Pump it Up:
Data Mining the Water Table

Prepared by: Kai Luo

Introduction

- Background
- Data
- Model Selection
- Features Engineering
- Model Performance
- Recommendations & Future Work

Background

- https://www.drivendata.org/competitions/7/pump-it-up-data-mining-the-water-table/
- Can you predict which water pumps are faulty?
- Using data from Taarifa and the Tanzanian Ministry of Water, can you predict which pumps are functional, which need some repairs, and which don't work at all? This is an intermediate-level practice competition. Predict one of these three classes based on a number of variables about what kind of pump is operating, when it was installed, and how it is managed. A smart understanding of which waterpoints will fail can improve maintenance operations and ensure that clean, potable water is available to communities across Tanzania.

Data

<u>**Feature**</u>

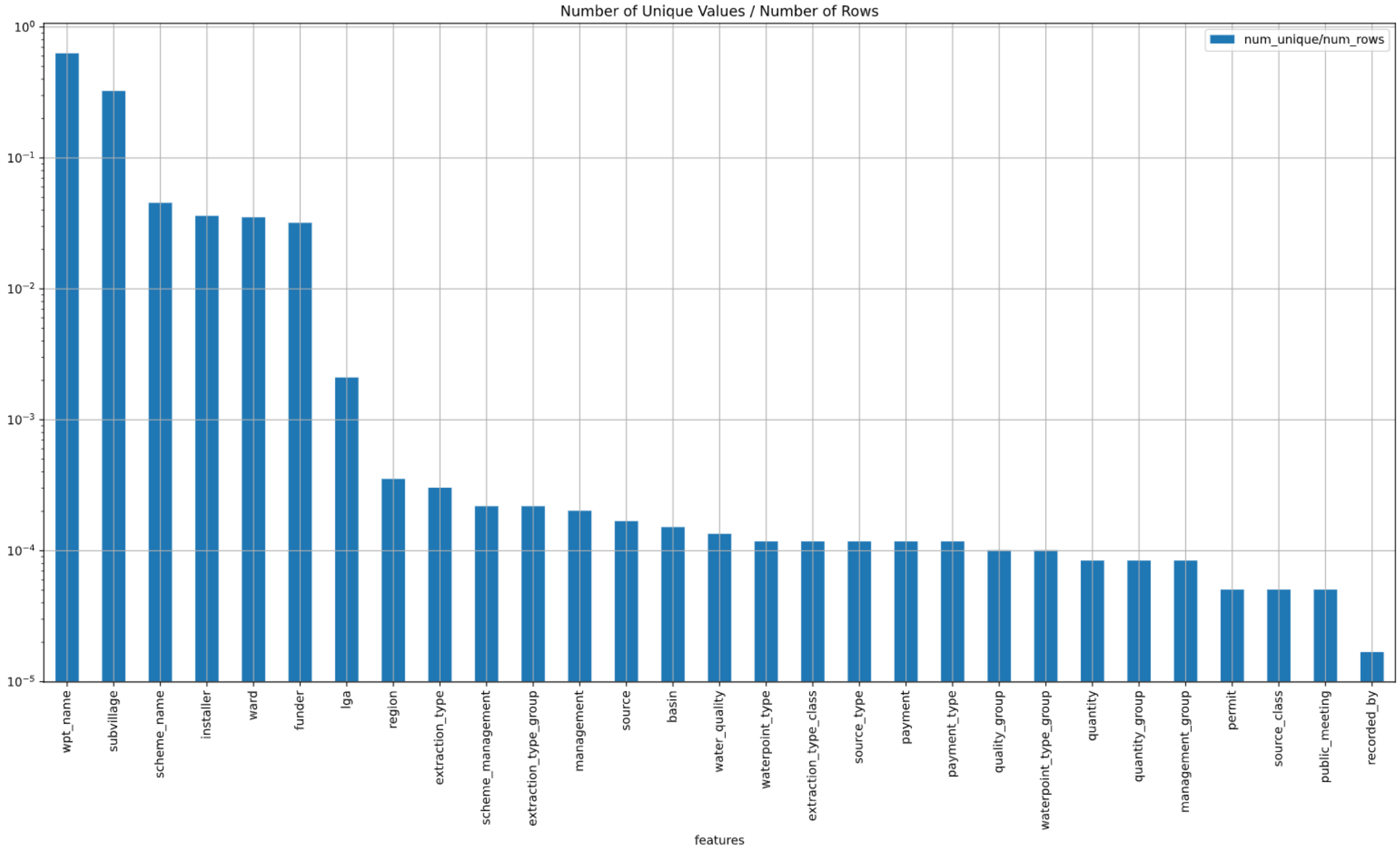
- **amount_tsh** - Total static head (amount water available to waterpoint)
- **date_recorded** - The date the row was entered **(not important)**
- **funder** - Who funded the well
- **gps_height** - Altitude of the well
- **installer** - Organization that installed the well
- **longitude** - GPS coordinate **(self identifying)**
- **latitude** - GPS coordinate **(self identifying)**
- **wpt_name** - Name of the waterpoint if there is one **(self identifying)**
- **num_private** -
- **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

