Module4Assignment2

Engel, Alec

drug = read\_csv("drug\_data.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\_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)

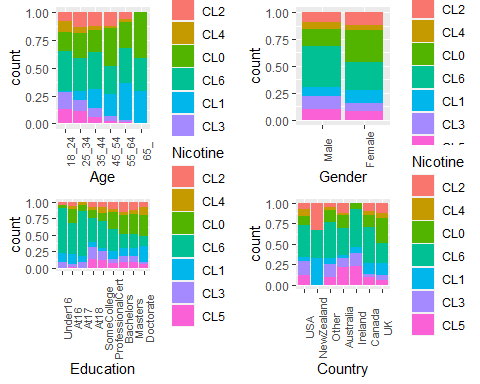
## Warning: `funs()` is deprecated as of dplyr 0.8.0.  
## Please use a list of either functions or lambdas:   
##   
## # Simple named list:   
## list(mean = mean, median = median)  
##   
## # Auto named with `tibble::lst()`:   
## tibble::lst(mean, median)  
##   
## # Using lambdas  
## list(~ mean(.,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

#str(drug\_clean)  
#skim(drug\_clean)

**No further missingness identified.**

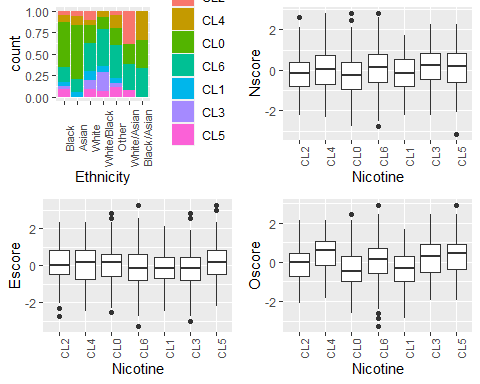
set.seed(1234)   
drug\_split = initial\_split(drug\_clean, prop = 0.7, strata = Nicotine)  
train = training(drug\_split)  
test = testing(drug\_split)

p1 = ggplot(train, aes(x = Age, fill = Nicotine)) + geom\_bar(position = "fill") + theme(axis.text.x = element\_text(angle = 90, size = 8))  
p2 = ggplot(train, aes(x = Gender, fill = Nicotine)) + geom\_bar(position = "fill") + theme(axis.text.x = element\_text(angle = 90, size = 8))  
p3 = ggplot(train, aes(x = Education, fill = Nicotine)) + geom\_bar(position = "fill") + theme(axis.text.x = element\_text(angle = 90, size = 8))  
p4 = ggplot(train, aes(x = Country, fill = Nicotine)) + geom\_bar(position = "fill") + theme(axis.text.x = element\_text(angle = 90, size = 8))  
grid.arrange(p1,p2,p3,p4)



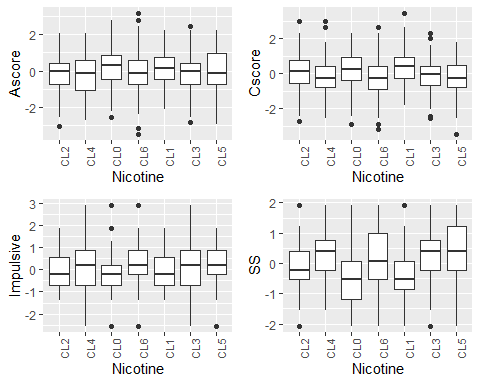
**With these visualizations, we can see that as age increases, the number of people who never used nicotine or used over a decade ago increases while the number of people who have used nicotine within the last decade or sooner greatly decreases. This obviously leads to a shorter life span with nicotine use. Nicotine use within the last day seems more prevalent in males and females have a larger portion of people that nave never used. With education, it appears that the more educated someone is, the less likely they are to have used nicotine in the last 10 years or even at all. The highest nicotine used within the last day looks to be with kids 17 and under. Daily nicotine users is relatively similar in each of these countries while New Zealand seems to have the lowest rate, if not zero nicotine users within the last decade. At a quick glance, Ireland appears to have the most prevalent nicotine use.**

p1 = ggplot(train, aes(x = Ethnicity, fill = Nicotine)) + geom\_bar(position = "fill") + theme(axis.text.x = element\_text(angle = 90, size = 8))  
p2 = ggplot(train, aes(x = Nicotine, y = Nscore)) + geom\_boxplot() + theme(axis.text.x = element\_text(angle = 90, size = 8))  
p3 = ggplot(train, aes(x = Nicotine, y = Escore)) + geom\_boxplot() + theme(axis.text.x = element\_text(angle = 90, size = 8))  
p4 = ggplot(train, aes(x = Nicotine, y = Oscore)) + geom\_boxplot() + theme(axis.text.x = element\_text(angle = 90, size = 8))  
grid.arrange(p1,p2,p3,p4)



