STT465: Bayesian Statistical Methods (Fall 2025)

Modeling the odds of malignancy for a breast mass using Bayesian logistic regression

Lowell Monis[*,†]

Abstract. This report presents a Bayesian approach to classifying breast cancer diagnoses (malignant vs. benign) using 30 features computed from a digitized image of a fine needle aspirate of breast masses. Bayesian logistic regression is employed with Metropolis-Hastings sampling for inference. Bayesian variable selection is implemented to identify the most important diagnostic features.

Introduction

Breast cancer remains one of the most prevalent and life-threatening diseases worldwide. Early and accurate diagnosis is crucial for effective treatment and improved patient outcomes. Fine needle aspiration (FNA) is a minimally invasive diagnostic procedure that extracts cells from a breast mass for analysis. The advent of digital imaging technology has enabled the extraction of quantitative features from these cell nuclei, opening new avenues for automated diagnosis using statistical and machine learning methods.

The Wisconsin Diagnostic Breast Cancer (WDBC) dataset, curated by Wolberg et al. (1993) at the University of Wisconsin, represents a landmark contribution to this field. The dataset contains measurements computed from digitized images of fine needle aspirates, capturing various characteristics of cell nuclei that distinguish malignant from benign breast masses. Previous research has demonstrated that these features contain sufficient information to achieve highly accurate classification, with some studies reporting predictive accuracy exceeding 97% using linear programming-based methods.

While classical machine learning approaches have proven effective, they often provide point estimates without quantifying the uncertainty in predictions or model parameters. Bayesian methods offer a principled framework for incorporating prior knowledge, quantifying uncertainty, and performing model selection in a probabilistic manner. In this analysis, a Bayesian logistic regression approach is adopted to model the probability of malignancy as a function of the extracted features. This approach not only provides predictions but also yields posterior information for all model parameters, enabling greater understanding of the effect of each individual quantifier and the role it plays in the malignancy of a breast mass when prior information is provided to the model.

Objectives

This study addresses the following question:

> Can an accurate Bayesian logistic regression model be developed to predict a breast cancer diagnosis from digitized cell nuclei features, and which of the thirty morphological characteristics are most predictive of malignancy and should be prioritized in clinical diagnosis?

Specifically, this study will:

- Build a Bayesian logistic regression model using all 30 available features and assess its predictive accuracy
- Quantify the importance of each feature through posterior probabilities
- Compare the Bayesian variable selection results with classical model selection approaches (AIC/BIC)

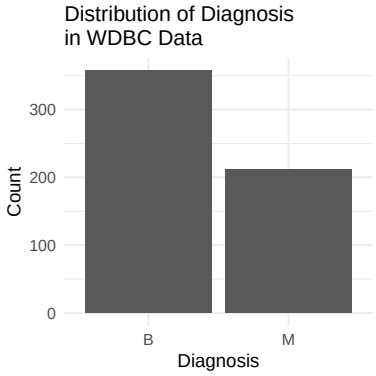
The Data

The Wisconsin Diagnostic Breast Cancer dataset was collected at the University of Wisconsin Clinical Sciences Center starting around 1989. It contains measurements from 569 patients who underwent fine needle aspiration of breast masses. For each patient, a digitized image of the aspirate was analyzed to compute quantitative features describing the characteristics of cell nuclei present in the image. The first feature, `id`, identifies each sample in the dataset. This feature can be discarded since it cannot be used as a predictor variable in the analysis.

```
## 'data.frame':    569 obs. of  32 variables:
##  $ id                : int  842302 842..
##  $ diagnosis         : chr  "M" "M" ""..
##  $ radius1           : num  18 20.6 19..
##  $ texture1          : num  10.4 17.8 ..
##  $ perimeter1        : num  122.8 132...
##  $ area1             : num  1001 1326 ..
##  $ smoothness1       : num  0.1184 0.0..
##  $ compactness1      : num  0.2776 0.0..
##  $ concavity1        : num  0.3001 0.0..
##  $ concave_points1   : num  0.1471 0.0..
##  $ symmetry1         : num  0.242 0.18..
##  $ fractal_dimension1: num  0.0787 0.0..
##  $ radius2           : num  1.095 0.54..
##  $ texture2          : num  0.905 0.73..
##  $ perimeter2        : num  8.59 3.4 4..
##  $ area2             : num  153.4 74.1..
##  $ smoothness2       : num  0.0064 0.0..
##  $ compactness2      : num  0.049 0.01..
##  $ concavity2        : num  0.0537 0.0..
##  $ concave_points2   : num  0.0159 0.0..
##  $ symmetry2         : num  0.03 0.013..
##  $ fractal_dimension2: num  0.00619 0...
##  $ radius3           : num  25.4 25 23..
##  $ texture3          : num  17.3 23.4 ..
##  $ perimeter3        : num  184.6 158...
##  $ area3             : num  2019 1956 ..
##  $ smoothness3       : num  0.162 0.12..
##  $ compactness3      : num  0.666 0.18..
##  $ concavity3        : num  0.712 0.24..
##  $ concave_points3   : num  0.265 0.18..
##  $ symmetry3         : num  0.46 0.275..
##  $ fractal_dimension3: num  0.1189 0.0..
```

The dataset contains one binary response variable where B = benign, and M = malignant.

```
## # A tibble: 2 x 3
##   diagnosis count percentage
##   <chr>     <int>      <dbl>
## 1 B           357       62.7
## 2 M           212       37.3
```



Apart from the response variable, there are also 30 continuous predictor variables. The 30 real-valued predictor variables are organized into three groups of ten measurements each.

- If the variable ends with 1, these are the mean values averaging each measured value across all cell nuclei in the image.
- If the variable ends with 2, this indicates the standard error of each measured value across all cell nuclei.
- If the variable ends with 3, the feature stores the mean of the three largest (or the worst possible measurements) of each measured value.

The ten base measurements computed for each cell nucleus are:

- *Radius*: Mean distance from center to points on the perimeter.
- *Texture*: Standard deviation of gray-scale values.
- *Perimeter*: Perimeter of the nucleus.
- *Area*: Area of the nucleus.

- *Smoothness*: Local variation in radius lengths
- *Compactness*: Computed as perimeter^2 / area - 1.0
- *Concavity:* Severity of concave portions of the contour
- *Concave points*: Number of concave portions of the contour
- *Symmetry*: Symmetry of the nucleus
- *Fractal dimension*: "Coastline approximation" - 1

```
##     radius1          texture1
##  Min.   : 6.981   Min.   : 9.71
##  1st Qu.:11.700   1st Qu.:16.17
##  Median :13.370   Median :18.84
##  Mean   :14.127   Mean   :19.29
##  3rd Qu.:15.780   3rd Qu.:21.80
##  Max.   :28.110   Max.   :39.28
##    perimeter1         area1
##  Min.   : 43.79   Min.   : 143.5
##  1st Qu.: 75.17   1st Qu.: 420.3
##  Median : 86.24   Median : 551.1
##  Mean   : 91.97   Mean   : 654.9
##  3rd Qu.:104.10   3rd Qu.: 782.7
##  Max.   :188.50   Max.   :2501.0
##   smoothness1       compactness1
##  Min.   :0.05263   Min.   :0.01938
##  1st Qu.:0.08637   1st Qu.:0.06492
##  Median :0.09587   Median :0.09263
##  Mean   :0.09636   Mean   :0.10434
##  3rd Qu.:0.10530   3rd Qu.:0.13040
##  Max.   :0.16340   Max.   :0.34540
##    concavity1      concave_points1
##  Min.   :0.00000   Min.   :0.00000
##  1st Qu.:0.02956   1st Qu.:0.02031
##  Median :0.06154   Median :0.03350
##  Mean   :0.08880   Mean   :0.04892
##  3rd Qu.:0.13070   3rd Qu.:0.07400
##  Max.   :0.42680   Max.   :0.20120
##    symmetry1      fractal_dimension1
##  Min.   :0.1060   Min.   :0.04996
##  1st Qu.:0.1619   1st Qu.:0.05770
##  Median :0.1792   Median :0.06154
##  Mean   :0.1812   Mean   :0.06280
##  3rd Qu.:0.1957   3rd Qu.:0.06612
##  Max.   :0.3040   Max.   :0.09744
##     radius2          texture2
##  Min.   :0.1115   Min.   :0.3602
##  1st Qu.:0.2324   1st Qu.:0.8339
##  Median :0.3242   Median :1.1080
##  Mean   :0.4052   Mean   :1.2169
##  3rd Qu.:0.4789   3rd Qu.:1.4740
##  Max.   :2.8730   Max.   :4.8850
##    perimeter2         area2
##  Min.   : 0.757   Min.   :  6.802
##  1st Qu.: 1.606   1st Qu.: 17.850
##  Median : 2.287   Median : 24.530
##  Mean   : 2.866   Mean   : 40.337
##  3rd Qu.: 3.357   3rd Qu.: 45.190
##  Max.   :21.980   Max.   :542.200
##   smoothness2        compactness2
##  Min.   :0.001713   Min.   :0.002252
##  1st Qu.:0.005169   1st Qu.:0.013080
##  Median :0.006380   Median :0.020450
##  Mean   :0.007041   Mean   :0.025478
##  3rd Qu.:0.008146   3rd Qu.:0.032450
##  Max.   :0.031130   Max.   :0.135400
##    concavity2      concave_points2
##  Min.   :0.00000   Min.   :0.000000
##  1st Qu.:0.01509   1st Qu.:0.007638
##  Median :0.02589   Median :0.010930
##  Mean   :0.03189   Mean   :0.011796
##  3rd Qu.:0.04205   3rd Qu.:0.014710
##  Max.   :0.39600   Max.   :0.052790
##    symmetry2        fractal_dimension2
##  Min.   :0.007882   Min.   :0.0008948
##  1st Qu.:0.015160   1st Qu.:0.0022480
##  Median :0.018730   Median :0.0031870
##  Mean   :0.020542   Mean   :0.0037949
##  3rd Qu.:0.023480   3rd Qu.:0.0045580
##  Max.   :0.078950   Max.   :0.0298400
##     radius3          texture3
##  Min.   : 7.93   Min.   :12.02
##  1st Qu.:13.01   1st Qu.:21.08
##  Median :14.97   Median :25.41
##  Mean   :16.27   Mean   :25.68
##  3rd Qu.:18.79   3rd Qu.:29.72
##  Max.   :36.04   Max.   :49.54
##    perimeter3         area3
##  Min.   : 50.41   Min.   : 185.2
##  1st Qu.: 84.11   1st Qu.: 515.3
##  Median : 97.66   Median : 686.5
##  Mean   :107.26   Mean   : 880.6
##  3rd Qu.:125.40   3rd Qu.:1084.0
##  Max.   :251.20   Max.   :4254.0
##   smoothness3        compactness3
##  Min.   :0.07117   Min.   :0.02729
##  1st Qu.:0.11660   1st Qu.:0.14720
##  Median :0.13130   Median :0.21190
```

```
##  Mean   :0.13237   Mean   :0.25427
##  3rd Qu.:0.14600   3rd Qu.:0.33910
##  Max.   :0.22260   Max.   :1.05800
##    concavity3     concave_points3
##  Min.   :0.0000   Min.   :0.00000
##  1st Qu.:0.1145   1st Qu.:0.06493
##  Median :0.2267   Median :0.09993
##  Mean   :0.2722   Mean   :0.11461
##  3rd Qu.:0.3829   3rd Qu.:0.16140
##  Max.   :1.2520   Max.   :0.29100
##    symmetry3      fractal_dimension3
##  Min.   :0.1565   Min.   :0.05504
##  1st Qu.:0.2504   1st Qu.:0.07146
##  Median :0.2822   Median :0.08004
##  Mean   :0.2901   Mean   :0.08395
##  3rd Qu.:0.3179   3rd Qu.:0.09208
##  Max.   :0.6638   Max.   :0.20750
```

All feature values were recorded with four significant digits, and the dataset contains no missing values according to its description. Previous research has shown that these 30 features contain sufficient information to achieve linear separability of the two diagnostic classes, with particularly strong predictive power coming from features such as worst area, worst smoothness, and mean texture.

