol{X_i\beta} - (1+\exp(\boldsymbol{X_i\beta}))
\end{aligned}
$$

(b)

To use the Newton-Raphson method, we need the first and the second derivatives with respect to $\boldsymbol{\beta}$:

$$
\begin{aligned}
\boldsymbol{g'}(\boldsymbol{\beta}) \equiv \frac{l(\boldsymbol{\beta};\boldsymbol{X})}{\partial\boldsymbol{\beta}} &= \sum_{i=1}^{n}\left[y_i \frac{\partial}{\partial\boldsymbol{\beta}}\boldsymbol{X_i\beta} - \frac{\partial}{\partial\boldsymbol{\beta}}\log(1+\exp(\boldsymbol{X_i\beta}))\right] \\
&= \sum_{i=1}^{n}\left[y_i \boldsymbol{X_i}^\top - \frac{\exp(\boldsymbol{X_i\beta})}{1+\exp(\boldsymbol{X_i\beta})}\boldsymbol{X_i}^\top\right] \\
&= \sum_{i=1}^{n}\left[y_i - \frac{\exp(\boldsymbol{X_i\beta})}{1+\exp(\boldsymbol{X_i\beta})}\right]\boldsymbol{X_i}^\top
\end{aligned}
$$

and

$$
\begin{aligned}
\boldsymbol{g''}(\boldsymbol{\beta}) \equiv \frac{\partial^2}{\partial\boldsymbol{\beta}} l(\boldsymbol{\beta};\boldsymbol{X}) &= \frac{\partial}{\partial\boldsymbol{\beta}}\sum_{i=1}^{n}\left[y_i - \frac{\exp(\boldsymbol{X_i\beta})}{1+\exp(\boldsymbol{X_i\beta})}\right]\boldsymbol{X_i}^\top \\
&= \sum_{i=1}^{n} -\boldsymbol{X}_i\boldsymbol{X}_i^\top \frac{\exp(\boldsymbol{X_i\beta})}{(1+\exp(\boldsymbol{X_i\beta}))^2}
\end{aligned}
$$

then, the iterative update is

$$
\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^t - \boldsymbol{g''}(\boldsymbol{\beta^{(t)}})^{-1}\boldsymbol{g'}(\boldsymbol{\beta^{(t)}})
$$

and we use an absolute convergence criteria so that we stop when

$$
\left|\boldsymbol{\beta}^{(t+1)} - \boldsymbol{\beta}^t\right|_2 < \epsilon
$$

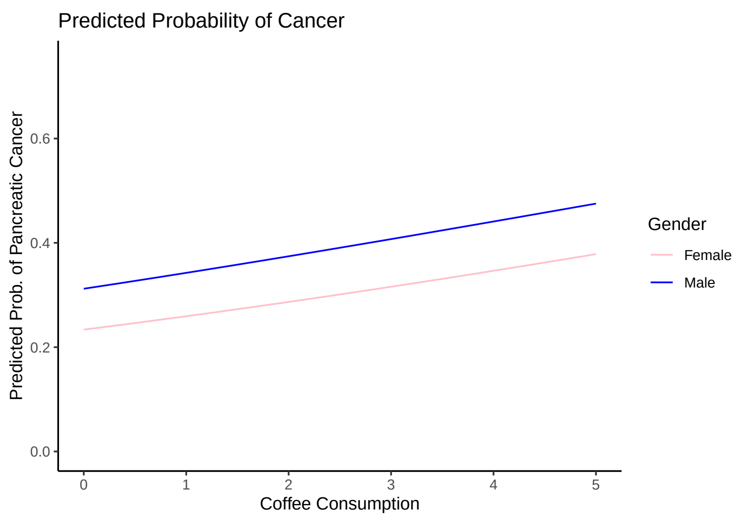
```
mod <- Newton_logit(b0=c(0,0,0),Y=Y,X=X,eps=1e-8)
```

Initialized at $(0,0,0)$, the optimizer converges after 5 iterations, yielding the estimates shown in the table below:

Table 3: Results of Estimation

	Coef. Est.	Std. Error
Beta0	-1.188	0.157
Beta1	0.138	0.043
Beta2	0.397	0.134

(c)



The estimated log odds of getting pancreatic cancer for male is

$$\log(\frac{\hat{p}}{1-\hat{p}}) = -0.791 + 0.138 * x_{coffee}$$

and the estimated log odds of getting pancreatic cancer for female is

$$\log(\frac{\hat{p}}{1-\hat{p}}) = -1.188 + 0.138 * x_{coffee}$$

This means that, on average:

- while holding gender constant, one additional cup of coffee consumption would be associated with an increase of 0.138 in the **log** odds of getting pancreatic cancer, or an increase in the odds by a factor of $\exp(0.138) = 1.148$.
- while holding coffee consumption constant, males are associated with an increase of 0.397 in the **log** odds of getting pancreatic cancer as compared to females, or an increase in the odds by a factor of $\exp(0.397) = 1.488$.

(d)

Using normal approximation (i.e. using a critical value of $z = 1.96$), testing against the null hypothesis that $H_0 : \beta_i = 0$ for $i \in (0, 1, 2)$, we have strong enough evidence to conclude that all the coefficients are significantly different from zero as their z-scores are all larger than the critical value (in magnitude).

```r
I <- -mod$Hessian # fisher information
var <- solve(I) # variance
sig <- sqrt(diag(var)) # standard deviation
z <- c(mod$Beta) / sig # Z-scores
# abs(z) > 1.96 ---> TRUE, TRUE, TRUE
z
```

```
## [1] -7.550765  3.236815  2.971351
```