Kasdin et al. (2025) show that dopamine in the brains of young zebra finches acts as a learning signal, increasing when they sing closer to their adult song and decreasing when they sing further away, effectively guiding their vocal development through trial-and-error. This suggests that complex natural behaviors, like learning to sing, are shaped by dopamine-driven reinforcement learning, similar to how artificial intelligence learns. You can find the paper at this link: https://www.nature.com/articles/s41586-025-08729-1..

Note they measure dopamine using fibre photometry, changes in the fluorescence indicate dopamine changes in realtime. Their specific measurement considers changes in flourescence in 100-ms windows between 200 and 300 ms from the start of singing, averaged across development.

1. Using the `pwr` package for `R` (Champely, 2020), conduct a power analysis. How many observations would the researchers need to detect a moderate-to-large effect ($d = 0.65$) when using $\alpha = 0.05$ and default power (0.80) for a two-sided one sample t test.

```
library(pwr)

power.result <- pwr.t.test(d = 0.65,
                           sig.level = 0.05,
                           power = 0.80,
                           type = "one.sample",
                           alternative = "two.sided")
print(power.result)

##
##      One-sample t test power calculation
##
##               n = 20.58039
##               d = 0.65
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
```

   In order to detect a moderate-to-large effect ($d = 0.65$) when using $\alpha = 0.05$ and default power (0.80) for a two-sided one sample t test, researchers should use a sample size of at least $n = 21$.

2. Click the link to go to the paper. Find the source data for Figure 2. Download the Excel file. Describe what you needed to do to collect the data for Figure 2(g). Note that you only need the `closer_vals` and `further_vals`. Ensure to `mutate()` the data to get a difference (e.g., `closer_vals - further_vals`).

   In order to collect the data for figure 2(g), I would use the `g.Farther_vals` (dopamine values when birds sang further away) and `g.Closer_vals` (dopamine values when birds sang closer to the adult song) tabs in the excel file, and save each sheet as an individual CSV file.

   Once both CSV files are opened in R, I would create a tibble with two columns, one that represents the values in the Farther data, and one that represents the values in the Closer data.

   Then, I would finally mutate the tibble and add a column that represents the difference between further and closer.

```
library(tidyverse)

closer <- read_csv("g.Closer_vals.csv")

## Rows:  24 Columns:  1
## -- Column specification --------------------------------------------------------
## Delimiter:  ","
## dbl (1):  0.275714146
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

further <- read_csv("g.Farther_vals.csv")

## Rows: 24 Columns: 1
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (1): -0.191685046
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

colnames(closer) <- "Value"
colnames(further) <- "Value"

data <- tibble(
  closer.vals = closer$Value,
  further.vals = further$Value
)

data <- data |>
  mutate(diff = closer.vals - further.vals)
```

3. Summarize the data.

   (a) Summarize the further data. Do the data suggest that dopamine in the brains of young zebra finches decreases when they sing further away?

```
further.summary <- data |>
  summarise(
    mean = mean(further.vals, na.rm = TRUE),
    sd = sd(further.vals, na.rm = TRUE),
    n = sum(!is.na(further.vals))
  )

further.summary
```

```
## # A tibble: 1 x 3
##     mean    sd     n
##    <dbl> <dbl> <int>
## 1 -0.203 0.133    24
```

   The mean is significantly less than 0, which supports the idea that dopamine decreases when birds sing further away.

   (b) Summarize the closer data. Do the data suggest that dopamine in the brains of young zebra finches increases when they sing closer to their adult song?

```
summary.closer <- data |>
  summarise(
    mean = mean(closer.vals, na.rm = TRUE),
    sd = sd(closer.vals, na.rm = TRUE),
    n = sum(!is.na(closer.vals))
```

```
  )

summary.closer
```

```
## # A tibble: 1 x 3
##    mean     sd     n
##   <dbl>  <dbl> <int>
## 1 0.151 0.0927    24
```

The mean is greater than 0, which suggests that dopamine increases when birds sing closer to the target.

(c) Summarize the paired differences. Do the data suggest that there is a difference between dopamine in the brains of young zebra finches when they sing further away compared to closer to their adult song?

```
summary.diff <- data |>
  summarise(
    mean = mean(diff, na.rm = TRUE),
    sd = sd(diff, na.rm = TRUE),
    n = sum(!is.na(diff))
  )

summary.diff
```

```
## # A tibble: 1 x 3
##    mean    sd     n
##   <dbl> <dbl> <int>
## 1 0.354 0.214    24
```

The mean is positive, which implies higher dopamine when singing closer vs. further.

