Wine Quality Analysis

DHANUSHKA SREE CB.SC.15DAS19036

Team no: 18

Introduction

Problem statement:

- predict the quality of wine on the basis of giving features.

This dataset has the fundamental features which may affecting the quality of the wine.

- By the use of few Machine learning models, we will predict the quality of the wine.
- Outlier detection algorithms could be used to detect the few excellent or poor wines.

Input variables :

- fixed acidity
- volatile acidity
- citric acid
- residual sugar
- Chlorides
- free sulfur dioxide
- total sulfur dioxide
- Density
- pH
- Sulphate
- alcohol

Output variable : - quality (score between 0 and 10)

Attribute Information :

Sample of the data set:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.70	0.00	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5.0
1	7.8	0.88	0.00	2.6	0.098	25	67	0.9968	3.20	0.68	9.8	5.0
2	7.8	0.76	0.04	2.3	0.092	15	54	0.9970	3.26	0.65	9.8	5.0
3	11.2	0.28	0.56	1.9	0.075	17	60	0.9980	3.16	0.58	9.8	6.0
4	7.4	0.70	0.00	1.9	0.076	11	34	0.9978	3.51	0.56	9.4	5.0

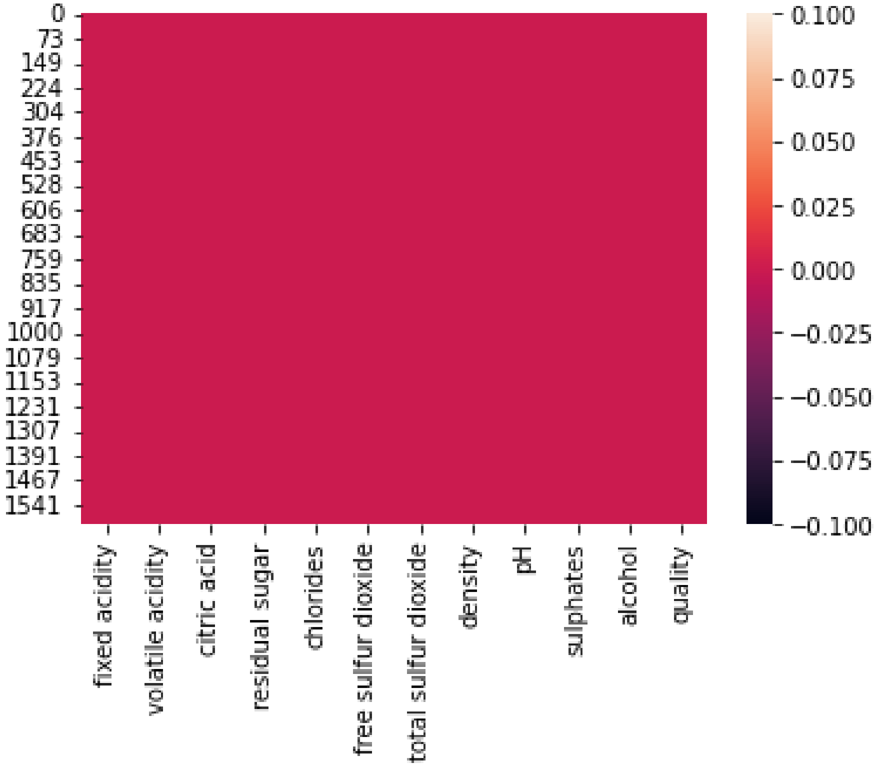
Implementation

Removing Null Values

graphical representation of null values using seaborn

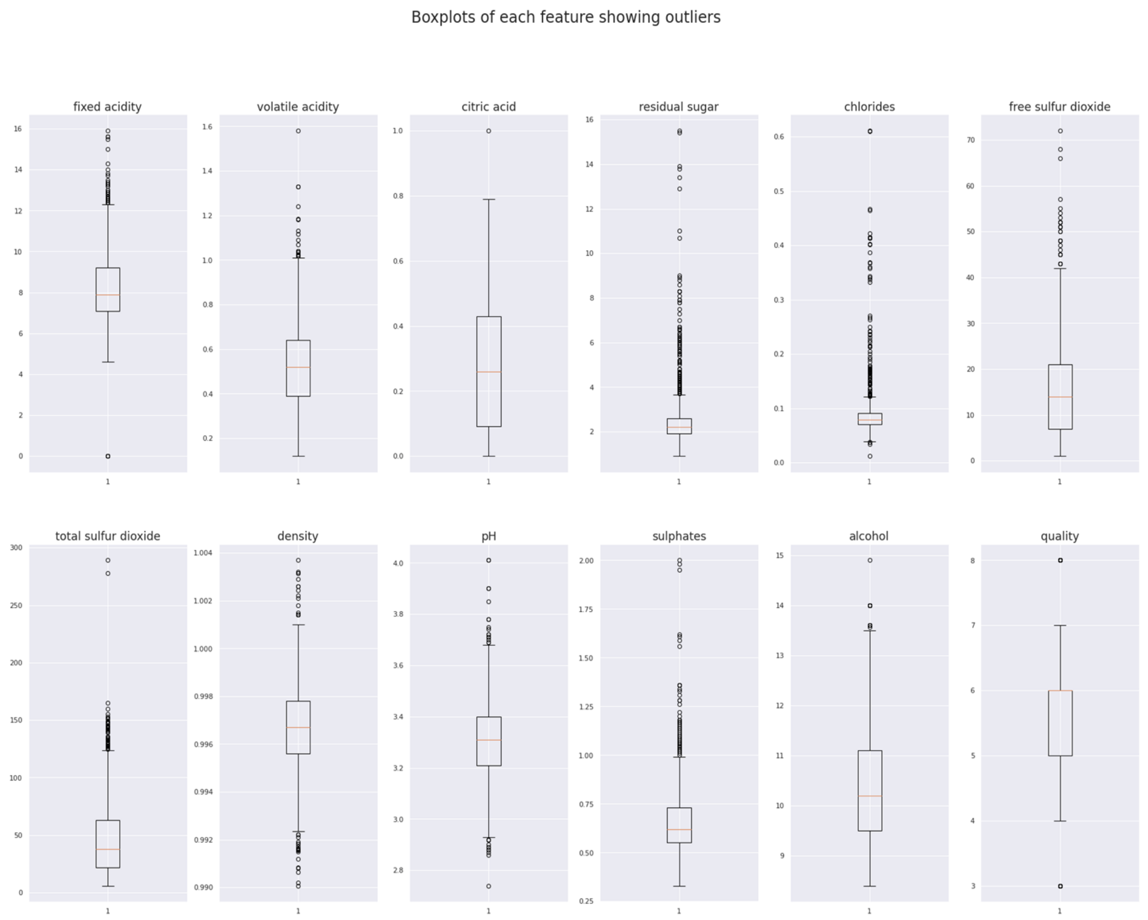
From this heap map , we can infer that there is no null values

```
#heatmap--- graphical representation of data that uses a system of color-coding to represent different values
```



