# Capstone Data Exploration

*Tom Thorpe*

*April 17, 2018*

## Objective

View different plots of the cleaned Forest Cover data from the previous section to learn more about the data.

Include required libraries.

```
progStart=Sys.time()
print(paste("R script started at",progStart))
```

```
## [1] "R script started at 2018-04-18 14:49:59"
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

Point to data. The forestcover_clean_full.csv is the cleaned data to be graphed.

```
infile="C:/Users/Tom/git/datasciencefoundation/ForestCoverage/forestcover_clean_full2.csv"
#infile="C:/Users/Tom/git/datasciencefoundation/ForestCoverage/forestcover_clean_full_sample2.csv"
out2file="C:/Users/Tom/git/datasciencefoundation/ForestCoverage/forestcover_graph.csv"
#out1file="C:/Users/Tom/git/datasciencefoundation/ForestCoverage/forestcoversmall_clean_full.csv"
#out2file="C:/Users/Tom/git/datasciencefoundation/ForestCoverage/forestcoversmall_clean.csv"

alphaVal<-0.01 # large data
alphaVal<-0.1  # small data
```

Load the data.

```
startTime=Sys.time()
print(paste("Data load started at",startTime))
```

```
## [1] "Data load started at 2018-04-18 14:49:59"
```

```
forestcover <- read.csv(infile,header=TRUE,sep=",") %>% tbl_df()

# Shorten some names
forestcover$ClimateName <- as.character(forestcover$ClimateName)
forestcover$ClimateName[forestcover$ClimateZone == 1] <- "MonLowDry" # was "Mont_LowDry"
forestcover$ClimateName[forestcover$ClimateZone == 2] <- "MonLow" # was "Montane_Low"
forestcover$ClimateName[forestcover$ClimateZone == 3] <- "MonDry" # was "Montane_Dry"
forestcover$ClimateName[forestcover$ClimateZone == 4] <- "Montane" # was "Montane"
```

```
forestcover$ClimateName[forestcover$ClimateZone == 5] <- "M&MDry" # was "Mon&Mon_Dry"
forestcover$ClimateName[forestcover$ClimateZone == 6] <- "MonSubAlp" # was "Mon_SubAlp"
forestcover$ClimateName[forestcover$ClimateZone == 7] <- "SubAlpine" # was "SubAlpine"
forestcover$ClimateName[forestcover$ClimateZone == 8] <- "Alpine" # was "Alpine"
forestcover$ClimateName <- as.factor(forestcover$ClimateName)

endTime=Sys.time()
  print(paste("Data load completed at",endTime))
```

```
## [1] "Data load completed at 2018-04-18 14:50:34"
```

```
  print(paste("Elapsed time=",endTime-startTime,"seconds."))
```

```
## [1] "Elapsed time= 34.4610548019409 seconds."
```

## Data Overview

The forest cover data has a row for each sample representing a 30 meter by 30 meter square area of land. Each cell sample is described by elevation, slope and direction the cell faces, distance to water, roads, and fire and binary columns for wilderness area and soil type. One of 4 possible wilderness areas and one of 40 possible aggregated soil types are set in each row. The predicted variable is the coverage type indicating 1 of 7 possible trees found in the cell sample.

The data is described in detail here: https://archive.ics.uci.edu/ml/machine-learning-databases/covtype/covtype.info. The data names have been abbreviated but can be related to the data descriptions easily.

```
glimpse(forestcover)
```

```
## Observations: 581,012
## Variables: 130
## $ Elev                  <int> 2596, 2590, 2804, 2785, 2595, 2579, 26...
## $ Aspect                <int> 51, 56, 139, 155, 45, 132, 45, 49, 45,...
## $ Slope                 <int> 3, 2, 9, 18, 2, 6, 7, 4, 9, 10, 4, 11,...
## $ H2OHD                 <int> 258, 212, 268, 242, 153, 300, 270, 234...
## $ H2OVD                 <int> 0, -6, 65, 118, -1, -15, 5, 7, 56, 11,...
## $ RoadHD                <int> 510, 390, 3180, 3090, 391, 67, 633, 57...
## $ Shade9AM              <int> 221, 220, 234, 238, 220, 230, 222, 222...
## $ Shade12PM             <int> 232, 235, 238, 238, 234, 237, 225, 230...
## $ Shade3PM              <int> 148, 151, 135, 122, 150, 140, 138, 144...
## $ FirePtHD              <int> 6279, 6225, 6121, 6211, 6172, 6031, 62...
## $ RWwild                <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ NEwild                <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ CMwild                <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ CPwild                <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST01                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST02                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST03                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST04                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST05                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST06                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST07                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST08                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST09                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST10                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST11                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
```

```
## $ ST12                  <int> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST13                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST14                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST15                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST16                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST17                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST18                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,...
## $ ST19                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST20                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST21                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST22                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST23                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST24                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST25                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST26                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST27                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST28                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST29                  <int> 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0,...
## $ ST30                  <int> 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1,...
## $ ST31                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST32                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST33                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST34                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST35                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST36                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST37                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST38                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST39                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ ST40                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ CovType               <int> 5, 5, 2, 2, 5, 2, 5, 5, 5, 5, 5, 2, 2,...
## $ STsum                 <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ Wildsum               <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ Wilderness_Area       <fct> Rawah, Rawah, Rawah, Rawah, Rawah, Raw...
## $ ClimateName           <fct> SubAlpine, SubAlpine, Montane, SubAlpi...
## $ GeoName               <fct> Ign_Meta, Ign_Meta, Ign_Meta, Ign_Meta...
## $ CovName               <fct> Aspen, Aspen, Lodgepole, Lodgepole, As...
## $ SoilType              <int> 29, 29, 12, 30, 29, 29, 29, 29, 29, 29...
## $ ClimateZone           <int> 7, 7, 4, 7, 7, 7, 7, 7, 7, 7, 6, 7, 7,...
## $ GeoZone               <int> 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,...
## $ Montane_low           <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Montane               <int> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,...
## $ SubAlpine             <int> 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ Alpine                <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Dry                   <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Non_Dry               <int> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,...
## $ Alluvium              <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Glacial               <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Sed_mix               <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Ign_Meta              <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ Aquolis_cmplx         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Argiborolis_Pachic    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Borohemists_cmplx     <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Bross                 <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Bullwark              <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
```

```
## $ Bullwark_Cmplx          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Catamount               <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Catamount_cmplx         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cathedral               <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Como                    <int> 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1,...
## $ Cryaquepts_cmplx        <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cryaquepts_Typic        <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cryaquolls              <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cryaquolls_cmplx        <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cryaquolls_Typic        <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cryaquolls_Typic_cmplx  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cryoborolis_cmplx       <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cryorthents             <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cryorthents_cmplx       <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cryumbrepts             <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cryumbrepts_cmplx       <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Gateview                <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Gothic                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Granile                 <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Haploborolis            <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Legault                 <int> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Legault_cmplx           <int> 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1,...
## $ Leighcan                <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Leighcan_cmplx          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Leighcan_warm           <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Moran                   <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Ratake                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Ratake_cmplx            <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Rogert                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,...
## $ Supervisor_Limber_cmplx <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Troutville              <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Unspecified             <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Vanet                   <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Wetmore                 <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Bouldery_ext            <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Rock_Land               <int> 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1,...
## $ Rock_Land_cmplx         <int> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Rock_Outcrop            <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Rock_Outcrop_cmplx      <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Rubbly                  <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Stony                   <int> 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Stony_extreme           <int> 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1,...
## $ Stony_very              <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,...
## $ Till_Substratum         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Spruce_Fir              <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ LodgepolePine           <int> 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 1, 1,...
## $ PonderosaPine           <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Cottonwood_Willow       <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Aspen                   <int> 1, 1, 0, 0, 1, 0, 1, 1, 1, 1, 1, 0, 0,...
## $ DouglasFir              <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ Krummholz               <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
```

## List Data Ranges for Non-Binary Data

List Data Ranges for Non-Binary Data.

```
myranges <- function(name,x) { c(name, min = min(x), mean = mean(x), max = max(x)) }

forestDataRanges <- data.frame("Data"=character(), "min"=double(), "mean"=double(), "max"=double(),
                               stringsAsFactors=FALSE)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("Elev",forestcover$Elev)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("Aspect",forestcover$Aspect)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("Slope",forestcover$Slope)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("H2OHD",forestcover$H2OHD)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("H2OVD",forestcover$H2OVD)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("RoadHD",forestcover$RoadHD)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("FirePtHD",forestcover$FirePtHD)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("Shade9AM",forestcover$Shade9AM)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("Shade12P",forestcover$Shade12PM)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("Shade3PM",forestcover$Shade3PM)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("RWwild",forestcover$RWwild)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("NEwild",forestcover$NEwild)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("CMwild",forestcover$CMwild)
forestDataRanges[nrow(forestDataRanges)+1,] <- myranges("CPwild",forestcover$CPwild)
forestDataRanges
```

```
##        Data  min              mean  max
## 1      Elev 1859  2959.36530054457 3858
## 2    Aspect    0  155.656807432549  360
## 3     Slope    0  14.1037035379648   66
## 4     H2OHD    0  269.428216628916 1397
## 5     H2OVD -173   46.418855376481  601
## 6    RoadHD    0  2350.14661142971 7117
## 7  FirePtHD    0    1980.291226343 7173
## 8  Shade9AM    0  212.146048618617  254
## 9  Shade12P    0  223.318716308785  254
## 10 Shade3PM    0   142.52826275533  254
## 11   RWwild    0 0.448865083681576    1
## 12   NEwild    0 0.051434393781884    1
## 13   CMwild    0 0.436073609495157    1
## 14   CPwild    0 0.063626913041383    1
```

The results show all the data values have reasonable values and there is no missing data. The elevation ranges from 1859 meters (6099 feet) to 3858 meters (12657 feet). These are valid ranges for elevation in the Colorado wilderness areas being sampled, but the rule of thumb for timberline (the maximum elevation for where trees are found) is 11500 feet. It might be interesting to see how accurate predictions are if samples above 11800 feet are removed.

The Aspect which is the compass heading that the terrain faces, ranges from 0 to 360 degrees and is a valid data range. The Slope is the steepness of the terrain with 0 degrees being flat and 90 degrees being vertical. The maximum Slope was found to be 66 degrees which seems logical since trees are not usually seen on near-vertical cliffs. (It's a different story in New Zealand!)

The horizontal distance to the nearest water features, range from 0 to 1397 meters which seems reasonable. The vertical distance to nearest water features, range from -173 to 601 meters which seems reasonable and can be negative since the nearest water may be below the forest cover data sample.

The horizontal distance to the nearest road ranges from 0 to 7117 meters which is reasonable. The horizontal distance to the nearest fire features range from 0 to 7173 meters which is reasonable. The amount of shade

present in a cell sample at 9AM, 12PM and 3PM ranges from 0 (full sun) to 254 (fully shaded).

```
##print(paste("Forest coverage ST__ column deletion started at",startTime))
#forestcover <- forestcover %>% select(-ST01,-ST02,-ST03,-ST04,-ST05,-ST06,-ST07,-ST08,-ST09,-ST10,
#                                      -ST11,-ST12,-ST13,-ST14,-ST15,-ST16,-ST17,-ST18,-ST19,-ST20,
#                                      -ST21,-ST22,-ST23,-ST24,-ST25,-ST26,-ST27,-ST28,-ST29,-ST30,
#                                      -ST31,-ST32,-ST33,-ST34,-ST35,-ST36,-ST37,-ST38,-ST39,-ST40,
#                                      -STsum, -Wildsum
#                                      )

#forestcover <- mutate(forestcover,CovName = "")
#forestcover$CovName[forestcover$CovType == 1] <- "Spruce&Fir"
#forestcover$CovName[forestcover$CovType == 2] <- "Lodgepole"
#forestcover$CovName[forestcover$CovType == 3] <- "Ponderosa"
#forestcover$CovName[forestcover$CovType == 4] <- "Cotton&Willow"
#forestcover$CovName[forestcover$CovType == 5] <- "Aspen"
#forestcover$CovName[forestcover$CovType == 6] <- "DouglasFir"
#forestcover$CovName[forestcover$CovType == 7] <- "Krummholz"

#forestcover <- mutate(forestcover,Climate = "")
#forestcover$Climate[forestcover$Montane_low == 1 & forestcover$Non_Dry ==1] <- "1Montane_Low_NonDry"
#forestcover$Climate[forestcover$Montane == 1] <- "2Montane"
#forestcover$Climate[forestcover$Subalpine == 1] <- "3SubAlpine"
#forestcover$Climate[forestcover$Alpine == 1] <- "4Alpine"

#forestcover <- mutate(forestcover,Wilderness_Area = "")
#forestcover$Wilderness_Area[forestcover$RWwild == 1] <- "Rawah"
#forestcover$Wilderness_Area[forestcover$NEwild == 1] <- "Neota"
#forestcover$Wilderness_Area[forestcover$CMwild == 1] <- "Comanche"
#forestcover$Wilderness_Area[forestcover$CPwild == 1] <- "Cache"
#forestcover$Wilderness_Area <- as.factor(forestcover$Wilderness_Area)
```

## Data distributions

Now check some basic distributions.

### Elevation Histogram - Figure 1

```
startTime=Sys.time()
  print(paste("Plot creation started at",startTime))
```

```
## [1] "Plot creation started at 2018-04-18 14:50:34"
```

```
# Figure 1
  jpeg(filename="Figure01.jpg")
  plot(table(forestcover$Elev))
  dev.off()
```

```
## pdf
##   2
```

The elevation histogram looks reasonable for Colorado high country.

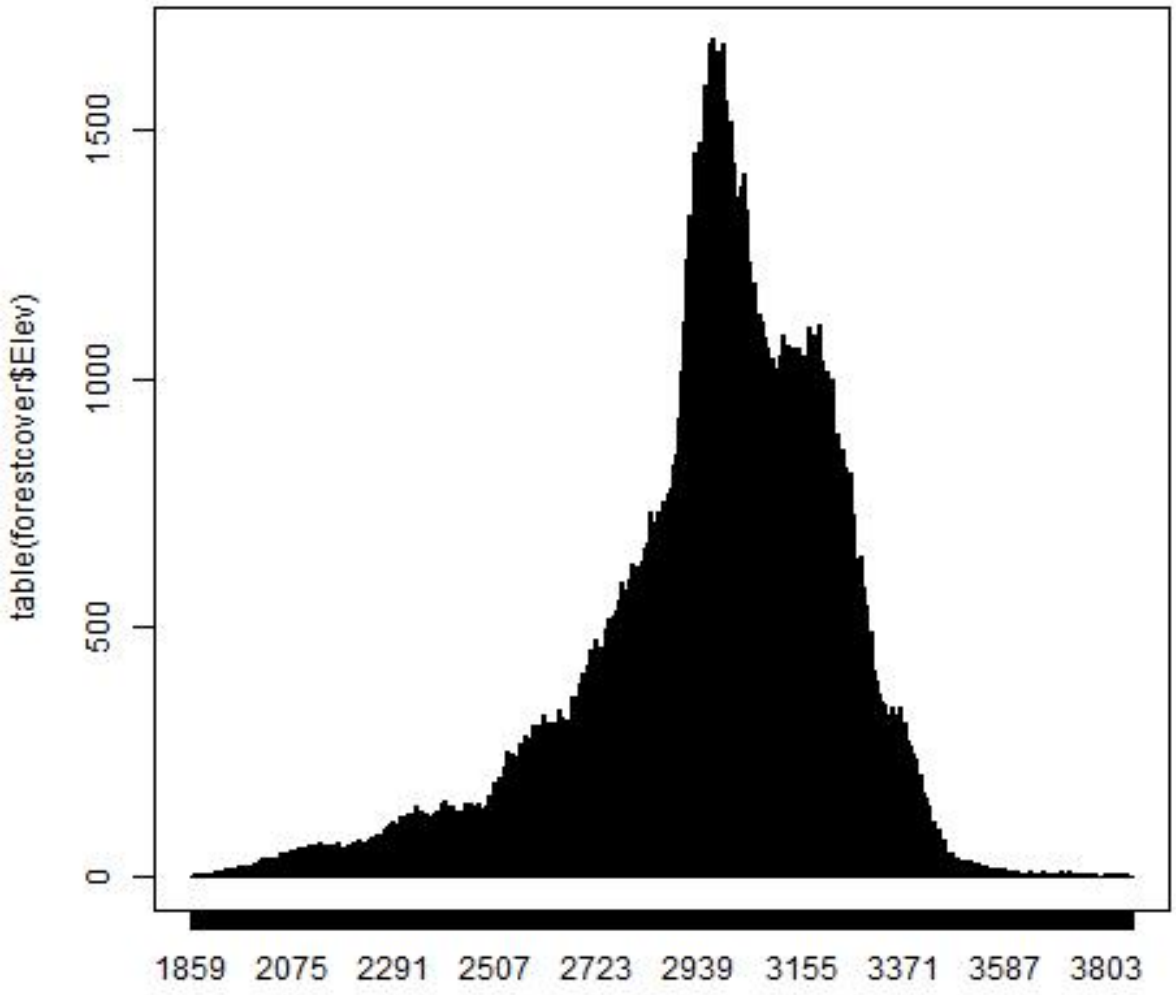


Figure 1: Elevation Histogram

**Terrain Aspect - Figure 2**

```
# Figure 2
  jpeg(filename="Figure02.jpg")
  plot(table(forestcover$Aspect))
  dev.off()
```

```
## pdf
##   2
```

The aspect, the direction the terrain is facing, forms an interesting sine wave with the most terrain facing 45 degrees or North East and the least amount of terrain facing 230 degrees or South West. There are spikes in the graph that look like anomalies due to the size of the bin used by the histogram. There are dips in the bins next to the spikes.

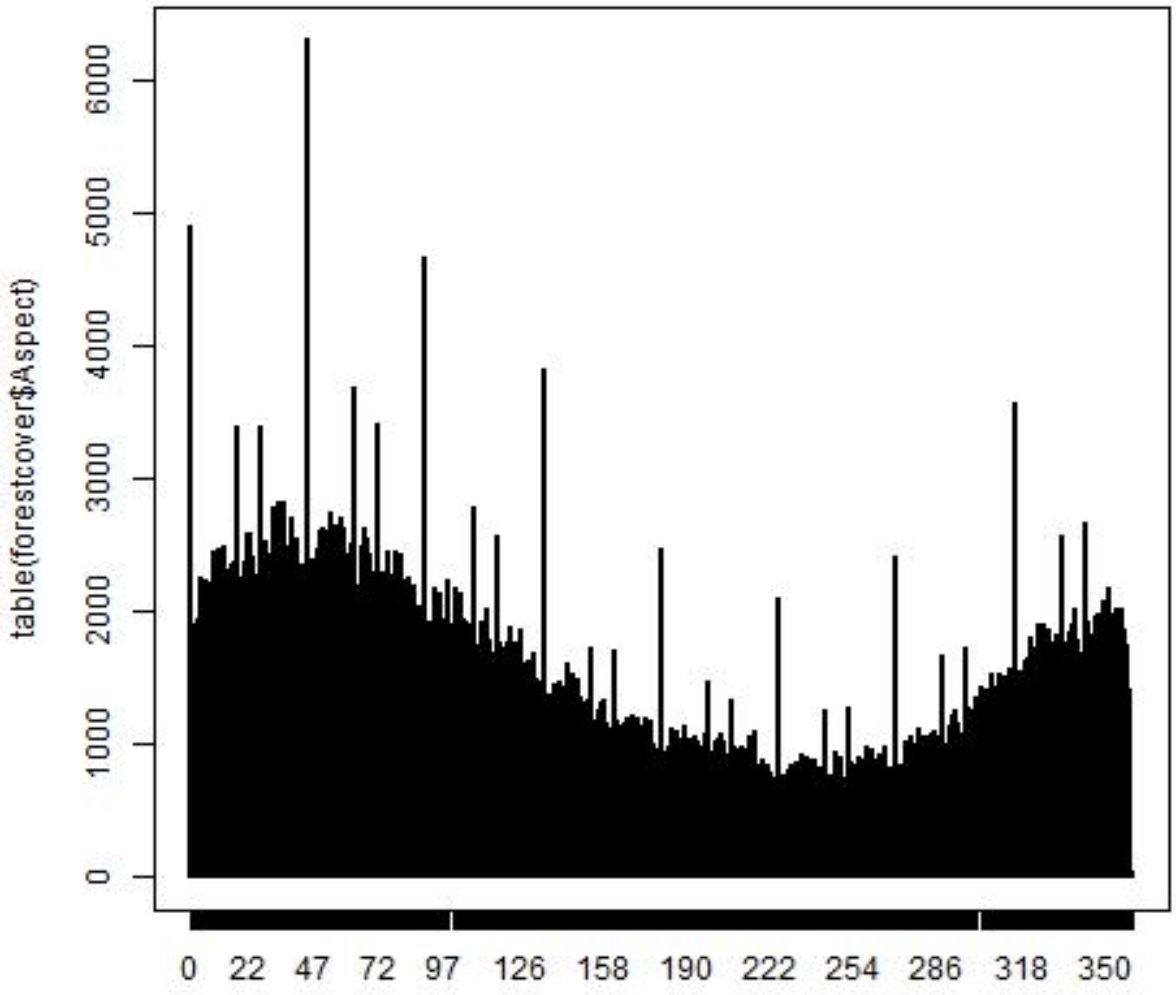


Figure 2: Terrain Aspect Histogram

**Terrain Slope - Figure 3**