**Individually, Black and Asian ethnicities look to have the highest amount of lifetime non-nicotine users and there seems to be much more prevalent nicotine usage in White, mixed racial and other ethnicities. Neurotism seems to be relatively consistent among nicotine users while we can observe a lower rate amongst those who have not used nicotine within the last decade or greater. Extraversion doesn’t appear to be too much of a predictor variable for nicotine usage as it seems very constant. Openness definitely increases with those who have used nicotine in the last year or sooner.**

p1 = ggplot(train, aes(x = Nicotine, y = Ascore)) + geom\_boxplot() + theme(axis.text.x = element\_text(angle = 90, size = 8))  
p2 = ggplot(train, aes(x = Nicotine, y = Cscore)) + geom\_boxplot() + theme(axis.text.x = element\_text(angle = 90, size = 8))  
p3 = ggplot(train, aes(x = Nicotine, y = Impulsive)) + geom\_boxplot() + theme(axis.text.x = element\_text(angle = 90, size = 8))  
p4 = ggplot(train, aes(x = Nicotine, y = SS)) + geom\_boxplot() + theme(axis.text.x = element\_text(angle = 90, size = 8))  
grid.arrange(p1,p2,p3,p4, ncol = 2)



**Agreeableness seems to be somewhat consistent among all respondents with those who have not used either at all or in the last decade having a slightly higher score. Conscientiousness unsurprisingly rated much higher in the individuals who haven’t used at all or within the last decade. Impulsive also seems to align with higher and more frequent nicotine usage among more impulsive people. Sensation seeing follows these same lines where more sensation seeing respondents are more likely to have used nicotine within the last day, week, month, or year.**

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  
)

##   
## Attaching package: 'rlang'

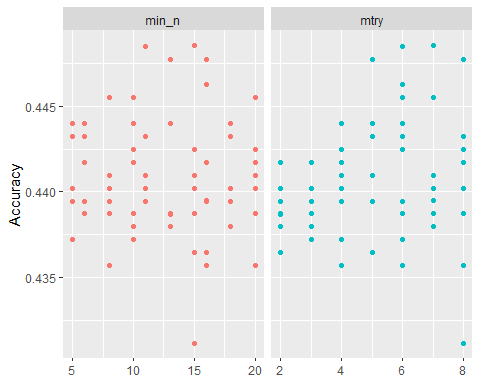
## The following objects are masked from 'package:purrr':  
##   
## %@%, as\_function, flatten, flatten\_chr, flatten\_dbl, flatten\_int,  
## flatten\_lgl, flatten\_raw, invoke, list\_along, modify, prepend,  
## splice

##   
## Attaching package: 'vctrs'

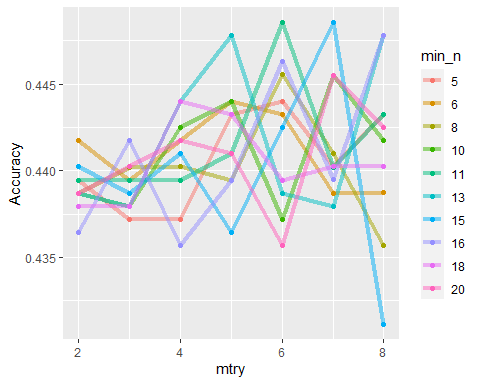
## The following object is masked from 'package:dplyr':  
##   
## data\_frame

