Analysis and Prediction of Water Potability

A Data Science Project by Dallas Collie

Did you know…?

- As of September 2021, there were 51 long-term drinking water advisories in effect in 33 First Nations communities in Canada.*
- 400 out of 618 First Nations in Canada have experienced boil water advisories or do-not-consume orders in recent years. **
- The exact percentage of First Nations people that do not have access to clean drinking water is not known, but it’s clear that a significant number do not have access.

* The Government of Canada's Indigenous Services Canada.
** A 2021 report by Human Rights Watch

Water Potability

- Water potability refers to the suitability of water for consumption by humans. Safety guidelines are put in place to ensure that potability of drinking water.
  - Microbial safety: free from harmful microorganisms (bacteria, viruses, and parasites)
  - Chemical safety: from harmful chemical contaminants (heavy metals, pesticides, and industrial chemicals).
  - Physical safety: free from physical contaminants (sediment, particulate matter, and other visible materials).
  - Radiological safety: free from radiological contaminants (uranium, radon and radioactive materials used in nuclear power plants).
  - Taste and odor: free from unpleasant taste and odor that may make it less desirable for consumption.

Data - Water Potability

- Acquired blind data: no locations of samples taken and when.
- Potability calculation: unknown.

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	7.080795	219.674262	22210.613083	5.875041	333.775777	398.517703	11.502316	112.412210	2.994259	0
1	6.783888	193.653581	13677.106441	5.171454	323.728663	477.854687	15.056064	66.396293	3.250022	0
2	6.010618	184.558582	15940.573271	8.165222	421.486089	314.529813	20.314617	83.707938	4.867287	1
3	8.097454	218.992436	18112.284447	6.196947	333.775777	376.569803	17.746264	59.909941	4.279082	1
4	8.072612	210.269780	16843.363927	8.793459	359.516169	559.167574	17.263576	68.738989	5.082206	0
...	...	...	...	...	...	...	...	...	...	...
4299	8.989900	215.047358	15921.412018	6.297312	312.931022	390.410231	9.899115	55.069304	4.613843	1
4300	6.702547	207.321086	17246.920347	7.708117	304.510230	329.266002	16.217303	28.878601	3.442983	1
4301	11.491011	94.812545	37188.826022	9.263166	258.930600	439.893618	16.172755	41.558501	4.369264	1
4302	6.069616	186.659040	26138.780191	7.747547	345.700257	415.886955	12.067620	60.419921	3.669712	1
4303	4.668102	193.681735	47580.991603	7.166639	359.948574	526.424171	13.894419	66.687695	4.435821	1

Data - Potability Criteria

- Potability: 1 (potable) or 0 (not potable)
- Potability Criteria:
  - pH - The acidity or alkalinity of a solution. Safe limits: 6.5 to 8.5
  - Hardness – Concentration of dissolved minerals, primarily calcium and magnesium. Max 200 mg/L.
  - Solids – Concentration of dissolved minerals such as calcium and magnesium, as well as other dissolved substances like salts and organic compounds. Max 500 mg/L
  - Chloramines – A type of disinfectant can react with organic matter in the water, forming potentially harmful by-products like trihalomethanes (THMs). Max 4 mg/L .
  - Sulfates - Naturally occurring mineral can have a laxative effect. Max 250 mg/L.
  - Conductivity - Ability of water to conduct an electrical current indicating presence of minerals, metals and organic compounds. Max 500 µS/cm.
  - Organic Carbon - Amount of carbon-based compounds, can react with chlorine to form by-products such as THMs. Max 10 mg/L.
  - Trihalomethanes (THM) - disinfection by-products that can form in drinking water when chlorine or other disinfectants react with organic matter, linked to an increased risk of cancer. Max 100 µg/L.
  - Turbidity – Cloudiness or haziness of water making it difficult to disinfect water effectively, particles can shield bacteria and other microorganisms. Max 1.0 nephelometric turbidity units.

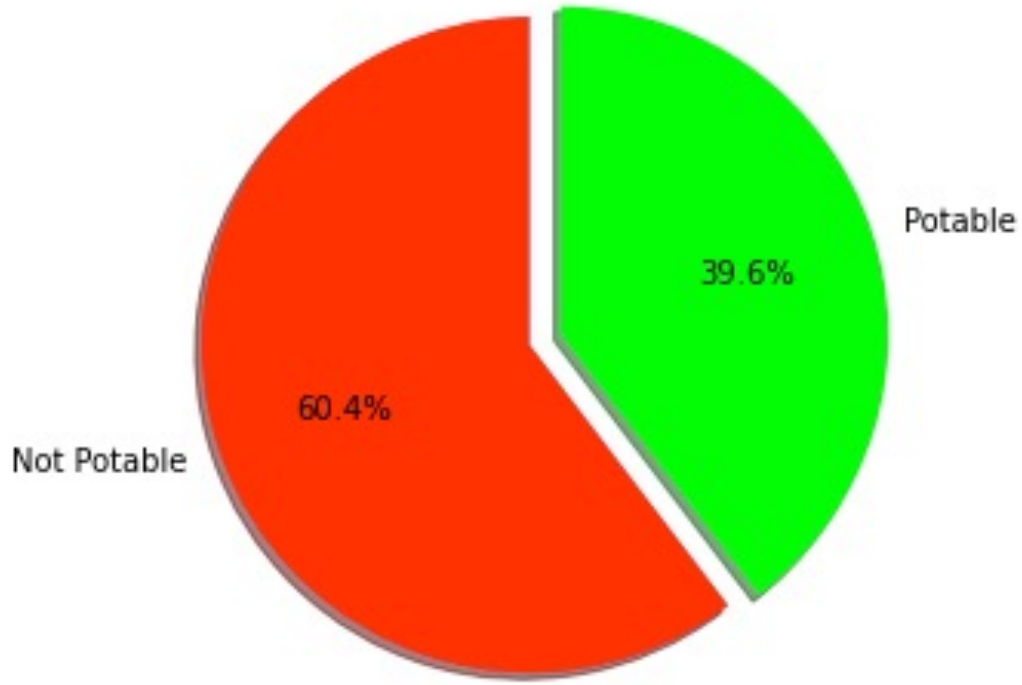
Data Prep and Overview

- Create a Data Frame of from csv file
- Remove all entries with null values and save as new Data Frame

