Exploratory Data Analysis

Grayson White

Data wrangling:

```r
# Import data
spatial <- read_csv("../data/plot_level/plt_spatial.csv")
response <- read_csv("../data/plot_level/plot_response.csv")

# Join data
## Keep only observations in both `spatial` and `response`
dat <- inner_join(spatial, response,
                  by = c("PLT_CN" = "PLT_CN",
                         "INVYR" = "INVYR"))

# Create columns for province, sections, and subsections
dat <- dat %>%
  mutate(
    subsection = ECOSUBCD.x,
    section = str_remove_all(ECOSUBCD.x, "[:lower:]"),
    province = str_sub(section, end = -2)
  )

# Select small subset of columns to work with for this EDA
dat_small <- dat %>%
  select(PLT_CN, INVYR, PLOT.x, LON_PUBLIC.x, LAT_PUBLIC.x, LON_PUBLIC.y, LAT_PUBLIC.y,
         ELEV_PUBLIC.x, ELEV_PUBLIC.y, forgrp, forprob, nlcd11, demLF, evtLF, forbio,
         BALIVE_TPA, CNTLIVE_TPA, BIOLIVE_TPA, VOLNLIVE_TPA, subsection, section, province)

# Check if a couple of columns are the same
all.equal(dat_small$LON_PUBLIC.x, dat_small$LON_PUBLIC.y)
```

```
## [1] TRUE
```

```r
all.equal(dat_small$LAT_PUBLIC.x, dat_small$LAT_PUBLIC.y)
```

```
## [1] TRUE
```

```r
all.equal(dat_small$ELEV_PUBLIC.x, dat_small$ELEV_PUBLIC.y)
```

```
## [1] TRUE
```

```r
# Remove redundent columns, rename columns for ease of use
dat_small <- dat_small %>%
  select(-LON_PUBLIC.y, -LAT_PUBLIC.y, -ELEV_PUBLIC.y) %>%
  rename(PLOT = PLOT.x,
         LON_PUBLIC = LON_PUBLIC.x,
         LAT_PUBLIC = LAT_PUBLIC.x,
         ELEV_PUBLIC = ELEV_PUBLIC.x)
```

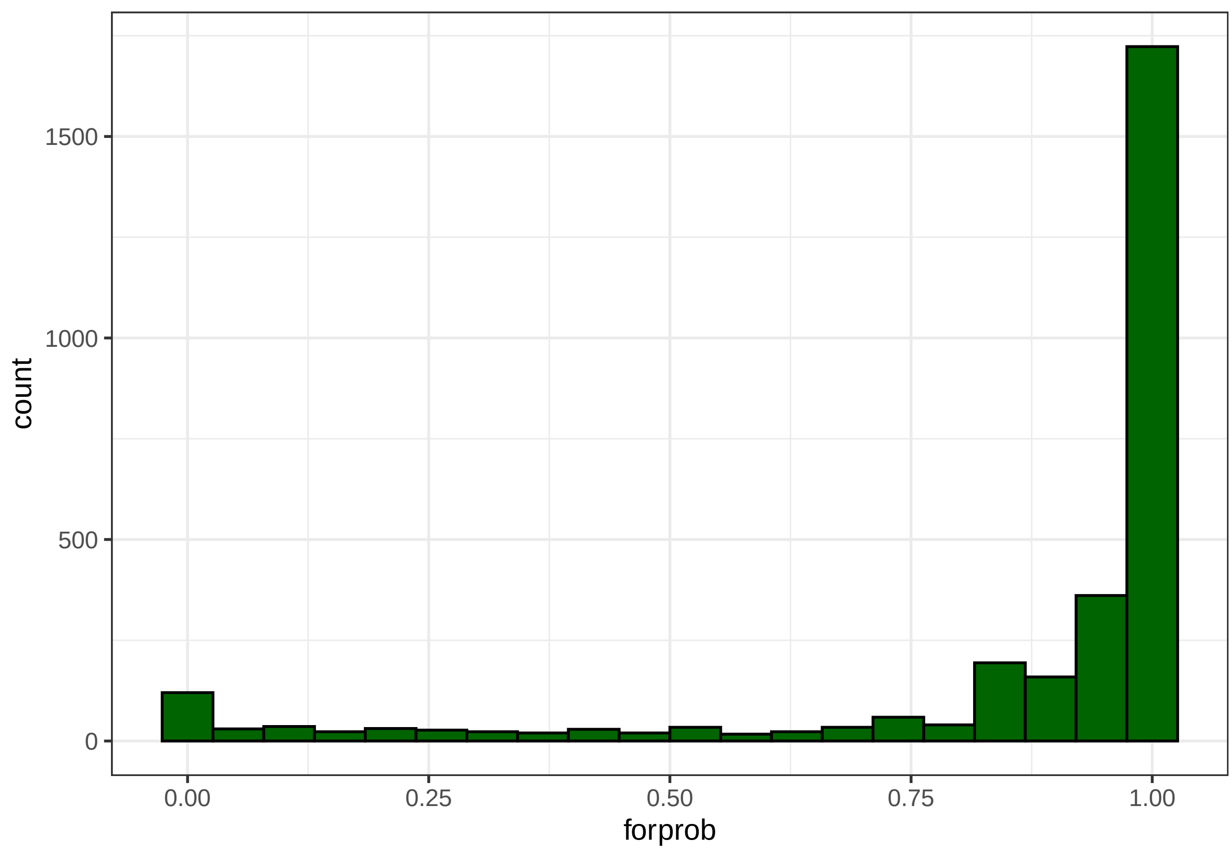
Exploratory Data Analysis

I will first look into the province `M333`, which is the Northern Rocky Mountain Forest. This province has a maritime-influenced cool temperate climate with warm, dry summers and cold, moist winters with heavy snowfall. Small areas of glaciers occur near the Canadian border. High-elevation, high-relief mountains are the main landforms.

