STA 6352, Report 9.1

Carson Slater *Baylor University*

Problem Description

We consider a Bayesian logistic regression model:

$$y_i \sim \text{Bernoulli}(p_i), \quad \text{logit}(p_i) = \beta_0 + \beta_1 x_i,$$

with diffuse priors:

$$\beta_0, \beta_1 \sim \mathcal{N}(0, 25).$$

We simulate 1000 datasets under the true values $\beta_0 = -2$ and $\beta_1 = 0.2, 0.5, 1.0$, for sample sizes $n = 100$, $n = 150$, and $n = 200$ and perform posterior inference using JAGS.

The goal is to:

- Estimate the average posterior means for β_0 and β_1,
- Compute 95% credible interval coverage,
- Estimate the empirical Bayesian power using the criterion:

$$\text{Reject } H_0 : \beta_1 = 0 \text{ if } \Pr(\beta_1 > 0 \mid \text{data}) > 0.95.$$

JAGS Model

The JAGS model defines the Bayesian logistic regression model structure. The linear predictor $\text{logit}(p_i)$ is modeled as a linear function of the covariate x_i. Diffuse priors are used to reflect minimal prior information:

```
model_string <- "
model {
  for (i in 1:N) {
    y[i] ~ dbern(p[i])
    logit(p[i]) <- beta0 + beta1 * x[i]
  }
  beta0 ~ dnorm(0, 0.04) # Equivalent to N(0, 25)
  beta1 ~ dnorm(0, 0.04)
}
"
```

Simulation Procedure (Parallelized)

To estimate posterior summaries efficiently across multiple simulated datasets, we parallelize the process using `furrr`.

The function below runs a full simulation study for a given combination of n and β_1. It:

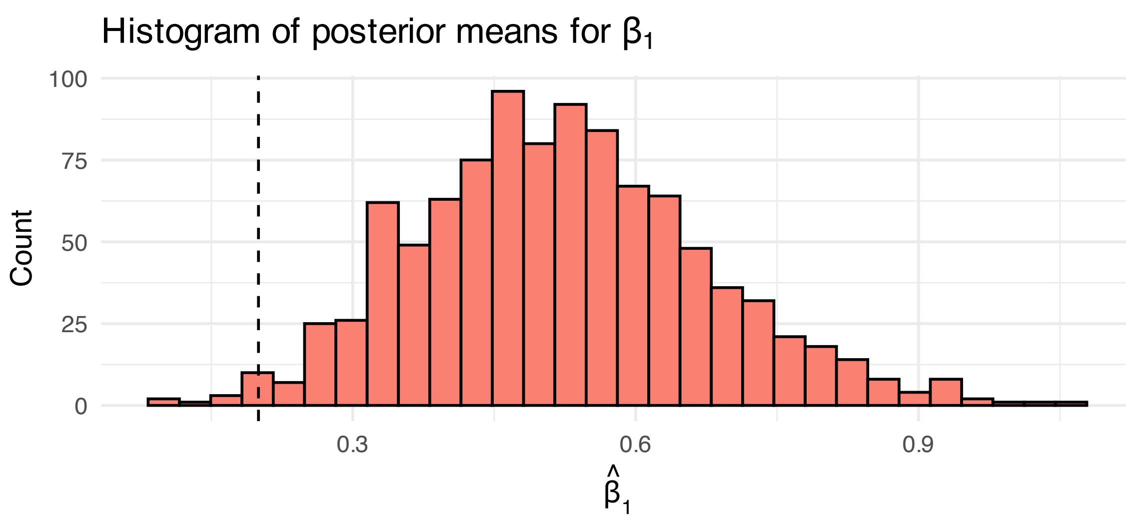
- Simulates covariates $x \sim U(0, 6)$.
- Computes success probabilities p using the logistic link.
- Simulates binary outcomes $y \sim \text{Bernoulli}(p)$.
- Fits the model in JAGS.
- Extracts posterior means and evaluates:
  - Whether the 95% CI covers the true β_1,
  - Whether $\Pr(\beta_1 > 0|\text{data}) > 0.95$ (power).