```
# Figure 3

  jpeg(filename="Figure03.jpg")
  plot(table(forestcover$Slope))
  dev.off()
```

```
## pdf
##   2
```

The slope distribution seems reasonable and smooth with most of the trees on slopes of 3 degrees to 30 degrees and very few trees on slopes greater than 40 degrees.

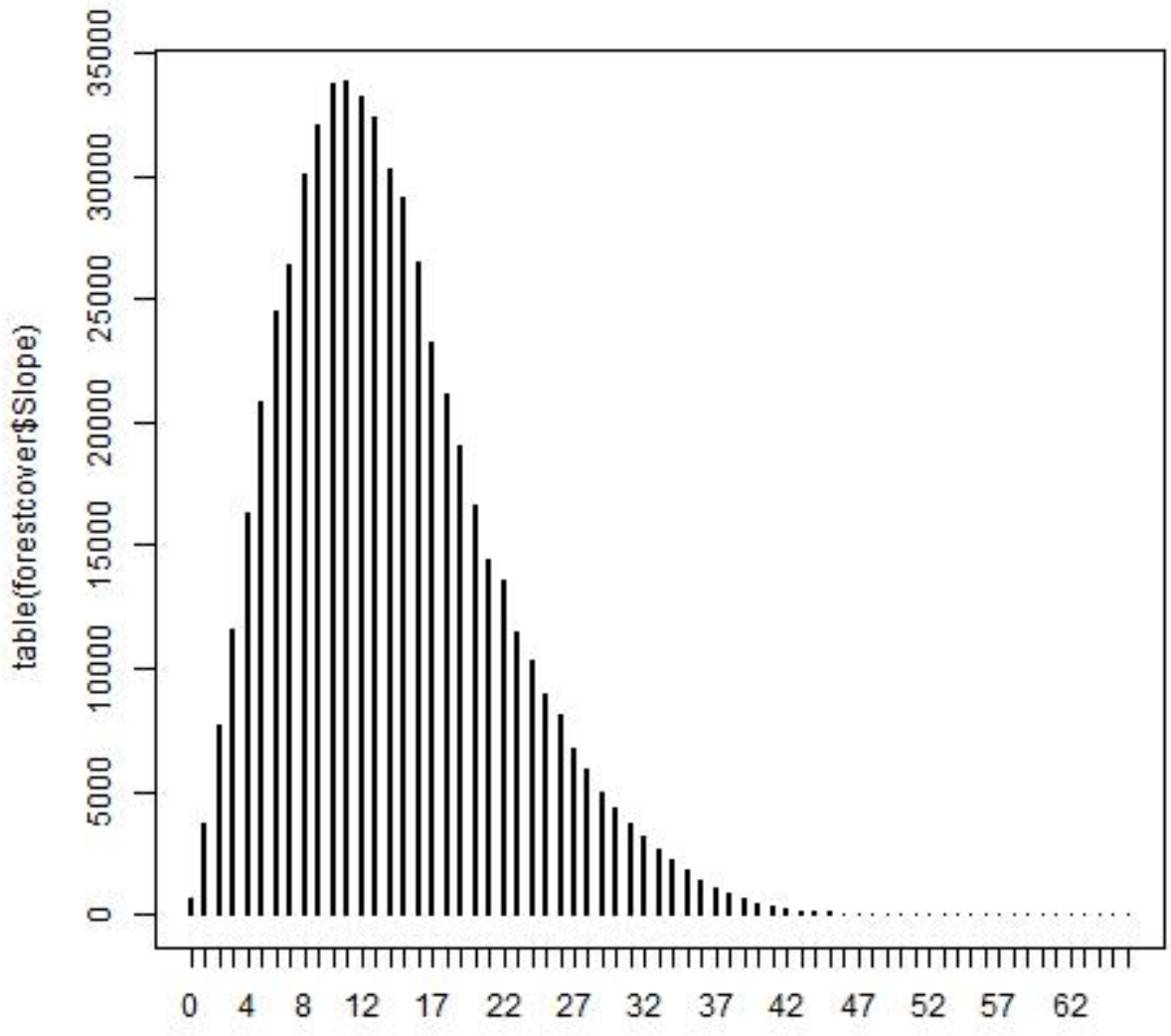


Figure 3: Terrain Slope Histogram

**Horizontal Distance to Water - Figure 4**

```
# Figure 4
  jpeg(filename="Figure04.jpg")
  plot(table(forestcover$H2OHD))
  dev.off()
```

```
## pdf
##   2
```

The horizontal distance to water features seems reasonable. Most of the trees are fairly close to water which shows trees thrive close to water.

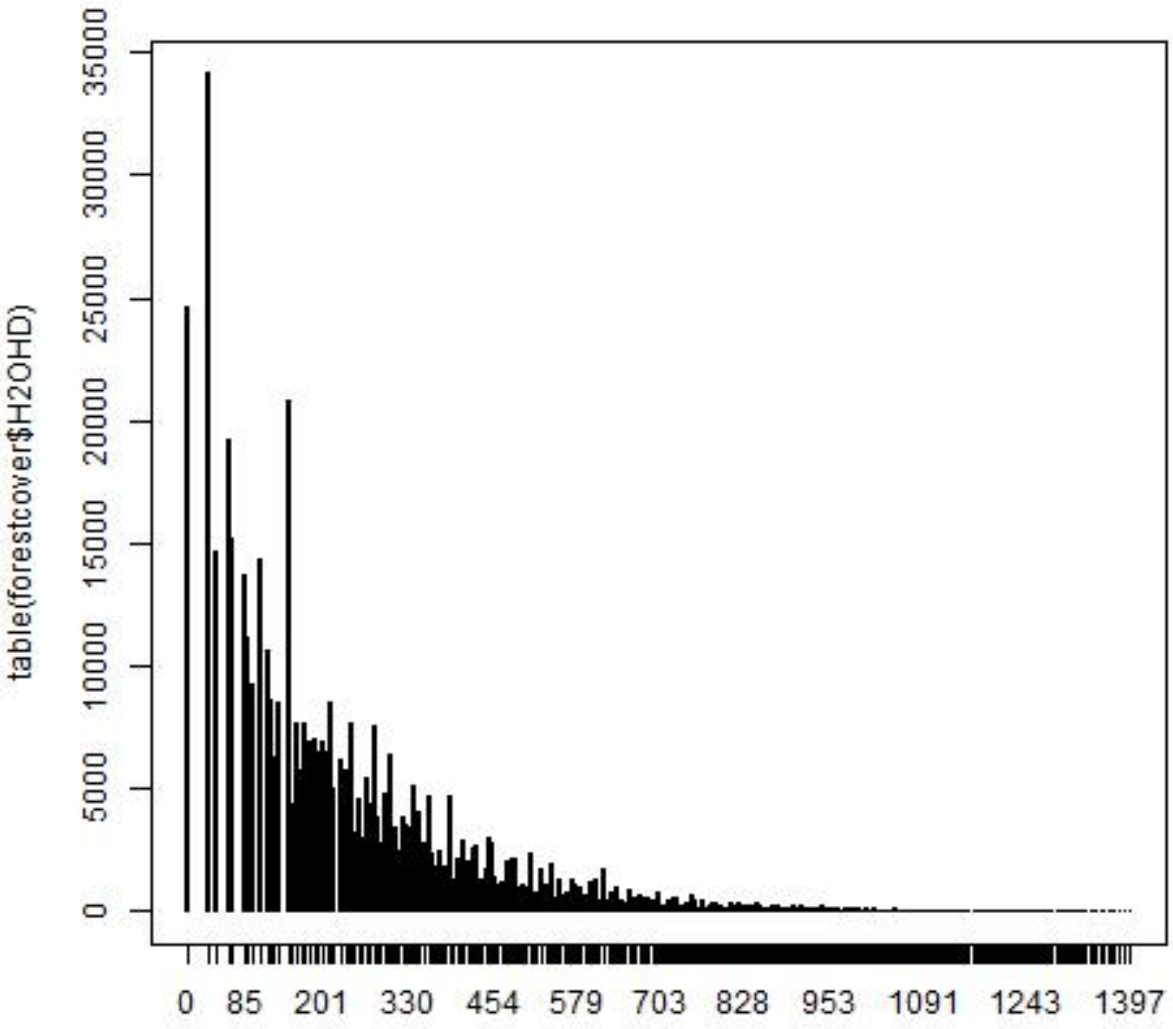


Figure 4: Horizontal Distance to Water Features Histogram

**Vertical Distance to Water - Figure 5**

```
# Figure 5
  jpeg(filename="Figure05.jpg")
  plot(table(forestcover$H2OVD))
  dev.off()
```

```
## pdf
##   2
```

The vertical distance to the nearest water also seems reasonable with the majority of trees having water from above within 5 to 150 meters. For some trees the nearest water is below within about 40 meters.

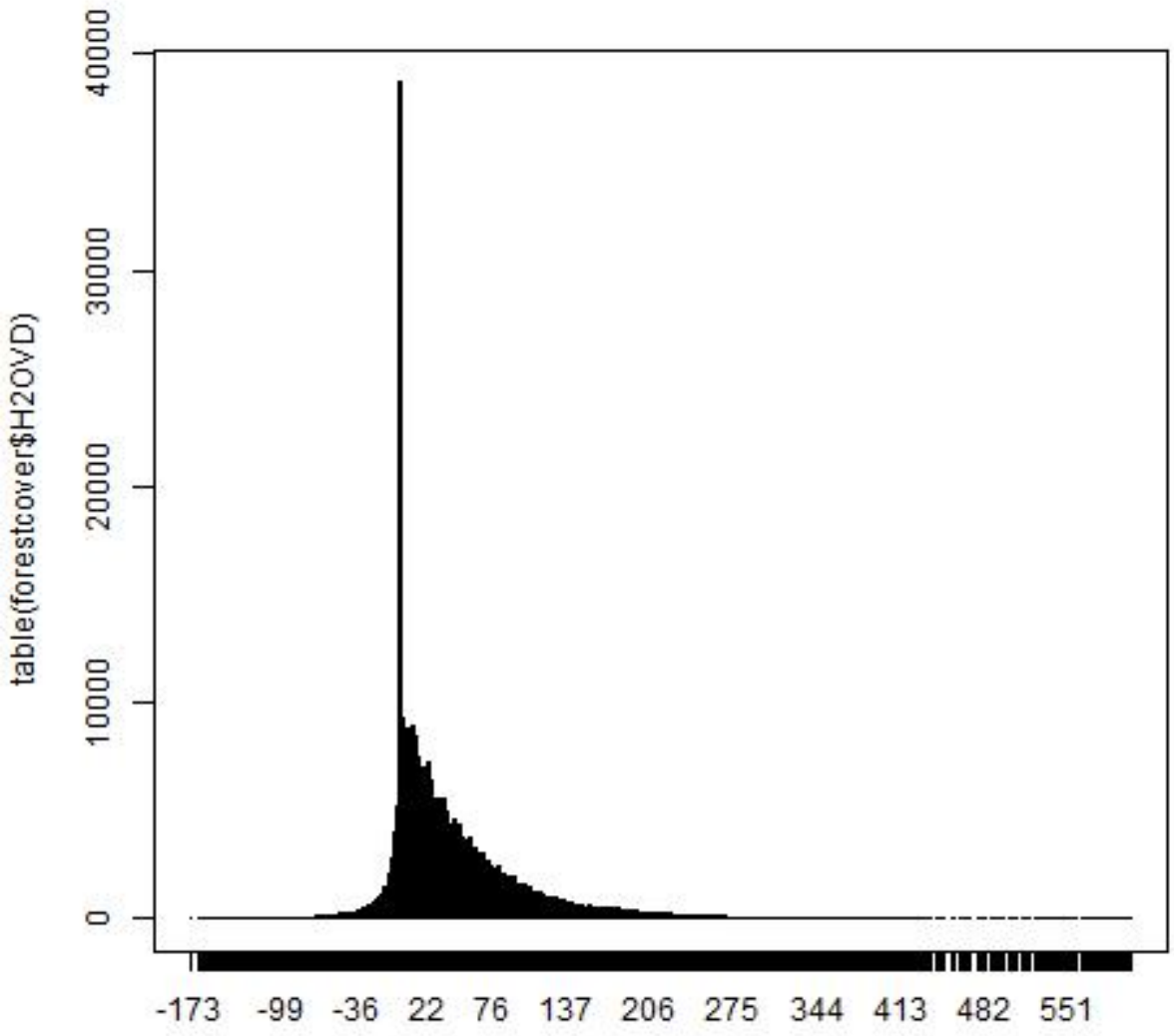


Figure 5: Vertical Distance to Water Features Histogram

**Horzontal Distance to Roads - Figure 6**

```
# Figure 6
  jpeg(filename="Figure06.jpg")
  plot(table(forestcover$RoadHD))
  dev.off()
```

```
## pdf
##   2
```

The nearest roads are closer than I would have expected for wilderness areas. Most of the trees are within 1700 meters or about a mile of a road.

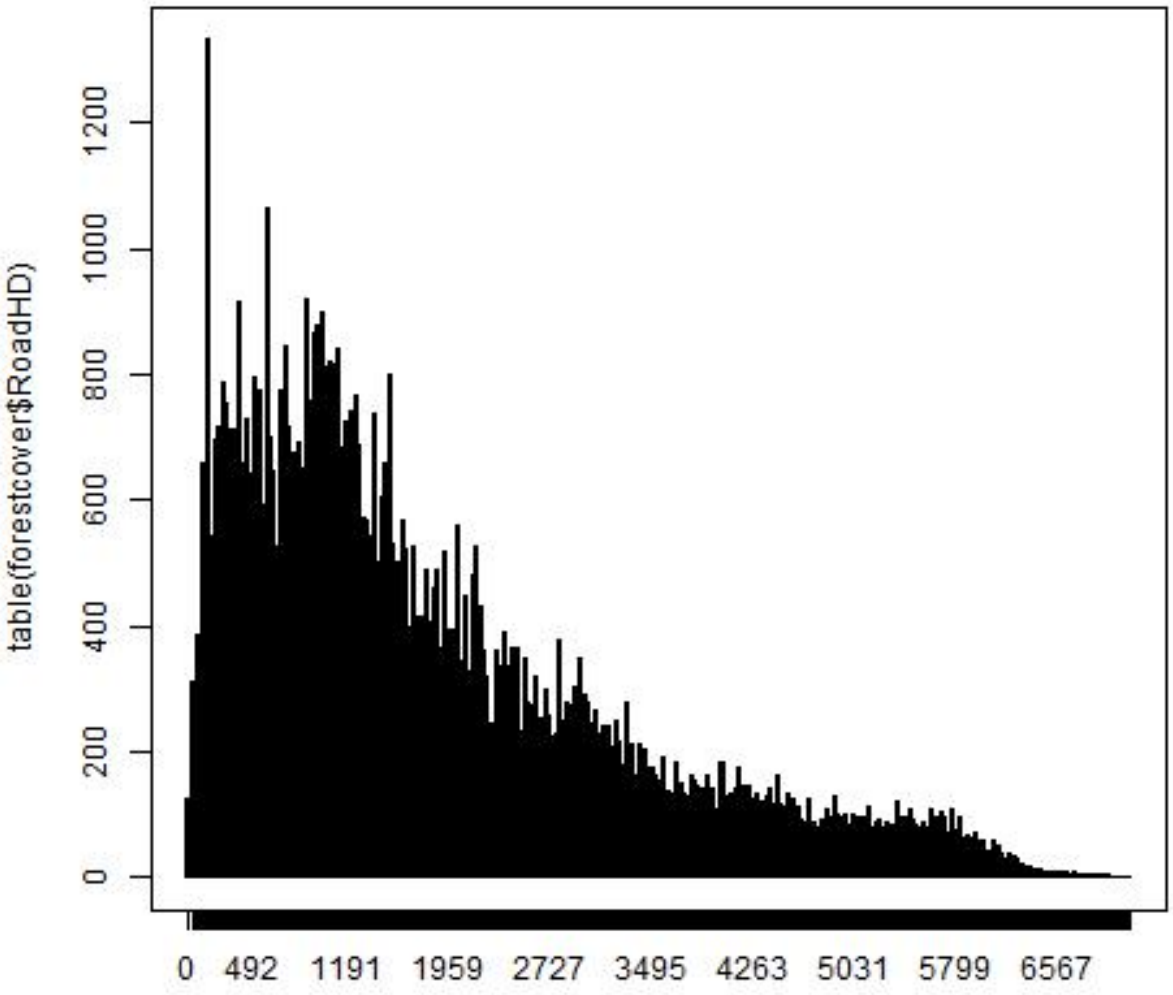


Figure 6: Horizontal Distance to Water Features Histogram

**Shade at 9am - Figure 7**

```
# Figure 7
  jpeg(filename="Figure07.jpg")
  plot(table(forestcover$Shade9AM))
  dev.off()
```