(d) **Optional Challenge:** Can you reproduce Figure 2(g)? Note that the you can use `geom_errorbar()` to plot the range created by adding the mean $\pm$ one standard deviation.

4. Conduct the inferences they do in the paper. Make sure to report the results a little more comprehensively – that is your parenthetical should look something like: ($t = 23.99$, $p < 0.0001$; $g = 1.34$; 95% CI: 4.43, 4.60).
Note: Your numbers may vary slightly as they performed some unclear correction of their p-values. I'm waiting to hear back from them via email!

(a) "The close responses differed significantly from 0 ($p = 1.63 \times 10^{-8}$)."

(b) "The far responses differed significantly from 0 ($p = 5.17 \times 10^{-8}$)."

(c) "The difference between populations was significant ($p = 1.04 \times 10^{-8}$)."

```
(t.closer <- t.test(data$closer.vals, mu = 0))
```

```
##
##  One Sample t-test
##
## data:  data$closer.vals
## t = 7.9945, df = 23, p-value = 4.34e-08
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  0.1121084 0.1903801
## sample estimates:
## mean of x
## 0.1512443
```

```r
(t.further <- t.test(data$further.vals, mu = 0))
```

```
##
##  One Sample t-test
##
## data:  data$further.vals
## t = -7.4785, df = 23, p-value = 1.337e-07
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.2593882 -0.1469806
## sample estimates:
##  mean of x
## -0.2031844
```

```r
(t.paired <- t.test(data$closer.vals, data$further.vals, paired = TRUE))
```

```
##
##  Paired t-test
##
## data:  data$closer.vals and data$further.vals
## t = 8.1073, df = 23, p-value = 3.41e-08
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  0.2639927 0.4448646
## sample estimates:
## mean difference
##       0.3544287
```

```r
library(effectsize)

g.closer <- hedges_g(data$closer.vals, mu = 0)
(g.closer)
```

```
## Hedges' g |       95% CI
## -------------------------
## 1.58      | [0.97, 2.17]
```

```r
g.further <- hedges_g(data$further.vals, mu = 0)
(g.further)
```

```
## Hedges' g |         95% CI
## ---------------------------
## -1.48     | [-2.04, -0.89]
```

```r
g.diff <- hedges_g(data$closer.vals, data$further.vals, paired = TRUE)

## For paired samples, 'repeated_measures_d()' provides more options.

(g.diff)
```

```
## Hedges' g |       95% CI
## -------------------------
## 1.60      | [0.99, 2.19]
```

