Bayes Factor Design Analysis for the Cover Stories Effects in the Illusion of Causality

Bayes Factor Design Analysis for the Cover Stories Effects in the Illusion of Causality

Table of contents

Bayes Factor Design Analysis for the Cover Stories Effects in the Illusion of Causality

1. Aim

We aim to conduct a design analysis to determine the necessary sample size for testing whether the cover story used in the Contingency Learning Task (CLT) can provoke variations in the illusion of causality.

Traditionally, the illusion of causality has been assessed using a Likert-type scale ranging from 0 to 100. However, in our study, we plan to measure causal judgments using a binary (Y/N) response format. To inform our analysis, we will first dichotomize the continuous causal judgment data from Dalla Bona et al. (2025), under the assumption that the 0–100 scale reflects implicit causal beliefs.

Typically, the illusion of causality is assessed using the "medicine task", which likely involves participants' prior knowledge about the effectiveness of medical treatments and presents a clearly probabilistic scenario. To isolate the influence of such factors, we propose two experimental conditions: (1) one that removes prior knowledge using an "alien on a planet" paradigm, and (2) another that removes the probabilistic component using a deterministic "lever" paradigm.

To test our hypotheses, we will conduct two independent A/B Bayesian tests. For the combination of the two tests, we will perform a Bayes Factor Design Analysis (BFDA) to determine the appropriate sample size. This approach is well outlined in two key articles by Schönbrodt and Wagenmakers (2016) and Stefan et al. (2017), which provide a comprehensive tutorial on BFDA.

2. Dichotomizing Causality Judgments

In previous studies, such as Dalla Bona et al. (2025), causal judgments were measured using a continuous Likert-type scale ranging from 0 to 100. For the purposes of our analysis and experimental design, where responses will be recorded in a binary (Y/N) format, we need to convert these continuous judgments from Dalla Bona et al. (2025) into a dichotomous variable.

To do this, we adopt a threshold-based approach: responses equal to or greater than 50 are

interpreted as indicating a belief in a causal relationship (“yes”), while responses below 50 indicate the absence of such a belief (“no”). This threshold is consistent with the midpoint of the scale and aligns with previous interpretations of causality strength.

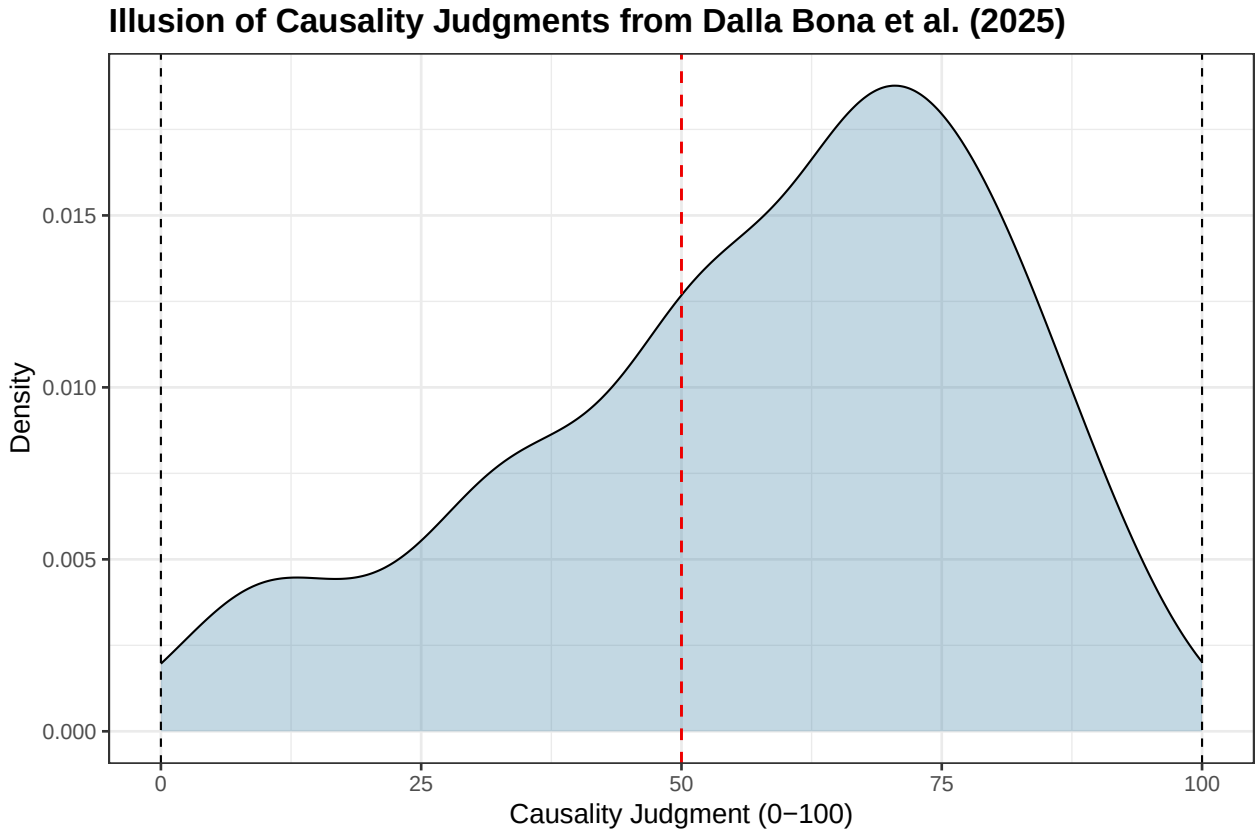
We find that approximately 70% of participants made a causal judgment (i.e., scored 50 on the 0–100 scale) in the medicine condition. This value will be used as the expected baseline proportion for the medicine condition in our design analysis.

```
# Load the .csv with the data and visualization
causality <- read.csv("Causalityjudgment.csv")


# General summary of the variable
summary(causality$x)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
   0.00   43.00   63.50   57.69   75.00   97.00 
```

```
# Plot
library(ggplot2)
library(viridisLite)
mako_color <- mako(5)[3]


ggplot(causality, aes(y = x)) +
  geom_density(fill= mako_color, alpha = 0.3) +
  coord_flip() +
  geom_hline(yintercept = 50, color = "red2", linetype = "dashed",
             lwd=.7) +
   theme_bw(base_size = 14) +
  geom_hline(yintercept = 0, linetype = "dashed") +
```

```
  geom_hline(yintercept = 100, linetype = "dashed") +
  theme(
    plot.title = element_text(face = "bold"),
    legend.position = "right"
  ) +
  labs(title =
  "Illusion of Causality Judgments from Dalla Bona et al. (2025)",
  y = "Causality Judgment (0-100)",
       x = "Density")
```




```
# Dichotomizing judgments
causality$binary_judgment <- ifelse(causality$x >= 50, 1, 0)


# Estimated proportion of 'yes' responses (x >= 50)
```

```r
mean(causality$binary_judgment == 1)
```

```
[1] 0.7227273
```

```r
# Standard deviation
sd(causality$binary_judgment)
```

```
[1] 0.4486732
```

3. Bayesian Logit Transformation Testing (LTT)

This model assumes that y_A and y_B follow binomial distributions with success probabilities θ_A and θ_B, respectively:

$$y_A \sim \text{Binomial}(n_A, \theta_A)$$

$$y_B \sim \text{Binomial}(n_B, \theta_B)$$

The success probabilities are modeled as a function of two parameters, γ and ψ, through a logit transformation:

$$\log\left(\frac{\theta_A}{1-\theta_A}\right) = \gamma - \frac{\psi}{2}$$

$$\log\left(\frac{\theta_B}{1-\theta_B}\right) = \gamma + \frac{\psi}{2}$$

Here, γ represents the grand mean of the log odds across the two conditions, and ψ denotes the log odds ratio, that is, the difference in log odds between condition B and condition A. These two parameters are assigned with normal priors by the function we will use.

