Bayesian Modeling and Inference: An Introduction to STAN for the Social Sciences

Bayesian Logistic with STAN

May 23, 2025

Contents

Objective

In this tutorial, we will:

- Implement a simple Bayesian logistic regression model in Stan.
- Fit the model to simulated data with a binary outcome using R.
- Inspect model convergence and posterior distributions.
- Perform posterior predictive checks for binary data.
- Interpret the results, including coefficients on the log-odds and odds ratio scales.
- Explore how changes in priors, data, and model specification affect the outcomes.

Prerequisites

- Basic understanding of R.
- R and RStudio installed.
- `rstan`, `bayesplot`, `ggplot2`, `dplyr`, `pROC` R packages installed.
- Conceptual understanding of Bayesian logistic regression.

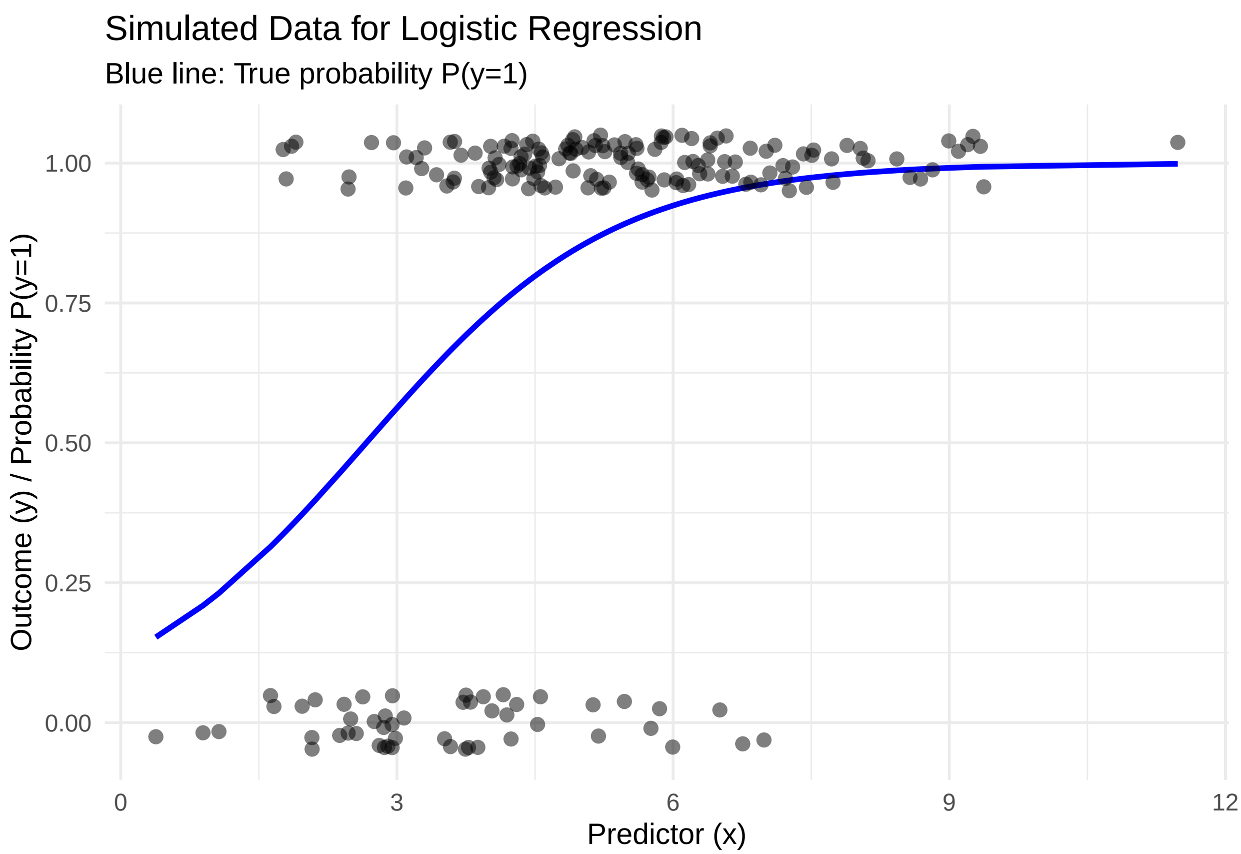
```r
# install.packages(c("rstan", "bayesplot", "ggplot2", "dplyr", "pROC"))
```

Section 1: Setting Up

```r
library(rstan)
library(bayesplot)
library(ggplot2)
library(dplyr)
library(pROC)

rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores())
```

Section 2: Simulating Data for Logistic Regression

```r
alpha_true_logodds <- -2.0
beta_true_logodds  <- 0.75
N_logistic         <- 200

set.seed(123)
x_logistic <- rnorm(N_logistic, mean = 5, sd = 2)
eta_logodds <- alpha_true_logodds + beta_true_logodds * x_logistic
prob_y_eq_1 <- 1 / (1 + exp(-eta_logodds))
y_logistic <- rbinom(N_logistic, size = 1, prob = prob_y_eq_1)

sim_data_logistic <- data.frame(x = x_logistic, y = y_logistic, prob = prob_y_eq_1)

ggplot(sim_data_logistic, aes(x = x)) +
  geom_line(aes(y = prob), color = "blue", size = 1) +
  geom_jitter(aes(y = y), width = 0, height = 0.05, alpha = 0.5, size=2) +
  labs(title = "Simulated Data for Logistic Regression",
       subtitle = "Blue line: True probability P(y=1)",
       x = "Predictor (x)", y = "Outcome (y) / Probability P(y=1)") +
  theme_minimal()
```



