## Random Forests

### Libraries

library(tidyverse)  
library(tidymodels)  
library(caret)  
library(gridExtra)  
library(vip)  
library(ranger)  
library(skimr)

drug = read\_csv("drug\_data-1.csv")

names(drug) = c("ID", "Age", "Gender", "Education", "Country", "Ethnicity","Nscore", "Escore", "Oscore", "Ascore", "Cscore", "Impulsive","SS", "Alcohol", "Amphet", "Amyl", "Benzos", "Caff", "Cannabis","Choc", "Coke", "Crack", "Ecstasy", "Heroin", "Ketamine", "Legalh","LSD", "Meth", "Mushrooms", "Nicotine", "Semer", "VSA")  
  
#str(drug)

drug[drug == "CL0"] = "No"  
drug[drug == "CL1"] = "No"  
drug[drug == "CL2"] = "Yes"  
drug[drug == "CL3"] = "Yes"  
drug[drug == "CL4"] = "Yes"  
drug[drug == "CL5"] = "Yes"  
drug[drug == "CL6"] = "Yes"

drug\_clean = drug %>%   
 mutate\_at(vars(Age:Ethnicity), funs(as\_factor)) %>%  
 mutate(Age = factor(Age, labels = c("18\_24", "25\_34", "35\_44","45\_54", "55\_64", "65\_"))) %>%  
 mutate(Gender = factor(Gender, labels = c("Male", "Female"))) %>%  
 mutate(Education = factor(Education, labels =c("Under16", "At16", "At17", "At18", "SomeCollege","ProfessionalCert", "Bachelors", "Masters", "Doctorate"))) %>%  
 mutate(Country = factor(Country,labels = c("USA", "NewZealand", "Other", "Australia","Ireland","Canada","UK"))) %>%  
 mutate(Ethnicity = factor(Ethnicity,labels = c("Black", "Asian", "White", "White/Black", "Other","White/Asian", "Black/Asian"))) %>%  
mutate\_at(vars(Alcohol:VSA), funs(as\_factor)) %>%  
 select(-ID)

#str(drug\_clean)

drug\_clean = drug\_clean %>%   
 select(!(Alcohol:Mushrooms)) %>%   
 select(!(Semer:VSA))  
names(drug\_clean)

## [1] "Age" "Gender" "Education" "Country" "Ethnicity" "Nscore"   
## [7] "Escore" "Oscore" "Ascore" "Cscore" "Impulsive" "SS"   
## [13] "Nicotine"

### Task 1: Missing data

There is no missing data

# skim(drug\_clean)  
# summary(drug\_clean)

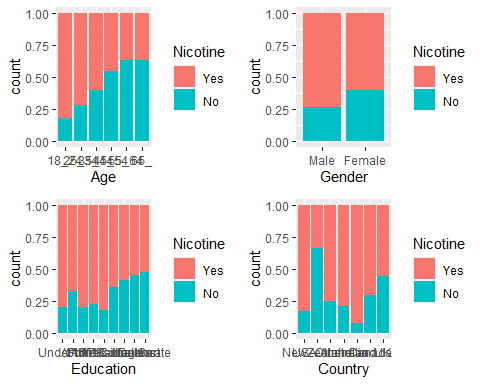
### Task 2: Split

set.seed(1234)  
drug\_split = initial\_split(drug\_clean, prop = 0.7, strata = Nicotine)  
train = training(drug\_split)  
test = testing(drug\_split)

### Task 3

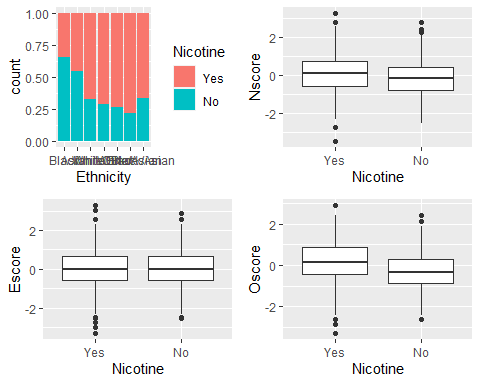
Age certainly appears to be a factor. Younger age categories are more likely to use Nicotine  
Males seem to be more likely to use Nicotine  
Education and Country also appear to have an effect

p1 = ggplot(train, aes(x = Age, fill = Nicotine)) + geom\_bar(position = "fill")  
p2 = ggplot(train, aes(x = Gender, fill = Nicotine)) + geom\_bar(position = "fill")  
p3 = ggplot(train, aes(x = Education, fill = Nicotine)) + geom\_bar(position = "fill")  
p4 = ggplot(train, aes(x = Country, fill = Nicotine)) + geom\_bar(position = "fill")  
grid.arrange(p1,p2,p3,p4)