4. Design (Point) Priors

As a heuristic criterion for interpreting differences between proportions, we consider a reduction in causal judgments to be meaningful when the expected proportion falls below 0.5. This threshold indicates increased skepticism among participants under the modified cover stories.

Specifically, we hypothesize that the deterministic cover story (lever condition) will lead to a substantial reduction in causal judgments, with an expected absolute difference of 0.30 compared to the baseline. The alien cover story, which removes prior knowledge, is expected to produce a moderate-to-strong reduction, with an expected absolute difference of 0.25. Accordingly, our alternative hypotheses for the two comparisons are defined as:

$$\mathcal{H}_{-} : \theta_A > \theta_B$$

These expected point estimate differences (i.e., $|\theta_A - \theta_B|$) will serve as point-targeted design priors for BFDA, functioning as the key simulation parameters.

5. Analysis Priors

5.1 Weakly Informative Analysis Priors

To assess the expected effects of the alien and lever cover stories on participants' causal judgments, we simulate binary responses based on hypothesized proportions. We set the baseline medicine condition proportion to $p_{\text{medicine}} = 0.7$ and model reductions in the alien and lever conditions by subtracting the expected effect sizes. Each condition receives an equal number of participants.

Using these simulated data, we run two independent Bayesian A/B tests comparing (i) Alien condition vs. Medicine condition and (ii) Lever condition vs. Medicine condition

We specify normal priors centered at zero for the effect size (ψ) and the grand mean (β) parameters with standard deviation 1, reflecting weakly informative priors. Prior model probabilities are equally split between the null hypothesis and the positive directional hypothesis.

```r
# Simulation parameters
p_medicine <- 0.7
p_alien <- p_medicine - 0.25
p_lever <- p_medicine - 0.3

# Number of participants
n_group <- 50

# Seed setted at current year and month
set.seed(202509)

# Binary responses
response_medicine <- rbinom(n_group, size = 1, prob = p_medicine)
response_alien    <- rbinom(n_group, size = 1, prob = p_alien)
response_lever    <- rbinom(n_group, size = 1, prob = p_lever)

# Alien vs Medicine
data_ab_alien <- list(
  y1 = sum(response_medicine), n1 = n_group,
  y2 = sum(response_alien), n2 = n_group
)

# Lever vs Medicine
data_ab_lever <- list(
  y1 = sum(response_medicine), n1 = n_group,
  y2 = sum(response_lever), n2 = n_group
)
```

```r
# Prior model probabilities
prior_prob <- c(H1 = 0, Hplus = 0, Hmin = 0.5, H0 = 0.5)
names(prior_prob) <- c("H1", "H+", "H-", "H0")

# Bayesian A/B test Alien vs Medicine
library(abtest)
bf_alien <- ab_test(
  data_ab_alien,
  prior_par = list(mu_psi = 0, sigma_psi = 1, mu_beta = 0, sigma_beta = 1),
  prior_prob = prior_prob
)
bf_alien
```

```
Bayesian A/B Test Results:

 Bayes Factors:

 BF10: 4.699165
 BF+0: 0.1136948
 BF-0: 9.277314

 Prior Probabilities Hypotheses:

 H-: 0.5
 H0: 0.5

 Posterior Probabilities Hypotheses:
```

```
 H-: 0.9027
 H0: 0.0973
```

```
# Bayesian A/B test Lever vs Medicine
bf_lever <- ab_test(
  data_ab_lever,
  prior_par = list(mu_psi = 0, sigma_psi = 1, mu_beta = 0, sigma_beta = 1),
  prior_prob = prior_prob
)
bf_lever
```

```
Bayesian A/B Test Results:

 Bayes Factors:

 BF10: 20.15016
 BF+0: 0.09686739
 BF-0: 40.89489

 Prior Probabilities Hypotheses:

 H-: 0.5
 H0: 0.5

 Posterior Probabilities Hypotheses:

 H-: 0.9761
```

```
 H0: 0.0239
```

