drug <- read\_csv("~/Ban502/Module 4/drug\_data-1.csv")

##   
## -- Column specification --------------------------------------------------------  
## cols(  
## .default = col\_character(),  
## Column1 = col\_double(),  
## Column2 = col\_double(),  
## Column3 = col\_double(),  
## Column4 = col\_double(),  
## Column5 = col\_double(),  
## Column6 = col\_double(),  
## Column7 = col\_double(),  
## Column8 = col\_double(),  
## Column9 = col\_double(),  
## Column10 = col\_double(),  
## Column11 = col\_double(),  
## Column12 = col\_double(),  
## Column13 = col\_double()  
## )  
## i Use `spec()` for the full column specifications.

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()` 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)  
  
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"

str(drug\_clean)

## tibble [1,885 x 13] (S3: tbl\_df/tbl/data.frame)  
## $ Age : Factor w/ 6 levels "18\_24","25\_34",..: 3 2 3 1 3 6 4 3 3 5 ...  
## $ Gender : Factor w/ 2 levels "Male","Female": 2 1 1 2 2 2 1 1 2 1 ...  
## $ Education: Factor w/ 9 levels "Under16","At16",..: 6 9 6 8 9 4 8 2 6 8 ...  
## $ Country : Factor w/ 7 levels "USA","NewZealand",..: 7 7 7 7 7 6 1 7 6 7 ...  
## $ Ethnicity: Factor w/ 7 levels "Black","Asian",..: 6 3 3 3 3 3 3 3 3 3 ...  
## $ Nscore : num [1:1885] 0.313 -0.678 -0.467 -0.149 0.735 ...  
## $ Escore : num [1:1885] -0.575 1.939 0.805 -0.806 -1.633 ...  
## $ Oscore : num [1:1885] -0.5833 1.4353 -0.8473 -0.0193 -0.4517 ...  
## $ Ascore : num [1:1885] -0.917 0.761 -1.621 0.59 -0.302 ...  
## $ Cscore : num [1:1885] -0.00665 -0.14277 -1.0145 0.58489 1.30612 ...  
## $ Impulsive: num [1:1885] -0.217 -0.711 -1.38 -1.38 -0.217 ...  
## $ SS : num [1:1885] -1.181 -0.216 0.401 -1.181 -0.216 ...  
## $ Nicotine : Factor w/ 2 levels "Yes","No": 1 1 2 1 1 1 1 2 1 1 ...

summary(drug\_clean)

## Age Gender Education Country   
## 18\_24:643 Male :943 SomeCollege :506 USA : 557   
## 25\_34:481 Female:942 Bachelors :480 NewZealand: 5   
## 35\_44:356 Masters :283 Other : 118   
## 45\_54:294 ProfessionalCert:270 Australia : 54   
## 55\_64: 93 At18 :100 Ireland : 20   
## 65\_ : 18 At16 : 99 Canada : 87   
## (Other) :147 UK :1044   
## Ethnicity Nscore Escore Oscore   
## Black : 33 Min. :-3.464360 Min. :-3.273930 Min. :-3.273930   
## Asian : 26 1st Qu.:-0.678250 1st Qu.:-0.695090 1st Qu.:-0.717270   
## White :1720 Median : 0.042570 Median : 0.003320 Median :-0.019280   
## White/Black: 20 Mean : 0.000047 Mean :-0.000163 Mean :-0.000534   
## Other : 63 3rd Qu.: 0.629670 3rd Qu.: 0.637790 3rd Qu.: 0.723300   
## White/Asian: 20 Max. : 3.273930 Max. : 3.273930 Max. : 2.901610   
## Black/Asian: 3   
## Ascore Cscore Impulsive   
## Min. :-3.464360 Min. :-3.464360 Min. :-2.555240   
## 1st Qu.:-0.606330 1st Qu.:-0.652530 1st Qu.:-0.711260   
## Median :-0.017290 Median :-0.006650 Median :-0.217120   
## Mean :-0.000245 Mean :-0.000386 Mean : 0.007216   
## 3rd Qu.: 0.760960 3rd Qu.: 0.584890 3rd Qu.: 0.529750   
## Max. : 3.464360 Max. : 3.464360 Max. : 2.901610   
##   
## SS Nicotine   
## Min. :-2.078480 Yes:1264   
## 1st Qu.:-0.525930 No : 621   
## Median : 0.079870   
## Mean :-0.003292   
## 3rd Qu.: 0.765400   
## Max. : 1.921730   
##

skim(drug\_clean)

Data summary

|  |  |
| --- | --- |
| Name | drug\_clean |
| Number of rows | 1885 |
| Number of columns | 13 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| factor | 6 |
| numeric | 7 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: factor**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| skim\_variable | n\_missing | complete\_rate | ordered | n\_unique | top\_counts |
| Age | 0 | 1 | FALSE | 6 | 18\_: 643, 25\_: 481, 35\_: 356, 45\_: 294 |
| Gender | 0 | 1 | FALSE | 2 | Mal: 943, Fem: 942 |
| Education | 0 | 1 | FALSE | 9 | Som: 506, Bac: 480, Mas: 283, Pro: 270 |
| Country | 0 | 1 | FALSE | 7 | UK: 1044, USA: 557, Oth: 118, Can: 87 |
| Ethnicity | 0 | 1 | FALSE | 7 | Whi: 1720, Oth: 63, Bla: 33, Asi: 26 |
| Nicotine | 0 | 1 | FALSE | 2 | Yes: 1264, No: 621 |

