Wine Sales Project

MSDS 410 | Prepared by Vincent Pun

SECTION ONE: DATA EXPLORATION

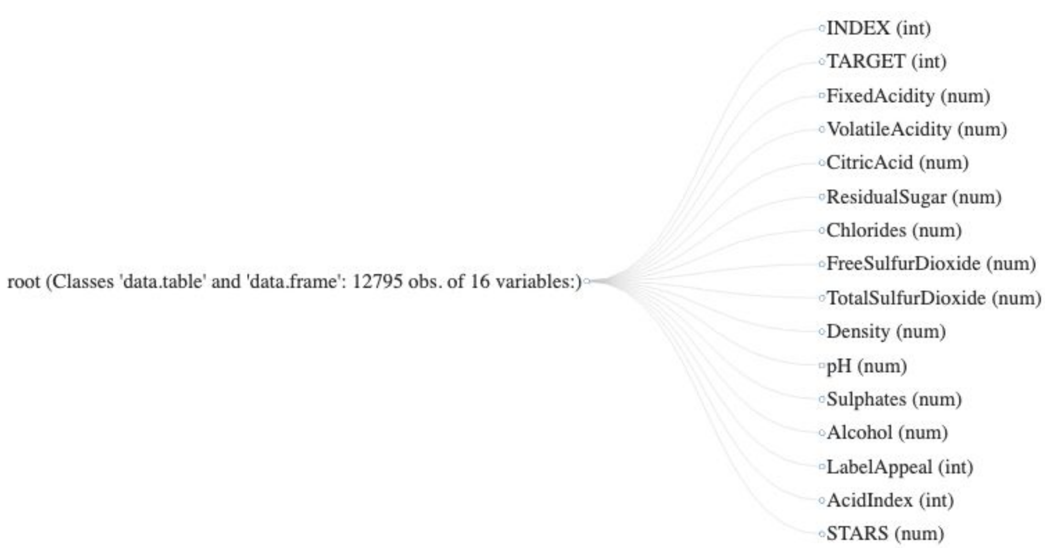
Facts:

This data set contains information on approximately 12,000 commercially available wines. The variables are mostly related to the chemical properties of the wine being sold. The target variable is the number of sample cases of wine that were purchased by wine distribution companies after sampling a wine. These cases would be used to provide tasting samples to restaurants and wine stores around the United States. The more sample cases purchased, the more likely a wine is to be sold at a high end restaurant.

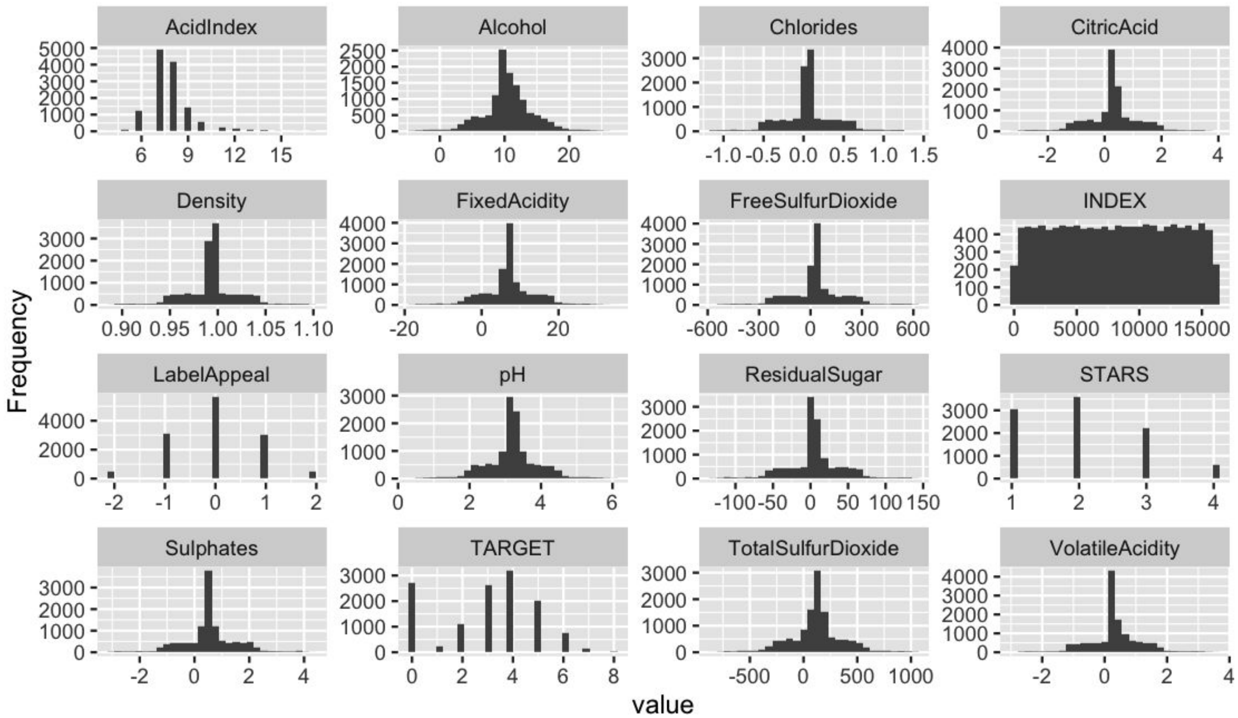
A large wine manufacturer is studying the data in order to predict the number of wine cases ordered based upon the wine characteristics. If the wine manufacturer can predict the number of cases, then that manufacturer will be able to adjust their wine offering to maximize sales.