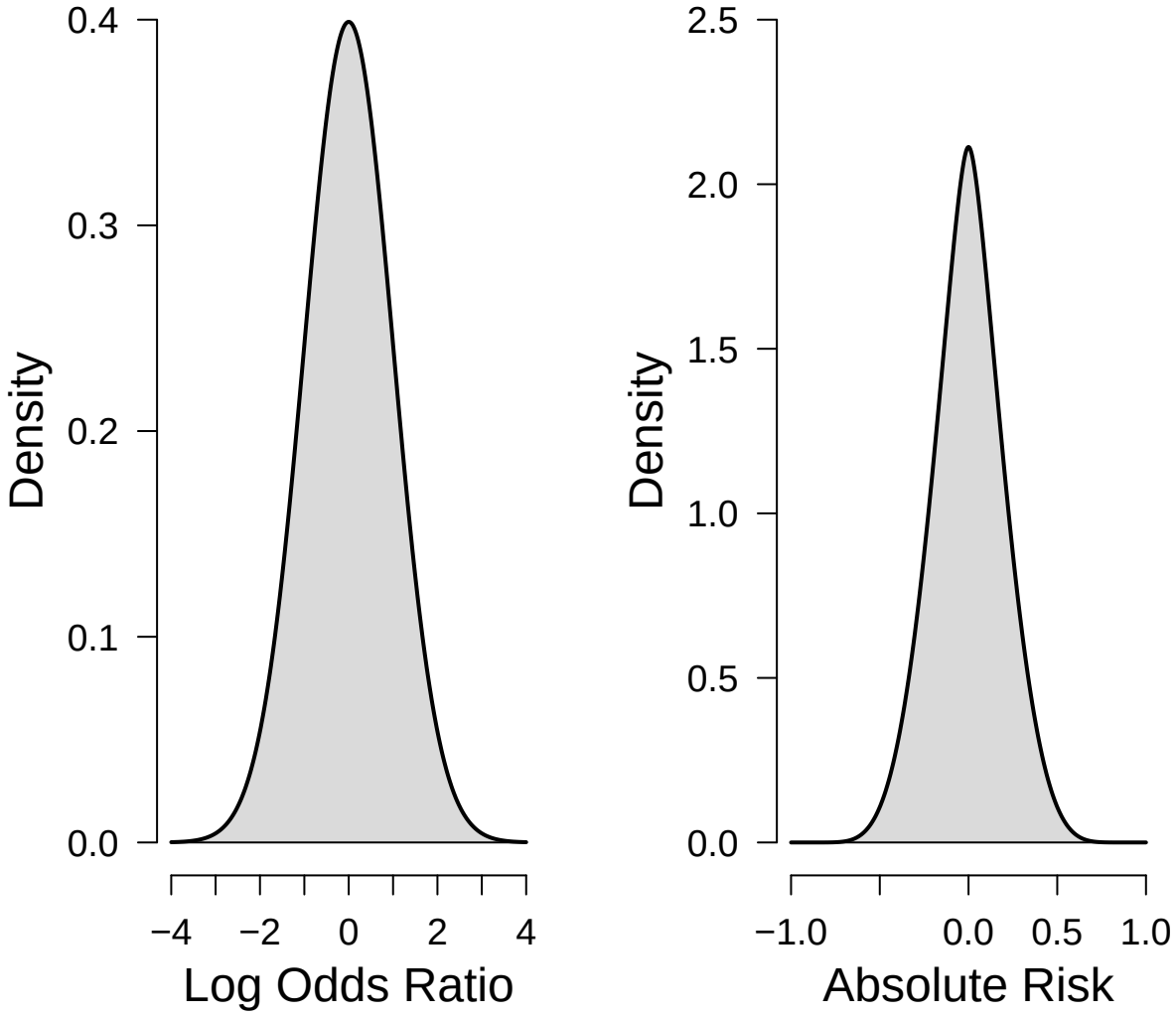
5.2 Analysis Priors

We visualized the analysis priors parameters and their corresponding plots, for each condition.

```r
# Analysis prior for the lever condition effect size
prior_par_lever <- elicit_prior(
  q = c(-1, 0, 1),
  prob = c(0.1, 0.5, 0.9),
  what = "arisk"
)


# Parameters
print(prior_par_lever)
```

```
$mu_psi
[1] 0


$sigma_psi
[1] 1


$mu_beta
[1] 0


$sigma_beta
[1] 1
```

```r
# Plot
par(mfrow=c(1,2))
plot_prior(prior_par_lever); plot_prior(prior_par = prior_par_lever,
                                        what = "arisk")
```

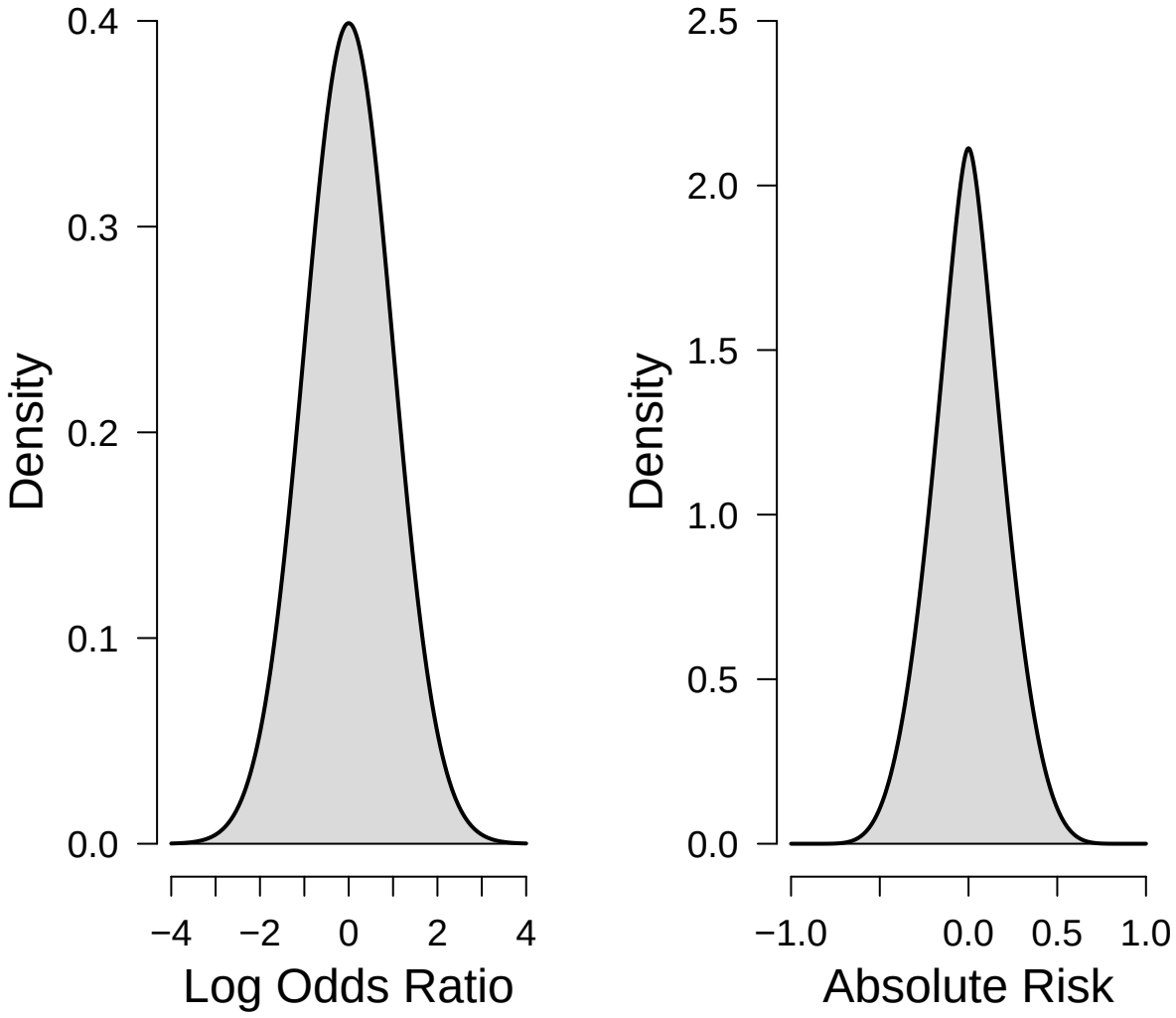



```r
# Analysis prior for the alien condition effect size
prior_par_alien <- elicit_prior(
  q = c(-1, 0, 1),
  prob = c(0.1, 0.5, 0.9),
  what = "arisk"
)
```

```r
# Parameters
print(prior_par_alien)
```

```
$mu_psi
[1] 0

$sigma_psi
[1] 1

$mu_beta
[1] 0

$sigma_beta
[1] 1
```

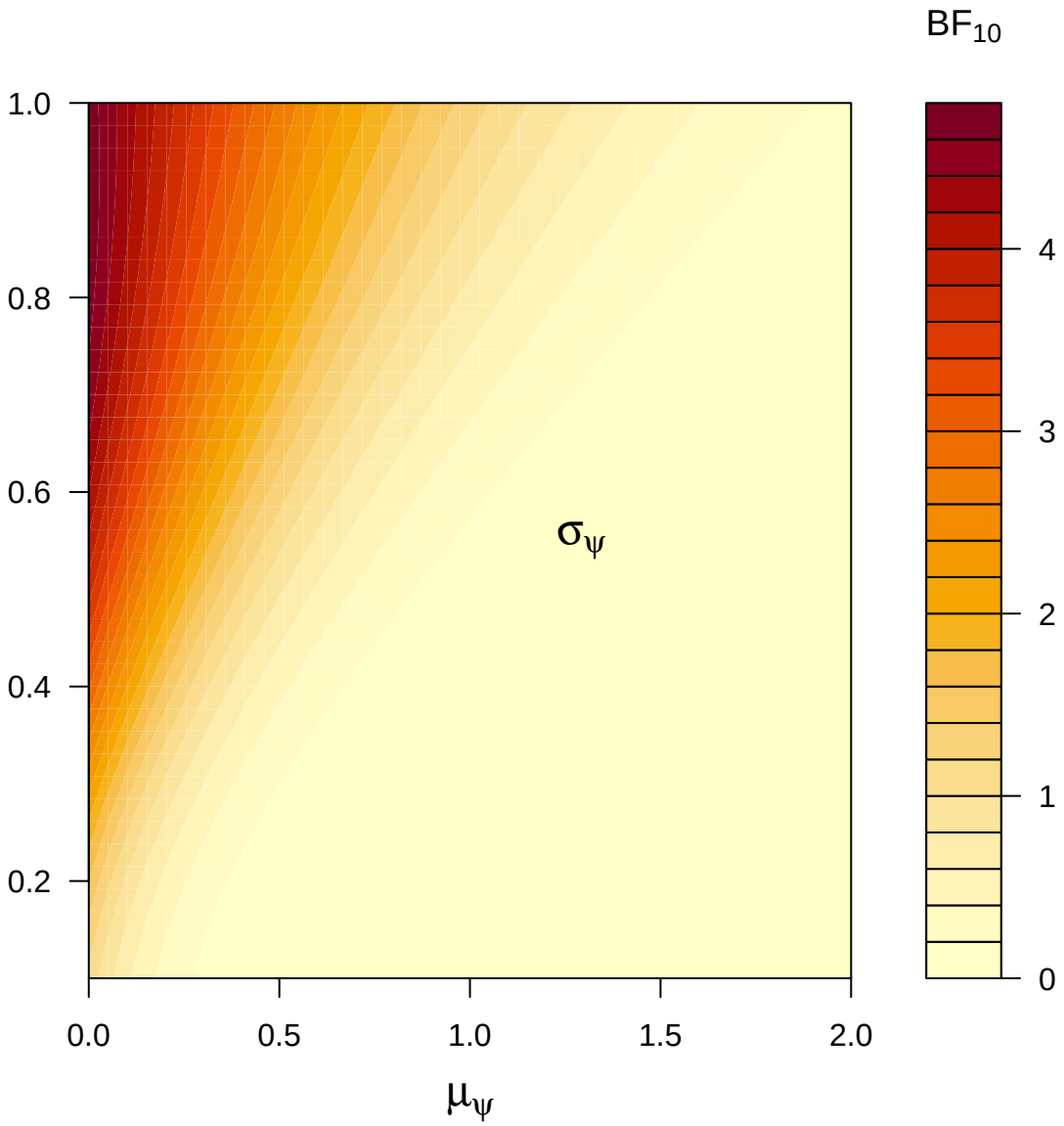
```r
# Plot
par(mfrow=c(1,2))
plot_prior(prior_par_alien); plot_prior(prior_par = prior_par_alien,
                                        what = "arisk")
```



