

STAT 238 - Bayesian Statistics

Lecture Seventeen

Spring 2026, UC Berkeley

Aditya Guntuboyina

04 March 2026

1 Logistic Regression

We are again in the usual regression setting where we observe data $(y_i, x_{i1}, x_{i2}, \dots, x_{im})$ for $i = 1, \dots, n$. There are m explanatory variables x_1, \dots, x_m and one response variable. x_{ij} denotes the value of the j^{th} explanatory variable for the i^{th} individual and y_i is the value of the response variable for the i^{th} individual. Suppose now that the response variable is binary i.e., y_1, \dots, y_n take values in $\{0, 1\}$. In this case, the logistic regression model assumes that:

$$y_i \stackrel{\text{independent}}{\sim} \text{Bernoulli} \left(\frac{\exp(\beta_0 + \beta_1 x_{i1} + \dots + \beta_m x_{im})}{1 + \exp(\beta_0 + \beta_1 x_{i1} + \dots + \beta_m x_{im})} \right) \quad \text{for } i = 1, \dots, n.$$

Letting $x_i = (1, x_{i1}, \dots, x_{im})^T$ and $\beta = (\beta_0, \beta_1, \dots, \beta_m)^T$, we can write the model also as

$$y_i \stackrel{\text{independent}}{\sim} \text{Bernoulli} \left(\frac{\exp(x_i^T \beta)}{1 + \exp(x_i^T \beta)} \right) \quad \text{for } i = 1, \dots, n$$

where β is the $(m+1) \times 1$ vector with components $\beta_0, \beta_1, \dots, \beta_m$.

This gives the likelihood:

$$\text{Likelihood} = \prod_{i=1}^n \left(\frac{\exp(x_i^T \beta)}{1 + \exp(x_i^T \beta)} \right)^{y_i} \left(1 - \frac{\exp(x_i^T \beta)}{1 + \exp(x_i^T \beta)} \right)^{1-y_i}$$

Suppose x_1^T, \dots, x_n^T form the rows of the $n \times p$ design matrix X (where $p = m+1$). The unknown parameters in the logistic regression model are β_0, \dots, β_m (note that, in contrast to the linear regression model, there is no σ parameter in logistic regression). As in linear regression, we use the prior:

$$\beta_0, \beta_1, \dots, \beta_m \stackrel{\text{i.i.d}}{\sim} \text{Unif}(-C, C) \quad (1)$$

for a large C . The posterior of β is then

$f_{\beta|\text{data}}(\beta) \propto \text{Likelihood} \times \text{prior}$

$$\begin{aligned} &\propto \left[\prod_{i=1}^n \left(\frac{\exp(x_i^T \beta)}{1 + \exp(x_i^T \beta)} \right)^{y_i} \left(1 - \frac{\exp(x_i^T \beta)}{1 + \exp(x_i^T \beta)} \right)^{1-y_i} \right] I\{\beta_0, \beta_1, \dots, \beta_p \in (-C, C)\} \\ &= \left[\prod_{i=1}^n \frac{\exp(y_i x_i^T \beta)}{1 + \exp(x_i^T \beta)} \right] I\{\beta_0, \beta_1, \dots, \beta_p \in (-C, C)\} \\ &= [\exp(\ell(\beta))] I\{\beta_0, \beta_1, \dots, \beta_p \in (-C, C)\} \end{aligned}$$

where

$$\ell(\beta) := \sum_{i=1}^n [y_i(x_i^T \beta) - \log(1 + \exp(x_i^T \beta))].$$

Note that $\ell(\beta)$ is simply the log-likelihood in this problem. This posterior density is not in standard form involving some well-known densities. If $p = 1$ or $p = 2$, then this can be plotted. One can use various MCMC techniques to obtain samples from this posterior.

Instead of MCMC, we shall derive a closed form multivariate normal approximation that works quite well in practice. This method is called Laplace Approximation or Posterior Normal Approximation. It turns out that Bayesian inference from this normal approximation to the posterior coincides with usual frequentist inference for logistic regression.

