Lab 3: Testing Mean Vectors

Anish Yakkala

For this lab, we will try to determine which factors contribute to the percieved quality of red wine. The following code will load your dataset.

```
# Load the data
wine = read.csv("http://kbodwin.web.unc.edu/files/2017/11/redWines.csv")
```

Here is a description of the dataset supplied by the creators:

```
P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis.
  Modeling wine preferences by data mining from physicochemical properties.
  In Decision Support Systems, Elsevier, 47(4):547-553. ISSN: 0167-9236.

  In the above reference, two datasets were created, using red and white wine samples.
  The inputs include objective tests (e.g. PH values) and the output is based on sensory data
  (median of at least 3 evaluations made by wine experts). Each expert graded the wine quality
  as 0 (bad) or 1 (good).

Attribute information:

   For more information, read [Cortez et al., 2009].

   Input variables (based on physicochemical tests):
   1 - fixed acidity (tartaric acid - g / dm^3)
   2 - volatile acidity (acetic acid - g / dm^3)
   3 - citric acid (g / dm^3)
   4 - residual sugar (g / dm^3)
   5 - chlorides (sodium chloride - g / dm^3
   6 - free sulfur dioxide (mg / dm^3)
   7 - total sulfur dioxide (mg / dm^3)
   8 - density (g / cm^3)
   9 - pH
   10 - sulphates (potassium sulphate - g / dm3)
   11 - alcohol (% by volume)
   Output variable (based on sensory data):
   12 - quality (score of 0 or 1)

Description of attributes:

   1 - fixed acidity: most acids involved with wine or fixed or nonvolatile (do not evaporate readily)

   2 - volatile acidity: the amount of acetic acid in wine, which at too high of levels can lead to an u

   3 - citric acid: found in small quantities, citric acid can add 'freshness' and flavor to wines

   4 - residual sugar: the amount of sugar remaining after fermentation stops, it's rare to find wines w
       less than 1 gram/liter and wines with greater than 45 grams/liter are considered sweet

   5 - chlorides: the amount of salt in the wine

   6 - free sulfur dioxide: the free form of SO2 exists in equilibrium between molecular SO2 (as a disso
```

```
    and bisulfite ion; it prevents microbial growth and the oxidation of wine

7 - total sulfur dioxide: amount of free and bound forms of SO2; in low concentrations,
    SO2 is mostly undetectable in wine, but at free SO2 concentrations over 50 ppm, SO2 becomes evid

8 - density: the density of water is close to that of water depending on the percent alcohol and sug

9 - pH: describes how acidic or basic a wine is on a scale from 0 (very acidic) to 14 (very basic);
    most wines are between 3-4 on the pH scale

10 - sulphates: a wine additive which can contribute to sulfur dioxide gas (SO2) levels,
     wich acts as an antimicrobial and antioxidant

11 - alcohol: the percent alcohol content of the wine

Output variable (based on sensory data):
12 - quality (score or 0 and 1)
```

Part One: In-Class

This section will be graded for completeness, not correctness. Do your best in class, even if you can't figure out the answers right away.

Exploring The Dataset

Our dataset today contains only some of the **red wines** from this study.

How many wines were rated "bad"? How many were rated "good"?