```
## pdf
##   2
```

The shade value is fairly high at 9am with most area having between 80% shade (207/254) and 90% shade (238/254).

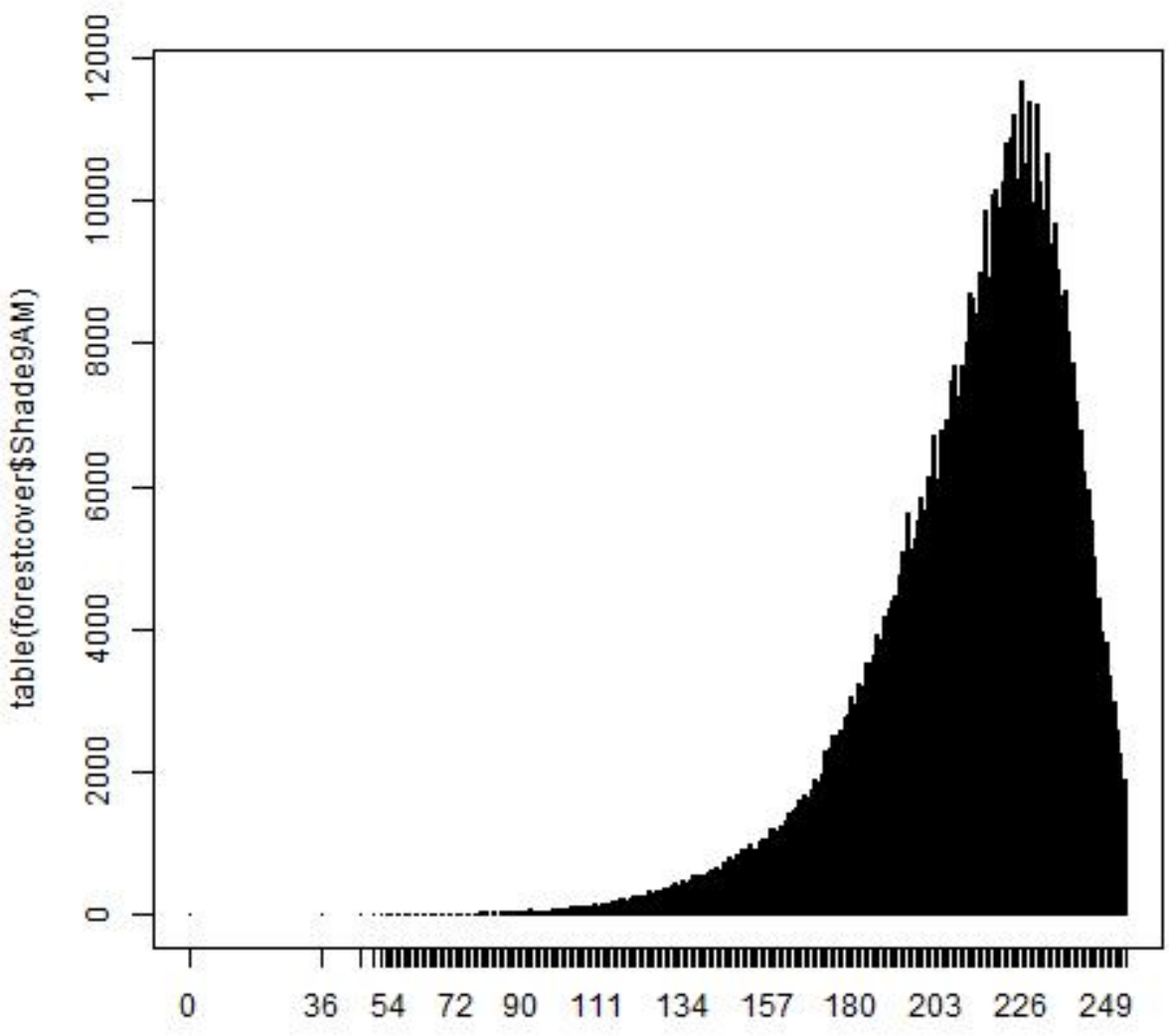


Figure 7: Shade at 9AM Histogram

**Shade at 12pm - Figure 8**

```
# Figure 8
  jpeg(filename="Figure08.jpg")
  plot(table(forestcover$Shade12PM))
  dev.off()
```

```
## pdf
##   2
```

There is even more shade at noon with most areas having between 82% (210/254) and 100% shade (252/254).

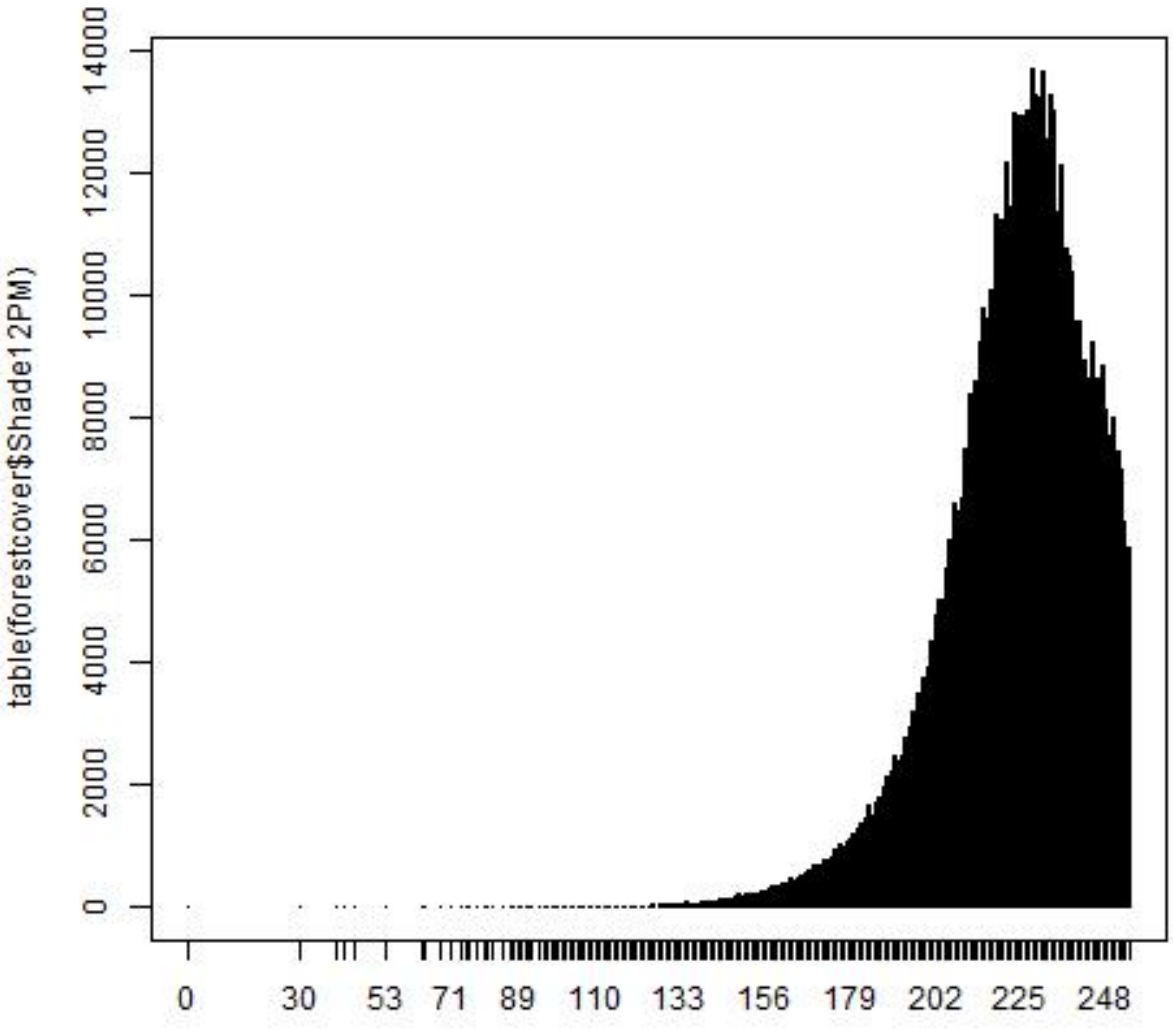


Figure 8: Shade at 12PM Histogram

**Shade at 3pm - Figure 9**

```
# Figure 9
  jpeg(filename="Figure09.jpg")
  plot(table(forestcover$Shade3PM))
  dev.off()
```

```
## pdf
##   2
```

There is more sun (less shade) at 3pm with most cells having between 37% (94/254) and 77% (196/254) shade.

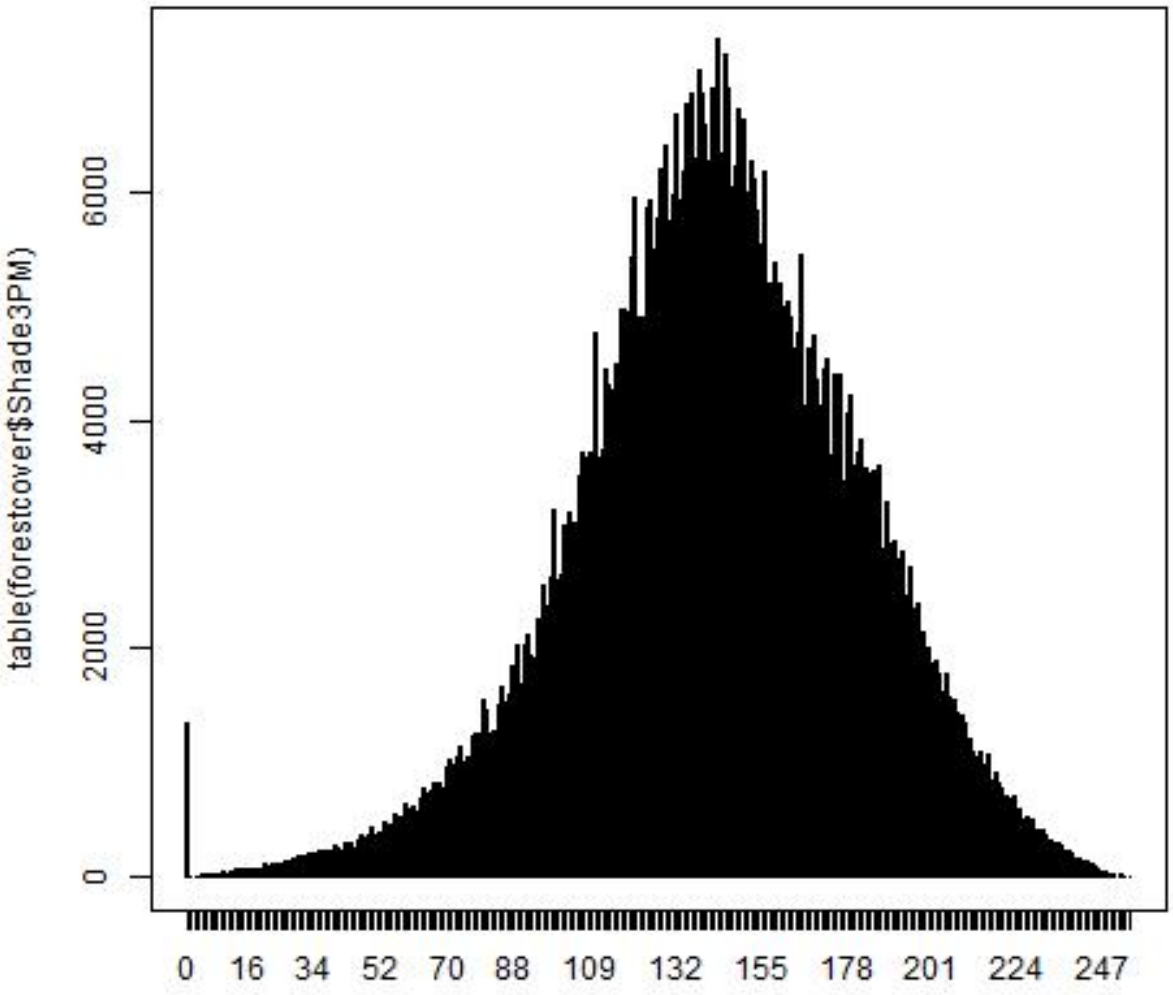


Figure 9: Shade at 3PM Histogram

**Horzontal Distance to Fire Points - Figure 10**

```
# Figure 10
  jpeg(filename="Figure10.jpg")
  plot(table(forestcover$FirePtHD))
  dev.off()
```

```
## pdf
##   2
```

The distance from fire occurrence is similar to the distance to roads graph. While there is correlation. It is not possible to determine causation.

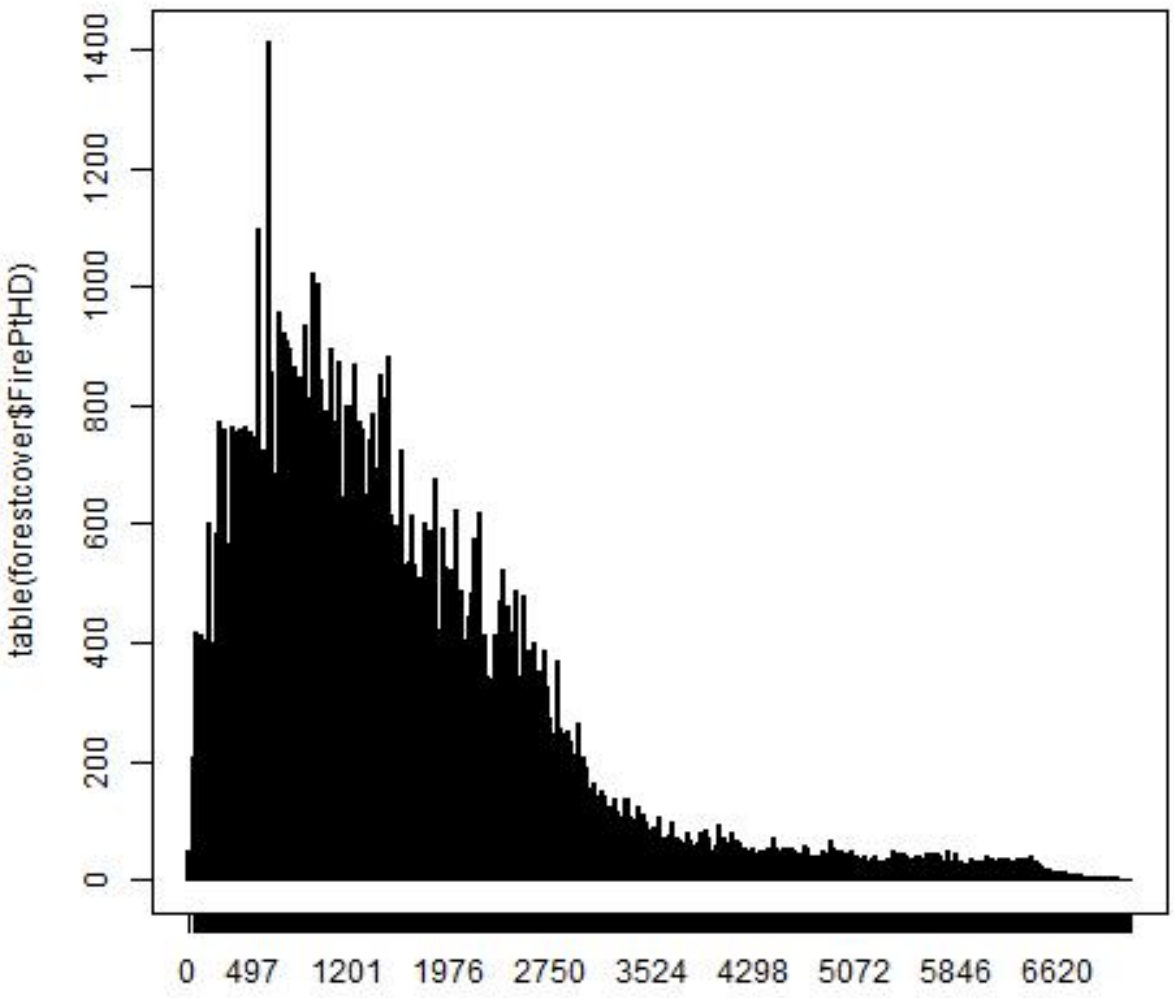


Figure 10: Horizontal Distance to Fire Points Histogram

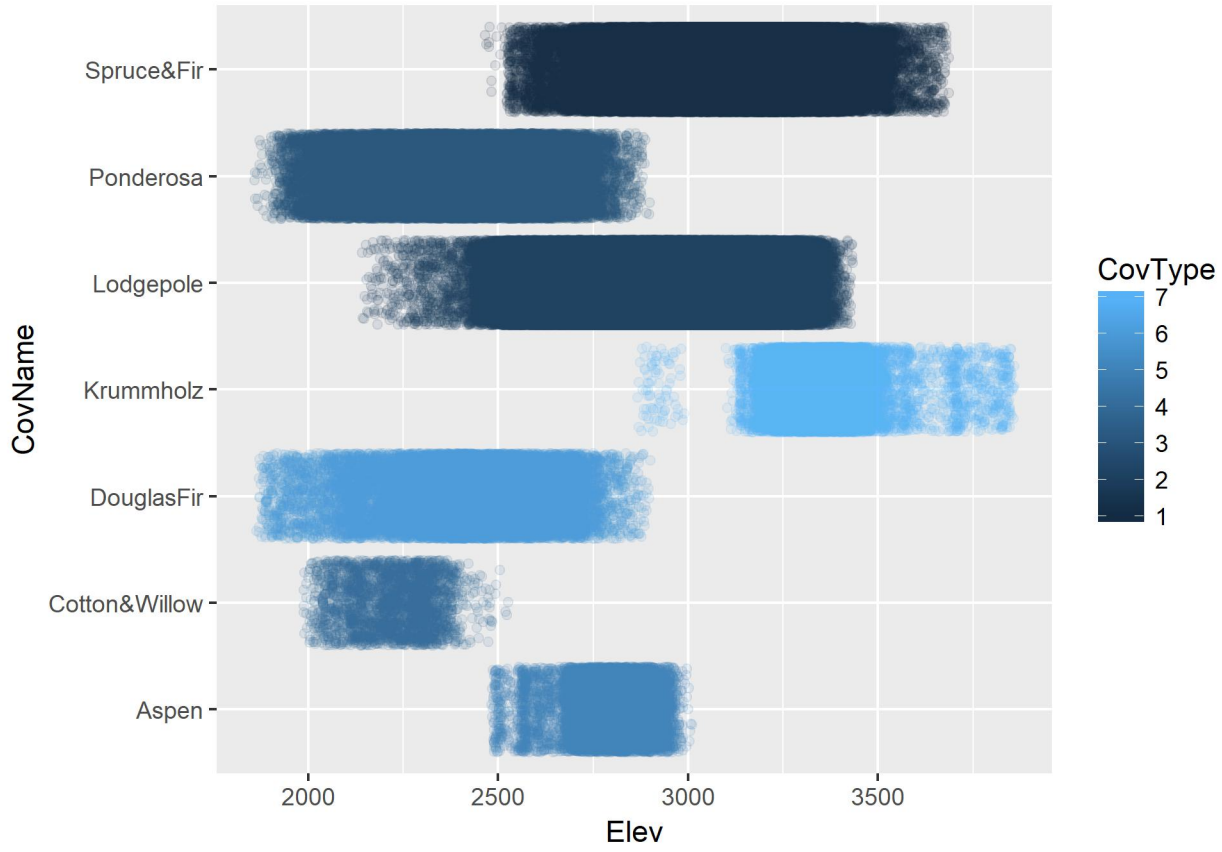


Figure 11: Tree Type vs Elevation

**Tree Type vs Elevation - Figure 11**

```
# Figure 11
  g <- ggplot(forestcover,aes(Elev, CovName,col=CovType)) +
        geom_jitter(alpha=alphaVal)
  ggsave("Figure11.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Elevation vs Tree Type shows that trees reside in a range of elevations and will help in determining tree type, but more information will be needed where there is overlap in elevation.

This graph looks a little strange. The next graph reverses the axes.