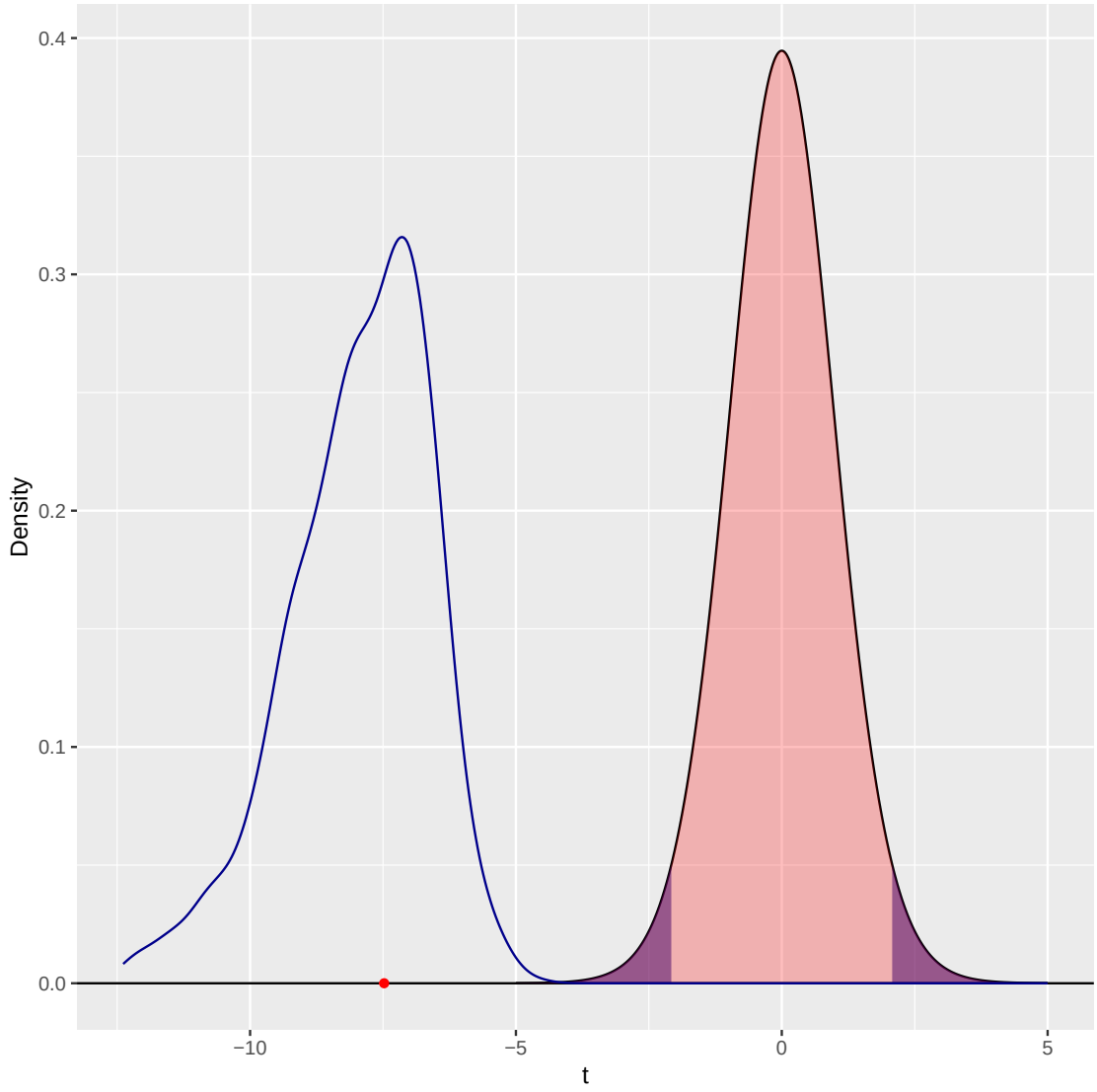
For the close responses, there is statistically discernible support for the researcher's alternative hypothesis ($t = 7.9945$,$p = 4.34 \times 10^{-8}$, $g = 1.58$, 95% CI: 0.1121, 0.1903).
For the far responses, there is statistically discernible support for the researcher's alternative hypothesis ($t = -7.4785$,$p = 1.337 \times 10^{-7}$, $g = -1.48$, 95% CI: -0.2593, -0.1469).

For the difference between close and far responses, there is statistically discernible support for the researcher's alternative hypothesis ($t = 8.1073$,$p = 3.41 \times 10^{-8}$, $g = 1.60$, 95% CI: 0.2639, 0.4448).

5. Reverse engineer the hypothesis test plot from Lecture 20 to create accurate hypothesis testing plots for each part of the previous question.

   (a) Question 4, part(a).

