Intro to Bayesian Methods

BIOS 6618

CU Anschutz

Introduction

Frequentist versus Bayesian

Almost all the methods covered this semester are rooted in the *frequentist* approach to statistics. This is based on long-run probabilities of how probable a dataset is given a null hypothesis *(for a refresher review the lectures on p-values and the NHST framework)*.

An alternative approach is known as *Bayesian* statistics, which focuses on the probability of a hypothesis given a certain dataset. Bayesian approaches are able to incorporate prior information to our analyses, make different assumptions about our modeling framework, and have different interpretations.

This lecture will give a brief, high-level introduction to Bayesian methods.

Bayesian Overview

Let $\mathbf{x}$ be our observed data and $\boldsymbol{\theta}$ be the parameter(s) we are interested in estimating (e.g., the sample mean ($\bar{X}$), coefficients in a regression model (β), etc.).

According to Bayes' theorem we have:

$$p(\boldsymbol{\theta}|\mathbf{x}) = \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{x})}$$

- $p(\boldsymbol{\theta}|\mathbf{x})$ is the *posterior*
- $p(\mathbf{x}|\boldsymbol{\theta})$ is the *likelihood*
- $p(\boldsymbol{\theta})$ is the *prior*
- $p(\mathbf{x})$ is the *normalizing constant*

Likelihood: $p(\mathbf{x}|\boldsymbol{\theta})$

The likelihood is the joint density function of our observed data to be analyzed:

$$L(\boldsymbol{\theta}) = L(\boldsymbol{\theta}|\mathbf{x}) = p(\mathbf{x}|\boldsymbol{\theta})$$

In our likelihood function, we assume our data is fixed and consider $\boldsymbol{\theta}$ over the whole range of possible parameter values.

For our class, we only assume cases with independent and identically distributed (iid) data, where we have a product of the PDFs at each observation $\mathbf{x}_i$. For a linear regression this is:

$$p(\mathbf{x}|\boldsymbol{\theta}) = \prod_{i=1}^{n} f(\mathbf{x}_i;\ \boldsymbol{\theta}) = \prod_{i=1}^{n} N(\beta_0 + \beta_1 X_{1i} + \cdots + \beta_k X_{ki}, \sigma^2_{Y|X})$$

Note, we use the likelihood in both frequentist and Bayesian statistics.

Prior: $p(\theta)$

The prior is our assumed distribution on parameters which quantifies a "belief" in the values of the parameters prior to observing study data. Priors are a unique aspect of Bayesian analyses.

The specification of priors in Bayesian models is one of the challenges since we often may be concerned about placing priors that may be seen as overly informative (e.g., overpowering the likelihood even if the data does not agree with our prior).

While any distribution may be specified, a common option in linear regression for our beta coefficients is $\beta_k \sim N(a, b)$ (where a and b are selected based on context).

In practice, it is recommended to evaluate the results across a range of priors to see if the posterior changes with different specifications.

Normalizing Constant: $p(\mathbf{x})$

The normalizing constant is estimated by integrating out the parameter(s):

$$p(\mathbf{x}) = \int p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}$$

Its role in Bayes' theorem is to ensure any probability function reduces to a probability density function with a total probability of 1.

Posterior: $p(\boldsymbol{\theta}|\mathbf{x})$

The posterior is what we ultimately are interested in using for statistical inference.

It expresses the uncertainty in the parameter(s) after accounting for both our observed data and incorporating our priors.

In some simpler contexts there are conjugate priors that result in closed form posterior distributions (e.g., a binomial likelihood with a beta prior on p; a normal likelihood with known variance and a normal prior on μ).

For most contexts, we likely need to use numerical integration to approximate the posterior distribution. This is achieved through *Markov chain Monte Carlo* (MCMC) methods.

