Topic: Pump it Up: Data Mining the Water Table.

1. Introduction

The “Pump it Up: Data Mining the Water Table” competition, hosted by DrivenData. The aim of “**DATA FOR SOCIAL GOOD**” project is to predict which water pumps in Tanzania are functional based on a set of characteristics. Tanzania currently has one of the fastest growing economies in Africa, however rural areas of the country remain in poverty with 1/3 of the population living below the poverty line. A lack of investment in water resources leaves many of the population without access to clean water. Poor drainage systems and insufficient capacity for storage or access means that only approximately 50% of the Tanzanian population have access to safe water. Ground water is often contaminated from poor drainage systems and surface water often contains human waste or bacteria.

Looking at the dataset of water pumps in Tanzania to predict the operating condition of a water point. By finding which water pumps are functional, functional needs repairs, and nonfunctional, the Tanzania Ministry of Water can improv the maintenance operations of the water pumps and make sure that clean, potable water is available to communities across Tanzania.

2. Problem Description

Predict which water pumps are faulty? (predict which pumps are functional, which need some repairs, and which don't work at all?)Using data from Taarifa and the Tanzanian Ministry of Water, predict the operating condition of a waterpoint for each record in the dataset.

A training dataset with information of nearly sixty thousand water points across Tanzania is provided and it is expected to build a model which will be able to predict which water points are functional, nonfunctional and functional but need repair on a test dataset. A model like this will help the Tanzanian government to tell which water point is likely to need repair and which are nonfunctional. This can help improving operational efficiency of water point maintenance.

3. Data Description

Taarifa is an open source platform for the crowd sourced reporting and triaging of infrastructure related issues. The data for this competition comes from the Taarifa waterpoints dashboard, which aggregates data from the Tanzania Ministry of Water.

https://www.drivendata.org/competitions/7/pump-it-up-data-mining-the-water-table/page/25/

The full training set provided has 59400 observations with 41 features.
Total number of attributes: 41
Number of training instances:59400
Number of testing instances: 14850
Class labels: Functional, Functional needs repair, Non-functional
The following are the 41 attributes used to build a model:

- amount_tsh - Total static head (amount water available to waterpoint)
- date_recorded - The date the row was entered
- funder - Who funded the well
- gps_height - Altitude of the well
- installer - Organization that installed the well
- longitude - GPS coordinate
- latitude - GPS coordinate
- wpt_name - Name of the waterpoint if there is one
- num_private - no description
- basin - Geographic water basin
- subvillage - Geographic location
- region - Geographic location
- region_code - Geographic location (coded)
- district_code - Geographic location (coded)
- lga - Geographic location
- ward - Geographic location
- population - Population around the well
- public_meeting - True/False
- recorded_by - Group entering this row of data
- scheme_management - Who operates the waterpoint
- scheme_name - Who operates the waterpoint
- permit - If the waterpoint is permitted
- construction_year - Year the waterpoint was constructed
- extraction_type - The kind of extraction the waterpoint uses
- extraction_type_group - The kind of extraction the waterpoint uses
- extraction_type_class - The kind of extraction the waterpoint uses
- management - How the waterpoint is managed
- management_group - How the waterpoint is managed
- payment - What the water costs
- payment_type - What the water costs
- water_quality - The quality of the water
- quality_group - The quality of the water
- quantity - The quantity of water
- quantity_group - The quantity of water
- source - The source of the water
- source_type - The source of the water

- source_class - The source of the water
- waterpoint_type - The kind of waterpoint
- waterpoint_type_group - The kind of waterpoint

For this project, I used Python’s scikit-learn library for running the Supervised Learning algorithms (KNN, DecisionTree and RandomForest). Used the pandas library for data manipulation and analysis and matplotlib for data visualization and plotting. For each of the four algorithms, I used the same general procedure for my analysis. Analysis included: a Model Complexity Curve (also called Validation Curve) to help tune hyperparameters, a Learning Curve to find the lower bound on the number of samples needed to learn this model and to investigate any issues due to high bias or high variance, a confusion matrix , classification report and ROC curve to evaluate the performance of different models.

4. Data Wrangling

Step 1: Remove redundant attributes

The data set has many similar features like (region_code, district_code, ward, lga, subvillage),(payment,payment_type),(extraction_type,extraction_type_group,extraction_type_class),(source,source_type,source_class),(waterpoint_type,waterpoint_type_group)gives similar information, so the correlation among the similar data are very high and removed those columns (date_recorded,installer,subvillage,region,district_code,lga,ward,public_meeting,permit,recorded_by,extraction_type_group,extraction_type_class,management_group,payment_type,quality_group,quantity_group,source_type,waterpoint_type_group) to avoid risk of overfitting.

Step 2: Missing values

All the values in num_private column is 0 and most of the values in schem_name are missing so, deleted num_private and scheme_name columns. The data set contains many zero values (amount_tsh, gps_hieght, population, construction_year and population).The zero values in (amount_tsh, gps_hieght, population, construction_year and population) are filled with mean values grouped by region_code. Due to the huge number of factors and the lack of a clear dominating value droped 'wpt_name' column.

The cleaned data set contains 19 features and one target variable(status_group). There are no significant outliers in the data set. It is also a good practice to know the columns and their corresponding data types, along with finding whether they contain null values or not. This has been done using pandas “.info()” methode.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 50580 entries, 0 to 59399
Data columns (total 20 columns):
id                   50580 non-null int64
amount_tsh           50580 non-null float64
funder               50580 non-null object
gps_height           50580 non-null float64
longitude            50580 non-null float64
latitude             50580 non-null float64
basin                50580 non-null object
region_code          50580 non-null int64
population           50580 non-null float64
scheme_management    50580 non-null object
construction_year    50580 non-null float64
extraction_type      50580 non-null object
```

