Forest Cover Analysis Notebook

Forest Cover Type Classificartion Problem

UCI Cover Type Dataset: https://archive.ics.uci.edu/ml/datasets/covertype (https://archive.ics.uci.edu/ml/datasets/covertype) UCI dataset link: https://archive.ics.uci.edu/ml/machine-learning-databases/covtype/ (https://archive.ics.uci.edu/ml/machine-learning-databases/covtype/)

```
#segData <- segData[, -(1:3)] remove columns
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------------
--------------------------------------------- tidyverse 1.2.1 --
```

```
## v ggplot2 3.0.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.6
## v tidyr   0.8.1     v stringr 1.3.1
## 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()
```

```
ct <- read.csv("C:/Users/Zeus/Desktop/DS Practicum/Kaggle Project/covtype.data") #UCI raw data
#Name UCI data Variables
names(ct) <- c("Elevation", "Aspect", "Slope", "Horizontal_Distance_To_Hydrology", "Vertical_Distance_To_Hydrol
ogy",
                        "Horizontal_Distance_To_Roadways", "Hillshade_9am", "Hillshade_Noon", "Hillshade_3pm",
"Horizontal_Distance_To_Fire_Points",
                        "Wilderness_Area1", "Wilderness_Area2", "Wilderness_Area3", "Wilderness_Area4", "Soil_T
ype1", "Soil_Type2","Soil_Type3","Soil_Type4",
                        "Soil_Type5","Soil_Type6","Soil_Type7","Soil_Type8","Soil_Type9","Soil_Type10","Soil_Ty
pe11","Soil_Type12","Soil_Type13","Soil_Type14",
                        "Soil_Type15","Soil_Type16","Soil_Type17","Soil_Type18","Soil_Type19","Soil_Type20","So
il_Type21","Soil_Type22","Soil_Type23","Soil_Type24",
                        "Soil_Type25","Soil_Type26","Soil_Type27","Soil_Type28","Soil_Type29","Soil_Type30","So
il_Type31","Soil_Type32","Soil_Type33","Soil_Type34",
                        "Soil_Type35","Soil_Type36","Soil_Type37","Soil_Type38","Soil_Type39","Soil_Type40", "C
over_Type")
str(ct)
```

```
## 'data.frame':    581011 obs. of  55 variables:
##  $ Elevation                         : int  2590 2804 2785 2595 2579 2606 2605 2617 2612 2612 ...
##  $ Aspect                            : int  56 139 155 45 132 45 49 45 59 201 ...
##  $ Slope                             : int  2 9 18 2 6 7 4 9 10 4 ...
##  $ Horizontal_Distance_To_Hydrology  : int  212 268 242 153 300 270 234 240 247 180 ...
##  $ Vertical_Distance_To_Hydrology    : int  -6 65 118 -1 -15 5 7 56 11 51 ...
##  $ Horizontal_Distance_To_Roadways   : int  390 3180 3090 391 67 633 573 666 636 735 ...
##  $ Hillshade_9am                     : int  220 234 238 220 230 222 222 223 228 218 ...
##  $ Hillshade_Noon                    : int  235 238 238 234 237 225 230 221 219 243 ...
##  $ Hillshade_3pm                     : int  151 135 122 150 140 138 144 133 124 161 ...
##  $ Horizontal_Distance_To_Fire_Points: int  6225 6121 6211 6172 6031 6256 6228 6244 6230 6222 ...
##  $ Wilderness_Area1                  : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Wilderness_Area2                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Wilderness_Area3                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Wilderness_Area4                  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type1                        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type2                        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type3                        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type4                        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type5                        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type6                        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type7                        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type8                        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type9                        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type10                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type11                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type12                       : int  0 1 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type13                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type14                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type15                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type16                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type17                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type18                       : int  0 0 0 0 0 0 0 0 0 1 ...
##  $ Soil_Type19                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type20                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type21                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type22                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type23                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type24                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type25                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type26                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type27                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type28                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type29                       : int  1 0 0 1 1 1 1 1 1 0 ...
##  $ Soil_Type30                       : int  0 0 1 0 0 0 0 0 0 0 ...
##  $ Soil_Type31                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type32                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type33                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type34                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type35                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type36                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type37                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type38                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type39                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type40                       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Cover_Type                        : int  5 2 2 5 2 5 5 5 5 5 ...
```

```
#Predictor Variable Dispersion in data set Table numbers and proportions
cbind(table(ct$Cover_Type), round(prop.table(table(ct$Cover_Type)),3)*100)
```

```
##     [,1] [,2]
## 1 211840 36.5
## 2 283301 48.8
## 3  35754  6.2
## 4   2747  0.5
## 5   9492  1.6
## 6  17367  3.0
## 7  20510  3.5
```

```r
ct$Cover_Type <- factor(ct$Cover_Type) #change to factor
ct$Cover_Type <- fct_recode(ct$Cover_Type, #code numbers to descriptive names
                                  "Spuce_Fir" = "1",
                                  "Lodgepole" = "2",
                                  "Ponderosa" = "3",
                                  "CottonWd_Willow" = "4",
                                  "Aspen" = "5",
                                  "DougFir" = "6",
                                  "Krummholz" ="7")
#Group Regions for visual
ct$region <- names(ct[11:14])[apply(ct[11:14], 1, match, x = 1)]
ct$region <- factor(ct$region)
#Check Proportions
cbind(table(ct$region), round(prop.table(table(ct$region)),3)*100)
```

```
##                    [,1] [,2]
## Wilderness_Area1 260795 44.9
## Wilderness_Area2  29884  5.1
## Wilderness_Area3 253364 43.6
## Wilderness_Area4  36968  6.4
```

```r
#Soil Type into one category
ct$soil <- names(ct[15:54])[apply(ct[15:54], 1, match, x = 1)]
ct$soil <- factor(ct$soil)
#Check Proportions
cbind(table(ct$soil), round(prop.table(table(ct$soil)),3)*100)
```

```
##                [,1] [,2]
## Soil_Type1    3031  0.5
## Soil_Type10  32634  5.6
## Soil_Type11  12410  2.1
## Soil_Type12  29971  5.2
## Soil_Type13  17431  3.0
## Soil_Type14    599  0.1
## Soil_Type15      3  0.0
## Soil_Type16   2845  0.5
## Soil_Type17   3422  0.6
## Soil_Type18   1899  0.3
## Soil_Type19   4021  0.7
## Soil_Type2    7525  1.3
## Soil_Type20   9259  1.6
## Soil_Type21    838  0.1
## Soil_Type22  33373  5.7
## Soil_Type23  57752  9.9
## Soil_Type24  21278  3.7
## Soil_Type25    474  0.1
## Soil_Type26   2589  0.4
## Soil_Type27   1086  0.2
## Soil_Type28    946  0.2
## Soil_Type29 115246 19.8
## Soil_Type3    4823  0.8
## Soil_Type30  30170  5.2
## Soil_Type31  25666  4.4
## Soil_Type32  52519  9.0
## Soil_Type33  45154  7.8
## Soil_Type34   1611  0.3
## Soil_Type35   1891  0.3
## Soil_Type36    119  0.0
## Soil_Type37    298  0.1
## Soil_Type38  15573  2.7
## Soil_Type39  13806  2.4
## Soil_Type4   12396  2.1
## Soil_Type40   8750  1.5
## Soil_Type5    1597  0.3
## Soil_Type6    6575  1.1
## Soil_Type7     105  0.0
## Soil_Type8     179  0.0
## Soil_Type9    1147  0.2
```

```
#Group Aspects into compass directions per standards
ct <- within(ct,{
  direction <- NA
  direction[Aspect <= 22.5] <- "North"
  direction[Aspect >= 22.6 & Aspect <= 67.5] <- "Northeast"
  direction[Aspect >= 67.6 & Aspect <= 112.5] <- "East"
  direction[Aspect >= 112.6 & Aspect <= 157.5] <- "Southeast"
  direction[Aspect >= 157.6 & Aspect <= 202.5] <- "South"
  direction[Aspect >= 202.6 & Aspect <= 247.5] <- "Southwest"
  direction[Aspect >= 247.6 & Aspect <= 292.5] <- "West"
  direction[Aspect >= 292.6 & Aspect <= 337.5] <- "Northwest"
  direction[Aspect >= 337.6 & Aspect <= 360] <- "North"
})
ct$direction <- factor(ct$direction)
#Check Proportions
cbind(table(ct$direction), round(prop.table(table(ct$direction)),3)*100)
```

```
##             [,1] [,2]
## East       96464 16.6
## North      96106 16.5
## Northeast 113905 19.6
## Northwest  70927 12.2
## South      49751  8.6
## Southeast  69914 12.0
## Southwest  40004  6.9
## West       43940  7.6
```

```
#Group Elevations into Plant Growth Zones for ROcky Mountain Region in COlorado
ct <- within(ct,{
  zone <- NA
  zone[Elevation >= 3353] <- "4_Alpine"
  zone[Elevation >= 2743 & Elevation <= 3352] <- "3_Subalpine"
  zone[Elevation >=2134 & Elevation <= 2742] <- "2_Montane"
  zone[Elevation <= 2133] <- "1_Foothills" })
ct$zone <- factor(ct$zone)
#Check Proportions
cbind(table(ct$zone), round(prop.table(table(ct$zone)),3)*100)
```

```
##               [,1] [,2]
## 1_Foothills   6423  1.1
## 2_Montane   105912 18.2
## 3_Subalpine 444548 76.5
## 4_Alpine     24128  4.2
```

```
#prepare as numeric
ct$Horizontal_Distance_To_Hydrology <- as.numeric(ct$Horizontal_Distance_To_Hydrology)
ct$Vertical_Distance_To_Hydrology <- as.numeric(ct$Vertical_Distance_To_Hydrology)
#Find Approx Surface Distance - Pythagorean Theorum
ct$Dist_water <- sqrt(ct$Horizontal_Distance_To_Hydrology^2 + ct$Vertical_Distance_To_Hydrology^2)
#Remove to not correlate
ct$Horizontal_Distance_To_Hydrology <- NULL
ct$Vertical_Distance_To_Hydrology <- NULL
#Check min-max
summary(ct$Dist_water)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0   108.5   229.5   276.1   393.8  1418.9
```

```
#Make a random small data sample to visualize the UCI data, is too large
library(caret)
```

```
## Loading required package: lattice
```

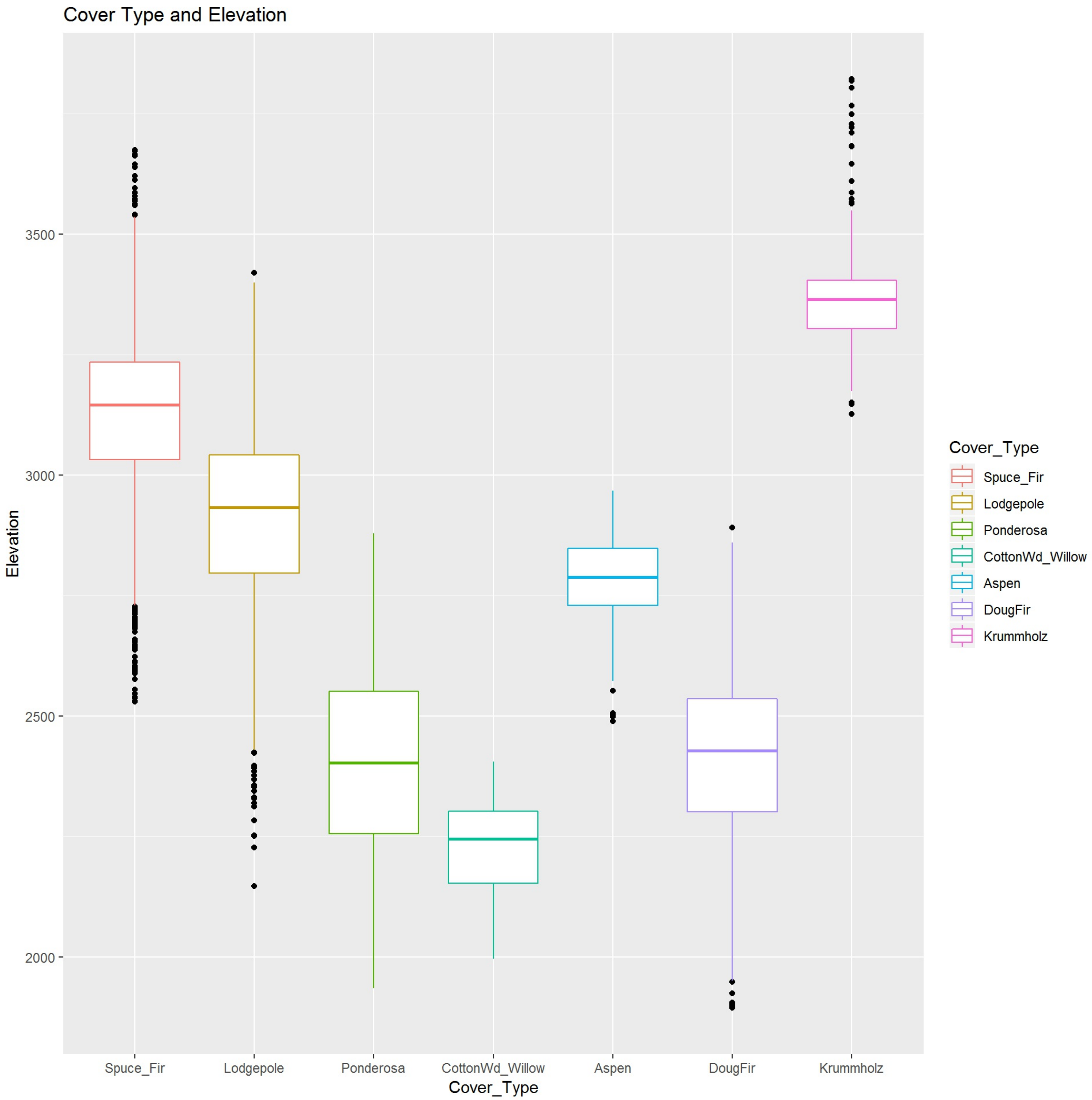
```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

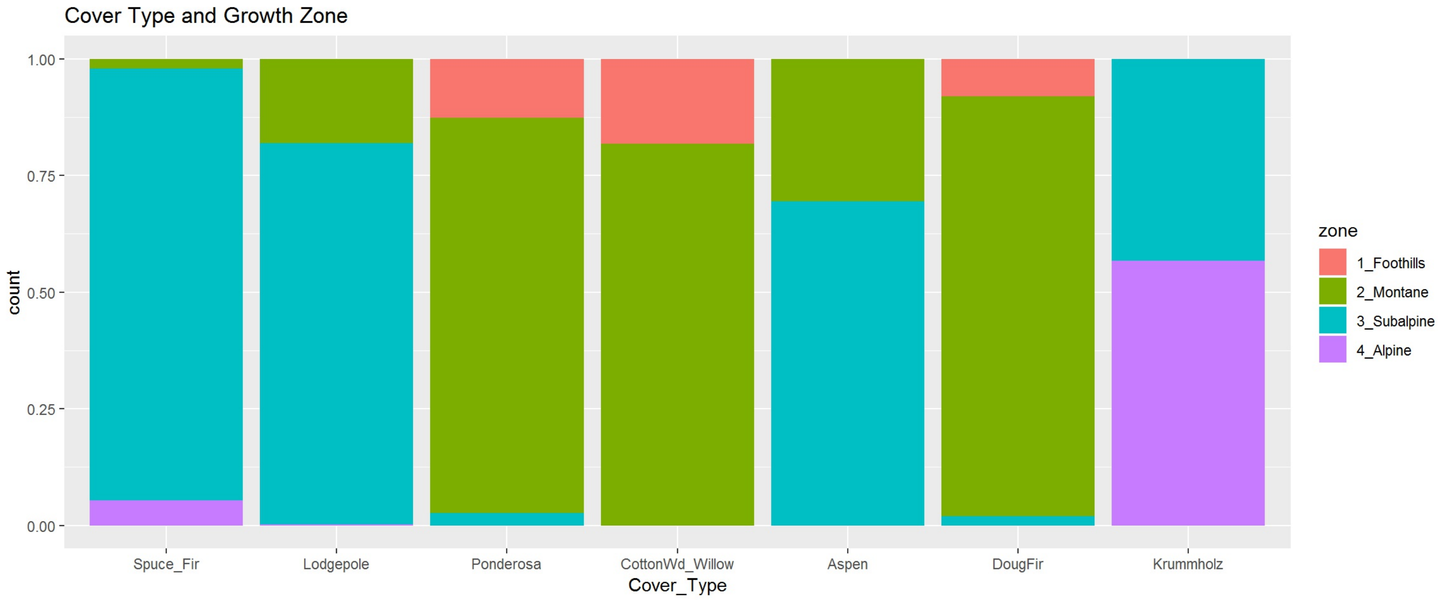
```
set.seed(1)
smlSet <- createDataPartition(ct$Cover_Type, p=.02,
                                  list = FALSE,
                                  times = 1)
smlSet <- ct[ smlSet,]
#check proportions for dependent variable
cbind(table(smlSet$Cover_Type), round(prop.table(table(smlSet$Cover_Type)),3)*100)
```

```
##                  [,1] [,2]
## Spuce_Fir        4237 36.5
## Lodgepole        5667 48.8
## Ponderosa         716  6.2
## CottonWd_Willow    55  0.5
## Aspen             190  1.6
## DougFir           348  3.0
## Krummholz         411  3.5
```

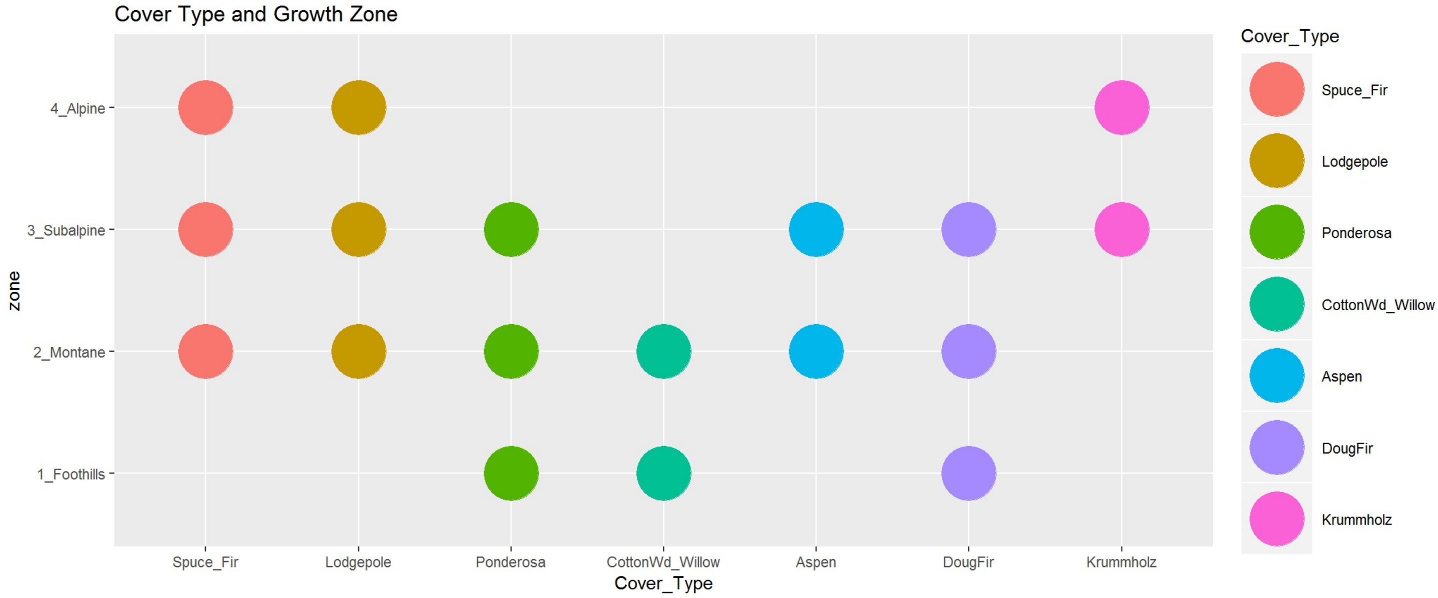
```
#Boxplots in color and outliers in black
ggplot(smlSet, aes(Cover_Type, Elevation)) +
  geom_boxplot(aes(color = Cover_Type), outlier.color = "black") + ggtitle("Cover Type and Elevation")
```



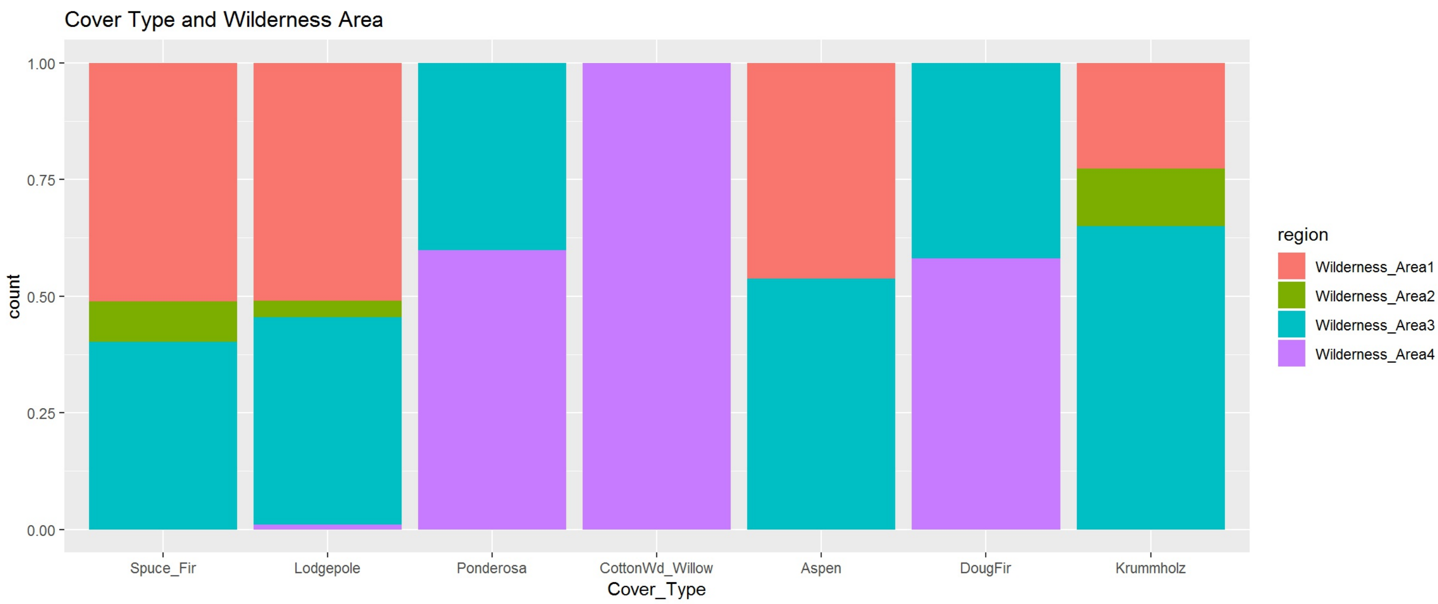

```
#BarPlot Cover Type to Zone(Elevation), note the growth zones of each tree
ggplot(smlSet) +
  geom_bar(mapping =aes(x =Cover_Type, fill=zone), position = 'fill') + ggtitle("Cover Type and Growth Zone")
```



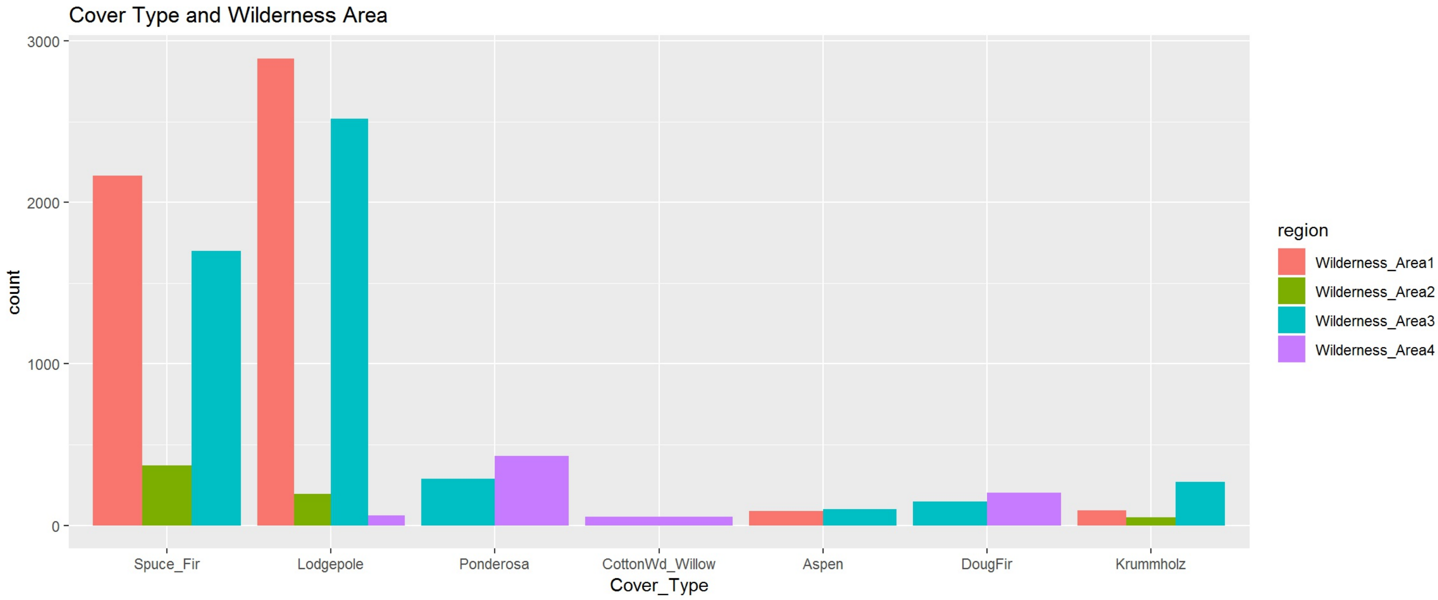

```
#plots CoverType to Elevation
ggplot(smlSet, aes(Cover_Type, zone, color=Cover_Type)) +
  geom_point(pch=19, size=15)  + ggtitle("Cover Type and Growth Zone")
```



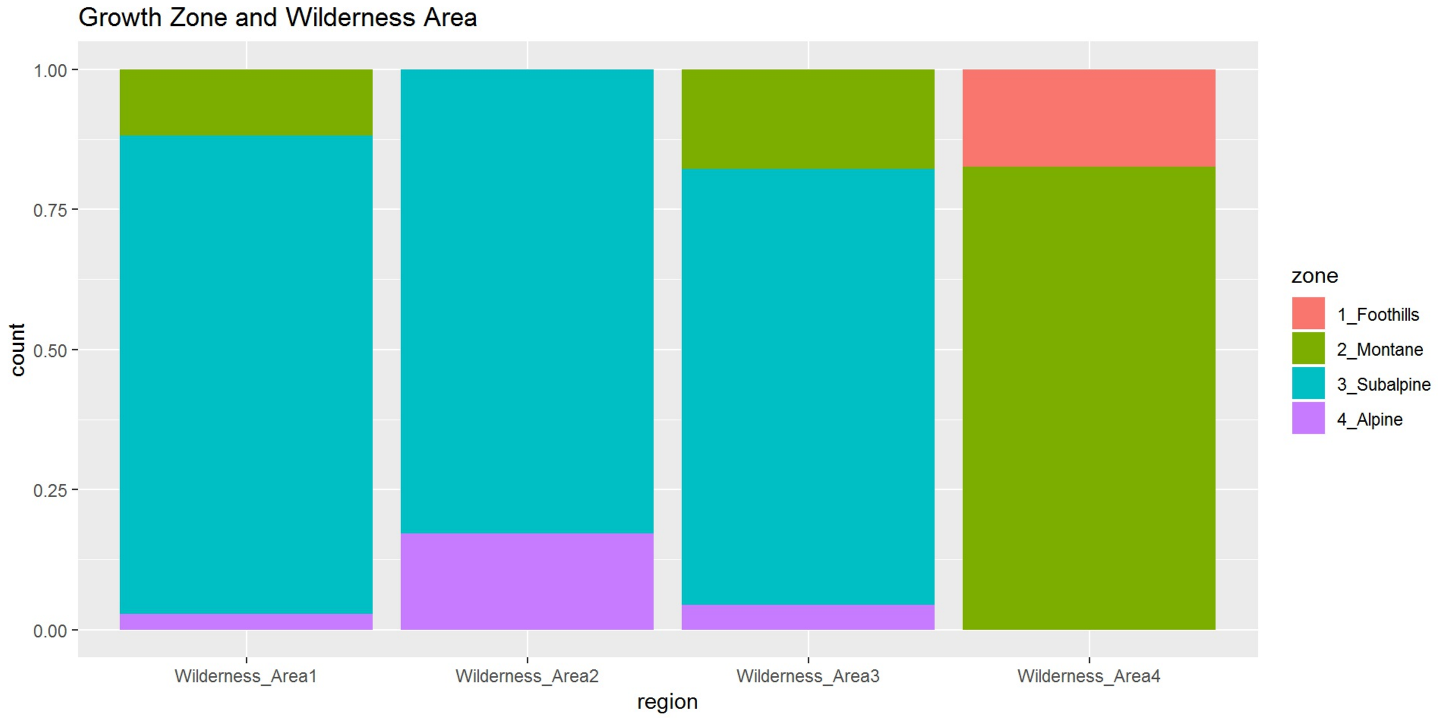

```
#Look at the Region and Covertype
ggplot(smlSet) +
  geom_bar(mapping =aes(x =Cover_Type, fill=region), position = 'fill')  + ggtitle("Cover Type and Wilderness A
rea")
```



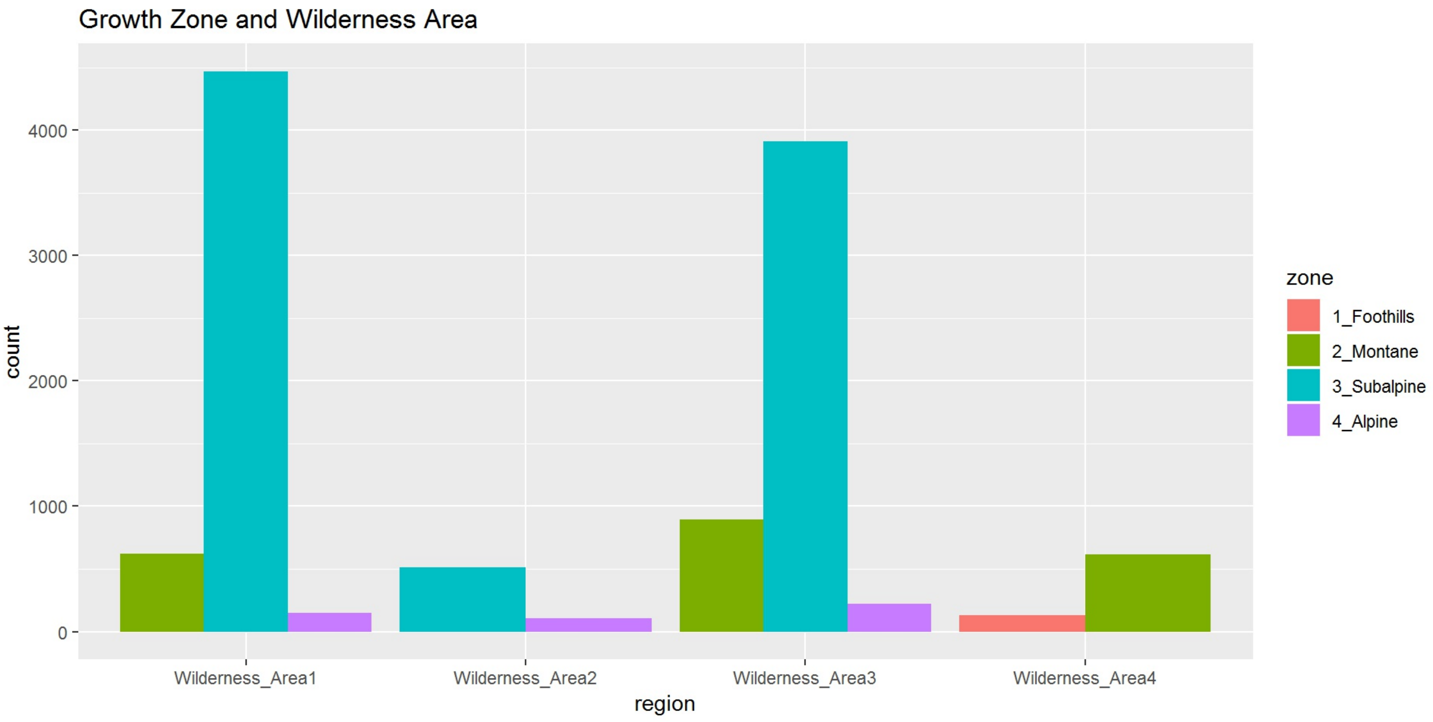
```
ggplot(smlSet) +
  geom_bar(mapping =aes(x =Cover_Type, fill=region), position = 'dodge') + ggtitle("Cover Type and Wilderness A
rea")
```



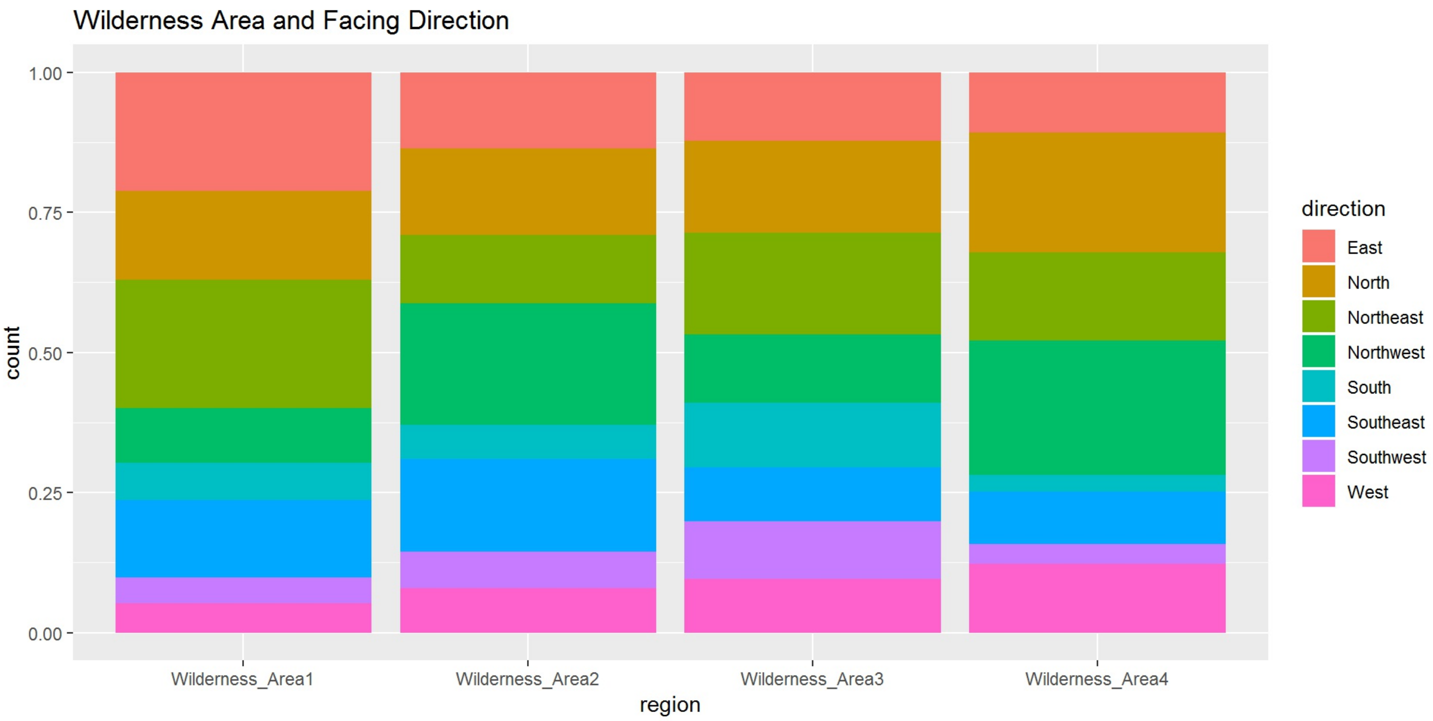