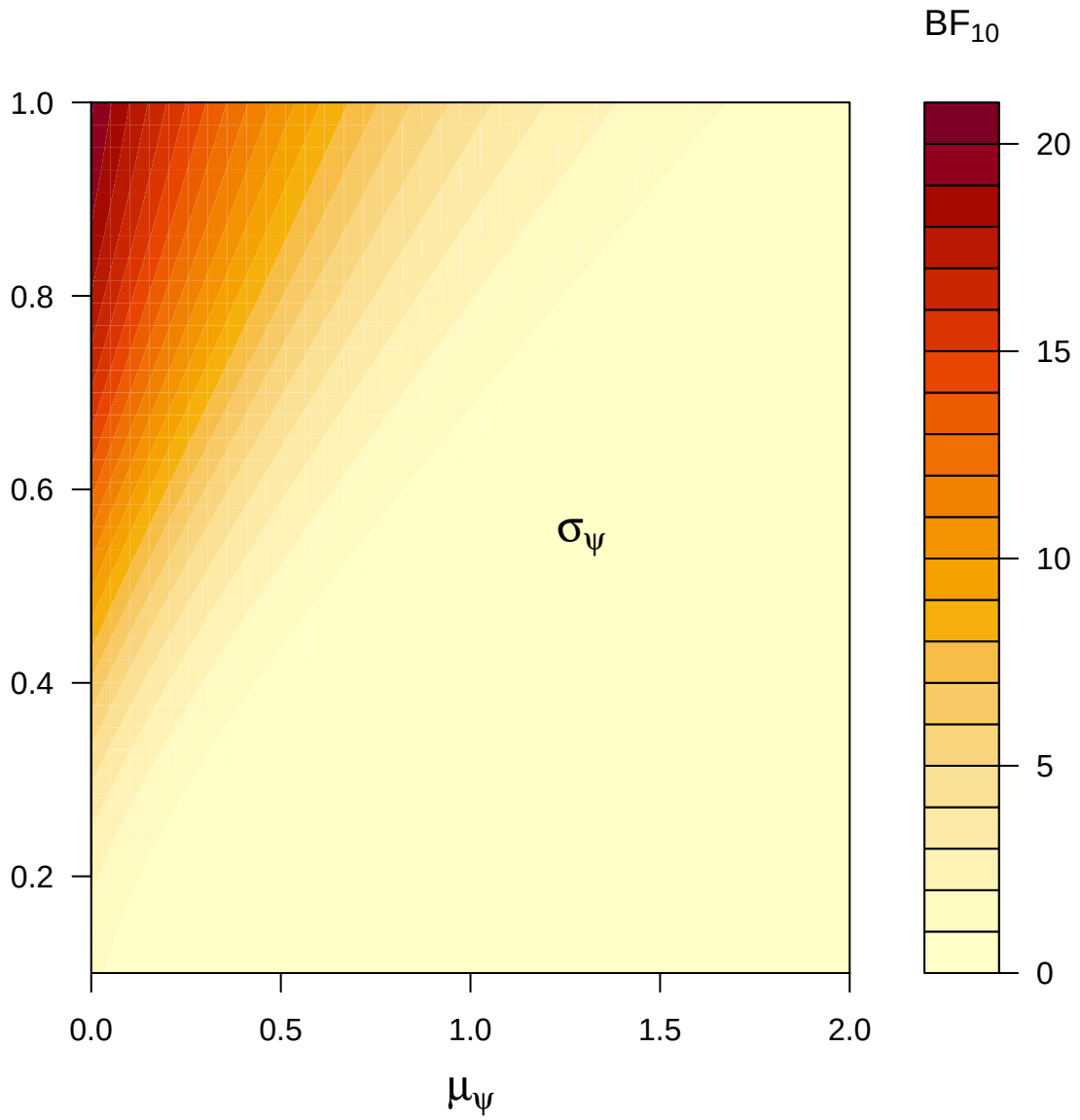
5.3 Bayes Factor robustness check

To evaluate the sensitivity of our Bayesian A/B test results to prior assumptions, we visualize the robustness of the Bayes factors across a range of prior mean and standard deviation values.

```
# Robustness plot Alien cover story
par(mfrow=c(1,2))
plot_robustness(bf_alien, mu_range = c(0,2),
                sigma_range = c(0.1,1))
```




```
# Robustness plot Lever cover story
plot_robustness(bf_lever, mu_range = c(0,2),
                sigma_range = c(0.1,1))
```



6. Simulating Under Different Scenarios

6.1 Inferential Risks for One Comparison

We simulate the statistical power and false positive rate for a single comparison. The simulations vary the sample size per group and the true effect size (difference in proportions).

```
# # Parameters
# sample_sizes <- c(150, 165, 180, 195, 210, 225)
# nsim <- 2000
# bf_threshold <- 3.5
```