Section 3: Fitting the Logistic Regression Model in R

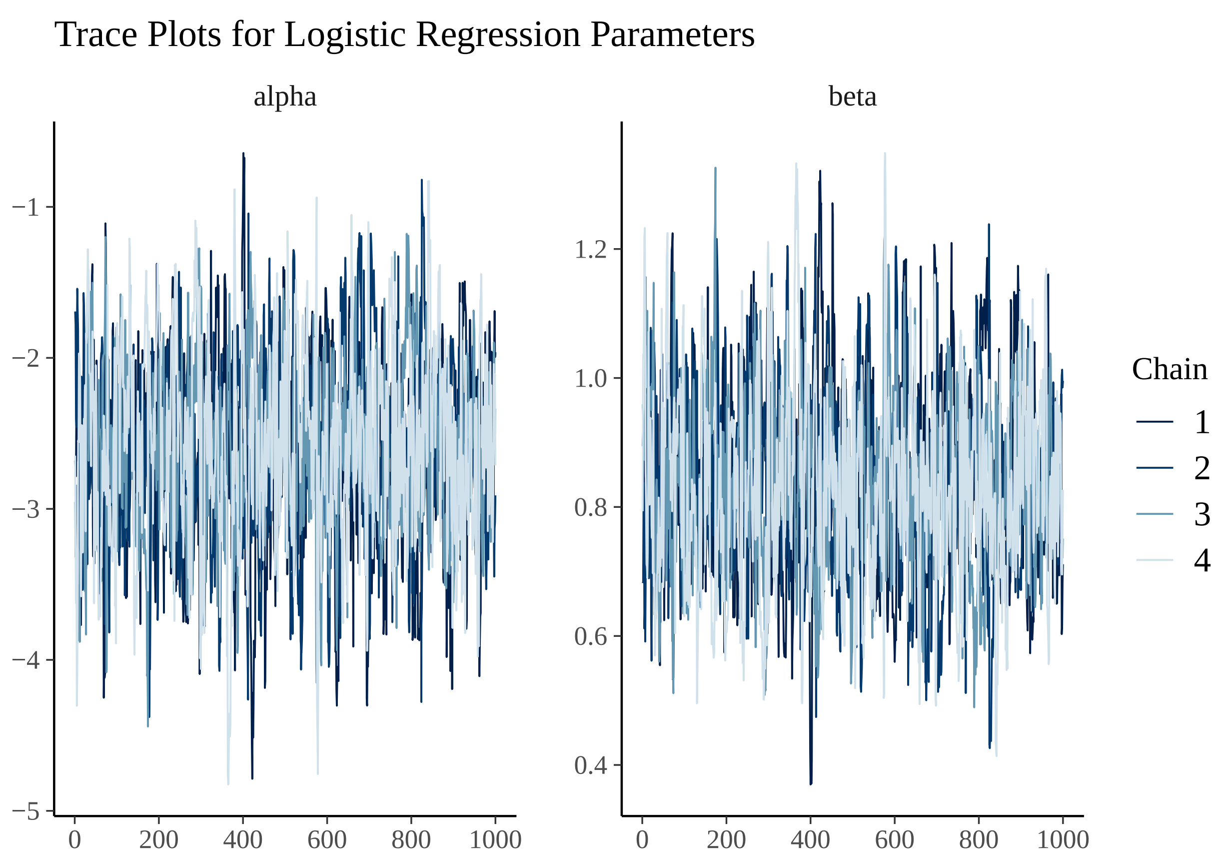
```r
setwd("/Users/user/Desktop/Lectures 2024/Bayesian Course - UoM/Bayesian Linear Regression")

stan_data_logistic <- list(
  N = N_logistic,
  x = sim_data_logistic$x,
  y = sim_data_logistic$y
)

model_logistic_compiled <- stan_model(file = "logistic_regression.stan")

fit_logistic <- sampling(
  object = model_logistic_compiled,
  data = stan_data_logistic,
  iter = 2000,
  warmup = 1000,
  chains = 4,
  seed = 456,
  refresh = 0
)
```

Section 4: Inspecting Model Fit and Convergence (Logistic)