```
#Look at Region and Zone(Elevation)
ggplot(smlSet) +
  geom_bar(mapping =aes(x =region, fill=zone), position = 'fill') + ggtitle("Growth Zone and Wilderness Area")
```



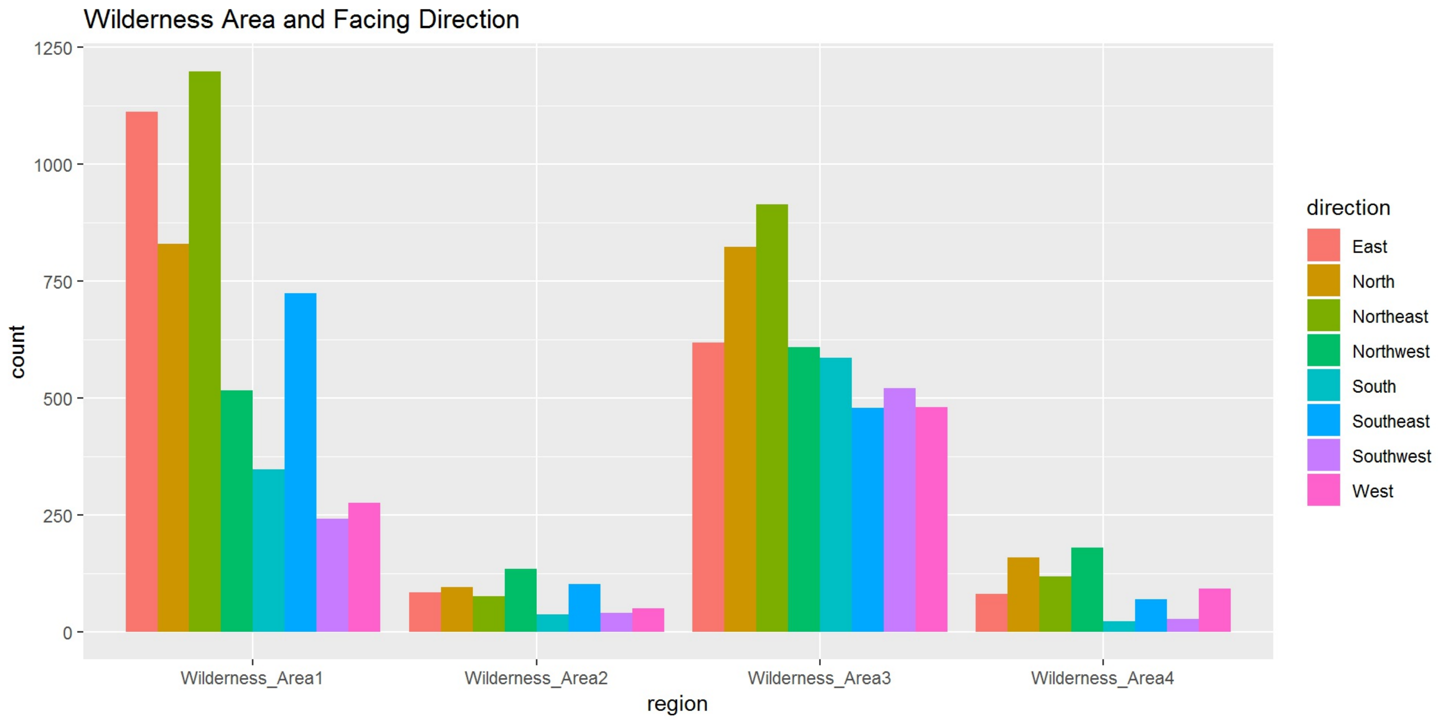

```
ggplot(smlSet) +
  geom_bar(mapping =aes(x =region, fill=zone), position = 'dodge') + ggtitle("Growth Zone and Wilderness Area")
```



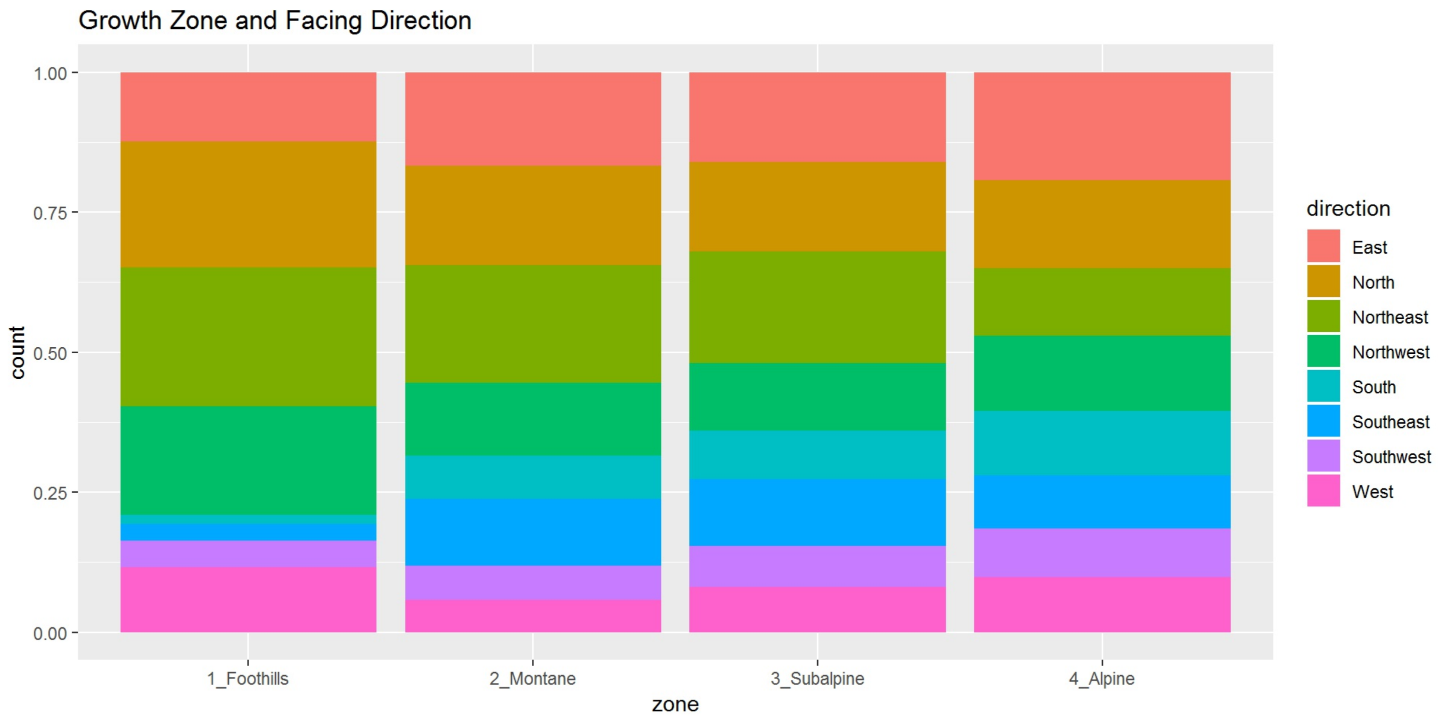

```
#Look at direction and CoverType
ggplot(smlSet) +
  geom_bar(mapping =aes(x =region, fill=direction), position = 'fill') + ggtitle("Wilderness Area and Facing Di
rection")
```



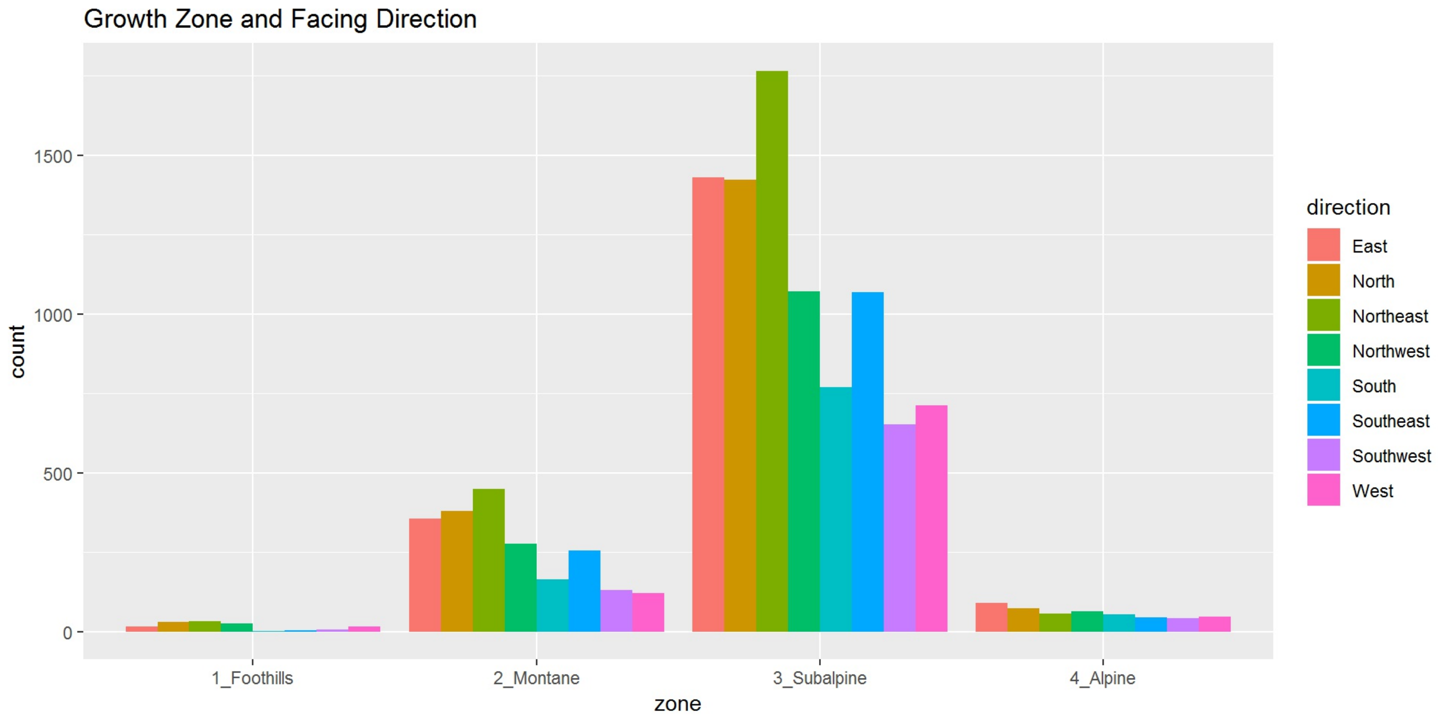

```
ggplot(smlSet) +
  geom_bar(mapping =aes(x =region, fill=direction), position = 'dodge') + ggtitle("Wilderness Area and Facing D
irection")
```



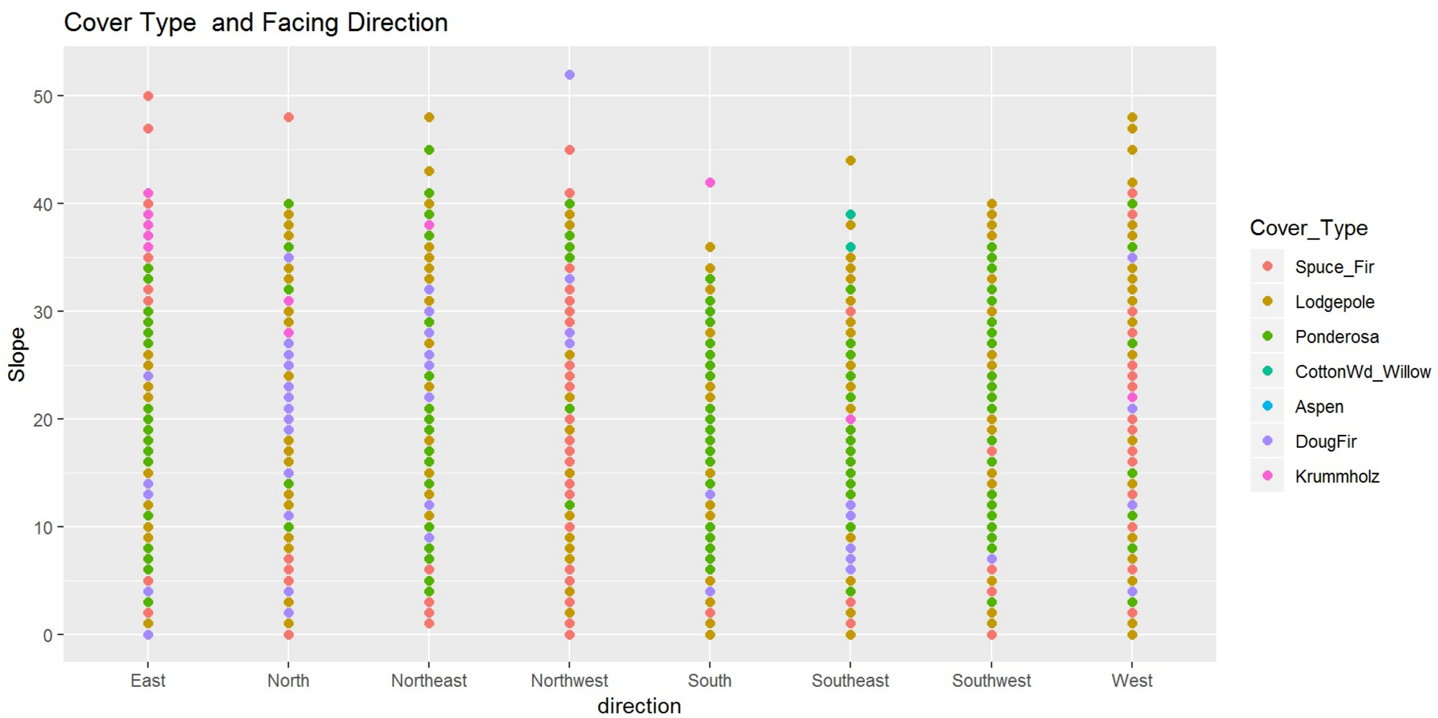

```
#Look at direction and zone
ggplot(smlSet) +
  geom_bar(mapping =aes(x =zone, fill=direction), position = 'fill') + ggtitle("Growth Zone and Facing Directio
n")
```



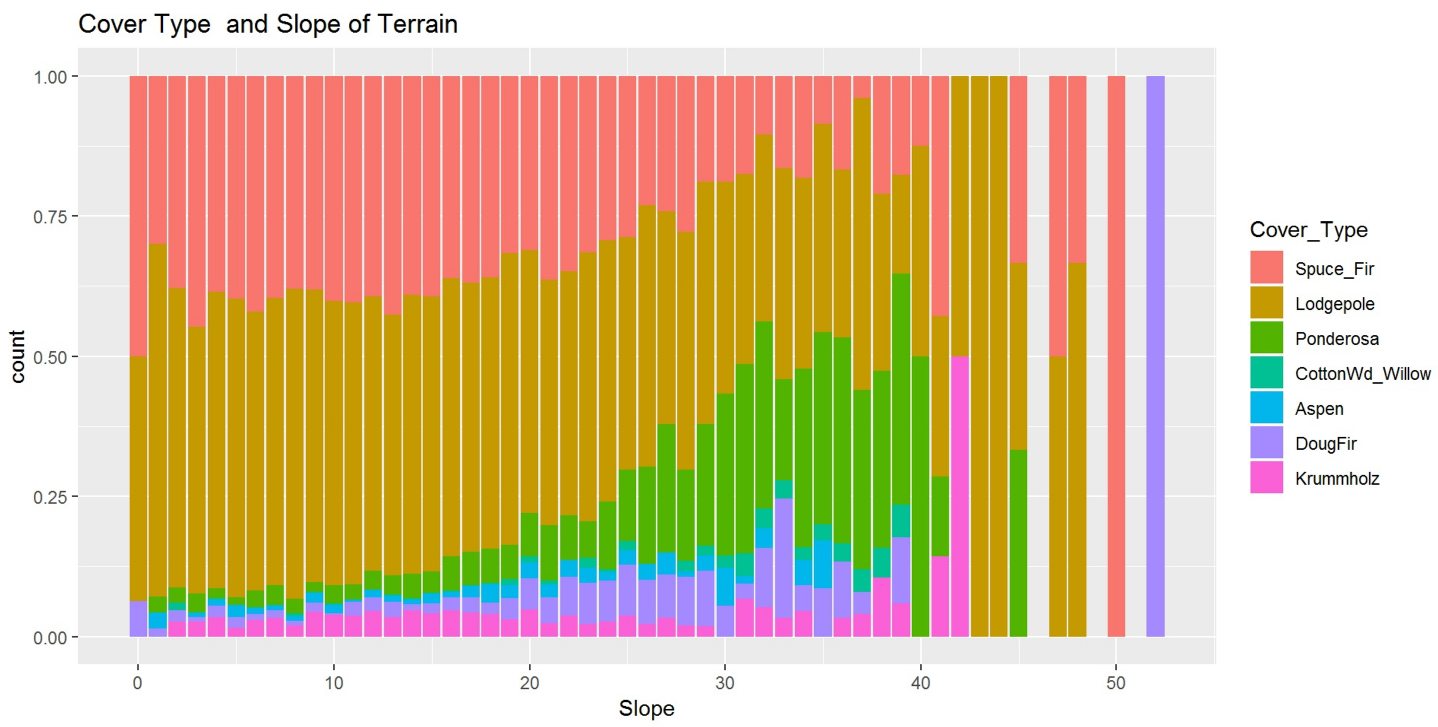

```
ggplot(smlSet) +
  geom_bar(mapping =aes(x =zone, fill=direction), position = 'dodge') + ggtitle("Growth Zone and Facing Directi
on")
```




```
#Slope to direction and Cover_Type
ggplot(smlSet, mapping=aes(x=direction, y=Slope)) +
  geom_point(mapping=aes(color= Cover_Type), pch=19, size=2) + ggtitle("Cover Type  and Facing Direction")
```

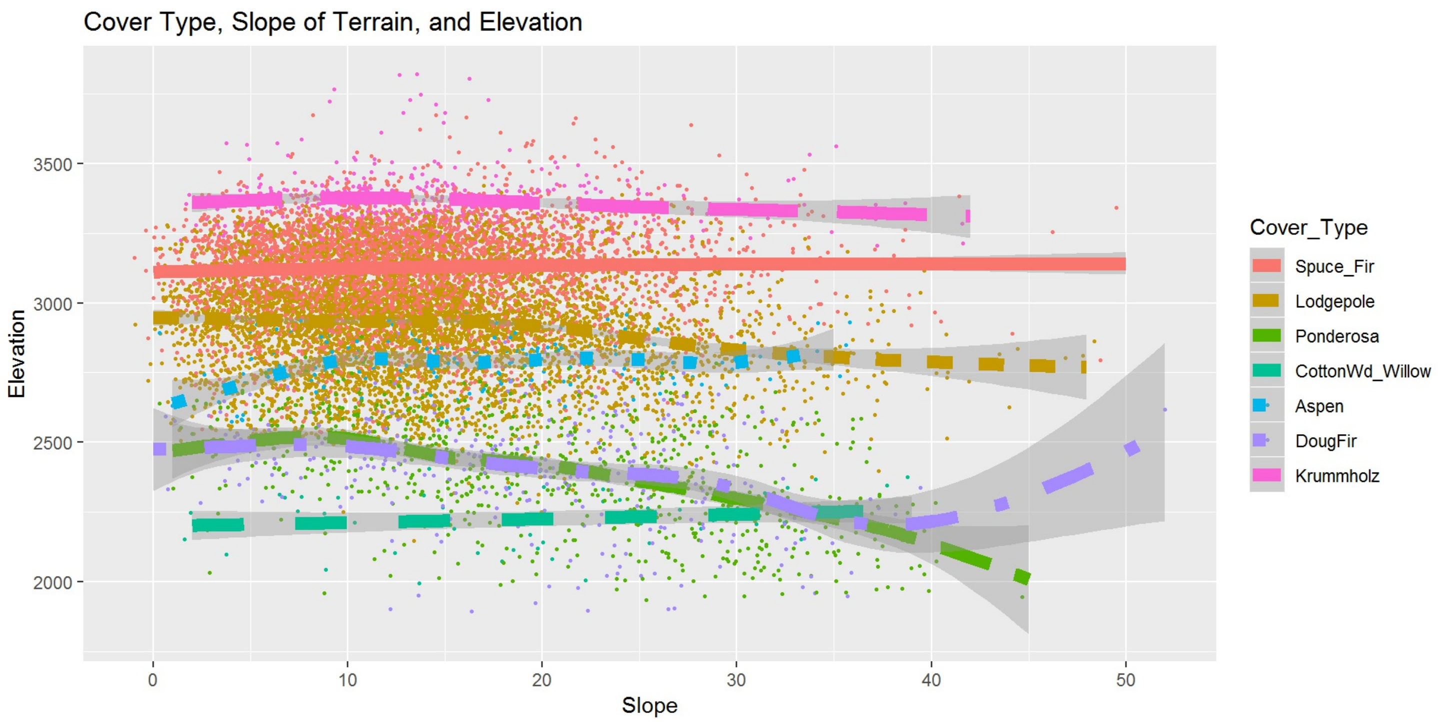



```
#######Slope and Cover_Type Note the four outliers at high slopes
ggplot(smlSet) +
  geom_bar(mapping =aes(x =Slope, fill=Cover_Type), position = 'fill') + ggtitle("Cover Type  and Slope of Terr
ain")
```



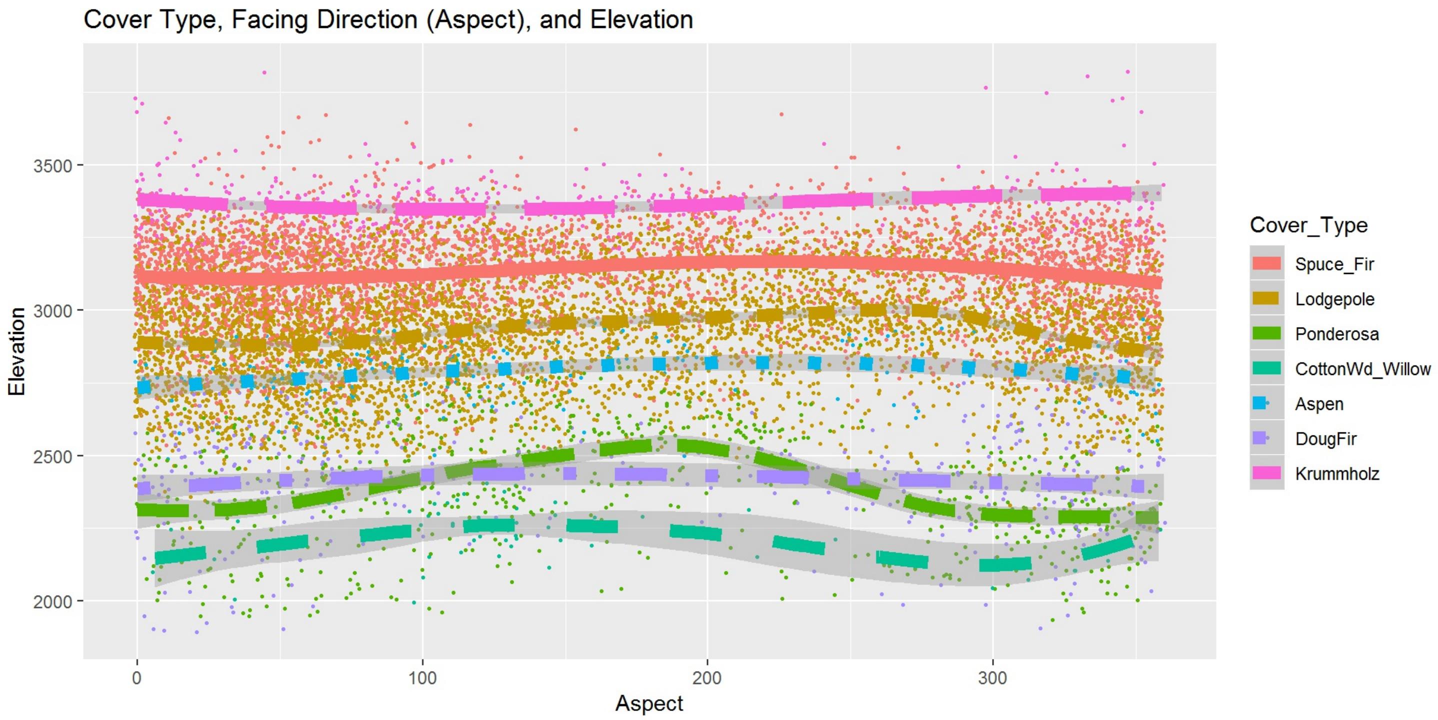

```
##### Slope, Elevation, CoverType, Smoothing Line
ggplot(smlSet, aes(Slope, Elevation)) +
  geom_jitter(mapping= aes(color= Cover_Type), width = 1, height = 0.5, size=.5) +
  geom_smooth(mapping = aes(Slope, Elevation, linetype = Cover_Type, color = Cover_Type), size=3) + ggtitle("Co
ver Type, Slope of Terrain, and Elevation")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



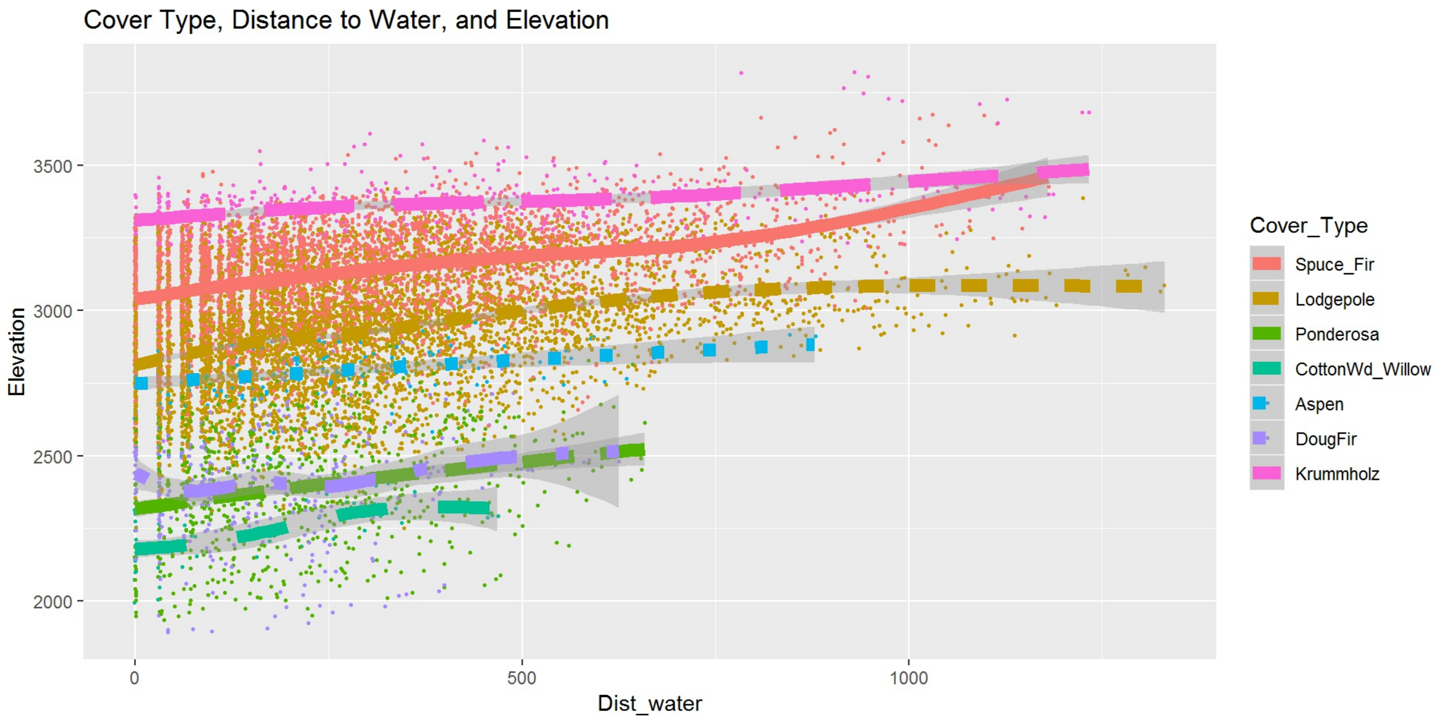

```
##### Aspect, elevation, Cover_type
ggplot(smlSet, aes(Aspect, Elevation)) +
  geom_jitter(mapping= aes(color= Cover_Type), width = 1, height = 0.5, size=.5) +
  geom_smooth(mapping = aes(Aspect, Elevation, linetype = Cover_Type, color = Cover_Type), size=3) + ggtitle("C
over Type, Facing Direction (Aspect), and Elevation")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



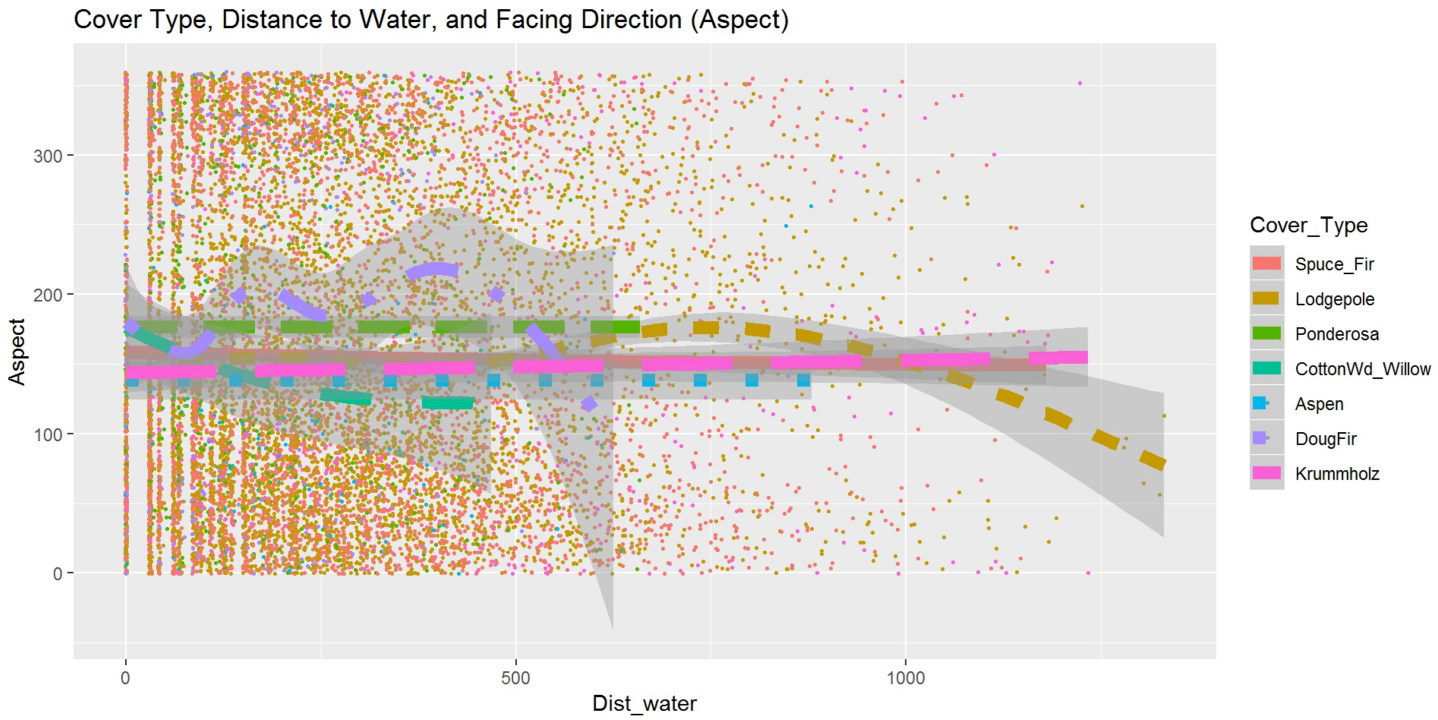

```
####Distance to water, elevation, Cover_type
ggplot(smlSet, aes(Dist_water, Elevation)) +
  geom_jitter(mapping= aes(color= Cover_Type), width = 1, height = 0.5, size=.5) +
  geom_smooth(mapping = aes(Dist_water, Elevation, linetype = Cover_Type, color = Cover_Type), size=3) + ggtitl
e("Cover Type, Distance to Water, and Elevation")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