```
management           50580 non-null object
payment              50580 non-null object
water_quality        50580 non-null object
quantity             50580 non-null object
source               50580 non-null object
source_class         50580 non-null object
waterpoint_type      50580 non-null object
status_group         50580 non-null object
dtypes: float64(6), int64(2), object (12)
memory usage: 10.6+ MB
```

- Data has 7 numerical and 13 categorical values.
- No variable column has null/missing values.

5. Exploratory Data Analysis

Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns,to spot anomalies,to test hypothesis and to check assumptions with the help of summary statistics and graphical representations. It is a good practice to understand the data first and try to gather as many insights from it. EDA is all about making sense of data in hand,before getting them dirty with it.

Possible questions need to be answer.

1.Individual counts of Functional, Non-functional and Functional Needs to repair.
2.Distribution of amount_tsh and gps_height among different status_grops.
3.Impact of population in different status_group.
4.Identify the region which contains most of the pumps.
5.Significance of categorical features (water_quality, extraction_type,waterpoint_type, payment, source,source_class,basin, funder, scheme_management,management,quantity) in the prediction of status_group.
Analysis of status_group has been done using “sns.countplot()” method, to understand the count of each status_group is used.

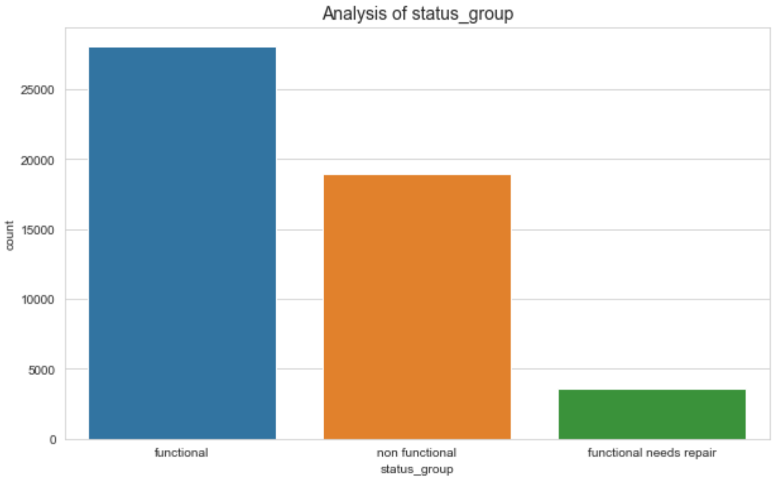


Fig.1 Count plot for Status_group

From the count plot we could identified that there are 28069 data points for functional pumps,18954,3557 data points for non-functional and functional needs to repair respectively.

To use classification algorithms for modelling, its necessary to remove correlated variables to improve model. It easy to find correlations using pandas “.corr()” function and can visualize the correlation matrix using a heatmap in seaborn.

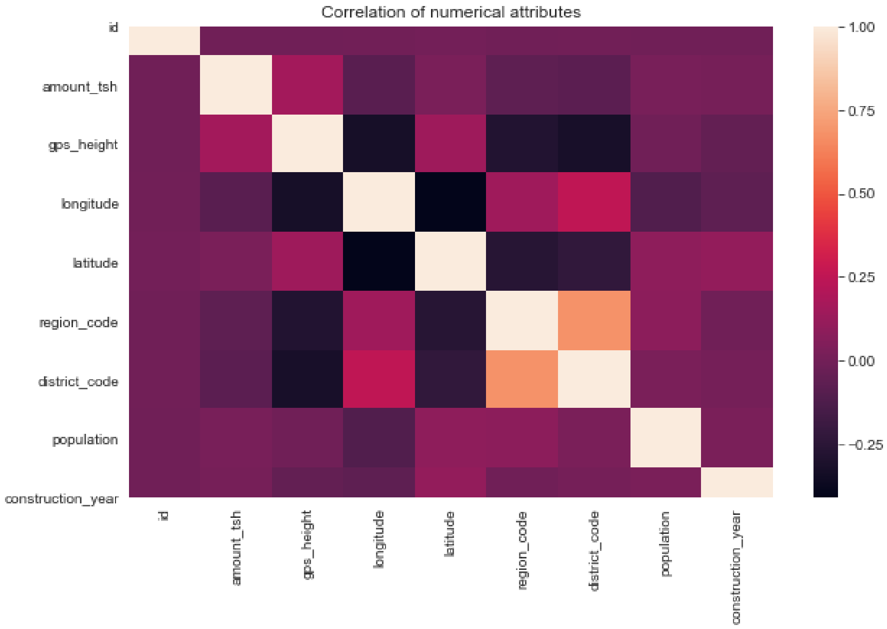


Fig.2 Heatmap for Numerical variables

Light shades represent positive correlation while darker shades represent negative correlation. It’s a good practice to remove correlated variables during feature selection. From the above heat map it is clear that, there is no correlated features.