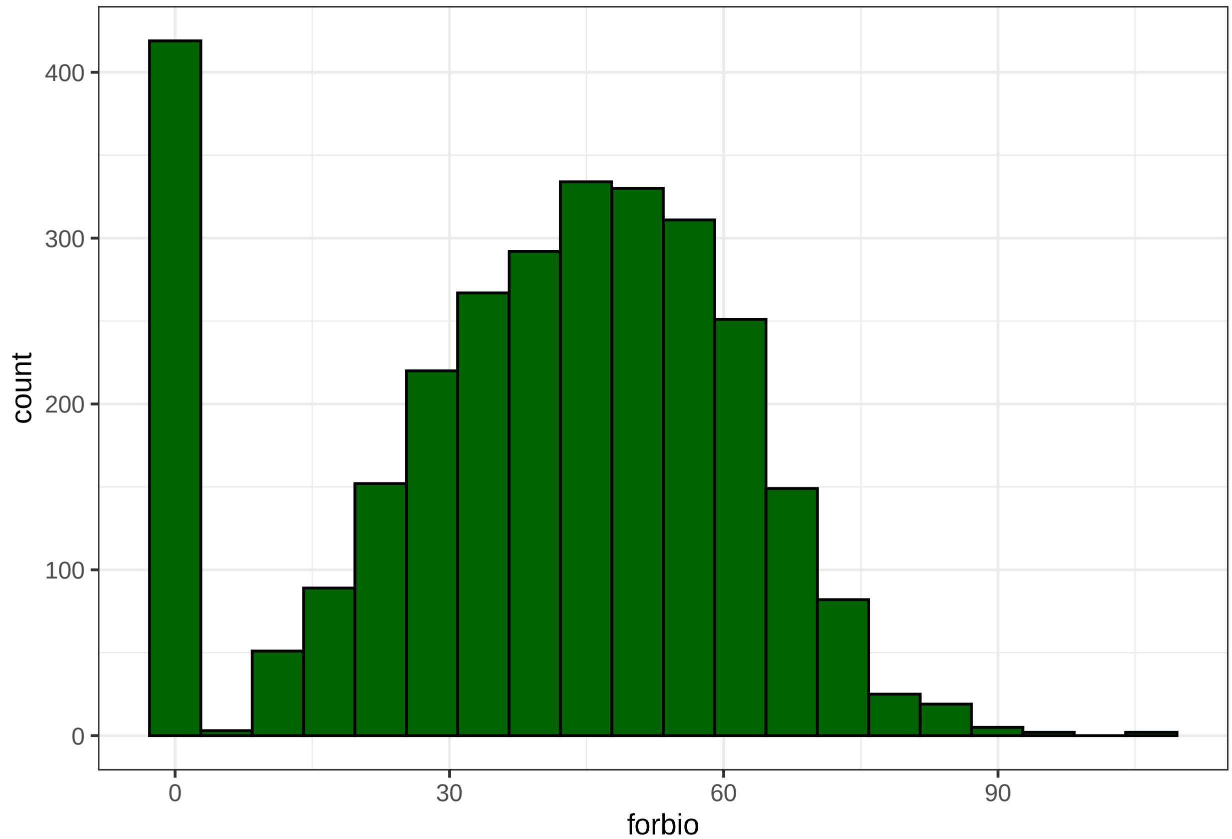
```
# Create dataframe
north_rocky <- dat_small %>%
  filter(province == "M333")

# Summary stats
north_rocky %>%
  summarize(
    mean_forprob = mean(forprob),
    mean_forbio = mean(forbio)
  )
```

```
## # A tibble: 1 x 2
##   mean_forprob mean_forbio
##          <dbl>       <dbl>
## 1        0.855        39.1
```

```
# Distribution of variables
ggplot(north_rocky,
       aes(x = forprob)) +
  geom_histogram(bins = 20, fill = "darkgreen",  color = "black") +
  theme_bw()
```



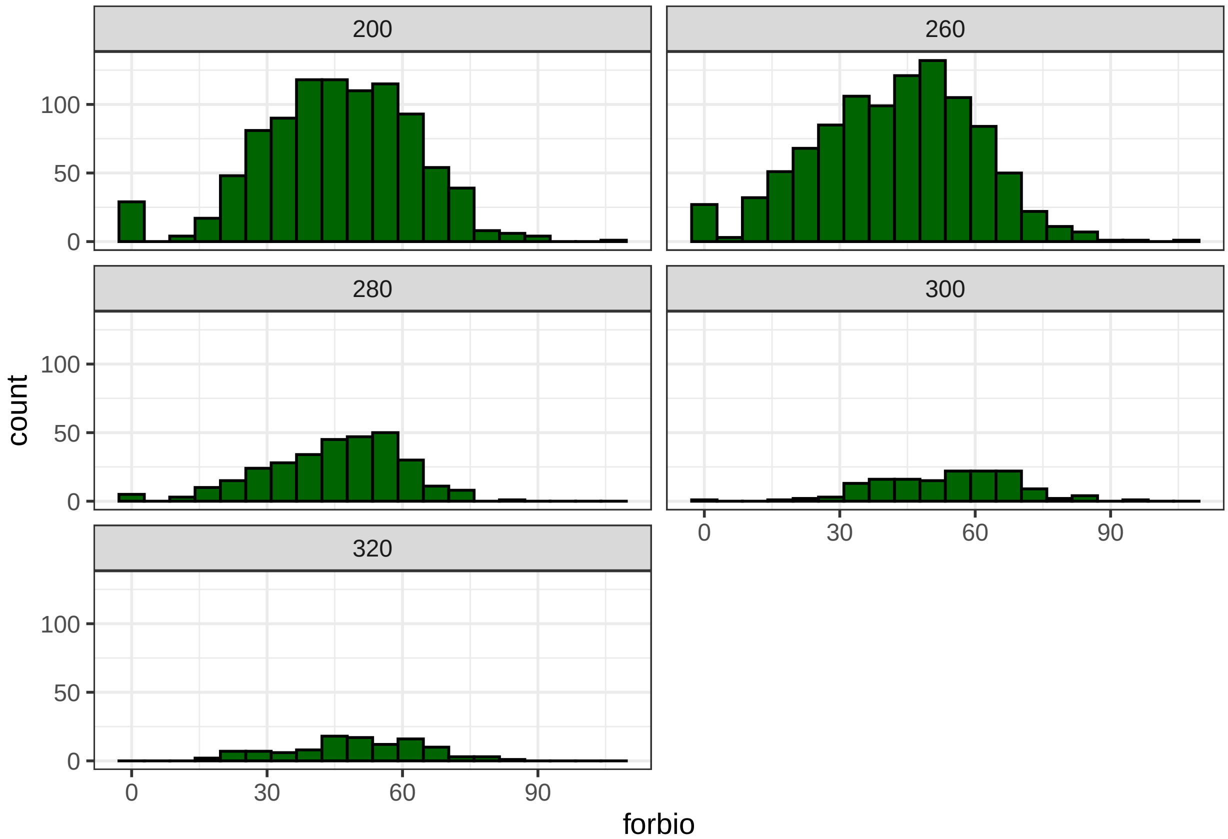
```r
ggplot(north_rocky,
       aes(x = forbio)) +
  geom_histogram(bins = 20, fill = "darkgreen",  color = "black") +
  theme_bw()
```




```r
# forest biomass, by forest group

## filter to groups with greater than 100 observations
forgrp_big <- north_rocky %>%
  group_by(forgrp) %>%
  summarize(count = n()) %>%
  filter(count > 100 & forgrp != 0) %>%
  pull(forgrp)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
north_rocky %>%
  filter(forgrp %in% forgrp_big) %>%
ggplot(aes(x = forbio)) +
  geom_histogram(bins = 20, fill = "darkgreen",  color = "black") +
  theme_bw() +
  facet_wrap(~forgrp, nrow = 3)
```



