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.analysis = pwr.t.test(d=0.65,
                            sig.level = 0.05,
                            power = 0.80,
                            type = "one.sample",
                            alternative = "two.sided")
```

   I got that n = 20.58039, so 21 observations are required in order to detect a moderate-to-large effect when using this significance level and power.

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`).

```
library(tidyverse)
dopamine.data = read_csv("DopamineData.csv")
dopamine.data = dopamine.data |>
  #create column that contains the difference between farther and closer values
  mutate(difference = Closer_vals - Farther_vals)
```

   To collect the data for Figure 2(g), I had to first download the entire source data file for all of Figure 2. Then, I extracted the data from the Farther_vals and Closer_vals tabs which correspond to Figure 2(g). Lastly, I put the columns side-by-side into a new Excel file, and downloaded it as a .csv file.

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?

   (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?

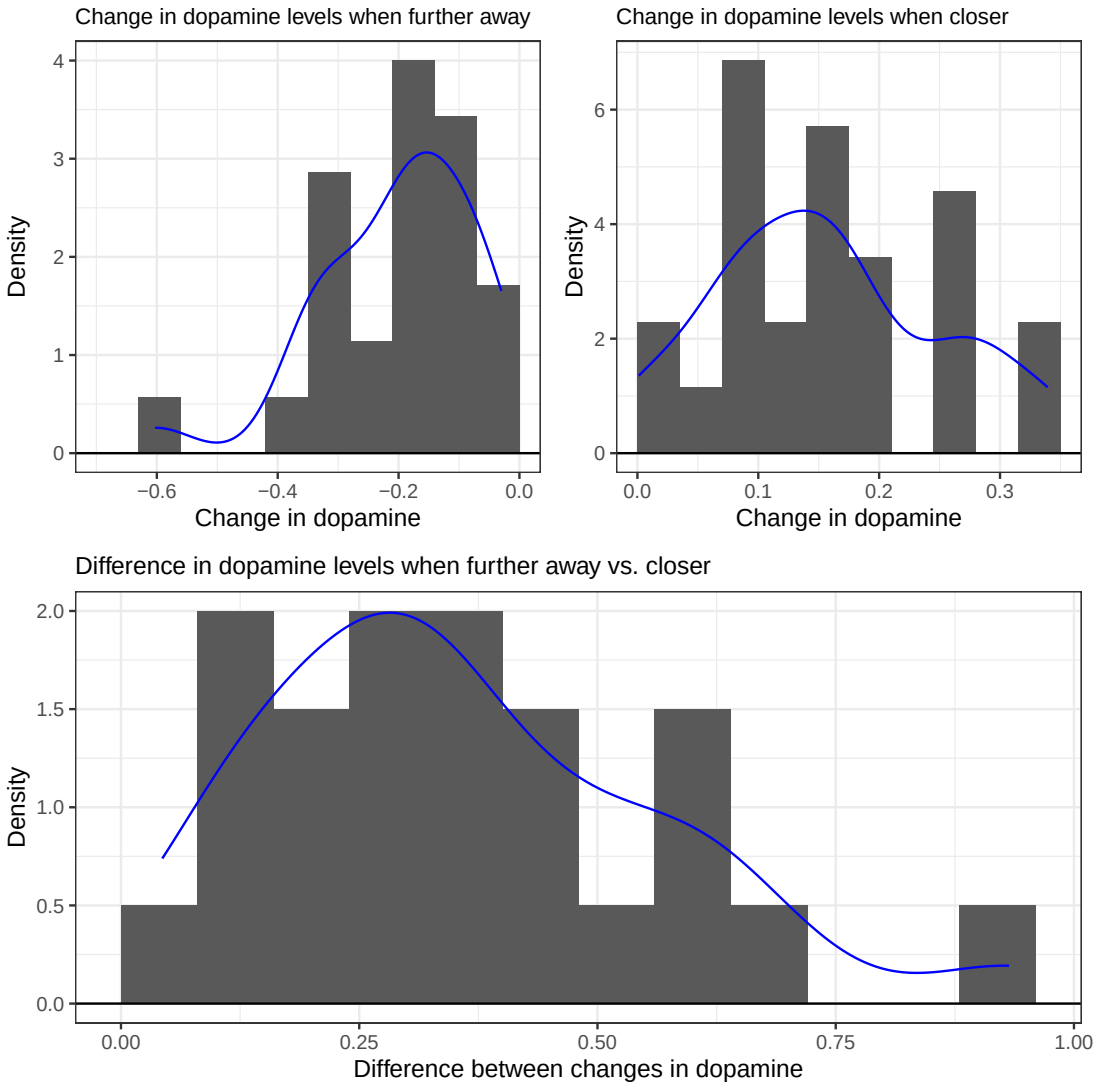
   (c) Summarize the paired differences. Do the data suggest that there is a difference between dopamine in the brains of young zebra finches when they sing further away compared to closer to their adult song?

	min	max	median	mean	sd
farther	-0.60	-0.03	-0.19	-0.20	0.13
closer	0.00	0.34	0.15	0.16	0.09
differences	0.04	0.93	0.33	0.36	0.21

Table 1: Numerical Summary for the further, closer, and differences data