<u>**Labels**</u>

- **functional -** the waterpoint is operational and there are no repairs needed
- **functional needs repair -** the waterpoint is operational, but needs repairs
- **non functional -** the waterpoint is not operational

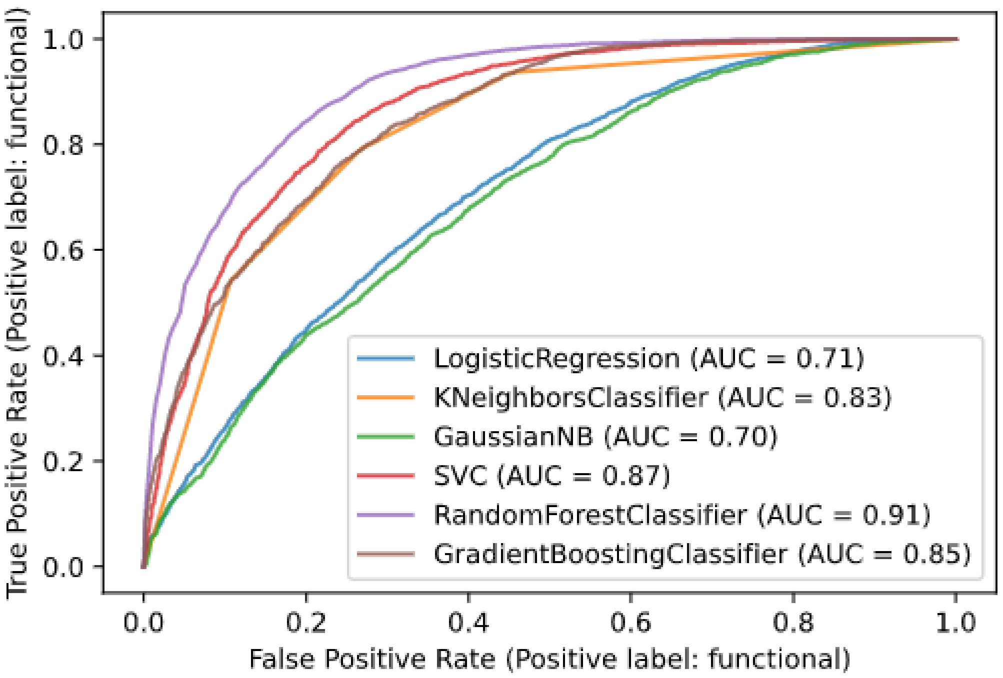
Data: Categorical Features Statistics

- These are the categorical features . There are numerical features as well.
- There are few features (the left most ones) that have high uniqueness.
  - Potentially these are self identifying features.
  - Also adds complexity to the model