Outlier treatment

The boxplots for scaled data to clearly see the outliers:



Fixed Acidity is mostly around 5 – 17.5

Volatile Acidity is mostly around 0.25 – 0.7

Citric Acid is mostly around 0.0 – 0.6

Residual sugar is high at 2.5

Chlorides is mostly around 0.0 – 0.1

Free sulfur dioxide is mostly around 0 – 40

Total sulfur dioxide is mostly around 0 -150

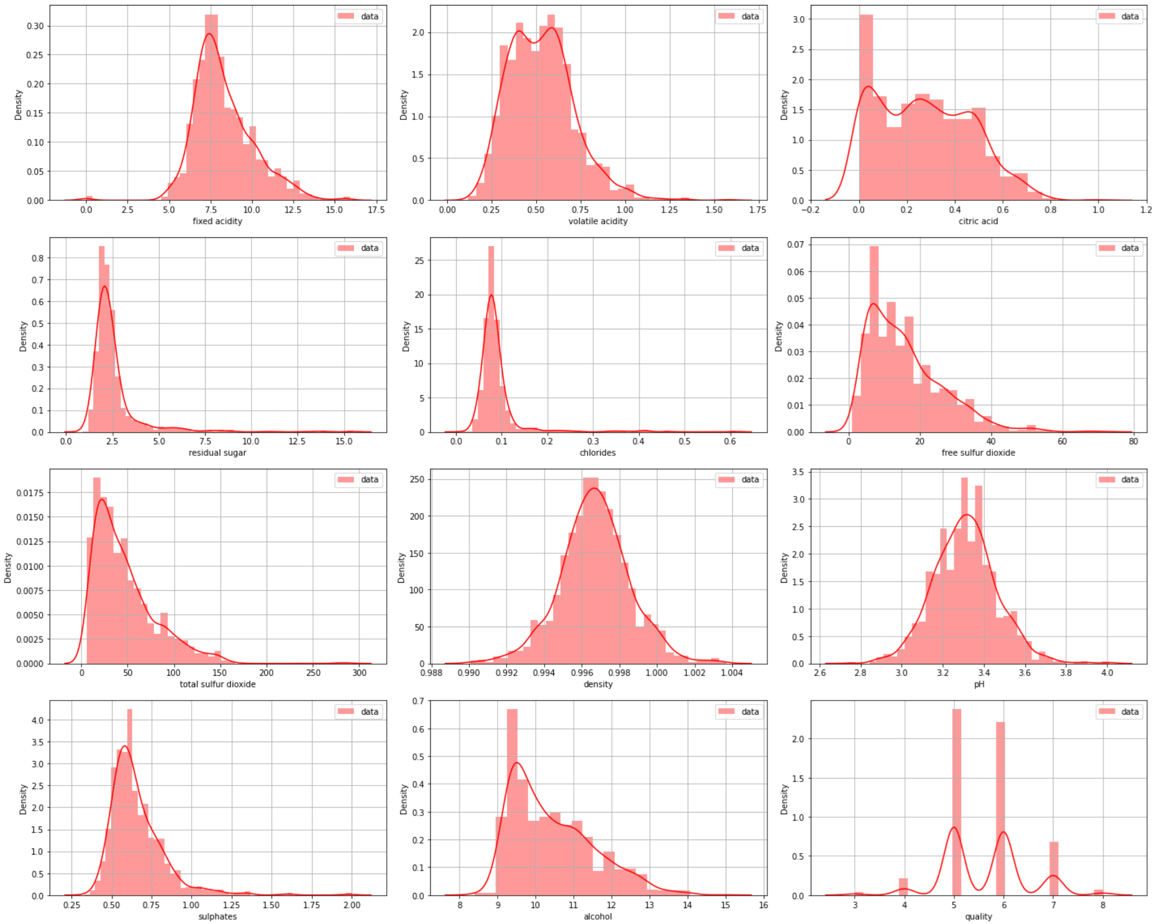
Density is high around 0.994 – 1.000

pH is highly around 3.2 – 3.6

Sulphates is around 0.5 – 0.75

Alcohol is around 9 – 13

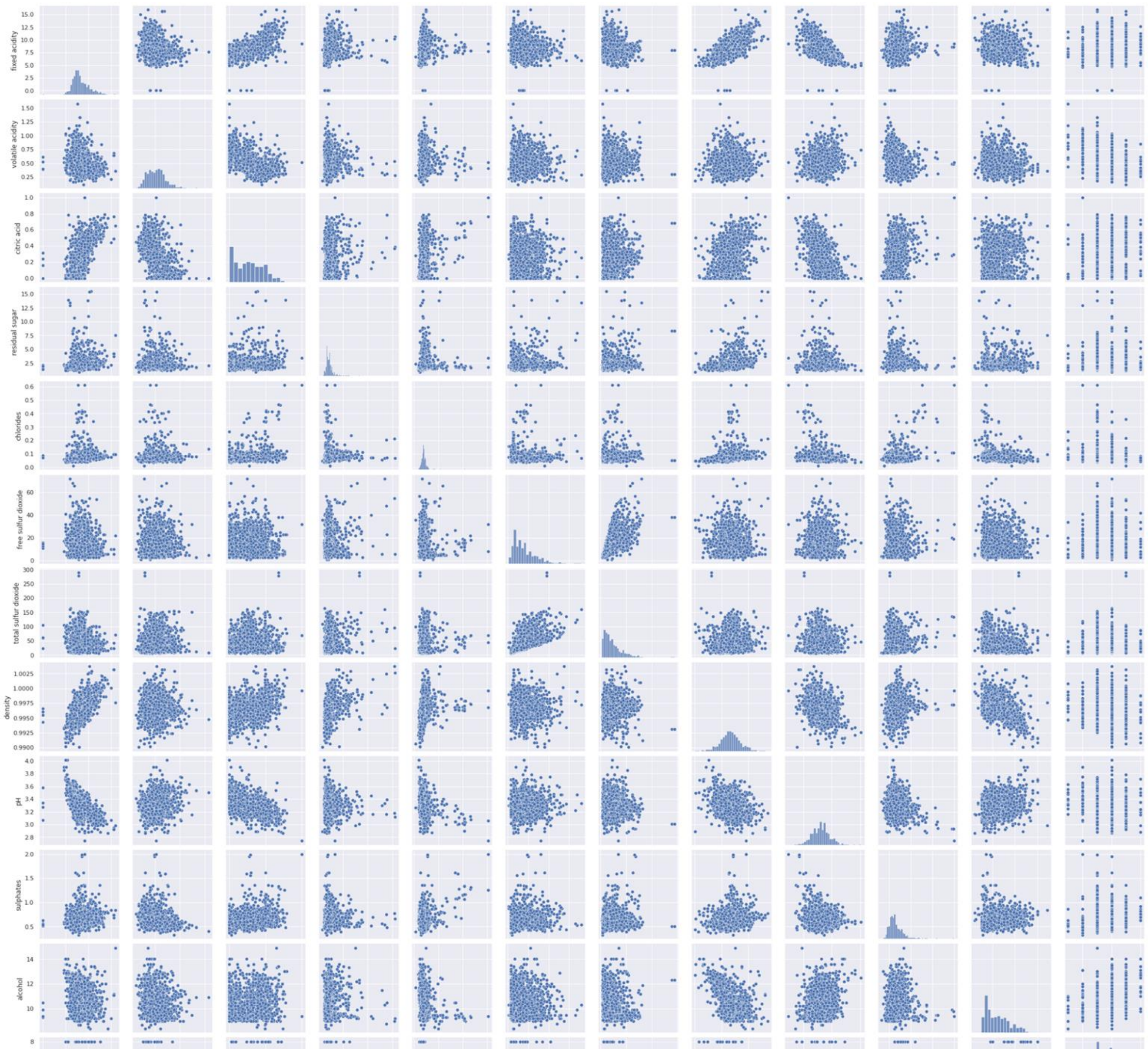
Quality is around 4,5,6,7



Histogram with line on it….distplot

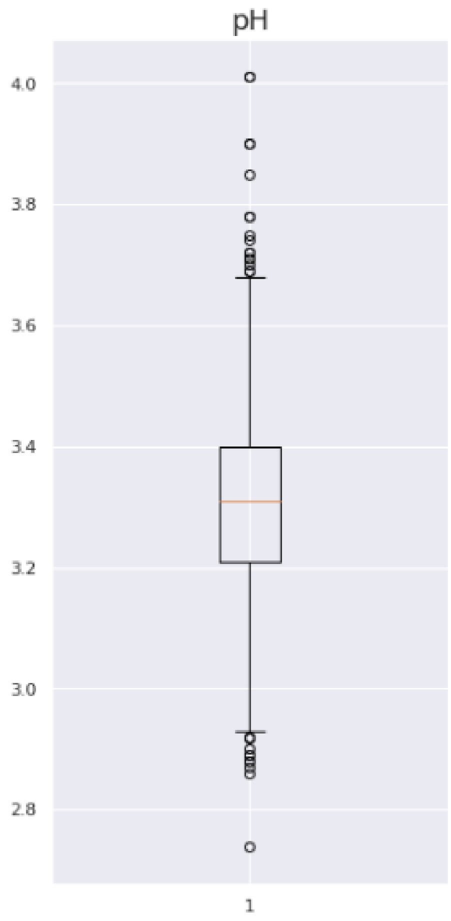
Wine Pairplot