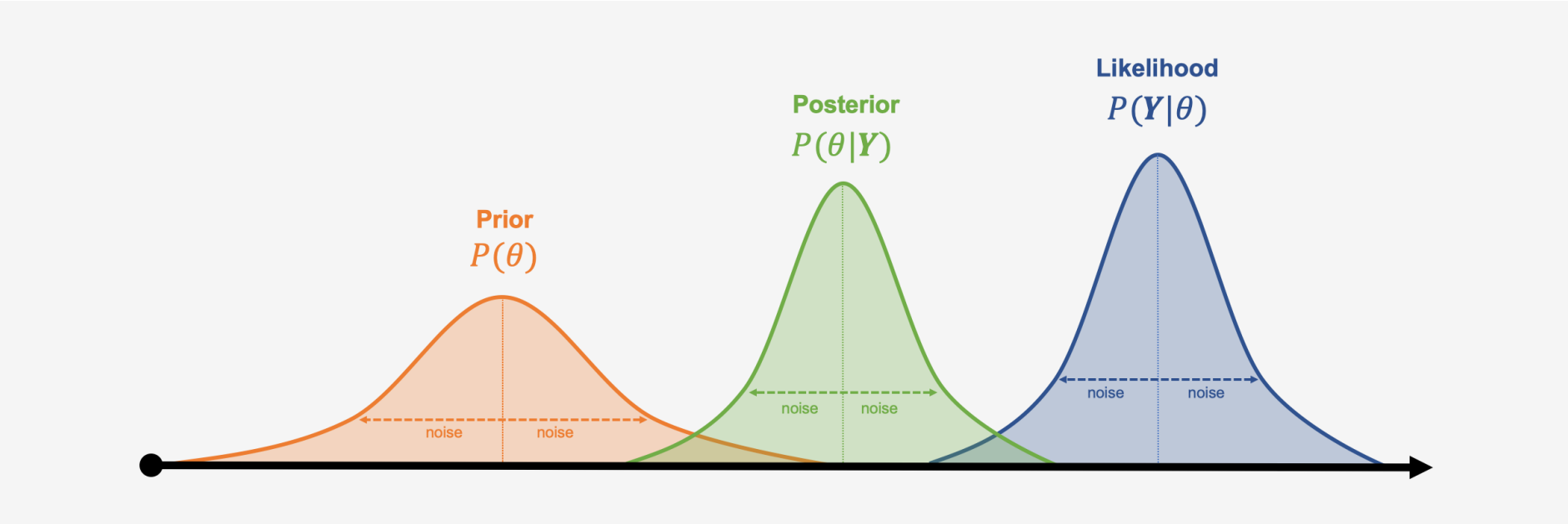
Bayesian Posterior and Proportional To

According to Bayes' theorem we have:

$$p(\boldsymbol{\theta}|\mathbf{x}) = \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{x})} \propto p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})$$

- $p(\boldsymbol{\theta}|\mathbf{x})$ is the *posterior*
- $p(\mathbf{x}|\boldsymbol{\theta})$ is the *likelihood*
- $p(\boldsymbol{\theta})$ is the *prior*
- $p(\mathbf{x})$ is the *normalizing constant*, but this is usually ignored as we describe the relationship of the posterior being **proportional to** ($\propto$) the likelihood $\times$ prior

Bayes Example Figure of Prior, Likelihood, Posterior



Source: https://hudsonthames.org/wp-content/uploads/2020/10/bayesian.png

MCMC and Diagnostics

MCMC

As mentioned previously, the posterior is often not known or available in closed form.

Instead, we can use Markov chain Monte Carlo (MCMC) methods to simulate samples from the posterior distribution.

There are numerous different MCMC algorithms that have been proposed included:

- Gibbs Sampler
- Metropolis Hasting
- No-U-Turn Sampler (NUTS)
- Reversible-Jump
- See over 40 different MCMC samplers at https://m-clark.github.io/docs/ld_mcmc/index_onepage.html

In general, our MCMC samplers can be thought of as recipes to explore the sample space as we approximate our posterior distribution.

MCMC in R/SAS/Stata

While we can often custom code our own MCMC in R, we can also leverage existing software to implement our models.

Two older Bayesian languages with R interfaces are BUGS (e.g., `R2OpenBUGS`) and JAGS (e.g., `rjags`) are based on Gibbs samplers. However, these often involve creating your own models and syntax.