To get the normal approximation, first let us drop the indicator which will be irrelevant when C is large to get

$$f_{\beta|Y=y}(\beta) \propto \exp(\ell(\beta)).$$

The normal approximation will be obtained by a second-order Taylor expansion of $\ell(\beta)$. We shall do the Taylor expansion around the MLE $\hat{\beta}$ because the posterior is peaked at $\hat{\beta}$ and the high regions of the posterior are most likely very close to $\hat{\beta}$. Recall that the MLE $\hat{\beta}$ is defined as the maximizer of the likelihood (or log-likelihood):

$$\hat{\beta} := \operatorname{argmax}_{\beta \in \mathbb{R}^p} \ell(\beta).$$

It is obtained by taking the gradient of the log-likelihood and solving the equation obtained by setting the gradient to zero. It is easy to check that the gradient of the log-likelihood is

$$\nabla \ell(\beta) = \sum_{i=1}^n \left(y_i - \frac{\exp(x_i^T \beta)}{1 + \exp(x_i^T \beta)} \right) x_i. \quad (2)$$

To get the maximum likelihood estimator $\hat{\theta}$, we need to set the gradient above to zero and solve the resulting equation for θ . This cannot be done in closed form and the usual method is to use an iterative scheme such as Newton's algorithm. The answer can be obtained from inbuilt functions in R or Python. More details behind the Newton algorithm will be provided a bit later.

Coming back to the posterior $\exp(\ell(\beta))$, Taylor expansion of $\ell(\beta)$ around the MLE $\hat{\beta}$ gives

$$f_{\beta|Y=y}(\beta) \propto \exp(\ell(\beta)) \approx \exp \left(\ell(\hat{\beta}) + \langle \nabla \ell(\hat{\beta}), \beta - \hat{\beta} \rangle + \frac{1}{2} (\beta - \hat{\beta})^T H \ell(\hat{\beta}) (\beta - \hat{\beta}) \right)$$

where $H \ell(\beta)$ denotes the Hessian of $\ell(\beta)$:

$$H \ell(\beta) = - \sum_{i=1}^n \frac{\exp(x_i^T \beta)}{(1 + \exp(x_i^T \beta))^2} x_i x_i^T.$$

Because the $\ell(\hat{\beta})$ term is a constant, it can be ignored in proportionality. Also $\nabla \ell(\hat{\beta})$ equals zero. We thus have

$$f_{\beta|Y=y}(\beta) \propto \exp \left(\frac{1}{2} (\beta - \hat{\beta})^T H \ell(\hat{\beta}) (\beta - \hat{\beta}) \right) = \exp \left(-\frac{1}{2} (\beta - \hat{\beta})^T \left(-H \ell(\hat{\beta}) \right) (\beta - \hat{\beta}) \right)$$

We have switched above to $-H\ell(\hat{\beta})$ because this matrix is positive semi-definite as $\hat{\beta}$ maximizes $\ell(\beta)$. The above term is simply the unnormalized density of the multivariate normal distribution with mean $\hat{\beta}$ and covariance matrix $(-H\ell(\hat{\beta}))^{-1}$. Observe that

$$-H\ell(\hat{\beta}) = \sum_{i=1}^n \frac{\exp(x'_i \hat{\beta})}{\left(1 + \exp(x'_i \hat{\beta})\right)^2} x_i x'_i.$$

Now let W denote the $n \times n$ diagonal matrix whose i^{th} diagonal entry is

$$\frac{\exp(x'_i \hat{\beta})}{\left(1 + \exp(x'_i \hat{\beta})\right)^2}$$

and also recall again that X is the $n \times p$ matrix with rows x'_1, \dots, x'_n . It is then easy to check that

$$-H\ell(\hat{\beta}) = X'WX. \quad (3)$$

The posterior normal approximation is thus

$$N(\hat{\beta}, (X'WX)^{-1}). \quad (4)$$

The standard errors corresponding to β_0, \dots, β_m can then be obtained by the square roots of the diagonal entries of $(X'WX)^{-1}$.

