Revised RMA/crop loss Random Forest Models

Rachel Schattman

March 19, 2019

# load libraries

##   
## 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

## -------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)

## -------------------------------------------------------------------------

##   
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

## Warning: package 'randomForest' was built under R version 3.5.3

## 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

## Warning: package 'Metrics' was built under R version 3.5.3

# Helpful resources:

## <https://www.r-bloggers.com/how-to-implement-random-forests-in-r/>

## <https://cran.r-project.org/web/packages/randomForest/randomForest.pdf>

## <https://rpubs.com/mbaumer/randomForest>

script\_path <- "C:/Users/rschattman/Documents/Research/RandomForestRMA/data"  
in\_dir <- "C:/Users/rschattman/Documents/Research/RandomForestRMA/data"  
out\_dir <- "C:/Users/rschattman/Documents/Research/RandomForestRMA/output/data"

# Read in data and combine into single dataframe

# Create new data frames with one dependent variable

#head(PAbeta\_wide)  
WetAcres <- PAbeta\_wide[,c(1,4,9:56)] #subset year, dependent variable, and all precip columes  
WetDollars <- PAbeta\_wide[,c(1,3,9:56)]  
DryAcres <- PAbeta\_wide[,c(1,7,9:56)]  
DryDollars <- PAbeta\_wide[,c(1,6,9:56)]

# Review data

# Split into trainning, validation, and test sets

# Create Random Forest Model and test performance metrics

## Wet Acres

Mod1 <- randomForest(WetAcres ~ .,   
 data = Wettrain,   
 ntree = 500,   
 #method = "anova",   
 importance = TRUE)  
  
#print(Mod1) # % of variance expalined is low. Tuning needed  
summary(Mod1)

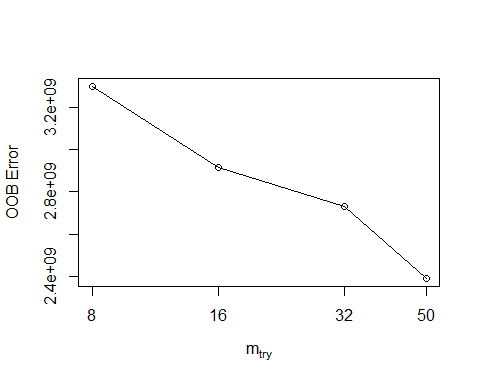
## Length Class Mode   
## call 5 -none- call   
## type 1 -none- character  
## predicted 14 -none- numeric   
## mse 500 -none- numeric   
## rsq 500 -none- numeric   
## oob.times 14 -none- numeric   
## importance 98 -none- numeric   
## importanceSD 49 -none- numeric   
## localImportance 0 -none- NULL   
## proximity 0 -none- NULL   
## ntree 1 -none- numeric   
## mtry 1 -none- numeric   
## forest 11 -none- list   
## coefs 0 -none- NULL   
## y 14 -none- numeric   
## test 0 -none- NULL   
## inbag 0 -none- NULL   
## terms 3 terms call

#plot(Mod1)  
  
pred <- predict(object = Mod1, newdata = Wettest)  
RMSE\_Mod1 <- rmse(actual = Wettest$WetAcres, #actual values  
 predicted = pred) #predicted values  
print(RMSE\_Mod1/mean(Wettest$WetAcres)) #tells us the %of the mean represented by RMSE. AKA "coefficient of variation"

## [1] 0.30291

# Tune mtry using OOB error  
set.seed(25)  
#train\_pred <- predict(object = Mod1, newdata = PAtrain)  
res <- tuneRF(x = Wettrain,  
 y = Wettrain$WetAcre,  
 ntree = 500,  
 stepfactor = 0.5,  
 doBest=TRUE, # Returns a random forest model with optimal mtry value  
 importance = TRUE)