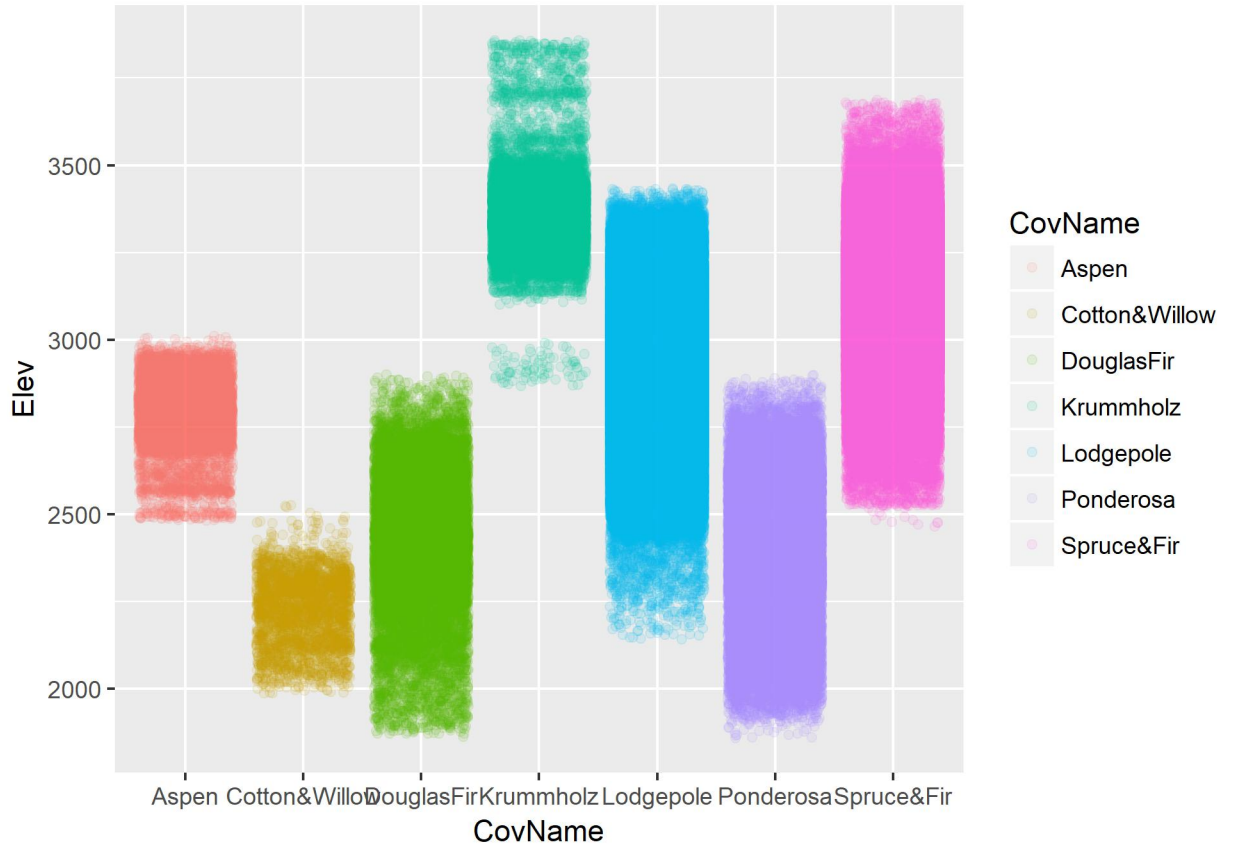


Figure 12: Figure 12

**Elevation vs Tree Type - Figure 12**

```
# Figure 12
  g <- ggplot(forestcover,aes(CovName,Elev,col=CovName)) +
        geom_jitter(alpha=alphaVal)
  ggsave("Figure12.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

This graph is more pleasant to my eyes, though I can't say exactly why. There is an unusual gap in the Krummholz between 3000 and 3125 meters. I wonder if this might be due to the different Wilderness areas.

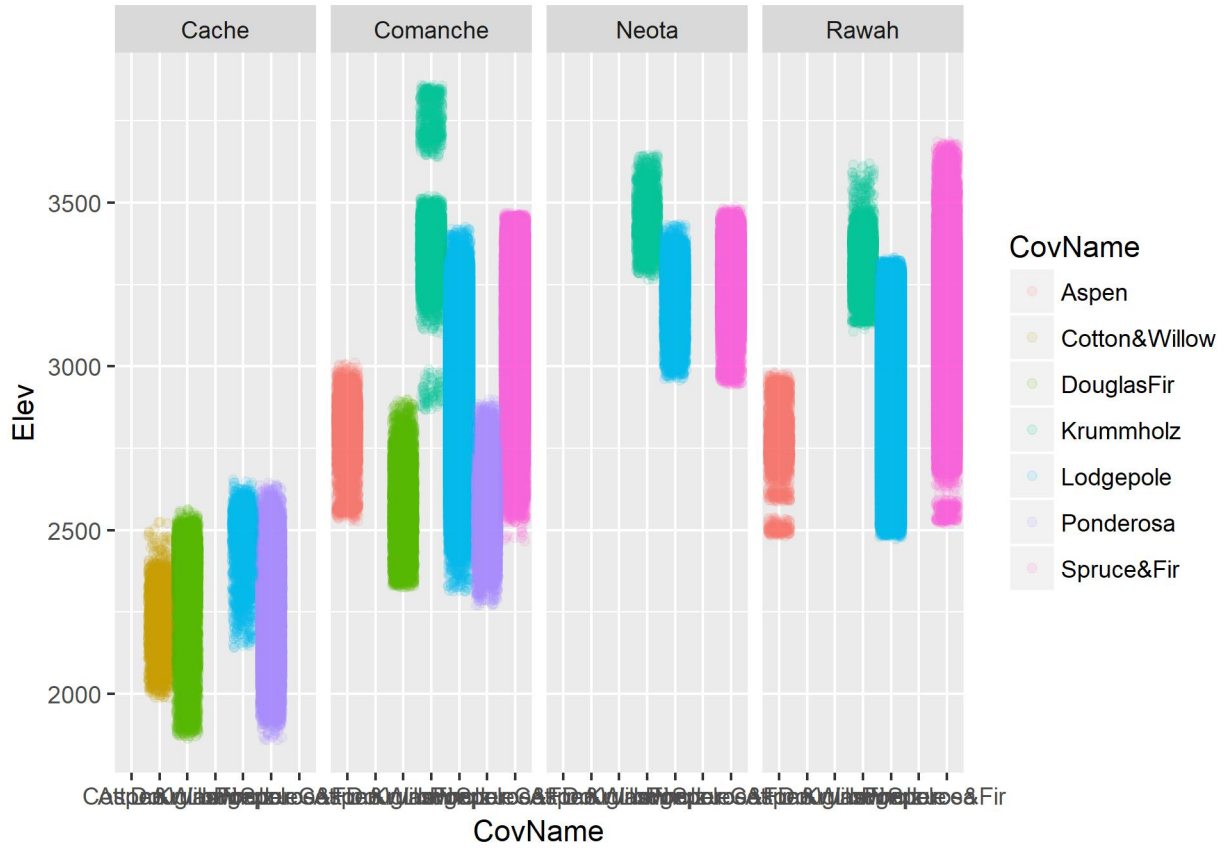


Figure 13: Elevations vs Tree Type and Wilderness Area

**Elevations vs Tree Type and Wilderness Area - Figure 13**

```
# Figure 13
  g <- ggplot(forestcover,aes(CovName,Elev,col=CovName)) +
        geom_jitter(alpha=alphaVal) +
    facet_grid(. ~ Wilderness_Area)
  ggsave("Figure13.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

The interesting elevation gap in Krummholz tree occurs in the Comanche wilderness area. But we see there are two gaps and must be due to some areas of terrain that vary significantly in the Comanche wilderness area. I was expecting that the Krummholz tree type should have been continuous in each wilderness area and the gap would be explained by different elevation ranges in different wilderness areas. This plot also shows that there are 1 to 4 types of trees missing in each wilderness area.

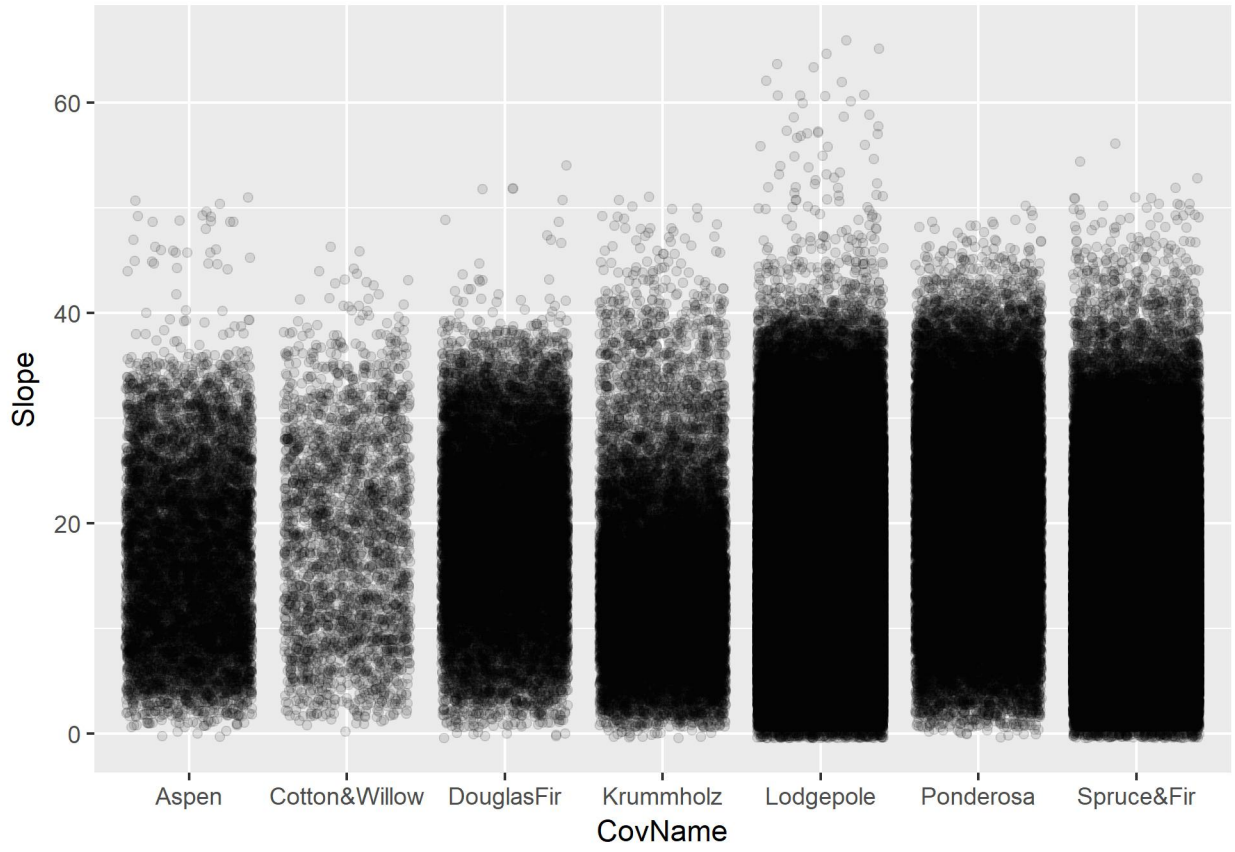


Figure 14: Slope vs Tree Type

**Slope vs Tree Type - Figure 14**

```
# Figure 14
g <- ggplot(forestcover,aes(CovName,Slope)) +
       geom_jitter(alpha=alphaVal)
  ggsave("Figure14.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

The distribution of tree type by slope appears to be pretty evenly distributed and the distribution is close to the same for each tree type. It doesn't look like slope will be a major factor to help determine the tree type.

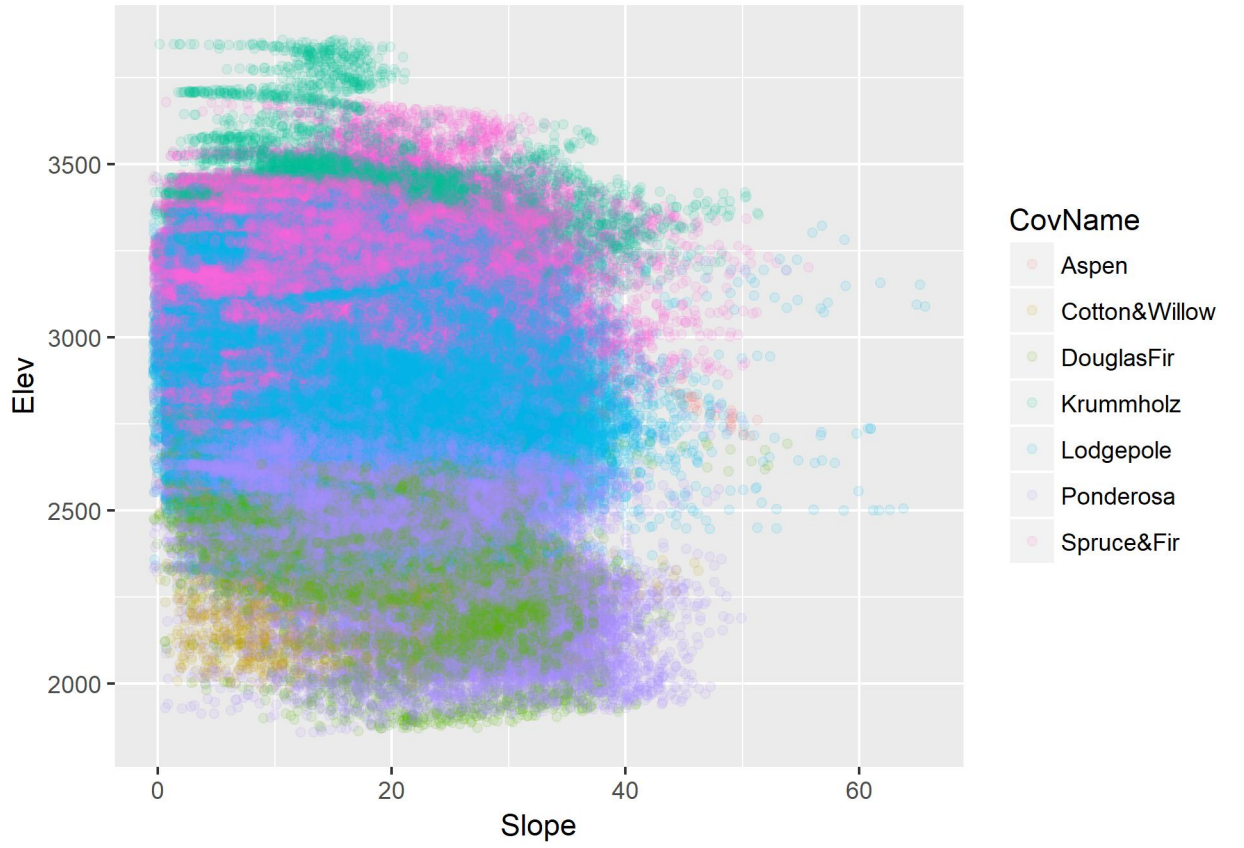


Figure 15: Slope vs Elev and Tree Type

**Slope vs Elev and Tree Type - Figure 15**

```
# Figure 15
  g <- ggplot(forestcover,aes(Slope,Elev,col=CovName)) +
      geom_jitter(alpha=alphaVal)
  ggsave("Figure15.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

It is hard to interpret this graph. There is a scattering of trees continuously over the range of the slope and elevation.

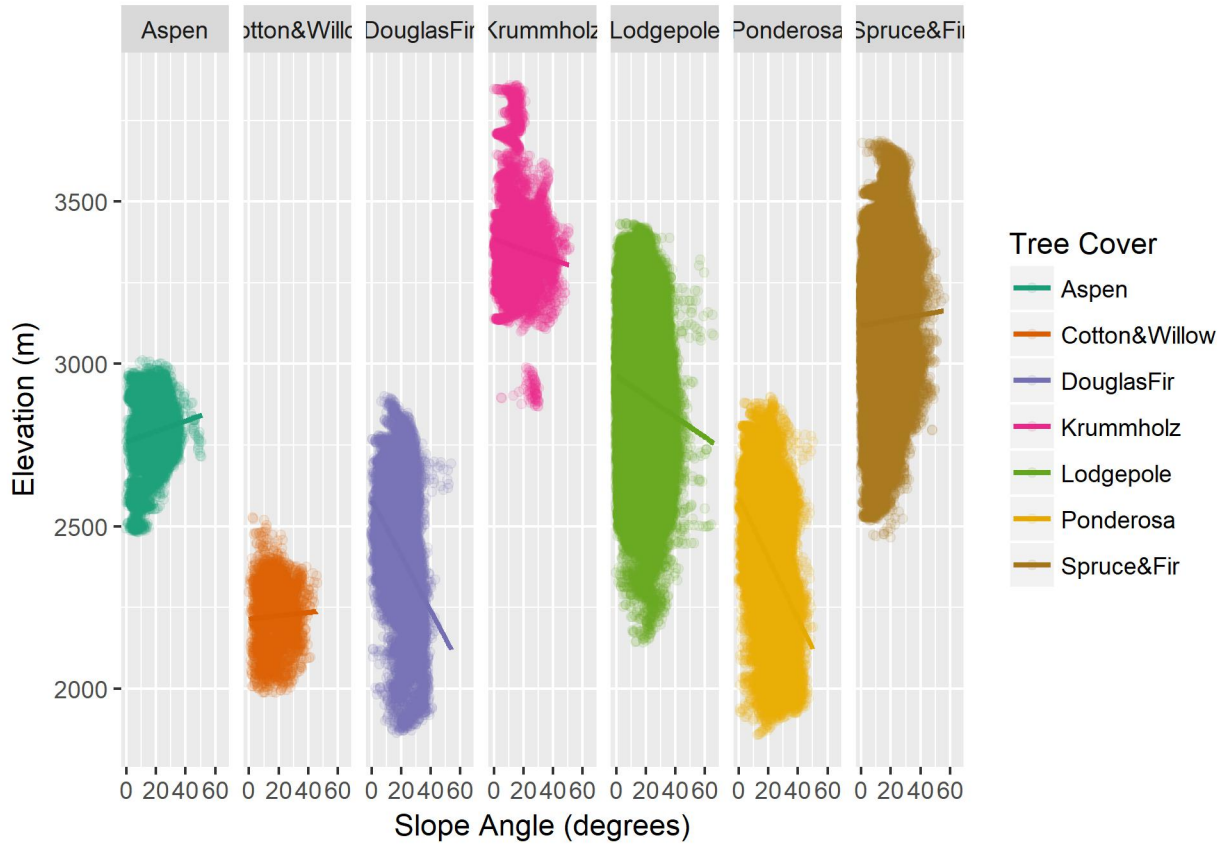


Figure 16: Elevation vs Slope and Tree Type

**Elevation vs Slope and Tree Type - Figure 16**

```
# Figure 16
library(RColorBrewer)
  myColors1 <- c(brewer.pal(7, "Dark2"), "black")

  g <- ggplot(forestcover,aes(Slope,Elev,col=CovName)) +
       geom_jitter(alpha=alphaVal) +
       stat_smooth(method = "lm", se = F) +
       facet_grid(. ~ CovName) +
       scale_color_manual("Tree Cover",values=myColors1) +
       labs(x = "Slope Angle (degrees)",
            y = "Elevation (m)")
  ggsave("Figure16.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

This graph replicates graph 12, so, no new info here. Adding the smooth line mainly helps to see the color of the legend when the alpha value is so low.

**Elevation vs Terrain Aspect - Figure 17**

```
# Figure 17
myColors1 <- c(brewer.pal(7, "Dark2"), "black")
  g <- ggplot(forestcover,aes(Aspect,Elev,col=CovName)) +
       geom_jitter(alpha=alphaVal) +
       stat_smooth(method = "lm", se = F) +
#       facet_grid(. ~ CovName) +
       scale_color_manual("Tree Cover",values=myColors1) +
       labs(x = "Aspect Direction (degrees)",
            y = "Elevation (m)")
  ggsave("Figure17.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

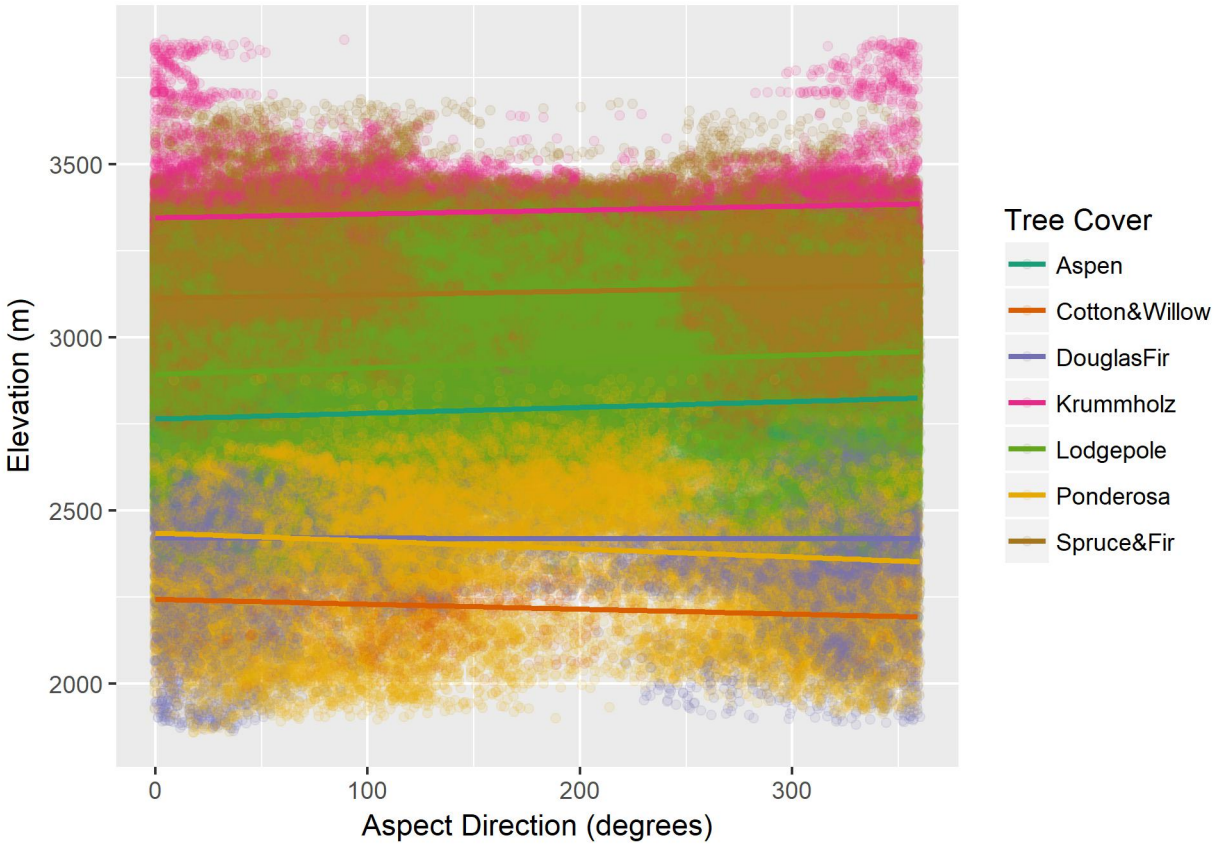


Figure 17: Elevation vs Terrain Aspect

The elevation vs aspect shows that, like slope, tree type is not related much to Aspect.