EDA:

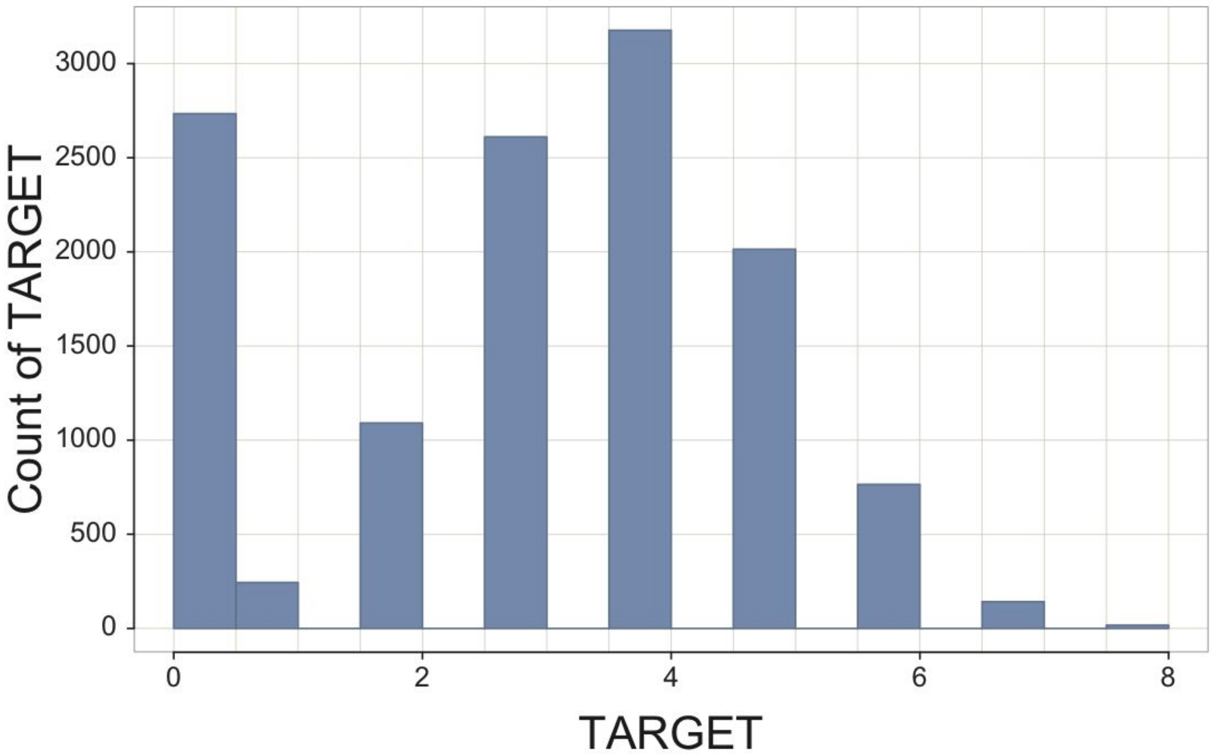
We see that there are a total of sixteen variables in the training dataset. Our response variable is TARGET (int), and our goal is to identify the variables that will best predict whether purchases of wine are made.



Looking at the histogram below, we notice that the TARGET distribution contains a large amount of zeros. Additionally, we see that there is a large number of records with the STARS value equal to 2. Overall, it seems that distributions appear either normally distributed or right skewed.