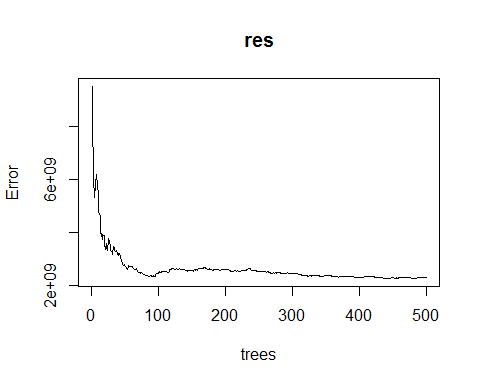
## mtry = 16 OOB error = 2914205385   
## Searching left ...  
## mtry = 8 OOB error = 3298852744   
## -0.1319905 0.05   
## Searching right ...  
## mtry = 32 OOB error = 2729903281   
## 0.06324266 0.05   
## mtry = 50 OOB error = 2389996818   
## 0.1245123 0.05



#localImp = TRUE)  
print(res)

##   
## Call:  
## randomForest(x = x, y = y, mtry = res[which.min(res[, 2]), 1], importance = TRUE, stepfactor = 0.5)   
## Type of random forest: regression  
## Number of trees: 500  
## No. of variables tried at each split: 50  
##   
## Mean of squared residuals: 2327616278  
## % Var explained: 38.23

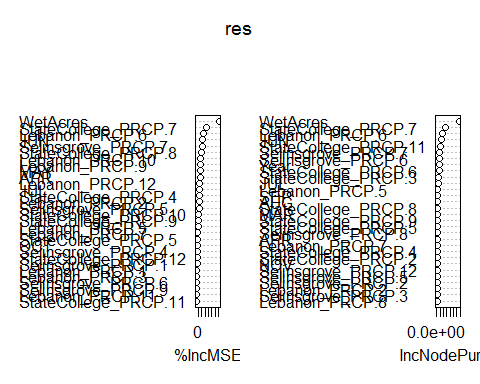
plot(res)



res$importance

## %IncMSE IncNodePurity  
## Year -109431051.6 1440070720  
## WetAcres 1638125031.0 13770856736  
## StateCollege\_PRCP.1 -4733884.7 80833423  
## Lebanon\_PRCP.1 -6500132.3 92026562  
## Selinsgrove\_PRCP.1 -10225659.2 132811591  
## StateCollege\_PRCP.2 -13008812.1 373751147  
## Lebanon\_PRCP.2 8380990.3 273896885  
## Selinsgrove\_PRCP.2 -17239217.3 311433786  
## StateCollege\_PRCP.3 -53086413.1 900979540  
## Lebanon\_PRCP.3 -2572467.7 19679452  
## Selinsgrove\_PRCP.3 -30124976.5 252789475  
## StateCollege\_PRCP.4 11369068.2 381437988  
## Lebanon\_PRCP.4 -7567639.5 133675909  
## Selinsgrove\_PRCP.4 -10107776.9 163044369  
## StateCollege\_PRCP.5 -4144266.8 477492561  
## Lebanon\_PRCP.5 -7398350.9 725088401  
## Selinsgrove\_PRCP.5 2323924.5 313882708  
## StateCollege\_PRCP.6 -70362820.0 1353576849  
## Lebanon\_PRCP.6 94539650.8 3273483135  
## Selinsgrove\_PRCP.6 -28547977.8 1465084235  
## StateCollege\_PRCP.7 358106318.2 5281572778  
## Lebanon\_PRCP.7 -8669100.9 406893033  
## Selinsgrove\_PRCP.7 76996208.7 1535402490  
## StateCollege\_PRCP.8 8973248.4 522455560  
## Lebanon\_PRCP.8 -11875237.3 198071456  
## Selinsgrove\_PRCP.8 -14988925.8 461748404  
## StateCollege\_PRCP.9 121913.8 512537159  
## Lebanon\_PRCP.9 12042594.2 99610538  
## Selinsgrove\_PRCP.9 -3578446.3 69994962  
## StateCollege\_PRCP.10 2724908.9 138663847  
## Lebanon\_PRCP.10 2377790.0 1531761  
## Selinsgrove\_PRCP.10 -12996609.5 78202811  
## StateCollege\_PRCP.11 -14767658.3 2512260417  
## Lebanon\_PRCP.11 -1536273.2 16417429  
## Selinsgrove\_PRCP.11 -15736934.4 99961596  
## StateCollege\_PRCP.12 -4916204.2 120912482  
## Lebanon\_PRCP.12 1188068.7 58754394  
## Selinsgrove\_PRCP.12 -20191049.3 337078531  
## JAN -1447456.8 46944372  
## FEB -1669443.7 11968207  
## MAR -28538647.5 520555140  
## APR 14668599.7 435175167  
## MAY 3862634.9 88421615  
## JUN 113912561.5 2886872611  
## JUL 18424743.4 864655186  
## AUG -22778188.8 578963238  
## SEP -13608918.3 599111685  
## OCT -1268644.7 126387871  
## NOV -15770189.4 356886370  
## DEC -17004711.0 68772498

