Some explanations about the IWLS algorithm to fit generalized linear models

Christophe Dutang

Some explanations about the IWLS algorithm to fit generalized linear models

Christophe Dutang

Laboratoire Manceau de Mathématiques, Le Mans Université, France

August 2017

This short note focuses on the estimation procedure generally used for generalized linear models (GLMs), see e.g. McCullagh, P. (1984). Generalized linear models. European Journal of Operational Research, 16(3), 285-292.

1 Fitting GLMs

1.1 Definition of the log-likelihood and the score function

The parametrization of the exponential family generally used for GLMs is given by the following density or mass probability function:

$$f_Y(y;\theta,\phi) = e^{\frac{y\theta - b(\theta)}{a(\phi)} + c(y,\phi)}, y \in S,$$

where S is the support of the distribution, typically $\mathbb{N}$ or $\mathbb{R}$, and a, b, c are known smooth functions.

Note that $E(Y) = b'(\theta) = \mu$ and $Var(Y) = \phi b''(\theta) = \phi V(\mu)$. Let us start with the iid case, where Y_i are independent and identically distributed. In that case, the score is defined as

$$S(\theta) = \frac{\partial \log f_Y(Y;\theta,\phi)}{\partial\theta} = \frac{Y - b'(\theta)}{a(\phi)}.$$

It is well known that $E(S) = 0$ and $Var(S) = -E(S'(\theta)) = b''(\theta)/\phi$.

Now, we focus on the GLM context. That is $Y_i \sim \mathcal{F}_{exp}(\theta_i, \phi_i)$ for all $i = 1, \dots, n$ where the explanatory variables are linked to the expectation by

$$g(b'(\theta_i)) = g(\mu_i) = \beta_1 x_{i1} + \cdots + \beta_p x_{ip},$$

with $p < n$ for identifiability reasons. Note that an intercept is generally included so that $x_{i1} = 1$ for all i. The log-density of Y_i is

$$l_i(\beta_i) = \log f_{Y_i}(y_i;\theta_i(\beta_i),\phi_i) = \frac{y_i\theta_i(\beta_i) - b(\theta_i(\beta_i))}{a(\phi_i)} + c(y_i,\phi_i).$$

The log-likelihood of the GLM for observations $y_1, \dots, y_n$ is simply obtained by adding l_i contributions

$$L(\beta) = \sum_{i=1}^n l_i(\beta_i) = \sum_{i=1}^n \left(\frac{y_i\theta_i(\beta_i) - b(\theta_i(\beta_i))}{a(\phi_i)} + c(y_i,\phi_i) \right).$$

A common choice for the dispersion parameter is $\phi_i = \phi/w_i$ with w_i a known weigth.

The score function is defined as the expectation of the gradient of the log-likelihood. Using $\theta_i = (b')^{-1}\left(g^{-1}(\beta_1 x_{i1} + \cdots + \beta_p x_{ip})\right)$, $\eta_i = \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$, $(f^{-1})' = 1/f' \circ f^{-1}$, $((f')^{-1})' = 1/f'' \circ (f')^{-1}$, we derive the partial derivative

$$\frac{\partial \theta_i}{\partial \beta_j} = ((b')^{-1})'(g^{-1}(\eta_i)) \times (g^{-1})'(\eta_i) \times x_{ij} = \frac{1}{b''((b')^{-1}(g^{-1}(\eta_i)))} \times \frac{x_{ij}}{g'(g^{-1}(\eta_i))} = \frac{1}{b''(\theta_i)} \times \frac{x_{ij}}{g'(\mu_i)}.$$

Therefore, using this partial derivative w.r.t. β_j leads to the following score

$$S_j(\beta) = \frac{\partial L(\beta)}{\partial \beta_j} = \sum_{i=1}^{n} U_i(\theta_i) \frac{\partial \theta_i}{\partial \beta_j} = \sum_{i=1}^{n} \frac{y_i - b'(\theta_i)}{a(\phi_i)} \frac{1}{b''(\theta_i)} \frac{x_{ij}}{g'(\mu_i)} = \sum_{i=1}^{n} \frac{y_i - \mu_i}{a(\phi_i) V(\mu_i)} \frac{x_{ij}}{g'(\mu_i)}$$

where $\mu_i = b'(\theta_i)$ and $V(\mu_i) = b''(\theta_i)$ for $j = 1, \ldots, p$. The parameter β is found by solving the score equations

$$S_j(\beta) = 0, \ j = 1, \ldots, p.$$

1.2 Objective of the optimization procedure

The question we may ask is whether it is equivalent to solve the score function or to minimize the opposite of the log-likelihood by the (exact) Newton method?