A newer approach based on a Hamiltonian Markov Chain is the Stan language. There are two popular sets of packages (`rstan`/`rstanarm` and `brms`) that both help to implement Stan models using standard `glm` syntax. You can also directly use Stan syntax to define your models.

It is worth noting SAS (e.g., PROC MCMC) and Stata have their own implementations.

MCMC Output and Terminology

Output from our MCMC algorithms will usually be in the form of a rectangular dataset with a column for each *chain* and a row for each *iteration*.

Chains are separate instances of the MCMC algorithm, often with different initial values, that help to provide more exploration of the sample space.

Each chain has a *burn-in period* where the first $N_{\text{burn-in}}$ of each chain are discarded to provide time for convergence.

Each software/package will have its own defaults for number of chains, iterations, and burn-in period.

MCMC Diagnostics

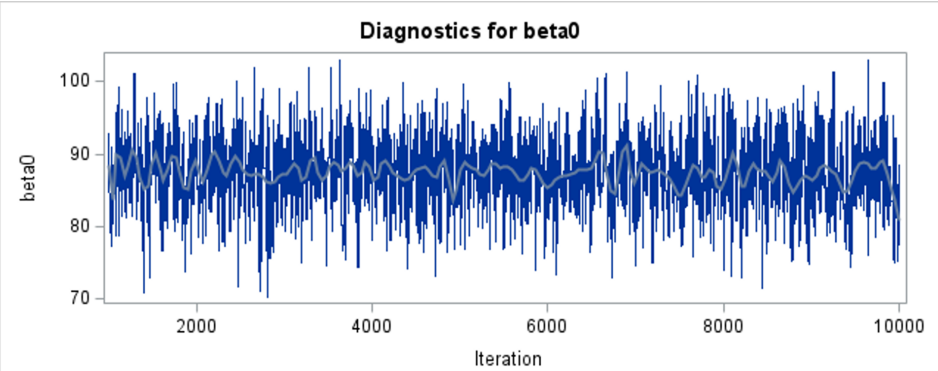
Since the MCMC is an approximation to estimate the posterior distribution, we will want to evaluate how well we think our model/sampler has done and if changes may be needed (e.g., longer burn-in, different models, etc.).

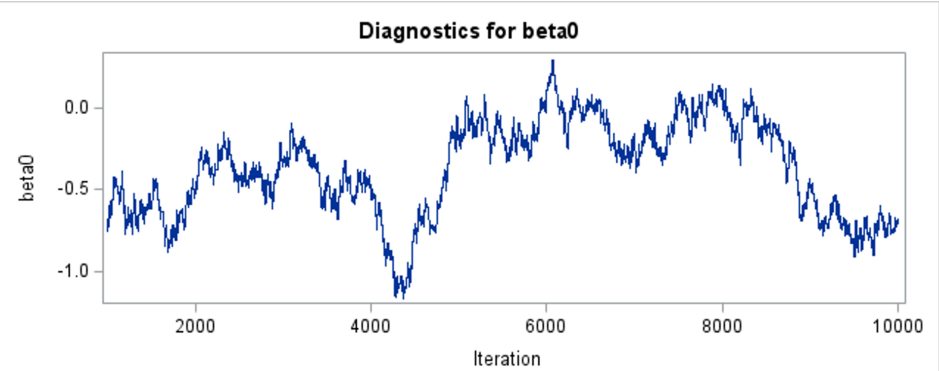
While there are many proposed approaches, we will focus on:

- Trace plots
- Autocorrelation plots
- Density plots
- Numerical diagnostics

Trace Plots

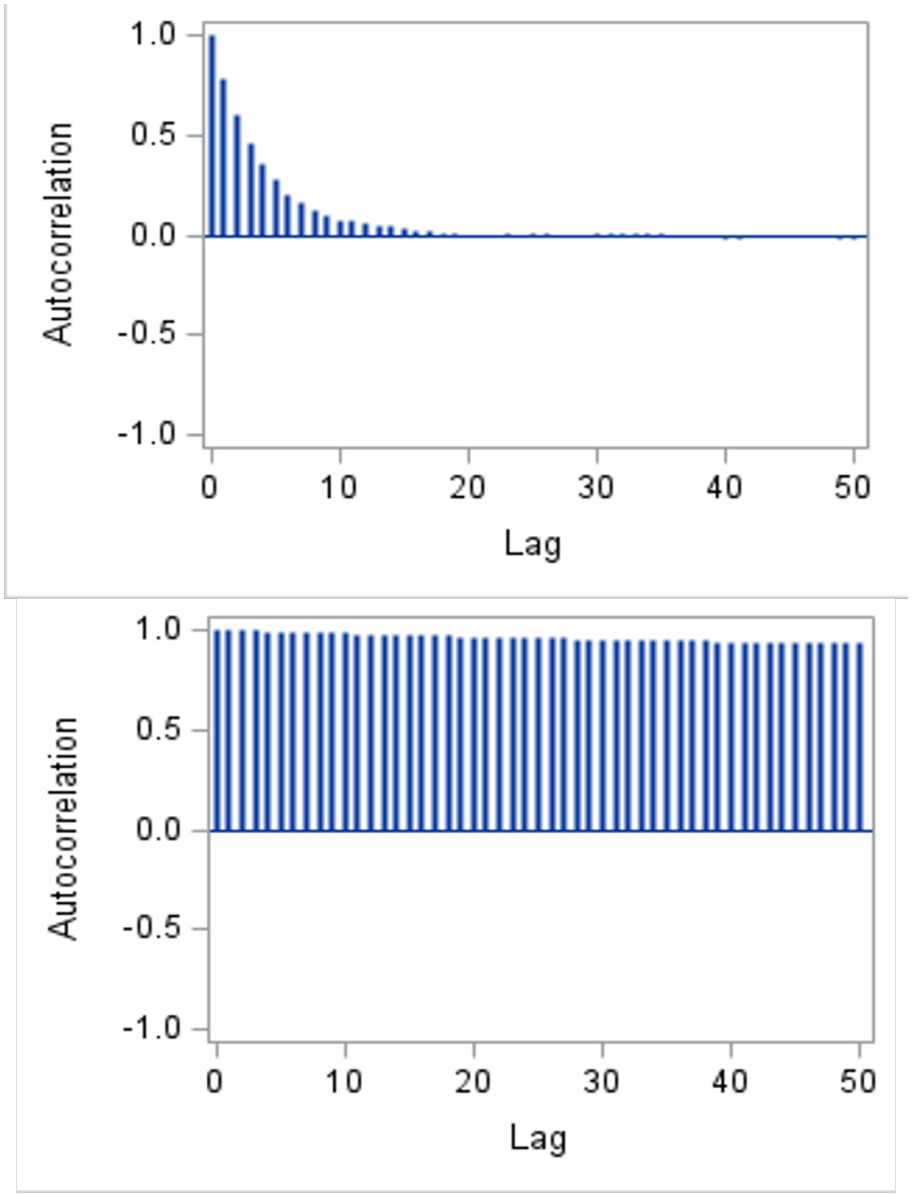
- Plots posterior estimates for each iteration across chain(s)
- Value of the parameter on the y-axis
- If convergence is achieved, should look like random noise
- Example from SAS's PROC MCMC with burn-in of 1000 iterations already removed for intercept in regression model (good convergence on left, poor convergence on right):





