Practica_2_final

Ana Marrodan i Mireia Solanich

03/01/2021

0.Introducció

El següent projecte s'inclou dins l'assignatura 'Tipologia i cicle de vida de les dades' del programa del màster Ciència de Dades de la UOC.

Integrants del projecte: - Ana Marrodan Badell - Mireia Solanich Ventura

1. Descripció del dataset. Perquè és important i quina pregunta/problema pretén respondre?

l'objecte del següent projecte cosisteix en predir quines de les següents bombes d'aigua són o no defectuoses.

A partir de les dades facilitades per Taarifa i el Ministeri d'Aigua de Tanzània, caldrà predir quines bombes són funcionals, quines necessiten algunes reparacions i quines no funcionen en absolut. Per a fer la classificació del funcionament dels pous caldrà fer-ne l'analisis a partir de les variables facilitades sobre quin tipus de bomba, quan es van instal · lar i com es gestiona.

El següent estudi permetrà coneixer i predir quins punts d'aigua fracassaran i permetrà millorar les operacions de manteniment i assegurar que l'aigua potable i neta estigui disponible per a les comunitats de Tanzània.

El projecte descrit forma part de la competició activa a a la plataforma "DrivenData.org" (DrivenData.org)

Per a realitzar els següent estudi es faciliten 4 fitxers 'csv': - Submission format: El format per enviar les vostres prediccions - Test set values: Les variables independents que necessiten prediccions - Training set labels: La variable dependent (status_group) de cadascuna de les files dels valors del conjunt d'entrenament - Training set values: Les variables independents del conjunt d'entrenament

Descripció dels camps:

Fitxers 'SubmissionFormat.csv' i 'training_set_lablels.csv' contenten els següents camps: - **id**: Codi identificador de cada pou - **status_group**: Estat del funcionament de cada pou

Fitxers 'test_set_values.csv' i 'training_set_lablels.csv' contenten els següents camps: - **id**: Codi identificador de cada pou - **amount_tsh**: Quantitat d'aigua disponible a cada pou (Total static head) - **date_recorded**: Data en que s'ha creat el registre - **funder**: Qui a financiat el pou - **gps_height**: altitud del pou GPS - **installer**: Organitzacio que va instal · lar el pou - **longitude**: Coordenada longitud GPS - **latitude**: Coordenada latitud GPS - **wpt_name**: Nom del punt d'aigua (si n'hi ha) - **num_private**: Conca hidrogràfica - **basin**: Localització conca hidrogràfica - **subvillage**: Localització geogràfica - **region**: Ubicació geogràfica de la regió - **region_code**: Codi ubicació geogràfica de la regió - **district_code**: Codi ubicació geogràfica del districte - **lga**: Localització geogràfica - **ward**: Localització geogràfica - **population**: Població al voltant del pou - **public_meeting**: cert / fals - **recorded_by**: Grup que introdueix aquesta fila de dades - **scheme_management**: Qui opera el punt d'aigua - **scheme_name**: Qui opera el punt d'aigua - **permit**: Si es permet el punt d'aigua - **construction_year**: Any en què es va construir el punt d'aigua - **extraction_type**: Tipus d'extracció que fa servir el punt d'aigua - **extraction_type_group**: Tipus d'extracció que fa servir el punt d'aigua - **extraction_type_class**: Tipus d'extracció que fa servir el punt

d'aigua - **management**: Com es gestiona el punt d'aigua - **management_group**: Com es gestiona el punt d'aigua - **payment**: Què costa l'aigua - **payment_type**: El que costa l'aigua - **water_quality**: Qualitat de l'aigua - **quality_group**: Qualitat de l'aigua - **quantity**: Quantitat d'aigua - **quantity_group**: Quantitat d'aigua - **source**: Font de l'aigua - **source_type**: Font de l'aigua - **source_class**: Font de l'aigua - **waterpoint_type**: Tipus de punt d'aigua - **waterpoint_type_group**: Tipus de punt d'aigua

Carreguem els paquets R que utilitzarem

```
library(ggplot2)
library(dplyr)
library(knitr)
library(caret)
library(rminer)
library(randomForest)
```

2. Integració i selecció de les dades d'interès a analitzar.

```
water_pumps <- read.csv('data/training_set_values.csv', header = TRUE,
                        sep = ',', stringsAsFactors = FALSE)
water_pumps_class <- read.csv('data/training_set_lablels.csv',
                              header = TRUE, sep = ',', stringsAsFactors = FALSE)

water_pumps_complete <- water_pumps %>% inner_join(water_pumps_class, by = "id")

head(water_pumps,n=5)
```

```
##      id amount_tsh date_recorded       funder gps_height    installer longitude
## 1 69572       6000    2011-03-14        Roman       1390        Roman  34.93809
## 2  8776          0    2013-03-06      Grumeti       1399      GRUMETI  34.69877
## 3 34310         25    2013-02-25 Lottery Club        686 World vision  37.46066
## 4 67743          0    2013-01-28       Unicef        263       UNICEF  38.48616
## 5 19728          0    2011-07-13  Action In A          0      Artisan  31.13085
##     latitude            wpt_name num_private                 basin
## 1  -9.856322                none           0             Lake Nyasa
## 2  -2.147466            Zahanati           0           Lake Victoria
## 3  -3.821329          Kwa Mahundi           0                 Pangani
## 4 -11.155298 Zahanati Ya Nanyumbu           0 Ruvuma / Southern Coast
## 5  -1.825359              Shuleni           0           Lake Victoria
##   subvillage  region region_code district_code       lga       ward population
## 1   Mnyusi B  Iringa          11             5    Ludewa   Mundindi        109
## 2    Nyamara    Mara          20             2 Serengeti      Natta        280
## 3    Majengo Manyara          21             4 Simanjiro    Ngorika        250
## 4 Mahakamani  Mtwara          90            63  Nanyumbu   Nanyumbu         58
## 5 Kyanyamisa  Kagera          18             1   Karagwe Nyakasimbi          0
##   public_meeting             recorded_by scheme_management
## 1           True GeoData Consultants Ltd              VWC
## 2                GeoData Consultants Ltd            Other
## 3           True GeoData Consultants Ltd              VWC
## 4           True GeoData Consultants Ltd              VWC
```

```
## 5           True GeoData Consultants Ltd
##                    scheme_name permit construction_year extraction_type
## 1                        Roman  False              1999         gravity
## 2                              True              2010         gravity
## 3 Nyumba ya mungu pipe scheme   True              2009         gravity
## 4                              True              1986     submersible
## 5                              True                 0         gravity
##   extraction_type_group extraction_type_class management management_group
## 1               gravity               gravity        vwc       user-group
## 2               gravity               gravity        wug       user-group
## 3               gravity               gravity        vwc       user-group
## 4            submersible           submersible        vwc       user-group
## 5               gravity               gravity      other            other
##          payment payment_type water_quality quality_group     quantity
## 1   pay annually     annually          soft          good       enough
## 2      never pay    never pay          soft          good insufficient
## 3 pay per bucket   per bucket          soft          good       enough
## 4      never pay    never pay          soft          good          dry
## 5      never pay    never pay          soft          good     seasonal
##   quantity_group               source          source_type source_class
## 1         enough               spring               spring  groundwater
## 2   insufficient rainwater harvesting rainwater harvesting      surface
## 3         enough                  dam                  dam      surface
## 4            dry          machine dbh             borehole  groundwater
## 5       seasonal rainwater harvesting rainwater harvesting      surface
##               waterpoint_type waterpoint_type_group
## 1          communal standpipe    communal standpipe
## 2          communal standpipe    communal standpipe
## 3 communal standpipe multiple    communal standpipe
## 4 communal standpipe multiple    communal standpipe
## 5          communal standpipe    communal standpipe
```

```
head(water_pumps_class,n=5)
```

```
##      id   status_group
## 1 69572     functional
## 2  8776     functional
## 3 34310     functional
## 4 67743 non functional
## 5 19728     functional
```

```
head(water_pumps_complete,n=5)
```

```
##      id amount_tsh date_recorded       funder gps_height    installer longitude
## 1 69572       6000    2011-03-14        Roman       1390        Roman  34.93809
## 2  8776          0    2013-03-06      Grumeti       1399      GRUMETI  34.69877
## 3 34310         25    2013-02-25 Lottery Club        686 World vision  37.46066
## 4 67743          0    2013-01-28       Unicef        263       UNICEF  38.48616
## 5 19728          0    2011-07-13  Action In A          0      Artisan  31.13085
##     latitude             wpt_name num_private                 basin
## 1  -9.856322                 none           0             Lake Nyasa
## 2  -2.147466              Zahanati           0          Lake Victoria
## 3  -3.821329          Kwa Mahundi           0                Pangani
```

```
## 4 -11.155298 Zahanati Ya Nanyumbu           0 Ruvuma / Southern Coast
## 5  -1.825359              Shuleni           0           Lake Victoria
##   subvillage  region region_code district_code       lga       ward population
## 1   Mnyusi B  Iringa          11             5    Ludewa   Mundindi        109
## 2    Nyamara    Mara          20             2 Serengeti      Natta        280
## 3    Majengo Manyara          21             4 Simanjiro    Ngorika        250
## 4 Mahakamani  Mtwara          90            63  Nanyumbu   Nanyumbu         58
## 5 Kyanyamisa  Kagera          18             1   Karagwe Nyakasimbi          0
##   public_meeting             recorded_by scheme_management
## 1           True GeoData Consultants Ltd               VWC
## 2                GeoData Consultants Ltd             Other
## 3           True GeoData Consultants Ltd               VWC
## 4           True GeoData Consultants Ltd               VWC
## 5           True GeoData Consultants Ltd
##                   scheme_name permit construction_year extraction_type
## 1                       Roman  False              1999         gravity
## 2                             True              2010         gravity
## 3 Nyumba ya mungu pipe scheme   True              2009         gravity
## 4                             True              1986     submersible
## 5                             True                 0         gravity
##   extraction_type_group extraction_type_class management management_group
## 1               gravity               gravity        vwc       user-group
## 2               gravity               gravity        wug       user-group
## 3               gravity               gravity        vwc       user-group
## 4           submersible           submersible        vwc       user-group
## 5               gravity               gravity      other            other
##          payment payment_type water_quality quality_group     quantity
## 1   pay annually     annually          soft          good       enough
## 2      never pay    never pay          soft          good insufficient
## 3 pay per bucket   per bucket          soft          good       enough
## 4      never pay    never pay          soft          good          dry
## 5      never pay    never pay          soft          good     seasonal
##   quantity_group               source          source_type source_class
## 1         enough               spring               spring  groundwater
## 2   insufficient rainwater harvesting rainwater harvesting      surface
## 3         enough                  dam                  dam      surface
## 4            dry          machine dbh             borehole  groundwater
## 5       seasonal rainwater harvesting rainwater harvesting      surface
##               waterpoint_type waterpoint_type_group   status_group
## 1          communal standpipe    communal standpipe     functional
## 2          communal standpipe    communal standpipe     functional
## 3 communal standpipe multiple    communal standpipe     functional
## 4 communal standpipe multiple    communal standpipe non functional
## 5          communal standpipe    communal standpipe     functional
```

3. Neteja de les dades.

3.1. Les dades contenen zeros o elements buits? Com gestionaries aquests casos? + 4.1. Selecció dels grups de dades que es volen analitzar/comparar

En aquest apartat farem un primer anàlisi dels camps que ens permetrà comprobar la compleció de les dades així com descartar unes quantes dimensions de cara al nostre anàlisi final en base a la seva qualitat.

```r
# Carreguem els paquets R que utilitzarem

#summary(water_pumps)
#summary(water_pumps_class)
summary(water_pumps_complete)
```

```
##        id          amount_tsh       date_recorded         funder
##  Min.   :    0   Min.   :     0.0   Length:59400       Length:59400
##  1st Qu.:18520   1st Qu.:     0.0   Class :character   Class :character
##  Median :37062   Median :     0.0   Mode  :character   Mode  :character
##  Mean   :37115   Mean   :   317.7
##  3rd Qu.:55657   3rd Qu.:    20.0
##  Max.   :74247   Max.   :350000.0
##    gps_height      installer           longitude        latitude
##  Min.   : -90.0   Length:59400       Min.   : 0.00   Min.   :-11.649
##  1st Qu.:   0.0   Class :character   1st Qu.:33.09   1st Qu.: -8.541
##  Median : 369.0   Mode  :character   Median :34.91   Median : -5.022
##  Mean   : 668.3                      Mean   :34.08   Mean   : -5.706
##  3rd Qu.:1319.2                      3rd Qu.:37.18   3rd Qu.: -3.326
##  Max.   :2770.0                      Max.   :40.35   Max.   :  0.000
##    wpt_name          num_private           basin            subvillage
##  Length:59400       Min.   :   0.0000   Length:59400       Length:59400
##  Class :character   1st Qu.:   0.0000   Class :character   Class :character
##  Mode  :character   Median :   0.0000   Mode  :character   Mode  :character
##                     Mean   :   0.4741
##                     3rd Qu.:   0.0000
##                     Max.   :1776.0000
##     region           region_code   district_code       lga
##  Length:59400       Min.   : 1.0   Min.   : 0.00   Length:59400
##  Class :character   1st Qu.: 5.0   1st Qu.: 2.00   Class :character
##  Mode  :character   Median :12.0   Median : 3.00   Mode  :character
##                     Mean   :15.3   Mean   : 5.63
##                     3rd Qu.:17.0   3rd Qu.: 5.00
##                     Max.   :99.0   Max.   :80.00
##      ward             population      public_meeting     recorded_by
##  Length:59400       Min.   :    0.0   Length:59400       Length:59400
##  Class :character   1st Qu.:    0.0   Class :character   Class :character
##  Mode  :character   Median :   25.0   Mode  :character   Mode  :character
##                     Mean   :  179.9
##                     3rd Qu.:  215.0
##                     Max.   :30500.0
##  scheme_management  scheme_name           permit          construction_year
##  Length:59400       Length:59400       Length:59400       Min.   :   0
##  Class :character   Class :character   Class :character   1st Qu.:   0
##  Mode  :character   Mode  :character   Mode  :character   Median :1986
##                                                           Mean   :1301
##                                                           3rd Qu.:2004
##                                                           Max.   :2013
##  extraction_type    extraction_type_group extraction_type_class
##  Length:59400       Length:59400          Length:59400
##  Class :character   Class :character      Class :character
##  Mode  :character   Mode  :character      Mode  :character
##
```

```
##
##
##   management        management_group     payment           payment_type
##  Length:59400       Length:59400       Length:59400       Length:59400
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  water_quality      quality_group        quantity          quantity_group
##  Length:59400       Length:59400       Length:59400       Length:59400
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##     source          source_type        source_class       waterpoint_type
##  Length:59400       Length:59400       Length:59400       Length:59400
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##  waterpoint_type_group status_group
##  Length:59400          Length:59400
##  Class :character      Class :character
##  Mode  :character      Mode  :character
##
##
##
```

```
#lapply(water_pumps_complete, class)
```

