Numerical methods for maximum likelihood estimation of GLM parameters

Amal Agarwal

STAT 511
Project Report

Pennsylvania State University

Supervisor: Prof. Ephraim Hanks

December 2014

Contents

Chapter 1

Literature Review

1.1 Maximum Likelihood Estimation

The theory of Maximum Likelihood Estimation (MLE) forms the basis of parameter estimation in frequentist framework of inference. The likelihood is basically defined as the probability model chosen to describe a given set of data. This model contains a set of unknown parameters which can be estimated by maximizing the likelihood function. By maximizing the likelihood function, we are essentially maximizing the probability of obtaining the given data under the specified model.

The theory can be explained better by taking an example. Let $X_1, X_2, ..., X_n$ be a random sample with realizations $x_1, x_2, ..., x_n$ and let the joint probability be $P(X_1 = x_1, ..., X_n = x_n|\theta)$ for the discrete case or the joint pdf $f(Y_1 = y_1, ..., Y_n = y_n|\theta)$ for the continuous case. Here $\boldsymbol{\theta} = (\theta_1, ..., \theta_p)^T$ is the parameter vector that has to estimated. For given realizations $x_1, ..., x_n$, this joint probability mass function or joint probability density function is considered as a function of θ and is called the likelihood, denoted by $L(\boldsymbol{\theta})$. The principle of maximum likelihood then says that the maximum likelihood estimate (MLE) $\hat{\boldsymbol{\theta}}$ for $\boldsymbol{\theta}$ is that value which maximizes $L(\boldsymbol{\theta})$

Maximizing the likelihood function is easy and analytically tractable only for some simple cases of probability distributions. However this is not always possible, in which case we have to numerically maximize the likelihood function. This project focusses on one such particular case that arises in the context on estimating parameters in Generalized Linear Models.

1.2 Parameter estimation in Generalized Linear Models[1]

In Generalized Linear Models (GLM's), we have the response variable Y_i from an exponential family of distributions $Y_i \sim f(Y_i|\theta_i)$ together with $g(E[Y_i]) = \eta_i$ where

$$f(Y_i|\theta_i) = exp\left(\frac{y_i\theta_i - b(\theta_i)}{\phi}w_i + c(y_i, \phi, w_i)\right) \tag{1.1}$$

g is the link function and $\eta_i = \boldsymbol{X_i^T}\boldsymbol{\beta}$ is the linear predictor.

The log-likelihood function is given as

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{n} l_i(\boldsymbol{\beta}) \tag{1.2}$$

This function depends on the regression parameters $\boldsymbol{\beta}$ through the natural parameter θ_i of the exponential family as

$$E[Y_i] = \mu_i = b'(\theta_i) = h(\boldsymbol{x_i^T}\boldsymbol{\beta}) \tag{1.3}$$

The log likelihood contribution of each observation $(y_i, \boldsymbol{x_i})$ is given by:

$$l_i(\boldsymbol{\beta}) = log(f(y_i|\boldsymbol{\beta})) = \frac{y_i\theta_i - b(\theta_i)}{\phi}w_i \tag{1.4}$$

Note that here the additive constant c has been dropped since it does not affect the maximization of log-likelihood.

In the following subsections, we derive the ML estimator in GLM's.

1.2.1 The score function

The score function $\boldsymbol{s}(\beta) = \partial l(\boldsymbol{\beta})/\partial\boldsymbol{\beta}$ can be obtained by applying the chain rule to the individual score function contributions as:

$$\begin{aligned} \boldsymbol{s_i}(\boldsymbol{\beta}) &= \frac{\partial l_i(\boldsymbol{\beta})}{\partial\boldsymbol{\beta}} \\ &= \frac{\partial\eta_i}{\partial\boldsymbol{\beta}}\frac{\partial\mu_i}{\partial\eta_i}\frac{\partial\theta_i}{\partial\mu_i}\frac{\partial(y_i\theta_i - b(\theta_i))}{\partial\theta_i}\frac{w_i}{\phi} \end{aligned} \tag{1.5}$$

The first term is given as

$$\frac{\partial\eta_i}{\partial\boldsymbol{\beta}} = \boldsymbol{x_i} \tag{1.6}$$

The second term can be simplified as

$$\frac{\partial\mu_i}{\partial\eta_i} = \frac{\partial h(\eta_i)}{\partial\eta_i} = h'(\eta_i) = d_i(say) \tag{1.7}$$

where $h = g^{-1}$ is the response function.