Consider $f : \mathbb{R}^n \mapsto \mathbb{R}$ a twice differentiable function with a gradient vector $g(x) = \nabla f(x)$, and a Hessian matrix $H(x) = \nabla^2 f(x)$. Let $F : \mathbb{R}^n \mapsto \mathbb{R}^n$ be a differentiable function. The Jacobian matrix is denoted by $\mathrm{Jac}\, F(x) \in \mathbb{R}^{n \times n}$.

From classical optimization books, e.g. Nocedal, J. & Wright, S. J. (2006), Numerical Optimization, Springer Science+Business Media, a (local) optimization method consists in computing the following sequence $x_{k+1} = x_k + d_k$ where d_k is computed according to a scheme. In addition, a globalization technique may be used in conjunction such as a line search. But, the globalization scheme is seldom done for fitting GLMs.

The exact Newton method (also called the Newton-Raphson method) to find the minimum of a function f uses the direction $d_k = -H(x_k)^{-1} g(x_k)$. In comparison, the steepest descent method to find the minimum of f considers $d_k = -g(x_k)$. Furthermore, the exact Newton method to find the root of F uses the direction $d_k = -\mathrm{Jac}\,(x_k)^{-1} F(x_k)$. Hence, the direction is exactly the same between the minimization problem and the root problem, when the root function F is the gradient ∇f of the objective. Hence, finding the roots of the score equations is equivalent to maximizing the log-likelihood.

1.3 Derivation of the Newton method for the score equations

The Newton method to find the root of the score equations is

$$\beta^{(k+1)} = \beta^{(k)} - \mathrm{Jac}\, S\left(\beta^{(k)}\right)^{-1} S(\beta^{(k)}).$$

The exponent $^{(k)}$ is used to denote the kth iteration since subscript are used for indexing observation and/or component. Let us compute the Jacobian of the score or the Hessian of the log-likelihood.

$$\begin{aligned}\frac{\partial^2 L(\beta)}{\partial \beta_j \partial \beta_l} &= \sum_{i=1}^{n} \frac{\partial}{\partial \beta_l}\left(\frac{y_i - b'(\theta_i)}{a(\phi_i)}\right) \frac{1}{b''(\theta_i)} \frac{x_{ij}}{g'(\mu_i)} + \sum_{i=1}^{n} \frac{\partial}{\partial \beta_l}\left(\frac{1}{b''(\theta_i)}\right) \frac{y_i - b'(\theta_i)}{a(\phi_i)} \frac{x_{ij}}{g'(\mu_i)} \\ &\quad + \sum_{i=1}^{n} \frac{\partial}{\partial \beta_l}\left(\frac{x_{ij}}{g'(\mu_i)}\right) \frac{y_i - b'(\theta_i)}{a(\phi_i)} \frac{1}{b''(\theta_i)}.\end{aligned}$$

The first term is

$$\frac{\partial}{\partial \beta_l}\left(\frac{y_i - b'(\theta_i)}{a(\phi_i)}\right) = \frac{-b''(\theta_i)}{a(\phi_i)} \times \frac{\partial \theta_i}{\partial \beta_l} = \frac{-b''(\theta_i)}{a(\phi_i)} \times \frac{1}{b''(\theta_i)} \times \frac{x_{il}}{g'(\mu_i)} = \frac{-x_{il}}{a(\phi_i) g'(\mu_i)}.$$

The second term is

$$\frac{\partial}{\partial\beta_l}\left(\frac{1}{b''(\theta_i)}\right) = -\frac{\partial(b''(\theta_i))}{\partial\beta_l}\frac{1}{(b''(\theta_i))^2} = -\frac{b'''(\theta_i)}{(b''(\theta_i))^2}\frac{\partial\theta_i}{\partial\beta_l} = -\frac{b'''(\theta_i)}{(b''(\theta_i))^3}\frac{x_{il}}{g'(\mu_i)} = -\frac{b'''(\theta_i)}{(V(\mu_i))^3}\frac{x_{il}}{g'(\mu_i)}.$$

The third term is

$$\frac{\partial}{\partial\beta_l}\left(\frac{x_{ij}}{g'(\mu_i)}\right) = -\frac{x_{ij}}{(g'(\mu_i))^2}\frac{\partial(g'(\mu_i))}{\partial\beta_l} = -\frac{x_{ij}g''(\mu_i)}{(g'(\mu_i))^2}\frac{\partial\mu_i}{\partial\beta_l} = -\frac{x_{ij}x_{il}g''(\mu_i)}{(g'(\mu_i))^3},$$

since

$$\frac{\partial\mu_i}{\partial\beta_l} = \frac{\partial(b'(\theta_i))}{\partial\beta_l} = b''(\theta_i)\frac{\partial\theta_i}{\partial\beta_l} = b''(\theta_i)\times\frac{1}{b''(\theta_i)}\times\frac{x_{il}}{g'(\mu_i)} = \frac{x_{il}}{g'(\mu_i)}.$$

