Homework 2

GitHub Repo: https://github.com/rw417/bios731_hw2_wan

Context

This assignment reinforces ideas in Module 2: Optimization. We focus specifically on implementing the Newton's method, EM, and MM algorithms.

Due date and submission

Please submit (via Canvas) a PDF containing a link to the web address of the GitHub repo containing your work for this assignment; git commits after the due date will cause the assignment to be considered late. Due date is Wednesday, 2/19 at 10:00AM.

Problem 0

This "problem" focuses on structure of your submission, especially the use git and GitHub for reproducibility, R Projects to organize your work, R Markdown to write reproducible reports, relative paths to load data from local files, and reasonable naming structures for your files.

To that end:

- create a public GitHub repo + local R Project; I suggest naming this repo / directory bios731_hw2_YourLastName (e.g. bios731_hw2_wrobel for Julia)
- Submit your whole project folder to GitHub
- Submit a PDF knitted from Rmd to Canvas. Your solutions to the problems here should be implemented in your .Rmd file, and your git commit history should reflect the process you used to solve these Problems.

Algorithms for logistic regression

For a given subject in a study, we are interested in modeling $\pi_i = P(Y_i = 1|X_i = x_i)$, where $Y_i \in \{0, 1\}$. The logistic regression model takes the form

$$\text{logit}(\pi_i) = \log\left(\frac{\pi_i}{1-\pi_i}\right) = \log\left(\frac{P(Y_i = 1|X_i)}{1 - P(Y_i = 1|X_i)}\right) = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \ldots + \beta_p X_{pi}$$

- $Y_1, Y_2, \ldots, Y_n \sim Bernoulli(\pi)$
- PDF is $f(y_i; \pi) = \pi^{y_i}(1-\pi)^{1-y_i}$

Problem 1: Newton's method

- Derive likelihood, gradient, and Hessian for logistic regression for an arbitrary number of predictors p.

ANSWER:

Likelihood:

We know that,

$$\log(\frac{\pi_i}{1-\pi_i}) = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \ldots + \beta_p X_{pi} = \sum_{j=0}^{p} \beta_j X_{ji}, \text{where } X_{0i} = 1$$

$$\pi_i = \frac{1}{1+\exp(-\sum_{j=0}^{p} \beta_j X_{ji})}$$

Thus,

$$L(y_i, \pi_i) = \prod_{i=1}^{n} \pi_i^{y_i} (1-\pi_i)^{1-y_i}$$

$$L(y_i, \beta_j) = \prod_{i=1}^{n} \left(\frac{1}{1+\exp(-\sum_{j=0}^{p} \beta_j X_{ji})} \right)^{y_i} \left(\frac{\exp(-\sum_{j=0}^{p} \beta_j X_{ji})}{1+\exp(-\sum_{j=0}^{p} \beta_j X_{ji})} \right)^{(1-y_i)}$$

$$l(y_i, \beta_j) = \sum_{i=1}^{n} \left\{ y_i \log(\frac{1}{1+\exp(-\sum_{j=0}^{p} \beta_j X_{ji})}) + (1-y_i) \log(\frac{\exp(-\sum_{j=0}^{p} \beta_j X_{ji})}{1+\exp(-\sum_{j=0}^{p} \beta_j X_{ji})}) \right\}$$

$$= \sum_{i=1}^{n} \left\{ y_i \left[log(\frac{1}{1+\exp(-\sum_{j=0}^{p} \beta_j X_{ji})}) - \log(\frac{\exp(-\sum_{j=0}^{p} \beta_j X_{ji})}{1+\exp(-\sum_{j=0}^{p} \beta_j X_{ji})}) \right] - \log(1+\exp(\sum_{j=0}^{p} \beta_j X_{ji})) \right\}$$

$$= \sum_{i=1}^{n} \left\{ y_i \sum_{j=0}^{p} \beta_j X_{ji} - \log(1+\exp(\sum_{j=0}^{p} \beta_j X_{ji})) \right\}$$