The third term is obtained by first computing the reciprocal as

$$\frac{\partial\mu_i}{\partial\theta_i} = \frac{\partial b'(\theta_i)}{\partial\theta_i} = b''(\theta_i) = \frac{w_iVar(y_i)}{\phi} = \frac{w_i\sigma_i^2}{\phi} \tag{1.8}$$

and therefore

$$\frac{\partial\theta_i}{\partial\mu_i} = \frac{\phi}{w_i\sigma_i^2} \tag{1.9}$$

The fourth term gets simplified as

$$\frac{\partial(y_i\theta_i - b(\theta_i))}{\partial\theta_i} = y_i - b'(\theta_i) = y_i - \mu_i \tag{1.10}$$

Putting all the pieces together we get the score function as

$$\boldsymbol{s}(\boldsymbol{\beta}) = \sum_i \boldsymbol{x_i}d_i\frac{\phi}{w_i\sigma_i^2}(y_i - \mu_i)\frac{w_i}{\phi} = \sum_i \boldsymbol{x_i}\frac{d_i}{\sigma_i^2}(y_i - \mu_i) \tag{1.11}$$

Now define $\boldsymbol{y} = (y_1, ..., y_n)^T$, $\boldsymbol{\mu} = (\mu_1, ..., \mu_n)^T$ and $D = diag(d_1, ..., d_n)$, $\Sigma = diag(\sigma_1^2), ..., \sigma_n^2)$ Using these, we obtain

$$\boldsymbol{s}(\boldsymbol{\beta}) = \boldsymbol{X^T D\Sigma^{-1}(y - \mu)} \tag{1.12}$$

1.2.2 The information matrix

Fisher matrix is given as

$$\begin{aligned}\boldsymbol{F}(\boldsymbol{\beta}) &= E[\boldsymbol{s}(\boldsymbol{\beta})\boldsymbol{s'}(\boldsymbol{\beta})] \\ &= \sum_i E[\boldsymbol{s_i}(\boldsymbol{\beta})\boldsymbol{s_i'}(\boldsymbol{\beta})] \\ &= \sum_i E[\boldsymbol{x_i x_i'}\frac{d_i^2}{(\sigma_i^2)^2}(y_i-\mu_i)^2] \\ &= \sum_i \boldsymbol{x_i x_i'}\frac{d_i^2}{(\sigma_i^2)^2}E[(y_i-\mu_i)^2] \\ &= \sum_i \boldsymbol{x_i x_i'}\frac{d_i^2}{(\sigma_i^2)^2}Var(y_i) \\ &= \sum_i \boldsymbol{x_i x_i'}\frac{d_i^2}{\sigma_i^2} \\ &= \sum_i \boldsymbol{x_i x_i'}\tilde{w_i}\end{aligned} \tag{1.13}$$

where $\tilde{w_i}$ are the working weights and depend on $\boldsymbol{\beta}$ as

$$\tilde{w_i} = \frac{d_i^2}{\sigma_i^2} = (h'(\eta_i))^2\frac{w_i}{b''(\theta_i)\phi} \tag{1.14}$$

In matrix notation, $\boldsymbol{F}(\boldsymbol{\beta})$ can be written as

$$\boldsymbol{F}(\boldsymbol{\beta}) = \boldsymbol{X^T W X} \tag{1.15}$$

where $\boldsymbol{W} = diag(\tilde{w_1}, ..., \tilde{w_n})$ is the diagonal matrix of working weights. Note $\boldsymbol{W} = \boldsymbol{D^2\Sigma^{-1}}$.

1.2.3 The Fisher scoring algorithm

Computation of the ML estimator $\hat{\boldsymbol{\beta}}$ is usually based on the Fisher scoring algorithm with the following iterative procedure:

$$\hat{\boldsymbol{\beta}}^{(\boldsymbol{t+1})} = \hat{\boldsymbol{\beta}}^{(\boldsymbol{t})} + \boldsymbol{F}^{-\boldsymbol{1}}(\hat{\boldsymbol{\beta}}^{(\boldsymbol{t})})\boldsymbol{s}(\hat{\boldsymbol{\beta}}^{(\boldsymbol{t})}), \quad t = 0, 1, 2, ... \tag{1.16}$$

Chapter 2

Simulations

2.1 Poisson Regression

2.1.1 Fisher scoring algorithm for Poisson model with log link

For the Poisson Regression, the GLM model is given as

$$Y_i \sim Poisson(\lambda_i) \tag{2.1}$$

$$g(E[Y_i]) = log(E[Y_i]) = log(\lambda_i) = \eta_i = \boldsymbol{X_i^T \beta} \tag{2.2}$$

where we have used the canonical log link.

In this case the response function is $h(\eta_i) = e^{\eta_i}$. Thus

$$d_i = h'(\eta_i) = \frac{\partial e^{\eta_i}}{\partial \eta_i} = e^{\eta_i} \tag{2.3}$$

This defines the diagonal matrix $\boldsymbol{D}$ which will be used to calculate the score function.

The variance of Y_i is given as

$$\sigma_i^2 = \lambda_i = e^{\eta_i} \tag{2.4}$$

Thus here we have $\boldsymbol{\Sigma} = \boldsymbol{D}$. The score function becomes

$$\boldsymbol{s(\beta)} = \boldsymbol{X^T D \Sigma^{-1} (y - \lambda)} = \boldsymbol{X^T}(\boldsymbol{y} - e^{\boldsymbol{X^T \beta}}) \tag{2.5}$$

Also the working weights matrix is given as

$$\boldsymbol{W} = \boldsymbol{D^2 \Sigma^{-1}} = \boldsymbol{D} \tag{2.6}$$

The Fisher matrix becomes

$$\boldsymbol{F(\beta)} = \boldsymbol{X^T W X} = \boldsymbol{X^T D X} \tag{2.7}$$

2.1.2 Results of the simulation study

Using the above score function and Fisher matrix, the Fisher algorithm was run on a set of data generated from the Poisson model (see the comments in attached R code for details). The convergence criteria was set as

$$\sup_{k=\{1,2,\ldots,p\}} |\beta_k^{(i)} - \beta_k| \le \epsilon \tag{2.8}$$

where $\beta_k^{(i)}$ is the kth component of the ith iterate of the parameter vector, β_k is the kth component of the true parameter vector and ϵ was set at 10^{-2}.