A box plot (or box-and-whisker plot) shows the distribution of quantitative data in a way that facilitates comparisons between variables. The box shows the quartiles of the dataset while the whiskers extend to show the rest of the distribution.

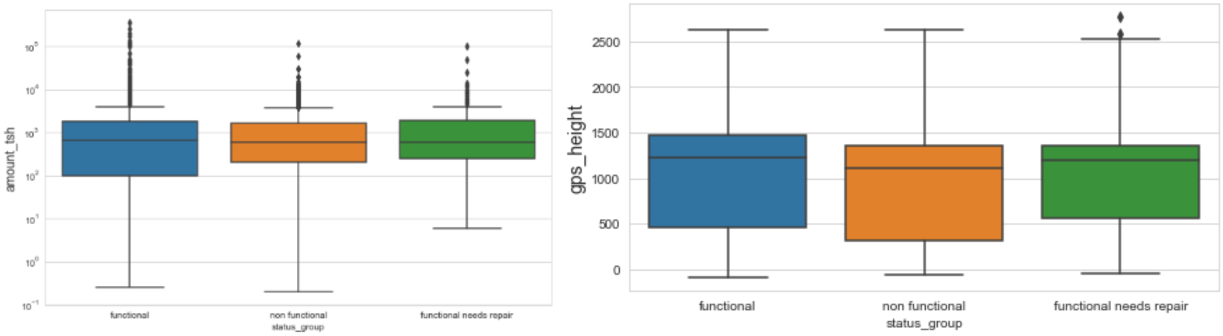


Fig.3 Box plots for amount_tsh and gps_height

From the above box plots for amount_tsh and gps_height the mean of different status_group are varying. So, we can assume that these two features have significant role in prediction. There are some outliers in the amount_tsh column.

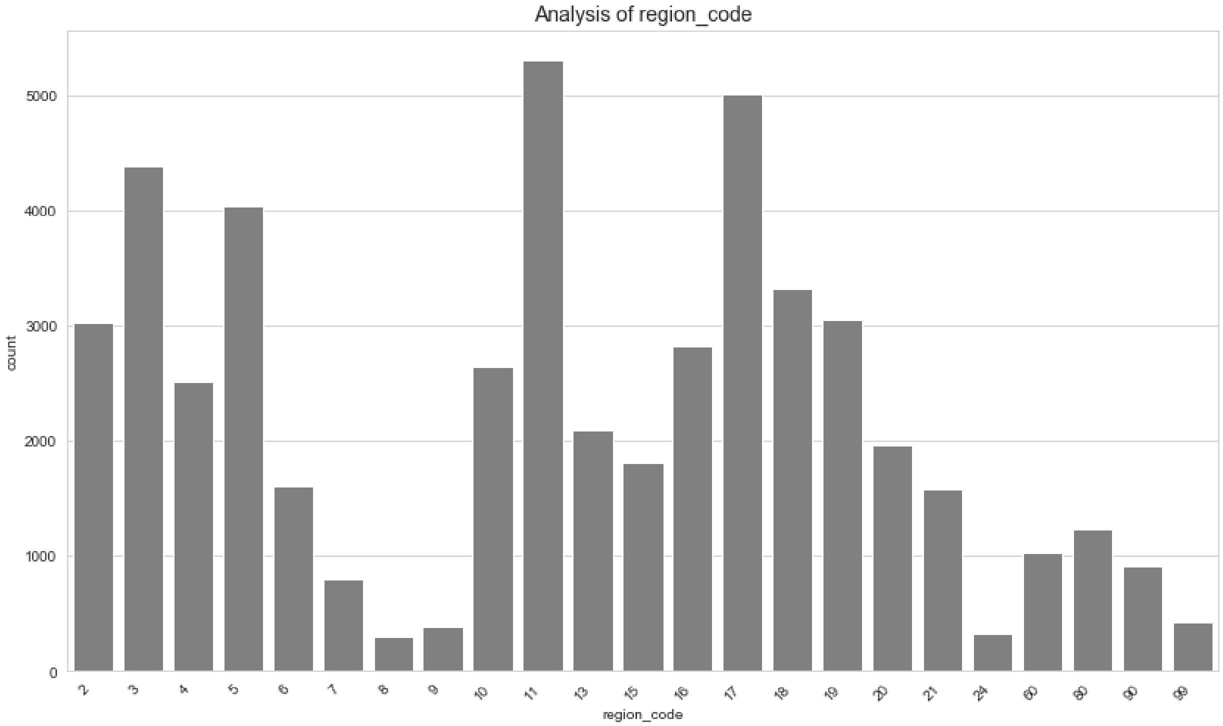


Fig.4 Analysis of region_code

From the Analysis of region_code it is obvious that most of the pumps are in region_code 11.

Analysis of categorical features using Bar plot

- Status_group vs water_quality:

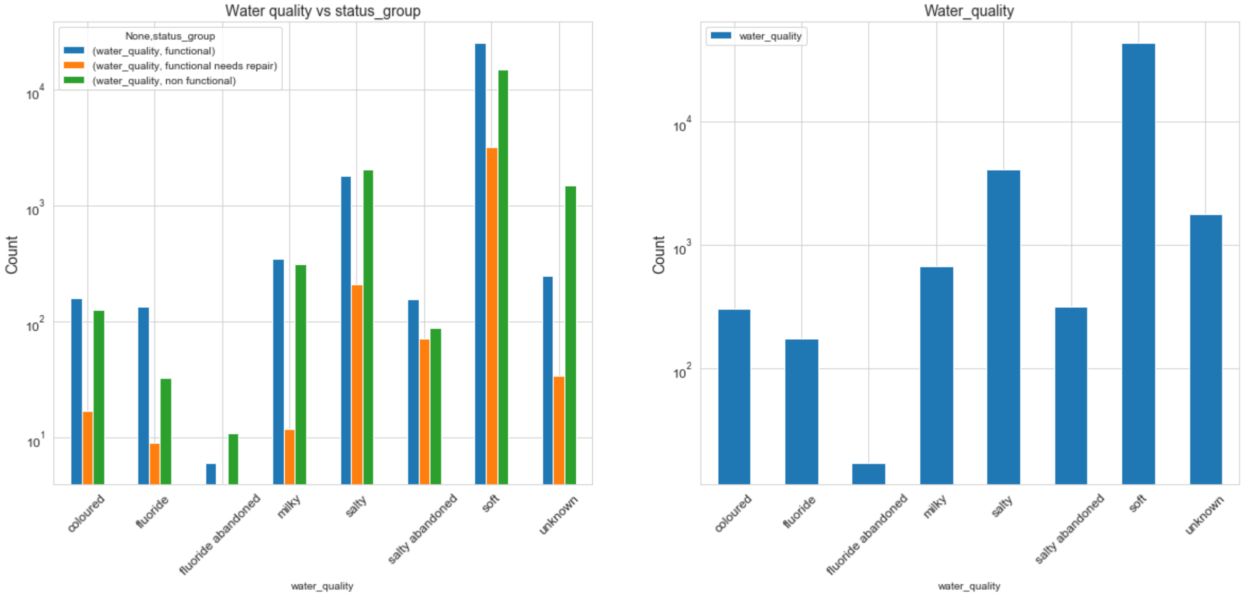


Fig.5 Status_group vs water_quality

The graph shows that most of the pumps carry soft water and pumps which carry abandoned fluoride are non-functional.

- Status_group vs waterpoint_type

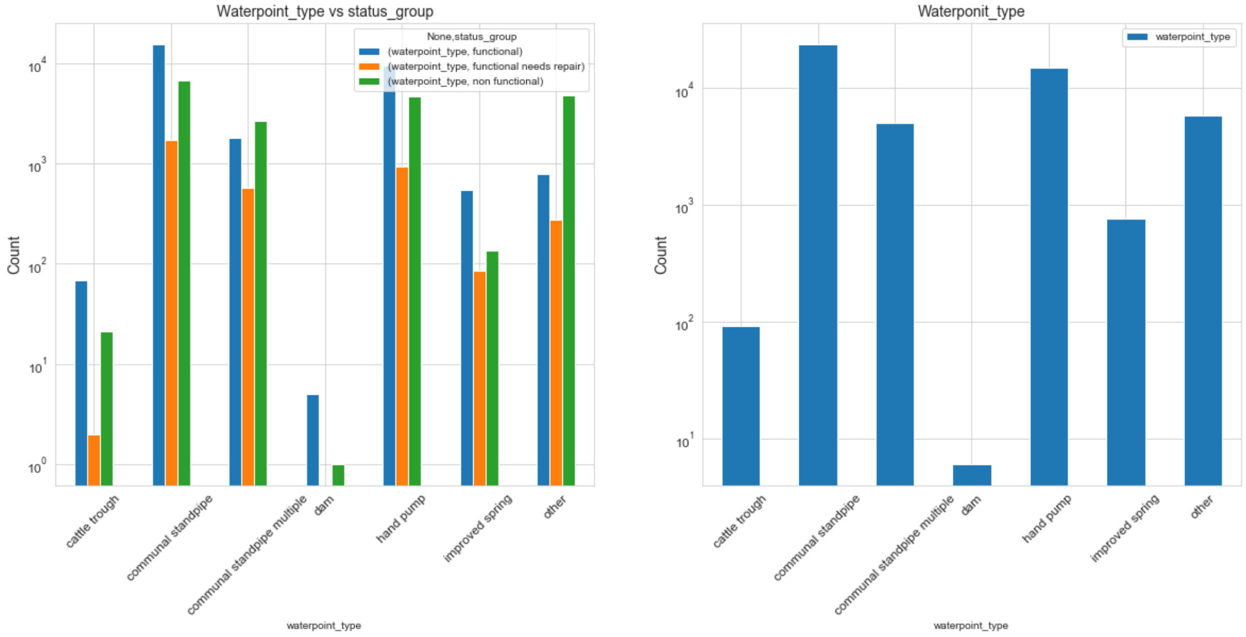


Fig.6 Status_group vs waterpoint_type

Most of pumps belongs to communal standpipe and hand pump type. From the graph it is clear that, there is no pumps needs repair in the 'dam' waterpoint_type.

- Status_group vs extraction_type

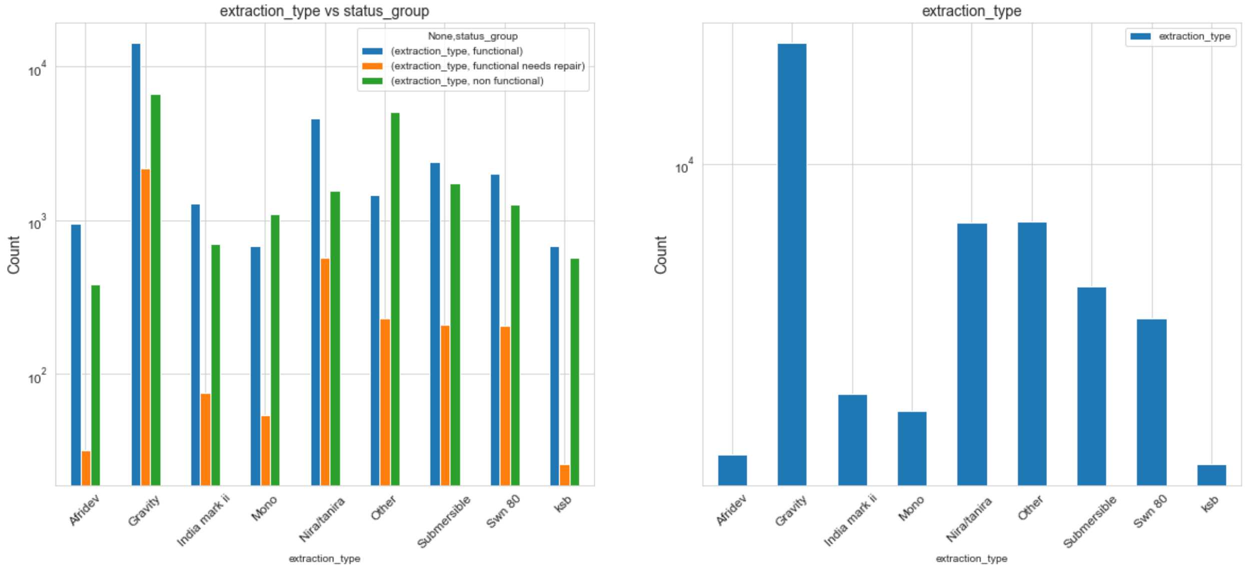


Fig.7 Status_group vs extraction_type

More pumps have gravity extraction_type and number of non-functional pumps exceeds in Mono extraction_type.

- Status_group vs source_class

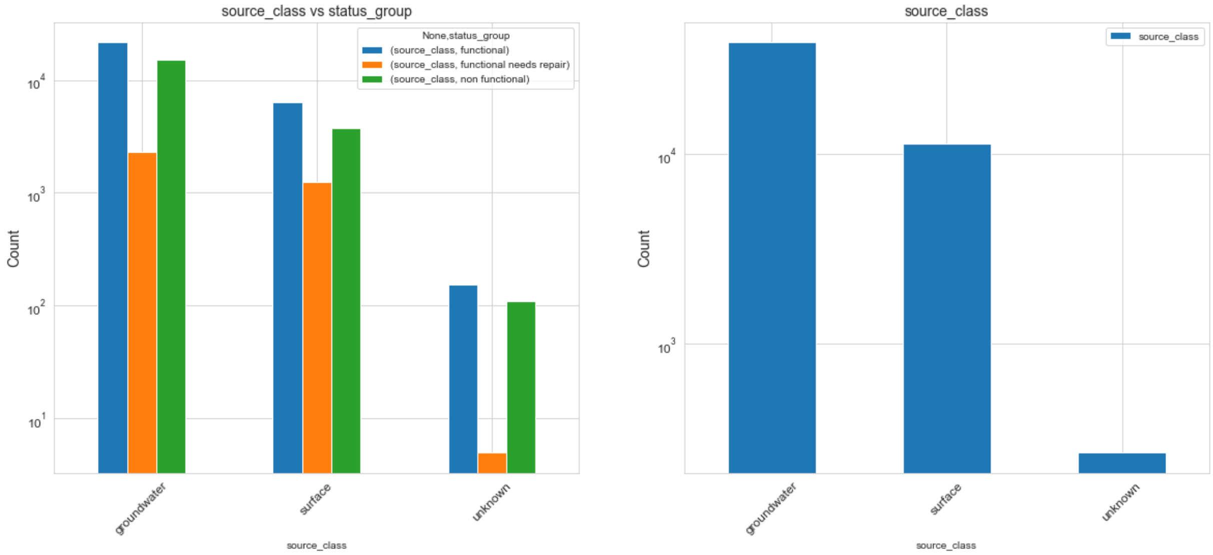


Fig.8 Status_group vs source_class

Groundwater is the main source of water for all the types of pumps.

6. Statistical Data Analysis

To find the significance of different variable to predict the status of pumps I have conducted the following statistical tests.

T-test: A t-test is a type of inferential statistic used to determine if there is a significant difference between the means of two groups, which may be related in certain features. The t-test is one of many tests used for the purpose of hypothesis testing in statistics.

The t-test is conducted on two numeric features, amount_tsh and gps_height across different status group.

Ho: The average mean of amount_tsh is same for both functional and nonfunctional group.

H1: The average mean of amount_tsh is not same for both functional and nonfunctional group.

alpha=.05

if p-value<alpha: reject null hypothesis

if p-value>=alpha: fail to reject null hypothesis