**Variable type: numeric**

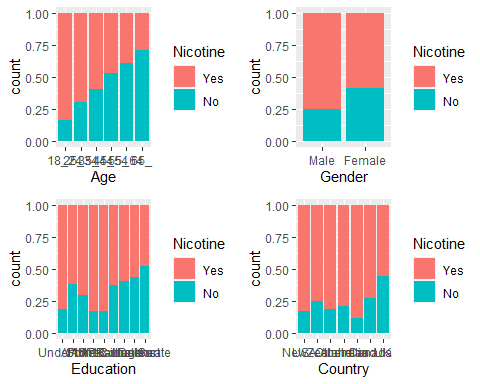
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| Nscore | 0 | 1 | 0.00 | 1.00 | -3.46 | -0.68 | 0.04 | 0.63 | 3.27 | ▁▃▇▅▁ |
| Escore | 0 | 1 | 0.00 | 1.00 | -3.27 | -0.70 | 0.00 | 0.64 | 3.27 | ▁▃▇▃▁ |
| Oscore | 0 | 1 | 0.00 | 1.00 | -3.27 | -0.72 | -0.02 | 0.72 | 2.90 | ▁▃▇▆▁ |
| Ascore | 0 | 1 | 0.00 | 1.00 | -3.46 | -0.61 | -0.02 | 0.76 | 3.46 | ▁▃▇▃▁ |
| Cscore | 0 | 1 | 0.00 | 1.00 | -3.46 | -0.65 | -0.01 | 0.58 | 3.46 | ▁▃▇▃▁ |
| Impulsive | 0 | 1 | 0.01 | 0.95 | -2.56 | -0.71 | -0.22 | 0.53 | 2.90 | ▁▆▇▃▁ |
| SS | 0 | 1 | 0.00 | 0.96 | -2.08 | -0.53 | 0.08 | 0.77 | 1.92 | ▂▇▇▇▅ |

### Task 1 Response:

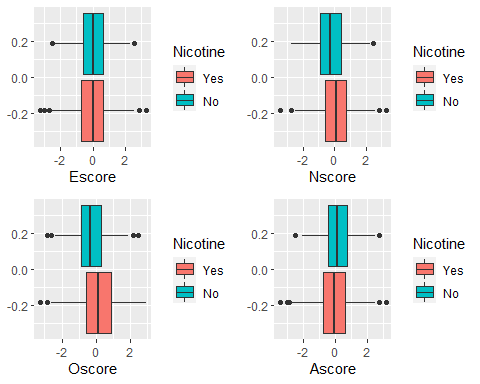
By skimming and reviewing the summary of the data, no missingness could be found at this point.

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

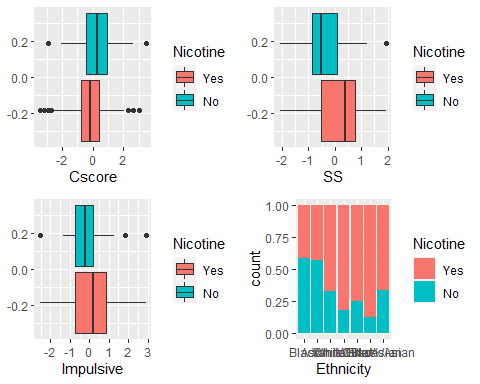
p1 = ggplot(train, aes(x = Age, fill = Nicotine)) + geom\_bar(position = "fill")  
p2 = ggplot(train, aes(x = Gender, fill = Nicotine)) + geom\_bar(position = "fill")  
p3 = ggplot(train, aes(x = Education, fill = Nicotine)) + geom\_bar(position = "fill")  
p4 = ggplot(train, aes(x = Country, fill =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



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



p1 = ggplot(train, aes(x = Cscore, fill = Nicotine)) + geom\_boxplot()  
p2 = ggplot(train, aes(x = SS, fill = Nicotine)) + geom\_boxplot()  
p3 = ggplot(train, aes(x = Impulsive, fill = Nicotine)) + geom\_boxplot()  
p4 = ggplot(train, aes(x = Ethnicity, fill = Nicotine)) + geom\_bar(position = "fill")  
grid.arrange(p1,p2,p3,p4)



### Task 3 Response:

Age: There appears to be a noticeable decrease in the number of nicotine users by age group.  
Gender: While there are fewer females that report nicotine use, it is not drastically different than men.  
Education: No consistent pattern overall, although the number of users appears to decrease with each level of graduate degree.  
Country: No consistent pattern, but UK appeasr to have significantly fewer people to report nicotine use.  
Escore: Not a significant difference in medians. Nscore: Small gaps in medians, does not appear very significant.  
Oscore: A moderate discrepancy in medians, with higher scores indicating more nicotine use. Ethnicity.  
Ascore: Small gaps in medians, does not appear very significant.  
Cscore: A moderate discrepancy in medians, with higher scores indicating more nicotine use.  
SS: Appears to have a large discrepancy in medians, with higher scores indicating more nicotine use.  
Impulsive: A moderate discrepancy in medians, with higher scores indicating more nicotine use.  
Ethnicity: No discernable patterns, Black and Asian participants reported the highest counts.