	count	mean	std	min	25%	50%	75%	max
ph	4304.0	7.084005	1.527266	0.000000	6.158179	7.080795	7.958871	14.000000
Hardness	4304.0	196.193037	32.539158	47.432000	176.749219	197.028926	216.444406	317.338124
Solids	4304.0	22001.028235	8655.692108	320.942611	15751.175300	21051.311141	27296.294045	56488.672413
Chloramines	4304.0	7.133619	1.578817	0.530351	6.139464	7.138343	8.107067	13.127000
Sulfate	4304.0	333.314535	38.573181	129.000000	312.015798	333.775777	354.510553	481.030642
Conductivity	4304.0	425.955979	80.467727	201.619737	365.091587	421.926811	481.737559	753.342620
Organic_carbon	4304.0	14.289231	3.333973	2.200000	12.066127	14.221757	16.570558	28.300000
Trihalomethanes	4304.0	66.453421	16.070327	0.738000	56.104252	66.396293	77.215957	124.000000
Turbidity	4304.0	3.968357	0.784600	1.450000	3.442121	3.958396	4.512068	6.739000
Potability	4304.0	0.396375	0.489201	0.000000	0.000000	0.000000	1.000000	1.000000

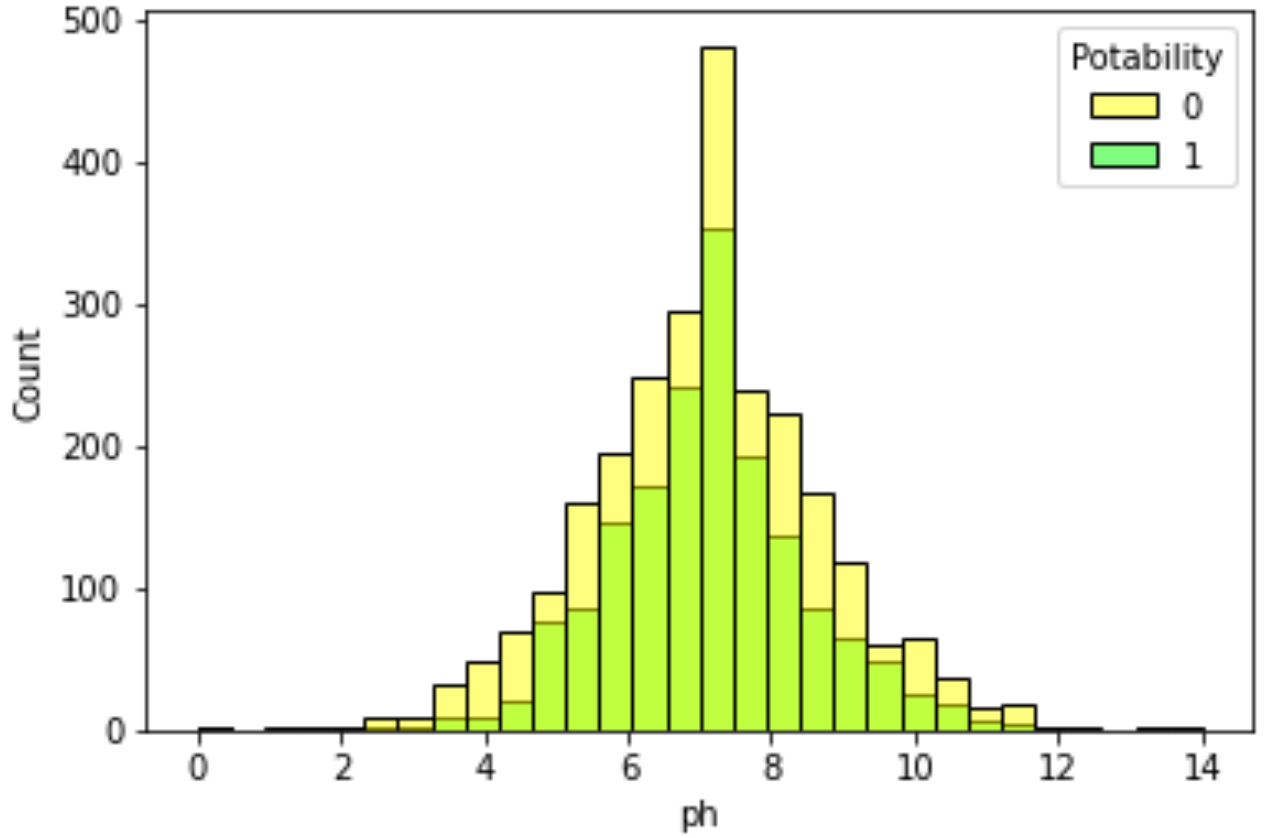
Data Analysis - Potability Count

```
0    2598
1    1706
Name: Potability, dtype: int64
```



