Red Wine Quality

Nailah Rawnaq

12/6/2021

Datasets

This data set consist of 1,599 rows and 12 columns. The description of the columns are given below.

- fixed acidity : most acids involved with wine or fixed or nonvolatile.
- volatile acidity : the amount of acetic acid in wine, which at too high of levels can lead to an unpleasant, vinegar taste.
- citric acid : found in small quantities, citric acid can add 'freshness' and flavor to wines.
- residual sugar : the amount of sugar remaining after fermentation stops.
- chlorides : the amount of salt in the wine.
- free sulfur dioxide : the free form of SO2 exists in equilibrium between molecular SO2 (as a dissolved gas) and bisulfite ion.
- total sulfur dioxide : amount of free and bound forms of S02.
- density : the density of water is close to that of water depending on the percent alcohol and sugar content.
- pH : describes how acidic or basic a wine is on a scale from 0 (very acidic) to 14 (very basic).
- sulphates : a wine additive which can contribute to sulfur dioxide gas (S02) levels.
- alcohol : the percent alcohol content of the wine
- quality (score between 0 and 10)

This is a data set from *kaggle* in a title of "Red Wine Quality".

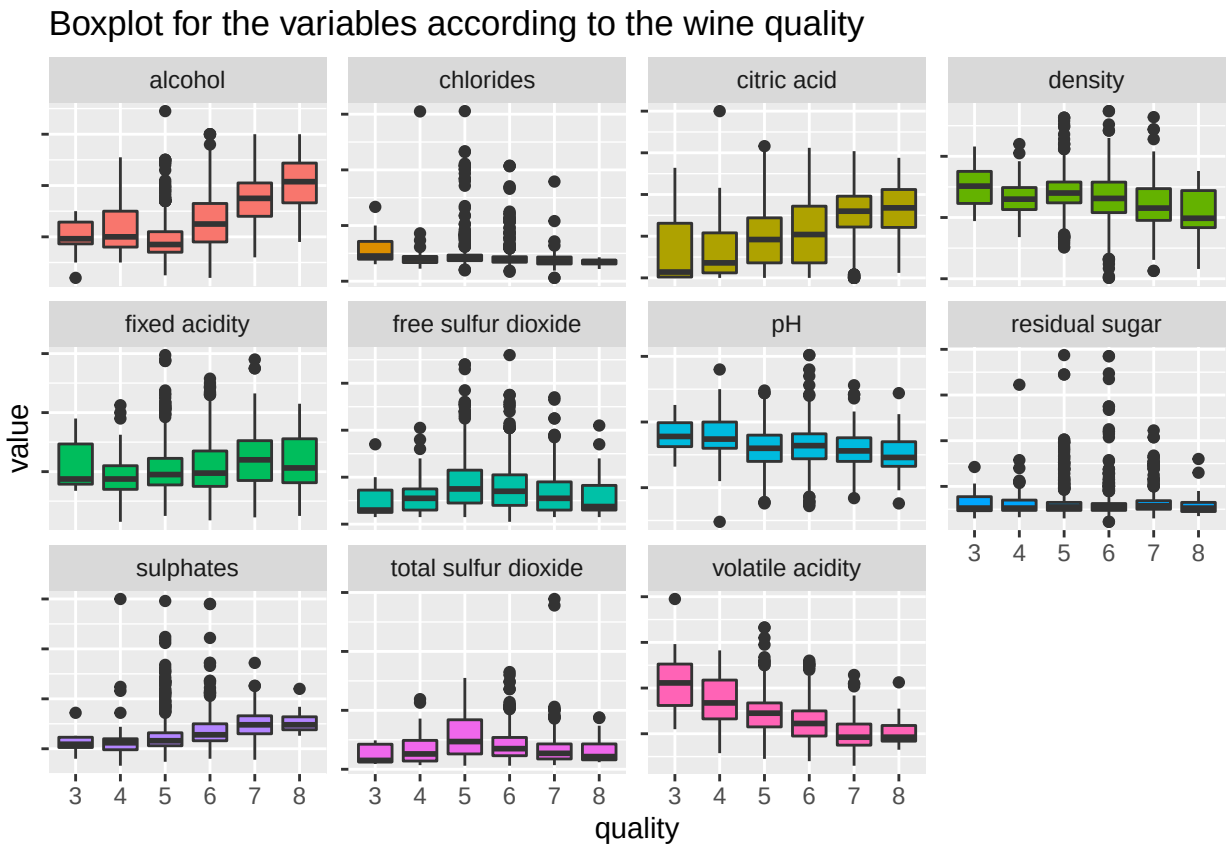
Basic Stats

```
df %>%
    summary()
```

```