varImpPlot(res)



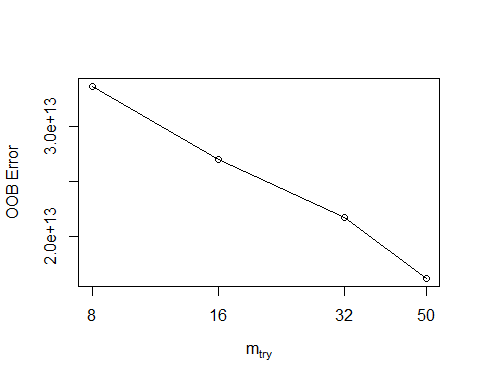
## Wet Dollars

# Split into trainning, validation, and test sets  
set.seed(25)  
assignment <- sample(1:3, size = nrow(WetDollars), prob = c(0.7, 0.15, 0.15), replace = TRUE)  
  
Wettrain2 <- WetDollars[assignment == 1,]  
Wetvalid2 <- WetDollars[assignment == 2,]  
Wettest2 <- WetDollars[assignment == 3,]  
  
#summary(Wettrain2)  
#summary(Wetvalid2)  
#summary(Wettest2)  
  
  
Mod2 <- randomForest(WetDollars ~ .,   
 data = Wettrain2,   
 ntree = 500,   
 #method = "anova",   
 importance = TRUE)  
  
#print(Mod2) # % of variance expalined is low. Tuning needed  
#summary(Mod2)  
#plot(Mod2)  
  
pred2 <- predict(object = Mod2, newdata = Wettest2)  
RMSE\_Mod2 <- rmse(actual = Wettest2$WetDollars, #actual values  
 predicted = pred2) #predicted values  
print(RMSE\_Mod2/mean(Wettest2$WetDollars)) #tells us the %of the mean represented by RMSE. AKA "coefficient of variation"

## [1] 0.4575993

# Tune mtry using OOB error  
set.seed(25)  
#train\_pred <- predict(object = Mod1, newdata = PAtrain)  
res2 <- tuneRF(x = Wettrain2,  
 y = Wettrain2$WetDollars,  
 ntree = 500,  
 stepfactor = 0.5,  
 doBest=TRUE, # Returns a random forest model with optimal mtry value  
 importance = TRUE)

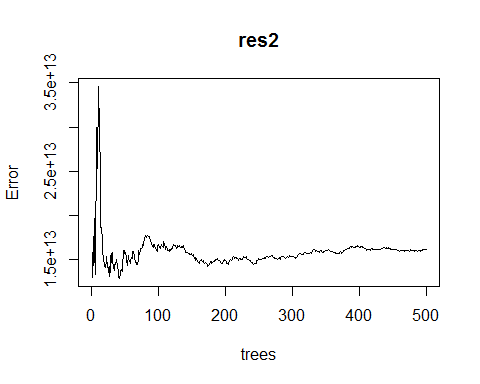
## mtry = 16 OOB error = 2.695644e+13   
## Searching left ...  
## mtry = 8 OOB error = 3.362091e+13   
## -0.2472309 0.05   
## Searching right ...  
## mtry = 32 OOB error = 2.168012e+13   
## 0.1957352 0.05   
## mtry = 50 OOB error = 1.616003e+13   
## 0.254615 0.05



