3303 Project

Lun Li.14415

2025-04-24

Introduction

In financial decision-making, historical returns is very important to guide strategy. Although past performance doesn't guarantee future results, it is still valuable to find patterns in past data. This project models stock performance as a sequence of binary outcomes—positive (1) or negative (0), to estimate the probability of a positive return, denoted by θ, for each stock.

Because stock returns are influenced by both sector-specific and market-wide factors, I use a hierarchical Bayesian model to account for dependencies at the stock, sector, and market levels. The goal is to estimate the posterior distributions of θ and determine which sectors and stocks are best to invest.

Data Description and Exploratory Data Analysis

The dataset contains 1,500 observations representing 30 time periods of return data for 50 stocks. These stocks are evenly split across five sectors, with 10 unique stocks in each sector. Each observation includes:

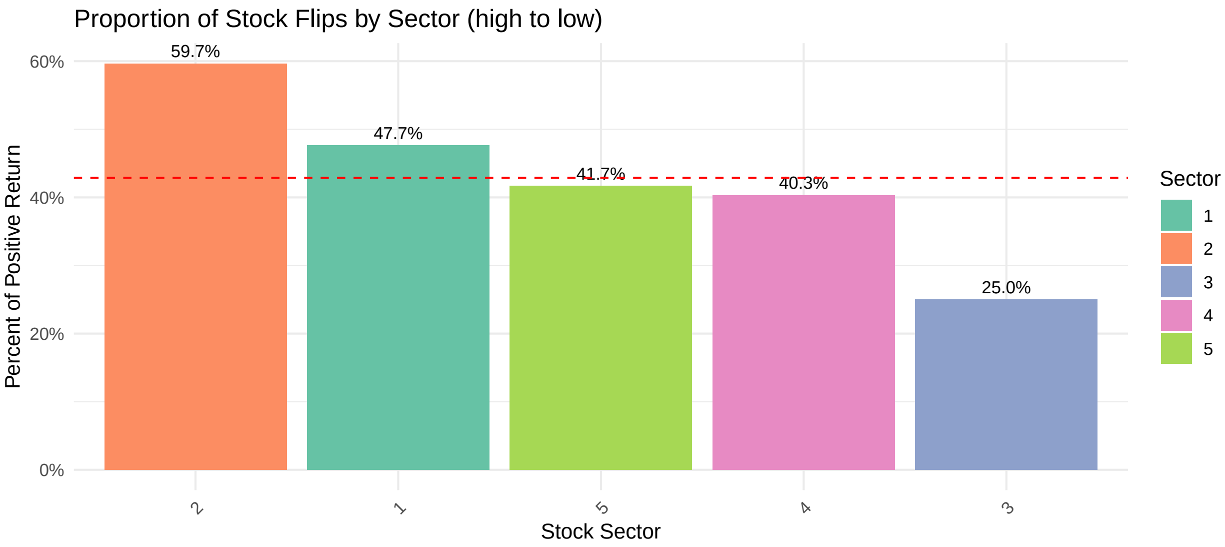
- `sector`: The industry group the stock belongs to (1 through 5).
- `stock`: An index representing the individual stock within a sector (1 through 10).
- `flip`: A binary indicator of whether the stock return was positive (1) or negative (0).

Exploratory Data Analysis (EDA)

Sector-Level Overview

Begin by summarizing the average flip (positive return) probability by sector.