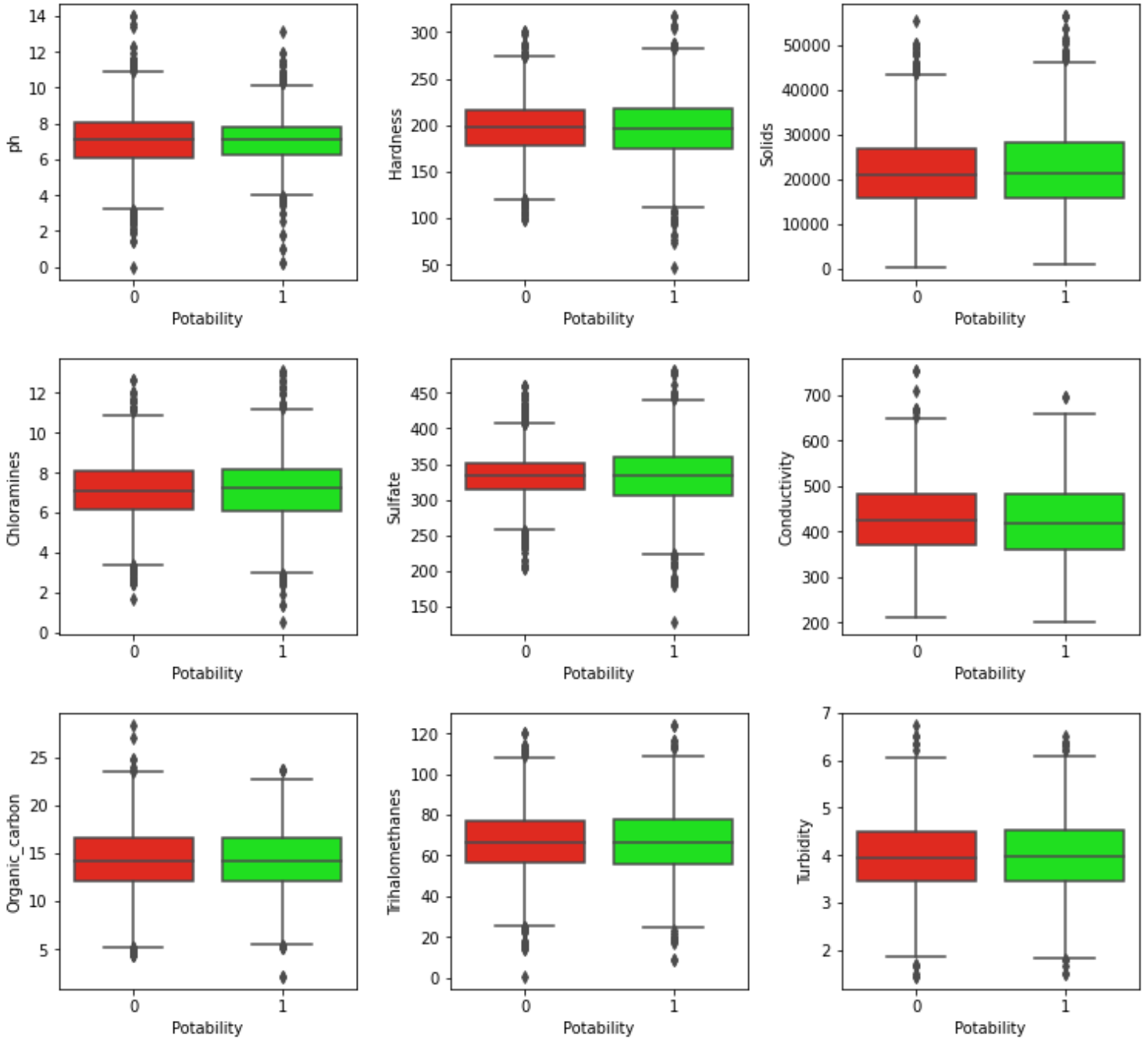
Data Analysis – Anomalous Data

- pH min and max show impossible measurements. Will this effect the data analysis?



Data Analysis

- Potability data ranges
- Water deemed potable has some data that is more extreme than the water that was not deemed potable



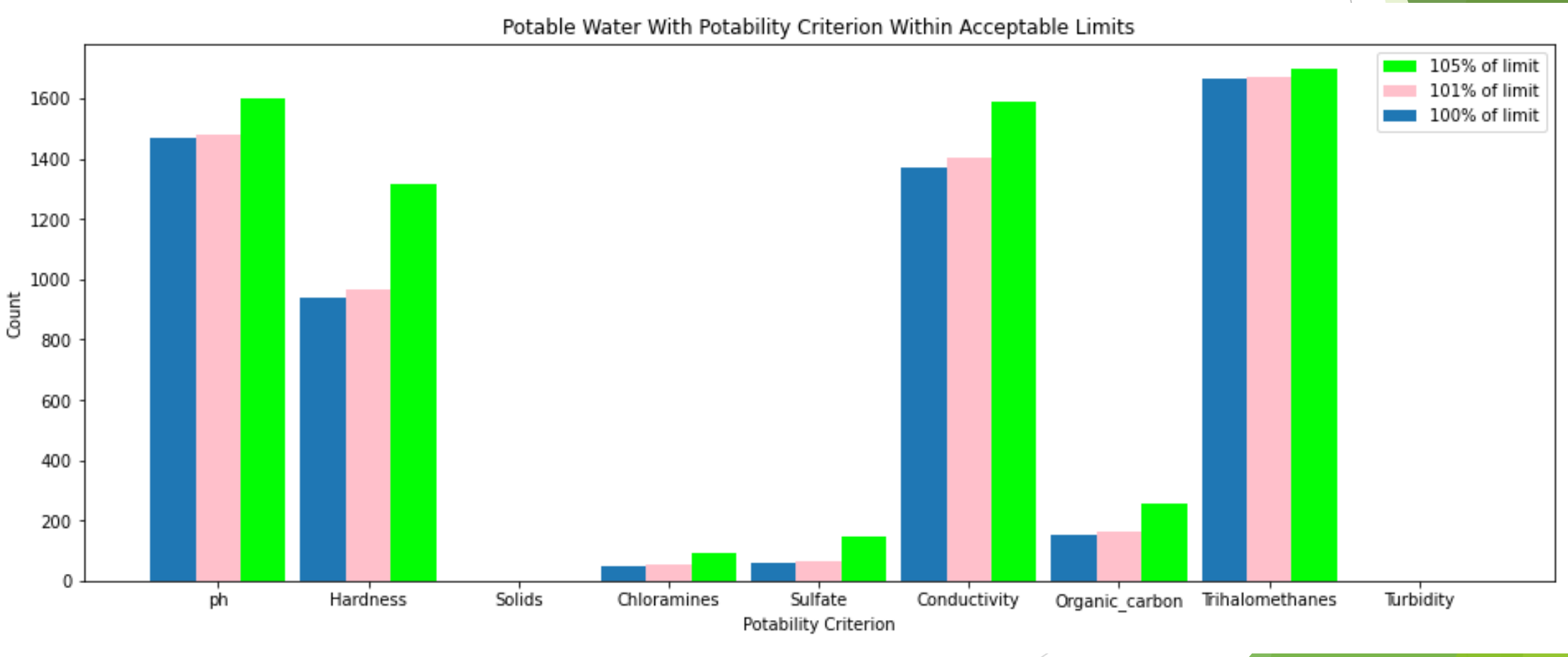
Data Analysis – Potability Breakdown

- Separated and created a data frame of just measurements of potable water.