For a sample size of 1000 and 3 predictors, convergence with the above criteria was obtained in 6 iterations.

2.2 Logistic Regression

2.2.1 Fisher scoring algorithm for Bernoulli model with logit link

For the Bernoulli Regression, the GLM model is given as

$$Y_i \sim Bernoulli(p_i) \tag{2.9}$$

$$g(E[Y_i]) = log\left(\frac{(E[Y_i]}{1 - E[Y_i]}\right) = log\left(\frac{p_i}{1 - p_i}\right) = \eta_i = \boldsymbol{X_i^T\beta} \tag{2.10}$$

where we have used the canonical logit link.

In this case the response function is $h(\eta_i) = \dfrac{e^{\eta_i}}{1 + e^{\eta_i}}$. Thus

$$d_i = h'(\eta_i) = \frac{\partial}{\partial\eta_i}\left(\frac{e^{\eta_i}}{1 + e^{\eta_i}}\right) = \frac{e^{\eta_i}}{(1 + e^{\eta_i})^2} \tag{2.11}$$

This defines the diagonal matrix $\boldsymbol{D}$ which will be used to calculate the score function.

The variance of Y_i is given as

$$\sigma_i^2 = p_i(1 - p_i) = \frac{e^{\eta_i}}{(1 + e^{\eta_i})^2} \tag{2.12}$$

Thus here we have $\boldsymbol{\Sigma = D}$. The score function becomes

$$\boldsymbol{s(\beta)} = \boldsymbol{X^T D \Sigma^{-1}(y - \lambda)} = \boldsymbol{X^T}(\boldsymbol{y} - e^{\boldsymbol{X^T\beta}}) \tag{2.13}$$

Also the working weights matrix is given as

$$\boldsymbol{W} = \boldsymbol{D^2\Sigma^{-1}} = \boldsymbol{D} \tag{2.14}$$

The Fisher matrix becomes

$$\boldsymbol{F(\beta)} = \boldsymbol{X^T W X} = \boldsymbol{X^T D X} \tag{2.15}$$

2.2.2 Results of the simulation study

Using the above score function and Fisher matrix, the Fisher algorithm was run on a set of data generated from the Bernoulli model. With the aforementioned convergence criteria, the iterates don't converge to the true parameter vector. Therefore the following convergence criteria was used to stop the iterations:

$$\sup_{k=\{1,2,\ldots,p\}} |\beta_k^{(i)} - \beta_k^{(i-1)}| \leq \epsilon \tag{2.16}$$

Here $\beta_k^{(i)}$ is the kth component of the ith iterate of the parameter vector, $\beta_k^{(i-1)}$ is the kth component of the $(i-1)$th iterate of the parameter vector and ϵ was set at 10^{-2}.

Again for a sample size of 1000 and 3 predictors, convergence with the above criteria was obtained in 4 iterations. It is worthwhile to note here that increasing the sample size to 10^4 does make the $\sup_{k=\{1,2,\ldots,p\}} |\beta_k^{(i)} - \beta_k^{(i-1)}| < 0.05$ which indicates that it might be possible to satisfy the earlier convergence criteria for large enough sample sizes.

Chapter 3

Conclusion

Fisher scoring algorithm was applied to numerically estimate the parameters in Poisson model with log link and Bernoulli model with logit link. The results of the simulation study shows that it takes approximately order 1 number of iterations to meet the convergence criteria in Poisson model for sample size of order 3. However for Bernoulli model, this convergence criteria does not work and the iterates never converge to the true vector precisely for sample size of order 3. This can be attributed to the fact that in the Bernoulli model we have response taking only two values 0 and 1 and so the information content in each observation is lower compared to the Poisson case where the response can take any value greater than zero. Increasing the sample size to order 4 does help in Bernoulli model and we obtain better convergence to the true parameter vector. Order 5 sample sizes could not be taken since it goes beyond this computer's resources. Since Fisher scoring algorithm gives an unbiased estimator, these results are expected.

Chapter 4

Future Scope

- Fisher scoring method can be applied to other possible distributions of the response variable in exponential family with canonical links.
- The rate of covergence can be compared with Newton Raphson algorithm in each case.
- Methods to estimate the paramter vectors for large sample sizes can be studied and implemented for the Bernoulli model, thus obtaining better convergence to the true parameter vector.

Bibliography

[1] Ludwig Fahrmeir, Thomas Kneib, Stefan Lang, Brian Marx *Regression: Models, Methods and Applications* 2013, SpringerLink.