```
library(patchwork)
summary = function(data) {
  mean = mean(data)
  min = min(data)
  max = max(data)
  sd = sd(data)
  median = median(data)
  return(c(min, max, median, mean, sd))
}
farther = summary(dopamine.data$`Farther_vals`)
closer = summary(dopamine.data$`Closer_vals`)
differences = summary(dopamine.data$difference)

table = data.frame()
table = rbind(farther,
              closer,
              differences) |> as.data.frame()
summary.table = table |>
  rename(min = "V1",
         max = "V2",
         median = "V3",
         mean = "V4",
         sd = "V5")

farther.plot = ggplot(data = dopamine.data, aes(x=`Farther_vals`))+
  geom_histogram(aes(y=after_stat(density)),
                 breaks=seq(-0.7,0,0.07)) +
  geom_hline(yintercept=0)+
  theme_bw()+
  geom_density(color = "blue") +
  xlab("Change in dopamine")+
  ylab("Density")+
  labs(color = "", title = "Change in dopamine levels when further away") +
  theme(plot.title = element_text(size = 10))
closer.plot = ggplot(data = dopamine.data, aes(x=`Closer_vals`))+
  geom_histogram(aes(y=after_stat(density)),
                 breaks=seq(0,0.35,0.035)) +
  geom_hline(yintercept=0)+
  theme_bw()+
  geom_density(color = "blue") +
  xlab("Change in dopamine")+
  ylab("Density")+
  labs(color = "", title = "Change in dopamine levels when closer") +
  theme(plot.title = element_text(size = 10))
differences.plot = ggplot(data = dopamine.data, aes(x=difference))+
  geom_histogram(aes(y=after_stat(density)),
                 breaks=seq(0,1,0.08)) +
  geom_hline(yintercept=0)+
  theme_bw()+
  geom_density(color = "blue") +
  xlab("Difference between changes in dopamine")+
  ylab("Density")+
  labs(title = "Difference in dopamine levels when further away vs. closer") +
```

```
  theme(plot.title = element_text(size = 11))
(farther.plot + closer.plot) / differences.plot
```



The data does suggest that dopamine levels decrease when young zebra finches sing further away from their adult song. Although some of the data is close to 0, every data point is negative, indicating dopamine decreased for every observation. We can say the opposite thing about young zebra finches when they sing closer to their adult song. Every observation is positive, meaning even if not always by much, they all increased in dopamine levels when singing close to their adult song.

When looking at the differences between the two sets of data, you can see there is quite a significant difference between changes in dopamine levels for the zebra finches, depending on whether they are closer or further away from their adult song. When doing closer - further, every observation is positive, meaning dopamine levels always change more positively when closer, than they do when further from their adult song.

Note that the plots don't follow a normal distribution very well, mainly due to the fact that we only have 25 observations for each data set.

(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 ± 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 ($p = 1.63 \times 10^{-8}$)."

```
library(effectsize)
closer.t.test = t.test(x=dopamine.data$`Closer_vals`,
                       alternative = "two.sided",
                       mu = 0,
                       conf.level = 0.95)
g = hedges_g(x = dopamine.data$`Closer_vals`, mu = 0, alternative = "two.sided")
```

   t = 8.3024, p <0.00000002, g = 1.61, 95% CI: 0.1173875, 0.1950586

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

```
farther.t.test = t.test(x=dopamine.data$`Farther_vals`,
                        alternative = "two.sided",
                        mu = 0,
                        conf.level = 0.95)
g = hedges_g(x = dopamine.data$`Farther_vals`, mu = 0, alternative = "two.sided")
```

   t = -7.778, p <0.00000006, g = 1.51, 95% CI: -0.2565176, -0.1489313

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