Mirem únicament a quin tipus correspont a cada variable

```
lapply(water_pumps_complete, class)
```

```
## $id
## [1] "integer"
##
## $amount_tsh
## [1] "numeric"
##
## $date_recorded
## [1] "character"
##
## $funder
## [1] "character"
##
## $gps_height
## [1] "integer"
##
## $installer
```

```
## [1] "character"
##
## $longitude
## [1] "numeric"
##
## $latitude
## [1] "numeric"
##
## $wpt_name
## [1] "character"
##
## $num_private
## [1] "integer"
##
## $basin
## [1] "character"
##
## $subvillage
## [1] "character"
##
## $region
## [1] "character"
##
## $region_code
## [1] "integer"
##
## $district_code
## [1] "integer"
##
## $lga
## [1] "character"
##
## $ward
## [1] "character"
##
## $population
## [1] "integer"
##
## $public_meeting
## [1] "character"
##
## $recorded_by
## [1] "character"
##
## $scheme_management
## [1] "character"
##
## $scheme_name
## [1] "character"
##
## $permit
## [1] "character"
##
## $construction_year
```

```
## [1] "integer"
##
## $extraction_type
## [1] "character"
##
## $extraction_type_group
## [1] "character"
##
## $extraction_type_class
## [1] "character"
##
## $management
## [1] "character"
##
## $management_group
## [1] "character"
##
## $payment
## [1] "character"
##
## $payment_type
## [1] "character"
##
## $water_quality
## [1] "character"
##
## $quality_group
## [1] "character"
##
## $quantity
## [1] "character"
##
## $quantity_group
## [1] "character"
##
## $source
## [1] "character"
##
## $source_type
## [1] "character"
##
## $source_class
## [1] "character"
##
## $waterpoint_type
## [1] "character"
##
## $waterpoint_type_group
## [1] "character"
##
## $status_group
## [1] "character"
```

verifiquem la dimensió de les taules

```
dim(water_pumps)
```

```
## [1] 59400    40
```

```
dim(water_pumps_class)
```

```
## [1] 59400     2
```

```
dim(water_pumps_complete)
```

```
## [1] 59400    41
```

renombrem la taula per a treballar amb ella

```
water <- water_pumps_complete
```

convertim els camps 'string' en factors i verifiquem

```
# factoritzar
cols<-c("date_recorded","funder","installer","wpt_name","basin","subvillage",
        "region","lga","ward","public_meeting","recorded_by",
        "scheme_management","scheme_name","permit","extraction_type",
        "extraction_type_group","extraction_type_class","management",
        "management_group","payment","payment_type","water_quality",
        "quality_group","quantity","quantity_group","source","source_type",
        "source_class","waterpoint_type","waterpoint_type_group")
for (i in cols){
  water[,i] <- as.factor(water[,i])
}


lapply(water, class)
```

```
## $id
## [1] "integer"
##
## $amount_tsh
## [1] "numeric"
##
## $date_recorded
## [1] "factor"
##
## $funder
## [1] "factor"
##
## $gps_height
## [1] "integer"
##
## $installer
## [1] "factor"
```

```
##
## $longitude
## [1] "numeric"
##
## $latitude
## [1] "numeric"
##
## $wpt_name
## [1] "factor"
##
## $num_private
## [1] "integer"
##
## $basin
## [1] "factor"
##
## $subvillage
## [1] "factor"
##
## $region
## [1] "factor"
##
## $region_code
## [1] "integer"
##
## $district_code
## [1] "integer"
##
## $lga
## [1] "factor"
##
## $ward
## [1] "factor"
##
## $population
## [1] "integer"
##
## $public_meeting
## [1] "factor"
##
## $recorded_by
## [1] "factor"
##
## $scheme_management
## [1] "factor"
##
## $scheme_name
## [1] "factor"
##
## $permit
## [1] "factor"
##
## $construction_year
## [1] "integer"
```

```
##
## $extraction_type
## [1] "factor"
##
## $extraction_type_group
## [1] "factor"
##
## $extraction_type_class
## [1] "factor"
##
## $management
## [1] "factor"
##
## $management_group
## [1] "factor"
##
## $payment
## [1] "factor"
##
## $payment_type
## [1] "factor"
##
## $water_quality
## [1] "factor"
##
## $quality_group
## [1] "factor"
##
## $quantity
## [1] "factor"
##
## $quantity_group
## [1] "factor"
##
## $source
## [1] "factor"
##
## $source_type
## [1] "factor"
##
## $source_class
## [1] "factor"
##
## $waterpoint_type
## [1] "factor"
##
## $waterpoint_type_group
## [1] "factor"
##
## $status_group
## [1] "character"
```

comprovem el nombre de files sense NA i nombre de files amb NA

```
(c.cases <- sum(complete.cases(water)) )
```

```
## [1] 59400
```

```
(na.cases <- nrow(water) - c.cases)
```

```
## [1] 0
```

veiem que no hi ha cap fila amb valor NA.

comprobem quants camps estan buits i en aquest cas els subtituirem per NA, després en tornem a fer el recompte

```
water[,][water[,] == ""] <- NA
kable(colSums(is.na(water)), col.names = c("NA Count") )
```

	NA Count
id	0
amount_tsh	0
date_recorded	0
funder	3635
gps_height	0
installer	3655
longitude	0
latitude	0
wpt_name	0
num_private	0
basin	0
subvillage	371
region	0
region_code	0
district_code	0
lga	0
ward	0
population	0
public_meeting	3334
recorded_by	0
scheme_management	3877
scheme_name	28166
permit	3056
construction_year	0
extraction_type	0
extraction_type_group	0
extraction_type_class	0
management	0
management_group	0
payment	0
payment_type	0
water_quality	0
quality_group	0
quantity	0

	NA Count
quantity_group	0
source	0
source_type	0
source_class	0
waterpoint_type	0
waterpoint_type_group	0
status_group	0

```
(c.cases <- sum(complete.cases(water)) )
```

```
## [1] 27813
```

```
(na.cases <- nrow(water) - c.cases)
```

```
## [1] 31587
```

```
na.cases / nrow(water)
```

```
## [1] 0.5317677
```

Tenim 27813 casos complets después de completar els valors de la cadena en blanc amb NA

31587 files amb NA

53,17% de las files tenen NA

Analisis dels camps

Primer de tot renombrarem la columna 'id' ja que cada registre és únic

```
rownames(water) <- water$id
water$id <- NULL
head(water, n=5)
```

```
##       amount_tsh date_recorded       funder gps_height    installer longitude
## 69572       6000    2011-03-14        Roman       1390        Roman  34.93809
## 8776           0    2013-03-06      Grumeti       1399      GRUMETI  34.69877
## 34310         25    2013-02-25 Lottery Club        686 World vision  37.46066
## 67743          0    2013-01-28       Unicef        263       UNICEF  38.48616
## 19728          0    2011-07-13  Action In A          0      Artisan  31.13085
##         latitude             wpt_name num_private                 basin
## 69572  -9.856322                 none           0             Lake Nyasa
## 8776   -2.147466              Zahanati           0           Lake Victoria
## 34310  -3.821329          Kwa Mahundi           0                Pangani
## 67743 -11.155298 Zahanati Ya Nanyumbu           0 Ruvuma / Southern Coast
## 19728  -1.825359               Shuleni           0           Lake Victoria
##       subvillage  region region_code district_code       lga       ward
## 69572   Mnyusi B  Iringa          11             5    Ludewa   Mundindi
## 8776     Nyamara    Mara          20             2 Serengeti      Natta
```

```
## 34310    Majengo Manyara          21             4 Simanjiro    Ngorika
## 67743 Mahakamani  Mtwara          90            63  Nanyumbu   Nanyumbu
## 19728 Kyanyamisa  Kagera          18             1   Karagwe Nyakasimbi
##       population public_meeting             recorded_by scheme_management
## 69572        109           True GeoData Consultants Ltd              VWC
## 8776         280           <NA> GeoData Consultants Ltd            Other
## 34310        250           True GeoData Consultants Ltd              VWC
## 67743         58           True GeoData Consultants Ltd              VWC
## 19728          0           True GeoData Consultants Ltd             <NA>
##                      scheme_name permit construction_year extraction_type
## 69572                      Roman  False              1999         gravity
## 8776                       <NA>   True              2010         gravity
## 34310 Nyumba ya mungu pipe scheme   True              2009         gravity
## 67743                      <NA>   True              1986     submersible
## 19728                      <NA>   True                 0         gravity
##       extraction_type_group extraction_type_class management management_group
## 69572               gravity               gravity        vwc       user-group
## 8776                gravity               gravity        wug       user-group
## 34310               gravity               gravity        vwc       user-group
## 67743           submersible           submersible        vwc       user-group
## 19728               gravity               gravity      other            other
##              payment payment_type water_quality quality_group     quantity
## 69572   pay annually     annually          soft          good       enough
## 8776       never pay    never pay          soft          good insufficient
## 34310 pay per bucket   per bucket          soft          good       enough
## 67743      never pay    never pay          soft          good          dry
## 19728      never pay    never pay          soft          good     seasonal
##       quantity_group               source          source_type source_class
## 69572         enough               spring               spring  groundwater
## 8776    insufficient rainwater harvesting rainwater harvesting      surface
## 34310         enough                  dam                  dam      surface
## 67743            dry          machine dbh             borehole  groundwater
## 19728       seasonal rainwater harvesting rainwater harvesting      surface
##                  waterpoint_type waterpoint_type_group   status_group
## 69572          communal standpipe    communal standpipe     functional
## 8776           communal standpipe    communal standpipe     functional
## 34310 communal standpipe multiple    communal standpipe     functional
## 67743 communal standpipe multiple    communal standpipe non functional
## 19728          communal standpipe    communal standpipe     functional
```

anem a veure quants registres únics té cada camp

```
# Valors unics: Per a quines variables tindria sentit un procés de discretització?
apply(water,2, function(x) length(unique(x)))
```

```
##            amount_tsh         date_recorded               funder
##                    98                   356                 1898
##            gps_height            installer             longitude
##                  2428                  2146                 55366
##              latitude               wpt_name           num_private
##                 57517                 37400                    65
##                 basin             subvillage                region
##                     9                 19288                    21
```

```
##           region_code         district_code                   lga
##                    27                    20                   125
##                  ward            population        public_meeting
##                  2092                  1049                     3
##           recorded_by     scheme_management          scheme_name
##                     1                    13                  2697
##                permit     construction_year       extraction_type
##                     3                    55                    18
## extraction_type_group extraction_type_class           management
##                    13                     7                    12
##      management_group               payment          payment_type
##                     5                     7                     7
##         water_quality         quality_group              quantity
##                     8                     6                     5
##        quantity_group                source           source_type
##                     5                    10                     7
##          source_class       waterpoint_type waterpoint_type_group
##                     3                     7                     6
##          status_group
##                     3
```

A continuació analitzarem els camps, en algunes ocasions, podem comprovar que hi ha camps que son iguals o similar a altres. Amb aquest estudi pretenem veure quins camps poden ser descartats i ens permetra **reduirne la dimensionalitat**.

amount_tsh[1]

```
q_atsh <- count(water, amount_tsh) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_atsh,n=5)
```

```
##   amount_tsh     n      perc
## 1          0 41639 70.099327
## 2        500  3102  5.222222
## 3         50  2472  4.161616
## 4       1000  1488  2.505051
## 5         20  1463  2.462963
```

veiem que aproximadament el 70% dels registres son 0, però en el cas d'aqeust dataset, i donada la importàcia d'aquest camp els considerarem zeros significatius i els deixarem intactes

date_recorded [2]

```
q_date <- count(water, date_recorded) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_date,n=5)
```

```
##   date_recorded   n      perc
## 1    2011-03-15 572 0.9629630
```

```
## 2    2011-03-17 558 0.9393939
## 3    2013-02-03 546 0.9191919
## 4    2011-03-14 520 0.8754209
## 5    2011-03-16 513 0.8636364
```

no hi ha cap registre que destaqui entre ells, segons la definició del camp podem descartar aquest camp ja que tan sols ens indica quan es va introduir el registre al dataset.

funder[3]

```
q_funder <- count(water, funder) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_funder,n=5)
```

```
##                   funder    n      perc
## 1 Government Of Tanzania 9084 15.292929
## 2                   <NA> 3635  6.119529
## 3                 Danida 3114  5.242424
## 4                 Hesawa 2202  3.707071
## 5                  Rwssp 1374  2.313131
```

veiem que el 15,29% dels pous son fundats per part del Govern de Tanzania, a més veiem que un 6% dels registres són desconeguts. Aquest camp també el descartarem ja que només ens indica el fundador i no es preveu que tingui relació amb el funcionament de les bombes de bombeig

gps_height[4]

```
q_gpshe <- count(water, gps_height) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_gpshe, n=5)
```

```
##   gps_height     n        perc
## 1          0 20438 34.40740741
## 2        -15    60  0.10101010
## 3        -16    55  0.09259259
## 4        -13    55  0.09259259
## 5        -20    52  0.08754209
```

hem obtingut que el 34,4% dels registres són 0.

installer[5]

```
q_instal <- count(water, installer) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_instal,n=5)
```

```
##    installer     n      perc
## 1        DWE 17402 29.296296
## 2       <NA>  3655  6.153199
## 3 Government  1825  3.072391
## 4        RWE  1206  2.030303
## 5      Commu  1060  1.784512
```

al igual que funder, també descartarem aquest camp, no aporta informació del funcionament del pous.

longitude[6], latitude[7]

```
q_longitude <- count(water, longitude) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_longitude,n=5)
```

```
##   longitude    n        perc
## 1   0.00000 1812 3.050505051
## 2  31.61953    2 0.003367003
## 3  32.91986    2 0.003367003
## 4  32.92489    2 0.003367003
## 5  32.92601    2 0.003367003
```

```
q_latitude <- count(water, latitude) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_latitude,n=5)
```

```
##      latitude    n        perc
## 1 -0.00000002 1812 3.050505051
## 2 -9.28934920    2 0.003367003
## 3 -7.17720290    2 0.003367003
## 4 -7.17715478    2 0.003367003
## 5 -7.17517443    2 0.003367003
```

Ubicació geografica dels pous/bombes, veiem que el 3% del registres son descartables, ja que aquestes coodenades estan ubicades al mar.

wpt_name[8]

```
q_wpt <- count(water, wpt_name) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_wpt,n=5)
```

```
##    wpt_name    n      perc
## 1      none 3563 5.9983165
## 2   Shuleni 1748 2.9427609
## 3  Zahanati  830 1.3973064
## 4 Msikitini  535 0.9006734
## 5  Kanisani  323 0.5437710
```

camp descartat.

