Likelihood Theory and Extensions (BIOS:7110)
Breheny

Assignment 10
Due: Monday, November 18

1. *Simulation study comparing Wald, score, and likelihood ratio tests.* This is a continuation of the problem "Score, Wald, and likelihood ratio tests for the rate parameter of the Gamma distribution." In particular, you should re-use your code from that problem – there is no need to start from scratch. In this problem, you will carry out simulations in which you generate samples of size 20 from the gamma distribution:

```
x <- rgamma(20, shape=2, rate = 1)
```

For each sample, you will then carry out Wald, likelihood ratio, and score tests using the code that you have written previously. For the simulations in (a) and (d) below, repeat this process for $N = 10,000$ independent samples.

   (a) Carry out Wald, likelihood ratio, and score tests of $H_0 : \beta = 1$. Report the observed type I error using $p < 0.05$ as a rejection threshold; note that this is 1 - the coverage of the 95% confidence interval, but the tests are easier to calculate.

   (b) Now, suppose we reparameterize the problem in terms of the scale parameter:

$$f(x|\alpha, \theta) = \frac{1}{\Gamma(\alpha)\theta^\alpha} x^{\alpha-1} e^{-x/\theta}.$$

   Re-derive the score and information matrix in terms of this new parameterization.

   (c) Is it possible for the observed information matrix to have negative values along the diagonal? What about the Fisher information? Explain how this could occur and whether it would cause problems for the tests in this problem.

   (d) Carry out Wald, likelihood ratio, and score tests of $H_0 : \theta = 1$. Report the observed type I error rate; note in this particular case, $\beta^* = \theta^* = 1$. Note that you can re-use much of the code you have already written; for example, $\hat{\alpha}$ is the same in both parameterizations.

2. *Newton's method and iteratively weighted least squares.* This problem explores the relationship between Newton's method and weighted least squares.

   (a) Suppose $\mathbf{y} \sim \mathrm{N}(\mathbf{X}\boldsymbol{\beta}, \mathbf{V})$, where $\mathbf{V}$ is a known positive definite variance-covariance matrix and $\mathbf{X}$ is a full-rank $n \times d$ design matrix. Derive the MLE of $\boldsymbol{\beta}$.

   (b) As you showed in a previous assignment, any (regular) likelihood can be approximated by that of a normal distribution. In the specific context of a GLM, where the likelihood of y_i depends on a linear predictor $\eta_i = \mathbf{x}_i^\top \boldsymbol{\beta}$, taking a Taylor series approximation about $\tilde{\boldsymbol{\eta}} = \mathbf{X}\widetilde{\boldsymbol{\beta}}$ yields expressions of the form

$$\ell(\boldsymbol{\beta}) \approx -\tfrac{1}{2}(\tilde{\mathbf{y}} - \mathbf{X}\boldsymbol{\beta})^\top \mathbf{W}(\tilde{\mathbf{y}} - \mathbf{X}\boldsymbol{\beta});$$

   derive $\tilde{\mathbf{y}}$ and $\mathbf{W}$ in the specific context of logistic regression. Note that:

- You are taking the Taylor series approximation of the likelihood with respect to the *linear predictor* $\boldsymbol{\eta}$, not the regression parameter $\boldsymbol{\beta}$.
- You have already derived the likelihood and taken many of the relevant derivatives in a much earlier problem, "Logistic regression derivatives"; there is no need to start from scratch here.

(c) In class, we derived the Newton update

$$\widehat{\boldsymbol{\beta}} = \widetilde{\boldsymbol{\beta}} + (\mathbf{X}^\top \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^\top (\mathbf{y} - \boldsymbol{\mu}),$$

by taking a first-order approximation to the score function about $\widetilde{\boldsymbol{\beta}}$ (recall that $\mathbf{W}$ and $\boldsymbol{\mu}$ were evaluated at $\widetilde{\boldsymbol{\beta}}$). Suppose we instead take the second-order Taylor series expansion of the log-likelihood as in (b), then solve for the MLE using our result from (a). Show that this MLE of the approximate likelihood is equal to the Newton update. Note: Again, there is no need to start from scratch; if you use the results from (a) and (b), this is a short problem.

(d) Code your own function to solve for the logistic regression MLE. I should be able to call the function as `logistic(X, y)`, where `X` is the design matrix and `y` the vector of responses. It should return the vector $\widehat{\boldsymbol{\beta}}$ and nothing else. As in previous assignments, turn in a separate .R file that defines the function (as well as any helper functions) and contains no other code.