# Computational Statistics Report - Group 14

## A general Probit model

The probit model is a regression model used in binary classification problems, where we expect the values Y to be Bernoulli distributed. It is characterised by the link function:
$\mu_i = \Phi(\eta_i)$

## Probit with Fisher Scoring

### Algorithm description

In our implementation, beta is initialised as a null vector, which is used to compute the initial values of $\mu$ (since it is $X\beta$) and $\eta$, through the link function that connects it to $\mu$. Those values are, in turn, used to calculate the entries of W and Z, as seen above. These elements are used in the update equation for beta, giving the values of beta for the next step, which will be used to compute the new values of $\mu$ and $\eta$ in an iterative process. We repeat this process until we reach convergence according to our selected convergence criterion.

### Equation derivation of the Fisher Scoring

The Fisher scoring method can be obtained by substituting the second derivative of the log-likelihood with the observed fisher information. This leads to the following formulation of the update equation:
$\beta_{t+1} = (X^T W_t X)^{-1} X^T W_t Z_t$

Here Z is a vector whose elements are $Z_{i,t} = \eta_{i,t} + (Y_i - \mu_{i,t}) \frac{\partial \eta_{i,t-1}}{\partial \mu_{i,t-1}}$, and

W is a diagonal matrix with elements $W_{ii} = \frac{1}{Var(Y_i)} \cdot \left(\frac{\partial \eta_i}{\partial \mu_i}\right)^{-2}$.

Since Y is bernoulli distributed, its variance is $Var(Y_i) = p(1-p) = \mu_i(1-\mu_i)$.

Therefore, we need to derive the derivative of $\eta$ with respect to $\mu$ in order to be able to derive the update equation.
$\frac{\partial \eta_i}{\partial \mu_i} = \frac{\partial}{\partial \mu_i} \Phi^{-1}(\mu_i)$

We can apply the rule for the derivative of the inverse of a function, $\left(f^{-1}\right)'(x) = \frac{1}{f'(f^{-1}(x))}$, leading to:

$\frac{\partial}{\partial \mu_i} \Phi^{-1}(\mu_i) = \left(\Phi^{-1}\right)'(\mu_i) = \frac{1}{\Phi'\left(\Phi^{-1}(\mu_i)\right)}$

In order to find the derivative of a Gaussian cumulative distribution function, we can apply the Leibniz integral rule for differentiation under the integral sign, since $\Phi(x) = \int_{-\infty}^{x} f(y)dy$, where $f(y)$ is the probability distribution function. The Leibniz rule can be written as follows:

$$\frac{d}{dx}\left(\int_{a(x)}^{b(x)} g(x,\ t)dt\right) = g(x, b(x))\frac{d}{dx}b(x)\ -\ g(x,\ a(x))\frac{d}{dx}a(x)\ +\ \int_{a(x)}^{b(x)}\frac{d}{dx}g(x,\ t)dt$$

Which in our case becomes:

$$\frac{d}{dx}\left(\int_{-\infty}^{x} f(t)dt\right) = f(x)\frac{d}{dx}x\ +\ \int_{-\infty}^{x}\frac{d}{dx}f(t)dt$$

And, since $f(t)$ does not depend on $x$, $\frac{d}{dx}f(t)\ =\ 0$ and thus

$\Phi'(x)\ =\ f(x)$.

Therefore, we have that:

$$\frac{\partial \eta_i}{\partial \mu_i} = \frac{1}{f(\Phi'(\mu_i))} = \frac{1}{f(\eta_i)}$$

And the entries of the matrix W become:

$$W_{ii}\ =\ \frac{1}{Var(Y_i)}\cdot\left(\frac{\partial \eta_i}{\partial \mu_i}\right)^{-2} = \frac{f(\eta_i)^2}{\mu_i(1-\mu_i)}$$

The entries of vector Z are:

$$Z_i\ =\ \eta_i\ +\ \frac{Y_i-\mu_i}{f(\eta_i)}$$

### Output and convergence of Fisher scoring

The convergence criterion we used is the relative convergence criterion, given by the formula:

$\frac{|x_t - x_{t-1}|}{|x_t| + \epsilon} < \epsilon$ which in our case becomes $\frac{|\beta_t - \beta_0|}{|\beta_t| + \epsilon} < \epsilon$.