num_private[9]

```
q_priv <- count(water, num_private) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_priv,n=5)
```

```
##   num_private     n        perc
## 1           0 58643 98.72558923
## 2           6    81  0.13636364
## 3           1    73  0.12289562
## 4           5    46  0.07744108
## 5           8    46  0.07744108
```

camp descartat, el 98,7% dels registres són 0

basin[10], subvillage[11]

```
q_basin <- count(water, basin) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
q_basin
```

```
##                     basin     n      perc
## 1           Lake Victoria 10248 17.252525
## 2                 Pangani  8940 15.050505
## 3                  Rufiji  7976 13.427609
## 4                Internal  7785 13.106061
## 5         Lake Tanganyika  6432 10.828283
## 6             Wami / Ruvu  5987 10.079125
## 7              Lake Nyasa  5085  8.560606
## 8 Ruvuma / Southern Coast  4493  7.563973
## 9              Lake Rukwa  2454  4.131313
```

```
q_subvillage <- count(water, subvillage) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_subvillage, n=5)
```

```
##   subvillage   n      perc
## 1   Madukani 508 0.8552189
## 2    Shuleni 506 0.8518519
## 3    Majengo 502 0.8451178
## 4       Kati 373 0.6279461
## 5       <NA> 371 0.6245791
```

Camps descartats, és informació geogràfica (que ja tenim en altres camps) amb molt alta granularitat.

region[12], region_code[13], district_code[14]

```
q_region <- count(water, region) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
q_region
```

```
##           region    n     perc
## 1         Iringa 5294 8.912458
## 2      Shinyanga 4982 8.387205
## 3          Mbeya 4639 7.809764
## 4    Kilimanjaro 4379 7.372054
## 5       Morogoro 4006 6.744108
## 6         Arusha 3350 5.639731
## 7         Kagera 3316 5.582492
## 8         Mwanza 3102 5.222222
## 9         Kigoma 2816 4.740741
## 10        Ruvuma 2640 4.444444
## 11         Pwani 2635 4.436027
## 12         Tanga 2547 4.287879
## 13        Dodoma 2201 3.705387
## 14       Singida 2093 3.523569
## 15          Mara 1969 3.314815
## 16        Tabora 1959 3.297980
## 17         Rukwa 1808 3.043771
## 18        Mtwara 1730 2.912458
## 19       Manyara 1583 2.664983
## 20         Lindi 1546 2.602694
## 21 Dar es Salaam  805 1.355219
```

```
q_reg_code <- count(water, region_code) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
q_reg_code
```

```
##    region_code    n        perc
## 1           11 5300 8.922558923
## 2           17 5011 8.436026936
## 3           12 4639 7.809764310
## 4            3 4379 7.372053872
## 5            5 4040 6.801346801
## 6           18 3324 5.595959596
## 7           19 3047 5.129629630
## 8            2 3024 5.090909091
## 9           16 2816 4.740740741
## 10          10 2640 4.444444444
## 11           4 2513 4.230639731
## 12           1 2201 3.705387205
## 13          13 2093 3.523569024
## 14          14 1979 3.331649832
## 15          20 1969 3.314814815
## 16          15 1808 3.043771044
## 17           6 1609 2.708754209
## 18          21 1583 2.664983165
## 19          80 1238 2.084175084
## 20          60 1025 1.725589226
```

```
## 21          90  917 1.543771044
## 22           7  805 1.355218855
## 23          99  423 0.712121212
## 24           9  390 0.656565657
## 25          24  326 0.548821549
## 26           8  300 0.505050505
## 27          40    1 0.001683502
```

```
q_district_code <- count(water, district_code) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(perc))
q_district_code
```

```
##    district_code     n        perc
## 1              1 12203 20.54377104
## 2              2 11173 18.80976431
## 3              3  9998 16.83164983
## 4              4  8999 15.14983165
## 5              5  4356  7.33333333
## 6              6  4074  6.85858586
## 7              7  3343  5.62794613
## 8              8  1043  1.75589226
## 9             30   995  1.67508418
## 10            33   874  1.47138047
## 11            53   745  1.25420875
## 12            43   505  0.85016835
## 13            13   391  0.65824916
## 14            23   293  0.49326599
## 15            63   195  0.32828283
## 16            62   109  0.18350168
## 17            60    63  0.10606061
## 18             0    23  0.03872054
## 19            80    12  0.02020202
## 20            67     6  0.01010101
```

Camps d'informació geogràfica que reduïrem a 2 per evitar la redundància.

lga[15], ward[16]

```
q_lga <- count(water, lga) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_lga,n=5)
```

```
##            lga    n     perc
## 1       Njombe 2503 4.213805
## 2 Arusha Rural 1252 2.107744
## 3  Moshi Rural 1251 2.106061
## 4      Bariadi 1177 1.981481
## 5       Rungwe 1106 1.861953
```

```
q_ward <- count(water, ward) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_ward,n=5)
```

```
##        ward   n      perc
## 1     Igosi 307 0.5168350
## 2  Imalinyi 252 0.4242424
## 3 Siha Kati 232 0.3905724
## 4    Mdandu 231 0.3888889
## 5   Nduruma 217 0.3653199
```

Camps descartats, informació que aporten tan sols és geogràfica

population[17]

```
q_population <- count(water, population) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_population,n=5)
```

```
##   population     n      perc
## 1          0 21381 35.994949
## 2          1  7025 11.826599
## 3        200  1940  3.265993
## 4        150  1892  3.185185
## 5        250  1681  2.829966
```

veiem que aproximadament el 36% dels registres son 0

public meeting[18]

```
q_public <- count(water,public_meeting) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
q_public
```

```
##   public_meeting     n      perc
## 1           True 51011 85.877104
## 2          False  5055  8.510101
## 3           <NA>  3334  5.612795
```

camp descartat.

recorded_by[19]

```
q_record <- count(water,recorded_by) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
q_record
```

```
##               recorded_by     n perc
## 1 GeoData Consultants Ltd 59400  100
```

No aporta informació diferent entre registres.

scheme_management[20], scheme_name[21]

```
q_sch_manag <- count(water,scheme_management) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(perc))
q_sch_manag
```

```
##    scheme_management     n         perc
## 1                VWC 36793 61.941077441
## 2                WUG  5206  8.764309764
## 3               <NA>  3877  6.526936027
## 4    Water authority  3153  5.308080808
## 5                WUA  2883  4.853535354
## 6        Water Board  2748  4.626262626
## 7         Parastatal  1680  2.828282828
## 8   Private operator  1063  1.789562290
## 9            Company  1061  1.786195286
## 10             Other   766  1.289562290
## 11               SWC    97  0.163299663
## 12             Trust    72  0.121212121
## 13              None     1  0.001683502
```

```
q_sch_name <- count(water,scheme_name) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_sch_name,n=5)
```

```
##     scheme_name     n       perc
## 1          <NA> 28166 47.4175084
## 2             K   682  1.1481481
## 3          None   644  1.0841751
## 4      Borehole   546  0.9191919
## 5 Chalinze wate   405  0.6818182
```

Camps descartats, més endevant veurem que la informació que aporten és igual a 'management'

permit[22]

```
q_permit <- count(water,permit) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
q_permit
```

```
##   permit     n      perc
## 1   True 38852 65.407407
## 2  False 17492 29.447811
## 3   <NA>  3056  5.144781
```

Camp descartat

construction_year[23]

```
q_const <- count(water,construction_year) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_const,n=5)
```

```
##   construction_year     n      perc
## 1                 0 20709 34.863636
## 2              2010  2645  4.452862
## 3              2008  2613  4.398990
## 4              2009  2533  4.264310
## 5              2000  2091  3.520202
```

el 35% dels registres son 0. Aquests 0 no tenen sentit per les característiques del camp i són equivalents a un NA. Com que el nostre dataset és gran, eliminarem les rows que tinguin 0 en aquest camp. A més, més endavant aquest camp ens

veiem que hi ha parelles de camps que donen informació similar o redundant per tant, ara analitzarem a veure si podem reduir en alguns camps, també en calcularem el seu percentatge en la mostra

extraction_type[24], extraction_type_group[25], extraction_type_class[26]

```
# Valors unics
apply(water[,c(24,25,26)],2, function(x) length(unique(x)))
```

```
##       extraction_type extraction_type_group extraction_type_class
##                    18                   13                    7
```

```
q_extraction <- count(water,extraction_type, extraction_type_group,
                      extraction_type_class) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(perc))
head(q_extraction,n=5)
```

```
##   extraction_type extraction_type_group extraction_type_class     n      perc
## 1         gravity               gravity               gravity 26780 45.084175
## 2     nira/tanira           nira/tanira              handpump  8154 13.727273
## 3           other                 other                 other  6430 10.824916
## 4     submersible           submersible           submersible  4764  8.020202
## 5          swn 80                swn 80              handpump  3670  6.178451
```

```
q_extraction_v1 <- count(water,extraction_type_class,extraction_type_group,
                         extraction_type) %>% mutate(perc = n/sum(n)*100) %>% arrange(desc(extraction_ty
head(q_extraction_v1,n=5)
```

```
##   extraction_type_class extraction_type_group   extraction_type    n       perc
## 1          wind-powered          wind-powered          windmill  117  0.1969697
## 2           submersible           submersible               ksb 1415  2.3821549
## 3           submersible           submersible       submersible 4764  8.0202020
## 4             rope pump             rope pump other - rope pump  451  0.7592593
## 5                 other                 other             other 6430 10.8249158
```

```
q_extraction_v2 <- count(water, extraction_type_group,
                         extraction_type_class) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(perc))
head(q_extraction_v2,n=5)
```

```
##   extraction_type_group extraction_type_class     n      perc
## 1               gravity               gravity 26780 45.084175
## 2          nira/tanira              handpump  8154 13.727273
## 3                other                 other  6430 10.824916
## 4          submersible           submersible  6179 10.402357
## 5               swn 80              handpump  3670  6.178451
```

```
q_extraction_v3 <- count(water, extraction_type_class, extraction_type_group) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(perc))
head(q_extraction_v3,n=5)
```

```
##   extraction_type_class extraction_type_group     n      perc
## 1               gravity               gravity 26780 45.084175
## 2              handpump          nira/tanira  8154 13.727273
## 3                 other                 other  6430 10.824916
## 4          submersible           submersible  6179 10.402357
## 5              handpump               swn 80  3670  6.178451
```

ens quedarem amb el camp **extraction_type_group**

management[27], management_group[28]

```
# Valors unics
apply(water[,c(27,28)],2, function(x) length(unique(x)))
```

```
##       management management_group
##               12                5
```

```
q_management <- count(water,management, management_group) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(perc))
q_management
```

```
##          management management_group     n       perc
## 1               vwc       user-group 40507 68.1936027
## 2               wug       user-group  6515 10.9680135
## 3       water board       user-group  2933  4.9377104
## 4               wua       user-group  2535  4.2676768
## 5  private operator       commercial  1971  3.3181818
## 6        parastatal       parastatal  1768  2.9764310
## 7   water authority       commercial   904  1.5218855
## 8             other            other   844  1.4208754
## 9           company       commercial   685  1.1531987
## 10          unknown          unknown   561  0.9444444
## 11   other - school            other    99  0.1666667
## 12            trust       commercial    78  0.1313131
```

```
q_management_v1 <- count(water,management_group, management) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(management_group))
q_management_v1
```

```
##    management_group       management     n       perc
## 1        user-group              vwc 40507 68.1936027
## 2        user-group      water board  2933  4.9377104
## 3        user-group              wua  2535  4.2676768
## 4        user-group              wug  6515 10.9680135
## 5           unknown          unknown   561  0.9444444
## 6        parastatal       parastatal  1768  2.9764310
## 7             other            other   844  1.4208754
## 8             other   other - school    99  0.1666667
## 9        commercial          company   685  1.1531987
## 10       commercial private operator  1971  3.3181818
## 11       commercial            trust    78  0.1313131
## 12       commercial  water authority   904  1.5218855
```

```
q_management_v2 <- count(water,management_group, management, scheme_management,
                         scheme_name) %>% mutate(perc = n/sum(n)*100) %>%
  arrange(desc(management_group))
head(q_management_v2,n=5)
```

```
##   management_group management scheme_management                scheme_name  n
## 1       user-group        vwc           Company              Bagamoyo wate 37
## 2       user-group        vwc           Company              Bagamoyo Wate  3
## 3       user-group        vwc           Company                   Borehole  1
## 4       user-group        vwc           Company Borehole drilling project  1
## 5       user-group        vwc           Company                   Ikela Wa 89
##          perc
## 1 0.062289562
## 2 0.005050505
## 3 0.001683502
## 4 0.001683502
## 5 0.149831650
```

ens quedarem amb el camp **management** conté més informació

payment[29], payment_type[30]

```
# Valors unics
apply(water[,c(29,30)],2, function(x) length(unique(x)))
```

```
##      payment payment_type
##            7            7
```

```
q_payment <- count(water,payment, payment_type) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(perc))
q_payment
```

```
##                 payment payment_type     n      perc
## 1             never pay    never pay 25348 42.673401
## 2        pay per bucket   per bucket  8985 15.126263
## 3           pay monthly      monthly  8300 13.973064
## 4               unknown      unknown  8157 13.732323
## 5 pay when scheme fails   on failure  3914  6.589226
## 6           pay annually     annually  3642  6.131313
## 7                 other        other  1054  1.774411
```

veiem que contenen els mateixos registres, es indiferent el camp escollit, en aquest cas seleccionem **payment**

water_quality[31], quality_group[32]

```
# Valors unics
apply(water[,c(31,32)],2, function(x) length(unique(x)))
```

```
## water_quality quality_group
##             8             6
```

```
q_quality <- count(water,water_quality, quality_group) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(perc))
q_quality
```

```
##        water_quality quality_group     n        perc
## 1               soft          good 50818 85.55218855
## 2              salty         salty  4856  8.17508418
## 3            unknown       unknown  1876  3.15824916
## 4              milky         milky   804  1.35353535
## 5           coloured       colored   490  0.82491582
## 6    salty abandoned         salty   339  0.57070707
## 7           fluoride      fluoride   200  0.33670034
## 8 fluoride abandoned      fluoride    17  0.02861953
```

seleccionem: **quality_group**

water_quality[33], quality_group[34]

```
# Valors unics
apply(water[,c(33,34)],2, function(x) length(unique(x)))
```

```
##       quantity quantity_group
##              5              5
```

```
q_quantity <- count(water,quantity, quantity_group) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(perc))
q_quantity
```

```
##       quantity quantity_group     n      perc
## 1       enough         enough 33186 55.868687
## 2 insufficient   insufficient 15129 25.469697
## 3          dry            dry  6246 10.515152
## 4     seasonal       seasonal  4050  6.818182
## 5      unknown        unknown   789  1.328283
```