## The following object is masked from 'package:tibble':  
##   
## data\_frame

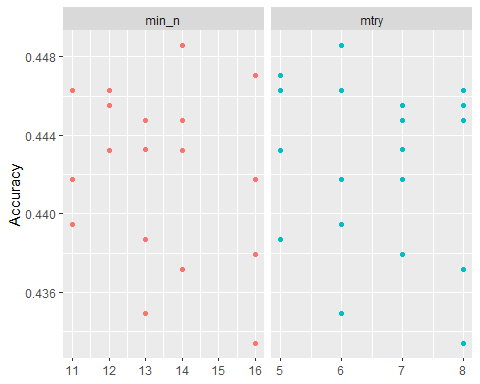
rf\_res\_tuned %>%  
 collect\_metrics() %>%  
 filter(.metric == "accuracy") %>%  
 select(mean, min\_n, mtry) %>%  
 pivot\_longer(min\_n:mtry,  
 values\_to = "value",  
 names\_to = "parameter"  
 ) %>%  
 ggplot(aes(value, mean, color = parameter)) +  
 geom\_point(show.legend = FALSE) +  
 facet\_wrap(~parameter,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



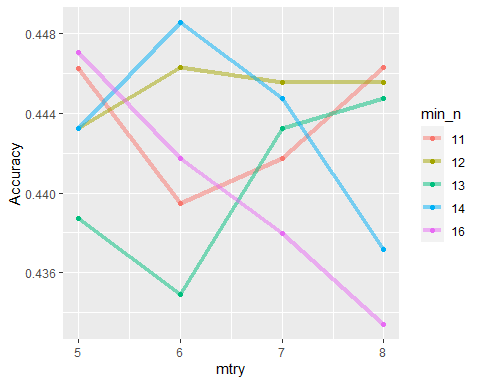
rf\_res\_tuned %>%  
 collect\_metrics() %>%  
 filter(.metric == "accuracy") %>%  
 mutate(min\_n = factor(min\_n)) %>%  
 ggplot(aes(mtry, mean, color = min\_n)) +  
 geom\_line(alpha = 0.5, size = 1.5) +  
 geom\_point() +  
 labs(y = "Accuracy")



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(5, 8)),  
 min\_n(range = c(11, 16)),  
 levels = 5  
)  
  
set.seed(123)  
rf\_res\_tuned = tune\_grid(  
 nicotine\_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")



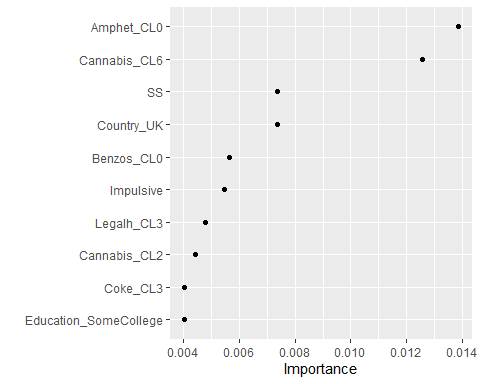
rf\_res\_tuned %>%  
 collect\_metrics() %>%  
 filter(.metric == "accuracy") %>%  
 mutate(min\_n = factor(min\_n)) %>%  
 ggplot(aes(mtry, mean, color = min\_n)) +  
 geom\_line(alpha = 0.5, size = 1.5) +  
 geom\_point() +  
 labs(y = "Accuracy")



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 = 14  
##   
## Engine-Specific Arguments:  
## importance = permutation  
##   
## Computational engine: ranger

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



**Here we learn that the most important variables for nicotine usage is if someone has used cannabis in the last day or if someone has never used amphetamines. Sensation seeing, impulsiveness, people who have never used benzodiazepine and people living in the UK are of relatively high importance to nicotine usage as well.**

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

## # A tibble: 6 x 1  
## .pred\_class  
## <fct>   
## 1 CL2   
## 2 CL4   
## 3 CL2   
## 4 CL6   
## 5 CL0   
## 6 CL6

