Capstone Data Logistic Regression - Predict Douglas Fir

Tom Thorpe

July 25, 2018

Objective

Use Logistic regression to predict tree coverage.

```
# Include required libraries.

library(gsubfn)
```

```
## Loading required package: proto
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(ggridges) # for easier viewing of sub-group distributions
library(ROCR)
```

```
## Loading required package: gplots
##
## Attaching package: 'gplots'
## The following object is masked from 'package:stats':
##
##     lowess
```

```
suppressMessages(library(latticeExtra, warn.conflicts = FALSE, quietly=TRUE))
#library(latticeExtra)

  curTime=Sys.time()
  print(paste("Forest Cover Logistic script started at",curTime))
```

```
## [1] "Forest Cover Logistic script started at 2018-08-12 18:11:22"
```

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

calcROC <- 1
saveFileName="ForestCoverLogisticStats.csv"

infile="C:/Users/Tom/git/datasciencefoundation/ForestCoverage/forestcover_clean_full.csv"
#infile="C:/Users/Tom/git/datasciencefoundation/ForestCoverage/forestcover_clean.csv"
#infile="C:/Users/Tom/git/datasciencefoundation/ForestCoverage/forestcoversmall_clean_full.csv"
```

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

alphaVal<-0.05 # large data
#alphaVal<-0.1  # small data
```

```
forestcover <- read.csv(infile,header=TRUE,sep=",") %>% tbl_df()
  curTime=Sys.time()
  print(paste("Forest Cover data load completed at",curTime))
```

```
## [1] "Forest Cover data load completed at 2018-08-12 18:12:00"
```

```
forestcover$SoilType<-as.factor(forestcover$SoilType)
forestcover$ClimateZone<-as.factor(forestcover$ClimateZone)
forestcover$GeoZone<-as.factor(forestcover$GeoZone)

# glimpse(forestcover)

# table(forestcover$Sed_mix)
#knitr::knit_exit()

# Coverage binary outcome Vars:
# Aspen
# Cottonwood_Willow
# DouglasFir
# Krummholz
# LodgepolePine
# PonderosaPine
# Spruce_Fir
```

A table showing the number of occurrences for each tree type is shown below.

```
covCount<-data.frame(table(forestcover$CovName))
totCount<-nrow(forestcover)
covCount <- mutate(covCount,Percent = as.integer(covCount$Freq*1000/totCount)/10)
LodgePct<-covCount$Percent[covCount$Var1=="Lodgepole"]
SpruceAndFirPct<-covCount$Percent[covCount$Var1=="Spruce&Fir"]
LodgeAndSpruceAndFir<-LodgePct+SpruceAndFirPct
#```
#```{r echo=TRUE}
covCount
```

```
##            Var1   Freq Percent
## 1         Aspen   9493     1.6
## 2 Cotton&Willow   2747     0.4
## 3    DouglasFir  17367     2.9
## 4     Krummholz  20510     3.5
## 5     Lodgepole 283301    48.7
## 6     Ponderosa  35754     6.1
## 7    Spruce&Fir 211840    36.4
```

Lodge pole Pine represents 48.7 percent of the sample. So always guessing "Lodge pole" would provide success rate of 48.7 percent and can be used as a baseline for comparing our predictions. Spruce & Fir represent the next largest number of trees. The two together represent 85.1 percent.

Logistic Model Accuracy Function

A function to help determine threshold for best accuracy and testing is shown below.

```
source("logisticAccuracy.R") # for function calcLogisticModelAccuracy
#save("calcLogisticModelAccuracy", file="logisticAccuracy.Rdata")
bestThreshIndex=11
```

Create Training and Testing Sets

Split data into training and testing data for logistic regression. The split is based on cover type so that the different coverage types will be split proportionately for all cover types in the training and test sets.

```
library(caTools)
set.seed(127)
split = sample.split(forestcover$CovType, 0.70) # we want 65% in the training set
forestTrain = subset(forestcover, split == TRUE)
forestTest  = subset(forestcover, split == FALSE)
```

Check training set coverage percentages and compare with test set to ensure there is a representative amount of data in each set for each coverage type.

View Training Set Coverage Percentages

Check training set coverage percentages.

```
covCount<-data.frame(table(forestTrain$CovName))
totCount<-nrow(forestTrain)
covCount <- mutate(covCount,Percent = as.integer(covCount$Freq*1000/totCount)/10)
covCount
```

```
##            Var1   Freq Percent
## 1         Aspen   6645     1.6
## 2 Cotton&Willow   1923     0.4
## 3    DouglasFir  12157     2.9
## 4     Krummholz  14357     3.5
## 5     Lodgepole 198311    48.7
## 6     Ponderosa  25028     6.1
## 7    Spruce&Fir 148288    36.4
```

View Test Set Coverage Percentages

Check test set coverage percentages.

```
covCount<-data.frame(table(forestTest$CovName))
totCount<-nrow(forestTest)
covCount <- mutate(covCount,Percent = as.integer(covCount$Freq*1000/totCount)/10)
covCount
```

```
##            Var1  Freq Percent
## 1         Aspen  2848     1.6
## 2 Cotton&Willow   824     0.4
## 3    DouglasFir  5210     2.9
## 4     Krummholz  6153     3.5
## 5     Lodgepole 84990    48.7
```

```
## 6     Ponderosa 10726     6.1
## 7    Spruce&Fir 63552    36.4
```

```
# knitr::knit_exit() # exit early

#glimpse(forestTrain)
#glimpse(forestTest)
#summary(forestTrain)
#summary(forestTest)
#table(forestTrain$Sed_mix)
#table(forestTrain$GeoName)
#table(forestTrain$Spruce_Fir)
#table(forestTest$Spruce_Fir)

# the above all work without error.

#table(forestTest$Rock_Land)
# Get the following error with above code:
#  Error in table(SpfFir_test$Rock_Land) : object 'SpfFir_test' not found
#    Calls: <Anonymous> ... withCallingHandlers -> withVisible -> eval -> eval -> table


#table(forestTrain$Rock_Land)
#table(forestTest$Rock_Land)
#table(forestTrain$Rubbly)
#table(forestTest$Rubbly)

#table(forestTrain$Sed_mix)
#table(forestTrain$Gateview)
#table(forestTrain$Rubbly)
#table(forestTest$Sed_mix)
#table(forestTest$Gateview)
#table(forestTest$Rubbly)

############# Start Start Start Start Start Start Start Start ##################
```

Douglas Fir Logistic Regression

Logistic regression models are created and compared for the Douglas Fir coverage type. The outcome is based on the binary 'DouglasFir' variable.

Douglas Fir Logistic Regression - All Variables

Create Douglas Fir Logistic Model - All Vars

Create the Douglas Fir logistic model for the Aggregated Soil data using all independent variables.

Douglas Fir All Aggregated Soil Types

The original project used aggregated Soil Types. Compute a logistic regression model using the aggregated soil types to see how the dis-aggregated / individuated variables compare.

```r
# You can remove the levels of the factor variables using the option exclude:
#   lm(dependent ~ factor(independent1, exclude=c('b','d')) + independent2)
#   This way the factors b, d will not be included in the regression.

curTime=Sys.time()
print(paste("DouglasFir aggregated Logistic Model Calculation started at",curTime))
```

```
## [1] "DouglasFir aggregated Logistic Model Calculation started at 2018-08-12 18:12:02"
```

```r
DougFir_Agg_LogMod =
  glm(DouglasFir ~
        Elev +      # Elevation in meters of data cell
        Aspect +    # Direction in degrees slope faces
        Slope +     # Slope / steepness of hill in degrees (0 to 90)
        H2OHD +     # Horizontal distance in meters to nearest water
        H2OVD +     # Vertical distance in meters to nearest water
        RoadHD +    # Horizontal distance in meters to nearest road
        FirePtHD + # Horizontal distance in meters to nearest fire point
        Shade9AM + Shade12PM + Shade3PM + # Amount of shade at 9am, 12pm and 3pm
        # Wilderness areas:
          RWwild + NEwild + CMwild + CPwild +
        # Aggregated Soil type:
          ST01 + ST02 + ST03 + ST04 + ST05 + ST06 + ST07 + ST08 + ST09 + ST10 +
          ST11 + ST12 + ST13 + ST14 + ST15 + ST16 + ST17 + ST18 + ST19 + ST20 +
          ST21 + ST22 + ST23 + ST24 + ST25 + ST26 + ST27 + ST28 + ST29 + ST30 +
          ST31 + ST32 + ST33 + ST34 + ST35 + ST36 + ST37 + ST38 + ST39 + ST40 ,
        data=forestTrain, family=binomial)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
# save model for later use
DougFir_Agg_All_LogMod = DougFir_Agg_LogMod
save("DougFir_Agg_All_LogMod", file="DougFir_Agg_All_LogMod.Rdata")

DougFir_Agg_All_aic<-as.integer(DougFir_Agg_LogMod$aic)
DougFir_Agg_All_aic
```

```
## [1] 57784
```

```r
curTime=Sys.time()
print(paste("DouglasFir aggregated Logistic Model Calculation completed at",curTime))
```

```
## [1] "DouglasFir aggregated Logistic Model Calculation completed at 2018-08-12 18:14:28"
```

Check the coefficients for the Douglas Fir model using all aggregated data.

```r
summary(DougFir_Agg_LogMod)
```

```
##
## Call:
## glm(formula = DouglasFir ~ Elev + Aspect + Slope + H2OHD + H2OVD +
##     RoadHD + FirePtHD + Shade9AM + Shade12PM + Shade3PM + RWwild +
##     NEwild + CMwild + CPwild + ST01 + ST02 + ST03 + ST04 + ST05 +
##     ST06 + ST07 + ST08 + ST09 + ST10 + ST11 + ST12 + ST13 + ST14 +
##     ST15 + ST16 + ST17 + ST18 + ST19 + ST20 + ST21 + ST22 + ST23 +
##     ST24 + ST25 + ST26 + ST27 + ST28 + ST29 + ST30 + ST31 + ST32 +
##     ST33 + ST34 + ST35 + ST36 + ST37 + ST38 + ST39 + ST40, family = binomial,
```

```
##     data = forestTrain)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.1226  -0.0894   0.0000   0.0000   4.3276
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.110e+09  3.433e+11  -0.003    0.997
## Elev        -4.165e-03  1.040e-04 -40.040  < 2e-16 ***
## Aspect       1.385e-05  1.228e-04   0.113    0.910
## Slope        4.384e-02  5.615e-03   7.807 5.86e-15 ***
## H2OHD       -1.847e-03  1.163e-04 -15.886  < 2e-16 ***
## H2OVD        7.026e-05  3.046e-04   0.231    0.818
## RoadHD       3.020e-05  1.912e-05   1.580    0.114
## FirePtHD     3.275e-04  2.128e-05  15.386  < 2e-16 ***
## Shade9AM     1.455e-01  5.829e-03  24.954  < 2e-16 ***
## Shade12PM   -1.511e-01  4.921e-03 -30.707  < 2e-16 ***
## Shade3PM     1.331e-01  4.880e-03  27.268  < 2e-16 ***
## RWwild      -1.712e+01  9.362e+01  -0.183    0.855
## NEwild      -1.363e+01  1.720e+02  -0.079    0.937
## CMwild       8.183e-01  4.048e-02  20.214  < 2e-16 ***
## CPwild              NA         NA      NA       NA
## ST01         1.110e+09  3.433e+11   0.003    0.997
## ST02         1.110e+09  3.433e+11   0.003    0.997
## ST03         1.110e+09  3.433e+11   0.003    0.997
## ST04         1.110e+09  3.433e+11   0.003    0.997
## ST05         1.110e+09  3.433e+11   0.003    0.997
## ST06         1.110e+09  3.433e+11   0.003    0.997
## ST07         1.110e+09  3.433e+11   0.003    0.997
## ST08         1.110e+09  3.433e+11   0.003    0.997
## ST09         1.110e+09  3.433e+11   0.003    0.997
## ST10         1.110e+09  3.433e+11   0.003    0.997
## ST11         1.110e+09  3.433e+11   0.003    0.997
## ST12         1.110e+09  3.433e+11   0.003    0.997
## ST13         1.110e+09  3.433e+11   0.003    0.997
## ST14         1.110e+09  3.433e+11   0.003    0.997
## ST15         1.110e+09  3.433e+11   0.003    0.997
## ST16         1.110e+09  3.433e+11   0.003    0.997
## ST17         1.110e+09  3.433e+11   0.003    0.997
## ST18         1.110e+09  3.433e+11   0.003    0.997
## ST19         1.110e+09  3.433e+11   0.003    0.997
## ST20         1.110e+09  3.433e+11   0.003    0.997
## ST21         1.110e+09  3.433e+11   0.003    0.997
## ST22         1.110e+09  3.433e+11   0.003    0.997
## ST23         1.110e+09  3.433e+11   0.003    0.997
## ST24         1.110e+09  3.433e+11   0.003    0.997
## ST25         1.110e+09  3.433e+11   0.003    0.997
## ST26         1.110e+09  3.433e+11   0.003    0.997
## ST27         1.110e+09  3.433e+11   0.003    0.997
## ST28         1.110e+09  3.433e+11   0.003    0.997
## ST29         1.110e+09  3.433e+11   0.003    0.997
## ST30         1.110e+09  3.433e+11   0.003    0.997
## ST31         1.110e+09  3.433e+11   0.003    0.997
```

```
## ST32        1.110e+09  3.433e+11   0.003    0.997
## ST33        1.110e+09  3.433e+11   0.003    0.997
## ST34        1.110e+09  3.433e+11   0.003    0.997
## ST35        1.110e+09  3.433e+11   0.003    0.997
## ST36        1.110e+09  3.433e+11   0.003    0.997
## ST37        1.110e+09  3.433e+11   0.003    0.997
## ST38        1.110e+09  3.433e+11   0.003    0.997
## ST39        1.110e+09  3.433e+11   0.003    0.997
## ST40        1.110e+09  3.433e+11   0.003    0.997
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 109294  on 406708  degrees of freedom
## Residual deviance:  57677  on 406655  degrees of freedom
## AIC: 57785
##
## Number of Fisher Scoring iterations: 20
```

WOW! The intercept is huge and listed as not significant. Wilderness area and several soil types are not significant and can be removed in the next iteration.

