20592 - Statistics and Probability
Final Project

Group 12
Lalic Pavle, Mancinelli Pierluigi, Martinelli Aurora,
Nenkov Nikola Nikolaev, Paoli Susanna

Jan 11, 2024

1 Description of the algorithms

In the project, we implement and analyze the performance of two algorithms for estimating the coefficients of a probit regression with computational approximations. We compare Fisher scoring and random walk Metropolis-Hastings. In this way, we can tackle the same problem from both a Frequentist and a Bayesian approach. First, we implement Fisher scoring using the equations derived in the following section.

Fisher scoring, which is a variation of the general Newton-Raphson method in which we use the Fisher information of the sample instead of the Hessian matrix in the update equation, converges more rapidly than gradient descent, thanks to the incorporation of the second derivate. Using Fisher scoring we can find an estimate of the parameters using the Weighted Least Squares (WLS) method. WLS would boil down to OLS if, in the context of generalized linear models, we had $\mu_i = \eta_i$, namely, the link function was given by the identity function. However, in our case, the link function is given by the cumulative density function of a standard normal and thus we don't have any simplification in this sense.

Second, we implement a Markov Chains Monte Carlo approximation to tackle the same problem from a Bayesian perspective. Specifically, we implement a random walk Metropolis-Hastings algorithm, where the proposal distribution is symmetric $g(\theta^*|\theta^{(t-1)}) = g(\theta^{(t-1)}|\theta^*)$. In other words, the probability of moving "back and forth" is the same. Metropolis-Hastings is an algorithm that exploits MCMC to perform computational approximations in the realm of Bayesian analysis. If we want to approximate the value of a parameter, θ, the general idea behind MCMC is to construct a sequence of dependent θ's that converge in distribution to the true posterior distribution of the parameter. The idea is to construct an ergodic Markov chain whose unique stationary distribution converges to the posterior distribution of the parameter. The key feature of Metropolis-Hastings concerns the "acceptance probability". That is, at each iteration, we sample from the proposal distribution and accept the sampled value with probability α (refer to Section 3 for the derivation). With probability $(1 - \alpha)$, we set the new value of iteration t equal to the previous value $\theta^{(t-1)}$. We chose to use a random walk Metropolis-Hastings approach to tackle this problem because of its flexibility and natural ability to work well with multivariate distributions, even though it probably requires more time to converge due to the acceptance probability step. Gibbs sampling might have provided more computational efficiency, but given that we are dealing with only a few regressors, we chose to prioritize the global exploration of RWMH rather than computational efficiency. We discuss some challenges we faced related to the convergence of MH in Section 4.

2 Derivation of update equations for Fisher scoring method

As previously said, we are dealing with probit regression, therefore we have a binary dependent variable (the random component of the generalized linear model), distributed as a Bernoulli random variable, with the canonical parameter θ. More specifically, we have $Y_i \sim \text{Be}(\frac{e^\theta}{1+e^\theta})$, where $\frac{e^\theta}{1+e^\theta}$ can be derived by writing the density function of the dependent variable in its exponential family form. The link function, in this case, is given by the inverse of the CDF of a standard normal distribution $N(0,1)$. Specifically, we have $\Phi^{-1}(\mu_i) = \eta_i$, with $\mu_i = \frac{e^\theta}{1+e^\theta}$, and $\eta_i = X_i^T\beta$, the systematic component of the model. In this section, we derive the updating equations for the implementation of the Fisher-scoring algorithm to find the optimal values of the parameters.

$$\frac{\delta\eta_i}{\delta\mu_i} = (\frac{\delta\mu_i}{\delta\eta_i})^{-1}$$

$$\frac{\delta\mu_i}{\delta\eta_i} = \frac{\delta\Phi(\eta_i)}{\delta\eta_i} = \phi(\eta_i) = \frac{1}{\sqrt{2\Pi}}\text{exp}(-\frac{\eta_i^2}{2})$$

$$\frac{\delta\eta_i}{\delta\mu_i} = (\frac{\delta\mu_i}{\delta\eta_i})^{-1} = \sqrt{2\Pi}\text{exp}(\frac{\eta_i^2}{2})$$