#localImp = TRUE)  
print(res2)

##   
## Call:  
## randomForest(x = x, y = y, mtry = res[which.min(res[, 2]), 1], importance = TRUE, stepfactor = 0.5)   
## Type of random forest: regression  
## Number of trees: 500  
## No. of variables tried at each split: 50  
##   
## Mean of squared residuals: 1.614258e+13  
## % Var explained: 67.73

plot(res2)



res2$importance

## %IncMSE IncNodePurity  
## Year 1.492216e+11 2.076613e+12  
## WetDollars 3.507522e+13 3.209201e+14  
## StateCollege\_PRCP.1 4.199247e+09 4.140208e+12  
## Lebanon\_PRCP.1 -1.238743e+11 5.027268e+12  
## Selinsgrove\_PRCP.1 1.373866e+11 9.356924e+12  
## StateCollege\_PRCP.2 -2.000074e+11 2.061557e+12  
## Lebanon\_PRCP.2 1.217833e+10 1.724166e+12  
## Selinsgrove\_PRCP.2 3.063961e+09 1.893287e+12  
## StateCollege\_PRCP.3 -2.325823e+10 4.617288e+12  
## Lebanon\_PRCP.3 1.890071e+11 8.751792e+12  
## Selinsgrove\_PRCP.3 4.033225e+10 5.486445e+11  
## StateCollege\_PRCP.4 7.260511e+11 9.140074e+12  
## Lebanon\_PRCP.4 -1.994035e+10 2.069271e+11  
## Selinsgrove\_PRCP.4 -5.267015e+10 1.040687e+12  
## StateCollege\_PRCP.5 -3.695164e+10 3.493911e+11  
## Lebanon\_PRCP.5 2.618359e+11 9.143805e+12  
## Selinsgrove\_PRCP.5 6.655302e+10 8.356351e+11  
## StateCollege\_PRCP.6 6.112282e+10 4.884671e+12  
## Lebanon\_PRCP.6 5.679026e+12 9.446013e+13  
## Selinsgrove\_PRCP.6 -1.522097e+11 3.392124e+12  
## StateCollege\_PRCP.7 4.675452e+11 1.717404e+13  
## Lebanon\_PRCP.7 -8.727803e+10 7.772359e+11  
## Selinsgrove\_PRCP.7 2.500518e+11 6.938950e+12  
## StateCollege\_PRCP.8 3.532244e+11 2.435488e+13  
## Lebanon\_PRCP.8 -9.636112e+09 8.567342e+11  
## Selinsgrove\_PRCP.8 4.334026e+11 2.763132e+13  
## StateCollege\_PRCP.9 -4.612282e+09 2.265460e+12  
## Lebanon\_PRCP.9 2.302684e+10 5.692063e+11  
## Selinsgrove\_PRCP.9 -2.917793e+10 1.201969e+12  
## StateCollege\_PRCP.10 -5.092807e+10 1.369840e+12  
## Lebanon\_PRCP.10 8.829517e+10 7.156616e+11  
## Selinsgrove\_PRCP.10 -1.183691e+09 6.219557e+11  
## StateCollege\_PRCP.11 -1.380161e+10 4.590524e+11  
## Lebanon\_PRCP.11 1.642646e+10 8.108650e+11  
## Selinsgrove\_PRCP.11 1.171689e+11 8.091539e+11  
## StateCollege\_PRCP.12 -2.868009e+10 9.017920e+11  
## Lebanon\_PRCP.12 -1.441217e+11 3.877619e+12  
## Selinsgrove\_PRCP.12 5.340822e+10 1.527548e+12  
## JAN 3.273513e+11 9.919190e+12  
## FEB -6.555884e+10 1.021864e+12  
## MAR -1.915666e+10 3.375041e+12  
## APR 5.343148e+09 1.111013e+12  
## MAY -9.151809e+10 1.721220e+12  
## JUN 1.066423e+12 1.446601e+13  
## JUL 5.870861e+10 1.079916e+12  
## AUG 3.158142e+11 1.091745e+13  
## SEP 1.356905e+10 4.704531e+12  
## OCT -8.681158e+10 1.657728e+11  
## NOV 2.608009e+10 5.791294e+11  
## DEC -3.060431e+11 3.397829e+12

varImpPlot(res2)

