PROJECT REPORT

PUMP IT UP: DATAMINING THE WATER TABLE

Team

Gayatri Balakumar(gxb170030)

Mohana Prudvi Dongala(mxd165330)

Sowmya Bhupathiraju(sxb162130)

Teja Kiran Chunduri(txc163430)

1. INTRODUCTION

As a part of our project we have taken part in the driven data competition: “Pump it Up: Data Mining the Water Table”. For this challenge we have to develop a classifier that can classify various water points of Tanzania water resources as “Functional”, “Non-Functional” and “Functional needs repair”. These values should be predicted upon the large array of attributes provided to us in the dataset. Then we have developed various classifiers and submitted an .csv file for the test values that have been provided. The motto of the classification is to know the understanding and maintenance of the water points to provide better service across the communities of Tanzania.

2. PROBLEM DESCRIPTION

The learner we are developing for the project is a part of the competition Pump it up –Driven Data challenge. The objective of this challenge is to predict the working condition of the water point whether it is functional, functional needs repair or non-functional. The data for this competition comes from the “Tariff water point’s dashboard”, which aggregates data from the Tanzania Ministry of Water.

They have provided with

- Training set values -The variables for the training set
- Training set labels-The class label for each variable in training set
- Test set values- The variables that need predictions

The problem definition is to find out whether or not a water pipe if functional or needs a repair or not functioning

3. DATASET DESCRIPTION

The training data set given for the given competition had 59400 training values where each has one ID and class label.

- Total features : 39 attributes and 1 class label
- Total number of training instances: 59400
- Total number of testing instances: 14850

3.1 Type of Attributes

- Number of categorical attributes: 31
- Number of numeric attributes: 7
- Number of data attributes: 2

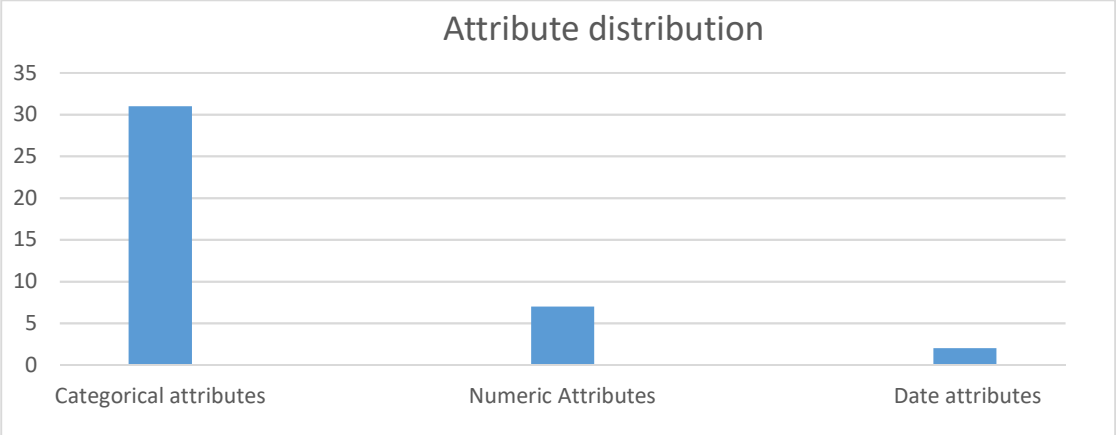

Fig 3.1. Types of Attributes

Attributes Description

Name of the attribute	**Attribute description**
Amount_tsh	The amount of total static head
Date_recorded	The date when the data point was entered
Funder	The organization funding the well
Gps_height	Altitude of the well
Installer	The organization which installed the well
Latitude	GPS coordinate of the well
Longitude	GPS coordinate of the well
Wpt_name	Name of the water point if present else null
Num_private	
Basin	Geographic location of the basin
Sub_village	Geographic location i.e., name of village
Region	Geographic location i.e., region name
Region_code	Geographic location i.e., zip code of region
District_code	Geographic location i.e., unique region code
Lga	Geographic location
Ward	Geographic location
Population	Population around the water point
Public_meeting	True /False
Recorded_by	Group which entered this row of data
Scheme_management	Organization that operates the water point
Scheme_name	Name of organization that operates the water point
Permit	Whether the water point is permitted
Construction_year	Year of water point construction
Extraction_type	Extraction method used by water point

Extraction_type_group	Group of Extraction method used by waterpoint
Extraction_type_class	The kind of extraction the waterpoint uses
Management	In what manner the waterpoint is managed
Management_group	In what manner the waterpoint is managed
Payment	Cost of the water
Payment_type	Cost of the water
Water_quality	Quality of the water
Quality_group	Quality of the water
Quantity	Quantity of the water
Quantity_group	The quantity of water
Source	The source of waterpoint
Source_type	The source of waterpoint
Source_class	The source of waterpoint
Waterpoint_type	The kind of waterpoint
Waterpoint_type_group	The kind of waterpoint

3.2 Class Labels

The Class labels in this dataset:

- functional - the water point is functional and no repairs needed
- functional needs repair - the water point is functional, but repairs needed
- non-functional - the water point is not functional

Class Labels Distribution

- Number of functional labels: 32259
- Number of functional needs repair: 4317
- Number of nonfunctional: 22824

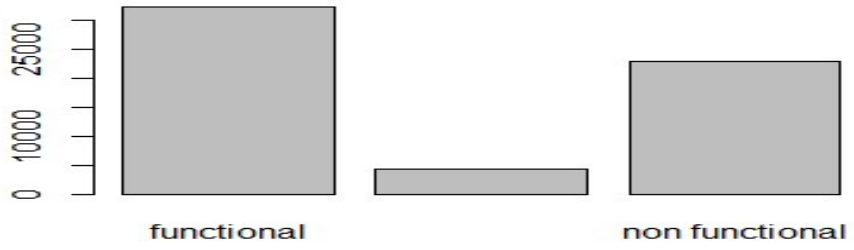

Fig 3.2 Distribution of labels

4. PRE-PROCESSING TECHNIQUES

4.1 ATTRIBUTE ANALYSIS

The raw data may contain lot of noise that is, it may contain number of attributes which may not contribute to classification or it may mislead the learner. So we checked each and every attribute to know how much important it is and how it effects the learner's accuracy .We performed a number of operations like calculating correlation ,plotting histograms against attributes and class labels ,checking their null values percentage etc.

4.1.1 Eliminating attributes

Calculating correlations

A correlation matrix is constructed for all the attributes to check out the correlations .It is huge 40/40 matrix. From the correlation matrix we tried to observe the correlation with the class label as well as the correlation with the other attributes.

We observed the following:

- Checking with the class labels
  - Good correlation with the class labels :Theses attributes help in classification
  - Low correlation with the class labels: Further check the reason behind low correlation
- Checking amongst other attributes
  - Redundant attributes

By carefully checking the correlations values obtained, we began by assuming a threshold of 0.8. The attribute pairs which have more than 0.8 are further checked to see which one of them is more useful. We analyzed the raw data to find out the reason behind the high correlation. In case the attributes were way too similar .i.e., their contribution to classification is same (from values and their distribution) we concluded such attributes as redundant ones.

