# BAN 502 - Random Forest

### Khayrayyah Haamid-Day

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)

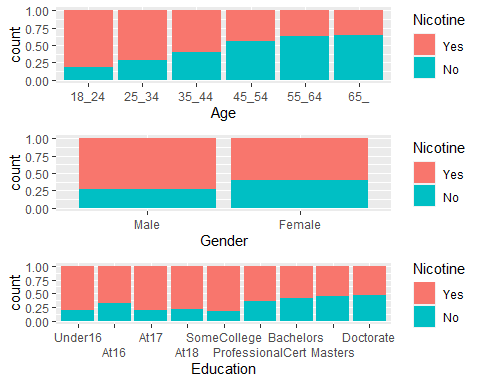
## Warning: `funs()` was deprecated in dplyr 0.8.0.  
## Please use a list of either functions or lambdas:   
##   
## # Simple named list:   
## list(mean = mean, median = median)  
##   
## # Auto named with `tibble::lst()`:   
## tibble::lst(mean, median)  
##   
## # Using lambdas  
## list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was generated.

#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"

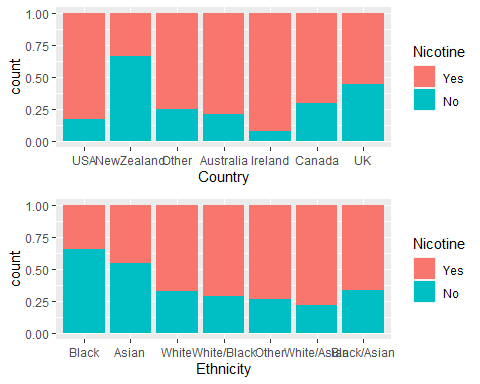
#Task 1  
#skim(drug\_clean)  
  
#Task 2  
  
set.seed(1234)  
drugclean\_split = initial\_split(drug\_clean, prop = 0.7, strata = Nicotine)  
train = training(drugclean\_split)  
test = testing(drugclean\_split)  
  
#str(drug\_clean)

#Task 3  
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") +  
 scale\_x\_discrete(guide = guide\_axis(n.dodge=2))  
grid.arrange(p1,p2,p3)



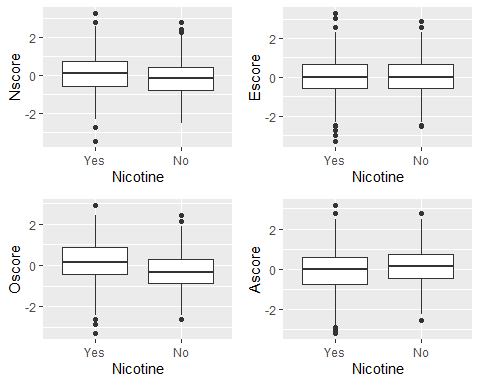
This shows that younger people and females are more likely to use nicotine. In terms of education, it looks like less education increases liklihood of using nicotine.

p5 = ggplot(train, aes(x = Ethnicity, fill = Nicotine)) + geom\_bar(position = "fill")  
p4 = ggplot(train, aes(x = Country, fill = Nicotine)) + geom\_bar(position = "fill")  
grid.arrange(p4,p5)



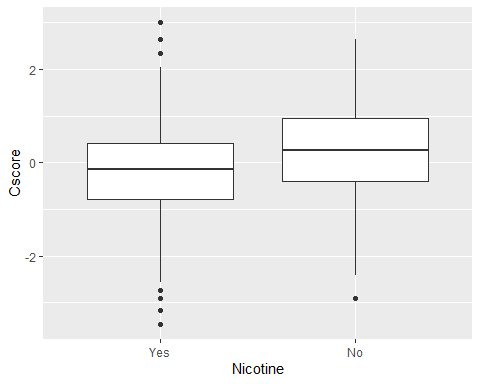
Most countries have higher amounts of nicotine use except for New Zealand where there is significantly lower nicotine use. Asian and black ethnicities have less nicotine use compared to the other ethnicities.

