STAT 600 - HW 2

Kevin Korsurat

All Rcpp/RcppArmadillo can be found in my GitHub.

Question 1

(a)

First, consider the likelihood and the log-likelihood function.

$$\begin{aligned}
\mathrm{L}(\theta) &= \prod_{i=1}^{n} \frac{1}{\pi\left(1 + (x_i - \theta)^2\right)} \\
l(\theta) &= \log\left(L(\theta)\right) \\
&= \log\left(\prod_{i=1}^{n} \frac{1}{\pi\left(1 + (x_i - \theta)^2\right)}\right) \\
&= \sum_{i=1}^{n} \log\left(\frac{1}{\pi\left(1 + (x_i - \theta)^2\right)}\right) \\
&= -\sum_{i=1}^{n} \log\left(\pi\left(1 + (x_i - \theta)^2\right)\right) \\
&= -n\log(\pi) - \sum_{i=1}^{n} \log\left(1 + (x_i - \theta)^2\right)
\end{aligned}$$

Then, consider the derivative of the log-likelihood, $l'(\theta)$.

$$\begin{aligned}
l'(\theta) &= \frac{d}{d\theta} l(\theta) \\
&= -\sum_{i=1}^{n} \frac{1}{1 + (x_i - \theta)^2}\left[\frac{d}{d\theta}(x_i - \theta)^2\right] \\
&= 2\sum_{i=1}^{n} \frac{x_i - \theta}{1 + (x_i - \theta)^2}
\end{aligned}$$

```r
dat <- c(-8.86, -6.82, -4.03, -2.84, 0.14, 0.19, 0.24, 0.27, 0.49, 0.62, 0.76, 1.09,
         1.18, 1.32, 1.36, 1.58, 1.58, 1.78, 2.13, 2.15, 2.36, 4.05, 4.11, 4.12,
         6.83)
rangeTheta <- seq(-10, 10, 0.01)
data.frame(theta = rangeTheta, dll = sapply(rangeTheta, dloglik, x = dat)) %>%
  ggplot(aes(x = theta, y = dll)) +
  geom_line() +
```

```
theme_bw() +
labs(x = TeX("\\theta"), y = TeX("l'(\\theta)"),
     title = "The plot of the derivative of the log-likelihood based on 25 daat points")
```

