Water Pumps

Steven Gusenius, Zuber Saiyed, Margarita Linets

About this Project

Using data from Taarifa and the Tanzanian Ministry of Water, we set out to predict where water pumps were likely to be functional, in need of repair of not functional at a certainl locale.

A smart understanding of which water pumps will fail can improve maintenance operations and ensure that clean, potable water is available to communities across Tanzania.

More information about the challenge and the dataset can be found here - https://www.drivendata.org/competitions/7/pump-it-up-data-mining-the-water-table/page/23/

About the Dataset

The datasets for this project were downloaded from www.drivendata.org and consisted of a two files of comma separated format. This first file contains 40 characteristic data of each water pump, indexed by a pump ID, to be used as predictors. A list of these predictors is provided in *APPENDIX A*.

The second file contains the status_group for each water pump, also indexed by pump ID. The status_group is the response we are attempting to predict and indicates the condition of a water pump. Its value can be: Functional (F), FunctionalNeedsRepair (FNR), or NonFunctional(NF). The respective percentages of each are: 54.3%, 7.3%, 38.4%.

In total, there is data for 59,400 water pumps.

Data Cleaning

Data Modification Initially the datasets were cleaned to make them compatible with processing. Primarily this consisted of addressing missing data and special characters. Then the predictor data was merged with the response data into a single dataset.

Data Excluded Following the merge, the pump ID was eliminated as it is not a meaningful predictor. One predictor, **recorded_by** was excluded because it had minimal variation for all water pumps. Several other categorical predictors were eliminated for having an excessive number of (greater than 30) levels. A list of these factor variables, and their associated number of levels, is available in *APPENDIX B*. This step was needed when Lasso was used. This because Lasso requires the inputs to be of type *model.matrix*. A model matrix creates a separate column of data for each level of each factor variable. This has a detrimental impact on both memory requirements and processing speed. In this case, the retention of all such factor variables exceeded the capacity of the R software. Further, it is a reasonable assumption that if a large proportion of the data is spread across many nominal factor levels, that factor variable will have diminished predictive power.

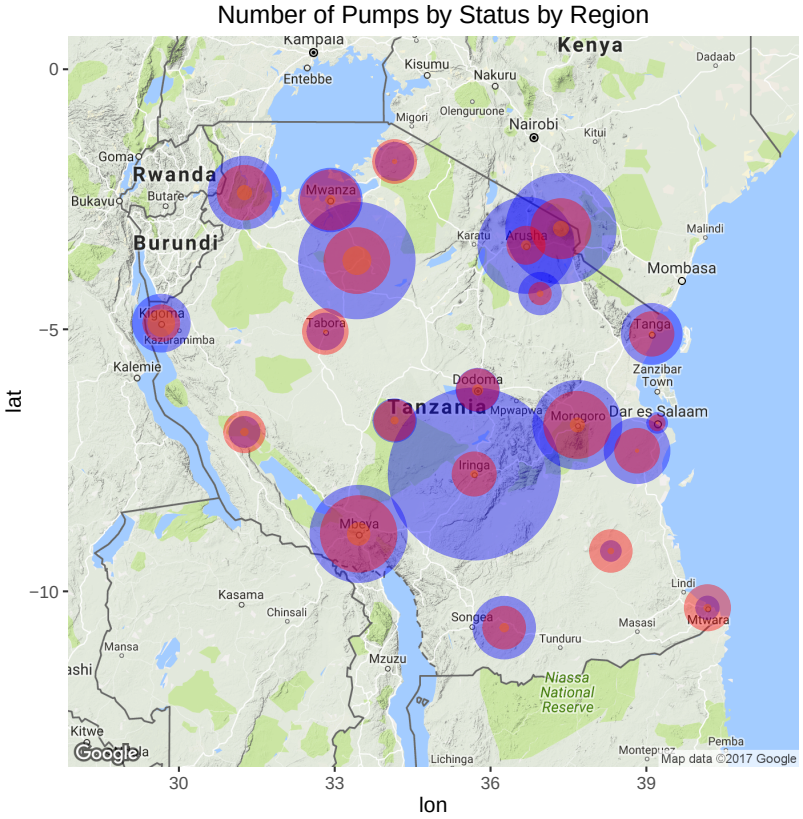
Data Exploration

Prior to model fitting, some effort was invested in understanding the content of the data. Various hypotheses were made and then evaluated through a number of simple, ad hoc analyses.