Preprocessing

The first, and most important preprocessing step while preparing the data for analysis is to encode the response variable into something more acceptable to a numerical model. Since this model aims to predict malignancy in a breast mass, a response that the mass is malignant is set to 1, while a response that the mass is benign is set to 0.

$$y = \begin{cases} 0, & \text{mass is benign} \\ 1, & \text{mass is malignant} \end{cases}$$

```
##
## y     B   M
##   0 357   0
##   1   0 212
```

The design matrix needs to be constructed as follows:

$$\mathbf{X} = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,30} \\ 1 & x_{2,1} & x_{2,2} & \cdots & x_{2,30} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & x_{n,2} & \cdots & x_{n,30} \end{bmatrix}_{n \times p}$$

where $n = 569$ patients, $p = 31$ parameters (1 intercept + 30 features), and $x_{i,j}$ represents the value of feature j for patient i.

To do this, the features are bound with an all-ones vector column-wise:

$$\mathbf{X} = [\mathbf{1}_n \mid \mathbf{X}_{\text{features}}] \in \mathbb{R}^{n \times p}$$

Following through the observations made earlier about the difference in the scales of each feature, the design matrix can be standardized to $\mathcal{N}(0, 1)$ to improve the performance of algorithms like Metropolis-Hastings, and make the interpretation of coefficients more meaningful.

$$\mathbf{X}_{\text{scaled}} = \frac{\mathbf{X} - \mu}{\sigma}$$

Model Setup

Since the response variable for this analysis is binary (malignant or benign), a logistic regression model is the appropriate choice for classification. The goal is to estimate the probability that a breast mass is malignant given the observed feature measurements. A Bayesian approach will be employed to quantify uncertainty in both predictions and parameter estimates.

For each observation $i = 1, \ldots, n$, where $n = 569$ patients, the response variable Y_i follows a Bernoulli distribution:

$$Y_i \mid \mathbf{x}_i, \beta \sim \text{Bernoulli}(\theta_i)$$

Here $Y_i \in \{0, 1\}$ indicates whether patient i has a malignant or benign mass. $\mathbf{x}_i$ is the vector of 30 feature measurements for patient i (and an intercept value), and $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_{30})^T$ is the vector of regression coefficients to be estimated,

containing the intercept and 30 coefficients corresponding to each feature, which quantify the effect of each measurement on the odds of malignancy. The probability of malignancy for patient i is modeled:

$$\theta_i = P(Y_i = 1 \mid \mathbf{x}_i, \boldsymbol{\beta}) = \frac{e^{\boldsymbol{\beta}^T \mathbf{x}_i}}{1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}}$$

This function, known as the logistic or sigmoid function, ensures that predicted probabilities lie between 0 and 1, making it suitable for binary classification problems.

Equivalently, this can also be expressed using its logit transformation as follows:

$$\begin{aligned} \text{logit}(\theta_i) &= \log\left(\frac{\theta_i}{1-\theta_i}\right) \\ &= \boldsymbol{\beta}^T \mathbf{x}_i \\ &= \beta_0 + \sum_{j=1}^{30} \beta_j x_{i,j} \end{aligned}$$

The left-hand side represents the log-odds of malignancy, which is modeled as a linear combination of the features.

The likelihood function quantifies the probability of observing the response $\mathbf{y} = (y_1, y_2, \ldots, y_n)^T$ given the feature matrix $\mathbf{X}$ and coefficient vector $\boldsymbol{\beta}$. Under the assumption that observations are independent, the joint likelihood is the product of individual Bernoulli probabilities:

$$\begin{aligned} p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\beta}) &= \prod_{i=1}^{n} \theta_i^{y_i} (1-\theta_i)^{1-y_i} \\ &= \prod_{i=1}^{n} \frac{e^{y_i \boldsymbol{\beta}^T \mathbf{x}_i}}{1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}} \end{aligned}$$

The expression follows from the Bernoulli probability mass function—when $y_i = 1$ (malignant), the contribution is θ_i, and when $y_i = 0$ (benign), the contribution is $1 - \theta_i$. The expansion of the expression is provided by substituting the logistic function θ_i.

For numerical stability and computational efficiency, it is standard practice to work with the log-likelihood. Taking the natural logarithm of both sides and applying the properties of logarithms, one can obtain:

$$\begin{aligned} \log p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\beta}) &= \sum_{i=1}^{n} \log\left[\frac{e^{y_i \boldsymbol{\beta}^T \mathbf{x}_i}}{1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}}\right] \\ &= \sum_{i=1}^{n} \left[\log\left(e^{y_i \boldsymbol{\beta}^T \mathbf{x}_i}\right) - \log\left(1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}\right)\right] \\ &= \sum_{i=1}^{n} \left[y_i \boldsymbol{\beta}^T \mathbf{x}_i - \log\left(1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}\right)\right] \end{aligned}$$

The log-likelihood form is computationally advantageous due to its ability to convert products into sums, which are numerically more stable, and avoid potential underflow issues when multiplying many small probabilities. It is also the foundation for constructing the acceptance ratio in the Metropolis-Hastings algorithm, which will be explored later in this study.

Prior Distribution

In Bayesian inference, prior distributions encode beliefs about parameters before observing the data. For this analysis, a weakly informative multivariate normal prior is adopted for the coefficient vector $\boldsymbol{\beta}$:

$$\boldsymbol{\beta} \sim \mathcal{N}(\mu_o, \Sigma_0)$$

where $\mu_0 = \mathbf{0}_p$, $p = 31$, giving a a 31-dimensional all-zeros vector, and $\Sigma_0 = \sigma_0^2 \mathbf{I}_p$, where $\sigma_0^2 = 100$, and $\mathbf{I}_p$ is the 31×31 identity matrix. This choice reflects several important considerations. First, centering the prior at zero with prior mean set to 0 indicates no prior preference for the direction or magnitude of effects–it is not assumed *a priori* that any feature increases or decreases the probability of malignancy. Second, the large prior variance, set to 100, makes this a weakly informative prior, meaning the data will largely determine the posterior estimates, rather than being heavily influenced by prior assumptions. The independence

structure impose by the diagonal covariance matrix Σ_0 assumes that, prior to seeing the data, there is no reason to believe the coefficients are correlated with one another.

Despite being weakly informative, this prior serves an important regularization function by gently discouraging extremely large coefficient values that might lead to overfitting or numerical instability. However, unlike conjugate priors in simpler models such as the normal-normal model for linear regression, this prior does not combine with the logistic likelihood to produce a posterior distribution with a known, closed-form expression. Consequently, the posterior summaries cannot be obtained analytically, and computational methods must be used.

Posterior Distribition

The posterior distribution combines the likelihood and prior through Bayes' theorem:

$$p(\boldsymbol{\beta} \mid \mathbf{y}, \mathbf{X}) = \frac{p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\beta}) \cdot p(\boldsymbol{\beta})}{p(\mathbf{y} \mid \mathbf{X})}$$

The denominator $p(\mathbf{y} \mid \mathbf{X}) = \int p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\beta}) p(\boldsymbol{\beta}) d\boldsymbol{\beta}$ is the marginal likelihood, which serves as a normalizing constant ensuring the posterior integrates to one. Since this integral is difficult to handle for logistic support, the unnormalized posterior is used, which is proportional to the product of the likelihood and the prior:

$$\begin{aligned} p(\boldsymbol{\beta} \mid \mathbf{y}, \mathbf{X}) &\propto p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\beta}) \cdot p(\boldsymbol{\beta}) \\ &\propto \left[\prod_{i=1}^{n} \frac{e^{y_i \boldsymbol{\beta}^T \mathbf{x}_i}}{1 + e^{\boldsymbol{\beta}^T \mathbf{x}_i}} \right] \cdot e^{-\frac{1}{2\sigma_0^2} \boldsymbol{\beta}^T \boldsymbol{\beta}} \end{aligned}$$

The first term is the likelihood contribution as computed above, representing the fit of the model to the observed data, while the second term is the prior contribution, which penalizes coefficients with large magnitudes. The product of these two components balances data fit with regularization.

Once again, computational methods are needed due to the fact that the posterior distribution does not have a closed-form expression. This is because the logistic likelihood is not conjugate to the normal prior. The product of these two densities does not simplify to a recognizable probability distribution. As a result, posterior means, variances, and credible intervals cannot be computed using direct integration or algebraic manipulation.

This necessitates the use of Markov Chain Monte Carlo methods to generate samples from the posterior distribution. By drawing a large number of samples $\boldsymbol{\beta}^{(1)}, \boldsymbol{\beta}^{(2)}, \ldots, \boldsymbol{\beta}^{(S)}$ from $p(\boldsymbol{\beta} \mid \mathbf{y}, \mathbf{X})$ for a total of S simulations, the posterior summaries can be approximated empirically: posterior means can be estimated by sample averages, and credible intervals can be constructed from sample quantiles. The Metropolis-Hastings algorithm, described in the following section, provides the computational framework for generating these samples. The Gibbs sampler is not directly applicable, since the full conditional distributions do not have closed-form expressions either.

Parameter Tuning

Algorithm Overview

Since the posterior distribution $p(\boldsymbol{\beta} \mid \mathbf{y}, \mathbf{X})$ lacks a closed-form expression, it cannot be sampled from directly using standard Gibbs sampling methods. Instead, the Metropolis algorithm is employed. This is a Markov Chain Monte Carlo (MCMC) technique that generates a sequence of samples $\boldsymbol{\beta}^{(1)}, \boldsymbol{\beta}^{(2)}, \ldots, \boldsymbol{\beta}^{(S)}$ that approximate draws from the posterior distribution. The key idea is to construct a Markov chain whose stationary distribution is the target posterior. After the chain converges, samples can be used to estimate posterior means, credible intervals, and other quantities of interest.

The Metropolis algorithm is a special case of the more general Metropolis-Hastings algorithm where the proposal distribution is symmetric– meaning the probability of proposing a move from state A to state B is the same as proposing a move from B to A. This symmetry simplifies the acceptance ratio calculation, as seen below, making the Metropolis algorithm partic-

ularly convenient when symmetric proposals are natural for the problem at hand.

Proposal Distribution

At each iteration k, the algorithm proposes a candidate value $\boldsymbol{\beta}^*$ from a symmetric proposal distribution centered at the current state $\boldsymbol{\beta}^{(k)}$:

$$\boldsymbol{\beta}^* \mid \boldsymbol{\beta}^{(k)} \sim N(\boldsymbol{\beta}^{(k)}, \Sigma_{\text{prop}})$$

This multivariate normal proposal is symmetric because the probability of proposing $\boldsymbol{\beta}^{(k)}$ given $\boldsymbol{\beta}^*$ is the same as the probability of proposing $\boldsymbol{\beta}^*$ given $\boldsymbol{\beta}^{(k)}$. This symmetry causes the proposal ratio to cancel in the Metropolis-Hastings acceptance probability, simplifying computations.

The choice of the proposal covariance matrix Σ_{prop} is critical for the algorithm's efficiency.

$$\Sigma_{\text{prop}} = \sigma^2_{\text{prop}}(\mathbf{X}^T\mathbf{X})^{-1}$$

This form incorporates the correlation structure among the features through the $\mathbf{X}^T\mathbf{X})^{-1}$ term. When features are highly correlated, this proposal allows the algorithm to propose coordinated changes to multiple coefficients simultaneously, improving exploration of the posterior distribution. The scalar tuning parameter σ^2_{prop} controls the overall scale of the proposals: larger values lead to bolder moves through the parameter space, while smaller values result in more conservative steps.

The purpose of tuning the proposal variance is to balance two competing goals. If σ^2_{prop} is too small, the algorithm takes tiny steps and accepts nearly every proposal, but explores the posterior very slowly, leading to a high acceptance rate but poor mixing. Conversely, if it is too large, the algorithm proposes extreme values that are frequently rejecte,d again leading to slow exploration but with lower acceptance rate and poor mixing. Empirical research suggests that an acceptance rate between 20% and 50% typically yields efficient exploration for multivariate problems. σ^2_{prop} is tuned to achieve rates within or close to this range.