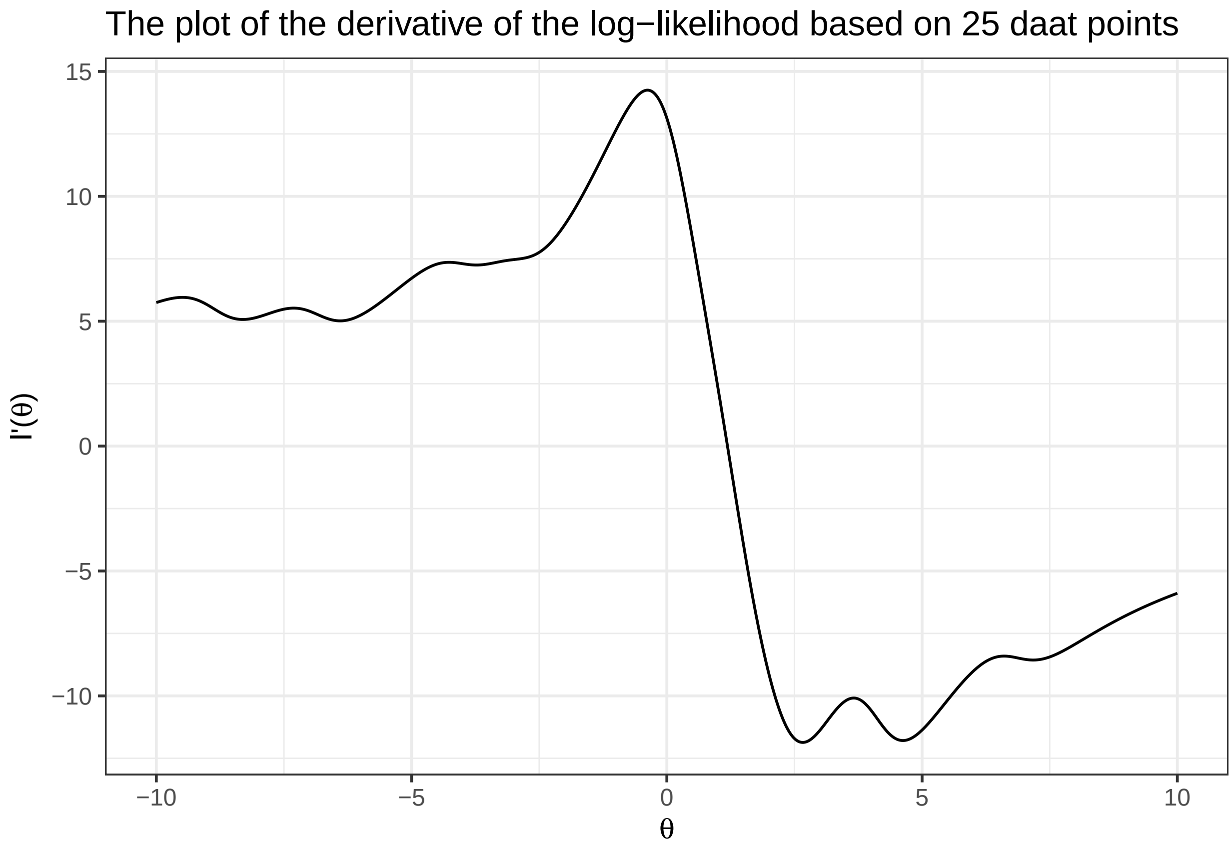

Figure 1: The plot of the derivative of the log-likelihood on the original dataset.

(b)

This is the second derivation for the log-likelihood function.

$$
\begin{aligned}
l''(\theta) &= \frac{d}{d\theta} l'(\theta) \\
&= \frac{d}{d\theta} 2 \sum_{i=1}^{n} \frac{x_i - \theta}{1 + (x_i - \theta)^2} \\
&= 2 \sum_{i=1}^{n} \frac{d}{d\theta} \frac{x_i - \theta}{1 + (x_i - \theta)^2} \\
&= 2 \sum_{i=1}^{n} \frac{-1 + (x_i - \theta)^2}{\left(1 + (x_i - \theta)^2\right)^2}
\end{aligned}
$$

(c)

Below are the result from each methods. Note that I have set the ϵ to be 1×10^{-5}.

```
### Run all methods
eps_set <- 1e-5
bs_dat <- bisect_q1(min(dat), max(dat), dat, eps = eps_set)
nr_dat <- nr_q1(x0 = 0, dat = dat, eps = eps_set)
fs_dat <- fs_q1(x0 = 0, dat = dat, eps = eps_set)
sc_dat <- sc_q1(x0 = 0, x1 = 1e-5, dat = dat, eps = eps_set)

### Create the table
data.frame(theta = c(bs_dat$xt, nr_dat$xt, fs_dat$xt, sc_dat$xt),
           iter = c(bs_dat$n_iter, nr_dat$n_iter, fs_dat$n_iter, sc_dat$n_iter)) %>%
  `rownames<-`(c("Bisection", "Newton-Raphson", "Fisher Scoring", "Secant Method")) %>%
  kable(digits = 5, col.names = c("$\\hat\\theta$", "Number of iteration"),
        caption = "The result from each methods with only 25 observations.")
```

Table 1: The result from each methods with only 25 observations.

	$\hat{\theta}$	Number of iteration
Bisection	1.18795	20
Newton-Raphson	1.18795	5
Fisher Scoring	1.18795	5
Secant Method	1.18794	5

(d)

For the convergence criteria used in this problem, I decided to employ the absolute convergence criterion, as the $x^{(t)}$ might be close to 0 in some iterations, as indicated by the plot shown in part (a). Additionally, the value of x is neither too tiny nor too huge compared to ϵ.