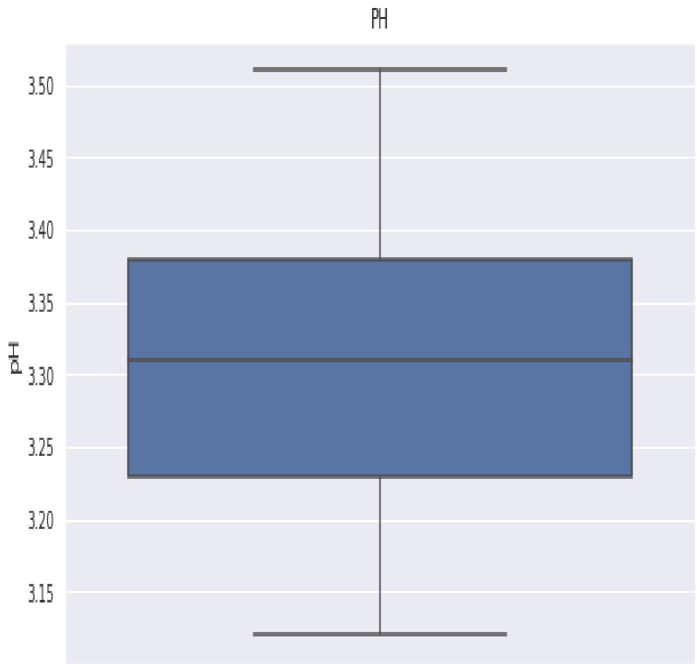
This pairplot shows the relationship among various features for all the possible pair .



Removing Outliers

After removing the outliers, we can see that the graph has no outliers.



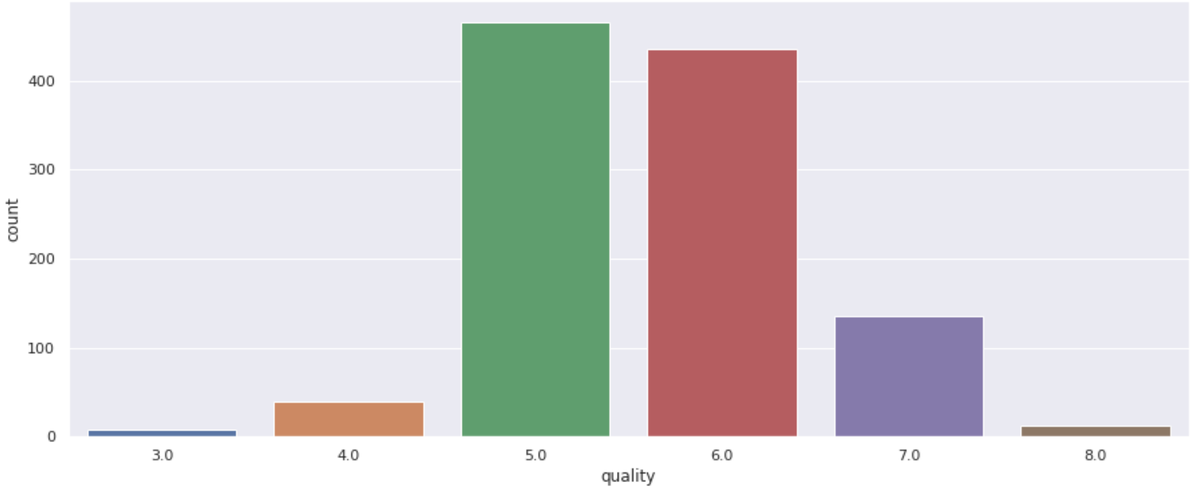


Variate Analysis

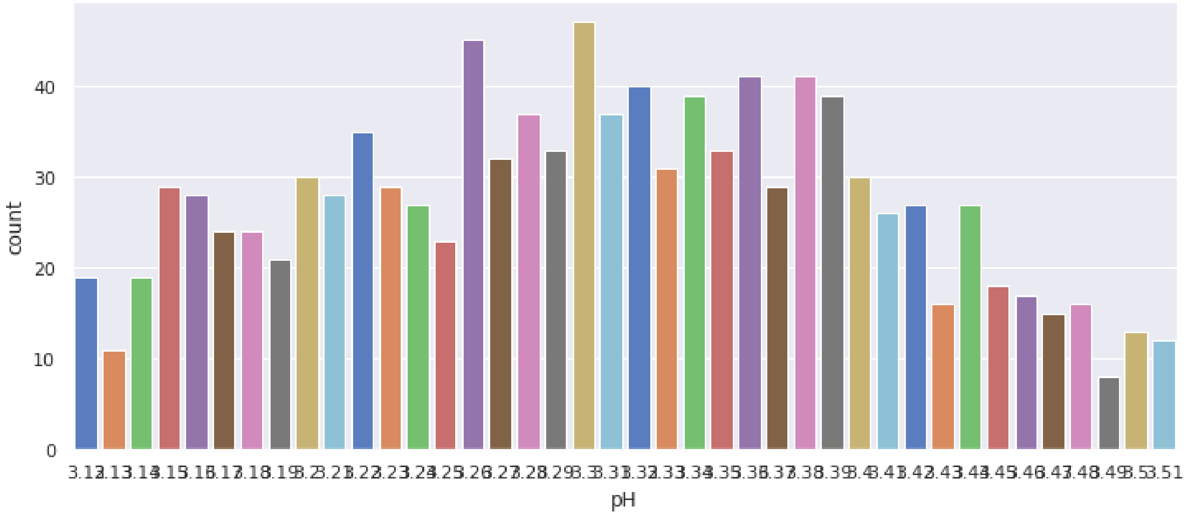
- Univariate
- Bivariate
- Multi-variate