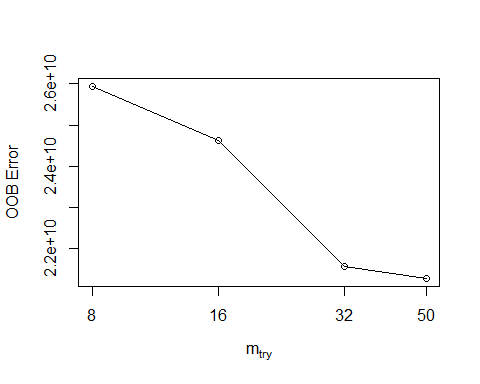
 ## Dry Acres

# Split into trainning, validation, and test sets  
set.seed(25)  
assignment <- sample(1:3, size = nrow(DryAcres), prob = c(0.7, 0.15, 0.15), replace = TRUE)  
  
Drytrain <- DryAcres[assignment == 1,]  
Dryvalid <- DryAcres[assignment == 2,]  
Drytest <- DryAcres[assignment == 3,]  
  
#summary(Drytrain)  
#summary(Dryvalid)  
#summary(Drytest)  
  
  
Mod3 <- randomForest(DryAcres ~ .,   
 data = Drytrain,   
 ntree = 500,   
 #method = "anova",   
 importance = TRUE)  
  
#print(Mod3) # % of variance expalined is low. Tuning needed  
#summary(Mod3)  
#plot(Mod3)  
  
pred3 <- predict(object = Mod3, newdata = Drytest)  
RMSE\_Mod3 <- rmse(actual = Drytest$DryAcres, #actual values  
 predicted = pred3) #predicted values  
print(RMSE\_Mod3/mean(Drytest$DryAcres)) #tells us the %of the mean represented by RMSE. AKA "coefficient of variation"

## [1] 20.68058

# Tune mtry using OOB error  
set.seed(25)  
#train\_pred <- predict(object = Mod1, newdata = PAtrain)  
res3 <- tuneRF(x = Drytrain,  
 y = Drytrain$DryAcres,  
 ntree = 500,  
 stepfactor = 0.5,  
 doBest=TRUE, # Returns a random forest model with optimal mtry value  
 importance = TRUE)

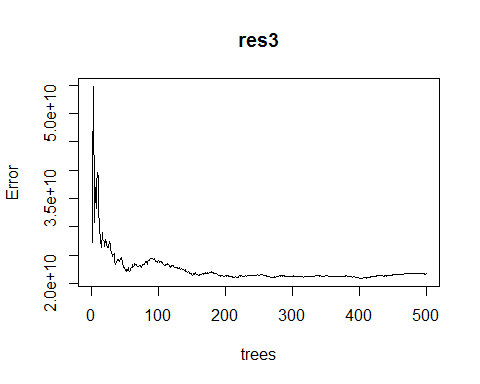
## mtry = 16 OOB error = 24613961386   
## Searching left ...  
## mtry = 8 OOB error = 25940516129   
## -0.0538944 0.05   
## Searching right ...  
## mtry = 32 OOB error = 21563445393   
## 0.1239344 0.05   
## mtry = 50 OOB error = 21272628416   
## 0.01348657 0.05



#localImp = TRUE)  
print(res3)

##   
## Call:  
## randomForest(x = x, y = y, mtry = res[which.min(res[, 2]), 1], importance = TRUE, stepfactor = 0.5)   
## Type of random forest: regression  
## Number of trees: 500  
## No. of variables tried at each split: 50  
##   
## Mean of squared residuals: 21696289130  
## % Var explained: 6.98