veiem que contenen els mateixos registres, es indiferent el camp escollit, en aquest cas seleccionem **quantity_group**

source[35], source_type[36], source_class[37]

```
# Valors unics
apply(water[,c(35,36,37)],2, function(x) length(unique(x)))
```

```
##       source  source_type source_class
##           10            7            3
```

```
q_source <- count(water,source, source_type, source_class) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(perc))
q_source
```

```
##                  source          source_type source_class     n       perc
## 1                spring               spring  groundwater 17021 28.6548822
## 2          shallow well         shallow well  groundwater 16824 28.3232323
## 3           machine dbh             borehole  groundwater 11075 18.6447811
## 4                 river           river/lake      surface  9612 16.1818182
## 5  rainwater harvesting rainwater harvesting      surface  2295  3.8636364
## 6              hand dtw             borehole  groundwater   874  1.4713805
## 7                  lake           river/lake      surface   765  1.2878788
## 8                   dam                  dam      surface   656  1.1043771
## 9                 other                other      unknown   212  0.3569024
## 10              unknown                other      unknown    66  0.1111111
```

```
q_source_v1 <- count(water,source_class, source_type, source) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(source_class))
q_source_v1
```

```
##    source_class          source_type               source     n       perc
## 1       unknown                other                other   212  0.3569024
## 2       unknown                other              unknown    66  0.1111111
## 3       surface                  dam                  dam   656  1.1043771
## 4       surface rainwater harvesting rainwater harvesting  2295  3.8636364
## 5       surface           river/lake                lake   765  1.2878788
## 6       surface           river/lake               river  9612 16.1818182
## 7   groundwater             borehole             hand dtw   874  1.4713805
## 8   groundwater             borehole          machine dbh 11075 18.6447811
## 9   groundwater         shallow well         shallow well 16824 28.3232323
## 10  groundwater               spring               spring 17021 28.6548822
```

en aquest cas seleccionem **source**

waterpoint_type[38], waterpoint_type_group[39]

```
# Valors unics
apply(water[,c(38,39)],2, function(x) length(unique(x)))
```

```
##       waterpoint_type waterpoint_type_group
##                     7                     6
```

```
q_waterpoint <- count(water, waterpoint_type, waterpoint_type_group) %>%
  mutate(perc = n/sum(n)*100) %>% arrange(desc(perc))
q_waterpoint
```

```
##               waterpoint_type waterpoint_type_group     n        perc
## 1          communal standpipe    communal standpipe 28522 48.01683502
## 2                   hand pump             hand pump 17488 29.44107744
## 3                       other                 other  6380 10.74074074
## 4 communal standpipe multiple    communal standpipe  6103 10.27441077
## 5             improved spring       improved spring   784  1.31986532
## 6               cattle trough         cattle trough   116  0.19528620
## 7                         dam                   dam     7  0.01178451
```

seleccionem **waterpoint_type**

aixi doncs els camps seleccionats seran:

```
water_select <- select(water, status_group, amount_tsh,
                   region_code, district_code
                   , population, construction_year, extraction_type_group,
                   management, payment, quality_group, quantity_group, source,
                   waterpoint_type)

water_net <- subset(water_select, construction_year >0 )

write.csv(water_net,"data\\trainig_set_CLEAN.csv", row.names = FALSE)

head(water_net, 10)
```

```
##         status_group amount_tsh region_code district_code population
## 69572     functional       6000          11             5        109
## 8776      functional          0          20             2        280
## 34310     functional         25          21             4        250
## 67743 non functional          0          90            63         58
## 9944      functional         20           4             8          1
## 49056     functional          0          60            43        345
## 50409     functional        200          10             5        250
## 50495     functional          0           3             7          1
## 61848     functional          0          15             2        200
## 48451 non functional        500          11             4         35
##       construction_year extraction_type_group       management
## 69572              1999               gravity              vwc
## 8776               2010               gravity              wug
```

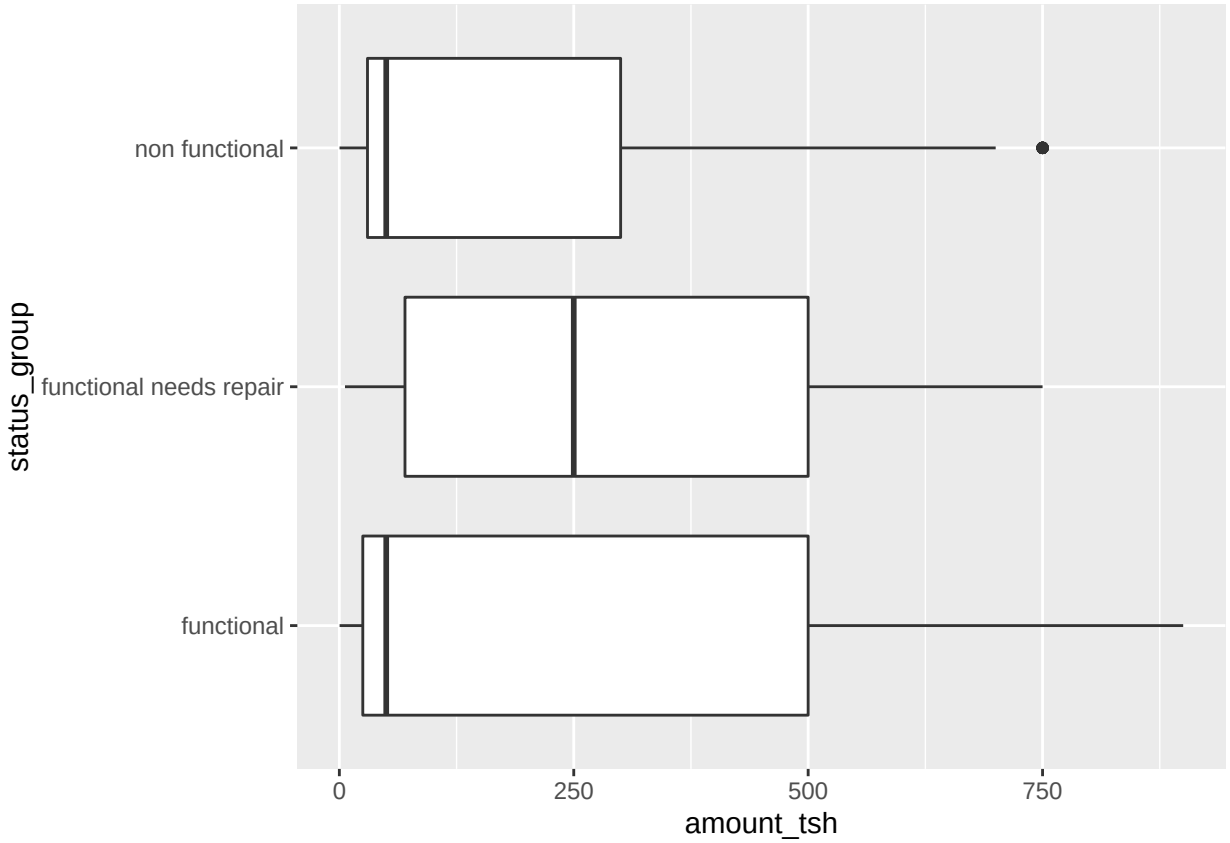
```
## 34310              2009              gravity              vwc
## 67743              1986          submersible              vwc
## 9944               2009          submersible              vwc
## 49056              2011          submersible private operator
## 50409              1987               swn 80              wug
## 50495              2009              gravity      water board
## 61848              1991               swn 80              vwc
## 48451              1978              gravity              wua
##                    payment quality_group quantity_group               source
## 69572          pay annually          good         enough               spring
## 8776              never pay          good   insufficient rainwater harvesting
## 34310        pay per bucket          good         enough                  dam
## 67743             never pay          good            dry          machine dbh
## 9944         pay per bucket         salty         enough                other
## 49056             never pay         salty         enough          machine dbh
## 50409 pay when scheme fails          good   insufficient         shallow well
## 50495           pay monthly          good         enough               spring
## 61848             never pay          good         enough          machine dbh
## 48451           pay monthly          good            dry                river
##                   waterpoint_type
## 69572          communal standpipe
## 8776           communal standpipe
## 34310 communal standpipe multiple
## 67743 communal standpipe multiple
## 9944  communal standpipe multiple
## 49056                      other
## 50409                  hand pump
## 50495          communal standpipe
## 61848                  hand pump
## 48451          communal standpipe
```

3.2. Identificació i tractament de valors extrems. + 4.1. Selecció dels grups de dades que es volen analitzar/comparar (planificació dels anàlisis a aplicar). + 4.1. Selecció dels grups de dades que es volen analitzar/comparar (planificació dels anàlisis a aplicar).

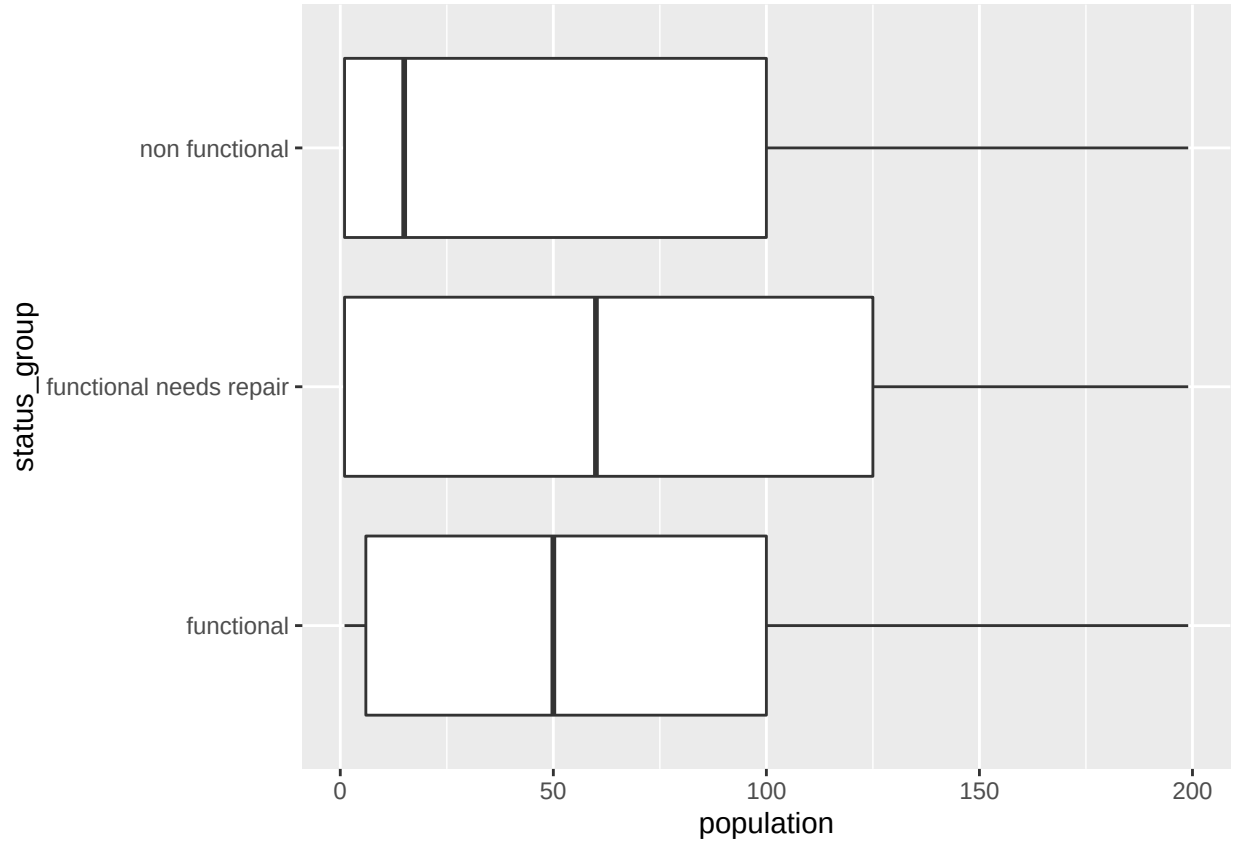
Variables quantitatives Continuarem l'anàlisi graficant els camps. L'objectiu és detectar valors extrems i conèixer les distribucions de les nostres dades respecte del status de les bombes.

```
t <- water_net

sub1 <- subset(t, amount_tsh > 0  & amount_tsh < 1000)
ggplot(sub1, aes(x=amount_tsh, y=status_group)) + geom_boxplot()
```

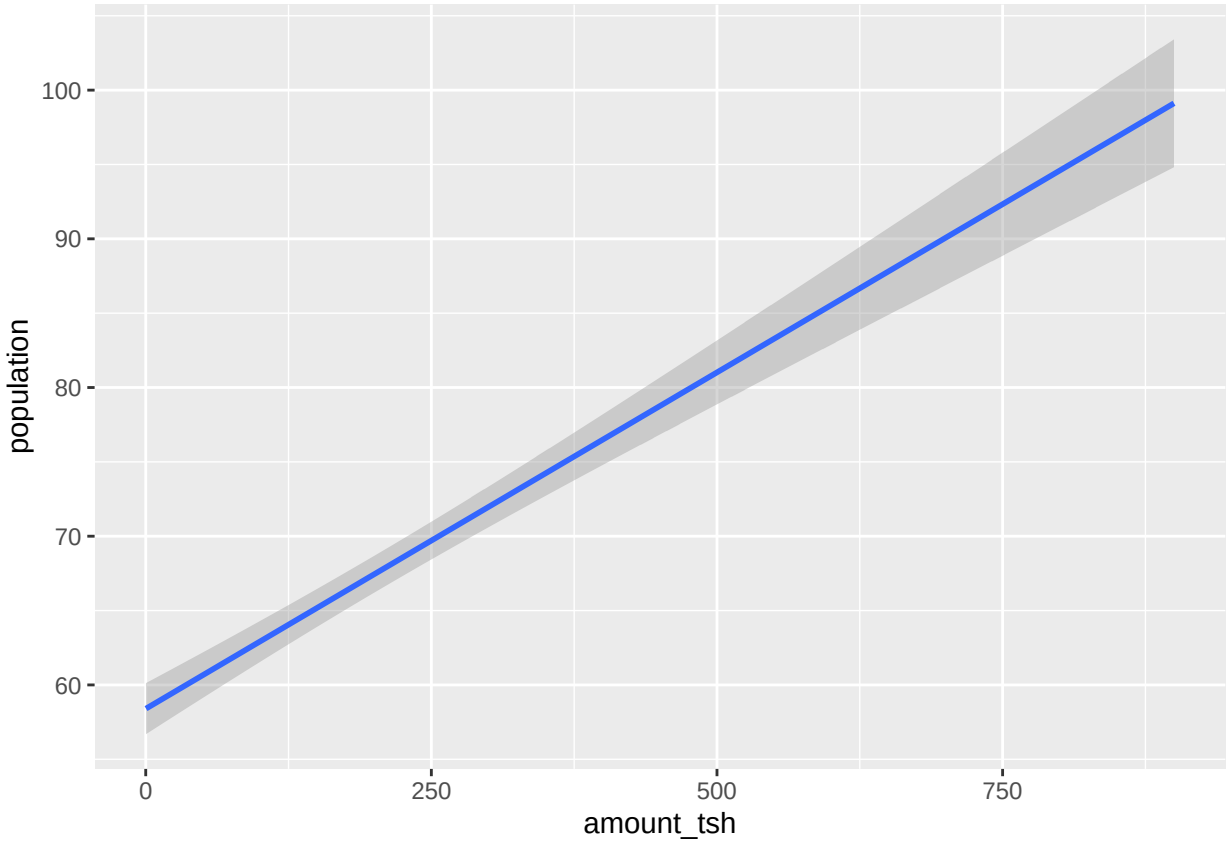



```
sub2 <- subset(t, population >0 & population < 200)
ggplot(sub2, aes(x=population, y=status_group)) + geom_boxplot()
```

