ADS506-01-FA22 - Final Project

Team 1

11/05/2022

RMarkdown global setup

```
knitr::opts_chunk$set(fig.align = 'center')
```

```
library(AppliedPredictiveModeling)
library(BioStatR)
library(car)
library(caret)
library(class)
library(corrplot)
library(datasets)
library(e1071)
library(Hmisc)
library(mlbench)
library(gridExtra)
library(psych)
library(randomForest)
library(RANN)
library(rpart)
library(rpart.plot)
library(scales)
library(tidyverse)

set.seed(1699)
```

Create function to generate boxplots for continuous variables

```
# Define function to produce formatted boxplots
box_comp <- function(xcol = c(), df = NA, rtn_met = TRUE) {
  sig <- 3
  metrics_df01 <- data.frame(metric = c("",
                                        "Total N:",
                                        "Count",
                                        "NA Count",
                                        "Mean",
                                        "Median",
                                        "Standard Deviation",
                                        "Variance",
                                        "Range",
                                        "Min",
                                        "Max",
```

```
                                        "25th Percentile",
                                        "75th Percentile",
                                        "Subset w/o Outliers:",
                                        "Count",
                                        "%",
                                        "Outlier %",
                                        "NA Count",
                                        "Mean",
                                        "Median",
                                        "Standard Deviation",
                                        "Variance",
                                        "Range",
                                        "Min",
                                        "Max"
                                        ))
for (var in xcol) {
  df_s1 <- df[, var]
  df_s1s1 <-data.frame(df_s1)
  df_s1_fit <- preProcess(df_s1s1,
                          method = c("center", "scale"))
  df_s1_trans <- predict(df_s1_fit, df_s1s1)


  # Calculate quartiles
  var_iqr_lim <- IQR(df_s1) * 1.5
  var_q1 <- quantile(df_s1, probs = c(.25))
  var_otlow <- var_q1 - var_iqr_lim
  var_q3 <- quantile(df_s1, probs = c(.75))
  var_othigh <- var_q3 + var_iqr_lim

  # Subset non-outlier data
  var_non_otlr_df01 <- subset(df, (abs(df_s1_trans) <= 3))
  #var_non_otlr_df01 <- subset(df, (df_s1 > var_otlow & df_s1 < var_othigh))
  df_s2 <- var_non_otlr_df01[, var]

  # Begin calculating measures of centrality & dispersion
  var_mean <- mean(df_s1)
  var_non_otlr_df01_trunc_mean <- mean(df_s2)
  var_med <- median(df_s1)
  var_non_otlr_df01_trunc_med <- median(df_s2)
  var_mode <- mode(df_s1)
  var_non_otlr_df01_trunc_mode <- mode(df_s2)
  var_stde <- sd(df_s1)
  var_non_otlr_df01_trunc_stde <- sd(df_s2)
  var_vari <- var(df_s1)
  var_non_otlr_df01_trunc_vari <- var(df_s2)
  var01_min <- min(df[, var])
  var01_max <- max(df[, var])
  var01_range <- var01_max - var01_min
  var02_min <- min(var_non_otlr_df01[, var])
  var02_max <- max(var_non_otlr_df01[, var])
  var02_range <- var02_max - var02_min
```

```
    # Configure y-axis min & max to sync graphs
    plot_min <- min(var01_min, var02_min)
    plot_max <- max(var01_max, var02_max)
    nonoutlier_perc <- round((as.numeric(dim(var_non_otlr_df01)[1] /
↪  as.numeric(dim(df)[1]))) * 100, 1)
    measure_val01 <- c(paste0("Variable: ", var),
                       "",
                       as.character(dim(df)[1]),
                       sum(is.na(df_s1)),
                       round(var_mean, sig),
                       round(var_med, sig),
                       round(var_stde, sig),
                       round(var_vari, sig),
                       round(var01_range, sig),
                       round(var01_min, sig),
                       round(var01_max, sig),
                       round(var_q1, sig),
                       round(var_q3, sig),
                       "",
                       as.character(dim(var_non_otlr_df01)[1]),
                       paste0(nonoutlier_perc, "%"),
                       paste0(round(100 - nonoutlier_perc, 1), "%"),
                       sum(is.na(df_s2)),
                       round(var_non_otlr_df01_trunc_mean, sig),
                       round(var_non_otlr_df01_trunc_med, sig),
                       round(var_non_otlr_df01_trunc_stde, sig),
                       round(var_non_otlr_df01_trunc_vari, sig),
                       round(var02_range, sig),
                       round(var02_min, sig),
                       round(var02_max, sig)
                       )

    var_name <- paste0("Variable: ", var)
    metrics_df01[, ncol(metrics_df01) + 1] <- measure_val01
}
  boxplot(df)
  if(rtn_met == TRUE) {
    return(metrics_df01)
  }
}
```