	id	amount_tsh	date_recorded	funder	gps_height	installer	longitude	latitude	wpt_name
id	1.00	-0.01	0.00	-0.01	0.00	0.00	0.00	0.00	0.00
amount_tsh	-0.01	1.00	-0.02	0.01	0.08	0.01	0.02	-0.05	0.01
date_recorded	0.00	-0.02	1.00	0.01	0.28	0.02	-0.04	0.26	-0.10
funder	-0.01	0.01	0.01	1.00	0.01	0.58	-0.04	0.08	0.02
gps_height	0.00	0.08	0.28	0.01	1.00	0.04	0.15	-0.04	0.02
installer	0.00	0.01	0.02	0.58	0.04	1.00	0.00	0.07	0.02
longitude	0.00	0.02	-0.04	-0.04	0.15	0.00	1.00	-0.43	-0.02
latitude	0.00	-0.05	0.26	0.08	-0.04	0.07	-0.43	1.00	-0.04
wpt_name	0.00	0.01	-0.10	0.02	0.02	0.02	-0.02	-0.04	1.00
num_private	0.00	0.00	-0.02	0.01	0.01	0.02	0.02	0.01	0.00
basin	0.00	0.01	-0.21	-0.03	-0.16	0.01	0.22	-0.22	-0.02
subvillage	0.00	0.00	0.08	-0.03	0.02	-0.01	0.00	-0.01	0.10
region	0.00	-0.02	-0.05	0.11	-0.32	0.11	-0.14	-0.03	-0.02
region_code	0.00	-0.03	0.12	0.00	-0.18	0.03	0.03	-0.22	0.02
district_code	0.00	-0.02	-0.02	-0.02	-0.17	0.02	0.15	-0.20	0.00
lga	0.00	0.01	0.05	-0.04	0.14	0.05	0.23	-0.27	-0.04
ward	0.00	0.00	0.07	0.03	0.04	0.02	0.04	0.01	0.00
population	0.00	0.02	0.10	0.02	0.14	0.04	0.09	-0.02	0.01
public_meeting	0.00	0.03	-0.16	0.05	0.02	0.03	0.10	-0.05	-0.02
recorded_by	NA	NA	NA	NA	NA	NA	NA	NA	NA
scheme_management	0.00	0.01	0.06	0.01	0.05	0.04	-0.12	0.02	-0.06
scheme_name	0.00	0.03	0.03	0.03	0.25	0.01	0.13	-0.07	-0.02
permit	0.00	0.02	-0.01	0.15	0.02	0.16	0.06	0.06	-0.03
construction_year	0.00	0.07	0.25	-0.02	0.66	0.05	0.40	-0.25	0.00
extraction_type	0.00	-0.03	-0.04	0.00	-0.24	0.02	-0.03	-0.01	0.01

	num_private	basin	subvillage	region	region_code	district_code	lga	ward	population
id	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
amount_tsh	0.00	0.01	0.00	-0.02	-0.03	-0.02	0.01	0.00	0.02
date_recorded	-0.02	-0.21	0.08	-0.05	0.12	-0.02	0.05	0.07	0.10
funder	0.01	-0.03	-0.03	0.11	0.00	-0.02	-0.04	0.03	0.02
gps_height	0.01	-0.16	0.02	-0.32	-0.18	-0.17	0.14	0.04	0.14
installer	0.02	0.01	-0.01	0.11	0.03	0.02	0.05	0.02	0.04
longitude	0.02	0.22	0.00	-0.14	0.03	0.15	0.23	0.04	0.09
latitude	0.01	-0.22	-0.01	-0.03	-0.22	-0.20	-0.27	0.01	-0.02
wpt_name	0.00	-0.02	0.10	-0.02	0.02	0.00	-0.04	0.00	0.01
num_private	1.00	0.02	-0.01	0.04	-0.02	0.00	0.00	0.01	0.00
basin	0.02	1.00	0.03	-0.11	0.14	0.19	-0.01	0.04	0.07
subvillage	-0.01	0.03	1.00	0.01	0.03	0.03	0.02	0.06	0.02
region	0.04	-0.11	0.01	1.00	0.11	-0.02	0.19	0.05	0.00
region_code	-0.02	0.14	0.03	0.11	1.00	0.68	0.04	0.03	0.09
district_code	0.00	0.19	0.03	-0.02	0.68	1.00	0.11	0.04	0.06
lga	0.00	-0.01	0.02	0.19	0.04	0.11	1.00	0.06	0.00
ward	0.01	0.04	0.06	0.05	0.03	0.04	0.06	1.00	0.03
population	0.00	0.07	0.02	0.00	0.09	0.06	0.00	0.03	1.00
public_meeting	0.01	0.01	-0.04	-0.10	-0.05	0.03	-0.01	0.02	0.00
recorded_by	NA	NA	NA	NA	NA	NA	NA	NA	NA
scheme_management	0.00	-0.07	0.00	0.00	-0.09	-0.02	-0.05	0.02	-0.06
scheme_name	0.01	0.00	0.02	-0.23	-0.12	-0.07	0.02	0.10	-0.03
permit	0.01	0.21	-0.01	-0.05	0.01	-0.01	-0.15	0.03	-0.03
construction_year	0.03	0.27	0.07	-0.14	0.03	0.05	0.13	0.08	0.26
extraction_type	0.02	0.17	0.01	0.22	0.10	0.03	0.02	0.00	0.07

	public_meeting	recorded_by	scheme_management	scheme_name	permit	construction_year	extraction_type
id	0.00	NA	0.00	0.00	0.00	0.00	0.00
amount_tsh	0.03	NA	0.01	0.03	0.02	0.07	-0.03
date_recorded	-0.16	NA	0.06	0.03	-0.01	0.25	-0.04
funder	0.05	NA	0.01	0.03	0.15	-0.02	0.00
gps_height	0.02	NA	0.05	0.25	0.02	0.66	-0.24
installer	0.03	NA	0.04	0.01	0.16	0.05	0.02
longitude	0.10	NA	-0.12	0.13	0.06	0.40	-0.03
latitude	-0.05	NA	0.02	-0.07	0.06	-0.25	-0.01
wpt_name	-0.02	NA	-0.06	-0.02	-0.03	0.00	0.01
num_private	0.01	NA	0.00	0.01	0.01	0.03	0.02
basin	0.01	NA	-0.07	0.00	0.21	0.27	0.17
subvillage	-0.04	NA	0.00	0.02	-0.01	0.07	0.01
region	-0.10	NA	0.00	-0.23	-0.05	-0.14	0.22
region_code	-0.05	NA	-0.09	-0.12	0.01	0.03	0.10
district_code	0.03	NA	-0.02	-0.07	-0.01	0.05	0.03
lga	-0.01	NA	-0.05	0.02	-0.15	0.13	0.02
ward	0.02	NA	0.02	0.10	0.03	0.08	0.00
population	0.00	NA	-0.06	-0.03	-0.03	0.26	0.07
public_meeting	1.00	NA	0.08	0.14	0.11	0.00	-0.09
recorded_by	NA	1	NA	NA	NA	NA	NA
scheme_management	0.08	NA	1.00	0.13	-0.10	-0.06	-0.08
scheme_name	0.14	NA	0.13	1.00	0.13	0.17	-0.26
permit	0.11	NA	-0.10	0.13	1.00	0.07	-0.09
construction_year	0.00	NA	-0.06	0.17	0.07	1.00	-0.06
extraction_type	-0.09	NA	-0.08	-0.26	-0.09	-0.06	1.00

	extraction_type_group	extraction_type_class	management	management_group	payment	payment_type
id	0.00	0.00	0.00	0.00	0.00	0.00
amount_tsh	-0.02	-0.01	-0.01	0.00	0.01	-0.12
date_recorded	-0.07	0.00	0.05	-0.06	0.21	0.26
funder	0.02	0.03	0.06	-0.04	0.03	0.02
gps_height	-0.26	-0.23	-0.04	0.03	0.11	-0.10
installer	0.05	0.06	0.00	-0.06	-0.01	-0.01
longitude	-0.02	0.05	-0.18	-0.08	0.02	-0.01
latitude	0.00	0.00	0.05	-0.06	-0.04	0.15
wpt_name	0.01	-0.01	-0.05	-0.03	-0.04	-0.04
num_private	0.01	0.02	-0.01	-0.03	0.00	0.01
basin	0.19	0.18	-0.06	-0.07	-0.02	-0.03
subvillage	0.00	0.01	0.00	0.00	0.02	0.02
region	0.23	0.21	0.08	-0.02	-0.05	0.06
region_code	0.10	0.15	-0.01	0.03	0.03	0.14
district_code	0.02	0.07	-0.04	0.02	0.00	0.09
lga	-0.01	0.00	-0.08	-0.01	-0.11	-0.18
ward	0.00	0.02	0.00	-0.05	0.00	0.00
population	0.06	0.08	-0.05	-0.02	0.03	0.03
public_meeting	-0.10	-0.12	0.07	0.19	-0.15	-0.25
recorded_by	NA	NA	NA	NA	NA	NA
scheme_management	-0.08	-0.09	0.56	0.34	-0.02	-0.05
scheme_name	-0.27	-0.16	-0.03	0.02	0.01	-0.06
permit	-0.07	-0.07	-0.02	0.01	-0.02	-0.11
construction_year	-0.06	0.01	-0.13	-0.07	0.13	-0.02
extraction_type	0.95	0.70	0.04	-0.04	0.00	0.13