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

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction CL2 CL4 CL0 CL6 CL1 CL3 CL5  
## CL2 62 1 0 0 1 1 0  
## CL4 0 27 0 0 0 0 0  
## CL0 35 13 274 23 44 12 11  
## CL6 39 38 21 400 19 49 45  
## CL1 1 0 0 0 77 0 2  
## CL3 0 0 0 0 0 69 0  
## CL5 0 0 0 0 0 0 58  
##   
## Overall Statistics  
##   
## Accuracy : 0.7315   
## 95% CI : (0.7067, 0.7552)  
## No Information Rate : 0.32   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.647   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: CL2 Class: CL4 Class: CL0 Class: CL6 Class: CL1  
## Sensitivity 0.45255 0.34177 0.9288 0.9456 0.54610  
## Specificity 0.99747 1.00000 0.8656 0.7653 0.99746  
## Pos Pred Value 0.95385 1.00000 0.6650 0.6547 0.96250  
## Neg Pred Value 0.94033 0.95985 0.9769 0.9677 0.94847  
## Prevalence 0.10363 0.05976 0.2231 0.3200 0.10666  
## Detection Rate 0.04690 0.02042 0.2073 0.3026 0.05825  
## Detection Prevalence 0.04917 0.02042 0.3116 0.4622 0.06051  
## Balanced Accuracy 0.72501 0.67089 0.8972 0.8555 0.77178  
## Class: CL3 Class: CL5  
## Sensitivity 0.52672 0.50000  
## Specificity 1.00000 1.00000  
## Pos Pred Value 1.00000 1.00000  
## Neg Pred Value 0.95052 0.95411  
## Prevalence 0.09909 0.08775  
## Detection Rate 0.05219 0.04387  
## Detection Prevalence 0.05219 0.04387  
## Balanced Accuracy 0.76336 0.75000

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

## # A tibble: 6 x 1  
## .pred\_class  
## <fct>   
## 1 CL6   
## 2 CL6   
## 3 CL0   
## 4 CL2   
## 5 CL6   
## 6 CL0

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

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction CL2 CL4 CL0 CL6 CL1 CL3 CL5  
## CL2 2 0 1 0 0 1 0  
## CL4 0 0 0 0 0 0 0  
## CL0 28 3 91 27 21 16 5  
## CL6 32 26 40 159 30 36 34  
## CL1 5 0 0 1 1 0 0  
## CL3 0 0 0 0 0 0 1  
## CL5 0 0 1 0 0 1 1  
##   
## Overall Statistics  
##   
## Accuracy : 0.4512   
## 95% CI : (0.4095, 0.4933)  
## No Information Rate : 0.3321   
## P-Value [Acc > NIR] : 3.01e-09   
##   
## Kappa : 0.2234   
##   
## Mcnemar's Test P-Value : NA   
##   
## Statistics by Class:  
##   
## Class: CL2 Class: CL4 Class: CL0 Class: CL6 Class: CL1  
## Sensitivity 0.029851 0.00000 0.6842 0.8503 0.019231  
## Specificity 0.995968 1.00000 0.7674 0.4734 0.988258  
## Pos Pred Value 0.500000 NaN 0.4764 0.4454 0.142857  
## Neg Pred Value 0.883721 0.94849 0.8871 0.8641 0.908273  
## Prevalence 0.119005 0.05151 0.2362 0.3321 0.092362  
## Detection Rate 0.003552 0.00000 0.1616 0.2824 0.001776  
## Detection Prevalence 0.007105 0.00000 0.3393 0.6341 0.012433  
## Balanced Accuracy 0.512909 0.50000 0.7258 0.6618 0.503745  
## Class: CL3 Class: CL5  
## Sensitivity 0.000000 0.024390  
## Specificity 0.998035 0.996169  
## Pos Pred Value 0.000000 0.333333  
## Neg Pred Value 0.903915 0.928571  
## Prevalence 0.095915 0.072824  
## Detection Rate 0.000000 0.001776  
## Detection Prevalence 0.001776 0.005329  
## Balanced Accuracy 0.499018 0.510279

**We see an accuracy in our training set of 73.15% while only an accuracy of 45.12% in our testing set. This may mean we have narrowed down our mtry and min\_n ranges a bit too far. This is probably the max distance of accuracy we would want to see from our training set to our testing set at just below 30%.**

**This model could easily be used to correlate the usage of multiple drugs by an individual or to help identify possible gateway drugs. I could also see this model being used to narrow down specific countries with nicotine issues(like the UK) and putting those up against possible disease or cancer statistics to try and find any sort of correlation. The neuroticism and other personality traits are also a very interesting aspect of this data and you can clearly see important variables like sensation seeing or impulsiveness leading to higher drug usage. This is a nice broad model but I would want thousands more observations prior to recommending for real world use. My main concern would be the distance of accuracy this model is from our training data to our testing data. That would allow me to foresee issues with future data and know that if the accuracy were any further than this 30% amount difference, we wouldn’t be able to trust the models we’ve created.**