UNIVARIATE ANALYSIS

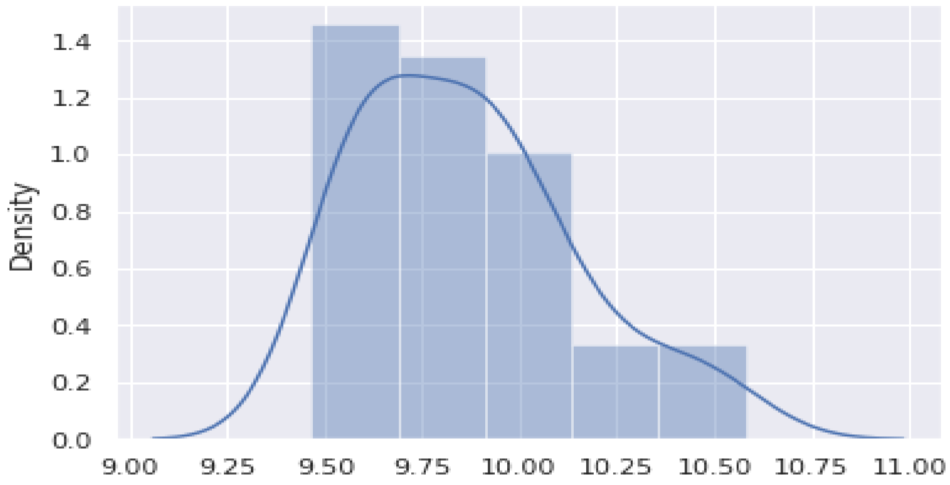
The Majority of the alcohol has a quality of 5 and 6.



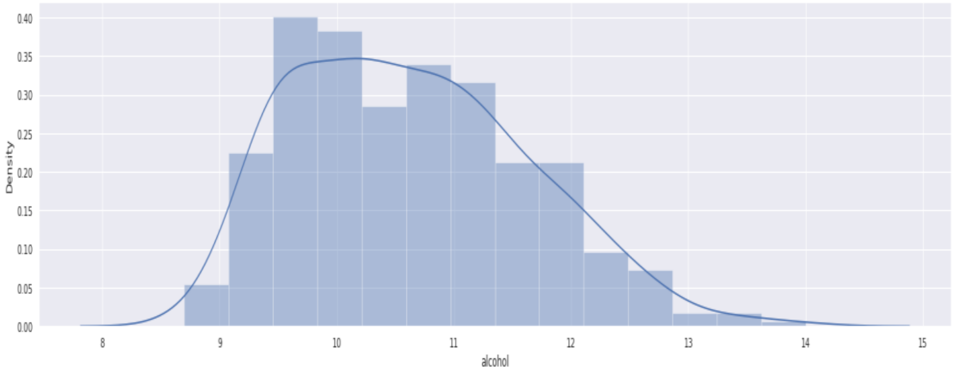
Majority of alcohols has a pH of 28 and 33, while the rest are almost equally distributed.

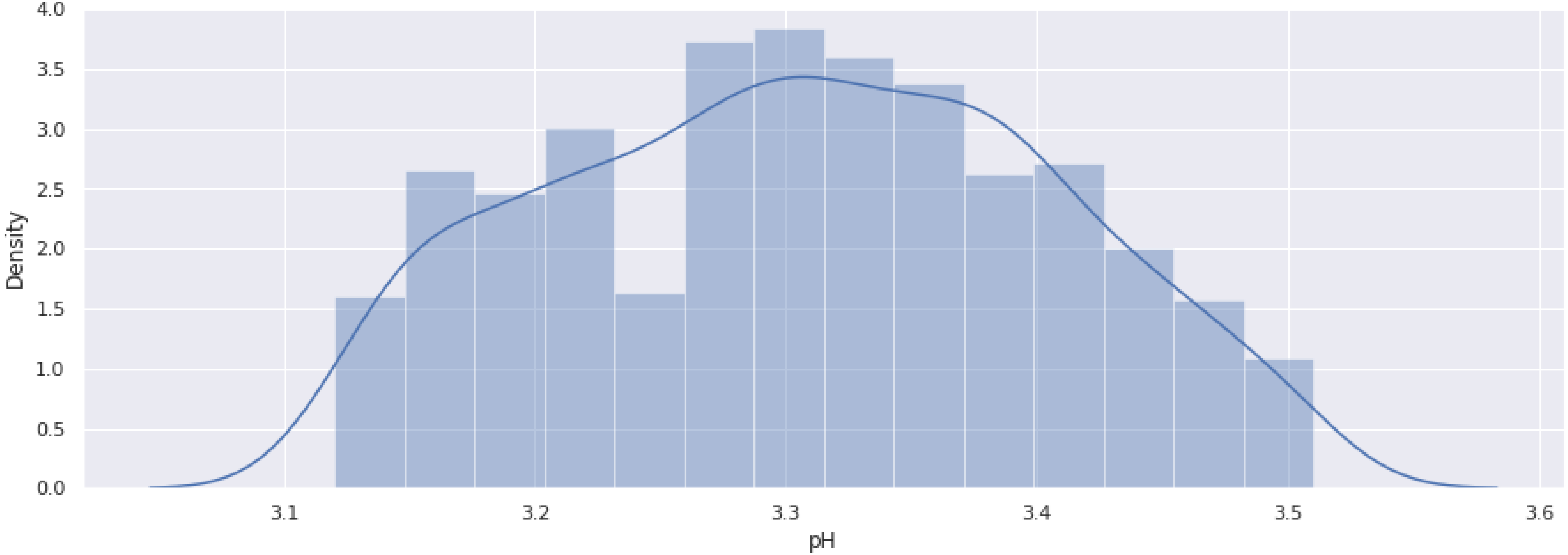


From this we can infer that the alcohol content is from 9.25 to 10.75 for density 5.