```
differences.t.test = t.test(x=dopamine.data$`difference`,
                        alternative = "two.sided",
                        mu = 0,
                        conf.level = 0.95)
g = hedges_g(x = dopamine.data$`difference`, mu = 0, alternative = "two.sided")
```

   t = 8.5109, p <0.00000002, g = 1.65, 95% CI: 0.2719028, 0.4459921
   For all three of these results, it is clear that there is statistically discernible support for these given hypotheses. In all cases, the p-values are much smaller than 0.05. This alone doesn't necessarily mean that the responses were very far away from zero, but that we are very confident that they at least somewhat differ from zero. However, in all cases, we have an effect size value much higher than 0.8, which is considered the threshold for a large effect. So not only are we very confident that there is a statistical difference from zero, but we can see that there is a substantial difference.

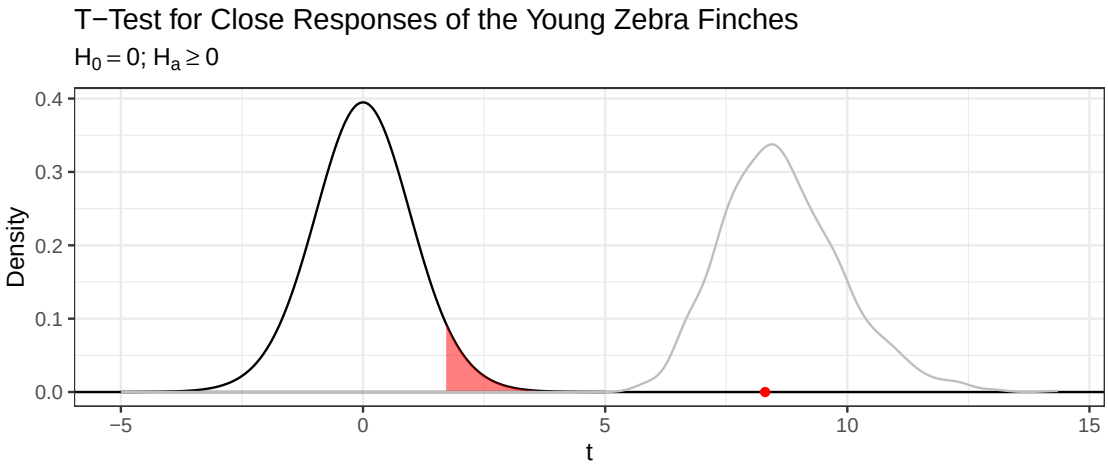
5. Reverse engineer the hypothesis test plot from Lecture 20 to create accurate hypothesis testing plots for each part of the previous question.

   (a) Question 4, part(a).

```
R = 1000 #number of resamples
mu0 = 0 #hypothesized mean
close.vec = dopamine.data$Closer_vals #vector for closer values
far.vec = dopamine.data$Farther_vals #vector for farther values
diff.vec = dopamine.data$difference #vector for difference
n = length(close.vec) #sample size
resamples = tibble(close = numeric(R),
                   far = numeric(R),
                   difference = numeric(R))
for(i in 1:R){
```

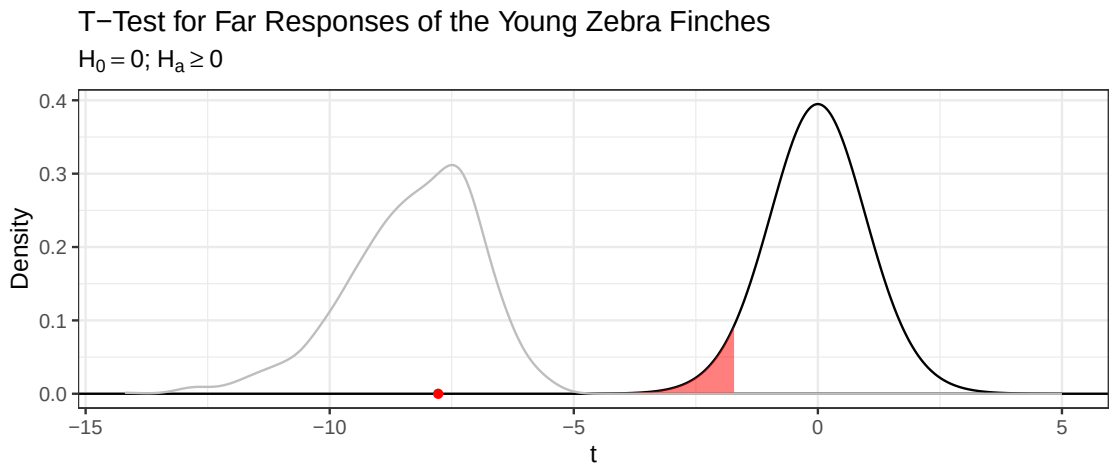
```r
  #resampling close responses
  close.sample <- sample(x=close.vec,
                         size=n,
                         replace=T)
  #resampling far responses
  far.sample <- sample(x=far.vec,
                          size=n,
                          replace=T)
  #resampling difference
  difference.sample <- sample(x=diff.vec,
                          size=n,
                          replace=T)
  #resampled t statistic for close values
  resamples$close[i] = (mean(close.sample)-mu0)/(sd(close.sample)/sqrt(n))
  #resampled t statistic for far values
  resamples$far[i] = (mean(far.sample)-mu0)/(sd(far.sample)/sqrt(n))
  #resampled t statistic for differences
  resamples$difference[i] = (mean(difference.sample)-mu0)/(sd(difference.sample)/sqrt(n))
}