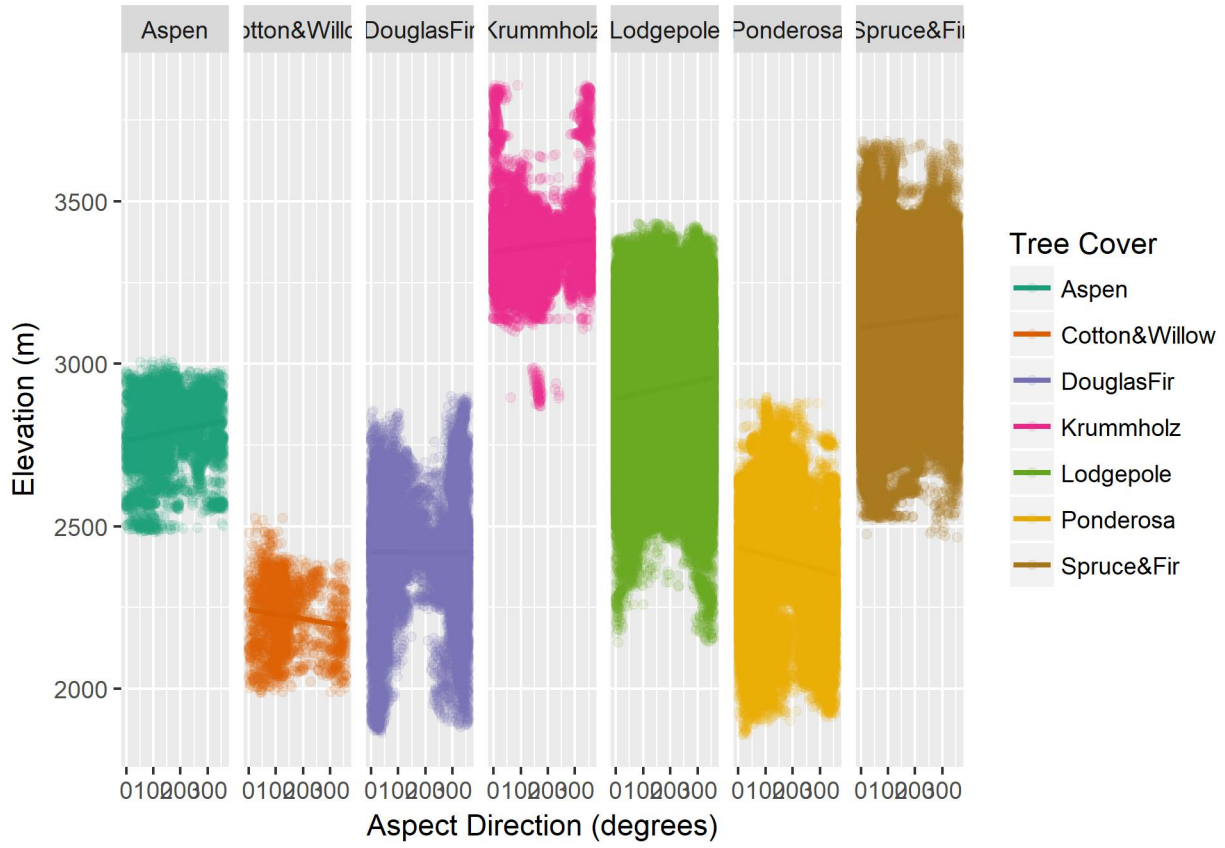


Figure 18: Elevation vs Terrain Aspect by Tree Type

**Elevation vs Terrain Aspect by Tree Type - Figure 18**

```
# Figure 18
  g <- ggplot(forestcover,aes(Aspect,Elev,col=CovName)) +
       geom_jitter(alpha=alphaVal) +
       stat_smooth(method = "lm", se = F) +
       facet_grid(. ~ CovName) +
       scale_color_manual("Tree Cover",values=myColors1) +
       labs(x = "Aspect Direction (degrees)",
            y = "Elevation (m)")
  ggsave("Figure18.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Another view of elevation vs aspect shows there are concentrations of tree types near 0 and 360 degrees. This occurs for all tree types and shows again that aspect is not going to help very much.

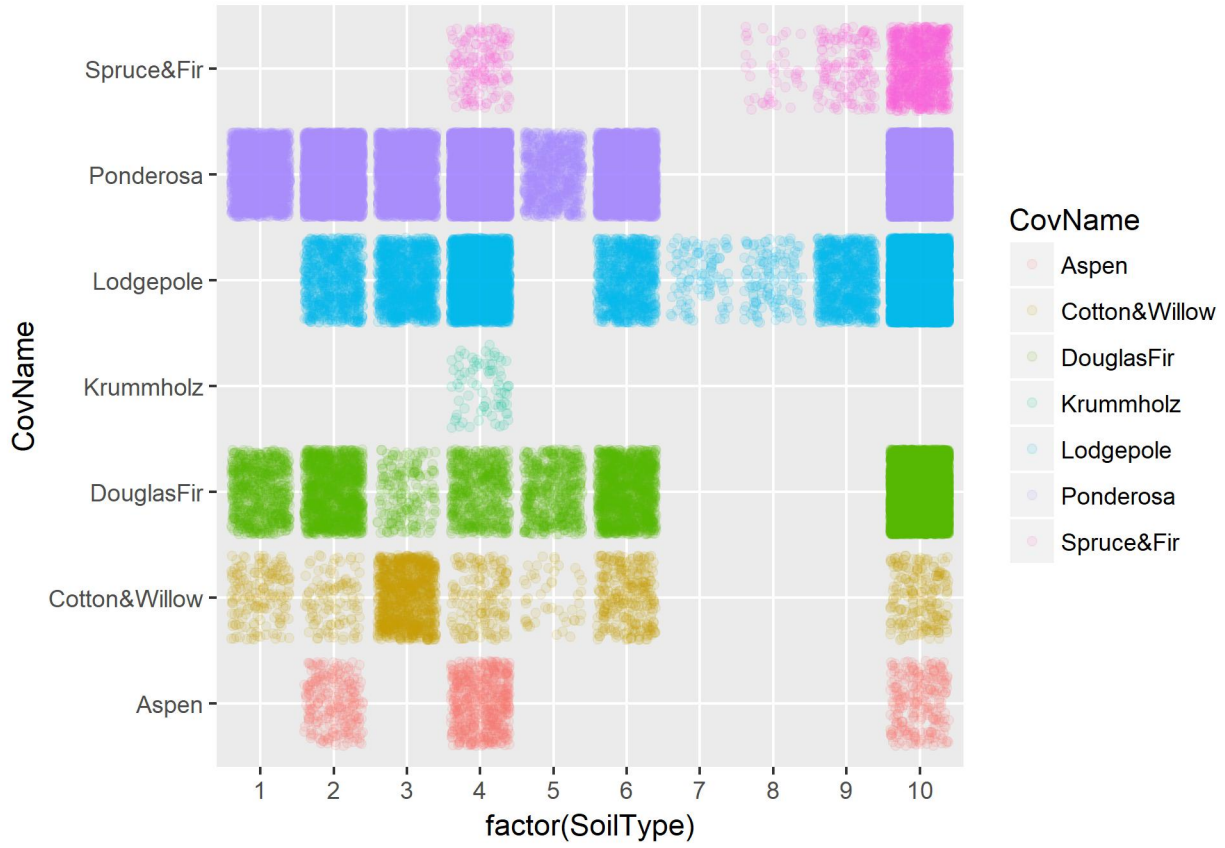


Figure 19: Tree Type vs Soil Types 1 thru 10

**Tree Type vs Soil Types 1 thru 10 - Figure 19**

```
# Figure 19
st1_10 <- forestcover[forestcover$SoilType<11,]

g <- ggplot(st1_10,aes(factor(SoilType), CovName,col=CovName)) +
      geom_jitter(alpha=alphaVal)
ggsave("Figure19.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

The aggregated soil types used by Dr Blackard as predictors in his PhD dissertation are shown in the next several graphs. Here we see soil types 1 through 10. It's difficult for me to draw any conclusions.

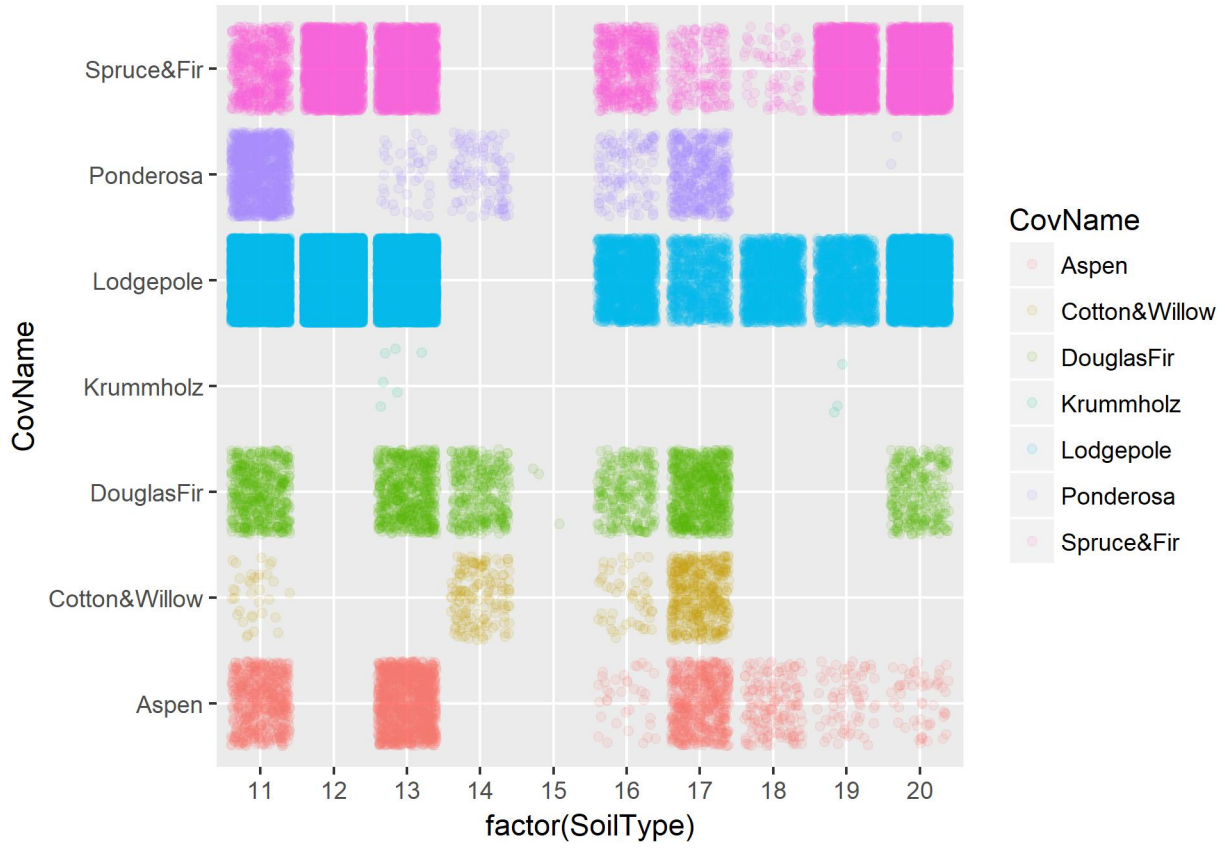


Figure 20: Tree Type vs Aggregated Soil Types 11 thru 20

**Tree Type vs Aggregated Soil Types 11 thru 20 - Figure 20**

```
# Figure 20
  st11_20 <- forestcover[forestcover$SoilType>10 & forestcover$SoilType<21,]

  g <- ggplot(st11_20,aes(factor(SoilType), CovName,col=CovName)) +
        geom_jitter(alpha=alphaVal)
  ggsave("Figure20.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Continuing with aggregated soil types 11 through 20.

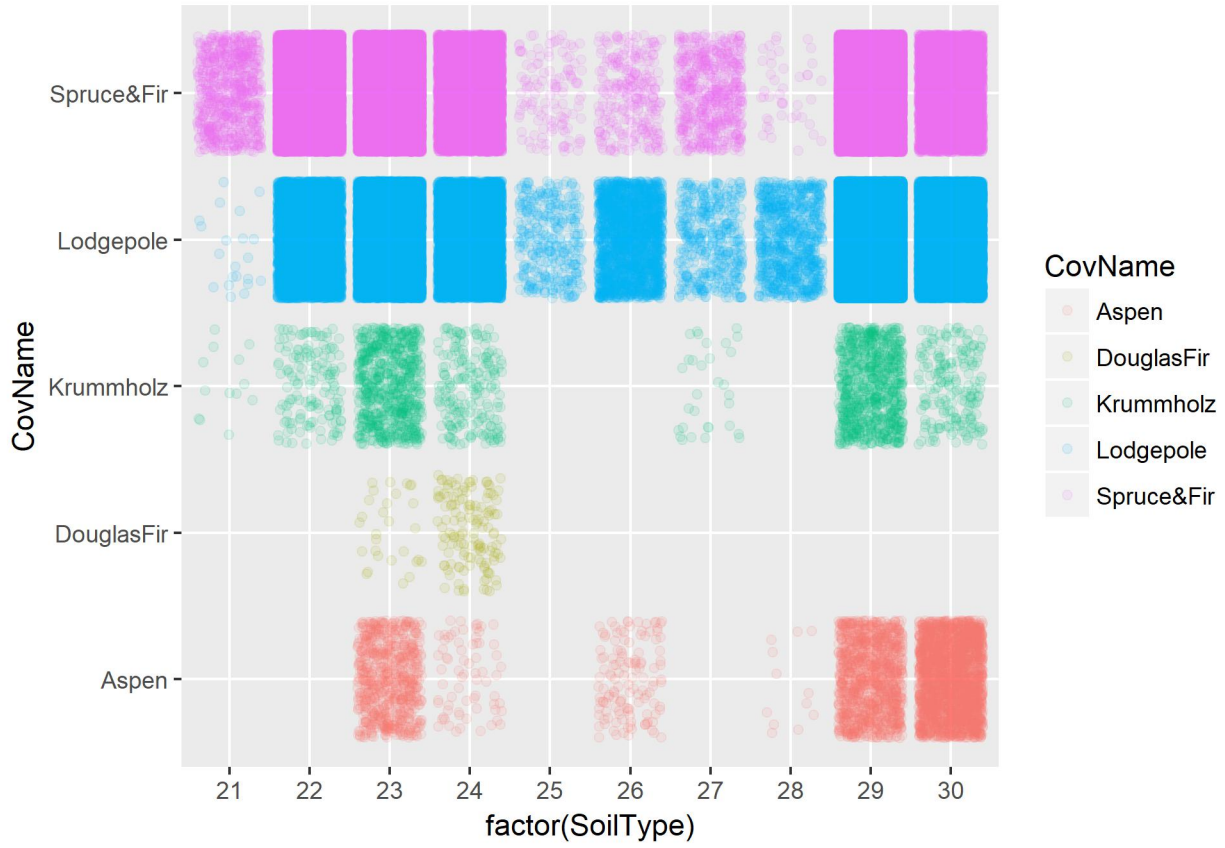


Figure 21: Tree Type vs Aggregated Soil Types 21 through 30

**Tree Type vs Aggregated Soil Types 21 through 30 - Figure 21**

```
# Figure 21
  st21_30 <- forestcover[forestcover$SoilType>20 & forestcover$SoilType<31,]
  g <- ggplot(st21_30,aes(factor(SoilType), CovName,col=CovName)) +
        geom_jitter(alpha=alphaVal)
  ggsave("Figure21.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Continuing with aggregated soil types 21 through 30.