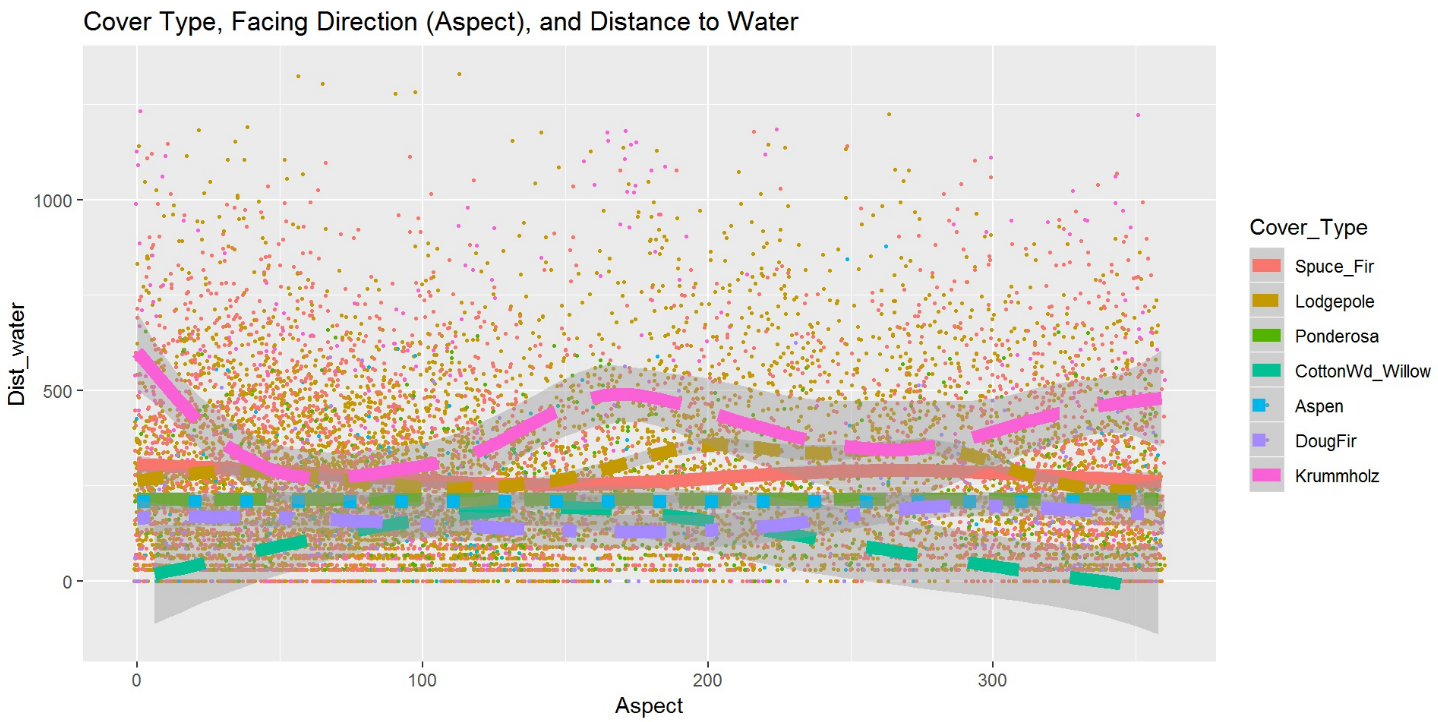

```
####Aspect, distance to water and CoverType
ggplot(smlSet, aes(Dist_water, Aspect)) +
  geom_jitter(mapping= aes(color= Cover_Type), width = 1, height = 0.5, size=.5) +
  geom_smooth(mapping = aes(Dist_water, Aspect, linetype = Cover_Type, color = Cover_Type), size=3) + ggtitle("
Cover Type, Distance to Water, and Facing Direction (Aspect)")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



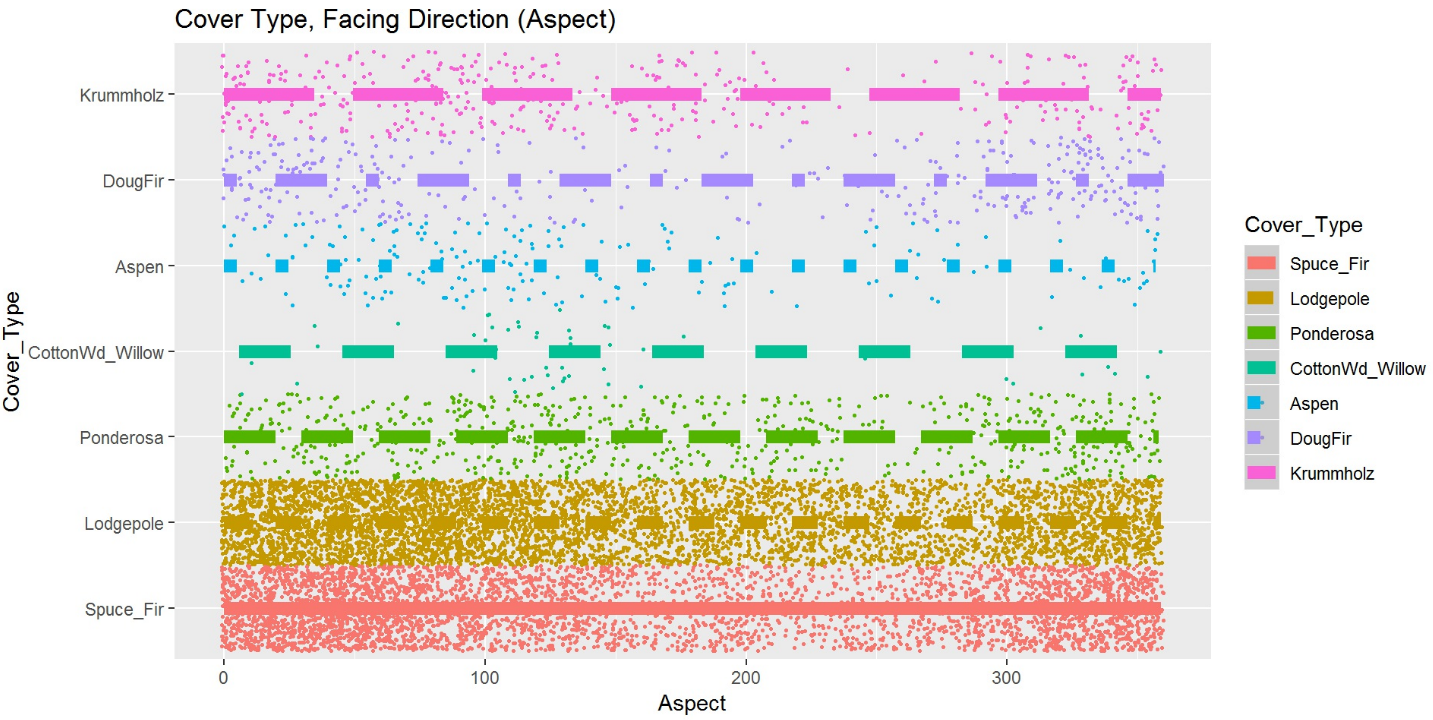

```
####Distance to water, aspect, Cover type
ggplot(smlSet, aes(Aspect, Dist_water)) +
  geom_jitter(mapping= aes(color= Cover_Type), width = 1, height = 0.5, size=.5) +
  geom_smooth(mapping = aes(Aspect, Dist_water, linetype = Cover_Type, color = Cover_Type), size=3) + ggtitle("
Cover Type, Facing Direction (Aspect), and Distance to Water")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

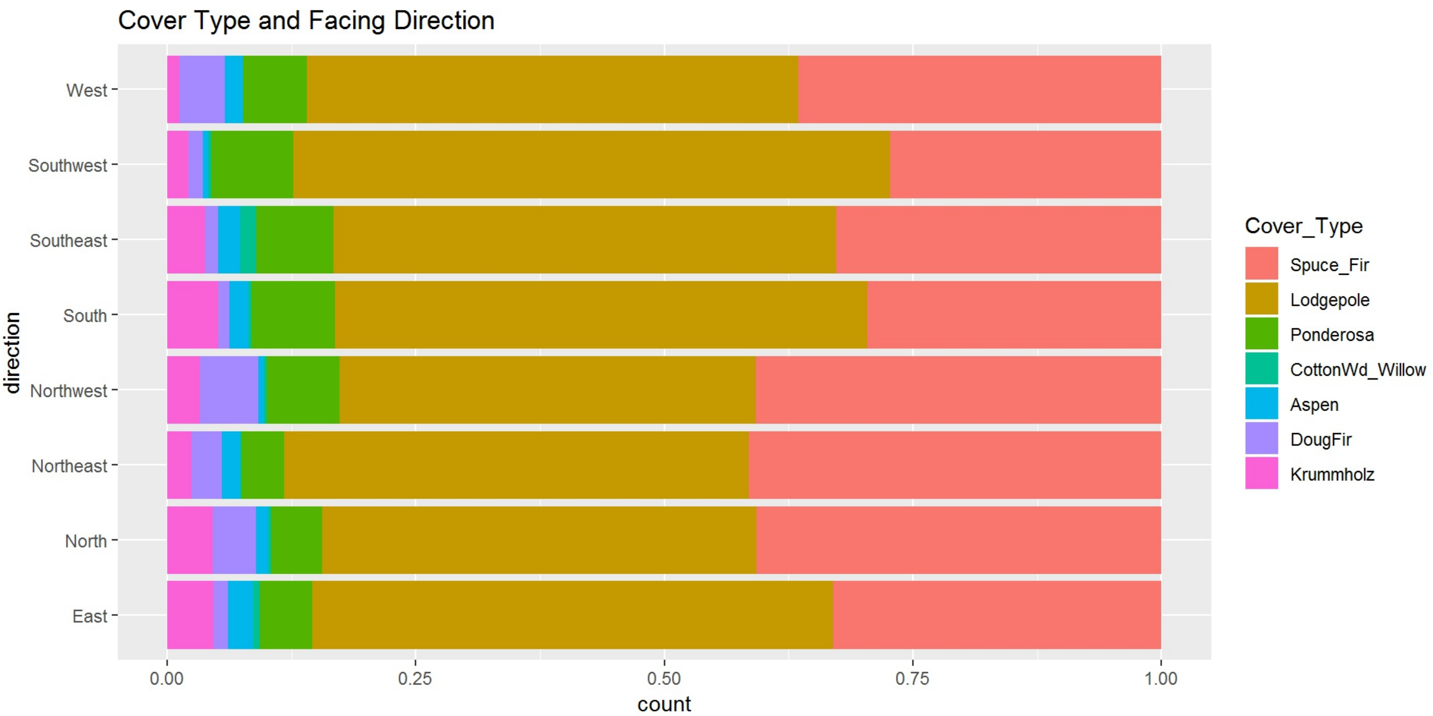



```
####direction and covertype
ggplot(smlSet, aes(Aspect, Cover_Type)) +
  geom_jitter(mapping= aes(color= Cover_Type), width = 1, height = 0.5, size=.5) +
  geom_smooth(mapping = aes(Aspect, Cover_Type, linetype = Cover_Type, color = Cover_Type), size=3)  + ggtitle(
"Cover Type, Facing Direction (Aspect)")
```

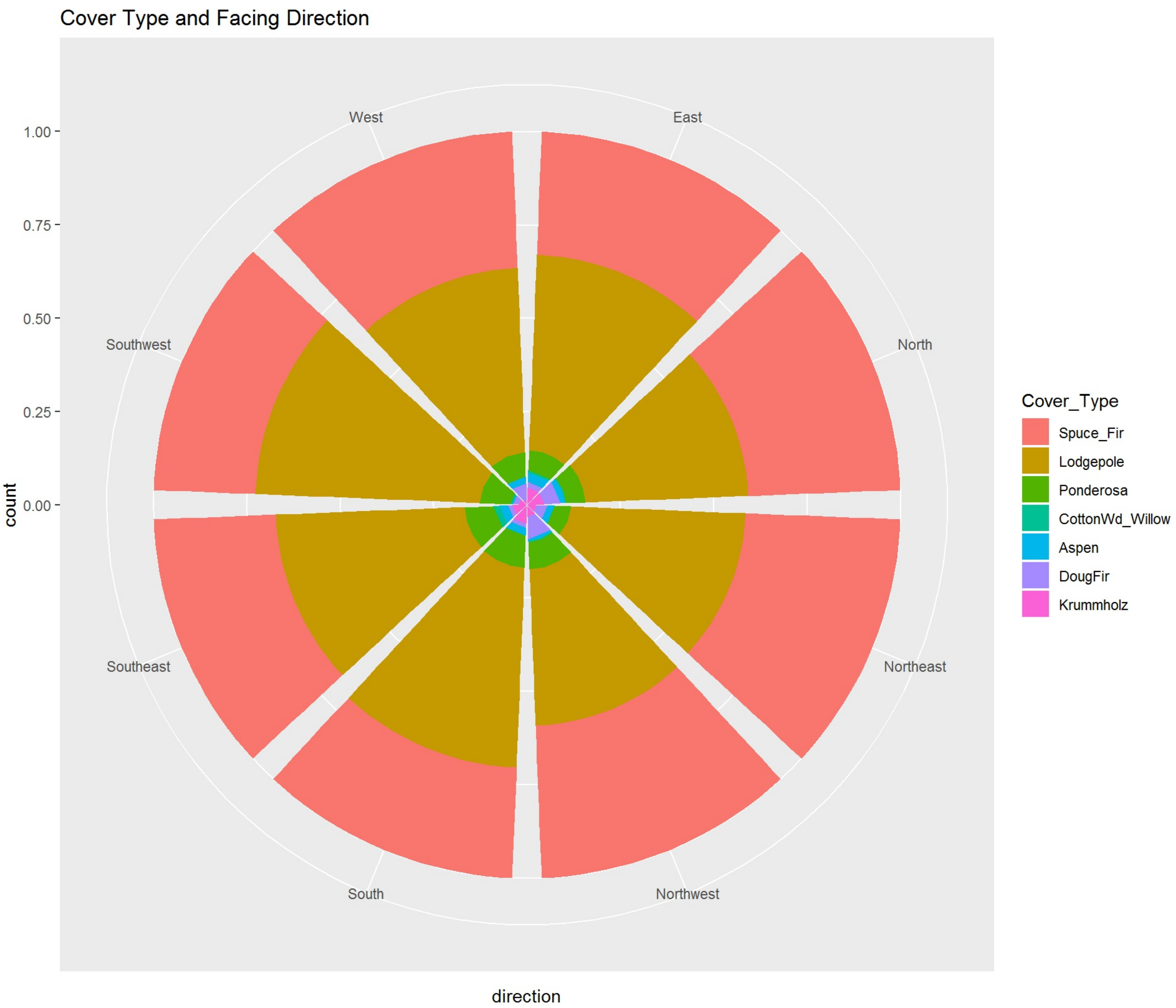
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




```
#direction and CoverType Polar coordinates
bar <- ggplot(data=smlSet) +
  geom_bar(mapping = aes(x =direction, fill = Cover_Type), position = 'fill')
bar + coord_flip() + ggtitle("Cover Type and Facing Direction")
```

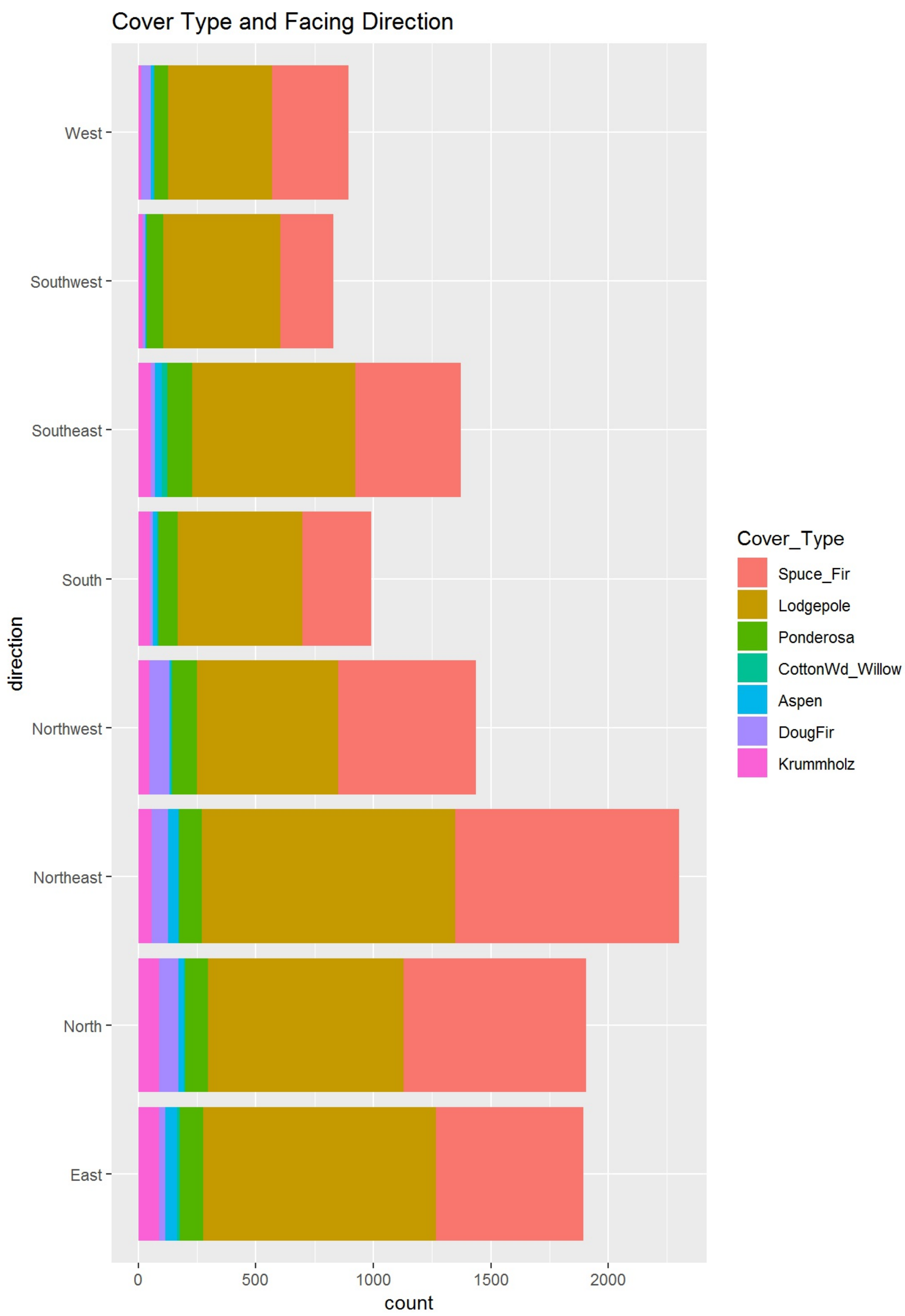



```
bar + coord_polar() + ggtitle("Cover Type and Facing Direction")
```

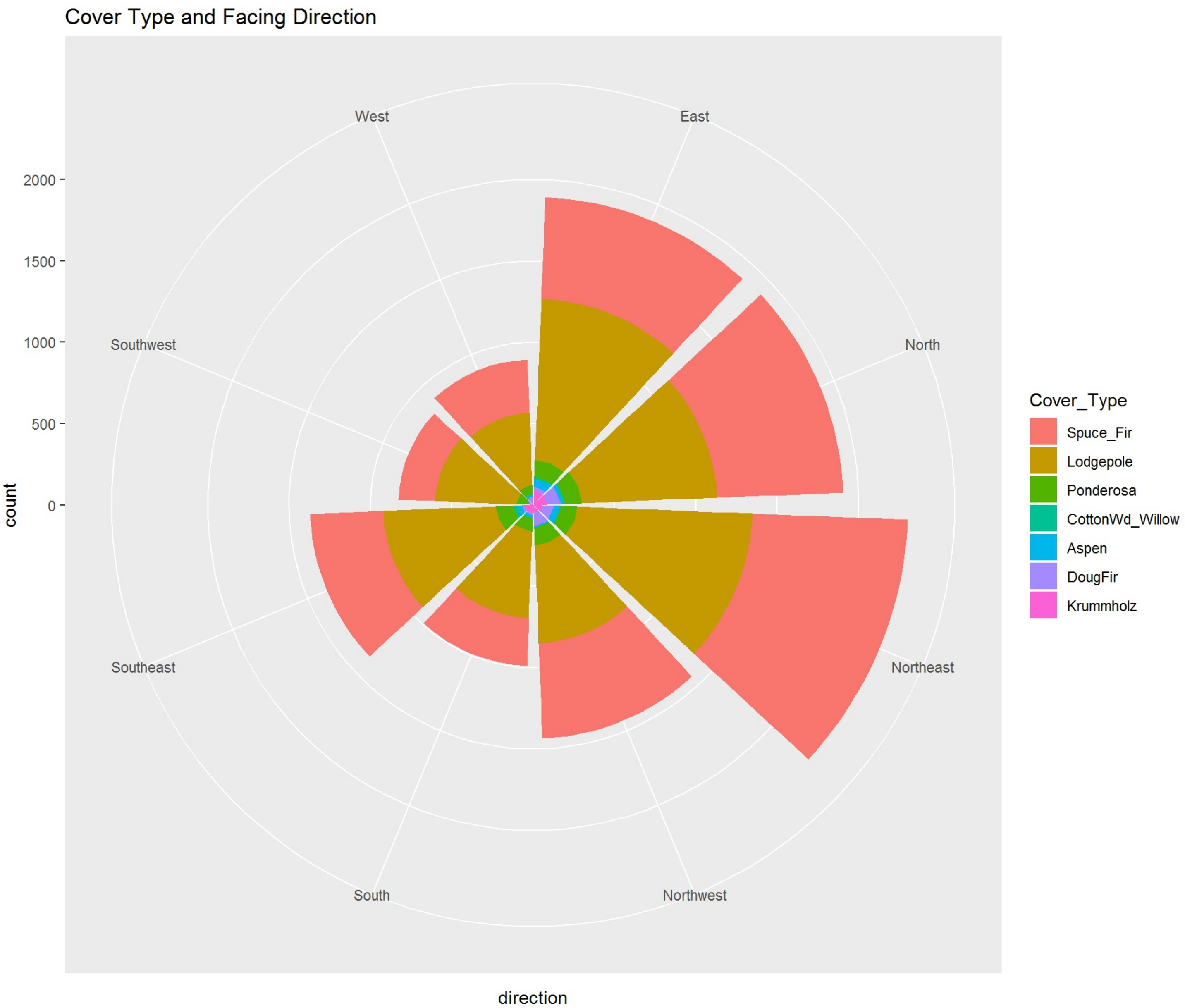



```
#Direction and Cover Type Bar and Polar Coord Plot
bar <- ggplot(data=smlSet) +
  geom_bar(mapping = aes(x =direction, fill = Cover_Type))
```

```
bar + coord_flip() + ggtitle("Cover Type and Facing Direction")
```

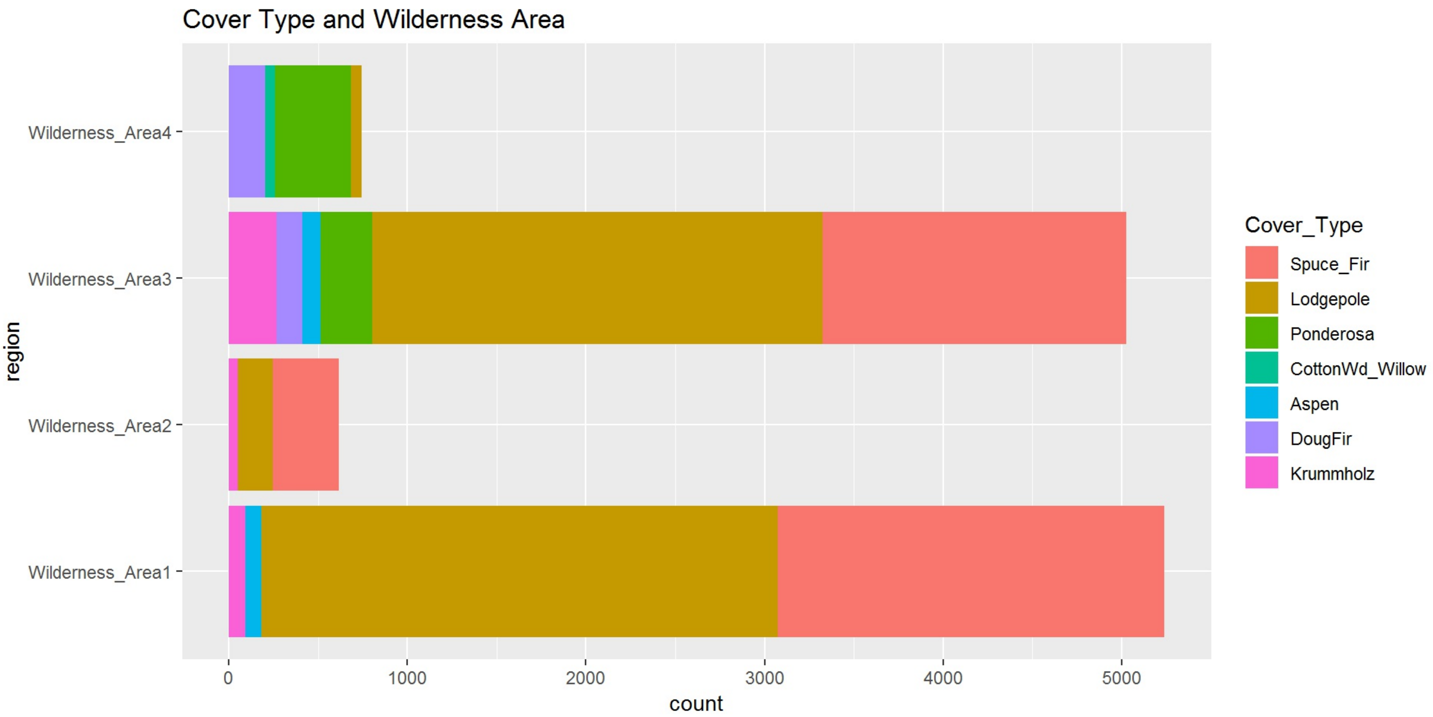



```
bar + coord_polar() + ggtitle("Cover Type and Facing Direction")
```

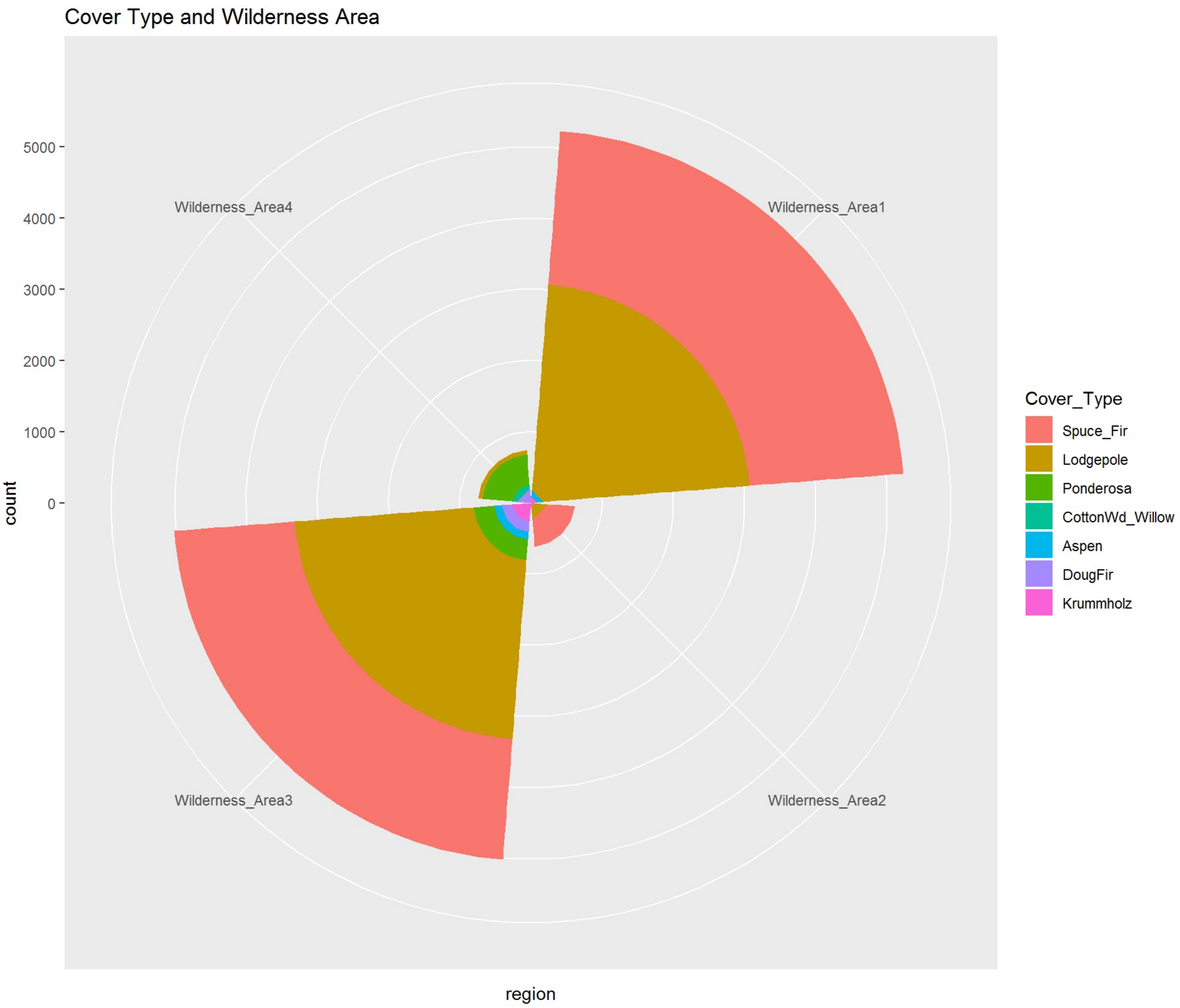



```
##Wilderness areas and Cover_Type bar and polar coord plot
bar <- ggplot(data=smlSet) +
  geom_bar(mapping = aes(x =region, fill = Cover_Type))
```

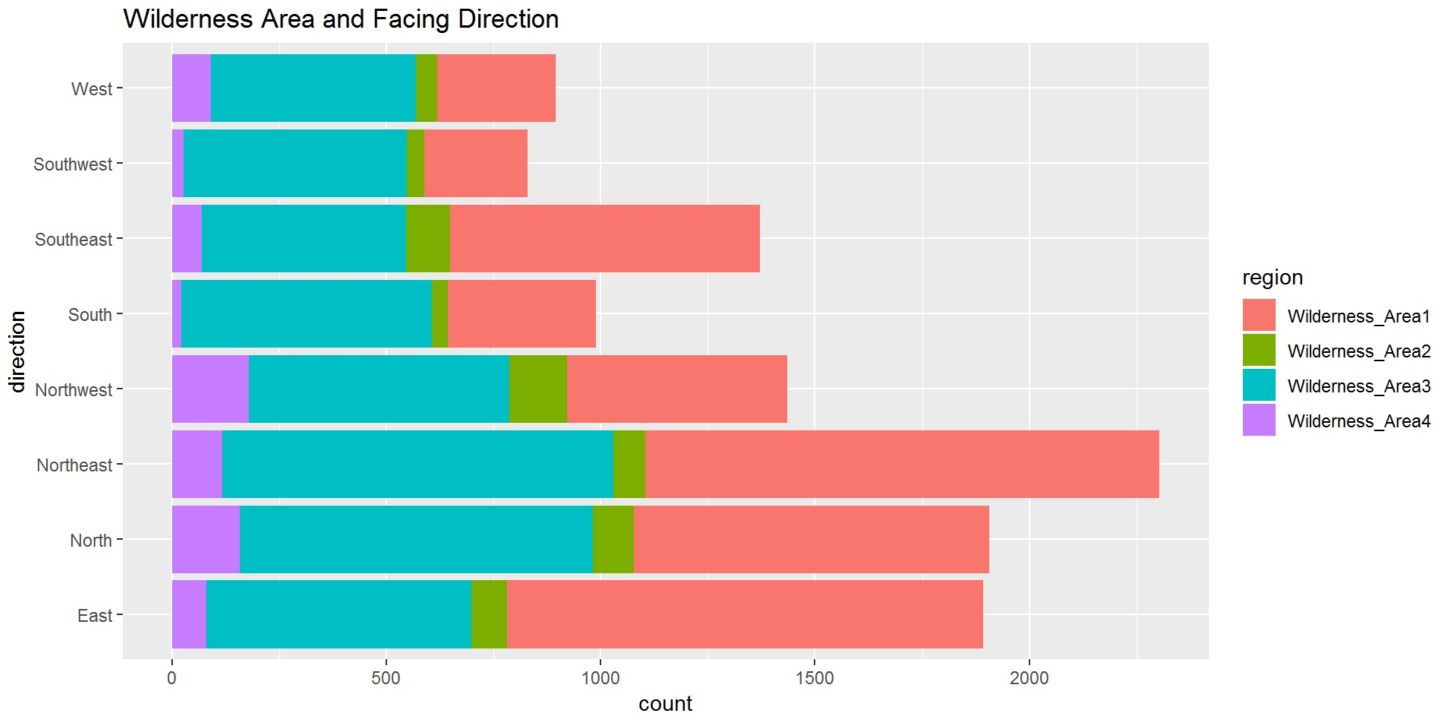
```
bar + coord_flip() + ggtitle("Cover Type and Wilderness Area")
```




```
bar + coord_polar() + ggtitle("Cover Type and Wilderness Area")
```

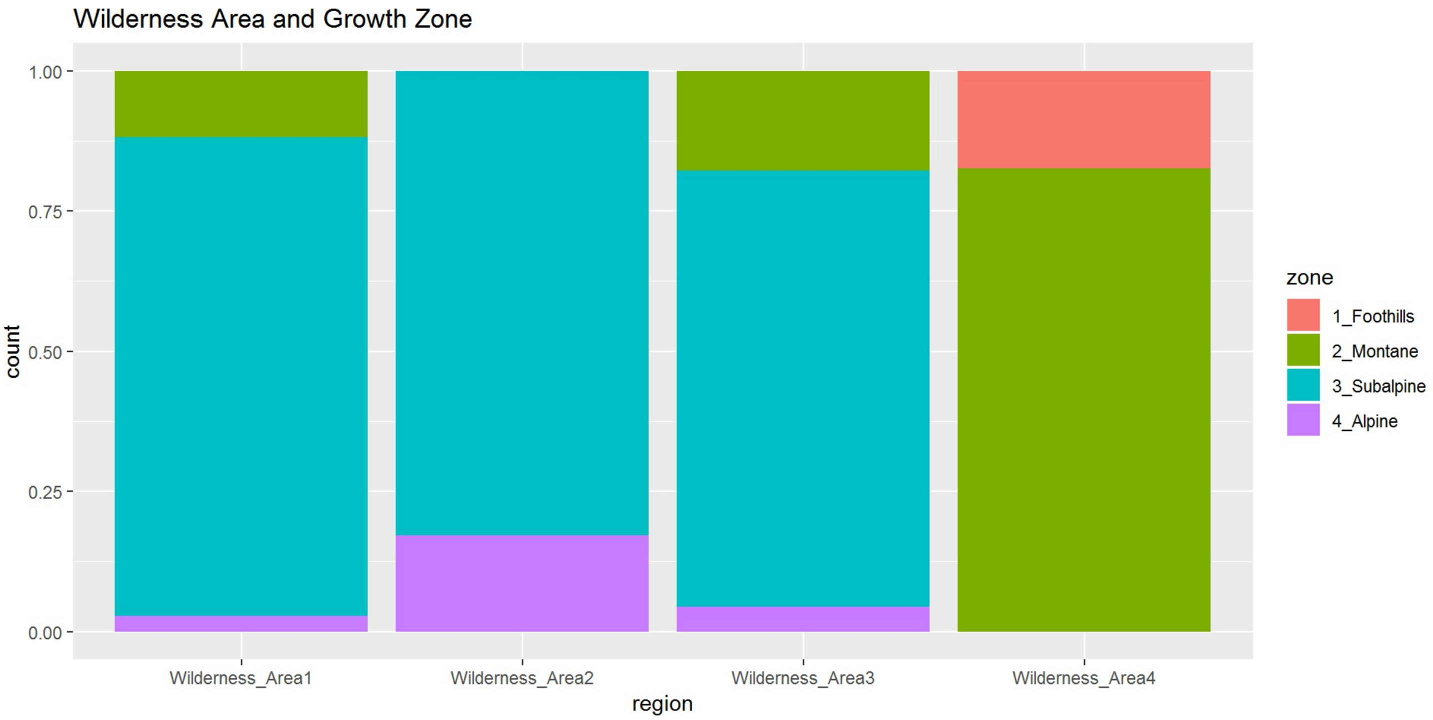



```
#Region and direction
bar <- ggplot(data=smlSet) +
  geom_bar(mapping = aes(x =direction, fill = region))
bar + coord_flip() + ggtitle("Wilderness Area and Facing Direction")
```