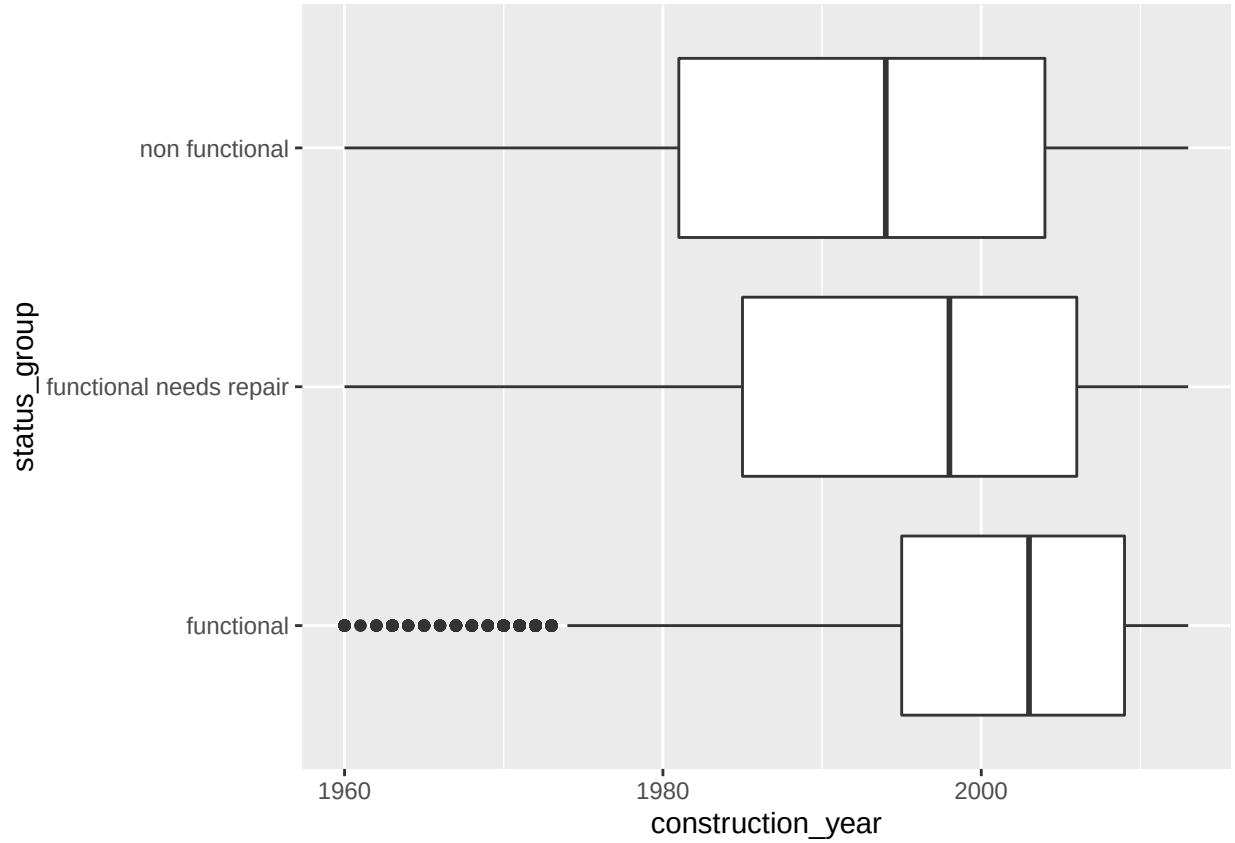



```
sub3 <- subset(t, population >0 & population < 200 & amount_tsh > 0  &
                 amount_tsh < 1000)
ggplot(sub3,aes(amount_tsh, population)) + geom_smooth(method='lm')
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
sub4 <- subset(t, construction_year > 0)
ggplot(sub4, aes(x=construction_year, y=status_group)) + geom_boxplot()
```




```
sub1 <- subset(t, amount_tsh == 0)
table(sub1$status_group)
```

```
##
##              functional functional needs repair          non functional
##                    9426                    1271                   10640
```

```
sub2 <- subset(t, amount_tsh > 0  & amount_tsh < 1000)
table(sub2$status_group)
```

```
##
##              functional functional needs repair          non functional
##                    9051                    1003                    3069
```

```
sub3 <- subset(t, amount_tsh > 1000)
table(sub3$status_group)
```

```
##
##              functional functional needs repair          non functional
##                    2178                     170                     484
```

```
sub2 <- subset(t, population ==0)
table(sub2$status_group)
```

```
##
##              functional functional needs repair          non functional
##                      995                       29                     323
```

```
sub2 <- subset(t, population >0 & population < 200)
table(sub2$status_group)
```

```
##
##              functional functional needs repair          non functional
##                    11721                     1208                    7560
```

```
sub2 <- subset(t, population > 200)
table(sub2$status_group)
```

```
##
##              functional functional needs repair          non functional
##                     7858                     1196                    5891
```

Podem Veure que aquests dos camps presenten un comportament similar marcat a trams, és més, en el tram central semblen presentar correlació. Observem dues coses:

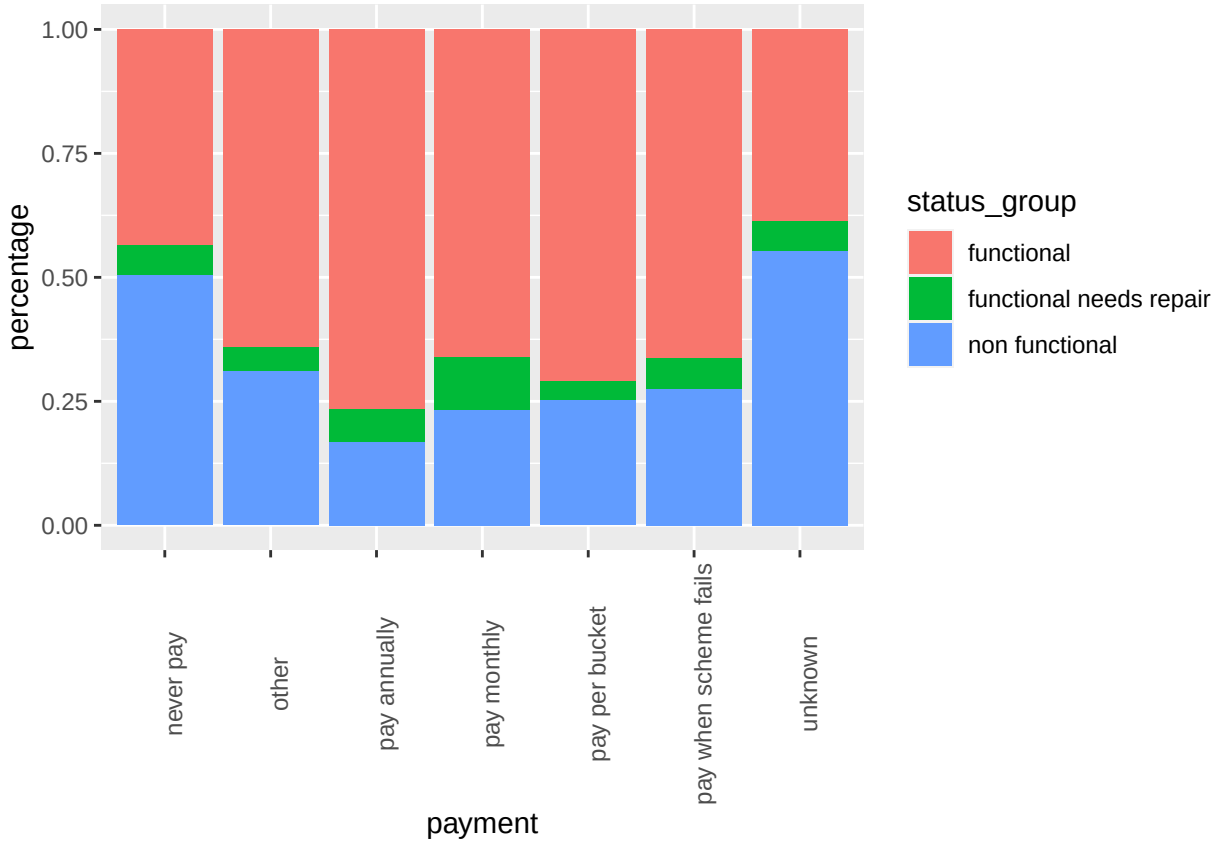
- El camp amount_ths te molt valors 0 o porpers a 0. Al tractar-se d'una mesura de cabal podria ser que aquelles bombes estiguessin desactivades en el moment de presa del registre. De moment tractarem aquesta dada com a vàlida però tindrem això en compte per anàlisis posteriors.
- Té sentit que la correlació que presenten els camps no es mantingui pels valors extrems del bigoti superior dels valors de població, doncs és d'esperar que cada bomba presenti un limit de cabal i al incrementar la població el que succeeixi és que incrementi el nombre de bombes.

De moment aparquem l'anàlisi aquí i el redrendrem al punt 4.3.

Variables qualitatives Pel que fa a les variables qualitatives veiem el següent:

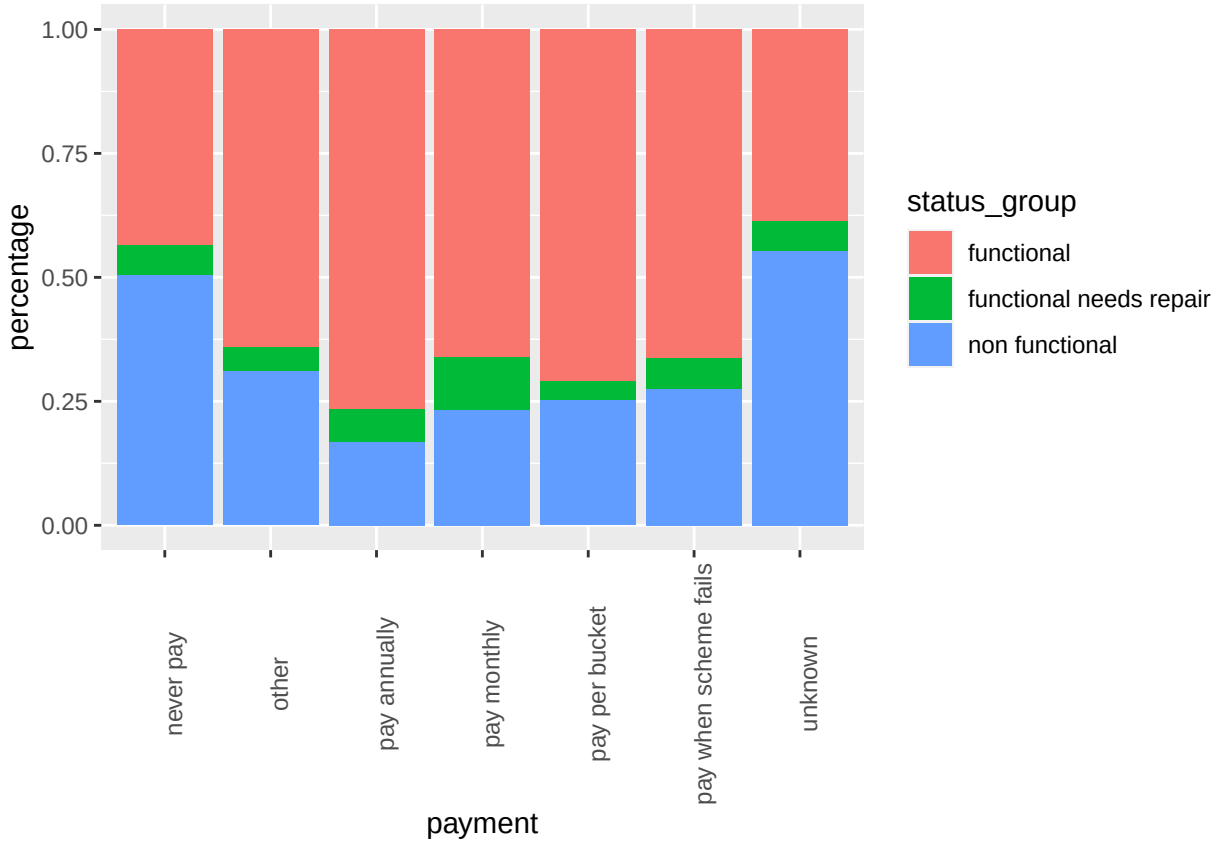
```
t %>%
    group_by(payment, status_group) %>%
    summarise(percentage = n()) %>%
    mutate(percentage = percentage / sum(percentage)) %>%
    ggplot(aes(x = payment, y = percentage, fill = status_group)) +
    geom_bar(stat = "identity", position = "stack")+ theme(axis.text.x = element_text(angle = 90))
```

```
## `summarise()` regrouping output by 'payment' (override with `.groups` argument)
```