(e)

(f)

(g)

```
add_dat <- c(-8.34, -1.73, -0.40, -0.24, 0.60, 0.94, 1.05, 1.06, 1.45, 1.50,
             1.54, 1.72, 1.74, 1.88, 2.04, 2.16, 2.39, 3.01, 3.01, 3.08, 4.66,
             4.99, 6.01, 7.06, 25.45)

### Run all methods with complete data
eps_set <- 1e-5
bs_cdat <- bisect_q1(min(c(dat, add_dat)), max(c(dat, add_dat)), dat = c(dat, add_dat), eps = eps_set)
nr_cdat <- nr_q1(x0 = 0.5, dat = c(dat, add_dat), eps = eps_set)
fs_cdat <- fs_q1(x0 = 0.5, dat = c(dat, add_dat), eps = eps_set)
sc_cdat <- sc_q1(x0 = 0, x1 = 0.5, dat = c(dat, add_dat), eps = eps_set)

### Create the table
```

```r
data.frame(theta = c(bs_cdat$xt, nr_cdat$xt, fs_cdat$xt, sc_cdat$xt),
           iter = c(bs_cdat$n_iter, nr_cdat$n_iter, fs_cdat$n_iter, sc_cdat$n_iter)) %>%
  `rownames<-`(c("Bisection", "Newton-Raphson", "Fisher Scoring", "Secant Method")) %>%
  kable(digits = 5, col.names = c("$\\hat\\theta$", "Number of iteration"),
        caption = "The result from each methods with all 50 observations.")
```

Table 2: The result from each methods with all 50 observations.

	$\hat{\theta}$	Number of iteration
Bisection	1.47131	21
Newton-Raphson	1.47130	5
Fisher Scoring	1.47130	5
Secant Method	1.47130	5

Question 2

Question 3

(a)

I will denote $\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}$ as $\boldsymbol{x}_i\boldsymbol{\beta}$. Since we know that $Y_i \sim \text{Ber}\left(\frac{\exp(\boldsymbol{x}_i\boldsymbol{\beta})}{1+\exp(\boldsymbol{x}_i\boldsymbol{\beta})}\right)$, then the likelihood and the log-likelihood can be derived as below.

$$\begin{aligned}
\mathrm{L}(\boldsymbol{\beta}) &= \prod_{i=1}^{n}\left[\frac{\exp(\boldsymbol{x}_i\boldsymbol{\beta})}{1+\exp(\boldsymbol{x}_i\boldsymbol{\beta})}\right]^{y_i}\left[1-\frac{\exp(\boldsymbol{x}_i\boldsymbol{\beta})}{1+\exp(\boldsymbol{x}_i\boldsymbol{\beta})}\right]^{1-y_i} \\
&= \prod_{i=1}^{n}\frac{\exp(\boldsymbol{x}_i\boldsymbol{\beta})^{y_i}}{1+\exp(\boldsymbol{x}_i\boldsymbol{\beta})} \\
l(\boldsymbol{\beta}) &= \log(\mathrm{L}(\boldsymbol{\beta})) \\
&= \sum_{i=1}^{n}\left[y_i(\boldsymbol{x}_i\boldsymbol{\beta}) - \log(1+\exp(\boldsymbol{x}_i\boldsymbol{\beta}))\right]
\end{aligned}$$

(b)

First, consider the first derivative of the log-likelihood w.r.t. $\boldsymbol{\beta}$, or the gradient.

$$\begin{aligned}
\nabla_{\boldsymbol{\beta}} l(\boldsymbol{\beta}) &= \frac{d}{d\boldsymbol{\beta}} l(\boldsymbol{\beta}) \\
&= \sum_{i=1}^{n}\left[y_i\frac{d}{d\boldsymbol{\beta}}\boldsymbol{x}_i\boldsymbol{\beta} - \frac{d}{d\boldsymbol{\beta}}\log(1+\exp(\boldsymbol{x}_i\boldsymbol{\beta}))\right] \\
&= \sum_{i=1}^{n}\left[y_i\boldsymbol{x}_i^T - \frac{\exp(\boldsymbol{x}_i\boldsymbol{\beta})}{1+\exp(\boldsymbol{x}_i\boldsymbol{\beta})}\boldsymbol{x}_i^T\right] \\
&= \sum_{i=1}^{n}\left[y_i - \frac{\exp(\boldsymbol{x}_i\boldsymbol{\beta})}{1+\exp(\boldsymbol{x}_i\boldsymbol{\beta})}\right]\boldsymbol{x}_i^T
\end{aligned}$$

We can rewrite the formula above in a matrix form as $\nabla_{\boldsymbol{\beta}} l(\boldsymbol{\beta}) = \boldsymbol{X}^T\left(\boldsymbol{Y} - \hat{\boldsymbol{Y}}\right)$, where $\hat{\boldsymbol{Y}}$ is a vector consisted of $\frac{\exp\left(\boldsymbol{x}_i^T\boldsymbol{\beta}\right)}{1+\exp\left(\boldsymbol{x}_i^T\boldsymbol{\beta}\right)}$ since we can think this quantity as a predicted probability of success for the observation i.