The iterative process stops when convergence is reached, namely when $\epsilon$ reaches ~1.341.

```
Convergence reached with value: 1.341365893593283e-07
```

| | betas |
|---|---|
| **constant** | -0.514337 |
| **sbp** | 0.077668 |
| **tobacco** | 0.221475 |
| **ldl** | 0.212949 |
| **adiposity** | 0.096447 |
| **famhist** | 0.265909 |
| **typea** | 0.231259 |
| **obesity** | -0.169230 |
| **alcohol** | 0.000479 |
| **age** | 0.383769 |

## Probit with Metropolis Hastings

### Algorithm description

In our implementation, the MH function follows a 3 step process.

First, it takes an initial beta which is passed to the function as a vector of zeros and proposes a new beta by adding to the initial beta vector an equal size vector where each element is sampled from a normal distribution with mean 0 and variance 0.1. This way the proposed beta is close to the current beta.

The second step requires us to compare the value of the acceptance ratio with the realisation from a uniform distribution U(0,1). Lastly, if the acceptance ratio is higher, we append to the list of beta vectors the proposed beta, otherwise we append the current beta and move to the next iteration (we run for 10,000 iterations).

For calculating the acceptance ratio we need both the posterior of the proposal and current beta, so we defined a posterior function that we then apply to calculate the acceptance ratio.

**The analytical description of the Probit Metropolis Hastings Algorithm**

We implemented the algorithm with two different priors, a uniform and a normal; the qualities and drawbacks of each of them will be discussed later. As the results that we obtained are very similar, in this paragraph we will focus on the non-informative prior, that is, the case where the pdf of the prior is P(β) = 1. In this case, the posterior is:

$$P(\beta|Y,X) = L(\beta;Y,X)\cdot P(\beta) = L(\beta;Y,X)\cdot P(\beta) = L(\beta;Y,X) = \prod_{i=1}^{n}\left[\Phi\left(\eta_i\right)\right]^{Y_i}\cdot\left[1-\Phi\left(\eta_i\right)\right]^{1-Y_i}$$

Therefore, the posterior probability is the likelihood itself.

At each step, a new beta (β*) will be proposed according to a normal distribution centred on the value of beta at the previous step: this way, the probability of proposing some vector a when we are in b is the same as the probability of proposing b when we are in a. We call this proposal function $g\left(\beta^{*}|\beta_{t-1}\right)$ which simply adds a vector of realisations from a N(0; 0.1) to the previously proposed beta ($\beta_{t-1}$).

The new proposal accepted if the value of the acceptance ratio $r$ is greater than a randomly sampled u, where:

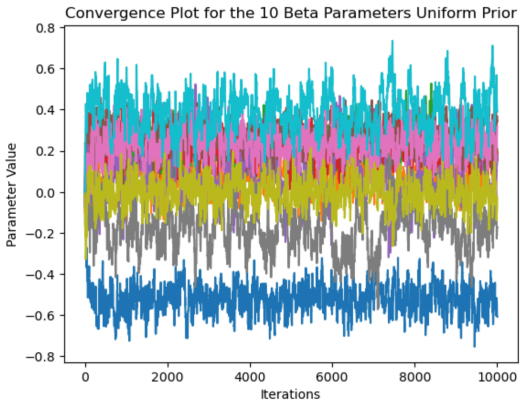
$$r = \min\left\{1; \frac{P(\beta^{*}|X,Y)\cdot g\left(\beta^{*}|\beta_{t-1}\right)}{P(\beta_{t-1}|X,Y)\cdot g\left(\beta_{t-1}|\beta^{*}\right)}\right\} \quad \text{and} \quad u \sim \text{Uniform}[0,1],$$