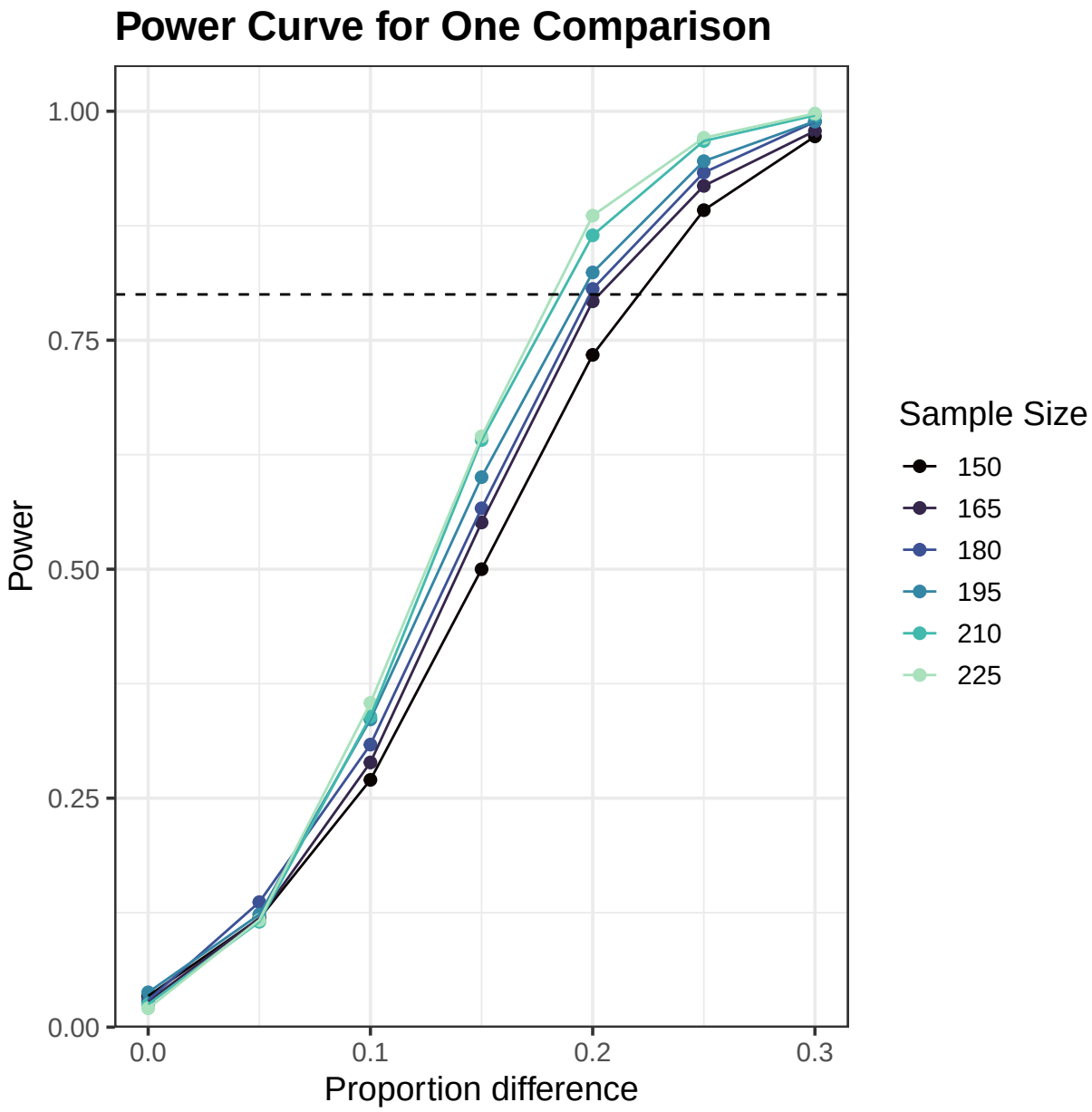
```
# effect_sizes <- seq(0, 0.3, by = 0.05)
#
# # Initialize results
# powerone <- data.frame(
#   sample_size = integer(),
#   effect_alien = numeric(),
#   power = numeric()
# )
#
# sim_counter <- 1
#
# for (n in sample_sizes) {
#   n_group <- as.integer(n / 2)
#
#   for (eff_alien in effect_sizes) {
#     detections <- 0
#
#     for (i in 1:nsim) {
#       # Set probabilities
#       p_medicine <- 0.7
#       p_alien <- p_medicine - eff_alien
#
#       # Responses
#       response_medicine <- rbinom(n_group, 1, p_medicine)
#       response_alien    <- rbinom(n_group, 1, p_alien)
#
#       data_ab_alien <- list(
```

```
#         y1 = sum(response_medicine), n1 = n_group,
#         y2 = sum(response_alien), n2 = n_group
#       )
#
#       # Priors
#       prior_prob <- c(H1 = 0, Hplus = 0, Hmin = 0.5, H0 = 0.5)
#       names(prior_prob) <- c("H1", "H+", "H-", "H0")
#
#       # Bayesian A/B test
#       bf_alien <- ab_test(
#         data_ab_alien,
#         prior_par = list(mu_psi = 0,
#                          sigma_psi = 1,
#                          mu_beta = 0, sigma_beta = 1),
#         prior_prob = prior_prob
#       )
#
#       #  BF
#       bf_a <- if (
#         !is.null(bf_alien$bf$bfminus0))
#         bf_alien$bf$bfminus0 else NA_real_
#
#       if (!is.na(bf_a) && bf_a > bf_threshold) {
#         detections <- detections + 1
#       }
#
#       sim_counter <- sim_counter + 1
```

```
#     }
#
#     power <- detections / nsim
#     powerone <- rbind(powerone, data.frame(
#       sample_size = n,
#       effect_alien = eff_alien,
#       power = power
#     ))
#
#     print(paste("N:", n,
#                 "Effect alien:", eff_alien,
#                 "Power:", round(power, 3)))
#   }
# }

# Save results
# save(powerone, file = "power_one_only.Rda")
```

```
# Load data and plot
load(file = "power_one_only.Rda")


ggplot(powerone, aes(x = effect_alien, y = power,
                     color = factor(sample_size))) +
  geom_point(size = 2) +
  geom_line()+
  scale_color_viridis_d(option="mako", end = .9)+
  scale_y_continuous(limits = c(0, 1.05), expand = c(0, 0)) +
```

```
  labs(
    title = "Power Curve for One Comparison",
    x = "Proportion difference",
    y = "Power",
    color = "Sample Size"
  ) +
  geom_hline(yintercept = .8, lty = "dashed")+
  theme_bw(base_size = 14) +
  theme(
    plot.title = element_text(face = "bold"),
    legend.position = "right"
  )
```




```
# Visualize power values for the Medicine vs Lever
rbind(
round(powerone$sample_size[powerone$effect_alien==.3],0),
powerone$power[powerone$effect_alien==.3]
)
```

```
          [,1]     [,2]     [,3]    [,4]     [,5]     [,6]
[1,] 150.0000 165.0000 180.0000 195.000 210.0000 225.0000
```

```
[2,]   0.9725   0.9785   0.9885   0.989   0.9955   0.9975
```

```r
# Visualize power values for the Medicine vs Alien
rbind(
round(powerone$sample_size[powerone$effect_alien==.3],0),
powerone$power[powerone$effect_alien==.25]
)
```

```
        [,1]     [,2]    [,3]     [,4]     [,5]    [,6]
[1,] 150.000 165.0000 180.000 195.0000 210.0000 225.000
[2,]   0.892   0.9185   0.933   0.9455   0.9675   0.971
```

6.2 Inferential Risks for Two Comparisons

When conducting two comparisons (i.e., Alien vs. Medicine and Lever vs. Medicine) it becomes essential to consider the risk of false evidence. In particular, we are interested in the probability of detecting one or both effects under different combinations of true and null hypotheses.

Assuming equal power and independence, if both null hypotheses are true, the probability of falsely rejecting at least one of them is given by:

$$P(\text{Reject at least one null} \mid \text{both nulls true}) = 1 - (1 - \alpha)^2$$

When one hypothesis is true and the other is null, the probability of simultaneously observing a true positive and a false positive can be expressed as:

$$P(\text{One true positive and one false positive}) = 2 \times \text{power} \times \alpha$$

When both hypotheses are true, and the directional alternative hypothesis holds in each comparison, the probability that both effects are detected is:

$$P(\text{Both detected}) = \text{power}^2$$

These expressions are based on the assumption that the two tests are independent and equally powered. However, these assumptions do not hold. Both the Alien and Lever conditions are compared against the same control group (Medicine), meaning the test statistics are not independent. The shared baseline introduces statistical dependence that can affect the joint probabilities of observing various inference patterns. To obtain more realistic estimates of inferential risks, we rely on simulation.

