# Module 4: Assignment 2

## Maliszewski, Angela

## Random Forests

Deliverable: All of your work for this assignment should be done in an R Markdown document. Knit your document into a Word file and submit the Word file as the deliverable for this assignment.

Libraries: For this assignment you will need the following libraries: tidyverse, tidymodels, caret, gridExtra, vip, and ranger.

Read in the “drug\_data.csv” dataset. This dataset deals with drug consumption in individuals across a wide spectrum of countries, drugs, ages, etc.

A description of the dataset is available here: <http://archive.ics.uci.edu/ml/datasets/Drug+consumption+%28quantified%29>

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

## Parsed with 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()  
## )

## See spec(...) for full column specifications.

This dataset (as in common with medical and healthcare datasets) requires quite a bit of cleaning and preparation before analysis. I’ll walk you through the cleaning before we start our random forest work.

Loading the Data: The columns do not have names, so we’ll supply names via the names function.

Comment out the str command before knitting your completed work (to save space) :)

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)

Next-up we change all CL0 and CL1 values to “No” and CL2, CL3, CL4, CL5, and CL6 values to “Yes”. CL0 and CL1 imply the drug was never used or used over a decade ago. CL2 through CL6 imply more recent drug use.

The code below finds any CL0, CL1, etc. values in the data frame and replaces them with the appropriate “No” or “Yes”.

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"

Next up we do a good bit of factor conversion and recoding. Note the use of mutate\_at to target specific ranges of variables.

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.

Take a peek at the cleaned data (comment this line out before knitting, to save space).

#str(drug\_clean)

We’ll focus on Nicotine use, so let’s get rid of the remaining drug use variables. We’ll use select for this.

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"

Now we’re in business.

# Task 1:

Check for missing data in our “drug\_clean” dataframe. Is there any missingness?

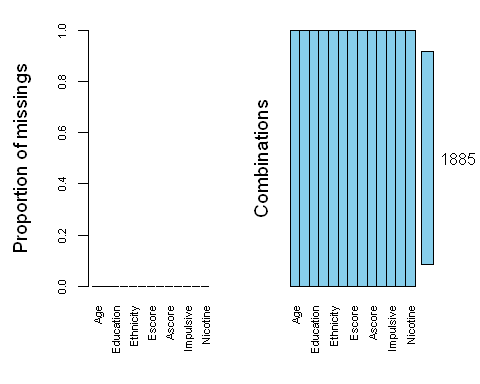
ANSWER: There is no missing data.

If so, identify and implement a reasonable strategy to deal with the missingness.

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   
##

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



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 2:

Split the dataset into training (70%) and testing (30%) sets. Use a set.seed of 1234. Stratify by the “Nicotine” variable.

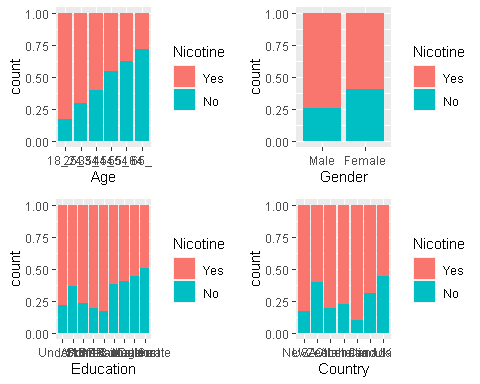
set.seed(1234)   
drug\_clean\_split = initial\_split(drug\_clean, prop = 0.7, strata = Nicotine) #70% in training  
train = training(drug\_clean\_split)  
test = testing(drug\_clean\_split)

# Task 3:

Create appropriate visualizations (12 in all) to examine the relationships between each variable and “Nicotine”. Use grid.arrange (from the gridExtra package) to organize these visuals (perhaps in groups of four visualizations?).