Gradient:

$$\nabla_{\beta_j} l(y_i, \beta_j) = \sum_{i=1}^{n} \left\{ y_i X_{ji} - \frac{\exp(\sum_{j=0}^{p} \beta_j X_{ji})}{1+\exp(\sum_{j=0}^{p} \beta_j X_{ji})} X_{ji} \right\}$$

$$= \sum_{i=1}^{n} \left\{ \left(y_i - \frac{\exp(\sum_{j=0}^{p} \beta_j X_{ji})}{1+\exp(\sum_{j=0}^{p} \beta_j X_{ji})} \right) X_{ji} \right\}$$

Hessian:

$$\nabla^2_{\beta_j \beta_k} l(y_i, \beta) = \sum_{i=1}^{n} \left\{ -\frac{\exp(\sum_{j=0}^{p} \beta_j X_{ji})}{(1+\exp(\sum_{j=0}^{p} \beta_j X_{ji}))^2} X_{ji} X_{ki} \right\}$$

- What is the Newton's method update for β for logistic regression?

ANSWER:

$$\beta_j^{(t+1)} = \beta_j^{(t)} - [\nabla^2_{\beta_j \beta_k} l(y_i, \beta)]^{-1} \nabla_{\beta_j} \mathbf{l}(\mathbf{y_i}, \beta_\mathbf{j})$$

- Is logistic regression a convex optimization problem? Why or why not?

ANSWER:

It is a convex optimization problem because the negative Hessian is convex. Thus, maximizing the log likelihood is the same as minimizing the negative log likelihood.

Problem 2: MM

(A) In constructing a minorizing function, first prove the inequality

$$-\log\{1+\exp x_i^T\theta\} \geq -\log\{1+\exp(X_i^T\theta^{(k)})\} - \frac{\exp(X_i^T\theta) - \exp(X_i^T\theta^{(k)})}{1+\exp(X_i^T\theta^{(k)})}$$

with equality when $\theta = \theta^{(k)}$. This eliminates the log terms.

ANSWER:

Treat $\exp(X_i^T\theta^{(k)})$ as x and $\exp(X_i^T\theta)$ as y. Then $f(z) = -\log(1+z)$. It's easy to see that $f(z) = \log(1+z)$ is a convex function. By the property of supporting hyperplanes of convex functions:

$$\begin{aligned} f(y) &\geq f(x) + f'(x)(y-x) \\ -\log(1+y) &\geq -\log(1+x) - \frac{1}{1+x}(y-x) \\ -\log(1+\exp(X_i^T\theta)) &\geq -\log(1+\exp(X_i^T\theta^{(k)})) - \frac{1}{1+\exp(X_i^T\theta^{(k)})}(\exp(X_i^T\theta) - \exp(X_i^T\theta^{(k)})), \end{aligned}$$

with equality when $\exp(X_i^T\theta) = \exp(X_i^T\theta^{(k)})$.

Moreover, $\exp(z)$ is a monotone increasing function of z, so $\exp(X_i^T\theta) = \exp(X_i^T\theta^{(k)}) \implies X_i^T\theta = X_i^T\theta^{(k)}$. When X_i^T is a vector, this implies $\theta = \theta^{(k)}$. Therefore, $\exp(X_i^T\theta) = \exp(X_i^T\theta^{(k)}) \implies \theta = \theta^{(k)}$.

(B) Now apply the arithmetic-geometric mean inequality to the exponential function $\exp(X_i^T\theta)$ to separate the parameters. Assuming that θ has p components and that there are n observations, show that these maneuvers lead to a minorizing function

$$g(\theta|\theta^{(k)}) = -\frac{1}{p}\sum_{i=1}^{n}\frac{\exp(X_i^T\theta^{(k)})}{1+\exp(X_i^T\theta^{(k)})}\sum_{j=1}^{p}\exp\{pX_{ij}(\theta_j - \theta_j^{(k)})\} + \sum_{i=1}^{n}Y_iX_i^T\theta = 0$$

up to a constant that does not depend on θ.