	water_quality	quality_group	quantity	quantity_group	source	source_type	source_class
id	0.00	0.00	0.00	0.00	0.00	0.00	-0.01
amount_tsh	0.01	-0.02	-0.01	-0.01	0.00	0.00	0.03
date_recorded	0.02	0.02	0.04	0.04	-0.03	-0.04	-0.04
funder	-0.06	-0.01	-0.01	-0.01	-0.08	-0.08	0.02
gps_height	0.14	-0.09	-0.03	-0.03	0.12	0.12	0.04
installer	-0.01	-0.03	0.00	0.00	-0.10	-0.08	0.03
longitude	-0.05	-0.03	0.02	0.02	-0.01	-0.05	0.02
latitude	-0.01	0.04	0.12	0.12	0.02	0.06	-0.04
wpt_name	0.01	-0.01	0.03	0.03	-0.01	-0.01	0.02
num_private	0.00	-0.01	0.00	0.00	-0.01	-0.01	0.01
basin	-0.08	0.04	-0.03	-0.03	-0.06	-0.07	0.03
subvillage	0.02	-0.01	0.02	0.02	-0.01	-0.01	0.00
region	-0.07	0.13	0.03	0.03	-0.16	-0.14	-0.01
region_code	-0.06	0.08	-0.07	-0.07	-0.13	-0.14	-0.08
district_code	-0.06	0.04	-0.03	-0.03	-0.05	-0.07	-0.05
lga	0.02	0.03	0.01	0.01	-0.07	-0.06	-0.02
ward	-0.01	0.00	0.00	0.00	-0.01	-0.02	0.03
population	-0.03	0.02	0.03	0.03	-0.09	-0.09	0.03
public_meeting	0.01	-0.11	-0.06	-0.06	0.07	0.05	0.01
recorded_by	NA	NA	NA	NA	NA	NA	NA
scheme_management	0.07	-0.09	-0.13	-0.13	0.03	0.02	0.00
scheme_name	0.11	-0.10	-0.09	-0.09	0.08	0.10	0.16
permit	-0.03	-0.13	-0.05	-0.05	0.07	0.09	0.09
construction_year	0.05	-0.01	-0.01	-0.01	-0.01	-0.03	0.10
extraction_type	-0.11	0.16	0.00	0.00	-0.33	-0.35	-0.23

Fig:4.1 Correlation matrices

To conclude upon which ones to retain, we used variable importance on different classifiers like random forests and chose the better one. We started from the least ranked attributes and followed the following steps to see whether to retain or remove

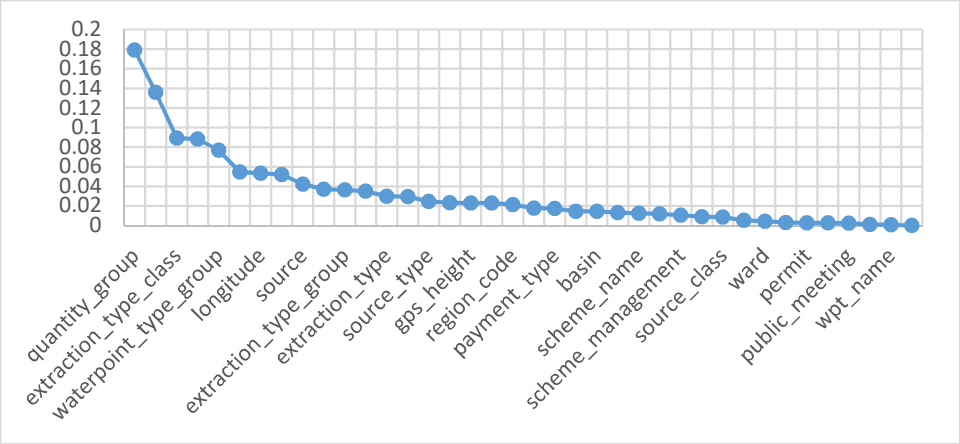

Fig :4.2 Variable importance of attributes on random forests

Checking for NULL values

We checked for the number of NULL values in the two attributes, the one which has most number of NULL values is removed. If we could not relate by checking NULL values, then we have further processed to check their factoring levels.

```
                id              amount_tsh           date_recorded                  funder
         0.0000000               0.0000000               0.0000000               6.1195286
        gps_height               installer               longitude                latitude
         0.0000000               6.1531987               0.0000000               0.0000000
          wpt_name             num_private                   basin              subvillage
         0.0000000               0.0000000               0.0000000               0.6245791
            region             region_code           district_code                     lga
         0.0000000               0.0000000               0.0000000               0.0000000
              ward              population          public_meeting            recorded_by
         0.0000000               0.0000000               5.6127946               0.0000000
scheme_management            scheme_name                  permit       construction_year
         6.5269360              47.4175084               5.1447811               0.0000000
   extraction_type extraction_type_group  extraction_type_class              management
         0.0000000               0.0000000               0.0000000               0.0000000
 management_group                 payment            payment_type           water_quality
         0.0000000               0.0000000               0.0000000               0.0000000
    quality_group                quantity           quantity_group                  source
         0.0000000               0.0000000               0.0000000               0.0000000
     source_type             source_class         waterpoint_type waterpoint_type_group
         0.0000000               0.0000000               0.0000000               0.0000000
```

Fig 4.3 Null value percentages for each Attributes

Plotting Histograms

We have plotted histograms and checked for the number of levels. If they have a considerable distribution on levels then the attribute which does not have proper distribution is removed.

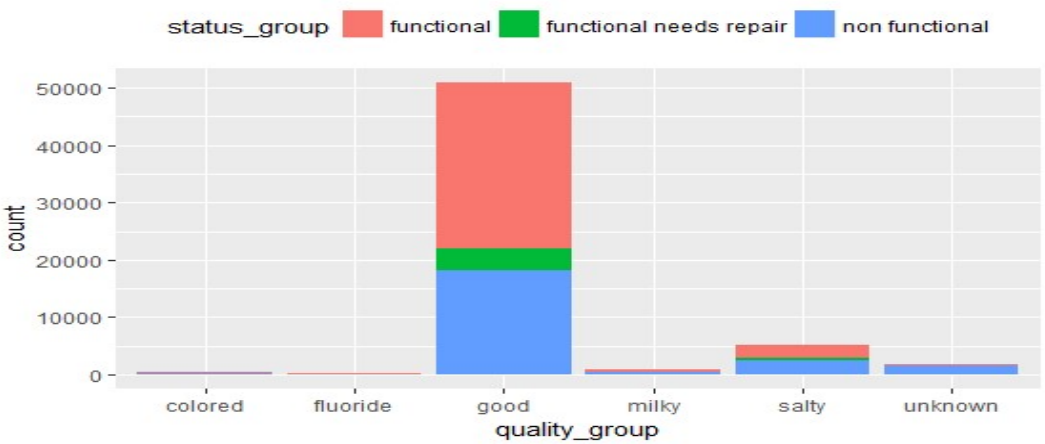

Fig: 4.4 Histogram plotted for quality_group

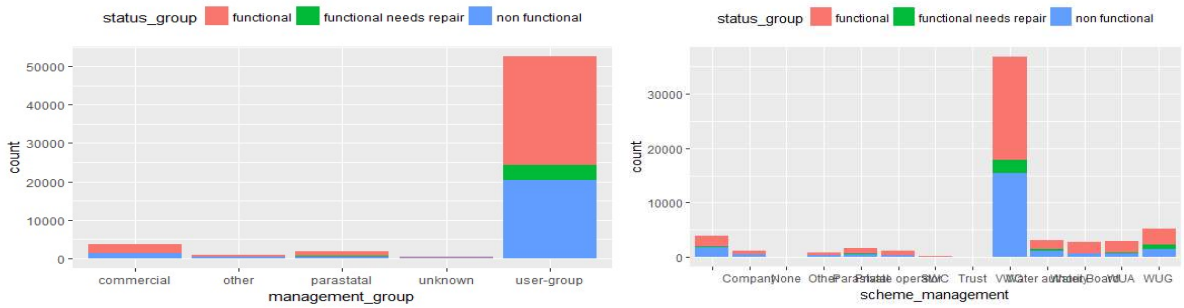

Fig:4.5 histograms plotted for management_group and schema_management

We have seen by plotting histogram for quality group that most of the rows value is “good “,so this does not contribute much for classification. The same with the other two attributes also.