##  fixed acidity   volatile acidity  citric acid    residual sugar
##  Min.   : 4.60   Min.   :0.1200   Min.   :0.000   Min.   : 0.900
##  1st Qu.: 7.10   1st Qu.:0.3900   1st Qu.:0.090   1st Qu.: 1.900
##  Median : 7.90   Median :0.5200   Median :0.260   Median : 2.200
##  Mean   : 8.32   Mean   :0.5278   Mean   :0.271   Mean   : 2.539
##  3rd Qu.: 9.20   3rd Qu.:0.6400   3rd Qu.:0.420   3rd Qu.: 2.600
##  Max.   :15.90   Max.   :1.5800   Max.   :1.000   Max.   :15.500
##    chlorides       free sulfur dioxide total sulfur dioxide    density
##  Min.   :0.01200   Min.   : 1.00       Min.   :  6.00       Min.   :0.9901
##  1st Qu.:0.07000   1st Qu.: 7.00       1st Qu.: 22.00       1st Qu.:0.9956
##  Median :0.07900   Median :14.00       Median : 38.00       Median :0.9968
##  Mean   :0.08747   Mean   :15.87       Mean   : 46.47       Mean   :0.9967
##  3rd Qu.:0.09000   3rd Qu.:21.00       3rd Qu.: 62.00       3rd Qu.:0.9978
##  Max.   :0.61100   Max.   :72.00       Max.   :289.00       Max.   :1.0037
##        pH          sulphates         alcohol         quality
##  Min.   :2.740   Min.   :0.3300   Min.   : 8.40   Min.   :3.000
##  1st Qu.:3.210   1st Qu.:0.5500   1st Qu.: 9.50   1st Qu.:5.000
##  Median :3.310   Median :0.6200   Median :10.20   Median :6.000
##  Mean   :3.311   Mean   :0.6581   Mean   :10.42   Mean   :5.636
##  3rd Qu.:3.400   3rd Qu.:0.7300   3rd Qu.:11.10   3rd Qu.:6.000
##  Max.   :4.010   Max.   :2.0000   Max.   :14.90   Max.   :8.000
```