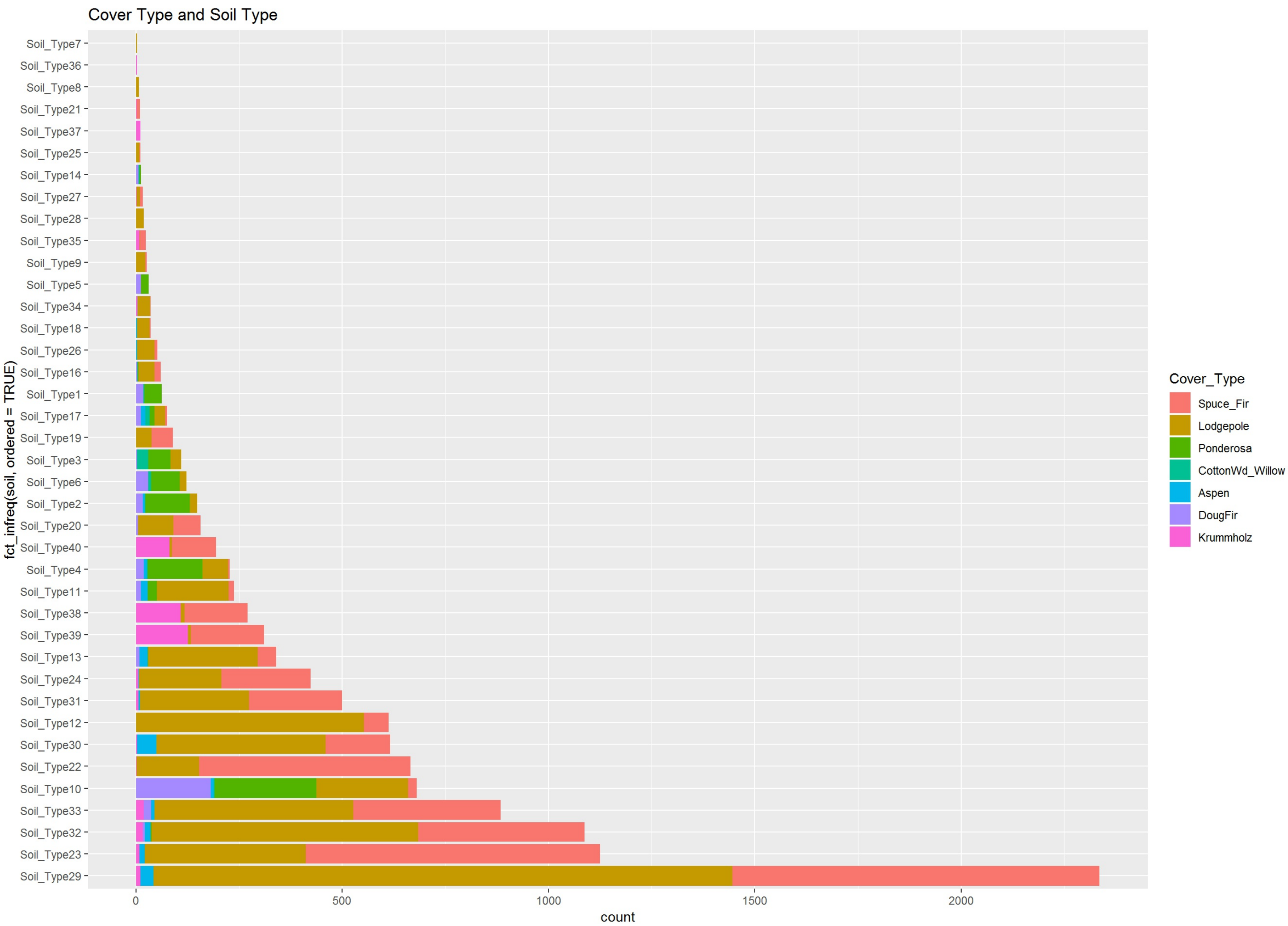



```
bar + coord_polar() + ggtitle("Wilderness Area and Facing Direction")
```

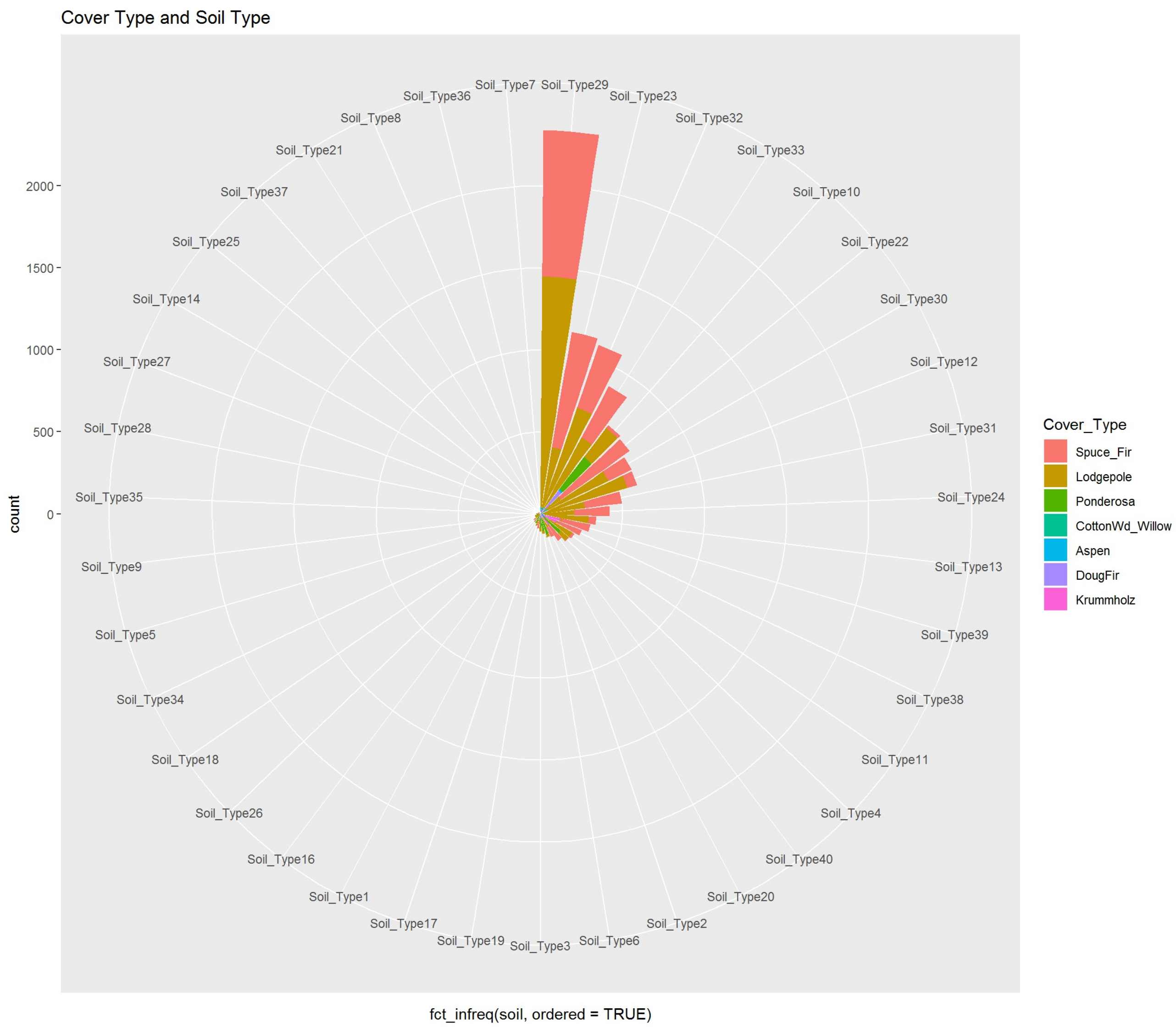

```
#Region and zone
ggplot(smlSet) +
  geom_bar(mapping =aes(x =region, fill=zone), position = 'fill') + ggtitle("Wilderness Area and Growth Zone")
```



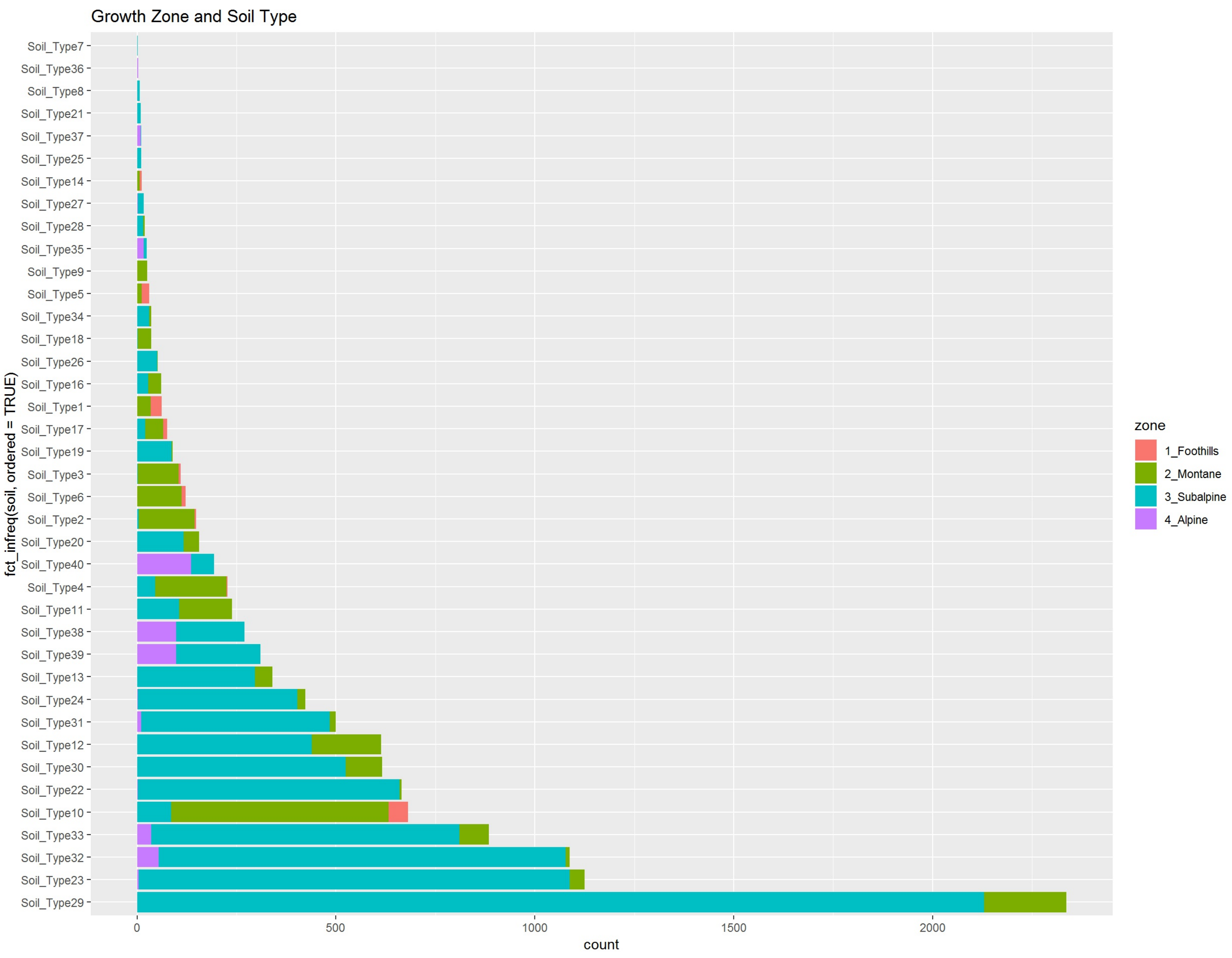

```
#Soil Type by Cover tpye
# library(forcats)
library(forcats)
bar <- ggplot(smlSet, aes(fct_infreq(soil,ordered = TRUE), fill=Cover_Type)) +
  geom_bar()
bar + coord_flip() + ggtitle("Cover Type and Soil Type")
```



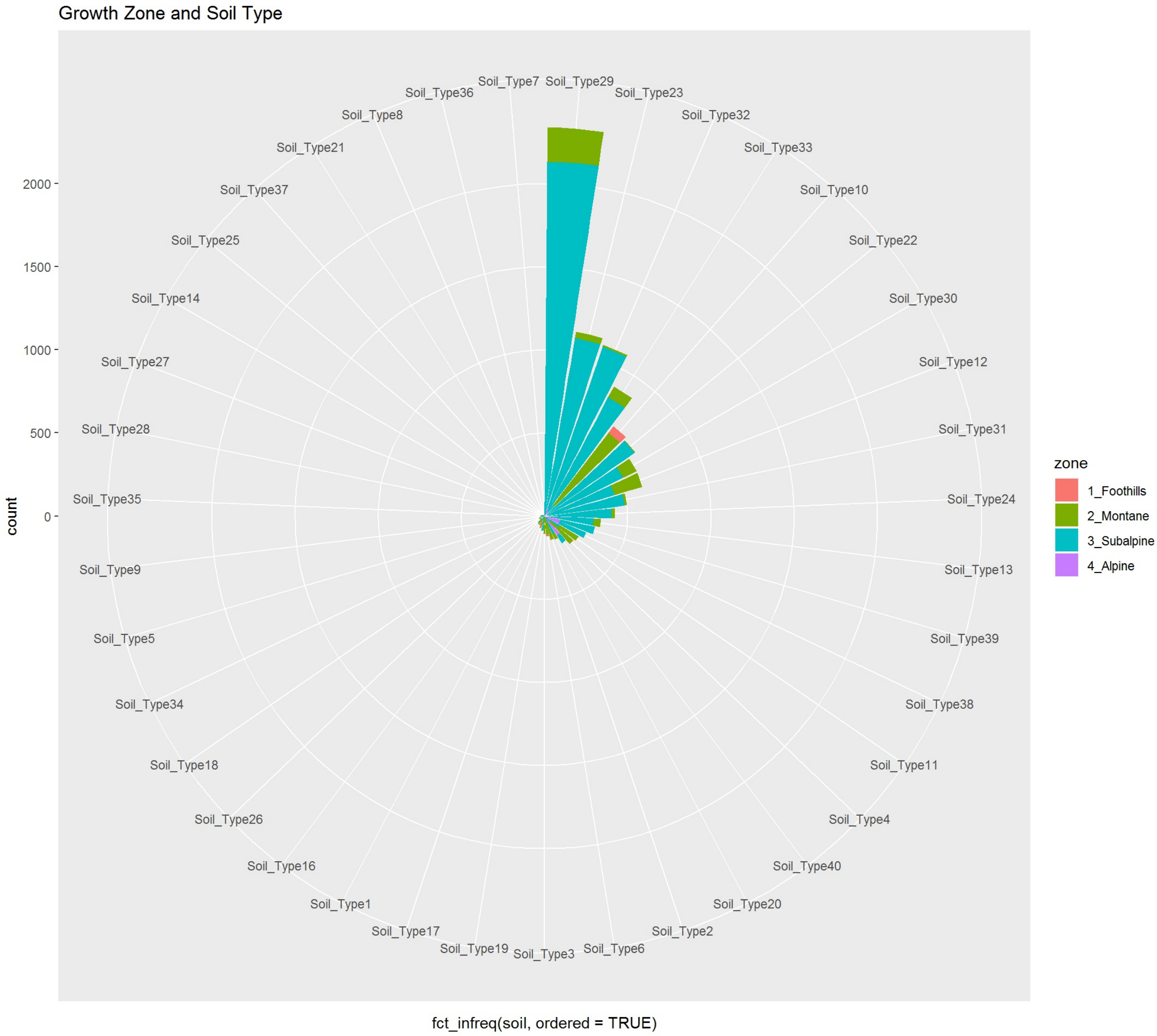

```
bar + coord_polar() + ggtitle("Cover Type and Soil Type")
```



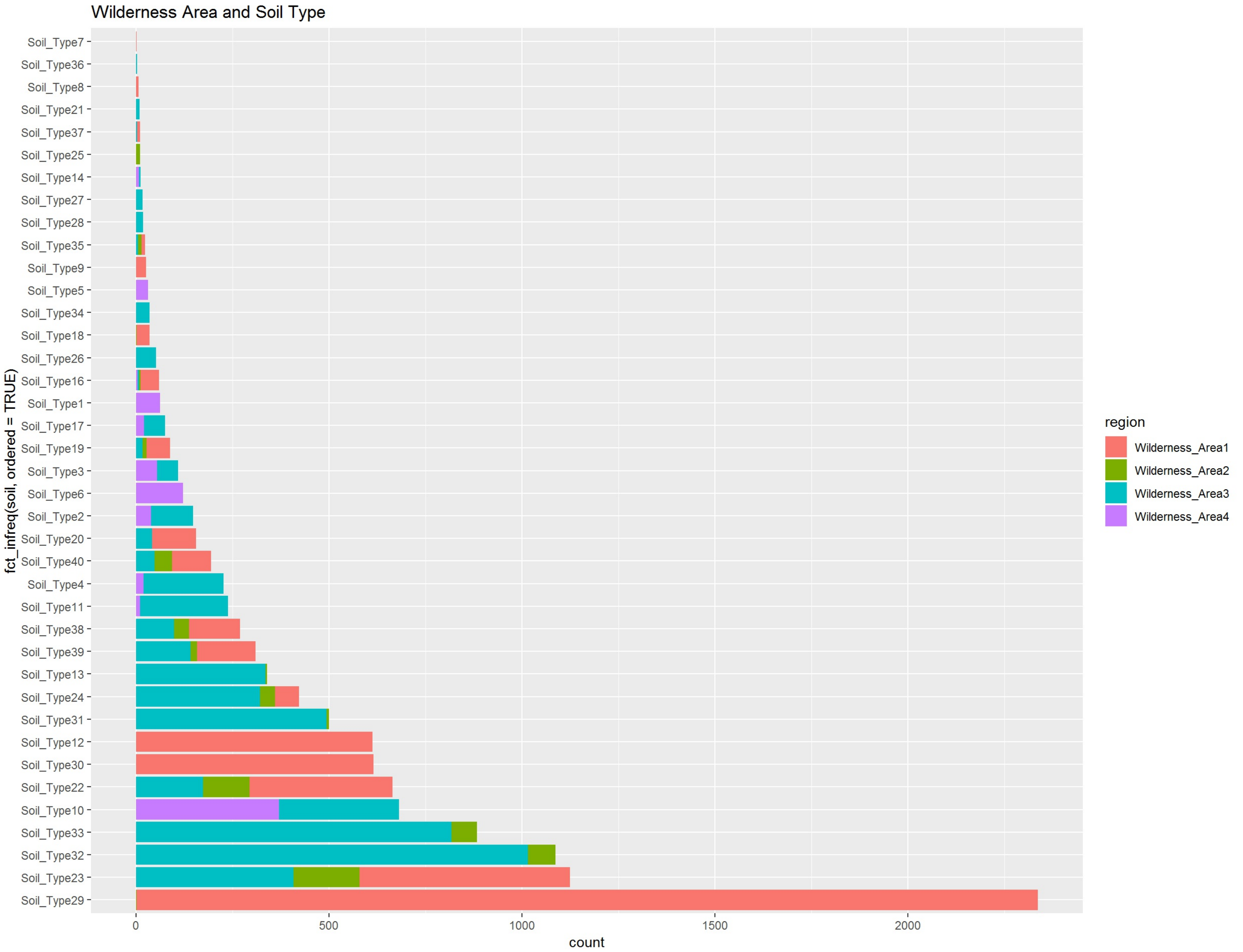

```
#Soil Type to Zone
bar <- ggplot(smlSet, aes(fct_infreq(soil,ordered = TRUE), fill=zone)) +
  geom_bar()
bar + coord_flip() + ggtitle("Growth Zone and Soil Type")
```



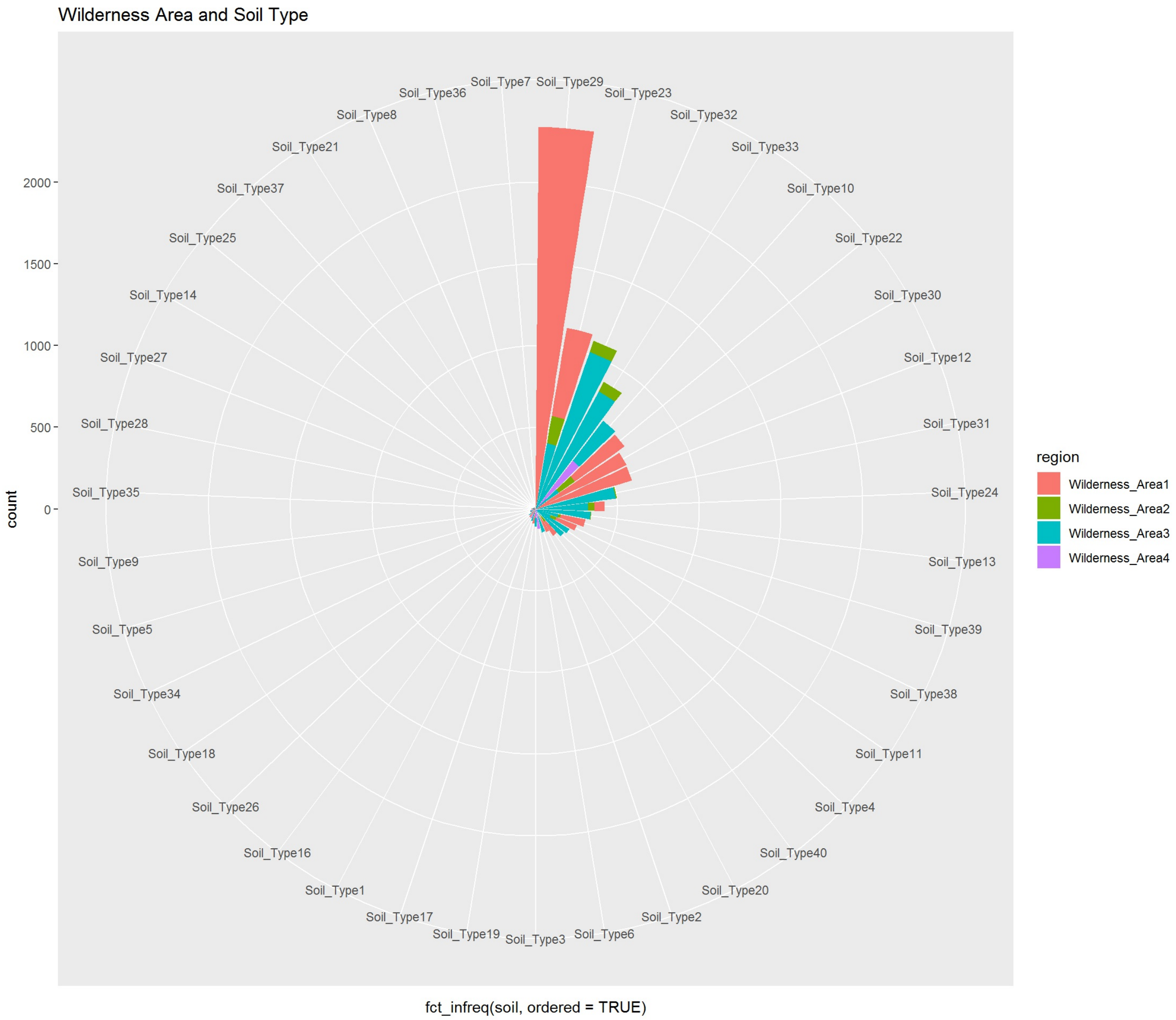

```
bar + coord_polar() + ggtitle("Growth Zone and Soil Type")
```



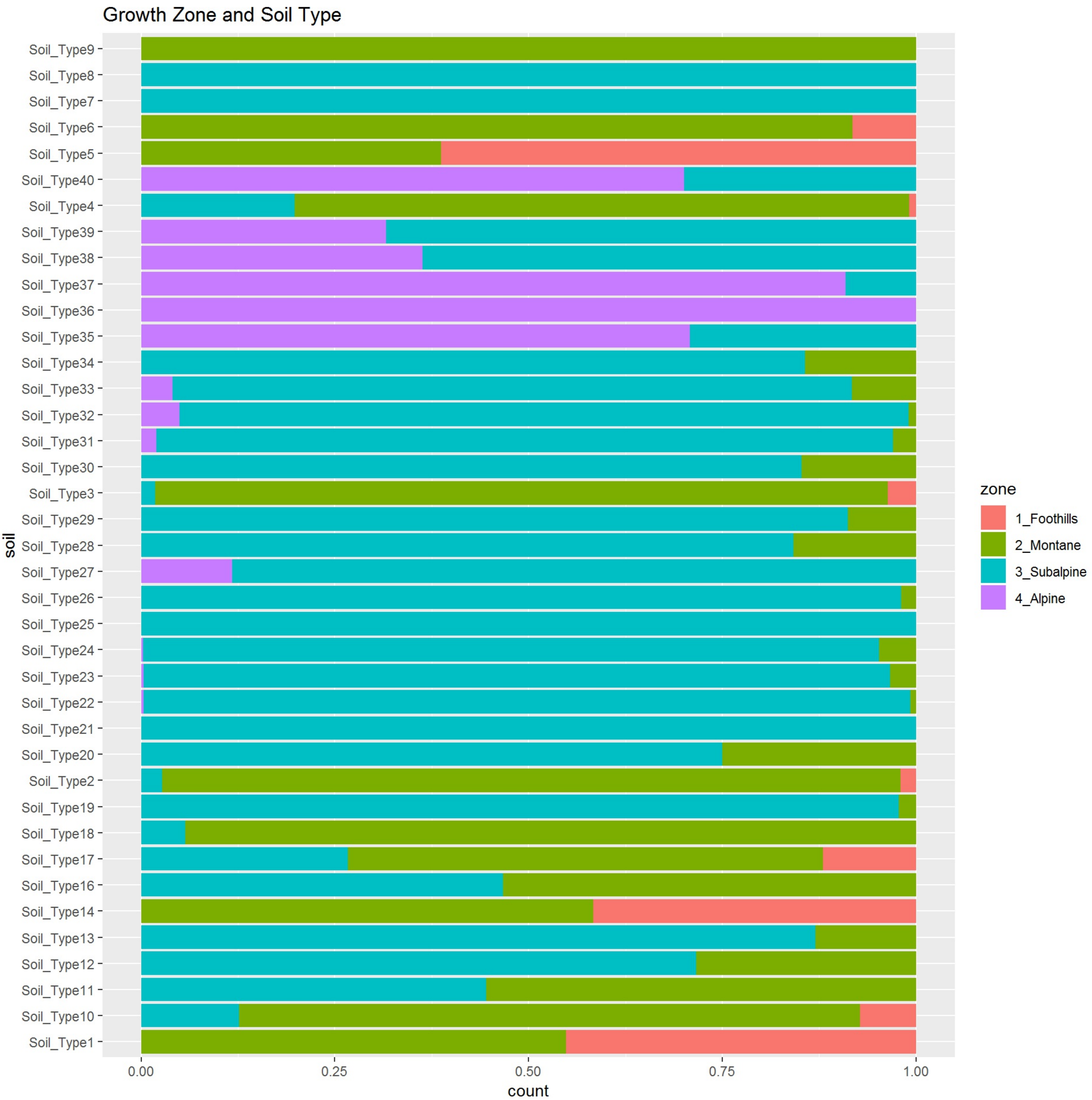

```
#Soil Type by Wilderness Area
bar <- ggplot(smlSet, aes(fct_infreq(soil,ordered = TRUE), fill=region)) +
  geom_bar()
bar + coord_flip() + ggtitle("Wilderness Area and Soil Type")
```



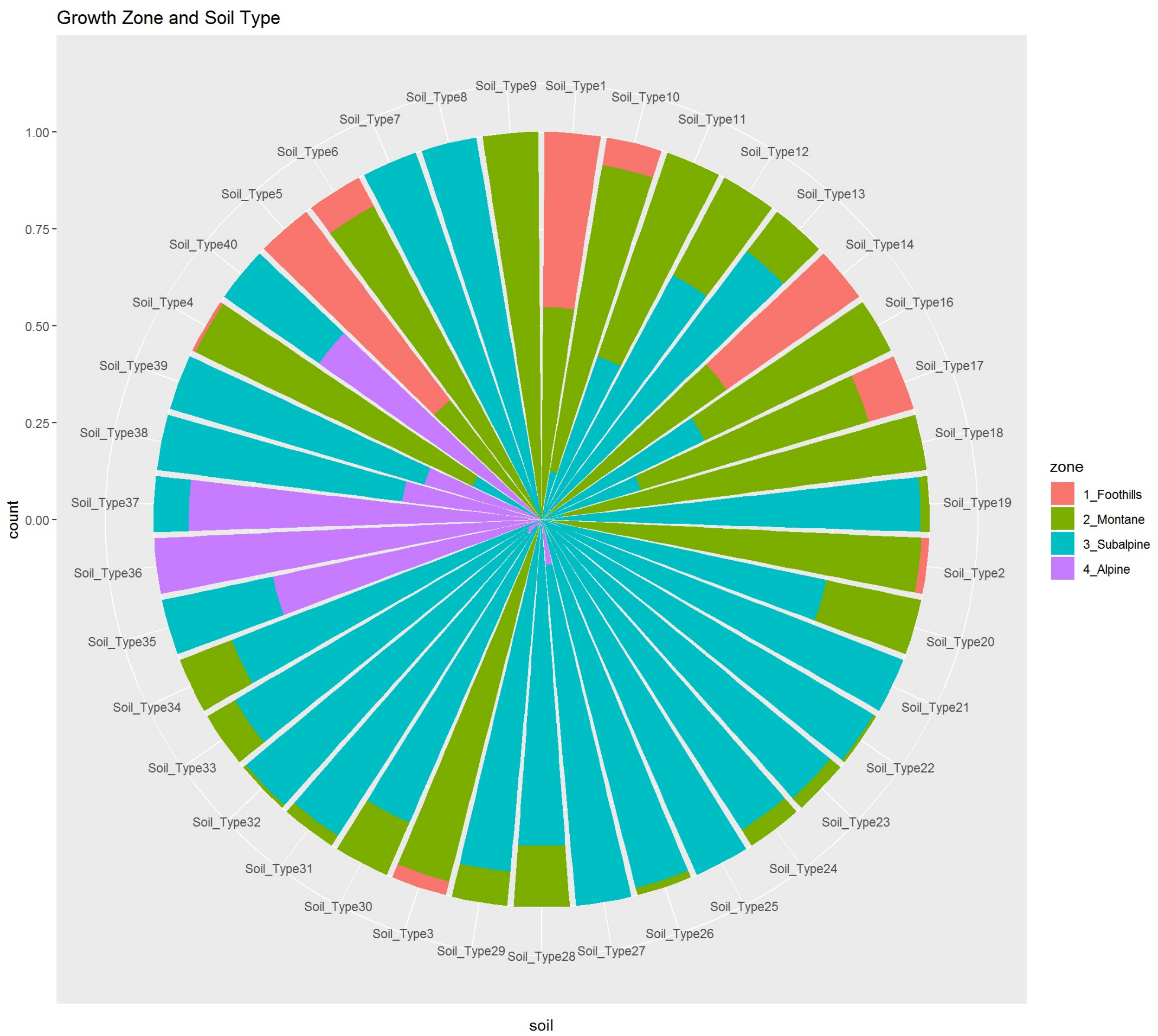

```
bar + coord_polar() + ggtitle("Wilderness Area and Soil Type")
```




```
#Soil Types by wilderness area (better to see differences)
bar <- ggplot(smlSet) +
  geom_bar(mapping =aes(x =soil, fill=zone), position = 'fill')
bar + coord_flip()  + ggtitle("Growth Zone and Soil Type")
```




```
bar + coord_polar()  + ggtitle("Growth Zone and Soil Type")
```



Growth Zone and Soil Type
count
1.00
0.75
0.50
0.25
0.00
Soil_Type9
Soil_Type1
Soil_Type10
Soil_Type11
Soil_Type12
Soil_Type13
Soil_Type14
Soil_Type16
Soil_Type17
Soil_Type18
Soil_Type19
Soil_Type2
Soil_Type20
Soil_Type21
Soil_Type22
Soil_Type23
Soil_Type24
Soil_Type25
Soil_Type26
Soil_Type27
Soil_Type28
Soil_Type29
Soil_Type3
Soil_Type30
Soil_Type31
Soil_Type32
Soil_Type33
Soil_Type34
Soil_Type35
Soil_Type36
Soil_Type37
Soil_Type38
Soil_Type39
Soil_Type4
Soil_Type40
Soil_Type5
Soil_Type6
Soil_Type7
Soil_Type8
soil
zone
1_Foothills
2_Montane
3_Subalpine
4_Alpine

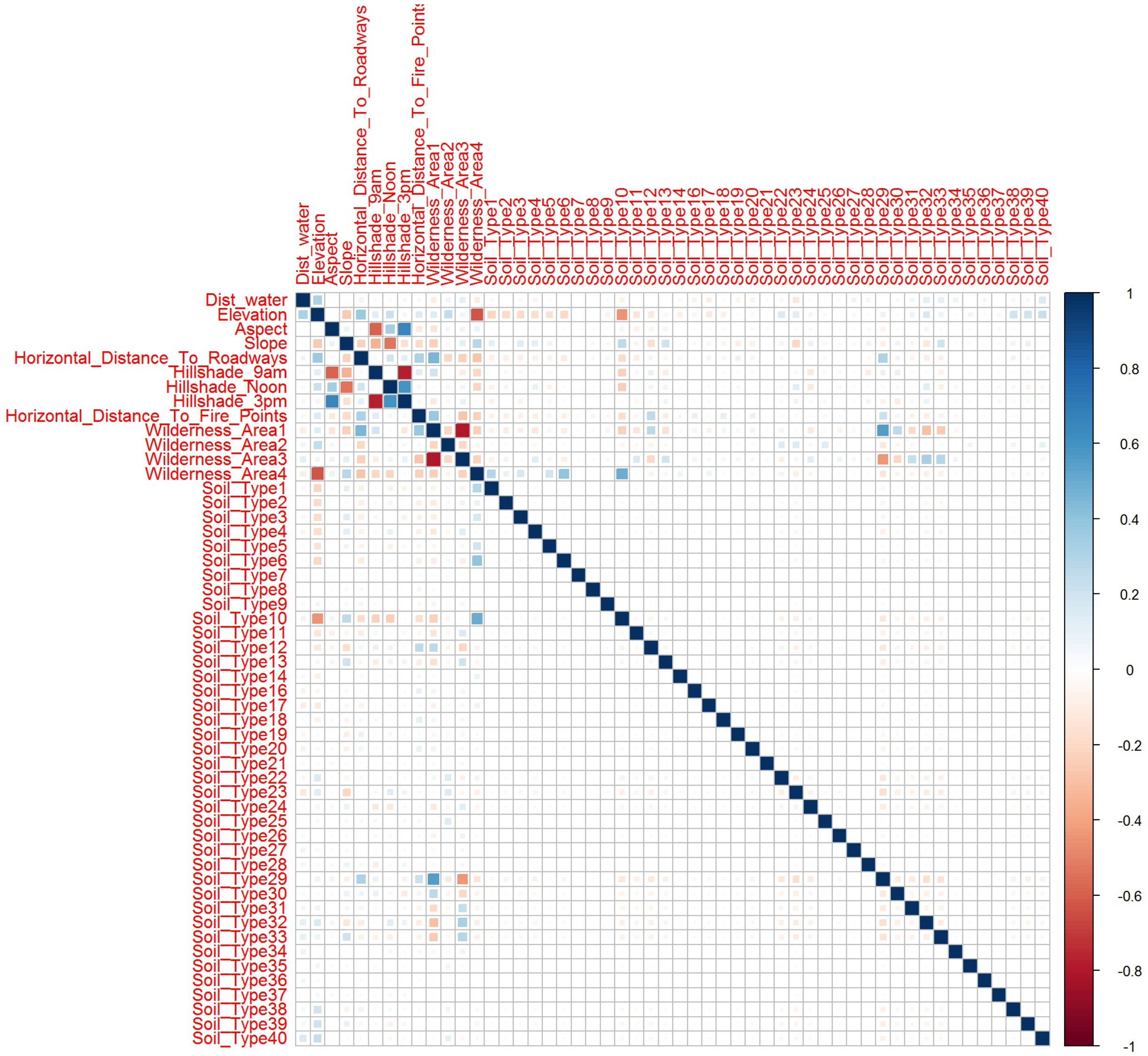
```
#Consolidate numeric variables
sampleSet <- select(smlSet,
                  Cover_Type,
                  Dist_water,
                  Elevation,
                  Aspect,
                  Slope,
                  Horizontal_Distance_To_Roadways,
                  Hillshade_9am,
                  Hillshade_Noon,
                  Hillshade_3pm,
                  Horizontal_Distance_To_Fire_Points,
                  Wilderness_Area1,
                  Wilderness_Area2,
                  Wilderness_Area3,
                  Wilderness_Area4,
                  Soil_Type1,
                  Soil_Type2,
                  Soil_Type3,
                  Soil_Type4,
                  Soil_Type5,
                  Soil_Type6,
                  Soil_Type7,
                  Soil_Type8,
                  Soil_Type9,
                  Soil_Type10,
                  Soil_Type11,
                  Soil_Type12,
                  Soil_Type13,
                  Soil_Type14,
                  Soil_Type16,
                  Soil_Type17,
                  Soil_Type18,
                  Soil_Type19,
                  Soil_Type20,
                  Soil_Type21,
                  Soil_Type22,
                  Soil_Type23,
                  Soil_Type24,
                  Soil_Type25,
                  Soil_Type26,
                  Soil_Type27,
                  Soil_Type28,
                  Soil_Type29,
                  Soil_Type30,
                  Soil_Type31,
                  Soil_Type32,
                  Soil_Type33,
                  Soil_Type34,
                  Soil_Type35,
                  Soil_Type36,
                  Soil_Type37,
                  Soil_Type38,
                  Soil_Type39,
                  Soil_Type40
)
#Create Correlation matrix less the dependent variable
library(corrplot)
```

```
## corrplot 0.84 loaded
```