	count	mean	std	min	25%	50%	75%	max
ph	1706.0	7.089571	1.392241	0.227499	6.282204	7.080795	7.841211	13.175402
Hardness	1706.0	195.682016	34.922142	47.432000	174.499822	196.654385	217.784278	317.338124
Solids	1706.0	22497.458015	9022.903478	728.750830	15896.365937	21363.740325	28174.620516	56488.672413
Chloramines	1706.0	7.167516	1.698013	0.530351	6.094134	7.212254	8.175744	13.127000
Sulfate	1706.0	332.479807	44.755204	129.000000	305.554444	333.775777	359.948574	481.030642
Conductivity	1706.0	424.261891	82.311378	201.619737	358.849003	418.548171	483.411484	695.369528
Organic_carbon	1706.0	14.241574	3.274677	2.200000	12.082600	14.203943	16.513838	23.604298
Trihalomethanes	1706.0	66.561131	16.278852	8.577013	56.040260	66.421884	77.368767	124.000000
Turbidity	1706.0	3.977342	0.779999	1.492207	3.439156	3.968748	4.515150	6.494249
Potability	1706.0	1.000000	0.000000	1.000000	1.000000	1.000000	1.000000	1.000000

Data Analysis – Limit Acceptability

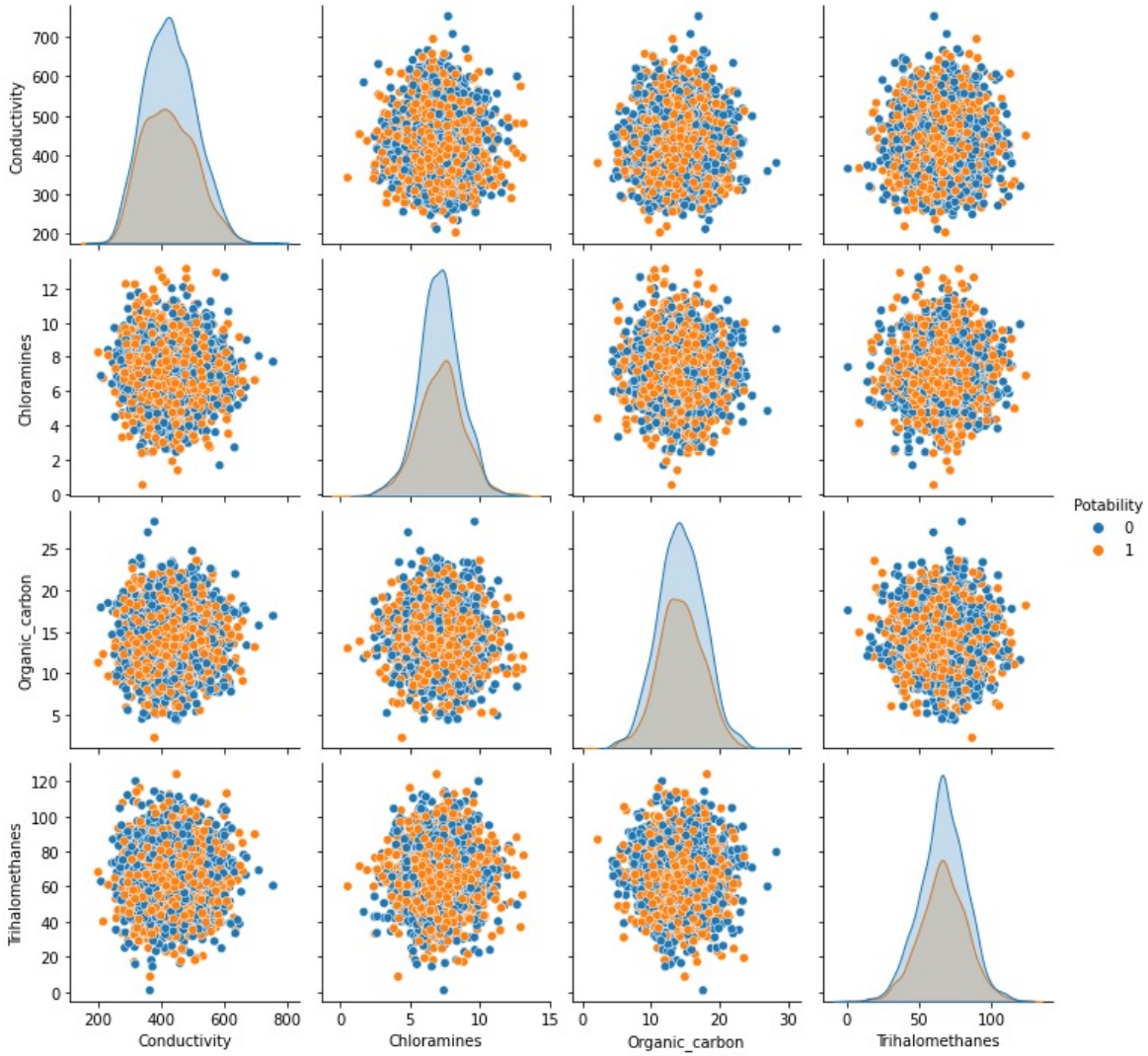
- None of the water samples deemed potable had criterion 100% within acceptable limits.
- All solids and turbidity measurements are outside of acceptable limits.