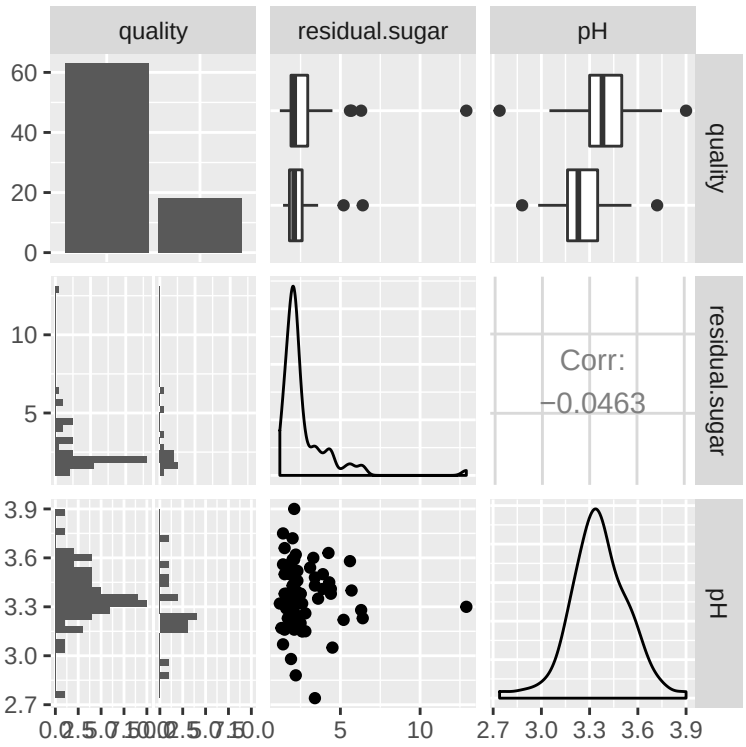
```
0: 63
1: 18
```

We will begin by looking at a small subset of the wine data, containing the residual sugar and pH level of the studied wines.

```
wine_sub <- wine %>%
  select(quality, residual.sugar, pH) %>%
  mutate(
    quality = factor(quality)
  )
```

Apply the function `ggpairs()` to your new subset to plot all the variable pairs.

```
ggpairs(wine_sub)
```



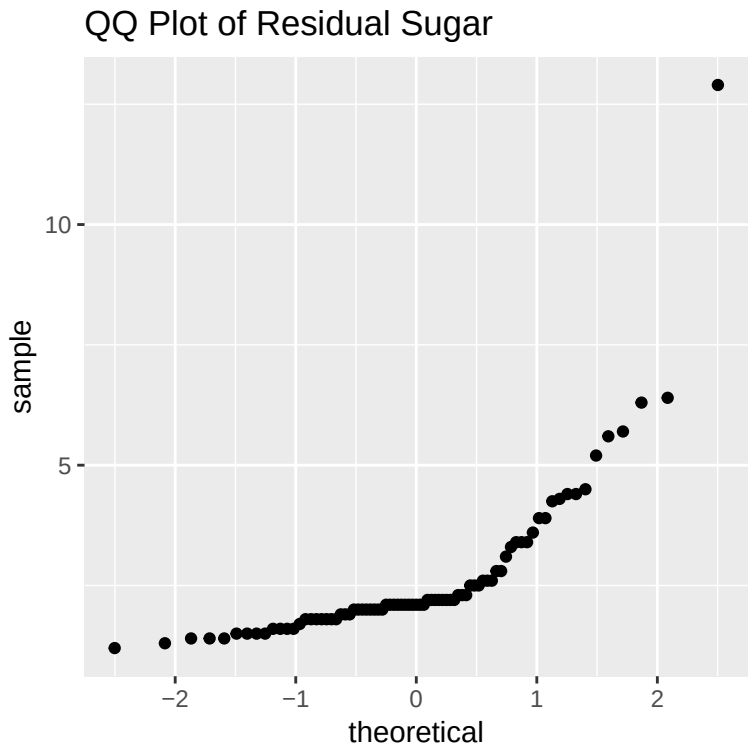
What can you say about this data based on the plot?

pH has normal looking distribution, there seems to be a weak correlation between residual sugar and pH. The Residual Sugars seem to be distributed fairly similiraily between the two groups.