Douglas Fir All Individuated Soil Types

Create a logistic model using the Individuated variables that were derived from the Soil Types. The Soil Type was the intersection of climate zone, geology zone, soil families, and rock content. These variables are used instead of the Soil types.

```
curTime=Sys.time()
print(paste("DouglasFir Individual Logistic Model Calculation started at",curTime))
```

```
## [1] "DouglasFir Individual Logistic Model Calculation started at 2018-08-12 18:14:29"
```

```
DougFir_Ind_LogMod =
  glm(DouglasFir ~
        Elev +     # Elevation in meters of cell
        Aspect +   # Direction in degrees slope faces
        Slope +    # Slope / steepness of hill in degrees (0 to 90)
        H2OHD +    # Horizontal distance in meters to nearest water
        H2OVD +    # Vertical distance in meters to nearest water
        RoadHD +   # Horizontal distance in meters to nearest road
        FirePtHD + # Horizontal distance in meters to nearest fire point
        Shade9AM + Shade12PM + Shade3PM + # Amount of shade at 9am, 12pm and 3pm
        # Wilderness areas:
          RWwild + NEwild + CMwild + CPwild +
        # Climate Zone:
        # ClimateName +
          Montane_low + Montane + SubAlpine + Alpine + Dry + Non_Dry +
        # Geology Zone:
        # GeoName +
          Alluvium + Glacial + Sed_mix + Ign_Meta +
        # Soil Family:
          Aquolis_cmplx + Argiborolis_Pachic + Borohemists_cmplx + Bross +
          Bullwark + Bullwark_Cmplx + Catamount + Catamount_cmplx +
```

```r
            Cathedral + Como + Cryaquepts_cmplx + Cryaquepts_Typic + Cryaquolls +
            Cryaquolls_cmplx + Cryaquolls_Typic + Cryaquolls_Typic_cmplx +
            Cryoborolis_cmplx + Cryorthents + Cryorthents_cmplx + Cryumbrepts +
            Cryumbrepts_cmplx + Gateview + Gothic + Granile + Haploborolis +
            Legault + Legault_cmplx + Leighcan + Leighcan_cmplx + Leighcan_warm +
            Moran + Ratake + Ratake_cmplx + Rogert + Supervisor_Limber_cmplx +
            Troutville + Unspecified + Vanet + Wetmore +
          # Soil Rock composition:
            Bouldery_ext + Rock_Land + Rock_Land_cmplx + Rock_Outcrop +
            Rock_Outcrop_cmplx + Rubbly + Stony + Stony_extreme + Stony_very +
            Till_Substratum ,
          data=forestTrain, family=binomial)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
# save model for later use
DougFir_Ind_All_LogMod = DougFir_Ind_LogMod
save("DougFir_Ind_All_LogMod", file="DougFir_Ind_All_LogMod.Rdata")

#table(forestTrain$GeoName)
#table(forestTrain$Sed_mix)
#table(forestTrain$Gateview)
# above: Error in table(SpfFir_test$Gateview) : object 'SpfFir_train' not found <-------

DougFir_Ind_All_aic<-as.integer(DougFir_Ind_LogMod$aic)
DougFir_Ind_All_aic
```

```
## [1] 57790
```

```r
summary(DougFir_Ind_LogMod)
```

```
##
## Call:
## glm(formula = DouglasFir ~ Elev + Aspect + Slope + H2OHD + H2OVD +
##     RoadHD + FirePtHD + Shade9AM + Shade12PM + Shade3PM + RWwild +
##     NEwild + CMwild + CPwild + Montane_low + Montane + SubAlpine +
##     Alpine + Dry + Non_Dry + Alluvium + Glacial + Sed_mix + Ign_Meta +
##     Aquolis_cmplx + Argiborolis_Pachic + Borohemists_cmplx +
##     Bross + Bullwark + Bullwark_Cmplx + Catamount + Catamount_cmplx +
##     Cathedral + Como + Cryaquepts_cmplx + Cryaquepts_Typic +
##     Cryaquolls + Cryaquolls_cmplx + Cryaquolls_Typic + Cryaquolls_Typic_cmplx +
##     Cryoborolis_cmplx + Cryorthents + Cryorthents_cmplx + Cryumbrepts +
##     Cryumbrepts_cmplx + Gateview + Gothic + Granile + Haploborolis +
##     Legault + Legault_cmplx + Leighcan + Leighcan_cmplx + Leighcan_warm +
##     Moran + Ratake + Ratake_cmplx + Rogert + Supervisor_Limber_cmplx +
##     Troutville + Unspecified + Vanet + Wetmore + Bouldery_ext +
##     Rock_Land + Rock_Land_cmplx + Rock_Outcrop + Rock_Outcrop_cmplx +
##     Rubbly + Stony + Stony_extreme + Stony_very + Till_Substratum,
##     family = binomial, data = forestTrain)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.1224  -0.0894   0.0000   0.0000   4.3277
##
## Coefficients: (17 not defined because of singularities)
```

```
##                         Estimate Std. Error z value Pr(>|z|)    
## (Intercept)           -3.755e+09  8.703e+11  -0.004    0.997    
## Elev                  -4.165e-03  1.022e-04 -40.740  < 2e-16 ***
## Aspect                 1.421e-05  1.298e-04   0.109    0.913    
## Slope                  4.384e-02  5.692e-03   7.702 1.34e-14 ***
## H2OHD                 -1.847e-03  1.189e-04 -15.531  < 2e-16 ***
## H2OVD                  7.035e-05  3.049e-04   0.231    0.818    
## RoadHD                 3.020e-05  1.917e-05   1.576    0.115    
## FirePtHD               3.274e-04  2.156e-05  15.186  < 2e-16 ***
## Shade9AM               1.455e-01  5.926e-03  24.546  < 2e-16 ***
## Shade12PM             -1.511e-01  5.065e-03 -29.830  < 2e-16 ***
## Shade3PM               1.331e-01  4.994e-03  26.647  < 2e-16 ***
## RWwild                -1.912e+01  2.545e+02  -0.075    0.940    
## NEwild                -1.564e+01  4.696e+02  -0.033    0.973    
## CMwild                 8.183e-01  4.068e-02  20.112  < 2e-16 ***
## CPwild                        NA         NA      NA       NA    
## Montane_low           -2.382e+09  3.669e+11  -0.006    0.995    
## Montane               -7.227e+09  8.210e+11  -0.009    0.993    
## SubAlpine              3.755e+09  8.703e+11   0.004    0.997    
## Alpine                 3.755e+09  8.703e+11   0.004    0.997    
## Dry                    9.096e+09  3.305e+12   0.003    0.998    
## Non_Dry                6.137e+09  9.646e+11   0.006    0.995    
## Alluvium               1.090e+09  9.693e+11   0.001    0.999    
## Glacial                2.613e+08  1.704e+12   0.000    1.000    
## Sed_mix                1.886e+09  2.494e+12   0.001    0.999    
## Ign_Meta                      NA         NA      NA       NA    
## Aquolis_cmplx         -5.341e+09  2.576e+12  -0.002    0.998    
## Argiborolis_Pachic            NA         NA      NA       NA    
## Borohemists_cmplx     -5.241e-01  3.345e+03   0.000    1.000    
## Bross                 -1.046e+00  8.522e+03   0.000    1.000    
## Bullwark               4.845e+09  7.485e+11   0.006    0.995    
## Bullwark_Cmplx         4.845e+09  7.485e+11   0.006    0.995    
## Catamount              1.883e+01  2.062e+03   0.009    0.993    
## Catamount_cmplx       -8.005e-01  1.744e-01  -4.589 4.45e-06 ***
## Cathedral             -5.633e-02  8.957e-02  -0.629    0.529    
## Como                  -1.213e+00  1.122e+03  -0.001    0.999    
## Cryaquepts_cmplx      -1.821e+00  2.592e+03  -0.001    0.999    
## Cryaquepts_Typic      -8.286e+08  2.090e+12   0.000    1.000    
## Cryaquolls             1.662e+00  1.029e+03   0.002    0.999    
## Cryaquolls_cmplx       6.679e-01  1.029e+03   0.001    0.999    
## Cryaquolls_Typic      -1.090e+09  9.693e+11  -0.001    0.999    
## Cryaquolls_Typic_cmplx -2.613e+08  1.704e+12   0.000    1.000    
## Cryoborolis_cmplx             NA         NA      NA       NA    
## Cryorthents           1.902e+01  2.320e+03   0.008    0.993    
## Cryorthents_cmplx     3.628e-01  4.773e+03   0.000    1.000    
## Cryumbrepts                  NA         NA      NA       NA    
## Cryumbrepts_cmplx            NA         NA      NA       NA    
## Gateview                     NA         NA      NA       NA    
## Gothic               -1.350e-02  1.203e+04   0.000    1.000    
## Granile              -2.009e+01  2.080e+03  -0.010    0.992    
## Haploborolis         -1.546e+00  1.186e-01 -13.029  < 2e-16 ***
## Legault               4.845e+09  7.485e+11   0.006    0.995    
## Legault_cmplx                NA         NA      NA       NA    
## Leighcan             -2.084e+00  1.062e+03  -0.002    0.998    
```

```
## Leighcan_cmplx          -2.092e+01  2.320e+03  -0.009    0.993
## Leighcan_warm           -1.076e+00  2.599e+03   0.000    1.000
## Moran                           NA         NA      NA       NA
## Ratake                  -1.106e+00  9.004e-02 -12.277  < 2e-16 ***
## Ratake_cmplx            -1.843e+01  2.062e+03  -0.009    0.993
## Rogert                   1.090e+09  9.693e+11   0.001    0.999
## Supervisor_Limber_cmplx         NA         NA      NA       NA
## Troutville               4.583e+09  1.892e+12   0.002    0.998
## Unspecified             -5.341e+09  2.576e+12  -0.002    0.998
## Vanet                           NA         NA      NA       NA
## Wetmore                 -4.328e-02  8.102e-02  -0.534    0.593
## Bouldery_ext            -2.613e+08  1.704e+12   0.000    1.000
## Rock_Land                2.336e-01  5.693e+02   0.000    1.000
## Rock_Land_cmplx         -1.980e+01  2.062e+03  -0.010    0.992
## Rock_Outcrop                    NA         NA      NA       NA
## Rock_Outcrop_cmplx      -1.814e+01  2.062e+03  -0.009    0.993
## Rubbly                          NA         NA      NA       NA
## Stony                           NA         NA      NA       NA
## Stony_extreme                   NA         NA      NA       NA
## Stony_very                      NA         NA      NA       NA
## Till_Substratum                 NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 109294  on 406708  degrees of freedom
## Residual deviance:  57677  on 406652  degrees of freedom
## AIC: 57791
##
## Number of Fisher Scoring iterations: 22
```

```
curTime=Sys.time()
print(paste("DouglasFir Individual Logistic Model Calculation completed at",curTime))
```

```
## [1] "DouglasFir Individual Logistic Model Calculation completed at 2018-08-12 18:18:55"
```

```
#table(forestTest$Rock_Land)
# Get the following error with above code:
#  Error in table(SpfFir_test$Rock_Land) : object 'SpfFir_test' not found
#    Calls: <Anonymous> ... withCallingHandlers -> withVisible -> eval -> table
```

Predict Douglas Fir Logistic Model Probabilities - All Aggregated Vars

Douglas Fir Probabilities - All Aggregated Data

Predict the probability of Douglas Fir for aggregated Data - all variables.

```
# Predict Douglas Fir Agg Data - all variables

  DougFir_Agg_Train_predict= predict(DougFir_Agg_LogMod, type="response")
  DougFir_Agg_Train_Logit= predict(DougFir_Agg_LogMod)
  summary(DougFir_Agg_Train_predict)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.000000 0.000000 0.000000 0.029891 0.005705 1.000000
```

```
str(DougFir_Agg_Train_predict)
```

```
##  Named num [1:406709] 4.61e-09 3.95e-09 5.92e-10 4.31e-09 2.85e-09 ...
##  - attr(*, "names")= chr [1:406709] "1" "2" "3" "4" ...
```

```
  #plot(table(DougFir_Agg_Train_predict))
  #plot(table(DougFir_Agg_Train_Logit))
  dens<-data.frame(table(DougFir_Agg_Train_predict))
# str(dens)

  DougFir_Agg_Test_predict= predict(DougFir_Agg_LogMod, type="response",newdata=forestTest)
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =
## ifelse(type == : prediction from a rank-deficient fit may be misleading
```

```
summary(DougFir_Agg_Test_predict)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00000 0.00000 0.02980 0.00566 0.99918
```

```
 str(DougFir_Agg_Test_predict)
```

```
##  Named num [1:174303] 8.44e-10 5.42e-09 4.51e-10 6.63e-10 7.70e-08 ...
##  - attr(*, "names")= chr [1:174303] "1" "2" "3" "4" ...
```

Douglas Fir Probabilities - All Individuated Data

Predict the probability of Douglas Fir for Individual Data - all variables.

```
DougFir_Ind_Train_predict= predict(DougFir_Ind_LogMod, type="response")
summary(DougFir_Ind_Train_predict)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.000000 0.000000 0.000000 0.029892 0.005705 1.000000
```

```
DougFir_Ind_Test_predict= predict(DougFir_Ind_LogMod, type="response",newdata=forestTest)
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =
## ifelse(type == : prediction from a rank-deficient fit may be misleading
```

```
summary(DougFir_Ind_Test_predict)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00000 0.00000 0.02980 0.00566 0.99918
```

Douglas Fir Receiver Operating Characteristic (ROC) - All Vars

Douglas Fir Receiver ROC - All Aggregated Data

Next, look at the True Positive and False Positive rates based on threshold value for the aggregated data.

```
if (calcROC) {
  curTime=Sys.time()
  print(paste("ROC graph 1 started at",curTime))

  ROCpred_DougFir_Agg = prediction(DougFir_Agg_Train_predict, forestTrain$DouglasFir)
  summary(ROCpred_DougFir_Agg)
  ROCperf_DougFir_Agg = performance(ROCpred_DougFir_Agg, "tpr", "fpr")
```

```
    summary(ROCperf_DougFir_Agg)

    DougFir_Agg_All_ROC_AUC = as.numeric(performance(ROCpred_DougFir_Agg, "auc")@y.values)
    DougFir_Agg_All_ROC_AUC=as.integer(as.numeric(DougFir_Agg_All_ROC_AUC)*1000)/10
    print(paste("DougFir_Agg_All_ROC_AUC=",DougFir_Agg_All_ROC_AUC))

    jpeg(filename="Fig-ROCR_perf_DougFir_Agg.jpg")
    plot(ROCperf_DougFir_Agg, colorize=TRUE, print.cutoffs.at=seq(0,1,0.1), text.adj=c(-0.2,1.7))
    dev.off()
  } else {
    DougFir_Agg_All_ROC_AUC = 84.2
  }
```

```
## [1] "ROC graph 1 started at 2018-08-12 18:19:01"
## [1] "DougFir_Agg_All_ROC_AUC= 96.4"
```