Then, we need to calculate the variance of Y_i. For a Bernoulli, it is given by

$$\text{Var}(Y_i) = \mu_i(1-\mu_i) = \Phi(\eta_i)(1-\Phi(\eta_i))$$

Now, we have everything we need to derive the updating equations for the Fisher-scoring algorithm for this Probit regression.

$$w_{ii} = \frac{1}{\text{Var}(Y_i)}(\frac{\delta\eta_i}{\delta\mu_i})^{-2} = \frac{1}{\Phi(\eta_i)(1-\Phi(\eta_i))}\frac{1}{2\Pi\text{exp}(\eta_i^2)}$$

$$z_i = \eta_i + (Y_i - \mu_i)\frac{\delta\eta_i}{\delta\mu_i} = \eta_i + (Y_i-\mu_i)\sqrt{2\Pi}\text{exp}(\frac{\eta_i^2}{2})$$

3 Derivation of acceptance probability for MCMC

As previously mentioned, we derive the acceptance probability for the random walk Metropolis-Hastings algorithm. First, in a general MH, with accept with probability α, where

$$\alpha = \min\{1, \frac{\Pi(\theta^*|Y)g(\theta^{(t-1)}|\theta^*)}{\Pi(\theta^{(t-1)}|Y)g(\theta^*|\theta^{(t-1)})}\}$$

However, as we explained before, we have $g(\theta^{(t-1)}|\theta^*) = g(\theta^*|\theta^{(t-1)})$, our ration simplifies to

$$\alpha = \min\{1, \frac{\Pi(\theta^*|Y)}{\Pi(\theta^{(t-1)}|Y)}\}$$

Thus, the acceptance probability exclusively depends on the posterior probability evaluated iteratively at the proposed value of the parameter and the previous step's parameter. In our context, the parameter to estimate is the vector of betas, the coefficients in the regression. After some brainstorming, we chose the standard normal distribution as a prior for the parameters. Thus, the ratio above becomes

$$\frac{\Pi(\beta^*|Y,X)}{\Pi(\beta^{(t-1)}|Y,X)}$$

Generally, we have the posterior proportional to the prior times the likelihood. In this case with Bernoulli($\Phi(\eta_i)$) likelihood and gaussian prior we get:

$$\Pi(\beta|Y,X) \propto \prod_{i=1}^{n}\Phi(\eta_i)^{y_i}(1-\Phi(\eta_i))^{1-y_i}e^{-\frac{1}{2}\sum_{j=1}^{k}\beta_j^2}$$

This expression can be simplified by simply using the logarithm, which maintains ordinal relations, given that it is monotonic. Thus, the above expression becomes

$$p(\beta|Y,X) \propto \sum_{i=1}^{n} log[\Phi(\eta_i)^{y_i}(1-\Phi(\eta_i))^{1-y_i}](-\frac{1}{2}\sum_{j=1}^{k}\beta_j^2)$$

$$p(\beta|Y,X) \propto -\sum_{i=1}^{n}\{y_i log\Phi(\eta_i) + (1-y_i)log(1-\Phi(\eta_i))\}\{\sum_{j=1}^{k}\beta_j^2\}$$

Since we need the ratio of two likelihoods, we get

$$\text{log ratio} = p(\beta^*|Y,X) - p(\beta^{(t-1)}|Y,X) = -\sum_{i=1}^{n}\{y_i log\Phi(\eta_i) + (1-y_i)log(1-\Phi(\eta_i))\}\{\sum_{j=1}^{k}\beta_j^{*2}\}+$$

$$+\sum_{i=1}^{n}\{y_i log\Phi(\eta_i) + (1-y_i)log(1-\Phi(\eta_i))\}\{\sum_{j=1}^{k}\beta_j^{(t-1)2}\}$$

where β^* is the new proposed value at iteration t from the proposal distribution and $\beta^{(t-1)}$ is the value of β at iteration $t-1$.