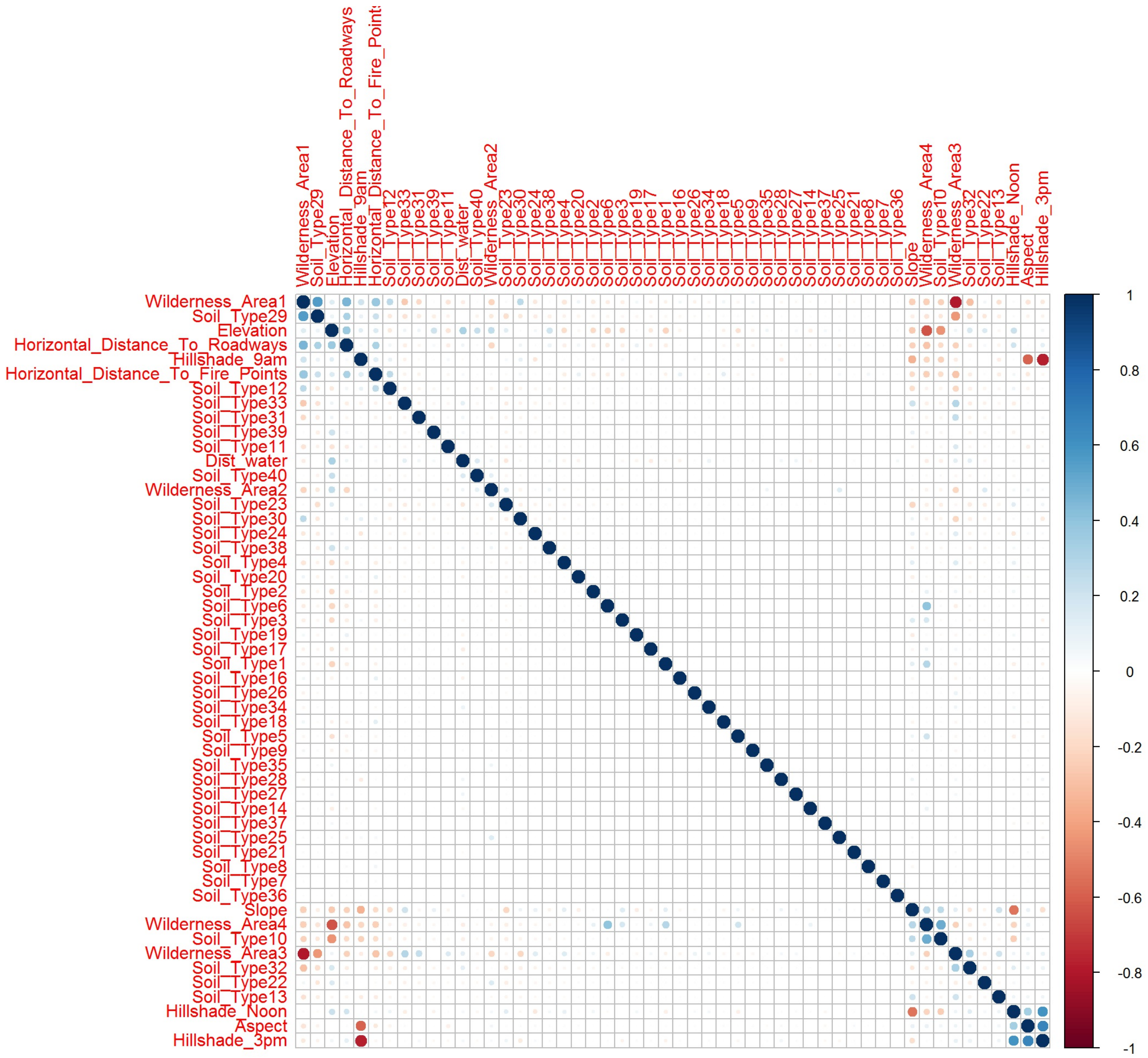
```
cor1 <- cor(sampleSet[-1]) #Correlation Lib
dim(cor1)
```

```
## [1] 52 52
```

```
#Create Correlation plot for the variables
corrplot::corrplot(cor1, method = "square") #Dont load PLS before Corrplot
```




```
#Clusters Correlations
corrplot::corrplot(cor1, order = "hclust")
```



```
#Zero and Near-Zero Variables
nzv <- nearZeroVar(sampleSet, saveMetrics= TRUE) #caret Lib
nzv[nzv$nzv,][1:10,]
```

	freqRatio <dbl>	percentUnique <dbl>	zeroVar <lgl>	nzv <lgl>
Soil_Type1	186.48387	0.01720578	FALSE	TRUE
Soil_Type2	77.54054	0.01720578	FALSE	TRUE
Soil_Type3	105.64220	0.01720578	FALSE	TRUE
Soil_Type4	50.20705	0.01720578	FALSE	TRUE
Soil_Type5	373.96774	0.01720578	FALSE	TRUE
Soil_Type6	94.27869	0.01720578	FALSE	TRUE
Soil_Type7	5811.00000	0.01720578	FALSE	TRUE
Soil_Type8	1659.57143	0.01720578	FALSE	TRUE
Soil_Type9	446.07692	0.01720578	FALSE	TRUE
Soil_Type11	47.84034	0.01720578	FALSE	TRUE

1-10 of 10 rows

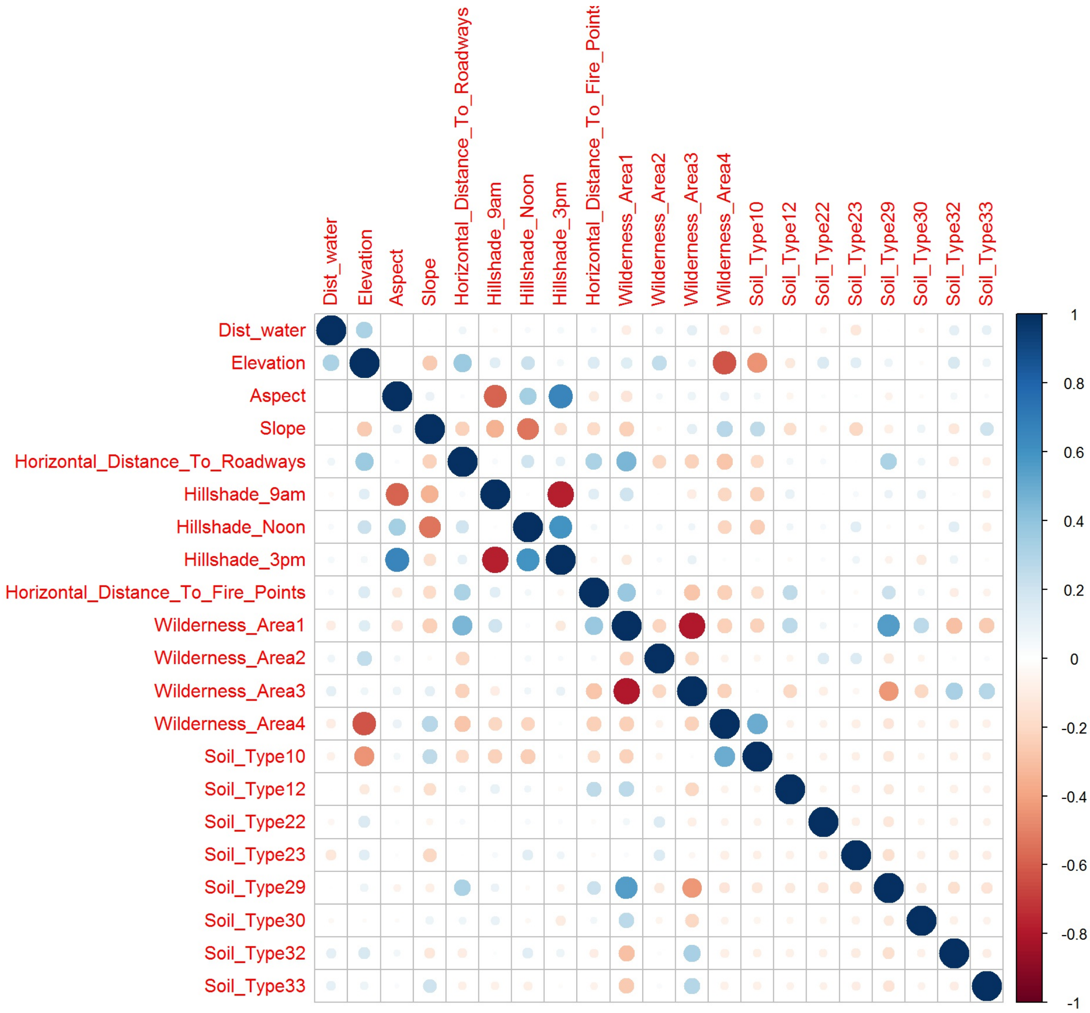
```
dim(sampleSet)
```

```
## [1] 11624    53
```

```
#Shows how many variables are close to Zero Variance
nzv <- nearZeroVar(sampleSet)
filteredDescr <- sampleSet[, -nzv]
dim(filteredDescr)
```

```
## [1] 11624    22
```

```
#Identify the Correlated Predictors
descrCor <- cor(filteredDescr[-1])
corrplot::corrplot(descrCor, method = "circle")
```




```
#Shows the correlation min-max values
summary(descrCor[upper.tri(descrCor)])
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.79045 -0.09696 -0.02774 -0.01915  0.06591  0.65044
```

```
############### PARTIAL LEAST SQUARES
library(pls)
```

```
##
## Attaching package: 'pls'
```

```
## The following object is masked from 'package:corrplot':
##
##     corrplot
```

```
## The following object is masked from 'package:caret':
##
##     R2
```

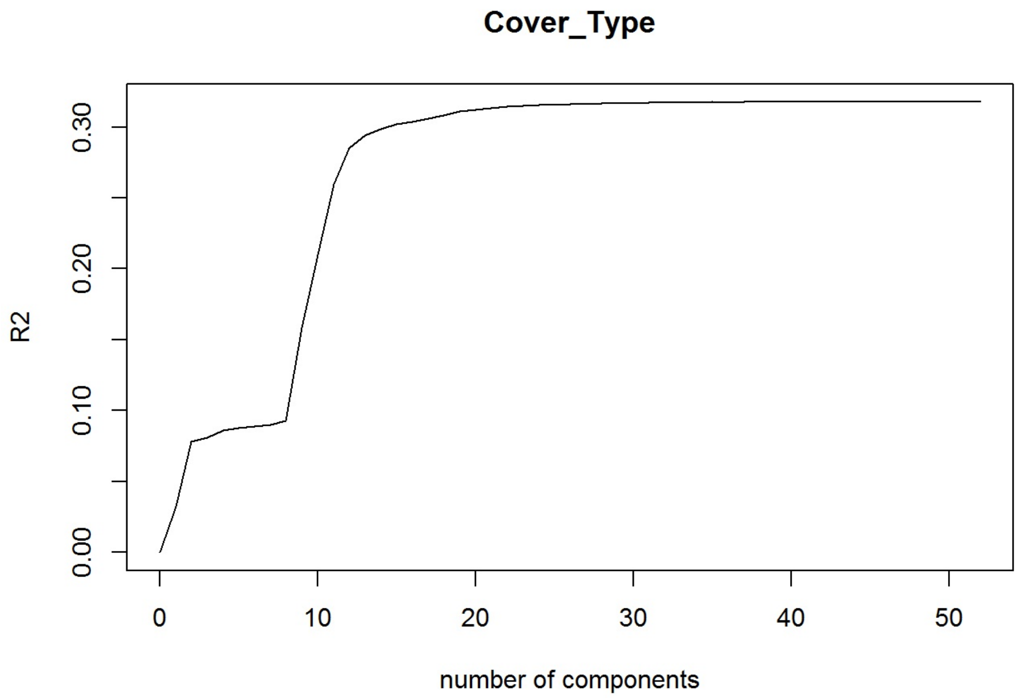
```
## The following object is masked from 'package:stats':
##
##     loadings
```

```
#Make another data frame of all numeric for the pls regression
sampleSet2pls <- as.data.frame(sapply( sampleSet, as.numeric))
```

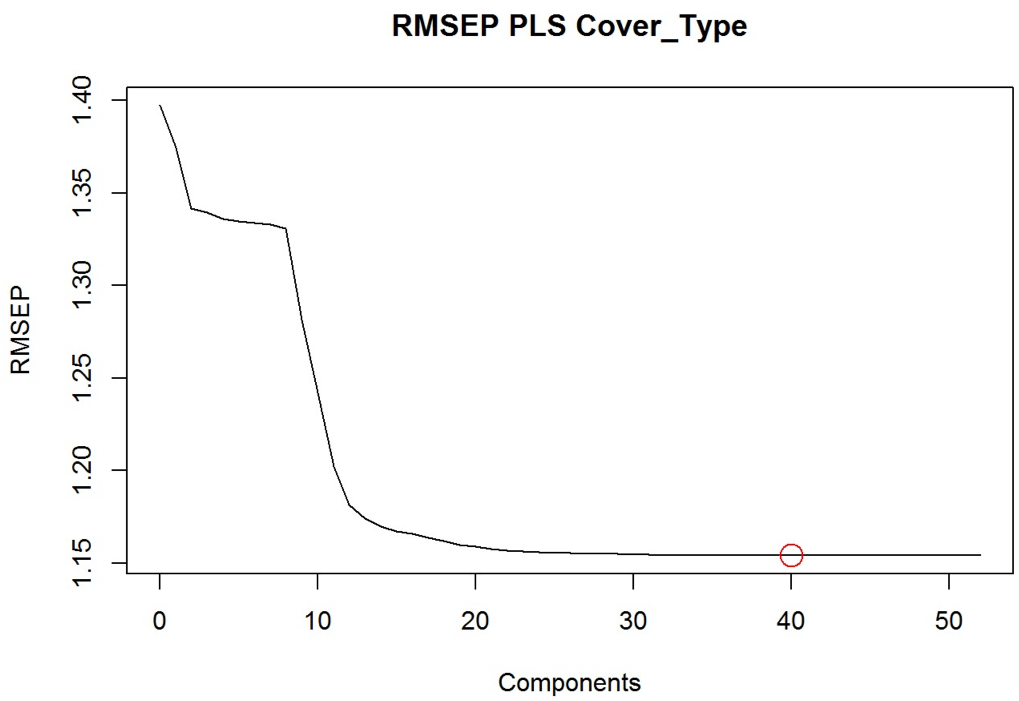
```
#this uses the pls package uses resources ok for now
set.seed(7)
fit1 <- plsr(Cover_Type~., data=sampleSet2pls, Scale=TRUE, validation ="CV") #PLS Lib conflict with Corrplot
summary(fit1)
```

```
## Data:    X dimension: 11624 52
##  Y dimension: 11624 1
## Fit method: kernelpls
## Number of components considered: 52
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV           1.397    1.375    1.342    1.339    1.336    1.335    1.334
## adjCV        1.397    1.375    1.341    1.340    1.336    1.335    1.334
##        7 comps  8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## CV       1.333    1.331    1.282     1.242     1.202     1.181     1.174
## adjCV    1.333    1.331    1.280     1.234     1.202     1.181     1.174
##        14 comps  15 comps  16 comps  17 comps  18 comps  19 comps
## CV         1.17     1.167     1.166     1.164     1.162     1.160
## adjCV      1.17     1.167     1.166     1.164     1.162     1.159
##        20 comps  21 comps  22 comps  23 comps  24 comps  25 comps
## CV        1.159     1.158     1.157     1.156     1.156     1.156
## adjCV     1.158     1.157     1.157     1.156     1.156     1.155
##        26 comps  27 comps  28 comps  29 comps  30 comps  31 comps
## CV        1.156     1.155     1.155     1.155     1.155     1.155
## adjCV     1.155     1.155     1.155     1.155     1.154     1.154
##        32 comps  33 comps  34 comps  35 comps  36 comps  37 comps
## CV        1.154     1.154     1.154     1.154     1.154     1.154
## adjCV     1.154     1.154     1.154     1.154     1.154     1.154
##        38 comps  39 comps  40 comps  41 comps  42 comps  43 comps
## CV        1.154     1.154     1.154     1.154     1.154     1.154
## adjCV     1.154     1.154     1.154     1.154     1.154     1.154
##        44 comps  45 comps  46 comps  47 comps  48 comps  49 comps
## CV        1.154     1.154     1.154     1.154     1.154     1.154
## adjCV     1.154     1.154     1.154     1.154     1.154     1.154
##        50 comps  51 comps  52 comps
## CV        1.154     1.154     1.154
## adjCV     1.154     1.154     1.154
##
## TRAINING: % variance explained
##             1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps
## X            65.620   68.181   98.490    99.66   99.732    99.99   99.999
## Cover_Type    3.222    7.972    8.139     8.68    8.978     9.04    9.126
##             8 comps  9 comps  10 comps  11 comps  12 comps  13 comps
## X           100.000   100.00     100.0    100.00       100    100.00
## Cover_Type    9.428    16.85      21.7     26.43        29     29.92
##             14 comps  15 comps  16 comps  17 comps  18 comps  19 comps
## X             100.00    100.00    100.00    100.00    100.00    100.00
## Cover_Type     30.49     30.89     31.11     31.46     31.78     32.04
##             20 comps  21 comps  22 comps  23 comps  24 comps  25 comps
## X             100.00    100.00    100.00    100.00    100.00    100.00
## Cover_Type     32.22     32.29     32.34     32.38     32.41     32.43
##             26 comps  27 comps  28 comps  29 comps  30 comps  31 comps
## X             100.00    100.00    100.00    100.00     100.0    100.00
## Cover_Type     32.45     32.47     32.48     32.49      32.5     32.53
##             32 comps  33 comps  34 comps  35 comps  36 comps  37 comps
## X             100.00    100.00    100.00    100.00    100.00    100.00
## Cover_Type     32.54     32.54     32.55     32.55     32.55     32.55
##             38 comps  39 comps  40 comps  41 comps  42 comps  43 comps
## X             100.00    100.00    100.00    100.00    100.00    100.00
## Cover_Type     32.55     32.55     32.55     32.55     32.55     32.55
##             44 comps  45 comps  46 comps  47 comps  48 comps  49 comps
## X             100.00    100.00    100.00    100.00    100.00    100.00
## Cover_Type     32.55     32.55     32.55     32.55     32.55     32.55
##             50 comps  51 comps  52 comps
## X             100.00    100.00    100.00
## Cover_Type     32.55     32.55     32.55
```

```
validationplot(fit1, val.type = "R2") #plots Optimal number of components Based on R^2
```




```
pls.RMSEP <- RMSEP(fit1, estimate="CV")
plot(pls.RMSEP, main="RMSEP PLS Cover_Type", xlab="Components")
min_comp <- which.min(pls.RMSEP$val)
points(min_comp, min(pls.RMSEP$val), pch=1, col="red", cex=2) #Show number of components in graph
```




```
#change all predictor variables to numeric
sampleSet <- sampleSet %>%
  mutate_if(is.integer, as.numeric)
#### PLS - Partial Least Squares
#split data
inTrain <- createDataPartition(y=sampleSet$Cover_Type, p=0.70, list = FALSE)
set.seed(78)
train <- sampleSet[ inTrain,]
test <- sampleSet[-inTrain,]
nrow(train)  # 8139 70%
```

```
## [1] 8139
```

```
nrow(test) #3485 30%
```

```
## [1] 3485
```

```
library(MLmetrics)
```

```
##
## Attaching package: 'MLmetrics'
```

```
## The following objects are masked from 'package:caret':
##
##     MAE, RMSE
```

```
## The following object is masked from 'package:base':
##
##     Recall
```

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(doParallel)
```

```
## Loading required package: foreach
```

```
##
## Attaching package: 'foreach'
```

```
## The following objects are masked from 'package:purrr':
##
##     accumulate, when
```

```
## Loading required package: iterators
```

```
## Loading required package: parallel
```

```
#Caret PLS Model Shows Cover_Type relationship to variables
#Train Control for Model
###RUN DoParallel on 6 cores for Caret train
ctrl <- trainControl(method = "repeatedcv",
                     repeats = 3, #tried 10, long run time
                     classProbs = TRUE,
                     summaryFunction = multiClassSummary)
###Set Cores for Parallel Processing
cl<- makePSOCKcluster(6)
registerDoParallel(cl)
```

```
plsFit <- train(Cover_Type~.,
                data = train,
                method = "pls",
                preProc = c("center", "scale"),
                tuneLength =40, #per an earlier run of this model
                trControl = ctrl,
                metric = "ROC") #MLMetrics Lib load after Caret and After using RMSE
```

```
## Warning in train.default(x, y, weights = w, ...): The metric "ROC" was not
## in the result set. logLoss will be used instead.
```

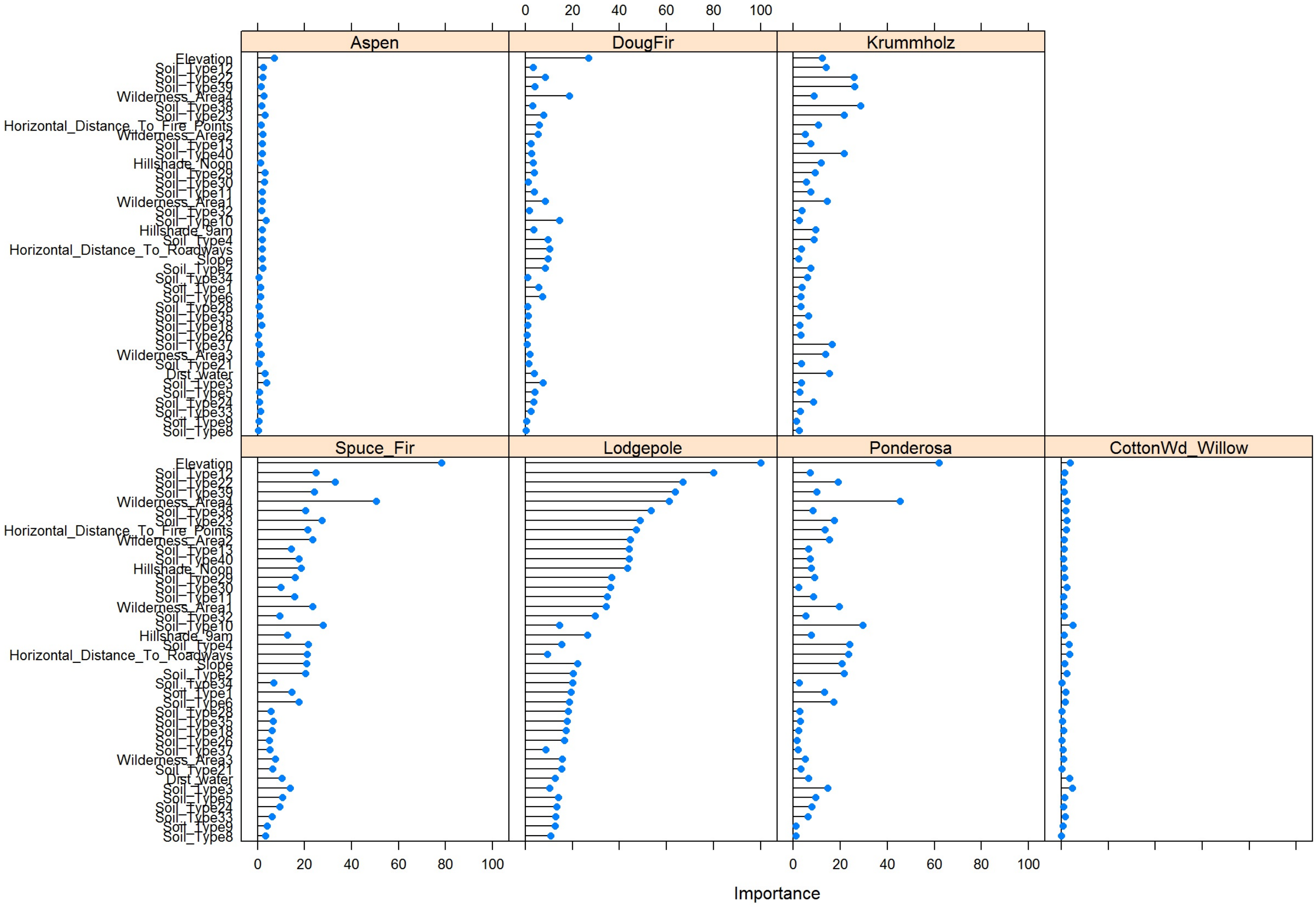
```
## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info =
## trainInfo, : There were missing values in resampled performance measures.
```

```
## Warning in fitFunc(X, Y, ncomp, Y.add = Y.add, center = center, ...): No convergence in 100 iterations

## Warning in fitFunc(X, Y, ncomp, Y.add = Y.add, center = center, ...): No convergence in 100 iterations

## Warning in fitFunc(X, Y, ncomp, Y.add = Y.add, center = center, ...): No convergence in 100 iterations

## Warning in fitFunc(X, Y, ncomp, Y.add = Y.add, center = center, ...): No convergence in 100 iterations

## Warning in fitFunc(X, Y, ncomp, Y.add = Y.add, center = center, ...): No convergence in 100 iterations

## Warning in fitFunc(X, Y, ncomp, Y.add = Y.add, center = center, ...): No convergence in 100 iterations
```

```
pltImp <- varImp(plsFit, scale = FALSE)
pltImp
```

```
## pls variable importance
##
##   variables are sorted by maximum importance across the classes
##   only 20 most important variables shown (out of 52)
##
##                                   Spuce_Fir Lodgepole Ponderosa
## Elevation                          0.050869  0.064914  0.040340
## Soil_Type12                        0.016167  0.051953  0.004685
## Soil_Type22                        0.021577  0.043462  0.012483
## Soil_Type39                        0.015761  0.041421  0.006532
## Wilderness_Area4                   0.032784  0.039797  0.029521
## Soil_Type38                        0.013273  0.034702  0.005444
## Soil_Type23                        0.017855  0.031793  0.011418
## Horizontal_Distance_To_Fire_Points  0.014021  0.030617  0.008797
## Wilderness_Area2                   0.015234  0.029001  0.009992
## Soil_Type13                        0.009393  0.028744  0.004337
## Soil_Type40                        0.011547  0.028727  0.004782
## Hillshade_Noon                     0.012112  0.028205  0.005122
## Soil_Type29                        0.010447  0.023812  0.005943
## Soil_Type30                        0.006549  0.023539  0.001592
## Soil_Type11                        0.010356  0.022721  0.005727
## Wilderness_Area1                   0.015363  0.022379  0.012781
## Soil_Type32                        0.006260  0.019291  0.003510
## Soil_Type10                        0.018174  0.009472  0.019224
## Hillshade_9am                      0.008303  0.017168  0.005054
## Soil_Type4                         0.014041  0.010062  0.015572
##                                   CottonWd_Willow    Aspen  DougFir
## Elevation                                0.0026503 0.004664 0.017569
## Soil_Type12                              0.0011460 0.001760 0.002252
## Soil_Type22                              0.0008023 0.001521 0.005524
## Soil_Type39                              0.0009055 0.001022 0.002713
## Wilderness_Area4                         0.0016188 0.001926 0.012191
## Soil_Type38                              0.0014172 0.001240 0.002159
## Soil_Type23                              0.0016780 0.002096 0.005153
## Horizontal_Distance_To_Fire_Points       0.0015798 0.001114 0.003902
## Wilderness_Area2                         0.0009458 0.001475 0.003640
## Soil_Type13                              0.0009842 0.001369 0.001569
## Soil_Type40                              0.0007426 0.001401 0.001844
## Hillshade_Noon                           0.0008710 0.001006 0.002202
## Soil_Type29                              0.0011371 0.002110 0.002499
## Soil_Type30                              0.0017010 0.001940 0.000845
## Soil_Type11                              0.0007592 0.001437 0.002606
## Wilderness_Area1                         0.0009343 0.001342 0.005512
## Soil_Type32                              0.0009234 0.001277 0.001151
## Soil_Type10                              0.0033004 0.002445 0.009538
## Hillshade_9am                            0.0009531 0.001425 0.002445
## Soil_Type4                               0.0022395 0.001370 0.006373
##                                   Krummholz
## Elevation                          0.008098
## Soil_Type12                        0.009112
## Soil_Type22                        0.016821
## Soil_Type39                        0.017051
## Wilderness_Area4                   0.005811
## Soil_Type38                        0.018715
## Soil_Type23                        0.014065
## Horizontal_Distance_To_Fire_Points  0.007040
## Wilderness_Area2                   0.003418
## Soil_Type13                        0.004958
## Soil_Type40                        0.014203
## Hillshade_Noon                     0.007784
## Soil_Type29                        0.006135
## Soil_Type30                        0.003645
## Soil_Type11                        0.004951
## Wilderness_Area1                   0.009509
## Soil_Type32                        0.002565
## Soil_Type10                        0.001795
## Hillshade_9am                      0.006215
## Soil_Type4                         0.005832
```

```
plot(varImp(plsFit), top =40)
```



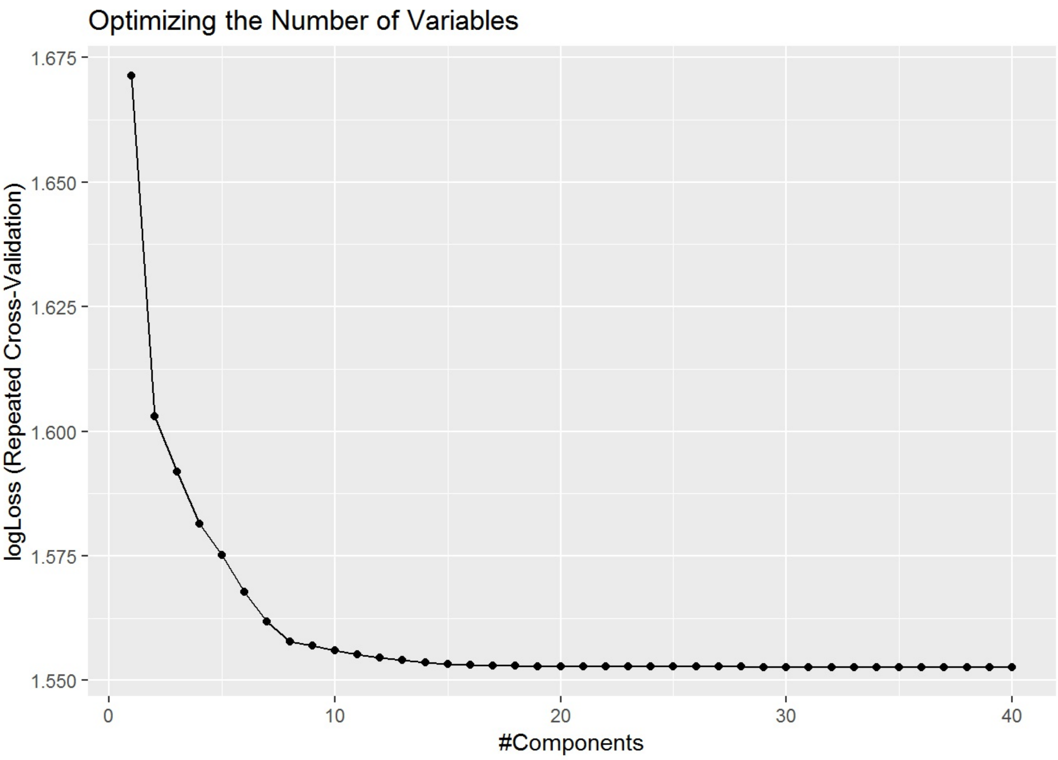

```
plsFit
```

```
## Partial Least Squares
##
## 8139 samples
##   52 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## Pre-processing: centered (52), scaled (52)
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 7326, 7324, 7328, 7326, 7325, 7324, ...
## Resampling results across tuning parameters:
##
##   ncomp  logLoss   AUC        prAUC      Accuracy   Kappa      Mean_F1
##    1     1.671411  0.8100441  0.3003262  0.5741132  0.2380725  NaN
##    2     1.603004  0.8730766  0.3771170  0.6436951  0.3814601  NaN
##    3     1.591944  0.8768968  0.3873379  0.6445948  0.3875054  NaN
##    4     1.581390  0.8831506  0.4168369  0.6484049  0.3984174  NaN
##    5     1.575186  0.8814134  0.4116739  0.6635166  0.4312264  NaN
##    6     1.567687  0.8856730  0.4252681  0.6687135  0.4436736  NaN
##    7     1.561775  0.8905267  0.4289618  0.6813691  0.4656633  NaN
##    8     1.557718  0.8913756  0.4409655  0.6930821  0.4875537  NaN
##    9     1.556925  0.8893991  0.4525591  0.6930411  0.4875276  NaN
##   10     1.555982  0.8981783  0.4658083  0.6931638  0.4876234  NaN
##   11     1.555194  0.9034791  0.4790882  0.6959081  0.4922149  NaN
##   12     1.554599  0.9067460  0.4810249  0.6976682  0.4957217  NaN
##   13     1.554140  0.9055322  0.4810007  0.6981598  0.4969839  NaN
##   14     1.553668  0.9042046  0.4857022  0.6985278  0.4979579  NaN
##   15     1.553286  0.9045352  0.4915406  0.6990187  0.4993185  NaN
##   16     1.553094  0.9031254  0.4937551  0.6993058  0.5000291  NaN
##   17     1.552975  0.9028767  0.4928247  0.6995108  0.5005019  NaN
##   18     1.552904  0.9031274  0.4934482  0.6995516  0.5005772  NaN
##   19     1.552848  0.9033010  0.4948359  0.6991412  0.5000192  NaN
##   20     1.552817  0.9036448  0.4946674  0.6993054  0.5003180  NaN
##   21     1.552796  0.9038361  0.4951374  0.6993462  0.5003921  NaN
##   22     1.552780  0.9042694  0.4961562  0.6993463  0.5004248  NaN
##   23     1.552774  0.9044634  0.4964854  0.6991004  0.5000404  NaN
##   24     1.552770  0.9043999  0.4974948  0.6992645  0.5002927  NaN
##   25     1.552768  0.9043961  0.4959682  0.6993874  0.5004879  NaN
```