from the test it is identified that both amount_tsh and gps_height have p-value less than alpha so, rejected null hypothesis. It is concluded that the mean of amount_tsh and gps_height is varying among status_group. These two features can have important role in the prediction.

One-Way Anova: The one-way analysis of variance (ANOVA) is used to determine whether there are any statistically significant differences between the means of three or more independent (unrelated) groups. From One-way Anova also, It is concluded that the mean of amount_tsh and gps_height is varying among status_group. These two features can have important role in the prediction.

Chi-square test: The Chi Square statistic is commonly used for testing relationships between categorical variables. The null hypothesis of the Chi-Square test is that no relationship exists on the categorical variables in the population; they are independent.

A chi-squared test of independence is run to verify the relation between each categorical variable and the Status_group (dependent) variable. A significance level of 0.05 is chosen. For each test, if the p-value is less than the significance level, the corresponding variable is kept being possibly used in the model.

From the chi_square test it is identified that,the parameters (water_quality, extraction_type, waterpoint_type, payment, source,source_class,basin, funder, scheme_management, management, quantity) have p-value less than alpha so,reject the null hypothesis.That means the above listed features plays a relevant role in the prediction of status_group.

7. Model Selection and Evaluation

I have used Three Supervised Learning algorithms (KNN, Decision Tree and RandomForest). At a high level, here is the general methodology I followed when analyzing each algorithm:

Step1: Split 80% of the dataset into a training set and 20% into a testing set, using a train_test_split.

Step 2: Used Pandas function pd.get_dummies for categorical variables. Pd.get_dummies create a new dataframe which consists of zeros and ones.

Step 3: Of the 80% of training set, use a Model Complexity curve to find the best hyperparameters for tuning the model (with 5-fold cross validation).

4. Of the same 80% training set and using the best estimator identified in Step 2, plot the Learning Curve for the model (with 5-fold cross validation, using 20% of data incrementally cumulated)