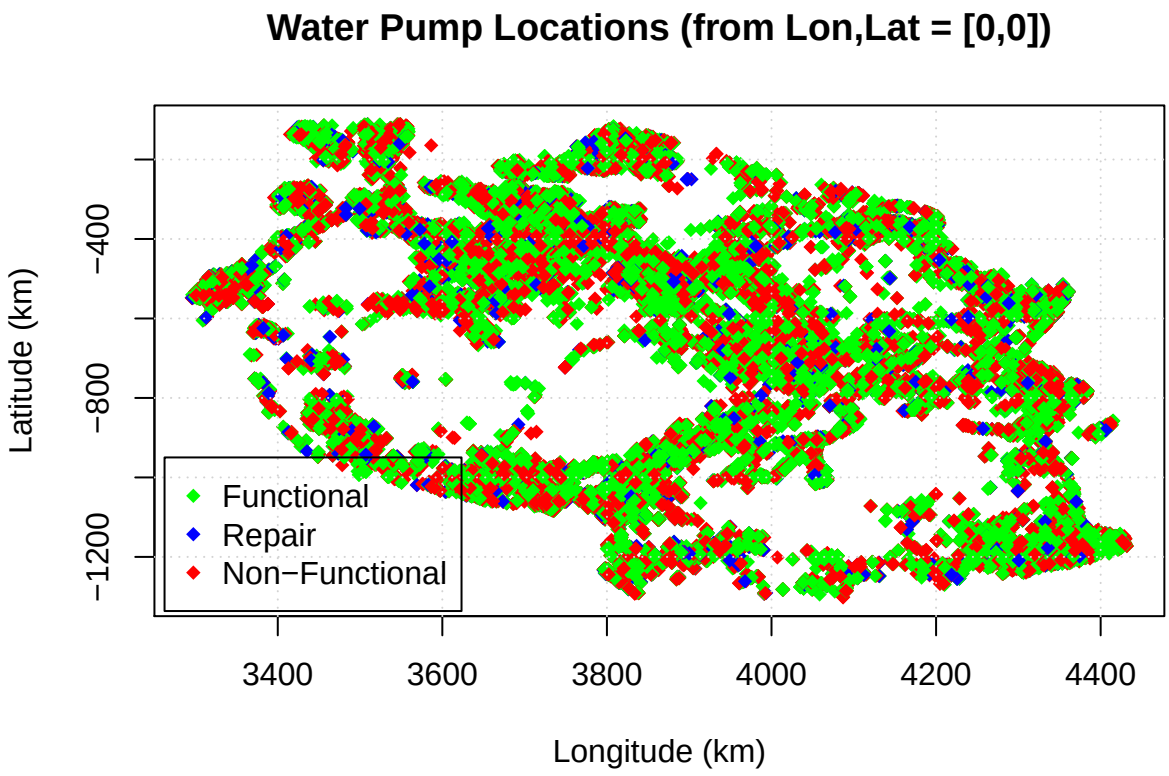
One such analysis was a data visualization where the frequency of the three **status_groups**, for each **region**, was plotted at the center of the respective region on a map of Tanzania. This map provide an understanding of how pump functionality was dispersed throughout the country, and serves as an indication of how many

water pumps were contained in each region and whether each **region** had similar proportions of F, FNR, and NF water pumps.

Based on this map, it appears that districts with the fewest water pumps might have a larger number of pumps that are NF. For this reason a variable **regionalPumpCount** was added to the dataset.



Additionally visual analysis was performed by plotting the position of each water pumps, color coded by **status_group**. A simple spherical earth transformation allowed the water pump positions in longitude and latitude to be plotted in on a flat plane using linear units of kilometers. Given that linear units are preferable for fitting models, these transformed positions were added to the dataset as **East_km** and **North_km**. Because these would exhibit exceptionally high correlation with **longitude** and **latitude**, respectively, the latter variables were removed from the dataset.



The plot above was examined to see if there were signs of clustering among pumps of a specific **status_group**. While it did appear that there were some areas of the country with elevated proportions NF pumps, there was no recognizable pattern that could be leveraged for this evaluation. A visual comparison against a mean annual rainfall map of Tanzania (available on the internet), looked like it might exhibit correlation between areas with more rain and the location of all water pumps. A similar map of Tanzania average temperatures showed a potential positive correlate between hot temperatures and NF pumps. However, defining these relationships is beyond the scope of this effort.

Fit Approaches

For all models, cross validation was used. This consisted of separating the data into **training** and **validation** sets. The models were constructed using the **training** data, then their performances were evaluated using the **validation** data. The split between the two sets was approximately 70% **training** and 30% **validation**.

Given that relatively few of the variables contained numeric data, model approaches that utilize Euclidean distances between datapoints could not be used.

Because a small proportion of water pumps were of **status_group** FNR, some model types would ignore this state completely their predictions.

One approach for addressing this was the use of a Binary Outcome Lasso using a one-vs-one selection strategy. With this strategy, three **sub-models** were built. Each sub-model was assigned a level of the response variable. The remaining two levels were given the value of **other**. This forced the sub-model to focus on fitting only its assigned level. The outcome of each sub-model was an estimated probability that the each datapoint belonged to the assigned level. Each point was assessed against these three sets of predictions. The level with the highest probability was selected.