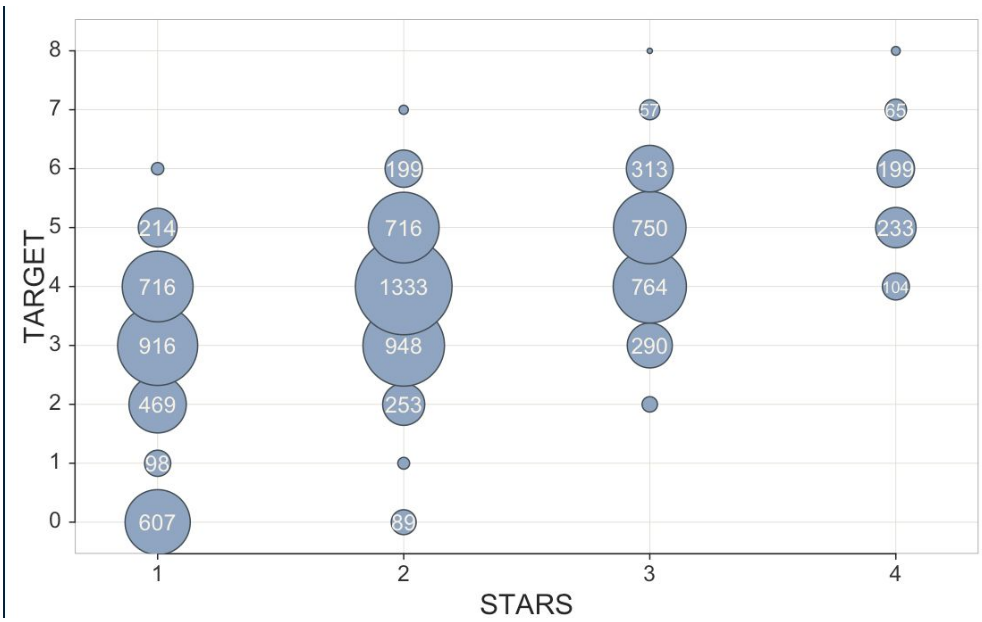


Enlarged histogram of TARGET:

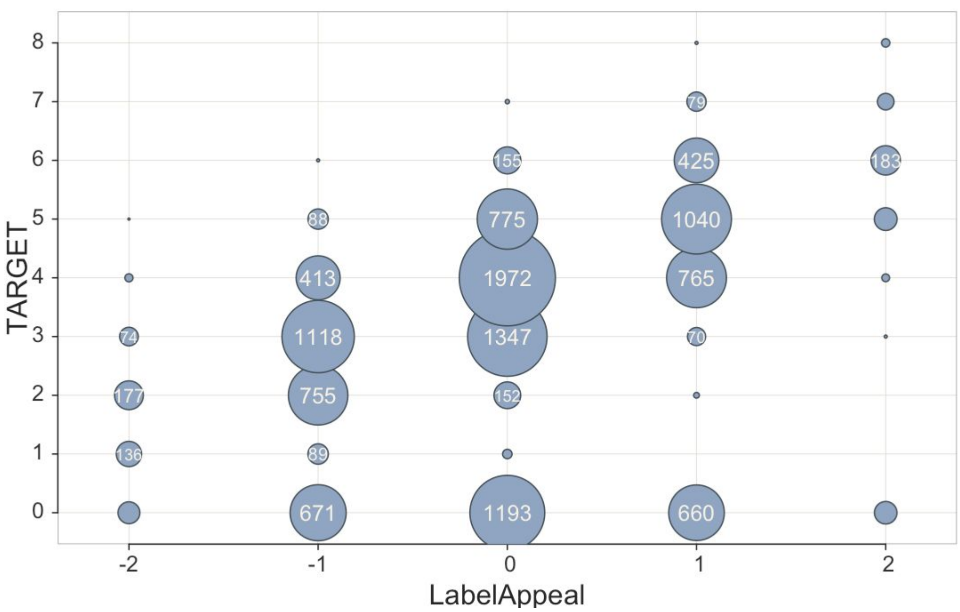


Count of TARGET
3000
2500
2000
1500
1000
500
0
0
2
4
6
8
TARGET

When applying a scatter plot to the TARGET and STARS variables, it is observed that the values of TARGET appear to be positively correlated with STARS to some degree. For example, there are no records indicating that a wine has four stars and is not purchased.



However, when looking at LabelAppeal, which is the marketing score indicating the appeal of label design for consumers, it does not seem that it is a strong predictor for TARGET. For example, a higher label appeal score of 1 contains 660 instances where TARGET equals zero, and a label appeal score of -1 contains 671 instances where TARGET equals zero.