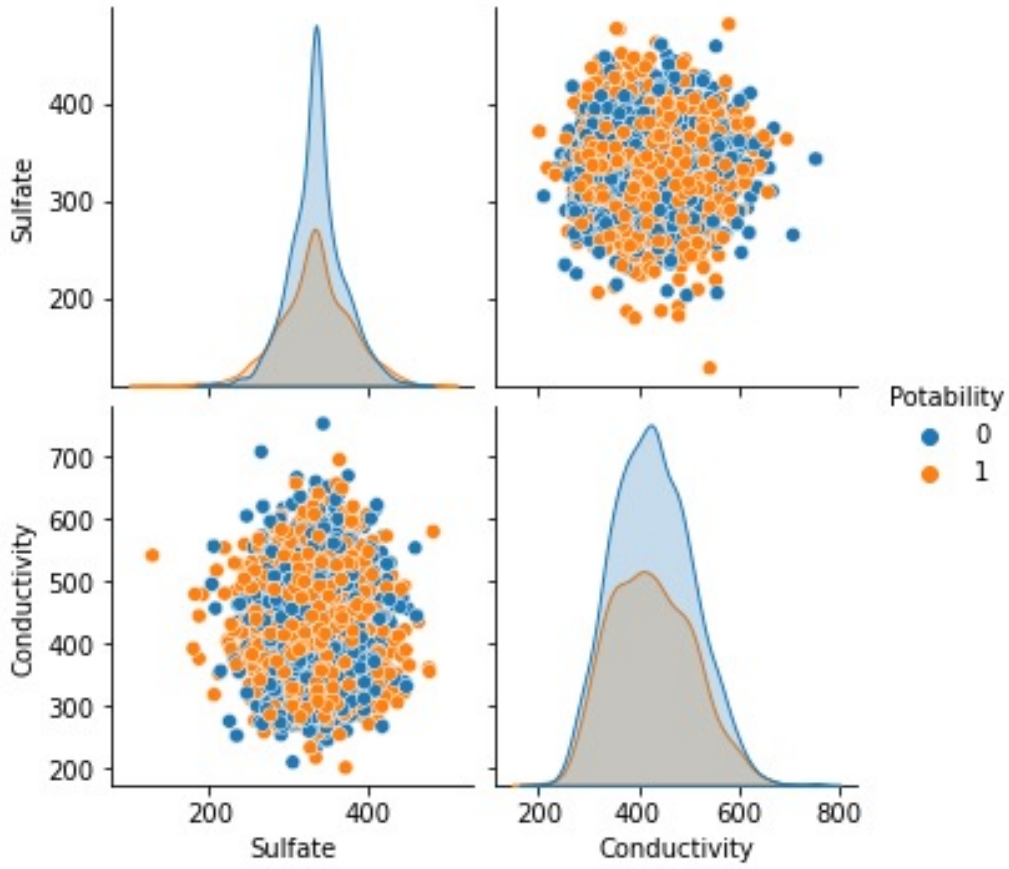
Data Analysis - Correlations

- One possible group that should give rise to a correlation:
  - Conductivity – Indicate chloride or organic compounds
  - Chloramines – combines with organic carbon to produce THMs
  - Organic Carbon – combines with chloramines to produced THMs
  - THMs – by-product of chloramines and organic carbon
- Another possible group that should give rise to a correlation:
  - Conductivity – can indicate the presence of sulfate
  - Sulfate

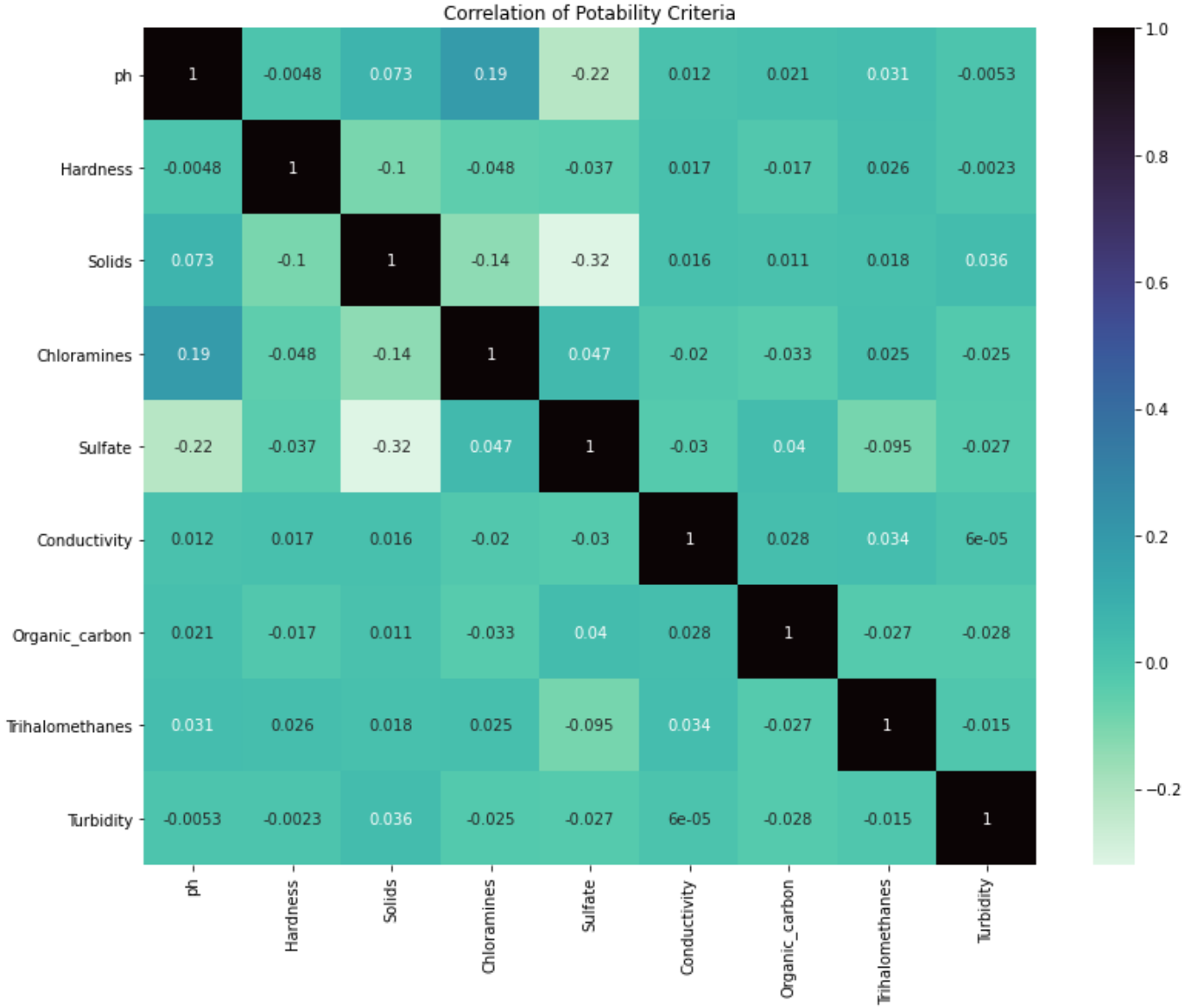
Data Analysis - Correlations



Data Analysis - Correlations



Data Analysis - Correlations



- Heatmap shows little to no correlation.
- Small correlation in chloramines and pH, possibly due the effect lower and higher pH levels have on chloramines, breaking them down.

