

# HW 3: Drafting Visuals

Visualizing forest health impacts of various treatment options

AUTHOR

Liv Hemond

PUBLISHED

February 25, 2025

## Part 1: Option Selection

I will be pursuing Option 1, which is an infographic.

## Part 2: Defining Viz Questions

I have rephrased my questions slightly since the previous two homeworks. They are now:

[Overarching] How well do different forest restoration treatments (e.g., prescribed fire, mechanical thinning, or a combination of the two) meet various management goals?

- [Question 1, Viz 1] Which treatments are best at **reducing fuel loads and fire risk?**
- [Question 2, Viz 2] Which treatments are best at **increasing forest resilience?** (i.e., reducing tree densities enough to protect against the impacts of drought, pests, or disease)
- [Question 3, Viz 3] Which treatments are best at **protecting forest carbon stocks?**
- [Additional info at the end, but likely not a full viz] How expensive are the different treatments?

## Part 3: Explanation of Variables

Question 1: Which treatments are best at **reducing fuel loads and fire risk?**

I have three data sets that I will join. One contains data on surface fuel amounts, one contains data on ground fuel amounts, and one contains the modeled probability of severe fire (“probability of torching” or ptorch). Each data set is structured similarly, with the data available for each plot in each treatment group, all at the end of the 18-year study period. I can join these data sets using plot ID numbers and treatment group, which are shared across data sets. I then have three variables to include in my visualization: fuel amounts (continuous data, for both surface and ground fuels), treatment group (categorical, which I use as a factor), and ptorch (continuous, which I use to set the opacity value for the boxplot fill, so that treatment groups with higher burn probabilities are shaded in darker).

## Question 2: Which treatments are best at **increasing forest resilience?**

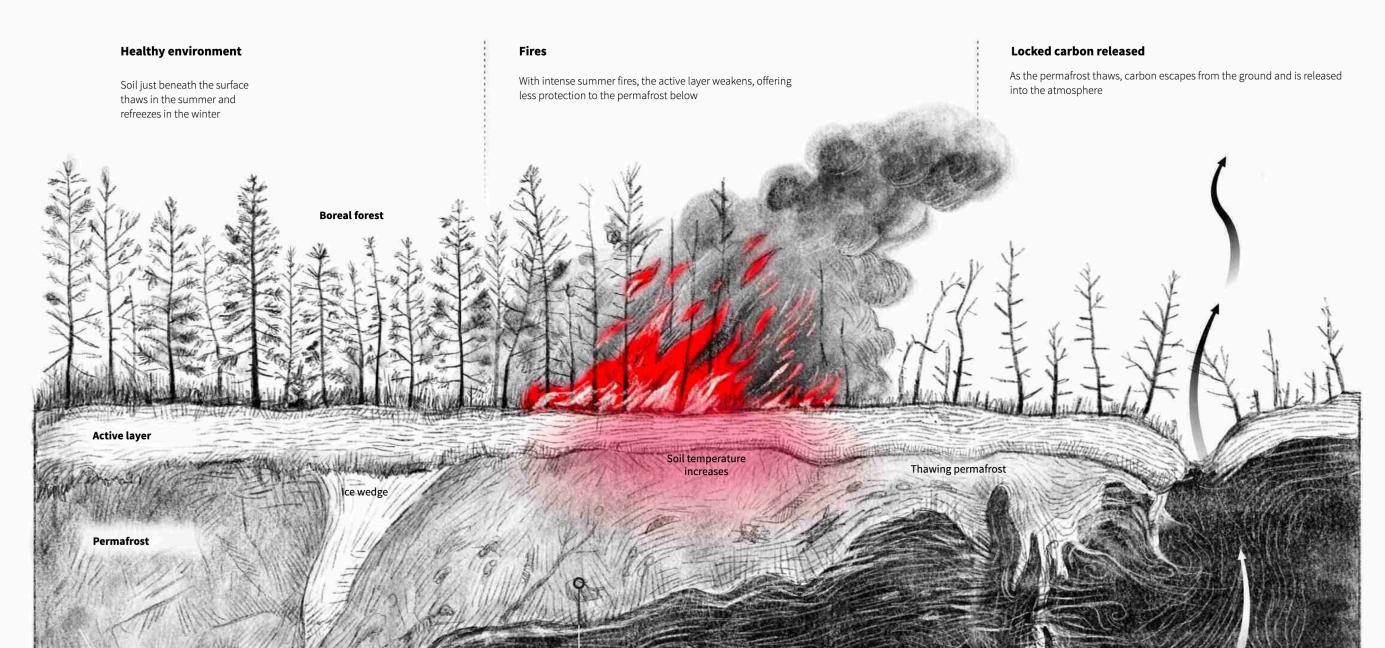
I have one data set which contains forest stand density information for each plot and treatment group at the end of the 18-year study period. It contains both the raw stand density numbers, and assigns each of them to four different “stand density index (SDI) zones” which are commonly discussed by foresters (for information on the different SDI zones, see reference sheet [here](#)). I will re-label these zones to reduce technical jargon. Then, I will calculate the number of forest plots in each treatment group and SDI zone and divide this by the total number of plots in each treatment group. This gives me the percent of plots, for each treatment group, that fall in each different SDI zone. I will then visualize my data using these three variables: SDI zone (categorical, treated as a factor and also mapped onto an opacity value for fill in my graph), percent of plots in each SDI zone (continuous), and treatment group (categorical, also a factor).

