Draft-1

STA 210 - Summer 2022

Team 2 - Alicia Gong, Ashley Chen, Abdel Shehata, Claire Tam

6-8-2022

Setup

Load packages and data:

```r
library(tidyverse)
library(tidymodels)
library(dplyr)
library(ggplot2)
library(cowplot)
library(knitr)
library(recipes)
library(caret)
library(InformationValue)
library(ISLR)
library(MASS)
library(nnet)
```

```r
redwine <- read.csv("data/winequality-red.csv", sep = ";")
whitewine <- read.csv("data/winequality-white.csv", sep = ";")
redwine<-redwine%>%mutate(color="red")
whitewine<-whitewine%>%mutate(color="white")
wine<-redwine%>%full_join(whitewine)
```

```
Joining, by = c("fixed.acidity", "volatile.acidity", "citric.acid",
"residual.sugar", "chlorides", "free.sulfur.dioxide", "total.sulfur.dioxide",
"density", "pH", "sulphates", "alcohol", "quality", "color")
```

```
wine<- slice(wine, sample(1:n()))
```

Introduction and Data

Introduction

About 234 million hectoliters of wine were consumed in 2020, worldwide, with the US making up approximetly 14% of that consumption (Karlsson 2020). Since Wine composition and wine quality varies widely, it raises the question: what makes a good wine?

To answer that question, we will analyze the wine quality dataset from Vinho Verde vinyard in Portogal, and more importantly try to narrow down our question to make it possible for it to be supported by evidence. Below is the introduction to our research:

Project Goal: To identify variables that are important in explaining variation in the response. "Vinho Verde" is the kind of wine exclusively produced in the demarcated region of Vinho Verde in northwestern Portugal. The Vinho Verde wine has its own unique system of production and is only produced from the indigenous grape varieties of the region. Vinho Verde region is one of the largest and oldest wine regions in the world, and is home to thousands of producers, generating a wealth of economic activity and jobs, and strongly contributes to the development of Minho province and the country. The Vinho Verde wine also enjoys high reputation worldwide. It is recurrently awarded in national and international competitions. The goal of this dataset is to model wine quality based on physicochemical tests. We believe that this dataset can also be used to analyze the relationship between different chemical compositions and the ratings of wine quality. We believe that this dataset can also be used to analyze what chemical factors are attributable to the final rating of Vinho Verde wine. Our research may shed light on future research and development directions for improving the quality of Vinho Verde wine, which may also contribute to the competitiveness of Portuguese wine industry.

Our goal is to produce a classification model that best explains how different chemical compositions of the Portuguese "Vinho Verde" wine affects the variation of the wine quality.

Data Introduction

The Wine Quality dataset was collected from Vinho Verde wine Samples, from the North of Portugal. The data was originally donated in 2009 by Professor Cortez. The specific mechanism of the collection of the data was lab work done on different wines to measure their chemical attributites (like acidity etc.). The quality of the wine however was obtained through the average rating of three wine experts. The dataset is divided into two: Red wine and White wine. Red Wine has 1599 observations, and white wine has 4898 observations (each observation being a specific wine). Information about the wine include but are not limited to:PH,Density,Acidity, and alcohol content.