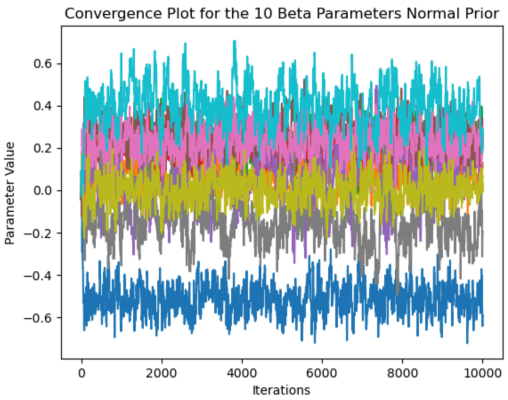
meaning that we accept the proposal with probability $r$.

**Basic diagnostics for the convergence of MH algorithm**

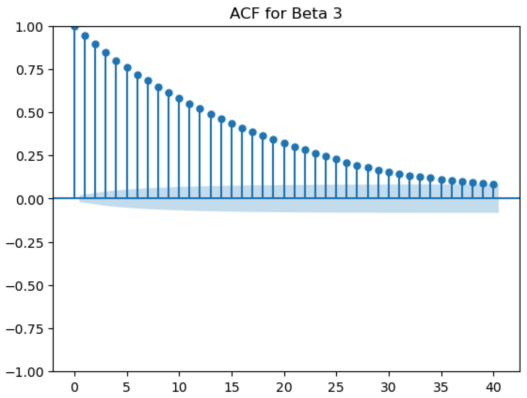
For convergence diagnostics we used mainly 2 tests.

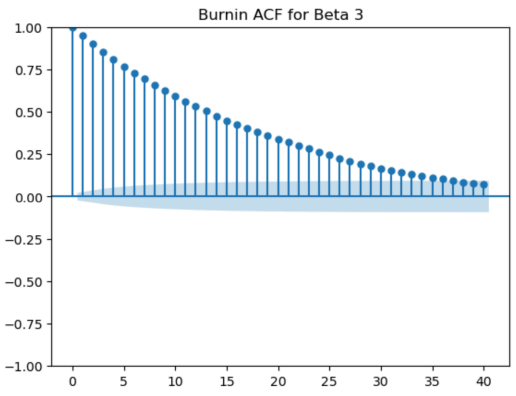
First we looked at the plot of all the beta parameters across the Metropolis Hastings iterations. We notice that most betas begin to converge after the first few hundred interactions.

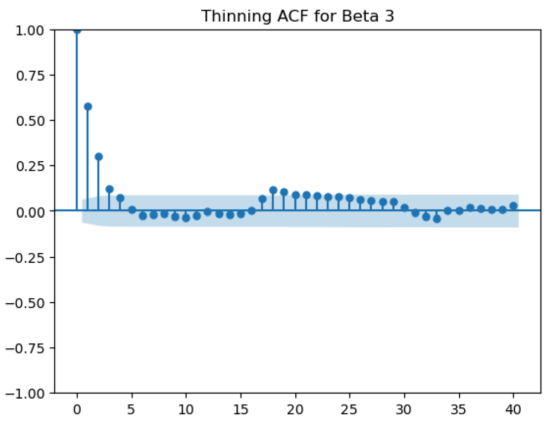



Secondly, we plot the **ACF** (autocorrelation function) of each individual beta with its past values, in this case we looked at the correlation between beta and its past 40 lags. We want the ACF curve to drop below the 95% credible bands as beta gets further from the initial value. While for the whole sample data convergence is slower, we notice that for most betas using burn-in after 1000 iterations leads to convergence at 40 lags. However, beta 4 and 7 are still resistant, but after using **thinning** we see indeed that all betas converge. In the code, we plot the ACF for all 10 betas for the whole sample, the sample with thinning and the sample with **burn-in**. Below we have an example of the 3 ACF plots for beta 3 which is the beta of the cholesterol *ldl*.