Acceptance Ratio

At iteration k, after proposing $\boldsymbol{\beta}^*$, it is decided whether to accept or reject it using the Metropolis acceptance ratio:

$$r = \frac{p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\beta}^*)p(\boldsymbol{\beta}^*)}{p(\mathbf{y} \mid \mathbf{X}, \boldsymbol{\beta}^{(k)})p(\boldsymbol{\beta}^{(k)})}$$

This ratio compares the unnormalized posterior density at the proposed value $\boldsymbol{\beta}^*$ to the density at the current value $\boldsymbol{\beta}^{(k)}$. Notice that the normalizing constant $p(\mathbf{y} \mid \mathbf{X})$ (which is intractable to compute) cancels in this ratio, allowing the evaluation of r using only the likelihood and prior, both of which are computable. If $r \geq 1$, the proposed value has higher posterior density and is always accepted. If $r < 1$, the proposed value is accepted with probability r, ensuring the chan can move to lower-density regions occasionally, which is necessary for convergence to the stationary distribution.

For numerical stability, the acceptance ratio is computed on the logarithmic scale. Probabilities in the likelihood can become extremely small (underflow) or large (overflow), but working with log probabilities avoids these issues. The log acceptance ratio is:

$$\log r = \sum_{i=1}^{n} \left[y_i(\boldsymbol{\beta}^{*T}\mathbf{x}_i - \boldsymbol{\beta}^{(k)T}\mathbf{x}_i) - \log \frac{1 + \exp(\boldsymbol{\beta}^{*T}\mathbf{x}_i)}{1 + \exp(\boldsymbol{\beta}^{(k)T}\mathbf{x}_i)} \right] - \frac{1}{2\sigma_0^2}(\|\boldsymbol{\beta}^*\|^2 - \|\boldsymbol{\beta}^{(k)}\|^2)$$

The first term is the log-likelihood ratio, capturing how much better (or worse) the proposed parameters explain the observed data. The second term is the log-prior ratio, reflecting the change in prior plausibility. The decision rule is:

$$\boldsymbol{\beta}^{(k+1)} = \begin{cases} \boldsymbol{\beta}^* & \text{if } \log u \leq \log r \\ \boldsymbol{\beta}^{(k)} & \text{else} \end{cases}$$

where $u \sim \text{Uniform}(0, 1)$. This acceptance mechanism ensures detailed balance, guaranteeing that the Markov chain converges to the target posterior distribution.

Tuning the Proposal Variance

Before running the full MCMC chain on the model, it is essential to select an appropriate value for σ^2_{prop}. This tuning process involves running short pilot chains, with different candidate values, and monitoring the acceptance rate. The goal is to find a value that produces an acceptance rate between 20% and 50%, which has been shown empirically to provide efficient exploration of the posterior in multivariate settings.

Thus, a grid of candidate values is tested by running short MCMC chains for each value and recording the acceptance rate. The optimal choice is the value whose acceptance rate is closes to 35%, which is the midpoint of the desired range, since there can be times where the rates may not be perfectly within the range. This tuning step is computationally inexpensive relative to the full analysis and substantially improves the quality of posterior samples by ensuring the chain mixes well and explores the posterior efficiently.

Once the optimal σ^2_{prop} is identified, the full MCMC sampler is run for a larger number of iterations using this tuned value. The resulting samples, after discarding an initial burn-in period where the chain adjusts from the prior value, constitute approximate draws from the posterior distribution and form the basis for all subsequent inference.

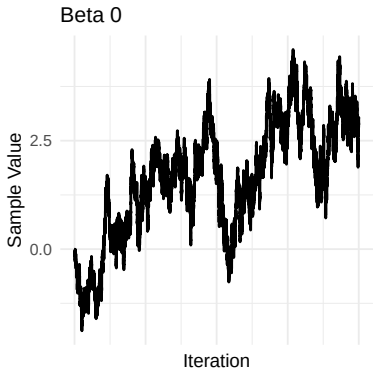
The values being tested as candidates for σ^2_{prop} are $\{0.5, 1, 2, 5, 10, 25, 50, 100\}$. After running chains for 2,000 iterations, the following acceptance rates are computed.

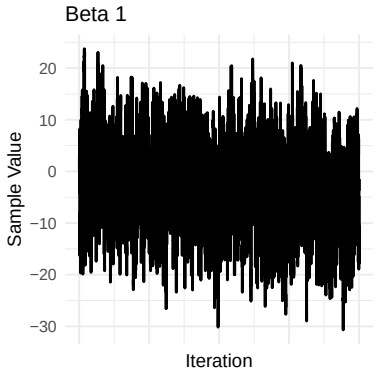
```
##   sigma2_prop acceptance_rate
## 1         0.5           0.645
## 2         1.0           0.579
## 3         2.0           0.482
## 4         5.0           0.315
## 5        10.0           0.176
## 6        25.0           0.066
## 7        50.0           0.028
## 8       100.0           0.025
```

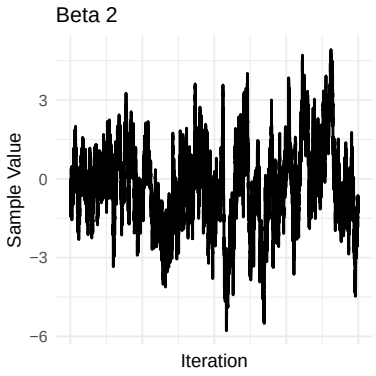
The value with acceptance rate closest to 35% is 5, which will be chosen as the optimal value.

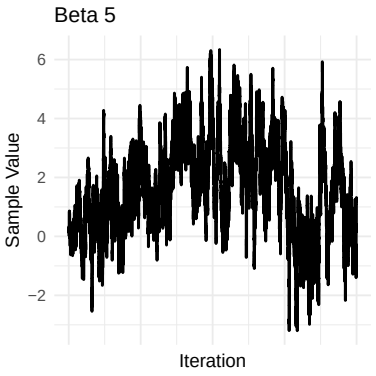
Constructing the Full Model using MCMC Diagnostics

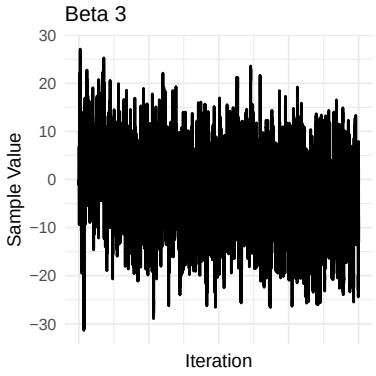
Now that the parameter has been tuned to the most optimal value, the number of iterations can be increased to 20,000 with the parameter set to $\sigma^2_{\text{prop}} = 5.0$. The diagnostic traceplots and autocorrelation function plots can be plotted after this to evaluate the coefficients, and determine the amount of burn-in and thinning needed to create the best version of the full model.