Boxplot for the variables according to the wine quality

```
df %>%
    pivot_longer(-quality) %>%
    mutate(quality = as.factor(quality)) %>%
    ggplot(aes(quality, value, fill = name)) +
    geom_boxplot(show.legend = F) +
    facet_wrap(~name, scales = "free_y") +
    scale_y_continuous(labels = NULL) +
    labs(title = "Boxplot for the variables according to the wine quality")
```



We can see from this plot that mean level of some variables like alcohol, citric acid, pH, sulphates and volatile acidity.

Correlation among the variables

```
corrplot::corrplot(cor(df))
```

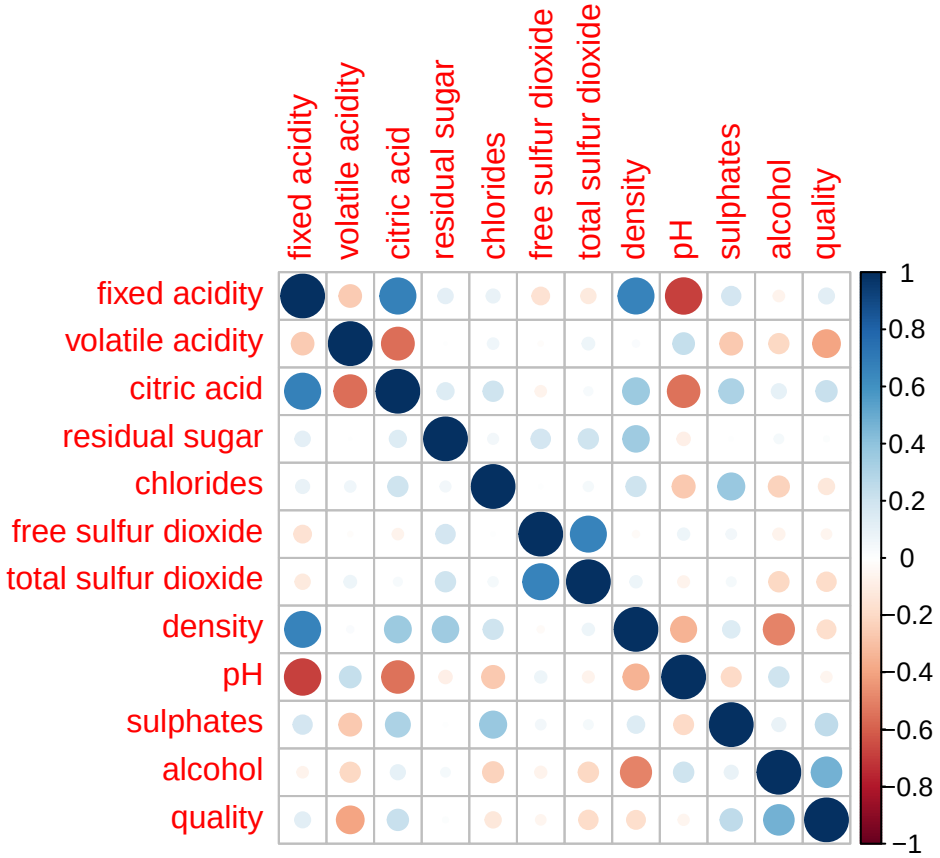


Table for shoing the important variable as a output of linear regression with highest magnitude

```
cor(select(df, -quality)) %>%
    as.data.frame() %>%
    rownames_to_column() %>%
    pivot_longer(-rowname) %>%
    filter(value != 1) %>%
    arrange(-abs(value)) %>%
    group_by(name) %>%
    summarise(mean_corr = mean(abs(value))) %>%
    arrange(-mean_corr) %>%
    pander::pander()
```

name	mean_corr
fixed acidity	0.2999
citric acid	0.2998
pH	0.2691
density	0.2691
alcohol	0.1708
volatile acidity	0.1679
sulphates	0.1667
total sulfur dioxide	0.153
chlorides	0.1525
free sulfur dioxide	0.1299
residual sugar	0.1194

Since fixed acidity has the higher mean absolute correlation so we going to eliminate this and fit a linear regression model to see find out the most important sets of variables.