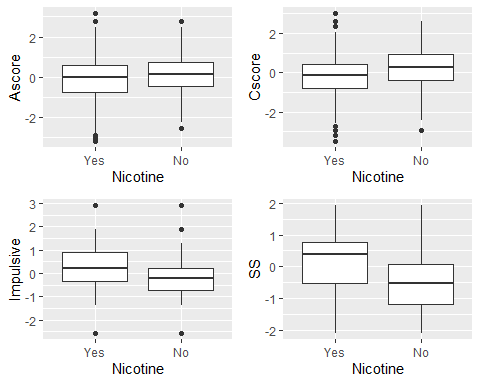
Ethnicity does appear to be a factor in Nicotine use Neuroticism, extroversion, and openness do not seem to have a strong relationship to Nicotine use. The medians and ranges are similar between groups

p1 = ggplot(train, aes(x = Ethnicity, fill = Nicotine)) + geom\_bar(position = "fill")  
p2 = ggplot(train, aes(x = Nicotine, y = Nscore)) + geom\_boxplot()  
p3 = ggplot(train, aes(x = Nicotine, y = Escore)) + geom\_boxplot()  
p4 = ggplot(train, aes(x = Nicotine, y = Oscore)) + geom\_boxplot()  
grid.arrange(p1,p2,p3,p4)



Agreeableness differs slightly between Nicotine groups. Those that do not use Nicotine are slightly more agreeable.  
Conscientiousness is also slightly lower in Nicotine users  
Impulsiveness and “sensation seeking” are higher in Nicotine users

p1 = ggplot(train, aes(x = Nicotine, y = Ascore)) + geom\_boxplot()  
p2 = ggplot(train, aes(x = Nicotine, y = Cscore)) + geom\_boxplot()  
p3 = ggplot(train, aes(x = Nicotine, y = Impulsive)) + geom\_boxplot()  
p4 = ggplot(train, aes(x = Nicotine, y = SS)) + geom\_boxplot()  
grid.arrange(p1,p2,p3,p4)

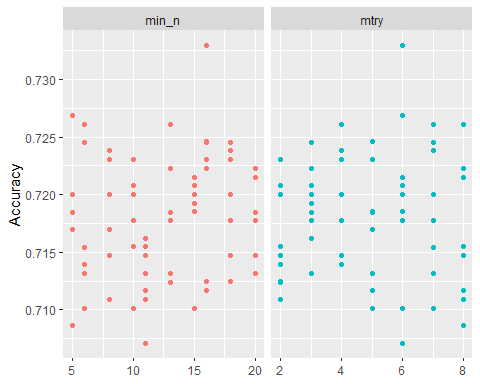


### Task 4: Random forest

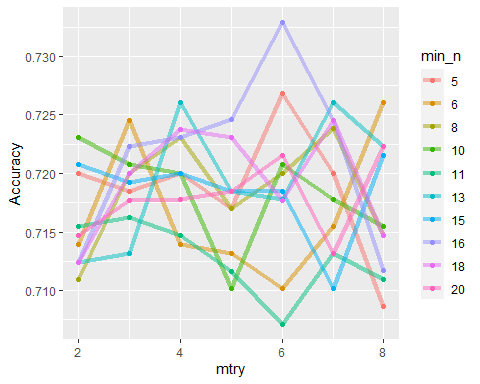
set.seed(123)  
rf\_folds = vfold\_cv(train, v=5)

drug\_recipe = recipe(Nicotine ~ ., train) %>%  
 step\_dummy(all\_nominal(), -all\_outcomes())  
  
rf\_model = rand\_forest(mtry = tune(), min\_n = tune(), trees = 100) %>%  
 set\_engine("ranger", importance = "permutation") %>%  
 set\_mode("classification")  
  
drug\_wflow =  
 workflow() %>%  
 add\_model(rf\_model) %>%  
 add\_recipe(drug\_recipe)  
  
rf\_grid = grid\_regular(  
 mtry(range = c(2, 8)),  
 min\_n(range = c(5,20)),  
 levels = 10  
)  
  
set.seed(123)  
rf\_res\_tuned = tune\_grid(  
 drug\_wflow,  
 resamples = rf\_folds,  
 grid = rf\_grid  
)

rf\_res\_tuned %>%  
 collect\_metrics() %>%  
 filter(.metric == "accuracy") %>%  
 select(mean, min\_n, mtry) %>%  
 pivot\_longer(min\_n:mtry,  
 values\_to = "value",  
 names\_to = "parameter"  
 ) %>%  
 ggplot(aes(value, mean, color = parameter)) +   
 geom\_point (show.legend = FALSE) +   
 facet\_wrap(~parameter, scales = "free\_x") +  
 labs(x = NULL, y = "Accuracy")



rf\_res\_tuned %>%  
 collect\_metrics() %>%  
 filter(.metric =="accuracy") %>%  
 mutate(min\_n = factor(min\_n)) %>%  
 ggplot(aes(mtry, mean, color = min\_n)) +  
 geom\_line(alpha = 0.5, size = 1.5) +   
 geom\_point() +   
 labs(y="Accuracy")