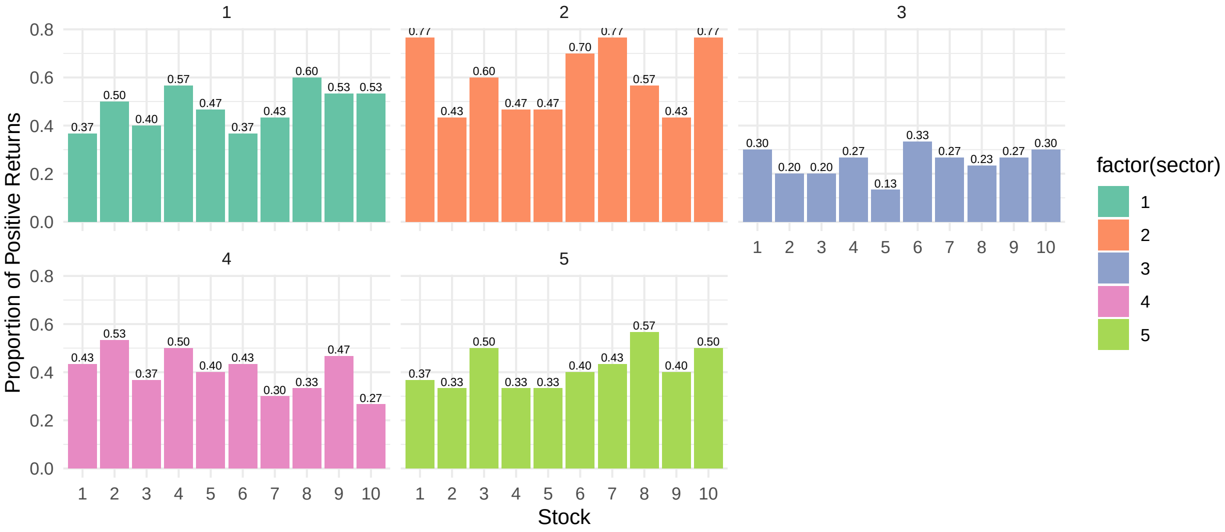
From the table above, Sector 2 shows the highest average flip rate (around 60%), while Sector 3 has the lowest (aroung 25%). Sector 1, 4, and 5 are performing aroung average level.



Sector 2 apparently outperforms the rest, exceeding the overall mean flip rate (indicated by the red dashed line). Sector 3 significantly underperforms.

Stock-Level Variation Within Sectors

Next, analyze average flip rates by stock within each sector. This reveals performance variation at the individual stock level.



Some sectors, such as Sector 2, show several stocks with flip rates near or above 0.70, indicating consistently strong performance. In contrast, Sector 3 stocks mostly fall below 0.40, confirming its overall bearish trend.

Data Limitations and Potential Sources of Bias

This data simplifies stock performance into binary outcomes, which is far more simpler than real situations. The magnitude of returns and potential time trends, most important parts of stock analysis in real life, are ignored. Also, the sample of stocks and sectors may not fully represent the broader market. The hierarchical characteristic of the mode may also obscure individual performances.

Model Description and Fitting

To estimate the probability that a stock yields a positive return, propose a hierarchical Bayesian logistic regression model. This model contains three nested levels of variation: stock-level effects, sector-level effects, and a market-wide effect.

Let:

- y_i be the observed return for the i-th observation ($y_i = 1$ if the return is positive, 0 otherwise),
- θ_i be the probability of a positive return for observation i,
- $j = \text{stock\_id}_i$ be the stock associated with observation i,
- $k = \text{stock\_sector}_j$ be the sector associated with stock j.

Model the data as:

$$y_i \sim \text{Bernoulli}(\theta_i), \quad \text{logit}(\theta_i) = \beta_{1,j}, \quad j = 1, 2, ...50$$

The log-odds of success for stock j is modeled as:

$$\beta_{1,j} \sim \mathcal{N}(\alpha_k, \tau_{\text{stock}}^{-1}) \quad k = 1, 2, 3, 4, 5$$

Each sector-level effect α_k is drawn from a normal distribution centered on the market-level average:

$$\alpha_k \sim \mathcal{N}(\beta_0, \tau_{\text{sector}}^{-1})$$

The overall market-level parameter:

$$\beta_0 \sim \mathcal{N}(0, 100^{-1})$$

Prior Specification

Use weakly informative priors for the precision (inverse variance) terms:

$$\tau_{\text{stock}} \sim \text{Gamma}(2, 2) \tau_{\text{sector}} \sim \text{Gamma}(2, 2)$$

The hierarchical structure enables seperate analysis for individual stocks and general sector performance: Estimates for each stock are influenced by its sector. And sector estimates are influenced by the market-level mean.