Important variables

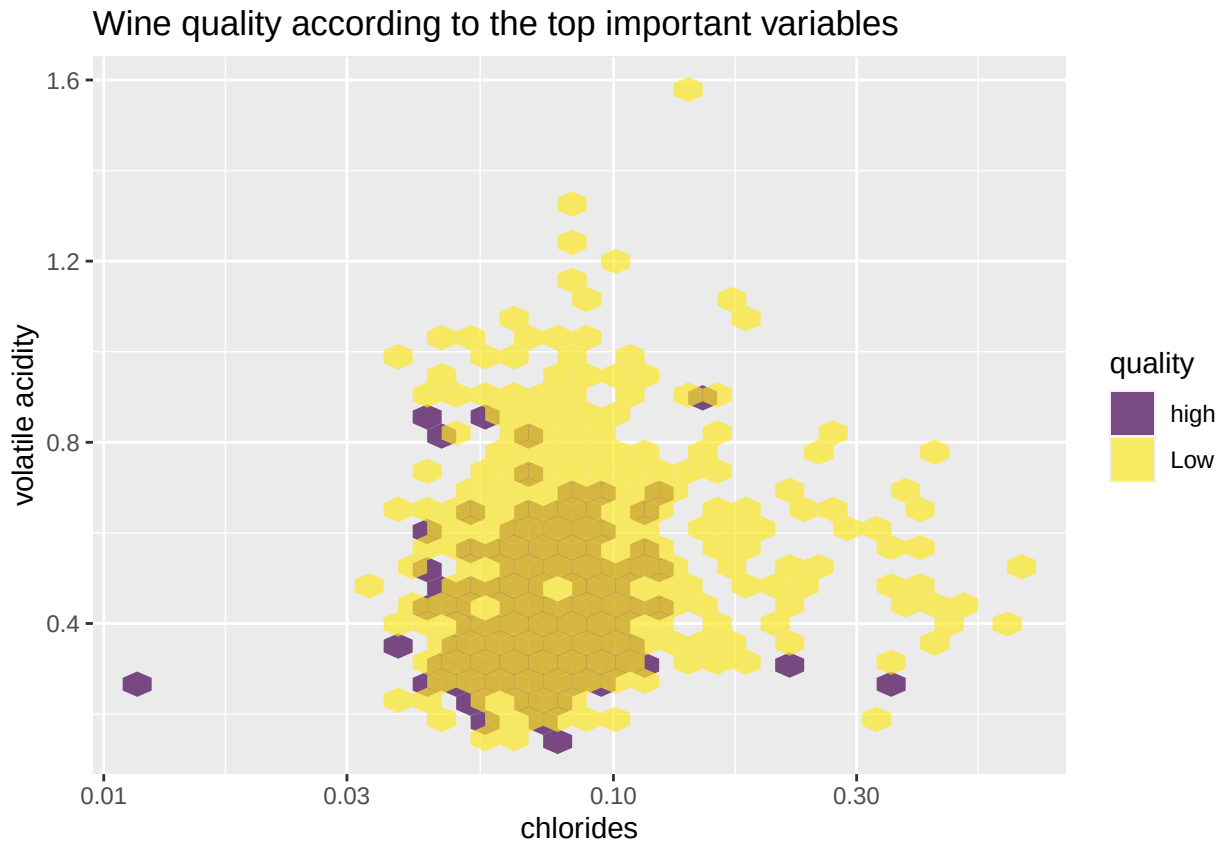
```
(df %>%
    select(-`fixed acidity`) %>%
    lm(formula = quality ~ .) %>%
    summary())$coefficients %>%
    as.data.frame() %>%
    rownames_to_column() %>%
    filter(`Pr(>|t|)` < .05) %>%
    arrange(-abs(Estimate)) %>%
    pander::pander()
```

rowname	Estimate	Std. Error	t value	Pr(>|t|)
chlorides	-1.968	0.4077	-4.828	1.51e-06
`volatile acidity`	-1.078	0.1209	-8.911	1.353e-18
sulphates	0.8996	0.113	7.961	3.232e-15
pH	-0.5462	0.1333	-4.099	4.358e-05
alcohol	0.2901	0.02223	13.05	4.917e-37
`free sulfur dioxide`	0.004592	0.002158	2.128	0.03352
`total sulfur dioxide`	-0.003427	0.0007089	-4.835	1.463e-06

So according to this, variable chlorides and volatile acidity are the important variabes. So we will plot those two variables according to the quatity of wine.