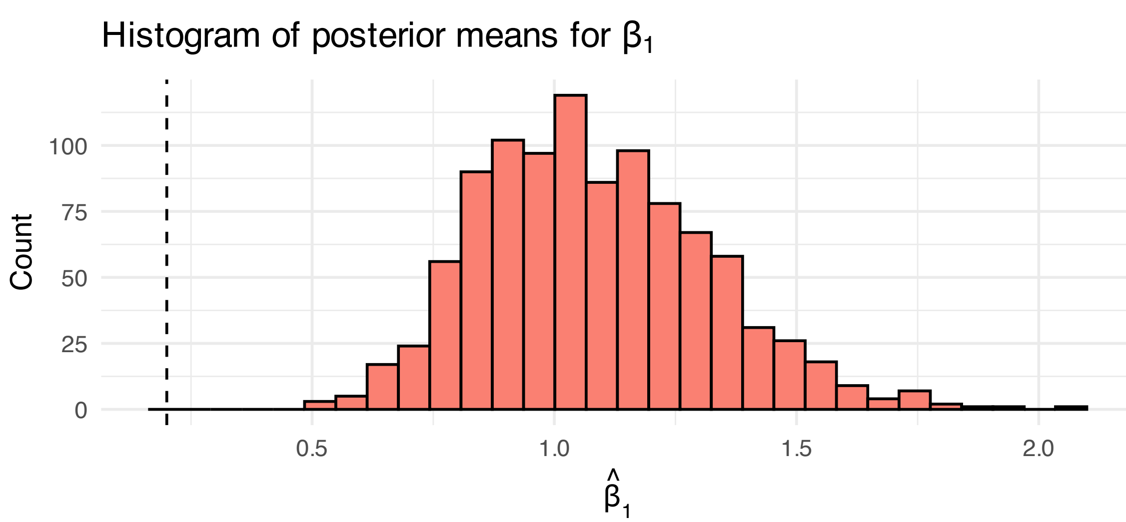
```
simulate_run_parallel <- function(n = 100, beta0_true = -2, beta1_true = 0.2,
  future_map_dfr(1:reps, function(r) {
    x <- runif(n, 0, 6)
    eta <- beta0_true + beta1_true * x
    p <- 1 / (1 + exp(-eta))
    y <- rbinom(n, 1, p)

    data_jags <- list(N = n, x = x, y = y)
    model <- jags.model(textConnection(model_string), data = data_jags, n.cha
    update(model, 500, progress.bar = "none")
    samples <- coda.samples(model, variable.names = c("beta0", "beta1"), n.ite
    summary_stats <- summary(samples)[[1]]

    beta1_samples <- as.matrix(samples)[, "beta1"]
    post_prob_beta1_gt_0 <- mean(beta1_samples > 0)
    beta1_ci <- quantile(beta1_samples, c(0.025, 0.975))

    tibble(
      beta0_mean = summary_stats["beta0", "Mean"],
      beta1_mean = summary_stats["beta1", "Mean"],
      beta1_gt_0 = post_prob_beta1_gt_0,
      beta1_covered = (beta1_true > beta1_ci[1] & beta1_true < beta1_ci[2])
    )
  }, .options = furrr_options(seed = TRUE))
}
```

Visualization of Posterior Estimates ($n = 100$)