6.3 Joint Power

To estimate the target sample size of our experimental design, we conducted a simulation-based power analysis using BFDA. The goal is to compute the joint detection power, defined as the proportion of simulations in which both comparisons (i.e., alien vs. medicine and lever vs. medicine) yield a Bayes Factor in favor of the alternative hypothesis above a detection threshold (set at BF > 3.5).

We simulate response data across a range of sample sizes and effect sizes. The expected effects are modeled as differences in success probabilities compared to the baseline medicine condition (with $p = 0.7$). For each combination of sample size and effect sizes, 2000 simulations are performed. We use weakly informative priors as defined above. A power of .8 is achieved at 195 participants (65 participants per group).

```
#Parameters
# sample_sizes <- seq(150, 225, by = 15)
# nsim <- 2000
# bf_threshold <- 3.5
# effect_sizes <- seq(0, 0.3, by = 0.05)
#
# power_results <- data.frame(
```

```
#   sample_size = integer(),
#   effect_alien = numeric(),
#   effect_lever = numeric(),
#   power = numeric()
# )
#
# bf_log <- data.frame(
#   sim_id = integer(),
#   sample_size = integer(),
#   effect_alien = numeric(),
#   effect_lever = numeric(),
#   bf_plus0_alien = numeric(),
#   bf_plus0_lever = numeric()
# )
#
# sim_counter <- 1
#
# for (n in sample_sizes) {
#   n_group <- as.integer(n / 3)
#
#   for (eff_alien in effect_sizes) {
#     for (eff_lever in effect_sizes) {
#       detections <- 0
#
#       for (i in 1:nsim) {
#         # Scenarios probabilities
#         p_medicine <- 0.7
```

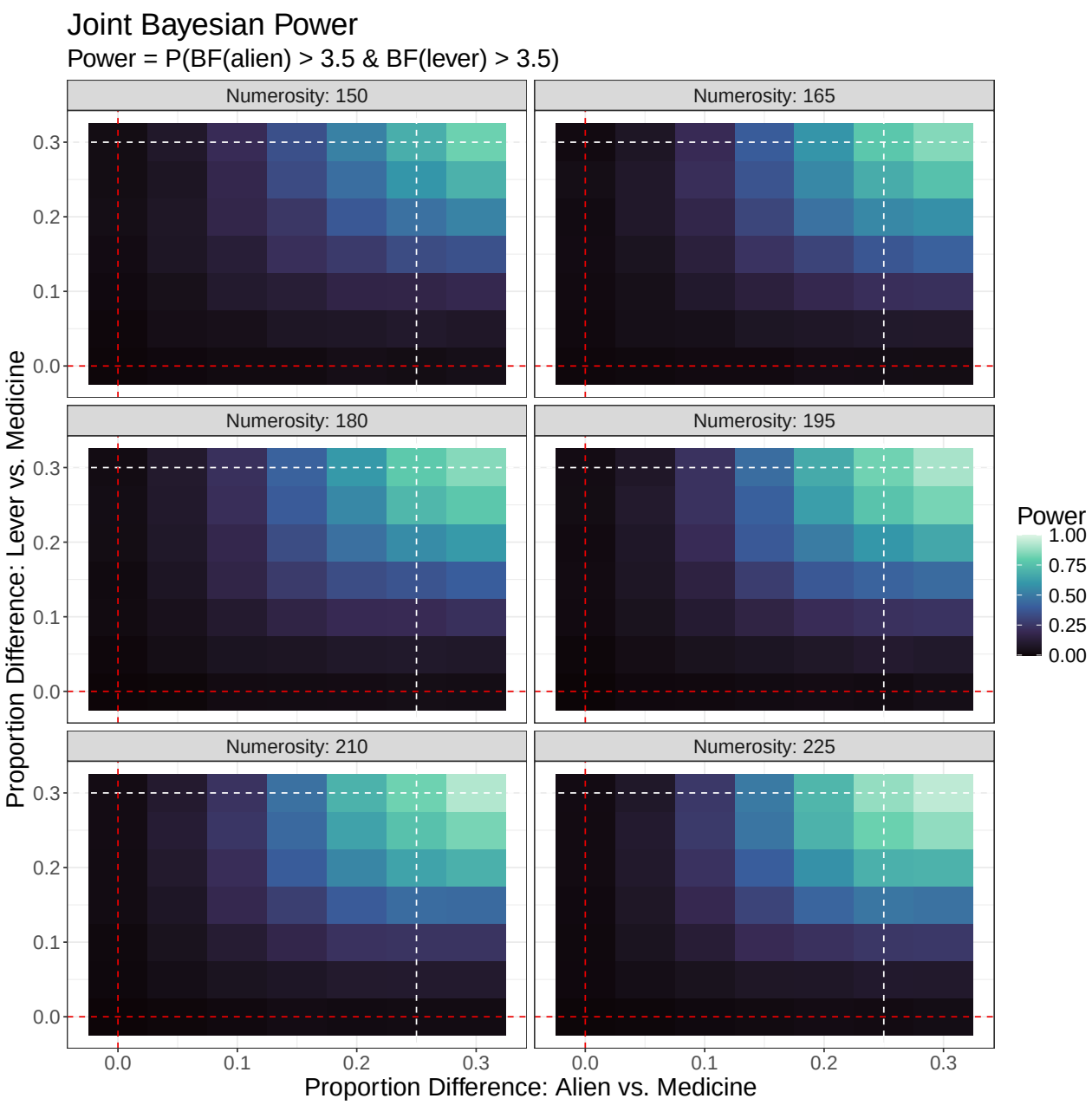
```
#         p_alien <- p_medicine - eff_alien
#         p_lever <- p_medicine - eff_lever
#
#         # Responses
#         response_medicine <- rbinom(n_group, 1, p_medicine)
#         response_alien    <- rbinom(n_group, 1, p_alien)
#         response_lever    <- rbinom(n_group, 1, p_lever)
#
#         data_ab_alien <- list(
#           y1 = sum(response_medicine), n1 = n_group,
#           y2 = sum(response_alien), n2 = n_group
#         )
#         data_ab_lever <- list(
#           y1 = sum(response_medicine), n1 = n_group,
#           y2 = sum(response_lever), n2 = n_group
#         )
#
#         # Prior probabilities
#         prior_prob <- c(H1 = 0, Hplus = 0, Hmin = 0.5, H0 = 0.5)
#         names(prior_prob) <- c("H1", "H+", "H-", "H0")
#
#         # Run Bayesian A/B tests
#         bf_alien <- ab_test(
#           data_ab_alien,
#           prior_par = list(mu_psi = 0,
#                            sigma_psi = 1,
#                            mu_beta = 0, sigma_beta = 1),
```

```
#                             prior_prob = prior_prob)
#
#         bf_lever <- ab_test(
#           data_ab_lever,
#           prior_par = list(mu_psi = 0,
#                            sigma_psi = 1,
#                            mu_beta = 0, sigma_beta = 1),
#                             prior_prob = prior_prob)
#
#         # BFs
#   bf_a <- if (!is.null(
#   bf_alien$bf$bfminus0)) bf_alien$bf$bfminus0 else NA_real_
#   bf_l <- if (!is.null(
#   bf_lever$bf$bfminus0)) bf_lever$bf$bfminus0 else NA_real_
#
#         # Log BFs
#         bf_log <- rbind(bf_log, data.frame(
#           sim_id = sim_counter,
#           sample_size = n,
#           effect_alien = eff_alien,
#           effect_lever = eff_lever,
#           bf_plus0_alien = bf_a,
#           bf_plus0_lever = bf_l
#         ))
#         sim_counter <- sim_counter + 1
#
#         # Joint detections
```

```
#         if (!is.na(bf_a) && !is.na(bf_l) &&
#             bf_a > bf_threshold && bf_l > bf_threshold) {
#           detections <- detections + 1
#         }
#       }
#
#       # Power result
#       power <- detections / nsim
#       power_results <- rbind(power_results, data.frame(
#         sample_size = n,
#         effect_alien = eff_alien,
#         effect_lever = eff_lever,
#         power = power
#       ))
#
#       print(paste("N:", n,
#                   "Effect alien:", eff_alien,
#                   "Effect lever:", eff_lever,
#                   "Joint Power:", round(power, 2)))
#     }
#   }
# }
# save(power_results, file = "Presults.Rda")
```

```
# Load data and plot
load("Presults.Rda")
names(power_results)[1] <- "Numerosity"
```

```r
ggplot(power_results, aes(x = effect_alien,
                          y = effect_lever, fill = power)) +
  geom_tile() +
  geom_vline(xintercept = 0.25, color = "white", linetype = "dashed") +
  geom_hline(yintercept = 0.3, color = "white", linetype = "dashed") +
  geom_vline(xintercept = 0, color = "red2", linetype = "dashed") +
  geom_hline(yintercept = 0, color = "red2", linetype = "dashed") +
  scale_fill_viridis_c(limits = c(0, 1),
                       name = "Power", option = "mako") +
  facet_wrap(~ Numerosity, ncol = 2, labeller = label_both) +
  labs(
    title = "Joint Bayesian Power",
    subtitle = "Power = P(BF(alien) > 3.5 & BF(lever) > 3.5)",
    x = "Proportion Difference: Alien vs. Medicine",
    y = "Proportion Difference: Lever vs. Medicine"
  ) +
  theme_bw() +
  theme(text = element_text(size = 18))
```




```
# Visualize power values
rbind(
  power_results$Numerosity[power_results$effect_alien==.25 &
                      power_results$effect_lever==.30],
power_results$power[power_results$effect_alien==.25 &
                      power_results$effect_lever==.30]
```

```
)
```

```
        [,1]    [,2]     [,3]     [,4]    [,5]    [,6]
[1,] 150.000 165.000 180.0000 195.0000 210.000 225.000
[2,]   0.684   0.774   0.7825   0.8215   0.819   0.875
```