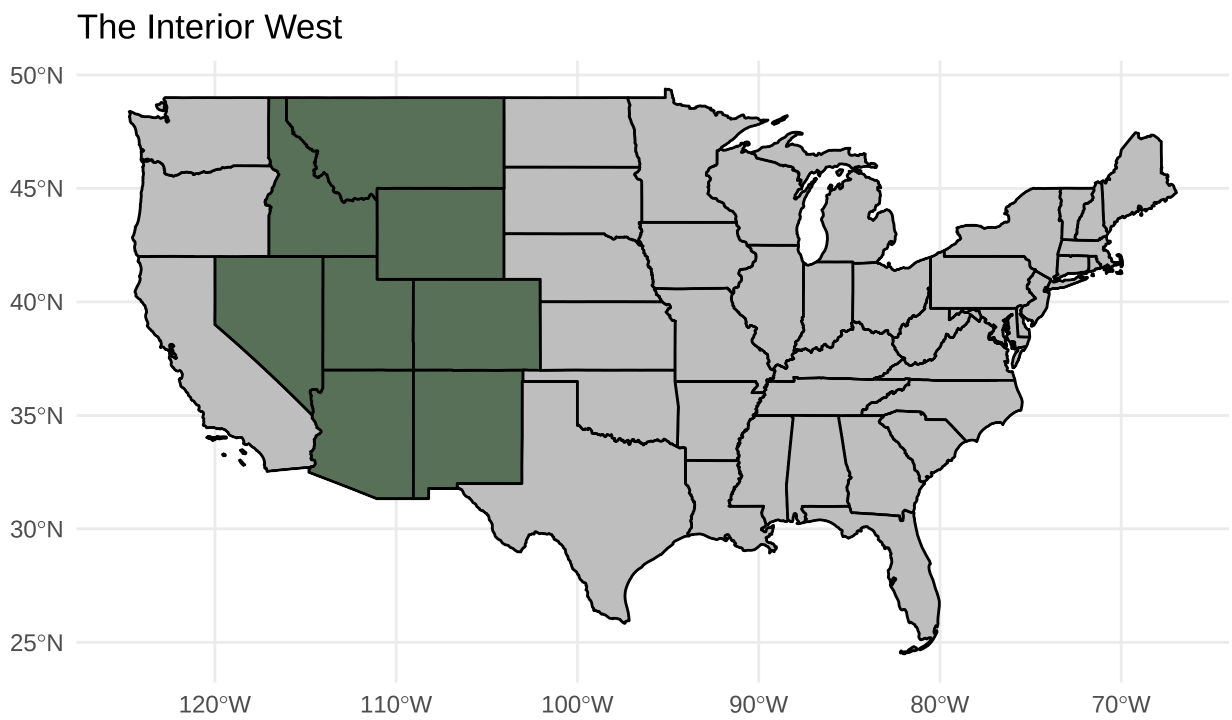
Here we go, time to make a map :-)

```
library(sf)
```

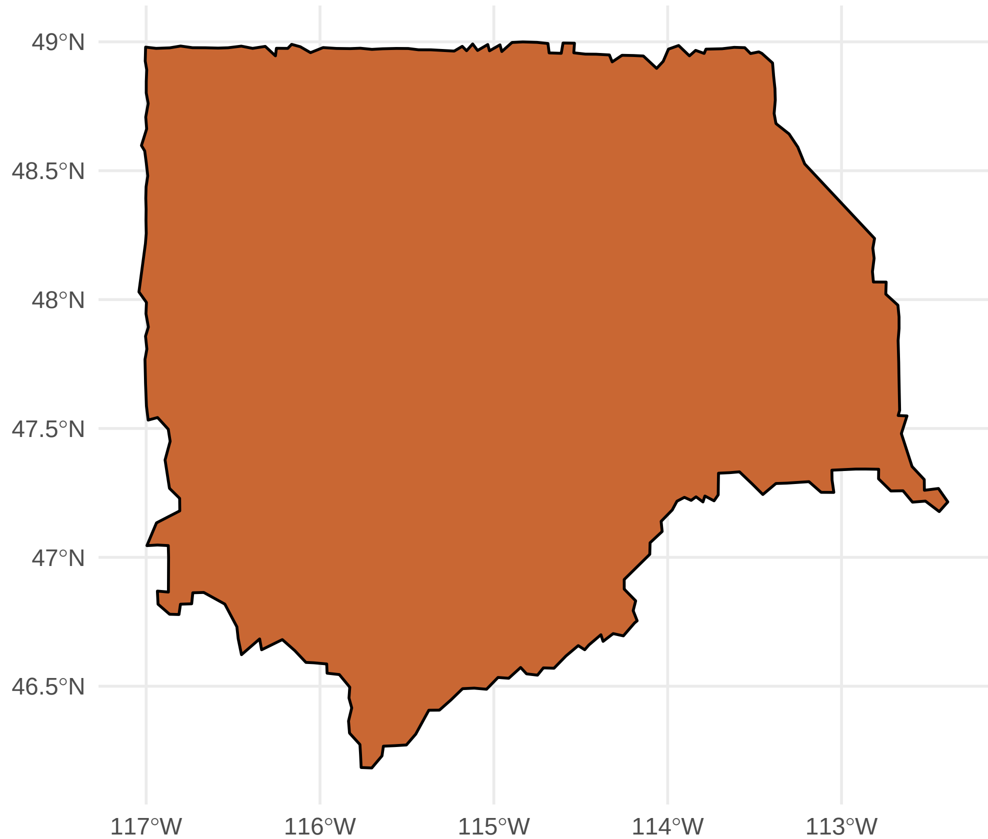
```
## Linking to GEOS 3.7.2, GDAL 2.4.2, PROJ 5.2.0
```

```
library(USAboundaries)
`%ni%` <- Negate(`%in%`)

interior_west <- c("AZ", "CO", "ID", "MT", "NV", "NM", "UT", "WY")

states <- data.frame(state.abb) %>%
  filter(state.abb %ni% interior_west & state.abb %ni% c("AK", "HI")) %>%
  pull()


# The interior west plotted in green on the USA
ggplot(data = north_rocky) +
  geom_sf(data = us_boundaries(type = "state",
                               states = interior_west),
          fill = "#597058",
          color = "black") +
  geom_sf(data = us_boundaries(type = "state",
                               states = states),
          fill = "grey",
          color = "black") +
  theme_minimal() +
```