Given their general suitability for datasets of this nature, Random Forest and Random Forest with Boosting were also used.

Results

Binary Outcome Lasso

Table 1: Training Data Performance

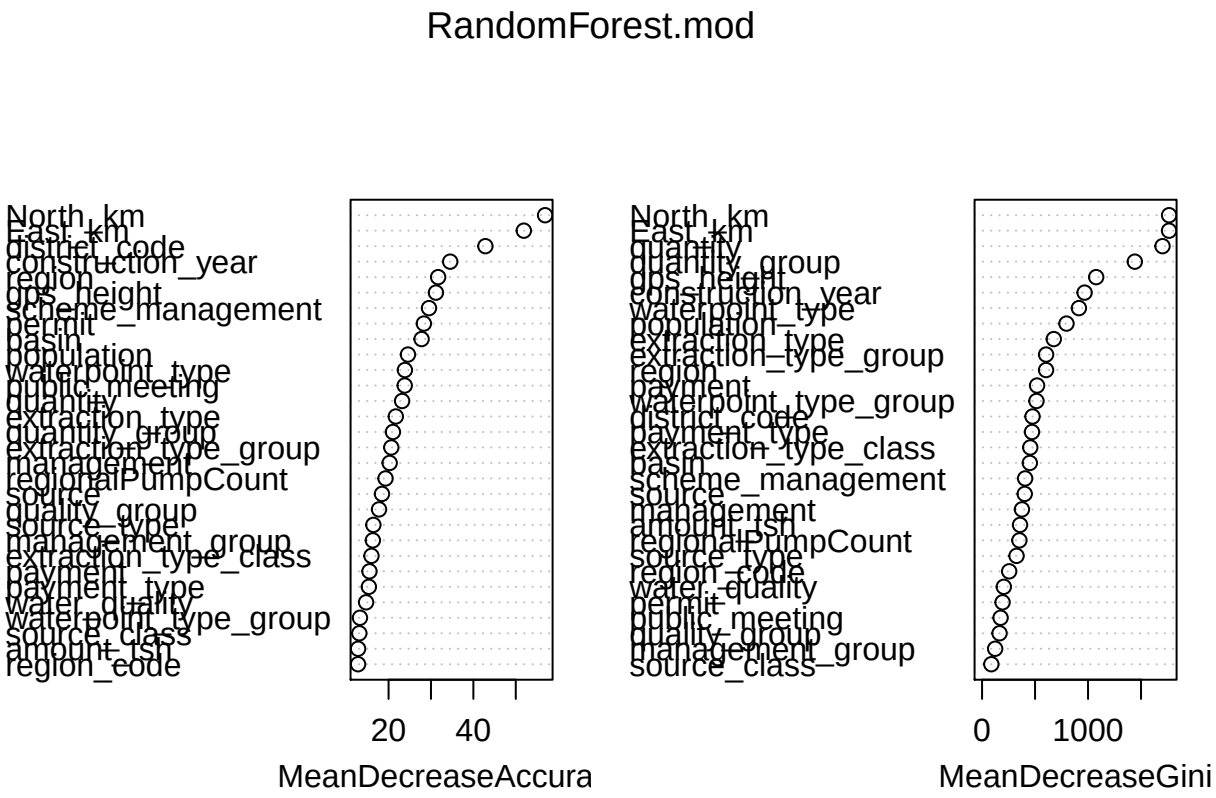
	Sensitivity	Specificity	Precision	Recall	Balanced.Accuracy
Class: functional	0.8980072	0.5738884	0.7136861	0.8980072	0.7359478
Class: functional needs repair	0.0517469	0.9966015	0.5451389	0.0517469	0.5241742
Class: non functional	0.6376522	0.8932134	0.7890589	0.6376522	0.7654328

Table 2: Validation Data Performance

	Sensitivity	Specificity	Precision	Recall	Balanced.Accuracy
Class: functional	0.8949424	0.5653732	0.7122638	0.8949424	0.7301578
Class: functional needs repair	0.0584567	0.9961904	0.5434783	0.0584567	0.5273236
Class: non functional	0.6227052	0.8892925	0.7766990	0.6227052	0.7559989

In this approach, we obtained training set prediction accuracy of 73.6%. By contrast, in the validation set, the prediction accuracy was 73.1%.

Random Forest



From the random forest procedure, we obtained training set prediction accuracy of 92.7%. By contrast, in the validation set, the prediction accuracy was only 80.4%. Based on this procedure, we also know that location, quantitry and pump age are some of the most powerful predictors.