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

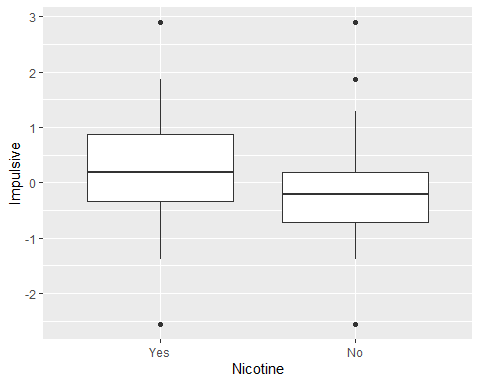


These scores do not appear to have a significant impact on nicotine use. There are some outliers.

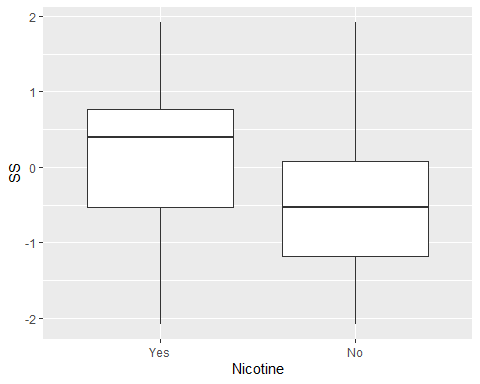
p5 = ggplot(train, aes(x = Nicotine, y = Cscore)) + geom\_boxplot()  
p6 = ggplot(train, aes(x = Nicotine, y = Impulsive)) + geom\_boxplot()  
p7 = ggplot(train, aes(x = Nicotine, y = SS)) + geom\_boxplot()  
(p5)



(p6)



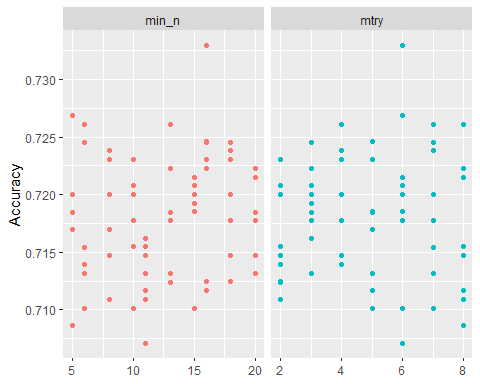
(p7)



It appears there is a relationship with nicotine use, SS and Impulsiveness.

#Task 4   
set.seed(123)  
rf\_folds = vfold\_cv(train, v = 5)  
  
nicotine\_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")  
  
nicotine\_wflow =   
 workflow() %>%   
 add\_model(rf\_model) %>%   
 add\_recipe(nicotine\_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(  
 nicotine\_wflow,  
 resamples = rf\_folds,  
 grid = rf\_grid   
)

#Visualization  
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")

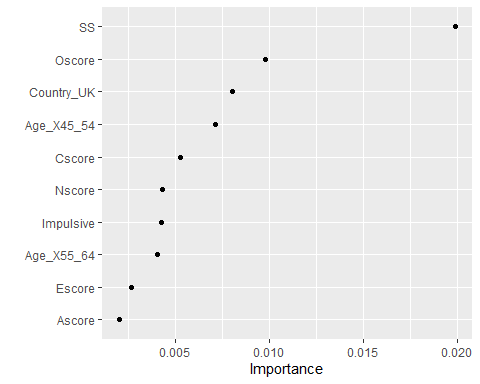


#Task 5   
best\_rf = select\_best(rf\_res\_tuned, "accuracy")  
  
final\_rf = finalize\_workflow(  
 nicotine\_wflow,  
 best\_rf  
)  
  
final\_rf

## == 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.



The top most important variables appear to be sensation seeing, openness to experiences, being from the UK, and an age range of 45-54

#Task 6  
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   
##

The accuracy drops a lot from the training set being 91% to 69% in the testing set, indicating a overfit model.

**Task 7**

This model could be used to predict a persons likelihood of using nicotine and as a result, target those with increased risk. I would recommend this model for real-world use but I would be concerned with the overfitting so I believe there would need to be some tweaking.