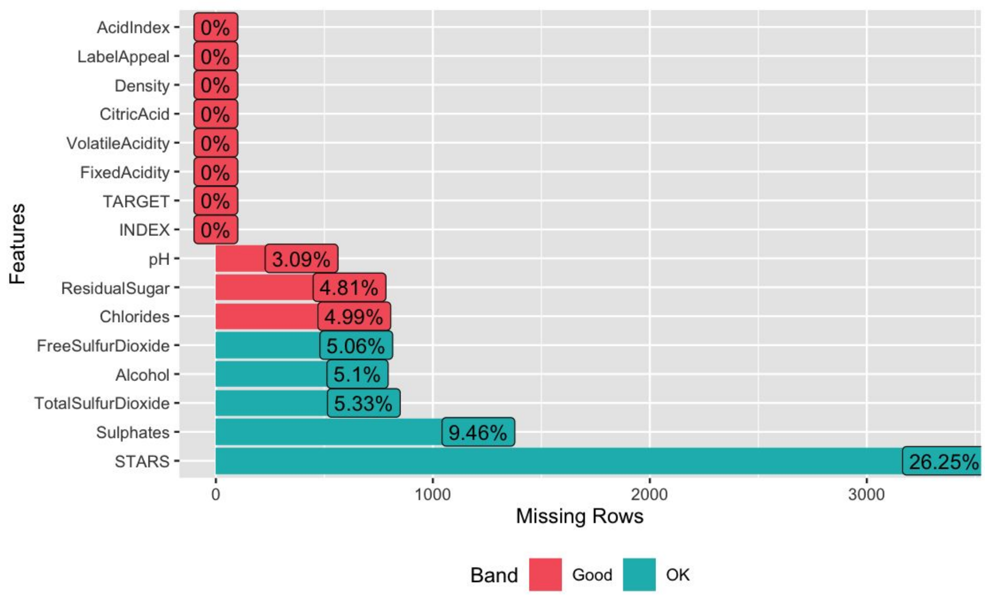


SECTION TWO: DATA PREPARATION

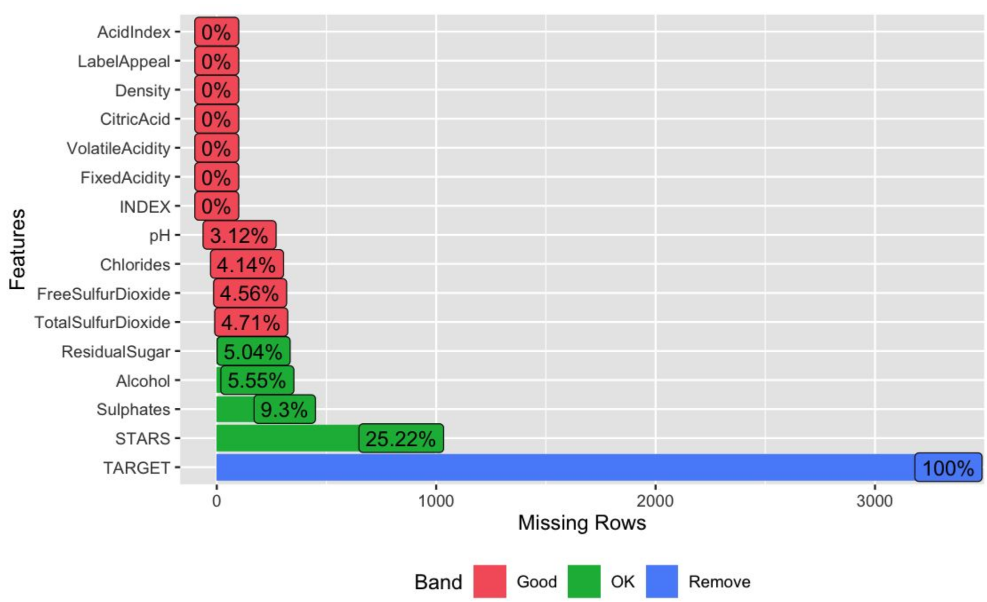
Missing Data:

While most of the features in the training dataset do not contain very many missing values (similar findings for test dataset), it appears that STARS contains a large amount of missing values, with approximately one-fourth of all values missing in that attribute.

Train Dataset



Test Dataset



Imputation is conducted on the features with missing values. For simplicity, all attributes are imputed with their column's median values except for STARS.

```
#TRAIN
train$Sulphates[is.na(train$Sulphates)] <- median(train$Sulphates, na.rm = T)
train$TotalSulfurDioxide[is.na(train$TotalSulfurDioxide)] <- median(train$TotalSulfurDioxide, na.rm = T)
train$Alcohol[is.na(train$Alcohol)] <- median(train$Alcohol, na.rm = T)
train$FreeSulfurDioxide[is.na(train$FreeSulfurDioxide)] <- median(train$FreeSulfurDioxide, na.rm = T)
train$Chlorides[is.na(train$Chlorides)] <- median(train$Chlorides, na.rm = T)
train$ResidualSugar[is.na(train$ResidualSugar)] <- median(train$ResidualSugar, na.rm = T)
train$pH[is.na(train$pH)] <- median(train$pH, na.rm = T)

#TEST
test$Sulphates[is.na(test$Sulphates)] <- median(test$Sulphates, na.rm = T)
test$TotalSulfurDioxide[is.na(test$TotalSulfurDioxide)] <- median(test$TotalSulfurDioxide, na.rm = T)
test$Alcohol[is.na(test$Alcohol)] <- median(test$Alcohol, na.rm = T)
test$FreeSulfurDioxide[is.na(test$FreeSulfurDioxide)] <- median(test$FreeSulfurDioxide, na.rm = T)
test$Chlorides[is.na(test$Chlorides)] <- median(test$Chlorides, na.rm = T)
test$ResidualSugar[is.na(test$ResidualSugar)] <- median(test$ResidualSugar, na.rm = T)
test$pH[is.na(test$pH)] <- median(test$pH, na.rm = T)
```

Feature Engineering:

Since we are only trying to predict whether a sale was made, we have created a new variable called TARGET_BINARY, which equals zero if no purchases were made (TARGET = 0) and one if any purchases were made (TARGET > 0).