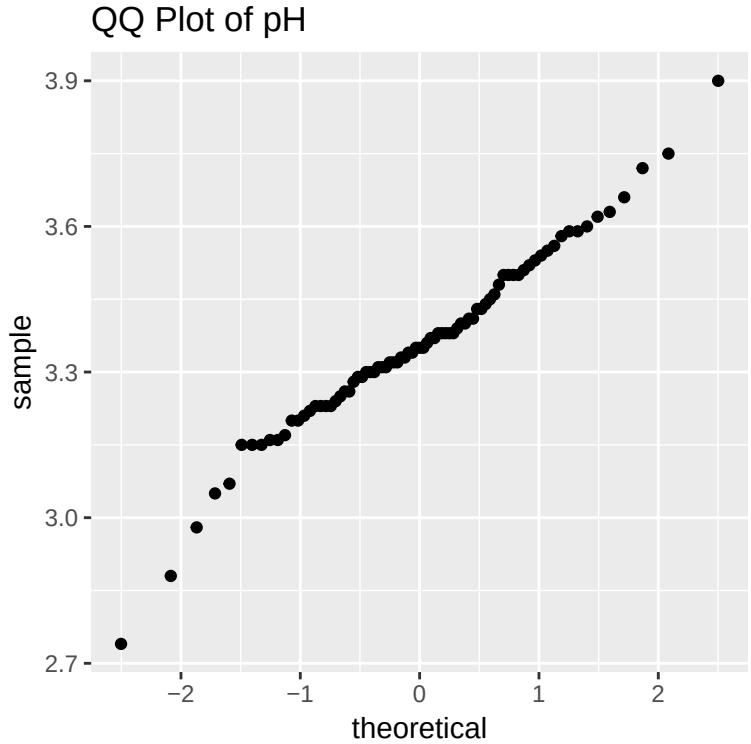
Checking Multivariate Normality

Add to the following code chunk to produce a Normal Q-Q plots for pH as well as residual sugar.

```
ggplot(wine_sub, aes(sample = residual.sugar)) + geom_qq() + ggtitle("QQ Plot of Residual Sugar")
```




```
ggplot(wine_sub, aes(sample = pH)) + geom_qq() + ggtitle("QQ Plot of pH")
```



Comment on these plots. What do you conclude?

```
Residual sugars seems to trail off the expected values near the end. Which gives suggestive evidence tha
```

Create a data matrix Y for the two numeric variables `pH` and `residual.sugar`. Then use the `mvn()` function from the MVN package to examine this data matrix.

```
### YOUR CODE HERE ###
Y <- wine_sub %>%
          select(residual.sugar,pH)

mvn(Y)
```

```
## $multivariateNormality
##              Test        Statistic              p value Result
## 1 Mardia Skewness 175.291308921256 7.64950540199223e-37     NO
## 2 Mardia Kurtosis 20.1559472401511                    0     NO
## 3             MVN             <NA>                 <NA>     NO
##
## $univariateNormality
##           Test       Variable Statistic   p value Normality
## 1 Shapiro-Wilk residual.sugar     0.639  <0.001      NO
## 2 Shapiro-Wilk       pH           0.979  0.1927      YES
##
## $Descriptives
##                 n Mean Std.Dev Median  Min  Max 25th 75th   Skew Kurtosis
## residual.sugar 81 2.66   1.631   2.10 1.20 12.9 1.80 2.80  3.528    17.31
## pH             81 3.36   0.186   3.35 2.74  3.9 3.25 3.48 -0.187     1.24
```

Is it reasonable to treat this data matrix as multivariate Normal? Why or why not?

```
Both the Mardia skewness and Kurtosis tests gave a p value of <.001. So at the 5% significance level we
```

Practice with Multivariate Normality

Suppose you are told that this data **is** from a multivariate Normal distribution. Find the sample mean vector and sample covariance matrix.

```
### EDIT THIS CODE ###
y_bar <- as.matrix(colMeans(Y))

S_y <- cov(Y)
```

Suppose it is hypothesized that on average for red wine, the residual sugar is equal to 80% of the pH. Test this hypothesis. Do **not** define a new variable in your dataset; instead, use your matrix Y and your calculations above. Be sure to state your hypotheses and conclusion - you may use R to find an exact p-value, or you may compare your result to a critical value of 2.

Hint: Define a new variable W that is a linear combination of variables in Y. What is the distribution of W?

```
a <- matrix(c(1,-0.8),2,1)

# Sample d
W <- t(a) %*% (y_bar)

S_w <- (t(a) %*% S_y %*% a)

SE <- sqrt((1/81 + 1/81) * S_w)

test_statistic <- W/SE
```

$$H_o : \mu_{residual.sugar} - 0.8 * \mu_{pH} = 0$$
$$H_a : \mu_{residual.sugar} - 0.8 * \mu_{pH} \neq 0$$
$$\alpha = 0.05$$

$$t = 0.0983$$

At the 5% significance level we can not reject the null hypothesis that residual sugar is equal to 80% of the pH, since our observed test statistic is 0.0983, which is not larger than the critical value of 2.

Testing Mean Vectors

Suppose it is now hypothesized that "good" wines have a different residual sugar and pH than "bad" wines.

Calculate the individual mean vectors and covariance matrices for good wines and for bad wines. Then calculate the pooled covariance matrix. Also find the generalized variance for each of these three covariance matrices.

```
### YOUR CODE HERE ###

n1 <- nrow(wine_sub %>%
    filter(quality == 0) %>%
      select(residual.sugar,pH))

n2 <- nrow(wine_sub %>%
    filter(quality == 1) %>%
      select(residual.sugar,pH))

y_bar_1 = as.matrix(colMeans(wine_sub %>%
                    filter(quality == 0) %>%
                      select(residual.sugar,pH)))
y_bar_2 = as.matrix(colMeans(wine_sub %>%
                    filter(quality == 1) %>%
                      select(residual.sugar,pH)))

S_1 = cov(wine_sub %>%
                    filter(quality == 0) %>%
                      select(residual.sugar,pH))
S_2 = cov(wine_sub %>%
                    filter(quality == 1) %>%
                      select(residual.sugar,pH))

S_p = (((n1 - 1) * S_1) + ((n2 - 1) * S_2))/(n1 + n2 - 2)

(det(S_1)^((n1-1)/2) * det(S_2)^((n2-1)/2))/det(S_p)^(18+63-2)
```

```
## [1] 1.39e+41
```