Seperating the high and low class points (important variables)

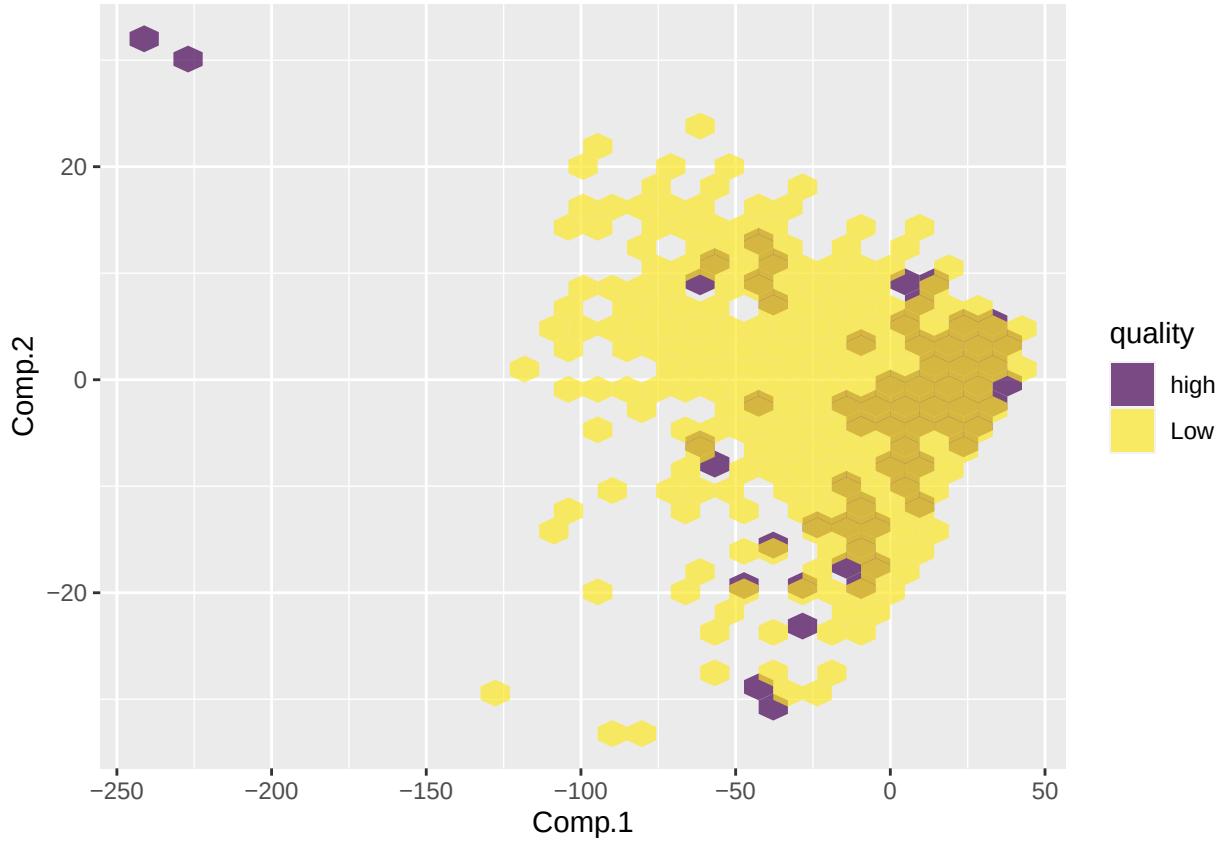
```
df %>%
    select(chlorides, `volatile acidity`, quality) %>%
    mutate(quality = if_else(quality <= 6,"Low", "high")) %>%
    ggplot(aes(chlorides, `volatile acidity`, fill = quality)) +
    geom_hex(alpha = .7) +
    scale_fill_viridis_d() +
    scale_x_log10() +
    labs(title = "Wine quality according to the top important variables")
```



We can see that this plot is not performing great to seperate the points. we will perform pca for this.

