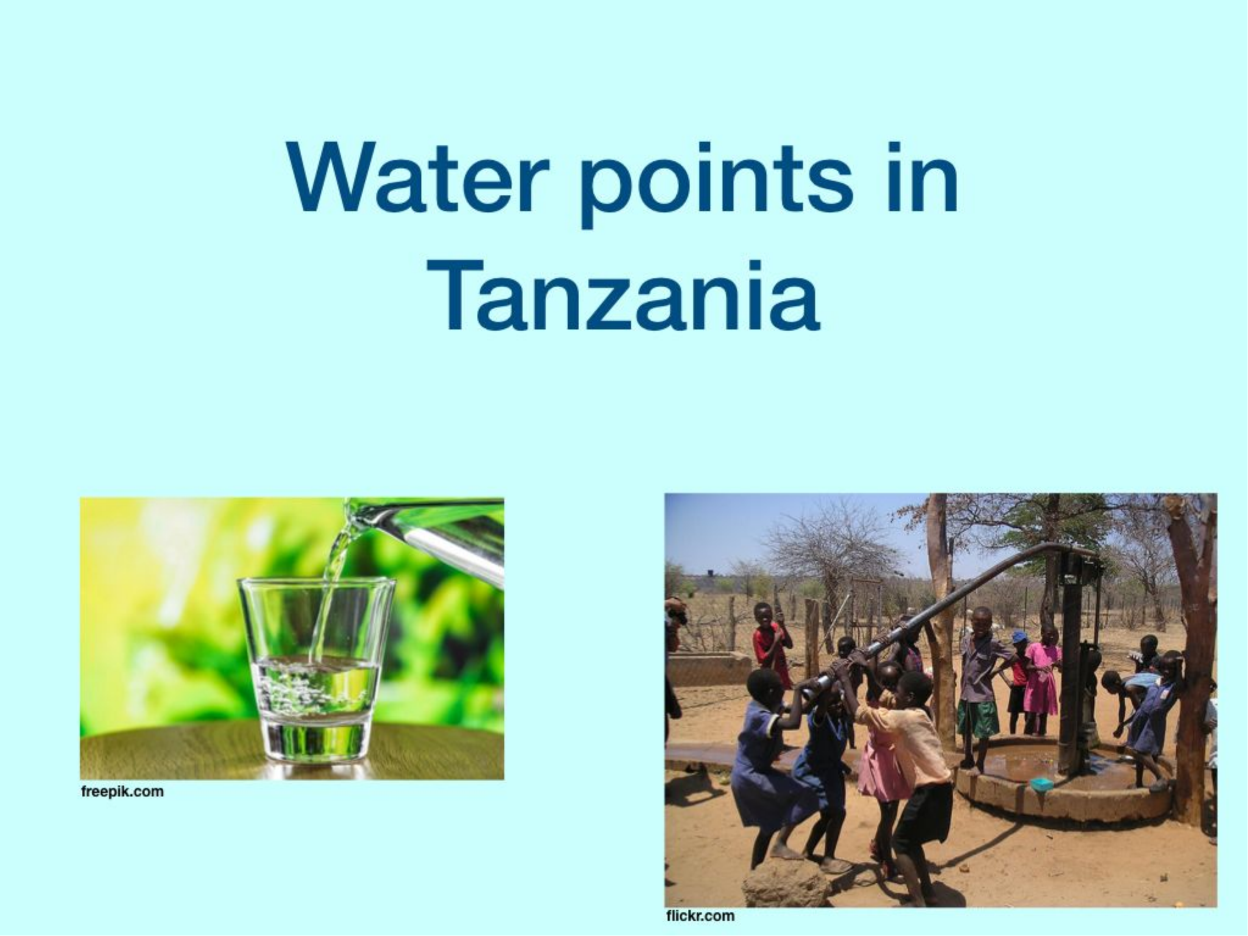


Water points in Tanzania

freepik.com

flickr.com

When you enjoy a glass of water, do you ever think of how this water comes to you? Do you think about the infrastructures, and all the hard work it takes to maintain a good quality and quantity of water?

In Tanzania, slightly more than half of the population has access to clean water.

This project’s objective is to help reduce the operation and maintenance costs, while improving continuity of water supply. For this objective, I try to predict the operational status of water points.

This is a DrivenData competition. The dataset comes from Taarifa and the Tanzanian Ministry of Water.

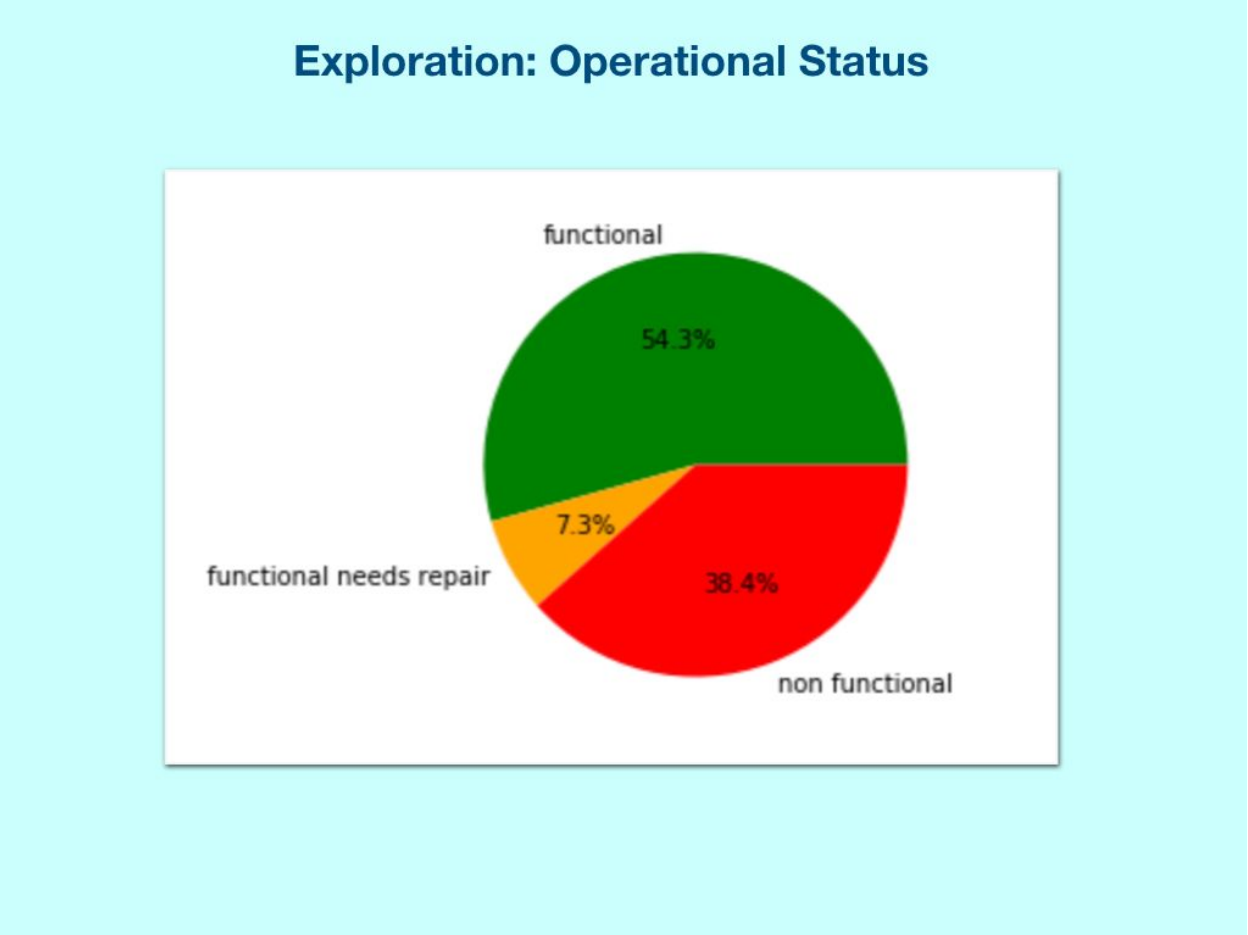
1 Exploration

2 Machine Learning

3 Synthesis

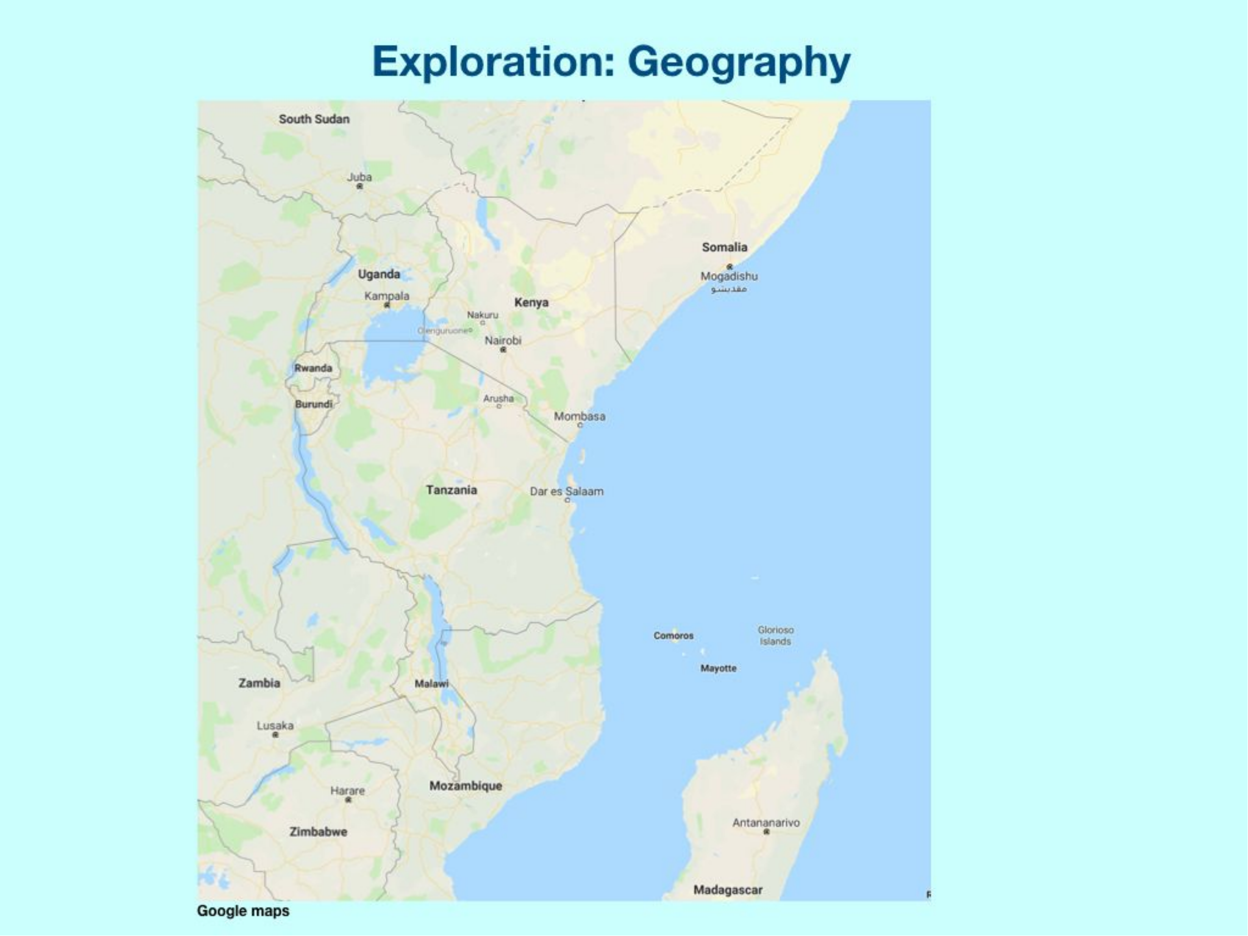
cleanwateraction.org

First, I will present the data exploration,
Then, I will explain how I made the machine learning model,
And finally how I synthesized it to predict the operational status of unseen water points.