Compare these to each other, and to the `S_y` you found above. Do you think it makes sense to pool the covariance?

```
Using the test of determinants on the covariances matrices we find that the statistic is not close to 1
```

Calculate a T-squared statistic for this test, by using matrix algebra on your above calculations.

```
library(Hotelling)
d_bar <- y_bar_1 - y_bar_2

S_dbar <- S_1/n1 + S_2/n2

T_square <- t(d_bar) %*% solve(S_dbar) %*% d_bar
```

Why might we be comfortable performing a Hotelling's T-Squared test, even if we are not confident that the data is Multivariate Normal?

`Multivariate Central Limit Theorem, using the fact that we have with sample sizes greater than 15.`

Now take a look at the function `hotelling.test()`. Then run the code supplied below.

```
#install.packages("Hotelling")
library(Hotelling)

htest <- hotelling.test(. ~ quality, data = wine_sub)
htest
```

```
## Test stat:  2.9573
## Numerator df:  2
## Denominator df:  78
## P-value:  0.05781
```

Report these results. What were the hypotheses? What was the test statistic? What do you conclude?

$$H_o : \vec{\mu}_{good} - \vec{\mu}_{bad} = 0$$
$$H_a : \vec{\mu}_{good} - \vec{\mu}_{bad} \neq 0$$
$$\alpha = 0.05$$

$$T^2 = 2.9573$$
$$\text{p-value } = 0.05781$$

At the 5% significance level we can not reject the null hypothesis that the difference in mean pH and residual sugar for good and bad wines are the same.

Note: Since this test involved only a 2x2 matrix, you might want to consider using this data to practice doing T^2 tests by hand, and then compare your answers to the R output.

Follow-up: Specific differences

We may wish to ask ourselves where, exactly, the differences in mean vectors lie. Do good wines have different pH than bad wines? Do they have different residual sugar? Use R to perform individual two-sample t-tests for the two variables. Hint: The function `t.test()` takes similar input as `hotelling.test()`

```
tidy(t.test(pH ~ quality, data = wine_sub))
```

```
## # A tibble: 1 x 10
##   estimate estimate1 estimate2 statistic p.value parameter conf.low
```

```
##      <dbl>     <dbl>     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>
## 1    0.117      3.38      3.27      2.24  0.0342      24.9  0.00940
## # ... with 3 more variables: conf.high <dbl>, method <chr>,
## #   alternative <chr>
```

```
tidy(t.test(residual.sugar ~ quality, data = wine_sub))
```

```
## # A tibble: 1 x 10
##   estimate estimate1 estimate2 statistic p.value parameter conf.low
##      <dbl>     <dbl>     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>
## 1    0.107      2.68      2.58     0.286   0.777      36.0   -0.653
## # ... with 3 more variables: conf.high <dbl>, method <chr>,
## #   alternative <chr>
```

What do you conclude based on the individual t tests?

```
That we cannot reject the null hypothesis that the mean residual sugar for good and bad wines are the sa
```

Why did we run a Hotelling's T-squared first, rather than simply performing two separate t-tests?

```
The seperate t-tests do not take into account the relationship between pH and residual.sugar. This is wh
Also the results for the two tests are not one to one.
```

YOUR TURN

Congratulations! You now own a vineyard! Of course, your goal is produce only the best red wines. Your chemists can control all the properties, but they need to be told which ones to focus their efforts on.

Consider the full "wine" dataset. Perform a proper hypothesis test for all of the variables involved. Make sure you justify all assumptions. Be sure your conclusion tells your chemists which specific chemical properties should be made higher, made lower, or ignored in their wine creation.

Note: While you are certainly welcome to answer this question using your knowledge from other advanced courses, you are not required to do so for full credit. I am only looking for basic analyses and plot, accompanied by an accurate and clear interpretation.