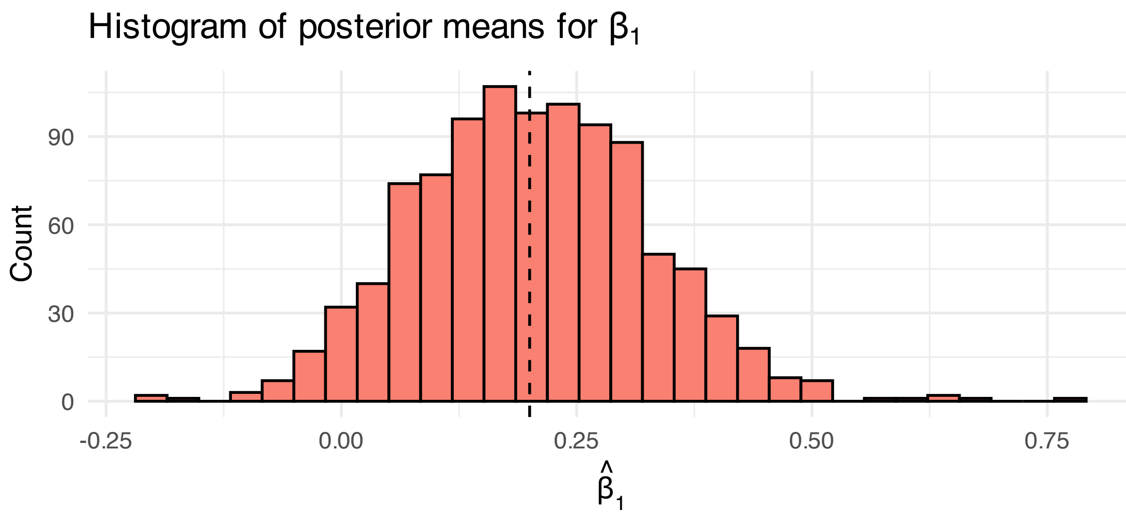
The histogram below shows the distribution of posterior means for β_1 when $n = 100$ and the true $\beta_1 = 0.2$. The dashed line indicates the true value. This visual helps evaluate bias and variance in the estimator.