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

Douglas Fir Receiver ROC - All Individuated Data

The Response Operating Curve for the individuated data is shown below.

```
  if (calcROC) {
    curTime=Sys.time()
    print(paste("ROCR graph 2 started at",curTime))

    ROCpred_DougFir_Ind = prediction(DougFir_Ind_Train_predict, forestTrain$DouglasFir)
    summary(ROCpred_DougFir_Ind)
    ROCperf_DougFir_Ind = performance(ROCpred_DougFir_Ind, "tpr", "fpr")
    summary(ROCperf_DougFir_Ind)

    DougFir_Ind_All_ROC_AUC = as.numeric(performance(ROCpred_DougFir_Ind, "auc")@y.values)
    DougFir_Ind_All_ROC_AUC=as.integer(as.numeric(DougFir_Ind_All_ROC_AUC)*1000)/10
    print(paste("DougFir_Ind_All_ROC_AUC=",DougFir_Ind_All_ROC_AUC))

    jpeg(filename="Fig-ROCR_perf_DougFir_Ind.jpg")
    plot(ROCperf_DougFir_Ind, colorize=TRUE, print.cutoffs.at=seq(0,1,0.1), text.adj=c(-0.2,1.7))
    dev.off()
  } else {
    DougFir_Ind_All_ROC_AUC = 84.2
  }
```

```
## [1] "ROCR graph 2 started at 2018-08-12 18:21:57"
## [1] "DougFir_Ind_All_ROC_AUC= 96.4"
```

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

The threshold graphs are essentially identical. This is making me think that there is not much difference between the two models. The AIC score for the Soil Type model is AIC: 351676 and for the individuated variables is: AIC: 351839. The Soil type model AIC score is 0.046% better than the individuated model.

```
  curTime=Sys.time()
  print(paste("ROCR graph 2 completed at",curTime))
```

```
## [1] "ROCR graph 2 completed at 2018-08-12 18:24:46"
```

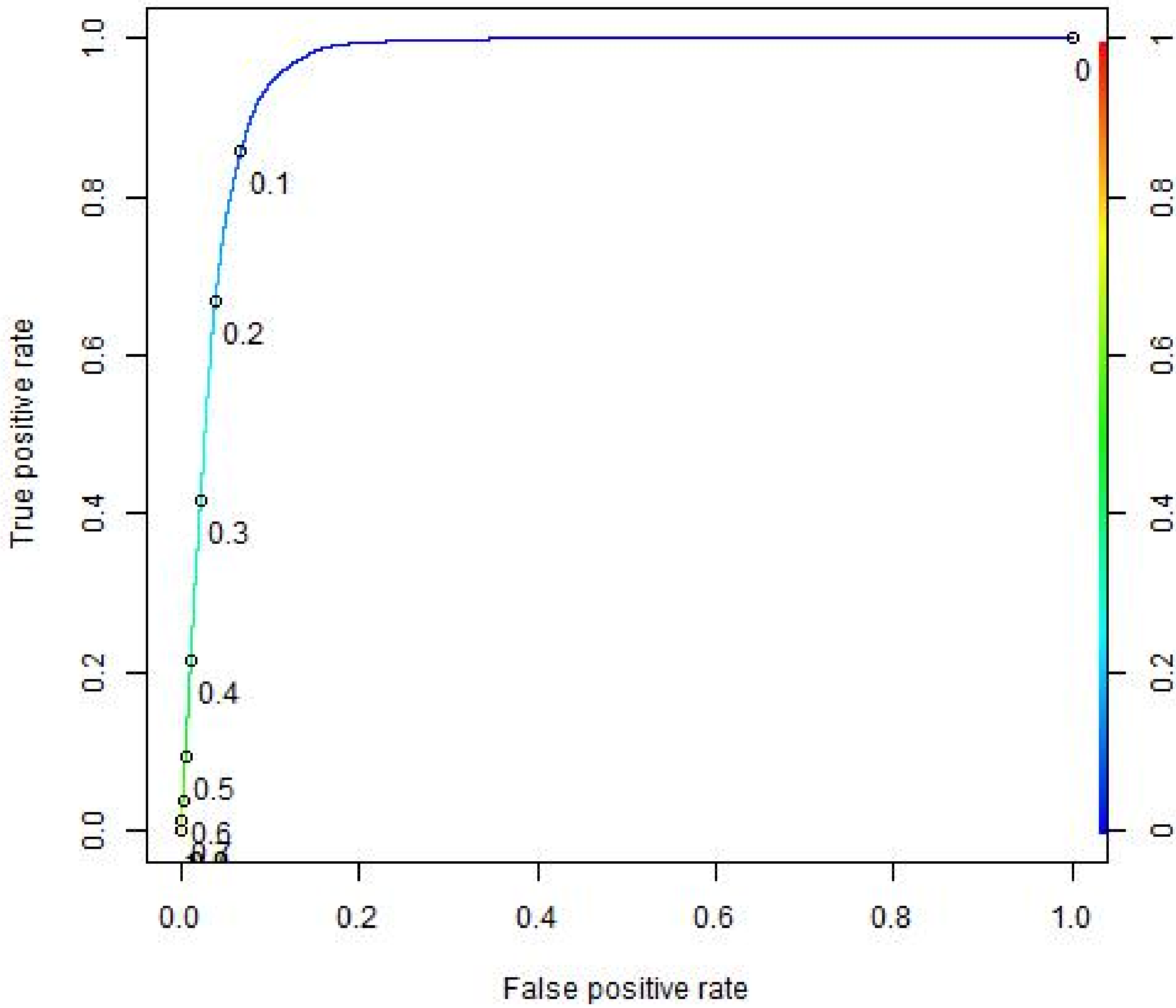

Figure 1: Douglas Fir ROC for All Aggregated Data

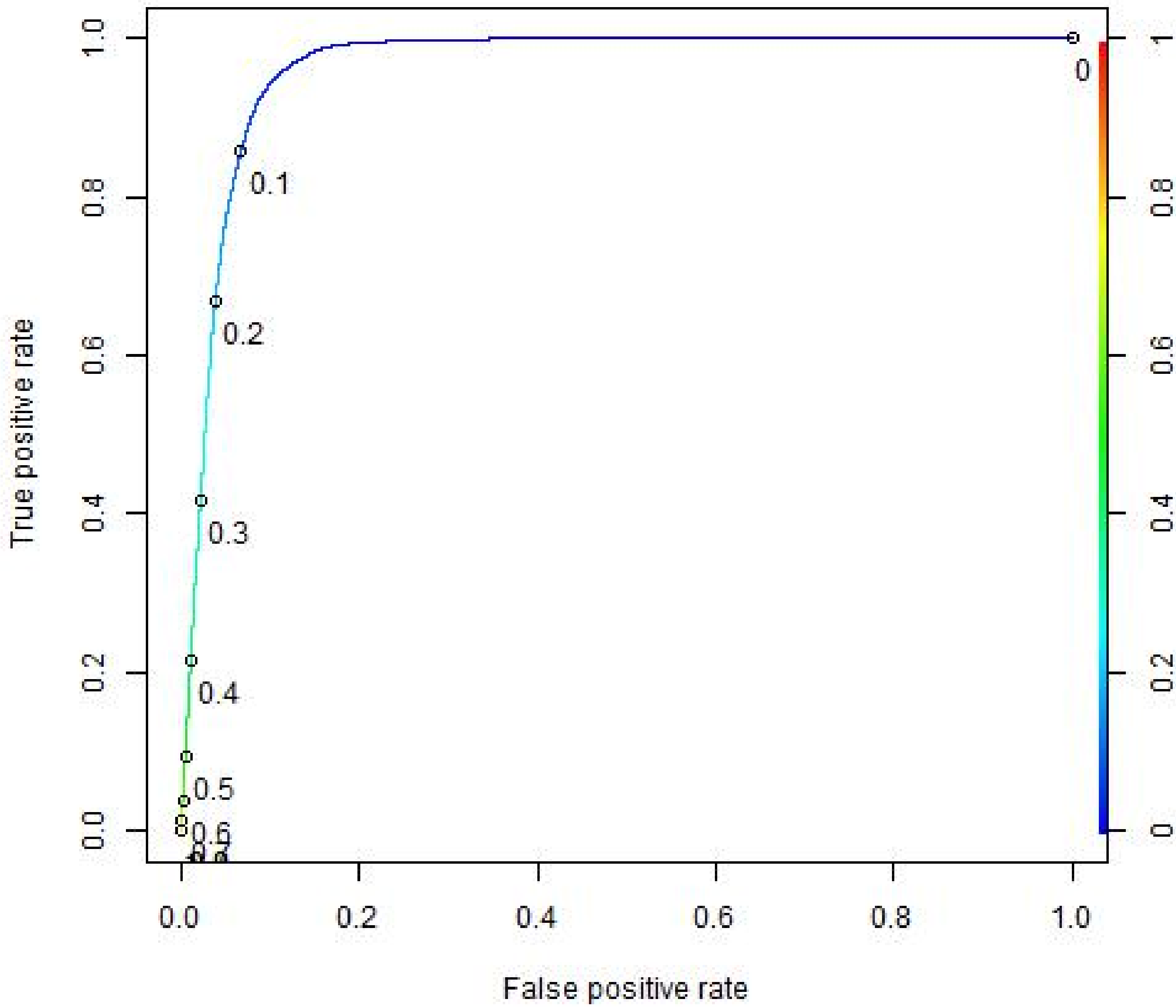

Figure 2: Douglas Fir ROC for All Individuated Data

Calculate Accuracy of Douglas Fir Logisitic Models - All Vars

Calculate Douglas Fir Aggregated Data Logisitic Model Accuracy - All Vars

Find best threshold for Douglas Fir using all aggregated data.

```
result = calcLogisticModelAccuracy (forestTrain$DouglasFir, DougFir_Agg_Train_predict,
                        0.0, 1, 10, "DouglasFir", "Other", 1,1)
```

```
## [1] "Searching for threshold producing best Sensitivity_Specificity"
## [1] "start= 0 end= 1 inc= 0.1"
## [1] "Thresh=0, Accuracy=2.9%, BaseAcc(Other)=97%, Sens=100%, Spec=0%, Sens^2+Spec^2=-2"
## [1] "Thresh=0.1, Accuracy=93%, BaseAcc(Other)=97%, Sens=85.7%, Spec=93.3%, Sens^2+Spec^2=1.606"
## [1] "Thresh=0.2, Accuracy=95.3%, BaseAcc(Other)=97%, Sens=66.8%, Spec=96.2%, Sens^2+Spec^2=1.372"
## [1] "Thresh=0.3, Accuracy=96.1%, BaseAcc(Other)=97%, Sens=41.7%, Spec=97.8%, Sens^2+Spec^2=1.131"
## [1] "Thresh=0.4, Accuracy=96.6%, BaseAcc(Other)=97%, Sens=21.5%, Spec=98.9%, Sens^2+Spec^2=1.025"
## [1] "Thresh=0.5, Accuracy=96.8%, BaseAcc(Other)=97%, Sens=9.4%, Spec=99.5%, Sens^2+Spec^2=0.999"
## [1] "Thresh=0.6, Accuracy=96.9%, BaseAcc(Other)=97%, Sens=3.7%, Spec=99.8%, Sens^2+Spec^2=0.998"
## [1] "Thresh=0.7, Accuracy=97%, BaseAcc(Other)=97%, Sens=1.3%, Spec=99.9%, Sens^2+Spec^2=0.999"
## [1] "Thresh=0.8, Accuracy=97%, BaseAcc(Other)=97%, Sens=0.1%, Spec=99.9%, Sens^2+Spec^2=0.999"
## [1] "Thresh=0.9, Accuracy=97%, BaseAcc(Other)=97%, Sens=0%, Spec=99.9%, Sens^2+Spec^2=0.999"
## [1] "Thresh=1, Accuracy=97%, BaseAcc(Other)=97%, Sens=0%, Spec=100%, Sens^2+Spec^2=-2"
## [1] "Best Sensitivity_Specificity threshold= 0.1 inc= 0.1"
## [1] "========================================"
## [1] "start= 0 end= 0.2 inc= 0.01"
## [1] "Thresh=0, Accuracy=2.9%, BaseAcc(Other)=97%, Sens=100%, Spec=0%, Sens^2+Spec^2=-2"
## [1] "Thresh=0.01, Accuracy=82%, BaseAcc(Other)=97%, Sens=99.2%, Spec=81.5%, Sens^2+Spec^2=1.651"
## [1] "Thresh=0.02, Accuracy=86.2%, BaseAcc(Other)=97%, Sens=97.9%, Spec=85.9%, Sens^2+Spec^2=1.697"
## [1] "Thresh=0.03, Accuracy=88.3%, BaseAcc(Other)=97%, Sens=96.2%, Spec=88.1%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.04, Accuracy=89.7%, BaseAcc(Other)=97%, Sens=94.8%, Spec=89.6%, Sens^2+Spec^2=1.703"
## [1] "Thresh=0.05, Accuracy=90.7%, BaseAcc(Other)=97%, Sens=93.4%, Spec=90.6%, Sens^2+Spec^2=1.695"
## [1] "Thresh=0.06, Accuracy=91.3%, BaseAcc(Other)=97%, Sens=92.1%, Spec=91.3%, Sens^2+Spec^2=1.683"
## [1] "Thresh=0.07, Accuracy=91.8%, BaseAcc(Other)=97%, Sens=90.7%, Spec=91.9%, Sens^2+Spec^2=1.667"
## [1] "Thresh=0.08, Accuracy=92.3%, BaseAcc(Other)=97%, Sens=89.2%, Spec=92.4%, Sens^2+Spec^2=1.651"
## [1] "Thresh=0.09, Accuracy=92.7%, BaseAcc(Other)=97%, Sens=87.5%, Spec=92.8%, Sens^2+Spec^2=1.628"
## [1] "Thresh=0.1, Accuracy=93%, BaseAcc(Other)=97%, Sens=85.7%, Spec=93.3%, Sens^2+Spec^2=1.606"
## [1] "Thresh=0.11, Accuracy=93.3%, BaseAcc(Other)=97%, Sens=84.1%, Spec=93.6%, Sens^2+Spec^2=1.585"
## [1] "Thresh=0.12, Accuracy=93.6%, BaseAcc(Other)=97%, Sens=82.4%, Spec=94%, Sens^2+Spec^2=1.563"
## [1] "Thresh=0.13, Accuracy=93.9%, BaseAcc(Other)=97%, Sens=80.8%, Spec=94.3%, Sens^2+Spec^2=1.544"
## [1] "Thresh=0.14, Accuracy=94.2%, BaseAcc(Other)=97%, Sens=78.9%, Spec=94.7%, Sens^2+Spec^2=1.52"
## [1] "Thresh=0.15, Accuracy=94.4%, BaseAcc(Other)=97%, Sens=77.1%, Spec=95%, Sens^2+Spec^2=1.498"
## [1] "Thresh=0.16, Accuracy=94.6%, BaseAcc(Other)=97%, Sens=75.1%, Spec=95.2%, Sens^2+Spec^2=1.472"
## [1] "Thresh=0.17, Accuracy=94.8%, BaseAcc(Other)=97%, Sens=73.2%, Spec=95.5%, Sens^2+Spec^2=1.449"
## [1] "Thresh=0.18, Accuracy=95%, BaseAcc(Other)=97%, Sens=71.2%, Spec=95.7%, Sens^2+Spec^2=1.424"
## [1] "Thresh=0.19, Accuracy=95.1%, BaseAcc(Other)=97%, Sens=69.2%, Spec=95.9%, Sens^2+Spec^2=1.4"
## [1] "Best Sensitivity_Specificity threshold= 0.03 inc= 0.01"
## [1] "========================================"
## [1] "start= 0.02 end= 0.04 inc= 0.001"
## [1] "Thresh=0.02, Accuracy=86.2%, BaseAcc(Other)=97%, Sens=97.9%, Spec=85.9%, Sens^2+Spec^2=1.697"
## [1] "Thresh=0.021, Accuracy=86.5%, BaseAcc(Other)=97%, Sens=97.6%, Spec=86.1%, Sens^2+Spec^2=1.697"
## [1] "Thresh=0.022, Accuracy=86.7%, BaseAcc(Other)=97%, Sens=97.5%, Spec=86.4%, Sens^2+Spec^2=1.698"
## [1] "Thresh=0.023, Accuracy=87%, BaseAcc(Other)=97%, Sens=97.3%, Spec=86.6%, Sens^2+Spec^2=1.699"
## [1] "Thresh=0.024, Accuracy=87.2%, BaseAcc(Other)=97%, Sens=97.1%, Spec=86.9%, Sens^2+Spec^2=1.7"
## [1] "Thresh=0.025, Accuracy=87.4%, BaseAcc(Other)=97%, Sens=97%, Spec=87.1%, Sens^2+Spec^2=1.702"
## [1] "Thresh=0.026, Accuracy=87.6%, BaseAcc(Other)=97%, Sens=96.9%, Spec=87.3%, Sens^2+Spec^2=1.702"
```

```
## [1] "Thresh=0.027, Accuracy=87.8%, BaseAcc(Other)=97%, Sens=96.7%, Spec=87.5%, Sens^2+Spec^2=1.703"
## [1] "Thresh=0.028, Accuracy=88%, BaseAcc(Other)=97%, Sens=96.6%, Spec=87.7%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.029, Accuracy=88.2%, BaseAcc(Other)=97%, Sens=96.4%, Spec=87.9%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.03, Accuracy=88.3%, BaseAcc(Other)=97%, Sens=96.2%, Spec=88.1%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.031, Accuracy=88.5%, BaseAcc(Other)=97%, Sens=96.1%, Spec=88.3%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.032, Accuracy=88.7%, BaseAcc(Other)=97%, Sens=96%, Spec=88.4%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.033, Accuracy=88.8%, BaseAcc(Other)=97%, Sens=95.8%, Spec=88.6%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.034, Accuracy=89%, BaseAcc(Other)=97%, Sens=95.7%, Spec=88.8%, Sens^2+Spec^2=1.706"
## [1] "Thresh=0.035, Accuracy=89.1%, BaseAcc(Other)=97%, Sens=95.6%, Spec=88.9%, Sens^2+Spec^2=1.706"
## [1] "Thresh=0.036, Accuracy=89.2%, BaseAcc(Other)=97%, Sens=95.4%, Spec=89%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.037, Accuracy=89.4%, BaseAcc(Other)=97%, Sens=95.3%, Spec=89.2%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.038, Accuracy=89.5%, BaseAcc(Other)=97%, Sens=95.1%, Spec=89.3%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.039, Accuracy=89.6%, BaseAcc(Other)=97%, Sens=95%, Spec=89.5%, Sens^2+Spec^2=1.704"
## [1] "========================================"
## [1] "Best Threshold=0.035"
## [1] "Best Sensitivity_Specificity=1.7067867706601"
```

```r
curThresh = as.numeric(result[bestThreshIndex])
DougFir_Agg_All_threshold = curThresh
```