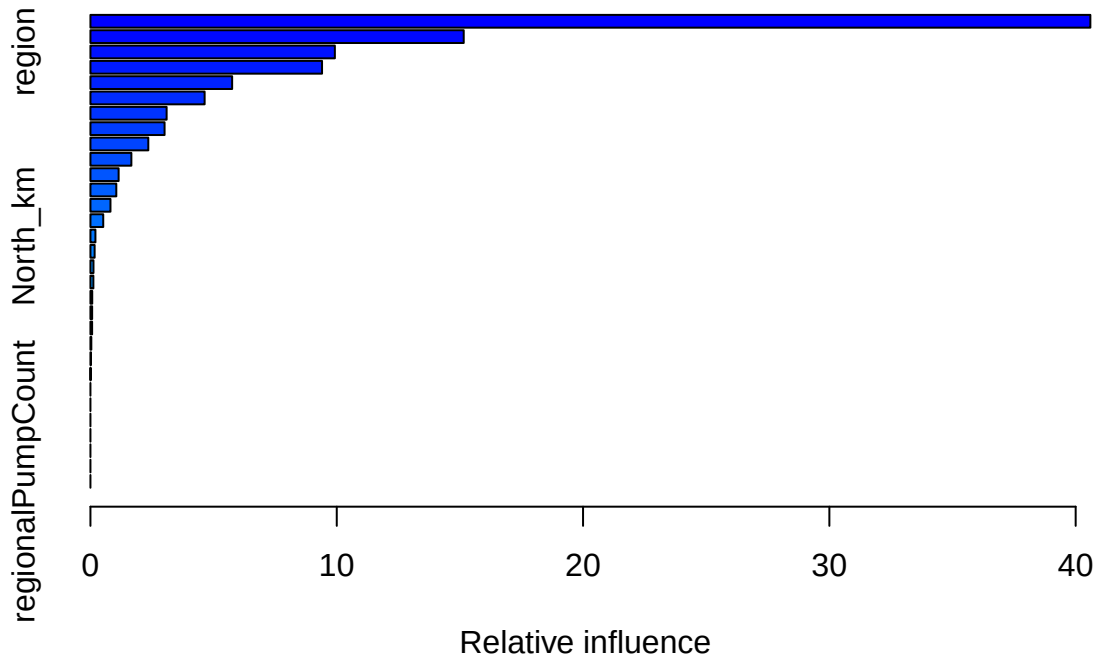
Table 3: Training Data Performance

	Sensitivity	Specificity	Precision	Recall	Balanced.Accuracy
Class: functional	0.9776308	0.8819361	0.9073571	0.9776308	0.9297834
Class: functional needs repair	0.6516150	0.9940590	0.8961922	0.6516150	0.8228370
Class: non functional	0.9078364	0.9781342	0.9629752	0.9078364	0.9429853

Table 4: Validation Data Performance

	Sensitivity	Specificity	Precision	Recall	Balanced.Accuracy
Class: functional	0.8966900	0.7256550	0.7971306	0.8966900	0.8111725
Class: functional needs repair	0.3086516	0.9822217	0.5739130	0.3086516	0.6454366
Class: non functional	0.7638420	0.9104532	0.8406336	0.7638420	0.8371476

Random Forest with Boosting




```
## NULL
```

Table 5: Training Data Performance

	Sensitivity	Specificity	Precision	Recall	Balanced.Accuracy
Class: functional	0.9217966	0.5751483	0.7195967	0.9217966	0.7484725
Class: functional needs repair	0.0909690	0.9957713	0.6287016	0.0909690	0.5433702
Class: non functional	0.6352170	0.9176217	0.8284877	0.6352170	0.7764194

Table 6: Validation Data Performance

	Sensitivity	Specificity	Precision	Recall	Balanced.Accuracy
Class: functional	0.9173520	0.5659911	0.7175941	0.9173520	0.7416715
Class: functional needs repair	0.1044427	0.9955857	0.6473430	0.1044427	0.5500142
Class: non functional	0.6194742	0.9129053	0.8147576	0.6194742	0.7661898

Tree Boosting Algorithm with XGBoost

This is an alternative boosting algorithm, which is often deemed as the most effective and most commonly used for the boost procedure. It has linear model solver as well as tree learning algorithm. While it may conceptually similar to the gbm algorithm, it does perform slightly better on the training and validation data sets.