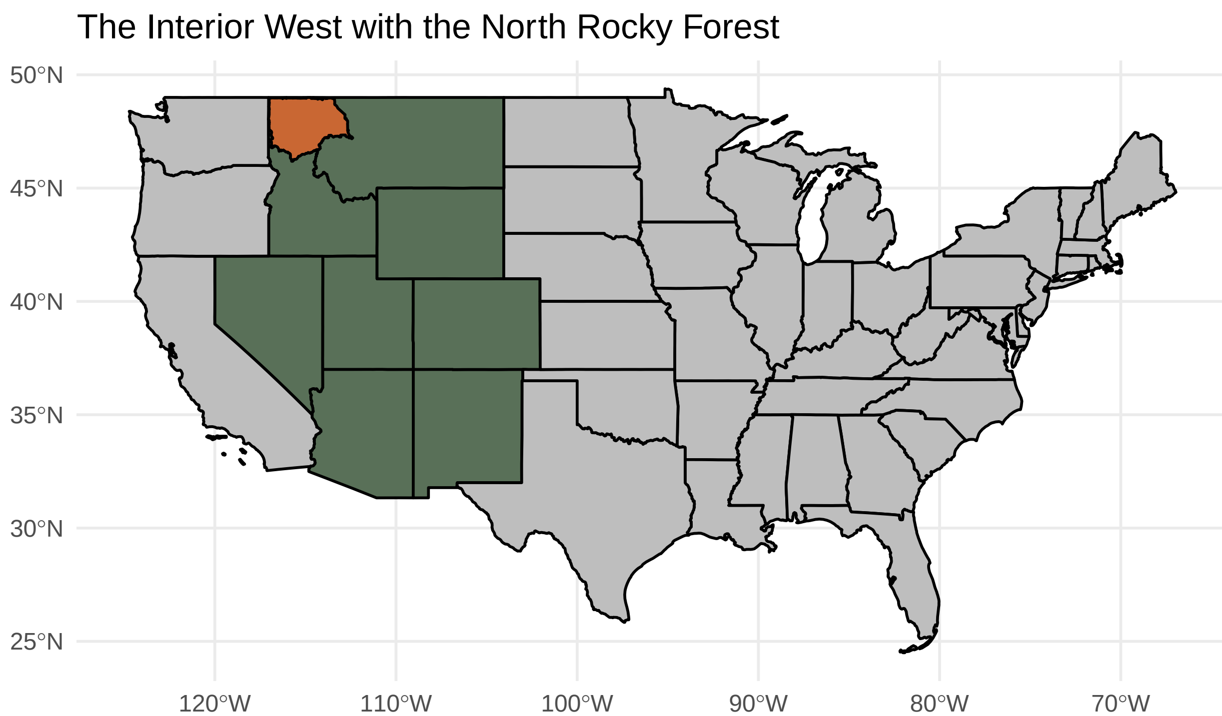
```
  labs(
    title = "The Interior West"
  )
```




```
# Add the north rocky forest onto the interior west map
library(concaveman)

points <- north_rocky %>%
  st_as_sf(coords = c("LON_PUBLIC", "LAT_PUBLIC"), crs = 4326)

forest <- concaveman(points)

ggplot(data = north_rocky) +
  geom_sf(data = forest,
          fill = "#C96733",
          color = "black") +
  theme_minimal()
```

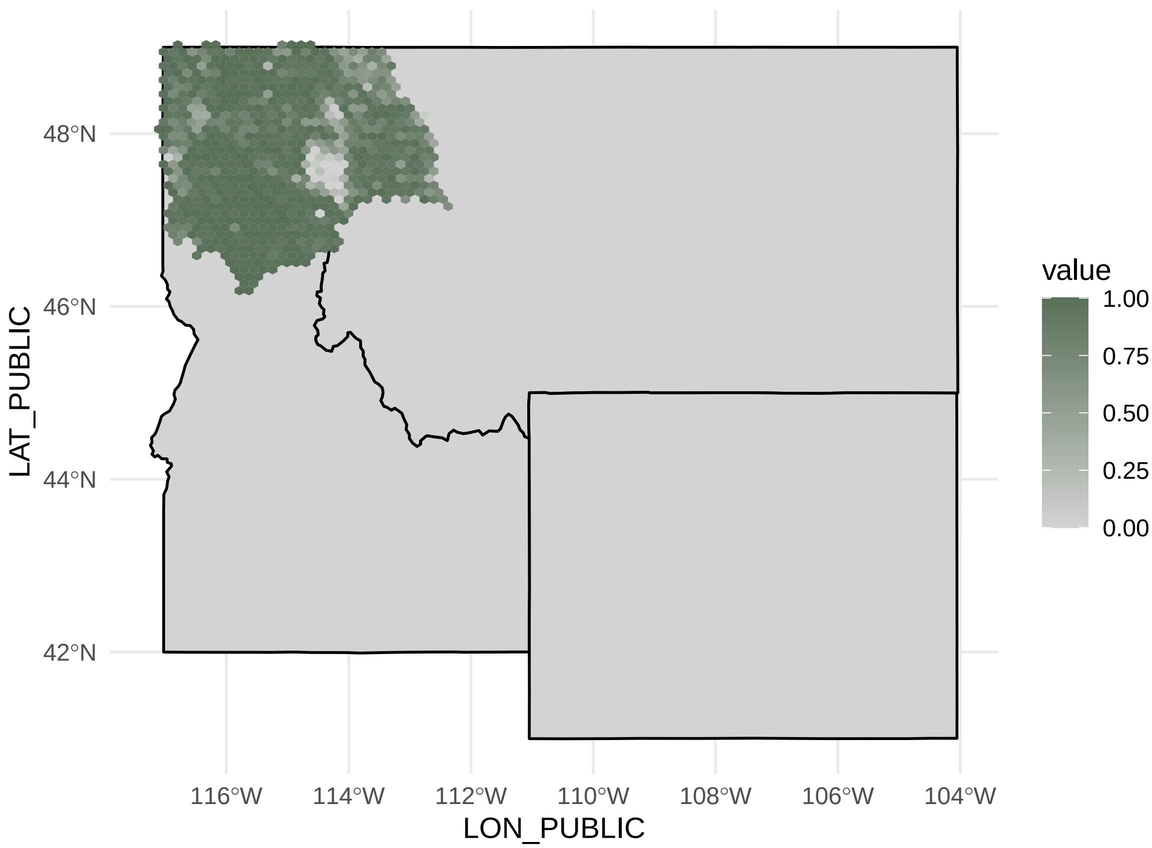



```r
ggplot(data = north_rocky) +
  geom_sf(data = us_boundaries(type = "state",
                               states = interior_west),
          fill = "#597058",
          color = "black") +
  geom_sf(data = us_boundaries(type = "state",
                               states = states),
          fill = "grey",
          color = "black") +
  geom_sf(data = forest,
          fill = "#C96733",
          color = "black") +
  theme_minimal() +
  labs(
    title = "The Interior West with the North Rocky Forest"
  )
```