4 Diagnostics for convergence of the algorithms

For Fisher scoring, reaching convergence was relatively easy. We initialized β_0 to 0 and set the convergence criterion constant ϵ to 0.01: this combination allowed us to obtain convergence very rapidly. We also tested an initialization of 1, which, however, performed more poorly. An important thing to note is that in the case of probit regression, similar to what we encountered when we worked on Poisson regression, the value of epsilon for the convergence criterion needed to be greater than what we needed in logistic regression, probably due to the fact that the cumulative distribution function used as link function allows for smoother convergence, and thus a larger ϵ was sufficient for the algorithm to converge (we underline that when we tried with a smaller value, convergence was never reached). To ensure convergence, we also needed to take care of the overflow problem, which was easily addressed by adding a small constant where needed.

For the convergence of the Metropolis-Hastings algorithm, instead, we needed to take care of many different things, like the number of iterations, the initial value of the vector of parameters, and the standard deviation for the proposal distribution, assumed to be known to avoid dealing with the inverse gamma. First, for the number of iterations, we performed different tests, from 100,000 iterations up to 4,000,000 iterations. However, we found that these large values were needed for two coefficients that hardly converged: the intercept and the dummy indicating the family history of heart diseases ("famhist"). Thus, even though the best result was obtained with fewer iterations, around 200,000, the strenuous convergence of those two parameters forced us to set the number of iterations to 2.5 million, always trying to strike a balance between computational efficiency and optimal values for the two difficult convergences. Second, we initialized the vector of parameters to 1, apart from the intercept set to -5. While in Fisher scoring we initialized it to zero, here a vector of ones worked more efficiently because it allowed the algorithm to explore the parameter space better, given the random nature of Metropolis-Hastings. Moreover, the choice of the intercept has to be based on the hypothesis that, without any covariate, the likelihood of having heart disease should be extremely low, around 0% marginal effect (apart from a natural predisposition, we should not be expecting large marginal effects in the likelihood of contracting a coronart disease if nothing changes). The most challenging part was related to the choice of the standard deviation proposal. First, to be consistent with the choice of the prior, we assumed no correlation among parameters, thus the matrix was diagonal.

Secondly, we did different tests to find a good balance between setting variances too high and never converging and setting the variances too low and blocking the algorithm from exploring the parameter space completely. We tested different values between 0.000001 and 1 and found that the best choice occurred with a variance of 0.01. After estimation, we also tried to find the best values for *burnin* and *thinning*, crucial for optimizing mixing and relational times. Specifically, the former has been set to around 500,000 (on 2,500,000 iterations) to drop these first values, which were only intermediate changes in the process of getting to the true value. Here the main issue came with the intercept and a categorical variable, whereas the continuous parameters converged very fast. For thinning, instead, we set it to 40 so that we could pick values at a 40-step distance. In this way, we could minimize the autocorrelation among values of the same coefficient.

5 Prior distribution choice for Bayesian regression model

Selecting the most appropriate prior distribution for the parameters was something that challenged us a lot, as we had to perform several trials before finding the combination that best suited the data at hand. We tested different prior distributions with different hyper-parameters, but what we found working best was a multivariate normal prior with mean of zeros and an identity matrix as the variance-covariance matrix. We chose to set the mean to a zero vector mainly based on the values we obtained through Fisher-scoring approximation (all around zero but the intercept), which we thought might provide a good path to follow. Thus, we thought that it would have been a good choice. We set the variance-covariance matrix to the identity matrix instead because we assumed, based on our prior beliefs, that the parameters were uncorrelated to each other. We thought this was a good assumption because we did not have any specific prior information about the parameters, and thus, imposing a more informative but wrong prior might have biased the results terribly. Thus, we opted for a more “neutral” approach.