Additionally, a new variable, STARSNA, is created. This variable equals 0 if there is a missing value in STARS and 1 otherwise. This is observed in a correlation plot (below) to see whether there is a strong correlation with either TARGET or TARGET_BINARY, which would indicate that there is a reason aside from error for the missing STARS values.

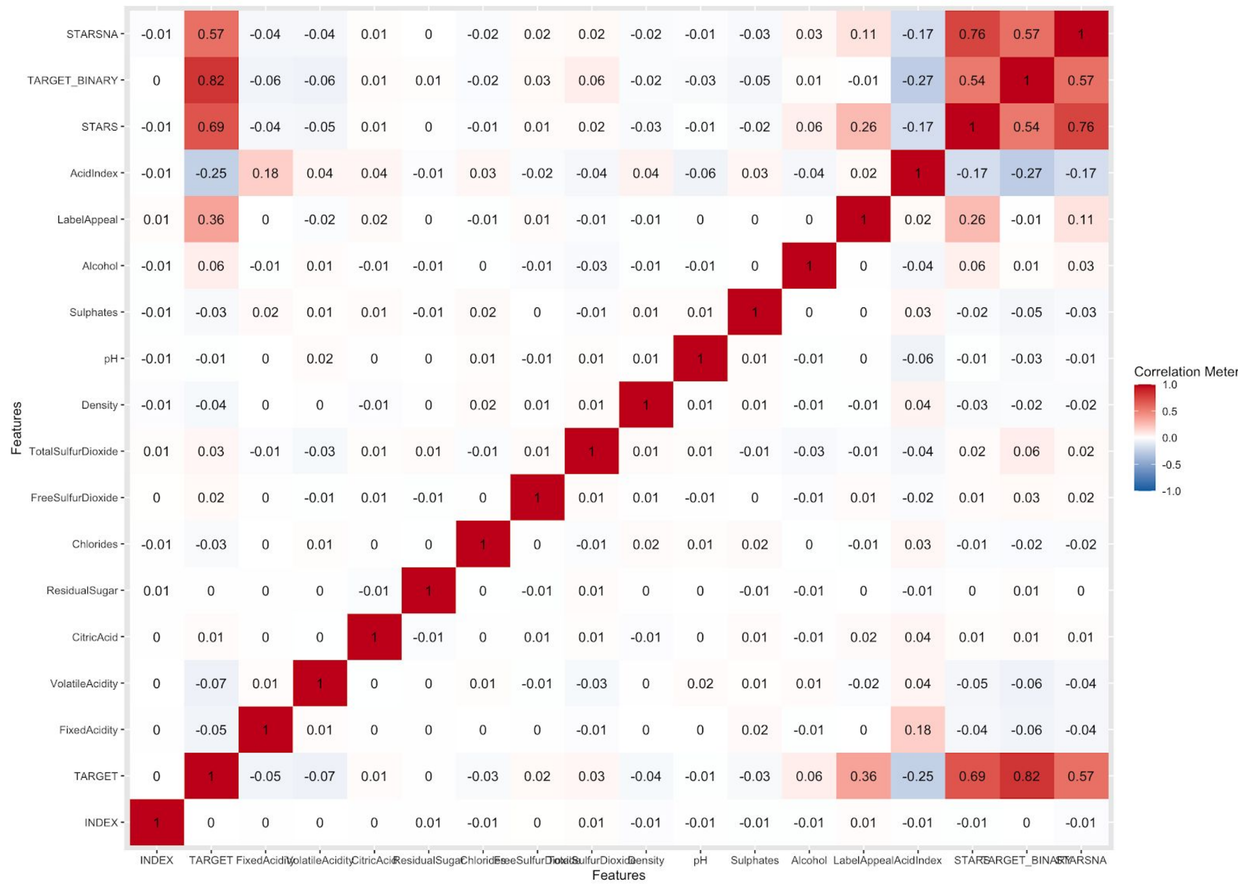
```
train$TARGET_BINARY <- ifelse(train$TARGET > 0, 1, 0)

train$STARSNA <- ifelse(is.na(train$STARS), 0, 1)
test$STARSNA <- ifelse(is.na(test$STARS), 0, 1)

train$STARS <- ifelse(is.na(train$STARS), 0, train$STARS)
test$STARS <- ifelse(is.na(test$STARS), 0, train$STARS)
```

Train Test Split (70/30):
The train dataset is split (line 288) 70/30 for train and validation purposes. The two datasets used to evaluate the models in the report are titled **train.split** and **test.split**. After a best model is determined, the best model will be retrained on 100% of the data (as **model_4**) for Kaggle submission purposes.

