For this project I have chosen a dataset from Kaggle about the quality of water. Specifically, the data collected is about the pH level, hardness, solids, chloramines, sulfates, conductivity, organic carbon, trihalomethanes, and turbidity and their effects on whether the water is safe to consume based on varying levels of each of the factors. I will use machine learning algorithms on this dataset to see if they can be trained to classify if water is safe to drink or not based on the factors listed above.

To start the project, I imported all the necessary libraries into the notebook. Next, I uploaded the dataset into the notebook. I ran the shuffle function in after loading the data to mix up all the 0's and 1's in the Potability column, so that they are not all clumped together with 0's all in the beginning and 1's in the end. Next, I took a look of the dataset with the head function. Then, I took a look at the rows and columns of the dataset with the shape function and found that the dataset has 3276 rows and 10 columns. After that, I took a look at the datatypes of each column to see if any one hot encoding is necessary because many machine learning models cannot use classifications as is and need to be transformed into integers. Finding that all the data is already in float and integer form, I took a look at the statistics of the dataset with the describe function. Then, I looked for any missing/null values in the dataset and found that there are 3 columns with missing data (pH, Sulfate, and Trihalomethanes.) I decided to not drop any of the columns because the missing data is less than 30% of the total population of the dataset which makes them usable. I ran a for loop to find all the null/missing values in the dataset and replace them with the mean value of the column. After cleaning this dataset, I went on to make visuals for the dataset.

For the visualizations I made 5 graphs. First, I made a bar plot of the Potability column to see how much of the data are safe to drink and how much of the data is not safe to drink. In this visual, I found that there is way less safe to drink water (1's) than unsafe water (0's). This may have led to one of my problems in the later part of this project. My second graph is a histogram of the Hardness column. Hardness of water is described as the calcium and magnesium salts that the water has com into contact with. I found that the spread here is normal with the center around the 200 level. My third graph is also a histogram of the Solids column. Solids are measured in Total Dissolved Solids (TDS). It is the measure of how much inorganic and organic minerals are dissolved in water. The range of 500 to 1000 ppm are safe to drink. The spread here relatively normal with a slight skew to the left and a small tail to the right. The fourth graph I made is a histogram of the Trihalomethanes column. Trihalomethanes (THM) is a chemical that is produced when water has been treated with chlorine. The level of THM can go up to 80 ppm to be considered safe to drink. The spread of this histogram is normal. My fifth graph is a heat correlation map of the dataset. This heatmap shows me that there is little to no correlation between the features and the label. This shows that there is no multicollinearity within this dataset.

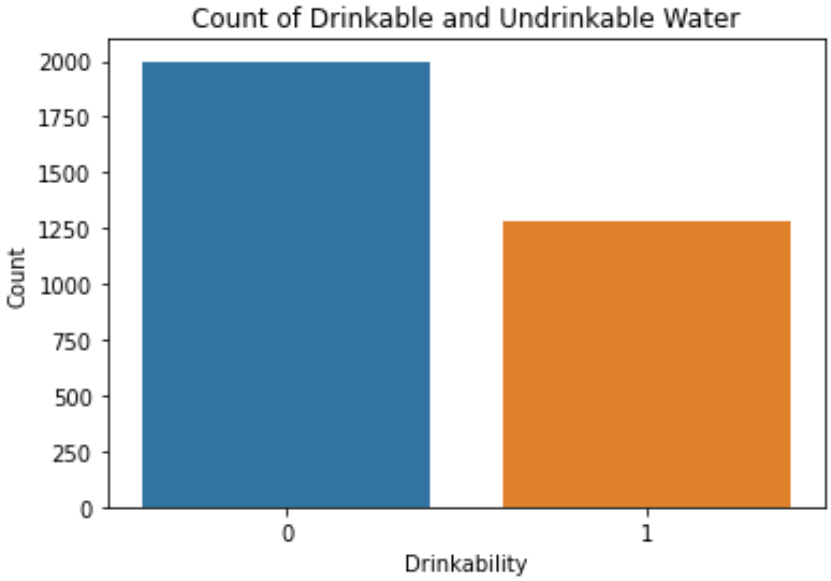
Next, I split the data into features (X) and labels (y) with the train, test, split function from sklearn. I made the X variable equal to the dataset's pH, hardness, solids, chloramines, sulfates, conductivity, organic carbon, trihalomethanes, and turbidity. The y variable is set to be the Potability column (0 meaning unsafe to drink and 1 meaning safe to drink). I standardized the columns to get rid of any skewedness that may be present in the features (X) variable