## Question 3: Which treatments are best at **protecting forest carbon stocks?**

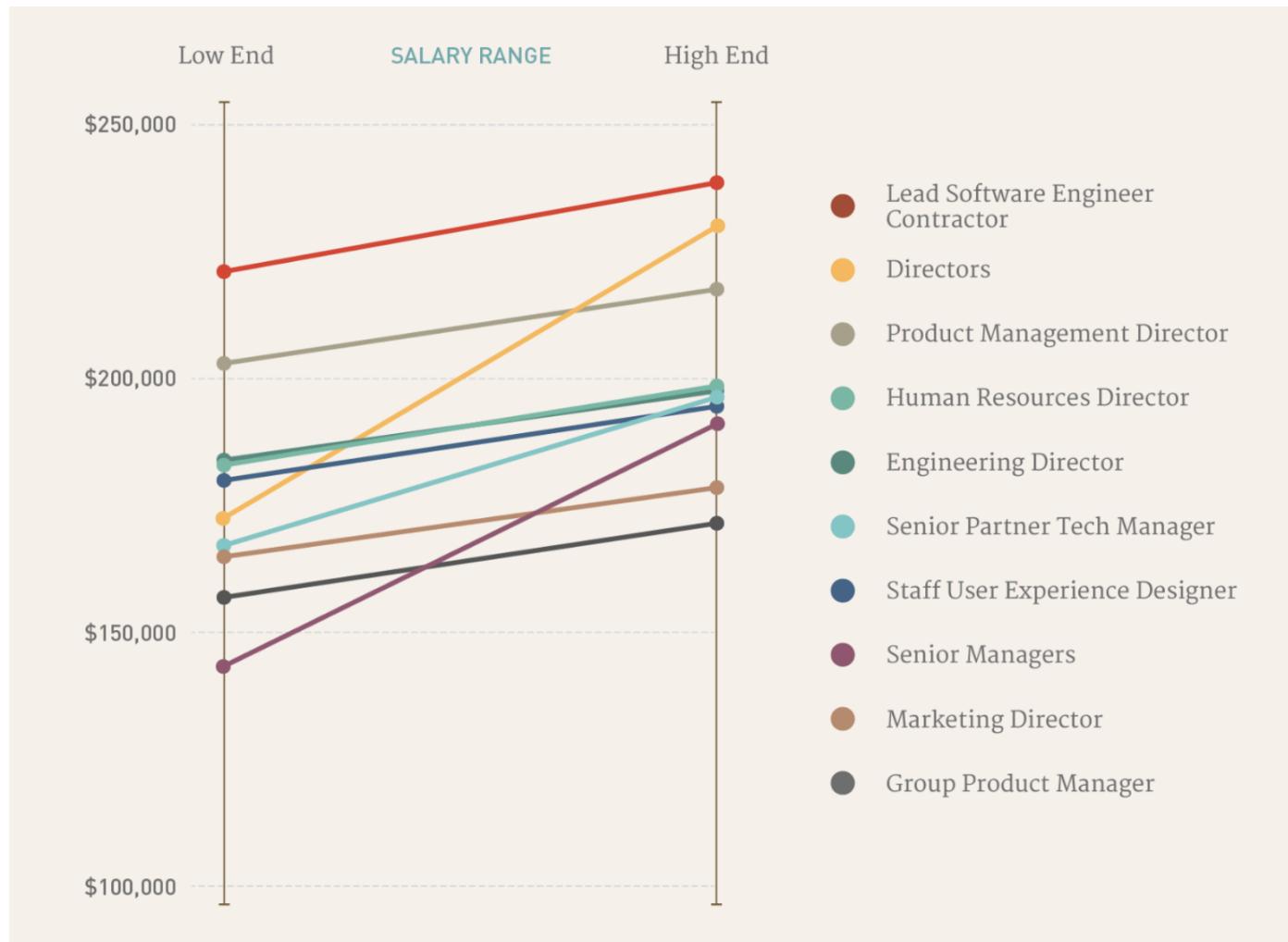
I have one data set which contains forest carbon for each plot and treatment group. There is both the total carbon stored aboveground in live trees (“live carbon”), and the amount of carbon predicted to remain in live trees after fire has passed through the system (“stable carbon”). I will calculate the average live carbon and stable carbon value for each treatment group, and use my viz to show the difference between the two. Thus, I’ll be showing three different variables: treatment group (categorical, factor), average live carbon (continuous), and average stable carbon (continuous).