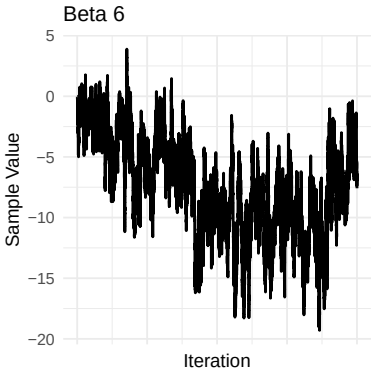


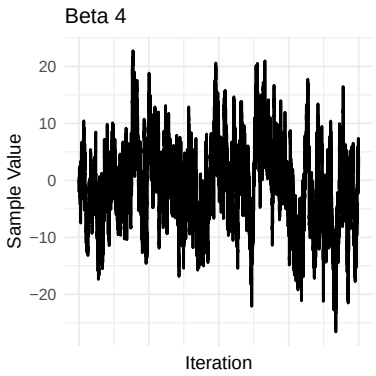


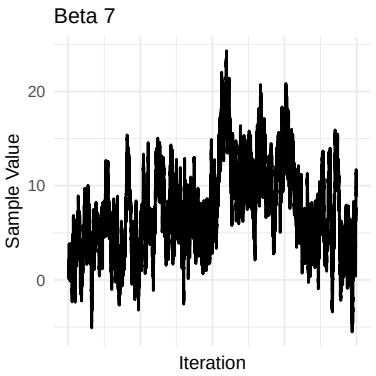

Beta 2
Sample Value
3
0
−3
−6
Iteration

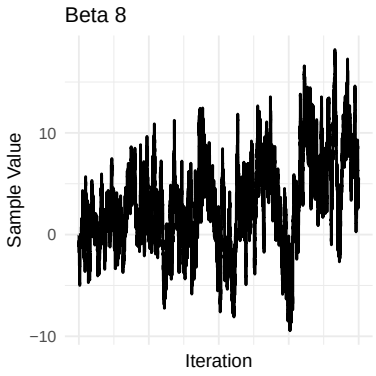

Beta 5
Sample Value
6
4
2
0
−2
Iteration

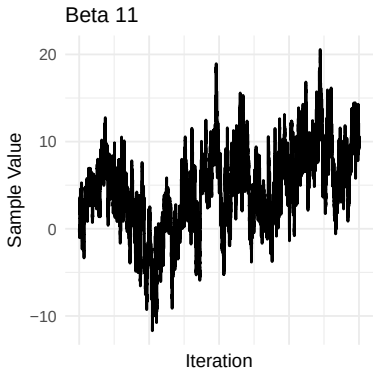

Beta 3
Sample Value
30
20
10
0
−10
−20
−30
Iteration

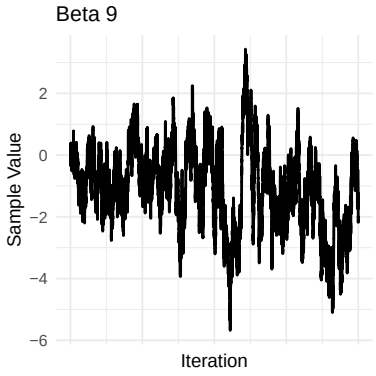

Beta 6
Sample Value
5
0
−5
−10
−15
−20
Iteration

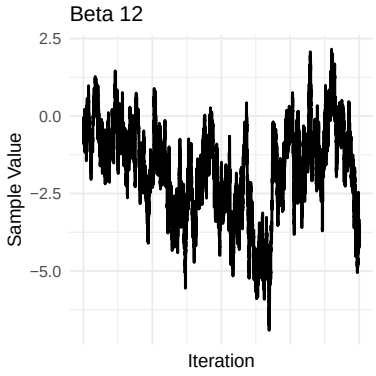

Beta 4
Sample Value
20
10
0
−10
−20
Iteration

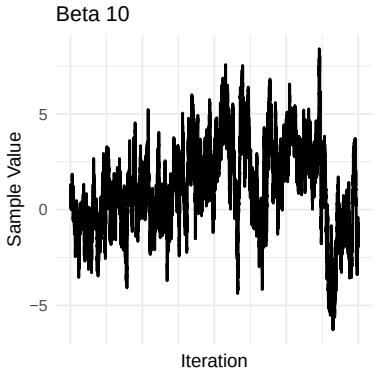

Beta 7
Sample Value
20
10
0
Iteration

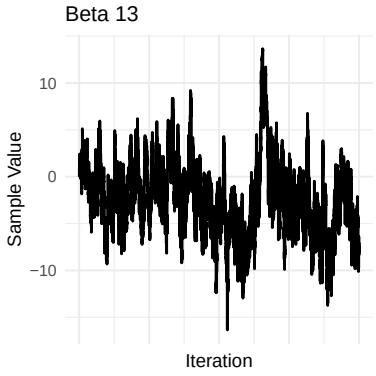

Beta 8
Sample Value
10
0
−10
Iteration

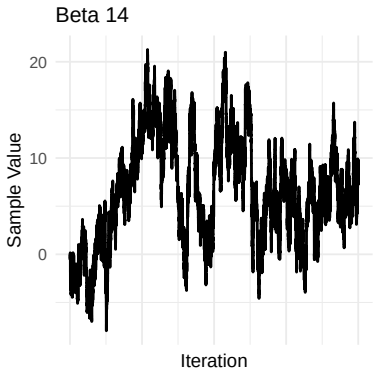

Beta 11
Sample Value
20
10
0
−10
Iteration

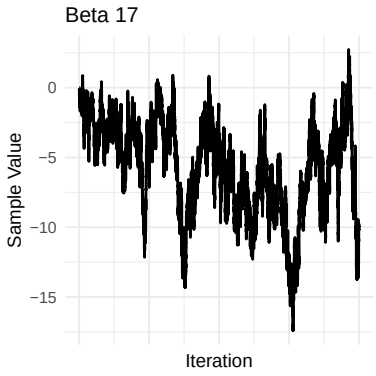

Beta 9
Sample Value
2
0
−2
−4
−6
Iteration

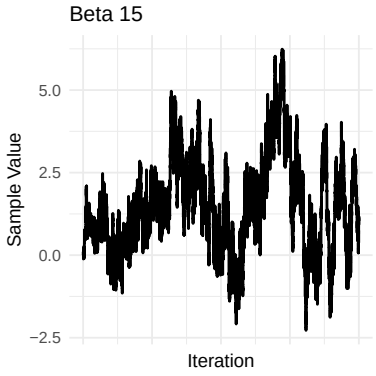

Beta 12
Sample Value
2.5
0.0
−2.5
−5.0
Iteration

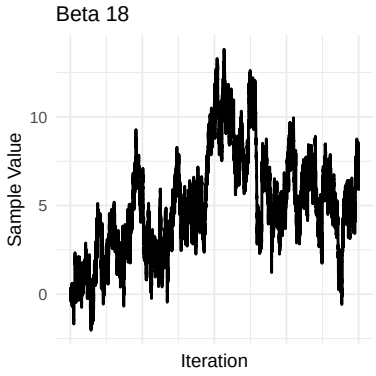

Beta 10
Sample Value
5
0
−5
Iteration

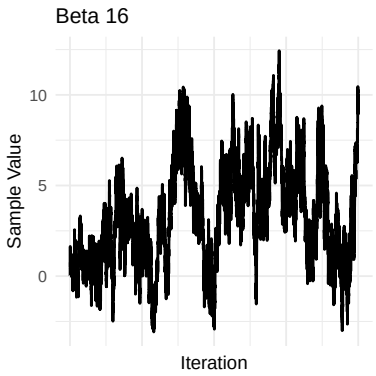

Beta 13
Sample Value
10
0
−10
Iteration

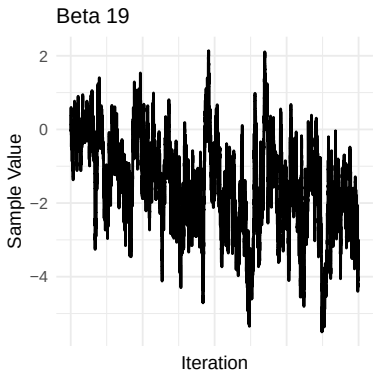

Beta 14
Sample Value
20
10
0
Iteration

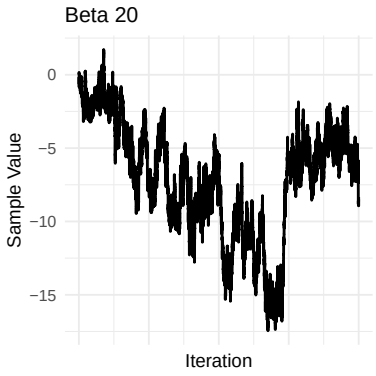

Beta 17
Sample Value
0
−5
−10
−15
Iteration

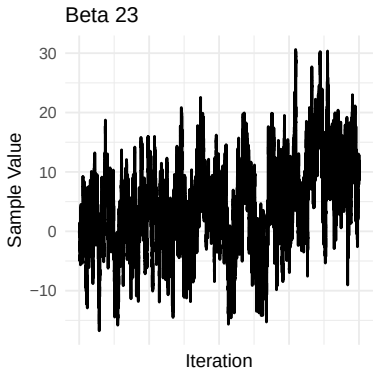

Beta 15
Sample Value
5.0
2.5
0.0
−2.5
Iteration

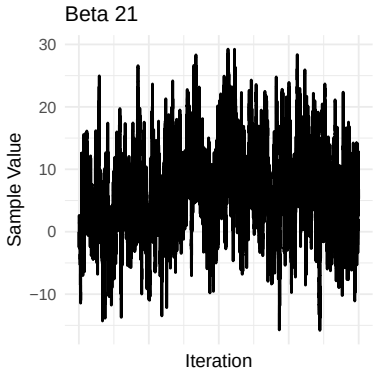

Beta 18
Sample Value
10
5
0
Iteration

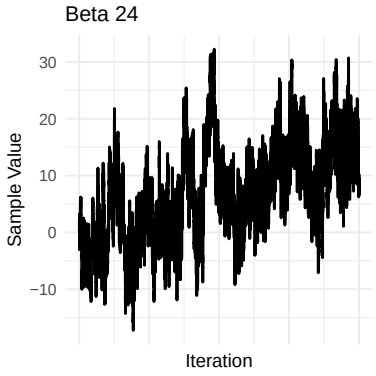

Beta 16
Sample Value
10
5
0
Iteration


Beta 19
Sample Value
2
0
−2
−4
Iteration

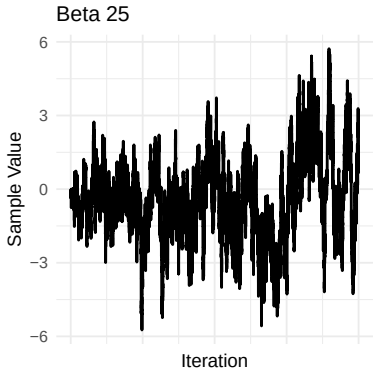

Beta 20
Sample Value
0
−5
−10
−15
Iteration

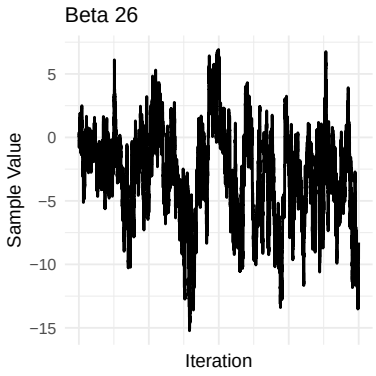

Beta 23
Sample Value
30
20
10
0
−10
Iteration

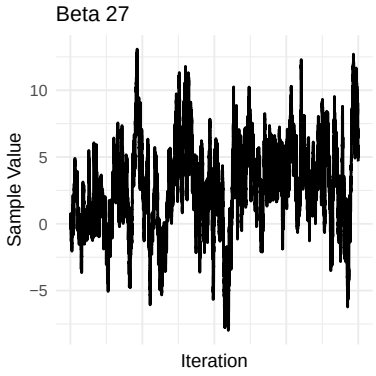

Beta 21
Sample Value
30
20
10
0
−10
Iteration

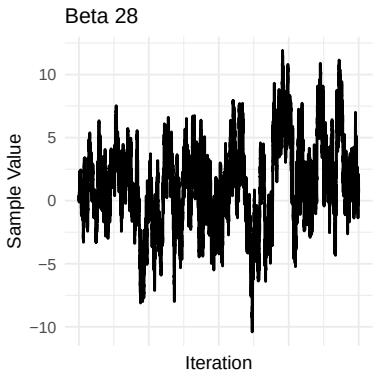

Beta 24
Sample Value
30
20
10
0
−10
Iteration

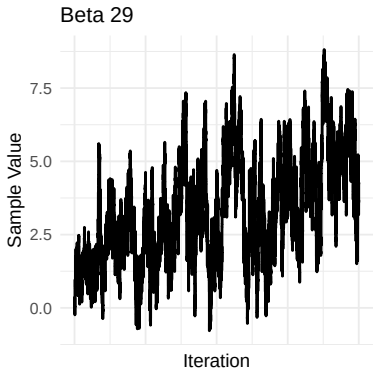

Beta 22
Sample Value
10
5
0
Iteration

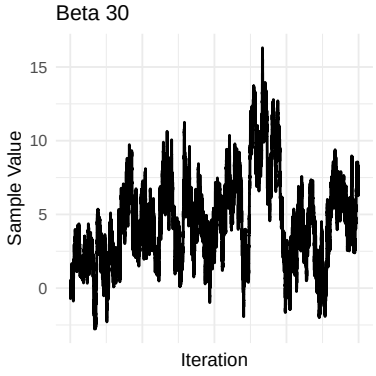

Beta 25
Sample Value
6
3
0
−3
−6
Iteration

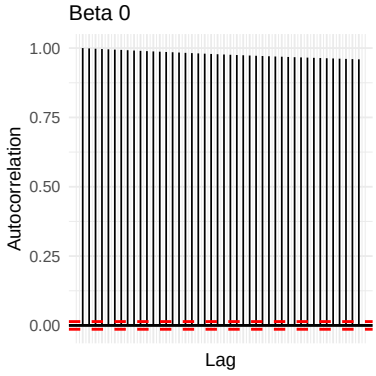

Beta 26
Sample Value
5
0
−5
−10
−15
Iteration

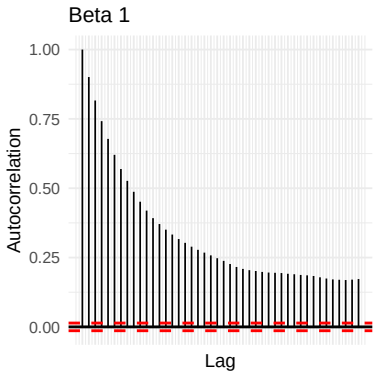

Beta 27
Sample Value
10
5
0
−5
Iteration

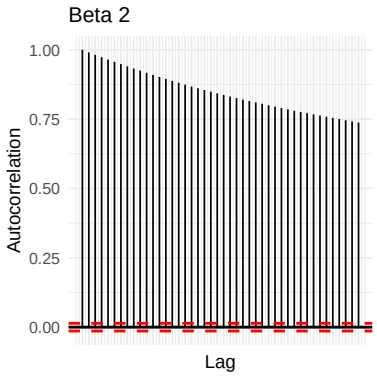

Beta 28
Sample Value
10
5
0
−5
−10
Iteration

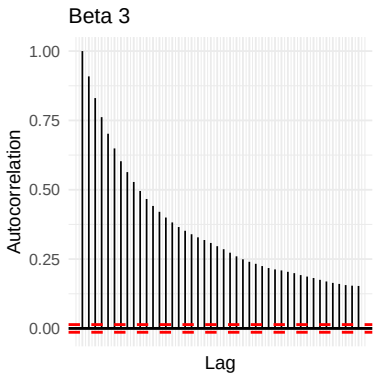

Beta 29
Sample Value
7.5
5.0
2.5
0.0
Iteration

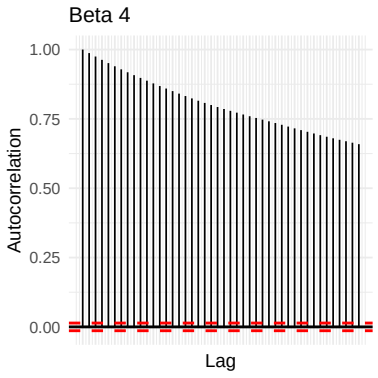

Beta 30
Sample Value
15
10
5
0
Iteration

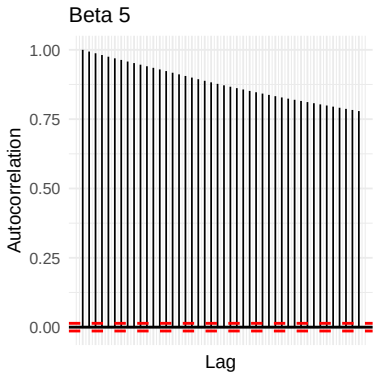

Beta 0
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

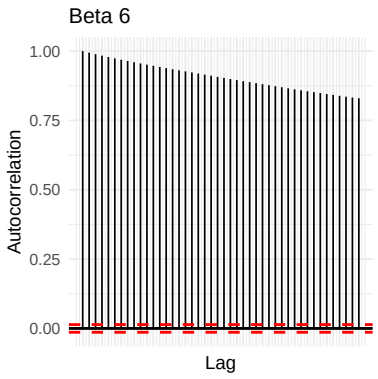

Beta 1
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

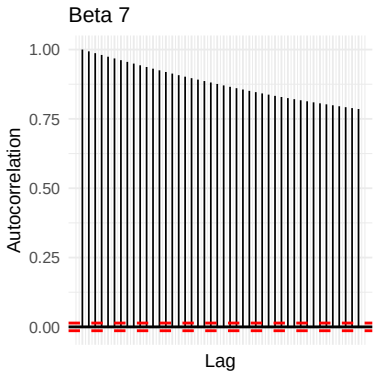

Beta 2
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

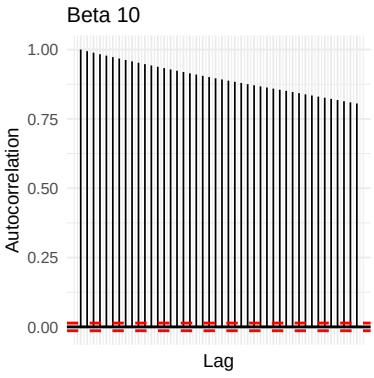

Beta 3