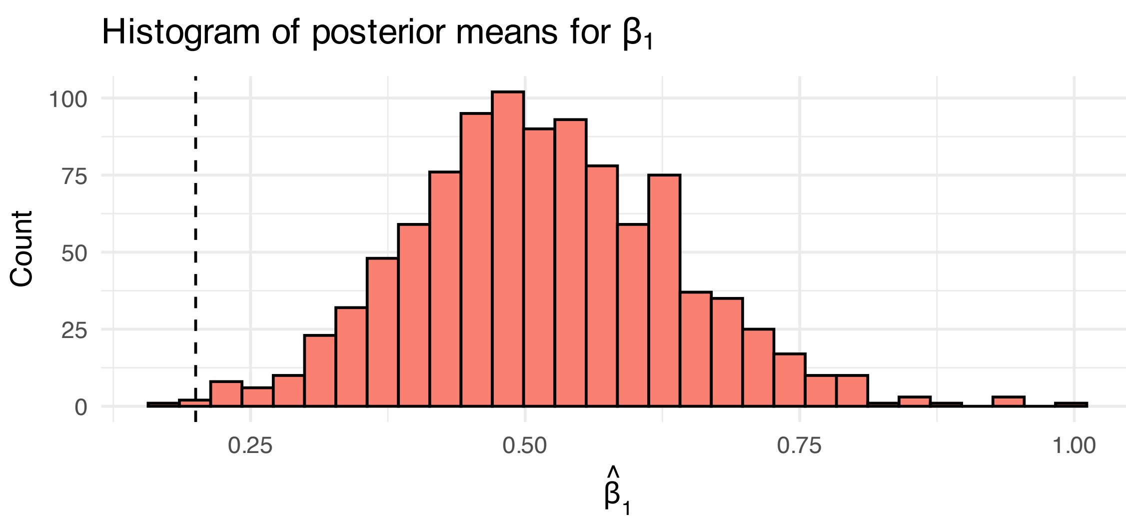

Histogram of posterior means for β1
Count
100
75
50
25
0
-0.3
0.0
0.3
0.6
β̂1

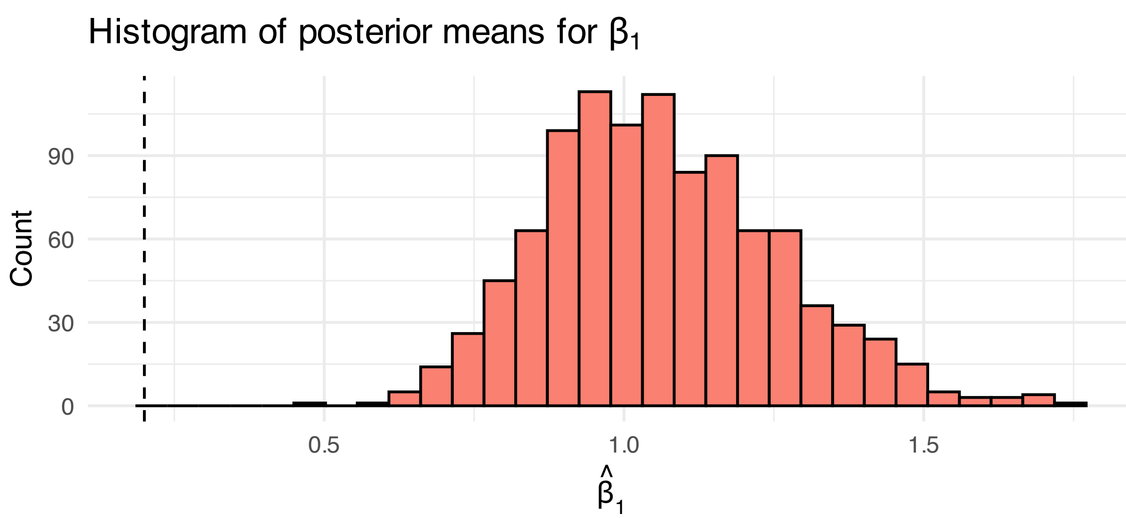

Histogram of posterior means for β1
Count
100
75
50
25
0
0.3
0.6
0.9
β̂1


Histogram of posterior means for β1
Count
100
75
50
25
0
0.5
1.0
1.5
2.0
β̂1

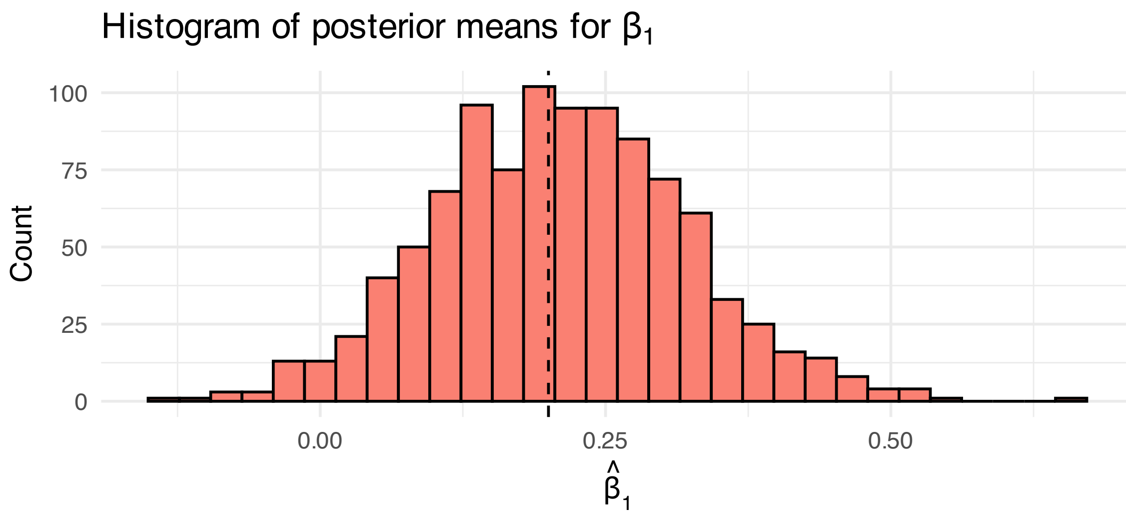
Visualization of Posterior Estimates ($n = 150$)