6.4 One True Positive / One False Positive

This section aims to estimate the probability of simultaneously observing one true positive and one false positive across the two comparisons. Specifically, we are interested in cases where one of the cover story effects is real and detected, while the other is null but still falsely detected. To estimate this, we simulated across a range of sample sizes and effect sizes. Across all conditions, the probability of simultaneously detecting both effects when only one is true varied between approximately 0.02 and 0.03.

At a total sample size of 195 participants, when the Lever effect is real and the Alien effect is null, the probability of observing both tests in favor of the directional alternative hypothesis is approximately 0.03. Conversely, when the Alien effect is real and the Lever effect is null, the probability of simultaneous detection is approximately 0.02. Assuming that either condition could be the true effect, and that these two scenarios are mutually exclusive, the combined probability of obtaining one true positive and one false positive is 0.05.

```
# Load data and plot
load("mix.rda")


ggplot(mixed_results, aes(x = factor(effect_alien), y = joint_detection, fill = factor(s
  geom_col(position = "dodge") +
  labs(
    title = "Joint Detection Rate (One True, One False)",
    x = "True Proportion Difference Size (Alien - Medicine)",
```

```
    y = "Probability Both Effects Detected",
    fill = "Sample Size"
  ) +
  scale_y_continuous(limits = c(0, 0.05), expand = c(0, 0)) +
  scale_fill_viridis_d(option = "mako", end = 0.9) +
  theme_bw(base_size = 14) +
  geom_hline(yintercept = 0.025, linetype = "dashed") +
  theme(
    plot.title = element_text(face = "bold"),
    legend.position = "right"
  )
```


```
# Prob. Alien true and Lever false

mixed_results$joint_detection[mixed_results$sample_size==195 &

                              mixed_results$effect_alien ==.25]
```

```
[1] 0.02
```

```
# Prob. Alien false and Lever true
mixed_results$joint_detection[mixed_results$sample_size==195 &
                              mixed_results$effect_alien ==.30]
```

```
[1] 0.03
```

```
# Combining
mixed_results$joint_detection[mixed_results$sample_size==195 &
                              mixed_results$effect_alien ==.25] +
  mixed_results$joint_detection[mixed_results$sample_size==195 &
                              mixed_results$effect_alien ==.30]
```

```
[1] 0.05
```

6.5 Joint False Positive Rate

In this section, we estimate the joint false positive rate (FPR) for our Bayesian A/B testing design. Results from the simulation show that, for a target sample size of 195 participants, the joint false positive rate is approximately 0.05.

```
#Parameters
# sample_sizes <- seq(150, 210, by = 15)
# nsim <- 2000
# bf_threshold <- 3.5
# effect_sizes <- 0
#
# power_resultsNULL <- data.frame(
#   sample_size = integer(),
#   effect_alien = numeric(),
#   effect_lever = numeric(),
```

```
#   power = numeric()
# )
#
# bf_log <- data.frame(
#   sim_id = integer(),
#   sample_size = integer(),
#   effect_alien = numeric(),
#   effect_lever = numeric(),
#   bf_plus0_alien = numeric(),
#   bf_plus0_lever = numeric()
# )
#
# sim_counter <- 1
#
# for (n in sample_sizes) {
#   n_group <- as.integer(n / 3)
#
#   for (eff_alien in effect_sizes) {
#     for (eff_lever in effect_sizes) {
#       detections <- 0
#
#       for (i in 1:nsim) {
#         # Scenarios probabilities
#         p_medicine <- 0.7
#         p_alien <- 0.7
#         p_lever <- 0.7
#
```