Methods of Predictions

- There are several modeling methods that may predict water potability if given the measurement data with the potability criteria. Each method has a training set and a testing set of data. The method is trained with the training set of data and a model is fit to that data. The test data can be run through the model to predict the target outcome, in this case potability, and the prediction can be verified with the actual values of the test data.
- These are the methods that were used in potability predictions:
  - Logistic Regression
  - K-class Nearest Neighbor (KNN)
  - Random Forest

Methods – Data Prep For Modeling

- New data frames are created.
  - X will contain all the potability criteria
  - Y will contain potability.
- X and Y is split up into training data and test data.

```
In [75]: cols = wp.drop('Potability', axis=1).columns
         cols

Out[75]: Index(['ph', 'Hardness', 'Solids', 'Chloramines', 'Sulfate', 'Conductivity',
                'Organic_carbon', 'Trihalomethanes', 'Turbidity'],
               dtype='object')

In [74]: X = wp[cols]
         Y = wp['Potability'].astype(float)
         X_train, X_test, y_train, y_test = train_test_split(X,Y, test_size=0.33, random_state=42)
```

Method - Logistic Regression

- The goal of logistic regression is to estimate the probability of the binary outcome variable taking the value of 1 given the values of the predictor variables. The logistic regression model accomplishes this by fitting a logistic function, which is an S-shaped curve that ranges from 0 to 1, to the data.

```
In [77]: lg=LogisticRegression(max_iter=10000)
         lg.fit(X_train, y_train)
         lg_pred_y = lg.predict(X_test)
```

```
print(lg.coef_)
print(lg.intercept_)
```

```
[[-3.37578873e-05 -1.78610519e-03  1.15644206e-05  4.98117832e-05
  -1.08712807e-03  1.13833936e-04 -9.60297878e-05  1.02881726e-04
  -2.06493848e-05]]
[-3.75033419e-06]
```

Methods - KNN

- K-class nearest neighbor method uses the idea that a new data point will take on the class of the nearest neighbour or neighbours depending on how many neighbours are to be analyzed nearest to the new data point.
- Since this method is dependent on position in its data space, the X data set must be scaled first to be accurately represented.

In [78]:
```
scal = StandardScaler()
scld_X_train = scal.fit_transform(X_train)
scld_X_test = scal.transform(X_test)
scaled_X_train = pd.DataFrame(scld_X_train, columns=cols)
scaled_X_test = pd.DataFrame(scld_X_test, columns=cols)

KNN = KNeighborsClassifier(n_neighbors = 1)
KNN.fit(scld_X_train, y_train)
KNN_pred_y = KNN.predict(scld_X_test)
```

Methods – Random Forest

- During training, the random forest method creates many decision trees by randomly selecting subsets of the data and features. Each tree is constructed independently based on a random subset of the data and features, using a decision tree algorithm.
- This might prove to be the best method as there the method will provide models not based on some of the subsets of data such as Solids and Turbidity which don't seem to play the potable classification.

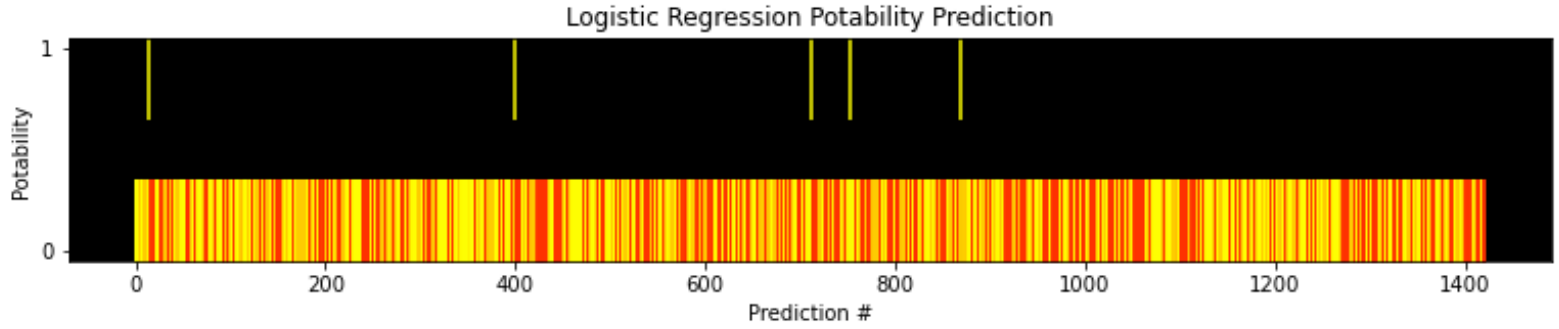
```
In [79]: rf = RandomForestClassifier(n_estimators=5)
         rf.fit(X_train,y_train)
         rf_pred_y=rf.predict(X_test)
```

Results – Logistic Regression

- 871 true negative (61.3%) / 545 false negatives (38.4%)
- 5 true positives (0.4%) / 0 false positive (0%)
- Accuracy of 62%

```
In [19]: print(confusion_matrix(y_test, lg_pred_y))
         print(classification_report(y_test, lg_pred_y))
```

```
[[871   0]
 [545   5]]
              precision    recall  f1-score   support

         0.0       0.62      1.00      0.76       871
         1.0       1.00      0.01      0.02       550

    accuracy                           0.62      1421
   macro avg       0.81      0.50      0.39      1421
weighted avg       0.76      0.62      0.47      1421
```