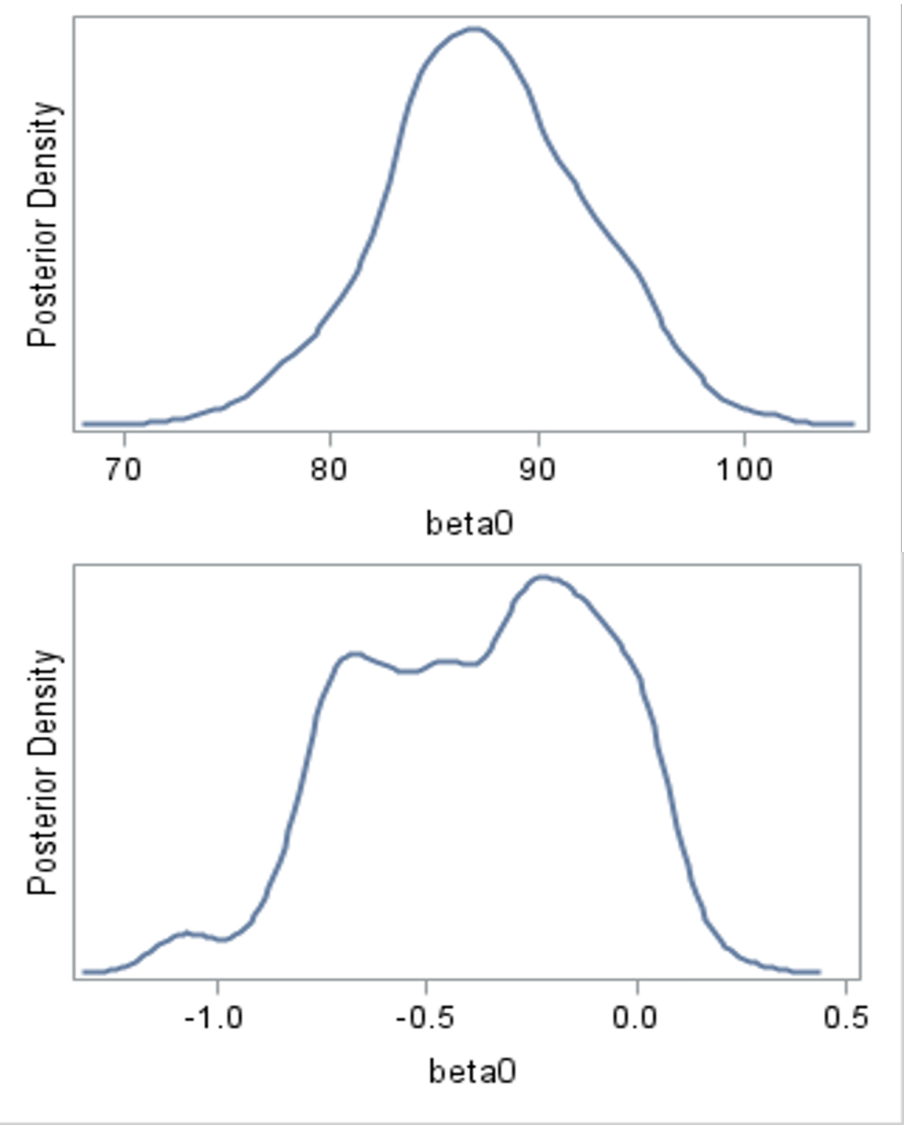
Autocorrelation Plots

- Pairwise correlation between iterations and different lags between iterations
- Well-functioning algorithms should have a fairly steep decline as lag increases
- This pattern suggests samples are likely to be random draws from posterior and simulation has adequately sampled the support of the parameter
- PROC MCMC example comparing better (top) and worse (bottom) autocorrelation:



Density/Histogram Plots

- Visualization of the posterior distribution as a histogram and/or density plot
- Typically looking for a fairly smooth shape with a single mode
- Bimodal or unusual patterns may indicate lack of convergence
- PROC MCMC example comparing better (top) and worse (bottom) convergence:



Numerical Diagnostics

In addition to plots to visualize convergence, other approaches exist to numerically summarize performance (and may differ by software).

Geweke Diagnostic:

- Compares values in early part of Markov chain to latter values
- Provides a two-sided test with a large absolute z-score indicating rejection of convergence

R-hat:

- A potential scale reduction factor proposed by Gelman and Rubin
- Estimates convergence based on variance of estimated parameter between chains and variance within chain
- Want values near 1 (some rules of thumb suggest <1.01 or <1.05 as adequate)

Bayesian Summaries from the Posterior Distribution

Point Estimate

Since our posterior distribution isn't a single value, but a distribution with varying densities over a range of possible parameter values, we have to decide how to summarize the point estimate.