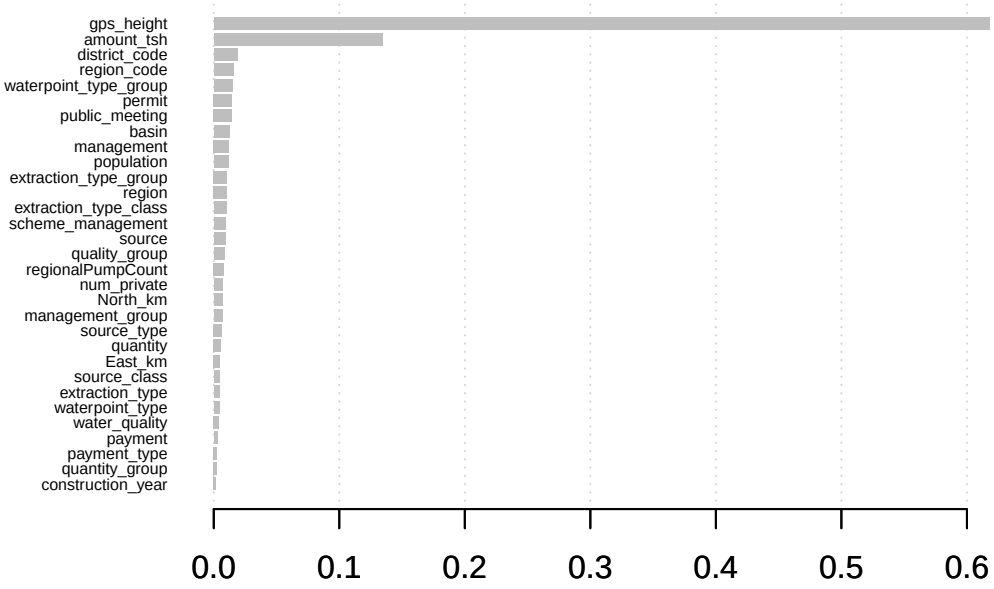


Table 7: Training Data Performance

	Sensitivity	Specificity	Precision	Recall	Balanced.Accuracy
Class: 1	0.8886028	0.9350434	0.9495362	0.8886028	0.9118231
Class: 2	0.8630600	0.9747029	0.6730389	0.8630600	0.9188815
Class: 3	0.9217862	0.9220558	0.8713081	0.9217862	0.9219210

Table 8: Validation Data Performance

	Sensitivity	Specificity	Precision	Recall	Balanced.Accuracy
Class: 1	0.7548135	0.7718206	0.8341900	0.7548135	0.7633171
Class: 2	0.4912281	0.9476560	0.3055339	0.4912281	0.7194420
Class: 3	0.7502791	0.8178197	0.6909972	0.7502791	0.7840494

Conclusion

Based on the results obtained with the previous models, it is evident that the random forest peroforms best. Below is the comparison of prediction accuracy of all three models over the validation dataset. We have also submitted out predictions to the competition. While we didn't really hit the top ten of the leaderboard, it was a great experience and something we would definitely do again.

Table 9: Model Comparison

Model	Accuracy
Binary Outcome Lasso	0.7306958
Random Forest	0.8035915
Boosted Random Forest	0.7450056
XGBoost	0.7414141

Appendices

Appendix A

Table 10: Metadata

Variable	Definition
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	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

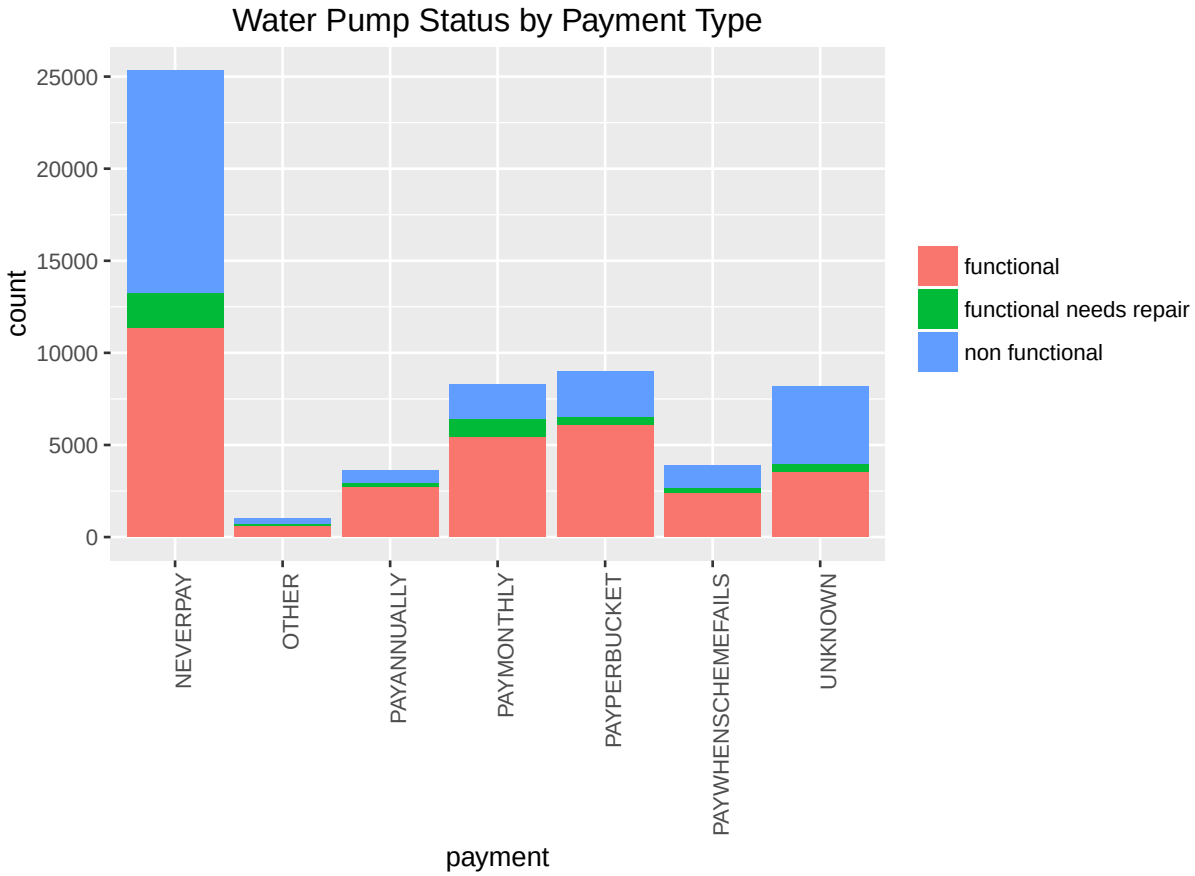
Variable	Definition
payment	What the water costs
payment_type	What the water costs
water_quality	The quality of the water
quality_group	The quality of the water
quantity	The quantity of water
quantity_group	The quantity of water
source	The source of the water
source_type	The source of the water
source_class	The source of the water
waterpoint_type	The kind of waterpoint
waterpoint_type_group	The kind of waterpoint