Another conclusions obtained by plotting the histograms

By plotting histograms we also observed many of the attributes with too many levels. Then we had to see the usefulness of these attributes before disregarding them completely because we can handle such attributes by bucketing them. Few of them were

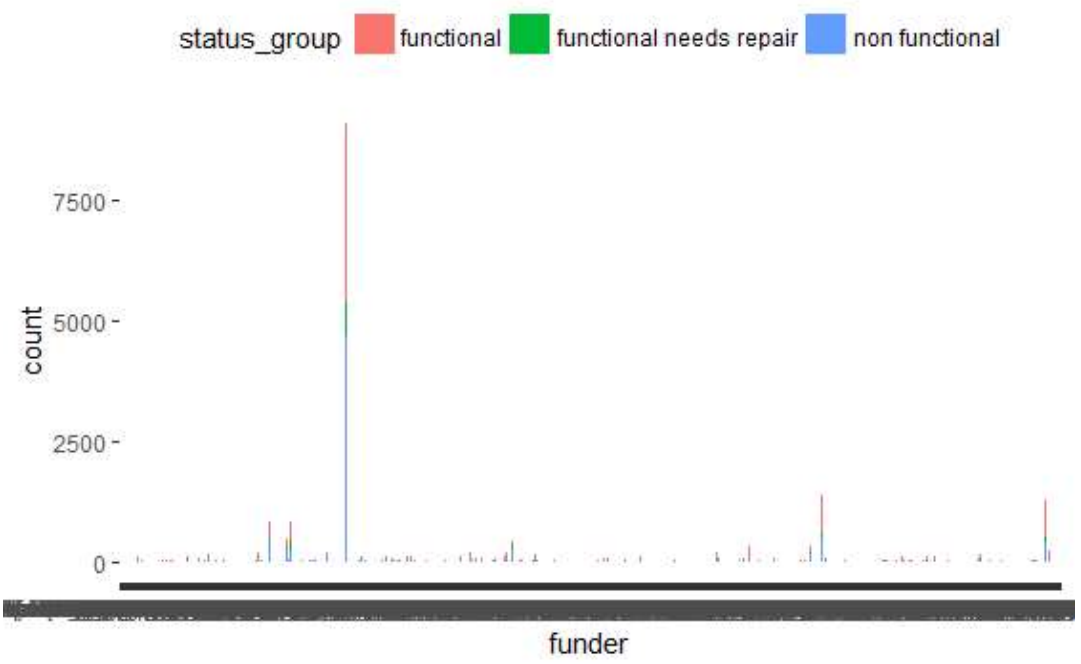

Fig:4.6 histograms plotted for funder

There were cases where the attributes had too many levels and did not have much significance with the class label such as:

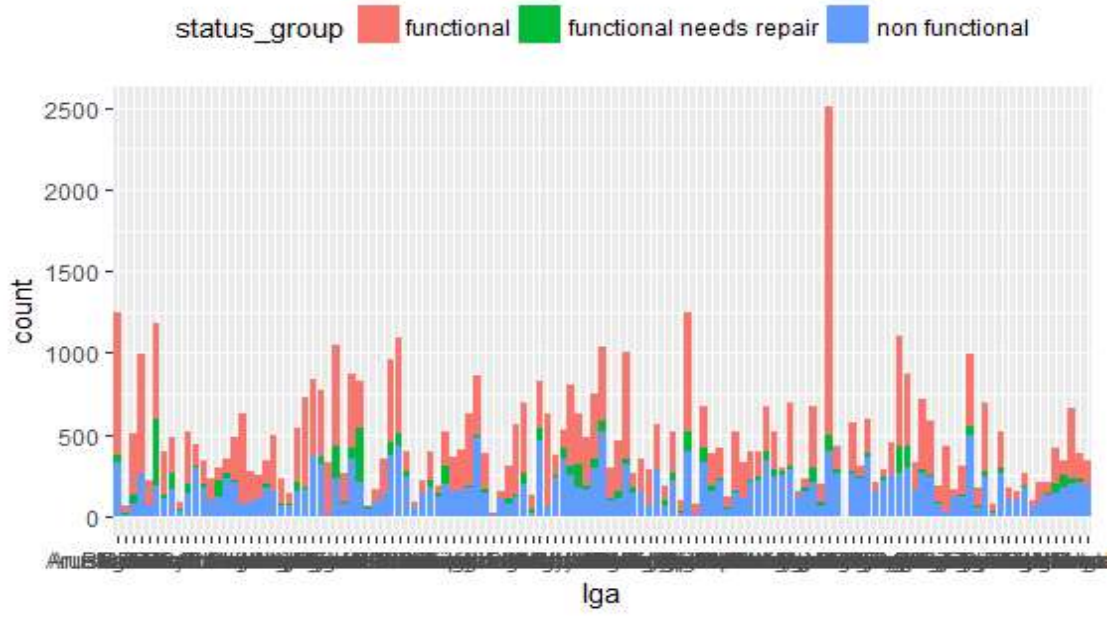

Fig: 4.7 histograms plotted for lga

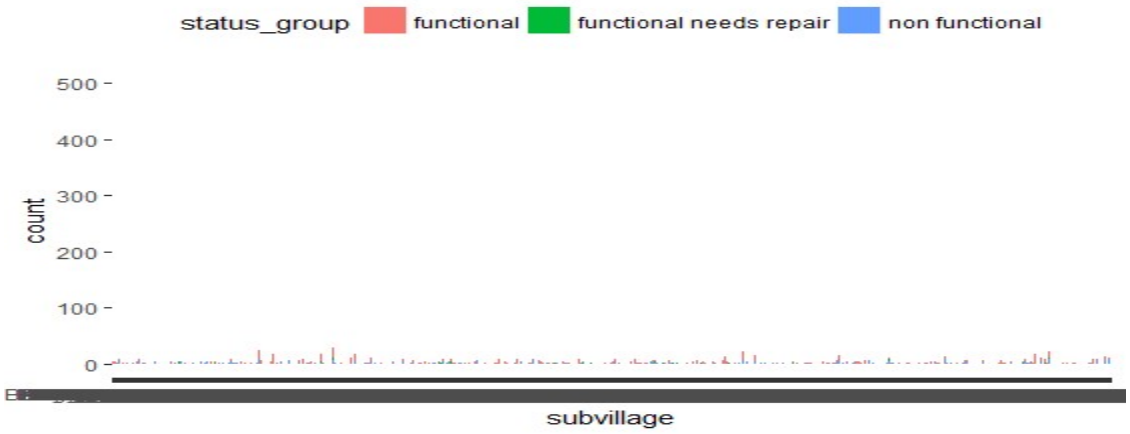

Fig 4.8 Histogram of sub village

Impact of removing attributes on classifier (random forests)

For impact on classifier we proceeded by running on random forest to cross verify the results and decide on few uncertain attributes. For those, we incrementally removed attributes and checked the impact on accuracy by retaining and removing them. This was performed by running experiments on a classifier (random forest) by providing attributes on all confusing combinations and recording the accuracy values. Doing this also helped us confirm on equivalent data attributes.