## Part 4: Inspiration

One piece of inspiration for me is this drawing of boreal forest fires and carbon release, created by Manas Sharma, Gloria Dickie, Adolfo Arranz and Simon Scarr in their article “[Why Arctic fires are releasing more carbon than ever](#)”. I would like to borrow this idea and create my own cross-section of a forest to use for my visualization. I think I could similarly divide it into three parts, the first showing where “ground” and “surface” fuels are found in the forest, the second illustrating different forest densities, and the third showing fire in the forest for where I show the data on stable carbon. I plan to use Affinity Designer to create this layout style with appropriate tree species to my study system.

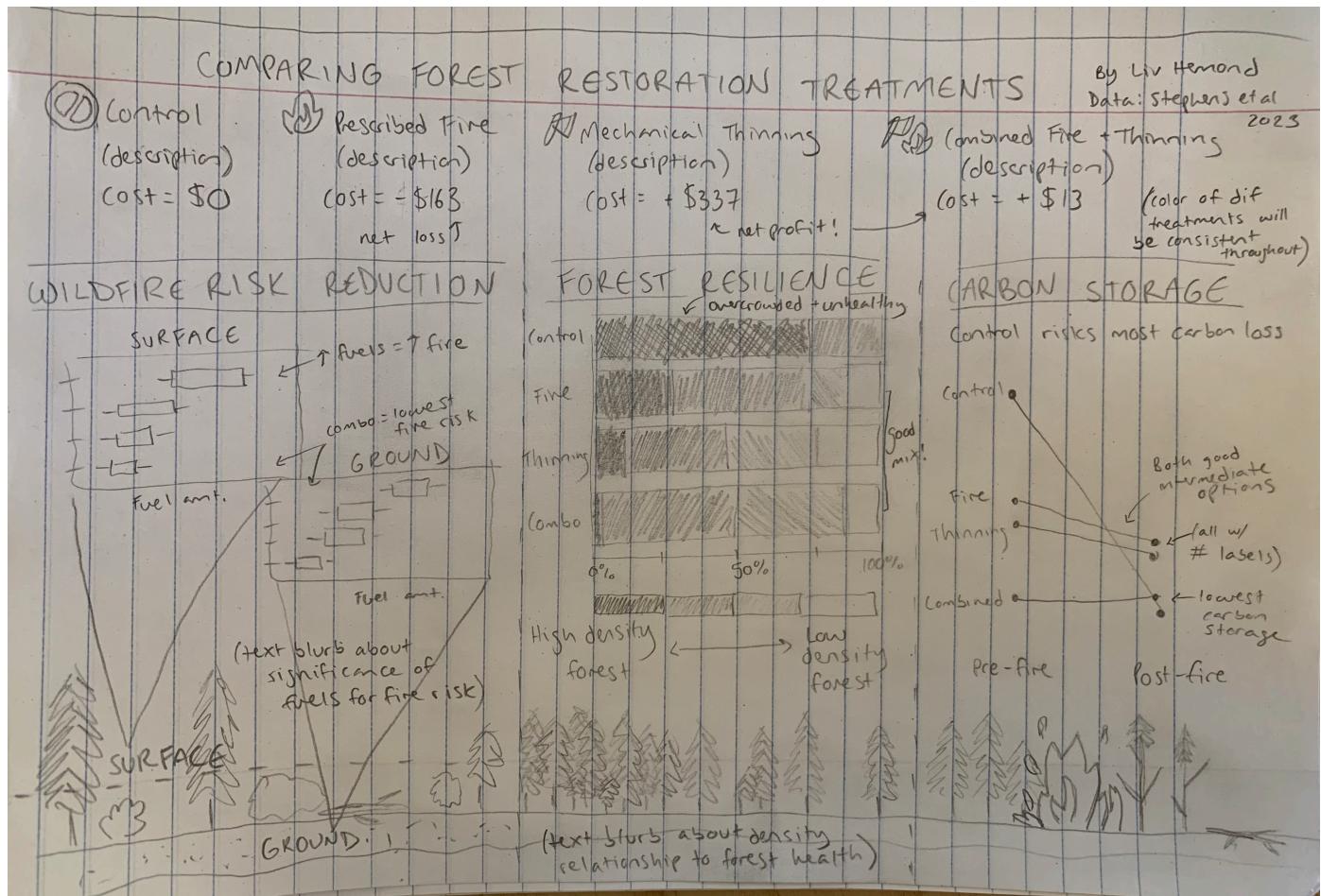


A second piece of inspiration is this type of chart design, called a slope chart. I was originally thinking of expressing the difference between live carbon and stable carbon in a dumbbell plot, but something felt off. Instead, I think I'd like to use a slope chart. On the left would be “pre-fire”, to show the live carbon amount, and on the right would be “post-fire”, showing the stable carbon amount left over after a fire. Each treatment group would then be shown as a different color, so that the viewer can track how they each change.