Exploration: Operational Status

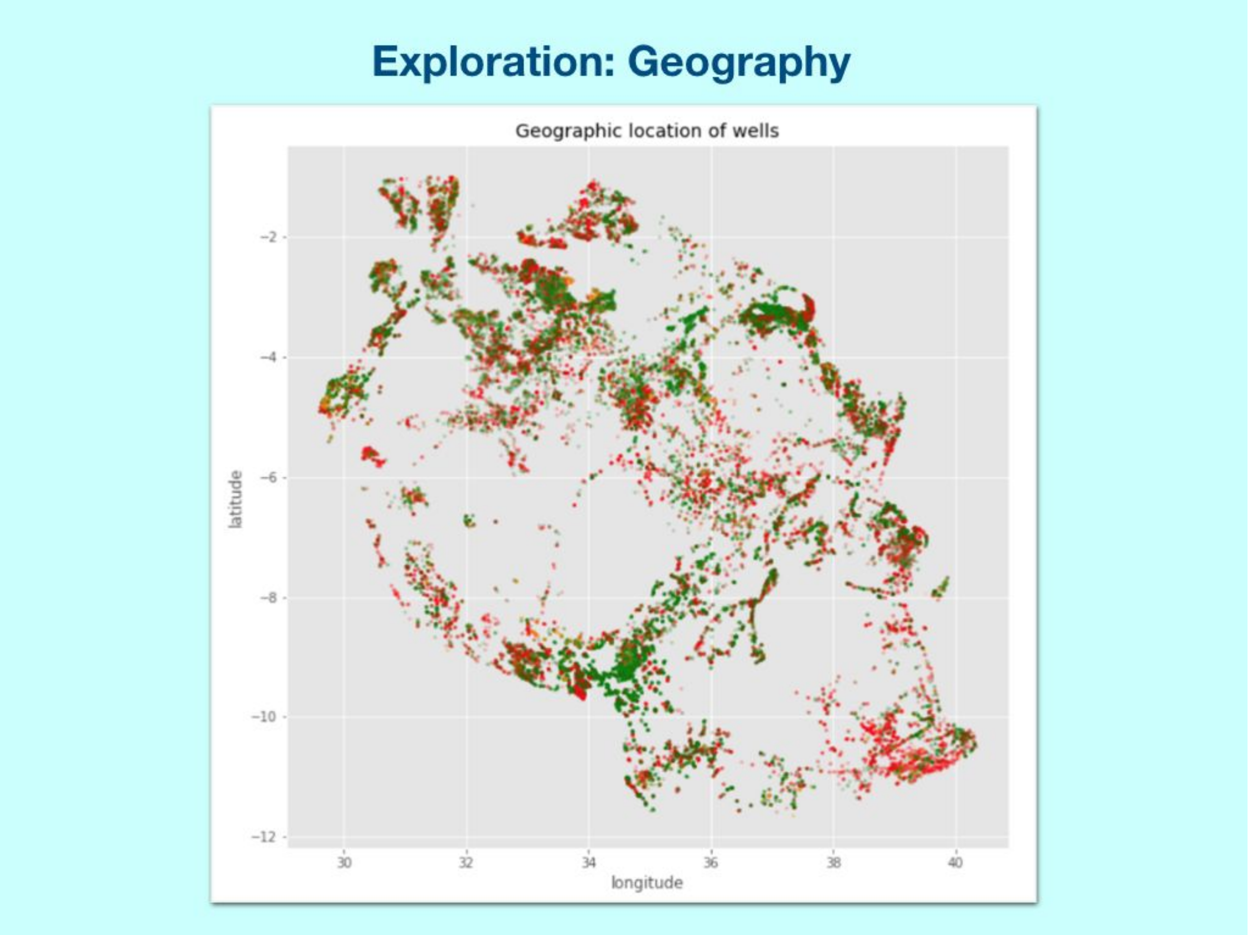
I have about 60,000 water points to study.
This is the operational status I will try to predict. It’s ordered.
Hopefully, a majority of water points are functional. Only a few needs repair, so they will be more difficult to classify.
This color code green, orange, red for the operational status will remain the same in this presentation.



Exploration: Geography

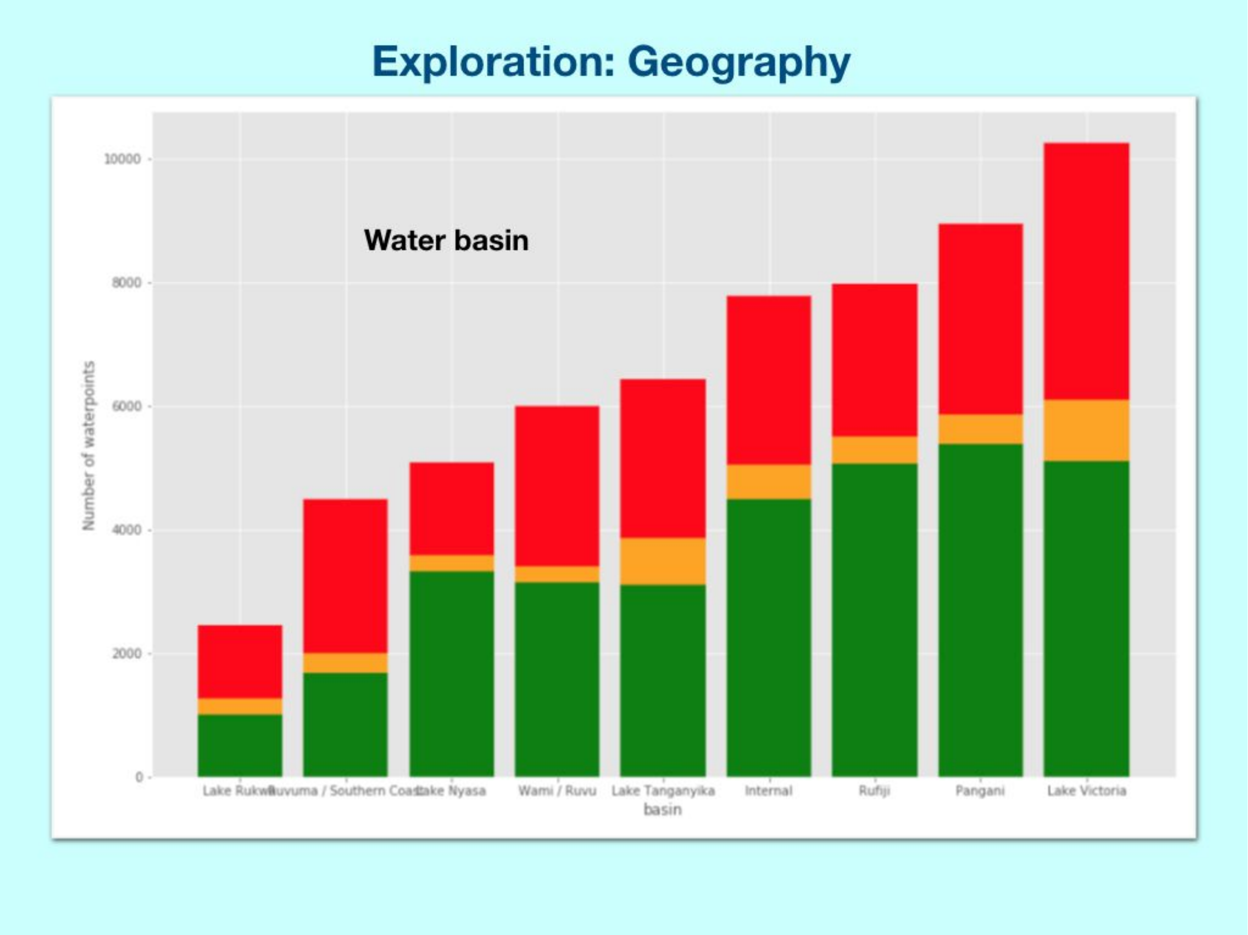
Google maps

Where is Tanzania?
It’s in East Africa, by the Indian Ocean.
Water wise, there is Lake Victoria in the North, Lake Tanganyika along the West border, and Lake Malawi in the South.



Exploration: Geography

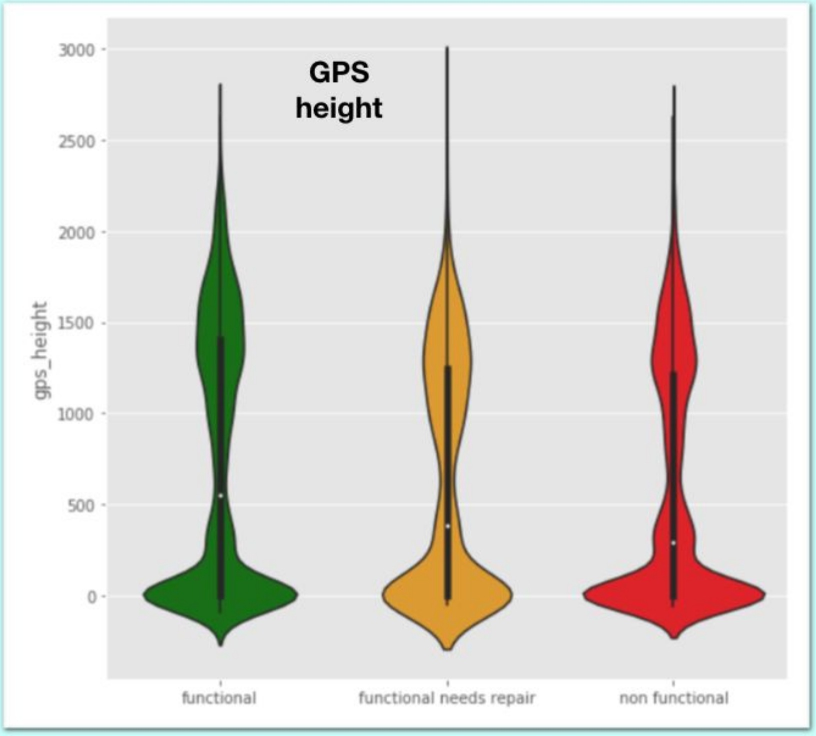
The water points data provides latitude and longitude, so I can plot their location on a map. There were 1812 erroneous coordinates (outside of Tanzania) that I replaced with the mean of their region.
We can see that some areas like the South East have a majority of non functional water points. Statistical test (chi-square test for independence) confirms that the operational status repartition is different according to the region.