Now, we will consider the Hessian for the log-likelihood.

$$
\begin{aligned}
H(\boldsymbol{\beta}) &= \nabla_{\boldsymbol{\beta}}\left(\nabla_{\boldsymbol{\beta}} l(\boldsymbol{\beta})\right) \\
&= \nabla_{\boldsymbol{\beta}} \sum_{i=1}^{n}\left[y_i - \frac{\exp\left(\boldsymbol{x}_i\boldsymbol{\beta}\right)}{1+\exp\left(\boldsymbol{x}_i\boldsymbol{\beta}\right)}\right]\boldsymbol{x}_i^T \\
&= -\sum_{i=1}^{n}\boldsymbol{x}_i^T\boldsymbol{x}_i\frac{\exp\left(\boldsymbol{x}_i\boldsymbol{\beta}\right)}{\left(1+\exp\left(\boldsymbol{x}_i\boldsymbol{\beta}\right)\right)^2}
\end{aligned}
$$

Similarly, we can rewite the Hessian matrix in the matrix form as $H(\boldsymbol{\beta}) = \boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}$, where $\boldsymbol{W}$ is a matrix consisted of $-\frac{\exp(\boldsymbol{x}_i\boldsymbol{\beta})}{(1+\exp(\boldsymbol{x}_i\boldsymbol{\beta}))^2}$ as a diagonal while the off-diagonal are 0.

By applying the Newton-Ralphson, we will update the parameters for the iteration t by using $\boldsymbol{\beta}^{(t)} = \boldsymbol{\beta}^{(t-1)} - \left[H\left(\boldsymbol{\beta}^{(t-1)}\right)\right]^{-1}\left[\nabla_{\boldsymbol{\beta}} l\left(\boldsymbol{\beta}^{(t-1)}\right)\right] = \boldsymbol{\beta}^{(t-1)} - \left[\boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}\right]^{-1}\boldsymbol{X}^T\left(\boldsymbol{Y} - \hat{\boldsymbol{Y}}\right)$.

```r
### Data
#### yi, intercept, x1 (coffee assumption), x2 (gender)
designMat <- rbind(c(1, 1, 0, 1), c(0, 1, 0, 1), c(1, 1, 2, 1), c(0, 1, 2, 1),
      c(1, 1, 4, 1), c(0, 1, 4, 1), c(1, 1, 5, 1), c(0, 1, 5, 1),
      c(1, 1, 0, 0), c(0, 1, 0, 0), c(1, 1, 2, 0), c(0, 1, 2, 0),
      c(1, 1, 4, 0), c(0, 1, 4, 0), c(1, 1, 5, 0), c(0, 1, 5, 0)) %>%
  as.matrix()

repTime <- c(9, 41 - 9, 94, 213 - 94, 53, 127 - 53, 60, 142 - 60,
             11, 67 - 11, 59, 211- 59, 53, 133 - 53, 28, 76 - 28)

designMat <- designMat[rep(1:nrow(designMat), times = repTime), ]
```

```r
### Run the optimization
resultQ3 <- optimQ3(desMat = designMat[, -1], Y = designMat[, 1],
                    b0 = c(0, 0, 0), eps = 1e-10)
resultQ3$bt
```

```
##            [,1]
## [1,] -1.1877905
## [2,]  0.1382988
## [3,]  0.3972517
```

(d)

```r
est_SD <- sqrt(diag(solve(t(designMat[, -1]) %*% resultQ3$W %*% designMat[, -1])))
z_stat <- (resultQ3$bt - 0)/est_SD

### Compare with z-statistics
(qnorm(0.05/2) <= z_stat) & (z_stat <= qnorm(1 - (0.05/2))) ## If TRUE, FTR H0
```

```
##        [,1]
## [1,] FALSE
## [2,] FALSE
## [3,] FALSE
```