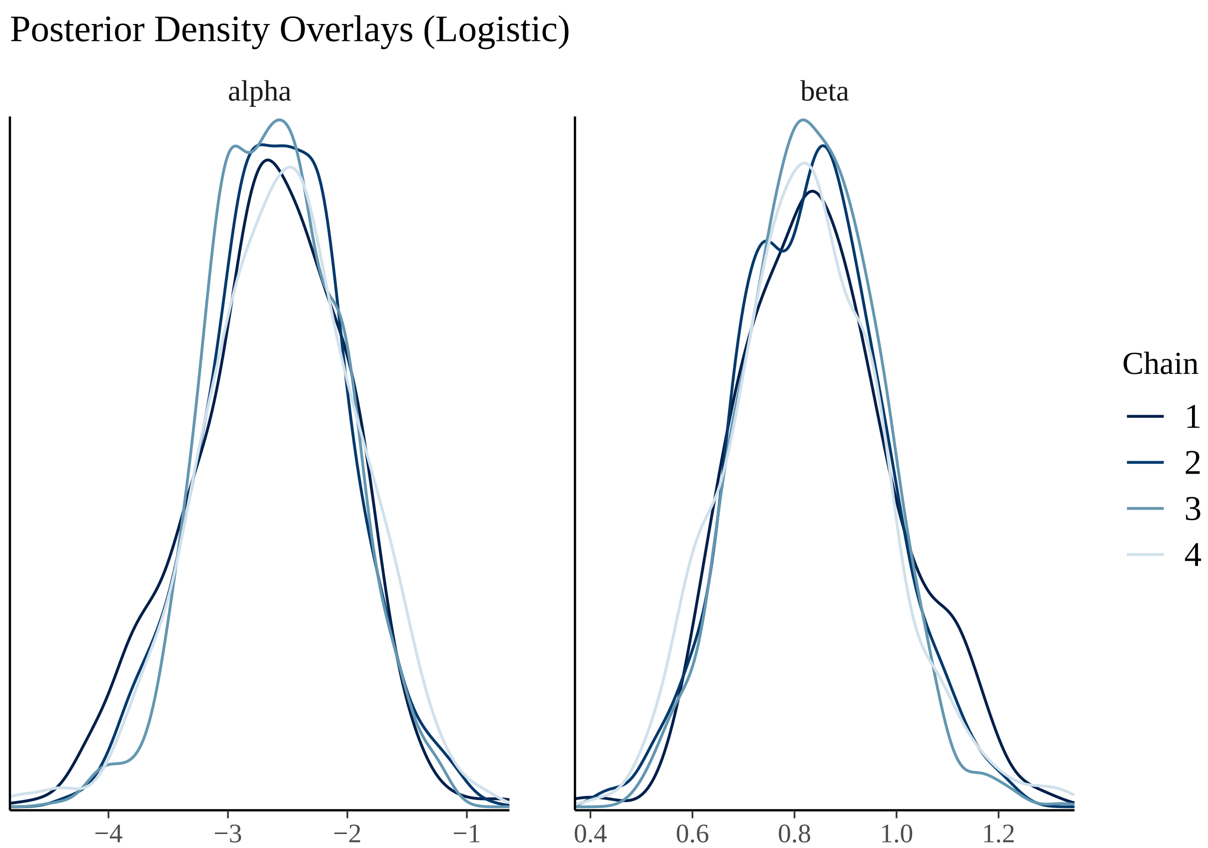
```
print(fit_logistic, pars = c("alpha", "beta"), probs = c(0.025, 0.5, 0.975))
```

```
## Inference for Stan model: anon_model.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##        mean se_mean   sd  2.5%   50% 97.5% n_eff Rhat
## alpha -2.61    0.02 0.61 -3.86 -2.60 -1.46   736    1
## beta   0.84    0.01 0.14  0.57  0.83  1.14   718    1
##
## Samples were drawn using NUTS(diag_e) at Mon May 19 11:25:53 2025.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

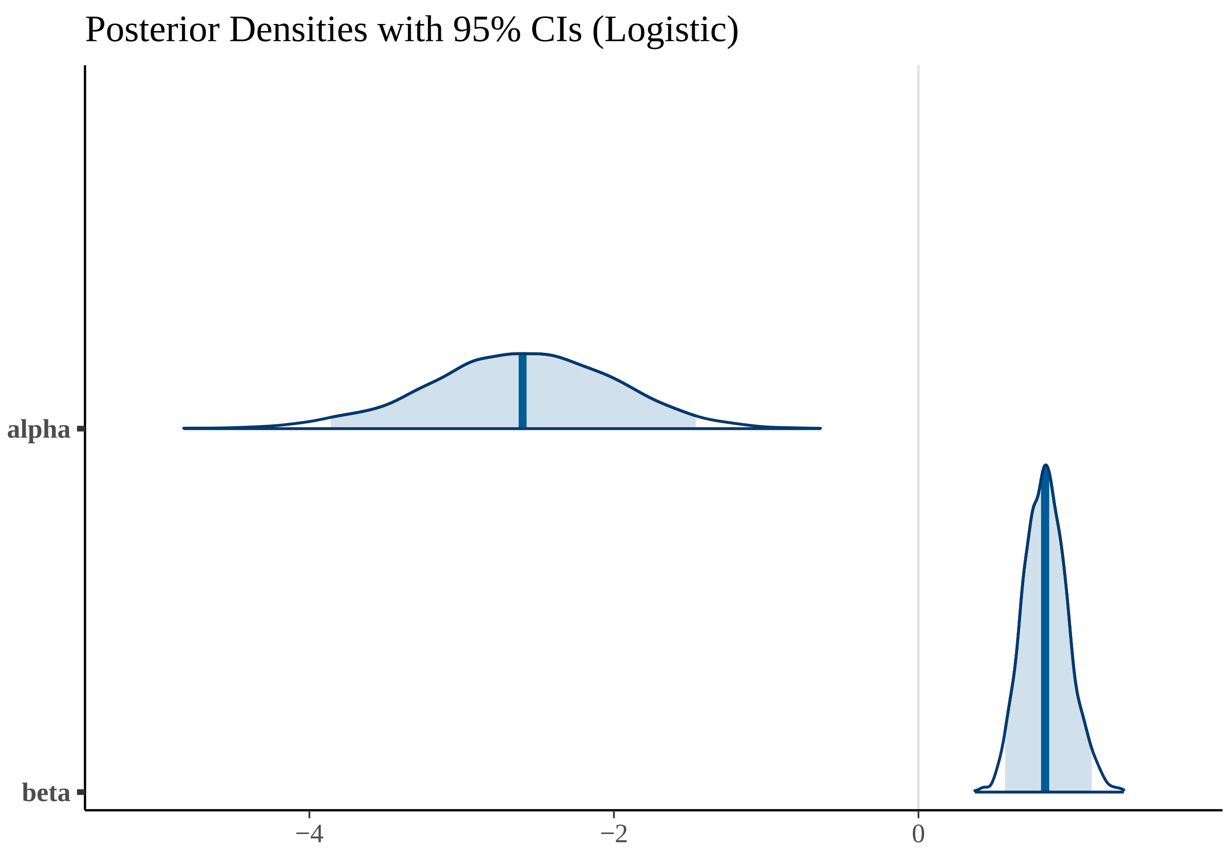
```
mcmc_trace(fit_logistic, pars = c("alpha", "beta")) +
  ggtitle("Trace Plots for Logistic Regression Parameters")
```