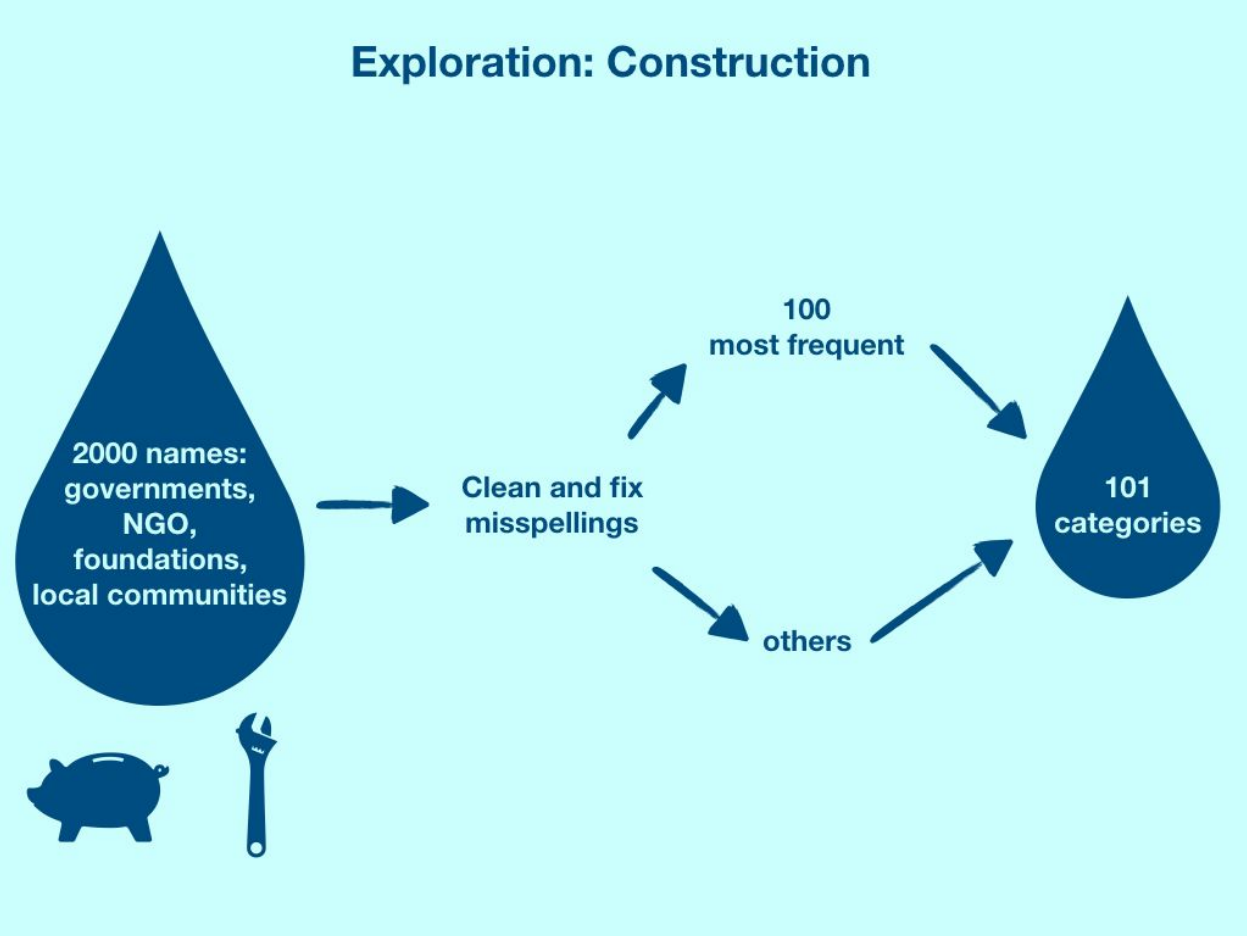
Exploration: Geography

Tanzania is divided in 9 water basins, and they have a different operation status repartition as well, as shown by the plot and the statistical test.

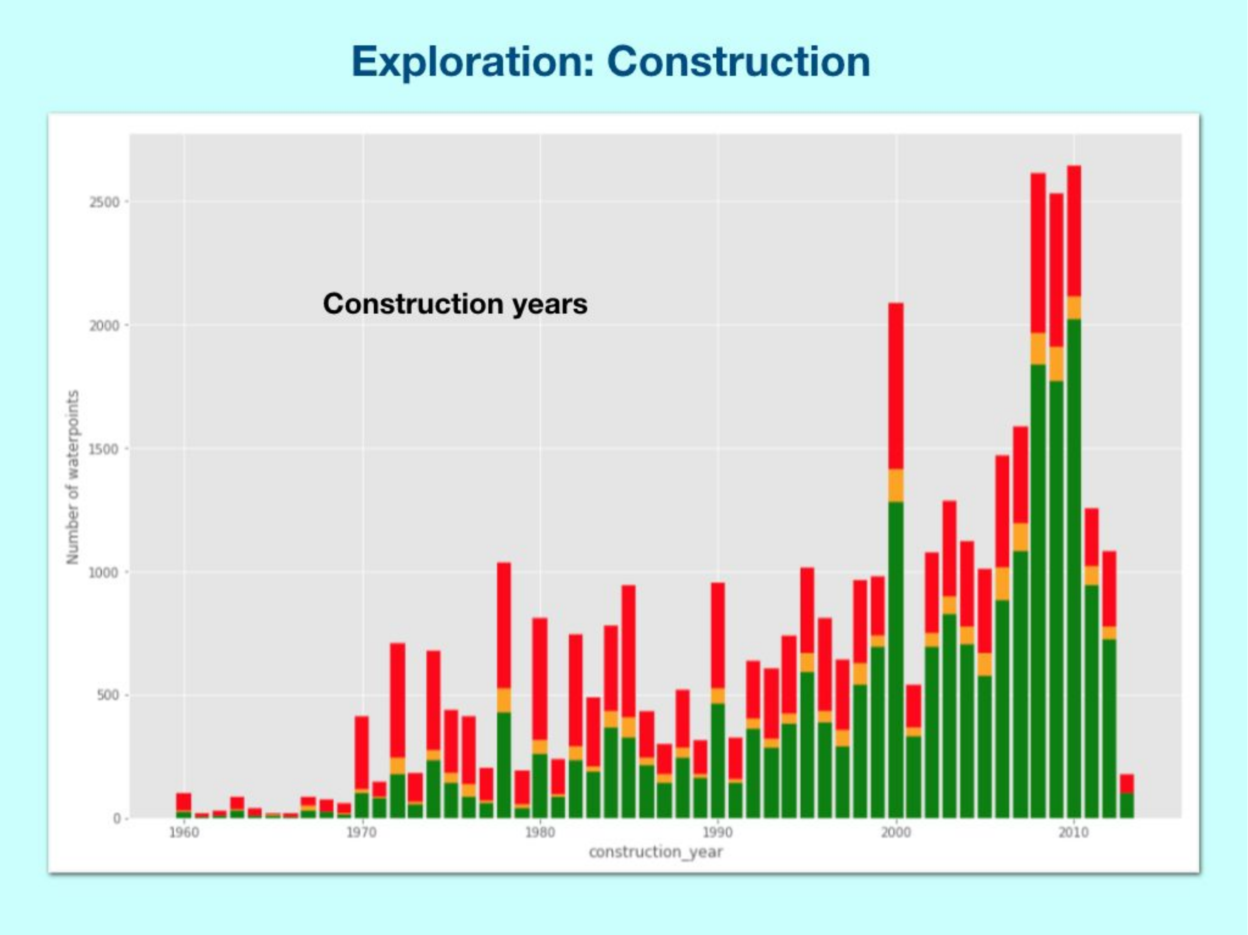
Exploration: Geography



The violins look very similar, so let’s test the independence of each status against the others.
The low p_values show the repartition of gps_height for these groups is actually very different.

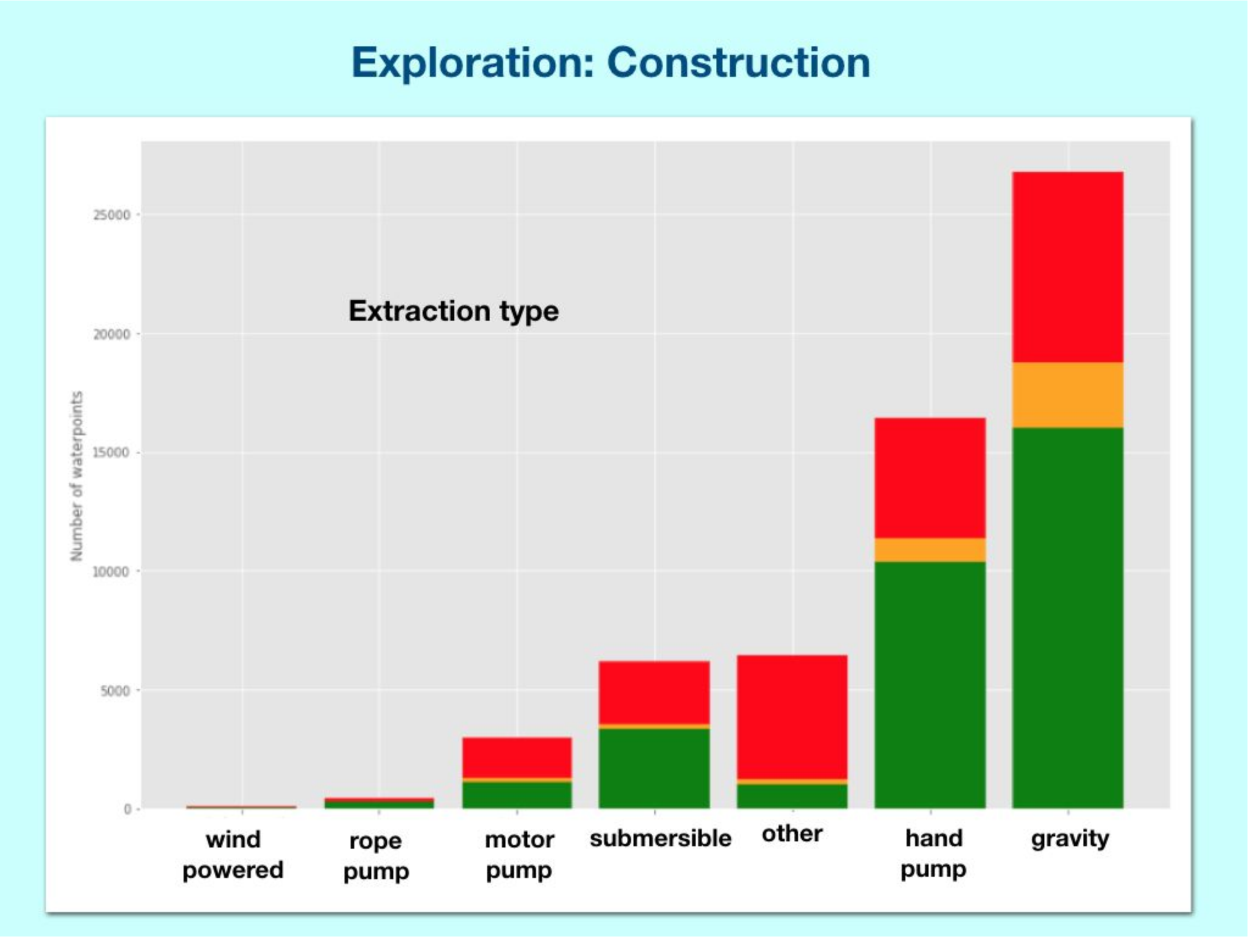

Exploration: Construction
2000 names: governments, NGO, foundations, local communities
Clean and fix misspellings
100 most frequent
others
101 categories

For the construction features, the founder and installer are important information, but they are also very messy. There is about 2000 different names of governments, NGO, foundations, and local communities. The first task is to clean and fix misspellings. Then, I picked the 100 most frequent organizations and set all the others to ‘Other’. I know have 101 categories I can work with.