Combination Of Attributes Removed	Manually Recorded Impact On Accuracy
recorded_by	0.1
recorded_by,latitude	0.3
recorded_by,wpt_name	0
recorded_by,wpt_name,public_meeting	0.3
recorded_by,wpt_name,public_meeting,subvillage	0.02
recorded_by,wpt_name,public_meeting,subvillage,quantity_group	0.09
recorded_by,wpt_name,public_meeting,subvillage,quantity_group,ward	-0.19
recorded_by,wpt_name,public_meeting,subvillage,quantity_group,ward,management_group	-0.16
recorded_by,wpt_name,public_meeting,subvillage,quantity_group,ward,management_group,source_type	0
recorded_by,wpt_name,public_meeting,subvillage,quantity_group,ward,management_group,source_type,extraction_type_group	-0.66
recorded_by,wpt_name,public_meeting,subvillage,quantity_group,ward,management_group,source_type,extraction_type_class	-0.91
recorded_by,wpt_name,public_meeting,subvillage,quantity_group,ward,management_group,source_type,extraction_type	-0.26
recorded_by,wpt_name,public_meeting,subvillage,quantity_group,ward,management_group,source_type,extraction_type_group,extraction_type_class,extraction_type	-0.99
recorded_by,wpt_name,public_meeting,subvillage,quantity_group,ward,management_group,source_type,extraction_type_group,extraction_type	0.16
recorded_by,wpt_name,public_meeting,subvillage,quantity_group,ward,management_group,source_type,extraction_type_group,extraction_type,region,district_code, region_code,lga	-0.31
recorded_by,wpt_name,public_meeting,subvillage,quantity_group,ward,management_group,source_type,extraction_type_group,extraction_type,region	0.12

Fig 4.9 Impact of Accuracy on classifier when adding and removing the attributes

After the above procedures, the following attributes are removed.

index	Attribute	Reason for removal
1	quantity_goup	redundant high correlation with quantity
2	extraction_type_group	redundant high correlation with extraction_type
3	source_type	redundant high correlation with source
4	payment_type	redundant high correlation with payment
5	quality_group	less variable importance most of the rows have one value effecting accuracy negatively
6	waterpoint_type_group	redundant high correlation with waterpoint_type
7	scheme_name	high NULL value percentage
8	recorded_by	same value in entire data
9	num_private	very less variable importance no significance more number of NULL values
10	sub_village	too many levels less variable importance
11	district_code	too many levels less variable importance
12	wpt_name	less variable importance
13	extraction_type	less consistent and similar data compared to extraction_type_class
14	Lga	too many levels
15	region_code	similar to region
16	management_group	similar to management
17	Installer	similar to funder
18	scheme_management	no significance
19	Ward	Similar to subvillage
20	Date_recorded	Modified to operation_age

4.1.2 Modification of existing attributes

Operations Age

Using date recorded and construction year. We haves seen that date recorded attribute have numerous level and construction years are mostly around 2000. To make these attributes useful we have constructed a new attribute called operationyear. This attribute gives the operation duration of water point in years. This is the subtraction of year from the date recorded and the construction year. This attribute now makes sense as we are taking in regard the feature which describes how old the water point is. This may help us to know about the working condition of the water point.

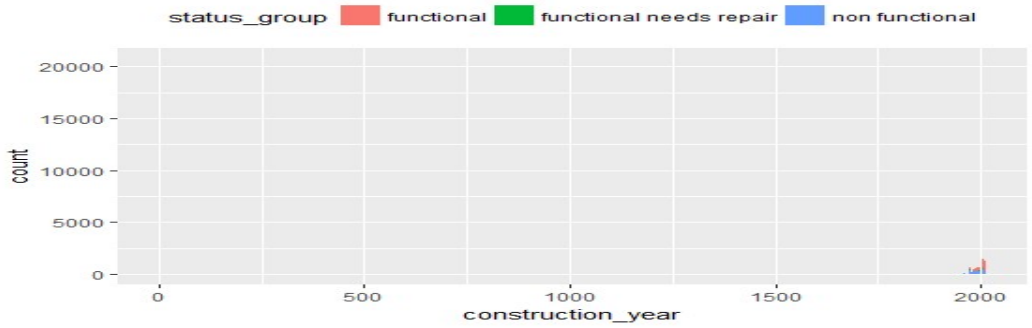

Fig 4.10 histogram of construction year showing all concentrated around 2000

Funder

The funder attribute is the name of the organization funding the water point. The data of funder has many human errors. Mostly punctuations and spelling mistakes. If these are not handled the number of levels is at 1898. So we have modifies the funder column and then factored it to 11 levels

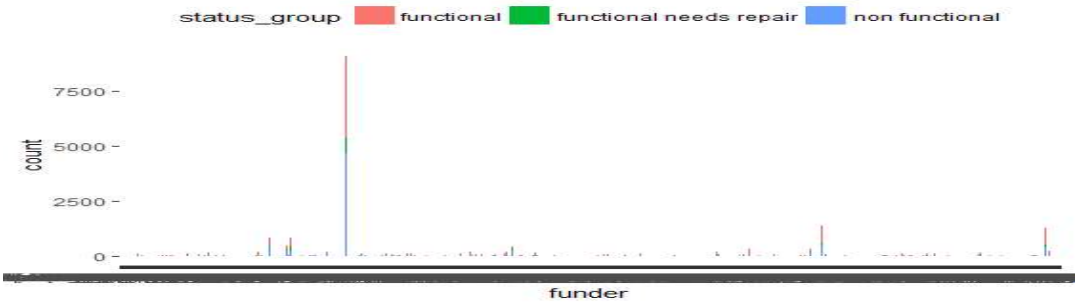

Fig 4.11 Histogram of Funder

Finally, after the attribute we considered the following attributes.

Index	Attributes
1	Amount_tsh
2	Funder
3	Gps_height
4	Longitutude
5	Latitude
6	Basin
7	Region
8	Population
9	Public_meeting
10	Permit
11	Extraction_type_class
12	Management
13	Payment
14	Water_quality
15	Quantity
16	Source
17	Source_class
18	Waterpoint_type
19	Operation_age

4.2 Pseudo code

- Firstly we have removed the attributes that we selected keeping those we want to change.

```
Training_Data_Complete$source_type = NULL
Testing_Data$source_type = NULL
```

- We have changed the attribute construction year and date recorded into a new attribute OperationAge using the year part of the attributes. Then removed the attributes construction year and date_recorded

```
t <- format(as.Date(Training_Data_Complete$date_recorded, '%Y-%m-%d'),'%Y')
tTest <- format(as.Date(Testing_Data$date_recorded, '%Y-%m-%d'),'%Y')

t <- as.numeric(t)
tTest <- as.numeric(tTest)

Training_Data_Complete$OperationAge <- t - Training_Data_Complete$construction_year
Testing_Data$OperationAge <- tTest - Testing_Data$construction_year

Training_Data_Complete$OperationAge[Training_Data_Complete$OperationAge > 2000] <- 0
Testing_Data$OperationAge[Testing_Data$OperationAge > 2000] <- 0
```

- On the column Funder: This column has too many levels and errors. TO handle them we identified similar values for this column we have firstly changed the case to lower case then removed the spaces in the values, lastly trimmed the values into first 3 characters in R . Also we have considered row values with garbage such as“.”,””,””,” ”,”-“,”_” as others. This way we are handling even null values here. Now the levels have reduced to around 400 .Further we have bucketed these levels to 11.

```
funders_levels <- names(summary(Training_Data_Complete$funder)[1:limitLevels])
funder <- factor(Training_Data_Complete$funder, levels=c(funders_levels, "Other"))
funder[is.na(funder)] <- "Other"
Training_Data_Complete$funder <- funder

funder <- factor(Testing_Data$funder, levels=c(funders_levels, "Other"))
funder[is.na(funder)] <- "Other"
Testing_Data$funder <- funder
```

- Changed gps_levels by bucketing similar to that done on funder column.
- Then we have run the classifier on the dataset. Few changes like making a data matrix for xgboost have been done according to the classifier functions and library used.

5. PROPOSED SOLUTIONS AND METHODS

Following Methods and classifiers have been used to check the best possible classifiers for the given problem. The classifiers which gave better accuracy results were further considered and worked with the parameter tuning in the next session

Decision trees

Decision trees are flexible .They are very good with categorical values. It is comparatively faster than the neural nets and deep learning techniques. As our task was classification we tried on decision trees and they gave considerate results

R package used: rpart

Random forests

Random forests are ensemble methods for classification and regression problems. The base classifiers used are decision trees. They construct multiple decision trees at the training time and gives the node of the trees as output in the testing time

R package used: randomforest

Bagging

Bootstrap aggregating or bagging is an ensemble methods .This method helps to avoid over fitting. Bootstrapping might not have a problem with noisy data. It does not work with stable methods.

R package used: adaBag, Ipred

Naïve Bayes

Naïve bayes are simply probabilistic classifiers based on the Bayes theorem with strong independence assumptions. It is very fast because it has no model building.