Model Fitting and Convergence

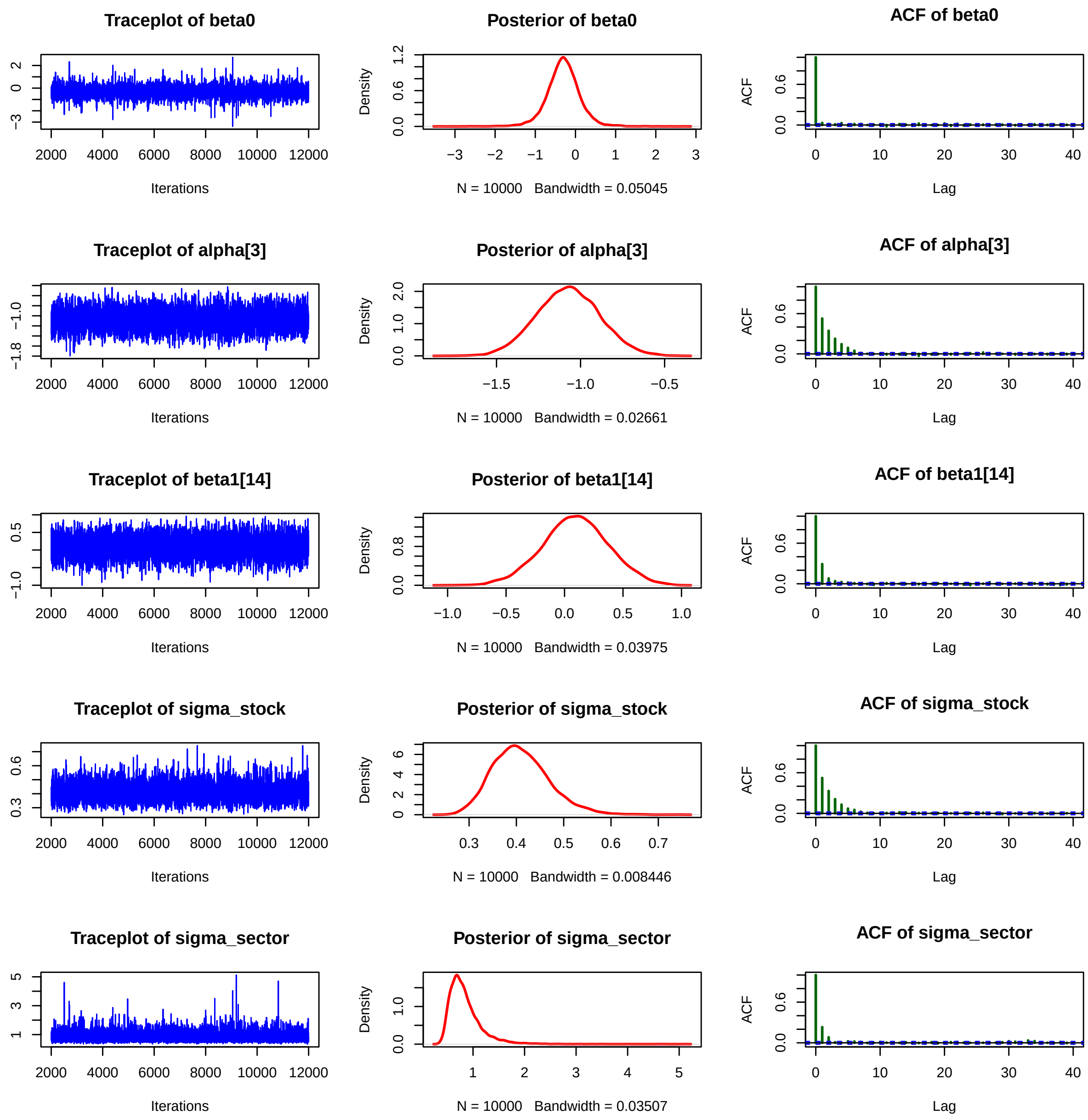
The model was implemented in JAGS and fitted using Markov Chain Monte Carlo (MCMC) with the following settings:

- **Iterations**: 10,000
- **Burn-in**: 1,000
- **Adaptation**: 1,000
- **Chains**: 1

Initial values were specified to aid convergence: - $\beta_0 = 0$ (centered at neutral market condition) - α_k and $\beta_{1,j}$ initialized using random normal values - τ values drawn from $\text{Gamma}(2, 2)$

Monitor convergence using trace plots and summary diagnostics. The trace plots showed stable mixing and no signs of divergence after burn-in. Posterior summaries, including means and credible intervals for all parameters, were extracted from the resulting MCMC samples.