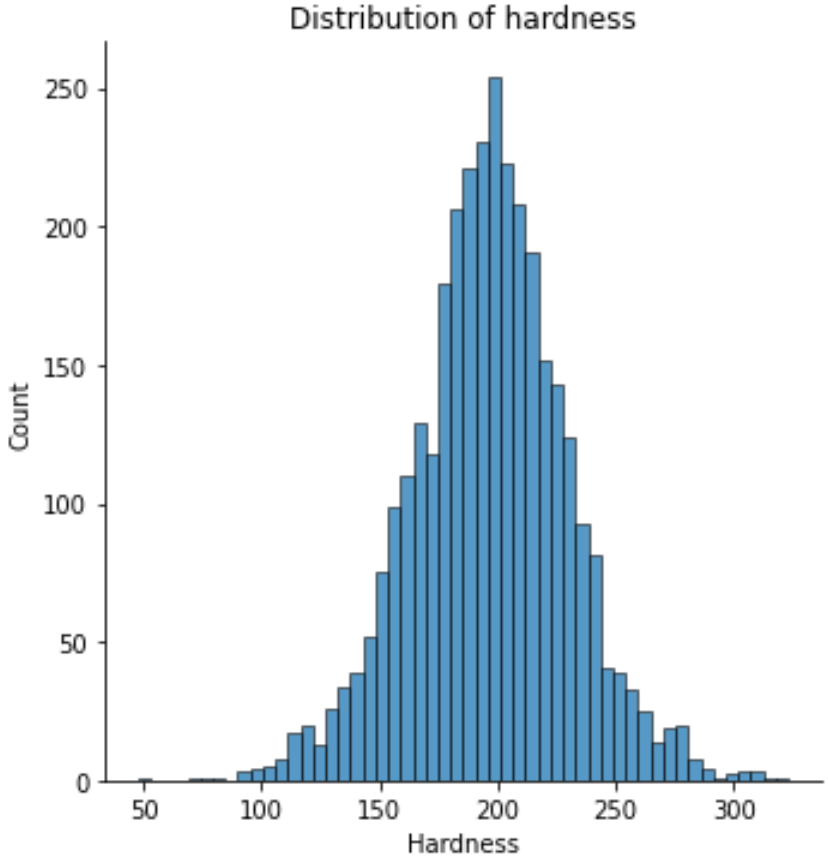
because larger numbers tend to skew the result when some columns are between 0 and 1. I start to train my data models by fitting my features and labels into them. I started with decision trees, followed by random forest, KNN, SVM (with kernels linear, RBF, polynomial, and sigmoid), Gaussian Naïve Bayes, and finally logistic regression. Before making predictions with each model, I ran a 5-fold cross validation in each model to see how well each model does on unseen data. After running the 5-fold cross validation, I make predictions with each model and generate classification reports with each model. It is here that I ran into a problem with the linear SVM model. The big difference between the 0's and 1's of the Potability column made it so that there are not enough 1's that the linear SVM model can use to run the model well. So, in the conclusion I have omitted this model and compared all the rest of the models.

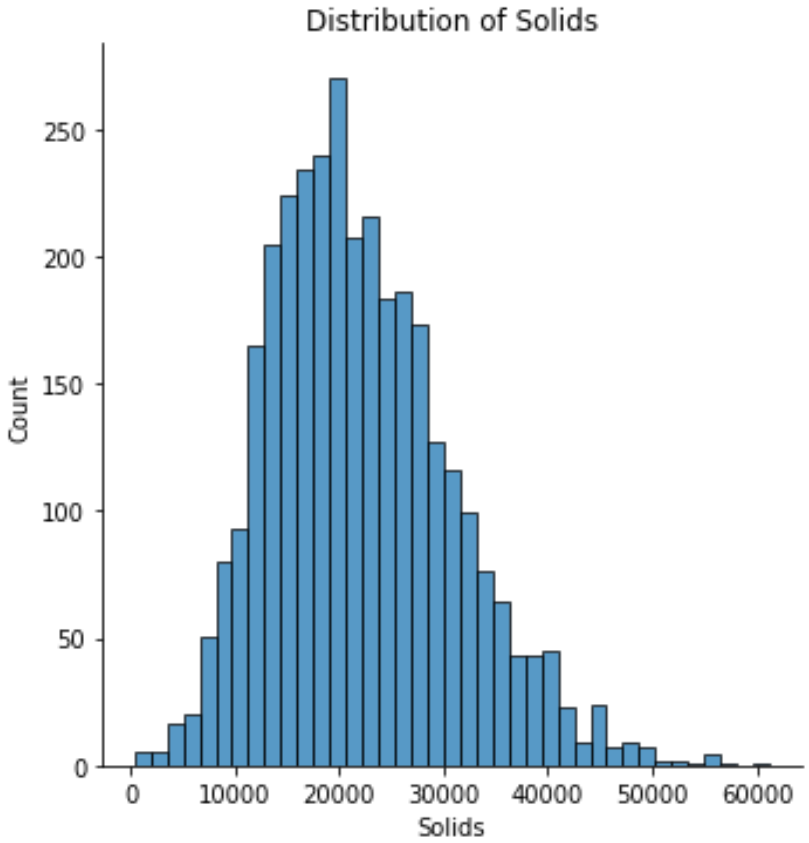
The results of all the models weren't as good as I would've have hoped. Let's talk about accuracy. There are 4 models that are tied with .63 accuracy and those are the KNN model, Gaussian Naïve Bayes model, logistic regression model, and the random forest model. In the precision category logistic is ranked the highest with a precision value of .70 followed by KNN and Gaussian Naïve Bayes models. For the recall value there is a 4-way tie again with .63 between the models KNN, Gaussian Naive Bayes, logistic regression, and random forest. As for F1 scores, KNN has the highest score at .61, followed by a tie between Gaussian Naïve Bayes model and the random forest model. When it comes the potability of water, I believe that lowering the type 2 errors (false negatives) as much as possible is the best because drinking water deemed to be safe when it isn't can actually lead to sickness. So, minimizing this would be for the best. This would make it that the models with the high accuracy and high recall would be good. Since KNN model, Gaussian Naïve Bayes model, logistic regression model, and the random forest model are all tied in these categories, the model with the highest f1 score will be the best model for this dataset. In this case, it will be the KNN model.