ANSWER:

$$\begin{aligned} \frac{\exp(X_i^T\theta)}{\exp(X_i^T\theta^{(k)})} &= \exp(X_i^T(\theta - \theta^{(k)})) \\ &= \exp(pX_i^T(\theta - \theta^{(k)}))^{1/p} \\ &= \exp(\sum_{j=1}^{p} pX_{ij}(\theta_j - \theta_j^{(k)}))^{1/p} \\ &= \prod_{j=1}^{p}\exp(pX_{ij}(\theta_j - \theta_j^{(k)}))^{1/p} \\ &\leq \frac{1}{p}\sum_{j=1}^{p}\exp(pX_{ij}(\theta_j - \theta_j^{(k)})) \end{aligned}$$

The log likelihood of the logistic regression is:

$$
\begin{aligned}
l(y_i, \beta_j) &= \sum_{i=1}^{n} \left\{ y_i \sum_{j=0}^{p} \beta_j X_{ji} - \log(1 + \exp(\sum_{j=0}^{p} \beta_j X_{ji})) \right\} \\
&= \sum_{i=1}^{n} \left\{ Y_i X_i^T \theta - \log(1 + \exp(X_i^T \theta) \right\} \\
&\geq \sum_{i=1}^{n} \left\{ Y_i X_i^T \theta - \log(1 + \exp(X_i^T \theta^{(k)}) - \frac{\exp(X_i^T \theta) - \exp(X_i^T \theta^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \right\} \\
&= \sum_{i=1}^{n} \left\{ Y_i X_i^T \theta - \frac{\exp(X_i^T \theta)}{1 + \exp(X_i^T \theta^{(k)})} - \log(1 + \exp(X_i^T \theta^{(k)}) + \frac{\exp(X_i^T \theta^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \right\} \\
l(y_i, \beta_j) &\geq \sum_{i=1}^{n} \left\{ Y_i X_i^T \theta - \frac{\exp(X_i^T \theta)}{1 + \exp(X_i^T \theta^{(k)})} \right\} + \sum_{i=1}^{n} \left\{ - \log(1 + \exp(X_i^T \theta^{(k)}) + \frac{\exp(X_i^T \theta^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \right\} \\
l(y_i, \beta_j) &\geq \sum_{i=1}^{n} \left\{ Y_i X_i^T \theta - \frac{1}{p} \sum_{j=1}^{p} \frac{\exp(X_i^T \theta^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \exp(p X_{ij} (\theta_j - \theta_j^{(k)})) \right\} + \sum_{i=1}^{n} \left\{ - \log(1 + \exp(X_i^T \theta^{(k)}) + \frac{\exp(X_i^T \theta^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \right\} \\
l(y_i, \beta_j) &\geq g(\theta | \theta^{(k)}) + \sum_{i=1}^{n} \left\{ - \log(1 + \exp(X_i^T \theta^{(k)}) + \frac{\exp(X_i^T \theta^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \right\}
\end{aligned}
$$

Note that:

$$
- \log(1 + \exp(X_i^T \theta^{(k)}) + \frac{\exp(X_i^T \theta^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \leq 1,
$$

which does not depend on θ. So while $g(\theta|\theta^{(k)})$ may be greater than the log-likelihood, its difference is constant w.r.t θ. As such, $g(\theta|\theta^{(k)})$ still works as a minorizing function.

(C) Finally, prove that maximizing $g(\theta|\theta^{(k)})$ consists of solving the equation

$$
- \sum_{i=1}^{n} \frac{\exp(X_i^T \theta^{(k)}) X_{ij} \exp(-p X_{ij} \theta_j^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \exp(p X_{ij} \theta_j) + \sum_{i=1}^{n} Y_i X_{ij} = 0
$$

for each j.