```
mcmc_dens_overlay(fit_logistic, pars = c("alpha", "beta")) +
  ggtitle("Posterior Density Overlays (Logistic)")
```

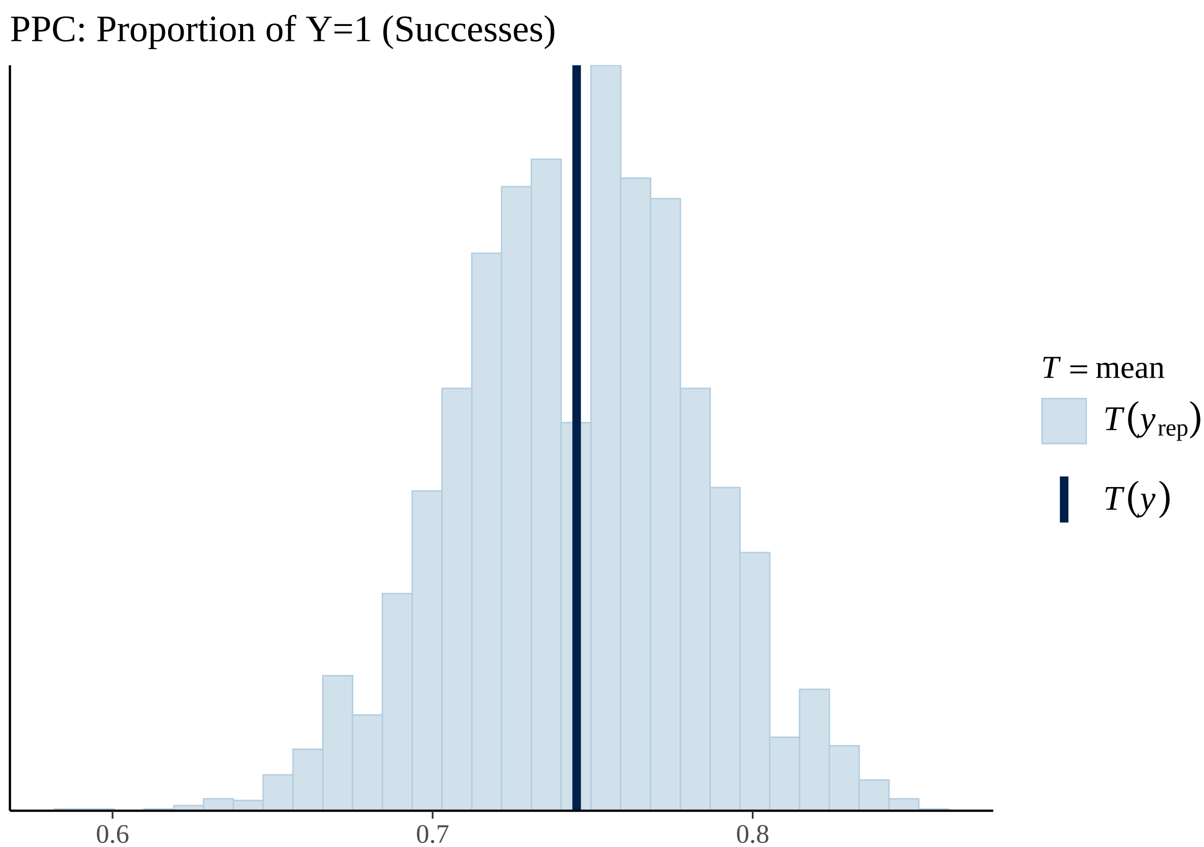