It turns out that Bayesian inference done with the above normal posterior approximation (4) coincides with the frequentist inference in the logistic regression model. It is easy to check this, say, in Python using statsmodels (construct a 95% credible interval for, say, one of the components of β and then compare it with the frequentist interval). Thus the usual frequentist inference for the logistic regression model can be viewed from a Bayesian perspective.

Note that the above analysis relies on two assumptions: (a) the prior for β is assumed to be uniform on the large cube $(-C, C)^p$, and (b) the posterior is approximated by a normal distribution. These assumptions may be of course not reasonable in a particular application. In such a situation, it is conceptually very clear as to how one would proceed: if the normal approximation to the posterior is not accurate, one needs to work with the actual posterior. If the uniform prior is not reasonable, one can do the full posterior analysis (or by taking a normal approximation to the posterior) for a more appropriate prior.

1.1 Details behind the Newton Algorithm for computing the MLE

The MLE $\hat{\beta}$ of β is the maximizer of $\ell(\beta)$. The maximizer of $\ell(\beta)$ cannot be computed in closed form. Newton's method is commonly used for maximizing $\ell(\beta)$. Newton's method uses the iterative scheme

$$\beta^{(m+1)} = \beta^{(m)} - \left(H\ell(\beta^{(m)})\right)^{-1} \nabla\ell(\beta^{(m)}) \quad (5)$$

As we saw in (2),

$$\nabla\ell(\beta) = \sum_{i=1}^n (y_i - \pi_i) x_i$$

where π_i is given by

$$\pi_i = \frac{\exp(x'_i \beta)}{1 + \exp(x'_i \beta)}$$

Letting π be the $n \times 1$ vector with entries π_1, \dots, π_n , we can write $\nabla \ell(\beta)$ in matrix notation as (note that X has rows x_1^T, \dots, x_n^T or, equivalently, X^T has columns x_1, \dots, x_n):

$$\nabla \ell(\beta) = X^T (y - \pi)$$

where, as usual in regression, y denotes the $n \times 1$ vector of response values. Also from (3), we can write

$$H\ell(\beta) = -X^T W X$$

where W is the $n \times n$ diagonal matrix whose i^{th} diagonal entry is $\pi_i(1 - \pi_i)$. Newton's iterative scheme (5) therefore becomes

$$\beta^{(m+1)} = \beta^{(m)} + (X^T W X)^{-1} X^T (y - \pi).$$

This can be rewritten as

$$\beta^{(m+1)} = (X^T W X)^{-1} X^T W Z \tag{6}$$

where

$$Z = X\beta^{(m)} + W^{-1}(y - \pi). \tag{7}$$

The method of obtaining the MLE $\hat{\beta}$ therefore proceeds iteratively as follows. First have an initial estimate of β . Call this initial estimator $\hat{\beta}^{(0)}$. Use this estimator to calculate π_i via

$$\pi_i = \frac{\exp(\hat{\beta}_0^{(0)} + \hat{\beta}_1^{(0)} x_{i1} + \dots + \hat{\beta}_p^{(0)} x_{ip})}{1 + \exp(\hat{\beta}_0^{(0)} + \hat{\beta}_1^{(0)} x_{i1} + \dots + \hat{\beta}_p^{(0)} x_{ip})}.$$

Use these values of π_i to create the response variable values Z_i via (7) and also use values of π_i to construct the matrix W . With Z and W , we can estimate β via

$$\hat{\beta}^{(1)} = (X^T W X)^{-1} X^T W Z.$$

Now replace the initial estimator $\hat{\beta}^{(0)}$ by $\hat{\beta}^{(1)}$ and repeat this process. Keep repeating this until two successive estimates $\hat{\beta}^{(m)}$ and $\hat{\beta}^{(m+1)}$ do not change much. At that point, stop and report the estimate of β in the logistic regression model as $\hat{\beta}^{(m)}$.

The expression $(X^T W X)^{-1} X^T W Z$ is reminiscent of the usual $(X^T X)^{-1} X^T y$ which is the usual estimate of β in the linear model. In fact, this is the least squares estimate in a weighted least squares model.