Hint: below is some code I wrote for you to create a function called `ttest_all_vars`, to run all the t-tests at once. You absolutely do **not** need to understand this function! Simply run this code, and the example below it for pH and residual sugar. You may use this function if you wish to make your process easier.

```
ttest_all_vars <- function(data, response){

  res <- data %>%
    select(-response) %>%
    map_df(~tidy(t.test(.x ~ data[,response]))) %>%
    mutate(
      Chemical.Property = names(wine %>%
                                  select(-response)),
      observed.diff = estimate,
      t.score = statistic
    )

  res <- res %>% select(Chemical.Property, observed.diff, t.score, p.value)

  return(res)
}
```

Normality

```
mvn(as.matrix(wine %>%
                select(-quality)))
```

```
## $multivariateNormality
##              Test        Statistic               p value Result
## 1 Mardia Skewness 1392.91885412925 7.77910302082619e-145     NO
## 2 Mardia Kurtosis   20.78199521824                     0     NO
## 3             MVN             <NA>                  <NA>     NO
##
## $univariateNormality
##            Test             Variable Statistic   p value Normality
## 1  Shapiro-Wilk    fixed.acidity         0.954  0.0053      NO
## 2  Shapiro-Wilk   volatile.acidity       0.957  0.0088      NO
## 3  Shapiro-Wilk     citric.acid          0.870  <0.001      NO
## 4  Shapiro-Wilk    residual.sugar        0.639  <0.001      NO
## 5  Shapiro-Wilk      chlorides           0.411  <0.001      NO
## 6  Shapiro-Wilk free.sulfur.dioxide      0.827  <0.001      NO
## 7  Shapiro-Wilk total.sulfur.dioxide     0.851  <0.001      NO
## 8  Shapiro-Wilk       density            0.989  0.7483      YES
## 9  Shapiro-Wilk          pH              0.979  0.1927      YES
## 10 Shapiro-Wilk      sulphates           0.710  <0.001      NO
## 11 Shapiro-Wilk       alcohol            0.942  0.0012      NO
##
## $Descriptives
##                        n    Mean  Std.Dev Median   Min    Max   25th   75th
## fixed.acidity         81  8.0259  1.77410  7.600 4.600  12.60  6.900  8.800
## volatile.acidity      81  0.6573  0.26069  0.620 0.230   1.58  0.460  0.845
## citric.acid           81  0.2220  0.22377  0.150 0.000   1.00  0.030  0.390
## residual.sugar        81  2.6611  1.63130  2.100 1.200  12.90  1.800  2.800
## chlorides             81  0.0897  0.06733  0.078 0.044   0.61  0.067  0.088
## free.sulfur.dioxide   81 12.3333  9.51578  9.000 3.000  42.00  5.000 16.000
## total.sulfur.dioxide  81 34.2222 26.03027 24.000 7.000 119.00 14.000 48.000
## density               81  0.9964  0.00193  0.996 0.991   1.00  0.995  0.998
## pH                    81  3.3581  0.18630  3.350 2.740   3.90  3.250  3.480
## sulphates             81  0.6312  0.21730  0.570 0.330   2.00  0.520  0.690
## alcohol               81 10.6333  1.26068 10.400 8.400  14.00  9.700 11.300
##                         Skew Kurtosis
## fixed.acidity          0.653  -0.0469
## volatile.acidity       0.683   0.4055
## citric.acid            0.910   0.2465
## residual.sugar         3.528  17.3084
## chlorides              5.985  41.7931
## free.sulfur.dioxide    1.362   1.1270
## total.sulfur.dioxide   1.232   0.8894
## density               -0.195   0.3459
## pH                    -0.187   1.2407
## sulphates              3.406  17.5144
## alcohol                0.723  -0.0838
```

```
nrow(wine)
```

```
## [1] 81
```

While most of the features in the Shapiro-Wilk test reject the null, there is no fear since we have the

T Tests

```
t_tests <- ttest_all_vars(wine, "quality") %>%
            mutate(abs_diff = abs(observed.diff),
                   increase = observed.diff < 0,
                   significant = p.value < 0.05)
```

Should Increase

```
t_tests %>%
  filter(significant & increase) %>%
  pull(Chemical.Property)
```

```
## [1] "citric.acid" "sulphates"   "alcohol"
```

Should Decrease

```
t_tests %>%
  filter(significant & !increase) %>%
  pull(Chemical.Property)
```

```
## [1] "volatile.acidity" "chlorides"        "density"
## [4] "pH"
```

Insignificant

```
t_tests %>%
  filter(!significant) %>%
  pull(Chemical.Property)
```

```
## [1] "fixed.acidity"        "residual.sugar"       "free.sulfur.dioxide"
## [4] "total.sulfur.dioxide"
```