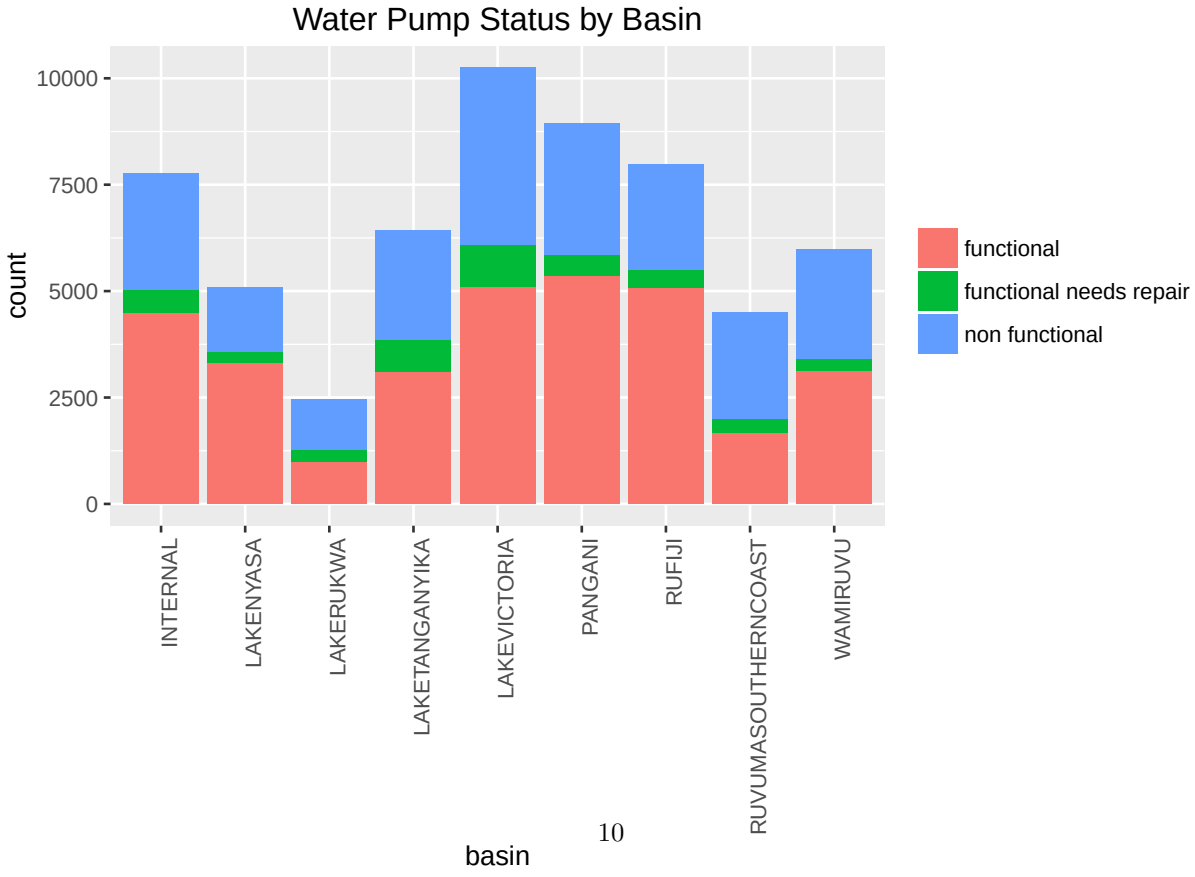
Appendix B

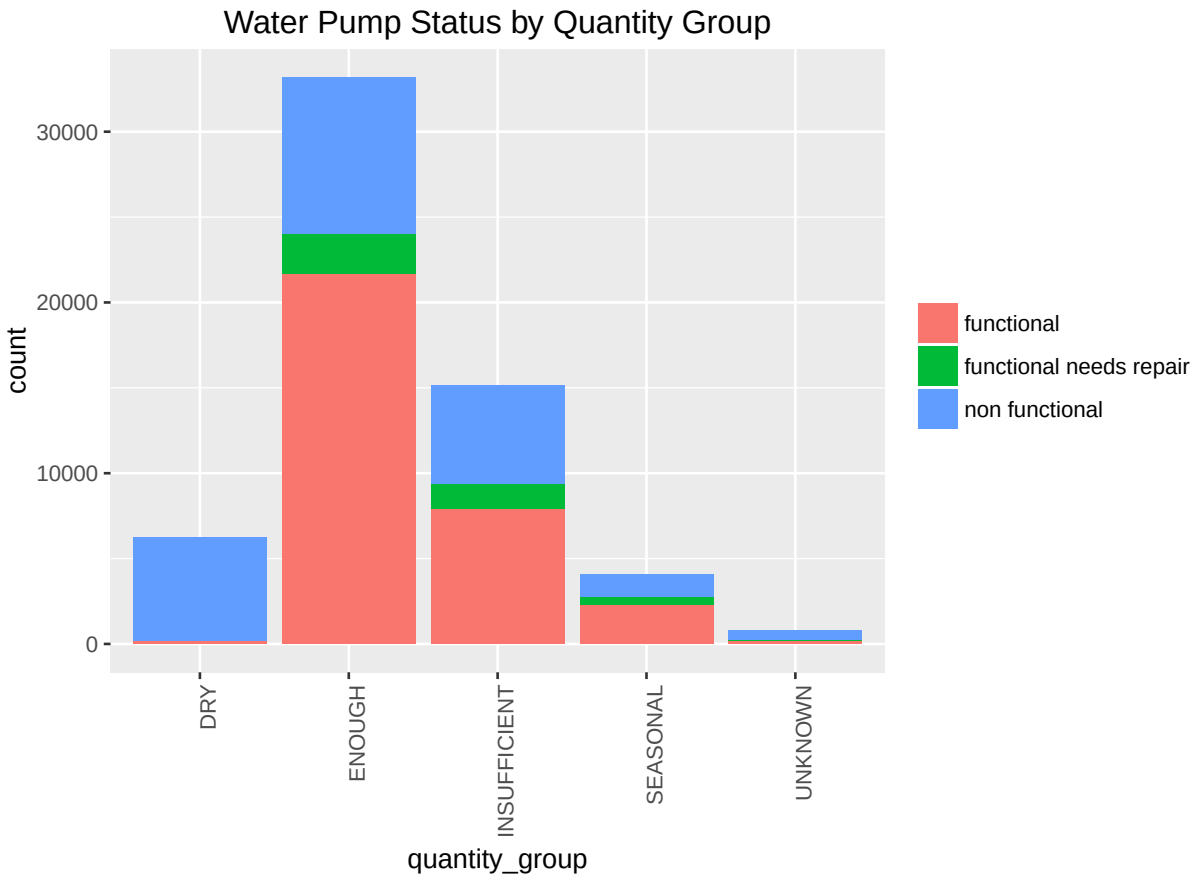
Table 11: Dropped Predictors

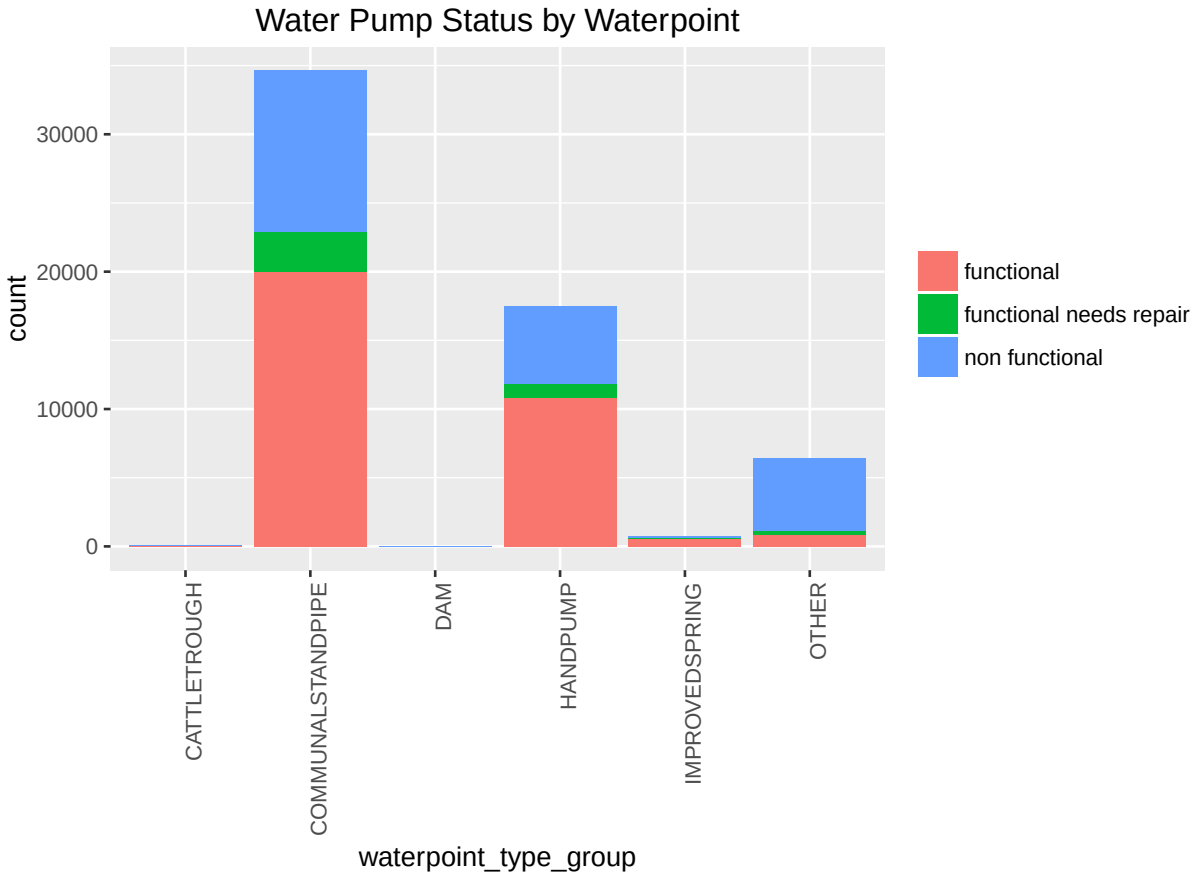
	factorlevels	vars
date_recorded	FALSE	date_recorded
funder	FALSE	funder
installer	FALSE	installer
wpt_name	FALSE	wpt_name
basin	TRUE	basin
subvillage	FALSE	subvillage
region	TRUE	region
lga	FALSE	lga
ward	FALSE	ward
public_meeting	TRUE	public_meeting
scheme_management	TRUE	scheme_management
scheme_name	FALSE	scheme_name
permit	TRUE	permit
extraction_type	TRUE	extraction_type
extraction_type_group	TRUE	extraction_type_group