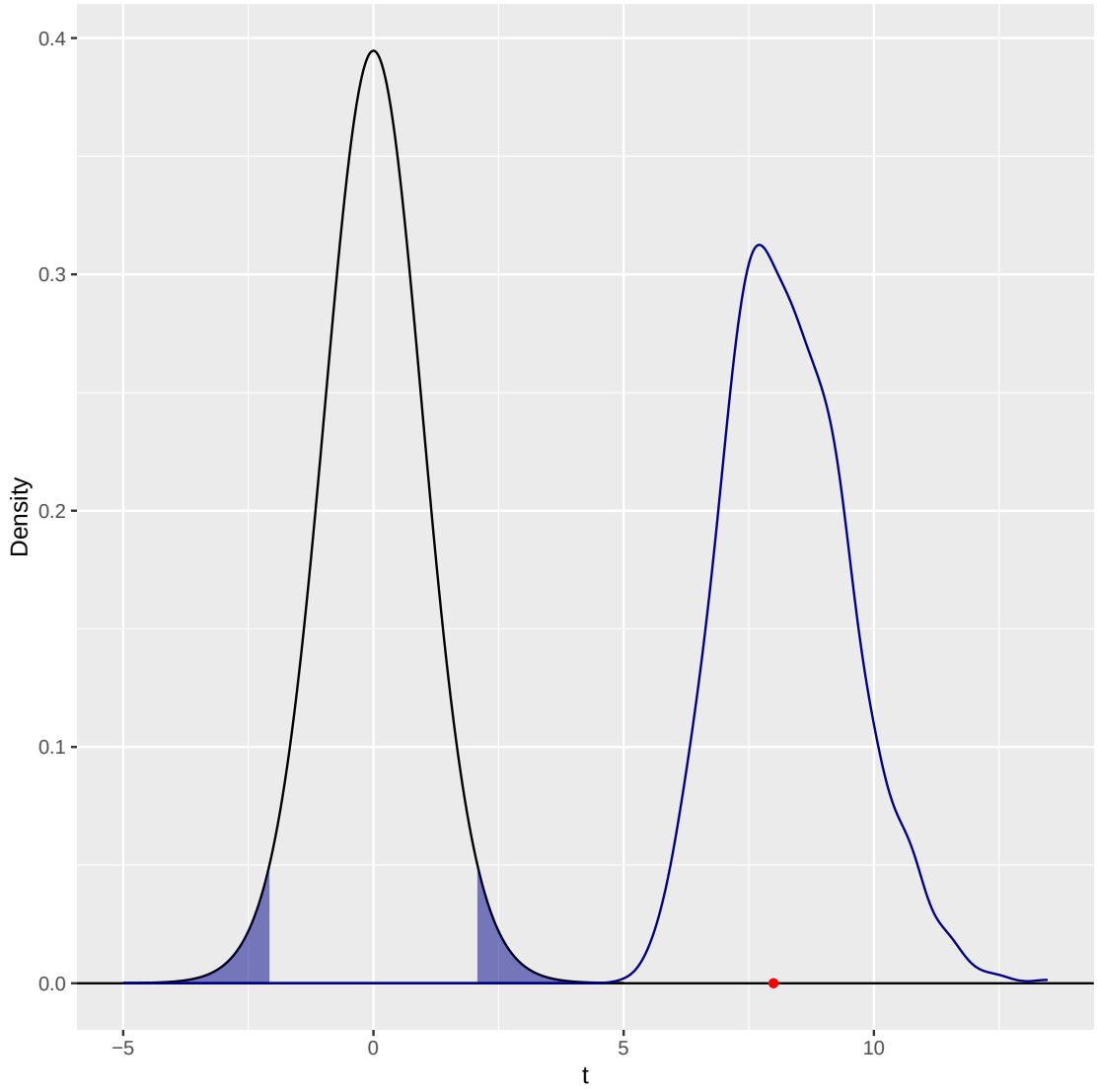
```
x <- data$further.vals
mu0 <- 0
xbar <- mean(x)
s <- sd(x)
n <- length(x)
t.stat <- (xbar - mu0)/(s/sqrt(n))

R <- 1000
resamples <- tibble(t = numeric(R))
for(i in 1:R){
  curr.sample <- sample(x = x, size = n, replace = TRUE)
  resamples$t[i] <- (mean(curr.sample) - mu0)/(sd(curr.sample)/sqrt(n))
}

ggdat.t <- tibble(t = seq(-5, 5, length.out = 1000)) |>
  mutate(pdf.null = dt(t, df = n - 1))
ggdat.obs <- tibble(t = t.stat, y = 0)

t.breaks <- c(-5, qt(0.025, df = n - 1), 0, qt(0.975, df = n - 1), 5, t.stat)
xbar.breaks <- t.breaks * s/(sqrt(n)) + mu0

ggplot() +
  geom_line(data = ggdat.t, aes(x = t, y = pdf.null)) +
  geom_hline(yintercept = 0) +
  geom_ribbon(data = subset(ggdat.t, t <= qt(0.025, df = n - 1)),
              aes(x = t, ymin = 0, ymax = pdf.null), fill = "darkblue", alpha = 0.5) +
  geom_ribbon(data = subset(ggdat.t, t >= qt(0.975, df = n - 1)),
              aes(x = t, ymin = 0, ymax = pdf.null), fill = "darkblue", alpha = 0.5) +
  geom_ribbon(data = subset(ggdat.t, t >= t.stat),
              aes(x = t, ymin = 0, ymax = pdf.null), fill = "red", alpha = 0.25) +
  geom_point(data = ggdat.obs, aes(x = t, y = y), color = "red") +
  stat_density(data = resamples, aes(x = t), geom = "line", color = "darkblue") +
  ylab("Density")
```