Importing Train/Test Datasets

```
train_x01_df01 <- read.csv("../data/Drinking
↪  Water/analyte_tests_drinking_water_datasd.csv", header = TRUE, sep = ",")
train_x02_df01 <- read.csv("../data/Campaign Funds/financial_support_2021_datasd_v1.csv",
↪  header = TRUE, sep = ",")
train_x03_df01 <- read.csv("../data/Ocean Water/water_quality_2020_2021_datasd.csv",
↪  header = TRUE, sep = ",")
#train_x01_df01 <- read.csv("../data/FD Incidents/fd_incidents_2022_datasd_v1.csv",
↪  header = TRUE, sep = ",")
#test_x01_df01 <- read.csv("../data/outlier-included/biodeg_test.csv", header = TRUE, sep
↪  = ",")
```

```
#train_y01_df01 <- read.csv("../data/outlier-included/response_train.csv", header = TRUE,
↪  sep = ",")
#test_y01_df01 <- read.csv("../data/outlier-included/response_test.csv", header = TRUE,
↪  sep = ",")

#train_y01_vc01 <- train_y01_df01[["x"]]
#test_y01_vc01 <- test_y01_df01[["x"]]

print(head(train_x01_df01))
```

```
##   date_sample sample_source sample_id   analyte value_qualifier analyte_value
## 1  2022-01-01       55A SYS  W1470689  FLUORIDE                         0.469
## 2  2022-01-02       174 SYS  W1470694  FLUORIDE                         0.438
## 3  2022-01-03       313 SYS  W1471820  FLUORIDE                         0.478
## 4  2022-01-03       50A SYS  W1471858     COLOR              ND            NA
## 5  2022-01-03       50A SYS  W1471858       TON                         1.000
## 6  2022-01-03       50A SYS  W1471858 TURBIDITY                         0.100
##   value_units                          source_description
## 1        MG/L                          5183 Arvinels Ave.
## 2        MG/L 3250 Camino Del Rio North; Sample Stanchion
## 3        MG/L                         11602 Calle Paracho
## 4       COLOR                          2693 Melbourne Dr.
## 5        ODOR                          2693 Melbourne Dr.
## 6         NTU                          2693 Melbourne Dr.
```

```
describe(train_x01_df01)
```

```
##                    vars     n    mean      sd  median trimmed     mad  min
## date_sample*          1 20323 1252.57  711.77 1262.00 1256.78  889.56 1.00
## sample_source*        2 20323   24.93   16.12   30.00   25.41   20.76 1.00
## sample_id*            3 20323 4403.67 2507.52 4438.00 4416.50 3149.04 1.00
## analyte*              4 20323    2.60    1.19    3.00    2.63    1.48 1.00
## value_qualifier*      5 20323    1.85    1.25    1.00    1.69    0.00 1.00
## analyte_value         6 15443    0.74    0.62    0.73    0.66    0.58 0.04
## value_units*          7 20323    2.59    1.18    3.00    2.61    1.48 1.00
## source_description*   8 20323   25.91   11.44   28.00   26.14   14.83 1.00
##                     max   range  skew kurtosis    se
## date_sample*       2467 2466.00 -0.05    -1.18  4.99
## sample_source*       45   44.00 -0.21    -1.60  0.11
## sample_id*         8693 8692.00 -0.04    -1.18 17.59
## analyte*              4    3.00 -0.22    -1.47  0.01
## value_qualifier*      4    3.00  1.02    -0.77  0.01
## analyte_value        10    9.96  1.55     8.63  0.00
## value_units*          4    3.00 -0.21    -1.45  0.01
## source_description*  45   44.00 -0.15    -1.22  0.08
```

```
print(head(train_x02_df01))
```

```
##   form schedule   schedule_description recipient_id             recipient_name
## 1  460        A Monetary contributions      1421046 Leventhal for Council 2020
## 2  460        A Monetary contributions      1421046 Leventhal for Council 2020
## 3  460        A Monetary contributions      1421046 Leventhal for Council 2020
## 4  460        A Monetary contributions      1414821 Todd Gloria for Mayor 2020
```

```