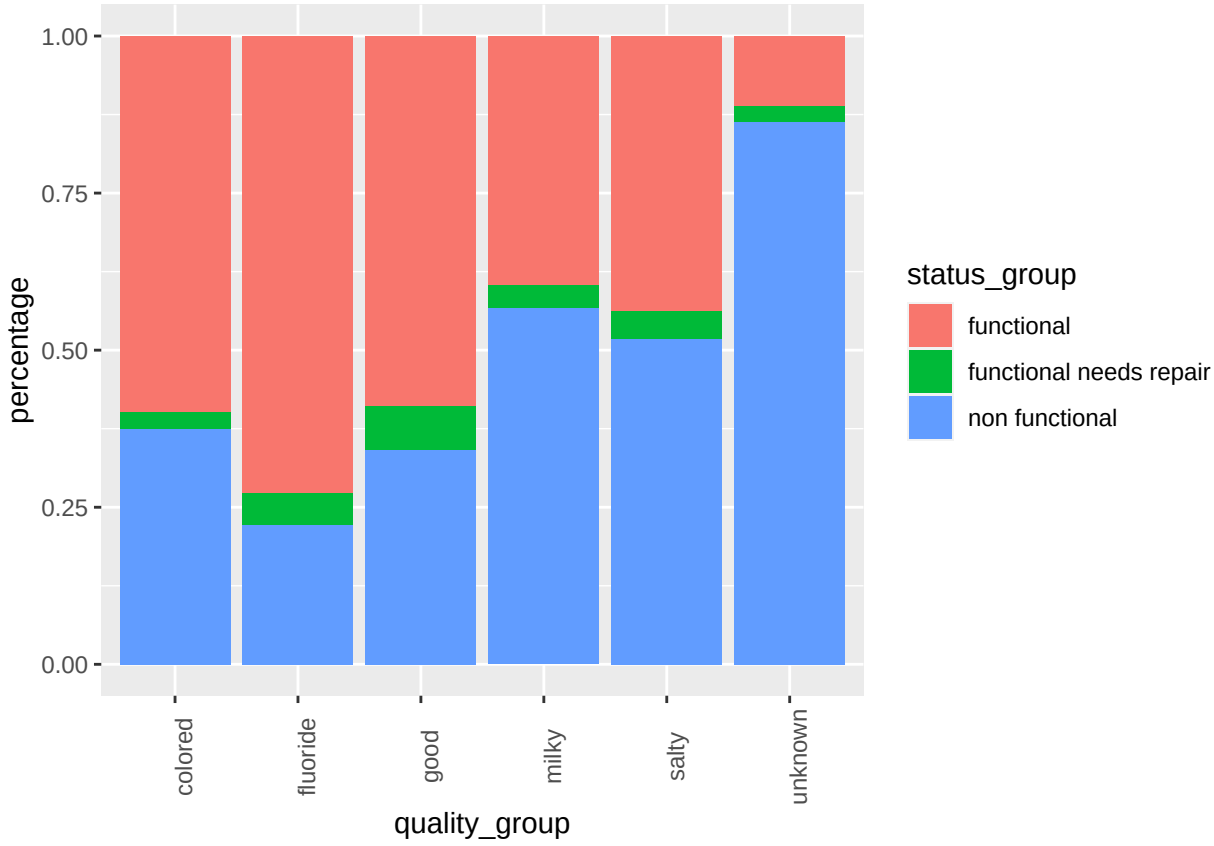

```
t %>%
    group_by(payment, status_group) %>%
    summarise(percentage = n()) %>%
    mutate(percentage = percentage / sum(percentage)) %>%
    ggplot(aes(x = payment, y = percentage, fill = status_group)) +
    geom_bar(stat = "identity", position = "stack")+ theme(axis.text.x = element_text(angle = 90))
```

```
## `summarise()` regrouping output by 'payment' (override with `.groups` argument)
```



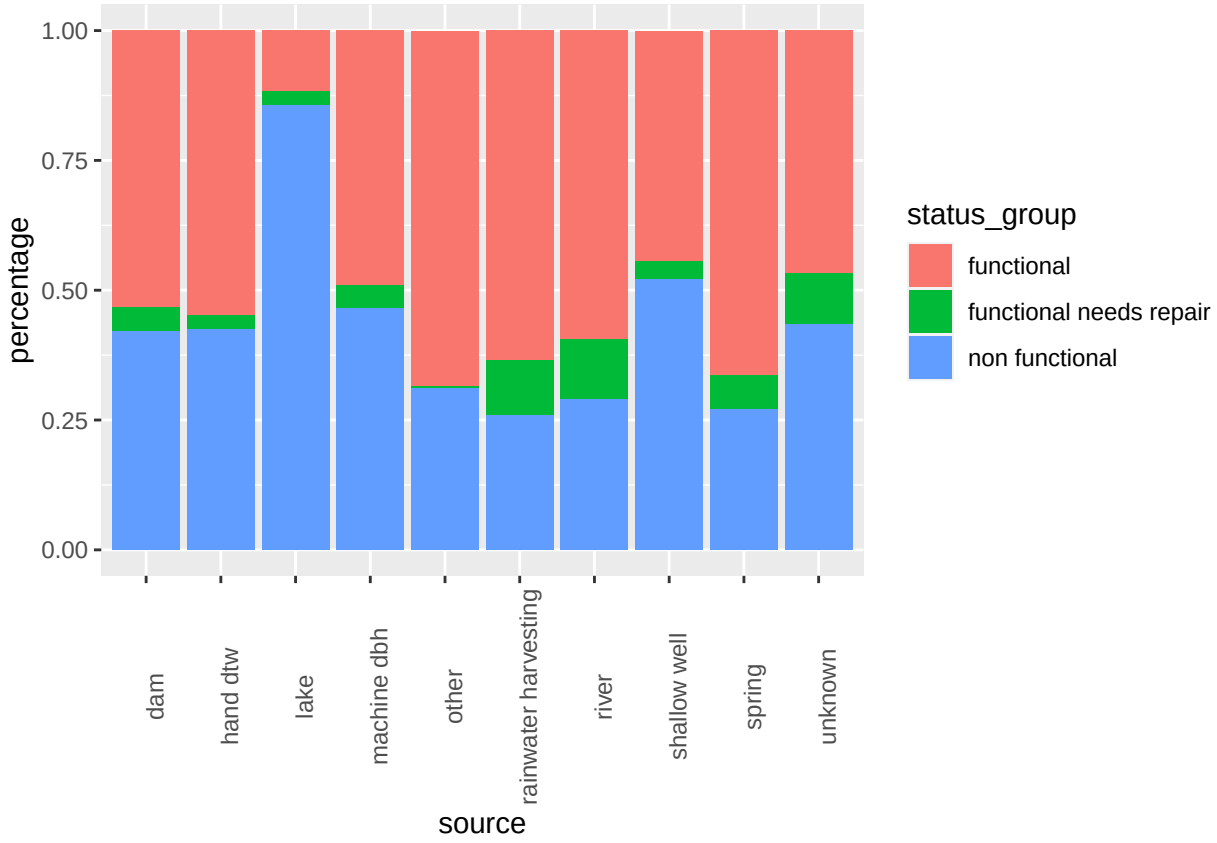

```
t %>%
    group_by(quality_group, status_group) %>%
    summarise(percentage = n()) %>%
    mutate(percentage = percentage / sum(percentage)) %>%
    ggplot(aes(x = quality_group, y = percentage, fill = status_group)) +
    geom_bar(stat = "identity", position = "stack") + theme(axis.text.x = element_text(angle = 90))
```

```
## `summarise()` regrouping output by 'quality_group' (override with `.groups` argument)
```



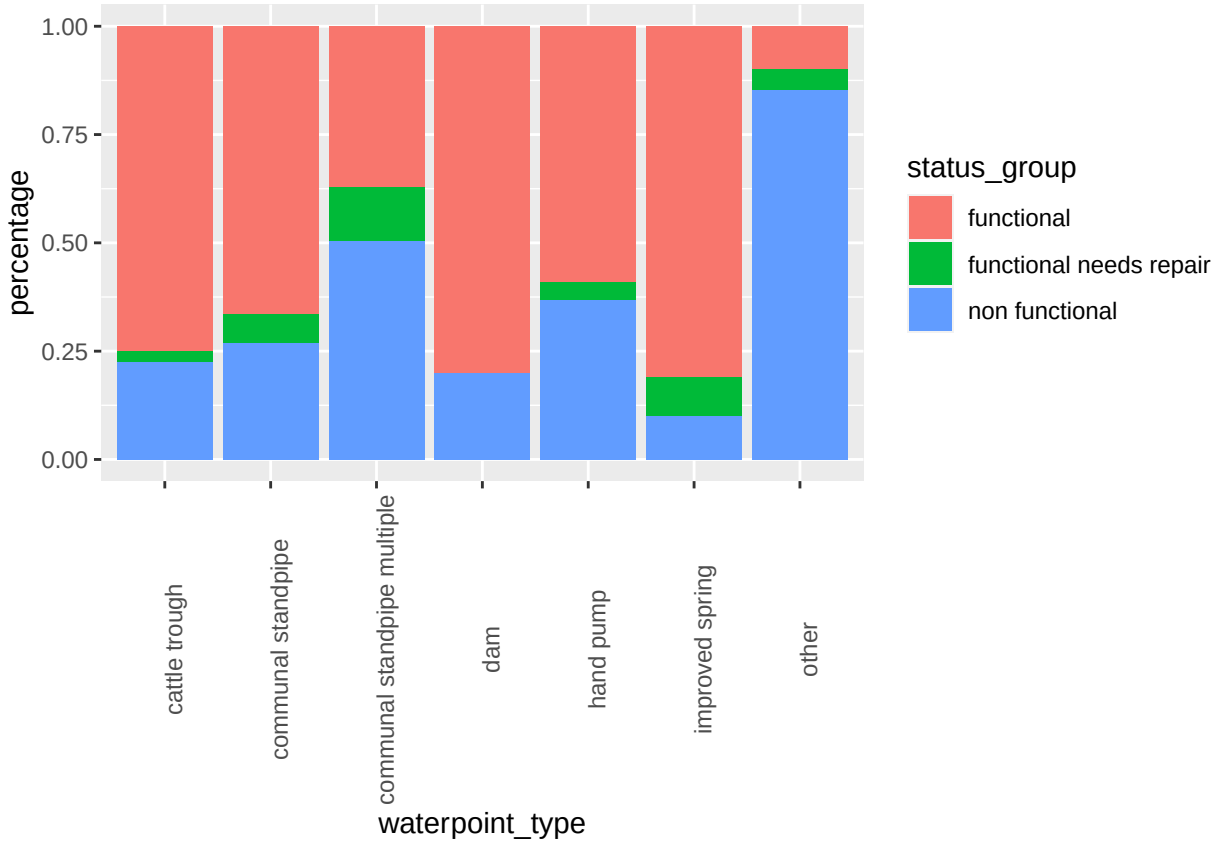

```
t %>%
    group_by(source, status_group) %>%
    summarise(percentage = n()) %>%
    mutate(percentage = percentage / sum(percentage)) %>%
    ggplot(aes(x = source, y = percentage, fill = status_group)) +
    geom_bar(stat = "identity", position = "stack")+ theme(axis.text.x = element_text(angle = 90))
```

```
## `summarise()` regrouping output by 'source' (override with `.groups` argument)
```



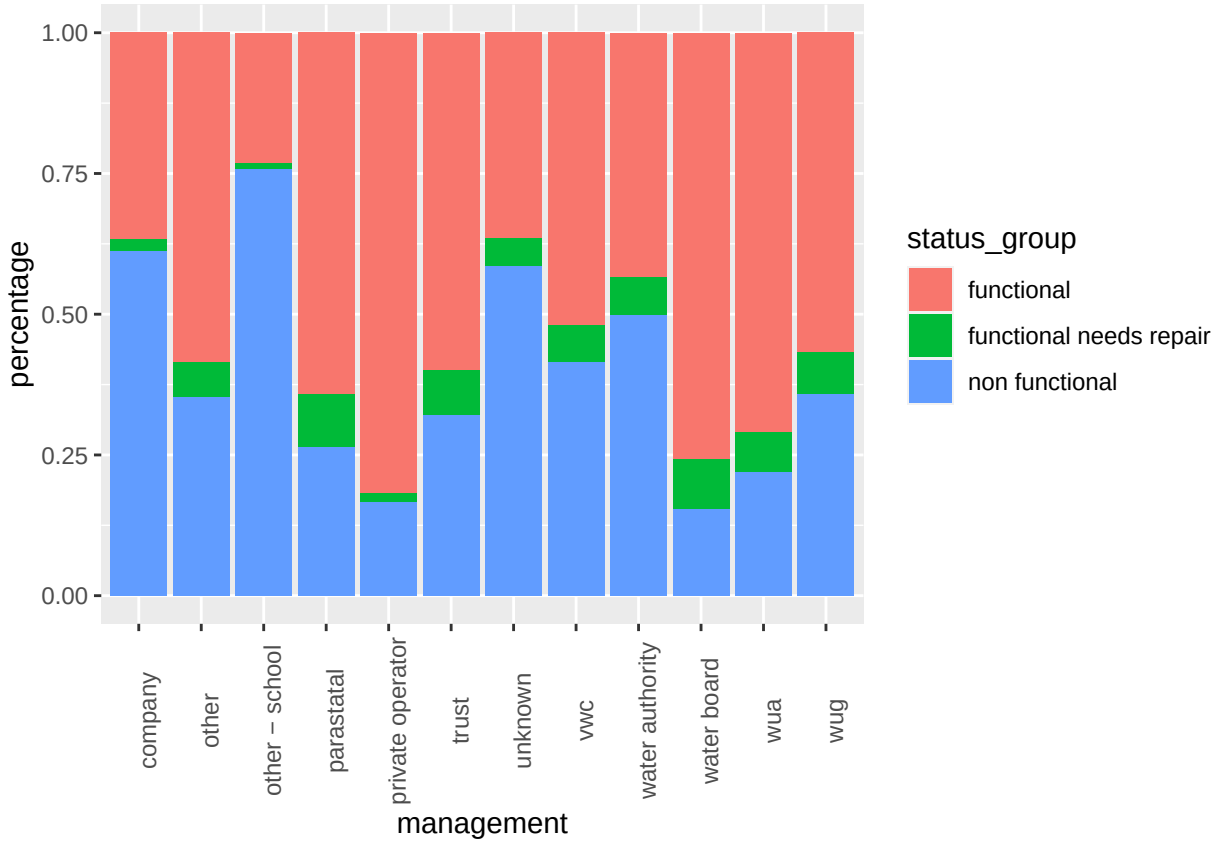

```
t %>%
    group_by(waterpoint_type, status_group) %>%
    summarise(percentage = n()) %>%
    mutate(percentage = percentage / sum(percentage)) %>%
    ggplot(aes(x = waterpoint_type, y = percentage, fill = status_group)) +
    geom_bar(stat = "identity", position = "stack")+ theme(axis.text.x = element_text(angle = 90))
```

```
## `summarise()` regrouping output by 'waterpoint_type' (override with `.groups` argument)
```




```
t %>%
    group_by(management, status_group) %>%
    summarise(percentage = n()) %>%
    mutate(percentage = percentage / sum(percentage)) %>%
    ggplot(aes(x = management, y = percentage, fill = status_group)) +
    geom_bar(stat = "identity", position = "stack")+ theme(axis.text.x = element_text(angle = 90))
```

```
## `summarise()` regrouping output by 'management' (override with `.groups` argument)
```




```
#install.packages("DescTools")
library(car)
library(DescTools)
```