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   
)

##   
## Attaching package: 'rlang'

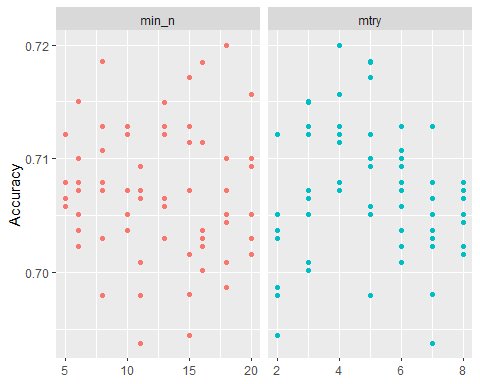
## The following objects are masked from 'package:purrr':  
##   
## %@%, as\_function, flatten, flatten\_chr, flatten\_dbl, flatten\_int,  
## flatten\_lgl, flatten\_raw, invoke, list\_along, modify, prepend,  
## splice

##   
## Attaching package: 'vctrs'

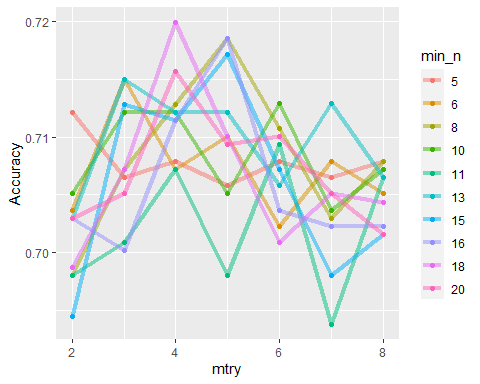
## The following object is masked from 'package:dplyr':  
##   
## data\_frame

## 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")

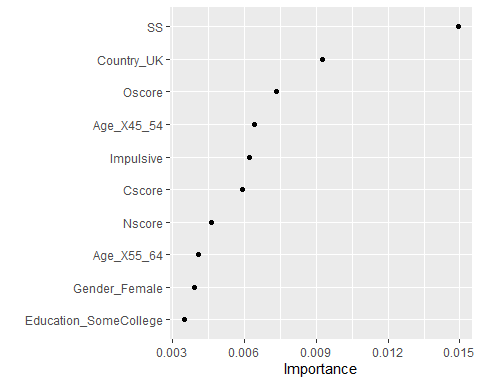


best\_rf = select\_best(rf\_res\_tuned, "accuracy")  
  
final\_rf = finalize\_workflow(  
 drug\_wflow,  
 best\_rf  
)  
  
final\_rf

## == 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 = 18  
##   
## Engine-Specific Arguments:  
## importance = permutation  
##   
## Computational engine: ranger

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

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



## Task 5 Response:

The most important variable is SS (sensation seeing), followed by country (UK), and Oscore (the openness to experience).

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

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

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

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction Yes No  
## Yes 929 142  
## No 19 324  
##   
## Accuracy : 0.8861   
## 95% CI : (0.8684, 0.9022)  
## No Information Rate : 0.6704   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.7238   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.9800   
## Specificity : 0.6953   
## Pos Pred Value : 0.8674   
## Neg Pred Value : 0.9446   
## Prevalence : 0.6704   
## Detection Rate : 0.6570   
## Detection Prevalence : 0.7574   
## Balanced Accuracy : 0.8376   
##   
## 'Positive' Class : Yes   
##

Predictions on test

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

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

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

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction Yes No  
## Yes 284 94  
## No 32 61  
##   
## Accuracy : 0.7325   
## 95% CI : (0.6901, 0.772)  
## No Information Rate : 0.6709   
## P-Value [Acc > NIR] : 0.002268   
##   
## Kappa : 0.3254   
##   
## Mcnemar's Test P-Value : 5.501e-08   
##   
## Sensitivity : 0.8987   
## Specificity : 0.3935   
## Pos Pred Value : 0.7513   
## Neg Pred Value : 0.6559   
## Prevalence : 0.6709   
## Detection Rate : 0.6030   
## Detection Prevalence : 0.8025   
## Balanced Accuracy : 0.6461   
##   
## 'Positive' Class : Yes   
##

## Task 6 Response:

On the training data, this model has an accuracy of .8861, which is better than the naive accuracy of .6704. Accuracy takes a significant hit on the testing data, dropping to .7325, which means the model may not fit data that it has not seen before as well.

## Task 7 Response:

This model may be used to determine if personality tests or life factors could indicate the need for early intervention to prevent nicotine (or with modification, illicit drug) use in programs such as D.A.R.E. or other school / education systems. I would not recommend this particular model for real-world use because the accuracy is not consistent when used against data it has not seen before and it may lead to the failure to properly identify people / students, wasting resources.