The accuracy for the best threshold on the training set for Douglas Fir using all aggregated data is shown below.

```r
result = calcLogisticModelAccuracy (forestTrain$DouglasFir, DougFir_Agg_Train_predict,
                        curThresh, curThresh, 1, "DouglasFir", "Other", 3)
```

```
## [1] "Model Performance for threshold= 0.035"
## [1] "predicted performance="
##                       Predicted
## Actual                 FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other           351007 (TN)         43545 (FP)
##   1=Actual:DouglasFir      526 (FN)            11631 (TP)
## [1] "Sensitivity= 0.956732746565765 (True positive rate of DouglasFir = TP/(TP+FN) = 11631 /( 11631 +
## [1] "Specificity= 0.889634319430645 (True negative rate of Other = TN/(TN+FP) = 351007 /( 351007 + 4
## [1] "Sens^2+Spec^2=1.706"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.891639"
```

```r
result = calcLogisticModelAccuracy (forestTrain$DouglasFir, DougFir_Agg_Train_predict,
                        0.0, 0.1, 10, "DouglasFir", "Other", 2)
```

```
## [1] "----------"
## [1] "Model Performance for threshold= 0"
## [1] "predicted performance="
##                       Predicted
## Actual                 FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other           0 (TN)              394552 (FP)
##   1=Actual:DouglasFir      0 (FN)              12157 (TP)
## [1] "Sensitivity= 1 (True positive rate of DouglasFir"
## [1] "Specificity= 0 (True negative rate of Other"
## [1] "Sens^2+Spec^2=-2"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.029891"
## [1] "----------"
## [1] "Model Performance for threshold= 0.01"
## [1] "predicted performance="
```

```
##                     Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          321820 (TN)         72732 (FP)
##   1=Actual:DouglasFir     87 (FN)             12070 (TP)
## [1] "Sensitivity= 0.992843629184832 (True positive rate of DouglasFir"
## [1] "Specificity= 0.815659279385227 (True negative rate of Other"
## [1] "Sens^2+Spec^2=1.651"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.820955"
## [1] "----------"
## [1] "Model Performance for threshold= 0.02"
## [1] "predicted performance="
##                     Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          339079 (TN)         55473 (FP)
##   1=Actual:DouglasFir     255 (FN)            11902 (TP)
## [1] "Sensitivity= 0.979024430369335 (True positive rate of DouglasFir"
## [1] "Specificity= 0.859402562906791 (True negative rate of Other"
## [1] "Sens^2+Spec^2=1.697"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.862978"
## [1] "----------"
## [1] "Model Performance for threshold= 0.03"
## [1] "predicted performance="
##                     Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          347793 (TN)         46759 (FP)
##   1=Actual:DouglasFir     451 (FN)            11706 (TP)
## [1] "Sensitivity= 0.962902031751254 (True positive rate of DouglasFir"
## [1] "Specificity= 0.881488371621485 (True negative rate of Other"
## [1] "Sens^2+Spec^2=1.704"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.883921"
## [1] "----------"
## [1] "Model Performance for threshold= 0.04"
## [1] "predicted performance="
##                     Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          353639 (TN)         40913 (FP)
##   1=Actual:DouglasFir     623 (FN)            11534 (TP)
## [1] "Sensitivity= 0.948753804392531 (True positive rate of DouglasFir"
## [1] "Specificity= 0.896305176503984 (True negative rate of Other"
## [1] "Sens^2+Spec^2=1.703"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.897872"
## [1] "----------"
## [1] "Model Performance for threshold= 0.05"
## [1] "predicted performance="
##                     Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          357579 (TN)         36973 (FP)
##   1=Actual:DouglasFir     791 (FN)            11366 (TP)
## [1] "Sensitivity= 0.934934605577034 (True positive rate of DouglasFir"
## [1] "Specificity= 0.906291185952675 (True negative rate of Other"
```

```
## [1] "Sens^2+Spec^2=1.695"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.907147"
## [1] "----------"
## [1] "Model Performance for threshold= 0.06"
## [1] "predicted performance="
##                      Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          360435 (TN)         34117 (FP)
##   1=Actual:DouglasFir     958 (FN)            11199 (TP)
## [1] "Sensitivity= 0.921197663897343 (True positive rate of DouglasFir"
## [1] "Specificity= 0.913529775542894 (True negative rate of Other"
## [1] "Sens^2+Spec^2=1.683"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.913758"
## [1] "----------"
## [1] "Model Performance for threshold= 0.07"
## [1] "predicted performance="
##                      Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          362702 (TN)         31850 (FP)
##   1=Actual:DouglasFir     1129 (FN)           11028 (TP)
## [1] "Sensitivity= 0.907131693674426 (True positive rate of DouglasFir"
## [1] "Specificity= 0.919275532756139 (True negative rate of Other"
## [1] "Sens^2+Spec^2=1.667"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.918912"
## [1] "----------"
## [1] "Model Performance for threshold= 0.08"
## [1] "predicted performance="
##                      Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          364747 (TN)         29805 (FP)
##   1=Actual:DouglasFir     1304 (FN)           10853 (TP)
## [1] "Sensitivity= 0.892736694908283 (True positive rate of DouglasFir"
## [1] "Specificity= 0.924458626492832 (True negative rate of Other"
## [1] "Sens^2+Spec^2=1.651"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.92351"
## [1] "----------"
## [1] "Model Performance for threshold= 0.09"
## [1] "predicted performance="
##                      Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          366526 (TN)         28026 (FP)
##   1=Actual:DouglasFir     1518 (FN)           10639 (TP)
## [1] "Sensitivity= 0.875133667845686 (True positive rate of DouglasFir"
## [1] "Specificity= 0.928967537865731 (True negative rate of Other"
## [1] "Sens^2+Spec^2=1.628"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.927358"
## [1] "----------"
## [1] "Model Performance for threshold= 0.1"
## [1] "predicted performance="
```

```
##                        Predicted
## Actual                 FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          368121 (TN)         26431 (FP)
##   1=Actual:DouglasFir     1730 (FN)           10427 (TP)
## [1] "Sensitivity= 0.857695155054701 (True positive rate of DouglasFir"
## [1] "Specificity= 0.933010097528336 (True negative rate of Other"
## [1] "Sens^2+Spec^2=1.606"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.930758"
```

The accuracy for the best threshold on the testing set for Douglas Fir using all aggregated data is shown below.

```
result = calcLogisticModelAccuracy (forestTest$DouglasFir, DougFir_Agg_Test_predict,
                        curThresh, curThresh, 1, "DouglasFir", "Other", 3,
                        saveFile=saveFileName, desc="Douglas Fir All Aggregate Vars",
                        AIC=DougFir_Agg_All_aic, AUC=DougFir_Agg_All_ROC_AUC)
```

```
## [1] "Model Performance for threshold= 0.035"
## [1] "predicted performance="
##                        Predicted
## Actual                 FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          150473 (TN)         18620 (FP)
##   1=Actual:DouglasFir     226 (FN)            4984 (TP)
## [1] "Sensitivity= 0.956621880998081 (True positive rate of DouglasFir = TP/(TP+FN) = 4984 /( 4984 + 2
## [1] "Specificity= 0.88988308209092 (True negative rate of Other = TN/(TN+FP) = 150473 /( 150473 + 186
## [1] "Sens^2+Spec^2=1.707"
## [1] "Baseline (Other) Accuracy=0.970109"
## [1] "Logistic Accuracy=0.891877"
```

```
  # retVal = c(modelPerformance, sensitivity,specificity) # TN, FN, FP, TP, sens, spec
  # c(funcStat,accuracy,baseline,retVal)
  list[RC, DougFir_Agg_All_model_acc, DougFir_Agg_All_baseline_acc,
      TN, FN, FP, TP, DougFir_Agg_All_sens, DougFir_Agg_All_spec] <- result
  if (RC != "OK") {
    print(paste("Error - terminating:",RC))
    knitr:knit_exit()
  }
  DougFir_Agg_All_model_acc = as.integer(as.numeric(DougFir_Agg_All_model_acc)*1000)/10
  DougFir_Agg_All_baseline_acc = as.integer(as.numeric(DougFir_Agg_All_baseline_acc)*1000)/10
  DougFir_Agg_All_sens = as.integer(as.numeric(DougFir_Agg_All_sens)*1000)/10
  DougFir_Agg_All_spec = as.integer(as.numeric(DougFir_Agg_All_spec)*1000)/10
```