 ### Task 5

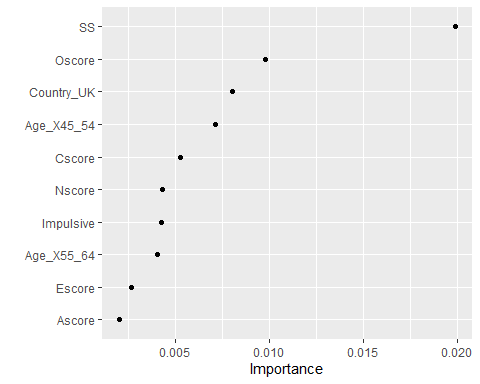
best\_rf = select\_best(rf\_res\_tuned, "accuracy")  
  
final\_rf = finalize\_workflow(  
 drug\_wflow,  
 best\_rf  
)  
  
final\_rf

## == Workflow ====================================================================  
## Preprocessor: Recipe  
## Model: rand\_forest()  
##   
## -- Preprocessor ----------------------------------------------------------------  
## 1 Recipe Step  
##   
## \* step\_dummy()  
##   
## -- Model -----------------------------------------------------------------------  
## Random Forest Model Specification (classification)  
##   
## Main Arguments:  
## mtry = 6  
## trees = 100  
## min\_n = 16  
##   
## Engine-Specific Arguments:  
## importance = permutation  
##   
## Computational engine: ranger

final\_rf\_fit = fit(final\_rf, train)

final\_rf\_fit %>% pull\_workflow\_fit() %>% vip(geom = "point")

## Warning: `pull\_workflow\_fit()` was deprecated in workflows 0.2.3.  
## Please use `extract\_fit\_parsnip()` instead.



**SS appears to be the most important variable in predicting Nicotine use. This Makes sense when comparing this result to the box plot of Nicotine vs SS above in which the median was much lower for non-Nicotine users. It also logically makes sense that this would be a personality trait that would lead to Nicotine use. Openness to Experience, living in the UK and being age 45-54 are the next most important variables. Openness makes sense according to the chart above however the bar charts do not suggest that being in the UK or being age 45-55 is a strong predictor or Nicotine use compared ot other countries and age groups**

### Task 6

The model performed very well on the training set with an accuracy of 91.6% which is significantly higher than the no information rate accuracy of 67% which would classify everyone as a Nicotine user. The model failed in the testing set however with an accuracy of 69.66% which is no longer a statistically significant improvement from the NIR.

trainpredrf = predict(final\_rf\_fit, train)  
head(trainpredrf)

## # A tibble: 6 x 1  
## .pred\_class  
## <fct>   
## 1 No   
## 2 No   
## 3 No   
## 4 No   
## 5 No   
## 6 No

confusionMatrix(trainpredrf$.pred\_class, train$Nicotine, positive = "Yes")

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction Yes No  
## Yes 871 97  
## No 13 337  
##   
## Accuracy : 0.9165   
## 95% CI : (0.9003, 0.9309)  
## No Information Rate : 0.6707   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.8013   
##   
## Mcnemar's Test P-Value : 2.498e-15   
##   
## Sensitivity : 0.9853   
## Specificity : 0.7765   
## Pos Pred Value : 0.8998   
## Neg Pred Value : 0.9629   
## Prevalence : 0.6707   
## Detection Rate : 0.6608   
## Detection Prevalence : 0.7344   
## Balanced Accuracy : 0.8809   
##   
## 'Positive' Class : Yes   
##

testpredrf = predict(final\_rf\_fit, test)  
head(testpredrf)

## # A tibble: 6 x 1  
## .pred\_class  
## <fct>   
## 1 No   
## 2 Yes   
## 3 No   
## 4 No   
## 5 No   
## 6 No

confusionMatrix(testpredrf$.pred\_class, test$Nicotine, positive = "Yes")

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction Yes No  
## Yes 327 119  
## No 53 68  
##   
## Accuracy : 0.6966   
## 95% CI : (0.657, 0.7343)  
## No Information Rate : 0.6702   
## P-Value [Acc > NIR] : 0.09699   
##   
## Kappa : 0.2462   
##   
## Mcnemar's Test P-Value : 7.188e-07   
##   
## Sensitivity : 0.8605   
## Specificity : 0.3636   
## Pos Pred Value : 0.7332   
## Neg Pred Value : 0.5620   
## Prevalence : 0.6702   
## Detection Rate : 0.5767   
## Detection Prevalence : 0.7866   
## Balanced Accuracy : 0.6121   
##   
## 'Positive' Class : Yes   
##

## Task 7

I would not recommend this model for real world use. While the accuracy was very high for the training set, it fell significantly with the testing set data. This suggests that over-fitting has occurred.