Seperating the high and low class points (PCA)

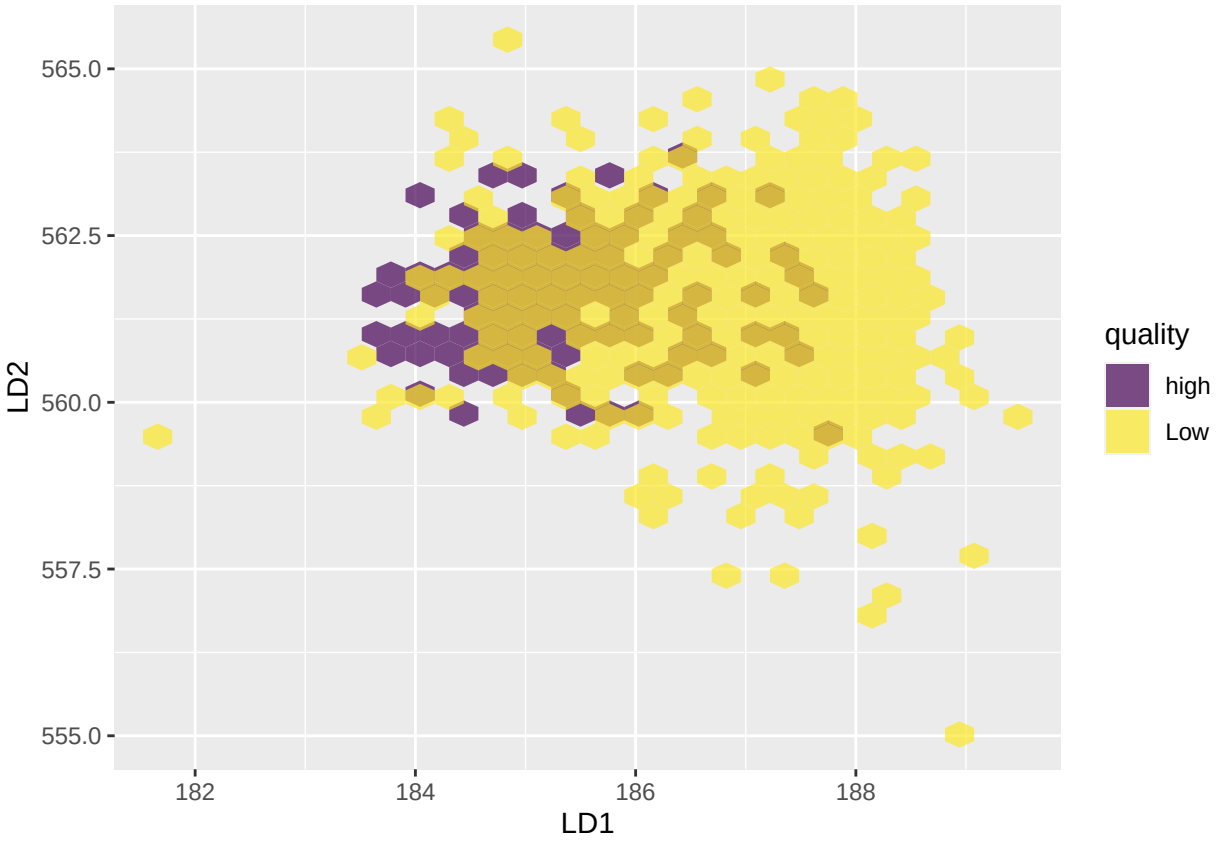
```
(df %>%
    select(-quality) %>%
    princomp())$score %>%
    as.data.frame() %>%
    select(Comp.1, Comp.2) %>%
    bind_cols(quality = df$quality) %>%
    mutate(quality = if_else(quality <= 6,"Low", "high")) %>%
    ggplot(aes(Comp.1, Comp.2, fill = quality)) +
    geom_hex(alpha = .7) +
    scale_fill_viridis_d()
```



This is not also performing well to seperate the points. we will use LDA for it

Seperating the high and low class points (LDA)