4.2 Comprovació de la normalitat i homogeneïtat de la variància

comprovació de la normalitat Mètode Kolmogorov-Smirnov

```
ks.test(water_net$amount_tsh, pnorm, mean(water_net$amount_tsh),
        sd(water_net$amount_tsh))
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  water_net$amount_tsh
## D = 0.4476, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
ks.test(water_net$population, pnorm, mean(water_net$population),
        sd(water_net$population))
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  water_net$population
## D = 0.31261, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
ks.test(water_net$construction_year, pnorm, mean(water_net$construction_year), sd(water_net$constructio
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  water_net$construction_year
## D = 0.1304, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

segons el metode Kolmogorov-Smirnov, per els camps "amount_tsh", "population", "construction_year", no es compleix es rebutja la hipòtesi nul · la

Mètode Shapiro-Wilk

```
shapiro.test(water_net$amount_tsh[0:5000])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  water_net$amount_tsh[0:5000]
## W = 0.10557, p-value < 2.2e-16
```

```
shapiro.test(water_net$population[0:5000])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  water_net$population[0:5000]
## W = 0.48299, p-value < 2.2e-16
```

```
shapiro.test(water_net$construction_year[0:5000])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  water_net$construction_year[0:5000]
## W = 0.9061, p-value < 2.2e-16
```

Segons veiem també que per el metode Shapiro-Wilk, aplicat a als 5000 primer registes deguts a la limitació de la funció, també es rebutja la hipòtesi nul · la.

De totes maneres com que el nombre de registres és molt gran, podem aplicar el teorema central del límit, podem considerar que les dades tendeixen a seguir una distribució normal

El teorema central del límit: a mesura que augmenta la mida de la mostra , la distribució de la mitjana de la mostra s'assembla cada vegada més a una distribució normal amb una (vertadera) mitjana de població i variància .

comprovació de l'homoscedasticitat Mètode Shapiro-Wilk

```
leveneTest(amount_tsh ~ status_group, data = water_net)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##          Df F value    Pr(>F)
## group     2  75.816 < 2.2e-16 ***
##       38688
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
leveneTest(population ~ status_group, data = water_net)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##          Df F value Pr(>F)
## group     2  0.9672 0.3802
##       38688
```

```
leveneTest(construction_year ~ status_group, data = water_net)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##          Df F value    Pr(>F)
## group     2  788.81 < 2.2e-16 ***
##       38688
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Es rebutja la hipotesi nul · la d'homoscedasticitat

```
fligner.test(amount_tsh ~ status_group, data = water_net)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  amount_tsh by status_group
## Fligner-Killeen:med chi-squared = 6190.9, df = 2, p-value < 2.2e-16
```

```
fligner.test(population ~ status_group, data = water_net)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  population by status_group
## Fligner-Killeen:med chi-squared = 251.71, df = 2, p-value < 2.2e-16
```

```
fligner.test(construction_year ~ status_group, data = water_net)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  construction_year by status_group
## Fligner-Killeen:med chi-squared = 1643.5, df = 2, p-value < 2.2e-16
```

```r
res.aov_v1 <- aov(amount_tsh ~ status_group, data = water_net)
res.aov_v2 <- aov(population ~ status_group, data = water_net)
res.aov_v3 <- aov(construction_year ~ status_group, data = water_net)
summary(res.aov_v1)
```

```
##                 Df    Sum Sq   Mean Sq F value Pr(>F)
## status_group     2 1.902e+09 951208871   76.16 <2e-16 ***
## Residuals    38688 4.832e+11  12490411
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(res.aov_v2)
```

```
##                 Df   Sum Sq Mean Sq F value  Pr(>F)
## status_group     2 3.04e+06 1520009   4.983 0.00686 **
## Residuals    38688 1.18e+10  305021
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(res.aov_v3)
```

```
##                 Df  Sum Sq Mean Sq F value Pr(>F)
## status_group     2  500055  250028    1753 <2e-16 ***
## Residuals    38688 5518248     143
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4.3. Aplicació de proves estadístiques per comparar els grups de dades. En funció de les dades i de l'objectiu de l'estudi, aplicar proves de contrast d'hipòtesis, correlacions, regressions, etc. Aplicar almenys tres mètodes d'anàlisi diferents.

Correlació Població & Amount TSH En els apartats anteriors, quan miràvem els valors extrems ens hem trobat que hi havia un parell de variables numèriques que valia la pena estudiar més ja que semblaven presentar un comportament similar en alguns trams: la població i l'amount_tsh. Ambdues variables presenten molts 0 i també uns valors extrems força alts.

Amount tsh és particularment interessant doncs al ser una mena de mesura de cabal té una gran rellevància i no obstant presenta una gran proporció de 0s que fan difícil treballar-hi, doncs desplacen enormement mesures com la mitjana. Si els deixem de banda tenim que:

```r
test <- subset(t, amount_tsh > 0)
quantile(test$amount_tsh)
```

```
##      0%     25%     50%     75%    100%
## 2.0e-01 5.0e+01 2.5e+02 6.0e+02 3.5e+05
```

Pel que fa a la població tenim el següent:

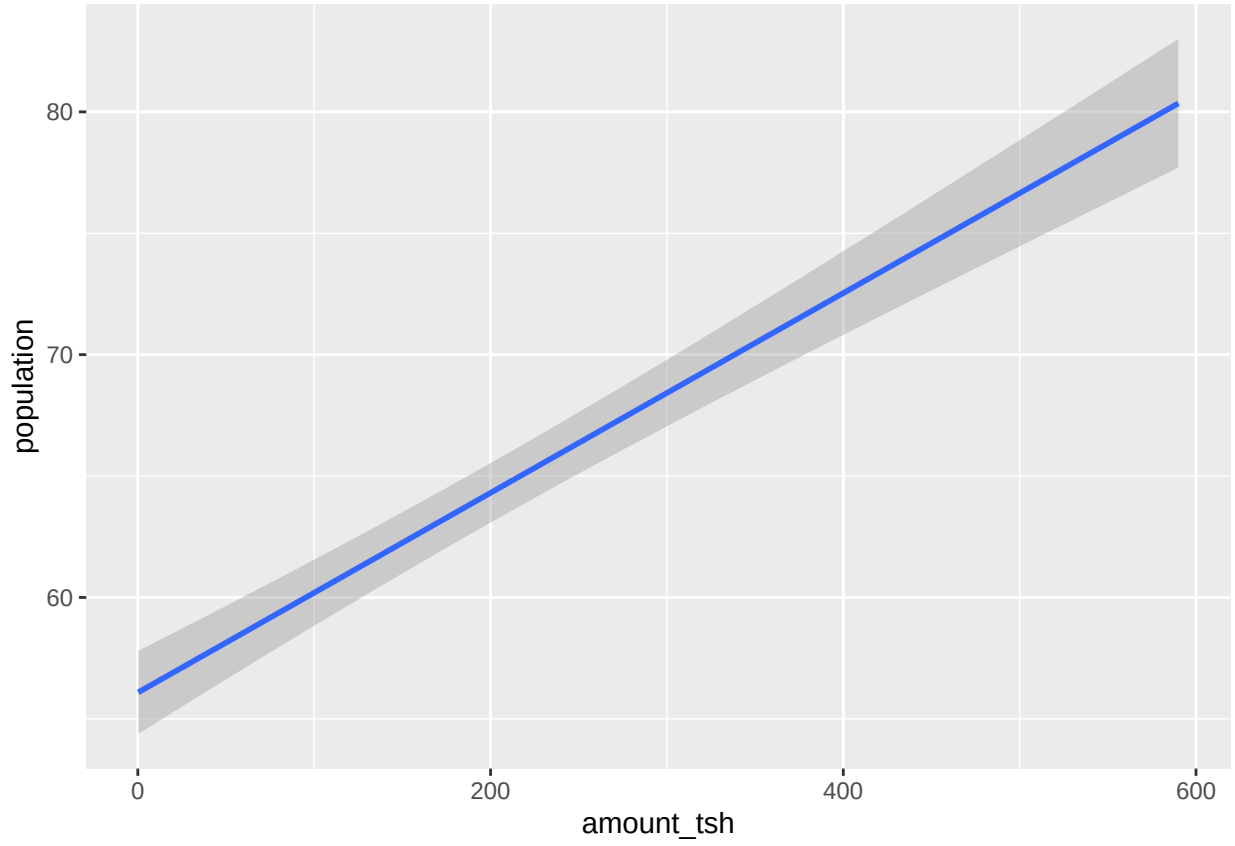
```
quantile(t$population)
```

```
##    0%   25%   50%   75%  100%
##     0    30   150   305 30500
```

Ara podem testar la correlació d'aquestes dues variables. Farem servir el 4t percentil com a referència per partir la mostra doncs el que volem és gràficament contrastar la correlació entre aquestes dues variables als dos trams de dades que es generen quan filtrem per aquest llindar que hem escollit:

```
sub1 <- subset(t, population < 200 & amount_tsh > 0  & amount_tsh < 600)
ggplot(sub1,aes(amount_tsh, population)) + geom_smooth(method='lm')
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
res <- cor.test(sub1$population, sub1$amount_tsh,
                    method = "pearson")
res
```

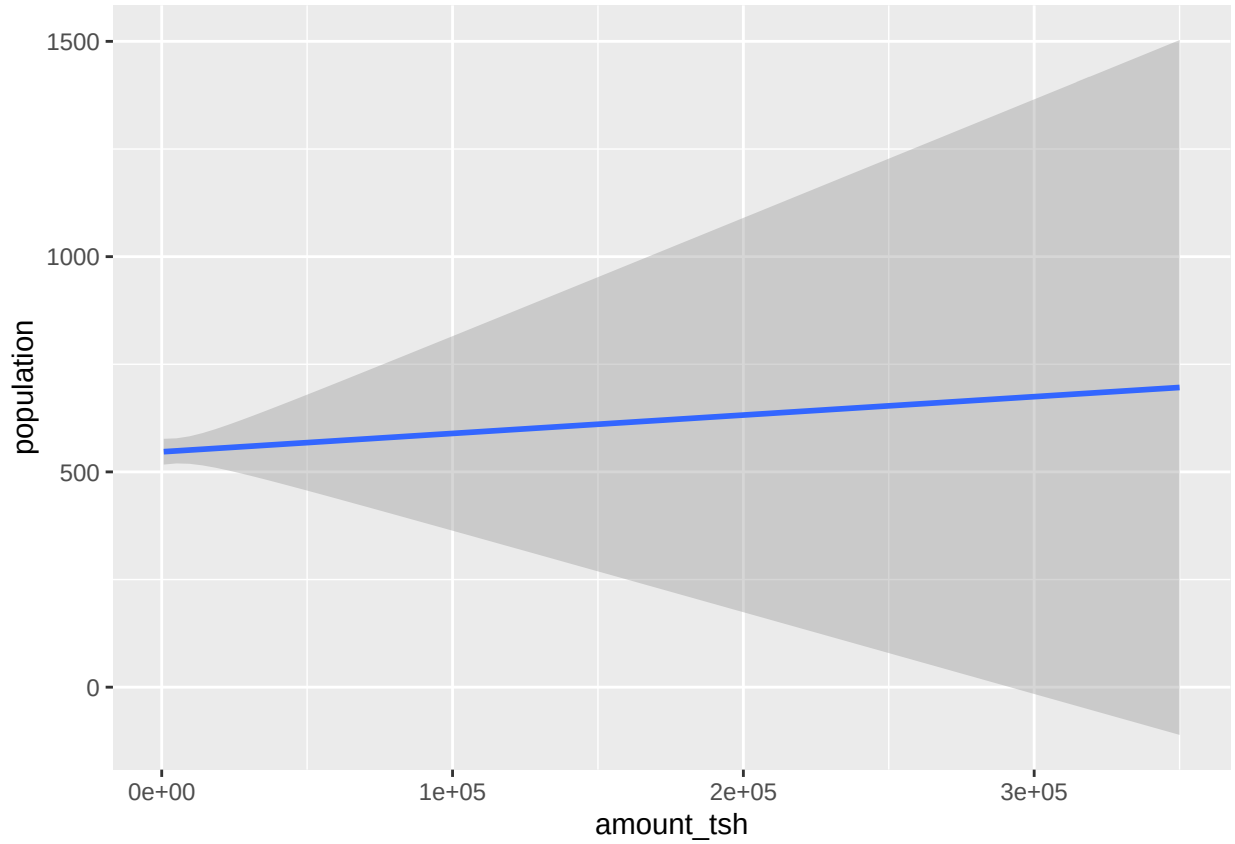
```
##
##  Pearson's product-moment correlation
##
## data:  sub1$population and sub1$amount_tsh
## t = 13.442, df = 7391, p-value < 2.2e-16
```

```
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1321476 0.1766513
## sample estimates:
##       cor
## 0.1544778
```

Veiem al primer tram, on hem establer que: population $<$ 200 & amount_tsh $>$ 0 & amount_tsh $<$ 600 tenim que el p valor per la correlació és molt menor al nivell de significació 0.05 per tant podem rebutjar la hipòtesi nul · la i considerar que les vaiables estan correlacionades.

```
sub2 <- subset(t, population > 200 & amount_tsh > 0  & amount_tsh > 600)
ggplot(sub2,aes(amount_tsh, population)) + geom_smooth(method='lm')
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
res1 <- cor.test(sub2$population, sub2$amount_tsh,
                    method = "pearson")
res1
```

```
##
##  Pearson's product-moment correlation
##
## data:  sub2$population and sub2$amount_tsh
```

```
## t = 0.35951, df = 1528, p-value = 0.7193
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.04093686  0.05928396
## sample estimates:
##         cor
## 0.009196645
```

En canvi, el següent tram de les dades on population > 200 & amount_tsh > 0 & amount_tsh > 600, ja només pel gràfic podem veure com la correlació es perd tal i com confirma un p-valor de 0.72.

Random Forest Ara que sabem que aquests dos trams de dades es comporten diferent, anem a intentar fer un anàlisi predictiu entrenant un model random forest de dues manetres diferents:

- 1 - Tota la mostra sencera
- 2 - La mostra escindida en dos trams utilitzant el filtre que hem establert a l'anàlisi anterior.

