SDS 383D: Modeling II **Spring 2017**

Section 3: Bayesian GLMs

Student: Natalia Zuniga-Garcia

3.1 Modeling non-Gaussian observations

So far, we've assumed real-valued observations. In this setting, our likelihood model is a univariate normal, parametrized by a mean $x_i^T\beta$ and some precision that does not directly depend on the value of x_i. In general, $x_i^T\beta$ will take values in $\mathbb{R}$

If we don't want to use a Gaussian likelihood, we typically won't be able to parametrize our data using a real-valued parameter. Instead, we must transform it via an appropriate link function. This is, in essence, the generalized linear model.

As a first step into other types of data, let's consider binary valued observations. Here, the natural likelihood model is a Bernoulli random variable; however we cannot directly parametrize this by $x_i^T\beta$. Instead, we must transform $x_i^T\beta$ to lie between 0 and 1 via some function $g^{-1}:\mathbb{R}\rightarrow(0,1)$. We can then write a linear model as

$$\begin{aligned}y_i|p_i \sim& \text{Bernoulli}(p_i)\\ p_i =& g^{-1}(x_i^T\beta)\\ \beta|\theta \sim& \pi_\theta(\beta)\end{aligned}$$

where $\pi_\theta(\beta)$ is our choice of prior on β. Unfortunately, there is no choice of prior here that makes the model conjugate.

Let's start off with a normal prior on β. One appropriate function for g^{-1} is the CDF of the normal distribution – known as the probit function. This is equivalent to assuming our data are generated according to

$$\begin{aligned}y_i =& \begin{cases}1 & ifz > 0\\ 0 & \text{otherwise}\end{cases}\\ z_i \sim& \text{N}(x_i^T\beta, \tau^2)\end{aligned}$$

If we put a normal-inverse gamma prior on β and τ, then we have a *latent* regression model on the (x_i, z_i) pairs, that is idential to what we had before! Conditioned on the z_i, we can easily sample values for β and τ.

Exercise 3.1 *To complete our Gibbs sampler, we must specify the conditional distribution $p(z_i|x_i, y_i, \beta, \tau)$. Write down the form of this conditional distribution, and write a Gibbs sampler to sample from the posterior distribution. Test it on the dataset* ***pima.csv****, which contains diabetes information for women of Pima indian heritage. The dataset is from National Institute of Diabetes and Digestive and Kidney Diseases, full information and explanation of variables is available at* ***http://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes****.*

Solution

$$p(z|x, y, \beta, \tau) \propto P(y|z)p(z|\beta, X) = \prod_{i=1}^{N} p(y_i|z_i)p(z_i|\beta, x_i)$$

$$\propto \begin{cases} N(z_i|x_i^T\beta, \tau)\mathbb{1}(z_i > 0, y_i = 1) \\ N(z_i|x_i^T\beta, \tau)\mathbb{1}(z_i \leq 0, y_i = 0) \end{cases}$$

Which is a normal truncated function.

The Gibb Sampler is presented in the script *Section3-1.R*,

Some details reviewed from: https://rpubs.com/cakapourani/bayesian-binary-probit-model

```r
library(MASS)
# unloadNamespace("MASS")
library(truncnorm) # Truncated Normal distribution

# Prior Hyperparameters
K.zero <- diag(rep(0.1, p)) # precision matriz pxp
beta.zero <- matrix(0, p) # prior guess on beta
a.zero <- 1 # prior sample size for the error variance
b.zero <- 1 # prior sum of square errors for the error variance

N1  <- sum(y)  # Number of successes
N0  <- n - N1  # Number of failures

# Sampling updated parameters
n.iter <- 5000
beta <- matrix(NA, n.iter, p)
tau <- matrix(NA, n.iter)
beta[1,] <- rep(0, p)
tau[1] <- 1
z <- rep(0, n)

# Gibb Sampler
for (i in 2:n.iter) {
  # Update mean of z based on beta
  mu_z <- X %*% beta[i-1,]
  # Get the latent variable from its distribution
  z[y == 0] <- rtruncnorm(N0, mean = mu_z[y == 0], sd = 1, a = -Inf, b = 0)
  z[y == 1] <- rtruncnorm(N1, mean = mu_z[y == 1], sd = 1, a = 0, b = Inf)

  # Get the Betas
  K.new <- K.zero + crossprod(X)
  beta.new <- solve(K.new) %*% (crossprod(X, z) + K.zero %*% beta.zero)
  beta[i,] <- mvrnorm(1, beta.new, solve(tau[i-1] * K.new))

  # Get the tau
  a.new <- a.zero + (n + 1) / 2
  s <- t(beta.zero) %*% K.zero %*% beta.zero + crossprod(z)
```

The accuracy obtained is 77.6% and the correlation coefficient is 0.51, Table 3.1 presents the coefficients.