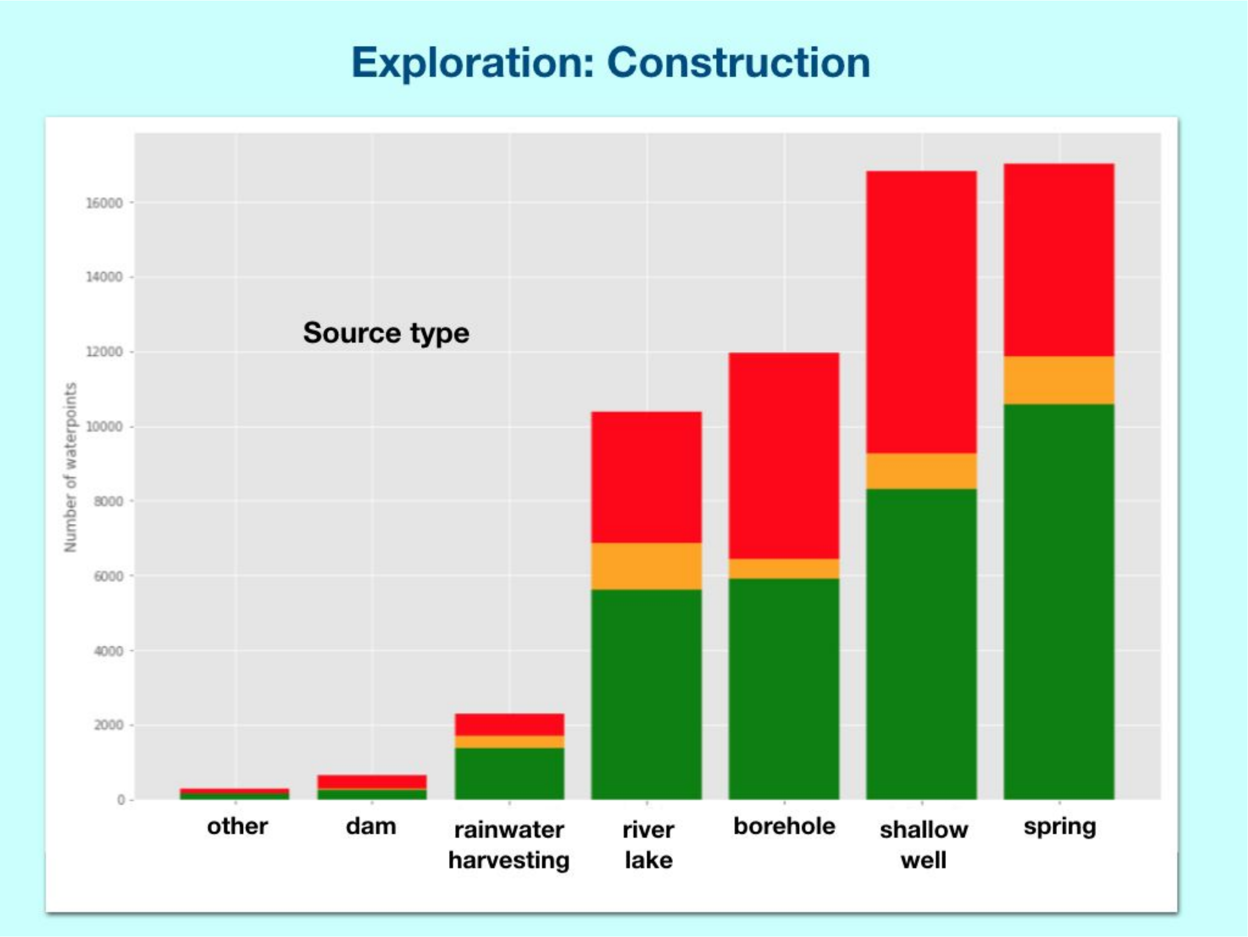
Exploration: Construction

This shows the number of new water points each year. Some values were missing and I imputed them with the mean. Older water points are more likely to be non functional.



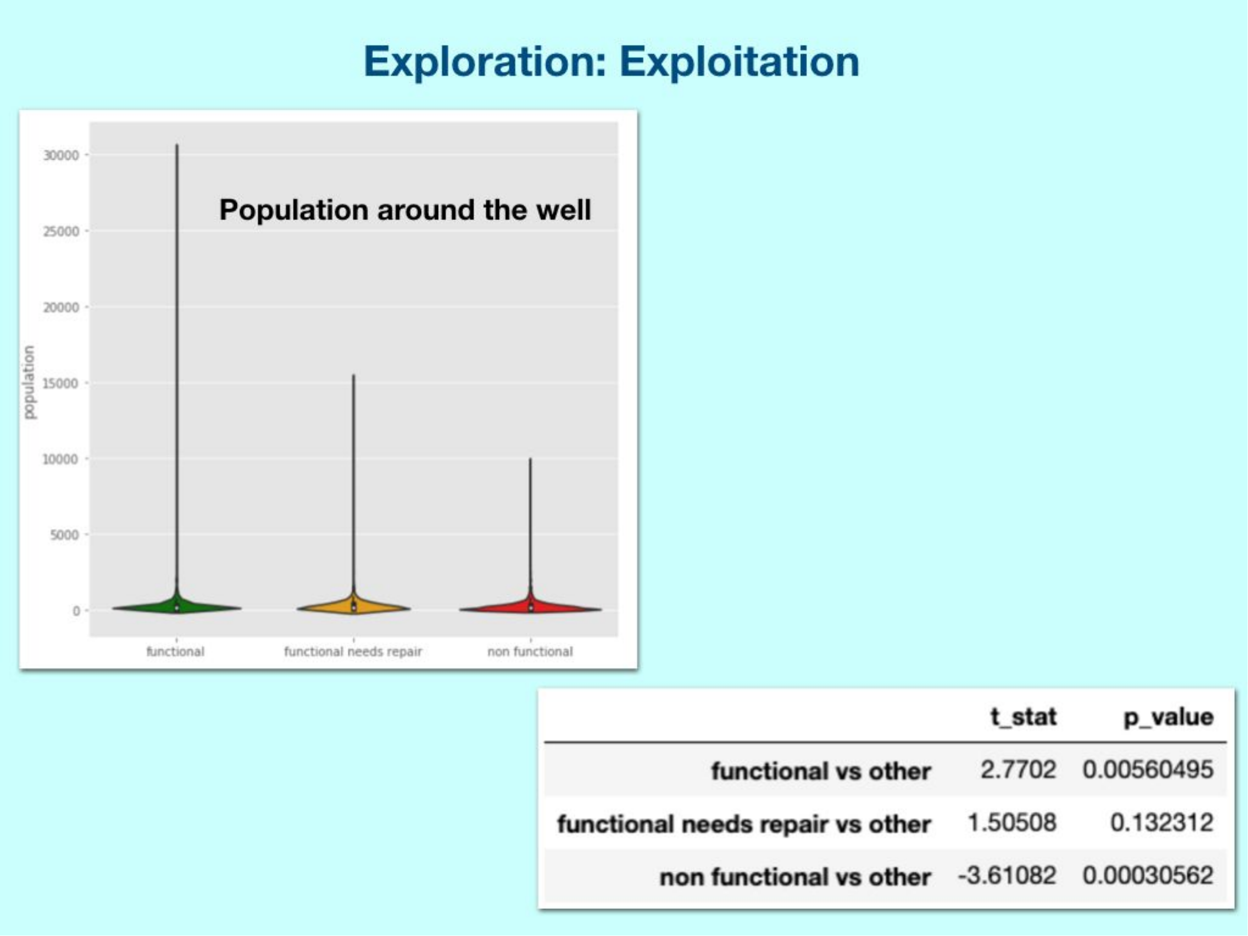
Exploration: Construction

The repartition of these extraction types are very different. Gravity pumps are more likely to be functional.