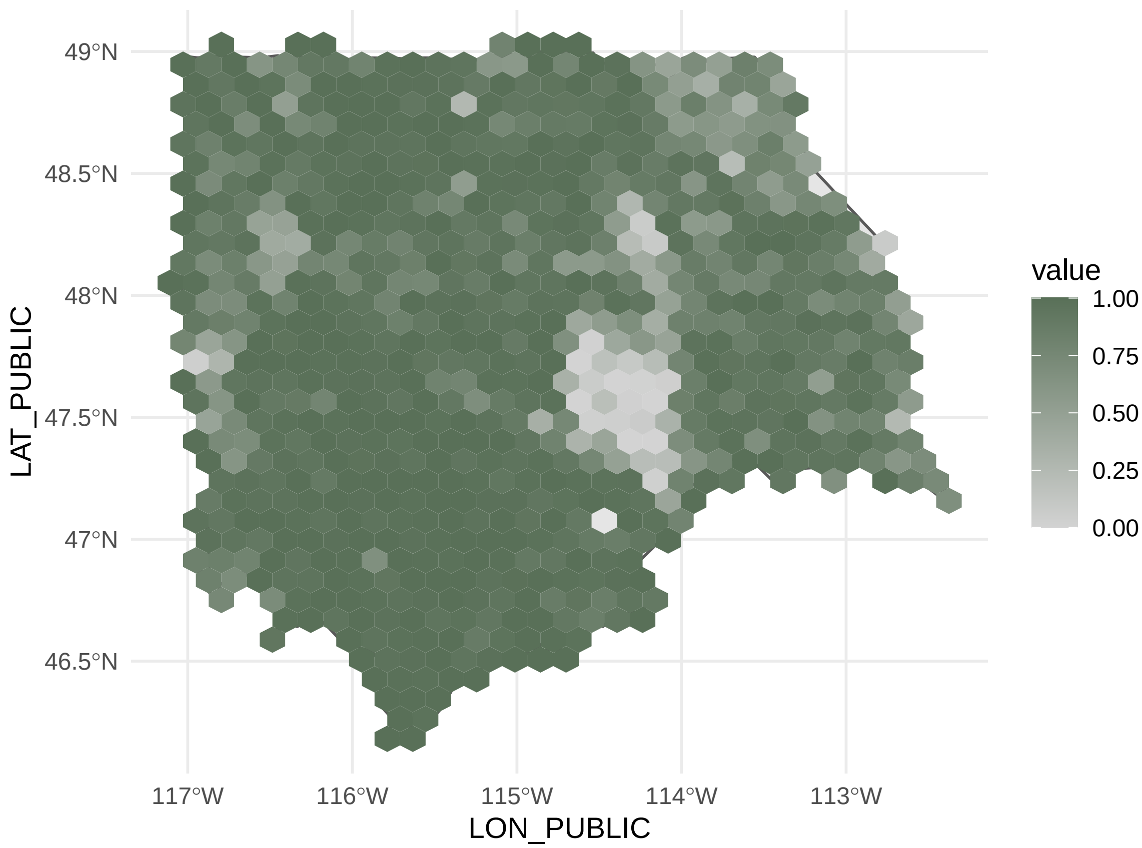


Time to use the data on these nice maps:

```
ggplot(data = north_rocky) +
  geom_sf(data = us_boundaries(type = "state",
                               states = c("ID", "MT", "WY")),
          fill = "lightgrey",
          color = "black") +
  # geom_sf(data = forest,
  #         fill = "#C96733",
  #         color = "black") +
  stat_summary_hex(data = north_rocky,
                fun = function(x) mean(x),
          aes(x = LON_PUBLIC,
              y = LAT_PUBLIC,
              z = forprob)) +
  scale_fill_gradient(low = "lightgrey", high = "#597058") +
  theme_minimal()
```