Each observation is a specific wine from the Vinho Verde region. Thus, there might be a little uncertainity in collecting the exact numbers for each numbers. However, since the Vinho Verde region is a vast region spreading 15500 hectareas of vineyards in far-north Portugal, this uncertainity shouldn't be significant in our analysis or project. Thus, we will assume that the datas are independent and random ::: {.cell}

```
glimpse(wine)
```

```
Rows: 6,497
Columns: 13
$ fixed.acidity        <dbl> 6.1, 8.8, 5.6, 6.3, 12.7, 6.6, 6.2, 9.4, 6.2, 10.~
$ volatile.acidity     <dbl> 0.380, 0.470, 0.235, 0.240, 0.600, 0.270, 0.320, ~
$ citric.acid          <dbl> 0.15, 0.49, 0.29, 0.22, 0.49, 0.32, 0.16, 0.32, 0~
$ residual.sugar       <dbl> 1.80, 2.90, 1.20, 11.90, 2.80, 1.30, 7.00, 6.50, ~
$ chlorides            <dbl> 0.072, 0.085, 0.047, 0.050, 0.075, 0.044, 0.045, ~
$ free.sulfur.dioxide  <dbl> 6.0, 17.0, 33.0, 65.0, 5.0, 18.0, 30.0, 20.0, 22.~
$ total.sulfur.dioxide <dbl> 19, 110, 127, 179, 19, 93, 136, 167, 143, 29, 282~
$ density              <dbl> 0.99550, 0.99820, 0.99100, 0.99659, 0.99940, 0.99~
$ pH                   <dbl> 3.42, 3.29, 3.34, 3.06, 3.14, 3.11, 3.18, 3.08, 3~
$ sulphates            <dbl> 0.57, 0.60, 0.50, 0.58, 0.57, 0.56, 0.47, 0.43, 0~
$ alcohol              <dbl> 9.40, 9.80, 11.00, 9.30, 11.40, 12.25, 9.60, 10.6~
$ quality              <int> 5, 5, 7, 6, 5, 5, 6, 5, 5, 6, 5, 6, 6, 5, 6, 5, 7~
$ color                <chr> "red", "red", "white", "white", "red", "white", "~
```

::: There are 6497 observations and 13 variables (14 if you include the new response variable added later).

```
any(is.na(wine))
```

```
[1] FALSE
```

There are no NAs in our data, so we shouldn't be concerned about missing data.

Data Editing

```
wine<-wine%>%
  mutate(good_wine=if_else(quality >= 7,"1","0"))
wine<-wine%>%
  mutate(good_wine=as.factor(good_wine))
```

```
wine<-wine%>%
  mutate(good_wine_names=if_else(good_wine=="1","Good wine","Bad or subpar wine"))
no1 <- colnames(wine)[1:11]
colnames(wine)[1:11] = paste("c_", no1, sep = "")
```

EDA

Methodology

###Logistic Model

```
set.seed(222)
wine_split <- initial_split(wine, prop = 3/4)
wine_train <- training(wine_split)
wine_test <- training(wine_split)
```

```
wine_spec <- logistic_reg() %>%
  set_engine("glm")
```

```
wine_rec1 <- recipe(
  good_wine ~., data = wine_train) %>%
  step_rm(good_wine_names,quality) %>%
  step_center(all_numeric_predictors())%>%
  step_dummy(all_nominal_predictors()) %>%
  step_zv(all_predictors())
```

```
wine_wflow1 <- workflow() %>%
  add_model(wine_spec) %>%
  add_recipe(wine_rec1)
```

```
wine_fit1 <- wine_wflow1 %>%
  fit(data = wine_train)
kable(tidy(wine_fit1), digits = 3)
```

Reduced Model

term	estimate	std.error	statistic	p.value
(Intercept)	-1.195	0.205	-5.818	0.000
c_fixed.acidity	0.528	0.077	6.864	0.000
c_volatile.acidity	-3.492	0.433	-8.058	0.000
c_citric.acid	-0.295	0.390	-0.757	0.449
c_residual.sugar	0.236	0.030	7.910	0.000
c_chlorides	-6.019	2.718	-2.214	0.027
c_free.sulfur.dioxide	0.011	0.003	3.188	0.001
c_total.sulfur.dioxide	-0.003	0.002	-2.248	0.025
c_density	-437.756	75.268	-5.816	0.000
c_pH	2.620	0.417	6.289	0.000
c_sulphates	2.350	0.328	7.158	0.000
c_alcohol	0.461	0.091	5.064	0.000
color_white	-0.889	0.271	-3.283	0.001