## Part 5: Hand-Drawn Infographic Idea

I plan to create an infographic using ggplot for all the charts and Affinity Designer to arrange them, add graphics, and add text annotations.



## Part 6: Mock-up of Viz

### I. Setup

#### i. Load libraries and data

##### ▼ Code

```
# .....load libraries.....
library(tidyverse)
library(here)
library(paletteer)
library(scales)

# .....set color palette.....
conifer <- paletteer_d("calecopal::conifer", n = 5)
# subset individual colors from palette
darkbrown <- "#765043FF"
fuels <- c("Surface" = "#979A6BFF", "Ground" = "#CC7540FF")
```

```
treatments <- c("Combined Thinning and Prescribed Fire" = "#979A6BFF",
              "Prescribed Fire" = "#CC7540FF",
              "Mechanical Thinning" = "#39692FFF",
              "Control" = "#765043FF")

# .....load data files.....
# part A: surface fuels and duff
# surface fuels
fuels_data <- read_csv(here("data", "surface_fuels_data.csv"))
# duff
duff_data <- read_csv(here("data", "duff_data.csv"))
# ptorch
ptorch_data <- read_csv(here("data", "ptorch_data.csv"))

# part B: stand density index
# stand density index (SDI)
sdi_data <- read_csv(here("data", "sdi_data.csv"))

# part C: stable live tree carbon
# stable live tree carbon (STLC)
stlc_data <- read_csv(here("data", "sltc_data.csv"))
```

## ii. Clean and wrangle data

## A. Surface fuels and duff

## ▼ Code

```

# calculate ptorch mean for each treatment and set it as an opacity value
ptorch_data_clean <- ptorch_data %>%
  # calculate mean probability of torching
  group_by(treatment) %>%
  summarize(mean_ptorch = mean(ptorch)) %>%
  mutate(opacity_val = scales::rescale(x = mean_ptorch,
                                         to = c(0.5, 0.8))) %>%
  # convert treatment to factor and set levels to match fuels data
  mutate(treatment = factor(treatment,
                            levels = c("mechburn", "burn", "mech", "control"),
                            labels = c("Combined Thinning and Prescribed Fire",
                                      "Mechanical Thinning", "Control")))

# join ptorch values with fuels data
fuels_duff_final <- left_join(fuels_duff_clean, ptorch_data_clean, by = join_by(

```

## B. Stand density index

### ▼ Code

```

# stand density index
sdi_clean <- sdi_data %>%
  # convert treatment and sdi_zone (competitive benchmark zones) to factors and :
  mutate(treatment = factor(treatment,
                            levels = c("mechburn", "burn", "mech", "control"),
                            labels = c("Combined Thinning and Prescribed Fire",
                                      "Mechanical Thinning", "Control"))),
  sdi_zone_long = factor(sdi_zone,
                         levels = c(">60", "35–60", "25–35", "<25"),
                         labels = c("Extremely High", "High", "Moderate",
                                   "Low"))
  # create a column for opacity based upon sdi_zone
  mutate(opacity_val = factor(sdi_zone,
                             levels = c(">60", "35–60", "25–35", "<25"),
                             labels = c(1, 0.8, 0.6, 0.4)))

```

## C. Stable live tree carbon

### ▼ Code

```

stlc_clean <- stlc_data %>%
  # convert treatment to factor and set levels
  mutate(treatment = factor(treatment,
                            levels = c("mechburn", "burn", "mech", "control")),

```

```
labels = c("Combined Thinning and Prescribed Fire",
         "Mechanical Thinning", "Control"))
```

## II. Visualizations

### A. Surface fuels and duff

Compare surface fuel loads and duff (aka ground fuel loads) across treatments.