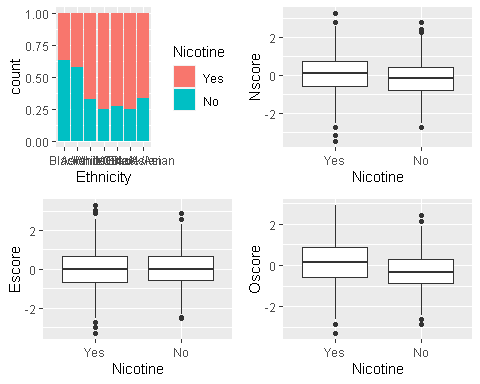
Comment on the relationship between each variable and “Nicotine”. Age - age seems to have an impact on nicotine use Gender - gender seems to have an impact on nicotine use Education - education seems to have an impact on nicotine use Country - country seems to have an impact on nicotine use

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



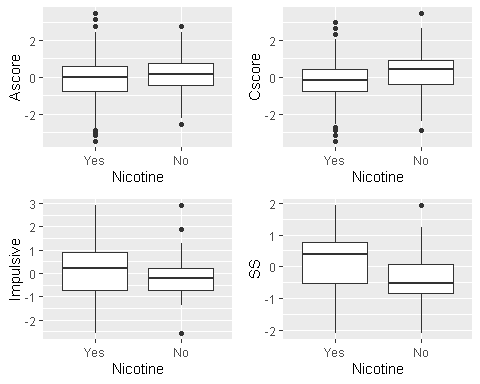
Ethnicity - ethnicity seems to have an impact on nicotine use Nscore - Nscore seems to have a slight impact on nicotine use Escore - Escore DOES NOT seem to have an impact on nicotine use Oscore - Oscore seems to have a slight impact on nicotine use

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



Ascore - Ascore DOES NOT seem to have an impact on nicotine use Cscore - Oscore seems to have a slight impact on nicotine use Impulsive - Impulsive seems to have a slight impact on nicotine use SS - SS seems to have an impact on nicotine use

p9 = ggplot(drug\_clean, aes(x = Nicotine, y = Ascore)) + geom\_boxplot()  
p10 = ggplot(drug\_clean, aes(x = Nicotine, y = Cscore)) + geom\_boxplot()  
p11 = ggplot(drug\_clean, aes(x = Nicotine, y = Impulsive)) + geom\_boxplot()  
p12 = ggplot(drug\_clean, aes(x = Nicotine, y = SS)) + geom\_boxplot()  
grid.arrange(p9,p10,p11,p12)



# Task 4:

Create a random forest model on the training set to predict Nicotine using all of the variables in the dataset. You 5-fold, k-fold cross-validation (random number seed of 123 for the folds). Allow R to select mtry values between 2 and 8 and min\_n values between 5 and 20. Use 10 levels in your “grid\_regular” function.

Set a random number seed of 123 for the tune\_grid function. Use 100 trees.

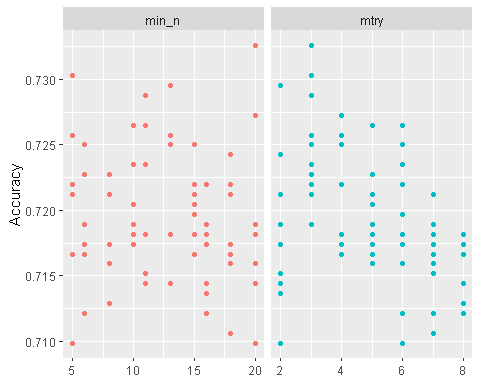
Visualize the relationships between parameters and performance metrics.

ANSWER: It looks like the best accuracy is with a mtry of 3 and min\_n of 20.

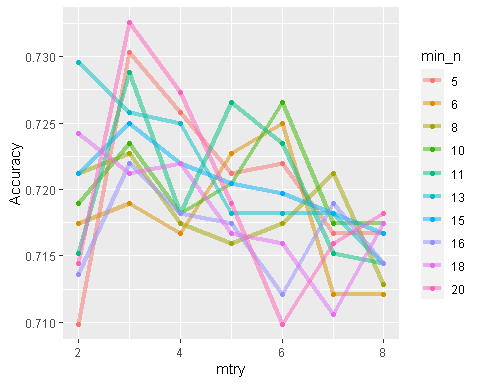
set.seed(123)  
rf\_folds = vfold\_cv(train, v = 5)

drug\_clean\_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\_clean\_wflow =   
 workflow() %>%   
 add\_model(rf\_model) %>%   
 add\_recipe(drug\_clean\_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\_clean\_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:

Use the best mtry and min\_n values from Task 4 to finalize the workflow and fit the model to training set. Examine variable importance.