5.Performed hyperparameter tuning using GridsearchCV to find the influence of other parameters in model performance.

6. Calculate the training accuracy of the model on the 80% that dataset from Step 1

7. Calculate the test accuracy of the model on the 20% of the dataset held out in Step 1

8.Model evaluation using confusion matrix, classification report and ROC curve.

- **K Nearest Neighbor Classifier (KNN)**

K-Nearest Neighbors is an instance-based learning algorithm. Rather than constructing a generalized model of the classification problem, it stores all training samples and classifies the testing data by taking a majority vote of the k nearest neighbors to the query point. Overfitting can occur in k-NN when we use a k value of 1. When k=1, each sample is in a neighborhood of its own and results in a model with low bias and high variance. As we increase k, we reduce the complexity in the model and we also reduce overfitting.

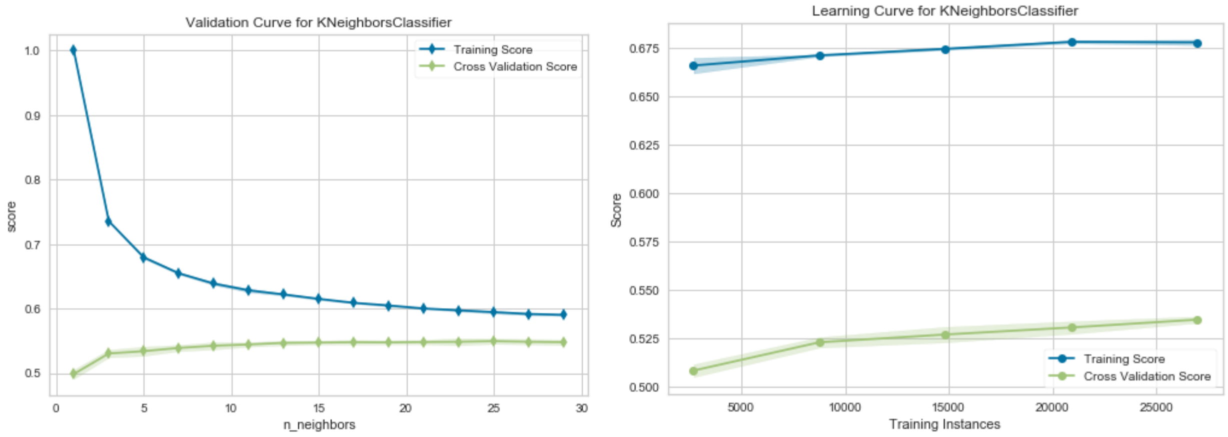


Fig.9 *Validation and Learning curve for KNN*

Using a validation curve seems like an excellent strategy for choosing k, and often it is. However, in the example above, all we can see is a decreasing variability in the cross-validated scores. the Model Complexity curve shows that overfitting occurs when k=1 and decreases as k increases. The peak validation accuracy I saw that was when k=30.This validation curve poses two possibilities: first, that do not have the correct param_range to find the best k and need to expand our search to larger values. The second is that other hyperparameters (such as uniform or distance-based weighting, or even the distance metric) may have more influence on the default model than k by itself does.

A learning curve shows the relationship of the training score versus the cross validated test score for an estimator with a varying number of training samples. As we give the model more training examples, the accuracy scores continue to increase and there is a large gap between the training and validation scores. This shows that the model suffers from high variance, which means more training data, or a larger k value will help improve the model's performance.

Although validation and learning curves can give some intuition about the performance of a model to a single hyperparameter, grid search is required to understand the performance of a model with respect to multiple hyperparameters.

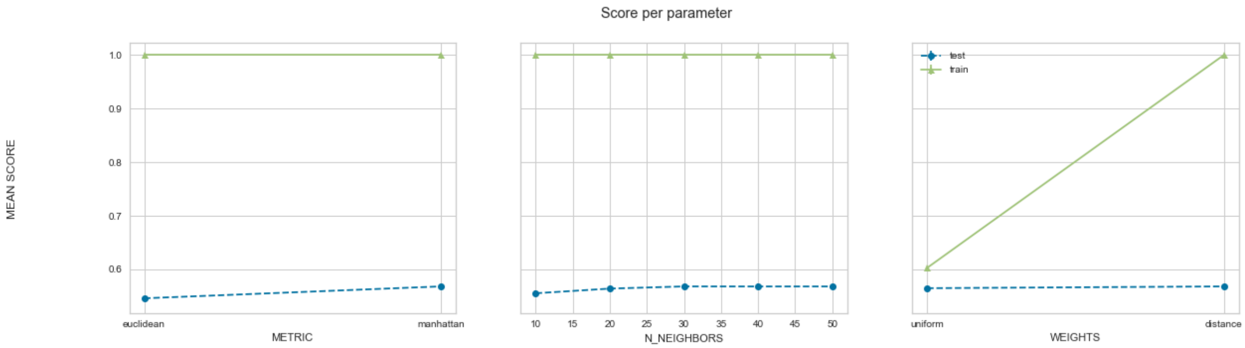