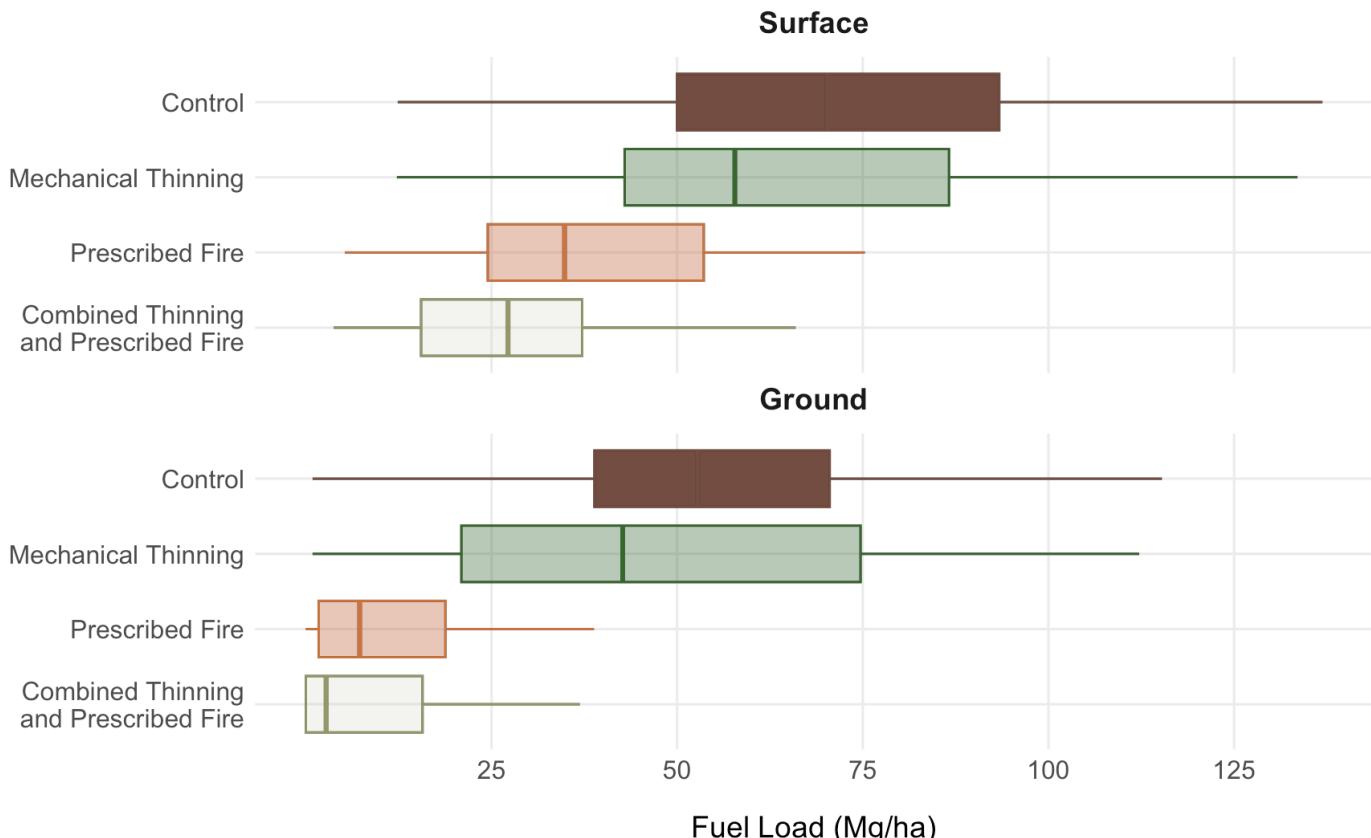
#### ▼ Code

```
# boxplot, using facet_wrap to compare
fuels_duff_final %>%
  ggplot(aes(x = treatment, y = amount_mgha)) +
  # add boxplot fill
  geom_boxplot(aes(fill = treatment, alpha = opacity_val), outliers = FALSE) +
  # add boxplot outline
  geom_boxplot(aes(color = treatment), alpha = 0, outliers = FALSE) +
  # facet by fuel type, create one column so they stack
  facet_wrap(~fuel_type, ncol = 1) +
  # apply color palette to boxplot fill and outline
  scale_fill_manual(values = treatments) +
  scale_color_manual(values = treatments) +
  # add titles
  labs(title = "Ground and Surface Fuel Loads Differ Across Treatments",
       subtitle = "Fuel loads and fire risk are lowest in combined treatment areas",
       caption = "Darker hues indicate higher fire risk. Data from Stephens et al. 2010",
       y = "Fuel Load (Mg/ha)") +
  # wrap long labels
  scale_x_discrete(labels = label_wrap(20)) +
  # adjust y axis breaks
  scale_y_continuous(breaks = c(25, 50, 75, 100, 125),
                     minor_breaks = NULL) +
  theme_minimal() +
  # switch axes
  coord_flip()
```



## Ground and Surface Fuel Loads Differ Across Treatments

Fuel loads and fire risk are lowest in combined treatment areas



Darker hues indicate higher fire risk. Data from Stephens et al. 2023.