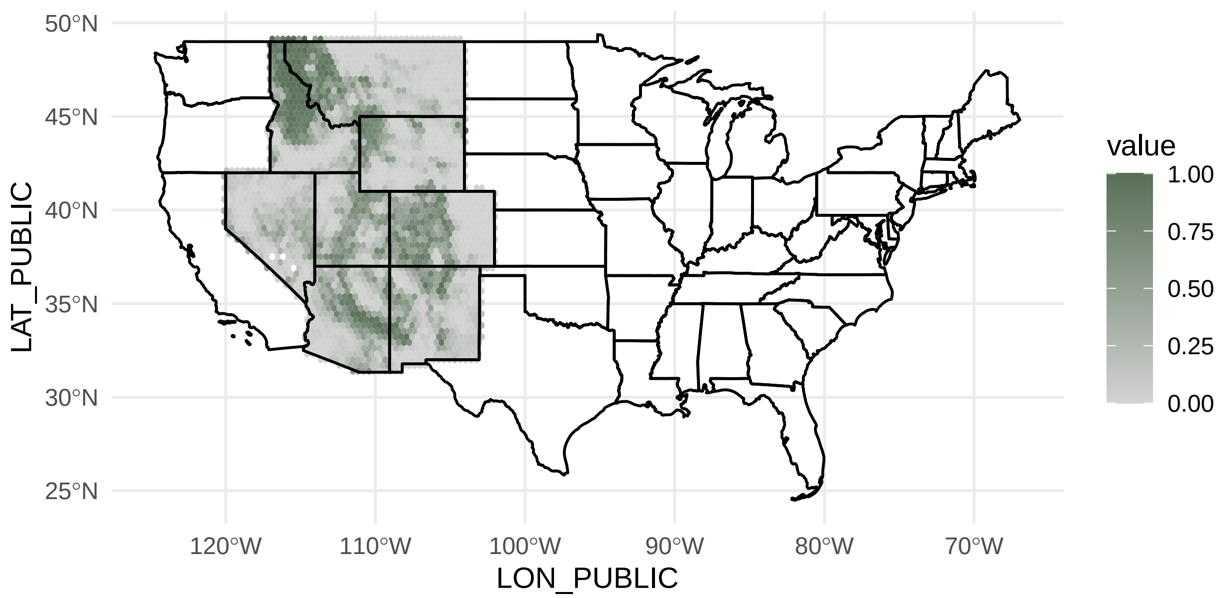

```r
ggplot(data = north_rocky) +
  geom_sf(data = forest) +
  stat_summary_hex(data = north_rocky,
                   fun = function(x) mean(x),
          aes(x = LON_PUBLIC,
              y = LAT_PUBLIC,
              z = forprob)) +
  theme_minimal() +
  scale_fill_gradient(low = "lightgrey", high = "#597058")
```



Let's take a step back and look at probabilty of forest over a larger area:

```
ggplot(data = north_rocky) +
  stat_summary_hex(
    data = dat_small,
    fun = function(x)
      mean(x),
    aes(x = LON_PUBLIC,
        y = LAT_PUBLIC,
        z = forprob),
    bins = 50
  ) +
  geom_sf(
    data = us_boundaries(type = "state",
                         states = c(states, interior_west)),
    fill = NA,
    color = "black"
  ) +
  scale_fill_gradient(low = "lightgrey", high = "#597058") +
  theme_minimal()
```

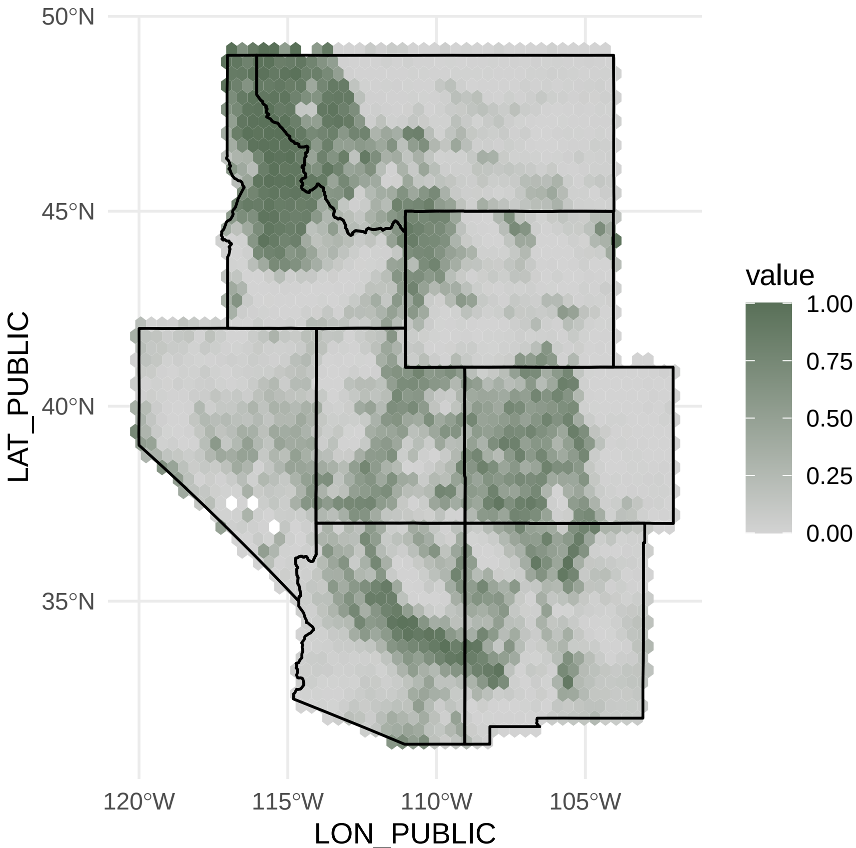
```
## Warning: Removed 1 rows containing non-finite values (stat_summary_hex).
```



Zooming into the interior west:

```r
ggplot(data = north_rocky) +
  stat_summary_hex(
    data = dat_small,
    fun = function(x)
      mean(x),
    aes(x = LON_PUBLIC,
        y = LAT_PUBLIC,
        z = forprob),
    bins = 50
  ) +
  geom_sf(
    data = us_boundaries(type = "state",
                         states = c(interior_west)),
    fill = NA,
    color = "black"
  ) +
  scale_fill_gradient(low = "lightgrey", high = "#597058") +
  theme_minimal()
```

```
## Warning: Removed 1 rows containing non-finite values (stat_summary_hex).
```



Now, let's explore areas that are likely (>0.75) forests.

```
forest75 <- dat_small %>%
  filter(forprob > 0.75)

# summary stats by province

forest75_summaries <- forest75 %>%
  group_by(province) %>%
  summarize(
    mean_basal_area = mean(BALIVE_TPA),
    mean_CNT = mean(CNTLIVE_TPA),
    mean_BIO = mean(BIOLIVE_TPA),
    mean_VOLN = mean(VOLNLIVE_TPA),
    mean_biomass = mean(forbio)
  )
```