ANSWER:

$$
\begin{aligned}
\frac{\partial}{\partial \theta_j} g(\theta | \theta^{(k)}) &= 0 \\
- \frac{1}{p} \sum_{i=1}^{n} \frac{\exp(X_i^T \theta^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \sum_{j=1}^{p} \frac{\partial}{\partial \theta_j} \exp\{p X_{ij} (\theta_j - \theta_j^{(k)})\} + \sum_{i=1}^{n} Y_i \sum_{j=1}^{p} \frac{\partial}{\partial \theta_j} X_{ij} \theta_j &= 0 \\
- \frac{1}{p} \sum_{i=1}^{n} \frac{\exp(X_i^T \theta^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \exp\{p X_{ij} (\theta_j)\} \cdot p X_{ij} \cdot \exp(-p X_i^T \theta^{(k)}) + \sum_{i=1}^{n} Y_i X_{ij} &= 0 \\
- \sum_{i=1}^{n} \frac{\exp(X_i^T \theta^{(k)}) X_{ij} \exp(-p X_{ij} \theta_j^{(k)})}{1 + \exp(X_i^T \theta^{(k)})} \exp(p X_{ij} \theta_j) + \sum_{i=1}^{n} Y_i X_{ij} &= 0
\end{aligned}
$$

Problem 3: simulation

Next we will implement logistic regression in R four different ways and compare the results using a short simulation study.

- implement using Newton's method from 1.1 in R
- implement using MM from 1.2 in R
- implement using `glm()` in R
- implement using `optim()` in R:
  - Use the option `method = "BFGS"`, which implements a Quasi-Newton approach

Simulation study specification:

- simulate from the model $logit(P(Y_i = 1|X_i)) = \beta_0 + \beta_1 X_i$
  - $\beta_0 = 1$
  - $\beta_1 = 0.3$
  - $X_i \sim N(0, 1)$
  - $n = 200$
  - $nsim = 1$
- For your implementation of MM and Newton's method, select your own starting value and stopping criterion, but make sure they are the same for the two algorithms

You only need to run the simulation using **one simulated dataset**. For each of the four methods, report:

- $\hat{\beta}_0$, $\hat{\beta}_1$
- 95% confidence intervals for $\hat{\beta}_0$, $\hat{\beta}_1$
- computation time
- number of iterations to convergence

Make 2-3 plots or tables comparing your results, and summarize these findings in one paragraph.

```
# Run this code to run the simulation
source(here::here("20_analysis", "run_simulations.R"))
```

Results Summary

Table 1: Summary of Simulation Results

	Newton	MM	GLM	Optim
Beta_0	1.0397	1.0397	1.0397	1.0397
Beta_0 95% CI	(0.73446, 1.3858)	(0.73457, 1.3858)	(0.73446, 1.3858)	(0.73446, 1.3858)
Beta_1	-0.14066	-0.14065	-0.14066	-0.14068
Beta_1 95% CI	(-0.43381, 0.16474)	(-0.43366, 0.16465)	(-0.43381, 0.16474)	(-0.43377, 0.16442)
Iterations	5.05	100	4	9.23
Computation Time	0.00055	1.7062	0.00225	0.00225

We see that all four optimization methods produced very similar estimates. GLM converged fastest, taking only 4 iterations. However, Newton converged quicker than GLM despite taking one more iteration, presumably because the `glm()` function has extra overhead in order to be more flexible. While Optim took 9 iterations on average to converge, its computation time was the same as GLM. The MM method took the longest and never converged. This is because we used are optimizing a minorization function that perhaps is not optimal. For both Newton and MM, I used the an error of 1e-08 and a max iteration of 100.

Problem 4: EM algorithm for censored exponential data

This will be a continuation of the lab problem on EM for censored exponential data. Suppose we have survival times $t_1, \ldots, t_n \sim Exponential(\lambda)$.

- Do not observe all survival times because some are censored at times $c_1, \ldots, c_n$.
- Actually observe $y_1, \ldots, y_n$, where $y_i = min(y_0, c_i)$

- Also have an indicator δ_i where $\delta_i = 1$ is $y_i \leq c_i$
  - i.e. $\delta_i = 1$ if not censored and $\delta_i = 0$ if censored

Do the following:

- Derive an EM algorithm to estimate the parameter λ. Show your derivation here and report updates for the **E-step** and **M-Step**.