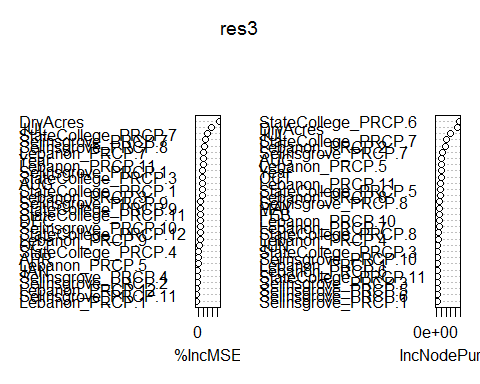
plot(res3) # looks pretty choppy?



res3$importance

## %IncMSE IncNodePurity  
## Year 299238276.9 9880369500  
## DryAcres 4104772189.6 44117123855  
## StateCollege\_PRCP.1 71856507.3 592910666  
## Lebanon\_PRCP.1 -21800063.5 1600822648  
## Selinsgrove\_PRCP.1 98974949.8 1086452003  
## StateCollege\_PRCP.2 -128082437.6 253368876  
## Lebanon\_PRCP.2 -440859831.0 18452680167  
## Selinsgrove\_PRCP.2 -6497537.7 271377052  
## StateCollege\_PRCP.3 202315860.2 1624088770  
## Lebanon\_PRCP.3 21719484.5 111281273  
## Selinsgrove\_PRCP.3 -48128522.4 1279555100  
## StateCollege\_PRCP.4 7945248.6 494528398  
## Lebanon\_PRCP.4 -127054649.9 2814985447  
## Selinsgrove\_PRCP.4 -6958775.7 371717040  
## StateCollege\_PRCP.5 -327392153.0 8966001261  
## Lebanon\_PRCP.5 -9903829.6 10776890980  
## Selinsgrove\_PRCP.5 -80341478.1 1211426768  
## StateCollege\_PRCP.6 -774362040.4 49345604973  
## Lebanon\_PRCP.6 -100052752.3 1540557359  
## Selinsgrove\_PRCP.6 -111293004.3 1103764361  
## StateCollege\_PRCP.7 592190141.2 21507044402  
## Lebanon\_PRCP.7 222175785.7 5252792462  
## Selinsgrove\_PRCP.7 425040046.9 12423563379  
## StateCollege\_PRCP.8 -169190848.8 3214770372  
## Lebanon\_PRCP.8 -443038439.4 7338356752  
## Selinsgrove\_PRCP.8 379706405.1 6416211412  
## StateCollege\_PRCP.9 31439644.4 211478120  
## Lebanon\_PRCP.9 11076648.0 344364466  
## Selinsgrove\_PRCP.9 9187413.5 62914288  
## StateCollege\_PRCP.10 -100823976.1 933374138  
## Lebanon\_PRCP.10 -261358260.5 5319217809  
## Selinsgrove\_PRCP.10 47984020.8 1622712408  
## StateCollege\_PRCP.11 49627185.4 1451193436  
## Lebanon\_PRCP.11 320557170.5 9102025322  
## Selinsgrove\_PRCP.11 -16767232.8 680993153  
## StateCollege\_PRCP.12 1196260.7 222178756  
## Lebanon\_PRCP.12 -16263984.7 598863613  
## Selinsgrove\_PRCP.12 -30942910.3 493801714  
## JAN -9033304.7 1056996311  
## FEB -333395969.6 5634712163  
## MAR -42287916.5 533945523  
## APR 297977.3 198597949  
## MAY -141239511.7 5929295564  
## JUN -217365968.2 1745961251  
## JUL 1737429736.1 28522082353  
## AUG 105068835.3 10841667440  
## SEP -8165055.0 43297984  
## OCT 87107670.4 9532435449  
## NOV -32509260.9 286119431  
## DEC 5542353.2 159565596

varImpPlot(res3)