Convergence can be demonstrated by the following trace plots, distribution plots, and ACF plots. For conciseness, representative examples are randomly selected for alpha and beta1.



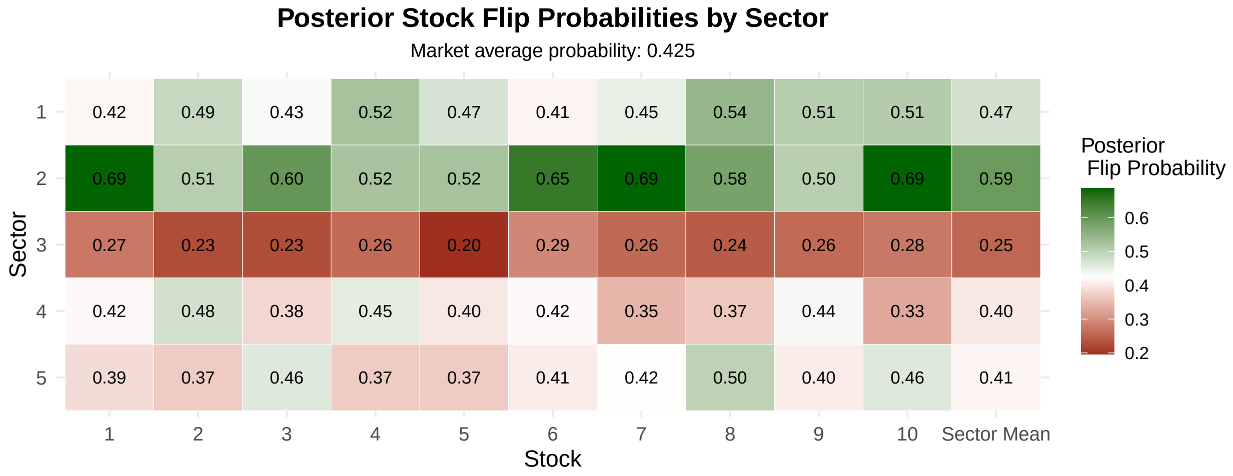
Posterior Inference and Interpretation

Visualizing Posterior Stock Probabilities

To better understand the inferred performance of individual stocks and sectors, I computed the posterior means of model parameters and transformed them to probabilities using the inverse logit function:

- θ_j: Probability of a positive return for stock j.
- α_k: Sector-level log-odds, converted to probability to represent average performance in sector k.
- β_0: Market-wide log-odds, converted to the overall market average flip probability.

I organized the posterior probabilities into a heatmap where each tile shows the estimated probability of a positive return for a given stock. An additional column labeled "Sector Mean" shows the posterior average probability for each sector. The color scale ranges from **dark red (underperforming)** to **dark green (overperforming)**, centered on the **market average** of approximately 0.42, which is calculated by invlogit(β_0).



Interpretation of Results

Market-Level Insight

The estimated market-wide probability of a positive return is **0.425**, serving as a reference for evaluating the relative performance of sectors and individual stocks.

Sector-Level Performance

- **Sector 1** demonstrates good performance, with only three stocks exhibiting posterior probabilities below market level of **0.425**.
- **Sector 2** consistently outperforms other sectors, with several stocks exhibiting posterior probabilities of 0.65 and highest reaching 0.69. Its sector mean is approximately 0.59, the highest among all groups.
- **Sector 3** demonstrates the weakest performance, with all ten stocks displaying posterior probabilities below 0.30 and a sector mean around **0.26**.
- Sectors **4 and 5** have sector means near the market average (0.40 and 0.41), reflecting slightly negative performance.

Stock-Level Highlights

- The top-performing stocks include **Stocks 1, 6, 7, and 10 in Sector 2**. The highest are Stock 1, 7, and 10, with posterior probabilities of **0.69**.
- In contrast, **Stocks 2, 3, and 5 in Sector 3** exhibit some of the lowest probabilities (as low as 0.20), indicating historical underperformance.
- Notable stocks in other sectors include:

  - **Stock 8 in Sector 1** (0.54)
  - **Stock 2 in Sector 4** (0.48)
  - **Stock 8 in Sector 5** (0.50)

Best Sector

- **Sector 2** was the top-ranked sector in the majority of iterations, with an estimated probability of **59%**.
- All other sectors had significantly lower probabilities of being the best.