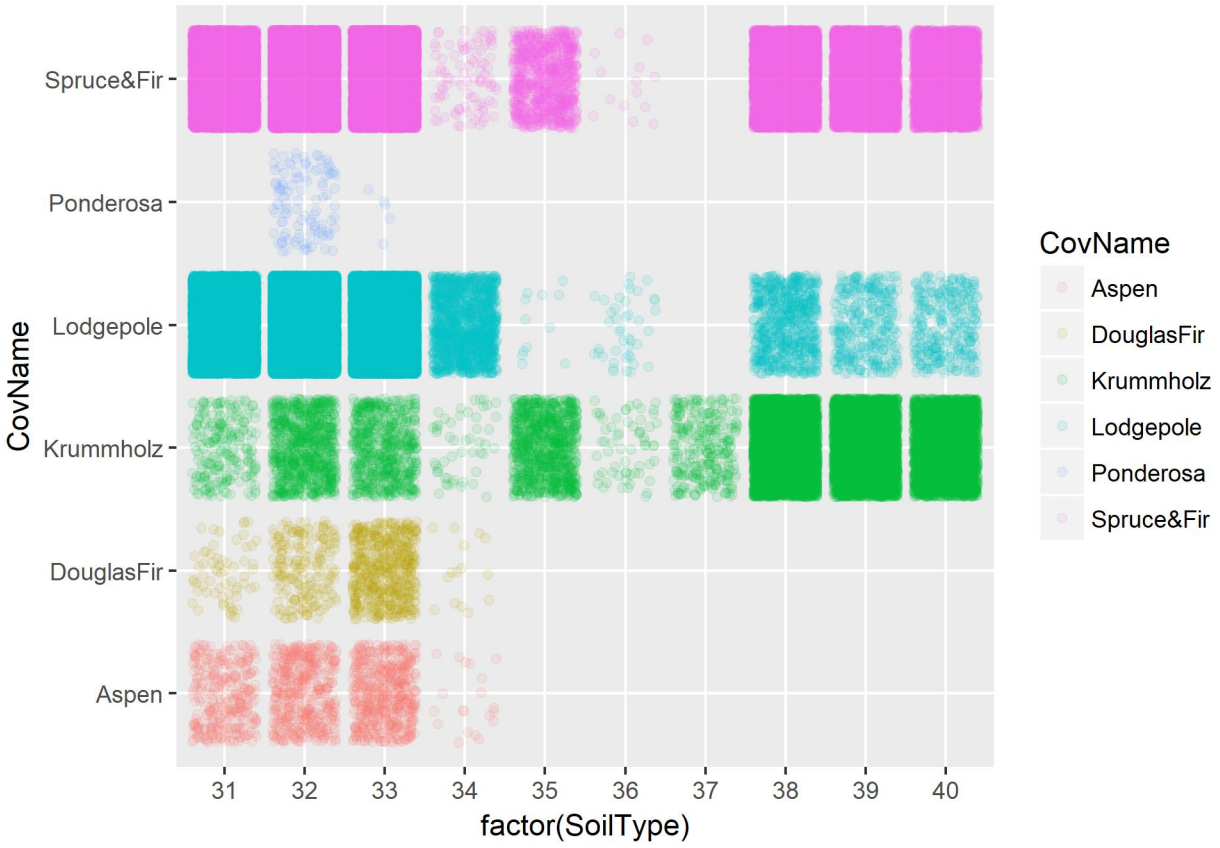


Figure 22: Tree Type vs Aggregated Soil Types 31 through 40

**Tree Type vs Aggregated Soil Types 31 through 40 - Figure 22**

```
# Figure 22
  st31_40 <- forestcover[forestcover$SoilType>30,]
  g <- ggplot(st31_40,aes(factor(SoilType), CovName,col=CovName)) +
        geom_jitter(alpha=alphaVal)
  ggsave("Figure22.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Continuing with aggregated soil types 31 through 40.

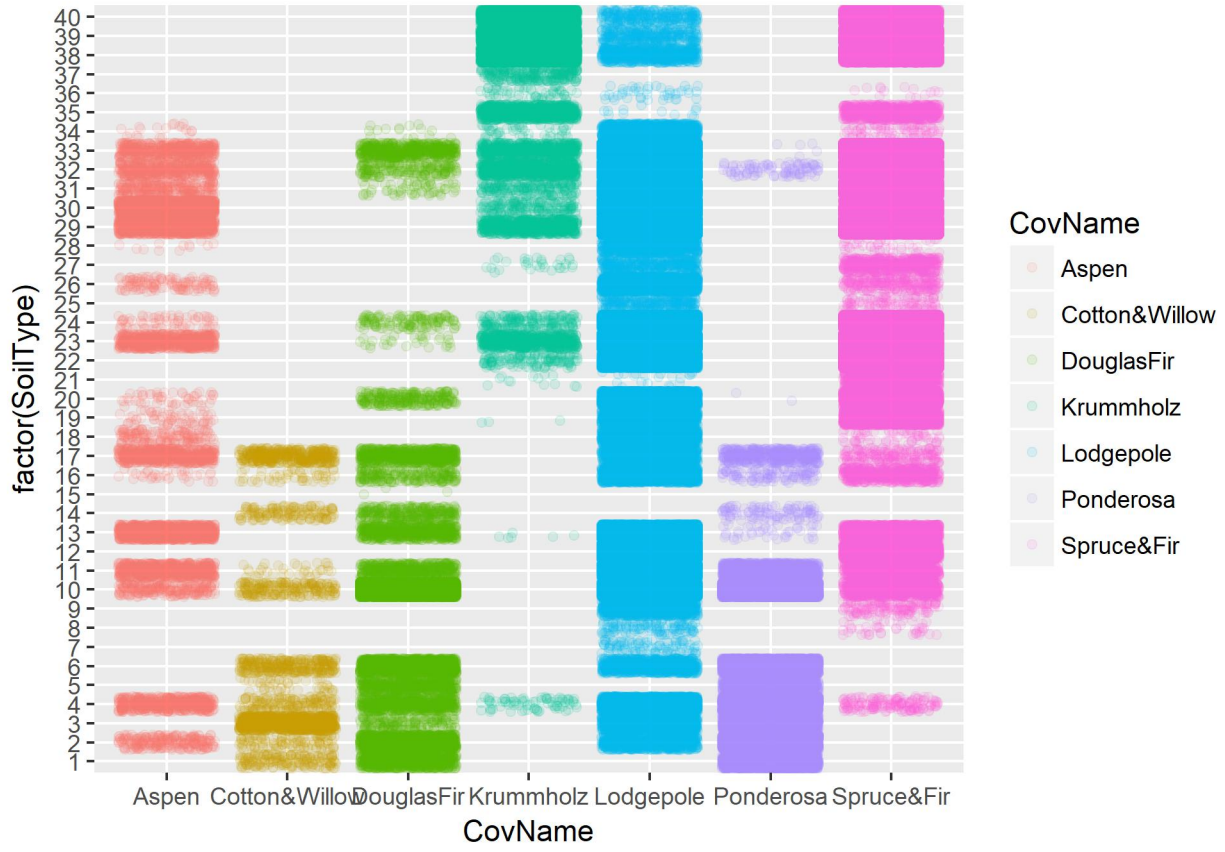


Figure 23: Tree Type vs Aggregated Soil Types 1 through 40

**Tree Type vs Aggregated Soil Types 1 through 40 - Figure 23**

```
# Figure 23
  g <- ggplot(forestcover,aes(CovName,factor(SoilType), col=CovName)) +
        geom_jitter(alpha=alphaVal)
  ggsave("Figure23.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Looking at aggregated soil types 1 through 40. Still nothing jumping out.

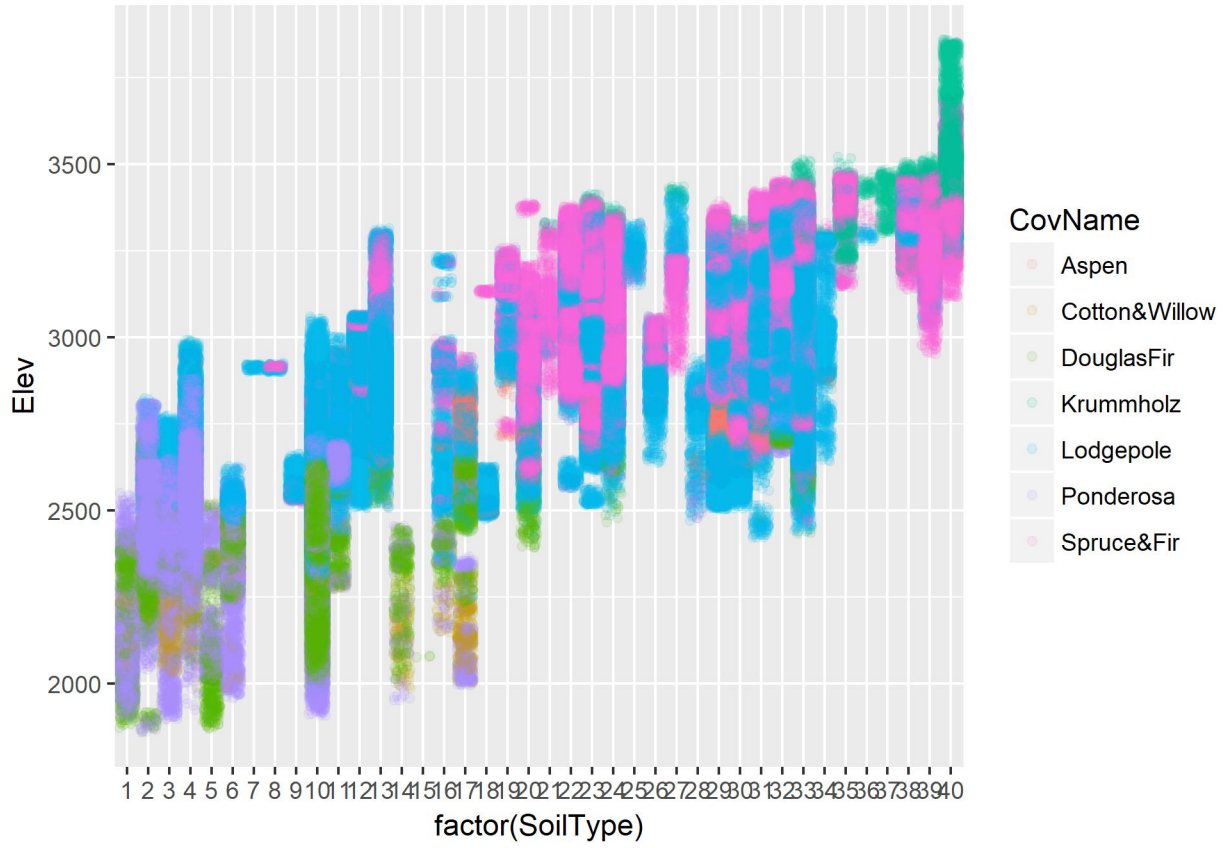


Figure 24: Elevation vs Aggregated Soil Type with Tree Type

**Elevation vs Aggregated Soil Type with Tree Type - Figure 24**

```
# Figure 24
g <- ggplot(forestcover,aes(factor(SoilType),Elev, col=CovName)) +
       geom_jitter(alpha=alphaVal)
  ggsave("Figure24.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Seeing if Elevation vs Aggregated Soil Type gives any insights. Nothing jumping out at me.

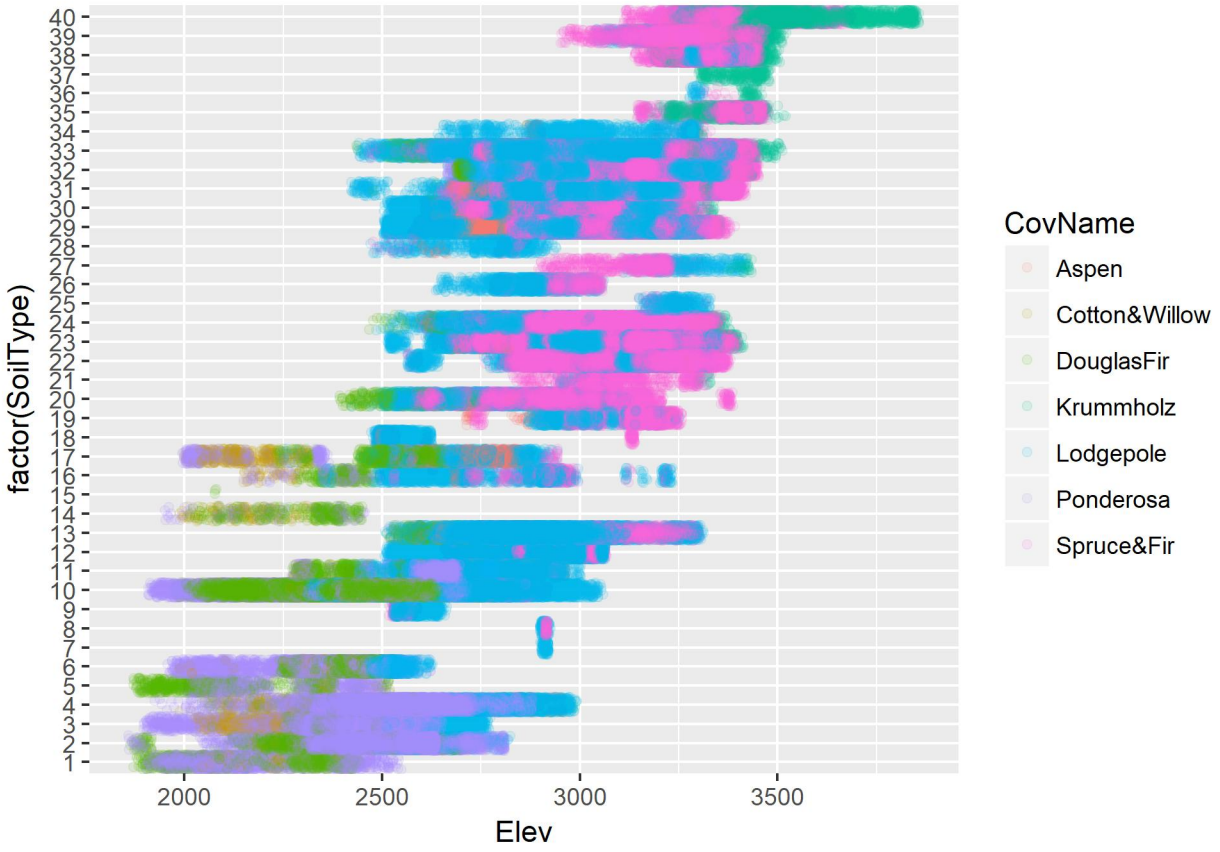


Figure 25: Aggregated Soil Type vs Elevation with Tree Type

**Aggregated Soil Type vs Elevation with Tree Type - Figure 25**

```
# Figure 25
g <- ggplot(forestcover,aes(Elev,factor(SoilType), col=CovName)) +
        geom_jitter(alpha=alphaVal)
  ggsave("Figure25.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Same thing as Figure 24 with axes reversed.

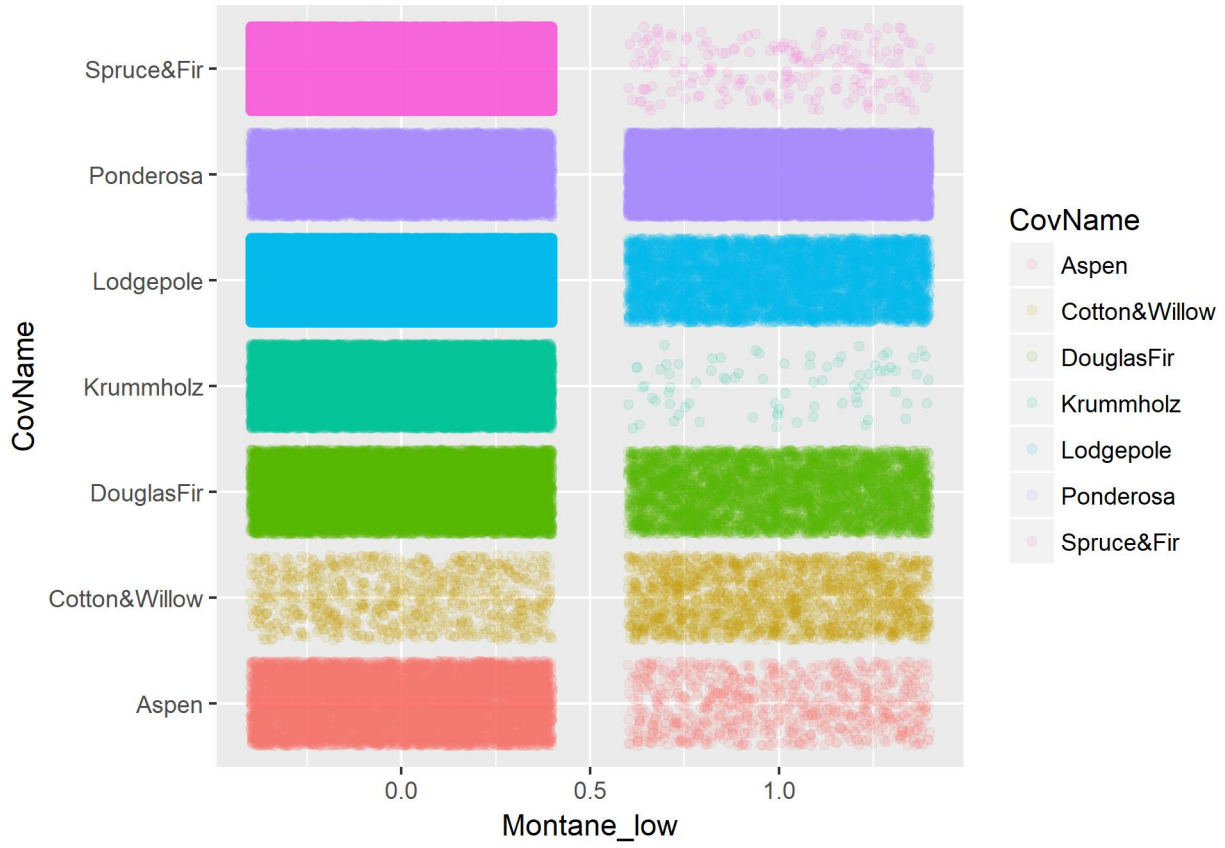


Figure 26: Tree Type vs "Montane Low" Soil Family

**Tree Type vs "Montane Low" Soil Family - Figure 26**

```
# Figure 26
g <- ggplot(forestcover,aes(Montane_low,CovName, col=CovName)) +
       geom_jitter(alpha=alphaVal)
  ggsave("Figure26.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Trying to determine how best to look at the different Soil Families. Looking at an individual family is not very pretty. We really only care about the entries with Montane_low value of '1'.

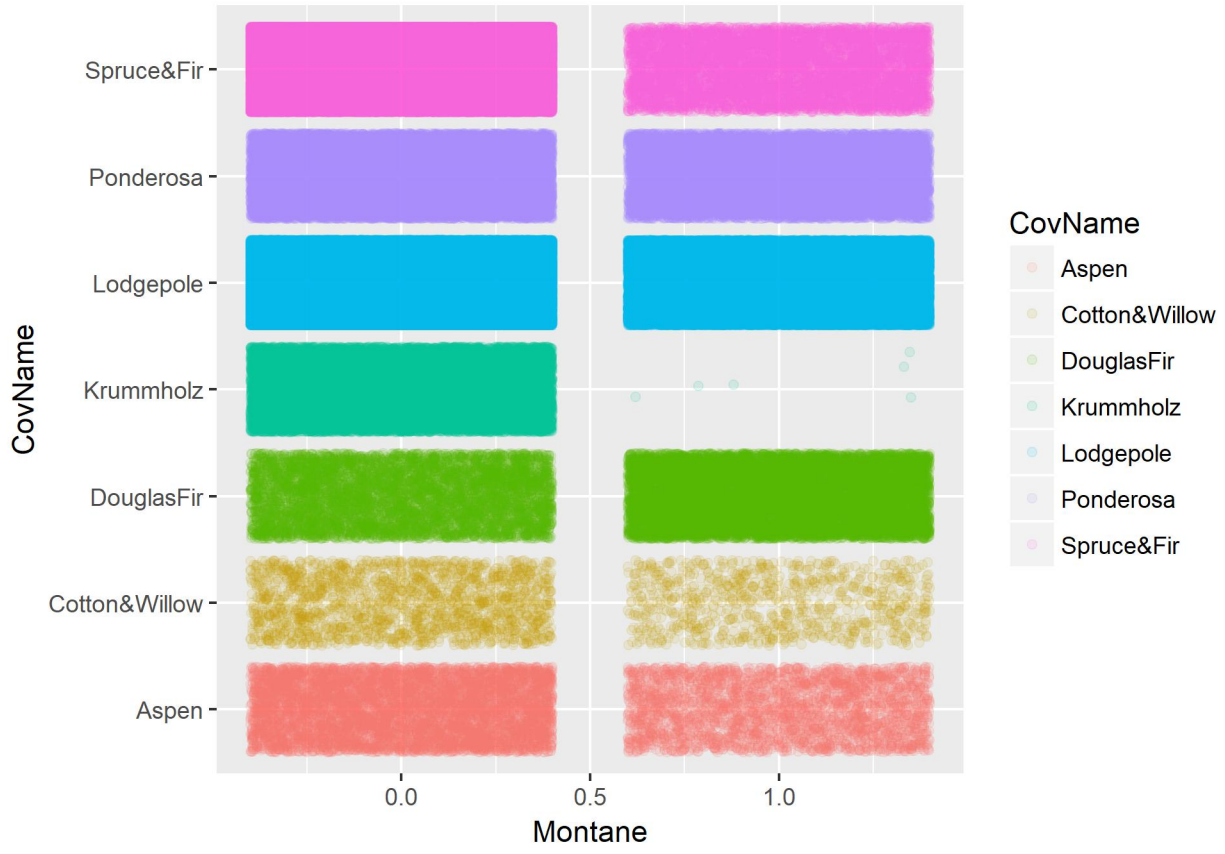


Figure 27: Tree Type vs "Montane" Soil Family

**Tree Type vs "Montane" Soil Family - Figure 27**

```
# Figure 27
g <- ggplot(forestcover,aes(Montane,CovName, col=CovName)) +
       geom_jitter(alpha=alphaVal)
  ggsave("Figure27.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Looking at 'Montane' soil family. No conclusions here.