extraction_type_class	TRUE	extraction_type_class
management	TRUE	management
management_group	TRUE	management_group
payment	TRUE	payment
payment_type	TRUE	payment_type
water_quality	TRUE	water_quality
quality_group	TRUE	quality_group
quantity	TRUE	quantity
quantity_group	TRUE	quantity_group
source	TRUE	source
source_type	TRUE	source_type
source_class	TRUE	source_class
waterpoint_type	TRUE	waterpoint_type
waterpoint_type_group	TRUE	waterpoint_type_group

Appendix C


Water Pump Status by Payment Type
count
25000
20000
15000
10000
5000
0
NEVERPAY
OTHER
PAYANNUALLY
PAYMONTHLY
PAYPERBUCKET
PAYWHENSCHEMEFAILS
UNKNOWN
payment
functional
functional needs repair
non functional


Water Pump Status by Basin
count
10000
7500
5000
2500
0
INTERNAL
LAKENYASA
LAKERUKWA
LAKETANGANYIKA
LAKEVICTORIA
PANGANI
RUFIJI
RUVUMASOUTHERNCOAST
WAMIRUVU
basin
functional
functional needs repair
non functional


Water Pump Status by Quantity Group
count
30000
20000
10000
0
DRY
ENOUGH
INSUFFICIENT
SEASONAL
UNKNOWN
quantity_group
functional
functional needs repair
non functional



Appendix D

Source code for the project can be found here: https://github.com/StatisticsGuru/WaterPump