6 Final discussion on the fitted models

After running the two algorithms, we obtained two vectors of coefficients, one for Fisher scoring and one for random walk Metropolis-Hastings. The estimated coefficients are, of course, slightly different. However, the coefficient interpretation is the same for both algorithms. First, let us recall what problem we are dealing with. We have a dataset of observational data related to coronary disease. Along with the final binary dependent variable indicating the presence of coronary disease, we also have a suite of regressors that can be analyzed to understand which factors mainly affect the dependent variable. Given that we are in probit regression and we are dealing with a generalized linear model that requires the usage of a link function, we know that the estimated coefficients give us the changes in the dependent variable as measured by the link function (the cumulative distribution function of a standard normal, in our case) for a one-unit change in the regressors. So, in our case, the betas give us the changes in the Z-score for a one-unit change in the independent variables. Given that providing a practical suggestion using a Z-score is not an optimal solution (especially if we need to communicate the results to people who are not experts in statistics), we also calculated the marginal effects that, in our case, give us a direct interpretation in terms of changes in the probability of contracting a coronary disease. To explain better, we consider the case of cholesterol, which is given by the variable “ldl” (low-density lipoprotein cholesterol) in the dataset. The beta coefficients are respectively 0.1028 (Fisher scoring) and 0.1025 (MCMC). These values give us the changes in the Z-score for a one-unit change in the level of cholesterol. In other words, as cholesterol increases by one unit, the Z-score deviates 0.103 more from the mean. However, as mentioned earlier, this interpretation is difficult to convey. Thus, we calculated the marginal effects, as anticipated before, using the formula $\phi(X\beta_i)\beta_i$. We chose to calculate marginal effects at the means; thus, in this case, X

represents the average observation in the sample. We obtained, respectively, 0.03643 (Fisher scoring) and 0.03634 (MCMC). Turning these values into percentages, we obtain that regardless of the algorithm used, a one-unit increase in cholesterol is expected to increase the probability of contracting a coronary disease by 3.6% (despite the different natures of the algorithms, we obtain consistent results for the marginal effects, which is ultimately the most important thing in real world applications). The first thing that can be noted is that despite the different natures of the algorithms, we obtained more or less similar values in terms of marginal effects. Second, comparing it with the other marginal effects, we notice how cholesterol plays a very important role in this context, with other coefficients affecting the probability of contracting a coronary disease by much less (especially in the case of MCMC, where some marginal effects are of the order of 10^{-5}).

Lastly, even though in our specific case we were more focused on estimation, prediction is a task that plays a fundamental role in general data analysis scenarios. Thus, to compare the two algorithms, we also calculated the accuracy and found values of around 73% for Fisher scoring and 57% for MCMC. Therefore, Fisher-scoring proved to be a bit more accurate. We avoided performing a training-testing split because, again, our focus was estimation. However, in the case where prediction was the main focus, more accurate tuning for that specific task should be implemented.

6.1 Coefficients' interpretation

Regarding interpretation, the marginal effect value tells us that a unitary increase in "ldl" produces a 3.6% increase in the likelihood of having CHD. We performed some exploratory analysis on the dataset and noticed that the "ldl" variable ranges from 0.98 to 15.3, which, after carrying out some research, we recognized as being on a different scale than the one used for indicating clinical levels of cholesterol, approximately 0-200 mg/dL of blood. Indeed, assuming these two scales to be proportional, we adapted the interpretation by re-scaling the variable: a unit increase in "ldl" would correspond to an increase of about 13.97 mg/dL which could be considered enough for increasing the likelihood of CHD by 3.6%.

Apart from cholesterol, many other coefficients were estimated. Trying to avoid being too pedantic, we will address only the most interesting ones. For instance, we understand that blood pressure ("sbp"), commonly known for being one of the major causes of death from CHD, is instead having an almost null impact on the likelihood of the disease's insurgence (marginal effects are of the order of 10^{-3} and 10^{-5}). This could be explained by the measurement error described in the dataset's details: this pressure has been recorded, for many individuals, only after patients had already undergone some treatments for reducing blood pressure, biasing the relationship between blood pressure and coronary diseases. Finally, we also wanted to address the role of family history, the only categorical (dummy) variable in the dataset. It represents the presence of genomic heritage in CHD, namely whether someone in the patient's family has already suffered from it. Indeed, as expected, having a family history of heart disease raises the probability of CHD by approximately 20%.

Lastly, to conclude and as a support for the medical cause, we would like to underline that tobacco is indeed dangerous for one's health: considering a unitary increase in the cumulative Kg of tobacco consumption, corresponding to about 750 cigarettes, we have an increase of about 1.9% in the likelihood of incurring coronary diseases. In real terms, considering that in South Africa, the average number of cigarettes smoked per smoker annually is approximately 3,857, corresponding to around 5kg of tobacco, we have that each smoker increases by about 10% their likelihood of CHD.