```r
mcmc_areas(fit_logistic, pars = c("alpha", "beta"), prob = 0.95) +
  ggtitle("Posterior Densities with 95% CIs (Logistic)")
```

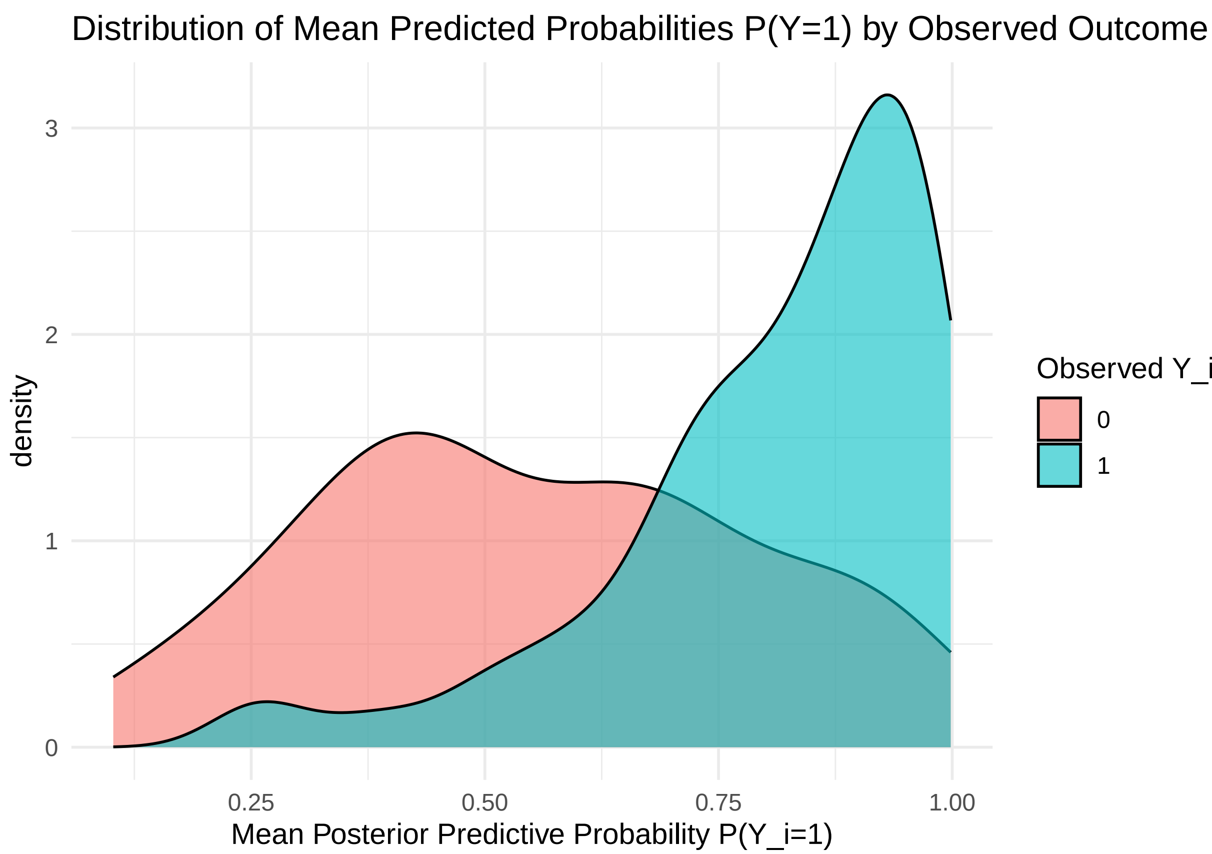


Section 5: Posterior Predictive Checks (PPCs) for Logistic Regression

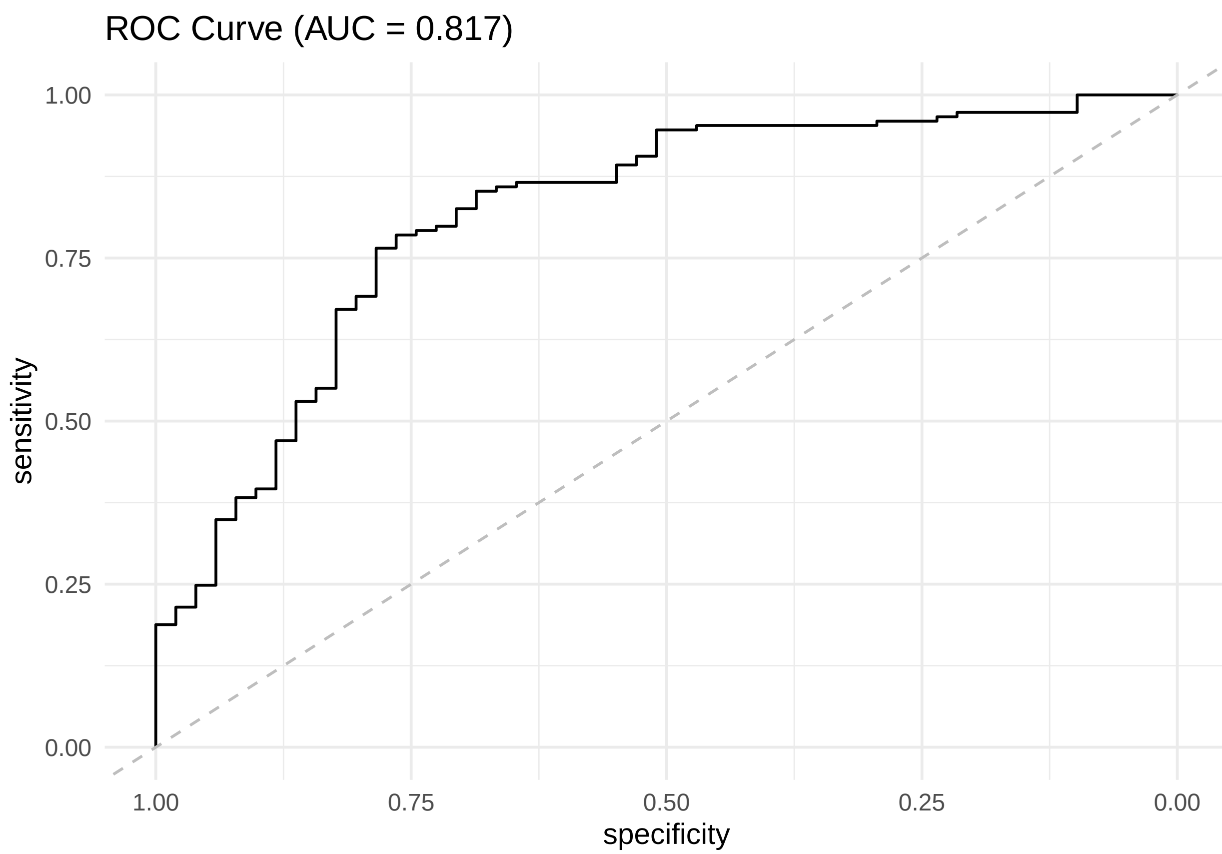
```
posterior_draws_logistic <- extract(fit_logistic)
y_rep_logistic_matrix <- posterior_draws_logistic$y_rep
prob_rep_logistic_matrix <- posterior_draws_logistic$prob_rep