Should we remove Citric Acid, lets do a quick Anova test ::: {.cell}

```
wine_rec2 <- recipe(
  good_wine ~., data = wine_train) %>%
  step_rm(good_wine_names,quality,c_citric.acid) %>%
  step_center(all_numeric_predictors())%>%
  step_dummy(all_nominal_predictors()) %>%
  step_zv(all_predictors())
wine_wflow2 <- workflow() %>%
  add_model(wine_spec) %>%
  add_recipe(wine_rec2)
wine_fit2 <- wine_wflow2 %>%
  fit(data = wine_train)
```

:::

```
fit_engine1<-extract_fit_engine(wine_fit1)
fit_engine2<-extract_fit_engine(wine_fit2)
anova(fit_engine2, fit_engine1, test = "Chisq") %>%
    kable(digits = 3)
```

Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
4860	3876.489	NA	NA	NA
4859	3875.914	1	0.575	0.448

We should remove cetric acid based on those results. (Insert Interruptation)

Full Model

```
wine_rec_full <- recipe(
  good_wine ~., data = wine_train) %>%
  step_rm(good_wine_names,quality,c_citric.acid) %>%
  step_dummy(color)%>%
  step_interact(terms = ~starts_with("c_"):starts_with("color")) %>%
  step_dummy(all_nominal_predictors()) %>%
  step_zv(all_predictors())
```

```
wine_flow_model <- workflow() %>%
  add_model(wine_spec) %>%
  add_recipe(wine_rec_full)
```

```
wine_fit_test <- wine_flow_model %>%
  fit(data = wine_train)

tidy(wine_fit_test,conf.int = T) %>%
  kable(digits = 3)
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	250.048	114.531	2.183	0.029	26.260	475.934
c_fixed.acidity	0.322	0.133	2.417	0.016	0.061	0.585
c_volatile.acidity	-2.611	0.703	-3.712	0.000	-4.016	-1.258
c_residual.sugar	0.255	0.079	3.230	0.001	0.096	0.408
c_chlorides	-6.772	3.328	-2.035	0.042	-14.455	-1.191
c_free.sulfur.dioxide	0.001	0.013	0.105	0.917	-0.025	0.027
c_total.sulfur.dioxide	-0.013	0.005	-2.741	0.006	-0.023	-0.004
c_density	-265.185	117.071	-2.265	0.024	-496.086	-36.451
c_pH	0.248	1.077	0.231	0.818	-1.877	2.351
c_sulphates	3.591	0.606	5.925	0.000	2.405	4.794
c_alcohol	0.739	0.137	5.413	0.000	0.474	1.010
color_white	457.236	159.417	2.868	0.004	145.221	770.431
c_fixed.acidity_x_color_white	0.276	0.170	1.618	0.106	-0.058	0.610
c_volatile.acidity_x_color_white	-0.958	0.889	-1.077	0.281	-2.691	0.797
c_residual.sugar_x_color_white	0.076	0.089	0.853	0.393	-0.097	0.255

term	estimate	std.error	statistic	p.value	conf.low	conf.high
c_chlorides_x_color_white	-4.365	5.447	-0.801	0.423	-14.937	6.620
c_free.sulfur.dioxide_x_color_white	0.007	0.014	0.518	0.604	-0.020	0.034
c_total.sulfur.dioxide_x_color_white	0.014	0.005	2.666	0.008	0.004	0.024
c_density_x_color_white	-466.493	162.305	-2.874	0.004	-785.334	-148.800
c_pH_x_color_white	3.294	1.188	2.772	0.006	0.973	5.636
c_sulphates_x_color_white	-1.490	0.729	-2.043	0.041	-2.930	-0.062
c_alcohol_x_color_white	-0.651	0.191	-3.418	0.001	-1.027	-0.280

As we can see from some variables p values and confidence interval, we can drop some of those valuables if we were to conduct to a hypothesis test since their p value would exceed 0.05, meaning that we would not have enough to rejec the null hypothesis. (better wording later)