```
# sense escindir la mostra

## ens assegurem que status_group esta correctament classificada:
t$status_group <-as.factor(t$status_group)

## dividim la mostra:
h<-holdout(t$status_group,ratio=2/3,mode="stratified")
data_train<-t[h$tr,]
data_test<-t[h$ts,]

## entrenem el model
m <- randomForest(status_group~., data = data_train)

## realitzem la predicció
prediction_se <- predict(m, data_test, type = "class")

## comprobem la bondat de l'ajsut:
confusionMatrix(prediction_se, data_test$status_group)
```

```
## Confusion Matrix and Statistics
##
##                          Reference
## Prediction                functional functional needs repair non functional
##   functional                    6593                     519           1087
##   functional needs repair         84                     196             47
##   non functional                 558                     128           3686
##
## Overall Statistics
##
##                Accuracy : 0.8121
##                  95% CI : (0.8053, 0.8189)
##     No Information Rate : 0.5609
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.6353
```

```
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
## Statistics by Class:
##
##                      Class: functional Class: functional needs repair
## Sensitivity                     0.9113                         0.23250
## Specificity                     0.7164                         0.98913
## Pos Pred Value                  0.8041                         0.59939
## Neg Pred Value                  0.8634                         0.94853
## Prevalence                      0.5609                         0.06536
## Detection Rate                  0.5112                         0.01520
## Detection Prevalence            0.6357                         0.02535
## Balanced Accuracy               0.8138                         0.61082
##                      Class: non functional
## Sensitivity                         0.7647
## Specificity                         0.9151
## Pos Pred Value                      0.8431
## Neg Pred Value                      0.8670
## Prevalence                          0.3737
## Detection Rate                      0.2858
## Detection Prevalence                0.3390
## Balanced Accuracy                   0.8399
```

Veiem que la curesa que obtenim, del 80.7% no està del tot malament però té molt espai de millora, especialment en la sensibilitat detectant bombes funcionals que necessiten reparació (el grup més abundant i pitjor classificat).

Ara provem amb la escisió:

```
t2 <- t
t2_1 <- subset(t2, population >= 215 & amount_tsh >= 600 )
t2_2 <- subset(t2, population < 215 & amount_tsh < 600 )
h1 <-holdout(t2_1$status_group,ratio=2/3,mode="stratified")
data_train1<-t2_1[h1$tr,]
data_test1<-t2_1[h1$ts,]
m1 <- randomForest(status_group~., data = data_train1)

h2 <-holdout(t2_2$status_group,ratio=2/3,mode="stratified")
data_train2<-t2_2[h2$tr,]
data_test2<-t2_2[h2$ts,]
m2 <- randomForest(status_group~., data = data_train2)
data_test12 <- rbind(data_test1, data_test2)

predictions1 <- predict(m1, data_test1, type = "class")
predictions2 <- predict(m2, data_test2, type = "class")

predictions <- rbind(predictions1, predictions2)
```

```
## Warning in rbind(predictions1, predictions2): number of columns of result is not
## a multiple of vector length (arg 1)
```

```r
data_test <- rbind(data_test1, data_test2)



confusionMatrix(predictions1, data_test1$status_group)
```

```
## Confusion Matrix and Statistics
##
##                          Reference
## Prediction                functional functional needs repair non functional
##   functional                     349                       34              39
##   functional needs repair         12                       20               5
##   non functional                  10                        3              65
##
## Overall Statistics
##
##                Accuracy : 0.8082
##                  95% CI : (0.7723, 0.8407)
##     No Information Rate : 0.6909
##     P-Value [Acc > NIR] : 5.597e-10
##
##                   Kappa : 0.5436
##
##  Mcnemar's Test P-Value : 3.321e-06
##
## Statistics by Class:
##
##                      Class: functional Class: functional needs repair
## Sensitivity                     0.9407                          0.35088
## Specificity                     0.5602                          0.96458
## Pos Pred Value                  0.8270                          0.54054
## Neg Pred Value                  0.8087                          0.92600
## Prevalence                      0.6909                          0.10615
## Detection Rate                  0.6499                          0.03724
## Detection Prevalence            0.7858                          0.06890
## Balanced Accuracy               0.7505                          0.65773
##                      Class: non functional
## Sensitivity                         0.5963
## Specificity                         0.9696
## Pos Pred Value                      0.8333
## Neg Pred Value                      0.9041
## Prevalence                          0.2030
## Detection Rate                      0.1210
## Detection Prevalence                0.1453
## Balanced Accuracy                   0.7830
```

```r
confusionMatrix(predictions2, data_test2$status_group)
```

```
## Confusion Matrix and Statistics
##
##                          Reference
## Prediction                functional functional needs repair non functional
```

```
##   functional                   3579                     237            590
##   functional needs repair        40                      91             14
##   non functional                319                      71           2136
##
## Overall Statistics
##
##                Accuracy : 0.8204
##                  95% CI : (0.8113, 0.8293)
##     No Information Rate : 0.5565
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.6507
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
## Statistics by Class:
##
##                      Class: functional Class: functional needs repair
## Sensitivity                     0.9088                         0.22807
## Specificity                     0.7365                         0.99191
## Pos Pred Value                  0.8123                         0.62759
## Neg Pred Value                  0.8656                         0.95557
## Prevalence                      0.5565                         0.05638
## Detection Rate                  0.5057                         0.01286
## Detection Prevalence            0.6226                         0.02049
## Balanced Accuracy               0.8227                         0.60999
##                      Class: non functional
## Sensitivity                         0.7796
## Specificity                         0.9101
## Pos Pred Value                      0.8456
## Neg Pred Value                      0.8673
## Prevalence                          0.3872
## Detection Rate                      0.3018
## Detection Prevalence                0.3569
## Balanced Accuracy                   0.8448
```

Contrast d'Hipòtesis En aquest cas examinarem la relació entre l'any de cosntrucció i l'estatus de la bomba. En concret testarem la hipòtesi de que la mitjada de l'any de cosntrucció per les bomes funcionals és més elevada (són més noves) que les no-funcionals. Farem l'anàlisi tenint en consideració un nivell de significància 0.05:

H0:$\mu_{functional} - \mu_{nonfunctional} = 0$ Ha:$\mu_{functional} - \mu_{nonfunctional} > 0$

```r
#escindim la mostra entre f(funcional) i m (non functional)
tf <- t[t$status_group == "functional",]
tm <- t[t$status_group == "non functional",]

#calculem el tamany de les dues mostres
nf <- nrow(tf)
nm <- nrow(tm)

#observem aquests valors
print(c(nf, nm))
```

```
## [1] 21704 14459
```

```
#mitjanes mostrals
xf <- sum(tf$construction_year)/nf
xm <- sum(tm$construction_year)/nm
print(c(xf, xm))
```

```
## [1] 1999.939 1992.398
```

```
#desviacions tipiques mostrals
dif_xf <- sapply(tf$construction_year, function(x) (x-xf)**2)
dif_xm <- sapply(tm$construction_year, function(x) (x-xm)**2)
sf <- sqrt(sum(dif_xf)/(nf-1))
sm <- sqrt(sum(dif_xm)/(nm-1))

#estadístic de contrast

s_fm <- sqrt(sf**2/nf+sm**2/nm)
z_fm <- (xf-xm)/s_fm

#p- valor
p_val_fm <- 2*pnorm(-abs(z_fm))
p_val_fm
```

```
## [1] 0
```

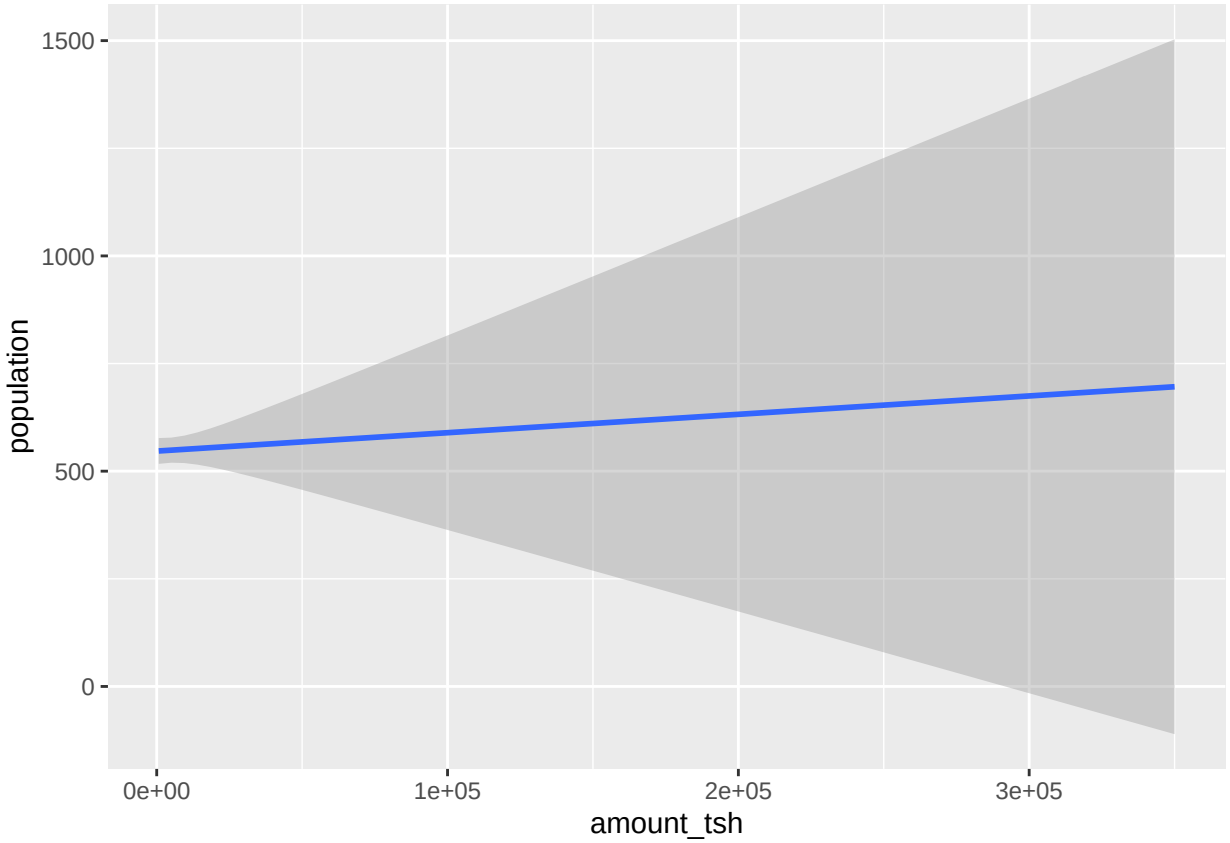
El valor p és menor que el nivell de significància lo qual ens permet rebutjar la hipòtesi nula.

5. Representació dels resultats a partir de taules i gràfiques.

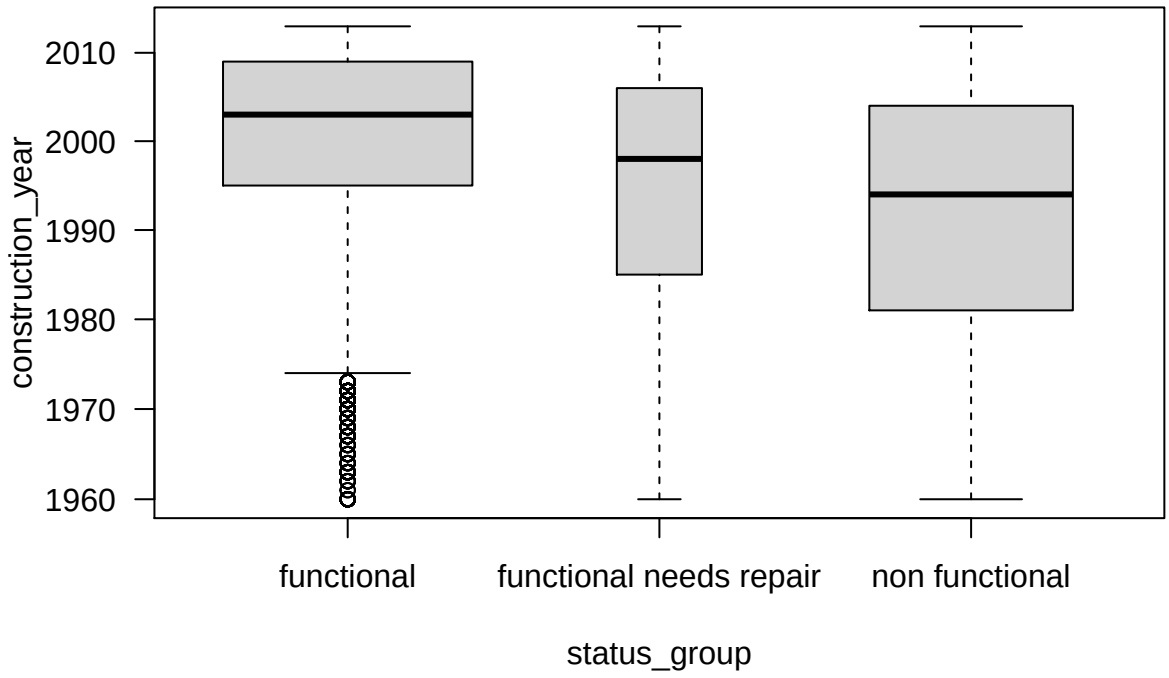
Les taules de confusió del model random forest es poden trobar a l'apartat 4.3.

```
ggplot(sub2,aes(amount_tsh, population)) + geom_smooth(method='lm')
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
boxplot(construction_year ~ status_group, data = t, varwidth = TRUE,
        log = "y", las = 1)
```



6. Resolució del problema. A partir dels resultats obtinguts, quines són les conclusions? Els resultats permeten respondre al problema?

Partíem d'un dataset molt complert, on el gruix de treball de neteja ha residit en reduïr-ne la dimensionalitat per tal d'adherir-nos a criteris de rellevància, reduïr la redundància d'alguns camps i simplificar l'anàlisi.

Pel que fa a l'anàlisi hem pogut confirmar amb un contrast d'hipòtesi que mitjana de l'any de construcció de les bombes que funcionen és major que el de les bombes que no funcionen. Això ens diu una cosa que intuïtivament té sentit, que les bombes més velles tenen més tendència a deixar de funcionar.

Més interessant encara, hem trobat que en dos dels camps, el camp població i el camp amount_tsh, tenen un comportament que varia per trams. Després de comprobar la correlació que presenten en el tram de population < 200 & amount_tsh > 0 & amount_tsh < 600, hem provat d'entrenar un model Random Forest amb el dataset sencer o bé escindint pels trams establerts per l'anàlisi anterior i el que hem trobat és que en el cas del model entrenat amb el dataset sencer té una curesa del 80% versus els dos models entrenats amb el dataset escindit, que han obtingut una curesa lleugerament superior (vora el 82%) en ambdos casos.

Referències

[Data Driven Pump it Up: Data Mining the Water Table] : https://www.drivendata.org/competitions/7/pump-it-up-data-mining-the-water-table/

[Taarifa] : http://taarifa.org/

[Tanzanian Ministry of Water] : https://www.maji.go.tz/

[MCA analysis in R] : http://www.gastonsanchez.com/visually-enforced/how-to/2012/10/13/MCA-in-R/

[maps] : https://www.littlemissdata.com/blog/maps

[randomForest] : https://rstudio-pubs-static.s3.amazonaws.com/245066_f7b5962e8ab84594829b84f06ced39b6.html

Glossari

VWC = Village water committee WUA = water users association WUG = water users group