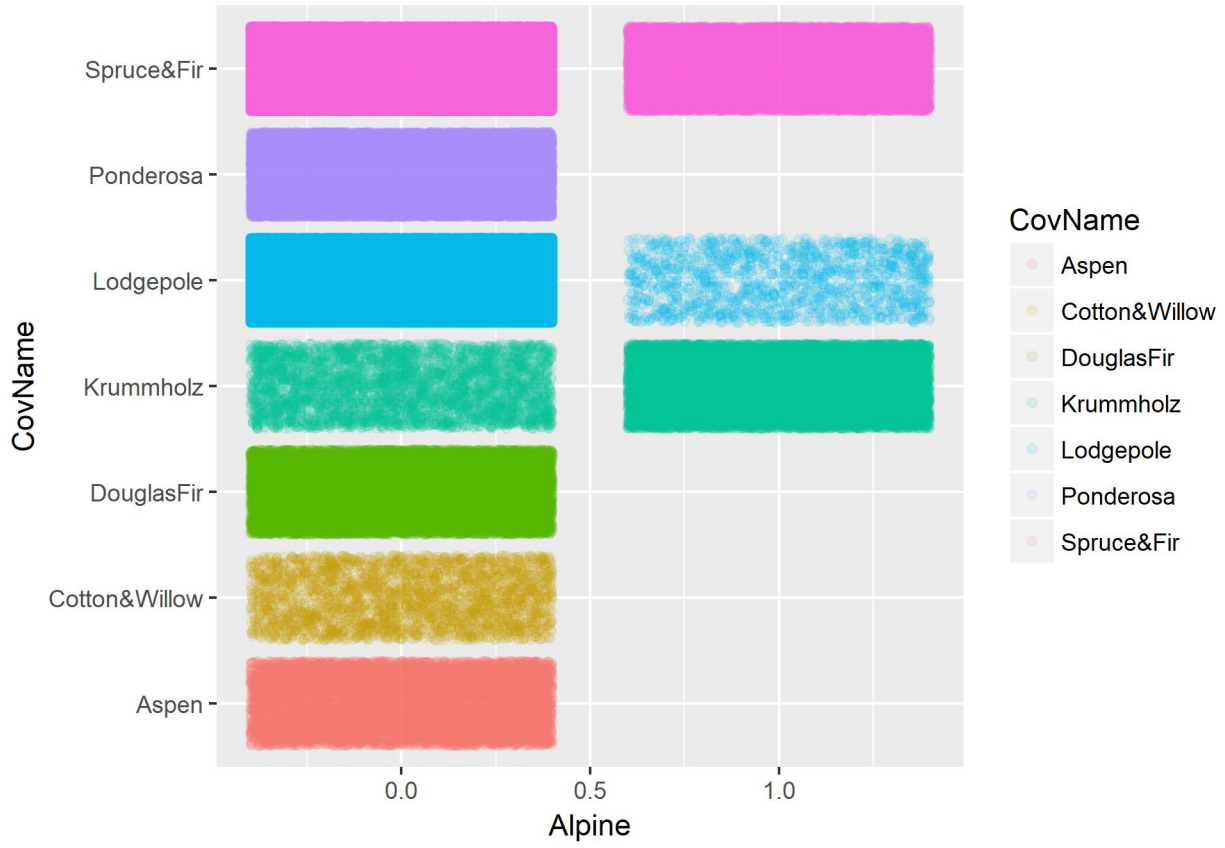


Figure 28: Tree Type vs "Alpine" Climate Zone

**Tree Type vs "Alpine" Climate Zone - Figure 28**

```
# Figure 28
g <- ggplot(forestcover,aes(Alpine,CovName, col=CovName)) +
       geom_jitter(alpha=alphaVal)
  ggsave("Figure28.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

The individual Alpine climate zone vs Tree cover. The Alpine Climate zone eliminates four of the tree types.

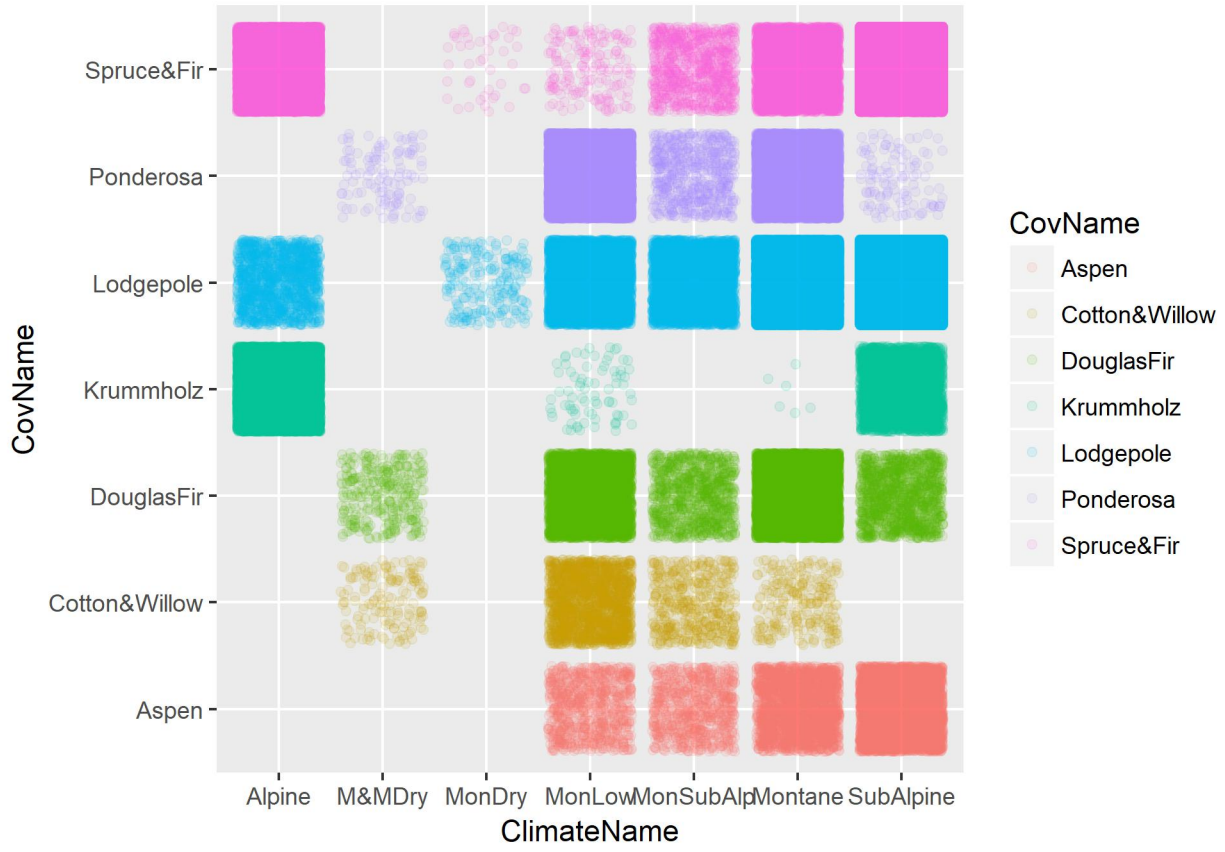


Figure 29: Tree Type vs Climate Zone

**Tree Type vs Climate Zone - Figure 29**

```
# Figure 29
g <- ggplot(forestcover,aes(ClimateName,CovName, col=CovName)) +
       geom_jitter(alpha=0.1)
  ggsave("Figure29.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

All of the Climate zones are graphed here. The Montane and Montane Dry (M&MDry) zone and the Montane Dry (MonDry) zones eliminate some tree types but the other zones are not as effective.

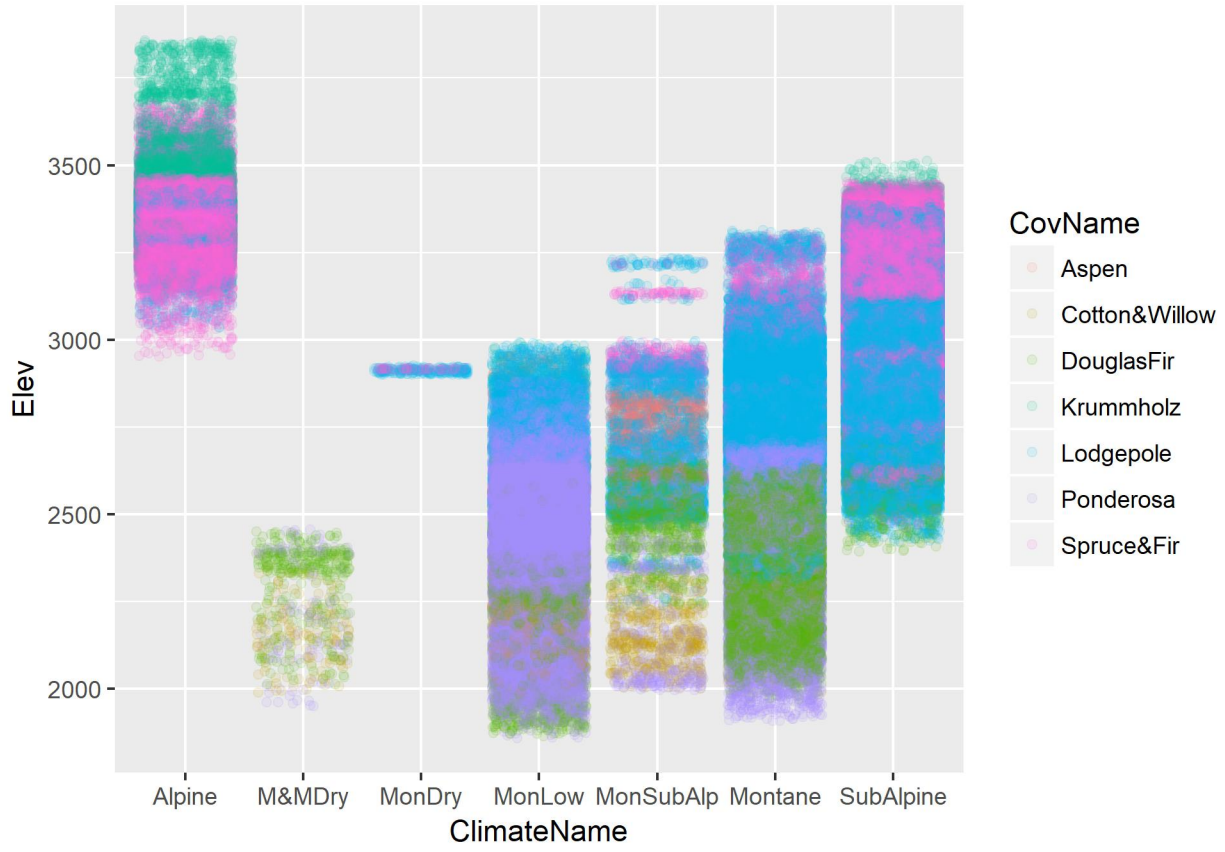


Figure 30: Elevation vs Climate with Tree Type

**Elevation vs Climate with Tree Type - Figure 30**

```
# Figure 30
g <- ggplot(forestcover,aes(ClimateName,Elev, col=CovName)) +
       geom_jitter(alpha=alphaVal)
  ggsave("Figure30.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

It's hard to see any patterns when looking at climate and Elevation with Tree type.

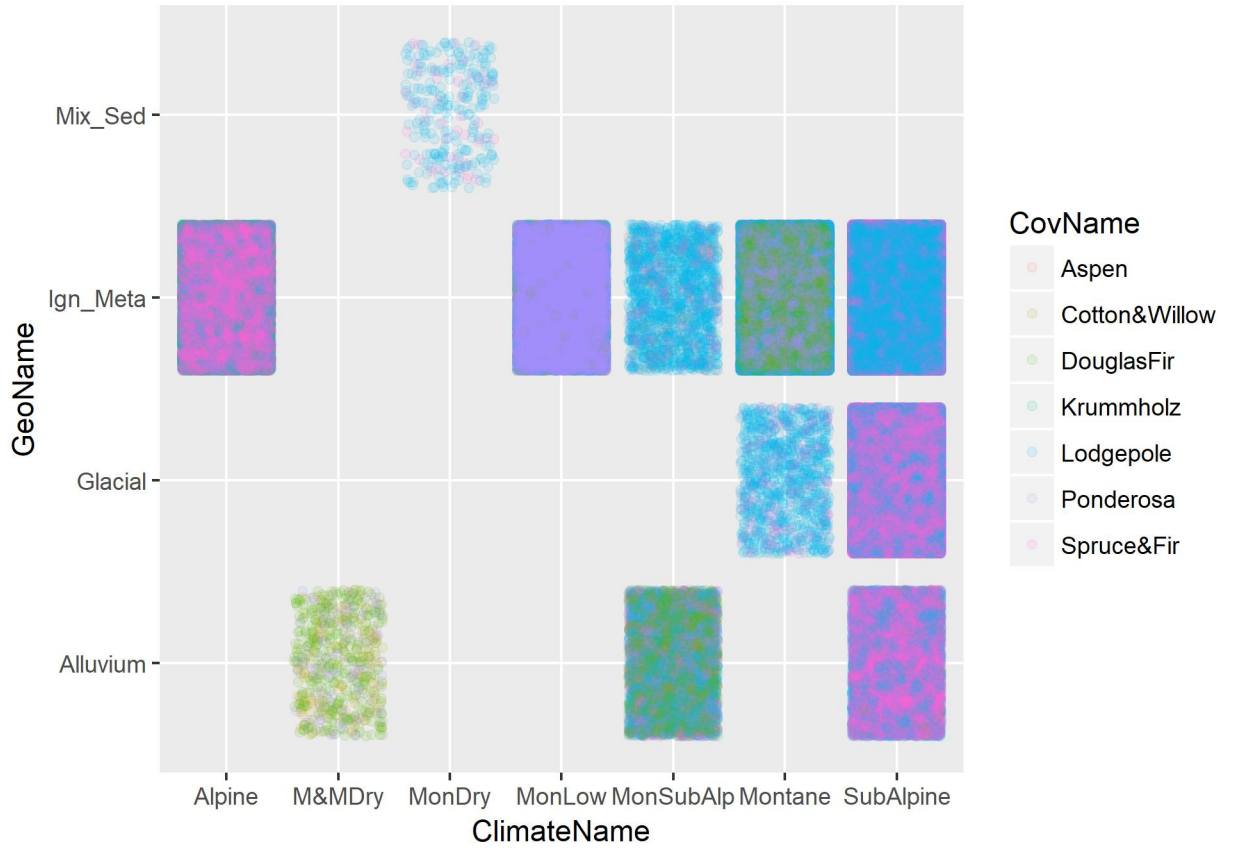


Figure 31: Geologic Zone vs Climate with Tree Type

**Geologic Zone vs Climate with Tree Type - Figure 31**

```
# Figure 31
g <- ggplot(forestcover,aes(ClimateName,GeoName, col=CovName)) +
       geom_jitter(alpha=alphaVal)
  ggsave("Figure31.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Plotting Climate and Geologic zones with Tree Type brings in too much data to see any clear patterns but it looks like it would aid in classifying the data.

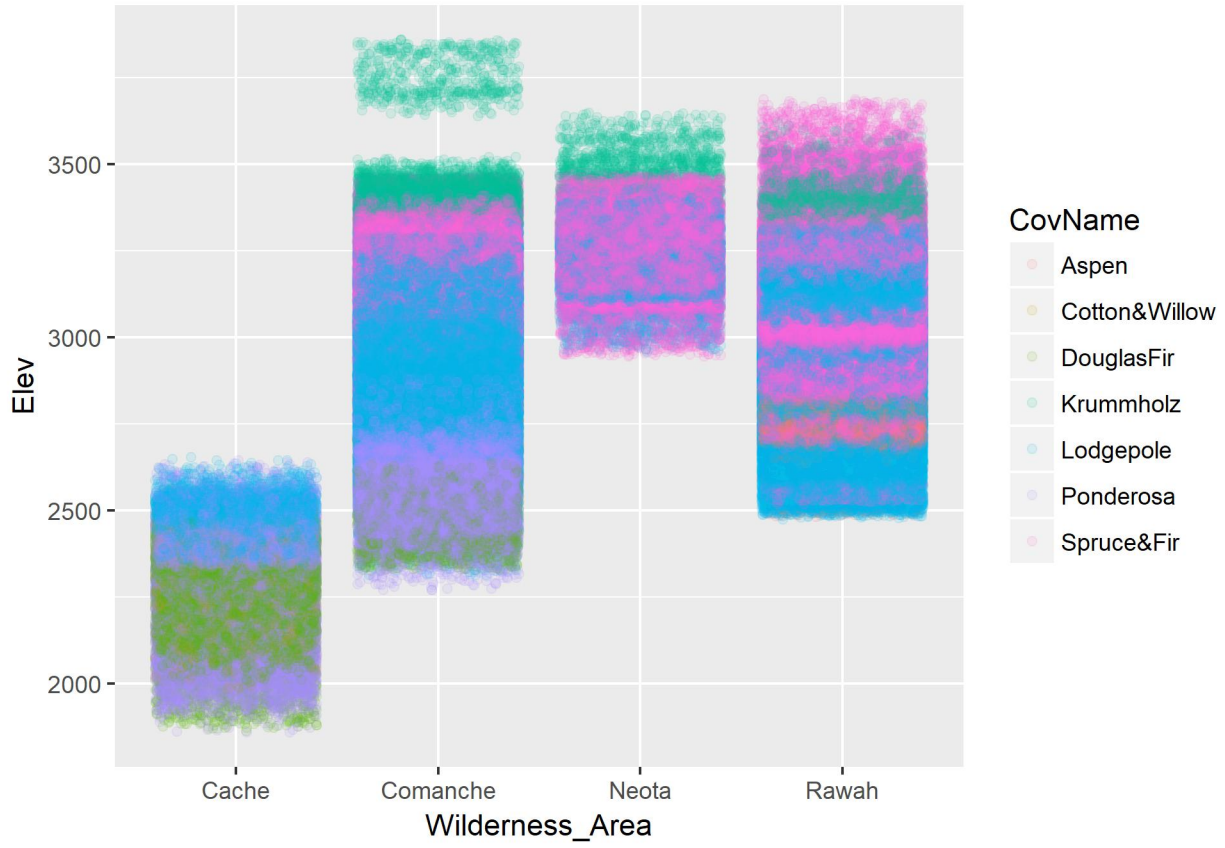


Figure 32: Elevation vs Wilderness Area with Tree Type

**Elevation vs Wilderness Area with Tree Type - Figure 32**

```
# Figure 32
g <- ggplot(forestcover,aes(Wilderness_Area,Elev,col=CovName)) +
        geom_jitter(alpha=0.1) # +
      #  facet_grid(. ~ CovName) +
  ggsave("Figure32.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Elevation vs Wilderness area shows the wilderness area should be able to help classifying tree type.

**Geology Count grouped Tree Type - Figure 33**

```
# Figure 33
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------- tidyverse
```

```
## v tibble  1.4.2     v purrr   0.2.4
## v tidyr   0.8.0     v stringr 1.3.0
## v readr   1.1.1     v forcats 0.3.0
```

```
## -- Conflicts ------------------------------------------------------------------------ tidyverse_conflicts
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
col1 <- grep("^CovName$", colnames(forestcover))
col2 <- grep("^Alluvium$", colnames(forestcover))
col3 <- grep("^Ign_Meta$", colnames(forestcover))
cols=c(col1,col2:col3)
#td2 <- gather(forestcover,Property,val,cols)
td2 <- forestcover[,cols]
td3<-gather(td2,Geology,Type,-1)
td4<- group_by(td3,CovName,Geology) %>%
  summarize(tot=sum(Type))
gr <- ggplot(td4, aes(CovName, tot, fill = Geology)) +
    geom_bar(stat = "identity", position = "dodge")
ggsave("Figure33.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

Plotting histograms of Tree Type grouped by Geologic Zone shows a similar shape between Geologic Zones.