Recalling that the Hessian matrix is defined as

$$H(\beta, y_1, \ldots, y_n) = \left(\frac{\partial^2 L(\beta)}{\partial\beta_j\partial\beta_l}\right)_{j,l},$$

and using $b''(\theta) = V(\mu)$, we get

$$\begin{aligned}
\frac{\partial^2 L(\beta)}{\partial\beta_j\partial\beta_l} &= -\sum_{i=1}^{n}\frac{x_{il}}{a(\phi_i)g'(\mu_i)}\frac{1}{b''(\theta_i)}\frac{x_{ij}}{g'(\mu_i)} - \sum_{i=1}^{n}\frac{b'''(\theta_i)}{(V(\mu_i))^3}\frac{x_{il}}{g'(\mu_i)}\frac{y_i - b'(\theta_i)}{a(\phi_i)}\frac{x_{ij}}{g'(\mu_i)} \\
&\quad -\sum_{i=1}^{n}\frac{x_{ij}x_{il}g''(\mu_i)}{(g'(\mu_i))^3}\frac{y_i - b'(\theta_i)}{a(\phi_i)}\frac{1}{b''(\theta_i)} \\
&= -\sum_{i=1}^{n}\frac{x_{il}x_{ij}}{a(\phi_i)(g'(\mu_i))^2V(\mu_i)} - \sum_{i=1}^{n}\frac{b'''(\theta_i)x_{il}x_{ij}(y_i-\mu_i)}{(V(\mu_i))^3(g'(\mu_i))^2a(\phi_i)} - \sum_{i=1}^{n}\frac{x_{ij}x_{il}g''(\mu_i)(y_i-\mu_i)}{(g'(\mu_i))^3\mu_i a(\phi_i}).
\end{aligned}$$

In practice, we use the expectation of this matrix w.r.t. the random variable Y_i. This procedure is known as the Fisher scoring method. Hence, two terms will cancel because $E(Y_i) = \mu_i$. So

$$\bar{H}(\beta) = E(H(\beta, Y_1, \ldots, Y_n)) = \left(-\sum_{i=1}^{n}\frac{x_{il}x_{ij}}{a(\phi_i)(g'(\mu_i))^2V(\mu_i)}\right)_{j,l}.$$

This matrix can be rewritten as the product of three matrices $\bar{H}(\beta) = -X^TW(\beta)X$ where

$$W(\beta) = \begin{pmatrix} \frac{1}{a(\phi_1)(g'(\mu_1))^2V(\mu_1)} & & \\ & \ddots & \\ & & \frac{1}{a(\phi_n)(g'(\mu_n))^2V(\mu_n)} \end{pmatrix}, X = \begin{pmatrix} x_{11} & \ldots & x_{1p} \\ & \vdots & \\ x_{n1} & \ldots & x_{np} \end{pmatrix}.$$

The (expected) Newton method is

$$\beta^{(k+1)} = \beta^{(k)} + \left(X^TW(\beta^{(k)})X\right)^{-1}S(\beta^{(k)}).$$

Let us write matricially the score vector

$$S_j(\beta) = \sum_{i=1}^{n}\frac{y_i-\mu_i}{\phi_iV(\mu_i)}\frac{x_{ij}}{g'(\mu_i)} = \sum_{i=1}^{n}(y_i-\mu_i)x_{ij}g'(\mu_i)\times\frac{1}{\phi_i(g'(\mu_i))^2V(\mu_i)} = X^TW(\beta)\tilde{Y}(\beta)$$

where we define a new vector $\tilde{Y}(\beta) = \big((y_i-\mu_i)g'(\mu_i)\big)_i \in \mathbb{R}^n$. The (expected) Newton method can be reformulated as

$$\beta^{(k+1)} = \beta^{(k)} + \left(X^TW(\beta^{(k)})X\right)^{-1}X^TW(\beta^{(k)})\tilde{Y}(\beta^{(k)}).$$