Best Stock Within Each Sector

Similarly, within each sector, the probability that a given stock was the top performer was estimated by comparing posterior $\beta_{1,j}$ values:

- In **Sector 2**, Stocks 1, 6, 7, and 10 were the most frequently top-ranked.
- In **Sector 3**, no standout performer emerged; the top stock had low probability due to generally poor performance.
- Other likely top stocks by sector:
  - **Sector 1**: Stock 8
  - **Sector 4**: Stock 2
  - **Sector 5**: Stock 8

Investment Implications

Based on the analysis, **Sector 2** presents the strongest investment potential, both in terms of sector-wide strength and individual stocks. **Sectors 1, 4, and 5** may offer moderate opportunities with careful selection. **Sector 3 is not recommended**, given its consistently low probability of generating positive returns.

The posterior probabilities derived from the hierarchical Bayesian model provide a rigorous framework for identifying both sector and stock-level strengths under uncertainty.

Appendix: R Code

```
invlogit <- function(x) {
  1 / (1 + exp(-x))
}
logit <- function (x) {
  log(x/(1-x))
}
```

```
library(coda)
library(rjags)
library(ggplot2)
library(dplyr)
library(tidyverse)
library(knitr)
library(kableExtra)
```

```
stock <- readr::read_csv("dataset20.csv")
head(stock)
```

```
mean_flip = mean(stock$flip)
print(mean_flip)
```

```
# Calculate mean flip for all stocks (market average)
mean_flip <- mean(stock$flip)

# Prepare sector summary
stock_sector_dist <- stock %>%
  group_by(sector) %>%
  summarize(proportion = mean(flip), .groups = "drop")

# Plot with value labels
ggplot(stock_sector_dist, aes(x = reorder(factor(sector), -proportion),
                              y = proportion, fill = factor(sector))) +
  geom_col() +
  geom_text(aes(label = scales::percent(proportion, accuracy = 0.1)),
            vjust = -0.5, size = 3) +  # <-- ADD TEXT LABELS
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(labels = scales::percent_format(accuracy = 1)) +
  geom_hline(yintercept = mean_flip, linetype = "dashed", color = "red") +
  labs(
    title = "Proportion of Stock Flips by Sector (high to low)",
    x = "Stock Sector",
    y = "Percent of Positive Return",
    fill = "Sector"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
stock %>%
  group_by(sector, stock) %>%
  summarize(p_flip = mean(flip), .groups = 'drop') %>%
  ggplot(aes(x = factor(stock), y = p_flip, fill = factor(sector))) +
  scale_fill_brewer(palette = "Set2") +
  geom_col() +
  geom_text(aes(label = sprintf("%.2f", p_flip)),
            vjust = -0.3, size = 2) +  # Add exact value labels
  facet_wrap(~sector) +
  ylab("Proportion of Positive Returns") +
  xlab("Stock") +
  theme_minimal()
```

```
stock_sector <- rep(1:5, each = 10) # Length-50 vector that maps
                                    # each stock to its sector
stock_id <- rep(1:50, each = 30) # Length-1500 vector that maps
                                 # each flip to its stock
```

```
# MCMC setup
niter <- 10000
nburn <- 1000
nadapt <- 1000

# Unique stock and sector identifiers
N_stock <- 50
N_sector <- length(unique(stock$sector))
N <- nrow(stock)

data_jags <- list(
  y = stock$flip,
  stock_id = stock_id,
  stock_sector = stock_sector,  # (precomputed!)
  N_stock = N_stock,  # (precomputed!)
  N_sector = N_sector,
  N = N
)
```

```
model_stock_2 <- "
model {
  for (i in 1:N) {
    y[i] ~ dbern(theta[i])
    logit(theta[i]) <- beta1[stock_id[i]]
  }


  # Stock-level effects, nested within sector
  for (j in 1:N_stock) {
    beta1[j] ~ dnorm(alpha[stock_sector[j]], tau_stock)
  }

  # Sector-level effects
  for (k in 1:N_sector) {
    alpha[k] ~ dnorm(beta0, tau_sector)
  }

  # Market-level effects, like inflation
  beta0 ~ dnorm(0, 1/100)