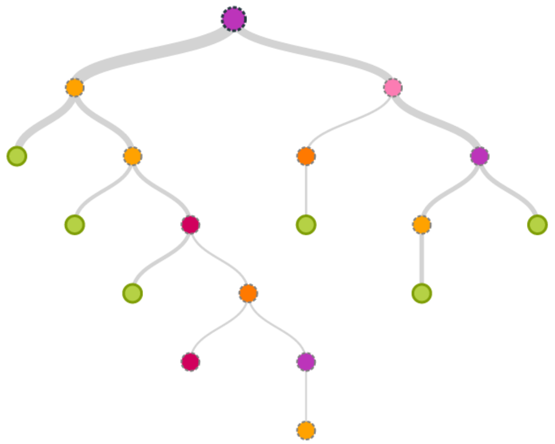
Model Comparison

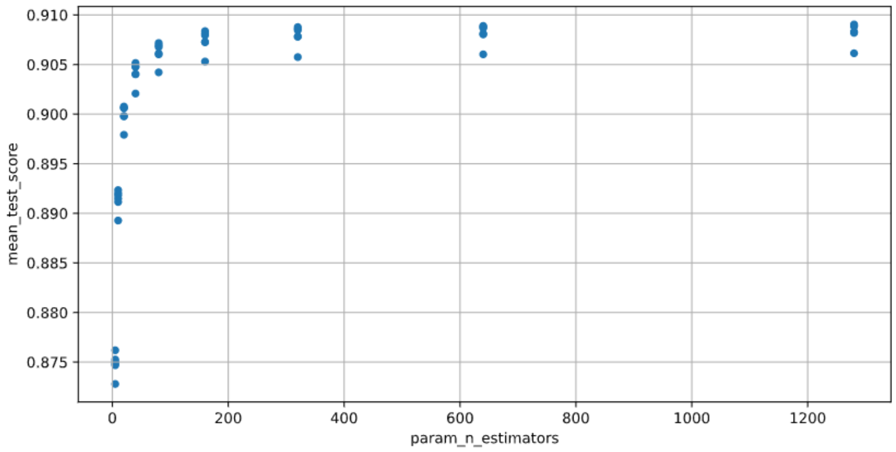
- For the sake of model comparison, since some of the models only works with binary labels, “functional needs repair” and “not functional” were combined.
- AUC = Area under Receiver Operating Curve
  - 1 is best
  - 0 is worse
- Default settings, without optimization
- Random Forest Classifier seems like the best performing.
  - Thus optimization and further exploration are focused on Random Forest
  - Output: probability of [f, fnr, nf]
    - f=functional, fnr=functional needs repair, nf=non function
  - i.e. [75%,20%,5%]

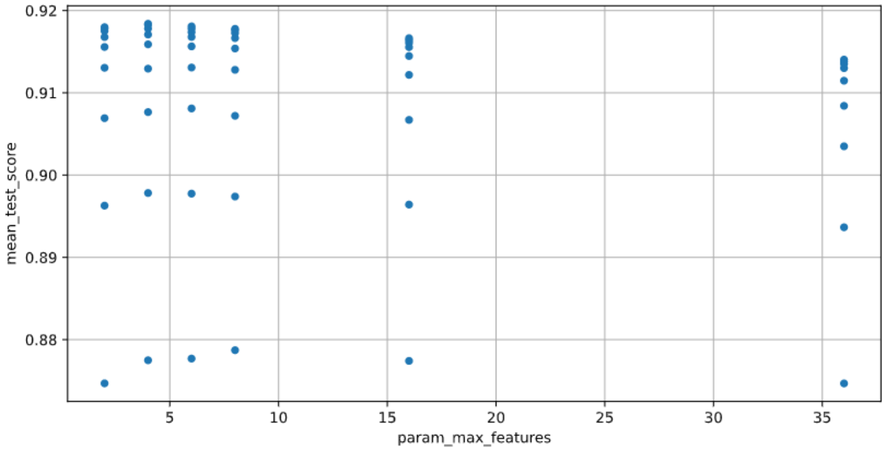


Parameters Selection

- test_score = AUC
- Random Forest Models are iterated over:
  - n_estimators: how many trees are in a forest.
  - max_features: how many features will be considered when splitting
- Performance doesn’t improve much over simpler models.