Calculate Douglas Fir Individuated Data Logisitic Model Accuracy - All Vars

Find best threshold for Douglas Fir using all individuated data.

```
result = calcLogisticModelAccuracy (forestTrain$DouglasFir, DougFir_Ind_Train_predict,
                        0.0, 1, 10, "DouglasFir", "Other", 1,1)
```

```
## [1] "Searching for threshold producing best Sensitivity_Specificity"
## [1] "start= 0 end= 1 inc= 0.1"
## [1] "Thresh=0, Accuracy=2.9%, BaseAcc(Other)=97%, Sens=100%, Spec=0%, Sens^2+Spec^2=-2"
## [1] "Thresh=0.1, Accuracy=93%, BaseAcc(Other)=97%, Sens=85.7%, Spec=93.3%, Sens^2+Spec^2=1.606"
## [1] "Thresh=0.2, Accuracy=95.3%, BaseAcc(Other)=97%, Sens=66.8%, Spec=96.1%, Sens^2+Spec^2=1.372"
## [1] "Thresh=0.3, Accuracy=96.1%, BaseAcc(Other)=97%, Sens=41.7%, Spec=97.8%, Sens^2+Spec^2=1.132"
```

```
## [1] "Thresh=0.4, Accuracy=96.6%, BaseAcc(Other)=97%, Sens=21.5%, Spec=98.9%, Sens^2+Spec^2=1.025"
## [1] "Thresh=0.5, Accuracy=96.8%, BaseAcc(Other)=97%, Sens=9.4%, Spec=99.5%, Sens^2+Spec^2=0.999"
## [1] "Thresh=0.6, Accuracy=96.9%, BaseAcc(Other)=97%, Sens=3.7%, Spec=99.8%, Sens^2+Spec^2=0.998"
## [1] "Thresh=0.7, Accuracy=97%, BaseAcc(Other)=97%, Sens=1.3%, Spec=99.9%, Sens^2+Spec^2=0.999"
## [1] "Thresh=0.8, Accuracy=97%, BaseAcc(Other)=97%, Sens=0.1%, Spec=99.9%, Sens^2+Spec^2=0.999"
## [1] "Thresh=0.9, Accuracy=97%, BaseAcc(Other)=97%, Sens=0%, Spec=99.9%, Sens^2+Spec^2=0.999"
## [1] "Thresh=1, Accuracy=97%, BaseAcc(Other)=97%, Sens=0%, Spec=100%, Sens^2+Spec^2=-2"
## [1] "Best Sensitivity_Specificity threshold= 0.1 inc= 0.1"
## [1] "========================================"
## [1] "start= 0 end= 0.2 inc= 0.01"
## [1] "Thresh=0, Accuracy=2.9%, BaseAcc(Other)=97%, Sens=100%, Spec=0%, Sens^2+Spec^2=-2"
## [1] "Thresh=0.01, Accuracy=82%, BaseAcc(Other)=97%, Sens=99.2%, Spec=81.5%, Sens^2+Spec^2=1.651"
## [1] "Thresh=0.02, Accuracy=86.2%, BaseAcc(Other)=97%, Sens=97.9%, Spec=85.9%, Sens^2+Spec^2=1.697"
## [1] "Thresh=0.03, Accuracy=88.3%, BaseAcc(Other)=97%, Sens=96.2%, Spec=88.1%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.04, Accuracy=89.7%, BaseAcc(Other)=97%, Sens=94.8%, Spec=89.6%, Sens^2+Spec^2=1.703"
## [1] "Thresh=0.05, Accuracy=90.7%, BaseAcc(Other)=97%, Sens=93.4%, Spec=90.6%, Sens^2+Spec^2=1.695"
## [1] "Thresh=0.06, Accuracy=91.3%, BaseAcc(Other)=97%, Sens=92.1%, Spec=91.3%, Sens^2+Spec^2=1.683"
## [1] "Thresh=0.07, Accuracy=91.8%, BaseAcc(Other)=97%, Sens=90.7%, Spec=91.9%, Sens^2+Spec^2=1.667"
## [1] "Thresh=0.08, Accuracy=92.3%, BaseAcc(Other)=97%, Sens=89.2%, Spec=92.4%, Sens^2+Spec^2=1.651"
## [1] "Thresh=0.09, Accuracy=92.7%, BaseAcc(Other)=97%, Sens=87.5%, Spec=92.8%, Sens^2+Spec^2=1.628"
## [1] "Thresh=0.1, Accuracy=93%, BaseAcc(Other)=97%, Sens=85.7%, Spec=93.3%, Sens^2+Spec^2=1.606"
## [1] "Thresh=0.11, Accuracy=93.3%, BaseAcc(Other)=97%, Sens=84.1%, Spec=93.6%, Sens^2+Spec^2=1.585"
## [1] "Thresh=0.12, Accuracy=93.6%, BaseAcc(Other)=97%, Sens=82.4%, Spec=94%, Sens^2+Spec^2=1.563"
## [1] "Thresh=0.13, Accuracy=93.9%, BaseAcc(Other)=97%, Sens=80.8%, Spec=94.3%, Sens^2+Spec^2=1.544"
## [1] "Thresh=0.14, Accuracy=94.2%, BaseAcc(Other)=97%, Sens=78.9%, Spec=94.7%, Sens^2+Spec^2=1.52"
## [1] "Thresh=0.15, Accuracy=94.4%, BaseAcc(Other)=97%, Sens=77.1%, Spec=95%, Sens^2+Spec^2=1.498"
## [1] "Thresh=0.16, Accuracy=94.6%, BaseAcc(Other)=97%, Sens=75.1%, Spec=95.2%, Sens^2+Spec^2=1.472"
## [1] "Thresh=0.17, Accuracy=94.8%, BaseAcc(Other)=97%, Sens=73.2%, Spec=95.5%, Sens^2+Spec^2=1.449"
## [1] "Thresh=0.18, Accuracy=95%, BaseAcc(Other)=97%, Sens=71.2%, Spec=95.7%, Sens^2+Spec^2=1.424"
## [1] "Thresh=0.19, Accuracy=95.1%, BaseAcc(Other)=97%, Sens=69.2%, Spec=95.9%, Sens^2+Spec^2=1.4"
## [1] "Best Sensitivity_Specificity threshold= 0.03 inc= 0.01"
## [1] "========================================"
## [1] "start= 0.02 end= 0.04 inc= 0.001"
## [1] "Thresh=0.02, Accuracy=86.2%, BaseAcc(Other)=97%, Sens=97.9%, Spec=85.9%, Sens^2+Spec^2=1.697"
## [1] "Thresh=0.021, Accuracy=86.5%, BaseAcc(Other)=97%, Sens=97.6%, Spec=86.1%, Sens^2+Spec^2=1.697"
## [1] "Thresh=0.022, Accuracy=86.7%, BaseAcc(Other)=97%, Sens=97.5%, Spec=86.4%, Sens^2+Spec^2=1.698"
## [1] "Thresh=0.023, Accuracy=87%, BaseAcc(Other)=97%, Sens=97.3%, Spec=86.6%, Sens^2+Spec^2=1.699"
## [1] "Thresh=0.024, Accuracy=87.2%, BaseAcc(Other)=97%, Sens=97.1%, Spec=86.9%, Sens^2+Spec^2=1.7"
## [1] "Thresh=0.025, Accuracy=87.4%, BaseAcc(Other)=97%, Sens=97%, Spec=87.1%, Sens^2+Spec^2=1.702"
## [1] "Thresh=0.026, Accuracy=87.6%, BaseAcc(Other)=97%, Sens=96.9%, Spec=87.3%, Sens^2+Spec^2=1.702"
## [1] "Thresh=0.027, Accuracy=87.8%, BaseAcc(Other)=97%, Sens=96.7%, Spec=87.5%, Sens^2+Spec^2=1.703"
## [1] "Thresh=0.028, Accuracy=88%, BaseAcc(Other)=97%, Sens=96.6%, Spec=87.7%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.029, Accuracy=88.2%, BaseAcc(Other)=97%, Sens=96.4%, Spec=87.9%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.03, Accuracy=88.3%, BaseAcc(Other)=97%, Sens=96.2%, Spec=88.1%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.031, Accuracy=88.5%, BaseAcc(Other)=97%, Sens=96.1%, Spec=88.3%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.032, Accuracy=88.7%, BaseAcc(Other)=97%, Sens=96%, Spec=88.4%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.033, Accuracy=88.8%, BaseAcc(Other)=97%, Sens=95.8%, Spec=88.6%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.034, Accuracy=89%, BaseAcc(Other)=97%, Sens=95.7%, Spec=88.8%, Sens^2+Spec^2=1.706"
## [1] "Thresh=0.035, Accuracy=89.1%, BaseAcc(Other)=97%, Sens=95.6%, Spec=88.9%, Sens^2+Spec^2=1.706"
## [1] "Thresh=0.036, Accuracy=89.2%, BaseAcc(Other)=97%, Sens=95.4%, Spec=89%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.037, Accuracy=89.4%, BaseAcc(Other)=97%, Sens=95.3%, Spec=89.2%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.038, Accuracy=89.5%, BaseAcc(Other)=97%, Sens=95.1%, Spec=89.3%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.039, Accuracy=89.6%, BaseAcc(Other)=97%, Sens=95%, Spec=89.5%, Sens^2+Spec^2=1.704"
## [1] "========================================"
```

```
## [1] "Best Threshold=0.035"
## [1] "Best Sensitivity_Specificity=1.70677324194118"
```

```
curThresh = as.numeric(result[bestThreshIndex])
DougFir_Ind_All_threshold = curThresh
```

The accuracy for the best threshold on the training set for Douglas Fir using all individuated data is shown below.

```
result = calcLogisticModelAccuracy (forestTrain$DouglasFir, DougFir_Ind_Train_predict,
                       curThresh, curThresh, 1, "DouglasFir", "Other", 3)
```

```
## [1] "Model Performance for threshold= 0.035"
## [1] "predicted performance="
##                      Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          351004 (TN)         43548 (FP)
##   1=Actual:DouglasFir     526 (FN)            11631 (TP)
## [1] "Sensitivity= 0.956732746565765 (True positive rate of DouglasFir = TP/(TP+FN) = 11631 /( 11631 +
## [1] "Specificity= 0.889626715870151 (True negative rate of Other = TN/(TN+FP) = 351004 /( 351004 + 43
## [1] "Sens^2+Spec^2=1.706"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.891632"
```

The accuracy for the best threshold on the testing set for Douglas Fir using all individuated data is shown below.

```
result = calcLogisticModelAccuracy (forestTest$DouglasFir, DougFir_Ind_Test_predict,
                       curThresh, curThresh, 1, "DouglasFir", "Other", 3,
                       saveFile=saveFileName, desc="Douglas Fir All Individualized Vars",
                       AIC=DougFir_Ind_All_aic, AUC=DougFir_Ind_All_ROC_AUC)
```

```
## [1] "Model Performance for threshold= 0.035"
## [1] "predicted performance="
##                      Predicted
## Actual                FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          150472 (TN)         18621 (FP)
##   1=Actual:DouglasFir     226 (FN)            4984 (TP)
## [1] "Sensitivity= 0.956621880998081 (True positive rate of DouglasFir = TP/(TP+FN) = 4984 /( 4984 + 2
## [1] "Specificity= 0.889877168185555 (True negative rate of Other = TN/(TN+FP) = 150472 /( 150472 + 18
## [1] "Sens^2+Spec^2=1.707"
## [1] "Baseline (Other) Accuracy=0.970109"
## [1] "Logistic Accuracy=0.891872"
```

```
list[RC, DougFir_Ind_All_model_acc, DougFir_Ind_All_baseline_acc,
      TN, FN, FP, TP, DougFir_Ind_All_sens, DougFir_Ind_All_spec] <- result
  if (RC != "OK") {
    print(paste("Error - terminating:",RC))
    knitr:knit_exit()
  }
  DougFir_Ind_All_model_acc = as.integer(as.numeric(DougFir_Ind_All_model_acc)*1000)/10
  DougFir_Ind_All_baseline_acc = as.integer(as.numeric(DougFir_Ind_All_baseline_acc)*1000)/10
  DougFir_Ind_All_sens = as.integer(as.numeric(DougFir_Ind_All_sens)*1000)/10
  DougFir_Ind_All_spec = as.integer(as.numeric(DougFir_Ind_All_spec)*1000)/10
```

The Douglas Fir aggregated model accuracy on the test data is 77.15% compared to 77.12% for the individuated data model, essentially identical. Both are ~ 14% better than the baseline model.

Douglas Fir Logistic Regression - Significant Variables

Create Douglas Fir Logistic Model - Sig Vars

Now create the logistic model for the Aggregated Soil data using just the significant variables and compare to the previous models.

Douglas Fir Logistic Model using Significant Aggregated Data

Variables that have been removed are commented out in the code below.

```
DougFir_Agg_LogMod =
  glm(DouglasFir ~
        Elev +     # Elevation in meters of cell
        # Aspect +   # Direction in degrees slope faces
        Slope +    # Slope / steepness of hill in degrees (0 to 90)
        H2OHD +    # Horizontal distance in meters to nearest water
        # H2OVD +    # Vertical distance in meters to nearest water
        # RoadHD +   # Horizontal distance in meters to nearest road
        FirePtHD + # Horizontal distance in meters to nearest fire point
        Shade9AM + Shade12PM + Shade3PM + # Amount of shade at 9am, 12pm and 3pm
        # Wilderness areas:
          # RWwild + NEwild +
          CMwild
          # CPwild +
        # Aggregated Soil type:
          # ST01 + ST02 + ST03 +
          #ST04 +
          # ST05 + ST06 + ST07 +
          #ST08 + ST09 + ST10 + ST11 + ST12 +
          # ST13 + ST14 + ST15 +
          #ST16 + ST17 + ST18 + ST19 + ST20 +
          #ST21 + ST22 + ST23 + ST24 + ST25 + ST26 + ST27 + ST28 + ST29 + ST30 +
          #ST31 + ST32 + ST33 +
          # ST34 + ST35 +
          #ST36 +
          # ST37 +
          #ST38 + ST39
          # + ST40
          ,
        data=forestTrain, family=binomial)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
# save model for later use
DougFir_Agg_Sig_LogMod = DougFir_Agg_LogMod
save("DougFir_Agg_Sig_LogMod", file="DougFir_Agg_Sig_LogMod.Rdata")

DougFir_Agg_Sig_aic<-as.integer(DougFir_Agg_LogMod$aic)
DougFir_Agg_Sig_aic
```

```
## [1] 64213
```

Check the coefficients of the Douglas Fir model using significant aggregated data.

```r
summary(DougFir_Agg_LogMod)
```

```
##
## Call:
## glm(formula = DouglasFir ~ Elev + Slope + H2OHD + FirePtHD +
##     Shade9AM + Shade12PM + Shade3PM + CMwild, family = binomial,
##     data = forestTrain)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -6.2060  -0.1411  -0.0630  -0.0307   4.8780
##
## Coefficients:
##               Estimate Std. Error  z value Pr(>|z|)
## (Intercept) -8.111e+00  9.180e-01   -8.836  < 2e-16 ***
## Elev        -7.340e-03  6.314e-05 -116.258  < 2e-16 ***
## Slope        8.150e-02  5.253e-03   15.516  < 2e-16 ***
## H2OHD       -9.713e-04  7.896e-05  -12.300  < 2e-16 ***
## FirePtHD    -5.030e-05  1.417e-05   -3.549 0.000387 ***
## Shade9AM     1.917e-01  5.636e-03   34.013  < 2e-16 ***
## Shade12PM   -1.926e-01  4.663e-03  -41.311  < 2e-16 ***
## Shade3PM     1.742e-01  4.673e-03   37.278  < 2e-16 ***
## CMwild       1.712e+00  2.908e-02   58.880  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 109294  on 406708  degrees of freedom
## Residual deviance:  64196  on 406700  degrees of freedom
## AIC: 64214
##
## Number of Fisher Scoring iterations: 8
```

The intercept looks much more reasonable. Some soil types that were significant previously are no longer significant.

Douglas Fir Logistic Model using Significant Individuated Data

Create a logistic model for the significant individuated variables.