  # Hyperpriors
  tau_stock ~ dgamma(1, 1)
  tau_sector ~ dgamma(1, 1)

  sigma_stock <- pow(tau_stock, -0.5)
  sigma_sector <- pow(tau_sector, -0.5)
}
"
```

```
inits_2 <- function() list(
  beta0 = 0,  # Center the overall market effect
  alpha = rnorm(N_sector, 0, 1),  # Reasonable starting sector effects
  beta1 = rnorm(N_stock, 0, 1),   # Starting stock-level log-odds
  tau_stock = rgamma(1, 1, 1),    # Centered around 1
  tau_sector = rgamma(1, 1, 1),   # Same here
  .RNG.name = "base::Wichmann-Hill",   # specify RNG type
  .RNG.seed = 123                      # specify RNG seed
)
```

```
set.seed(123)
fit2 = jags.model(textConnection(model_stock_2),
                  data = data_jags,
                  inits = inits_2,
                  n.chains = 1,
                  n.adapt = nadapt)

update(fit2, nburn)  # Burn-in phase

fit2.samples = coda.samples(fit2,
                           c("beta0", "alpha", "beta1", "sigma_stock",
                             "sigma_sector"),
                           n.iter = niter)

summary(fit2.samples)
```

```
# Extract MCMC samples into matrix
mcmc_matrix <- as.matrix(fit2.samples)

# Randomly select alpha[k] and beta1[j]
set.seed(123)
alpha_ids <- sample(1:5, 1)     # randomly pick 1 sectors
beta1_ids <- sample(1:50, 1)    # randomly pick 1 stocks

# Choose key parameters to visualize
params <- c("beta0",
            paste0("alpha[", alpha_ids, "]"),
            paste0("beta1[", beta1_ids, "]")
            , "sigma_stock", "sigma_sector")
```

```r
# Set up plotting layout: 5 rows, 3 columns (each parameter gets trace + density
# + autocorrelation)
par(mfrow = c(5, 3))  # 3 plots per row

# Loop through parameters and plot
for (param in params) {
  traceplot(fit2.samples[, param],
            main = paste("Traceplot of", param),
            col = "blue", lwd = 1)

  plot(density(mcmc_matrix[, param]),
       main = paste("Posterior of", param),
       col = "red", lwd = 2)

  acf(mcmc_matrix[, param],
      main = paste("ACF of", param),
      col = "darkgreen", lwd = 2)
}
```

```r
# Extract based on model summary order
sector_probs <- invlogit(posterior_means[1:5])   # alpha[1] to alpha[5]
market_prob <- invlogit(posterior_means[6])      # beta0
stock_probs <- invlogit(posterior_means[7:56])   # beta1[1] to beta1[50]

# Create a dataframe
stock_sector <- rep(1:5, each = 10)
stock_number <- rep(1:10, 5)

stock_df <- data.frame(
  Sector = factor(stock_sector),
  Stock = factor(stock_number),
  Probability = stock_probs
)

# Add a "fake" Stock 11 column = Sector Mean
sector_mean_df <- data.frame(
  Sector = factor(1:5),
  Stock = factor(11),    # new column after stock 10
  Probability = sector_probs
)

# Combine stock_df and sector_mean_df
full_df <- bind_rows(stock_df, sector_mean_df)

# Reverse sector order
full_df$Sector <- factor(full_df$Sector, levels = 5:1)

# Make the heatmap
ggplot(full_df, aes(x = Stock, y = Sector, fill = Probability)) +
  geom_tile(color = "white") +
  geom_text(aes(label = sprintf("%.2f", Probability)), size = 3) +  # Label each tile
  scale_fill_gradient2(
    midpoint = market_prob,
    low = "darkred", mid = "white", high = "darkgreen",
    limits = c(min(stock_probs), max(stock_probs)),
    name = "Posterior \n Flip Probability"
  ) +
  labs(title = "Posterior Stock Flip Probabilities by Sector",
       subtitle = paste0("Market average probability: ", round(market_prob, 3)),
       x = "Stock", y = "Sector") +
  scale_x_discrete(labels = c(1:10, "Sector Mean")) +  # Rename Stock 11
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face = "bold"),
    plot.subtitle = element_text(hjust = 0.5, size = 10),
    axis.text = element_text(size = 10),
    axis.title = element_text(size = 12)
  )
```