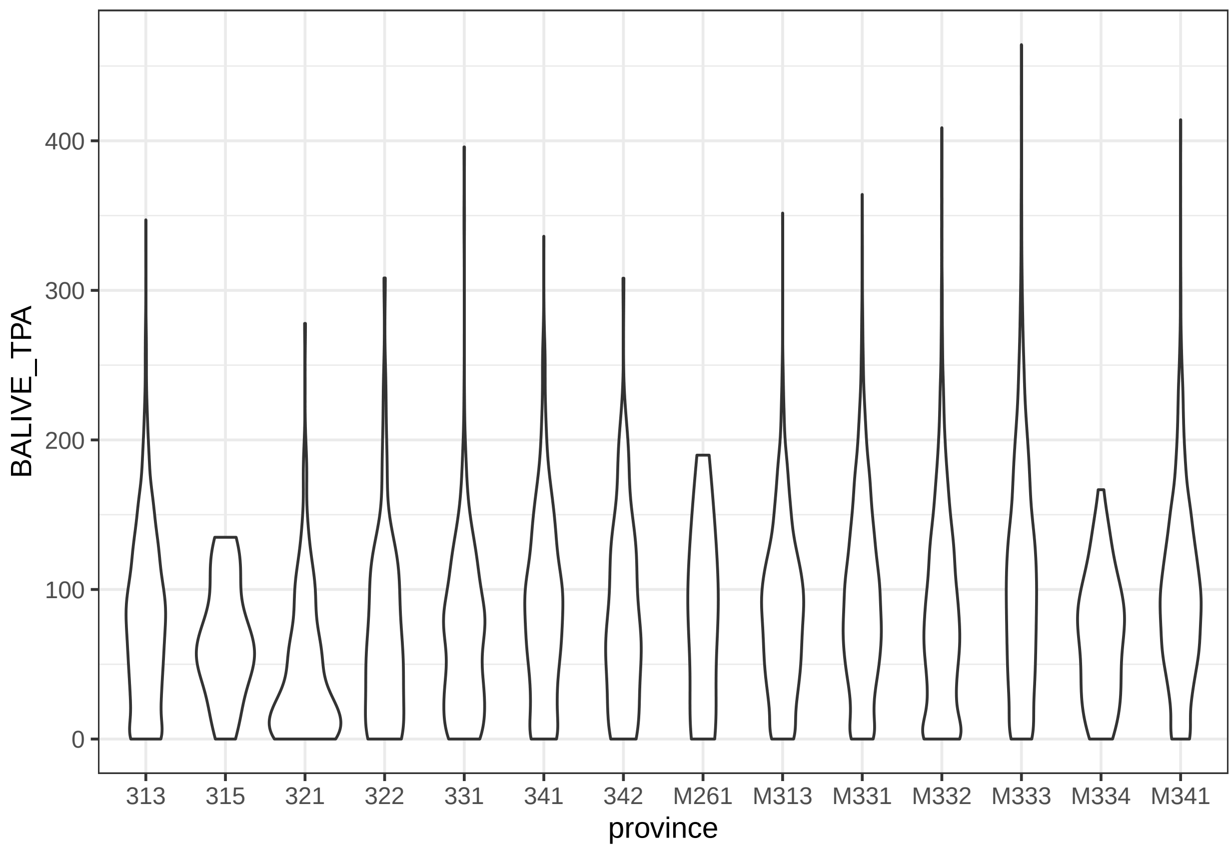
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
forest75_summaries
```

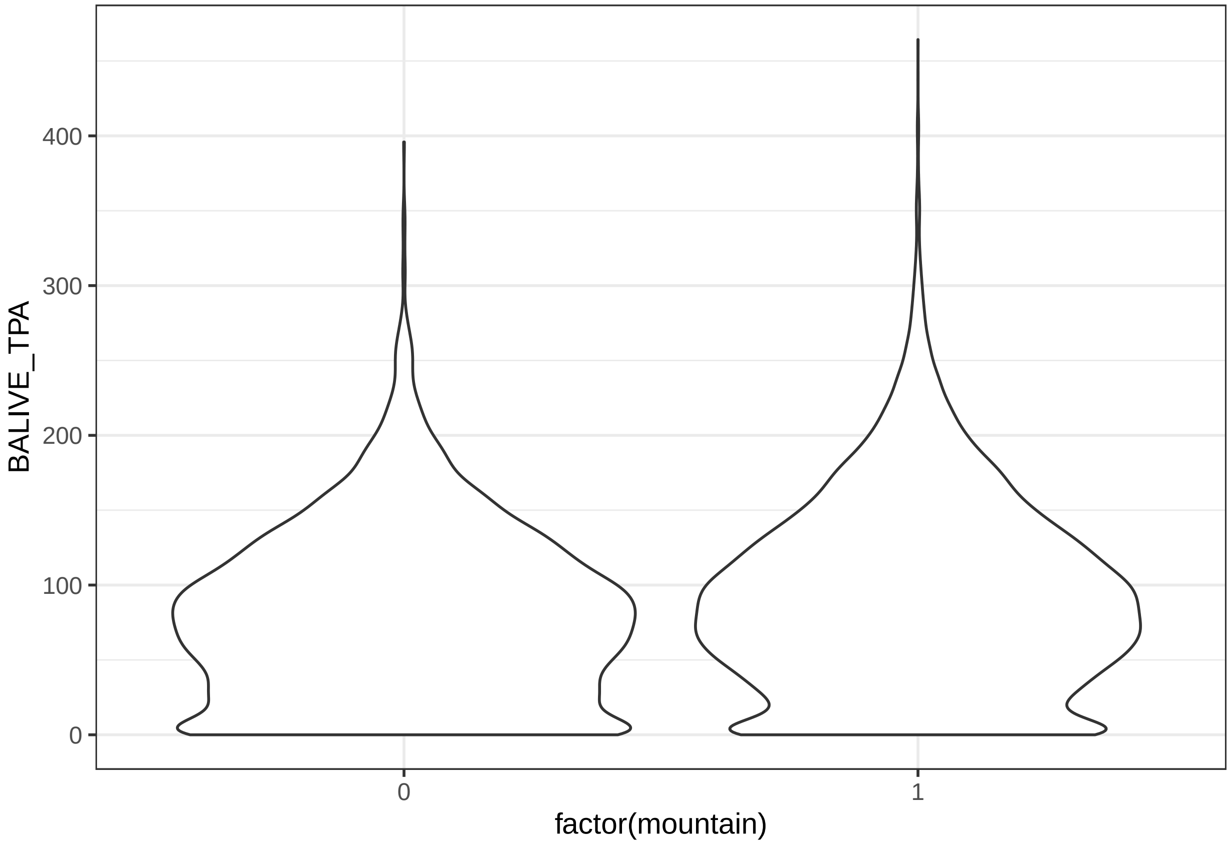
```
## # A tibble: 14 x 6
##    province mean_basal_area mean_CNT mean_BIO mean_VOLN mean_biomass
##    <chr>              <dbl>    <dbl>    <dbl>     <dbl>        <dbl>
##  1 313                 79.5     296.    15.7       846.         12.8
##  2 315                 65.6     211.     7.93      491.          6.23
##  3 321                 44.9     203.     7.97      404.         13.3
##  4 322                 68.6     186.    10.1       539.         10.1
```