Exploration: Construction

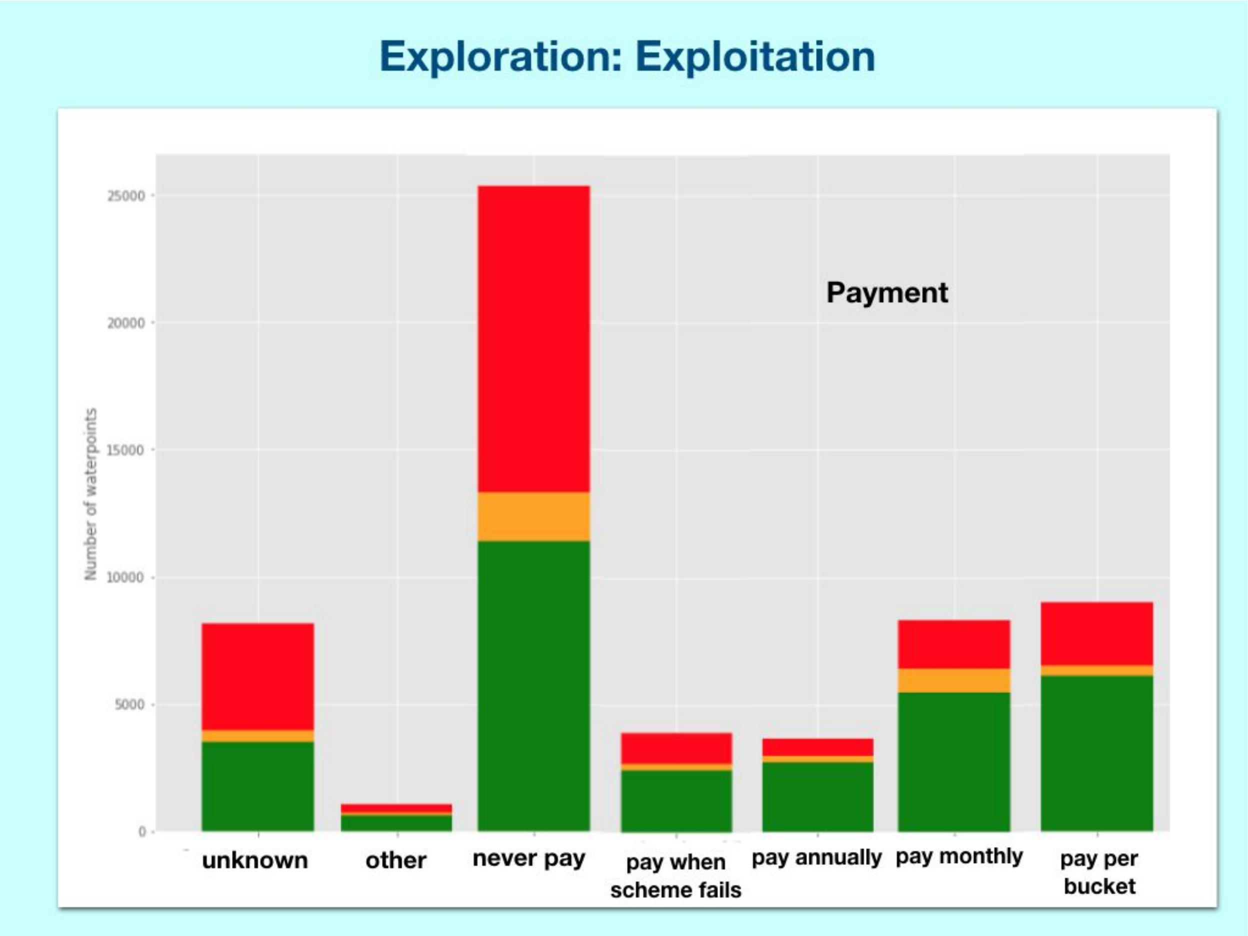
Shallow wells and bore holes are more likely to be non functional.
Spring and rainwater harvesting are more reliable.



Exploration: Exploitation

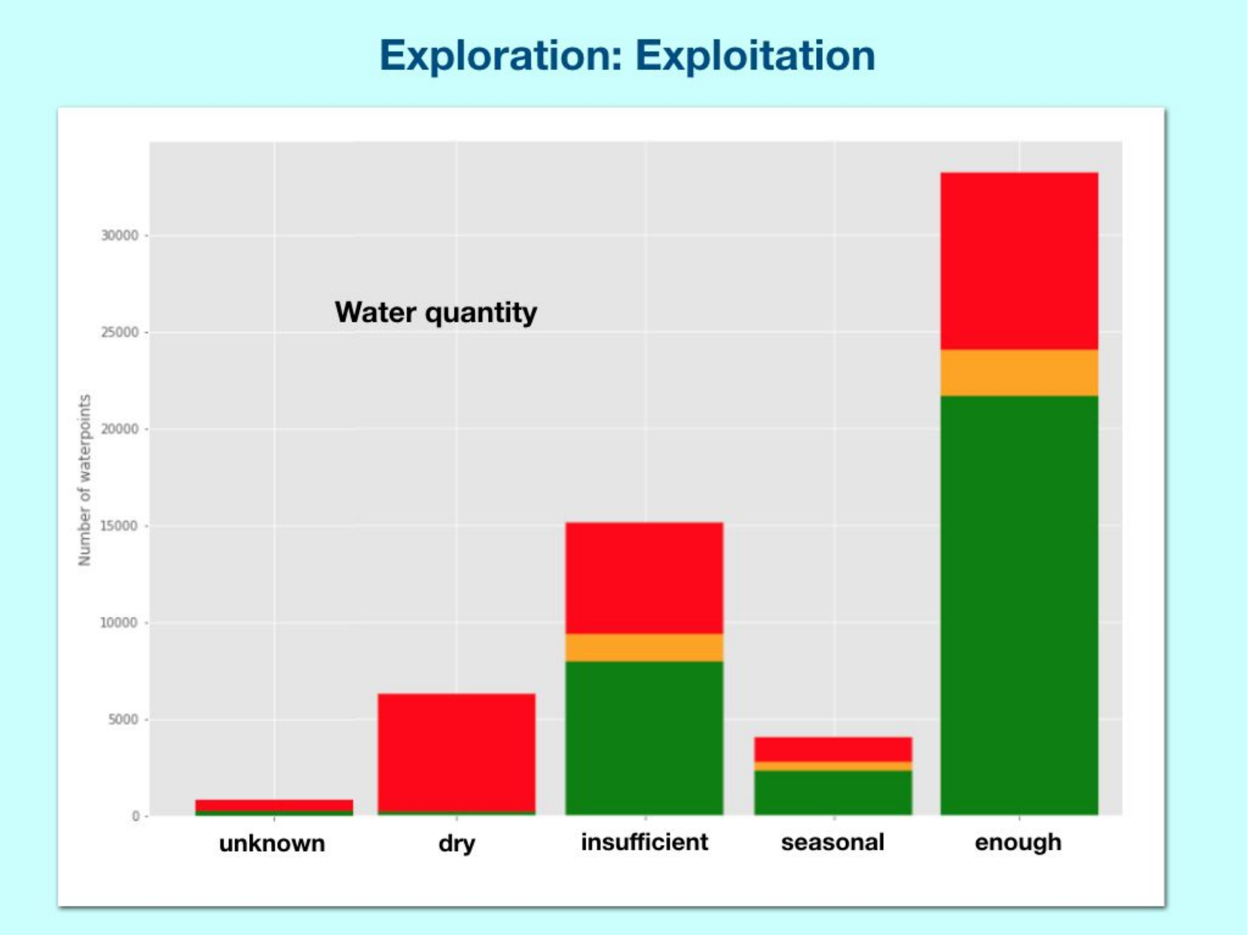
	t_stat	p_value
functional vs other	2.7702	0.00560495
functional needs repair vs other	1.50508	0.132312
non functional vs other	-3.61082	0.00030562

Hopefully, when there is a lot of people living around, the water point is more likely to be functional.
When I make a test independence of each category against the others, I see that the functional needs repair category is not significantly different. It means it will be more difficult to predict this category accurately.
This feature has a lot of missing values that I imputed with the median because I have extreme values.



Exploration: Exploitation

It seems that when there is payment for the water (either periodically, per bucket or on failure) the water point is more likely to be functional.

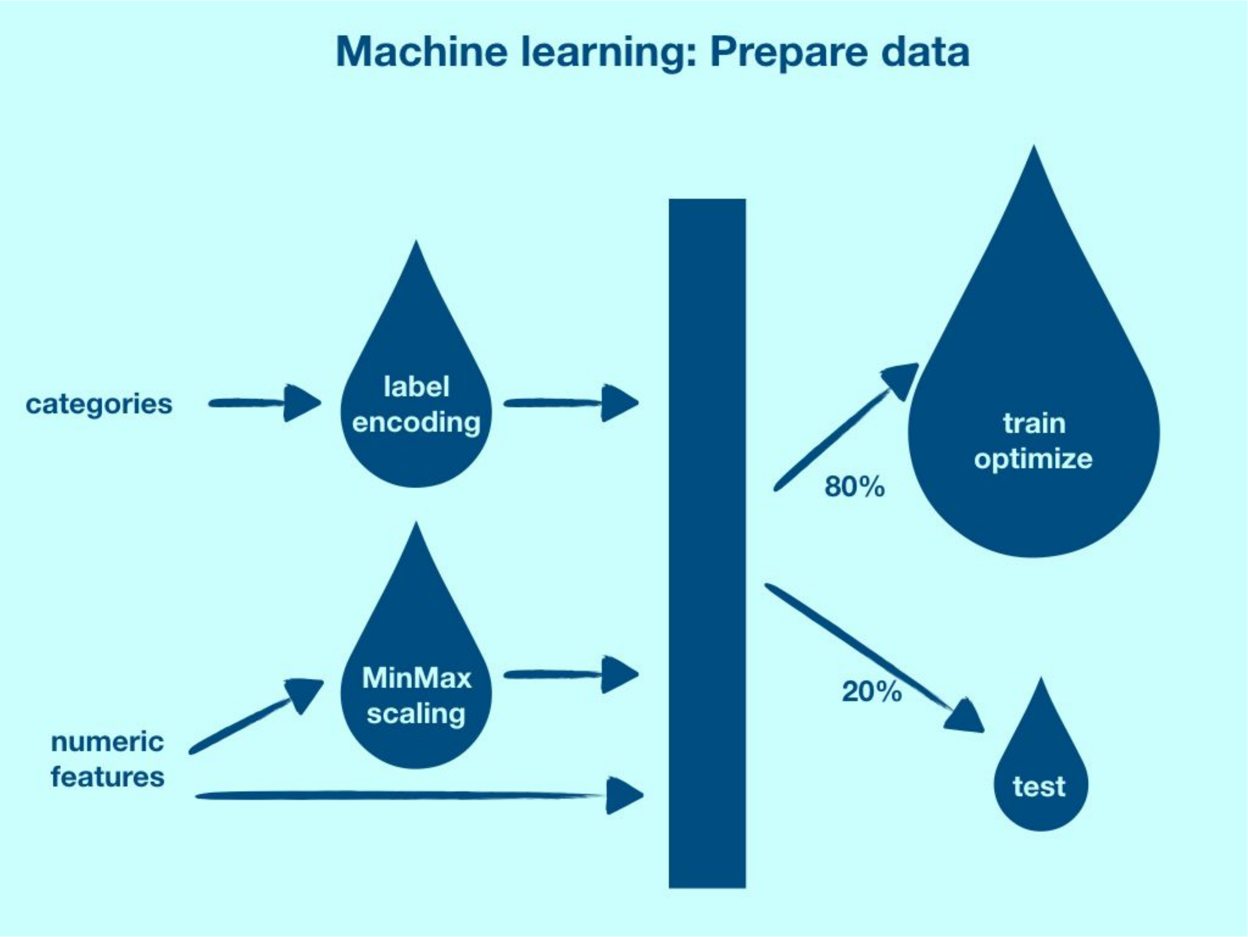


Exploration: Exploitation

As for the payment, this feature is an ordered category.
Unsurprisingly, the dry water points are more likely to be non functional.