## B. Stand density index

Show how plots of different treatments fall into different SDI zones (measure of competition between trees, greater SDI = greater competition and greater chance of mortality)

### ▼ Code

```
# Process the data to calculate % of plots, for that treatment, in each SDI zone
sdi_clean %>%

# first, count number of plots in each treatment type and SDI benchmark zone
group_by(treatment, sdi_zone_long, opacity_val) %>%
  summarize(count_treatment_zone = n()) %>%
  ungroup() %>%

# second, add the counts to get the total number of plots in each treatment type
group_by(treatment) %>%
  mutate(count_treatment = sum(count_treatment_zone),
```

```
# third, divide the number of plots per SDI zone by the number of total
pct_zone_per_treatment = count_treatment_zone/count_treatment) %>%

# set up ggplot
ggplot(aes(x = treatment, y = pct_zone_per_treatment, fill = treatment)) +

# create bar chart, with % as the length and SDI benchmark zone as the fill
geom_col(aes(alpha = opacity_val), position = position_stack(reverse = TRUE)) -

# wrap long labels
scale_x_discrete(labels = label_wrap(20)) +

# add percent labels
scale_y_continuous(labels = scales::percent_format(accuracy = 1)) +

# add fill colors
scale_fill_manual(values = treatments) +

# add title, subtitle, and y axis label
labs(title = "Percent of Treatment Plots that Fall Within Each Density Zone",
     subtitle = "Almost 75% of control plots face overcrowding and increased mo",
     caption = "Darker hues indicate higher density. Data from Stephens et al.",
     y = "Percent of Plots",
     alpha = "Forest Density:") +

# remove legend for fill colors
guides(fill = "none") +

# change legend text labels
scale_alpha_manual(
  values = c(1, 0.8, 0.6, 0.4),
  labels = c("Extremely High", "High", "Moderate", "Low")) +

coord_flip() +

# apply themeing
theme_classic() +
theme(
  # move legend
  legend.position = "top",
  # remove axis titles
  axis.title = element_blank(),
  #adjust title, subtitle, axis text
  plot.title = element_text(face = "bold",
                            size = 16,
                            hjust = 0.5,
```

```

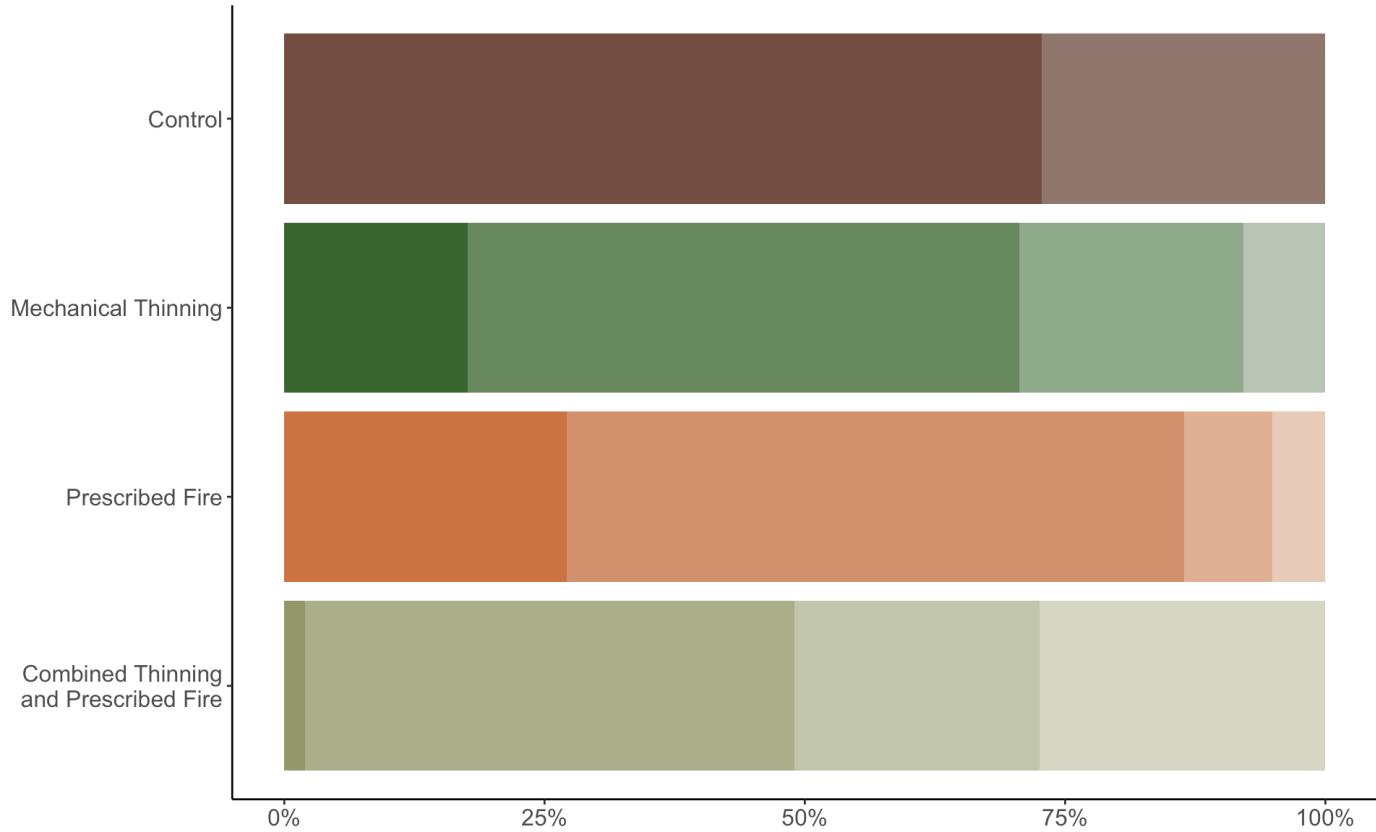
margin = margin(10, 0, 10, 0)),
plot.subtitle = element_text(size = 14,
                             hjust = 0.5,
                             margin = margin(0, 0, 10, 0)),
axis.text = element_text(size = 12),
legend.text = element_text(size = 12,
                           color = "#484848"),
legend.title = element_text(size = 14),
# adjust caption
plot.caption = element_text(face = "italic",
                            color = "#484848",
                            size = 11,
                            hjust = 1,
                            margin = margin(20, 0, 10, 0))
)