R package used: mass

Deeplearning

Deep learning is a new machine learning technique. These cascade to many layers of non-processing units for feature extraction and transformation

Rpackage used: h20

Xtreme Gradientboosting

Gradient Boosting is a machine learning technique which produces a prediction model in the form of weak prediction models .Generally decision trees are used as the classifiers. This is a very powerful technique.

Rpackage used: xgboost

The following table gives the initial accuracy which we tested on different classifiers

Classifier	**Average Accuracy**
Decision tree	76.517
Gradient Boosting	80.71
Random Forests	76.05
Deep Learning	68.185
Naïve Bayes	20
Bagging	54.34

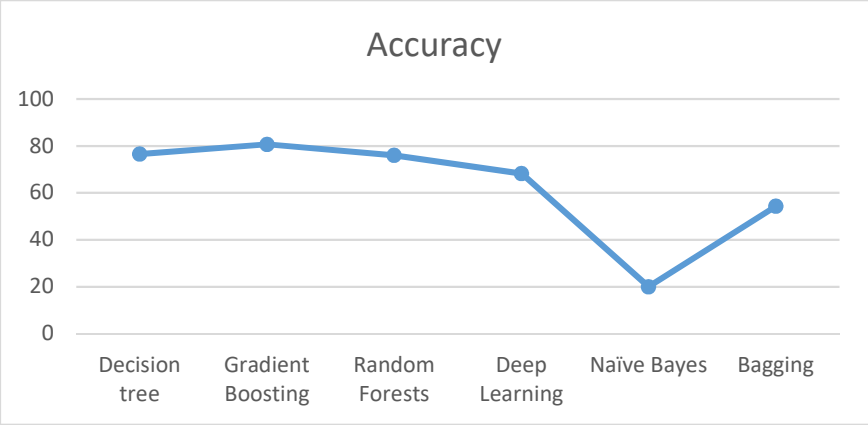

Fig 5.1 Average accuracies for classifiers

6. EXPERIMENTALRESULTS AND ANALYSIS

CLASSIFIER ANALYSIS AND PARAMETRE TUNING

Once we are done with the attribute selection and removal, we need to choose a learner to classify. We used Deep Learning techniques, Gradient Boosting, Decision Trees, and Random Forests

GRADIENT BOOSTING

XGBoost is one of the classifiers implementing .Ensemble methods and has proven to has better performance on various datsets.We've used "xgboost" library for implementing the xgboost algorithm, that includes a linear model and a tree learning algorithm. The following parameters are considered while implementing the "xgboost" method.

General Parameters

nthread = number of threads to be run in parallel

Tree Parameters

nfold = number of validation folds

max_depth = maximum depth of the trees

eta = step size to be shrinked after each boosting step

min_child_weight = minimum sum of instance weight in a child node(to stop further partition of the node)

subsample = sample ratio of training data

colsample_bytree = sample ratio of columns needed for tree construction

nrounds = number of passes on data

Learning Parameters

objective = learning task (Binary Classification or multiclass class classification)

num_class = number of classes

The following parameter values are fixed as these gives the best results

nthread=12,subsample=0.7,min_child_weight=3,colsample_bytree=0.5

```
Xtreme <- xgboost(data = matrix[,1:20],
                  nfold =5,
                  label = matrix[,21],
                  max_depth = 21,
                  eta = 0.02,
                  nthread = 12,
                  objective = "multi:softmax",
                  num_class =3,
                  subsample = 0.7,
                  colsample_bytree = 0.5,
                  min_child_weight = 3,
                  nrounds = 200,
                  maximize = FALSE)
```

Different runs for the Gradient Boosting

S.No	max_depth	Eta	nrounds	Training Accuracy	Validation Accuracy
1	20	0.2	400	100	80.54
2	15	0.2	400	100	80.257
3	21	0.2	400	100	80.31
4	21	0.5	350	100	79.57
5	21	0.1	250	99.99	80.74
6	**21**	**0.02**	**200**	**93.833**	**81.63**
7	21	0.02	150	93.57	81.97

By implementing the above tuned parameters on the testing data, we got highest accuracy for the parameters of the 6th run, which is **81.7**

```
Overall Statistics

               Accuracy : 0.8119
                 95% CI : (0.8016, 0.8219)
    No Information Rate : 0.5464
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.641
 Mcnemar's Test P-Value : < 2.2e-16

Statistics by Class:

                     Class: 0 Class: 1 Class: 2
Sensitivity            0.9170  0.27007   0.7622
Specificity            0.7210  0.98794   0.9184
Pos Pred Value         0.7984  0.63068   0.8528
Neg Pred Value         0.8782  0.94666   0.8616
Prevalence             0.5464  0.07086   0.3828
Detection Rate         0.5010  0.01914   0.2917
Detection Prevalence   0.6276  0.03034   0.3421
Balanced Accuracy      0.8190  0.62901   0.8403
```

Fig 6.1 Confusion matrix for the best run

Along with the accuracy, we have also considered other performance metrics from the confusion matrix such as precision, recall and ppv and npv

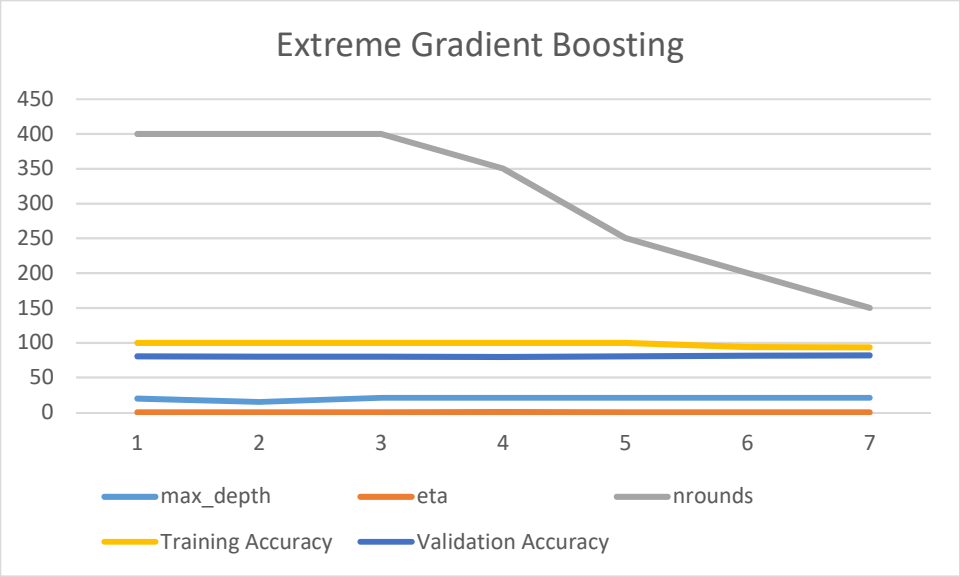

Fig 6.2 Graph comparing accuracy with the parameters

DECISION TREES

We have used rpart package for building the decision tree. The parameters tuned for improving with accuracy of the decision tree were:

- Minsplit- It is the minimum number of observations per leaf node.

- Cp- This controls the size of the decision tree by deciding the number of splits.
- Maxdepth- The maximum depth of the decision tree.

Different runs for the decision tree:

#run number	minsplit	cp	maxdepth	Accuracy
1.	10	0.0001	default	78.0812
2.	10	0.0001	5	70.83123
3.	10	0.0001	10	74.841
4.	10	0.0001	20	77.963
5.	100	0.0001	20	76.36
6.	5	0.0001	default	78.08
7.	5	0.00005	default	78.2835
8.	**5**	**0.000075**	**30**	**78.5**
9.	2	0.00001	30	75.46
10.	20	0.00001	30	77.255
11.	25	0.00001	30	77.00
12	25	0.00005	30	77.524

```
Overall Statistics

               Accuracy : 0.7781
                 95% CI : (0.7719, 0.7841)
    No Information Rate : 0.5382
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5847
 Mcnemar's Test P-Value : < 2.2e-16

Statistics by Class:

                     Class: functional Class: functional needs repair Class: non functional
Sensitivity                     0.8747                        0.29807                0.7336
Specificity                     0.7102                        0.97629                0.8917
Pos Pred Value                  0.7787                        0.49487                0.8120
Neg Pred Value                  0.8295                        0.94695                0.8399
Prevalence                      0.5382                        0.07229                0.3895
Detection Rate                  0.4708                        0.02155                0.2857
Detection Prevalence            0.6046                        0.04354                0.3518
Balanced Accuracy               0.7925                        0.63718                0.8126
```

Fig 6.3 Confusion Matrix for best run

Along with the accuracy ,we have also considered other performance metrics from the confusion matrix such as precision,recall and ppv and npv

The highest accuracy on training data set we achieved is 78.5. We have decided these as our best set of parameters. We have then populated the submission file for this and achieved an accuracy of **78.09** on submission.