Table 3.1: Coefficients

Int	Pregnant	Glucose	Blood Press	Skinfold	Insulin	BMI	Pedigre	Age
-4.8202	0.07172	0.01984	-0.008142	0.001419	0.0007390	0.05196	0.4942	0.009956

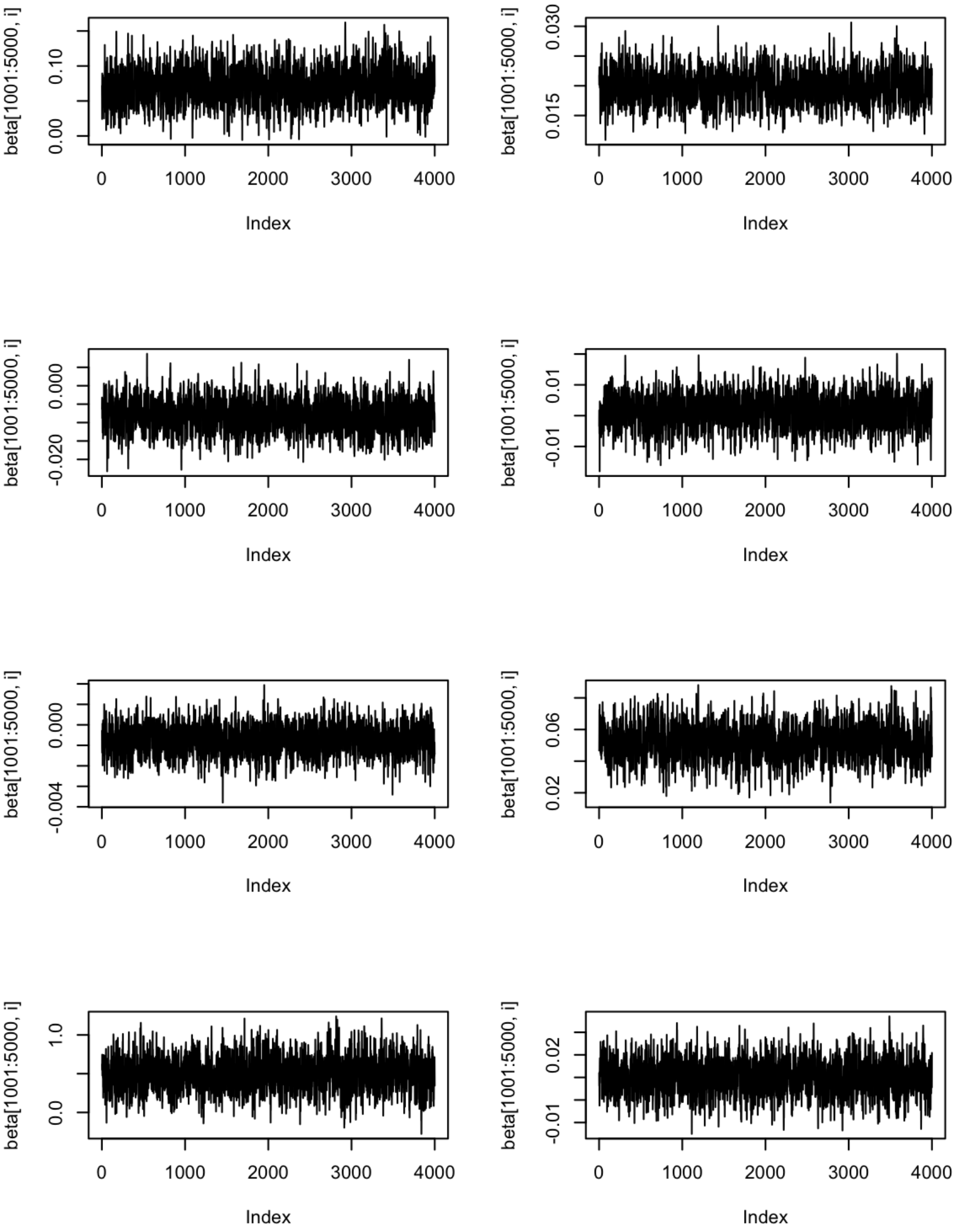

Figure 3.1: Trace plots

Another choice for $g^{-1}(\theta)$ might be the logit function, $\frac{1}{1+e^{-x^T\beta}}$. In this case, it's less obvious to see how we can construct an auxilliary variable representation (it's not impossible! See Polson et al. (2013). But for now, we'll assume we haven't come up with something). So, we're stuck with working with the posterior distribution over β.

Exercise 3.2 *Sadly, the posterior isn't in a "known" form. As a starting point, let's find the maximum a posteriori estimator (MAP). The dataset "titantic.csv" contains survival data from the Titanic; we're going to look at probability of survival as a function of age. For now, we're going to assume the intercept of our regression is zero – i.e. that β is a scalar. Write a function (that can use a black-box optimizer! No need to reinvent the wheel. It shouldn't be a long function) to estimate the MAP of β. Note that the MAP corresponds to the frequentist estimator using a ridge regularization penalty.*

Solution

We need to write a function that can estimate the MAP of β, we start by estimation the posterior:

$$p(\beta|y,X) \propto p(\beta)p(y|\beta,x)$$

Using a $N(1,0)$ prior on β, we have