What variables are most important in this model?

ANSWER: SS, Country UK, Oscore, Age 45-54 and Cscore are the top five most importance variables impacting nicotine use. SS is sensation seeking measured by ImpSS. Country is country of current residence of participant. Oscore is Openness to experience. Age is age of participant. Cscore is Conscientiousness.

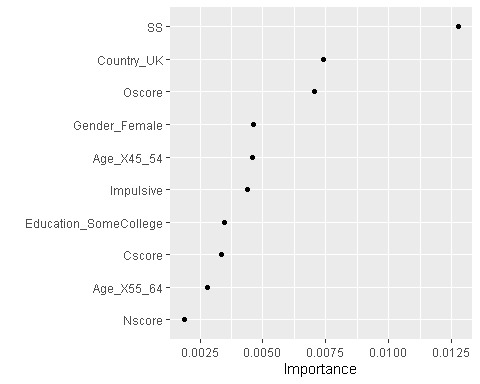
(Hint: Refer back to the dataset’s webpage if you need clarification as to meaning of any variables).

best\_rf = select\_best(rf\_res\_tuned, "accuracy")  
  
final\_rf = finalize\_workflow(  
 drug\_clean\_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 = 3  
## trees = 100  
## min\_n = 20  
##   
## Engine-Specific Arguments:  
## importance = permutation  
##   
## Computational engine: ranger

#fit the finalized workflow to our training data  
final\_rf\_fit = fit(final\_rf, train)

#to see variable importance  
final\_rf\_fit %>% pull\_workflow\_fit() %>% vip(geom = "point")



# Task 6:

How does the model perform on the training and testing sets?

ANSWER: The model provides 84.32% accuracy on the training set and only 71.15% accuracy on the testing set. Both sets are higher than the naive accuracy rate of 67.05%.

Predictions and Confusion matrix on the training set.

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 850 171  
## No 35 264  
##   
## Accuracy : 0.8439   
## 95% CI : (0.8232, 0.8631)  
## No Information Rate : 0.6705   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.6163   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.9605   
## Specificity : 0.6069   
## Pos Pred Value : 0.8325   
## Neg Pred Value : 0.8829   
## Prevalence : 0.6705   
## Detection Rate : 0.6439   
## Detection Prevalence : 0.7735   
## Balanced Accuracy : 0.7837   
##   
## 'Positive' Class : Yes   
##

Predictions and Confusion matrix on the test set.

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

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

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

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction Yes No  
## Yes 337 121  
## No 42 65  
##   
## Accuracy : 0.7115   
## 95% CI : (0.6722, 0.7486)  
## No Information Rate : 0.6708   
## P-Value [Acc > NIR] : 0.02111   
##   
## Kappa : 0.2676   
##   
## Mcnemar's Test P-Value : 9.999e-10   
##   
## Sensitivity : 0.8892   
## Specificity : 0.3495   
## Pos Pred Value : 0.7358   
## Neg Pred Value : 0.6075   
## Prevalence : 0.6708   
## Detection Rate : 0.5965   
## Detection Prevalence : 0.8106   
## Balanced Accuracy : 0.6193   
##   
## 'Positive' Class : Yes   
##

# Task 7:

Comment on how this model might be used in the “real-world.” Would you recommend this model for real-world use? What if any concerns would you have about using the model?

ANSWER: This model could be used to either a) educate the public on what human attributes are likely to lead to risk of being a consumer of specific drug(s) b) market specific drugs to segments of the population that are most likely to consume them based.

Specifically looking at nicotine use: With a 71% accuracy on the testing data. I would say that using this model to market to specific ages/countries would have a positive influence on revenue. I would not use this model to classify people into likely nicotine users given the 20-30% of population that could be classified incorrectly. The most influential variable is sensation seeking and there are so many ways people achieve this other than nicotine use, like eating.