Again, the non-significant variables have been commented out.

```r
  DougFir_Ind_LogMod =
    glm(DouglasFir ~
          Elev +     # Elevation in meters of cell
          # Aspect +   # Direction in degrees slope faces
          Slope +    # Slope / steepness of hill in degrees (0 to 90)
          H2OHD +    # Horizontal distance in meters to nearest water
          # H2OVD +    # Vertical distance in meters to nearest water
          # RoadHD +   # Horizontal distance in meters to nearest road
          FirePtHD + # Horizontal distance in meters to nearest fire point
          Shade9AM + Shade12PM + Shade3PM + # Amount of shade at 9am, 12pm and 3pm
          # Wilderness areas:
            # RWwild + NEwild +
```

```
          # CMwild +
          # CPwild +
        # Climate Zone:
        # ClimateName +
          # Montane_low + Montane +
          # SubAlpine + Alpine +
          # Dry + Non_Dry +
        # Geology Zone:
        # GeoName +
          # Alluvium + Glacial +
          # Sed_mix + Ign_Meta +
        # Soil Family:
          # Aquolis_cmplx +
          # Argiborolis_Pachic +
          # Borohemists_cmplx + Bross +
          # Bullwark + Bullwark_Cmplx + Catamount +
          Catamount_cmplx +
          # Cathedral + Como +
          # Cryaquepts_cmplx + Cryaquepts_Typic + Cryaquolls +
          # Cryaquolls_cmplx + Cryaquolls_Typic + Cryaquolls_Typic_cmplx +
          # Cryoborolis_cmplx +
          # Cryorthents +
          # Cryorthents_cmplx + Cryumbrepts + Cryumbrepts_cmplx + Gateview +
          # Gothic + Granile +
          Haploborolis +
          # Legault +
          # Legault_cmplx +
          # Leighcan + Leighcan_cmplx + Leighcan_warm +
          # Moran +
          Ratake
          # Ratake_cmplx + Rogert + Supervisor_Limber_cmplx +
          # Troutville + Unspecified + Vanet + Wetmore +
        # Soil Rock composition:
          # Bouldery_ext +
          # Rock_Land +
          # Rock_Land_cmplx + Rock_Outcrop +
          # Rock_Outcrop_cmplx ,
          # Rubbly + Stony + Stony_extreme + Stony_very + Till_Substratum ,
          ,
        data=forestTrain, family=binomial)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
# save model for later use
DougFir_Ind_Sig_LogMod = DougFir_Ind_LogMod
save("DougFir_Ind_Sig_LogMod", file="DougFir_Ind_Sig_LogMod.Rdata")

DougFir_Ind_Sig_aic<-as.integer(DougFir_Ind_LogMod$aic)
DougFir_Ind_Sig_aic
```

```
## [1] 67703
```

```
summary(DougFir_Ind_LogMod)
```

```
##
```

```
## Call:
## glm(formula = DouglasFir ~ Elev + Slope + H2OHD + FirePtHD +
##     Shade9AM + Shade12PM + Shade3PM + Catamount_cmplx + Haploborolis +
##     Ratake, family = binomial, data = forestTrain)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -6.3027  -0.1607  -0.0912  -0.0550   5.1493
##
## Coefficients:
##                   Estimate Std. Error  z value Pr(>|z|)
## (Intercept)     -1.450e+01  8.986e-01  -16.137   <2e-16 ***
## Elev            -5.424e-03  4.608e-05 -117.712   <2e-16 ***
## Slope            1.045e-01  5.157e-03   20.271   <2e-16 ***
## H2OHD           -1.585e-03  7.940e-05  -19.969   <2e-16 ***
## FirePtHD        -2.185e-04  1.235e-05  -17.695   <2e-16 ***
## Shade9AM         1.992e-01  5.533e-03   36.004   <2e-16 ***
## Shade12PM       -1.904e-01  4.608e-03  -41.329   <2e-16 ***
## Shade3PM         1.762e-01  4.591e-03   38.394   <2e-16 ***
## Catamount_cmplx -1.482e+00  1.418e-01  -10.452   <2e-16 ***
## Haploborolis    -1.347e+00  1.007e-01  -13.375   <2e-16 ***
## Ratake          -1.060e-01  5.814e-02   -1.823   0.0682 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 109294  on 406708  degrees of freedom
## Residual deviance:  67681  on 406698  degrees of freedom
## AIC: 67703
##
## Number of Fisher Scoring iterations: 8
```

Again the intercept looks much better. Also a few variables have become non-significant.

Predict Douglas Fir Logistic Model Probabilities - Sig Vars

Douglas Fir Probabilities using Significant Aggregated Data

Predict the probability of Douglas Fir for aggregated Data - significant variables.

```r
# Predict Douglas Fir Agg Data - significant variables

  DougFir_Agg_Train_predict= predict(DougFir_Agg_LogMod, type="response")
  summary(DougFir_Agg_Train_predict)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.0000000 0.0005692 0.0023602 0.0298912 0.0126581 1.0000000
```

```r
  DougFir_Agg_Test_predict= predict(DougFir_Agg_LogMod, type="response",newdata=forestTest)
  summary(DougFir_Agg_Test_predict)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## 0.0000000 0.0005697 0.0023339 0.0298994 0.0127332 0.9999979
```

Douglas Fir Probabilities using Significant Individuated Data

Predict the probability of DouglasFir using significant Individuated Data.

```
DougFir_Ind_Train_predict= predict(DougFir_Ind_LogMod, type="response")
summary(DougFir_Ind_Train_predict)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.000000 0.001736 0.004661 0.029891 0.015257 1.000000
```

```
DougFir_Ind_Test_predict= predict(DougFir_Ind_LogMod, type="response",newdata=forestTest)
summary(DougFir_Ind_Test_predict)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.000000 0.001741 0.004646 0.029742 0.015294 0.999999
```

```
print(paste("ROCR graph 2 completed at",curTime))
```

```
## [1] "ROCR graph 2 completed at 2018-08-12 18:24:46"
```

Douglas Fir Receiver Operating Characteristic (ROC) - Sig Vars

Look at the True Positive and False Positive rates based on threshold value.

```
if (calcROC) {
  ROCpred_DougFir_Agg = prediction(DougFir_Agg_Train_predict, forestTrain$DouglasFir)
  summary(ROCpred_DougFir_Agg)

  ROCperf_DougFir_Agg = performance(ROCpred_DougFir_Agg, "tpr", "fpr")
  summary(ROCperf_DougFir_Agg)

  DougFir_Agg_Sig_ROC_AUC = as.numeric(performance(ROCpred_DougFir_Agg, "auc")@y.values)
  DougFir_Agg_Sig_ROC_AUC=as.integer(as.numeric(DougFir_Agg_Sig_ROC_AUC)*1000)/10
  DougFir_Agg_Sig_ROC_AUC

  jpeg(filename="Fig-ROCR_perf_DougFir_Agg_Sig.jpg")
  plot(ROCperf_DougFir_Agg, colorize=TRUE, print.cutoffs.at=seq(0,1,0.1), text.adj=c(-0.2,1.7))
  dev.off()
} else {
  DougFir_Agg_Sig_ROC_AUC = 83.7
}
```

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

```
if (calcROC) {
  curTime=Sys.time()
  print(paste("ROCR graph 2 started at",curTime))

  ROCpred_DougFir_Ind = prediction(DougFir_Ind_Train_predict, forestTrain$DouglasFir)
  summary(ROCpred_DougFir_Ind)

  ROCperf_DougFir_Ind = performance(ROCpred_DougFir_Ind, "tpr", "fpr")
  summary(ROCperf_DougFir_Ind)

  DougFir_Ind_Sig_ROC_AUC = as.numeric(performance(ROCpred_DougFir_Ind, "auc")@y.values)
  DougFir_Ind_Sig_ROC_AUC=as.integer(as.numeric(DougFir_Ind_Sig_ROC_AUC)*1000)/10
```

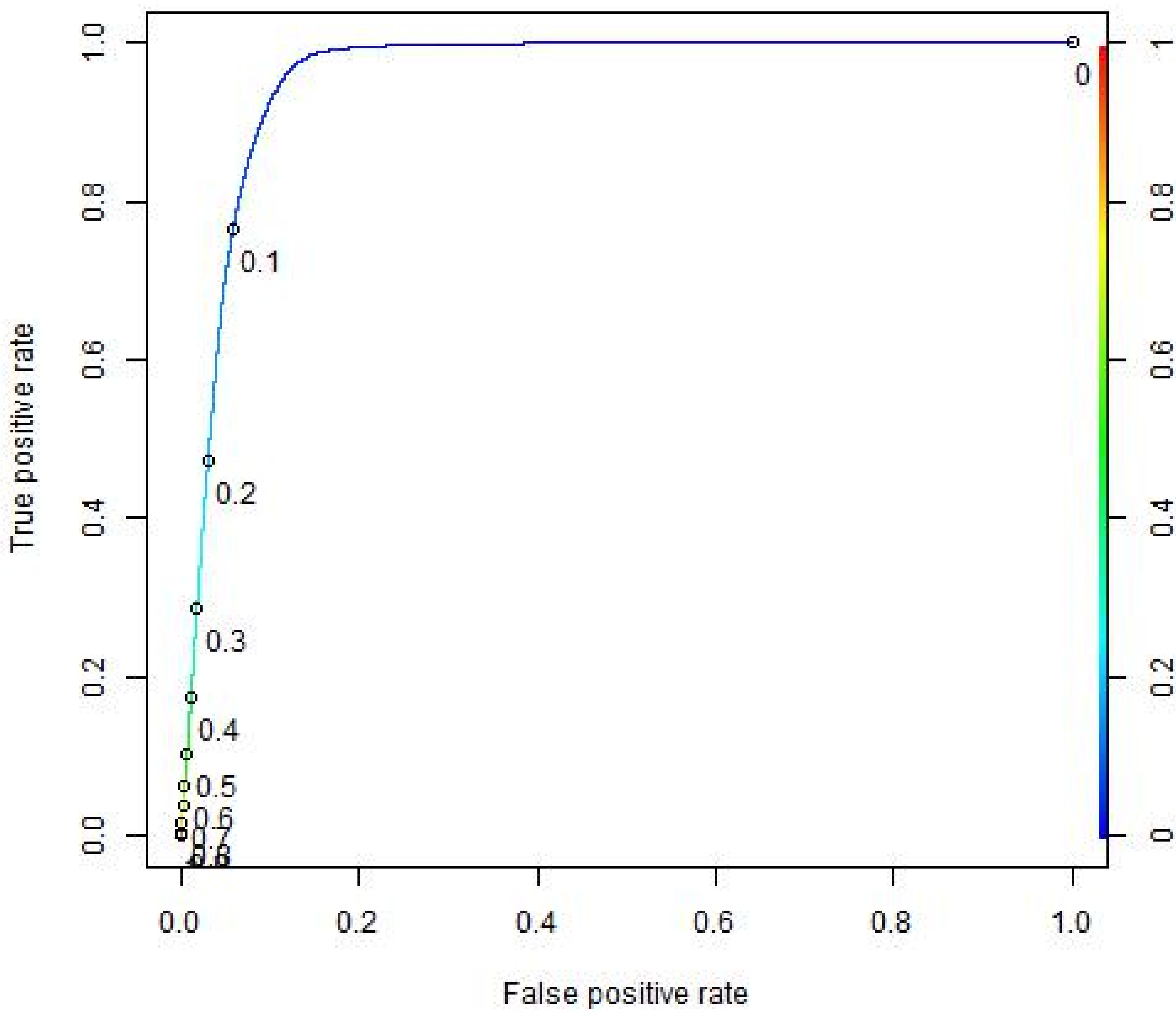

Figure 3: Douglas Fir ROC for Aggregated Significant Data

```r
    DougFir_Ind_Sig_ROC_AUC

    jpeg(filename="Fig-ROC_perf_DougFir_Ind_Sig.jpg")
    plot(ROCperf_DougFir_Ind, colorize=TRUE, print.cutoffs.at=seq(0,1,0.1), text.adj=c(-0.2,1.7))
    dev.off()
  } else {
    DougFir_Ind_Sig_ROC_AUC = 83.8
  }
```

```
## [1] "ROCR graph 2 started at 2018-08-12 18:29:40"
```

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

The threshold graphs are essentially identical. This is making me think that there is not much difference between the two models. The AIC score for the Soil Type model is AIC: 351676 and for the individuated variables is: AIC: 351839. The Soil type model AIC score is 0.046% better than the individuated model.

Calculate Accuracy of Douglas Fir Logisitic Model - Sig Vars

Calculate Douglas Fir Aggregated Data Logisitic Model Accuracy - Significant Vars