$$p(\beta|y,X) \propto e^{-\frac{1}{2}\beta^T\beta}\prod_{i=1}^{N} p(x_i)^{y_i}(1-p(x_i))^{1-y_i} \propto e^{-\frac{1}{2}\beta^T\beta}p(x)^y(1-p(x))^{(1-y)}$$

Now I estimate the likelihood:

$$L(\beta|y_i) = \prod_{i=1}^{N} p(\beta)p(\beta|y_i) = e^{-\frac{1}{2}\beta^2}\prod_{i=1}^{N} p(x_i)^{y_i}(1-p(x_i))^{1-y_i} = e^{-\frac{1}{2}\beta^2}\prod_{i=1}^{N}\left(\frac{1}{1+e^{-x_i\beta}}\right)^{y_i}\left(\frac{e^{-x_i\beta}}{1+e^{-x_i\beta}}\right)^{1-y_i}$$

$$\log(L(\beta|y_i)) = -\frac{1}{2}\beta^2 + \sum_{i=1}^{N}\left[-y_i\log(1+e^{-x_i\beta}) + (1-y_i)(-x_i\beta) + (1-y_i)(-\log(1+e^{-x_i\beta}))\right]$$

$$\log(L(\beta|y_i)) = -\frac{1}{2}\beta^2 - \sum_{i=1}^{N}\left[(1-y_i)x_i\beta + \log(1+e^{-x_i\beta})\right]$$

Then, we obtain:

$$\hat{\beta}_{MAP} = arg\min_{\beta}\left\{\frac{\beta^2}{2} + \sum_{i=1}^{N}\left[(1-y_i)x_i\beta + \log(1+e^{-x_i\beta})\right]\right\}$$

Solving in R - presented in the script *Section3-2-to-3-5.R* - the results is: $\hat{\beta} = -0.01101471$

Exercise 3.3 *OK, we don't know how to sample from the posterior, but we can at least look at it. Write a function to calculate the posterior pdf $p(\beta|\mathbf{x},\mathbf{y},\mu,\sigma^2)$, for some reasonable hyperparameter values μ and θ (up to a normalizing constant is fine!). Plot over a reasonable range of β (your MAP from the last question should give you a hint of a reasonable range).*

Solution

Solving in R - presented in the script *Section3-2-to-3-5.R* - the results is:

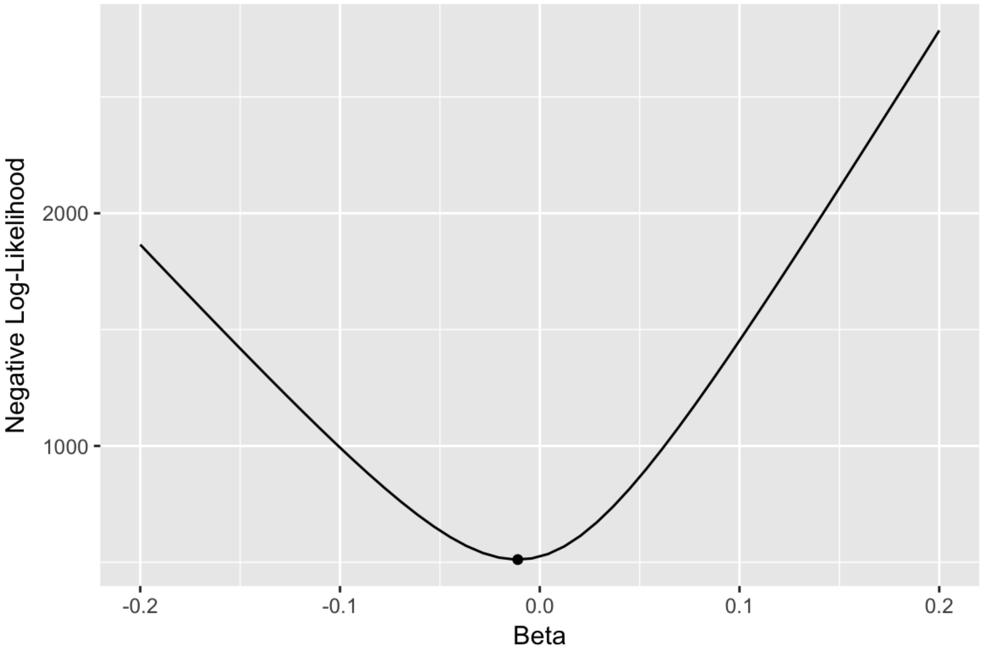

Figure 3.2: Negative Log-Likelihood vs. Beta values

The Laplace approximation is a method for approximating a distribution with a Gaussian, by matching the mean and variance at the mode.[1] Let P^* be the (unnormalized) PDF of a distribution we wish to approximate. We start by taking a Taylor expansion of the log (unnormalized) PDF at the global maximizing value x^*

$$\log P^*(x) \approx \log P^*(x^*) - \frac{c}{2}(x - x^*)^2$$

where $c = -\frac{\delta^2}{\delta x^2} \log P^*(x)\Big|_{x=x^*}$.

We approximate P^* with an unnormalized Gaussian, with the same mean and variance as P^*:

$$Q^*(x) = P^*(x^*) \exp\left\{-\frac{c}{2}(x - x^*)^2\right\}$$

Exercise 3.4 *Find the mean and precision of a Gaussian that can be used in a Laplace approximation to the posterior distribution over β.*

Solution We have,

$$Q^*(x) = P^*(x^*) \exp\left\{-\frac{c}{2}(x - x^*)^2\right\}$$

The the mean an precision are obtained directly from the Gaussian form:

[1] More generally, the Laplace approximation is used to approximate integrands of the form $\int_A e^{Nf(x)}dx$... but for our purposes we will always be working with PDFs.

- $x* = \hat{\beta}_{MAP}$ because it is the global maximum of the distribution of beta.
- $c = -\frac{\delta^2}{\delta x^2} \log P^*(x)\Big|_{x=x^*}$, we need to estimate this.

$$c = -\frac{\delta^2}{\delta x^2} \log P^*(x)\Big|_{x=x^*} = 1 + \sum_{i=1}^{N} \frac{x_i^2 e^{-x_i\beta}}{(1+e^{-x_i\beta})^2}$$

Exercise 3.5 *That's all well and good... but we probably have a non-zero intercept. We can extend the Laplace approximation to multivariate PDFs. This amounts to estimating the precision matrix of the approximating Gaussian using the negative of the Hessian – the matrix of second derivatives*

$$H_{ij} = \frac{\delta^2}{\delta x_i \delta x_j} \log P^*(x)\Big|_{x=x^*}$$

Use this to approximate the posterior distribution over β. Give the form of the approximating distribution, plus 95% marginal credible intervals for its elements.