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

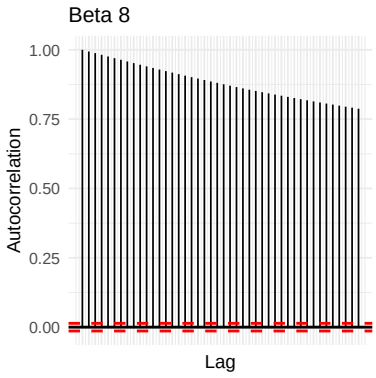

Beta 4
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

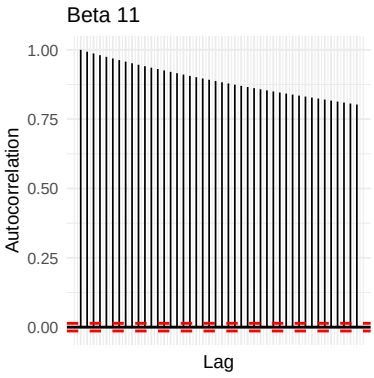

Beta 5
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

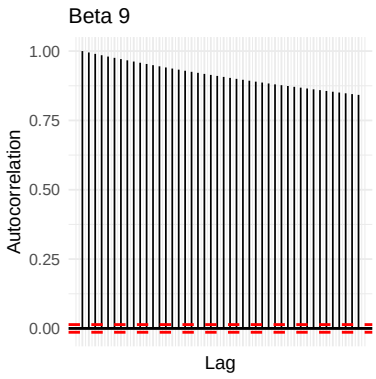

Beta 6
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

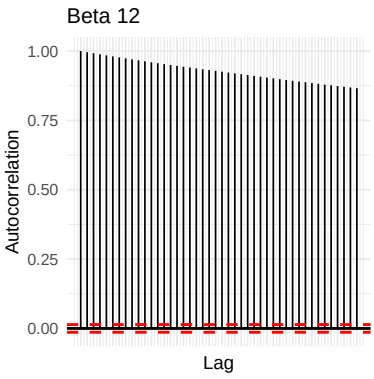

Beta 7
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

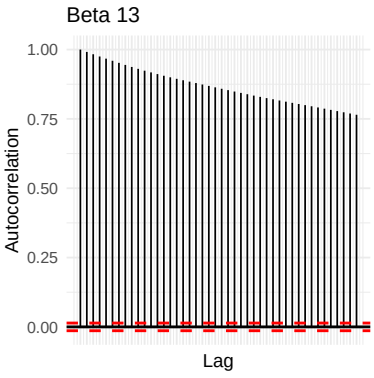

Beta 10
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

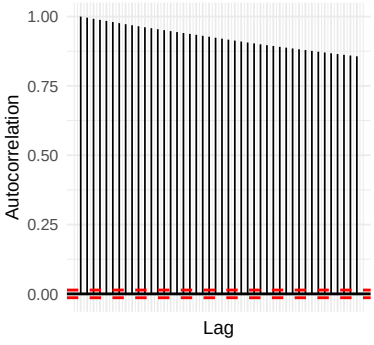

Beta 8
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

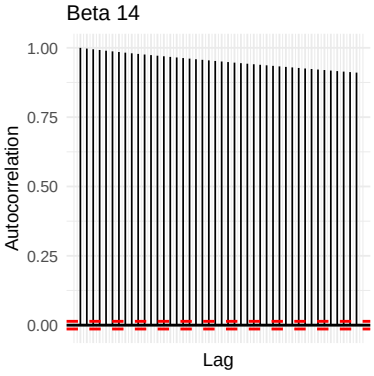

Beta 11
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

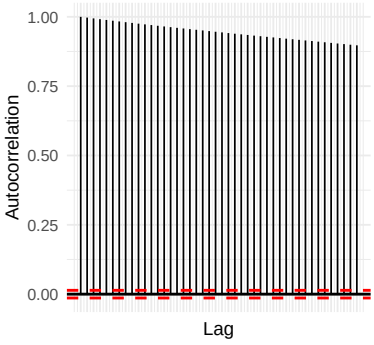

Beta 9
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

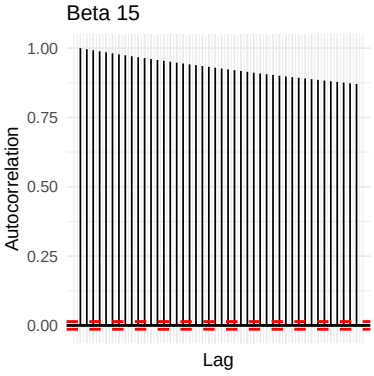

Beta 12
Autocorrelation
1.00
0.75
0.50
0.25
0.00
Lag

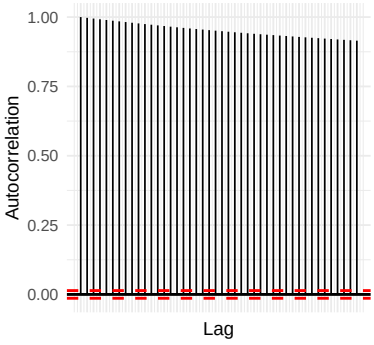

Beta 13
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

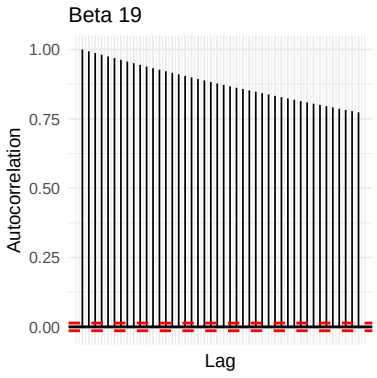

Beta 16
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

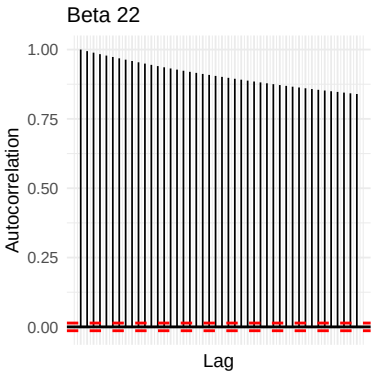

Beta 14
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

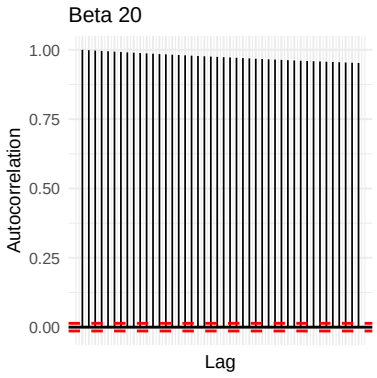

Beta 17
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

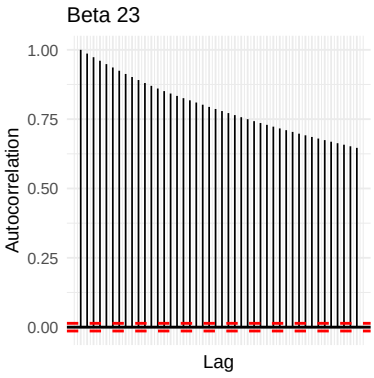

Beta 15
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

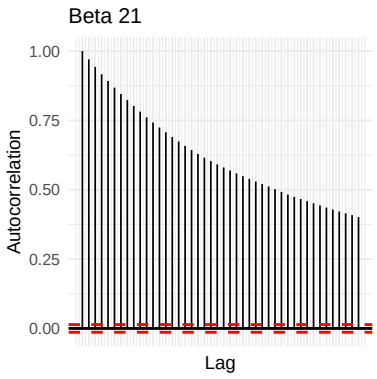

Beta 18
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

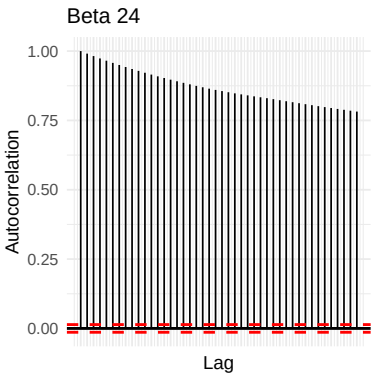

Beta 19
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

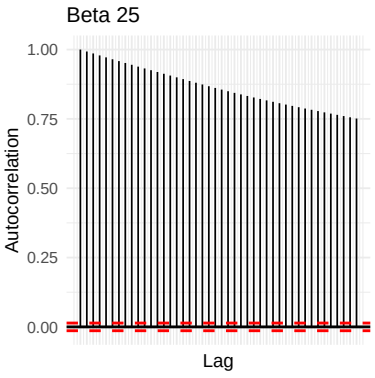

Beta 22
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

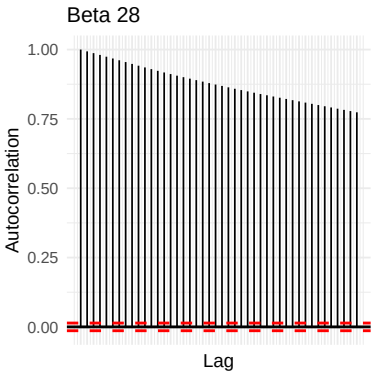

Beta 20
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

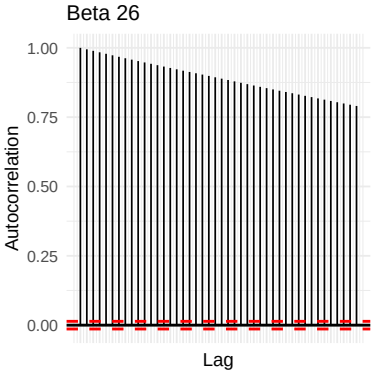

Beta 23
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

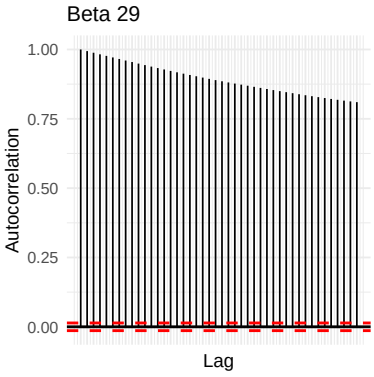

Beta 21
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

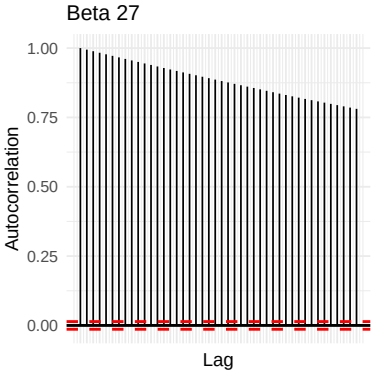

Beta 24
1.00
0.75
0.50
0.25
0.00
Autocorrelation
Lag

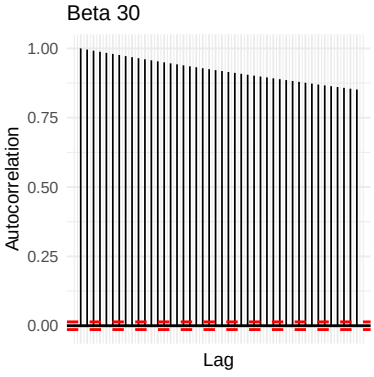












Before interpreting the plots, one needs to adjust the $\boldsymbol{\beta}$ samples by burning-in and thinning them. Upon taking a look at the traceplots, it looks like the chains take about the first 1,000 iterations to start converging, so the first thousand samples are discarded. Thinning may not be a good idea, since it will not make a difference, considering there is extremely high auto-

correlation for the samples of all β_i. Upon testing, the values of the coefficients do not change by a lot either. As for the diagnostics of this MCMC simulation, good mixing is only seen in a few coefficients, and not all.