Correlation Plot (Train Data):



Top 5 variables most correlated to **TARGET** (from greatest to least, excluding TARGET):

TARGET_BINARY, STARS, STARSNA, LabelApeal, AcidIndex

Top 5 variables most correlated to **TARGET_BINARY** (from greatest to least, excluding TARGET_BINARY):

TARGET, STARSNA, STARS, AcidIndex, FixedAcidity

SECTION THREE: BUILD MODELS

Build at least ***three*** *different models. Try a linear regression model and two logistic regression models.*

You may select the variables manually or use some other method. Describe the techniques you used. If you selected a variable for inclusion or exclusion indicate why.

Show all of your models and the statistical significance of the input variables.

Discuss the coefficients in the model, do they make sense?

In this case, about the only thing you can comment on is the number of stars and the wine label appeal.

However, you might comment on the coefficient and magnitude of variables and how they are similar or different from model to model. For example, you might say "pH seems to have a major positive impact in my regression model, but a negative effect elsewhere".

Model 1 - Linear Regression

The first model used is a multiple linear regression model that contains the top six most correlated variables to the response variable TARGET.

The model's Root Mean Square Error in addition to accuracy score is compared to a baseline score if every record were predicted as a sale. RMSE is slightly better (1.3 versus 1.9) compared to the baseline analysis, and accuracy score is almost identical (78.8% versus 78.3%), so it does not appear that Model 1 is adding much predictive value in this analysis.

```
##
## Call:
## lm(formula = TARGET ~ STARS + LabelAppeal + AcidIndex + Alcohol +
##     VolatileAcidity + FixedAcidity, data = train.split)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5841 -0.9671  0.0693  0.9106  5.6109
##
## Coefficients:
##                   Estimate Std. Error t value             Pr(>|t|)
## (Intercept)      3.1142310  0.1009280  30.856 < 0.0000000000000002
## STARS            0.9816117  0.0124747  78.688 < 0.0000000000000002
## LabelAppeal      0.4297640  0.0162670  26.419 < 0.0000000000000002
## AcidIndex       -0.2104211  0.0109254 -19.260 < 0.0000000000000002
## Alcohol          0.0132461  0.0039733   3.334             0.000860
## VolatileAcidity -0.0940058  0.0249615  -3.766             0.000167
## FixedAcidity    -0.0004926  0.0028342  -0.174             0.862035
##
## Residual standard error: 1.327 on 8983 degrees of freedom
## Multiple R-squared:  0.5243, Adjusted R-squared:  0.524
## F-statistic:  1650 on 6 and 8983 DF,  p-value: < 0.00000000000000022
```

```
sort(vif(model_1),decreasing=TRUE)
```

```
##           STARS     LabelAppeal       AcidIndex    FixedAcidity         Alcohol
##        1.117619        1.081669        1.067214        1.030545        1.003917
## VolatileAcidity
##        1.002117
```

RMSE Model 1 = 1.329
RMSE Baseline = 1.933

```
##        Predict
## Actual    0    1
##       0    6  803
##       1    3 2993
```

```
paste('Model 1 Accuracy = ',((2993+6)/(2993+6+803+3)))
```

```
## [1] "Model 1 Accuracy =  0.788173455978975"
```

Model 2 - Logistic Regression

Multiple logistic regression is used for Model 2. STARSNA and LabelAppeal are used to predict TARGET_BINARY. The reason STARSNA is used over STARS is because it appears to have a slightly stronger positive correlation to TARGET_BINARY.

Based on the coefficients, it appears that wines without missing STARS values have a much higher chance of being sold. However, the coefficient for LabelAppeal is negative, which means that a positive label score may decrease the chances of a wine being sold, which may be counterintuitive. The reason that LabelAppeal remains as an independent variable is that it decreases the AIC for this model. Overfitting will be monitored for

To evaluate this model, we observe that AIC equals 6605.3. Additionally, a confusion matrix is used to determine that the accuracy of this predicted model (on test.split) is 84.86%, which is a considerable improvement from the baseline accuracy of 78.3%.

```
Call:
glm(formula = TARGET_BINARY ~ STARSNA + LabelAppeal, family = binomial,
    data = train.split)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5127   0.3364   0.3834   0.4365   1.6275

Coefficients:
            Estimate Std. Error z value            Pr(>|z|)
(Intercept) -0.47503    0.04283 -11.092 < 0.0000000000000002
STARSNA      3.04835    0.06516  46.783 < 0.0000000000000002
LabelAppeal -0.27006    0.03541  -7.626   0.0000000000000242

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9338.3  on 8989  degrees of freedom
Residual deviance: 6599.3  on 8987  degrees of freedom
AIC: 6605.3

Number of Fisher Scoring iterations: 5
```

```
        Predict
Purchase    0    1
       0  571  238
       1  338 2658
[1] "model 2 accuracy =  0.84862023653088"
```

Model 3 - Multiple Logistic Regression

Model 3 is a bit more complex compared to Model 2, for a greater number of independent variables are used to predict TARGET_BINARY. Based on the coefficients, STARS is again a positive predictor in whether a wine is sold, and the increase in LabelAppeal, AcidIndex, and VolatileAcidity may decrease TARGET_BINARY.