Solution

Now, using matrix form:

$$\frac{\delta^2}{\delta\beta\delta\beta^T} \ln(p(\beta)) \propto -I - \sum_{i=1}^{N} \frac{x_i x_i^T e^{-x_i^T\beta}}{(1+e^{-x_i^T\beta})^2}$$

Let's try the same thing with a Poisson likelihood. Here, the obvious transformation is to let $g^{-1}(\theta) = e^{\theta}$, i.e.

$$\begin{aligned} y_i | p_i \sim & \text{Poisson}(\lambda_i) \\ \lambda_i = & e^{x_i^T\beta} \end{aligned}$$

We're going to work with the dataset `tea_discipline_oss.csv`, a dataset gathered by Texas Appleseed, looking at the number of out of school suspensions (ACTIONS) accross schools in Texas. The data is censored for privacy reasons – data points with fewer than 5 actions are given the code "-99". For now, we're going to exclude these data points.

Exercise 3.6 *We're going to use a Poisson model on the counts. Ignoring the fact that the data is censored, why is this not quite the right model? Hint: there are several answers to this – the most fundamental involve considering the support of the Poisson.*

Solution

We can observe from the histogram of *Actions* that the distribution is highly skewed. In addition, we know that the Poison distribution has $E[x] = Var[x] = \lambda$, however, for this data we have $E[x] = 15.93$ and $Var[x] = 460.91$ so the difference is very high.

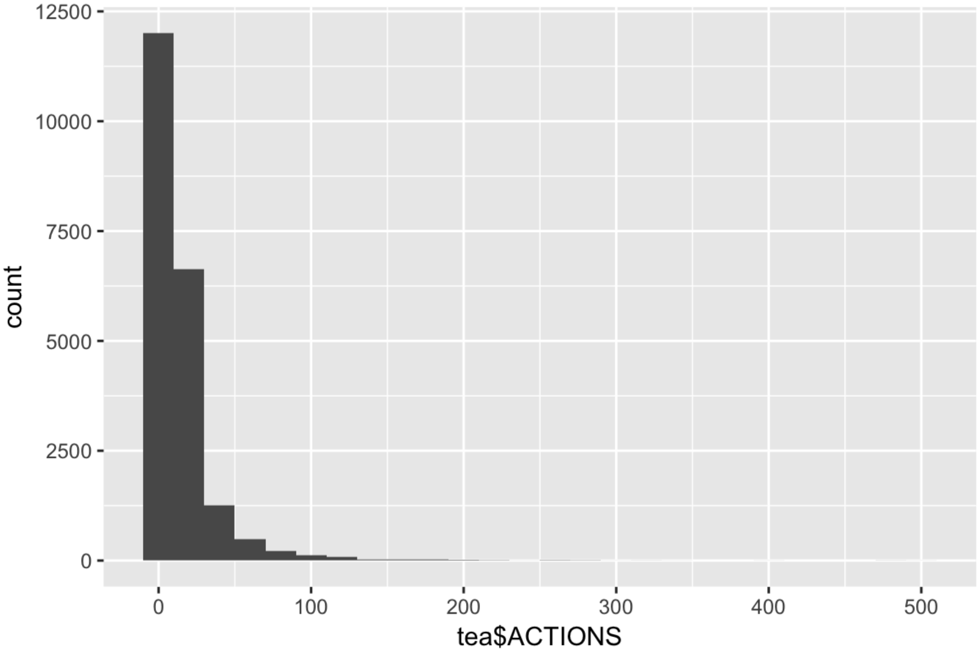

Figure 3.3: Histogram of *Actions*

Exercise 3.7 *Let's assume our only covariate of interest is GRADE*[2] *and put a normal prior on β. Using a Laplace approximation and an appropriately vague prior, find 95% marginal credible intervals for the entries of β. You'll probably want to use an intercept.*

Exercise 3.8 (Optional) *Repeat the analysis using a set of variables that interest you.*

Even though we don't have conjugacy, we can still use MCMC methods – we just can't use our old friend the Gibbs sampler. Since this isn't an MCMC course, let's use STAN, a probabilistic programming language available for R, python and Matlab. I'm going to assume herein that we're using RStan, and give appropriate scripts; it should be fairly straightforward to use if you're an R novice, or if you want to use a different language, there are hints on translating to PyStan at `http://pystan.readthedocs.io/en/latest/differences_pystan_rstan.html` and info on MatlabStan (which seems much less popular) at `http://mc-stan.org/users/interfaces/matlab-stan`.