```
##    26     1.552769  0.9043935  0.4961157  0.6993875  0.5004977  NaN
##    27     1.552769  0.9044149  0.4962346  0.6993055  0.5003430  NaN
##    28     1.552763  0.9043448  0.4958404  0.6993054  0.5002891  NaN
##    29     1.552722  0.9039328  0.4958389  0.6993055  0.5002622  NaN
##    30     1.552721  0.9039246  0.4958675  0.6992236  0.5001273  NaN
##    31     1.552721  0.9039154  0.4958682  0.6992646  0.5002074  NaN
##    32     1.552721  0.9039147  0.4958466  0.6992646  0.5002074  NaN
##    33     1.552721  0.9039139  0.4958677  0.6992646  0.5002074  NaN
##    34     1.552721  0.9039131  0.4959021  0.6992646  0.5002074  NaN
##    35     1.552721  0.9039162  0.4959063  0.6992236  0.5001344  NaN
##    36     1.552721  0.9039173  0.4959075  0.6992646  0.5002074  NaN
##    37     1.552721  0.9039161  0.4959070  0.6992646  0.5002074  NaN
##    38     1.552721  0.9039166  0.4959074  0.6992646  0.5002074  NaN
##    39     1.552721  0.9039166  0.4959074  0.6992646  0.5002074  NaN
##    40     1.552721  0.9039166  0.4959074  0.6992646  0.5002074  NaN
##    Mean_Sensitivity  Mean_Specificity  Mean_Pos_Pred_Value
##    0.2508613         0.8879881         NaN
##    0.2819738         0.9096136         NaN
##    0.3009732         0.9101515         NaN
##    0.3129003         0.9117903         NaN
##    0.3343189         0.9167783         NaN
##    0.3401998         0.9189600         NaN
##    0.3447986         0.9223504         NaN
##    0.3524747         0.9258210         NaN
##    0.3522432         0.9258335         NaN
##    0.3519032         0.9258613         NaN
##    0.3523356         0.9265704         NaN
##    0.3536623         0.9271491         NaN
##    0.3554392         0.9273601         NaN
##    0.3574214         0.9275102         NaN
##    0.3622414         0.9276741         NaN
##    0.3633378         0.9277973         NaN
##    0.3644489         0.9278620         NaN
##    0.3646008         0.9278730         NaN
##    0.3645965         0.9277981         NaN
##    0.3650930         0.9278360         NaN
##    0.3652873         0.9278451         NaN
##    0.3653035         0.9278540         NaN
##    0.3650329         0.9278036         NaN
##    0.3650851         0.9278374         NaN
##    0.3651211         0.9278674         NaN
##    0.3651253         0.9278696         NaN
##    0.3650890         0.9278446         NaN
##    0.3652511         0.9278230         NaN
##    0.3653529         0.9278119         NaN
##    0.3653248         0.9277913         NaN
##    0.3654890         0.9278005         NaN
##    0.3654890         0.9278005         NaN
##    0.3654890         0.9278005         NaN
##    0.3654890         0.9278005         NaN
##    0.3654729         0.9277891         NaN
##    0.3654890         0.9278005         NaN
##    0.3654890         0.9278005         NaN
##    0.3654890         0.9278005         NaN
##    0.3654890         0.9278005         NaN
##    0.3654890         0.9278005         NaN
##    Mean_Neg_Pred_Value  Mean_Precision  Mean_Recall  Mean_Detection_Rate
##    0.9071284            NaN             0.2508613    0.08201617
##    0.9198351            NaN             0.2819738    0.09195644
##    0.9205859            NaN             0.3009732    0.09208498
##    0.9204834            NaN             0.3129003    0.09262927
##    0.9236542            NaN             0.3343189    0.09478809
##    0.9242791            NaN             0.3401998    0.09553051
##    0.9274707            NaN             0.3447986    0.09733844
##    0.9303514            NaN             0.3524747    0.09901173
##    0.9303370            NaN             0.3522432    0.09900587
##    0.9304051            NaN             0.3519032    0.09902340
##    0.9311163            NaN             0.3523356    0.09941544
##    0.9314533            NaN             0.3536623    0.09966688
##    0.9315631            NaN             0.3554392    0.09973712
##    0.9316289            NaN             0.3574214    0.09978969
##    0.9317236            NaN             0.3622414    0.09985981
##    0.9318472            NaN             0.3633378    0.09990083
##    0.9319016            NaN             0.3644489    0.09993011
##    0.9319181            NaN             0.3646008    0.09993594
##    0.9317802            NaN             0.3645965    0.09987732
```

```
##   0.9318204            NaN             0.3650930    0.09990076
##   0.9318295            NaN             0.3652873    0.09990660
##   0.9318219            NaN             0.3653035    0.09990661
##   0.9317676            NaN             0.3650329    0.09987149
##   0.9318123            NaN             0.3650851    0.09989492
##   0.9318479            NaN             0.3651211    0.09991249
##   0.9318457            NaN             0.3651253    0.09991249
##   0.9318268            NaN             0.3650890    0.09990079
##   0.9318341            NaN             0.3652511    0.09990077
##   0.9318350            NaN             0.3653529    0.09990078
##   0.9318130            NaN             0.3653248    0.09988909
##   0.9318203            NaN             0.3654890    0.09989494
##   0.9318203            NaN             0.3654890    0.09989494
##   0.9318203            NaN             0.3654890    0.09989494
##   0.9318203            NaN             0.3654890    0.09989494
##   0.9318102            NaN             0.3654729    0.09988909
##   0.9318203            NaN             0.3654890    0.09989494
##   0.9318203            NaN             0.3654890    0.09989494
##   0.9318203            NaN             0.3654890    0.09989494
##   0.9318203            NaN             0.3654890    0.09989494
##   0.9318203            NaN             0.3654890    0.09989494
##   Mean_Balanced_Accuracy
##   0.5694247
##   0.5957937
##   0.6055623
##   0.6123453
##   0.6255486
##   0.6295799
##   0.6335745
##   0.6391478
##   0.6390383
##   0.6388822
##   0.6394530
##   0.6404057
##   0.6413996
##   0.6424658
##   0.6449578
##   0.6455675
##   0.6461555
##   0.6462369
##   0.6461973
##   0.6464645
##   0.6465662
##   0.6465788
##   0.6464182
##   0.6464612
##   0.6464943
##   0.6464975
##   0.6464668
##   0.6465370
##   0.6465824
##   0.6465581
##   0.6466448
##   0.6466448
##   0.6466448
##   0.6466448
##   0.6466310
##   0.6466448
##   0.6466448
##   0.6466448
##   0.6466448
##   0.6466448
##
## logLoss was used to select the optimal model using the smallest value.
## The final value used for the model was ncomp = 36.
```

```
ggplot(plsFit)  + ggtitle("Optimizing the Number of Variables")
```




```
plsClasses <- predict(plsFit, newdata = test)
plsProbs <- predict(plsFit, newdata = test, type = "prob")
head(plsProbs)
```

	Spuce_Fir <dbl>	Lodgepole <dbl>	Ponderosa <dbl>	CottonWd_Willow <dbl>	Aspen <dbl>	DougFir <dbl>	Krummholz <dbl>
3	0.1706668	0.2301951	0.1185397	0.1207259	0.1185485	0.1188924	0.1224318
9	0.1530704	0.2448218	0.1199687	0.1200330	0.1200292	0.1210265	0.1210503
11	0.1674798	0.2291630	0.1210949	0.1208964	0.1199811	0.1201384	0.1212464
14	0.1149465	0.3050844	0.1173616	0.1147919	0.1182572	0.1178363	0.1117220
17	0.1464141	0.2505476	0.1216634	0.1192392	0.1232829	0.1219128	0.1169400
18	0.1838887	0.2143382	0.1233069	0.1204475	0.1252827	0.1228429	0.1098931

6 rows

```
confusionMatrix(data = plsClasses, test$Cover_Type) #confusion matrix
```

```
## Confusion Matrix and Statistics
##
##                  Reference
## Prediction        Spuce_Fir Lodgepole Ponderosa CottonWd_Willow Aspen
##   Spuce_Fir             833       260         0               0     2
##   Lodgepole             419      1391        22               0    51
##   Ponderosa               1        45       192              16     4
##   CottonWd_Willow         0         0         0               0     0
##   Aspen                   0         0         0               0     0
##   DougFir                 0         0         0               0     0
##   Krummholz              18         4         0               0     0
##                  Reference
## Prediction        DougFir Krummholz
##   Spuce_Fir             0        87
##   Lodgepole            39         3
##   Ponderosa            65         0
##   CottonWd_Willow       0         0
##   Aspen                 0         0
##   DougFir               0         0
##   Krummholz             0        33
##
## Overall Statistics
##
##                Accuracy : 0.7027
##                  95% CI : (0.6872, 0.7179)
##     No Information Rate : 0.4878
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.505
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Spuce_Fir Class: Lodgepole Class: Ponderosa
## Sensitivity                    0.6554           0.8182          0.89720
## Specificity                    0.8424           0.7008          0.95995
## Pos Pred Value                 0.7047           0.7226          0.59443
## Neg Pred Value                 0.8098           0.8019          0.99304
## Prevalence                     0.3647           0.4878          0.06141
## Detection Rate                 0.2390           0.3991          0.05509
## Detection Prevalence           0.3392           0.5524          0.09268
## Balanced Accuracy              0.7489           0.7595          0.92857
##                      Class: CottonWd_Willow Class: Aspen Class: DougFir
## Sensitivity                        0.000000      0.00000        0.00000
## Specificity                        1.000000      1.00000        1.00000
## Pos Pred Value                          NaN          NaN            NaN
## Neg Pred Value                     0.995409      0.98364        0.97016
## Prevalence                         0.004591      0.01636        0.02984
## Detection Rate                     0.000000      0.00000        0.00000
## Detection Prevalence               0.000000      0.00000        0.00000
## Balanced Accuracy                  0.500000      0.50000        0.50000
##                      Class: Krummholz
## Sensitivity                  0.268293
## Specificity                  0.993456
## Pos Pred Value               0.600000
## Neg Pred Value               0.973761
## Prevalence                   0.035294
## Detection Rate               0.009469
## Detection Prevalence         0.015782
## Balanced Accuracy            0.630874
```

```r
library(randomForest)
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
#Rank the Variable Importance
ctrl2 <- trainControl(method="repeatedcv", number=10, repeats=3)
```

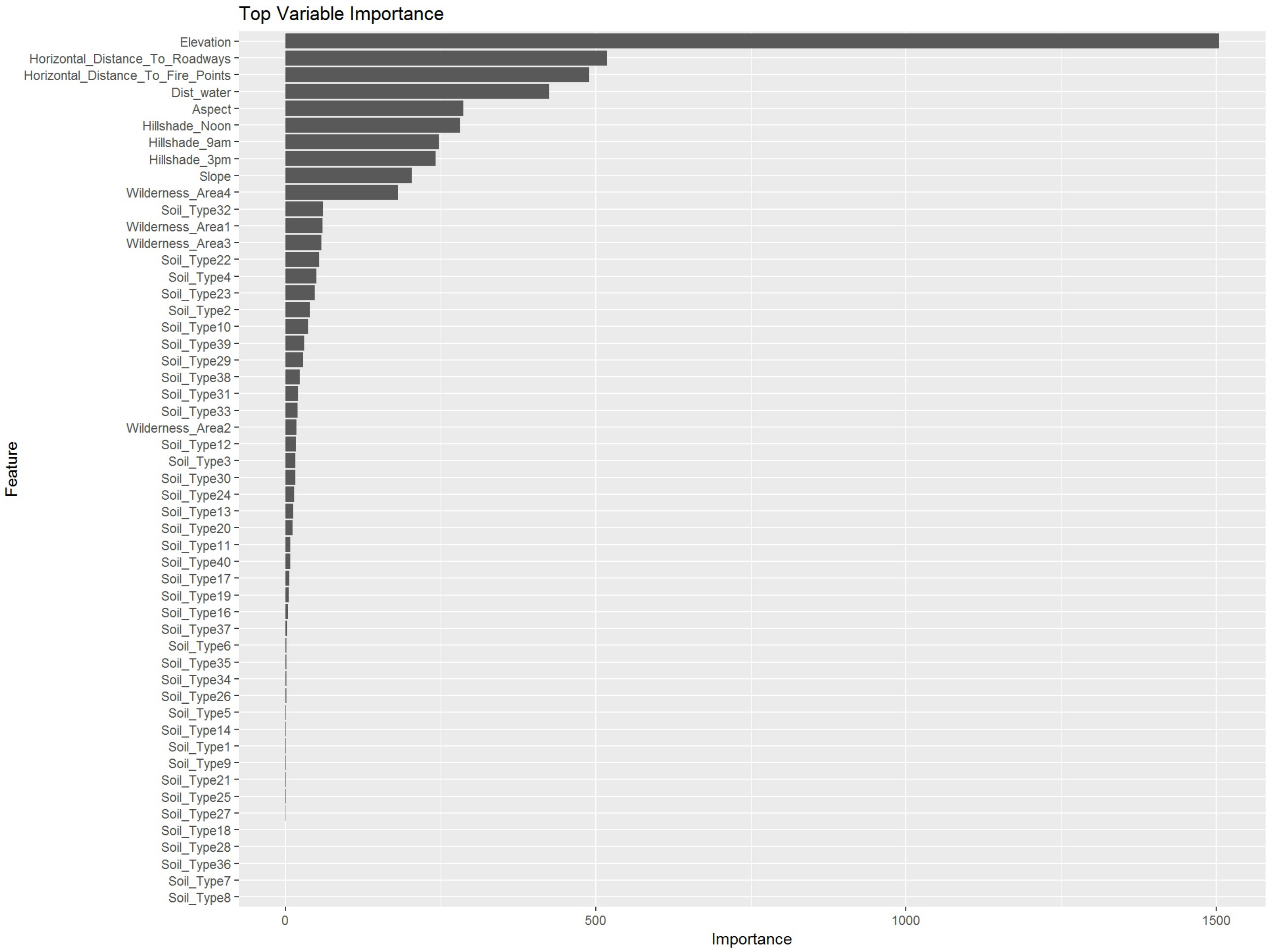
```
#Random Forest Model to select Features
set.seed(90)
mdl <- train(Cover_Type~.,
             data = train,
             method = "rf",
             preProcess = c("center", "scale"),
             trControl = ctrl2)
mdl
```

```
## Random Forest
##
## 8139 samples
##   52 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## Pre-processing: centered (52), scaled (52)
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 7325, 7325, 7325, 7327, 7325, 7325, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    2    0.6392247  0.3627254
##   27    0.8084478  0.6869038
##   52    0.8050506  0.6818103
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 27.
```

```
importance <- varImp(mdl, scale=FALSE)
((importance))
```

```
## rf variable importance
##
##   only 20 most important variables shown (out of 52)
##
##                                    Overall
## Elevation                          1503.96
## Horizontal_Distance_To_Roadways     518.32
## Horizontal_Distance_To_Fire_Points  489.67
## Dist_water                          425.60
## Aspect                              287.11
## Hillshade_Noon                      281.48
## Hillshade_9am                       247.90
## Hillshade_3pm                       242.07
## Slope                               204.13
## Wilderness_Area4                    181.37
## Soil_Type32                          60.76
## Wilderness_Area1                     59.80
## Wilderness_Area3                     58.65
## Soil_Type22                          54.69
## Soil_Type4                           49.99
## Soil_Type23                          47.58
## Soil_Type2                           39.40
## Soil_Type10                          36.70
## Soil_Type39                          30.93
## Soil_Type29                          28.35
```

```
ggplot(importance)  + ggtitle("Top Variable Importance")
```

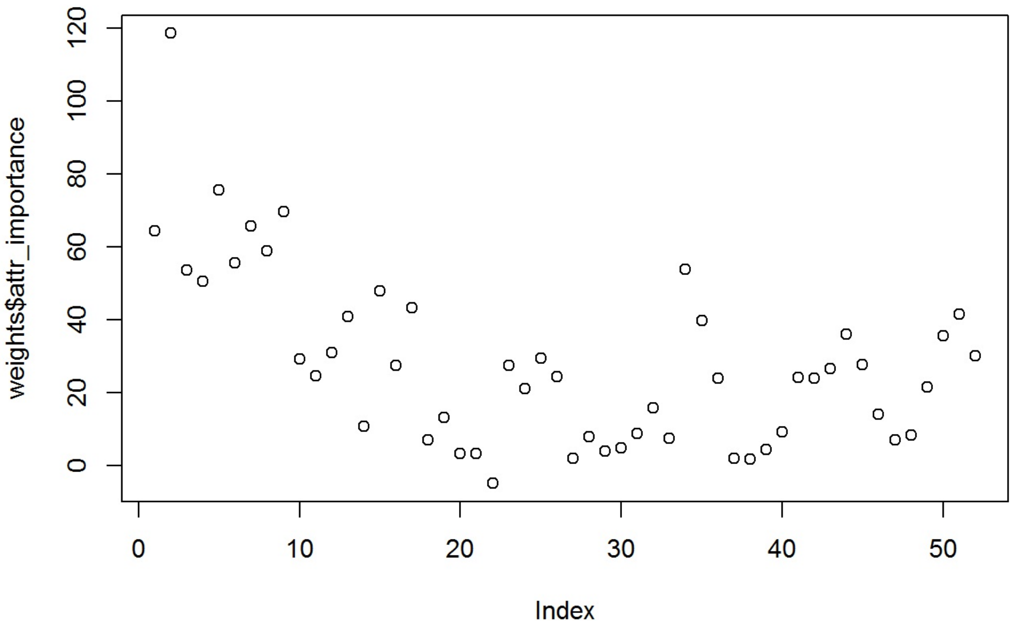



```r
#Feature Selection With FSelector Note very similar to Random Forest
library(FSelector)
set.seed(87)
weights <- random.forest.importance(Cover_Type ~., train, importance.type = 1)
((weights))
```

	attr_importance <dbl>
Dist_water	64.488261
Elevation	118.659635
Aspect	53.667005
Slope	50.664313
Horizontal_Distance_To_Roadways	75.591148
Hillshade_9am	55.689670
Hillshade_Noon	65.675508
Hillshade_3pm	59.035873
Horizontal_Distance_To_Fire_Points	69.746683
Wilderness_Area1	29.262278

1-10 of 52 rows Previous **1** 2 3 4 5 6 Next

```r
plot(weights$attr_importance)
```




```
subset <- cutoff.k(weights, 40) #top 40 variables
f <- as.simple.formula(subset, "Cover_Type")
((f))
```

```
## Cover_Type ~ Elevation + Horizontal_Distance_To_Roadways + Horizontal_Distance_To_Fire_Points +
##     Hillshade_Noon + Dist_water + Hillshade_3pm + Hillshade_9am +
##     Soil_Type22 + Aspect + Slope + Soil_Type2 + Soil_Type4 +
##     Soil_Type39 + Wilderness_Area4 + Soil_Type23 + Soil_Type32 +
##     Soil_Type38 + Wilderness_Area3 + Soil_Type40 + Soil_Type12 +
##     Wilderness_Area1 + Soil_Type33 + Soil_Type10 + Soil_Type3 +
##     Soil_Type31 + Wilderness_Area2 + Soil_Type13 + Soil_Type29 +
##     Soil_Type30 + Soil_Type24 + Soil_Type37 + Soil_Type11 + Soil_Type20 +
##     Soil_Type34 + Soil_Type6 + Soil_Type1 + Soil_Type28 + Soil_Type19 +
##     Soil_Type36 + Soil_Type16
## <environment: 0x0000000023ea5cc8>
```

```
#########Reduce data to Important Variables
trainRd <- select(train,
                  Cover_Type,
                  Elevation,
                  Horizontal_Distance_To_Roadways,
                  Horizontal_Distance_To_Fire_Points,
                  Dist_water,
                  Hillshade_Noon,
                  Hillshade_3pm,
                  Hillshade_9am,
                  Aspect,
                  Slope,
                  Wilderness_Area4,
                  Wilderness_Area3,
                  Wilderness_Area1,
                  Wilderness_Area2,
                  Soil_Type2,
                  Soil_Type3,
                  Soil_Type4,
                  Soil_Type6,
                  Soil_Type10,
                  Soil_Type11,
                  Soil_Type12,
                  Soil_Type13,
                  Soil_Type16,
                  Soil_Type17,
                  Soil_Type19,
                  Soil_Type20,
                  Soil_Type22,
                  Soil_Type23,
                  Soil_Type24,
                  Soil_Type29,
                  Soil_Type30,
                  Soil_Type31,
                  Soil_Type32,
                  Soil_Type33,
                  Soil_Type34,
                  Soil_Type36,
                  Soil_Type37,
                  Soil_Type38,
                  Soil_Type39,
                  Soil_Type40
)
str(trainRd)
```

```
## 'data.frame':	8139 obs. of  40 variables:
##  $ Cover_Type                        : Factor w/ 7 levels "Spuce_Fir","Lodgepole",..: 5 2 2 2 2 5 1 1 2 2 ..
.
##  $ Elevation                         : num  2489 2745 2847 2874 2791 ...
##  $ Horizontal_Distance_To_Roadways   : num  840 2416 4983 2869 1740 ...
##  $ Horizontal_Distance_To_Fire_Points: num  5254 6428 4727 3653 2294 ...
##  $ Dist_water                        : num  175.5 71.2 93.5 400.5 458.4 ...
##  $ Hillshade_Noon                    : num  232 234 190 236 201 234 218 219 234 203 ...
##  $ Hillshade_3pm                     : num  153 184 199 157 111 119 123 157 171 104 ...
##  $ Hillshade_9am                     : num  216 190 120 216 220 241 227 200 202 228 ...
##  $ Aspect                            : num  11 306 320 352 41 136 56 353 318 54 ...
##  $ Slope                             : num  4 11 33 2 17 14 11 12 7 16 ...
##  $ Wilderness_Area4                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Wilderness_Area3                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Wilderness_Area1                  : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Wilderness_Area2                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type2                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type3                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type4                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type6                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type10                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type11                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type12                       : num  0 0 0 0 1 0 0 0 1 0 ...
##  $ Soil_Type13                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type16                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type17                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type19                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type20                       : num  0 0 0 0 0 0 0 1 0 0 ...
##  $ Soil_Type22                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type23                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type24                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type29                       : num  0 1 1 1 0 0 1 0 0 1 ...
##  $ Soil_Type30                       : num  0 0 0 0 0 1 0 0 0 0 ...
##  $ Soil_Type31                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type32                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type33                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type34                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type36                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type37                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type38                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type39                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type40                       : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
testRd <- select(test,
                  Cover_Type,
                  Elevation,
                  Horizontal_Distance_To_Roadways,
                  Horizontal_Distance_To_Fire_Points,
                  Dist_water,
                  Hillshade_Noon,
                  Hillshade_3pm,
                  Hillshade_9am,
                  Aspect,
                  Slope,
                  Wilderness_Area4,
                  Wilderness_Area3,
                  Wilderness_Area1,
                  Wilderness_Area2,
                  Soil_Type2,
                  Soil_Type3,
                  Soil_Type4,
                  Soil_Type6,
                  Soil_Type10,
                  Soil_Type11,
                  Soil_Type12,
                  Soil_Type13,
                  Soil_Type16,
                  Soil_Type17,
                  Soil_Type19,
                  Soil_Type20,
                  Soil_Type22,
                  Soil_Type23,
                  Soil_Type24,
                  Soil_Type29,
                  Soil_Type30,
                  Soil_Type31,
                  Soil_Type32,
                  Soil_Type33,
                  Soil_Type34,
                  Soil_Type36,
                  Soil_Type37,
                  Soil_Type38,
                  Soil_Type39,
                  Soil_Type40
)
str(testRd)
```

```
## 'data.frame':    3485 obs. of  40 variables:
##  $ Cover_Type                        : Factor w/ 7 levels "Spuce_Fir","Lodgepole",..: 2 2 1 2 2 5 5 1 1 5 ..
.
##  $ Elevation                         : num  3073 2922 3047 2645 2822 ...
##  $ Horizontal_Distance_To_Roadways   : num  6836 4283 5664 467 1758 ...
##  $ Horizontal_Distance_To_Fire_Points: num  2735 4027 3811 1368 2765 ...
##  $ Dist_water                        : num  108 219 623 134 243 ...
##  $ Hillshade_Noon                    : num  246 237 222 230 237 210 212 230 224 187 ...
##  $ Hillshade_3pm                     : num  149 147 156 145 171 120 103 129 138 73 ...
##  $ Hillshade_9am                     : num  227 225 204 221 205 222 237 233 221 234 ...
##  $ Aspect                            : num  173 129 356 45 308 44 75 101 40 64 ...
##  $ Slope                             : num  12 3 10 4 5 13 15 8 7 23 ...
##  $ Wilderness_Area4                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Wilderness_Area3                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Wilderness_Area1                  : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Wilderness_Area2                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type2                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type3                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type4                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type6                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type10                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type11                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type12                       : num  0 0 0 1 0 0 0 0 0 0 ...
##  $ Soil_Type13                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type16                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type17                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type19                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type20                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type22                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type23                       : num  0 0 0 0 0 1 0 0 0 0 ...
##  $ Soil_Type24                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type29                       : num  1 1 1 0 1 0 1 0 1 0 ...
##  $ Soil_Type30                       : num  0 0 0 0 0 0 0 1 0 1 ...
##  $ Soil_Type31                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type32                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type33                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type34                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type36                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type37                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type38                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type39                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type40                       : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
preProcValues <- caret::preProcess(sampleSet, method = c("range")) #normalize, have tried center and scale
sampleSetTran <- predict(preProcValues, sampleSet)
str(sampleSetTran)
```

```
## 'data.frame':    11624 obs. of  53 variables:
##  $ Cover_Type                        : Factor w/ 7 levels "Spuce_Fir","Lodgepole",..: 5 2 2 2 2 2 5 1 2 1 ..
.
##  $ Dist_water                        : num  0.1318 0.0535 0.0812 0.0703 0.3009 ...
##  $ Elevation                         : num  0.308 0.441 0.612 0.494 0.508 ...
##  $ Aspect                            : num  0.0306 0.85 0.4806 0.8889 0.9778 ...
##  $ Slope                             : num  0.0769 0.2115 0.2308 0.6346 0.0385 ...
##  $ Horizontal_Distance_To_Roadways   : num  0.12 0.344 0.974 0.71 0.409 ...
##  $ Hillshade_9am                     : num  0.85 0.748 0.894 0.472 0.85 ...
##  $ Hillshade_Noon                    : num  0.864 0.877 0.951 0.605 0.889 ...
##  $ Hillshade_3pm                     : num  0.602 0.724 0.587 0.783 0.618 ...
##  $ Horizontal_Distance_To_Fire_Points: num  0.745 0.912 0.388 0.67 0.518 ...
##  $ Wilderness_Area1                  : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Wilderness_Area2                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Wilderness_Area3                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Wilderness_Area4                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type1                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type2                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type3                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type4                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type5                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type6                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type7                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type8                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type9                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type10                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type11                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type12                       : num  0 0 0 0 0 1 0 0 0 0 ...
##  $ Soil_Type13                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type14                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type16                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type17                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type18                       : num  1 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type19                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type20                       : num  0 0 0 0 0 0 0 0 0 1 ...
##  $ Soil_Type21                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type22                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type23                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type24                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type25                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type26                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type27                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type28                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type29                       : num  0 1 1 1 1 0 0 1 1 0 ...
##  $ Soil_Type30                       : num  0 0 0 0 0 0 1 0 0 0 ...
##  $ Soil_Type31                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type32                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type33                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type34                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type35                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type36                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type37                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type38                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type39                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type40                       : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
########## PROPORTIONS AND NORMALIZING
round(prop.table((table(sampleSetTran$Cover_Type))), 3) *100 #class imbalance
```

```
## 
##       Spuce_Fir       Lodgepole       Ponderosa CottonWd_Willow 
##            36.5            48.8             6.2             0.5 
##           Aspen         DougFir       Krummholz 
##             1.6             3.0             3.5
```

```
summary(sampleSetTran)
```

```
##            Cover_Type     Dist_water        Elevation     
##  Spuce_Fir      :4237   Min.   :0.00000   Min.   :0.0000  
##  Lodgepole      :5667   1st Qu.:0.08149   1st Qu.:0.4756  
##  Ponderosa      : 716   Median :0.17189   Median :0.5711  
##  CottonWd_Willow:  55   Mean   :0.20807   Mean   :0.5527
```

```
##  Aspen          : 190   3rd Qu.:0.29605   3rd Qu.:0.6606
##  DougFir        : 348   Max.   :1.00000   Max.   :1.0000
##  Krummholz      : 411
##      Aspect           Slope        Horizontal_Distance_To_Roadways
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.1611   1st Qu.:0.1731   1st Qu.:0.1578
##  Median :0.3528   Median :0.2500   Median :0.2853
##  Mean   :0.4328   Mean   :0.2729   Mean   :0.3359
##  3rd Qu.:0.7250   3rd Qu.:0.3462   3rd Qu.:0.4764
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##
##  Hillshade_9am    Hillshade_Noon   Hillshade_3pm
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.7795   1st Qu.:0.7469   1st Qu.:0.4685
##  Median :0.8543   Median :0.8272   Median :0.5630
##  Mean   :0.8339   Mean   :0.8098   Mean   :0.5620
##  3rd Qu.:0.9094   3rd Qu.:0.8951   3rd Qu.:0.6654
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##
##  Horizontal_Distance_To_Fire_Points Wilderness_Area1 Wilderness_Area2
##  Min.   :0.0000                     Min.   :0.0000   Min.   :0.00000
##  1st Qu.:0.1487                     1st Qu.:0.0000   1st Qu.:0.00000
##  Median :0.2417                     Median :0.0000   Median :0.00000
##  Mean   :0.2816                     Mean   :0.4507   Mean   :0.05299
##  3rd Qu.:0.3613                     3rd Qu.:1.0000   3rd Qu.:0.00000
##  Max.   :1.0000                     Max.   :1.0000   Max.   :1.00000
##
##  Wilderness_Area3 Wilderness_Area4    Soil_Type1         Soil_Type2
##  Min.   :0.0000   Min.   :0.00000   Min.   :0.000000   Min.   :0.00000
##  1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.000000   1st Qu.:0.00000
##  Median :0.0000   Median :0.00000   Median :0.000000   Median :0.00000
##  Mean   :0.4323   Mean   :0.06401   Mean   :0.005334   Mean   :0.01273
##  3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:0.000000   3rd Qu.:0.00000
##  Max.   :1.0000   Max.   :1.00000   Max.   :1.000000   Max.   :1.00000
##
##    Soil_Type3         Soil_Type4       Soil_Type5         Soil_Type6
##  Min.   :0.000000   Min.   :0.00000   Min.   :0.000000   Min.   :0.0000
##  1st Qu.:0.000000   1st Qu.:0.00000   1st Qu.:0.000000   1st Qu.:0.0000
##  Median :0.000000   Median :0.00000   Median :0.000000   Median :0.0000
##  Mean   :0.009377   Mean   :0.01953   Mean   :0.002667   Mean   :0.0105
##  3rd Qu.:0.000000   3rd Qu.:0.00000   3rd Qu.:0.000000   3rd Qu.:0.0000
##  Max.   :1.000000   Max.   :1.00000   Max.   :1.000000   Max.   :1.0000
##
##    Soil_Type7          Soil_Type8          Soil_Type9
##  Min.   :0.0000000   Min.   :0.0000000   Min.   :0.000000
##  1st Qu.:0.0000000   1st Qu.:0.0000000   1st Qu.:0.000000
##  Median :0.0000000   Median :0.0000000   Median :0.000000
##  Mean   :0.0001721   Mean   :0.0006022   Mean   :0.002237
##  3rd Qu.:0.0000000   3rd Qu.:0.0000000   3rd Qu.:0.000000
##  Max.   :1.0000000   Max.   :1.0000000   Max.   :1.000000
##
##   Soil_Type10       Soil_Type11       Soil_Type12       Soil_Type13
##  Min.   :0.00000   Min.   :0.00000   Min.   :0.00000   Min.   :0.00000
##  1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.00000
##  Median :0.00000   Median :0.00000   Median :0.00000   Median :0.00000
##  Mean   :0.05859   Mean   :0.02047   Mean   :0.05274   Mean   :0.02925
##  3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.00000
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.00000   Max.   :1.00000
##
##   Soil_Type14        Soil_Type16        Soil_Type17
##  Min.   :0.000000   Min.   :0.000000   Min.   :0.000000
##  1st Qu.:0.000000   1st Qu.:0.000000   1st Qu.:0.000000
##  Median :0.000000   Median :0.000000   Median :0.000000
##  Mean   :0.001032   Mean   :0.005162   Mean   :0.006452
##  3rd Qu.:0.000000   3rd Qu.:0.000000   3rd Qu.:0.000000
##  Max.   :1.000000   Max.   :1.000000   Max.   :1.000000
##
##   Soil_Type18        Soil_Type19        Soil_Type20
##  Min.   :0.000000   Min.   :0.000000   Min.   :0.00000
##  1st Qu.:0.000000   1st Qu.:0.000000   1st Qu.:0.00000
##  Median :0.000000   Median :0.000000   Median :0.00000
##  Mean   :0.003011   Mean   :0.007657   Mean   :0.01342
##  3rd Qu.:0.000000   3rd Qu.:0.000000   3rd Qu.:0.00000
##  Max.   :1.000000   Max.   :1.000000   Max.   :1.00000
##
##   Soil_Type21         Soil_Type22       Soil_Type23       Soil_Type24
```

```
##  Min.   :0.0000000   Min.   :0.00000   Min.   :0.00000   Min.   :0.00000  
##  1st Qu.:0.0000000   1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.00000  
##  Median :0.0000000   Median :0.00000   Median :0.00000   Median :0.00000  
##  Mean   :0.0007743   Mean   :0.05721   Mean   :0.09678   Mean   :0.03639  
##  3rd Qu.:0.0000000   3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.00000  
##  Max.   :1.0000000   Max.   :1.00000   Max.   :1.00000   Max.   :1.00000  
##                                                                            
##   Soil_Type25          Soil_Type26         Soil_Type27      
##  Min.   :0.0000000   Min.   :0.000000   Min.   :0.000000  
##  1st Qu.:0.0000000   1st Qu.:0.000000   1st Qu.:0.000000  
##  Median :0.0000000   Median :0.000000   Median :0.000000  
##  Mean   :0.0009463   Mean   :0.004474   Mean   :0.001463  
##  3rd Qu.:0.0000000   3rd Qu.:0.000000   3rd Qu.:0.000000  
##  Max.   :1.0000000   Max.   :1.000000   Max.   :1.000000  
##                                                             
##   Soil_Type28         Soil_Type29     Soil_Type30       Soil_Type31     
##  Min.   :0.000000   Min.   :0.000   Min.   :0.00000   Min.   :0.00000  
##  1st Qu.:0.000000   1st Qu.:0.000   1st Qu.:0.00000   1st Qu.:0.00000  
##  Median :0.000000   Median :0.000   Median :0.00000   Median :0.00000  
##  Mean   :0.001635   Mean   :0.201   Mean   :0.05299   Mean   :0.04301  
##  3rd Qu.:0.000000   3rd Qu.:0.000   3rd Qu.:0.00000   3rd Qu.:0.00000  
##  Max.   :1.000000   Max.   :1.000   Max.   :1.00000   Max.   :1.00000  
##                                                                        
##   Soil_Type32       Soil_Type33       Soil_Type34        Soil_Type35       
##  Min.   :0.00000   Min.   :0.00000   Min.   :0.000000   Min.   :0.000000  
##  1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.000000   1st Qu.:0.000000  
##  Median :0.00000   Median :0.00000   Median :0.000000   Median :0.000000  
##  Mean   :0.09351   Mean   :0.07605   Mean   :0.003011   Mean   :0.002065  
##  3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.000000   3rd Qu.:0.000000  
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.000000   Max.   :1.000000  
##                                                                           
##   Soil_Type36          Soil_Type37          Soil_Type38     
##  Min.   :0.0000000   Min.   :0.0000000   Min.   :0.00000  
##  1st Qu.:0.0000000   1st Qu.:0.0000000   1st Qu.:0.00000  
##  Median :0.0000000   Median :0.0000000   Median :0.00000  
##  Mean   :0.0002581   Mean   :0.0009463   Mean   :0.02323  
##  3rd Qu.:0.0000000   3rd Qu.:0.0000000   3rd Qu.:0.00000  
##  Max.   :1.0000000   Max.   :1.0000000   Max.   :1.00000  
##                                                            
##   Soil_Type39       Soil_Type40     
##  Min.   :0.00000   Min.   :0.00000  
##  1st Qu.:0.00000   1st Qu.:0.00000  
##  Median :0.00000   Median :0.00000  
##  Mean   :0.02667   Mean   :0.01669  
##  3rd Qu.:0.00000   3rd Qu.:0.00000  
##  Max.   :1.00000   Max.   :1.00000  
## 
```