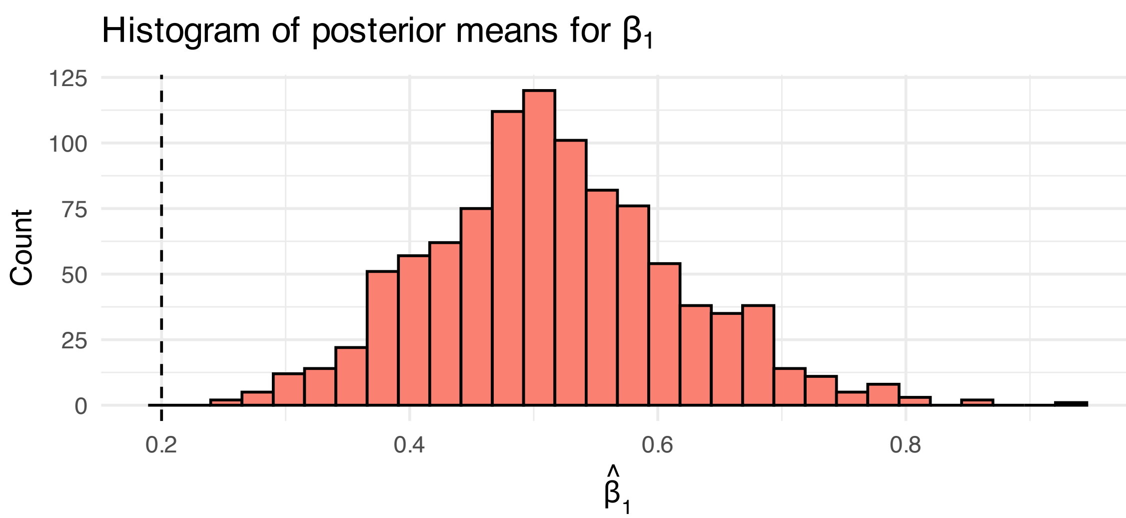


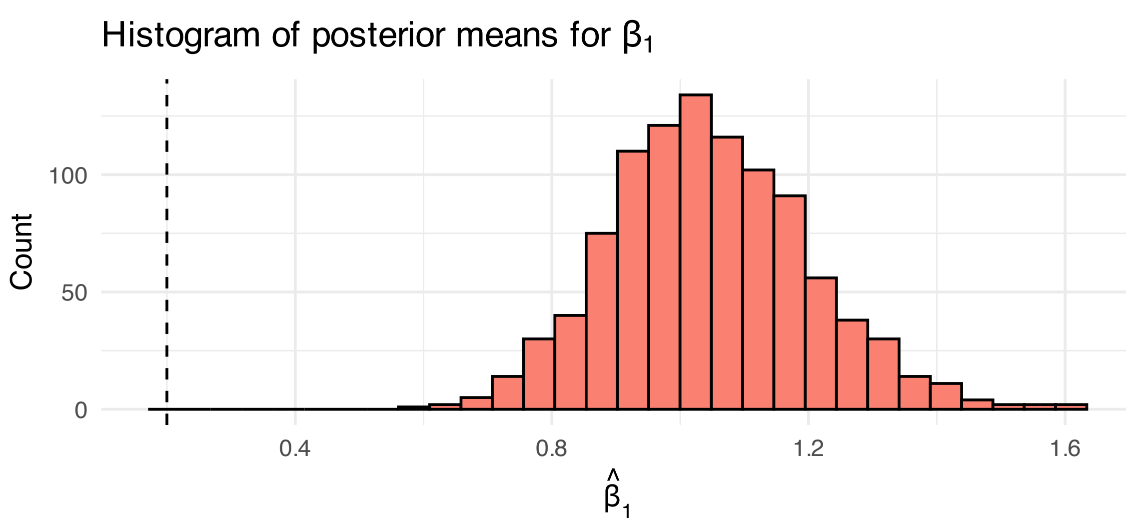




Visualization of Posterior Estimates ($n = 200$)







Summary Table

To formally report power and coverage for each scenario, we computed the following in the table:

n	True Beta1	Beta0 Mean	Beta1 Mean	Beta1 Coverage	Bayes Power
100	0.2	-2.057887	0.1945739	0.924	0.362
100	0.5	-2.071483	0.5207814	0.930	0.989
100	1.0	-2.158651	1.0832293	0.932	1.000
150	0.2	-2.057416	0.2034353	0.950	0.503
150	0.5	-2.069989	0.5171831	0.936	1.000
150	1.0	-2.121963	1.0618832	0.921	1.000
200	0.2	-2.078023	0.2106462	0.938	0.638
200	0.5	-2.067524	0.5166395	0.939	1.000
200	1.0	-2.097323	1.0469409	0.935	1.000