Random Forests

Morgan Nanni

2/10/2022

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)

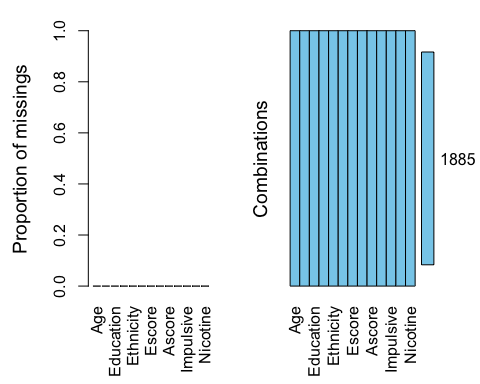
## Warning: `funs()` was deprecated in dplyr 0.8.0.  
## Please use a list of either functions or lambdas:   
##   
## # Simple named list:   
## list(mean = mean, median = median)  
##   
## # Auto named with `tibble::lst()`:   
## tibble::lst(mean, median)  
##   
## # Using lambdas  
## list(~ mean(., 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)

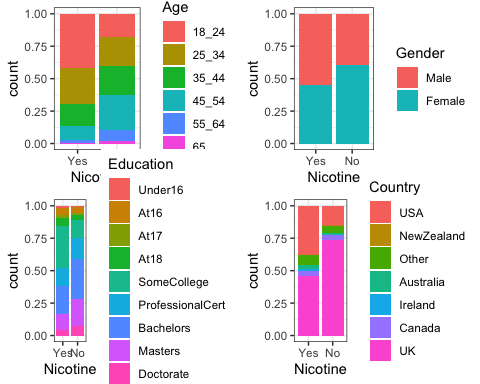
## [1] "Age" "Gender" "Education" "Country" "Ethnicity" "Nscore"   
## [7] "Escore" "Oscore" "Ascore" "Cscore" "Impulsive" "SS"   
## [13] "Nicotine"

vim\_plot = aggr(drug\_clean, numbers=TRUE, prop=c(TRUE,FALSE),cex.asix=.7)

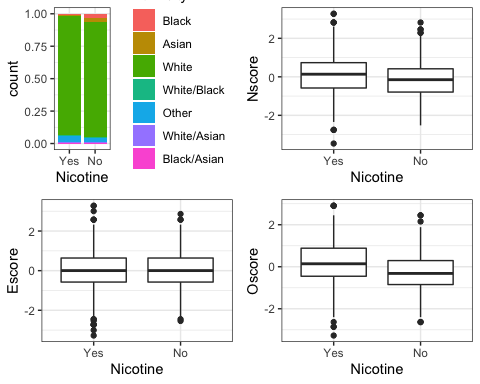
 There is no missing data.

set.seed(1234)  
drug\_clean\_split = initial\_split(drug\_clean, prop=0.70, strata=Nicotine)  
train = training(drug\_clean\_split)  
test = testing(drug\_clean\_split)

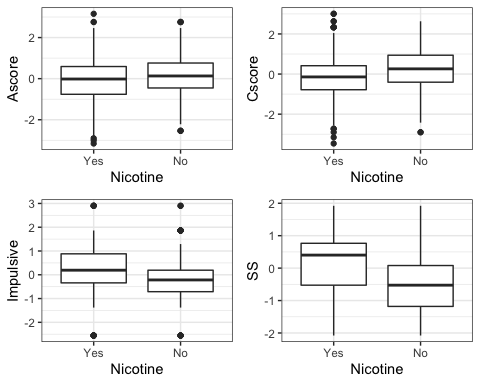
p1 = ggplot(train,aes(x=Nicotine, fill=Age)) + geom\_bar(position="fill") + theme\_bw()  
p2 = ggplot(train, aes(x=Nicotine, fill=Gender)) + geom\_bar(position="fill") + theme\_bw()  
p3 = ggplot(train, aes(x=Nicotine, fill=Education)) + geom\_bar(position="fill") + theme\_bw()  
p4 = ggplot(train, aes(x=Nicotine, fill=Country)) + geom\_bar(position="fill") + theme\_bw()  
grid.arrange(p1,p2,p3,p4, ncol=2)



p5 = ggplot(train, aes(x=Nicotine, fill=Ethnicity)) + geom\_bar(position="fill") + theme\_bw()  
p6 = ggplot(train, aes(x=Nicotine, y=Nscore)) + geom\_boxplot() + theme\_bw()  
p7 = ggplot(train, aes(x=Nicotine, y=Escore)) + geom\_boxplot() + theme\_bw()  
p8 = ggplot(train, aes(x=Nicotine, y=Oscore)) + geom\_boxplot() + theme\_bw()  
grid.arrange(p5,p6,p7,p8, ncol=2)



p9 = ggplot(train, aes(x=Nicotine, y=Ascore)) + geom\_boxplot() + theme\_bw()  
p10 = ggplot(train, aes(x=Nicotine, y=Cscore)) + geom\_boxplot() + theme\_bw()  
p11 = ggplot(train, aes(x=Nicotine, y=Impulsive)) + geom\_boxplot() + theme\_bw()  
p12 = ggplot(train, aes(x=Nicotine, y=SS)) + geom\_boxplot() + theme\_bw()  
grid.arrange(p9,p10,p11,p12, ncol=2)