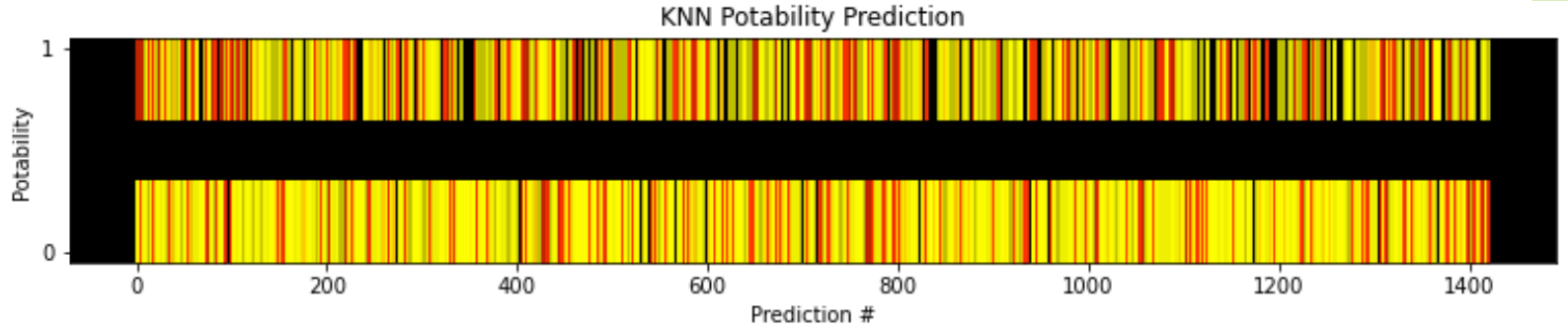


Results – KNN

- 716 true negative (50.4%) / 156 false negatives (11.0%)
- 394 true positives (27.7%) / 155 false positive (10.9%)
- Accuracy of 78%

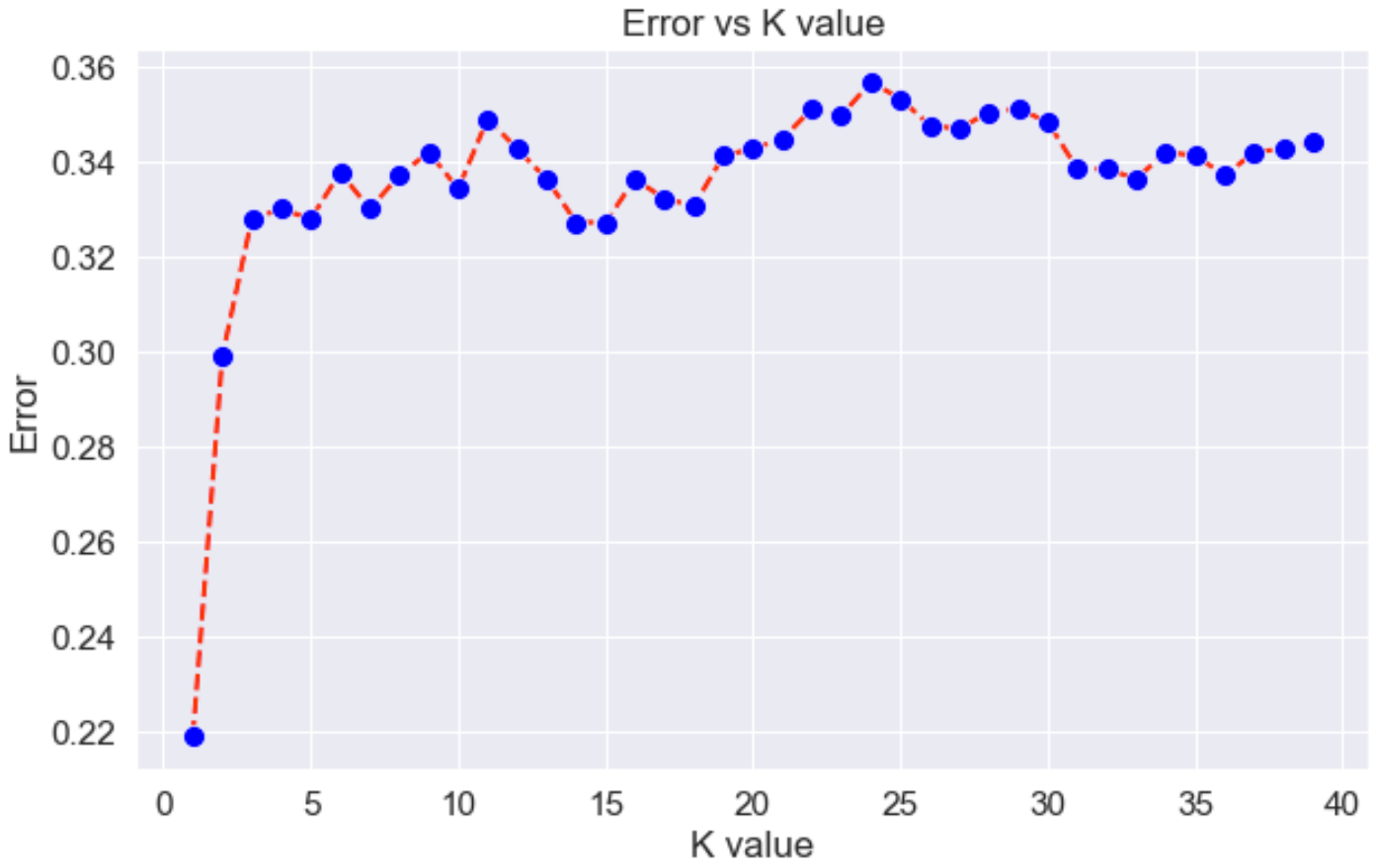
```
In [180]: print(confusion_matrix(y_test, KNN_pred_y))
          print(classification_report(y_test, KNN_pred_y))
```

```
[[716 155]
 [156 394]]
              precision    recall  f1-score   support

         0.0       0.82      0.82      0.82       871
         1.0       0.72      0.72      0.72       550

    accuracy                           0.78      1421
   macro avg       0.77      0.77      0.77      1421
weighted avg       0.78      0.78      0.78      1421
```



Results – KNN

- Ran the KNN model with k=1. The model was run through for k values up to 40 to see which k gives the lowest error. It was k = 1.