Find best Douglas Fir threshold for Aggregated Data using significant variables.

```r
result = calcLogisticModelAccuracy (forestTrain$DouglasFir, DougFir_Agg_Train_predict,
                        0.0, 1, 10, "DouglasFir", "Other", 1,1)
```

```
## [1] "Searching for threshold producing best Sensitivity_Specificity"
## [1] "start= 0 end= 1 inc= 0.1"
## [1] "Thresh=0, Accuracy=2.9%, BaseAcc(Other)=97%, Sens=100%, Spec=0%, Sens^2+Spec^2=-2"
## [1] "Thresh=0.1, Accuracy=93.6%, BaseAcc(Other)=97%, Sens=76.5%, Spec=94.1%, Sens^2+Spec^2=1.473"
## [1] "Thresh=0.2, Accuracy=95.5%, BaseAcc(Other)=97%, Sens=47.2%, Spec=97%, Sens^2+Spec^2=1.164"
## [1] "Thresh=0.3, Accuracy=96.2%, BaseAcc(Other)=97%, Sens=28.5%, Spec=98.2%, Sens^2+Spec^2=1.047"
## [1] "Thresh=0.4, Accuracy=96.4%, BaseAcc(Other)=97%, Sens=17.4%, Spec=98.9%, Sens^2+Spec^2=1.009"
## [1] "Thresh=0.5, Accuracy=96.6%, BaseAcc(Other)=97%, Sens=10.1%, Spec=99.3%, Sens^2+Spec^2=0.997"
## [1] "Thresh=0.6, Accuracy=96.8%, BaseAcc(Other)=97%, Sens=6.3%, Spec=99.6%, Sens^2+Spec^2=0.996"
## [1] "Thresh=0.7, Accuracy=96.9%, BaseAcc(Other)=97%, Sens=3.6%, Spec=99.8%, Sens^2+Spec^2=0.998"
## [1] "Thresh=0.8, Accuracy=97%, BaseAcc(Other)=97%, Sens=1.5%, Spec=99.9%, Sens^2+Spec^2=0.999"
## [1] "Thresh=0.9, Accuracy=97%, BaseAcc(Other)=97%, Sens=0.2%, Spec=99.9%, Sens^2+Spec^2=0.999"
## [1] "Thresh=1, Accuracy=97%, BaseAcc(Other)=97%, Sens=0%, Spec=100%, Sens^2+Spec^2=-2"
## [1] "Best Sensitivity_Specificity threshold= 0.1 inc= 0.1"
## [1] "========================================"
## [1] "start= 0 end= 0.2 inc= 0.01"
## [1] "Thresh=0, Accuracy=2.9%, BaseAcc(Other)=97%, Sens=100%, Spec=0%, Sens^2+Spec^2=-2"
## [1] "Thresh=0.01, Accuracy=75.1%, BaseAcc(Other)=97%, Sens=99.7%, Spec=74.3%, Sens^2+Spec^2=1.548"
## [1] "Thresh=0.02, Accuracy=82.8%, BaseAcc(Other)=97%, Sens=99.2%, Spec=82.3%, Sens^2+Spec^2=1.663"
## [1] "Thresh=0.03, Accuracy=86.5%, BaseAcc(Other)=97%, Sens=97.9%, Spec=86.2%, Sens^2+Spec^2=1.703"
## [1] "Thresh=0.04, Accuracy=88.7%, BaseAcc(Other)=97%, Sens=95.5%, Spec=88.5%, Sens^2+Spec^2=1.697"
## [1] "Thresh=0.05, Accuracy=90.1%, BaseAcc(Other)=97%, Sens=92.5%, Spec=90%, Sens^2+Spec^2=1.668"
## [1] "Thresh=0.06, Accuracy=91.2%, BaseAcc(Other)=97%, Sens=89.2%, Spec=91.2%, Sens^2+Spec^2=1.629"
## [1] "Thresh=0.07, Accuracy=92%, BaseAcc(Other)=97%, Sens=86.2%, Spec=92.1%, Sens^2+Spec^2=1.594"
## [1] "Thresh=0.08, Accuracy=92.6%, BaseAcc(Other)=97%, Sens=82.9%, Spec=92.9%, Sens^2+Spec^2=1.552"
## [1] "Thresh=0.09, Accuracy=93.2%, BaseAcc(Other)=97%, Sens=79.8%, Spec=93.6%, Sens^2+Spec^2=1.514"
## [1] "Thresh=0.1, Accuracy=93.6%, BaseAcc(Other)=97%, Sens=76.5%, Spec=94.1%, Sens^2+Spec^2=1.473"
## [1] "Thresh=0.11, Accuracy=94%, BaseAcc(Other)=97%, Sens=73.5%, Spec=94.6%, Sens^2+Spec^2=1.436"
```

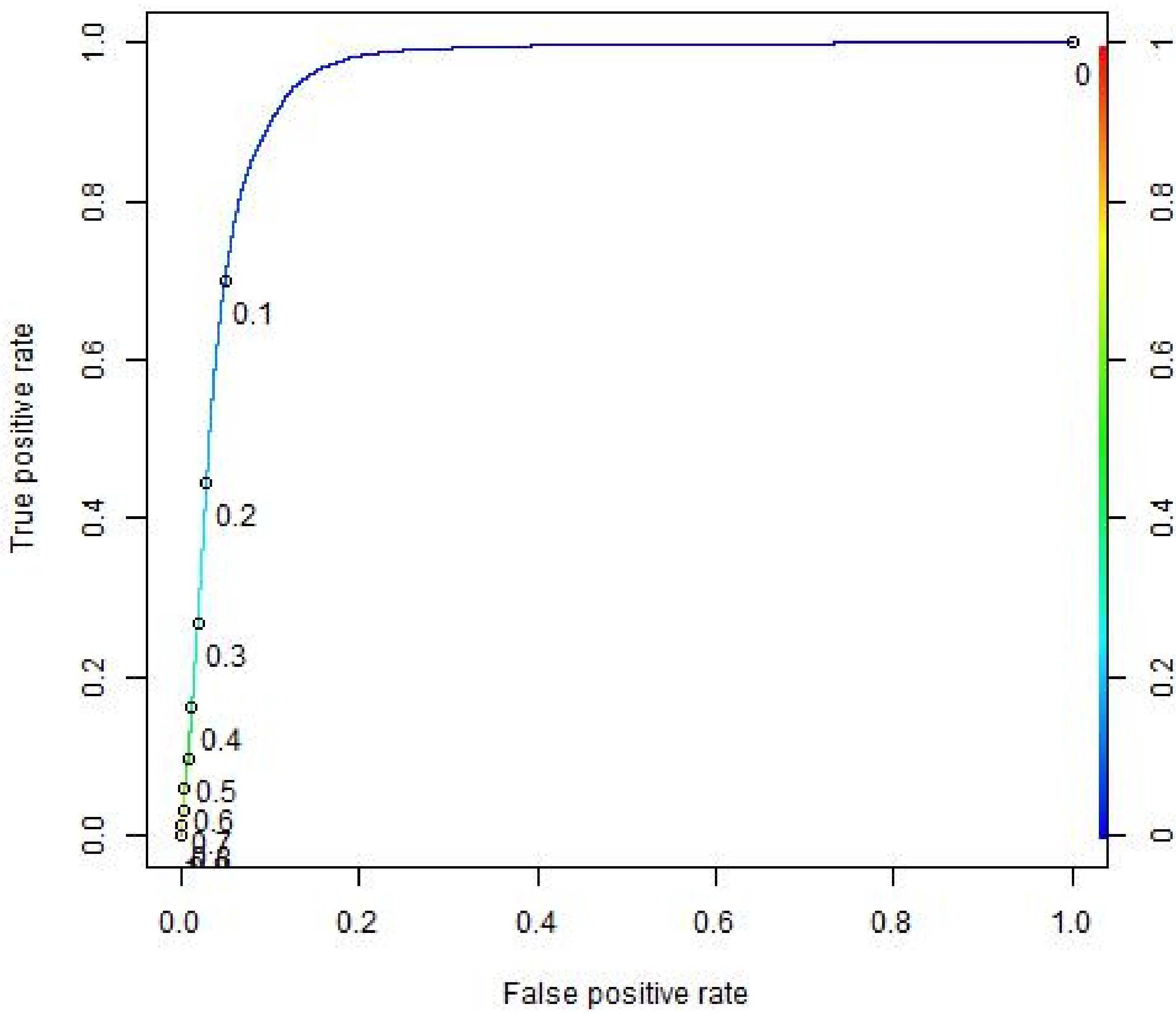

Figure 4: Douglas Fir ROC for Individuated Significant Data

```
## [1] "Thresh=0.12, Accuracy=94.3%, BaseAcc(Other)=97%, Sens=70.5%, Spec=95%, Sens^2+Spec^2=1.401"
## [1] "Thresh=0.13, Accuracy=94.5%, BaseAcc(Other)=97%, Sens=67.4%, Spec=95.3%, Sens^2+Spec^2=1.364"
## [1] "Thresh=0.14, Accuracy=94.7%, BaseAcc(Other)=97%, Sens=64.1%, Spec=95.7%, Sens^2+Spec^2=1.327"
## [1] "Thresh=0.15, Accuracy=94.9%, BaseAcc(Other)=97%, Sens=60.9%, Spec=95.9%, Sens^2+Spec^2=1.292"
## [1] "Thresh=0.16, Accuracy=95%, BaseAcc(Other)=97%, Sens=57.9%, Spec=96.2%, Sens^2+Spec^2=1.261"
## [1] "Thresh=0.17, Accuracy=95.2%, BaseAcc(Other)=97%, Sens=54.9%, Spec=96.4%, Sens^2+Spec^2=1.231"
## [1] "Thresh=0.18, Accuracy=95.3%, BaseAcc(Other)=97%, Sens=52.1%, Spec=96.6%, Sens^2+Spec^2=1.206"
## [1] "Thresh=0.19, Accuracy=95.4%, BaseAcc(Other)=97%, Sens=49.6%, Spec=96.8%, Sens^2+Spec^2=1.184"
## [1] "Best Sensitivity_Specificity threshold= 0.03 inc= 0.01"
## [1] "========================================"
## [1] "start= 0.02 end= 0.04 inc= 0.001"
## [1] "Thresh=0.02, Accuracy=82.8%, BaseAcc(Other)=97%, Sens=99.2%, Spec=82.3%, Sens^2+Spec^2=1.663"
## [1] "Thresh=0.021, Accuracy=83.3%, BaseAcc(Other)=97%, Sens=99.1%, Spec=82.8%, Sens^2+Spec^2=1.67"
## [1] "Thresh=0.022, Accuracy=83.7%, BaseAcc(Other)=97%, Sens=99%, Spec=83.3%, Sens^2+Spec^2=1.675"
## [1] "Thresh=0.023, Accuracy=84.2%, BaseAcc(Other)=97%, Sens=98.9%, Spec=83.7%, Sens^2+Spec^2=1.681"
## [1] "Thresh=0.024, Accuracy=84.6%, BaseAcc(Other)=97%, Sens=98.9%, Spec=84.1%, Sens^2+Spec^2=1.686"
## [1] "Thresh=0.025, Accuracy=85%, BaseAcc(Other)=97%, Sens=98.7%, Spec=84.5%, Sens^2+Spec^2=1.691"
## [1] "Thresh=0.026, Accuracy=85.3%, BaseAcc(Other)=97%, Sens=98.7%, Spec=84.9%, Sens^2+Spec^2=1.695"
## [1] "Thresh=0.027, Accuracy=85.6%, BaseAcc(Other)=97%, Sens=98.5%, Spec=85.2%, Sens^2+Spec^2=1.699"
## [1] "Thresh=0.028, Accuracy=86%, BaseAcc(Other)=97%, Sens=98.3%, Spec=85.6%, Sens^2+Spec^2=1.7"
## [1] "Thresh=0.029, Accuracy=86.3%, BaseAcc(Other)=97%, Sens=98.1%, Spec=85.9%, Sens^2+Spec^2=1.702"
## [1] "Thresh=0.03, Accuracy=86.5%, BaseAcc(Other)=97%, Sens=97.9%, Spec=86.2%, Sens^2+Spec^2=1.703"
## [1] "Thresh=0.031, Accuracy=86.8%, BaseAcc(Other)=97%, Sens=97.8%, Spec=86.5%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.032, Accuracy=87.1%, BaseAcc(Other)=97%, Sens=97.6%, Spec=86.8%, Sens^2+Spec^2=1.706"
## [1] "Thresh=0.033, Accuracy=87.3%, BaseAcc(Other)=97%, Sens=97.4%, Spec=87%, Sens^2+Spec^2=1.707"
## [1] "Thresh=0.034, Accuracy=87.5%, BaseAcc(Other)=97%, Sens=97.2%, Spec=87.3%, Sens^2+Spec^2=1.707"
## [1] "Thresh=0.035, Accuracy=87.8%, BaseAcc(Other)=97%, Sens=97%, Spec=87.5%, Sens^2+Spec^2=1.707"
## [1] "Thresh=0.036, Accuracy=88%, BaseAcc(Other)=97%, Sens=96.7%, Spec=87.7%, Sens^2+Spec^2=1.705"
## [1] "Thresh=0.037, Accuracy=88.2%, BaseAcc(Other)=97%, Sens=96.4%, Spec=87.9%, Sens^2+Spec^2=1.704"
## [1] "Thresh=0.038, Accuracy=88.4%, BaseAcc(Other)=97%, Sens=96.1%, Spec=88.1%, Sens^2+Spec^2=1.701"
## [1] "Thresh=0.039, Accuracy=88.5%, BaseAcc(Other)=97%, Sens=95.8%, Spec=88.3%, Sens^2+Spec^2=1.7"
## [1] "========================================"
## [1] "Best Threshold=0.033"
## [1] "Best Sensitivity_Specificity=1.70799981503938"
```

```
curThresh = as.numeric(result[bestThreshIndex])
DougFir_Agg_Sig_threshold = curThresh
```

The accuracy for the best threshold on the training set for Douglas Fir using significant aggregated data is shown below.

```
result = calcLogisticModelAccuracy (forestTrain$DouglasFir, DougFir_Agg_Train_predict,
                       curThresh, curThresh, 1, "DouglasFir", "Other", 3)
```

```
## [1] "Model Performance for threshold= 0.033"
## [1] "predicted performance="
##                        Predicted
## Actual                  FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other            343516 (TN)         51036 (FP)
##   1=Actual:DouglasFir       308 (FN)            11849 (TP)
## [1] "Sensitivity= 0.974664802171588 (True positive rate of DouglasFir = TP/(TP+FN) = 11849 /( 11849 +
## [1] "Specificity= 0.870648228877309 (True negative rate of Other = TN/(TN+FP) = 343516 /( 343516 + 5
## [1] "Sens^2+Spec^2=1.707"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.873757"
```

The accuracy for the best threshold on the testing set for Douglas Fir using significant aggregated data is shown below.

```
result = calcLogisticModelAccuracy (forestTest$DouglasFir, DougFir_Agg_Test_predict,
                       curThresh, curThresh, 1, "DouglasFir", "Other", 3,
                       saveFile=saveFileName, desc="Douglas Fir Sig Aggregate Vars",
                       AIC=DougFir_Agg_Sig_aic, AUC=DougFir_Agg_Sig_ROC_AUC)
```

```
## [1] "Model Performance for threshold= 0.033"
## [1] "predicted performance="
##                        Predicted
## Actual                  FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other          147081 (TN)         22012 (FP)
##   1=Actual:DouglasFir     123 (FN)            5087 (TP)
## [1] "Sensitivity= 0.976391554702495 (True positive rate of DouglasFir = TP/(TP+FN) = 5087 /( 5087 + 1
## [1] "Specificity= 0.869823115090512 (True negative rate of Other = TN/(TN+FP) = 147081 /( 147081 + 22
## [1] "Sens^2+Spec^2=1.709"
## [1] "Baseline (Other) Accuracy=0.970109"
## [1] "Logistic Accuracy=0.873008"
```

```
list[RC, DougFir_Agg_Sig_model_acc, DougFir_Agg_Sig_baseline_acc,
      TN, FN, FP, TP, DougFir_Agg_Sig_sens, DougFir_Agg_Sig_spec] <- result
  if (RC != "OK") {
    print(paste("Error - terminating:",RC))
    knitr:knit_exit()
  }
  DougFir_Agg_Sig_model_acc = as.integer(as.numeric(DougFir_Agg_Sig_model_acc)*1000)/10
  DougFir_Agg_Sig_baseline_acc = as.integer(as.numeric(DougFir_Agg_Sig_baseline_acc)*1000)/10
  DougFir_Agg_Sig_sens = as.integer(as.numeric(DougFir_Agg_Sig_sens)*1000)/10
  DougFir_Agg_Sig_spec = as.integer(as.numeric(DougFir_Agg_Sig_spec)*1000)/10
```