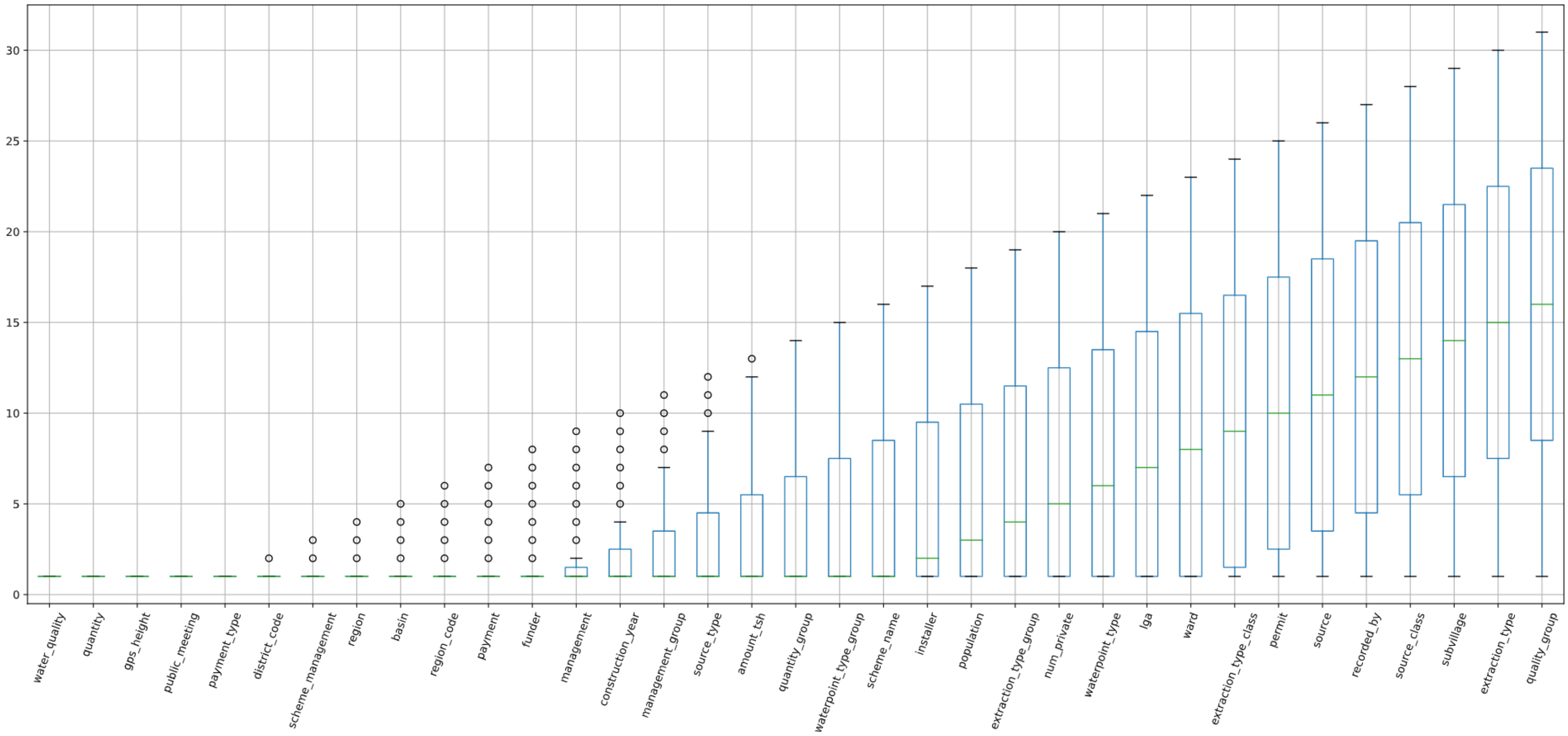






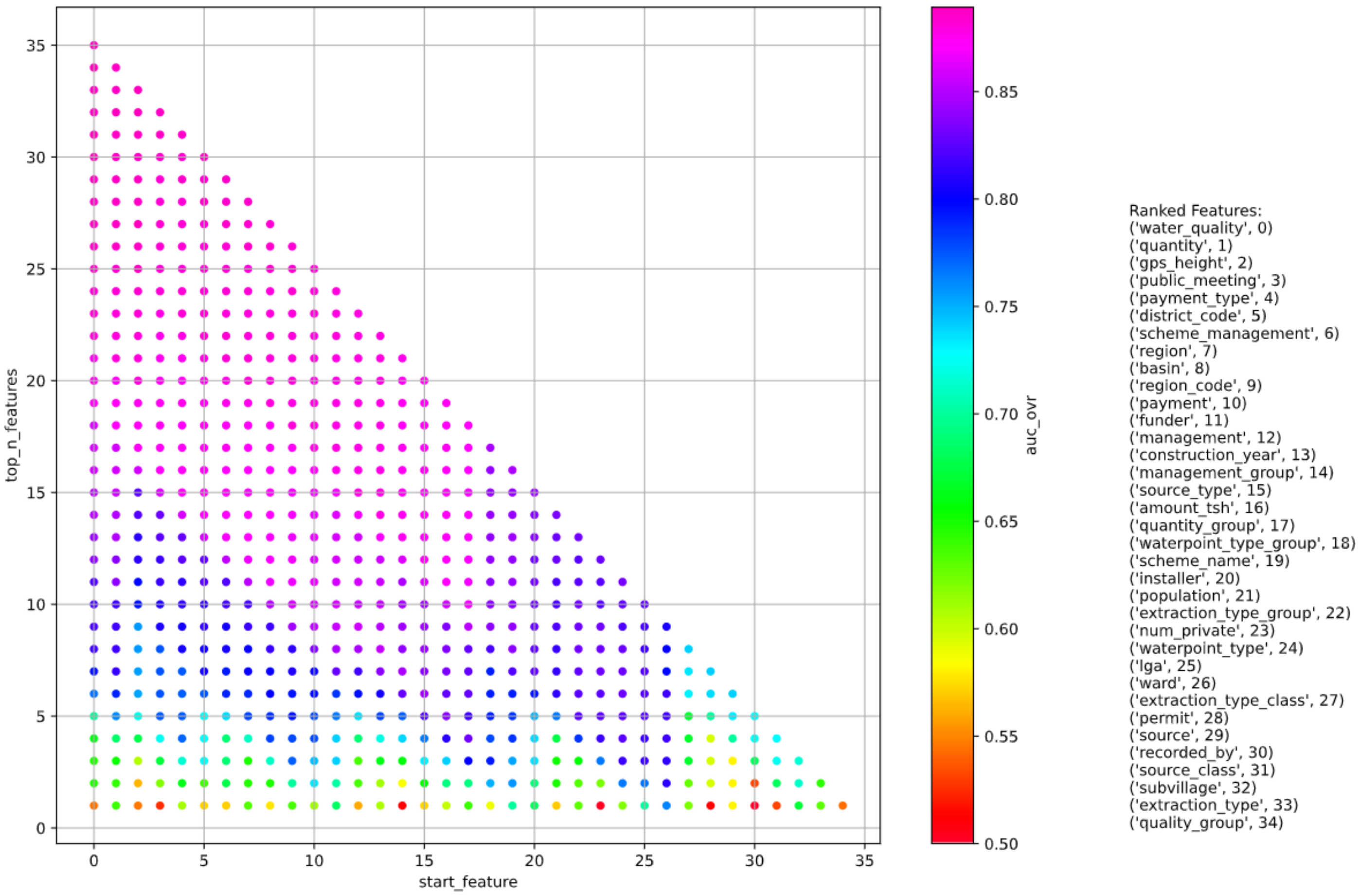
Features Ranking

Many random forests are made using different settings on the Recursive Feature Elimination algorithm and the features are ranked. The boxplot shows statistic summary of how they rank the features:



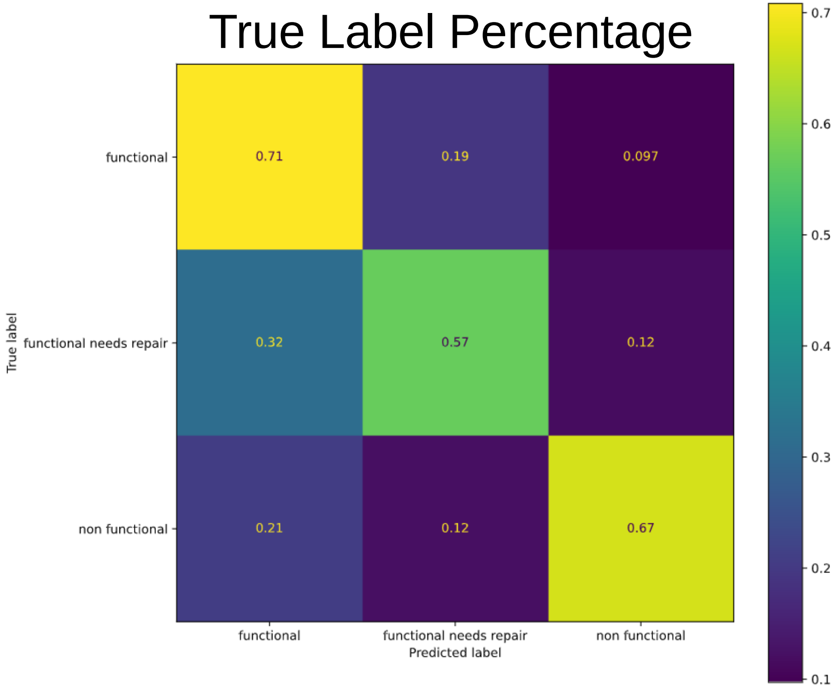
Feature Selection AUC

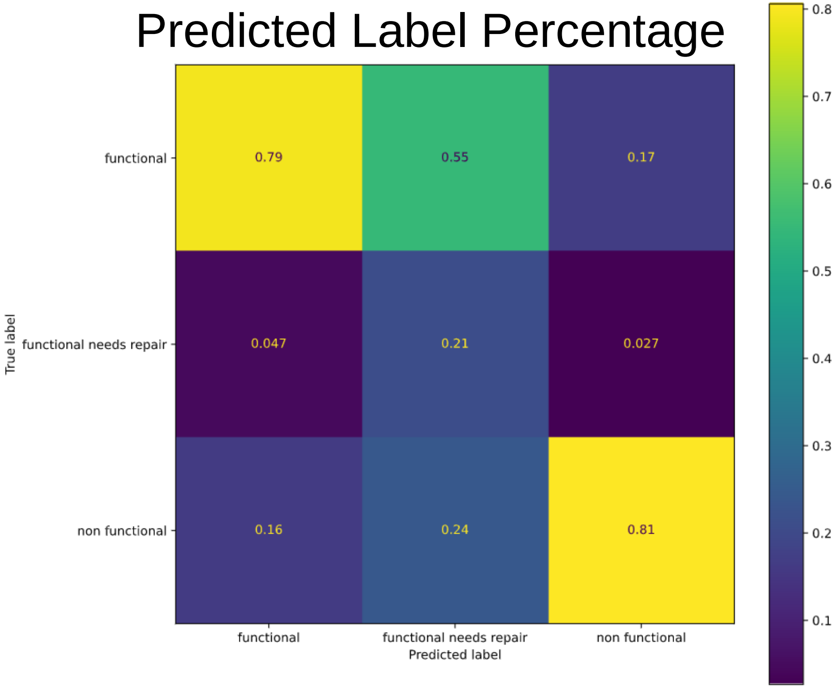
- AUC score of Random Forests with different start_feature and top_n_features are ploted.
- Note that it that the best ranked feature may not guarantee the best AUC score.
- Also note that lower top_n_features used may not necessarily be the worse performance model.



A Simple Model’s Performance

- Settings: start_feature=15, max_features=6, n_estimators=2. Selected to be simple to prove model performance doesn’t require extreme complexity.
  - AUC = 0.82
- It is most critical that when true “non functional” is predicted to “functional”.
- Also critical when true “functional needs repair” and predicted to be “functional”
- True Label Percentage:
  - This plot adds up to 1 row-wise.
  - 21% of true “non-functional” will be predicted to be “functional” by the model. 12% true “functional needs repair” will be predicted to be “functional”.
- Prediction Label Percentage:
  - This plot adds up to 1 column-wise.
  - 16% of predicted “functional” will be predicted to be true “non functional” by the model. 24% predicted “functional needs repair” will be predicted to be “non functional”.