```r
smpl <- select(sampleSet,
                 Cover_Type,
                 Elevation,
                 Horizontal_Distance_To_Roadways,
                 Horizontal_Distance_To_Fire_Points,
                 Dist_water,
                 Hillshade_Noon,
                 Hillshade_3pm,
                 Hillshade_9am,
                 Aspect,
                 Slope,
                 Wilderness_Area4,
                 Wilderness_Area3,
                 Wilderness_Area1,
                 Wilderness_Area2,
                 Soil_Type2,
                 Soil_Type3,
                 Soil_Type4,
                 Soil_Type6,
                 Soil_Type10,
                 Soil_Type11,
                 Soil_Type12,
                 Soil_Type13,
                 Soil_Type16,
                 Soil_Type17,
                 Soil_Type19,
                 Soil_Type20,
                 Soil_Type22,
                 Soil_Type23,
                 Soil_Type24,
                 Soil_Type29,
                 Soil_Type30,
                 Soil_Type31,
                 Soil_Type32,
                 Soil_Type33,
                 Soil_Type34,
                 Soil_Type36,
                 Soil_Type37,
                 Soil_Type38,
                 Soil_Type39,
                 Soil_Type40
)
```

```r
#Create smaller data split so we can run these calcs locally
inTrains <- createDataPartition(y=smpl$Cover_Type, p=0.70, list = FALSE)
set.seed(78)
training <- smpl[ inTrains,]
testing <- smpl[-inTrains,]
nrow(training)  # 8139 70%
```

```
## [1] 8139
```

```r
nrow(testing) #3485 30%
```

```
## [1] 3485
```

```r
####Ensemble with Super Learner, tells which will be the best model to use with weighted averages
library(SuperLearner) #caret, glmnet, randomForest, and xgboost have need pre-installed and needed for SuperLea
rner.
```

```
## Loading required package: nnls
```

```
## Super Learner
```

```
## Version: 2.0-24
```

```
## Package created on 2018-08-10
```

```
outcome <- smpl$Cover_Type #Move Response Variable
data <- subset(smpl, select = -Cover_Type) #Make a dataframe with predictors only
str(data) #confirm Structure
```

```
## 'data.frame':    11624 obs. of  39 variables:
##  $ Elevation                         : num  2489 2745 3073 2847 2874 ...
##  $ Horizontal_Distance_To_Roadways   : num  840 2416 6836 4983 2869 ...
##  $ Horizontal_Distance_To_Fire_Points: num  5254 6428 2735 4727 3653 ...
##  $ Dist_water                        : num  175.5 71.2 108 93.5 400.5 ...
##  $ Hillshade_Noon                    : num  232 234 246 190 236 201 234 218 237 219 ...
##  $ Hillshade_3pm                     : num  153 184 149 199 157 111 119 123 147 157 ...
##  $ Hillshade_9am                     : num  216 190 227 120 216 220 241 227 225 200 ...
##  $ Aspect                            : num  11 306 173 320 352 41 136 56 129 353 ...
##  $ Slope                             : num  4 11 12 33 2 17 14 11 3 12 ...
##  $ Wilderness_Area4                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Wilderness_Area3                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Wilderness_Area1                  : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ Wilderness_Area2                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type2                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type3                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type4                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type6                        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type10                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type11                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type12                       : num  0 0 0 0 0 1 0 0 0 0 ...
##  $ Soil_Type13                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type16                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type17                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type19                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type20                       : num  0 0 0 0 0 0 0 0 0 1 ...
##  $ Soil_Type22                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type23                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type24                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type29                       : num  0 1 1 1 1 0 0 1 1 0 ...
##  $ Soil_Type30                       : num  0 0 0 0 0 0 1 0 0 0 ...
##  $ Soil_Type31                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type32                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type33                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type34                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type36                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type37                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type38                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type39                       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Soil_Type40                       : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
dim(data) #Look at the size of data
```

```
## [1] 11624    39
```

```
set.seed(101)
train_obs <- sample(nrow(data), 150)#reduce size to help with calculation speed.
X_train <- data[train_obs, ] #training sample
X_hold <- data[-train_obs, ] #evaluate perfomance of the model
outcome_bin <- as.numeric(outcome) #outcomes binary
Y_train <- outcome_bin[train_obs]
Y_hold <- outcome_bin[-train_obs]
table(Y_train) #distribution
```

```
## Y_train
##  1  2  3  4  5  6  7
## 48 75 10  1  4  4  8
```

```
#
listWrappers() #Shows the prediction algorithms available.
```

```
## All prediction algorithm wrappers in SuperLearner:
```

```
##  [1] "SL.bartMachine"      "SL.bayesglm"         "SL.biglasso"        
##  [4] "SL.caret"            "SL.caret.rpart"      "SL.cforest"         
##  [7] "SL.dbarts"           "SL.earth"            "SL.extraTrees"      
## [10] "SL.gam"              "SL.gbm"              "SL.glm"             
## [13] "SL.glm.interaction"  "SL.glmnet"           "SL.ipredbagg"       
## [16] "SL.kernelKnn"        "SL.knn"              "SL.ksvm"            
## [19] "SL.lda"              "SL.leekasso"         "SL.lm"              
## [22] "SL.loess"            "SL.logreg"           "SL.mean"            
## [25] "SL.nnet"             "SL.nnls"             "SL.polymars"        
## [28] "SL.qda"              "SL.randomForest"     "SL.ranger"          
## [31] "SL.ridge"            "SL.rpart"            "SL.rpartPrune"      
## [34] "SL.speedglm"         "SL.speedlm"          "SL.step"            
## [37] "SL.step.forward"     "SL.step.interaction" "SL.stepAIC"         
## [40] "SL.svm"              "SL.template"         "SL.xgboost"
```

```
## 
## All screening algorithm wrappers in SuperLearner:
```

```
## [1] "All"
## [1] "screen.corP"            "screen.corRank"         "screen.glmnet"         
## [4] "screen.randomForest"    "screen.SIS"             "screen.template"       
## [7] "screen.ttest"           "write.screen.template"
```

```
sl_model <- c("SL.mean", #mean is lowest benchmark
              "SL.glmnet",
              "SL.randomForest",
              "SL.ranger",
              "SL.xgboost",
              "SL.nnet",
              "SL.ksvm",
              "SL.svm",
              "SL.nnls"
)
```

```
system.time({ #processing time for Super Learner
sl <- SuperLearner(Y=Y_train, X=X_train, family = gaussian(),
                   SL.library = sl_model)
})
```

```
## Loading required package: glmnet
```

```
## Loading required package: Matrix
```

```
## 
## Attaching package: 'Matrix'
```

```
## The following object is masked from 'package:tidyr':
## 
##     expand
```

```
## Loaded glmnet 2.0-16
```

```
## 
## Attaching package: 'glmnet'
```

```
## The following object is masked from 'package:pROC':
## 
##     auc
```

```
## Loading required package: nnet
```

```
## Loading required package: e1071
```

```
## Loading required package: ranger
```

```
##
## Attaching package: 'ranger'
```

```
## The following object is masked from 'package:randomForest':
##
##     importance
```

```
## Loading required package: xgboost
```

```
##
## Attaching package: 'xgboost'
```

```
## The following object is masked from 'package:dplyr':
##
##     slice
```

```
## Loading required package: kernlab
```

```
##
## Attaching package: 'kernlab'
```

```
## The following object is masked from 'package:purrr':
##
##     cross
```

```
## The following object is masked from 'package:ggplot2':
##
##     alpha
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type36' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
##    user  system elapsed
##   10.86    1.71   13.13
```

```
sl #Coef at zero means the model was not used at all
```

```
##
## Call:
## SuperLearner(Y = Y_train, X = X_train, family = gaussian(), SL.library = sl_model)
##
##
##
##                          Risk       Coef
## SL.mean_All          2.396137 0.00000000
## SL.glmnet_All        2.533589 0.00000000
## SL.randomForest_All 2.080519 0.13798896
## SL.ranger_All        2.051609 0.00000000
## SL.xgboost_All       2.975871 0.00000000
## SL.nnet_All          2.365376 0.00602686
## SL.ksvm_All          2.330691 0.08772647
## SL.svm_All           2.539382 0.00000000
## SL.nnls_All          6.953554 0.76825770
```

```
#randomforest had warning that the response has 5 or less unique values, due to the low sample rate and low pro
portions on some Cover_Types
```

```
set.seed(10)
system.time({
r <- CV.SuperLearner(Y=Y_train, X=X_train, family = gaussian(), V=10,
                     SL.library = sl_model)
})
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type36' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type17' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type17' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type36' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' and 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type19' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type3' and 'Soil_Type16' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type2' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type17' and 'Soil_Type34' and
## 'Soil_Type36' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type11' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type38' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type17' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type11' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type20' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' and 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type17' and 'Soil_Type34' and
## 'Soil_Type36' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type36' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type36' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type17' and 'Soil_Type34' and
## 'Soil_Type37' and 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type36' and
## 'Soil_Type37' and 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' and 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' and 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type20' and 'Soil_Type34' and
## 'Soil_Type37' and 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37' and
## 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type17' and 'Soil_Type34' and
## 'Soil_Type36' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type34' and 'Soil_Type37'
## constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type17' and 'Soil_Type19' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type36' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type19' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' and 'Soil_Type40' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type16' and 'Soil_Type19' and 'Soil_Type34' and
## 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type36' and 'Soil_Type37' constant. Cannot scale
## data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type19' and 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale
## data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type20' and 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale
## data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type37' and 'Soil_Type40' constant. Cannot scale
## data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
```

```
## Warning in svm.default(y = Y, x = X, nu = nu, type = type.reg, fitted =
## FALSE, : Variable(s) 'Soil_Type3' and 'Soil_Type16' and 'Soil_Type17' and
## 'Soil_Type34' and 'Soil_Type37' constant. Cannot scale data.
```

```
##    user  system elapsed
##   92.51   18.39  111.11
```

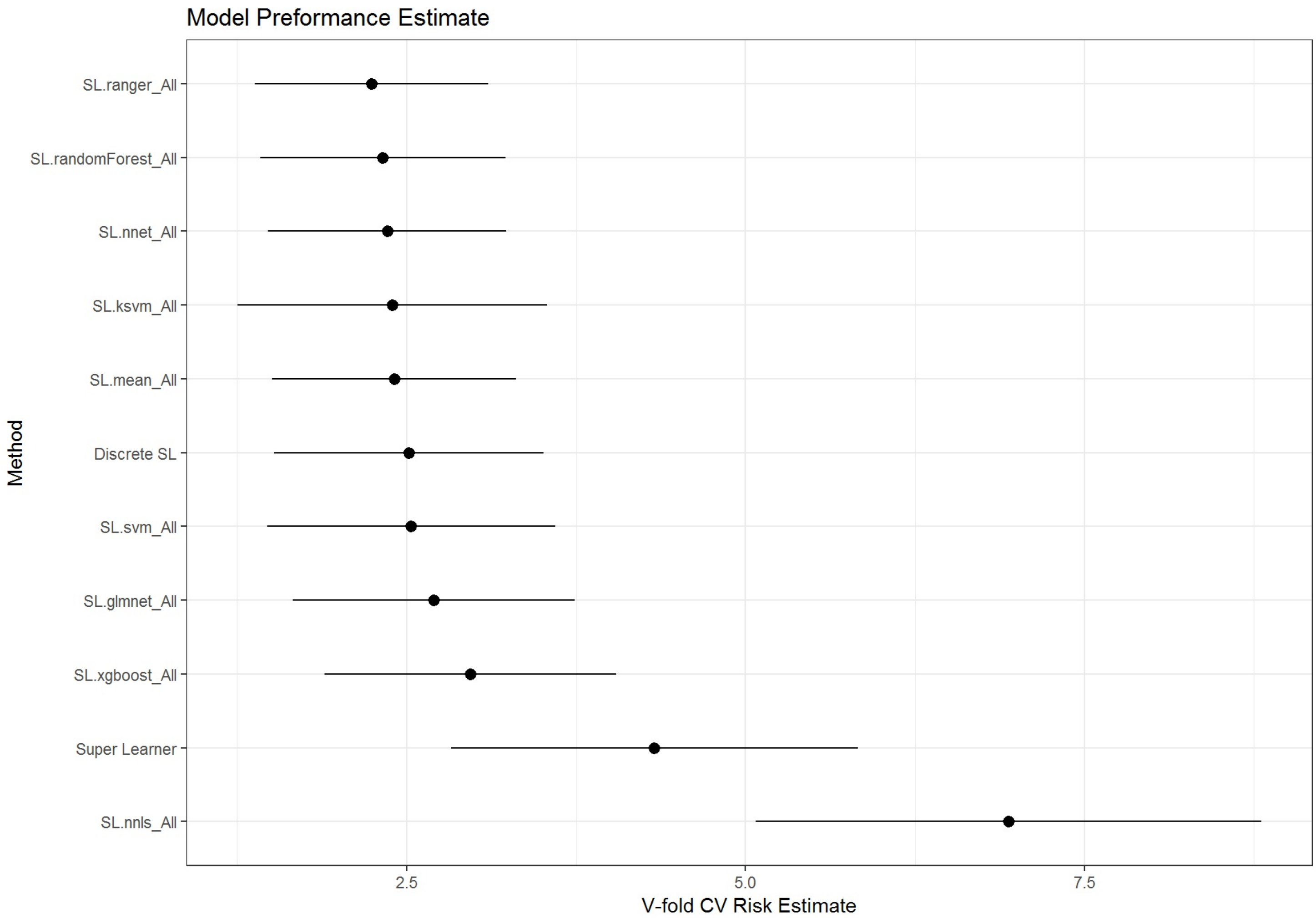
```
summary(r)
```

```
##
## Call:
## CV.SuperLearner(Y = Y_train, X = X_train, V = 10, family = gaussian(),
##     SL.library = sl_model)
##
## Risk is based on: Mean Squared Error
##
## All risk estimates are based on V =  10
##
##            Algorithm    Ave      se     Min     Max
##        Super Learner 4.3268 0.76592 0.64083  9.3838
##          Discrete SL 2.5176 0.50584 0.68260  6.3375
##          SL.mean_All 2.4078 0.45833 0.48000  4.9092
##        SL.glmnet_All 2.7012 0.53079 0.62881  5.2705
##  SL.randomForest_All 2.3258 0.46180 0.81687  6.3375
##        SL.ranger_All 2.2414 0.43988 0.68260  5.6609
##       SL.xgboost_All 2.9720 0.54812 1.28884  5.7196
##          SL.nnet_All 2.3569 0.44720 0.41824  4.9092
##          SL.ksvm_All 2.3938 0.58253 0.25161  5.4163
##           SL.svm_All 2.5346 0.54101 0.20505  5.5143
##          SL.nnls_All 6.9379 0.95142 2.88773 12.0341
```

```
table(simplify2array(r$whichDiscreteSL))
```

```
##
##         SL.ksvm_All SL.randomForest_All       SL.ranger_All
##                   1                   2                   7
```

```
plot(r) +
theme_bw() + ggtitle("Model Preformance Estimate")#plots CV
```




```
pred <- predict(sl, X_hold, onlySL = T)
head(pred$pred)
```

```
##           [,1]
## [1,] 0.5586227
## [2,] 0.5131341
## [3,] 0.3931873
## [4,] 0.6204339
## [5,] 0.4069709
## [6,] 0.4955275
```

```
str(pred)
```

```
## List of 2
##  $ pred           : num [1:11474, 1] 0.559 0.513 0.393 0.62 0.407 ...
##  $ library.predict: num [1:11474, 1:9] 0 0 0 0 0 0 0 0 0 0 ...
```

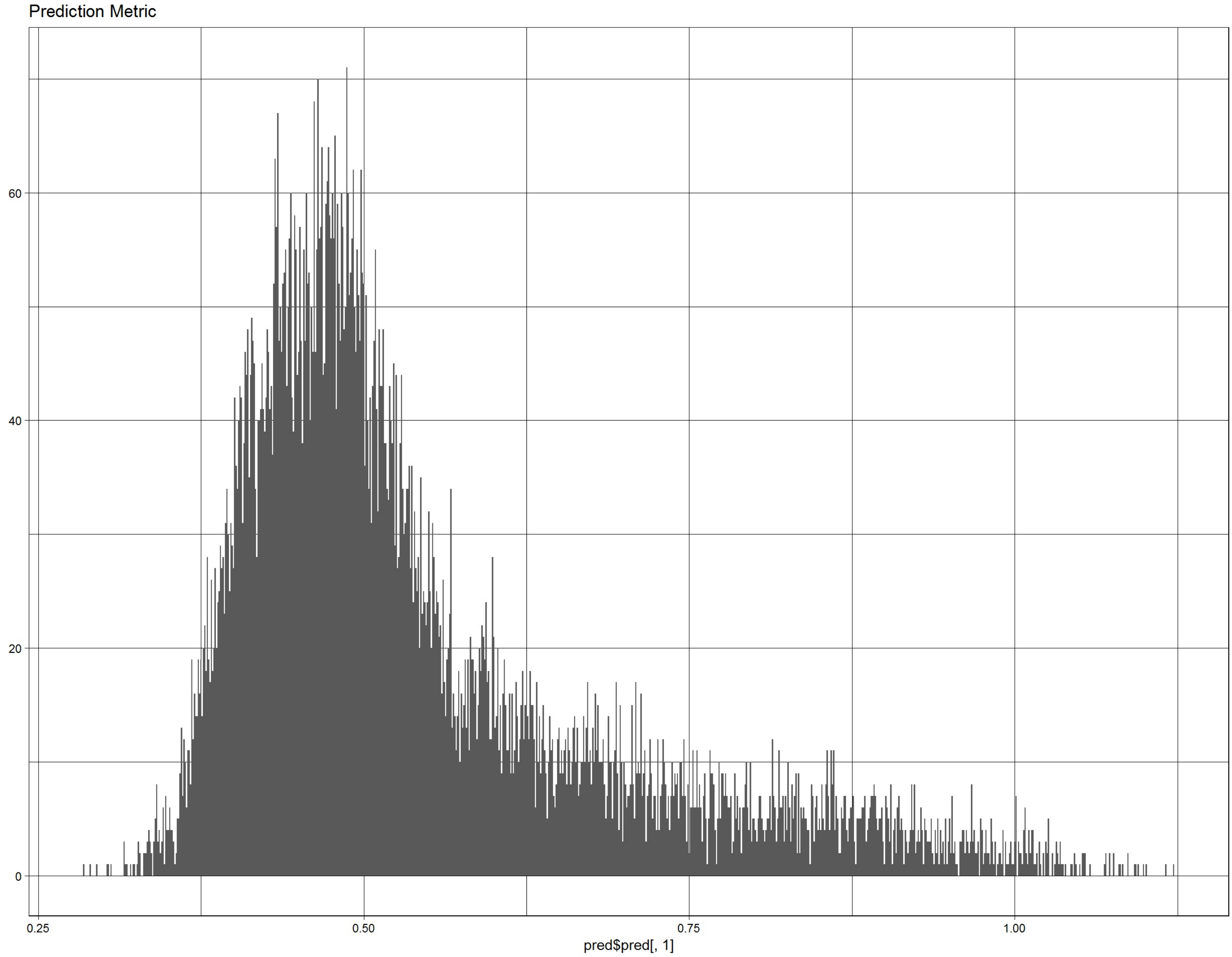
```
head(pred$library.predict)
```

```
##      [,1] [,2]     [,3] [,4] [,5]     [,6]     [,7] [,8]       [,9]
## [1,]    0    0 2.572230    0    0 1.750000 1.999888    0 0.02303025
## [2,]    0    0 2.075217    0    0 1.750000 1.960696    0 0.05756550
## [3,]    0    0 1.496138    0    0 2.226027 1.437022    0 0.06151050
## [4,]    0    0 2.829967    0    0 2.226027 2.029291    0 0.05010200
## [5,]    0    0 1.834617    0    0 2.226027 1.072005    0 0.06033766
## [6,]    0    0 2.013933    0    0 2.226027 1.779931    0 0.06256227
```

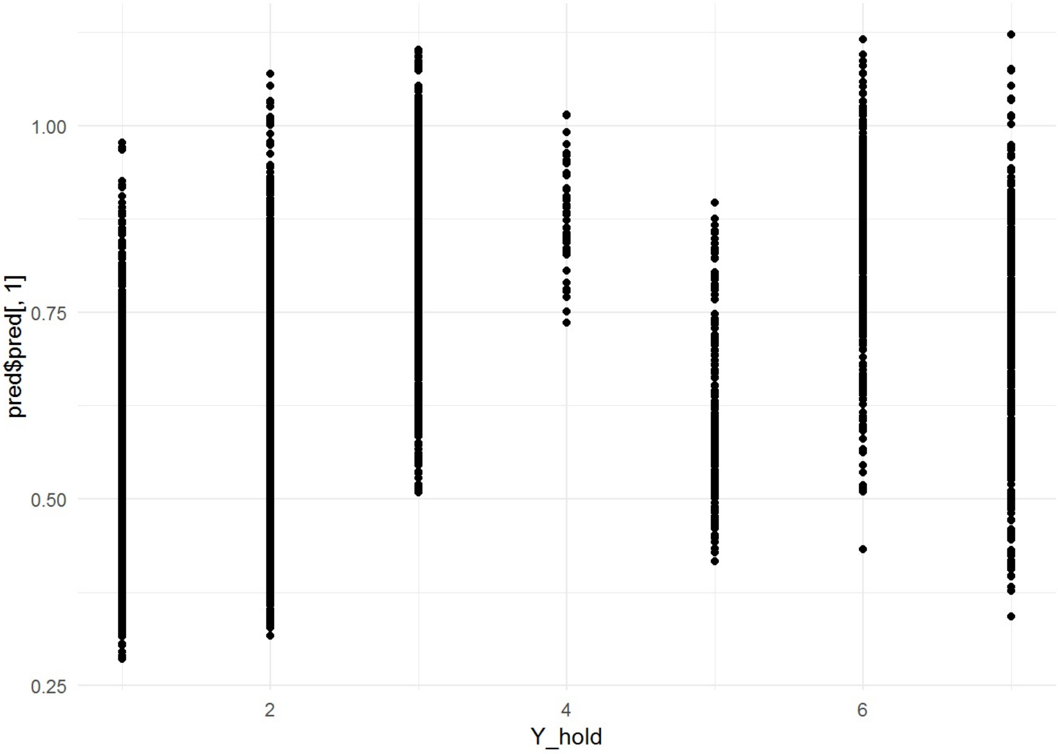
```
#Each one is the SL learner 1 through 9
summary(pred$library.predict)
```

```
##        V1          V2          V3              V4          V5
##  Min.   :0   Min.   :0   Min.   :1.216   Min.   :0   Min.   :0
##  1st Qu.:0   1st Qu.:0   1st Qu.:1.715   1st Qu.:0   1st Qu.:0
##  Median :0   Median :0   Median :1.979   Median :0   Median :0
##  Mean   :0   Mean   :0   Mean   :2.239   Mean   :0   Mean   :0
##  3rd Qu.:0   3rd Qu.:0   3rd Qu.:2.543   3rd Qu.:0   3rd Qu.:0
##  Max.   :0   Max.   :0   Max.   :5.458   Max.   :0   Max.   :0
##        V6              V7                V8          V9
##  Min.   :1.750   Min.   :0.7701   Min.   :0   Min.   :0.01589
##  1st Qu.:2.226   1st Qu.:1.6739   1st Qu.:0   1st Qu.:0.05522
##  Median :2.226   Median :1.8669   Median :0   Median :0.06138
##  Mean   :2.217   Mean   :1.9185   Mean   :0   Mean   :0.06909
##  3rd Qu.:2.226   3rd Qu.:2.0414   3rd Qu.:0   3rd Qu.:0.07861
##  Max.   :2.226   Max.   :3.4669   Max.   :0   Max.   :0.22052
```

```
#ggplot
#Histogram plot
qplot(pred$pred[, 1], binwidth=0.001) +
  theme_linedraw()  + ggtitle("Prediction Metric")
```




```
#Scatterplot
qplot(Y_hold, pred$pred[, 1]) + theme_minimal()
```




```
###Tune the Random Forest Model
SL.randomForest.tune <- function(...,mtry=30,ntree=1000,nodesize=3){
  SL.randomForest(...,mtry=mtry,ntree=ntree,nodesize=nodesize)
}
tuneGrid <-  expand.grid(mtry = c(3,5), ntree=c(500,1000), nodesize=c(1,3))
for(i in seq(nrow(tuneGrid))) {
  eval(parse(text = paste0("SL.randomForest",tuneGrid[i,1],"_nt",tuneGrid[i,2],"_ns",tuneGrid[i,3],
                           "<- function(..., mtry = ", tuneGrid[i, 1], ", ntree = ", tuneGrid[i, 2],
                           ", nodesize = ", tuneGrid[i,3],") { SL.randomForest(..., mtry = mtry,
                           ntree = ntree, nodesize=nodesize)}")))
}
```

```
SL.randomForest.tune
```

```
## function(...,mtry=30,ntree=1000,nodesize=3){
##   SL.randomForest(...,mtry=mtry,ntree=ntree,nodesize=nodesize)
## }
```

```
allObjects <- ls()
myRfObjects <- grep("SL.randomForest",allObjects)
allRf <- allObjects[myRfObjects]
allRf
```

```
## [1] "SL.randomForest.tune"        "SL.randomForest3_nt1000_ns1"
## [3] "SL.randomForest3_nt1000_ns3" "SL.randomForest3_nt500_ns1"
## [5] "SL.randomForest3_nt500_ns3"  "SL.randomForest5_nt1000_ns1"
## [7] "SL.randomForest5_nt1000_ns3" "SL.randomForest5_nt500_ns1"
## [9] "SL.randomForest5_nt500_ns3"
```

```
rf.sl <- SuperLearner(
  Y = Y_train,
  X = X_train,
  family = gaussian(),
  method="method.NNLS",
  SL.library = allRf
)
```

```
rf.sl
```

```
##
## Call:
## SuperLearner(Y = Y_train, X = X_train, family = gaussian(), SL.library = allRf,
##     method = "method.NNLS")
##
##
##                                      Risk      Coef
## SL.randomForest.tune_All        2.189556 0.2813379
## SL.randomForest3_nt1000_ns1_All 2.097515 0.0000000
## SL.randomForest3_nt1000_ns3_All 2.112036 0.0000000
## SL.randomForest3_nt500_ns1_All  2.087561 0.4405424
## SL.randomForest3_nt500_ns3_All  2.088577 0.2781197
## SL.randomForest5_nt1000_ns1_All 2.097743 0.0000000
## SL.randomForest5_nt1000_ns3_All 2.097063 0.0000000
## SL.randomForest5_nt500_ns1_All  2.087214 0.0000000
## SL.randomForest5_nt500_ns3_All  2.097057 0.0000000
```

```
############Random Forest in caret
library(e1071)
ctrl3 <- trainControl(method='repeatedcv',
                        number=10,
                        repeats=3)
metric <- "Accuracy"
set.seed(200)
mtry <- sqrt(ncol(training))
tunegrid <- expand.grid(.mtry=mtry)
rf_ <- train(Cover_Type~.,
                    data=training,
                    method='rf',
                  # preProcess =c("center", "scale"),#add this after first run @0.777/0.6333
                  # preProcess = ("range"),#after range 0.776/0.631
                    metric='Accuracy', #after center and scale 0.777/0.631
                    tuneGrid=tunegrid, #reran and got 0.779/0.636 didnt set seed mtry=6.32 for all cases
                    trControl=ctrl3)
((rf_))
```

```
## Random Forest
##
## 8139 samples
##   39 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 7326, 7326, 7324, 7324, 7326, 7324, ...
## Resampling results:
##
##   Accuracy   Kappa
##   0.7737689  0.6280133
##
## Tuning parameter 'mtry' was held constant at a value of 6.324555
```

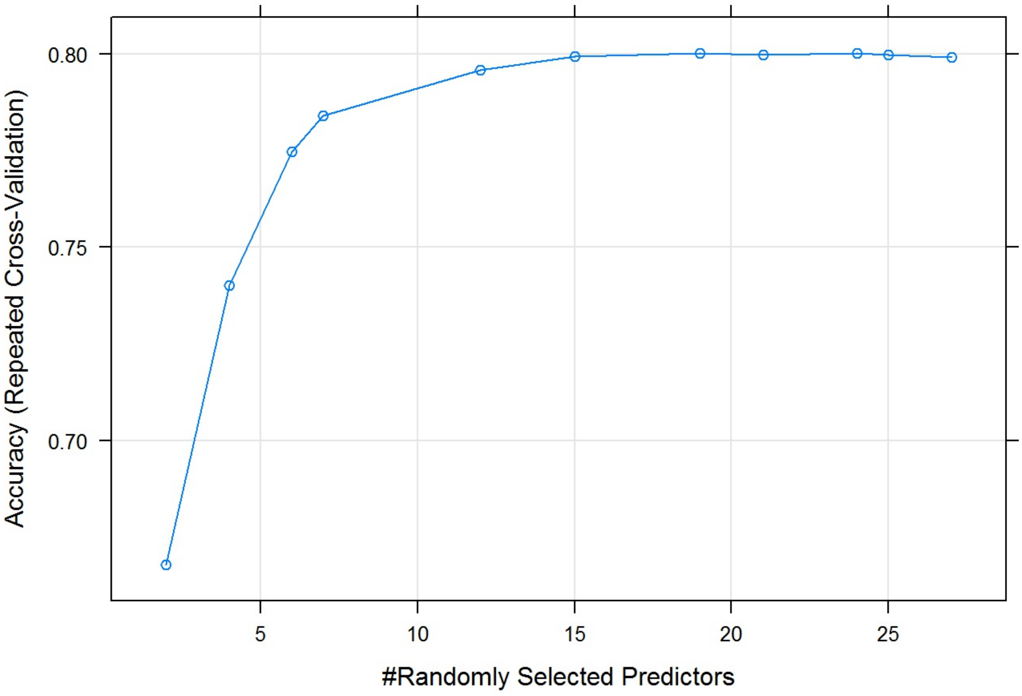
```
#trying centering and scaling the data in preprocess
rf_ <- train(Cover_Type~.,
             data = train,
             method = "rf",
             preProcess = c("center", "scale"),
             trControl = ctrl3)
((rf_))
```

```
## Random Forest
##
## 8139 samples
##   52 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## Pre-processing: centered (52), scaled (52)
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 7324, 7325, 7325, 7324, 7323, 7324, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    2    0.6409506  0.3661029
##   27    0.8064494  0.6837369
##   52    0.8032139  0.6786936
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 27.
```

```
#random variables
mtry <- sqrt(ncol(training))
ntree <- 3
control <- trainControl(method='repeatedcv',
                        number=10,
                        repeats=3,
                        search = 'random')
set.seed(201)
rf_random <- train(Cover_Type~.,
                   data = training,
                   method = 'rf',
                   metric = 'Accuracy',
                   tuneLength  = 15,
                   trControl = control)
((rf_random))#21 mtry @0.804/0.680
```

```
## Random Forest
##
## 8139 samples
##   39 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 7324, 7323, 7324, 7326, 7325, 7326, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    2    0.6679788  0.4246290
##    4    0.7401403  0.5689606
##    6    0.7747500  0.6296615
##    7    0.7839653  0.6456068
##   12    0.7957626  0.6657538
##   15    0.7992007  0.6714793
##   19    0.8001011  0.6729075
##   21    0.7997332  0.6722684
##   24    0.8001841  0.6730778
##   25    0.7996911  0.6723089
##   27    0.7990777  0.6712961
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 24.
```

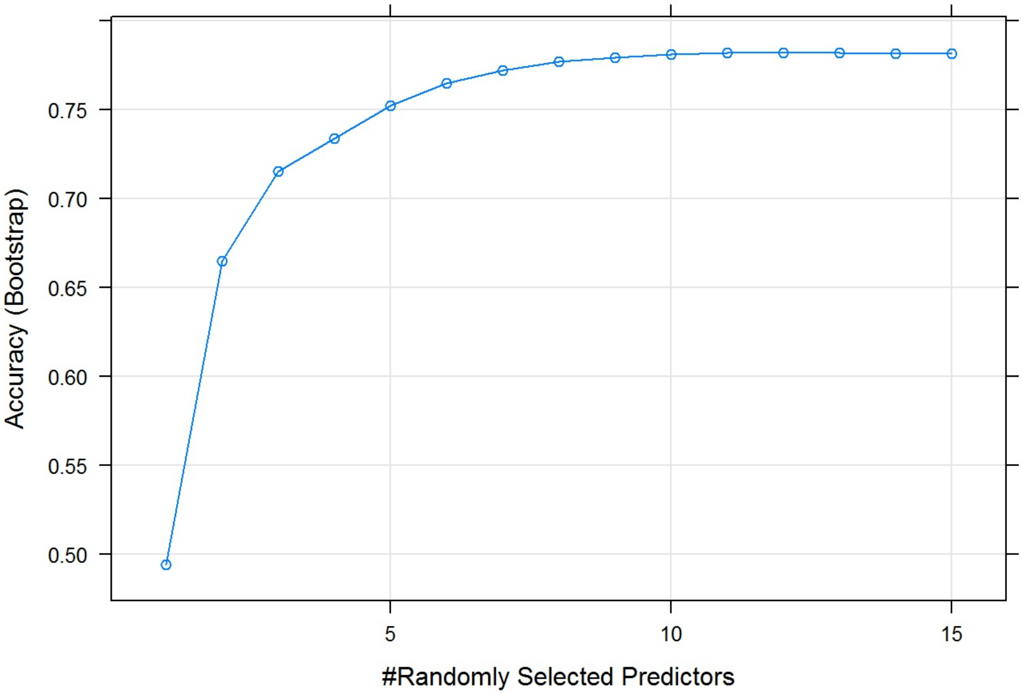
```
plot(rf_random)
```