In practice, we often use one of the measures of central tendency:

- Mean of the posterior distribution (may be affected by outliers/skewed posteriors)
- Median of the posterior distribution
- Mode of the posterior distribution (may not make sense of continuous outcomes)

These can all be estimated either from the conjugate distribution (i.e., we know the distribution and can estimate the value from it) or from pooling together all MCMC posterior chains after discarding the burn-in periods. The choice may also depend on the shape of the posterior distribution.

Defaults differ by software, so one should check what summary is used.

Posterior Probability

From our posterior distribution, we can estimate the *posterior probability* of any given hypothesis of interest. This may be calculated directly from a conjugate distribution (e.g., beta-binomial) or approximated from the MCMC posterior.

This is often thought of as analogous to the frequentist p-value. However, where the p-value has a very nuanced interpretation (i.e., the probability of observing something as or more extreme under the null hypothesis), the posterior probability has a very straight-forward interpretation (i.e., the probability of our hypothesis).

For example, if the posterior probability of the mean being greater than 0 is $P(\mu > 0) = 0.89$, we would say the probability the mean is greater than 0 is 89%.

Credible Intervals

From our posterior distribution, we can also estimate the *credible interval* (CrI) around our parameter(s) of interest.

This is often thought of as analogous to the frequentist confidence interval (CI). However, where the CI has a nuanced interpretation (i.e., we are 95% confident...), the CrI has a very straight-forward interpretation (i.e., there is a 95% probability that θ falls in our interval).

Unlike CIs, CrIs are not unique on a posterior distribution and different assumptions exist for how to calculate the interval (similar to some choices for bootstrap CIs):

- **Highest posterior density** (HPD): identifying the narrowest possible interval (always includes the mode)
- **Equal-tailed**: choosing the interval where the probability of being below and above the interval is equal (always includes the median)
- **Mean centered**: choosing the interval so it is centered at the mean

Prior Specification

Motivation

In this final section we will compare some different priors on a simulated data set estimating the posterior mean for a one-sample data set using `brms` in R. We will walk through an example with interpreting output and diagnostics in a separate lecture.

Let's simulate a sample of $n = 20$ observations from a normal distribution with a mean (SD) of 10 (2.5):

```
library(brms)

# simulate data
set.seed(6618)
dat <- data.frame(y = rnorm(n=20, mean=10, sd=2.5))
mean(dat$y); sd(dat$y)
```

```
## [1] 8.632887
```

```
## [1] 3.090654
```

Comparing Priors

Priors are often given descriptors like vague, noninformative, informative, etc. These are not well-defined and can depend on the eye of the beholder.

For our problem, we will implement a linear regression model with only an intercept considering:

1. $\beta_0 \sim N(0, 1000)$ (potentially the most "vague")
2. $\beta_0 \sim N(0, 1)$ (potentially informative due to $\sigma = 1$)
3. $\beta_0 \sim N(10, 1000)$ (center at simulated mean)
4. $\beta_0 \sim N(10, 1)$ (center at simulated mean, smaller SD)
5. $\beta_0 \sim U(-50, 50)$ (uniform)
6. $\beta_0 \sim U(-5, 5)$ (uniform, narrow range)