ppc_stat(y = sim_data_logistic$y, yrep = y_rep_logistic_matrix, stat = "mean") +
  ggtitle("PPC: Proportion of Y=1 (Successes)")
```




```r
df_probs_obs <- data.frame(
  prob_pred = colMeans(prob_rep_logistic_matrix),
  observed_y = as.factor(sim_data_logistic$y)
)

ggplot(df_probs_obs, aes(x = prob_pred, fill = observed_y)) +
  geom_density(alpha = 0.6) +
  labs(title = "Distribution of Mean Predicted Probabilities P(Y=1) by Observed Outcome",
       x = "Mean Posterior Predictive Probability P(Y_i=1)",
       fill = "Observed Y_i") +
  theme_minimal()
```




```r
mean_pred_probs <- colMeans(prob_rep_logistic_matrix)
roc_obj <- roc(response = sim_data_logistic$y, predictor = mean_pred_probs, quiet=TRUE)
auc_value <- auc(roc_obj)

ggroc(roc_obj) +
  geom_abline(slope=1, intercept=1, linetype="dashed", color="grey") +
  ggtitle(paste0("ROC Curve (AUC = ", round(auc_value, 3), ")")) +
  theme_minimal()
```




```
print(paste("Area Under ROC Curve (AUC):", round(auc_value, 3)))
```

```
## [1] "Area Under ROC Curve (AUC): 0.817"
```

Section 6: Interpretation and Visualization (Logistic)

```
beta_logodds_samples <- posterior_draws_logistic$beta
beta_or_samples <- exp(beta_logodds_samples)