The posterior means, 95% credible intervals, and significance of the various features are in the table below. Significance is determined by whether zero is included within the significance interval. If zero is not included, this means that there is a clear direction of the effect of the given variable on the malignancy of the mass, and the variable is likely to be significant.

```
##                Feature Posterior.Mean
## 1                            1.7754860
## 2              radius1     -3.4493988
## 3             texture1     -0.2689530
## 4           perimeter1     -2.6283274
## 5                area1     -0.7781131
## 6          smoothness1      1.6904161
## 7         compactness1     -7.6811684
## 8           concavity1      7.6316236
## 9      concave_points1      3.2774668
## 10          symmetry1     -1.0895585
## 11 fractal_dimension1      1.2469487
## 12             radius2      4.6995198
## 13            texture2     -1.8238974
## 14          perimeter2     -2.5221828
## 15               area2      6.9847849
## 16         smoothness2      1.5147851
## 17        compactness2      3.4526549
## 18          concavity2     -5.8965451
## 19     concave_points2      5.2522748
## 20          symmetry2     -1.5805192
## 21 fractal_dimension2     -7.3571682
## 22             radius3      6.6663185
## 23            texture3      5.1753524
## 24          perimeter3      4.5529981
## 25               area3      7.3644317
## 26         smoothness3     -0.4141473
## 27        compactness3     -3.0742975
## 28          concavity3      2.9748868
## 29     concave_points3      1.3182111
## 30          symmetry3      3.3746718
## 31 fractal_dimension3      4.7492725
##       Lower.CI   Upper.CI Significance
## 1   -0.8609714  3.8724485           No
## 2  -18.2375516 11.6857163           No
## 3   -3.4856376  2.9946985           No
## 4  -18.1270565 12.4704710           No
## 5  -15.0994008 14.3428624           No
## 6   -1.4481216  4.7211568           No
## 7  -15.4038902 -0.6953401          Yes
## 8   -0.5706768 17.7769508           No
## 9   -4.9496555 12.3973170           No
## 10  -3.8871346  1.3998645           No
## 11  -3.3940364  5.6095899           No
## 12  -5.3871781 13.5534819           No
## 13  -5.0118690  0.6785438           No
## 14  -9.8685213  6.5431827           No
## 15  -3.3532157 17.2591935           No
## 16  -0.9753628  4.5373823           No
## 17  -1.4584953  9.1899987           No
## 18 -13.2243149 -0.4488643          Yes
## 19   0.4025852 11.2145611          Yes
## 20  -4.2082020  0.7855442           No
## 21 -15.6389864 -1.2344497          Yes
## 22  -6.9512896 20.8152030           No
## 23   0.7802777 10.1108130          Yes
## 24  -9.7288741 19.2618953           No
## 25  -7.6918795 23.9887362           No
## 26  -3.7978384  3.3578309           No
## 27 -10.7738418  4.0033313           No
## 28  -4.0465014  9.5626004           No
## 29  -5.1513972  7.9451506           No
## 30   0.3100122  6.9465286          Yes
## 31  -0.4953099 11.7846683           No
```

Model selection within the Bayesian framework is still required, however. Variables selected by "significance" as done in linear regression, do not show good mixing in the traceplots.

Variable Selection

In the previous section, all 30 regression coefficients were estimated simultaneously. However, not all features may be equally important for predicting malignancy. Including irrelevant features can lead to overfitting and reduced interpretability. Bayesian variable selection provides a principled framework for identifying which features are most informative while accounting for uncertainty in the selection process itself.

Model Formulation

Rather than estimating a single model, all possible subsets of features are considered. The data determines which features should be included. Binary indicator variables $\mathbf{z} = (z_1, \ldots, z_p)$ are introduced where $z_j \in \{0, 1\}$ indicates whether feature j is included in the model:

$$\text{logit}(\theta_i) = b_0 + \sum_{j=1}^{30} z_j b_j x_{i,j}$$

If $z_j = 1$, feature j is included in the model, contributing $b_j x_{i,j}$ to the linear predictor. If $z_j = 0$, feature j is excluded from the model, contributing nothing (effectively $\beta_j = 0$).

The intercept b_0 is always included (no indicator variable). The joint parameter vector is now $\boldsymbol{\theta} = (\mathbf{z}, \mathbf{b})$, where $\mathbf{b} = (b_0, b_1, \ldots, b_{30})$ are the regression coefficients. Importantly, $\mathbf{b}$ represents the coefficients if the corresponding features were included–the actual effect is $j\beta_j = z_j \cdot b_j$.

Prior Specifications

One must specify prior distributions for both the coefficients $\mathbf{b}$ and the indicators $\mathbf{z}$. For the coefficients, one must use the same weakly informative multivariate normal prior as before:

$$\mathbf{b} \sim \mathcal{N}(\mathbf{0}, \sigma_b^2 \mathbf{I}_{31}), \quad \sigma_b^2 = 100$$

For the indicators, equal prior probability is assigned to inclusion or exclusion:

$$p(z_j = 1) = 0.5, \quad \text{independently for } j = 1, \ldots, 30$$

This uniform prior on each z_j implies equal prior probability across all $2^{30} \approx 1.07$ billion possible models. No *a priori* assumption is made about which features are important, allowing the data to guide the selection.

Metropolis-Hastings Algorithm

Since there now are two types of parameters–discrete indicators $\mathbf{z}$ and continuous coefficients $\mathbf{b}$-—they can be updated separately within each MCMC iteration using a two-step Metropolis-Hastings procedure.

First, $\mathbf{z}$ is updated given current $\mathbf{b}$. At iteration k, a new indicator vector $\mathbf{z}^*$ is proposed by randomly flipping each indicator with probability $p_{\text{flip}} = 0.2$:

$$z_j^* = \begin{cases} z_j^{(k)} & \text{with } p_{\text{flip}} = 0.8 \text{ (no flip)} \\ 1 - z_j^{(k)} & \text{with } p_{\text{flip}} = 0.2 \text{ (flip)} \end{cases}$$

This proposal is symmetric: the probability of proposing $\mathbf{z}^*$ from $\mathbf{z}^{(k)}$ equals the probability of proposing $\mathbf{z}^{(k)}$ from $\mathbf{z}^*$. The acceptance ratio simplifies to:

$$r_z = \frac{p(\mathbf{y} \mid \mathbf{X}, \mathbf{z}^*, \mathbf{b}^{(k)})}{p(\mathbf{y} \mid \mathbf{X}, \mathbf{z}^{(k)}, \mathbf{b}^{(k)})}$$

Next, $\mathbf{b}$ is updated given current $\mathbf{z}$. After updating $\mathbf{z}$, the coefficients $\mathbf{z}$ are updated using the same symmetric normal proposal as in the full model:

$$\mathbf{b}^* \sim N(\mathbf{b}^{(k)}, \boldsymbol{\Sigma}_{\text{prop}})$$

where $\boldsymbol{\Sigma}_{\text{prop}} = 0.5 \cdot (\mathbf{X}^T\mathbf{X})^{-1}$. The acceptance ratio is:

$$r_b = \frac{p(\mathbf{y}|\mathbf{X}, \mathbf{z}^{(k+1)}, \mathbf{b}^*)p(\mathbf{b}^*)}{p(\mathbf{y}|\mathbf{X}, \mathbf{z}^{(k+1)}, \mathbf{b}^{(k)})p(\mathbf{b}^{(k)})}$$

Note that this ratio is evaluated using the updated indicator vector $\mathbf{z}^{(k+1)}$ from the first step. This ensures that the coefficient updates are conditioned on the current model structure.

Computing the Effective Coefficient Vector

For likelihood evaluation, the effective coefficient vector is computed:

$$\boldsymbol{\beta}_{\text{effective}} = \begin{bmatrix} b_0 \\ z_1 b_1 \\ z_2 b_2 \\ \vdots \\ z_{30} b_{30} \end{bmatrix}$$

When $z_j = 0$, the corresponding coefficient is zeroed out, effectively excluding that feature from the model.

After running the MCMC sampler, samples $\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \ldots, \mathbf{z}^{(S)}$ are obtained of the indicator variables. The posterior inclusion probability for feature j is simply the proportion of iterations where $z_j = 1$:

$$P(z_j = 1 \mid \mathbf{y}, \mathbf{X}) \approx \frac{1}{S - S_{\text{burnin}}} \sum_{s=S_{\text{burnin}}+1}^{S} z_j^{(s)}$$

This probability quantifies the posterior belief that feature j should be included in the model. Features with high inclusion probabilities (e.g., > 0.5) are strongly supported by the data, while features with low probabilities can be considered unimportant. All features and their inclusion probabilities are displayed below:

```
##                Feature
## 2             texture1
## 5          smoothness1
## 6         compactness1
## 17          concavity2
## 21             radius3
## 27          concavity3
## 3           perimeter1
## 4                area1
## 15         smoothness2
## 20 fractal_dimension2
## 11             radius2
## 29          symmetry3
## 18    concave_points2
## 28    concave_points3
## 1              radius1
## 14               area2
## 13          perimeter2
## 7           concavity1
## 23          perimeter3
## 9            symmetry1
## 8     concave_points1
## 16       compactness2
## 25         smoothness3
## 19           symmetry2
## 12            texture2
## 30 fractal_dimension3
## 10 fractal_dimension1
## 22           texture3
## 24              area3
## 26       compactness3
##    Inclusion_Probability
## 2             1.00000000
## 5             1.00000000
## 6             1.00000000
## 17            1.00000000
## 21            1.00000000
## 27            1.00000000
## 3             0.96957895
## 4             0.86126316
## 15            0.72673684
## 20            0.70757895
## 11            0.67263158
## 29            0.61726316
## 18            0.58226316
## 28            0.54289474
## 1             0.53542105
## 14            0.52884211
## 13            0.49836842
## 7             0.48378947
## 23            0.26952632
## 9             0.24700000
## 8             0.21700000
## 16            0.19236842
## 25            0.12589474
## 19            0.11815789
## 12            0.08057895
## 30            0.06752632
## 10            0.00000000
## 22            0.00000000
## 24            0.00000000
## 26            0.00000000
```

The important features are as follows:

```
##  [1] "texture1"
##  [2] "smoothness1"
##  [3] "compactness1"
##  [4] "concavity2"
##  [5] "radius3"
##  [6] "concavity3"
##  [7] "radius2"
##  [8] "concave_points3"
##  [9] "radius1"
## [10] "perimeter2"
## [11] "concavity1"
```

```
## [12] "symmetry1"
## [13] "concave_points1"
## [14] "fractal_dimension1"
## [15] "texture3"
## [16] "area3"
```

Posterior Estimates for the Selected Model

To obtain coefficient estimates that account for variable selection uncertainty, the effective coefficients for each MCMC sample are computed by multiplying the coefficient values by their corresponding indicators:

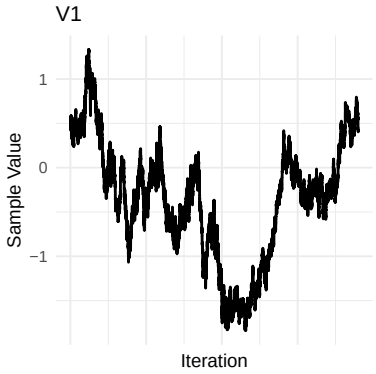
$$\beta_j^{(s)} = z_j^{(s)} \cdot b_j^{(s)}$$

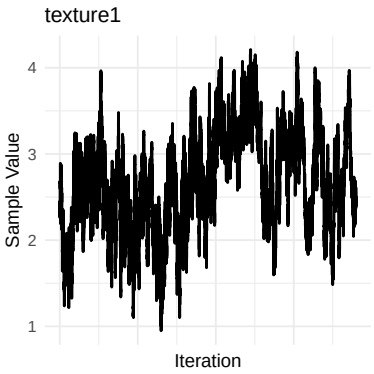
The posterior mean and credible intervals for these effective coefficients incorporate both parameter uncertainty (variation in b_j) and model uncertainty (whether $z_j = 0$ or 1).

