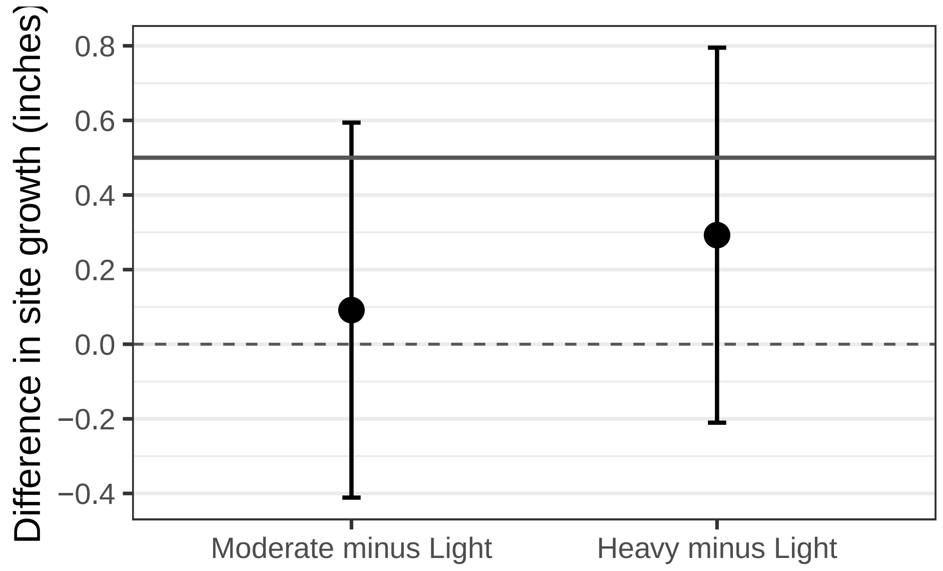
# FES 524 Winter 2022 Lab 2

## Bonus graphics

We will combine the summary information by thinning group and the figure we made in Lab 2 to make a final graphic. The figure we will build on is named `g1` and the summary table is named `sumdat`. These were both created in Lab 2 (see `lab2.example.handout.pdf`).

This is what the figure we'll be working with looked like at the end of Lab 2:



In order to combine a table with a graph, we will need to turn the table into a graphical object. Graphical objects are referred to as "grobs". Use `tableGrob()` from the package **gridExtra** with some help from `gtable_add_grob()` from package **gtable** to do this. The figure `g1` is already a graphical object.

```r
library(grid)
library(gridExtra)
library(gtable)
```

We've already created the summary table, but we can improve the names of the thinning group variable `Thinning` by capitalizing the names.

```r
sumdat
```

```
# A tibble: 3 x 7
  Thinning     n  Mean    SD Median Minimum Maximum
  <fct>    <int> <dbl> <dbl>  <dbl>   <dbl>   <dbl>
1 light        7   4     0.5    4.1     3.1     4.5
2 moderate     7   4.1   0.4    4.1     3.6     4.7
3 heavy        7   4.3   0.5    4.3     3.8     4.9
```

```r
# Improve category names in sumdat
sumdat$Thinning = c("Light", "Moderate", "Heavy")
```

I am only going to use the columns in `sumdat` that represent the group, sample size, means, and standard deviations of change in diameter. I'll name the object with only these columns in it `d1`.

```r
( d1 = sumdat[ , 1:4] )
```