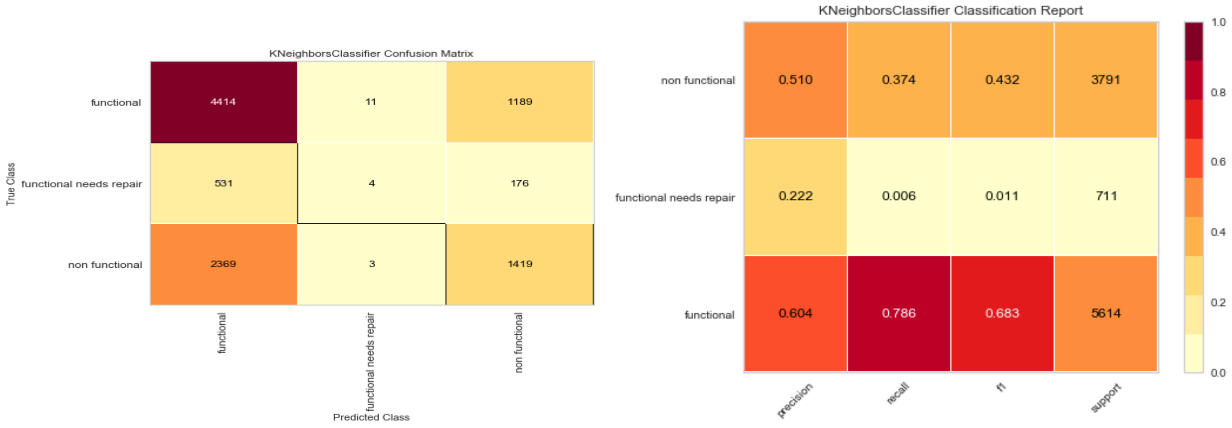


Fig.10 *GridSearch Result for KNN*

After doing hyperparameter turning using gridsearch I found that the model performs better when metric is Euclidian, n_neighbors=30 and weights=distance.

The final training accuracy was 100% and testing accuracy was 57.7%. As expected, the accuracy is lower.

A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. A Classification report is used to measure the quality of predictions from a classification algorithm. How many predictions are True and how many are False? More specifically, True Positives, False Positives, True negatives and False Negatives are used to predict the metrics of a classification report.



	functional	functional needs repair	non functional
functional	4414	11	1189
functional needs repair	531	4	176
non functional	2369	3	1419

	precision	recall	f1	support
non functional	0.510	0.374	0.432	3791
functional needs repair	0.222	0.006	0.011	711
functional	0.604	0.786	0.683	5614

Fig.11 Classification matrix and Classification report for K Nearest Neighbor Classifier

KNN is doing well on 'functional' and 'non-functional'. 'Functional needs repair' is the hardest to predict. Most of the real ones end up in 'functional'.

The Receiver Operating Characteristic (ROC) is a measure of a classifier's predictive quality that compares and visualizes the tradeoff between the model's sensitivity and specificity. When plotted, a ROC curve displays the true positive rate on the Y axis and the false positive rate on the X axis on both a global average and per-class basis. The ideal point is therefore the top-left corner of the plot: false positives are zero and true positives are one. This leads to another metric, area under the curve (AUC), which is a computation of the relationship between false positives and true positives. The higher the AUC, the better the model generally is. However, it is also important to inspect the "steepness" of the curve, as this describes the maximization of the true positive rate while minimizing the false positive rate.

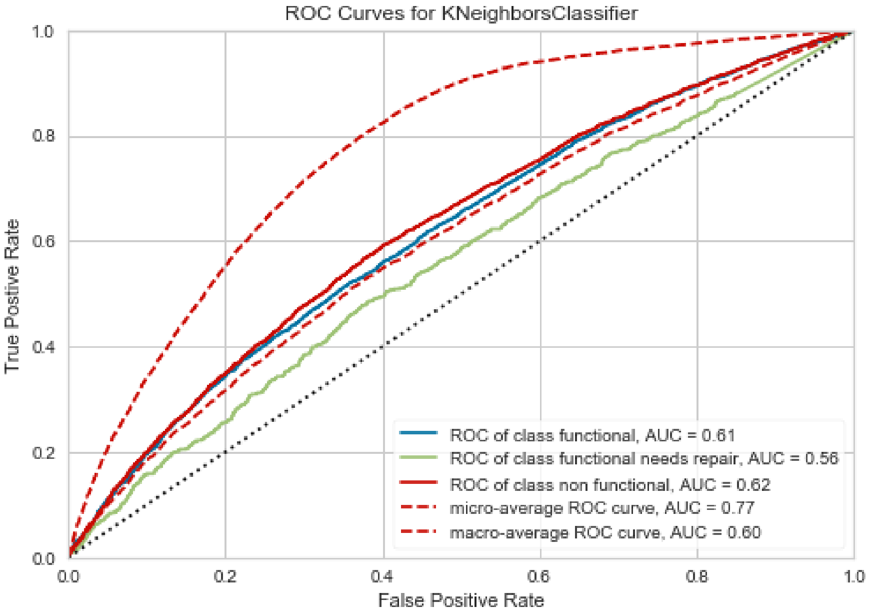


Fig.12 ROC Curves for K Neighbors Classifier

AUC for functional and non-functional is more than that of the functional needs repair.

- **Decision Tree classifier**

Tree based learning algorithms are one of the best and mostly used supervised learning methods. Tree based methods empower predictive models with high accuracy, stability and ease of interpretation. I have used decision tree classifier to get better performance.