```

### Percent of Treatment Plots that Fall Within Each Density Zone

Almost 75% of control plots face overcrowding and increased mortality risk

Forest Density:  Extremely High  High  Moderate  Low



*Darker hues indicate higher density. Data from Stephens et al. 2023.*

## C. Stable live tree carbon

Compare stable live tree carbon (that which is expected to survive after a wildfire event) with existing carbon stocks

▼ Code

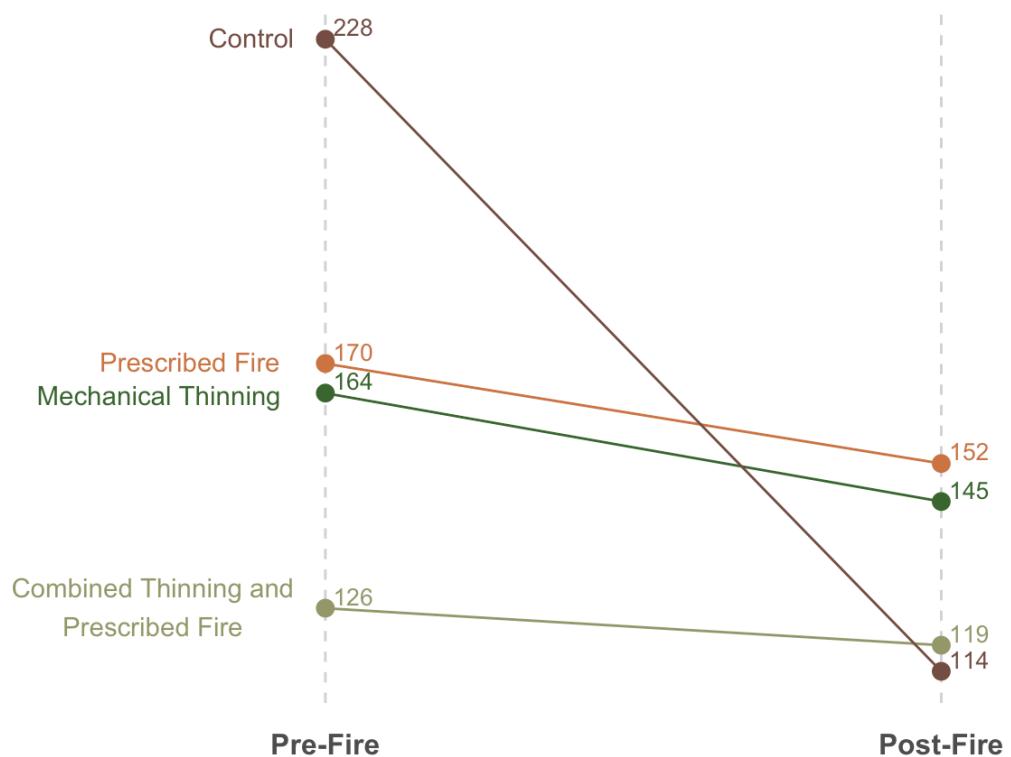
```
# first, calculate the mean stlc and mean existing carbon for each treatment type
stlc_clean %>%
  group_by(treatment) %>%
  summarize(mean_carbon_mgha = mean(carbon_mgha),
            mean_stable_ltc_mgha = mean(stable_ltc_mgha)) %>%
  pivot_longer(mean_carbon_mgha:mean_stable_ltc_mgha,
               names_to = "carbon_type",
               values_to = "carbon_mgha") %>%

# now create slope chart
ggplot(aes(x = carbon_type, y = carbon_mgha, color = treatment, group = treatment))
# add vertical lines as background
geom_vline(xintercept = 1, color = "lightgray", linetype = "dashed") +
  geom_vline(xintercept = 2, color = "lightgray", linetype = "dashed") +
  geom_point(size = 3) +
# add lines connecting the points
  geom_line() +
  # add number labels to each line (showing carbon in mg/ha)
  geom_text(aes(label = round(carbon_mgha, 0)),
            hjust = -0.2,
            vjust = -0.2,
            size = 3.5) +
  # add annotations to label each line with the treatment group name
  annotate(
    geom = "text",
    x = 0.88, y = 228,
    label = "Control",
    color = "#765043FF") +
  annotate(
    geom = "text",
    x = 0.78, y = 170,
    label = "Prescribed Fire",
    color = "#CC7540FF") +
  annotate(
    geom = "text",
    x = 0.73, y = 164,
    label = "Mechanical Thinning",
    color = "#39692FFF") +
```

```
annotate(
  geom = "text",
  x = 0.72, y = 126,
  label = "Combined Thinning and\nPrescribed Fire",
  color = "#979A6BFF") +
  
# apply color palette based upon treatment
scale_color_manual(values = treatments) +
  
# change x axis labels
scale_x_discrete(labels = c("Pre-Fire", "Post-Fire")) +
  
labs(title = "Carbon Stocks in Mg/ha Before and After a Simulated Fire",
  subtitle = "Control plots risk losing the greatest amount of carbon from \",
  caption = "Data from Stephens et al. 2023.") +
  
theme(
  # remove legend, gridlines, axis lines, axis titles
  legend.position = "none",
  panel.background = element_blank(),
  axis.text.y = element_blank(),
  axis.ticks = element_blank(),
  axis.title = element_blank(),
  #adjust title, subtitle, and axis text
  plot.title = element_text(face = "bold",
    size = 16,
    hjust = 0.5,
    margin = margin(10, 0, 10, 0)),
  plot.subtitle = element_text(size = 14,
    hjust = 0.5,
    margin = margin(0, 0, 20, 0)),
  axis.text.x = element_text(face = "bold",
    size = 12,
    margin = margin(10, 0, 0, 0)),
  # adjust caption
  plot.caption = element_text(face = "italic",
    color = "#484848",
    size = 11,
    hjust = 0.7,
    margin = margin(20, 0, 10, 0)))
)
```