cat("Posterior summary for beta (Odds Ratio):
")
```

```
## Posterior summary for beta (Odds Ratio):
```

```
cat("Mean OR:", round(mean(beta_or_samples), 3), "
")
```

```
## Mean OR: 2.332
```

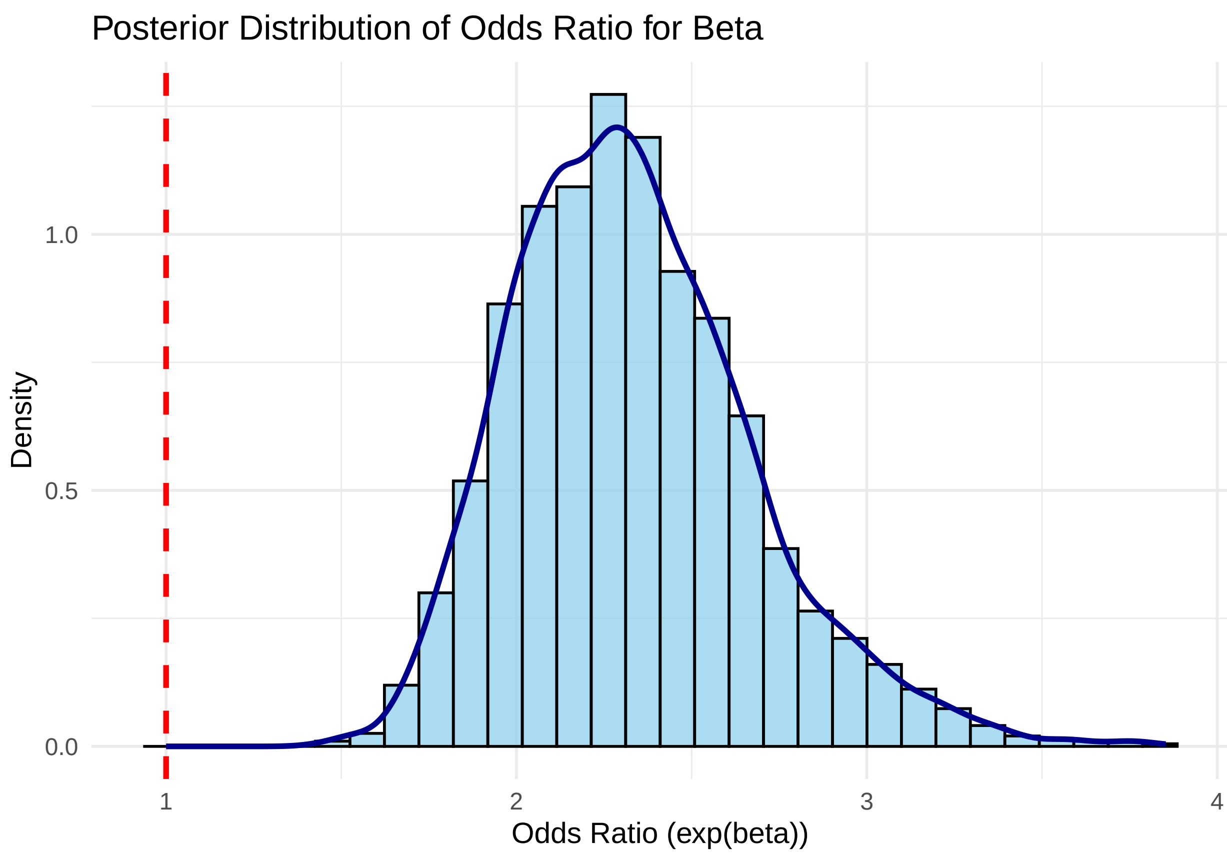
```r
cat("Median OR:", round(median(beta_or_samples), 3), "
")
```

```
## Median OR: 2.298
```

```r
cat("95% CI for OR: [",
    round(quantile(beta_or_samples, 0.025), 3), ", ",
    round(quantile(beta_or_samples, 0.975), 3), "]
")
```

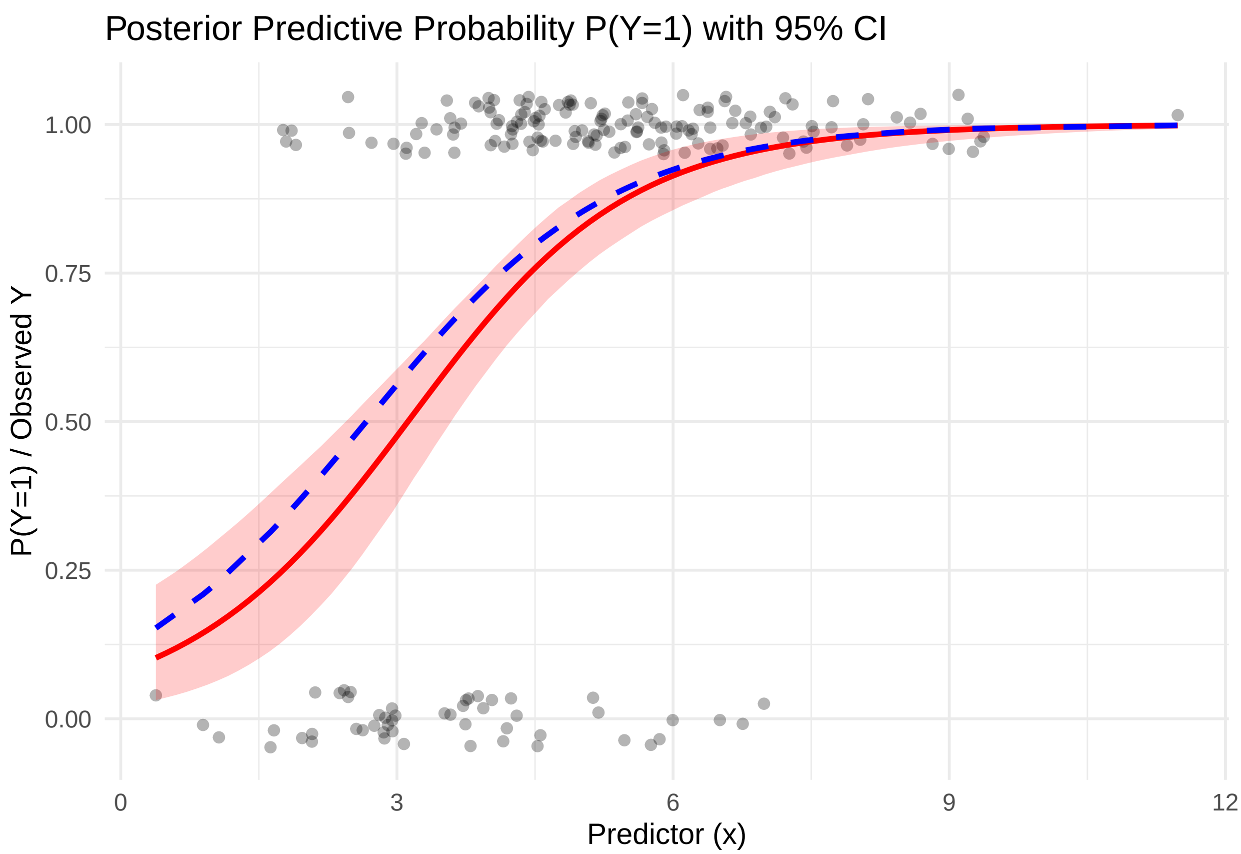
```
## 95% CI for OR: [ 1.767 ,  3.122 ]
```