ANSWER:

In class, we treated the survival time of the censored subjects as missing, denoted as z_i, and $t_i = \delta_i y_i + (1-\delta_i) z_i)$. We derived the total data likelihood as

$$p(y, t|\lambda) = \prod_{i=1}^{n} \lambda \exp\{-\lambda t\}$$
$$p(y, z|\lambda) = \prod_{i=1}^{n} \lambda \exp\{-\lambda(\delta_i y_i + (1-\delta_i) z_i)\}.$$

The log-likelihood is

$$l(y, t|\lambda) = n \log \lambda - \lambda \sum_{i=1}^{n} \left[\delta_i y_i + (1-\delta_i) z_i\right].$$

We derived the Q-function as

$$Q(\lambda|\lambda^{(k)}) = n \log \lambda - \lambda \sum_{i=1}^{n} \left[\delta_i y_i + (1-\delta_i) E[z_i|y_i, \lambda^{(k)}]\right].$$

E-Step The memory-less property of the exponential distribution gives us

$$E[z_i|y_i, \lambda^{(k)}] = E[z_i|z_i \geq y_i, \lambda^{(k)}] = y_i + \frac{1}{\lambda^{(k)}}.$$

Thus, the Q-function becomes

$$Q(\lambda|\lambda^{(k)}) = n \log \lambda - \lambda \sum_{i=1}^{n} \left[\delta_i y_i + (1-\delta_i)\left(y_i + \frac{1}{\lambda^{(k)}}\right)\right] = n \log \lambda - \lambda \sum_{i=1}^{n} \left[y_i + \frac{1-\delta_i}{\lambda^{(k)}}\right].$$

M-Step The M-step is to maximize the Q-function. Taking the first derivative of the Q-function and setting it to 0 produces

$$\frac{\partial}{\partial \lambda} Q(\lambda|\lambda^{(k)}) = \frac{n}{\lambda} - \sum_{i=1}^{n} \left[y_i + \frac{1-\delta_i}{\lambda^{(k)}}\right] = 0$$
$$\lambda^{(k+1)} = \frac{n}{\sum_{i=1}^{n} \left[y_i + \frac{1-\delta_i}{\lambda^{(k)}}\right]}.$$

- Implement your EM in R and fit it to the `veteran` dataset from the `survival` package.
  - Report your fitted λ value. How did you monitor convergence?
  - Report a 95% confidence interval for λ, and explain how it was obtained.
  - Compare 95% confidence interval and results from those obtained by fitting an accelerated failure time model (AFT) in R with exponential errors. You can fit an AFT model using the `phreg()` function from the `survival` package. If you choose `dist = "weibull` and `shape = 1` as parameter arguments, this will provide exponential errors.

Table 2: Comparison of My EM Algo and Survfit

	EM	Survreg
Lambda	4.8641	4.8689
Lambda 95% CI	(4.6421, 5.0838)	(4.6957, 5.0422)

The λ I obtained using my EM algorithm is very close to the result provided by the `Survreg()` function, although its 95% confidence interval is slightly larger than that of `Survreg()`. To obtain the confidence interval for the EM algorithm, I took bootstrap samples 1,000 times. For the `Survreg()` function, I used the confidence interval provided by the function rather than opting for bootstrap. I monitored convergence of EM by monitoring the difference in the observed log-likelihood between iterations.

Extra credit (up to 10 points)! Expected vs. observed information

Part A: Show that the expected and observed information are equivalent for logistic regression

Part B: Let's say you are instead performing probit regression, which is similar to logistic regression but with a different link function. Specifically, probit regression uses a probit link:

$$\Phi^{-1}(Pr[Y_i = 1|X_i]) = X_i^T\beta,$$

where Φ^{-1} is inverse of the CDF for the standard normal distribution. **Are the expected and observed information equivalent for probit regression?** Justify why or why not.