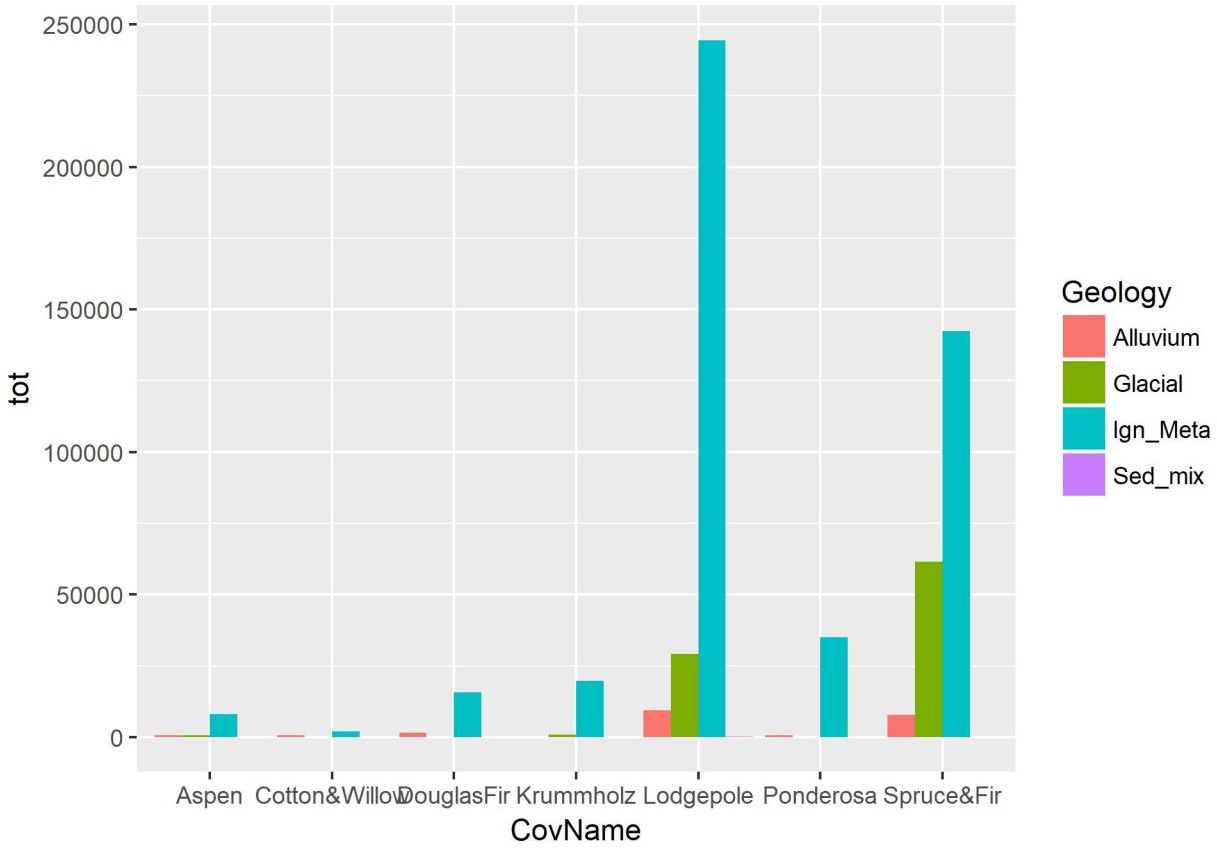


Figure 33: Geology Count grouped Tree Type

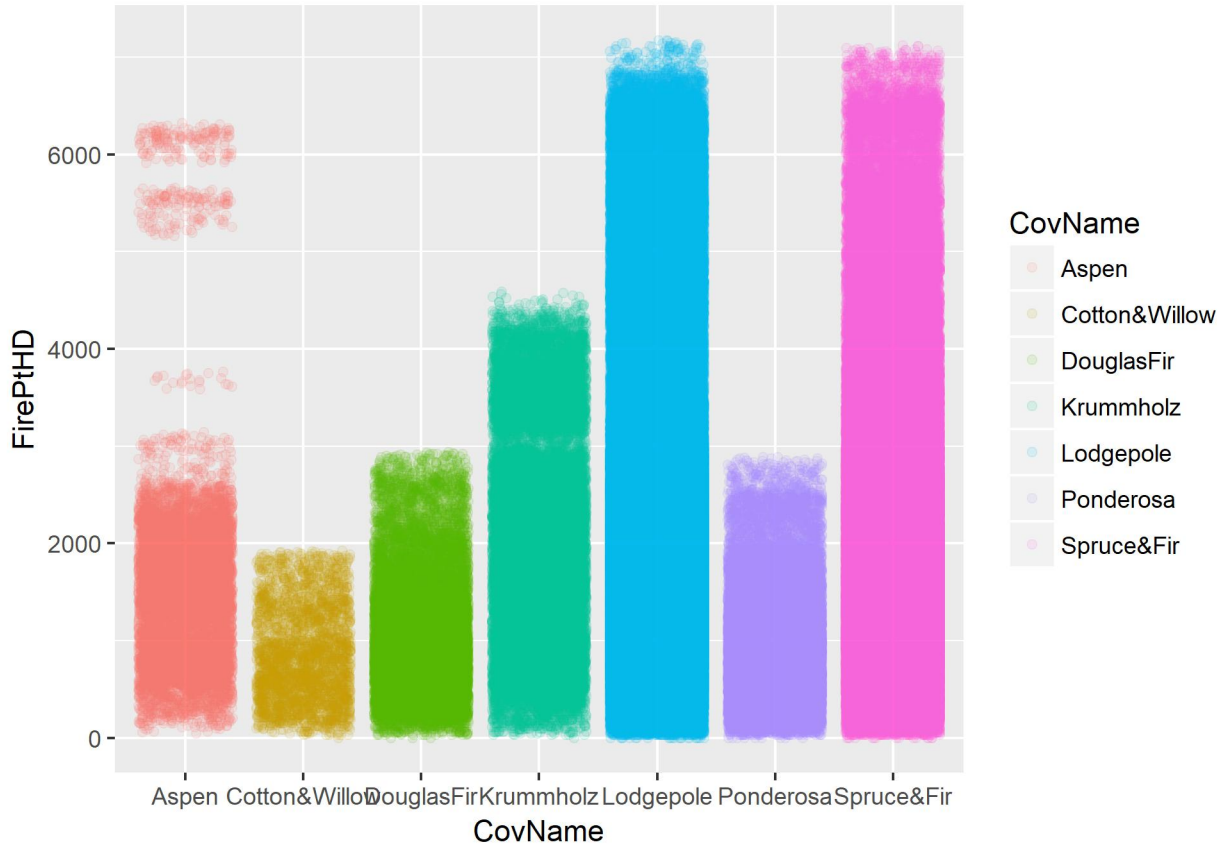


Figure 34: Fire Point Horizontal Distance vs Tree Type

**Fire Point Horizontal Distance vs Tree Type - Figure 34**

```
# Figure 34
  g <- ggplot(forestcover,aes(CovName, FirePtHD,col=CovName)) +
        geom_jitter(alpha=0.1) # +
      #  facet_grid(. ~ CovName) +
  ggsave("Figure34.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

It looks like the Fire Point distance can help aid in classifying the tree type by eliminating some tree types based on increasing distance.

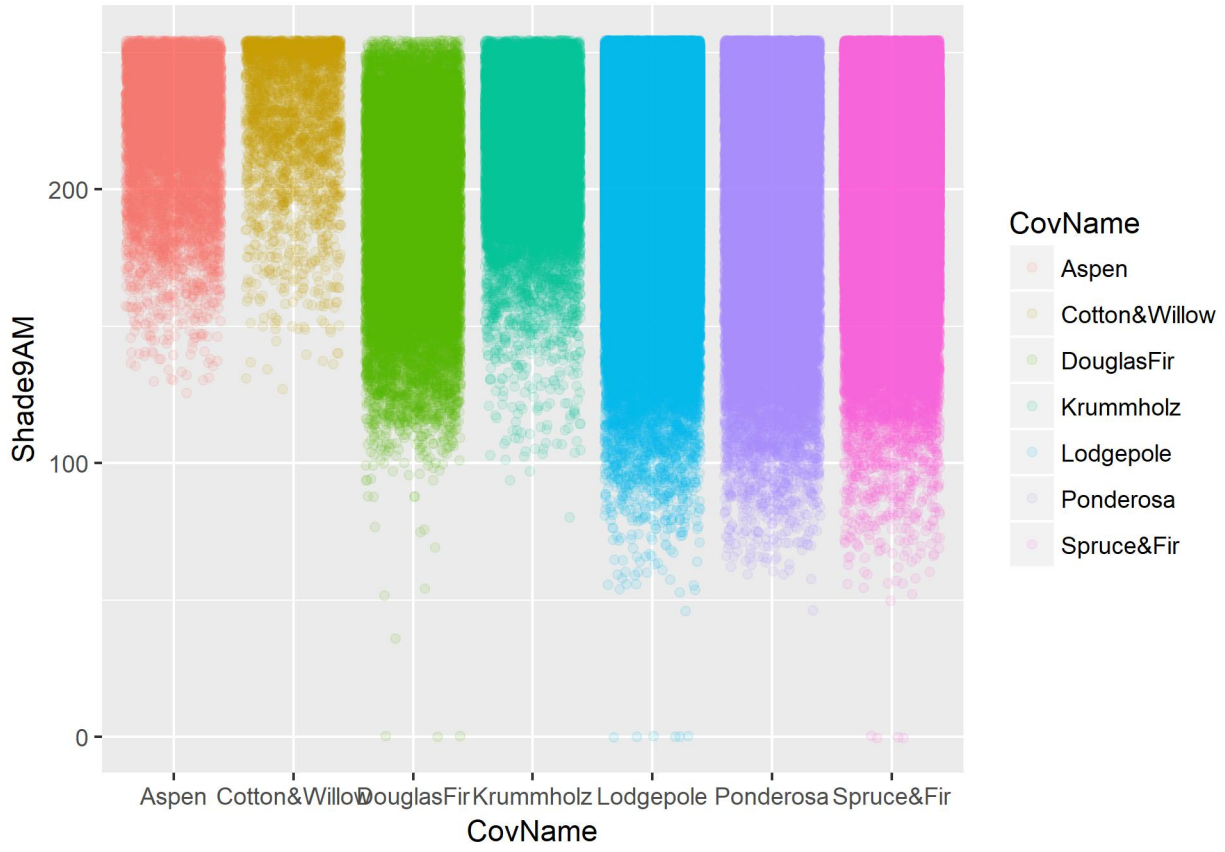


Figure 35: 9AM Shade vs Tree Type

**9AM Shade vs Tree Type - Figure 35**

```
# Figure 35
  g <- ggplot(forestcover,aes(CovName, Shade9AM,col=CovName)) +
        geom_jitter(alpha=0.1) # +
      #  facet_grid(. ~ CovName) +
  ggsave("Figure35.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

The Shade9AM value can be used to eliminate some trees based on low shade value.

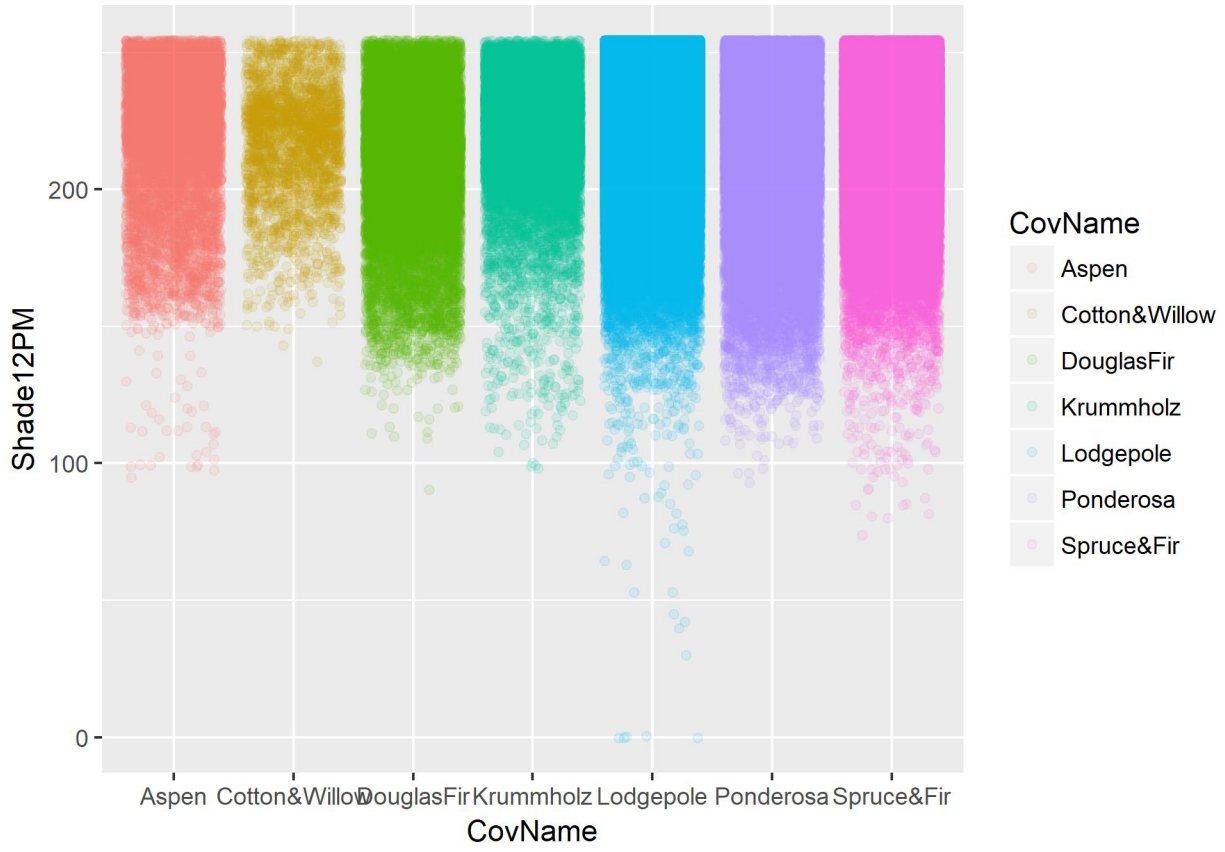


Figure 36: 12PM Shade vs Tree Type

**12PM Shade vs Tree Type - Figure 36**

```
# Figure 36
  g <- ggplot(forestcover,aes(CovName, Shade12PM,col=CovName)) +
        geom_jitter(alpha=0.1) # +
      #  facet_grid(. ~ CovName) +
  ggsave("Figure36.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

It looks like Shade12PM value can be used to help classify trees similar to Shade9AM but not as effective.

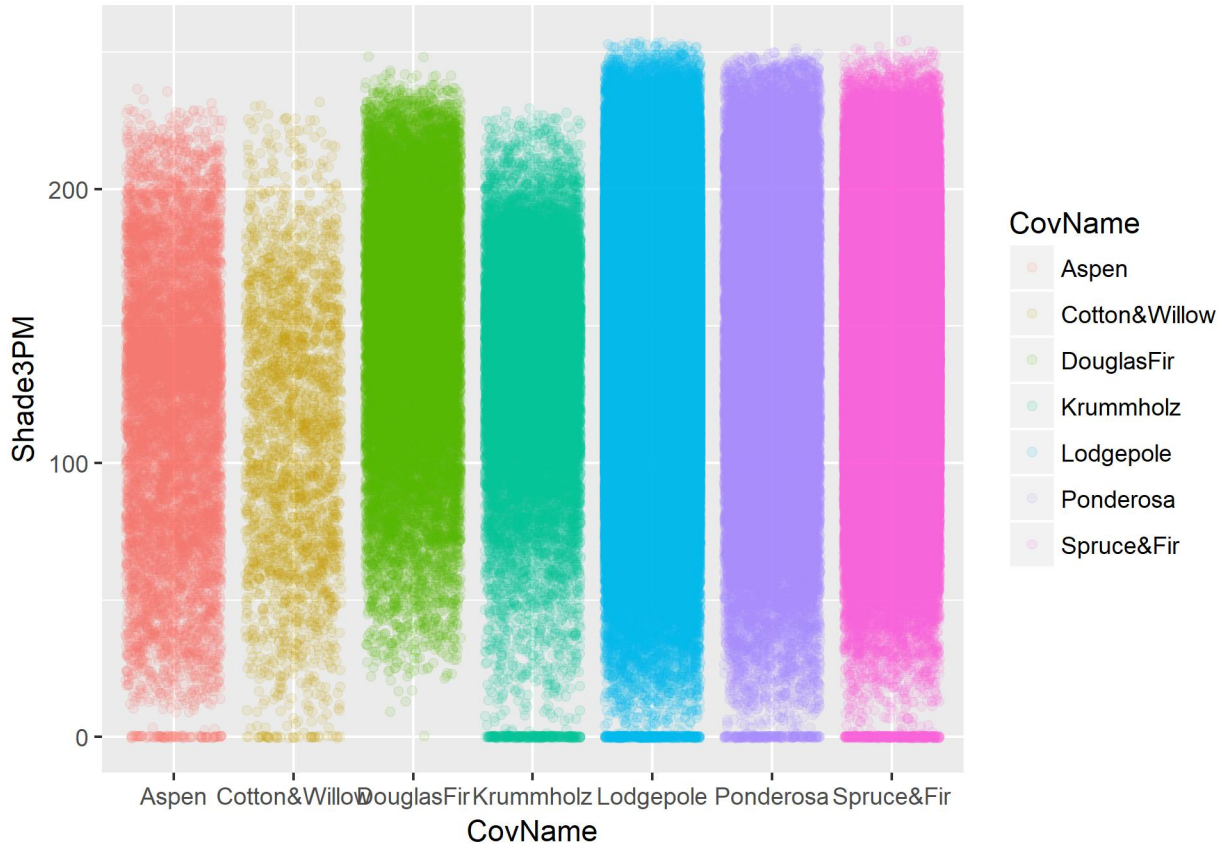


Figure 37: 3PM Shade vs Tree Type

**3PM Shade vs Tree Type - Figure 37**

```
# Figure 37
  g <- ggplot(forestcover,aes(CovName, Shade3PM,col=CovName)) +
        geom_jitter(alpha=0.1) # +
      #  facet_grid(. ~ CovName) +
  ggsave("Figure37.jpg")
```

```
## Saving 6.5 x 4.5 in image
```

The Shade3PM data does not look like it will help much to help classify tree type.

```
  endTime=Sys.time()
  print(paste("Figures completed at",endTime))
```

```
## [1] "Figures completed at 2018-04-18 15:18:07"
```

```
  print(paste("Elapsed time=",round(endTime-startTime),"seconds."))
```

```
## [1] "Elapsed time= 28 seconds."
```

```
progEnd=Sys.time()
print(paste("R script  started  at",progStart))
```

```
## [1] "R script  started  at 2018-04-18 14:49:59"
```

```
print(paste("R script completed at",progEnd))
```

```
## [1] "R script completed at 2018-04-18 15:18:07"
```

```
#print(paste("Elapsed time=",progEnd-progStart,"seconds."))
```

This concludes the current data exploration on my capstone data.