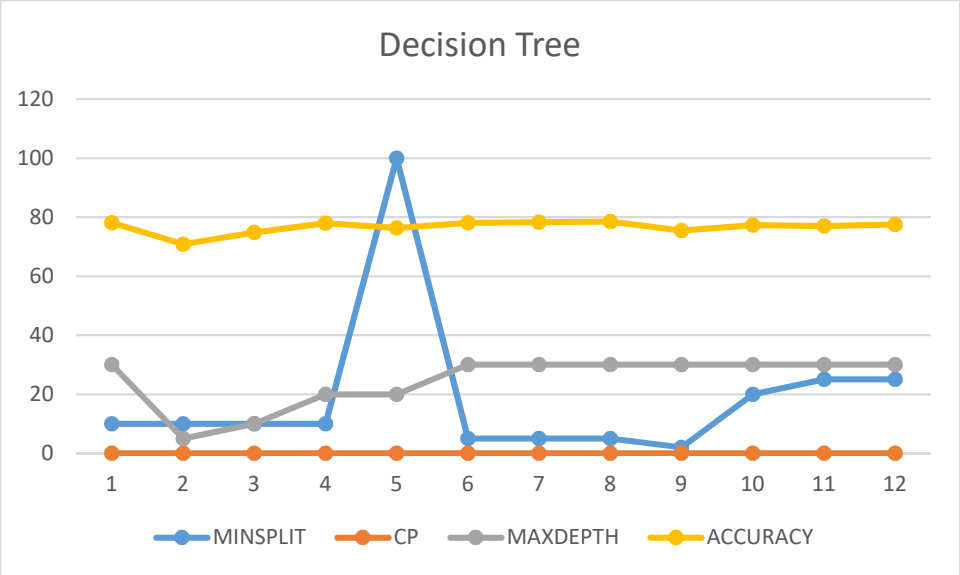

Fig 6.4 Graph comparing accuracy with the parameters

```
fit    <- rpart(status_group ~amount_tsh + funder + gps_height + longitude + latitude +
                  basin + region +  population + public_meeting + permit +  extraction_type_class
                + management + payment + water_quality + quantity + source + source_class +
                  waterpoint_type + OperationAge, data  = data.train, method  = 'class',
                parms = list(split  = "information"), control=rpart.control(minsplit=25, cp=0.00005))
DTprediction = predict(fit, data.test, type='class')
```

RANDOM FORESTS

Random Forest

The parameters tuned for improving with accuracy of the Random Forest were:

Parameters:

- mtry- the number of variables randomly selected at each split
- n-tree- number of trees to grow

Different runs for the Random Forest

#no of run	mtry	n-tree	Accuracy
1	10	1	74.08
2	15	1	73.75
3	5	1	74.01
4	5	2	72.01
5	10	10	79.32
6	10	15	79.55
7	**10**	**50**	**79.65**

```
Overall Statistics

               Accuracy : 0.7867
                 95% CI : (0.776, 0.797)
    No Information Rate : 0.5391
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6043
 Mcnemar's Test P-Value : < 2.2e-16

Statistics by Class:

                     Class: functional Class: functional needs repair Class: non functional
Sensitivity                     0.8601                        0.34339                0.7674
Specificity                     0.7405                        0.97410                0.8860
Pos Pred Value                  0.7949                        0.50859                0.8105
Neg Pred Value                  0.8190                        0.95002                0.8570
Prevalence                      0.5391                        0.07240                0.3885
Detection Rate                  0.4636                        0.02486                0.2982
Detection Prevalence            0.5832                        0.04888                0.3679
Balanced Accuracy               0.8003                        0.65875                0.8267
```

Fig 6.5 Confusion Matrix for the best run

Along with the accuracy ,we have also considered other performance metrics from the confusion matrix such as precision,recall and ppv and npv

The highest accuracy on training data set we achieved is 78.5. We have decided these as our best set of parameters. We have then populated the submission file for this and achieved an accuracy of **78.09** on submission.

```
model_forest <- randomForest(status_group ~amount_tsh + funder + gps_height + longitude + latitude +
                    basin + region +  population + public_meeting + permit +  extraction_type_class +
                    management + payment + water_quality + quantity + source + source_class +
                    waterpoint_type + OperationAge ,data = data.train, importance = TRUE,mtry = 10, ntree=1)
```

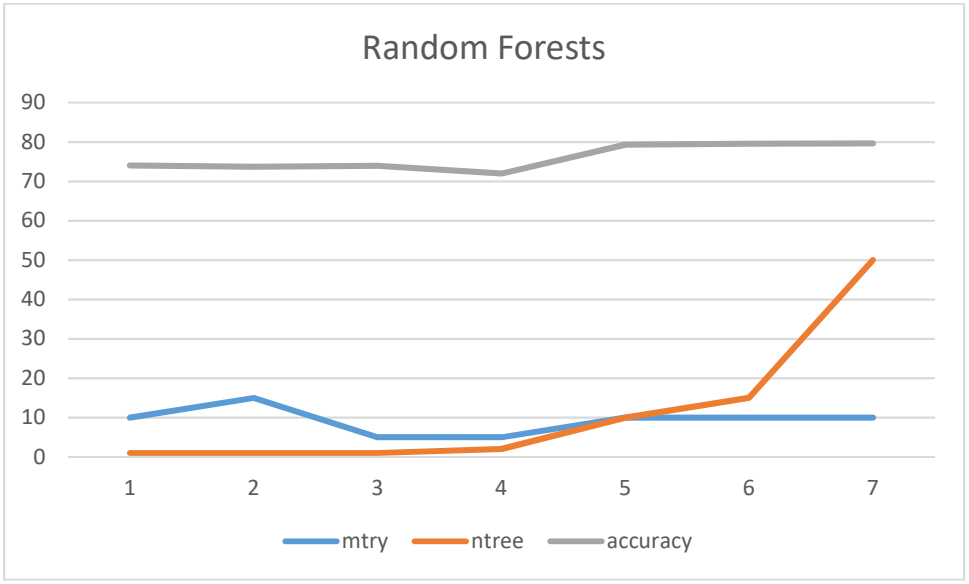

Fig 6.6 Graph comparing accuracy with the parameters

DEEP LEARNING

We used H20 package to build a multi-layer, feed forward, back-propagating neural network as our deep learning model. Below are the parameters we considered for building and tuning the model.

- Distribution – determines the type of predictions and the output nodes required.
- Activation Function – Weighted combination of input signals are activated through this function.
- Hidden – defines how many hidden layers to be built and the sets the number of neurons for each layer. This accounts to the complexity of the model.
- Epochs – Specifies how many times the network iterates thorough the dataset.
- Adaptive rate : Boolean attributes that is set to true or false for enabling adaptive learning rate.

```
deep_learning_model <- h2o.deeplearning(x=3:21, y="status_group",training_frame=splits[[1]]
                       validation_frame = splits[[2]],distribution="multinomial",
                       activation = "MaxoutWithDropout",hidden = c(285,285,285,285),
                       input_dropout_ratio = 0.2,hidden_dropout_ratios = c(0.1,0.1,0.1,0.1),
                       adaptive_rate=TRUE,sparse = TRUE,l1 = 1e-5,l2 = 1e-5,epochs = 20,
                       nfolds = 5,fold_assignment="Modulo")
```

Different runs for the Deep Learning

#no of run	Distribution	Activation Function	No. of hidden layers	No. of nodes in hidden layer	epochs	accuracy	Time taken (min)
1.	Quantile	RectifierWithDropout	3	250	50	67.01	130
2.	Quantile	MaxoutWIthDropout	4	220	100	70.6	190
3.	Quantile	MaxoutWIthDropout	5	200	100	63.12	240
4.	Multinomial	MaxoutWIthDropout	4	175	20	71.8	62
5.	Multinomial	MaxoutWIthDropout	3	205	20	64.02	68
6.	Multinomial	MaxoutWIthDropout	4	250	20	72.56	113

```
Confusion Matrix: Extract with `h2o.confusionMatrix(<model>,valid = TRUE)`)
=========================================================================
Confusion Matrix: vertical: actual; across: predicted
                        functional functional needs repair non functional  Error
functional                    6909                      56            1139 0.1475
functional needs repair        724                      71             248 0.9319
non functional                2013                       7            3696 0.3534
Totals                        9646                     134            5083 0.2817
                                Rate
functional                =  1,195 / 8,104
functional needs repair   =    972 / 1,043
non functional            =  2,020 / 5,716
Totals                    = 4,187 / 14,863
```

Fig 6.7 Confusion Matrix

Along with the accuracy ,we have also considered other performance metrics from the confusion matrix .