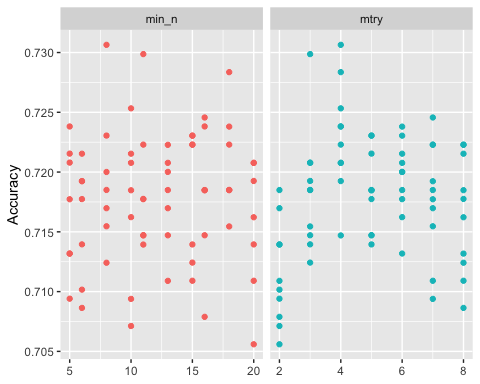


People of younger age (18-34) seem to be more likely to consumer nicotine than those of older ages. Male and Female are pretty similar in regards to nicotine consumption with males being a little more likely to consume nicotine. The responses of “yes” and “no” for the different levels of education are pretty similar with SomeCollege having larger response of “yes” to nicotine consumption than “no”. The UK and USA make up the majority of the data set for country and UK has higher responses of no nicotine consumption and USA has higher responses of yes to nicotine consumption. Ethnicity is almost equal between “yes” and “no” responses. People with higher Nscores are more likely to consume nicotine. Escore has no significant impact on nicotine consumption. Higher Oscores are shown to be more likely to consume nicotine. There doesn’t seem to be a significant impact on nicotine consumption from Ascores, but no nicotine consumption are shown to have slightly higher Ascores. Higher Cscores are likely to have no nicotine consumption and higher impulsiveness is likely to have nicotine consumption. Lastly, higher SS levels is shown to be more likely to consume nicotine.

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")



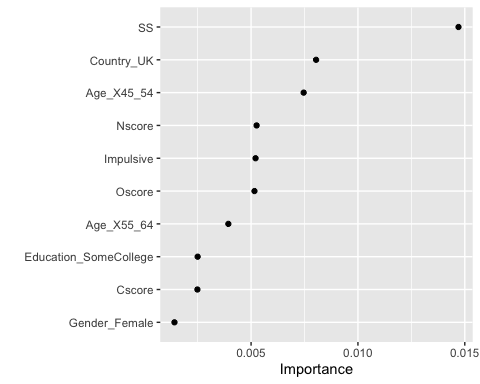
best\_rf = select\_best(rf\_res\_tuned, "accuracy")  
  
final\_rf = finalize\_workflow(  
 drug\_wflow,  
 best\_rf  
)  
  
final\_rf

## ══ Workflow ════════════════════════════════════════════════════════════════════  
## Preprocessor: Recipe  
## Model: rand\_forest()  
##   
## ── Preprocessor ────────────────────────────────────────────────────────────────  
## 1 Recipe Step  
##   
## • step\_dummy()  
##   
## ── Model ───────────────────────────────────────────────────────────────────────  
## Random Forest Model Specification (classification)  
##   
## Main Arguments:  
## mtry = 4  
## trees = 100  
## min\_n = 8  
##   
## 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 is the most important which represents sensation, this makes sense as a higher level of sensation could drive motivation for nicotine consumption. The second most important is country, countries have different laws and social norms which could significantly impact the amount of nicotine consumption. The third most important is age, people of older age are less likely to consume nicotine due to health issues as opposed to younger ages who are less concerned with their health.

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

## # A tibble: 6 × 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 879 78  
## No 5 356  
##   
## Accuracy : 0.937   
## 95% CI : (0.9225, 0.9495)  
## No Information Rate : 0.6707   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.8511   
##   
## Mcnemar's Test P-Value : 2.722e-15   
##   
## Sensitivity : 0.9943   
## Specificity : 0.8203   
## Pos Pred Value : 0.9185   
## Neg Pred Value : 0.9861   
## Prevalence : 0.6707   
## Detection Rate : 0.6669   
## Detection Prevalence : 0.7261   
## Balanced Accuracy : 0.9073   
##   
## 'Positive' Class : Yes   
##

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

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

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

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction Yes No  
## Yes 332 115  
## No 48 72  
##   
## Accuracy : 0.7125   
## 95% CI : (0.6733, 0.7495)  
## No Information Rate : 0.6702   
## P-Value [Acc > NIR] : 0.01708   
##   
## Kappa : 0.2846   
##   
## Mcnemar's Test P-Value : 2.347e-07   
##   
## Sensitivity : 0.8737   
## Specificity : 0.3850   
## Pos Pred Value : 0.7427   
## Neg Pred Value : 0.6000   
## Prevalence : 0.6702   
## Detection Rate : 0.5855   
## Detection Prevalence : 0.7884   
## Balanced Accuracy : 0.6294   
##   
## 'Positive' Class : Yes   
##

The accuracy of this model on the training set is 0.94 and the accuracy on the testing set is 0.71. The model performs well on the training set but there is a much lower accuracy on the testing set implying it did not perform as well.

This model could be used in real-world scenarios like in healthcare, disease prevention or even stock market predictions. I would recommend this model for real-world use because it does well with missing data while still maintaining the accuracy of the whole data set and gives higher accuracy from using cross validation. While using this model for the assignment I didn’t really notice any concerns that would carry over in to real-world scenarios.