#data for null T disribution
null.dat = tibble(t=seq(-5, 5, length.out = 1000)) |>
  mutate(density.null = dt(t, df=24))

#Part a

close.t.stat = (mean(close.vec) - mu0)/(sd(close.vec)/sqrt(n)) #close t statistic
#data for plotting observed point
close.obs = tibble(t = close.t.stat,
             y = 0)
#plot for close values
close.plot = ggplot() +
  # null distribution
  geom_line(data=null.dat,
            aes(x=t, y=density.null))+
  geom_hline(yintercept=0)+
  # rejection region
  geom_ribbon(data=subset(null.dat, t>=qt(0.95, df=n-1)),
              aes(x=t, ymin=0, ymax=density.null),
              fill="red", alpha=0.5) +
  #p-value region
  geom_ribbon(data=subset(null.dat, t>=close.t.stat),
              aes(x=t, ymin=0, ymax=density.null),
              fill="blue", alpha=0.5) +
  #observation point (t statistic)
  geom_point(data=close.obs, aes(x=t, y=y), color="red")+
  #Resampling Distribution
  stat_density(data=resamples,
               aes(x=close),
               geom="line", color="grey")+
  #clean up aesthetics
  theme_bw()+
  ylab("Density")+
```

```
  ggtitle("T-Test for Close Responses of the Young Zebra Finches",
          subtitle=bquote(H[0]==0*";"~H[a]>=0))
close.plot
```



(b) Question 4, part(b).

```
far.t.stat = (mean(far.vec) - mu0)/(sd(far.vec)/sqrt(n)) #far t statistic
#data for plotting observed point
far.obs = tibble(t = far.t.stat,
                 y = 0)
#plot for far values
far.plot = ggplot() +
  # null distribution
  geom_line(data=null.dat,
            aes(x=t, y=density.null))+
  geom_hline(yintercept=0)+
  # rejection region
  geom_ribbon(data=subset(null.dat, t<=qt(0.05, df=n-1)),
              aes(x=t, ymin=0, ymax=density.null),
              fill="red", alpha=0.5) +
  #p-value region
  geom_ribbon(data=subset(null.dat, t<=far.t.stat),
              aes(x=t, ymin=0, ymax=density.null),
              fill="blue", alpha=0.5) +
  #observation point (t statistic)
  geom_point(data=far.obs, aes(x=t, y=y), color="red")+
  #Resampling Distribution
  stat_density(data=resamples,
               aes(x=far),
               geom="line", color="grey")+
  #clean up aesthetics
  theme_bw()+
  ylab("Density")+
  ggtitle("T-Test for Far Responses of the Young Zebra Finches",
          subtitle=bquote(H[0]==0*";"~H[a]>=0))
far.plot
```



(c) Question 4, part(c).

```
diff.t.stat = (mean(diff.vec) - mu0)/(sd(diff.vec)/sqrt(n)) #difference t statistic
#data for plotting observed point
diff.obs = tibble(t = diff.t.stat,
                  y = 0)
#plot for far values
diff.plot = ggplot() +
  # null distribution
  geom_line(data=null.dat,
            aes(x=t, y=density.null))+
  geom_hline(yintercept=0)+
  # rejection regions
  geom_ribbon(data=subset(null.dat, t<=qt(0.025, df=n-1)),
              aes(x=t, ymin=0, ymax=density.null),
              fill="red", alpha=0.5) +
  geom_ribbon(data=subset(null.dat, t>=qt(0.975, df=n-1)),
              aes(x=t, ymin=0, ymax=density.null),
              fill="red", alpha=0.5) +
  #p-value region
  geom_ribbon(data=subset(null.dat, t>=diff.t.stat),
              aes(x=t, ymin=0, ymax=density.null),
              fill="blue", alpha=0.5) +
  geom_ribbon(data=subset(null.dat, t<=-diff.t.stat),
              aes(x=t, ymin=0, ymax=density.null),
              fill="blue", alpha=0.5) +
  #observation point (t statistic)
  geom_point(data=diff.obs, aes(x=t, y=y), color="red")+
  #Resampling Distribution
  stat_density(data=resamples,
               aes(x=difference),
               geom="line", color="grey")+
  #clean up aesthetics
  theme_bw()+
  ylab("Density")+
  ggtitle("T-Test for Difference in Close and Far Responses for Young Zebra Finches",
          subtitle=bquote(H[0]==0*";"~H[a]>=0))
diff.plot
```

As you can see by the red dots on each of these plots, the test statistic is significantly out of range of the null distribution. Given that, along with having very small p-values for each of the hypothesis tests, we can say there is statistically discernible support for the alternative hypotheses.

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.