```
wine_full_reduced <- recipe(
  good_wine ~., data = wine_train) %>%
  step_rm(good_wine_names,quality,c_citric.acid) %>%
  step_dummy(color)%>%
  step_interact(terms = ~starts_with("c_"):starts_with("color")) %>%
  step_rm(c_sulphates_x_color_white,c_free.sulfur.dioxide_x_color_white,c_chlorides_x_color_w
          c_residual.sugar_x_color_white,c_volatile.acidity_x_color_white,c_sulphates_x_color
          c_fixed.acidity_x_color_white,c_pH)%>%
  step_dummy(all_nominal_predictors()) %>%
  step_zv(all_predictors())
```

```
wine_full_reduced_workflow<- workflow() %>%
  add_model(wine_spec) %>%
  add_recipe(wine_full_reduced)

wine_fit_test <- wine_full_reduced_workflow %>%
  fit(data = wine_train)

tidy(wine_fit_test,conf.int = T) %>%
  kable(digits = 3)
```

term	estimate	std.error	statistic	p.value	conf.low	conf.high
(Intercept)	303.019	80.959	3.743	0.000	144.963	462.700
c_fixed.acidity	0.408	0.066	6.230	0.000	0.280	0.537
c_volatile.acidity	-3.149	0.424	-7.435	0.000	-3.989	-2.329
c_residual.sugar	0.280	0.031	9.029	0.000	0.219	0.341

term	estimate	std.error	statistic	p.value	conf.low	conf.high
c_chlorides	-8.512	2.913	-2.922	0.003	-14.573	-3.260
c_free.sulfur.dioxide	0.009	0.003	2.602	0.009	0.002	0.016
c_total.sulfur.dioxide	-0.013	0.003	-4.046	0.000	-0.020	-0.007
c_density	-316.798	81.116	-3.905	0.000	-476.803	-158.461
c_sulphates	2.400	0.334	7.181	0.000	1.743	3.054
c_alcohol	0.700	0.115	6.107	0.000	0.478	0.928
color_white	269.424	75.410	3.573	0.000	121.650	417.503
c_total.sulfur.dioxide_x_color_white	0.013	0.003	3.714	0.000	0.006	0.020
c_density_x_color_white	-277.075	74.987	-3.695	0.000	-424.319	-130.128
c_pH_x_color_white	2.825	0.393	7.195	0.000	2.058	3.598
c_alcohol_x_color_white	-0.467	0.140	-3.338	0.001	-0.745	-0.196

```
AIC_fit<- logistic_reg() %>%
  set_engine("glm") %>%
  fit(good_wine~.-c_citric.acid-quality-good_wine_names,
  data = wine_train)
AIC_fit<- repair_call(AIC_fit, data = wine_train)
AIC_fit_engine<-AIC_fit %>% extract_fit_engine()
```

```
best_AIC_model<-stepAIC(AIC_fit_engine,direction="forward",trace=FALSE)
```

```
best_AIC_model%>%tidy()
```

Stepwise

```
# A tibble: 12 x 5
   term                      estimate std.error statistic  p.value
   <chr>                        <dbl>     <dbl>     <dbl>    <dbl>
 1 (Intercept)                420.       74.1          5.67 1.42e- 8
 2 c_fixed.acidity              0.515     0.0750       6.87 6.50e-12
 3 c_volatile.acidity          -3.38      0.408       -8.29 1.10e-16
 4 c_residual.sugar             0.236     0.0298       7.93 2.23e-15
 5 c_chlorides                 -6.14      2.70        -2.27 2.30e- 2
 6 c_free.sulfur.dioxide     0.0108    0.00334      3.22 1.29e- 3
 7 c_total.sulfur.dioxide   -0.00350   0.00150     -2.33 1.96e- 2
 8 c_density              -441.       75.1         -5.88 4.16e- 9
 9 c_pH                      2.63      0.416        6.33 2.46e-10
10 c_sulphates               2.34      0.328        7.13 9.78e-13
11 c_alcohol                 0.451     0.0900       5.01 5.43e- 7
12 colorwhite               -0.901     0.270       -3.33 8.58e- 4
```

Multnomial Regression