Calculate Douglas Fir Individuated Data Logisitic Model Accuracy - Significant Vars

Find best Douglas Fir threshold for Inividuated Data using significant variables.

```
result = calcLogisticModelAccuracy (forestTrain$DouglasFir, DougFir_Ind_Train_predict,
                       0.0, 1, 10, "DouglasFir", "Other", 1,1)
```

```
## [1] "Searching for threshold producing best Sensitivity_Specificity"
## [1] "start= 0 end= 1 inc= 0.1"
## [1] "Thresh=0, Accuracy=2.9%, BaseAcc(Other)=97%, Sens=100%, Spec=0%, Sens^2+Spec^2=-2"
## [1] "Thresh=0.1, Accuracy=94.3%, BaseAcc(Other)=97%, Sens=69.9%, Spec=95.1%, Sens^2+Spec^2=1.394"
## [1] "Thresh=0.2, Accuracy=95.6%, BaseAcc(Other)=97%, Sens=44.5%, Spec=97.1%, Sens^2+Spec^2=1.143"
## [1] "Thresh=0.3, Accuracy=96%, BaseAcc(Other)=97%, Sens=26.7%, Spec=98.1%, Sens^2+Spec^2=1.035"
## [1] "Thresh=0.4, Accuracy=96.3%, BaseAcc(Other)=97%, Sens=16.1%, Spec=98.8%, Sens^2+Spec^2=1.003"
## [1] "Thresh=0.5, Accuracy=96.6%, BaseAcc(Other)=97%, Sens=9.7%, Spec=99.2%, Sens^2+Spec^2=0.995"
## [1] "Thresh=0.6, Accuracy=96.8%, BaseAcc(Other)=97%, Sens=5.9%, Spec=99.6%, Sens^2+Spec^2=0.995"
## [1] "Thresh=0.7, Accuracy=96.9%, BaseAcc(Other)=97%, Sens=3.1%, Spec=99.8%, Sens^2+Spec^2=0.997"
## [1] "Thresh=0.8, Accuracy=97%, BaseAcc(Other)=97%, Sens=1.1%, Spec=99.9%, Sens^2+Spec^2=0.999"
## [1] "Thresh=0.9, Accuracy=97%, BaseAcc(Other)=97%, Sens=0%, Spec=99.9%, Sens^2+Spec^2=0.999"
## [1] "Thresh=1, Accuracy=97%, BaseAcc(Other)=97%, Sens=0%, Spec=100%, Sens^2+Spec^2=-2"
## [1] "Best Sensitivity_Specificity threshold= 0.1 inc= 0.1"
## [1] "========================================"
## [1] "start= 0 end= 0.2 inc= 0.01"
## [1] "Thresh=0, Accuracy=2.9%, BaseAcc(Other)=97%, Sens=100%, Spec=0%, Sens^2+Spec^2=-2"
```

```
## [1] "Thresh=0.01, Accuracy=70.3%, BaseAcc(Other)=97%, Sens=99.3%, Spec=69.4%, Sens^2+Spec^2=1.469"
## [1] "Thresh=0.02, Accuracy=82%, BaseAcc(Other)=97%, Sens=97.9%, Spec=81.5%, Sens^2+Spec^2=1.623"
## [1] "Thresh=0.03, Accuracy=86.8%, BaseAcc(Other)=97%, Sens=95.1%, Spec=86.5%, Sens^2+Spec^2=1.655"
## [1] "Thresh=0.04, Accuracy=89.3%, BaseAcc(Other)=97%, Sens=91.2%, Spec=89.3%, Sens^2+Spec^2=1.63"
## [1] "Thresh=0.05, Accuracy=91%, BaseAcc(Other)=97%, Sens=87.4%, Spec=91.1%, Sens^2+Spec^2=1.594"
## [1] "Thresh=0.06, Accuracy=92.2%, BaseAcc(Other)=97%, Sens=84%, Spec=92.4%, Sens^2+Spec^2=1.562"
## [1] "Thresh=0.07, Accuracy=93%, BaseAcc(Other)=97%, Sens=80.5%, Spec=93.4%, Sens^2+Spec^2=1.521"
## [1] "Thresh=0.08, Accuracy=93.6%, BaseAcc(Other)=97%, Sens=76.8%, Spec=94.1%, Sens^2+Spec^2=1.476"
## [1] "Thresh=0.09, Accuracy=94%, BaseAcc(Other)=97%, Sens=73.1%, Spec=94.6%, Sens^2+Spec^2=1.431"
## [1] "Thresh=0.1, Accuracy=94.3%, BaseAcc(Other)=97%, Sens=69.9%, Spec=95.1%, Sens^2+Spec^2=1.394"
## [1] "Thresh=0.11, Accuracy=94.6%, BaseAcc(Other)=97%, Sens=67.1%, Spec=95.4%, Sens^2+Spec^2=1.362"
## [1] "Thresh=0.12, Accuracy=94.8%, BaseAcc(Other)=97%, Sens=64.2%, Spec=95.7%, Sens^2+Spec^2=1.33"
## [1] "Thresh=0.13, Accuracy=95%, BaseAcc(Other)=97%, Sens=61.4%, Spec=96%, Sens^2+Spec^2=1.299"
## [1] "Thresh=0.14, Accuracy=95.1%, BaseAcc(Other)=97%, Sens=58.7%, Spec=96.2%, Sens^2+Spec^2=1.272"
## [1] "Thresh=0.15, Accuracy=95.2%, BaseAcc(Other)=97%, Sens=56.2%, Spec=96.4%, Sens^2+Spec^2=1.246"
## [1] "Thresh=0.16, Accuracy=95.3%, BaseAcc(Other)=97%, Sens=53.9%, Spec=96.6%, Sens^2+Spec^2=1.224"
## [1] "Thresh=0.17, Accuracy=95.4%, BaseAcc(Other)=97%, Sens=51.4%, Spec=96.7%, Sens^2+Spec^2=1.201"
## [1] "Thresh=0.18, Accuracy=95.5%, BaseAcc(Other)=97%, Sens=49.1%, Spec=96.9%, Sens^2+Spec^2=1.181"
## [1] "Thresh=0.19, Accuracy=95.5%, BaseAcc(Other)=97%, Sens=46.8%, Spec=97%, Sens^2+Spec^2=1.161"
## [1] "Best Sensitivity_Specificity threshold= 0.03 inc= 0.01"
## [1] "========================================"
## [1] "start= 0.02 end= 0.04 inc= 0.001"
## [1] "Thresh=0.02, Accuracy=82%, BaseAcc(Other)=97%, Sens=97.9%, Spec=81.5%, Sens^2+Spec^2=1.623"
## [1] "Thresh=0.021, Accuracy=82.6%, BaseAcc(Other)=97%, Sens=97.6%, Spec=82.2%, Sens^2+Spec^2=1.629"
## [1] "Thresh=0.022, Accuracy=83.2%, BaseAcc(Other)=97%, Sens=97.4%, Spec=82.8%, Sens^2+Spec^2=1.635"
## [1] "Thresh=0.023, Accuracy=83.8%, BaseAcc(Other)=97%, Sens=97.1%, Spec=83.4%, Sens^2+Spec^2=1.64"
## [1] "Thresh=0.024, Accuracy=84.3%, BaseAcc(Other)=97%, Sens=96.9%, Spec=83.9%, Sens^2+Spec^2=1.645"
## [1] "Thresh=0.025, Accuracy=84.8%, BaseAcc(Other)=97%, Sens=96.7%, Spec=84.4%, Sens^2+Spec^2=1.649"
## [1] "Thresh=0.026, Accuracy=85.3%, BaseAcc(Other)=97%, Sens=96.3%, Spec=84.9%, Sens^2+Spec^2=1.651"
## [1] "Thresh=0.027, Accuracy=85.7%, BaseAcc(Other)=97%, Sens=96.1%, Spec=85.3%, Sens^2+Spec^2=1.652"
## [1] "Thresh=0.028, Accuracy=86.1%, BaseAcc(Other)=97%, Sens=95.7%, Spec=85.8%, Sens^2+Spec^2=1.653"
## [1] "Thresh=0.029, Accuracy=86.4%, BaseAcc(Other)=97%, Sens=95.4%, Spec=86.1%, Sens^2+Spec^2=1.653"
## [1] "Thresh=0.03, Accuracy=86.8%, BaseAcc(Other)=97%, Sens=95.1%, Spec=86.5%, Sens^2+Spec^2=1.655"
## [1] "Thresh=0.031, Accuracy=87.1%, BaseAcc(Other)=97%, Sens=94.9%, Spec=86.9%, Sens^2+Spec^2=1.655"
## [1] "Thresh=0.032, Accuracy=87.4%, BaseAcc(Other)=97%, Sens=94.6%, Spec=87.2%, Sens^2+Spec^2=1.656"
## [1] "Thresh=0.033, Accuracy=87.7%, BaseAcc(Other)=97%, Sens=94.2%, Spec=87.5%, Sens^2+Spec^2=1.654"
## [1] "Thresh=0.034, Accuracy=88%, BaseAcc(Other)=97%, Sens=93.7%, Spec=87.8%, Sens^2+Spec^2=1.651"
## [1] "Thresh=0.035, Accuracy=88.2%, BaseAcc(Other)=97%, Sens=93.3%, Spec=88.1%, Sens^2+Spec^2=1.648"
## [1] "Thresh=0.036, Accuracy=88.5%, BaseAcc(Other)=97%, Sens=92.9%, Spec=88.4%, Sens^2+Spec^2=1.644"
## [1] "Thresh=0.037, Accuracy=88.7%, BaseAcc(Other)=97%, Sens=92.4%, Spec=88.6%, Sens^2+Spec^2=1.641"
## [1] "Thresh=0.038, Accuracy=88.9%, BaseAcc(Other)=97%, Sens=92%, Spec=88.8%, Sens^2+Spec^2=1.637"
## [1] "Thresh=0.039, Accuracy=89.1%, BaseAcc(Other)=97%, Sens=91.6%, Spec=89.1%, Sens^2+Spec^2=1.634"
## [1] "========================================"
## [1] "Best Threshold=0.032"
## [1] "Best Sensitivity_Specificity=1.6565093046246"
```

```r
curThresh = as.numeric(result[bestThreshIndex])
DougFir_Ind_Sig_threshold = curThresh
```

The accuracy for the best threshold on the training set for Douglas Fir using significant individuated data is shown below.

```r
result = calcLogisticModelAccuracy (forestTrain$DouglasFir, DougFir_Ind_Train_predict,
                       curThresh, curThresh, 1, "DouglasFir", "Other", 3)
```

```
## [1] "Model Performance for threshold= 0.032"
## [1] "predicted performance="
##                       Predicted
## Actual                 FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other           344130 (TN)         50422 (FP)
##   1=Actual:DouglasFir      651 (FN)            11506 (TP)
## [1] "Sensitivity= 0.946450604589948 (True positive rate of DouglasFir = TP/(TP+FN) = 11506 /( 11506 +
## [1] "Specificity= 0.872204424258399 (True negative rate of Other = TN/(TN+FP) = 344130 /( 344130 + 50
## [1] "Sens^2+Spec^2=1.656"
## [1] "Baseline (Other) Accuracy=0.970108"
## [1] "Logistic Accuracy=0.874423"
```

The accuracy for the best threshold on the testing set for Douglas Fir using significant individuated data is shown below.

```r
result = calcLogisticModelAccuracy (forestTest$DouglasFir, DougFir_Ind_Test_predict,
                       curThresh, curThresh, 1, "DouglasFir", "Other", 3,
                       saveFile=saveFileName, desc="Douglas Fir Sig Individualized Vars",
                       AIC=DougFir_Ind_Sig_aic, AUC=DougFir_Ind_Sig_ROC_AUC)
```

```
## [1] "Model Performance for threshold= 0.032"
## [1] "predicted performance="
##                       Predicted
## Actual                 FALSE=Predict:Other TRUE=Predict:DouglasFir
##   0=Actual:Other           147416 (TN)         21677 (FP)
##   1=Actual:DouglasFir      280 (FN)            4930 (TP)
## [1] "Sensitivity= 0.946257197696737 (True positive rate of DouglasFir = TP/(TP+FN) = 4930 /( 4930 + 2
## [1] "Specificity= 0.871804273388017 (True negative rate of Other = TN/(TN+FP) = 147416 /( 147416 + 2
## [1] "Sens^2+Spec^2=1.655"
## [1] "Baseline (Other) Accuracy=0.970109"
## [1] "Logistic Accuracy=0.874029"
```

```r
list[RC, DougFir_Ind_Sig_model_acc, DougFir_Ind_Sig_baseline_acc,
      TN, FN, FP, TP, DougFir_Ind_Sig_sens, DougFir_Ind_Sig_spec] <- result
  if (RC != "OK") {
    print(paste("Error - terminating:",RC))
    knitr:knit_exit()
  }
  DougFir_Ind_Sig_model_acc = as.integer(as.numeric(DougFir_Ind_Sig_model_acc)*1000)/10
  DougFir_Ind_Sig_baseline_acc = as.integer(as.numeric(DougFir_Ind_Sig_baseline_acc)*1000)/10
  DougFir_Ind_Sig_sens = as.integer(as.numeric(DougFir_Ind_Sig_sens)*1000)/10
  DougFir_Ind_Sig_spec = as.integer(as.numeric(DougFir_Ind_Sig_spec)*1000)/10

############# End End End End End End End End End End End End #################
```

The accuracy of the models is shown below:

Logistic Model	Accuracy	Sens	Spec	AIC	AUC	Threshold
Douglas Fir Aggregate All Vars	89.1%	95.6%	88.9%	57784	96.4%	0.035
Douglas Fir Individual All Vars	89.1%	95.6%	88.9%	57790	96.4% 0.035	
Douglas Fir Aggregate Sig Vars	87.3%	97.6%	86.9%	64213	95.8%	0.033
Douglas Fir Individual Sig Vars	87.4%	94.6%	87.1%	67703	95.4%	0.032

There is a slight degradation in the accuracy with insignificant variables eliminated, but not by much.

Conclusion

It is beginning to look like there is no advantage to dis-aggregating the Soil Type variables into their component parts. I was hoping there would be some improvement by allowing the individual variables to be "more finely" tuned. There is probably a mathematical explanation that proves there is no advantage of breaking out aggregated variables. I have to think about that more.

The logistic regression results for Douglas Fir are 7% better than the original paper this project was modeled after. These tests need to be done for the remaining 6 forest cover types to see how regression does overall.

```
curTime=Sys.time()
print(paste("Forest Cover Logistic script ended at",curTime))
```

```
## [1] "Forest Cover Logistic script ended at 2018-08-12 18:33:29"
```