Exercise 3.9 *Download the sample STAN script* `poisson.stan` *and corresponding R script* `run_poisson_stan.R`*. The R script should run the regression vs GRADE from earlier (feel free to change the prior parameters). Run it and see how the results differ from the Laplace approximation. Modify the scripy to include more variables, and present your results.*

Solution

Results are: $\beta_{int} = 0.05026685$ and $\beta_{grade} = 2.389339$. The trace-plots are shown as follows,

[2] I have manually replaced Kindergarten and Pre-K with Grades 0 and -1, respectively.

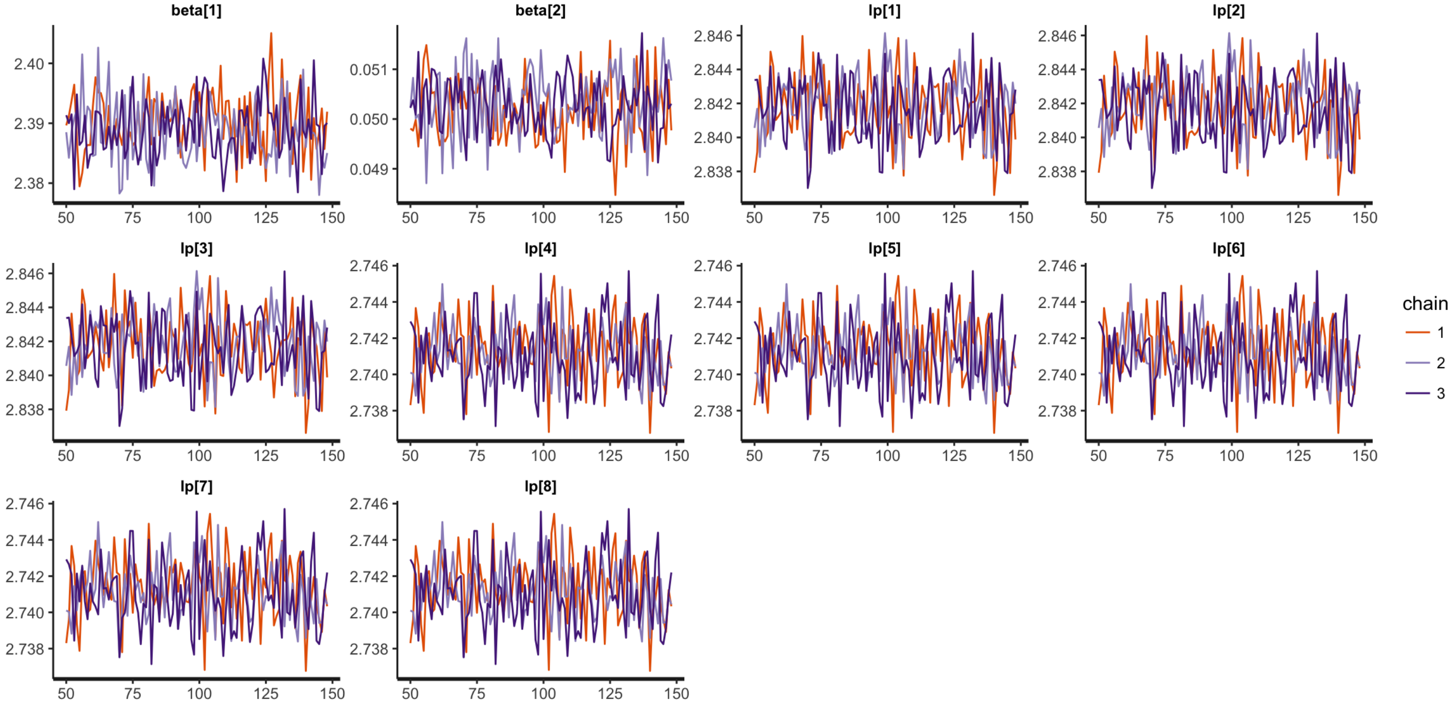

Figure 3.4: Trace plots

References

Polson, N. G., Scott, J. G., and Windle, J. (2013). Bayesian inference for logistic models using Pólya–gamma latent variables. *Journal of the American statistical Association*, 108(504):1339–1349.