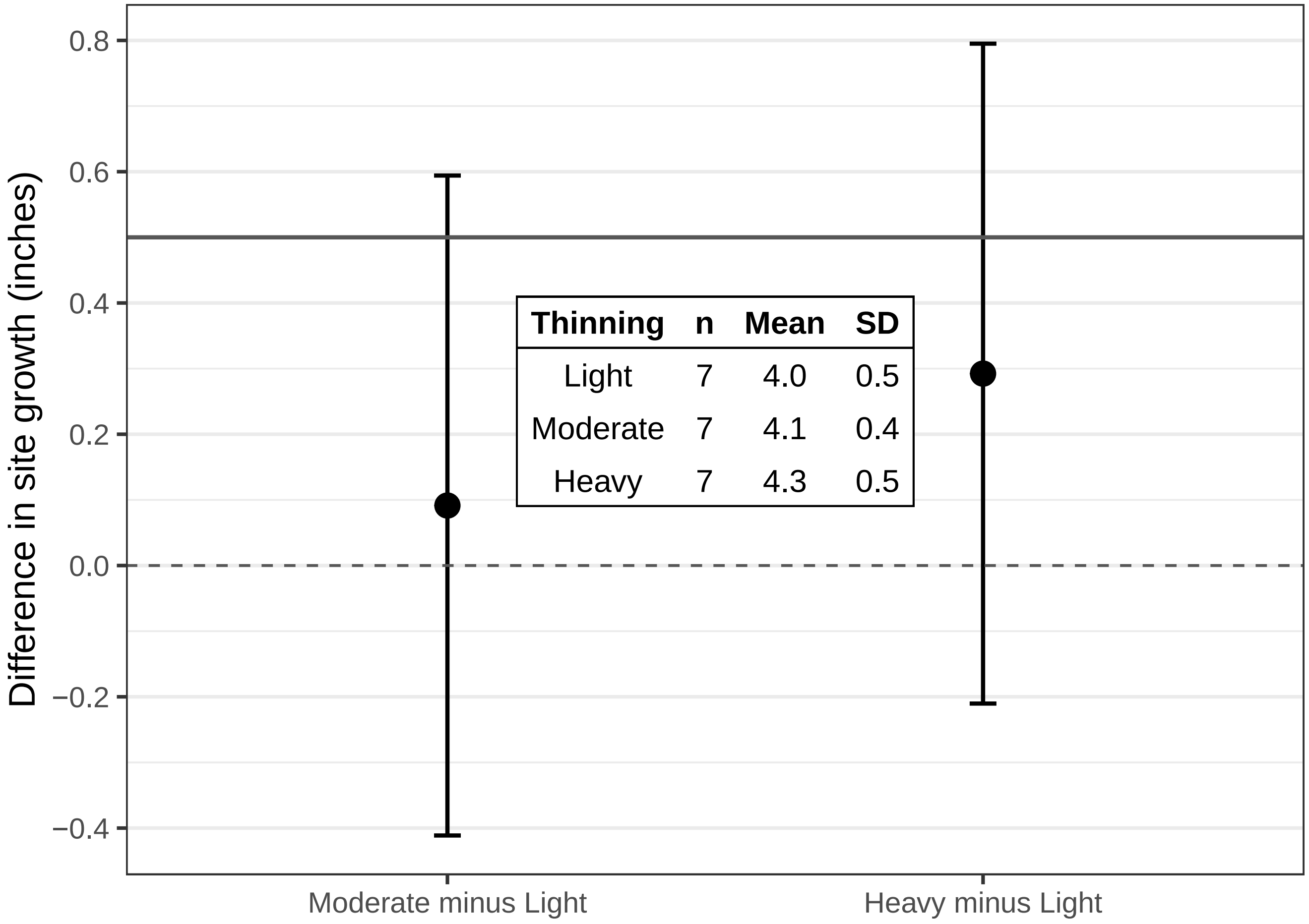
```
# A tibble: 3 x 4
  Thinning     n  Mean    SD
  <chr>    <int> <dbl> <dbl>
1 Light        7   4     0.5
2 Moderate     7   4.1   0.4
3 Heavy        7   4.3   0.5
```

Now we can use `tableGrob()` on `d1` to make a table graphical object to add to our current plot, which I name `t1`. In this example, I make the background color white in both the columns and column names. I then add a black box around the outside of `t1` and a line below the column names with `gtable_add_grob()`.

```
t1 = tableGrob(sumdat[,1:4], rows = NULL,
            theme = ttheme_default(core = list(bg_params = list(fill = "white") ),
                                   colhead = list(bg_params = list(fill = "white", col = "white") ) ) )
# Add rectangle around box
t1 = gtable_add_grob(t1, grobs = rectGrob(gp = gpar(fill = NA, lwd = 2) ),
                 t = 1,
                 b = nrow(t1),
                 r = ncol(t1),
                 l = 1)
# Add line under column names
t1 = gtable_add_grob(t1,grobs = rectGrob(gp = gpar(fill = NA, lwd = 2) ),
                 t = 1,
                 r = ncol(t1),
                 l = 1)
```

We will add the graphical object table `t1` to the grob `g1` using `annotation_custom()` and name the combined object `g1fin`. Notice how I define where the table will be placed in `annotation_custom()` by giving coordinates for the left and right (`xmin` and `xmax`, respectively) and top and bottom (`ymax` and `ymin`) of the `tableGrob()` object. I placed the table in the empty space between the two lines. Object placement using plot coordinates can take some practice, and may take you several tries to get it just right.

```
g1fin = g1 +
    annotation_custom(grob = t1,
                      xmin = 1, xmax = 2,
                      ymin = 0, ymax = .5)
g1fin
```



| Thinning | n | Mean | SD |
|---|---|---|---|
| Light | 7 | 4.0 | 0.5 |
| Moderate | 7 | 4.1 | 0.4 |
| Heavy | 7 | 4.3 | 0.5 |

If you wanted to save the final figure `g1fin`, you could do so using `ggsave()` or the `Export` drop-down menu in the RStudio Plots pane. In the example here we'll save `g1fin` as a jpeg named `lab2figure`.

```r
ggsave("lab2figure.jpg", plot = g1fin, height = 5, width = 7)
```