**Discussion of the prior distributions chosen in Metropolis Hastings Probit model**

In our first implementation of the algorithm, we chose the uniform prior, which is a non-informative prior. This is a good choice when we have no prior information at all about the data at hand. In this case, the maximum a posteriori estimator and the maximum likelihood estimator coincide, as the posterior function is equal to the likelihood.
This way, we were able to do statistical inference via Metropolis Hastings without injecting our subjective opinion into the analysis which is contrary to the goal of making scientific inference as objective as possible.

To compare the results of the noninformative prior, we also used a standard normal as prior, given that it is a symmetric distribution centred at zero but with values being less likely the further they are from the mean. This prior also allowed us to regularise our results by decreasing the probability of getting large values, as using a normal prior corresponds to performing a ridge regression (L2 regularisation).

**<u>Discussion on the fitted models & interpretation of the parameters</u>**

In probit regression, the coefficients represent the change in the z-score (standard normal deviate) for a one-unit change in the corresponding predictor variable. A positive coefficient implies an increase in the z-score, while a negative coefficient suggests a decrease. Finally, the magnitude of the coefficient indicates the strength of the effect.

In the case of colesterol represented by the variable *ldl* we see that the point estimate we chose for its coefficient - the average of last 100 *ldl* betas - is around 0.22 and is significant at the 5% level according to our credible intervals hence it has a relatively high positive impact for the z-score. Also, the point estimates of the betas of *famhist* (~0.27) and *age* (~0.39) are significant at 5%, and we see that among all positive betas they are the two with the highest positive impact for the z-score. As expected, *tobacco* usage leads to an increased risk of developing cardiac disease. Lastly, although the -0.17 beta of *obesity* seems surprising, it is not significant at 5% level, which suggests that the usually observed correlation between obesity and heart disease is explained entirely by other factors that are strongly correlated with obesity, such as high levels of low-density cholesterol.
Also surprising is the low relative impact on the z-score of *sbp* (blood pressure), but then we notice *sbp* is not significant at 5% level. The results hold in credible intervals for both posteriors (the one with uniform and normal gaussian prior) as shown in the tables below.

| | Betas uniform | 2.5% Bound | 97.5% Bound | 5% Significance |
|---|---|---|---|---|
| **constant** | -0.520874 | -0.659918 | -0.383349 | True |
| **sbp** | 0.081710 | -0.052851 | 0.216755 | False |
| **tobacco** | 0.229182 | 0.085545 | 0.388720 | True |
| **ldl** | 0.223603 | 0.075938 | 0.380268 | True |
| **adiposity** | 0.090660 | -0.161072 | 0.362183 | False |
| **famhist** | 0.270302 | 0.142635 | 0.401762 | True |
| **typea** | 0.233738 | 0.085286 | 0.375168 | True |
| **obesity** | -0.172210 | -0.379356 | 0.032641 | False |
| **alcohol** | -0.000601 | -0.138043 | 0.129696 | False |
| **age** | 0.386629 | 0.201365 | 0.577206 | True |

| | Betas gaussian | 2.5% Bound | 97.5% Bound | 5% Significance |
|---|---|---|---|---|
| **constant** | -0.513747 | -0.643607 | -0.370908 | True |
| **sbp** | 0.072575 | -0.064579 | 0.203708 | False |
| **tobacco** | 0.230080 | 0.098930 | 0.363383 | True |
| **ldl** | 0.219334 | 0.082597 | 0.350737 | True |
| **adiposity** | 0.062336 | -0.199996 | 0.302300 | False |
| **famhist** | 0.267332 | 0.131415 | 0.411350 | True |
| **typea** | 0.232484 | 0.098728 | 0.376748 | True |
| **obesity** | -0.145045 | -0.346461 | 0.068710 | False |
| **alcohol** | -0.000455 | -0.132180 | 0.129361 | False |
| **age** | 0.395891 | 0.214069 | 0.580468 | True |