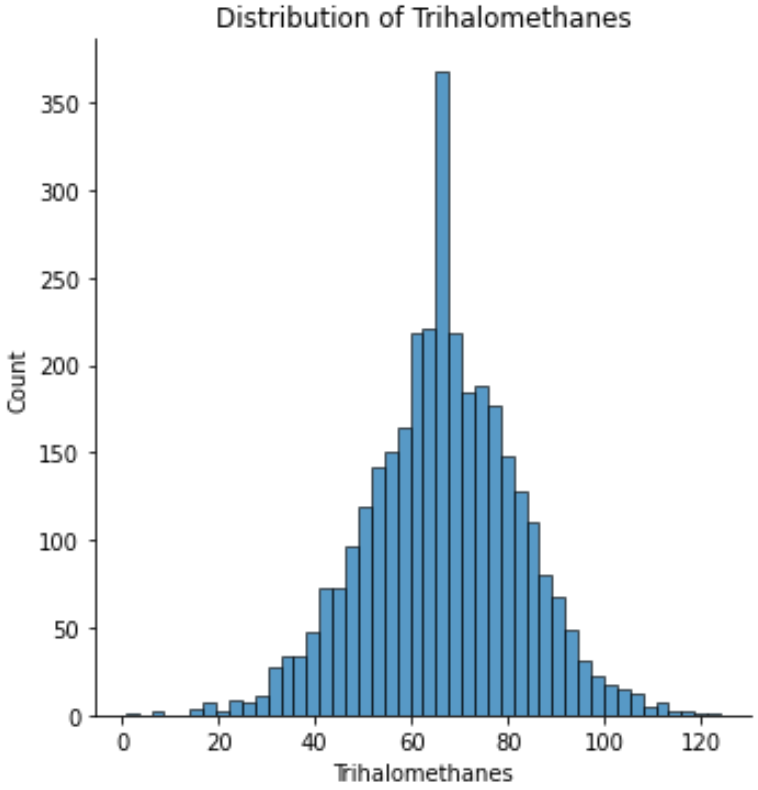
After finding all the classification reports I made one final graph that will show the performance of each model according to their accuracy, precision, recall, and f1 scores lined up against each other. First, I made a list of all the models to be considered without linear SVM. Then, I made a list for accuracy, precision, recall and f1 to match it up with their respective models. After that, I made a bar chart for each measure as well as adding titles, a legend, and also naming the axes.