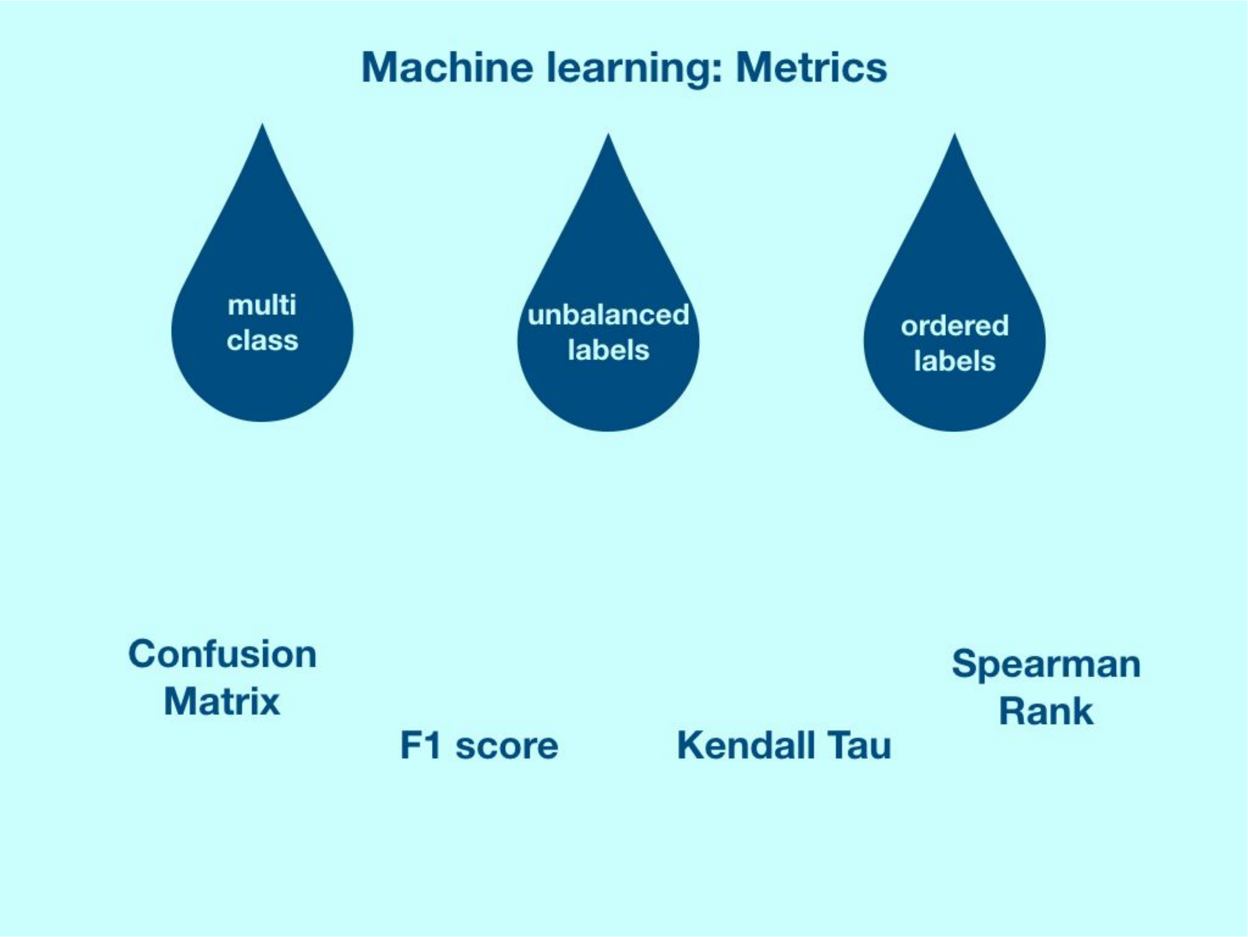
1. Exploration
2. Machine Learning
3. Synthesis

cleanwateraction.org



Machine learning: Prepare data

Before training some models, I first have to encode the categories. I choose label encoding because I have ordered categories, like payment, water quality, … and some features like funder, installer have 100 categories.
For some models, I will need to scale the numeric features. I choose MinMax scaler because I need positive values for some models.
Finally, I split the data set to work on 80% of it, and keep 20% only for test. I make sure to have the same proportion of each operational status in both groups.



Machine learning: Metrics

multi class

unbalanced labels

ordered labels

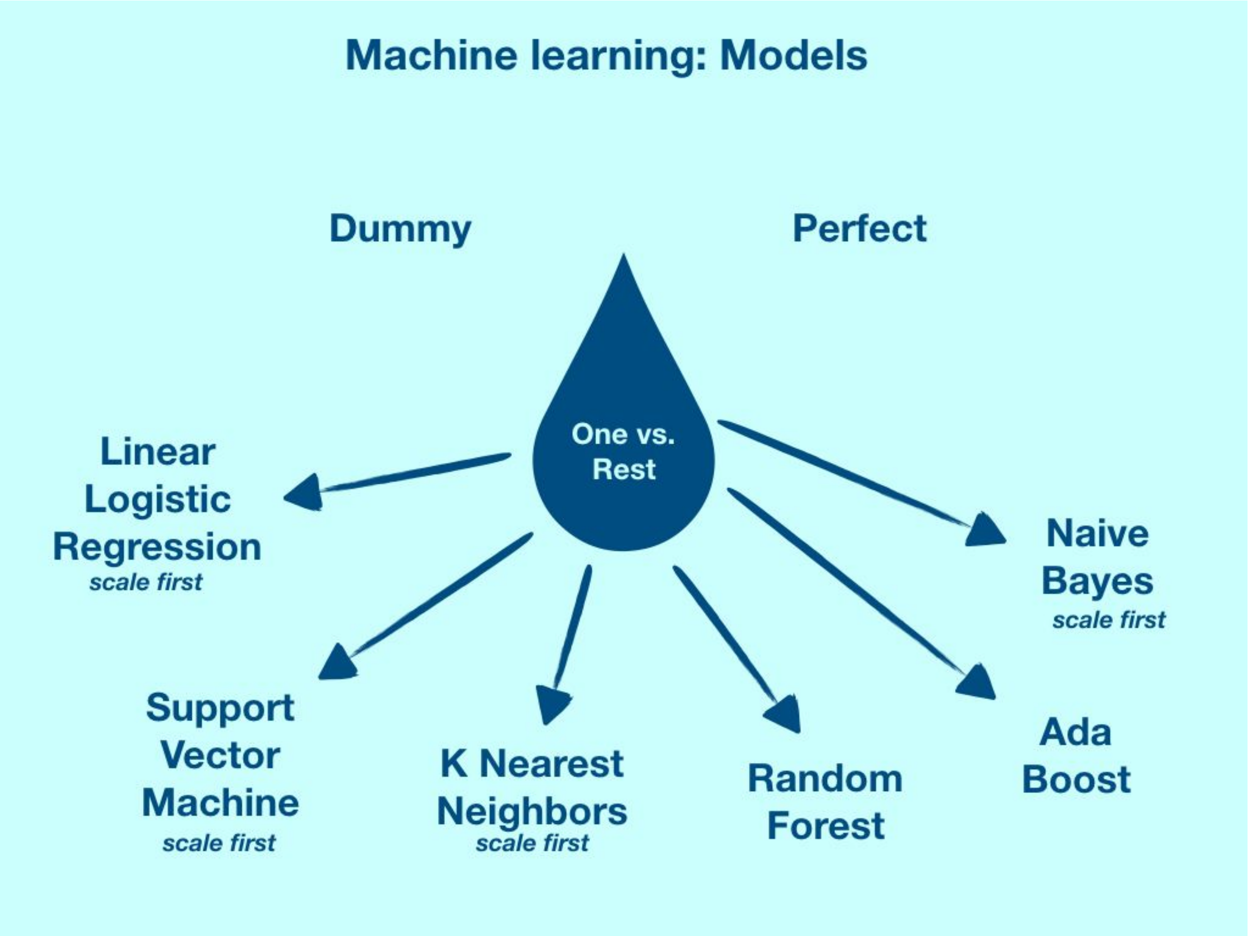
Confusion Matrix

F1 score

Kendall Tau

Spearman Rank

Because I have multi class, it’s very interesting to have a look at the confusion matrix to see how each operational status is predicted.
Because the labels are unbalanced (very few Functional needs repair), I will not use the accuracy metrics. Instead, I will use F1 score.
Because the labels are ordered, I will use Kendall Tau and Spearman rank. They measure correspondence between two rankings.



Machine learning: Models

Dummy

Perfect

One vs. Rest

Linear Logistic Regression *scale first*

Support Vector Machine *scale first*

K Nearest Neighbors *scale first*

Random Forest

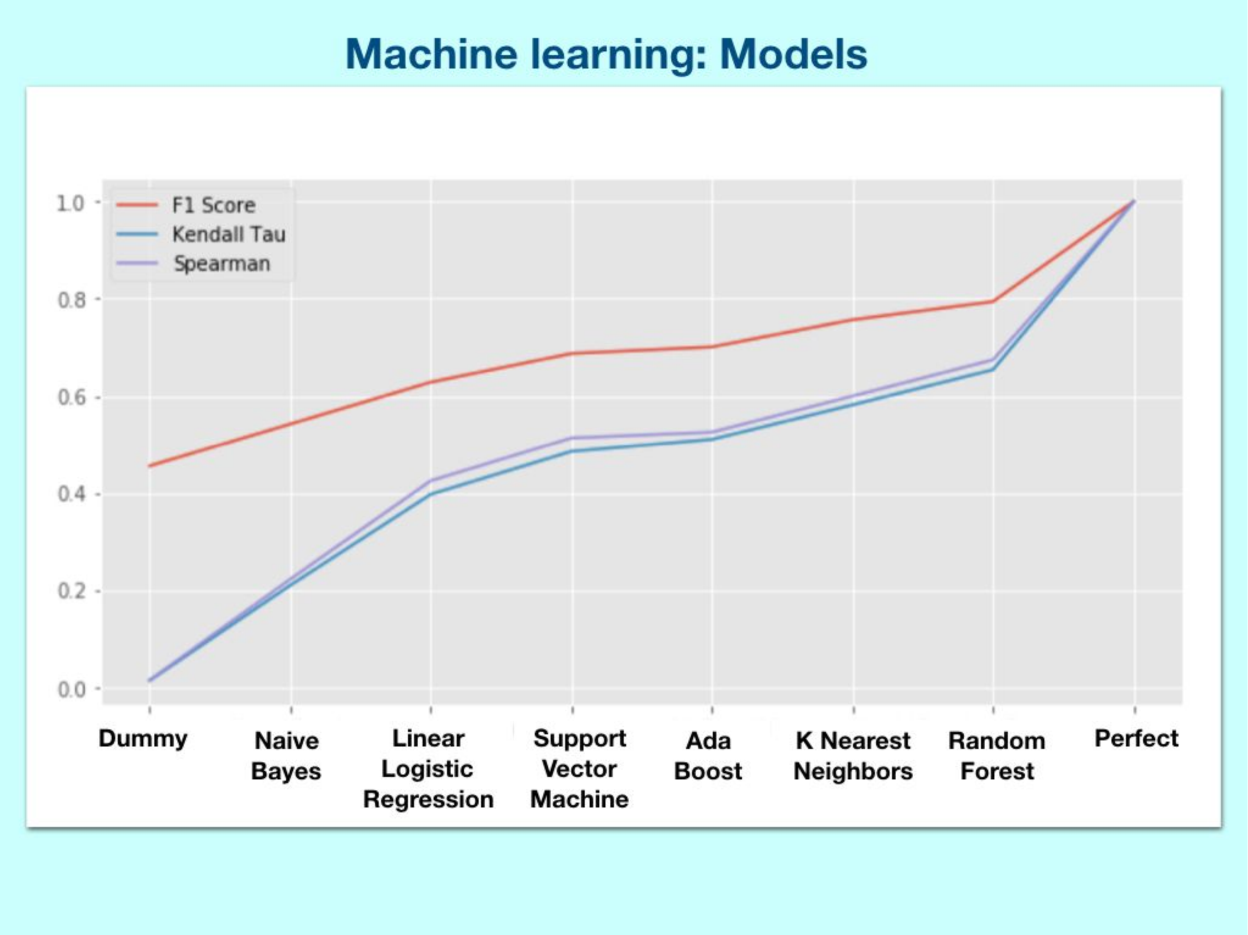
Ada Boost

Naive Bayes *scale first*

To be able to rank my models, I will first compute metrics for a dummy model that chooses labels randomly but with the same proportions as the training data, and compute metrics for a perfect model that is right all the time.
I will use One vs Rest approach: It isolates one label against the others, and trains 3 models.
I will try different classifiers with this approach:

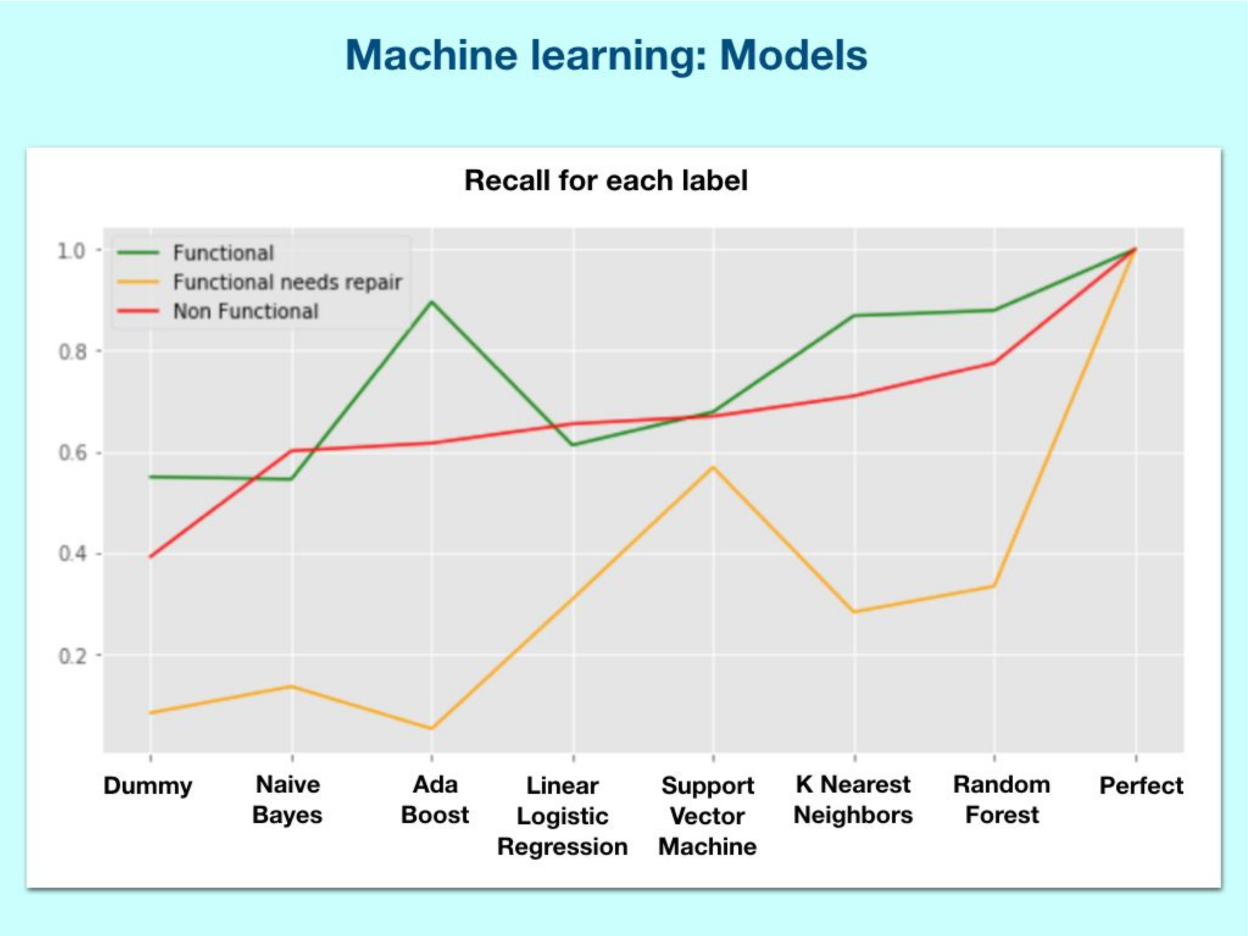
- Linear Logistic Regression because it's a simple model.
- Support Vector Machine because it defines a boundary between labels, and maybe able to better classify the functional needs repair that lies between the two other labels.
- K Nearest Neighbors for the same reason. It defines clusters of same labels water points and maybe able to better recall the functional needs repair.
- Random Forest because it usually performs well.
- AdaBoost is a kind of Random Forest that gives more weight to misclassified elements, so it may be able to improve the recall of functional needs repair water points.
- Naive Bayes because it can be good with categorical features. It needs scaling to have only positive values.

The other models that need scaling use distance to classify.



Machine learning: Models

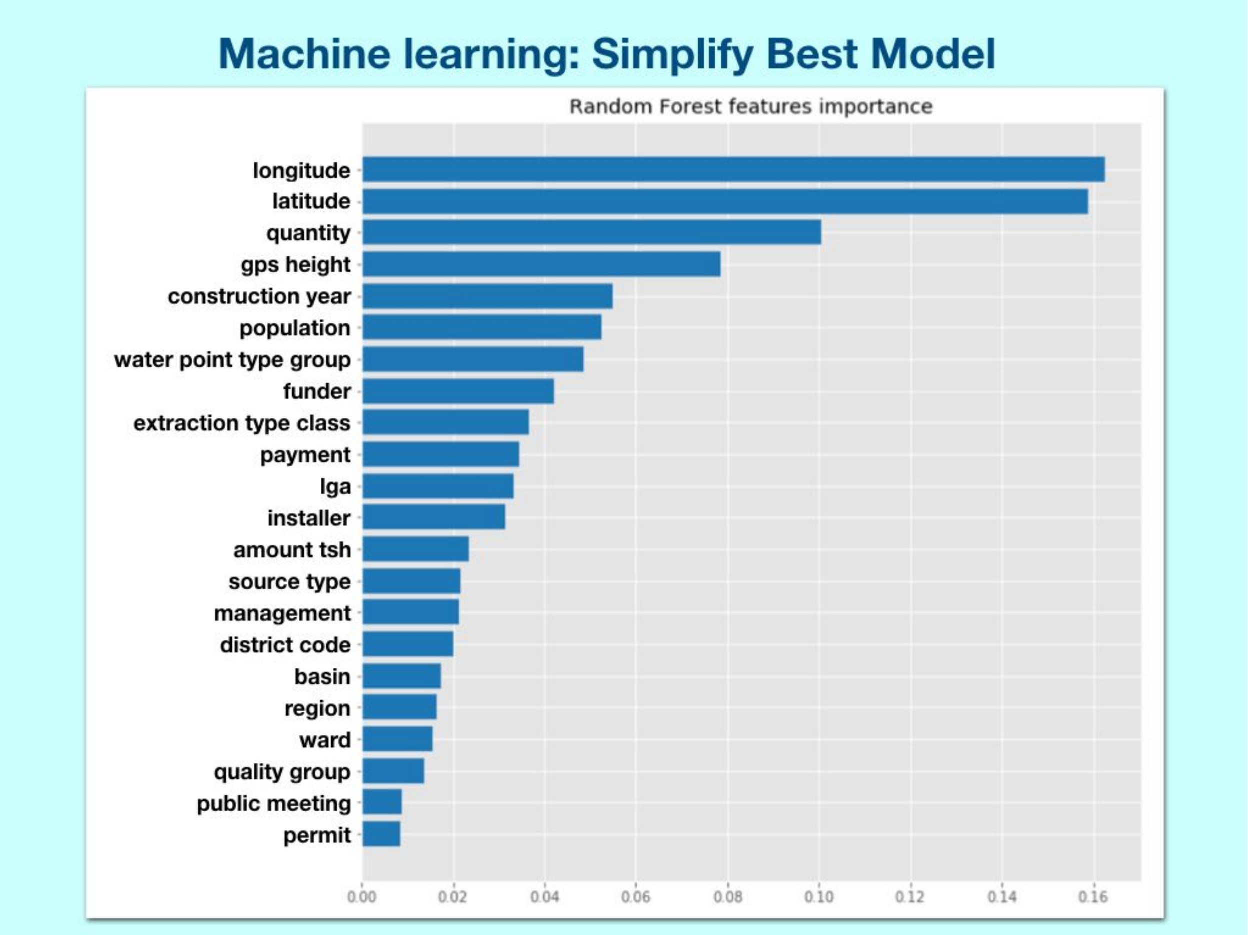
F1 score can only be positive. It’s close to 0.5 with the dummy model, and 1 with the perfect model.
Kendall Tau and Spearman are correlation measures, so they are close to 0 when there is no correlation, and close to 1 when there is positive correlation.
The best model is Random Forest, with 0.8 F1 score.