1.4 Reformulation as an iterative weighted least square (IWLS) problem

Let us rewrite β as a matrix product

$$\beta = \left(X^T W(\beta^{(k)}) X\right)^{-1} X^T W(\beta^{(k)}) X\beta = \left(X^T W(\beta^{(k)}) X\right)^{-1} X^T W(\beta^{(k)}) \tilde{X},$$

where $\tilde{X}(\beta) = X\beta$ is the vector of linear predictor η_i. In other words, the (expected) Newton method can be factorized as

$$\beta^{(k+1)} = \left(X^T W(\beta^{(k)}) X\right)^{-1} X^T W(\beta^{(k)}) \left(\tilde{X}(\beta^{(k)}) + \tilde{Y}(\beta^{(k)})\right) = \left(X^T W(\beta^{(k)}) X\right)^{-1} X^T W(\beta^{(k)}) Z(\beta^{(k)})$$

with a new vector $Z(\beta) = \left(\eta_i(\beta) + (y_i - \mu_i(\beta))\, g'(\mu_i(\beta))\right)_i$.

That is $\beta^{(k+1)}$ is the solution of a weighted least square problem with weights $W^{(k)}$, response vector $Z^{(k)}$ and explanatory variable $X^{(k)}$.

1.5 The IWLS Algorithm

The iterative weighted least square algorithm used to fit GLM is as follows

1. Initialization:
   (a) Use original data with a small shift $\mu_i^{(0)} = y_i + 0.1$ to compute $\eta_i^{(0)} = g(\mu_i^{(0)})$.
   (b) Compute working responses $Z^{(0)} = (\eta_i^{(0)} + (y_i - \mu_i^{(0)}) g'(\mu_i^{(0)}))_i$.
   (c) Compute working weights $W^{(0)} = \text{diag}(w_1, \ldots, w_n)$ and $w_i = \frac{1}{a(\phi_i)(g'(\mu_i^{(0)}))^2 V(\mu_i^{(0)})}$.
   (d) Solve the system to get $\beta^{(0)}$
   $$X^T W^{(0)} X\beta^{(0)} = X^T W^{(0)} Z^{(0)}.$$
2. Iteration: for $k = 1, \ldots, m$ do
   (a) Compute working responses $Z^{(k)} = (z_i)_i$ and $z_i = \eta_i(\beta^{(k)}) + (y_i - \mu_i(\beta^{(k)}))g'(\mu_i(\beta^{(k)}))$.
   (b) Compute working weights $W^{(k)} = \text{diag}(w_1, \ldots, w_n)$ and $w_i = \frac{1}{a(\phi_i)(g'(\mu_i(\beta^{(k)}))^2 V(\mu_i(\beta^{(k)}))}$.
   (c) Solve the system to get $\beta^{(k+1)}$
   $$X^T W^{(k)} X\beta^{(k+1)} = X^T W^{(k)} Z^{(k)}.$$
   (d) Verify convergence on the deviance: $||Dev(\beta^{(k+1)}) - Dev(\beta^{(k)})|| \leq \epsilon$.

In practice the linear system $X^T W^{(k)} X\beta^{(k+1)} = X^T W^{(k)} Z^{(k)}$ is solved via a QR decomposition, see e.g. Green (1984).

2 Numerical illustration

In this section, we carry out simple examples of GLMs on simulated datasets in the R statistical software, R Core Team (2017), R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