The highest accuracy on training data set we achieved is 72.56. We have decided these as our best set of parameters. We have then populated the submission file for this and achieved an accuracy of **50.67**on submission.

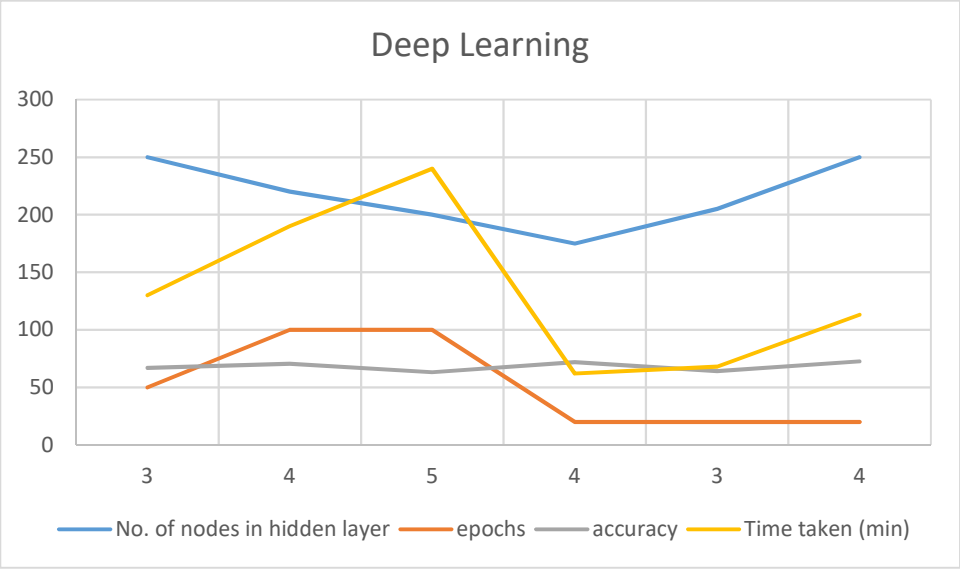

Fig: 6.8 Graph comparing accuracy with the parameters

7. CONCLUSION

The final algorithm used for submitting in the competition is gradient boosting. For Deep learning the tradeoff between accuracy of the model and the time consumed is not good enough to consider .Decision trees did not give good accuracies. Random forests was a strong learner and has a good theoretical base for prediction .But extreme gradient boosting gave promising results.

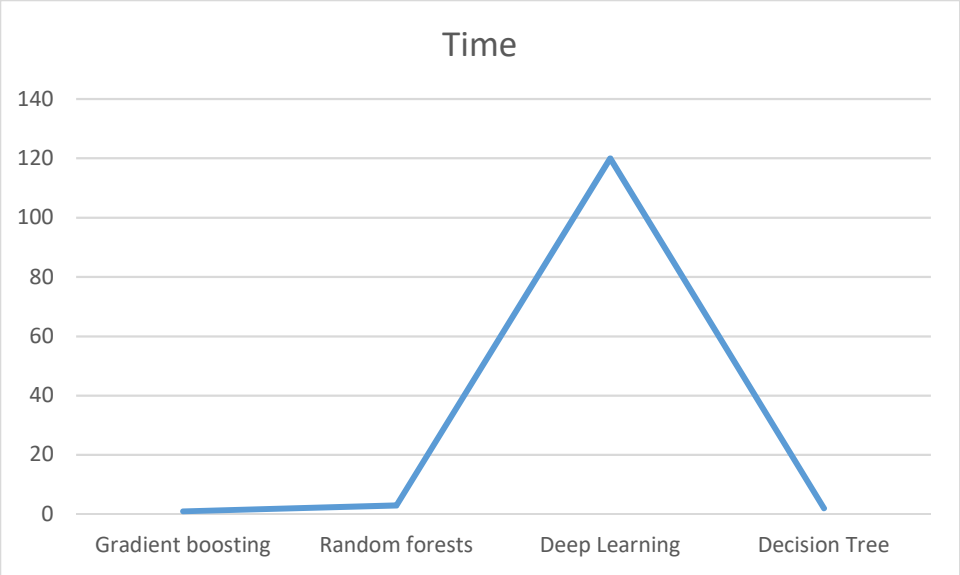

Fig: 7.1 Time graphs for all classifiers

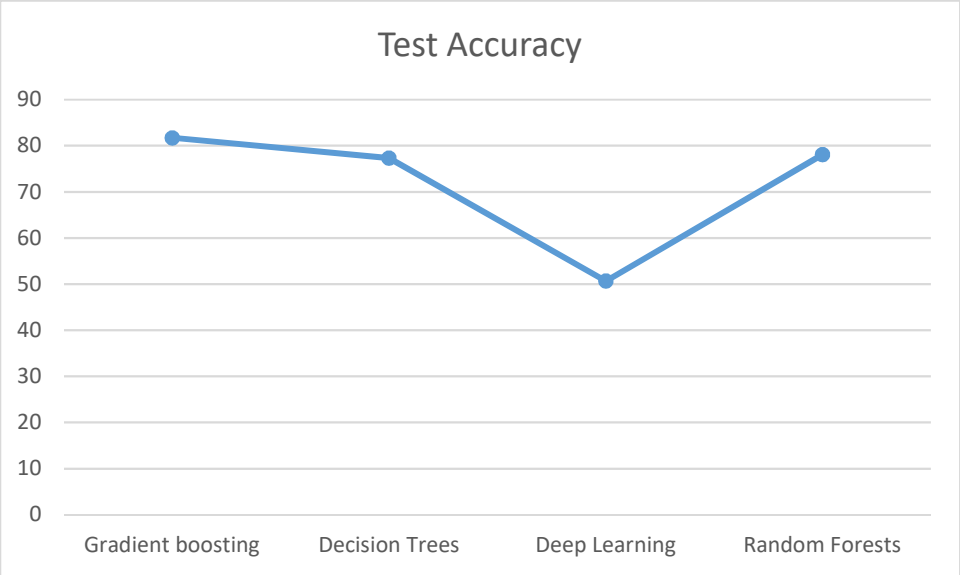

Fig: 7.2 Accuracy graphs for all classifiers

8. FUTURE WORK

In the future the data preprocessing can be handled even more sophisticatedly. For example the attributes longitude and latitude could be treated as (x,y) coordinates that can be used to convert into distance from a fixed point . This way we are reducing the attributes as well as making use of their values.

Even the attributes such as regions could be bucketed into north, south, east and west zones by gaining proper geographical knowledge of Tanzania.

Many more such improvisations can improve the classification power of the classifier. In future we can improve our classifier further by using such techniques

9. CONTRIBUTION OF TEAM MEMBERS

Understanding the datasets	Teja,Sowmya,Mohana,Gayathri
Preprocessing data	Teja,Sowmya,Mohana,Gayathri
Status report	Teja,Mohana
Proposed solutions	Teja,Sowmya,Mohana,Gayathri
Decision tree	Mohana
Random Forests	Sowmya
Xgboost	Teja
Deep Learning	Gayathri
Report	Teja,Sowmya,Mohana,Gayathri

10. REFERENCES

[1] “Machine Learning Report format” CS 39IL1: Machine Learning

[2] “R news and tutorials” R Bloggers

[3] “Welcome to deep Learning” Deep Learning

[4] “Correlation based feature selection for Machine Learning”, Mark A Hill, April 1999