When evaluating this model on the test data (test.split), it appears that the accuracy score is slightly higher than that of Model 2. Additionally, AIC shows a slight improvement over Model 2 as well. This suggests that adding more variables is not overfitting the model.

```
Call:
glm(formula = TARGET_BINARY ~ STARS + LabelAppeal + AcidIndex +
    VolatileAcidity, family = binomial, data = train.split)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-3.09173   0.03989   0.19156   0.44943   2.24432

Coefficients:
                Estimate Std. Error z value             Pr(>|z|)
(Intercept)      2.67180    0.20367  13.118 < 0.0000000000000002
STARS            2.04029    0.05087  40.110 < 0.0000000000000002
LabelAppeal     -0.45744    0.03932 -11.635 < 0.0000000000000002
AcidIndex       -0.39201    0.02484 -15.781 < 0.0000000000000002
VolatileAcidity -0.16078    0.05936  -2.709              0.00676

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9338.3  on 8989  degrees of freedom
Residual deviance: 5472.5  on 8985  degrees of freedom
AIC: 5482.5

Number of Fisher Scoring iterations: 6
```

```
        Predict
Purchase    0    1
       0  521  288
       1  255 2741
[1] "model 3 accuracy =  0.857293035479632"
```

SECTION FOUR: SELECT MODELS

Decide on the criteria for selecting the “Best Model”.
Will you use a metric such as AIC or Average Squared Error? Will you select a model with slightly worse performance if it makes more sense or is more parsimonious? Discuss why you selected your model.

If you happen to like the standard regression model the best, then that is OK. Please say that you like it the best and why you like it. HOWEVER, you MUST select a model for grading.

```
#78.63% of the records has a purchase
#this is the number to beat
table(train$TARGET_BINARY, dnn = c("Purchase"))
```

```
## Purchase
##     0     1
##  2734 10061
```

Baseline Accuracy = 78.63%

Model	Validation Accuracy	AIC	Kaggle (RMSE)
model_1	78.8%	--	--
model_2	84.86%	6605.03	2.83753
model_3	85.73%	5482.5	3.10252

It seems that model_3 should have performed better on the Kaggle dataset based on Validation Accuracy and AIC (compared to model_2), but it appears that there is a significant difference in RMSE. This suggests that model_3 could have been overfitted since it uses multiple independent variables to predict TARGET_BINARY.

Overall Model 2 seems to be a good fit, as using only STARSNA and LabelAppeal is more parsimonious. Additionally, it takes into account that some records will simply have missing/null values for STARs, which is acceptable for this analysis.

SECTION FIVE: FORMULA FOR MODEL

Write an equation for your model that will allow someone else to implement it. They should be able to score new data and predict the number of wine cases that will be sold based upon the qualities of the wine.

The variable should be named:
P_TARGET

The model equation will need to include:
a. All the variable transformations such as fixing missing values
b. The model formulas

For the purpose of predicting whether a sale occurs, we will want to determine whether P_TARGET equals zero or one.

```
Call:
glm(formula = TARGET_BINARY ~ STARSNA + LabelAppeal, family = binomial,
    data = train)
```

```
Coefficients:
             Estimate Std. Error z value             Pr(>|z|)
(Intercept) -0.48449    0.03613 -13.408 <0.0000000000000002
STARSNA      3.05083    0.05468  55.791 <0.0000000000000002
LabelAppeal -0.26784    0.02981  -8.986 <0.0000000000000002
```