(b) Question 4, part(b).

```
x <- data$closer.vals
mu0 <- 0
xbar <- mean(x)
s <- sd(x)
n <- length(x)
t.stat <- (xbar - mu0)/(s/sqrt(n))

R <- 1000
resamples <- tibble(t = numeric(R))
for(i in 1:R){
  curr.sample <- sample(x = x, size = n, replace = TRUE)
  resamples$t[i] <- (mean(curr.sample) - mu0)/(sd(curr.sample)/sqrt(n))
}

ggdat.t <- tibble(t = seq(-5, 5, length.out = 1000)) |>
```

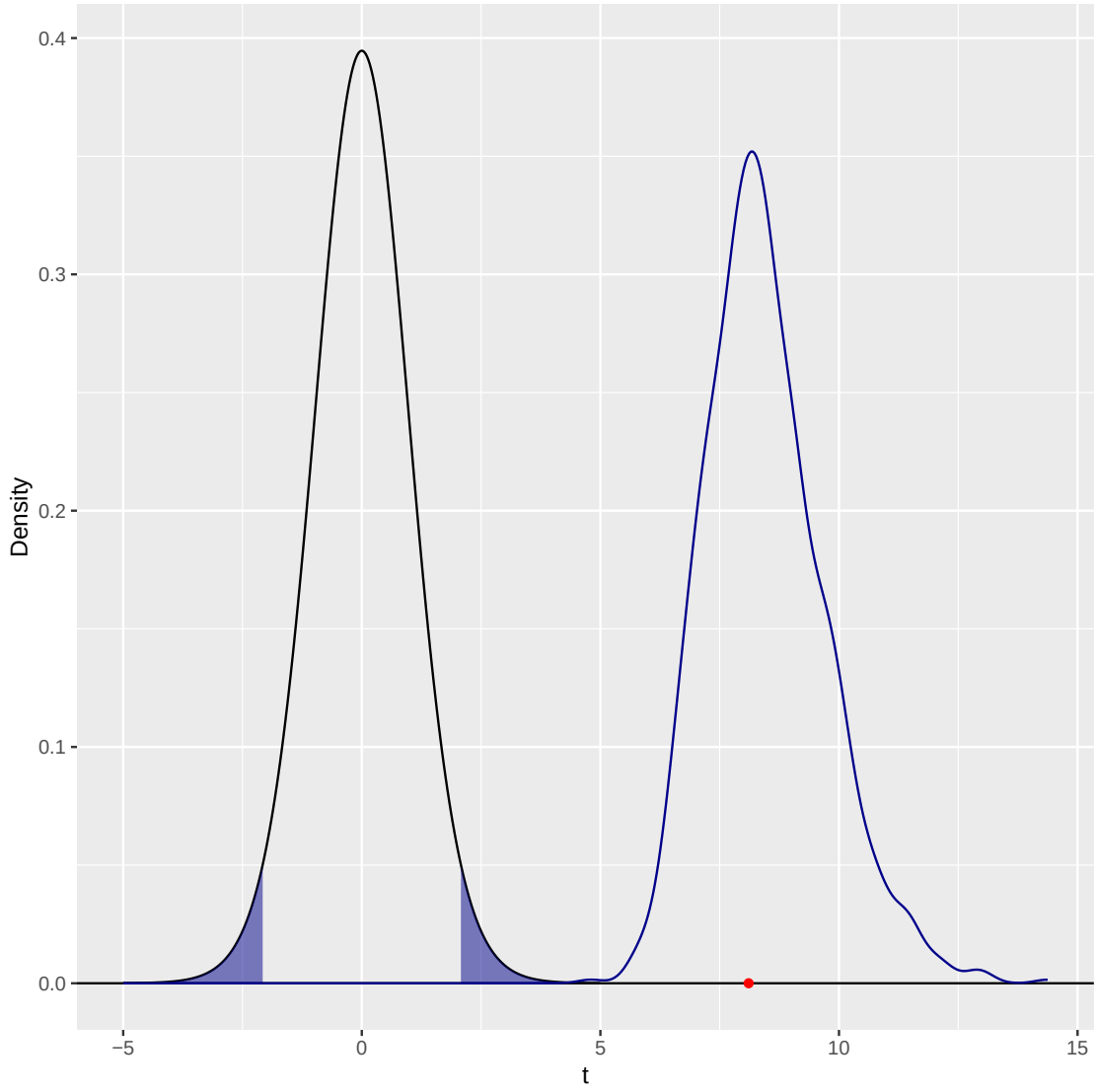
```
  mutate(pdf.null = dt(t, df = n - 1))
ggdat.obs <- tibble(t = t.stat, y = 0)

t.breaks <- c(-5, qt(0.025, df = n - 1), 0, qt(0.975, df = n - 1), 5, t.stat)
xbar.breaks <- t.breaks * s/(sqrt(n)) + mu0

ggplot() +
  geom_line(data = ggdat.t, aes(x = t, y = pdf.null)) +
  geom_hline(yintercept = 0) +
  geom_ribbon(data = subset(ggdat.t, t <= qt(0.025, df = n - 1)),
              aes(x = t, ymin = 0, ymax = pdf.null), fill = "darkblue", alpha = 0.5) +
  geom_ribbon(data = subset(ggdat.t, t >= qt(0.975, df = n - 1)),
              aes(x = t, ymin = 0, ymax = pdf.null), fill = "darkblue", alpha = 0.5) +
  geom_ribbon(data = subset(ggdat.t, t >= t.stat),
              aes(x = t, ymin = 0, ymax = pdf.null), fill = "red", alpha = 0.25) +
  geom_point(data = ggdat.obs, aes(x = t, y = y), color = "red") +
  stat_density(data = resamples, aes(x = t), geom = "line", color = "darkblue") +
  ylab("Density")
```



