**Bayesian Analysis of the Multiple Linear Regression (MLR) Model (STT465)**

Gustavo de los Campos

([gustavoc@msu.edu)](mailto:gustavoc@msu.edu))

In this notes I summarize the discussion we had in class about the multiple linear regression model. Some of the topics addressed in this note are discussed in the book with a slightly different approach. The reference chapters for this topic in the book are:

**1. Setting the stage**

In a MLR model a quantitative outcome () is expressed as the sum of a linear function of covariates plus an error term (

[1]

Typically, the first covariate is used to accommodate an intercept, that is

The *simple linear regression* is a special case of [1] with *p=2*, , here is the ‘y-intercept’ and is the slope of the regression.

***2. Matrix representation***

Expression [1] is the data-equation for the ith datum. Sacking these equations, we obtain the following system

where is a vector of covariates for the ith subject. In compact notation we write

[2]

where: is an n-dimensional vector with the response, is an nxp matrix (rows corresponding to individuals and columns correspond to predictors), and is a vector of model residuals. Expression [2] is the standard matrix representation of a multiple linear regression model.

***3. Ordinary Least Squares (OLS)***

OLS estimates are obtained by minimizing the residual sum of squares, that is

[3]

To solution to the above optimization problem has a closed-form, if f **X** is full-column rank we have that . The entry [OLS.md](https://github.com/gdlc/STT465/blob/master/OLS.md) in our repository demonstrates how to obtain OLS estimates in R.

***4. Bayesian model***

A Bayesian model is defined by the likelihood function and the prior distribution of the unknowns, we begin by describing each of these elements of the model.

*4.1. Likelihood with IID Gaussian errors*

If the error terms are independent and identically distributed, each following a normal distribution with null mean and variance , denoted as , then, it follows from [1] and from the properties of the normal distribution that

[4]

From the independence assumption of the error terms we have that the joint distribution of the data given the parameters (i.e., the likelihood function) is

where is the residual sum of squares which is a function of the response (**y**), the incidence matrix for effects (**X**) and the vector of regression coefficients (**b**).

4.2. *Maximum Likelihood estimation*

As with other models in the course, once we set the likelihood we see a short detour to discuss Maximum likelihood (ML). ML estimates are obtained by maximizing expression [5] with respect to the unknown parameters . Equivalently, MLE can be obtained by minimizing the negative log-likelihood that is

It can be seen that the MLE of **b** is the argument that minimizes the RSS; therefore, in this model MLE(**b**)=. Differentiating with respect to the error variance renders the following first order condition .

*4.3. Prior distribution-I*

The parameters entering in the likelihood (expression [5]) include the vector of regression coefficients and the error variance. The Bayesian model is completed by specifying a prior for these unknown parameters. A standard approach consists of assigning IID normal priors to the regression coefficients, , and an independent scaled-inverse Chi-square prior for the error variance, ; therefore,

[6]

*4.4.Controlling the influence of the prior on inferences*

An important consideration in Bayesian analyses is how much inferences will be influenced by the prior distribution. In some cases, we may want to use weekly informative priors. In other cases (e.g., regressions involving large number of coefficients) informative priors will yield more precise estimates. The hyper-parameters are the parameters that index the prior. These need to be specified by the analyst (we will discuss extensions later on where some may be inferred from data), in this case the hyper-parameters include

Since the prior on regression coefficient is normal, the amount of information provided by this prior can be controlled by specifying the variance. Choosing a very large makes the prior for effects “flat”, in this case inferences about effects will be largely driven by the information provided by the likelihood and Bayesian estimates of regression coefficients will be very close to MLE(**b**). The expected value, mean and mode of the scaled-inverse chi-square are and , respectively. The mean is defined for . One possibility is to choose to be small but greater than 2 (e.g., and then use either the equation for the man or the model to solve for the scale as a function of df and the expected error variance, for instance, using the equation for the mode, we can use . Here, is the sample variance of the data and our prior guess about the proportion of variance that will be explained by the model. If the is chosen to be small, these rules will lead to a relatively weak prior (although the scaled-inverse chi-square cannot be made strictly flat).

4*.5. Joint Posterior Distribution-I*

According to Bayes theorem the joint posterior distribution is proportional to the product of the likelihood times the prior

Therefore,

[7]

Expression [7] does not have a closed form; therefore, inferences on this model are often carried out using Monte Carlo (MC) methods. There are several approaches that can be followed, here we focus on the Gibbs Sampler.

***5. Gibbs Sampler-I***

In a Gibbs sampler samples from the posterior distribution are collected by sampling from the fully conditional distributions. An outline of the algorithm is as follows



Above, nIter is an algorithm-control-variable that determines the number of samples to be collected.

Before we present an implementation of the Gibbs sampler we need to derive the fully conditional distributions.

*5.1. Fully-conditional distributions*

To derive the fully-conditionals: (i) remove for posterior any proportionality constant that does not involve the unknown parameter and (ii) combine terms in search for a closed form. Since we are using conjugate priors all the fully conditionals will have closed forms.

*Regression coefficients*

After removing proportionality constants that do not involve we get

We now need to combine the two exponentials. The term can be written as where is an ‘off-set’ formed by substracting from the data the contribution to the regression function of all the terms that do not involve the kth regression coefficient. Since we are sampling from all the regression coefficients, except the kth one, can be treated as known. Therefore, can be treated as data, thus

The RSS can be written as . Likewise, the quadratic form entering in the second to the right exponential can be written as ; therefore,

The first and third exponentials do not involve the unknown parameter, thus

or

We now combine the terms involving and

or

where and . The expression above has a form very similar to the kernel of a normal distribution for the random variable with mean and variance . The term being missing in the exponential is . This term does not involve the unknown coefficient; therefore we can write

Combining the two exponentials renders

Therefore, we conclude that the fully-conditional density is normal, specifically

[8]

where, as stated before and .

*Error variance*

Removing from the joint posterior distribution (expression [7]) the terms that do not involve the error variance we get

Note that because we are sampling from the fully conditional distribution of the error variance, all the regression coefficients can be treated as known; thus, can be treated as known constant.

Combining the exponentials yields

[9]

The above expression is proportional to the kernel of a Scaled-inverse Chi-squared density with scale parameter and degree of freedom .

*5.2. Implementation*

The algorithm described in Box 1 is implemented in the script provided in [gibbsMLR.md](https://github.com/gdlc/STT465/blob/master/gibbsMLR.md). This implementation uses a few ‘computational tricks’, some are listed below.

- Some quantities required to sample from expressions [8] and [9] do not vary across iterations of the sampler; we compute this quantities in advance and do not re-compute them during the execution of the sampler (an example of this is the sum of squares of each of each of the predictors).

- The mixing of the algorithm can be improved by making each of the predictors orthogonal to the incidence vector for the intercept, this is achieved by centering each predictor around its mean.

-

[in progress!!]