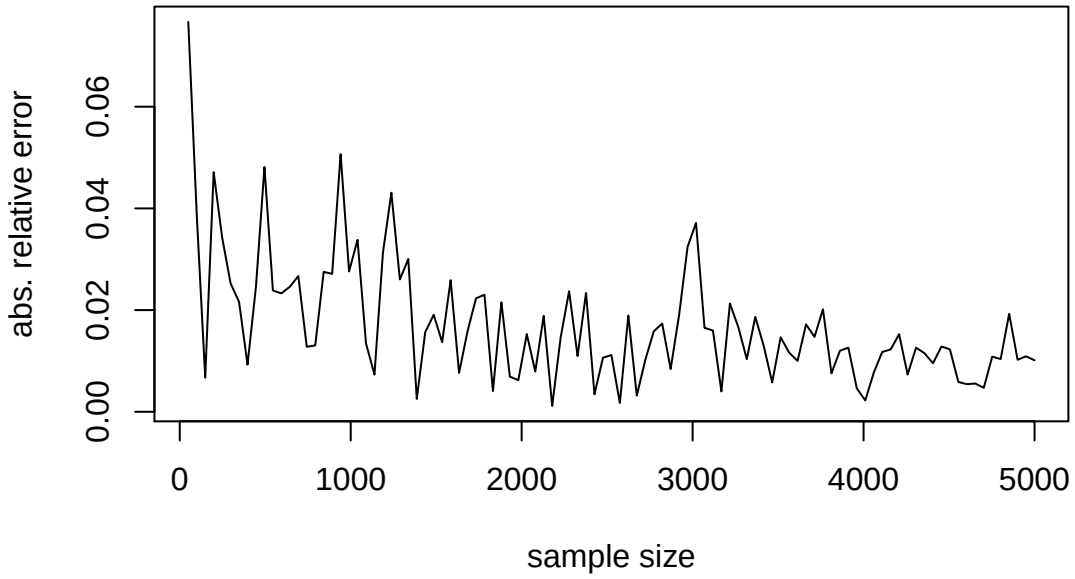
2.1 Poisson regression

A Poisson distribution has the following probabilty mass function $P(X = x) = \lambda^x e^{-\lambda}/x!$ for $x \in \mathbb{N}$. We rewrite as

$$\log(f(x)) = x\log(\lambda) - \log(x!) - \lambda = \frac{x\log(\lambda) - \lambda}{1} - \log(x!).$$

So $\theta = \log(\lambda) \Leftrightarrow \lambda = e^\theta$, $b(x) = e^x$, $\phi = 1$, $a(x) = x$ and $c(x, \phi) = -\log(x!)$. In particular $(b')^{-1}(x) = log(x)$.

Below we make a simple Poisson regression with a single categorical variable where an explicit solution exists. We plot the absolute relative error of the GLM estimator.



2.2 Gamma regression

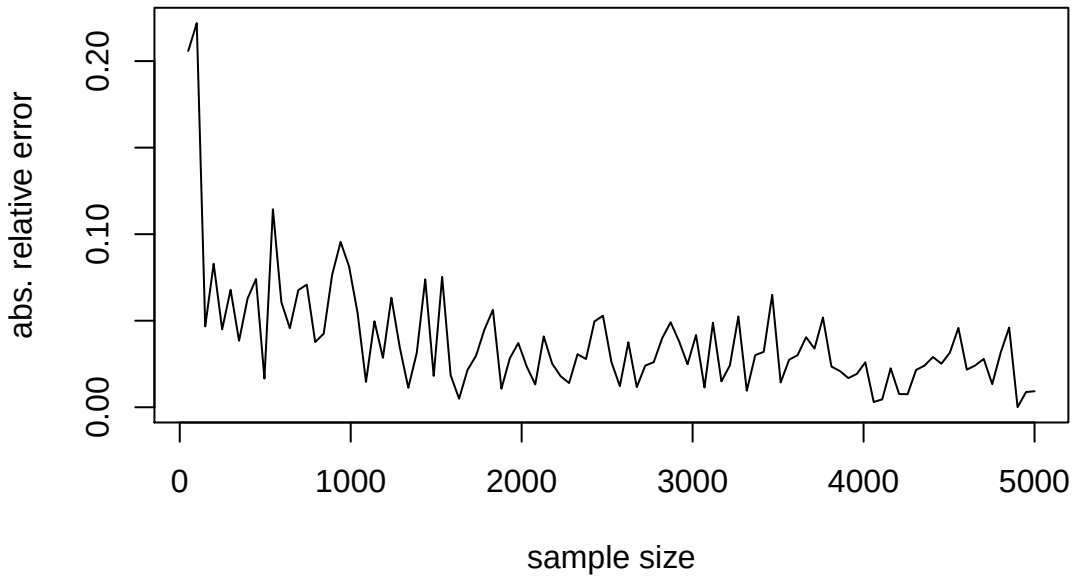
A gamma distribution has the following density function $f(x) = \frac{\lambda^\alpha x^{\alpha-1} e^{-\lambda x}}{\Gamma(\alpha)}$ for $x \in \mathbb{X} = \mathbb{R}_+$, $\lambda, \alpha > 0$. We rewrite as

$$\log(f(x)) = \frac{\frac{-\lambda}{\alpha}x - (-\log(-\frac{-\lambda}{\alpha}))}{1/\alpha} + \alpha\log(\alpha) + (\alpha - 1)\log(x) - \log(\Gamma(\alpha))$$

So $\theta = \frac{-\lambda}{\alpha}$, $\Theta = \mathbb{R}_-$, $\phi = 1/\alpha$, $a(x) = x$, $b(x) = -\log(-x)$ and

$$c(x, \phi) = \log(1/\phi)/\phi + (1/\phi - 1)\log(x) - \log(\Gamma(1/\phi)).$$

In particular $(b')^{-1}(x) = 1/x$. Below we make a simple gamma regression with a single categorical variable where an explicit solution exists. We plot the absolute relative error of the GLM estimator.


abs. relative error
0.00
0.10
0.20
0
1000
2000
3000
4000
5000
sample size