```
full_fit1<- multinom_reg() %>%
  set_engine("nnet") %>%
  fit(as.factor(quality)~.-good_wine_names-good_wine,
  data = wine_train)
full_fit1<- repair_call(full_fit1, data = wine_train)
tidy(full_fit1)
```

Data Editing for Regression

```
# A tibble: 78 x 6
   y.level term                   estimate std.error statistic   p.value
   <chr>   <chr>                     <dbl>     <dbl>     <dbl>     <dbl>
 1 4       (Intercept)             -2.57     0.104     -24.7   1.39e-134
 2 4       c_fixed.acidity         -0.757    0.144      -5.27  1.39e-  7
 3 4       c_volatile.acidity      -0.209    0.524      -0.398 6.91e-  1
 4 4       c_citric.acid            4.49     0.581       7.73  1.11e- 14
 5 4       c_residual.sugar        -0.125    0.0636     -1.96  4.99e-  2
 6 4       c_chlorides            -19.0      0.0858   -221.    0
 7 4       c_free.sulfur.dioxide   -0.0748   0.0135     -5.53  3.25e-  8
 8 4       c_total.sulfur.dioxide  -0.0107   0.00566    -1.90  5.77e-  2
 9 4       c_density               31.1      0.104     299.    0
10 4       c_pH                    -3.14     0.457      -6.87  6.55e- 12
# ... with 68 more rows
```

```
full_fit1_engine<-full_fit1 %>% extract_fit_engine()
newmodel<-stepAIC(full_fit1_engine,direction="both",trace=FALSE)
```

```
tidy(newmodel)
```

```
# A tibble: 72 x 6
   y.level term                     estimate std.error statistic       p.value
   <chr>   <chr>                       <dbl>     <dbl>     <dbl>         <dbl>
 1 4       (Intercept)            -18.0       0.101    -178.    0
 2 4       c_fixed.acidity         -0.374     0.142      -2.63  0.00845
 3 4       c_volatile.acidity      -1.29      0.502      -2.58  0.00990
 4 4       c_residual.sugar        -0.0905    0.0601     -1.51  0.132
 5 4       c_chlorides            -15.0       0.0734   -205.    0
 6 4       c_free.sulfur.dioxide   -0.0783    0.0136     -5.78  0.00000000763
 7 4       c_total.sulfur.dioxide  -0.00555   0.00578    -0.960 0.337
 8 4       c_density               37.4       0.101     370.    0
 9 4       c_pH                    -1.78      0.440      -4.04  0.0000542
10 4       c_sulphates              2.85      0.574       4.97  0.000000679
# ... with 62 more rows
```

Results

```
fit2_aug <- augment(wine_fit2, new_data = wine_test)

fit2_conf<-fit2_aug%>%
  count(good_wine,.pred_class,.drop=FALSE)%>%
    pivot_wider(names_from = .pred_class, values_from = n)
fit2_conf
```

```
# A tibble: 2 x 3
  good_wine   `0`   `1`
  <fct>     <int> <int>
1 0          3709   184
2 1           702   277
```

```
predicted <- predict(wine_fit2, wine_test)
predicted<-predicted%>%mutate(.pred_class=as.numeric(.pred_class))
optimal <- optimalCutoff(as.numeric(wine_test$good_wine), predicted)[1]

mis1<-misClassError(as.numeric(wine_test$good_wine), predicted, threshold=optimal)
accuracy <- mean(as.numeric(wine_test$good_wine)== as.numeric(predicted$.pred_class))
```

```
newmodel$AIC
```

```
[1] 10449.57
```

```
glance(full_fit1)$AIC
```

```
[1] 10460.85
```

```
training_pred <- predict(newmodel,wine_test)
training_pred<-data_frame(training_pred)
```

```
Warning: `data_frame()` was deprecated in tibble 1.1.0.
Please use `tibble()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```

```
accuracy <- mean(wine_test$quality == training_pred$training_pred)
```