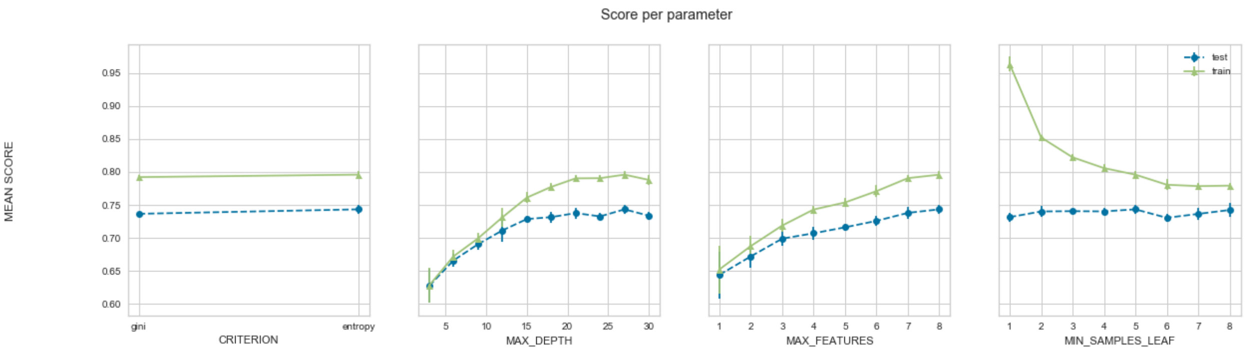


Fig.13 *GridSearch Result for Decision Tree Classifier*

The figure shows the gridsearch result for decision tree classifier. From the result we can select the better hyperparameters as criterion is entropy,Max_depth is 27,max_features is 8 and min_sample_leaf is 5. After running the classifier with the best parameter values the final training accuracy was 84.764% and testing accuracy was 74.149%. As expected, the accuracy is higher than the KNN classifier.

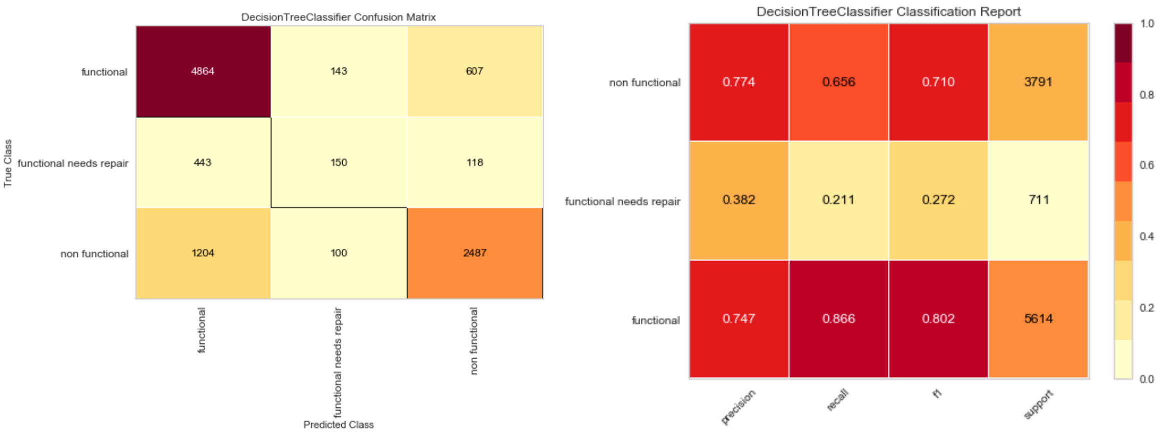


Fig.14 Confusion Matrix and Classification Report for Decision Tree Classifier

Decision Tree is doing well on 'functional' and 'non-functional'. 'Functional needs repair' is the hardest to predict. Most of the real ones end up in 'functional'.

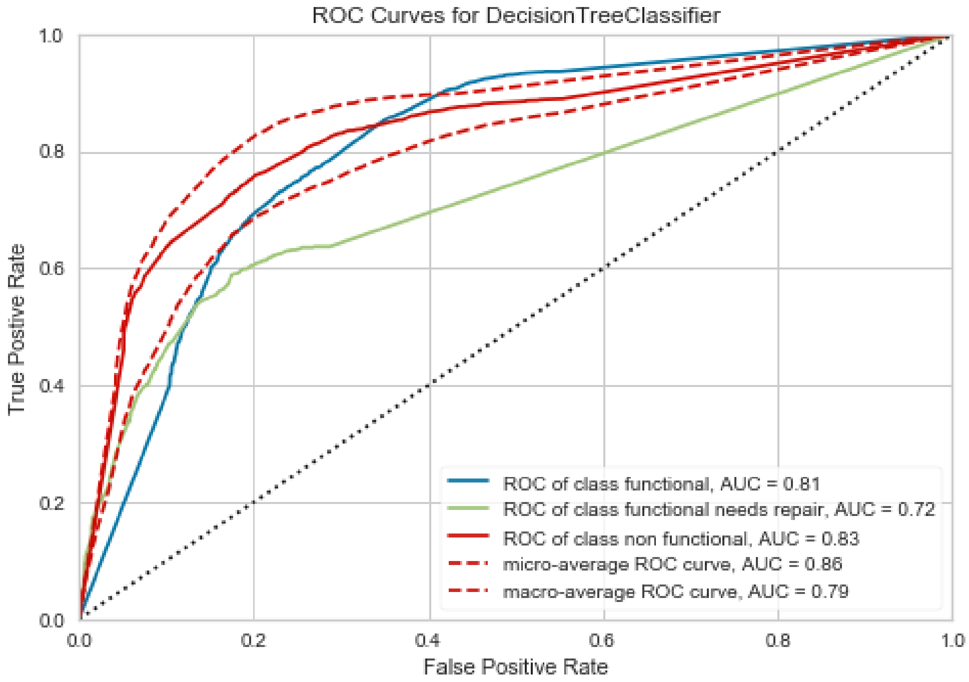


Fig.15 ROC Curves for Decision Tree Classifier

AUC for functional and non-functional is more than that of the functional needs repair.

- **Random Forest Classifier**

The Random Forest is a model made up of many decision trees. Random sampling of training data points when building trees. When training, each tree in a random forest learns from a **random** sample of the data points. The samples are drawn with replacement, known as *bootstrapping*, which means that some samples will be used multiple times in a single tree. The idea is that by training each tree on different samples, although each tree might have high variance with respect to a particular set of the training data, overall, the entire forest will have lower variance but not at the cost of increasing the bias.

At test time, predictions are made by averaging the predictions of each decision tree. This procedure of training each individual learner on different bootstrapped subsets of the data and then averaging the predictions is known as *bagging*, short for bootstrap aggregating.

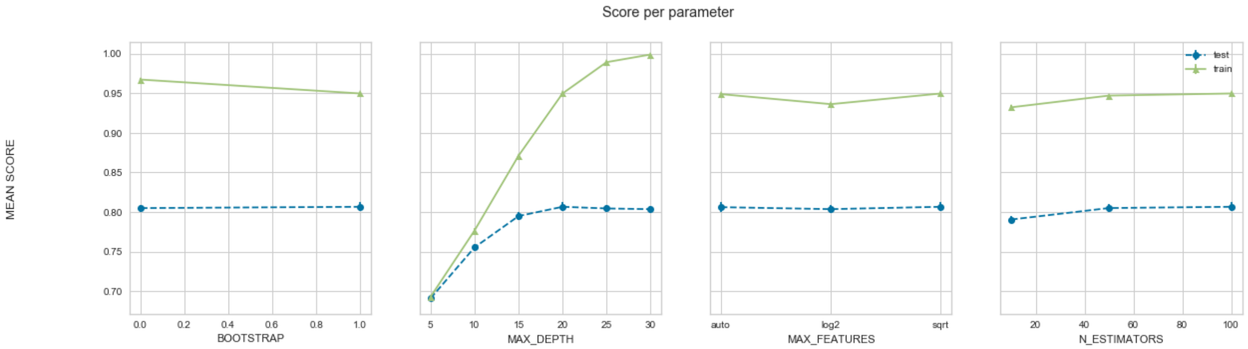
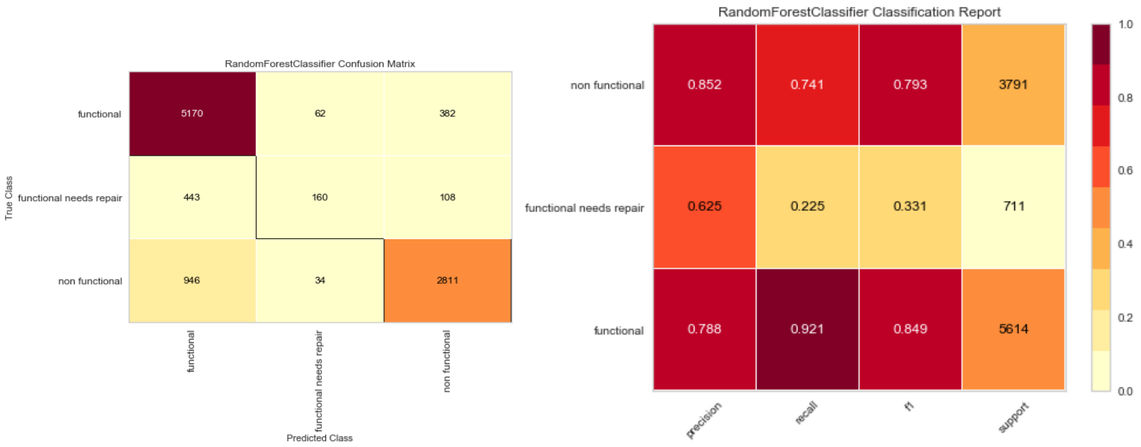


Fig.*16 GridSearch Result for RandomForest Classifier*

The figure shows the gridsearch result for RandomForest Classifier. From the result we can select the better hyperparameters as Max_depth is 20,max_features is ‘sqrt’ and n_estimators is 100.After running the classifier with the best parameter values the final training accuracy was 94.177% and testing accuracy was 80.476%. As expected, the accuracy is higher than the Decision Tree classifier.


