GLM estimation and model fitting

Patrick Breheny

February 14

Introduction

- In previous lectures, we've discussed the theoretical properties of $\widehat{\boldsymbol{\beta}}$, the regression coefficients of a generalized linear model
- We turn our attention today to a more practical matter: how do we actually solve for $\widehat{\boldsymbol{\beta}}$?
- This is a more challenging question than it sounds – in general, there is no closed form solutions for the maximum likelihood estimator $\widehat{\boldsymbol{\beta}}$
- Nevertheless, it turns out that we can com the ideas from our last two lectures (Taylor series approximations and iteratively reweighted least squares) to obtain an algorithm for obtaining $\widehat{\boldsymbol{\beta}}$

MLEs for GLMs

- As we have discussed previously, we obtain MLEs by setting the score vector equal to $\mathbf{0}$
- Recall that for a GLM using the canonical link function, the score vector is

$$\mathbf{u}(\boldsymbol{\beta}) = \phi^{-1}\mathbf{X}^T(\mathbf{y} - \boldsymbol{\mu})$$

- Note that in the above equation, $\boldsymbol{\mu}$ is a function of $\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta}$; however, it need not be a linear function, and if it is not, we lack a closed-form solution for $\boldsymbol{\beta}$

Taylor approximation for μ

- Nevertheless, we can apply a Taylor series approach to obtain the following approximation about the point $\tilde{\boldsymbol{\beta}}$:

$$\boldsymbol{\mu} \approx \tilde{\boldsymbol{\mu}} + \mathbf{W}(\mathbf{X}\boldsymbol{\beta} - \mathbf{X}\tilde{\boldsymbol{\beta}}),$$

  where $\tilde{\boldsymbol{\mu}} = g^{-1}(\mathbf{X}\tilde{\boldsymbol{\beta}})$

- Note that the above result rests on the following proposition
- **Proposition:** If g is the canonical link, then

$$\frac{d}{d\eta} g^{-1}(\eta) = W(\eta)$$

Main result

- Thus, we obtain the following linear approximation to the score for $\boldsymbol{\beta}$:

$$\frac{\partial \ell}{\partial \boldsymbol{\beta}} \approx \phi^{-1}\mathbf{X}^T\mathbf{W}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta}),$$

  where $\mathbf{z} = \mathbf{X}\tilde{\boldsymbol{\beta}} + \mathbf{W}^{-1}(\mathbf{y}-\tilde{\boldsymbol{\mu}})$ is known as the *adjusted response*)
- Note that this approximation is based at $\tilde{\boldsymbol{\beta}}$ or, equivalently, $\tilde{\boldsymbol{\mu}}$, which are treated as constants in the above expression, thereby rendering the score equation linear in $\boldsymbol{\beta}$ after the approximation
- Again, recall that this approximation will be accurate near the fitted values $\tilde{\boldsymbol{\mu}}$, but not necessarily accurate far away from them

Updates

- As we saw previously, this gives the maximum likelihood estimate

$$\widehat{\boldsymbol{\beta}}^{(m)} = (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\mathbf{z}$$

- Note that $\mathbf{W}$ here plays the role of the weights in weighted least squares, and for that reason is often referred to as the *weight matrix*
- Again, recall that for the canonical link, $\mathbf{W}$ is entirely determined by the mean-variance relationship, and that it plays a prominent role in the variability of $\widehat{\boldsymbol{\beta}}$ as well
- Note that in the above equation, we require a superscript on $\widehat{\boldsymbol{\beta}}^{(m)}$ because this is a case of unknown weights, where $\mathbf{W}$ (and $\mathbf{z}$) will change depending on $\widehat{\boldsymbol{\beta}}$ and vice versa

IRLS algorithm

As we saw earlier, one way to address this problem is to iterate the process of reweight–estimate–reweight–estimate–. . . until convergence; this *iteratively reweighted least squares* (IRLS) algorithm is how generalized linear models are fit:

(1) Choose an initial value $\widehat{\boldsymbol{\beta}}^{(0)}$

(2) For $m = 0, 1, 2, \ldots$,

  (a) Calculate $\mathbf{z}$ and $\mathbf{W}$ based on $\widehat{\boldsymbol{\beta}}^{(m)}$

  (b) Solve for $\widehat{\boldsymbol{\beta}}^{(m+1)}$

  (c) Check to see whether $\widehat{\boldsymbol{\beta}}$ has converged; if yes, then stop

The Newton-Raphson algorithm

- This IRLS algorithm is a special case of a more general approach to optimization called the *Newton-Raphson* algorithm
- The Newton-Raphson algorithm calculates iterative updates via

$$\widehat{\boldsymbol{\beta}}^{(m+1)} = \widehat{\boldsymbol{\beta}}^{(m)} - \mathbf{H}^{-1}\mathbf{u},$$

  where $\mathbf{u}$ is the score vector and $\mathbf{H}$ is the Hessian matrix (the first and second derivatives of the log-likelihood, respectively), both of which are evaluated at $\widehat{\boldsymbol{\beta}}^{(m)}$
- It can be shown (homework) that this produces the same iterative updates as IRLS

Unique solutions and rank

- Recall that, for linear regression, a full-rank design matrix $\mathbf{X}$ implied that there was exactly one unique solution $\widehat{\boldsymbol{\beta}}$ which minimized the residual sum of squares
- A similar result holds for generalized linear models: if $\mathbf{X}$ is not full rank, then there is no unique solution which maximizes the likelihood

Additional issues for GLMs

- However, two additional issues arise in generalized linear models:
  - Although a unique solution exists, the IRLS algorithm is not guaranteed to find it
  - It is possible for the unique solution to be infinite, in which case the estimates are not particularly useful and inference breaks down
- The first issue is uncommon; we will see an example of the second issue in an upcoming lecture