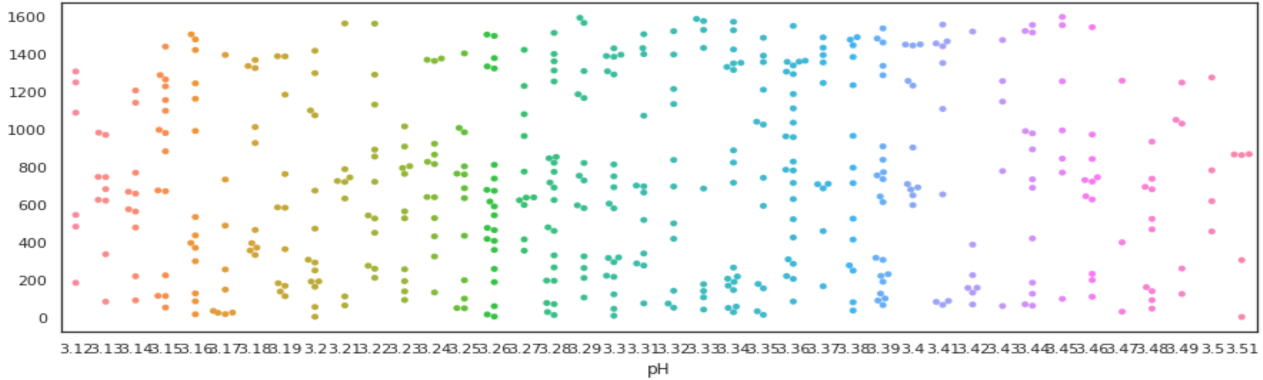
From this we can infer that the alcohol content is from 9 to 14 for density 6



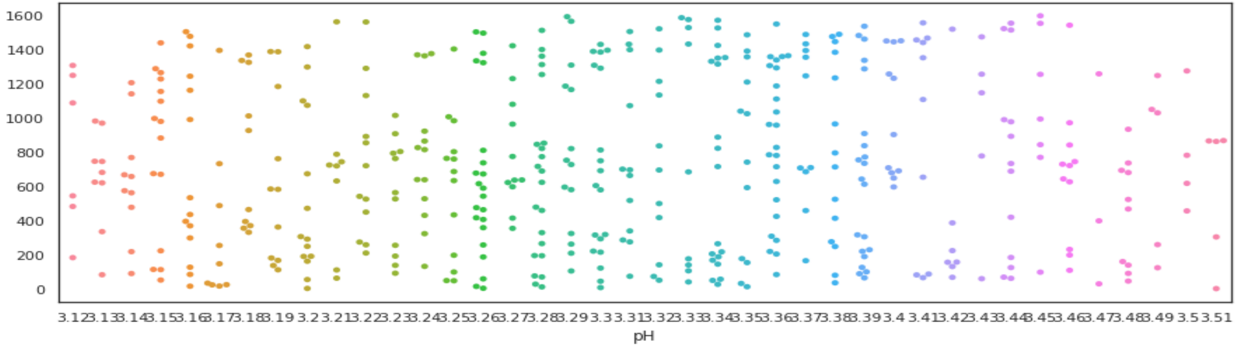


From this we can infer that the pH content is from 3.1 to 3.5

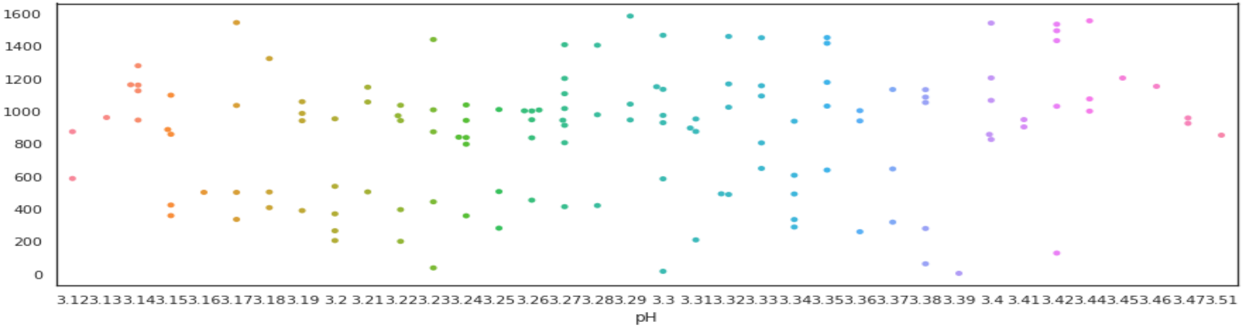
Quality = 5.0



Quality = 6.0



Quality = 7.0



Bivariate Analysis

Graph of pH and Alcohol, with hue as quality.