## Carbon Stocks in Mg/ha Before and After a Simulated Fire

Control plots risk losing the greatest amount of carbon from wildfires



Data from Stephens et al. 2023.

## Part 7: Reflection Questions

1. What challenges did you encounter or anticipate encountering as you continue to build / iterate on your visualizations in R? If you struggled with mocking up any of your three visualizations (from #6, above), describe those challenges here.

One challenge for me has been effectively using color in my visualizations. At first, I planned to have each viz have its own color scheme (though all be part of the same color palette). After working with the graphs a bit more, I've switched it so that each treatment group has its own color, which hopefully provides some continuity throughout the infographic. However, that means that I have to rely on other visual tools to convey the main messages of my graphics (shape length, opacity, point locations). I think once I put all these graphs together into one layout it'll be easier to tell whether the color is helping or hindering the overall message.

Another challenge has been the use of opacity to convey different variable values in my first two graphs. I really liked how the alien graphics from HW1 employed opacity to add

additional variation within a confined palette. However, in the first graph (A) I currently can't get the values I set (which should be from 0.5 to 0.8) to actually be implemented in my plot. Instead, it seems like the highest value is always 1, and I cannot tell what the others get set to. For the second graph (B), I was eventually able to get the opacity values to work how I wanted in the chart, but I don't necessarily love the grayscale of the legend. I'm not sure what other option I have there, though.

In general, just the fine-tuning of aesthetics has been challenging because it can be pretty time consuming, and often I need to render the whole file just to see how one position or size change on a graph now looks.

2. *What ggplot extension tools / packages do you need to use to build your visualizations? Are there any that we haven't covered in class that you'll be learning how to use for your visualizations?*

Beyond ggplot, I've only used the `palleteer` and `scales` packages to further customize my visualizations. I plan to use Affinity Designer to actually create my final infographic, so I don't think I will need many other ggplot extension tools.

3. *What feedback do you need from the instructional team and / or your peers to ensure that your intended message is clear?*

Is there any jargon I'm using that I could eliminate entirely? If/where I do have to use jargon, do my explanations for it make sense? (this will be more applicable once I've actually written up some annotations to add to my infographic)

Do the colors in my graphic enhance understanding, or make it more confusing? If confusing, how could I make it more clear?

What is a viewer's current takeaway from my planned infographic? How can I best emphasize the main points using annotations, without overwhelming the viewer with text?