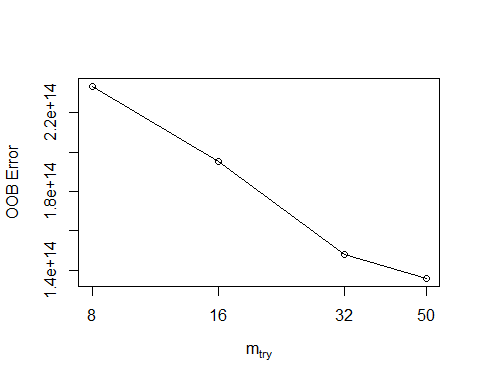
## Dry Dollars

# Split into trainning, validation, and test sets  
set.seed(25)  
assignment <- sample(1:3, size = nrow(DryDollars), prob = c(0.7, 0.15, 0.15), replace = TRUE)  
  
Drytrain2 <- DryDollars[assignment == 1,]  
Dryvalid2 <- DryDollars[assignment == 2,]  
Drytest2 <- DryDollars[assignment == 3,]  
  
#summary(Drytrain2)  
#summary(Dryvalid2)  
#summary(Drytest2)  
  
  
Mod4 <- randomForest(DryDollars ~ .,   
 data = Drytrain2,   
 ntree = 500,   
 #method = "anova",   
 importance = TRUE)  
  
#print(Mod4) # % of variance expalined is low. Tuning needed  
#summary(Mod4)  
#plot(Mod4)  
  
pred4 <- predict(object = Mod4, newdata = Drytest2)  
RMSE\_Mod4 <- rmse(actual = Drytest2$DryDollars, #actual values  
 predicted = pred4) #predicted values  
print(RMSE\_Mod4/mean(Drytest2$DryDollars)) #tells us the %of the mean represented by RMSE. AKA "coefficient of variation"

## [1] 32.21106

# Tune mtry using OOB error  
set.seed(25)  
#train\_pred <- predict(object = Mod1, newdata = PAtrain)  
res4 <- tuneRF(x = Drytrain2,  
 y = Drytrain2$DryDollars,  
 ntree = 500,  
 stepfactor = 0.5,  
 doBest=TRUE, # Returns a random forest model with optimal mtry value  
 importance = TRUE)

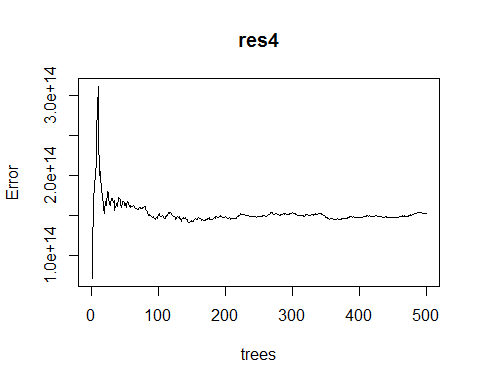
## mtry = 16 OOB error = 1.953765e+14   
## Searching left ...  
## mtry = 8 OOB error = 2.335376e+14   
## -0.1953211 0.05   
## Searching right ...  
## mtry = 32 OOB error = 1.482026e+14   
## 0.2414511 0.05   
## mtry = 50 OOB error = 1.357993e+14   
## 0.08369144 0.05



#localImp = TRUE)  
print(res4)

##   
## Call:  
## randomForest(x = x, y = y, mtry = res[which.min(res[, 2]), 1], importance = TRUE, stepfactor = 0.5)   
## Type of random forest: regression  
## Number of trees: 500  
## No. of variables tried at each split: 50  
##   
## Mean of squared residuals: 1.519727e+14  
## % Var explained: 36.67

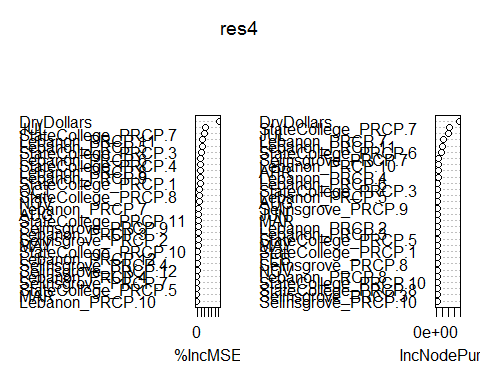
plot(res4) # looks pretty choppy?