```
set.seed(345)
control <- trainControl(method='repeatedcv',
                        number=10,
                        repeats=3,
                        search='grid')
tunegrid <- expand.grid(.mtry = (1:15))
rf_gridsearch <- train(Cover_Type ~ .,
                       data = training,
                       method = 'rf',
                       metric = 'Accuracy',
                       tuneGrid = tunegrid)
((rf_gridsearch))#mtry15 @0.787/0.652
```

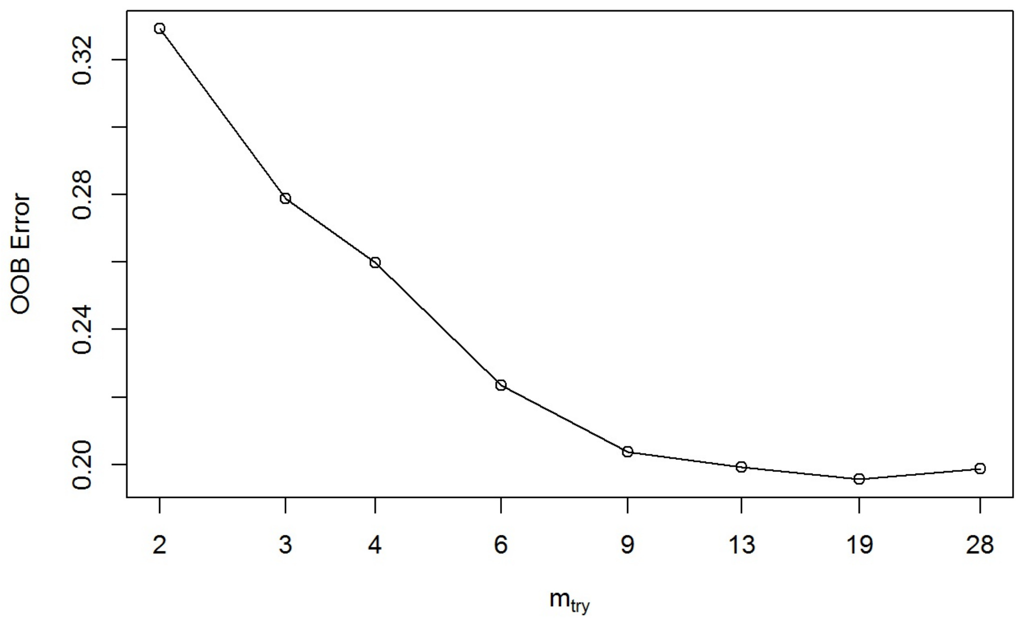
```
## Random Forest
##
## 8139 samples
##   39 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 8139, 8139, 8139, 8139, 8139, 8139, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    1    0.4939775  0.01722283
##    2    0.6647669  0.41850387
##    3    0.7154097  0.52259730
##    4    0.7338013  0.55785818
##    5    0.7520144  0.59001302
##    6    0.7649686  0.61286228
##    7    0.7720400  0.62529360
##    8    0.7769481  0.63379396
##    9    0.7790620  0.63748347
##   10    0.7812041  0.64130265
##   11    0.7820497  0.64285315
##   12    0.7821094  0.64297758
##   13    0.7819070  0.64274894
##   14    0.7816187  0.64238548
##   15    0.7816989  0.64249039
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 12.
```

```
plot(rf_gridsearch)
```




```
set.seed(16)
bestMtry <- tuneRF(training[,-1],training[,1],
                   mtryStart= 2,
                   stepFactor = 1.5,
                   improve = 1e-5,
                   ntree = 500)
```

```
## mtry = 2  OOB error = 32.92%
## Searching left ...
## Searching right ...
## mtry = 3     OOB error = 27.88%
## 0.1530422 1e-05
## mtry = 4     OOB error = 25.97%
## 0.06831203 1e-05
## mtry = 6     OOB error = 22.35%
## 0.1395459 1e-05
## mtry = 9     OOB error = 20.38%
## 0.08796042 1e-05
## mtry = 13    OOB error = 19.93%
## 0.02230259 1e-05
## mtry = 19    OOB error = 19.56%
## 0.01849568 1e-05
## mtry = 28    OOB error = 19.87%
## -0.01570352 1e-05
```




```
((bestMtry))#best oob error is 19
```

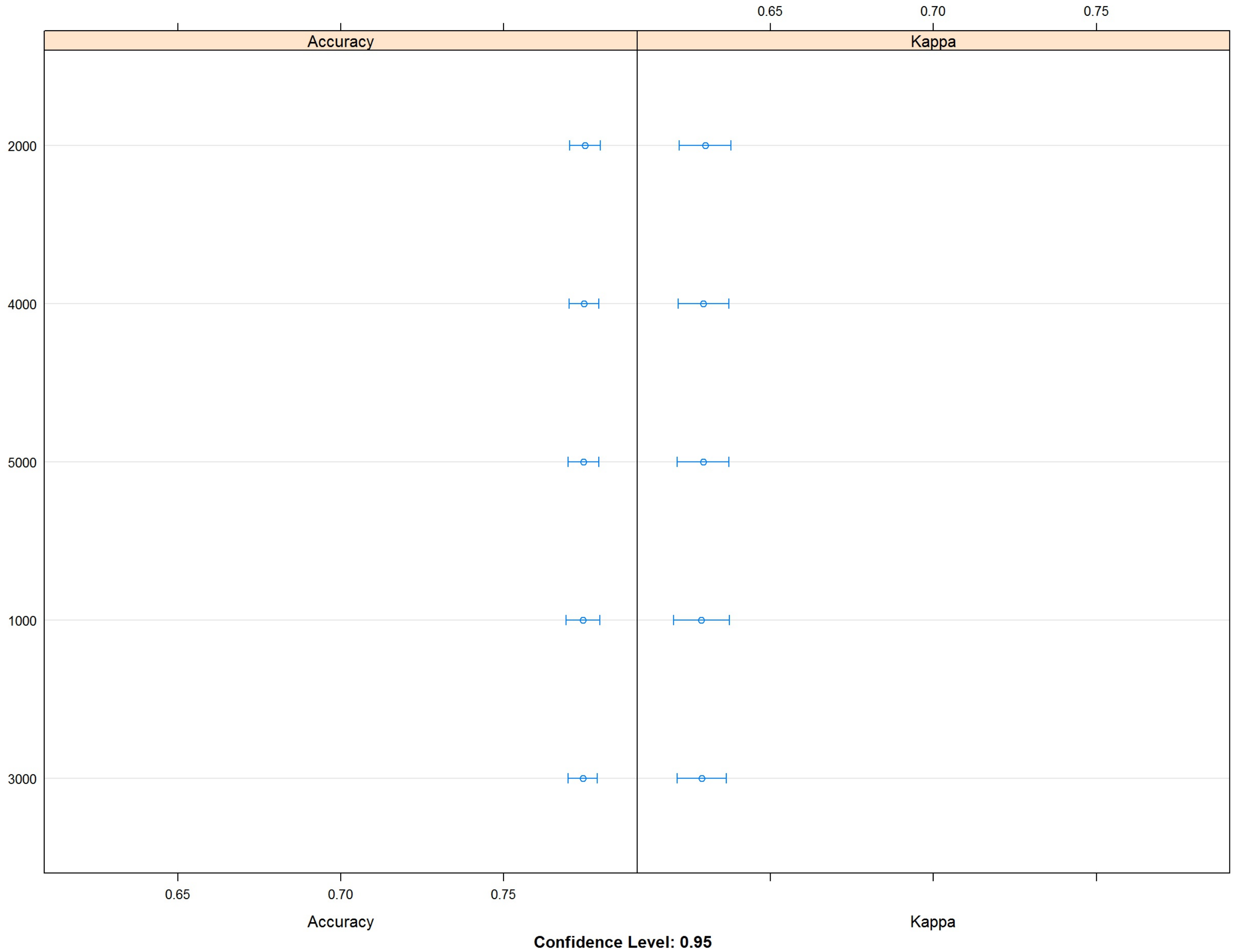
```
##          mtry   OOBError
## 2.OOB      2 0.3291559
## 3.OOB      3 0.2787812
## 4.OOB      4 0.2597371
## 6.OOB      6 0.2234918
## 9.OOB      9 0.2038334
## 13.OOB    13 0.1992874
## 19.OOB    19 0.1956014
## 28.OOB    28 0.1986731
```

```
set.seed(17)
control <- trainControl(method = 'repeatedcv',
                        number = 10,
                        repeats = 3,
                        search = 'grid')
tunegrid <- expand.grid(.mtry = c(sqrt(ncol(training))))
modellist <- list()
```

```
for (ntree in c(1000,2000,3000,4000,5000)){
  set.seed(17)
  fit <- train(Cover_Type~.,
               data = training,
               method = 'rf',
               metric = 'Accuracy',
               tuneGrid = tunegrid,
               trControl = control,
               ntree = ntree)
  key <- toString(ntree)
  modellist[[key]] <- fit
}
results <- caret::resamples(modellist) #2500 was up to 0.800/0.67 #3000-5000 was up to 0.814/0.695 kappa
summary(results)
```

```
##
## Call:
## summary.resamples(object = results)
##
## Models: 1000, 2000, 3000, 4000, 5000
## Number of resamples: 30
##
## Accuracy
##           Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## 1000 0.7420147 0.7660417 0.7739569 0.7743814 0.7839828 0.8009828    0
## 2000 0.7506143 0.7648873 0.7775045 0.7750783 0.7820476 0.8022113    0
## 3000 0.7493857 0.7664410 0.7770274 0.7743409 0.7798278 0.8009828    0
## 4000 0.7493857 0.7669426 0.7766157 0.7747085 0.7803006 0.8022113    0
## 5000 0.7481572 0.7666564 0.7768904 0.7745864 0.7800307 0.8009828    0
##
## Kappa
##           Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## 1000 0.5729660 0.6146188 0.6258167 0.6288168 0.6456703 0.6727630    0
## 2000 0.5883256 0.6140989 0.6308815 0.6299729 0.6428179 0.6747015    0
## 3000 0.5861945 0.6169314 0.6308447 0.6288738 0.6407800 0.6727630    0
## 4000 0.5859397 0.6166124 0.6296440 0.6294389 0.6407618 0.6747015    0
## 5000 0.5842956 0.6158975 0.6307233 0.6293220 0.6391602 0.6725989    0
```

```
lattice::dotplot(results)#errors multiple times in knitr
```




```
############# Ranger Package
library(ranger)
set.seed(500)
rf_fit <- train(Cover_Type~., data=training, method ="ranger")
rf_fit
```

```
## Random Forest
##
## 8139 samples
##   39 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 8139, 8139, 8139, 8139, 8139, 8139, ...
## Resampling results across tuning parameters:
##
##   mtry  splitrule   Accuracy   Kappa
##    2    gini        0.6661455  0.4205713
##    2    extratrees  0.6244327  0.3380173
##   20    gini        0.7820435  0.6431413
##   20    extratrees  0.7900143  0.6572566
##   39    gini        0.7774212  0.6356484
##   39    extratrees  0.7931995  0.6625957
##
## Tuning parameter 'min.node.size' was held constant at a value of 1
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were mtry = 39, splitrule =
##  extratrees and min.node.size = 1.
```

```
rf_predict <- predict(rf_fit, testing)#0.8089/0.6889 on first out of box run!!
caret::confusionMatrix(rf_predict, testing$Cover_Type) #didnt set seed got 0.82/0.48
```

```
## Confusion Matrix and Statistics
##
##                  Reference
## Prediction        Spuce_Fir Lodgepole Ponderosa CottonWd_Willow Aspen
##   Spuce_Fir            1028       191         0               0     2
##   Lodgepole             228      1487        16               0    36
##   Ponderosa               0        11       177               7     1
##   CottonWd_Willow         0         0        12               8     0
##   Aspen                   1         4         0               0    18
##   DougFir                 0         7         9               1     0
##   Krummholz              14         0         0               0     0
##                  Reference
## Prediction        DougFir Krummholz
##   Spuce_Fir             0        28
##   Lodgepole            18         2
##   Ponderosa            41         0
##   CottonWd_Willow       1         0
##   Aspen                 0         0
##   DougFir              44         0
##   Krummholz             0        93
##
## Overall Statistics
##
##                Accuracy : 0.8192
##                  95% CI : (0.806, 0.8319)
##     No Information Rate : 0.4878
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.7052
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Spuce_Fir Class: Lodgepole Class: Ponderosa
## Sensitivity                    0.8088           0.8747          0.82710
## Specificity                    0.9002           0.8319          0.98166
## Pos Pred Value                 0.8231           0.8321          0.74684
## Neg Pred Value                 0.8913           0.8746          0.98861
## Prevalence                     0.3647           0.4878          0.06141
## Detection Rate                 0.2950           0.4267          0.05079
## Detection Prevalence           0.3584           0.5128          0.06801
## Balanced Accuracy              0.8545           0.8533          0.90438
##                      Class: CottonWd_Willow Class: Aspen Class: DougFir
## Sensitivity                        0.500000     0.315789        0.42308
## Specificity                        0.996253     0.998541        0.99497
## Pos Pred Value                     0.380952     0.782609        0.72131
## Neg Pred Value                     0.997691     0.988735        0.98248
## Prevalence                         0.004591     0.016356        0.02984
## Detection Rate                     0.002296     0.005165        0.01263
## Detection Prevalence               0.006026     0.006600        0.01750
## Balanced Accuracy                  0.748126     0.657165        0.70902
##                      Class: Krummholz
## Sensitivity                   0.75610
## Specificity                   0.99584
## Pos Pred Value                0.86916
## Neg Pred Value                0.99112
## Prevalence                    0.03529
## Detection Rate                0.02669
## Detection Prevalence          0.03070
## Balanced Accuracy             0.87597
```

```
##Tune ranger
rf_pca <- preProcess(select(training, -Cover_Type),
                     method = c("center","scale","YeoJohnson", "nzv","pca"))
rf_pca
```

```
## Created from 8139 samples and 39 variables
##
## Pre-processing:
##   - centered (21)
##   - ignored (0)
##   - principal component signal extraction (21)
##   - removed (18)
##   - scaled (21)
##   - Yeo-Johnson transformation (6)
##
## Lambda estimates for Yeo-Johnson transformation:
## 0.4, 0.3, 0.43, 1.23, 0.47, 0.41
## PCA needed 16 components to capture 95 percent of the variance
```

```
rf_pca$method
```

```
## $center
##  [1] "Elevation"
##  [2] "Horizontal_Distance_To_Roadways"
##  [3] "Horizontal_Distance_To_Fire_Points"
##  [4] "Dist_water"
##  [5] "Hillshade_Noon"
##  [6] "Hillshade_3pm"
##  [7] "Hillshade_9am"
##  [8] "Aspect"
##  [9] "Slope"
## [10] "Wilderness_Area4"
## [11] "Wilderness_Area3"
## [12] "Wilderness_Area1"
## [13] "Wilderness_Area2"
## [14] "Soil_Type10"
## [15] "Soil_Type12"
## [16] "Soil_Type22"
## [17] "Soil_Type23"
## [18] "Soil_Type29"
## [19] "Soil_Type30"
## [20] "Soil_Type32"
## [21] "Soil_Type33"
##
## $scale
##  [1] "Elevation"
##  [2] "Horizontal_Distance_To_Roadways"
##  [3] "Horizontal_Distance_To_Fire_Points"
##  [4] "Dist_water"
##  [5] "Hillshade_Noon"
##  [6] "Hillshade_3pm"
##  [7] "Hillshade_9am"
##  [8] "Aspect"
##  [9] "Slope"
## [10] "Wilderness_Area4"
## [11] "Wilderness_Area3"
## [12] "Wilderness_Area1"
## [13] "Wilderness_Area2"
## [14] "Soil_Type10"
## [15] "Soil_Type12"
## [16] "Soil_Type22"
## [17] "Soil_Type23"
## [18] "Soil_Type29"
## [19] "Soil_Type30"
## [20] "Soil_Type32"
## [21] "Soil_Type33"
##
## $YeoJohnson
## [1] "Horizontal_Distance_To_Roadways"
## [2] "Horizontal_Distance_To_Fire_Points"
## [3] "Dist_water"
## [4] "Hillshade_3pm"
## [5] "Aspect"
## [6] "Slope"
##
## $pca
##  [1] "Elevation"
##  [2] "Horizontal_Distance_To_Roadways"
##  [3] "Horizontal_Distance_To_Fire_Points"
```

```
##  [4] "Dist_water"
##  [5] "Hillshade_Noon"
##  [6] "Hillshade_3pm"
##  [7] "Hillshade_9am"
##  [8] "Aspect"
##  [9] "Slope"
## [10] "Wilderness_Area4"
## [11] "Wilderness_Area3"
## [12] "Wilderness_Area1"
## [13] "Wilderness_Area2"
## [14] "Soil_Type10"
## [15] "Soil_Type12"
## [16] "Soil_Type22"
## [17] "Soil_Type23"
## [18] "Soil_Type29"
## [19] "Soil_Type30"
## [20] "Soil_Type32"
## [21] "Soil_Type33"
##
## $ignore
## character(0)
##
## $remove
##  [1] "Soil_Type2"  "Soil_Type3"  "Soil_Type4"  "Soil_Type6"  "Soil_Type11"
##  [6] "Soil_Type13" "Soil_Type16" "Soil_Type17" "Soil_Type19" "Soil_Type20"
## [11] "Soil_Type24" "Soil_Type31" "Soil_Type34" "Soil_Type36" "Soil_Type37"
## [16] "Soil_Type38" "Soil_Type39" "Soil_Type40"
```

```
rf_pca$rotation
```

```
##                                            PC1          PC2          PC3
## Elevation                           0.27432768 -0.307882582  0.249940704
## Horizontal_Distance_To_Roadways     0.32288160 -0.116949450 -0.087846823
## Horizontal_Distance_To_Fire_Points  0.30339207  0.008085476 -0.023757854
## Dist_water                          0.02669871 -0.133551664  0.138907526
## Hillshade_Noon                      0.13250464 -0.418337060 -0.155100758
## Hillshade_3pm                      -0.10298976 -0.442696692 -0.400408178
## Hillshade_9am                       0.24943924  0.231580917  0.378070579
## Aspect                             -0.11541529 -0.337766871 -0.332651722
## Slope                              -0.27940952  0.173907743  0.004686353
## Wilderness_Area4                   -0.30397472  0.243317729 -0.274921137
## Wilderness_Area3                   -0.26835924 -0.255803622  0.373488600
## Wilderness_Area1                    0.42419345  0.182718450 -0.271281290
## Wilderness_Area2                   -0.01230934 -0.108727109  0.080620027
## Soil_Type10                        -0.28608064  0.200511692 -0.188779161
## Soil_Type12                         0.13081278  0.062692272 -0.088292579
## Soil_Type22                         0.03437669 -0.059280206  0.008492571
## Soil_Type23                         0.04912963 -0.094413436  0.012595799
## Soil_Type29                         0.26743094  0.125230287 -0.200248190
## Soil_Type30                         0.08332849  0.106660651 -0.048638080
## Soil_Type32                        -0.07437515 -0.213853964  0.213227887
## Soil_Type33                        -0.12208374 -0.074032539  0.212579028
##                                            PC4           PC5         PC6
## Elevation                          -0.13498822  0.2577345088 -0.02511210
## Horizontal_Distance_To_Roadways    -0.24689607 -0.1518204606 -0.16235460
## Horizontal_Distance_To_Fire_Points -0.12603082  0.0009624863  0.17565723
## Dist_water                         -0.32582079  0.0736199237  0.47322276
## Hillshade_Noon                      0.23293765 -0.1835163034  0.01838791
## Hillshade_3pm                      -0.03944931 -0.0099307878  0.01095000
## Hillshade_9am                       0.24957382 -0.1330990954  0.01258783
## Aspect                             -0.07707540  0.0496393556 -0.04134520
## Slope                              -0.41801852  0.2354200722 -0.08227099
## Wilderness_Area4                    0.13072691 -0.0278574117  0.14602975
## Wilderness_Area3                   -0.10450112 -0.2848049627 -0.15149395
## Wilderness_Area1                   -0.07579144  0.0188668499 -0.04156812
## Wilderness_Area2                    0.25186282  0.6118902532  0.26265328
## Soil_Type10                         0.03163729 -0.0641900379  0.09173233
## Soil_Type12                         0.13240556 -0.2496957494  0.48072558
## Soil_Type22                         0.10501248  0.3841583750  0.18821116
## Soil_Type23                         0.43252604  0.1570020830 -0.44314173
## Soil_Type29                        -0.28023215 -0.0057479087 -0.07148458
## Soil_Type30                        -0.06799808  0.0725433721 -0.17585127
## Soil_Type32                         0.04734935 -0.2869500386  0.25788937
## Soil_Type33                        -0.31744390  0.1128568401 -0.14240913
```

```
##                                            PC7          PC8          PC9
## Elevation                          0.094038250 -0.027643640  0.036063886
## Horizontal_Distance_To_Roadways    0.004729975  0.038279798 -0.132088425
## Horizontal_Distance_To_Fire_Points -0.186957198  0.194737637  0.109931426
## Dist_water                         0.074185792 -0.024237283  0.288879380
## Hillshade_Noon                    -0.015921843 -0.082307717 -0.022232026
## Hillshade_3pm                     -0.037315201  0.038654451  0.008698438
## Hillshade_9am                      0.051628913 -0.089377521 -0.030243187
## Aspect                            -0.064602496 -0.128298986  0.024324979
## Slope                             -0.081615828 -0.060894385  0.046662510
## Wilderness_Area4                   0.110402457  0.008385414  0.062991360
## Wilderness_Area3                  -0.038449893  0.038348643 -0.145164684
## Wilderness_Area1                  -0.029368656 -0.082994355 -0.016815602
## Wilderness_Area2                   0.027665405  0.089340350  0.285394979
## Soil_Type10                        0.105912567  0.063331208  0.005485621
## Soil_Type12                       -0.542451429  0.184509269 -0.041535854
## Soil_Type22                        0.063537033 -0.197678020 -0.769517144
## Soil_Type23                       -0.071542258  0.254476948  0.269556358
## Soil_Type29                        0.474296683  0.213778181  0.064851043
## Soil_Type30                       -0.335162780 -0.752875830  0.256046413
## Soil_Type32                        0.340891897 -0.266795750  0.190635197
## Soil_Type33                       -0.390015165  0.284308013 -0.044817594
##                                          PC10         PC11        PC12
## Elevation                          0.16924592  0.100707478  0.05513681
## Horizontal_Distance_To_Roadways    0.29470360  0.234120927 -0.19821989
## Horizontal_Distance_To_Fire_Points  0.31303070 -0.314178388 -0.54141617
## Dist_water                         0.01531302  0.610757827  0.16722943
## Hillshade_Noon                    -0.33778574  0.151256123 -0.21227978
## Hillshade_3pm                      0.03480115 -0.010514753 -0.01792853
## Hillshade_9am                     -0.27479082  0.133161264 -0.14159134
## Aspect                            -0.14838461 -0.110128797  0.06929724
## Slope                              0.16662510 -0.189751032  0.20618358
## Wilderness_Area4                  -0.10825677  0.205125209 -0.11578969
## Wilderness_Area3                   0.11397463 -0.016988848 -0.04592992
## Wilderness_Area1                  -0.01241122  0.002539861  0.19481529
## Wilderness_Area2                  -0.10336212 -0.193172046 -0.20082644
## Soil_Type10                        0.29666767  0.290821597 -0.43824198
## Soil_Type12                        0.07280063 -0.083228613  0.33424879
## Soil_Type22                        0.11103889  0.074100409  0.01110353
## Soil_Type23                        0.32016989  0.219831869  0.24006523
## Soil_Type29                       -0.28161826 -0.123178427  0.05451624
## Soil_Type30                       -0.01425210  0.060773818 -0.15183253
## Soil_Type32                        0.15095856 -0.355702291  0.05600498
## Soil_Type33                       -0.44685867  0.063710445 -0.19831788
##                                           PC13        PC14        PC15
## Elevation                         -0.235774323  0.07385050 -0.02759981
## Horizontal_Distance_To_Roadways   -0.475289762  0.05162122  0.18328664
## Horizontal_Distance_To_Fire_Points  0.442180151 -0.12745684  0.25323608
## Dist_water                         0.281485520 -0.04660892  0.07245204
## Hillshade_Noon                     0.129067109  0.12511238  0.02955795
## Hillshade_3pm                      0.055339452 -0.19858478 -0.26563660
## Hillshade_9am                      0.007252544  0.27531816  0.29386737
## Aspect                            -0.025600029  0.43034480  0.53381812
## Slope                              0.045295792  0.24083858  0.25137904
## Wilderness_Area4                  -0.074657171 -0.32982779  0.39656510
## Wilderness_Area3                   0.213729438  0.19068473 -0.11657651
## Wilderness_Area1                  -0.071741987 -0.10866519 -0.01884401
## Wilderness_Area2                  -0.227938486  0.18317527 -0.13814160
## Soil_Type10                       -0.242509765  0.30875505 -0.19112008
## Soil_Type12                       -0.199415263  0.17542080 -0.05122562
## Soil_Type22                        0.106148413 -0.14862447  0.11983505
## Soil_Type23                        0.095957975 -0.13386074  0.23500271
## Soil_Type29                        0.081559492  0.10923822 -0.11357313
## Soil_Type30                        0.032978146 -0.06365220 -0.15503252
## Soil_Type32                       -0.321431981 -0.32504312  0.21048385
## Soil_Type33                       -0.287757835 -0.34802113  0.10025731
##                                          PC16
## Elevation                          0.02178985
## Horizontal_Distance_To_Roadways   -0.41167442
## Horizontal_Distance_To_Fire_Points  0.05811628
## Dist_water                         0.06419132
## Hillshade_Noon                    -0.07777275
## Hillshade_3pm                     -0.04693202
## Hillshade_9am                      0.03457958
## Aspect                             0.24268920
## Slope                             -0.13618939
```

```
## Wilderness_Area4                  -0.32017740
## Wilderness_Area3                  -0.03294776
## Wilderness_Area1                   0.27505073
## Wilderness_Area2                  -0.18161008
## Soil_Type10                        0.48614249
## Soil_Type12                        0.03208213
## Soil_Type22                        0.17395183
## Soil_Type23                        0.23267471
## Soil_Type29                        0.15428630
## Soil_Type30                        0.05422167
## Soil_Type32                        0.27298202
## Soil_Type33                        0.31686047
```

```
#########
fit_control <- trainControl(method = "cv",
                            number = 10)
set.seed(56)
rf_fit <- train(Cover_Type ~ .,
                data = training,
                method = "ranger",
                trControl = fit_control)
rf_fit
```

```
## Random Forest
##
## 8139 samples
##   39 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 7325, 7324, 7323, 7324, 7326, 7326, ...
## Resampling results across tuning parameters:
##
##   mtry  splitrule   Accuracy   Kappa
##    2    gini        0.6692624  0.4273693
##    2    extratrees  0.6294417  0.3500246
##   20    gini        0.8002243  0.6725144
##   20    extratrees  0.8042743  0.6810312
##   39    gini        0.7993610  0.6717280
##   39    extratrees  0.8086970  0.6878663
##
## Tuning parameter 'min.node.size' was held constant at a value of 1
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were mtry = 39, splitrule =
##  extratrees and min.node.size = 1.
```

```
#####   downsample
fit_control <- trainControl(method = "cv",
                            number = 10,
                            sampling = "down") #down sample class imbalance
set.seed(57)
rf_fit_down <- train(Cover_Type ~ .,
                data = training,
                method = "ranger",
                trControl = fit_control)
rf_fit_down #dropped 0.52/0.33
```

```
## Random Forest
##
## 8139 samples
##   39 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 7324, 7326, 7324, 7327, 7325, 7324, ...
## Addtional sampling using down-sampling
##
## Resampling results across tuning parameters:
##
##   mtry  splitrule   Accuracy   Kappa
##    2    gini        0.4335954  0.2342162
##    2    extratrees  0.4174997  0.2316850
##   20    gini        0.5191970  0.3350207
##   20    extratrees  0.5117244  0.3229175
##   39    gini        0.4977128  0.3087066
##   39    extratrees  0.5035991  0.3136745
##
## Tuning parameter 'min.node.size' was held constant at a value of 1
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were mtry = 20, splitrule = gini
##  and min.node.size = 1.
```

```
###########   Upsample
fit_control <- trainControl(method = "cv",
                            number = 10,
                            sampling = "up")#mtry39   extratrees  0.8161981  0.7035628
set.seed(57)
rf_fit_up <- train(Cover_Type ~ .,
                     data = training,
                     method = "ranger",
                     trControl = fit_control)
rf_fit_up
```

```
## Random Forest
##
## 8139 samples
##   39 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 7324, 7326, 7324, 7327, 7325, 7324, ...
## Addtional sampling using up-sampling
##
## Resampling results across tuning parameters:
##
##   mtry  splitrule   Accuracy   Kappa
##    2    gini        0.5180076  0.3317084
##    2    extratrees  0.4618409  0.2737728
##   20    gini        0.8012008  0.6796778
##   20    extratrees  0.8121395  0.6976766
##   39    gini        0.7824059  0.6514630
##   39    extratrees  0.8104197  0.6952550
##
## Tuning parameter 'min.node.size' was held constant at a value of 1
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were mtry = 20, splitrule =
##  extratrees and min.node.size = 1.
```

```
range_pred <- predict(rf_fit_up, testing)
caret::confusionMatrix(range_pred, testing$Cover_Type) #0.807/0.689 repeated CV
```

```
## Confusion Matrix and Statistics
##
##                  Reference
## Prediction        Spuce_Fir Lodgepole Ponderosa CottonWd_Willow Aspen
##   Spuce_Fir            1035       197         0               0     1
##   Lodgepole             212      1463        12               0    30
##   Ponderosa               0        14       178               8     1
##   CottonWd_Willow         0         0        11               7     0
##   Aspen                   2        11         0               0    25
##   DougFir                 1        15        13               1     0
##   Krummholz              21         0         0               0     0
##                  Reference
## Prediction        DougFir Krummholz
##   Spuce_Fir             0        27
##   Lodgepole            14         2
##   Ponderosa            42         0
##   CottonWd_Willow       1         0
##   Aspen                 0         0
##   DougFir              47         0
##   Krummholz             0        94
##
## Overall Statistics
##
##                Accuracy : 0.8175
##                  95% CI : (0.8043, 0.8302)
##     No Information Rate : 0.4878
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.7053
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Spuce_Fir Class: Lodgepole Class: Ponderosa
## Sensitivity                    0.8143           0.8606          0.83178
## Specificity                    0.8984           0.8487          0.98013
## Pos Pred Value                 0.8214           0.8442          0.73251
## Neg Pred Value                 0.8939           0.8647          0.98890
## Prevalence                     0.3647           0.4878          0.06141
## Detection Rate                 0.2970           0.4198          0.05108
## Detection Prevalence           0.3615           0.4973          0.06973
## Balanced Accuracy              0.8563           0.8547          0.90595
##                      Class: CottonWd_Willow Class: Aspen Class: DougFir
## Sensitivity                        0.437500     0.438596        0.45192
## Specificity                        0.996541     0.996208        0.99113
## Pos Pred Value                     0.368421     0.657895        0.61039
## Neg Pred Value                     0.997403     0.990717        0.98327
## Prevalence                         0.004591     0.016356        0.02984
## Detection Rate                     0.002009     0.007174        0.01349
## Detection Prevalence               0.005452     0.010904        0.02209
## Balanced Accuracy                  0.717020     0.717402        0.72152
##                      Class: Krummholz
## Sensitivity                   0.76423
## Specificity                   0.99375
## Pos Pred Value                0.81739
## Neg Pred Value                0.99139
## Prevalence                    0.03529
## Detection Rate                0.02697
## Detection Prevalence          0.03300
## Balanced Accuracy             0.87899
```

```
### Final
tuneGrid <- expand.grid(
  .mtry = 39,
  .splitrule = "extratrees",
  .min.node.size = 1
)
set.seed(201)
rf_final <- train(Cover_Type~.,
                  data = training,
                  method = "ranger",
                  tuneGrid = tuneGrid,
                  num.trees = 3000,
                  trControl = fit_control
                  )
```

```
## Growing trees.. Progress: 38%. Estimated remaining time: 50 seconds.
## Growing trees.. Progress: 73%. Estimated remaining time: 22 seconds.
```

```
rf_final
```

```
## Random Forest
##
## 8139 samples
##   39 predictor
##    7 classes: 'Spuce_Fir', 'Lodgepole', 'Ponderosa', 'CottonWd_Willow', 'Aspen', 'DougFir', 'Krummholz'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 7324, 7323, 7324, 7326, 7325, 7326, ...
## Addtional sampling using up-sampling
##
## Resampling results:
##
##   Accuracy   Kappa
##   0.8127548  0.6987721
##
## Tuning parameter 'mtry' was held constant at a value of 39
##  extratrees
## Tuning parameter 'min.node.size' was held constant at a
##  value of 1
```

```
rf_final$results
```

	mtry <dbl>	splitrule <fctr>	min.node.size <dbl>	Accuracy <dbl>	Kappa <dbl>	AccuracySD <dbl>	KappaSD <dbl>
1	39	extratrees	1	0.8127548	0.6987721	0.009159214	0.01469148

1 row

```
set.seed(202)
final_pred <- predict(rf_final, testing)
caret::confusionMatrix(final_pred, testing$Cover_Type)
```

```
## Confusion Matrix and Statistics
##
##                  Reference
## Prediction        Spuce_Fir Lodgepole Ponderosa CottonWd_Willow Aspen
##   Spuce_Fir            1042       198         0               0     1
##   Lodgepole             207      1462        14               0    29
##   Ponderosa               0        15       176               6     1
##   CottonWd_Willow         0         0        10               9     0
##   Aspen                   2        11         1               0    26
##   DougFir                 0        14        13               1     0
##   Krummholz              20         0         0               0     0
##                  Reference
## Prediction        DougFir Krummholz
##   Spuce_Fir             0        27
##   Lodgepole            15         1
##   Ponderosa            35         0
##   CottonWd_Willow       1         0
##   Aspen                 0         0
##   DougFir              53         0
##   Krummholz             0        95
##
## Overall Statistics
##
##                Accuracy : 0.8215
##                  95% CI : (0.8084, 0.8341)
##     No Information Rate : 0.4878
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.7118
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Spuce_Fir Class: Lodgepole Class: Ponderosa
## Sensitivity                    0.8198           0.8600          0.82243
## Specificity                    0.8979           0.8510          0.98257
## Pos Pred Value                 0.8218           0.8461          0.75536
## Neg Pred Value                 0.8967           0.8645          0.98831
## Prevalence                     0.3647           0.4878          0.06141
## Detection Rate                 0.2990           0.4195          0.05050
## Detection Prevalence           0.3638           0.4958          0.06686
## Balanced Accuracy              0.8589           0.8555          0.90250
##                      Class: CottonWd_Willow Class: Aspen Class: DougFir
## Sensitivity                        0.562500     0.456140        0.50962
## Specificity                        0.996829     0.995916        0.99172
## Pos Pred Value                     0.450000     0.650000        0.65432
## Neg Pred Value                     0.997980     0.991001        0.98502
## Prevalence                         0.004591     0.016356        0.02984
## Detection Rate                     0.002582     0.007461        0.01521
## Detection Prevalence               0.005739     0.011478        0.02324
## Balanced Accuracy                  0.779665     0.726028        0.75067
##                      Class: Krummholz
## Sensitivity                   0.77236
## Specificity                   0.99405
## Pos Pred Value                0.82609
## Neg Pred Value                0.99169
## Prevalence                    0.03529
## Detection Rate                0.02726
## Detection Prevalence          0.03300
## Balanced Accuracy             0.88320
```

```
stopCluster(cl)
```