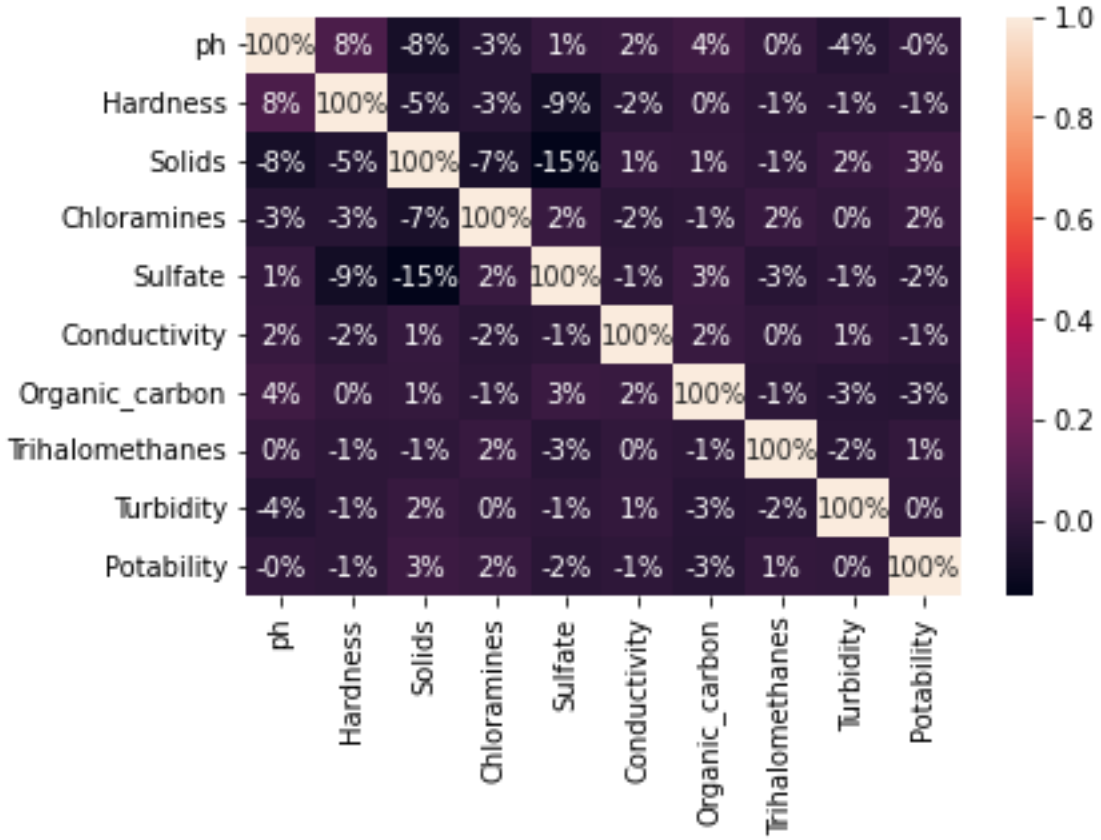
Appendix:

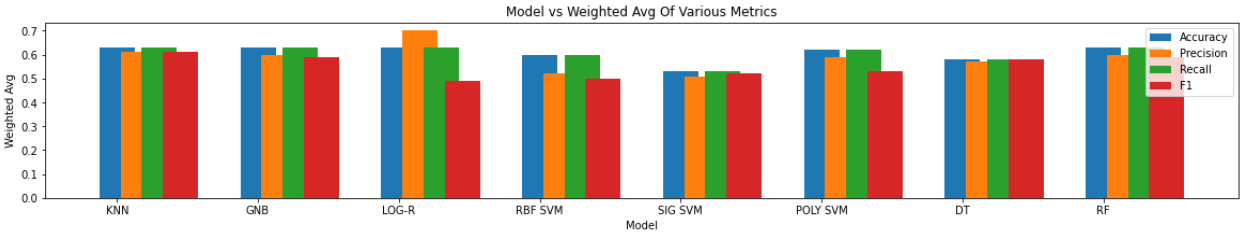

Count of Drinkable and Undrinkable Water
2000
1750
1500
1250
1000
750
500
250
0
Count
0
1
Drinkability


Distribution of hardness
250
200
150
100
50
0
Count
50
100
150
200
250
300
Hardness


Distribution of Solids
Count
Solids
0
50
100
150
200
250
0
10000
20000
30000
40000
50000
60000


Distribution of Trihalomethanes
Count
Trihalomethanes
0
50
100
150
200
250
300
350
0
20
40
60
80
100
120


ph 100% 8% -8% -3% 1% 2% 4% 0% -4% -0%
Hardness 8% 100% -5% -3% -9% -2% 0% -1% -1% -1%
Solids -8% -5% 100% -7% -15% 1% 1% -1% 2% 3%
Chloramines -3% -3% -7% 100% 2% -2% -1% 2% 0% 2%
Sulfate 1% -9% -15% 2% 100% -1% 3% -3% -1% -2%
Conductivity 2% -2% 1% -2% -1% 100% 2% 0% 1% -1%
Organic_carbon 4% 0% 1% -1% 3% 2% 100% -1% -3% -3%
Trihalomethanes 0% -1% -1% 2% -3% 0% -1% 100% -2% 1%
Turbidity -4% -1% 2% 0% -1% 1% -3% -2% 100% 0%
Potability -0% -1% 3% 2% -2% -1% -3% 1% 0% 100%
ph
Hardness
Solids
Chloramines
Sulfate
Conductivity
Organic_carbon
Trihalomethanes
Turbidity
Potability
1.0
0.8
0.6
0.4
0.2
0.0


Model vs Weighted Avg Of Various Metrics
Weighted Avg
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
KNN
GNB
LOG-R
RBF SVM
SIG SVM
POLY SVM
DT
RF
Model
Accuracy
Precision
Recall
F1