(c) Question 4, part(c).

```
  x <- data$closer.vals - data$further.vals
mu0 <- 0
xbar <- mean(x)
s <- sd(x)
n <- length(x)
t.stat <- (xbar - mu0)/(s/sqrt(n))

R <- 1000
resamples <- tibble(t = numeric(R))
for(i in 1:R){
  curr.sample <- sample(x = x, size = n, replace = TRUE)
  resamples$t[i] <- (mean(curr.sample) - mu0)/(sd(curr.sample)/sqrt(n))
}

ggdat.t <- tibble(t = seq(-5, 5, length.out = 1000)) |>
```

```r
  mutate(pdf.null = dt(t, df = n - 1))
ggdat.obs <- tibble(t = t.stat, y = 0)

t.breaks <- c(-5, qt(0.025, df = n - 1), 0, qt(0.975, df = n - 1), 5, t.stat)
xbar.breaks <- t.breaks * s/(sqrt(n)) + mu0

ggplot() +
  geom_line(data = ggdat.t, aes(x = t, y = pdf.null)) +
  geom_hline(yintercept = 0) +
  geom_ribbon(data = subset(ggdat.t, t <= qt(0.025, df = n - 1)),
              aes(x = t, ymin = 0, ymax = pdf.null), fill = "darkblue", alpha = 0.5) +
  geom_ribbon(data = subset(ggdat.t, t >= qt(0.975, df = n - 1)),
              aes(x = t, ymin = 0, ymax = pdf.null), fill = "darkblue", alpha = 0.5) +
  geom_ribbon(data = subset(ggdat.t, t >= t.stat),
              aes(x = t, ymin = 0, ymax = pdf.null), fill = "red", alpha = 0.25) +
  geom_point(data = ggdat.obs, aes(x = t, y = y), color = "red") +
  stat_density(data = resamples, aes(x = t), geom = "line", color = "darkblue") +
  ylab("Density")
```



References

Champely, S. (2020). *pwr: Basic Functions for Power Analysis*. R package version 1.3-0.

Kasdin, J., Duffy, A., Nadler, N., Raha, A., Fairhall, A. L., Stachenfeld, K. L., and Gadagkar, V. (2025). Natural behaviour is learned through dopamine-mediated reinforcement. *Nature*, pages 1–8.