RandomForestClassifier Confusion Matrix

True Class	functional	functional needs repair	non functional
functional	5170	62	382
functional needs repair	443	160	108
non functional	946	34	2811

RandomForestClassifier Classification Report

	precision	recall	f1	support
non functional	0.852	0.741	0.793	3791
functional needs repair	0.625	0.225	0.331	711
functional	0.788	0.921	0.849	5614

Fig.17 Confusion Matrix and Classification Report for RandomForest Classifier

Random Forest is doing very well on 'functional' and 'non-functional'. 'Functional needs repair' is the hardest to predict. Most of the real ones end up in 'functional'.

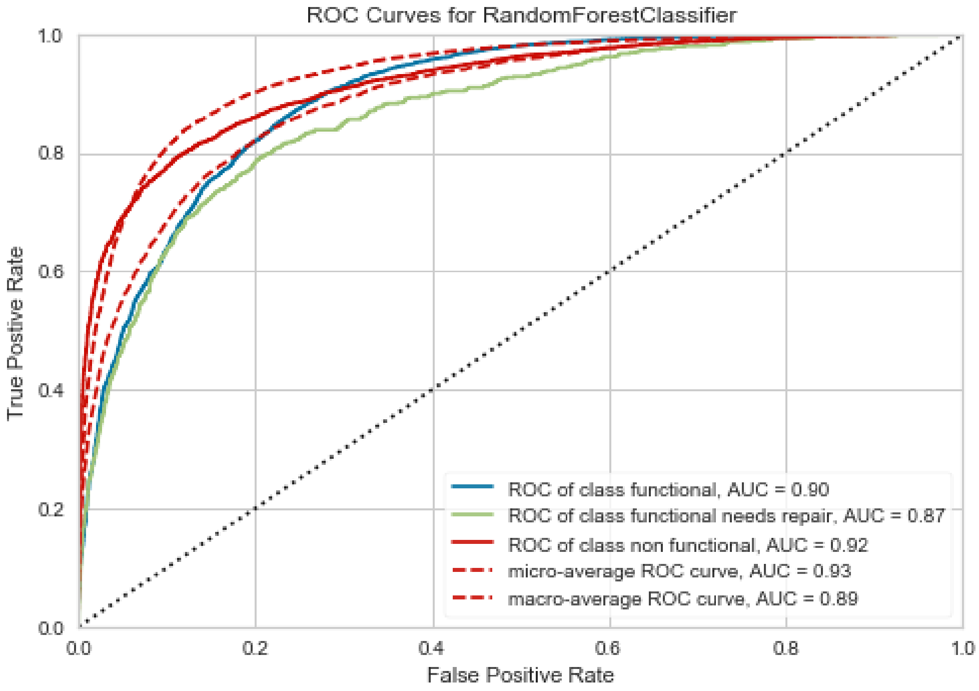


Fig.18 ROC Curves for RandomForest Classifier

AUC for functional and non-functional is more than that of the functional needs repair.

Comparison of different models:

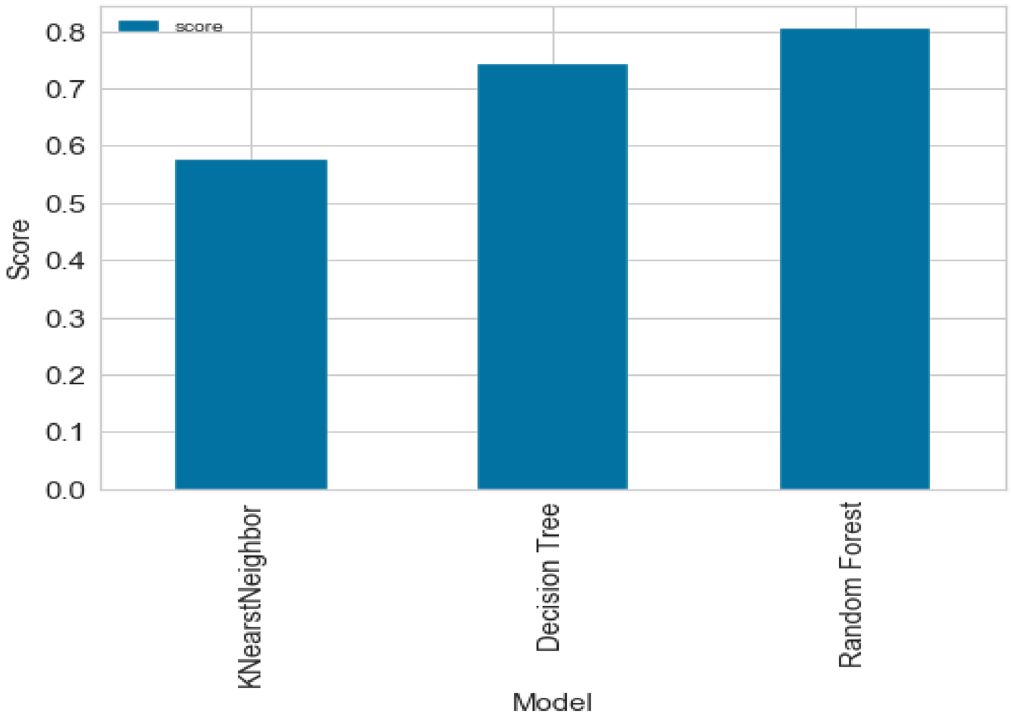


Fig.19 Comparison of Different

The above figure shows the performance score of different models. Among three models, KNN gives poor performance Random Forest gives better performance.

8. Conclusion and Future Work

Ideally, the aim of the project was to Predict the operating condition of a waterpoint for each record in the dataset. As you can see from the model comparison graph random forest model has managed to classify a good portion of the values with an overall accuracy of around 80.4%.

Future investigations are necessary to overcome class imbalance, the data set has severe class imbalance, with 32259 data points for functional water pumps,4317 data points for functional pumps but needs repair and 22824 data points for non-functional pumps. To mitigate the issue of class imbalance we can use various sampling methods like oversampling (Oversampling increases the weight of the minority class by replicating the minority class examples) and under-sampling(remove examples from the training dataset that belong to the majority class in order to better balance the class distribution).

9. Reference

- https://www.drivendata.org/competitions/7/pump-it-up-data-mining-the-water-table/page/24/
- https://www.scikityb.org/en/latest/search.html?q=spider+chart&check_keywords=yes&area=default
- https://medium.com/@vaibhavshukla182/pump-it-up-data-mining-the-water-table-f903d4cfc7a8
- http://scikit-learn.org/stable/
- https://pandas.pydata.org/
- https://matplotlib.org/