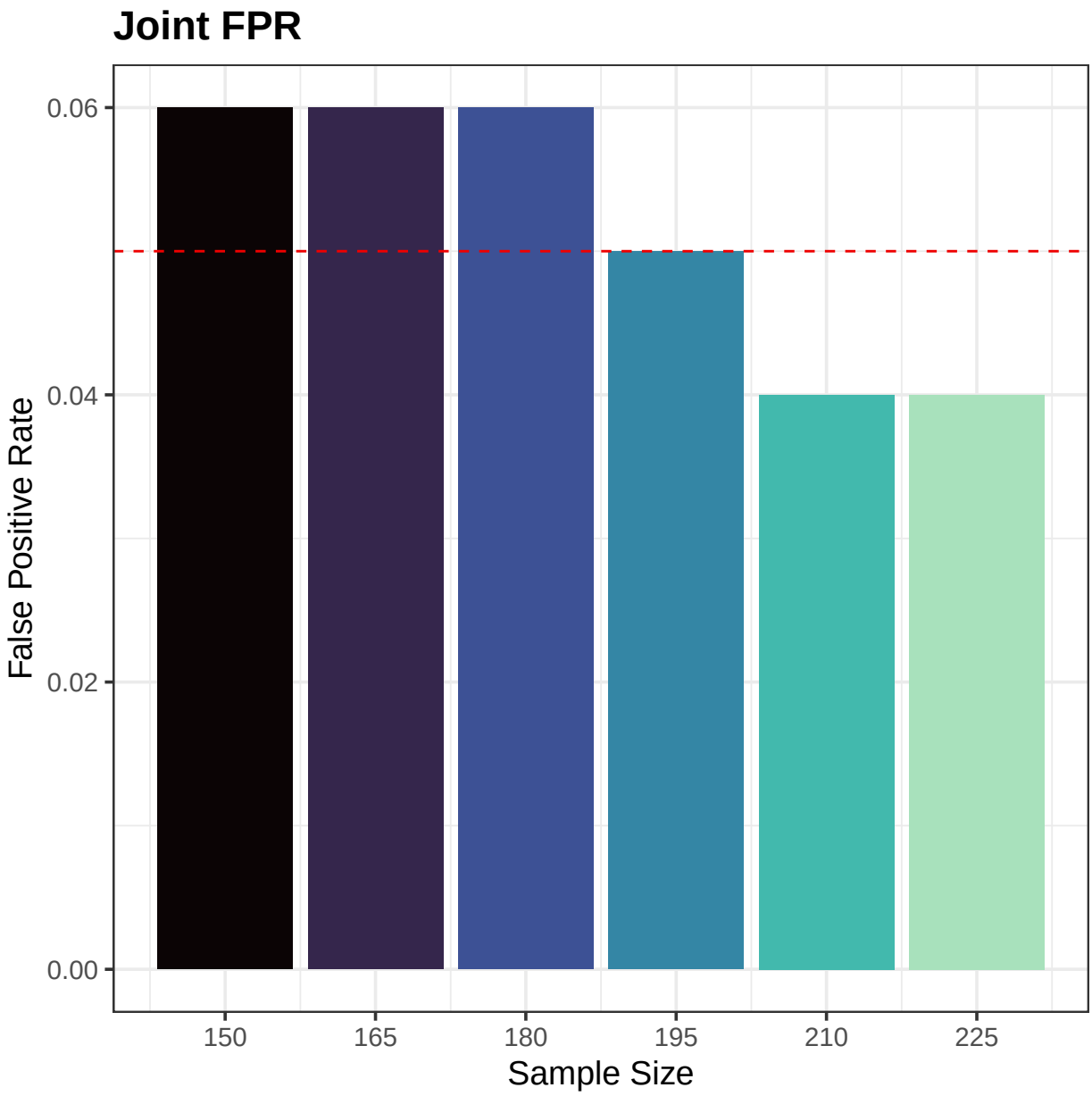
```
#         # Responses
#         response_medicine <- rbinom(n_group, 1, p_medicine)
#         response_alien    <- rbinom(n_group, 1, p_alien)
#         response_lever    <- rbinom(n_group, 1, p_lever)
#
#         data_ab_alien <- list(
#           y1 = sum(response_medicine), n1 = n_group,
#           y2 = sum(response_alien), n2 = n_group
#         )
#         data_ab_lever <- list(
#           y1 = sum(response_medicine), n1 = n_group,
#           y2 = sum(response_lever), n2 = n_group
#         )
#
#         # Prior probabilities
#         prior_prob <- c(H1 = 0, Hplus = 0, Hmin = 0.5, H0 = 0.5)
#         names(prior_prob) <- c("H1", "H+", "H-", "H0")
#
#         # Run Bayesian A/B tests
#         bf_alien <- ab_test(
#           data_ab_alien,
#           prior_par = list(mu_psi = 0,
#                            sigma_psi = 1,
#                            mu_beta = 0, sigma_beta = 1),
#                             prior_prob = prior_prob)
#
#         bf_lever <- ab_test(
```

```
#           data_ab_lever,
#           prior_par = list(mu_psi = 0,
#                            sigma_psi = 1,
#                            mu_beta = 0, sigma_beta = 1),
#                             prior_prob = prior_prob)
#
#         # BFs
#   bf_a <- if (!is.null(
#   bf_alien$bf$bfminus0)) bf_alien$bf$bfminus0 else NA_real_
#   bf_l <- if (!is.null(
#   bf_lever$bf$bfminus0)) bf_lever$bf$bfminus0 else NA_real_
#
#         # Log BFs
#         bf_log <- rbind(bf_log, data.frame(
#           sim_id = sim_counter,
#           sample_size = n,
#           effect_alien = eff_alien,
#           effect_lever = eff_lever,
#           bf_plus0_alien = bf_a,
#           bf_plus0_lever = bf_l
#         ))
#         sim_counter <- sim_counter + 1
#
#         # Joint detections
#         if (!is.na(bf_a) && !is.na(bf_l) &&
#             bf_a > bf_threshold || bf_l > bf_threshold) {
#           detections <- detections + 1
```

```
#         }
#       }
#
#       # Power result
#       power <- detections / nsim
#       power_resultsNULL <- rbind(power_resultsNULL, data.frame(
#         sample_size = n,
#         effect_alien = eff_alien,
#         effect_lever = eff_lever,
#         power = round(power,2)
#       ))
#
#
#     }
#   }
# }
# save(power_resultsNULL, file = "resultsNULL.Rda")
```

```
# Load data and plot
load("resultsNULL.Rda")


ggplot(power_resultsNULL, aes(x = sample_size,
                              y = power, fill = as.factor(sample_size))) +
  geom_col() +
  labs(
    title = "Joint FPR",
    x = "Sample Size",
```

```
    y = "False Positive Rate",
    color = "Sample Size"
  ) +
  scale_fill_viridis_d(option = "mako", end =.9) +
  scale_x_continuous(breaks = seq(150, 225, by = 15)) +
  theme_bw(base_size = 14) +
  geom_hline(yintercept = .05, lty = "dashed", color = "red2")+
  theme(
    plot.title = element_text(face = "bold"),
     legend.position = "none"
  )
```




```
# Visualize joint FPR

power_resultsNULL
```

```
  sample_size effect_alien effect_lever power
1         150            0            0  0.06
2         165            0            0  0.06
3         180            0            0  0.06
4         195            0            0  0.05
```

5	210	0	0	0.04
6	225	0	0	0.04

References

References marked with an asterisk indicate studies included in the meta-analysis.

Dalla Bona, S., Vicovaro, M., & Navarrete, E. (2025). *The foreign language effect on the illusion of causality: A replication attempt and an explorative analysis of the mechanisms*. https://doi.org/10.17605/OSF.IO/HVGKX.

Gronau, Q. F., Raj K. N., A., & Wagenmakers, E.-J. (2021). Informed Bayesian inference for the A/B test. *Journal of Statistical Software*, *100*(17), 1–39. https://doi.org/10.18637/jss.v100.i17

Masked Citation. (n.d.). *Masked Title*.

Schönbrodt, F. D., & Stefan, A. M. (2019). *BFDA: An r package for bayes factor design analysis* (Version 0.5.0).

Schönbrodt, F. D., & Wagenmakers, E. (2016). Bayes factor design analysis: Planning for compelling evidence. *Psychonomic Bulletin & Review*, *23*(2), 254–271. https://doi.org/10.3758/s13423-017-1230-y

Stan Development Team. (2024). *RStan: The R interface to Stan*. https://mc-stan.org/

Stefan, A. M., Gronau, Q. F., Schönbrodt, F. D., & Wagenmakers, E. (2017). A tutorial on bayes factor design analysis using an informed prior. *Behavior Research Methods*, *49*(2), 413–428. https://doi.org/10.3758/s13428-018-01189-8

Wickham, H. (2016). *ggplot2: Elegant graphics for data analysis*. Springer-Verlag New York.