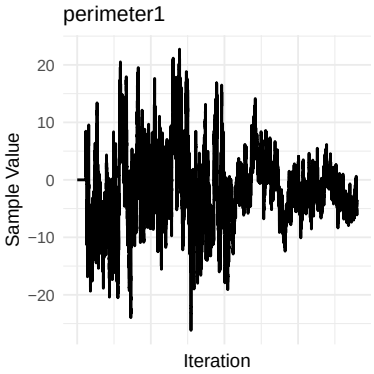
```
##                Feature Inclusion_Prob
## 1                            1.00000000
## 2              radius1     0.53542105
## 3             texture1     1.00000000
## 4           perimeter1     0.96957895
## 5                area1     0.86126316
## 6          smoothness1     1.00000000
## 7         compactness1     1.00000000
## 8           concavity1     0.48378947
## 9      concave_points1     0.21700000
## 10          symmetry1     0.24700000
## 11 fractal_dimension1     0.00000000
## 12             radius2     0.67263158
## 13            texture2     0.08057895
## 14          perimeter2     0.49836842
## 15               area2     0.52884211
## 16         smoothness2     0.72673684
## 17        compactness2     0.19236842
## 18          concavity2     1.00000000
## 19     concave_points2     0.58226316
## 20           symmetry2     0.11815789
## 21 fractal_dimension2     0.70757895
## 22             radius3     1.00000000
## 23            texture3     0.00000000
## 24          perimeter3     0.26952632
## 25               area3     0.00000000
## 26         smoothness3     0.12589474
## 27        compactness3     0.00000000
## 28          concavity3     1.00000000
## 29     concave_points3     0.54289474
## 30           symmetry3     0.61726316
## 31 fractal_dimension3     0.06752632
##      Post_Mean    CI_Lower
## 1  -0.36995562  -1.6628080
## 2  -1.65944780 -14.4222991
## 3   2.67392396   1.5502169
## 4  -1.58125765 -13.6273034
## 5  -0.75663039 -10.5585875
## 6   1.88475774   0.1327450
## 7  -3.40528590  -6.6355650
## 8   1.50767820  -0.0317417
## 9   0.12862344  -3.0097286
## 10 -0.29223032  -1.9698534
## 11  0.00000000   0.0000000
## 12  1.93842218  -1.8443072
## 13 -0.02025018  -0.4594021
## 14 -0.40212524  -4.8276812
## 15  2.49403402   0.0000000
## 16  0.94360290   0.0000000
## 17 -0.03647296  -1.2877484
## 18 -3.46134538  -6.2190711
## 19  1.65461316  -0.0889657
## 20  0.11877276   0.0000000
## 21 -1.49637658  -3.4927494
## 22 12.17590190   1.1319955
## 23  0.00000000   0.0000000
## 24  0.13992429  -7.4664087
## 25  0.00000000   0.0000000
## 26 -0.08956376  -1.4202561
## 27  0.00000000   0.0000000
## 28  5.44035795   1.0965329
## 29  2.35460982   0.0000000
## 30  0.80671130   0.0000000
## 31  0.03674546   0.0000000
##         CI_Upper
## 1   0.8728454937
## 2   8.0885531493
## 3   3.8189359532
## 4  12.9706383321
## 5  10.4023848826
## 6   3.4902249693
## 7   0.0566711468
## 8   6.4329458271
## 9   3.8544998518
## 10  0.0000000000
## 11  0.0000000000
## 12  7.4869133881
## 13  0.0000000000
```

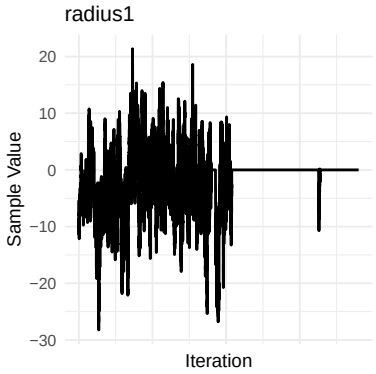
```
## 14  3.4268658484
## 15  7.1156958488
## 16  2.3140542685
## 17  0.8125925790
## 18 -1.6477107371
## 19  5.2973520748
## 20  1.2408535966
## 21  0.3110486179
## 22 20.2764899183
## 23  0.0000000000
## 24  8.7106883638
## 25  0.0000000000
## 26  0.0001140985
## 27  0.0000000000
## 28  8.7035620731
## 29  7.6234487248
## 30  2.5257268242
## 31  0.8500493773
```

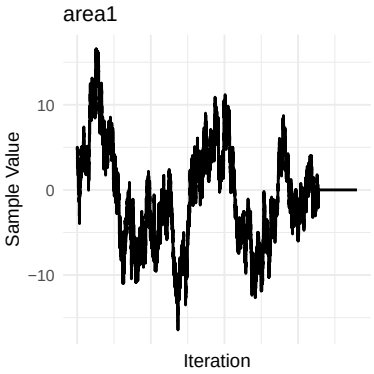
The traceplots for the coefficients are as follows:



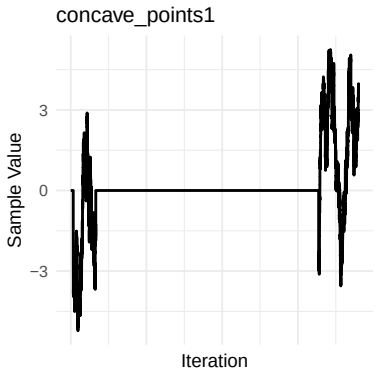


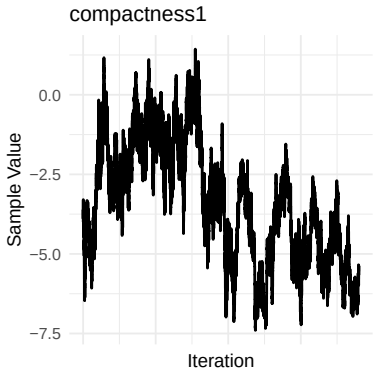


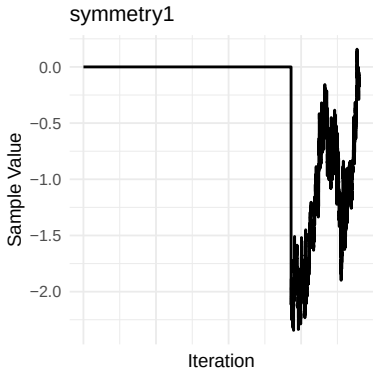


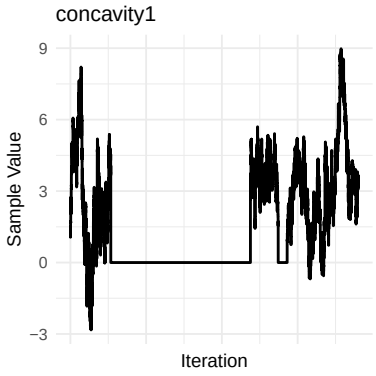


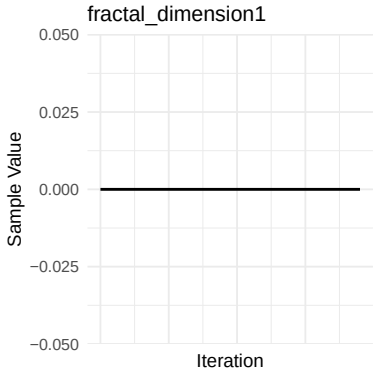

smoothness1
Sample Value
Iteration
4
3
2
1
0

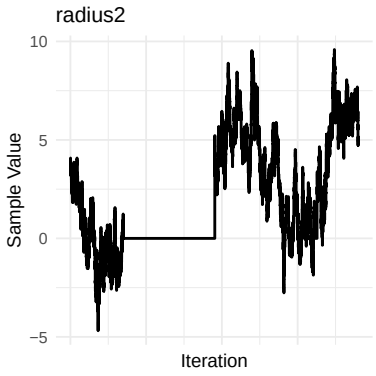

concave_points1
Sample Value
Iteration
3
0
−3

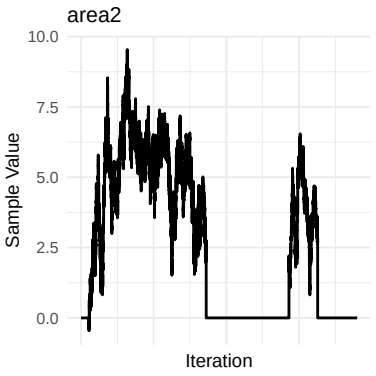

compactness1
Sample Value
Iteration
0.0
−2.5
−5.0
−7.5

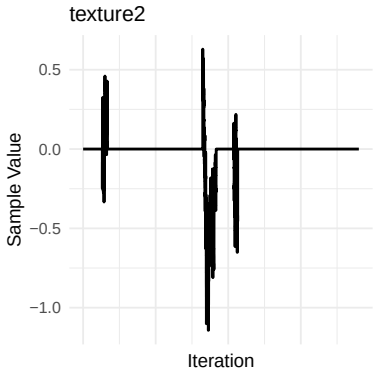

symmetry1
Sample Value
Iteration
0.0
−0.5
−1.0
−1.5
−2.0


concavity1
Sample Value
Iteration
9
6
3
0
−3

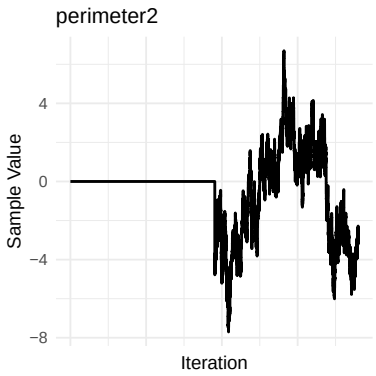

fractal_dimension1
Sample Value
Iteration
0.050
0.025
0.000
−0.025
−0.050

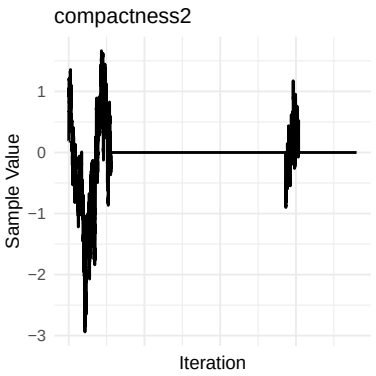

radius2
Sample Value
Iteration
10
5
0
−5

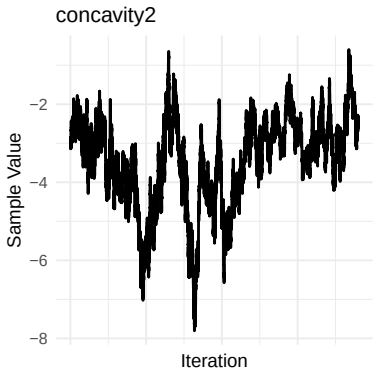

area2
Sample Value
Iteration
10.0
7.5
5.0
2.5
0.0

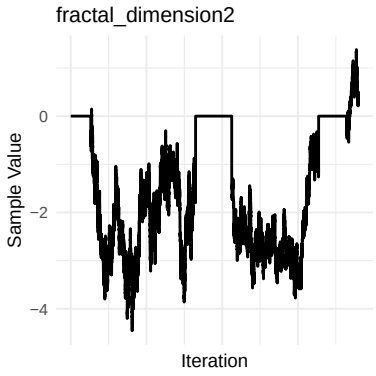

texture2
Sample Value
Iteration
0.5
0.0
−0.5
−1.0

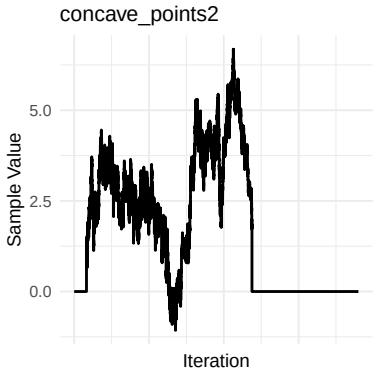

smoothness2
Sample Value
Iteration
2
1
0

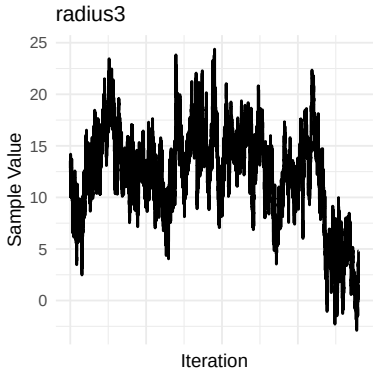

perimeter2
Sample Value
Iteration
4
0
−4
−8

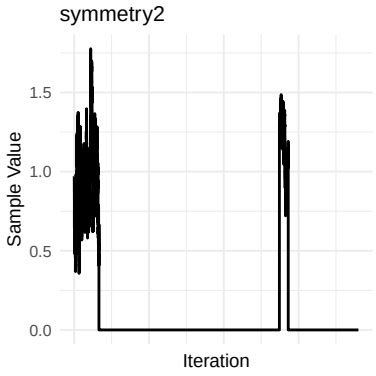

compactness2
Sample Value
Iteration
1
0
−1
−2
−3

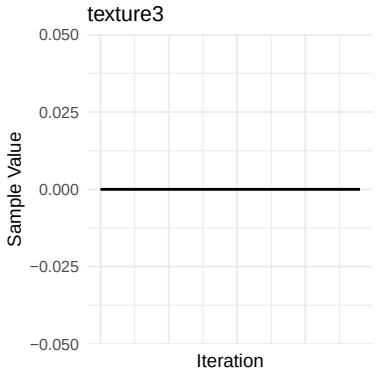

concavity2
Sample Value
−2
−4
−6
−8
Iteration

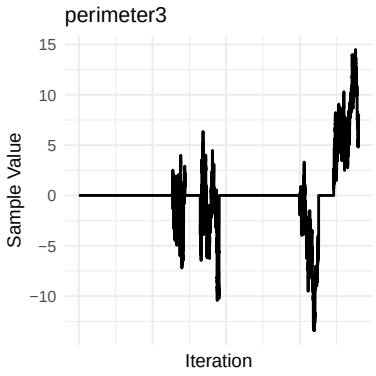

fractal_dimension2
Sample Value
0
−2
−4
Iteration

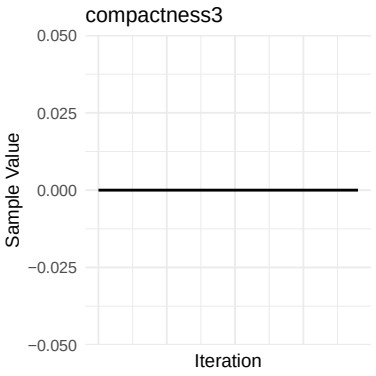

concave_points2
Sample Value
5.0
2.5
0.0
Iteration

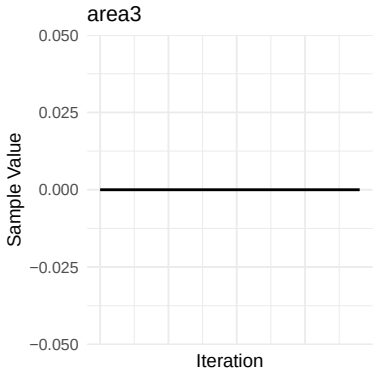

radius3
Sample Value
25
20
15
10
5
0
Iteration

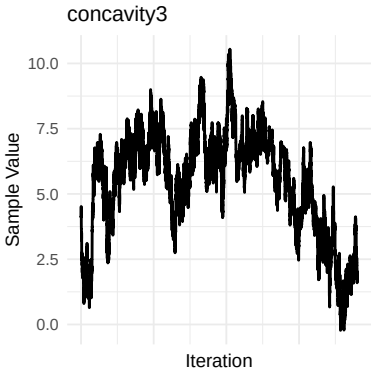

symmetry2
Sample Value
1.5
1.0
0.5
0.0
Iteration

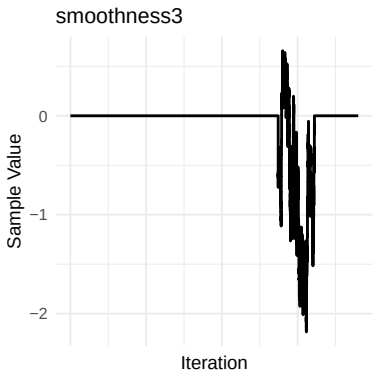

texture3
Sample Value
0.050
0.025
0.000
−0.025
−0.050
Iteration

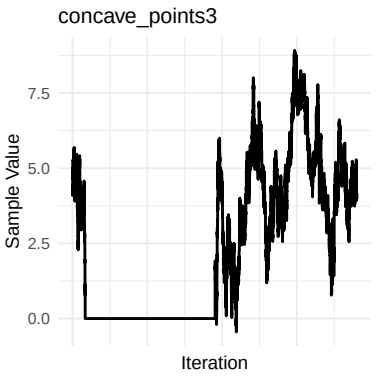

perimeter3
Sample Value
15
10
5
0
−5
−10
Iteration

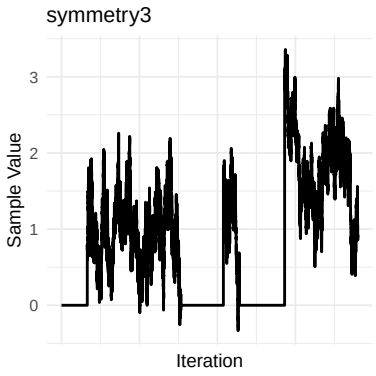

compactness3
Sample Value
0.050
0.025
0.000
−0.025
−0.050
Iteration

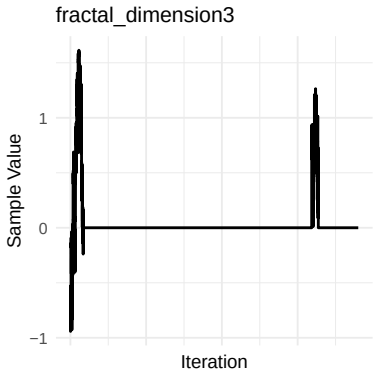

area3
Sample Value
0.050
0.025
0.000
−0.025
−0.050
Iteration


concavity3
Sample Value
10.0
7.5
5.0
2.5
0.0
Iteration


smoothness3
Sample Value
0
−1
−2
Iteration


concave_points3
Sample Value
7.5
5.0
2.5
0.0
Iteration





Variables with posterior inclusion probability set to zero, like `fractal_dimension1`, have no exploration in their traceplot at all.

The effective posterior means will be exactly zero for features with $z_j = 0$ in all samples, and will be shrunken toward zero for features that are only sometimes included. This provides automatic regularization while maintaining interpretability.

Unlike classical methods that select a single "best" model, Bayesian variable selection explores the entire model space and assigns posterior probabilities to each model. The specific combinations of features the MCMC sampler visited most frequently can be identified.

Results and Conclusion

The ultimate goal of this analysis is not merely to build a Bayesian model, but to understand which cellular characteristics are most indicative of malignancy. This understanding can inform clinical decision-making and suggest biological mechanisms underlying breast cancer.

Coefficient Interpretation

For features with high posterior inclusion probabilities, one interprets the coefficients on the odds scale. Recall that in logistic regression, $\exp(\beta_j)$ represents the multiplicative change in odds of malignancy associated with a one-unit increase in feature j.

Since features were standardized, a "one-unit increase" corresponds to a *one standard deviation increase* in the original measurement. This makes coefficients directly comparable across features measured on different scales.

All important features are defined as features with posterior inclusion probability greater than 50%, or features that were included in more than half of the possible combinations.

```
##
## radius1:
##   Coefficient: -1.6594
##   Odds ratio: 0.1902
##   95% CI for OR: (0, 3256.972)
##   A 1-SD increase in radius1
## multiplies the odds of
## malignancy by 0.19
##
## texture1:
##   Coefficient: 2.6739
##   Odds ratio: 14.4967
##   95% CI for OR: (4.7125, 45.5557)
##   A 1-SD increase in texture1
## multiplies the odds of
## malignancy by 14.5
##
## perimeter1:
##   Coefficient: -1.5813
##   Odds ratio: 0.2057
##   95% CI for OR: (0, 429612.2)
##   A 1-SD increase in perimeter1
## multiplies the odds of
## malignancy by 0.21
##
```

```
## area1:
##   Coefficient: -0.7566
##   Odds ratio: 0.4692
##   95% CI for OR: (0, 32938.09)
##   A 1-SD increase in area1
## multiplies the odds of
## malignancy by 0.47
##
## smoothness1:
##   Coefficient: 1.8848
##   Odds ratio: 6.5848
##   95% CI for OR: (1.142, 32.7933)
##   A 1-SD increase in smoothness1
## multiplies the odds of
## malignancy by 6.58
##
## compactness1:
##   Coefficient: -3.4053
##   Odds ratio: 0.0332
##   95% CI for OR: (0.0013, 1.0583)
##   A 1-SD increase in compactness1
## multiplies the odds of
## malignancy by 0.03
##
## radius2:
##   Coefficient: 1.9384
##   Odds ratio: 6.9478
##   95% CI for OR: (0.1581, 1784.535)
##   A 1-SD increase in radius2
## multiplies the odds of
## malignancy by 6.95
##
## area2:
##   Coefficient: 2.494
##   Odds ratio: 12.11
##   95% CI for OR: (1, 1231.14)
##   A 1-SD increase in area2
## multiplies the odds of
## malignancy by 12.11
##
## smoothness2:
##   Coefficient: 0.9436
##   Odds ratio: 2.5692
##   95% CI for OR: (1, 10.1154)
##   A 1-SD increase in smoothness2
## multiplies the odds of
## malignancy by 2.57
##
## concavity2:
##   Coefficient: -3.4613
##   Odds ratio: 0.0314
##   95% CI for OR: (0.002, 0.1925)
##   A 1-SD increase in concavity2
## multiplies the odds of
## malignancy by 0.03
##
## concave_points2:
##   Coefficient: 1.6546
##   Odds ratio: 5.2311
##   95% CI for OR: (0.9149, 199.807)
##   A 1-SD increase in concave_points2
## multiplies the odds of
## malignancy by 5.23
##
## fractal_dimension2:
##   Coefficient: -1.4964
##   Odds ratio: 0.2239
##   95% CI for OR: (0.0304, 1.3649)
##   A 1-SD increase in fractal_dimension2
## multiplies the odds of
## malignancy by 0.22
##
## radius3:
##   Coefficient: 12.1759
##   Odds ratio: 194056
##   95% CI for OR: (3.1018, 639687234)
##   A 1-SD increase in radius3
## multiplies the odds of
## malignancy by 194056
##
## concavity3:
##   Coefficient: 5.4404
##   Odds ratio: 230.5247
##   95% CI for OR: (2.9938, 6024.333)
##   A 1-SD increase in concavity3
## multiplies the odds of
## malignancy by 230.52
##
## concave_points3:
##   Coefficient: 2.3546
##   Odds ratio: 10.534
##   95% CI for OR: (1, 2045.605)
##   A 1-SD increase in concave_points3
## multiplies the odds of
## malignancy by 10.53
##
## symmetry3:
##   Coefficient: 0.8067
##   Odds ratio: 2.2405
##   95% CI for OR: (1, 12.5)
```

```
##   A 1-SD increase in symmetry3
## multiplies the odds of
## malignancy by 2.24
```

Example Interpretation: Consider `symmetry3`. If there a one standard deviation increase in the worst symmetry measurement, the odds of malignancy multiplies by approximately $e^{0.8067} \approx 2.24$, holding all other features constant.

Ranking the Most Important Features by Inclusion Probability

While it is known what the important features are, which of these are the most important?

The following features, as per the above computations, have a posterior inclusion probability of 1–they are included in every single model combination.

- The mean measurement of texture
- The mean measurement of smoothness
- The mean measurement of compactness
- The standard error of concavity

- The worst measurement of radius = The worst measurement of concavity

References

Wolberg, W., Mangasarian, O., and Street, W. N. (1993). "Breast Cancer Wisconsin (Diagnostic)." UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5DW2B. 1