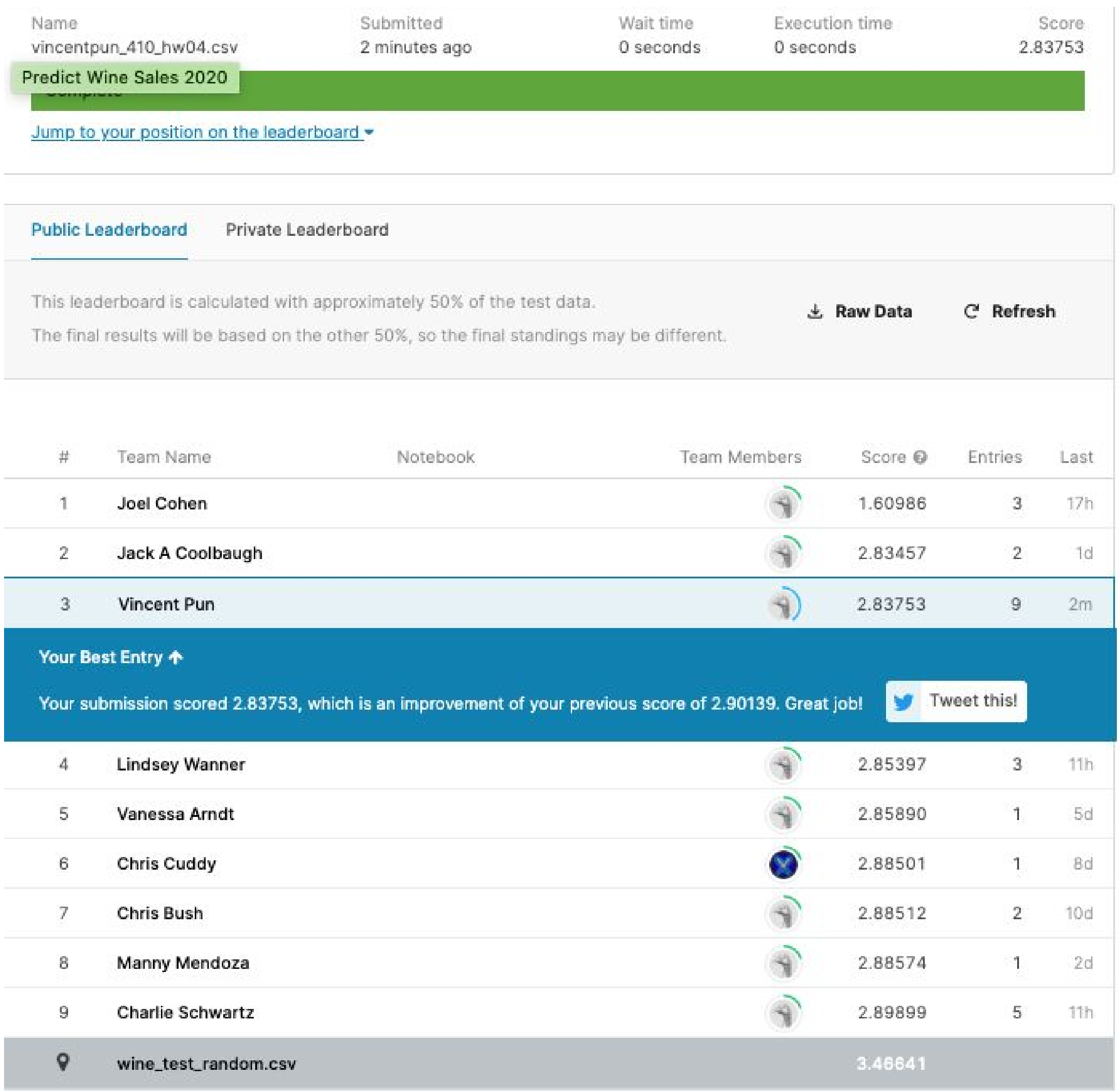
train$STARSNA <- ifelse(is.na(train$STARS), 0, 1)
test$STARSNA <- ifelse(is.na(test$STARS), 0, 1)

Formula:

P_TARGET = ROUND(3.06083*STARSNA -0.26784*LabelAppeal,0)

SECTION SIX: SCORED DATA FILE

Score the data file wine_test.csv. Create a file that has only TWO variables for each Record: **yourname_410_hw04.csv**
The second value, P_TARGET is 0 for no sales or 1 meaning some number of cases sold.
INDEX
P_TARGET



Name	Submitted	Wait time	Execution time	Score
vincentpun_410_hw04.csv	2 minutes ago	0 seconds	0 seconds	2.83753

Predict Wine Sales 2020

Jump to your position on the leaderboard

Public Leaderboard Private Leaderboard

This leaderboard is calculated with approximately 50% of the test data.
The final results will be based on the other 50%, so the final standings may be different.

Raw Data Refresh

#	Team Name	Notebook	Team Members	Score	Entries	Last
1	Joel Cohen			1.60986	3	17h
2	Jack A Coolbaugh			2.83457	2	1d
3	Vincent Pun			2.83753	9	2m
4	Lindsey Wanner			2.85397	3	11h
5	Vanessa Arndt			2.85890	1	5d
6	Chris Cuddy			2.88501	1	8d
7	Chris Bush			2.88512	2	10d
8	Manny Mendoza			2.88574	1	2d
9	Charlie Schwartz			2.89899	5	11h
	wine_test_random.csv			3.46641		

Your Best Entry

Your submission scored 2.83753, which is an improvement of your previous score of 2.90139. Great job!

Tweet this!