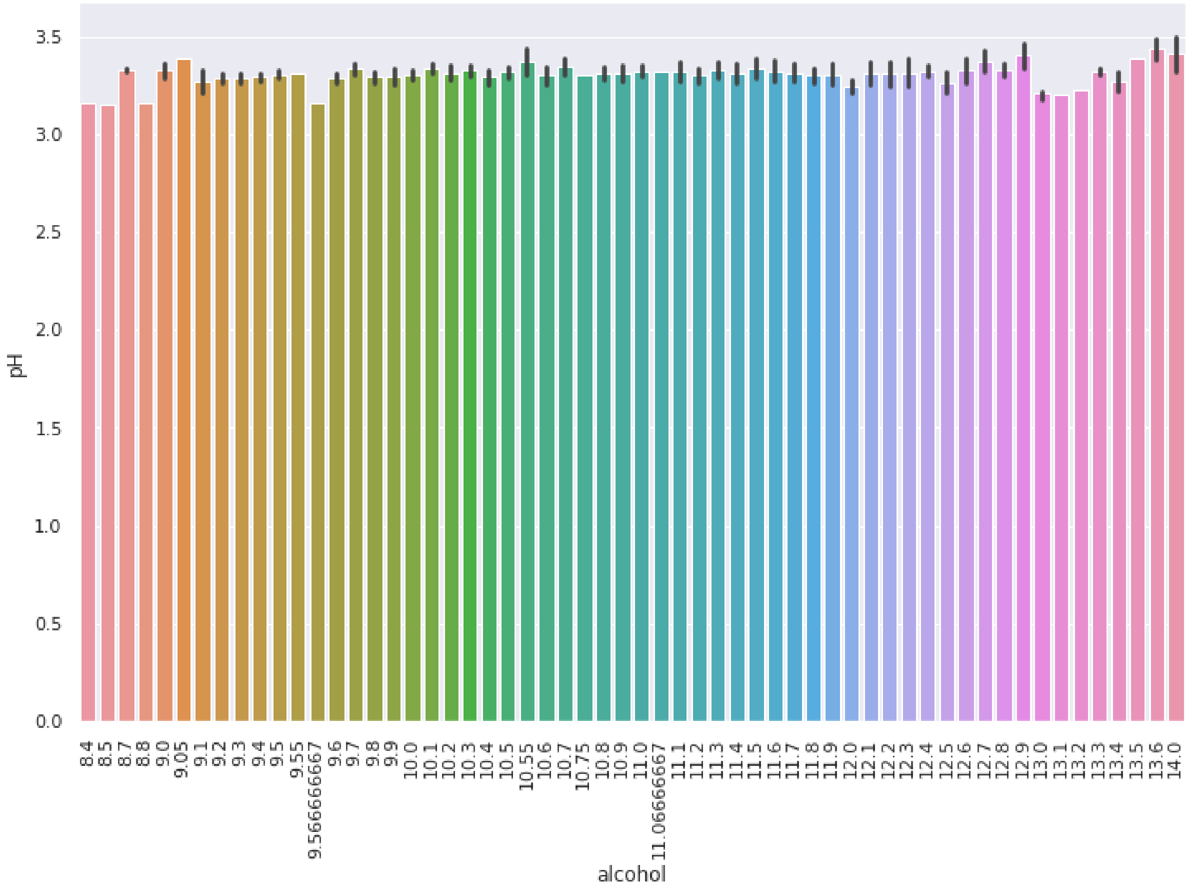
Doesn’t form any clusters or trends.

Graph of pH and Alcohol, with hue as density.

From this graph we can infer the pH level for each alcohol level.

We can see that all the alcohols levels have almost the same ph.

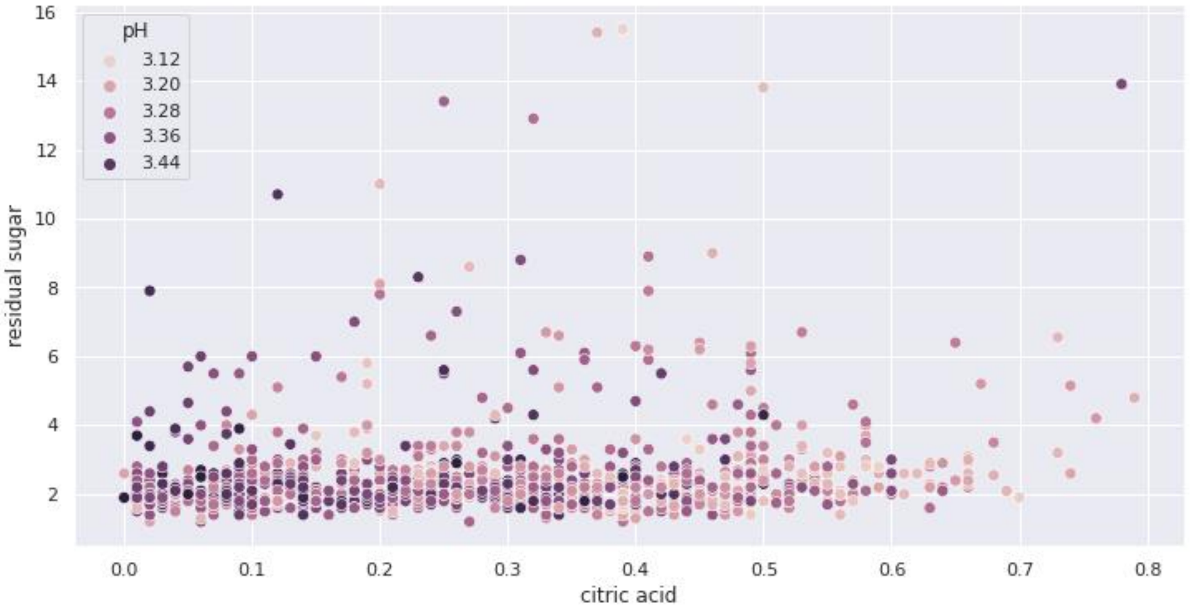
Almost uniformly distributed.