```
##  5 331                  68.3     350.    18.8      1013.         22.0
##  6 341                  86.4     333.    12.9       683.         13.9
##  7 342                  80.4     227.    20.9      1131.         30.2
##  8 M261                 82.2     242.    26.7      1578.         78.9
##  9 M313                 83.9     294.    20.4      1112.         21.9
## 10 M331                 93.1     486.    29.7      1675.         34.3
## 11 M332                 81.8     356.    31.2      1721.         40.4
## 12 M333                104.      473.    45.2      2529.         45.8
## 13 M334                 68.3     308.    26.3      1380.         23.5
## 14 M341                 94.5     341.    15.6       852.         17.2
```

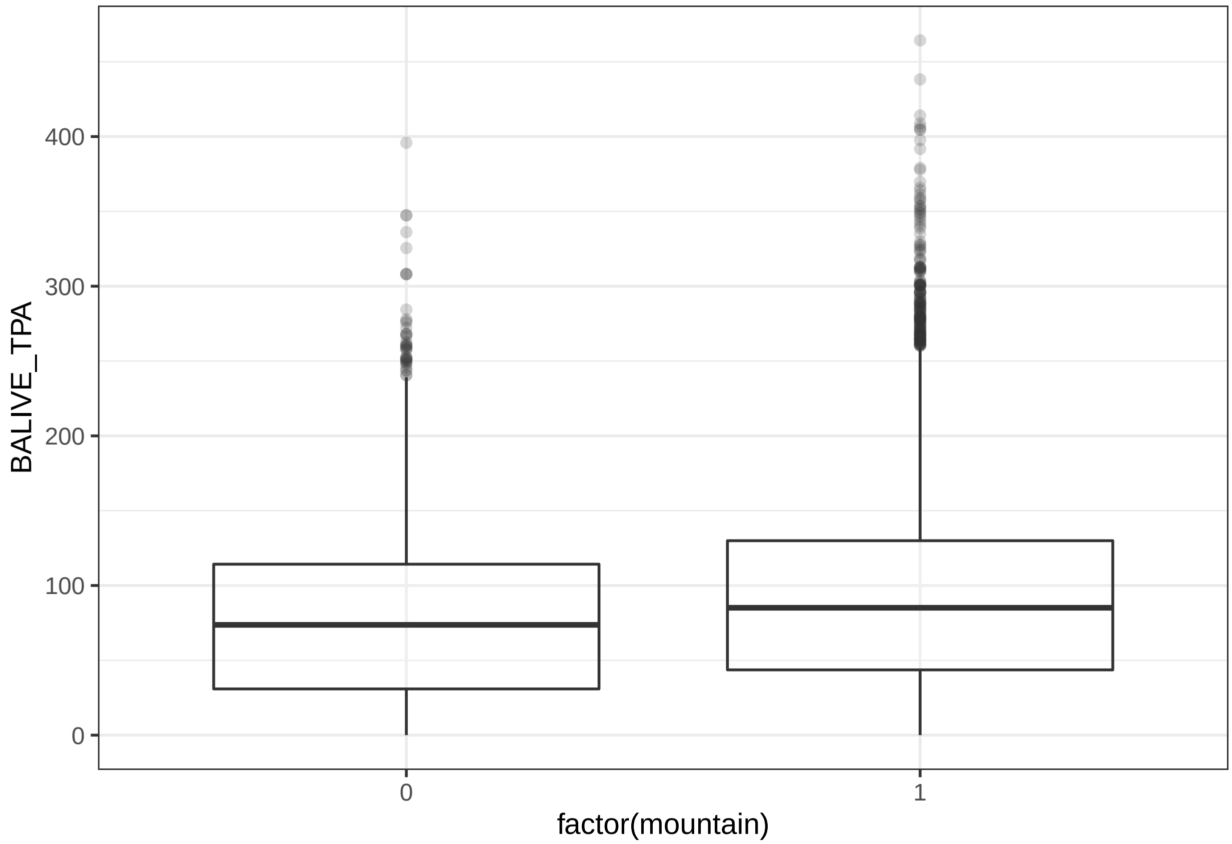
```r
# Learning more about distribution of these variables in likely forests:
## Mountain or not variable:
forest75 <- forest75 %>%
  mutate(
    mountain = case_when(
      province %in% c("M313", "M331", "M341", "M333", "M332", "M261", "M334") ~ 1,
      TRUE ~ 0
    )
  )

## By province
ggplot(forest75,
       aes(y = BALIVE_TPA,
           x = province)) +
  geom_violin() +
  theme_bw()
```



```r
## by mtn vs. not
ggplot(forest75,
       aes(x = factor(mountain),
           y = BALIVE_TPA)) +
  geom_violin() +
  theme_bw()
```




```r
ggplot(forest75,
       aes(x = factor(mountain),
           y = BALIVE_TPA)) +
  geom_boxplot(alpha = 0.2) +
  theme_bw()
```




```
## we could do a t test to see if these are really different
t.test(BALIVE_TPA ~ mountain, data = forest75)
```

```
##
##  Welch Two Sample t-test
##
## data:  BALIVE_TPA by mountain
## t = -12.323, df = 6922, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -15.35693 -11.14163
## sample estimates:
## mean in group 0 mean in group 1
##        78.08654        91.33582
```

```
## do any response variables have very similar means between mountain vs not?
t.test(BIOLIVE_TPA ~ mountain, data = forest75)
```

```
##
##  Welch Two Sample t-test
##
## data:  BIOLIVE_TPA by mountain
## t = -42.617, df = 11870, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -16.40570 -14.96289
## sample estimates:
```

```
## mean in group 0 mean in group 1
##        14.64030        30.32459
```

```r
t.test(CNTLIVE_TPA ~ mountain, data = forest75)
```

```
##
##  Welch Two Sample t-test
##
## data:  CNTLIVE_TPA by mountain
## t = -14.588, df = 6818, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -130.50749  -99.58853
## sample estimates:
## mean in group 0 mean in group 1
##         297.339         412.387
```

```r
t.test(VOLNLIVE_TPA ~ mountain, data = forest75)
```

```
##
##  Welch Two Sample t-test
##
## data:  VOLNLIVE_TPA by mountain
## t = -42.399, df = 12761, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -945.9213 -862.3247
## sample estimates:
## mean in group 0 mean in group 1
##        785.4787       1689.6017
```

```r
### no, these are all pretty different. interesting.
```