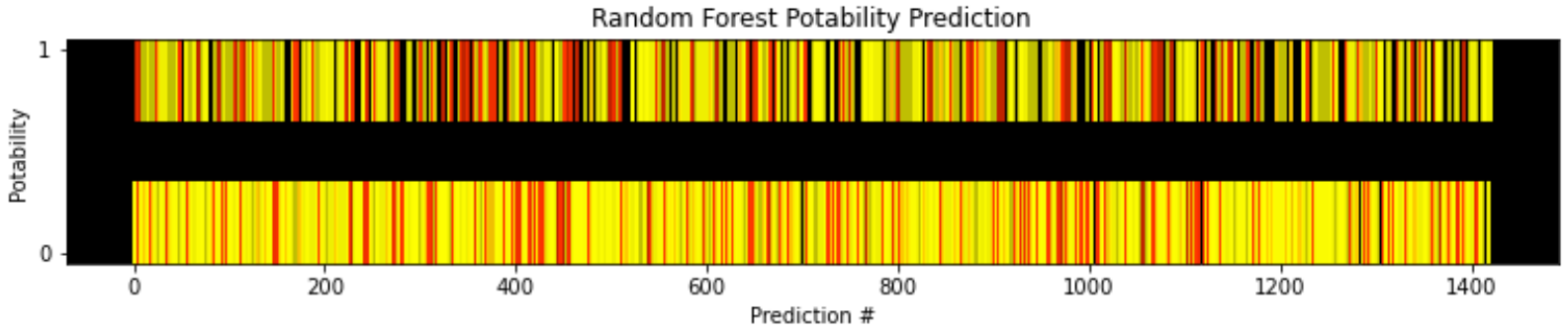


Results – Random Forest

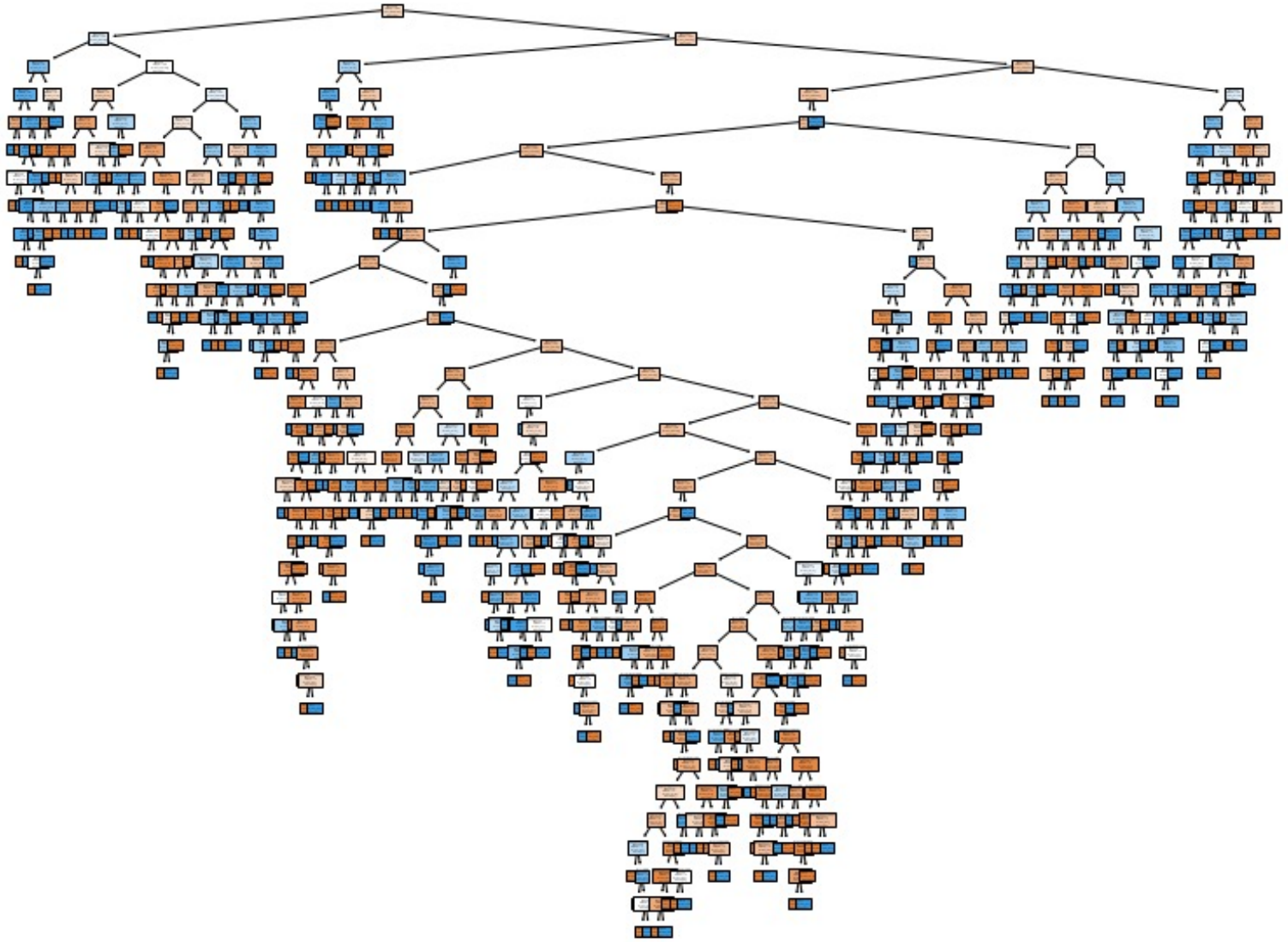
- 729 true negative (51.3%) / 169 false negatives (11.9%)
- 381 true positives (26.8%) / 142 false positive (10.0%)
- Accuracy of 78%

```
In [21]: print(confusion_matrix(y_test, rf_pred_y))
         print(classification_report(y_test, rf_pred_y))
```

```
[[729 142]
 [169 381]]
              precision    recall  f1-score   support

         0.0       0.81      0.84      0.82       871
         1.0       0.73      0.69      0.71       550

    accuracy                           0.78      1421
   macro avg       0.77      0.76      0.77      1421
weighted avg       0.78      0.78      0.78      1421
```



Results – Random Forest



Conclusion

- Prediction models KNN (k=1) and Random Forest (5 estimators) produced water potability predictions with the highest accuracy.
- KNN produced an accuracy of 78% with 50.4% true negatives and 27.7% true positives
- Random Forest produced an accuracy of 78% with 51.3% true negatives 26.8% true positives.
- No correlation could be detected between any of the potability criteria.

Future Potability Study

- Future potability data analysis and predictions:
  - Remove some of the criteria like solids and turbidity so see if the models had a higher accuracy in prediction.
  - To test out a new set of data to see if the models maintain prediction accuracy.
  - Acquire water portability data of First Nation communities.
  - Acquire data that could included temperature, location, elevation, distance to nearest fresh water source to base new prediction models on.