Machine learning: Models

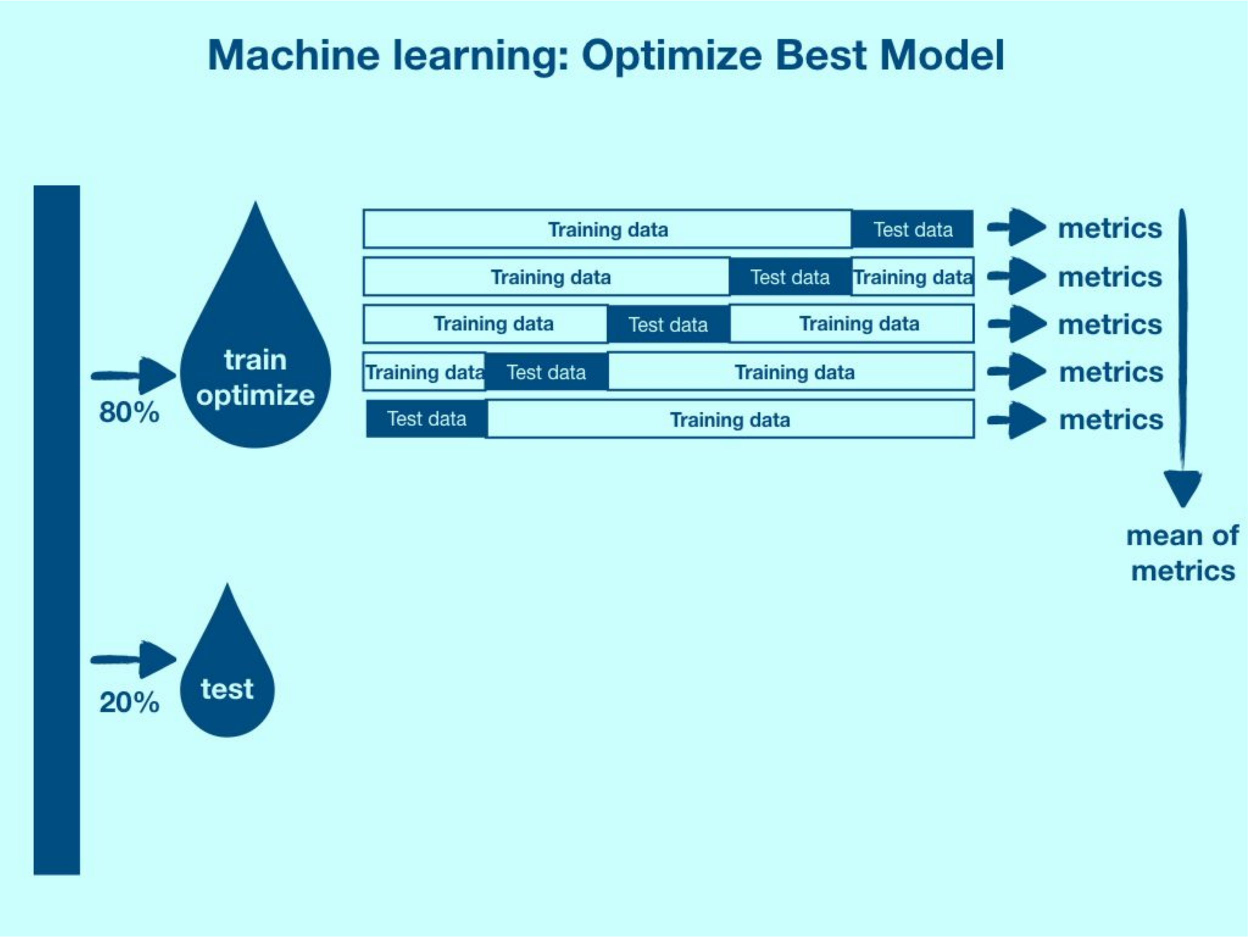
Recall for each label

This plot shows the recall (proportion of accurate prediction for each label).
As expected, functional needs repair is more difficult to predict.
We can see that SVM recalls the functional needs repair water points better than Random Forest, but at the expense of a bad precision. Overall Random Forest is better.



Machine learning: Simplify Best Model

Let’s have a look at the Random Forest features importance.
The most important features are longitude, latitude, quantity and gps_height.
The least important features are public_meeting and permit. I will drop these features to simplify the model.



Machine learning: Optimize Best Model

80%

train optimize

Training data | Test data → metrics

Training data | Test data | Training data → metrics

Training data | Test data | Training data → metrics

Training data | Test data | Training data → metrics

Test data | Training data → metrics

mean of metrics

20%

test

To optimize the Random Forest model, I tune its parameters.

I use cross validation. I use the training data and for each parameter, I will train and test the model 5 times, each time using 80% of it for training and 20% for testing. Then, I take the mean of the metrics over the 5 tests.

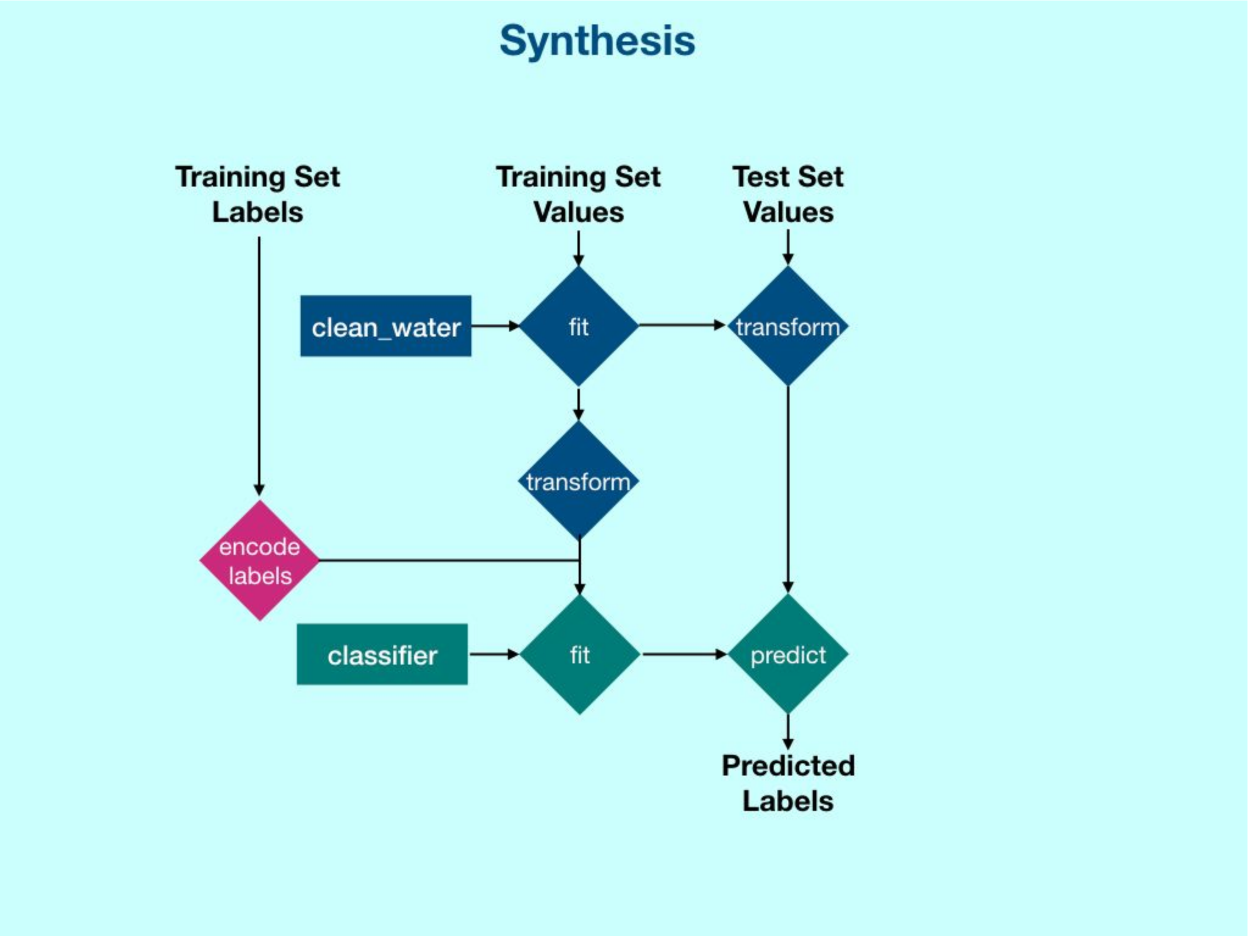
Using this technique, I optimize n_estimators: number of trees in the forest, and max_depth, the depth of the trees in the forest.

1 Exploration

2 Machine Learning

3 Synthesis

cleanwateraction.org

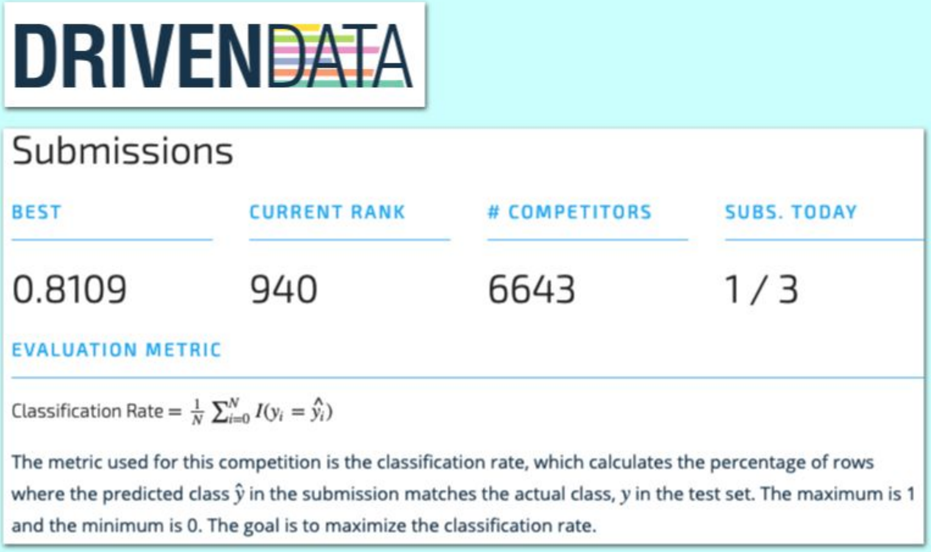


Now that I known how to process the data and which model to use, I synthesize the snippets of code for data cleaning, model training and prediction.
This will allow me to easily work on any training and test datasets.
I can train the model on all the available data set, and then predict the labels of the validation set.
I build a class clean_water with methods:

- fit to get the the means, medians, and most frequent values that I will need
- transform to clean the dataset and encode values

The training set labels are encoded as well.
Then, the classifier OneVsRest Random Forest with optimized parameters is fitted on all the training data set.
Finally, the test set is cleaned and sent to the fitted model to predict labels.

Synthesis



Submissions

BEST	CURRENT RANK	# COMPETITORS	SUBS. TODAY
0.8109	940	6643	1 / 3

EVALUATION METRIC

Classification Rate $= \frac{1}{N} \sum_{i=0}^{N} I(y_i = \hat{y}_i)$

The metric used for this competition is the classification rate, which calculates the percentage of rows where the predicted class $\hat{y}$ in the submission matches the actual class, y in the test set. The maximum is 1 and the minimum is 0. The goal is to maximize the classification rate.

The best score is 0.8286

Once the labels were predicted, I sent them to Driven Data to get a score.
Here are the performances of this model.

Conclusion

- Real data needs cleaning before using it: understand the data, then impute or drop missing data.
- Geography, construction and operation data influence the operational status of the water point.
- The best model to predict the operational status is One vs rest Random Forest.
- Creating cleaning class and associated methods makes it easy to train and predict on any dataset.
- The score of this model is 0.8109

Conclusion

Ideas to improve the model:

- improve data cleaning and features selection
- try using bootstrap to create more ‘functional needs repair’
- try one-hot encoding
- try One vs One Classifier

flickr.com