```r
ggplot(data.frame(OR_beta = beta_or_samples), aes(x = OR_beta)) +
  geom_histogram(aes(y = ..density..), bins = 30, fill = "skyblue", color = "black", alpha = 0.7) +
  geom_density(color = "darkblue", size = 1) +
  geom_vline(xintercept = 1, linetype = "dashed", color = "red", size = 1) +
  labs(title = "Posterior Distribution of Odds Ratio for Beta",
       x = "Odds Ratio (exp(beta))",
       y = "Density") +
  theme_minimal()
```



Section 7: Visualize Posterior Predictive Probability Curve:

```
x_seq <- seq(min(sim_data_logistic$x), max(sim_data_logistic$x), length.out = 100)
pred_probs_matrix <- matrix(NA, nrow = length(posterior_draws_logistic$alpha), ncol = length(x_seq))

for (i in 1:length(posterior_draws_logistic$alpha)) {
  eta_seq <- posterior_draws_logistic$alpha[i] + posterior_draws_logistic$beta[i] * x_seq
  pred_probs_matrix[i, ] <- 1 / (1 + exp(-eta_seq))
}

mean_pred_probs_curve <- colMeans(pred_probs_matrix)
lower_ci_probs_curve <- apply(pred_probs_matrix, 2, quantile, probs = 0.025)
upper_ci_probs_curve <- apply(pred_probs_matrix, 2, quantile, probs = 0.975)

pred_df <- data.frame(
  x_val = x_seq,
  mean_prob = mean_pred_probs_curve,
  lower_prob = lower_ci_probs_curve,
  upper_prob = upper_ci_probs_curve
)

ggplot(sim_data_logistic, aes(x = x, y = y)) +
  geom_jitter(width = 0, height = 0.05, alpha = 0.3, size=1.5) +
  geom_line(data = pred_df, aes(x = x_val, y = mean_prob), color = "red", size = 1) +
  geom_ribbon(data = pred_df, aes(x = x_val, ymin = lower_prob, ymax = upper_prob),
              fill = "red", alpha = 0.2, inherit.aes = FALSE) +
  geom_line(aes(y = prob), color = "blue", linetype = "dashed", size = 1) +
  labs(title = "Posterior Predictive Probability P(Y=1) with 95% CI",
       x = "Predictor (x)", y = "P(Y=1) / Observed Y") +
  theme_minimal()
```



Section 8: Student Exploration Questions (Logistic)

1. Influence of Priors

- Q1.1: Change the prior for `beta` to `normal(0, 10)`. Does it affect inference significantly?
- Q1.2: Set an off-centre prior for `alpha`, e.g., `alpha ~ normal(2, 1)`. How does it influence the posterior?

2. Impact of Data

- Q2.1: Reduce `N_logistic` to 50. How do posterior uncertainties change?
- Q2.2: Simulate with `beta_true_logodds <- 0.1`. Can the model detect the weak effect?

3. Interpreting Odds Ratios

- Q3.1: If `beta` = 0.693, `OR = exp(0.693) (approx 2)`. Interpret this.
- Q3.2: If the 95% CI for OR is [0.85, 2.5], what does this mean for the effect?

4. Model Fit Assessment

- Q4.1 (Conceptual): Discuss classification metrics: accuracy, precision, recall, F1-score.
- Q4.2: Try:

```r
ppc_error_binned(y = sim_data_logistic$y, yrep = y_rep_logistic_matrix)
```

5. Multiple Predictors (Advanced)

- Simulate a second predictor `x2_logistic`.
- Update Stan code to use matrix `X` and vector `beta`.
- Adjust likelihood to `y ~ bernoulli_logit(alpha + X * beta);`.
- Re-fit and check recovery of both coefficients.