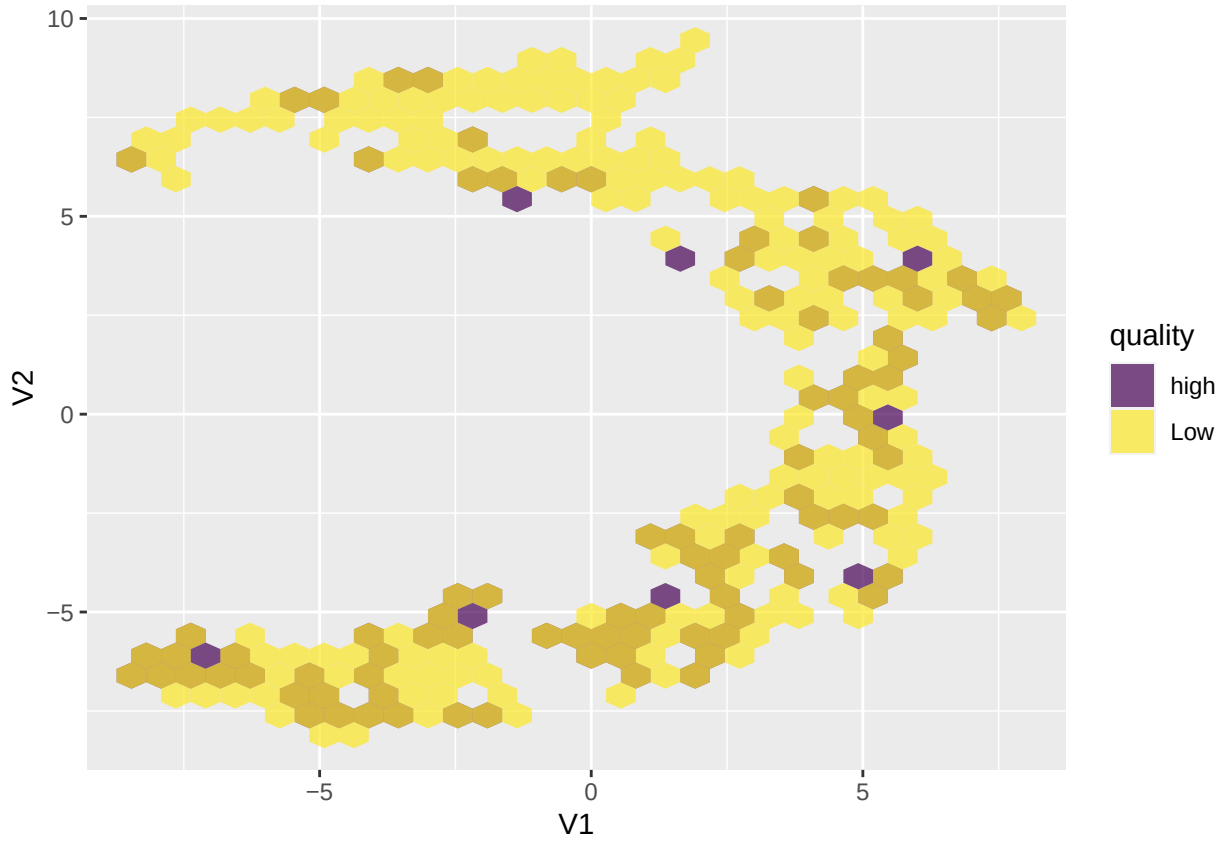
```
(as.matrix(select(df, -quality)) %*%
(MASS::lda(quality ~ .,
          mutate(df,quality =
                     case_when(quality <= 4 ~ "Low",
                               quality <= 6 ~ "Medium",
                               TRUE ~ "high"))))$scaling) %>%
    as.data.frame() %>%
    bind_cols(quality = if_else(df$quality <= 6,"Low", "high")) %>%
    ggplot(aes(LD1, LD2, fill = quality)) +
    geom_hex(alpha = .7) +
    scale_fill_viridis_d()
```



The output is much better than the previous two. And there is a seperable line for that.

Seperating the high and low class points(UMAP)

```
umap::umap(select(df, - quality))$layout %>%
    as.data.frame() %>%
    bind_cols(quality = if_else(df$quality <= 6,"Low", "high")) %>%
    ggplot(aes(V1, V2, fill = quality)) +
    geom_hex(alpha = .7) +
    scale_fill_viridis_d()
```



No the performance of Umap is not better than the LDA.