Multivariate analysis

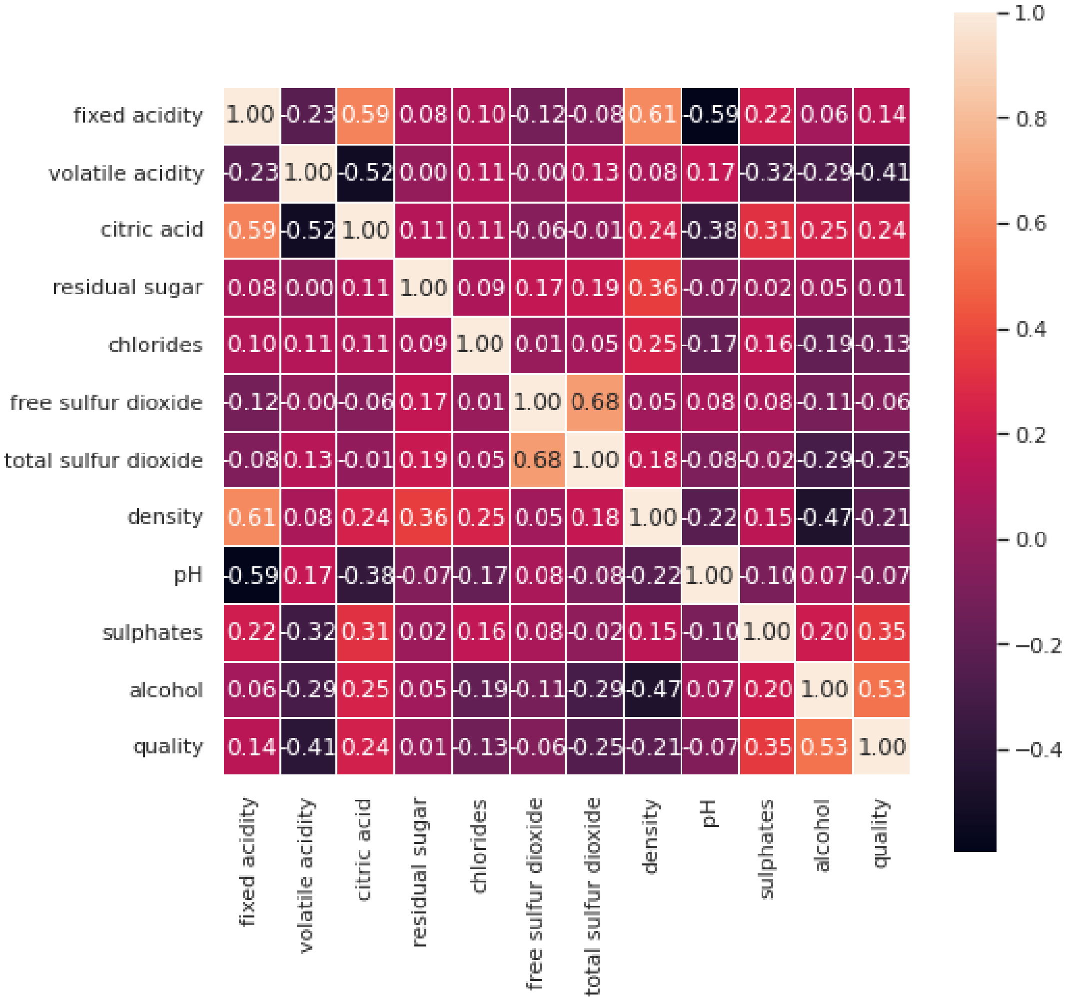
We can infer the, residual sugar level for citric acid.

We can see the residual sugar and citric acid combinedly has pH of mostly 3.44,3.36, 3.28



Low positive correlation between total sulfur dioxide and free sulfur dioxide, vise versa.

Majority doesn’t have correlation.



Machine learning algorithm:

Select column no.
Choose pH (to be predicted) as target variable .

Regression Analysis

```
[ ]
```

```
#Defining the independent variables and dependent variables
x = a.iloc[:,[0,1,3,4,5]]
y = a['pH']
#Getting Test and Training Set
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=.1,random_state=353)
x_train.head()
y_train.head()
```

```
32      3.17
237     3.37
267     3.35
1280    3.37
1474    3.16
Name: pH, dtype: float64
```

```
[ ] x_train.shape
```

```
(986, 5)
```

Linear regression :

linear regresion model

```
[ ]  #Prepare a Linear Regression Model
     reg=LinearRegression()
     reg.fit(x_train,y_train)
     y_pred=reg.predict(x_test)
     from sklearn.metrics import r2_score
     r2_score(y_test,y_pred)

     0.2500528699043001
```

Linear Regression is a supervised machine learning algorithm where the predicted output is continuous and has a constant slope.

1. Get data
2. Fit the line(using reg.fit.)
3. In next test value,when we give x, ,beta0 and beta1 get changed and we get y hat using y_pred
4. Using r-squared we find the accuracy .

Decision tree:

decision tree

```
#Prepairng a Decision Tree Regression
from sklearn.tree import DecisionTreeRegressor
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=.1,random_state=105)
DTree=DecisionTreeRegressor(min_samples_leaf=.01)
DTree.fit(x_train,y_train)
y_predict=DTree.predict(x_test)
from sklearn.metrics import r2_score
r2_score(y_test,y_predict)
```

```
0.32572803603714806
```

Performance metrics:

performance metrices

```
[ ] ls=[0.2500528699043001,0.3257280360371475]
    ls2=["Linear regression","Decision Tree"]
    df=pd.DataFrame(list(zip(ls2,ls)),columns=["Algorithms","Performance Values"])
    df
```

	Algorithms	Performance Values
0	Linear regression	0.250053
1	Decision Tree	0.325728

```
df.plot.scatter(x="Algorithms",y="Performance Values",s=200,c="orange",ec="black")
plt.show()
```

Inference for the machine learning algorithm:

Since the R-squared value of linear regression and decision tree are 0.2500528699043001 and 0.32572803603714806 respectively.

Comparing the R-squared value helps us to find out the more efficient algorithm

Hence , we can infer that decision tree regression algorithm provides better accuracy for our data set than linear regression.

Form the above performance value can infer that , the performance value 0.325728 of decision tree is more accurate than linear regression(0.250053).

From the above , we understand that the data is underfit. It becomes underfit when it did not include even the slightest variation within the data set.