res4$importance

## %IncMSE IncNodePurity  
## Year -1.257588e+12 7.485663e+12  
## DryDollars 9.678800e+13 7.803428e+14  
## StateCollege\_PRCP.1 1.240844e+12 2.148664e+13  
## Lebanon\_PRCP.1 -7.445366e+11 2.596882e+12  
## Selinsgrove\_PRCP.1 -3.230536e+11 4.479912e+12  
## StateCollege\_PRCP.2 -9.644629e+11 3.911407e+12  
## Lebanon\_PRCP.2 -1.329095e+12 2.328484e+13  
## Selinsgrove\_PRCP.2 4.291987e+10 6.013608e+12  
## StateCollege\_PRCP.3 3.532587e+12 3.838812e+13  
## Lebanon\_PRCP.3 1.751498e+11 2.296063e+13  
## Selinsgrove\_PRCP.3 -5.977370e+11 1.140791e+13  
## StateCollege\_PRCP.4 7.745024e+11 6.903249e+12  
## Lebanon\_PRCP.4 -4.886271e+11 4.337561e+13  
## Selinsgrove\_PRCP.4 -6.458586e+10 7.272525e+12  
## StateCollege\_PRCP.5 -3.474317e+11 2.252890e+13  
## Lebanon\_PRCP.5 1.787660e+12 3.338344e+13  
## Selinsgrove\_PRCP.5 -3.249469e+11 8.788137e+12  
## StateCollege\_PRCP.6 -2.078092e+12 1.440883e+14  
## Lebanon\_PRCP.6 1.998420e+12 4.085765e+13  
## Selinsgrove\_PRCP.6 -5.703824e+11 3.492520e+12  
## StateCollege\_PRCP.7 1.195760e+13 4.246825e+14  
## Lebanon\_PRCP.7 1.350897e+12 2.255572e+14  
## Selinsgrove\_PRCP.7 -3.911334e+11 7.280817e+13  
## StateCollege\_PRCP.8 5.521410e+11 1.238309e+13  
## Lebanon\_PRCP.8 8.623041e+11 1.352186e+13  
## Selinsgrove\_PRCP.8 -1.139597e+12 2.015877e+13  
## StateCollege\_PRCP.9 -2.293295e+11 1.128118e+12  
## Lebanon\_PRCP.9 1.524880e+11 4.515581e+12  
## Selinsgrove\_PRCP.9 3.342746e+11 3.254731e+13  
## StateCollege\_PRCP.10 -4.034864e+10 1.262548e+13  
## Lebanon\_PRCP.10 -1.118145e+12 6.069811e+13  
## Selinsgrove\_PRCP.10 -8.974447e+11 1.061688e+13  
## StateCollege\_PRCP.11 2.407377e+11 9.330622e+12  
## Lebanon\_PRCP.11 9.777438e+12 1.977376e+14  
## Selinsgrove\_PRCP.11 -2.265589e+11 1.633435e+12  
## StateCollege\_PRCP.12 -9.879181e+10 4.526553e+12  
## Lebanon\_PRCP.12 -1.078657e+10 1.045181e+12  
## Selinsgrove\_PRCP.12 -3.163922e+10 7.441735e+11  
## JAN -5.484841e+11 5.226413e+12  
## FEB -9.897391e+11 2.126656e+13  
## MAR -1.918309e+11 3.217588e+13  
## APR -2.598753e+12 5.485225e+13  
## MAY 1.752739e+11 2.181556e+13  
## JUN -3.214735e+12 3.250348e+13  
## JUL 1.390737e+13 4.041759e+14  
## AUG 3.842642e+11 3.334031e+13  
## SEP -5.545358e+11 9.993260e+12  
## OCT 1.633435e+11 1.645026e+12  
## NOV 2.504988e+11 1.359481e+13  
## DEC -1.642402e+11 5.942526e+11

varImpPlot(res4)



End of Script