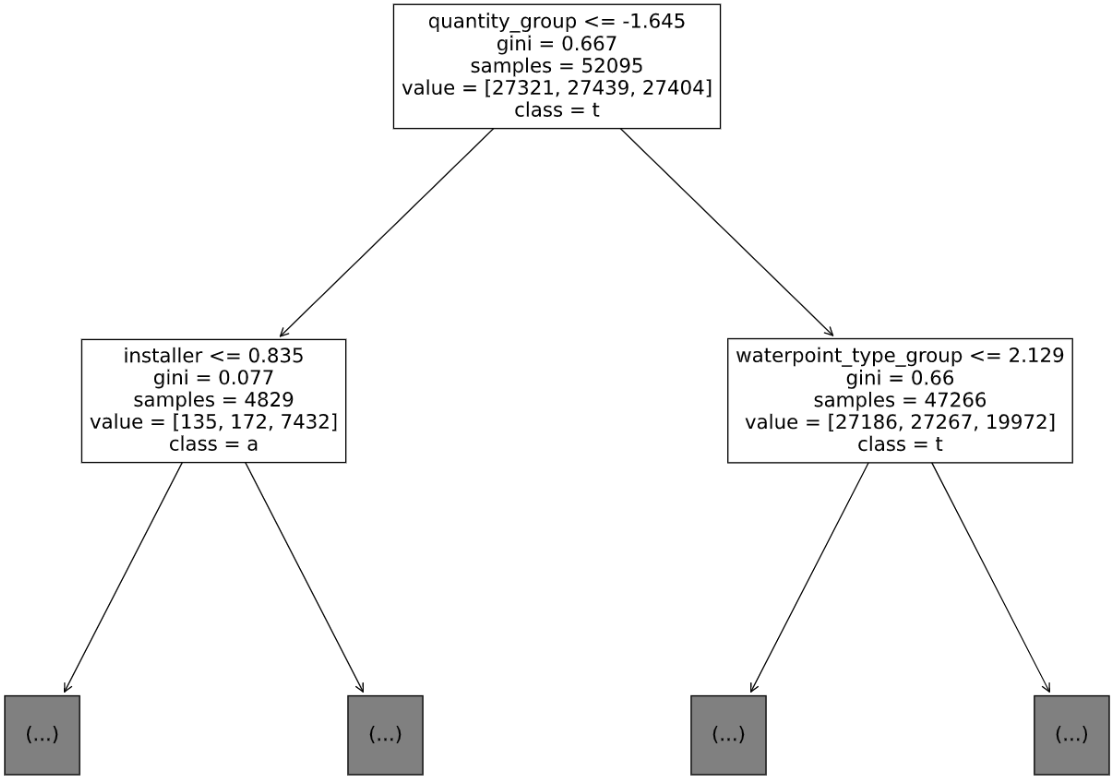
Recommendations and Future Works

1) Build a simpler model while real time updates the model. Could train multiple simple model based on date intervals as well.

2) Develop a categorical random forest library and visualization tools to further invest the correlation of “functional”, “functional needs repair”, and “non functional”. This can also help with the root cause investigation and physical prevention method development of the water pumps and wells.

3) The goal should be lower the false “functional” rate, since this slows the true “non functional” or “functional needs repair” replacement or repair process.

Attempt to Visualize Decision Tree



Summary

- Using water well pump data to predict the pumps are “functional”, “functional needs repair”, or “non functional”.
- Random Forest is the best model.
- Model performs well without too much complexity added by choosing high n_estimators and n_top_features.
- Recommendation:
  - Use simpler model to real time update model as more data come on. Could train multiple simple model based on date intervals as well.
  - Develop libraries and visualization tools for categorical Random Forest Classifier
    - To later help with root cause investigation and mitigation plan design.
  - The goal should be lower the false “functional” rate, since this slows the true “non functional” or “functional needs repair” replacement or repair process.

Thank you for you attention!

Any questions?