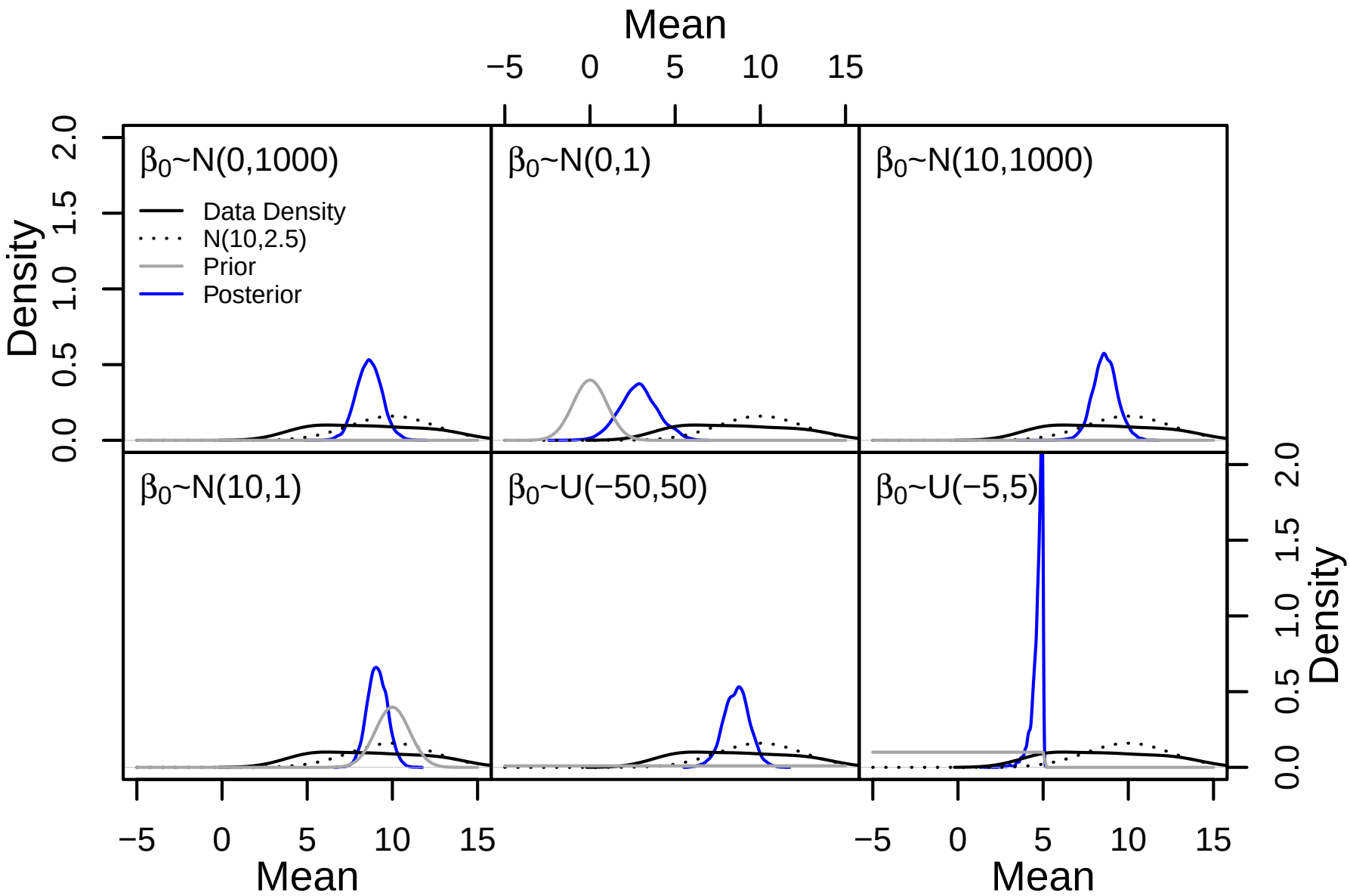
Basics in brms

We will fit the models in brms and then plot the results:

```
# Model with N(0,1000) prior
mod1 <- brm( y ~ 1, data=dat,
             prior = c(set_prior("normal(0,1000)",
                                 class="Intercept")),
             seed = 123
             )

# Model with U(-50,50) prior
mod5 <- brm( y ~ 1, data=dat,
             prior = c(set_prior("uniform(-50,50)",
                                 lb=-50, ub=50,
                                 class="Intercept")),
             seed = 123
             )
```

Plots of Likelihood, Prior, Posterior



Summary

Bayesian methods have a different approach to inference than frequentist methods.

The choice of prior specification may be challenging but gives great flexibility to incorporate prior beliefs or information.

Bayesian quantities (e.g., posterior probability, credible intervals) have intuitive estimates in comparison to frequentist quantities (e.g., p-values, confidence intervals).

We will explore the use of Bayesian methods in a multiple linear regression analysis in another lecture.