## 5  460        A Monetary contributions      1414821 Todd Gloria for Mayor 2020
## 6  460        A Monetary contributions      1414821 Todd Gloria for Mayor 2020
##             date_report_period_from             date_report_period_to
## 1 2021-01-01T00:00:00.0000000-08:00 2021-01-15T00:00:00.0000000-08:00
## 2 2021-01-01T00:00:00.0000000-08:00 2021-01-15T00:00:00.0000000-08:00
## 3 2021-01-01T00:00:00.0000000-08:00 2021-01-15T00:00:00.0000000-08:00
## 4 2021-01-01T00:00:00.0000000-08:00 2021-04-07T00:00:00.0000000-07:00
## 5 2021-01-01T00:00:00.0000000-08:00 2021-04-07T00:00:00.0000000-07:00
## 6 2021-01-01T00:00:00.0000000-08:00 2021-04-07T00:00:00.0000000-07:00
##   contributor_code contributor_last contributor_first address_city_contributor
## 1              IND        Leventhal              Joe                San Diego
## 2              IND        Leventhal              Joe                San Diego
## 3              IND        Leventhal              Joe                San Diego
## 4              IND            Adams          Matthew                San Diego
## 5              IND         Andersen              Jim             San Clemente
## 6              IND        Armstrong             Eric                 Carlsbad
##   address_state_contributor address_zip_contributor
## 1                        CA                  92127
## 2                        CA                  92127
## 3                        CA                  92127
## 4                        CA                  92119
## 5                        CA                  92672
## 6                        CA                  92009
##                               contributor_emp
## 1                          Dinsmore and Shohl
## 2                          Dinsmore and Shohl
## 3                          Dinsmore and Shohl
## 4 Building Industry Association San Diego  County
## 5                  Chelsea Investment Corporation
## 6                          Fuscoe Engineering
##                      contributor_occ                 date_contribution
## 1                           Attorney 2021-01-15T00:00:00.0000000-08:00
## 2                           Attorney 2021-01-15T00:00:00.0000000-08:00
## 3                           Attorney 2021-01-15T00:00:00.0000000-08:00
## 4 Vice President of Government Affairs 2021-01-22T00:00:00.0000000-08:00
## 5                   Affordable Housing 2021-01-21T00:00:00.0000000-08:00
## 6                       Civil Engineer 2021-01-14T00:00:00.0000000-08:00
##   contribution_amount contribution_annual      contribution_desc contributor_id
## 1                2500                   0 Forgiven Loan Received             NA
## 2               20000                   0 Forgiven Loan Received             NA
## 3                1250                   0 Forgiven Loan Received             NA
## 4                 150                 150                                    NA
## 5                 150                 150                                    NA
## 6                 150                 150                                    NA
##   intermediary_last intermediary_first address_city_intermediary
## 1                                   NA
## 2                                   NA
## 3                                   NA
## 4                                   NA
## 5                                   NA
## 6                                   NA
##   address_state_intermediary address_zip_intermediary intermediary_emp
## 1                                                 NA               NA
## 2                                                 NA               NA
```

```
## 3                                                    NA               NA
## 4                                                    NA               NA
## 5                                                    NA               NA
## 6                                                    NA               NA
##   intermediary_occ filing_id year_report
## 1               NA 195770416        2021
## 2               NA 195770416        2021
## 3               NA 195770416        2021
## 4               NA 200459049        2021
## 5               NA 200459049        2021
## 6               NA 200459049        2021
```

```
describe(train_x02_df01)
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf
```

```
##                            vars   n         mean        sd    median
## form                          1 191       460.00      0.00       460
## schedule*                     2 191         1.31      0.99         1
## schedule_description*         3 191         2.21      0.72         2
## recipient_id                  4 191   1414459.30  22341.52   1415989
## recipient_name*               5 191         5.49      1.59         5
## date_report_period_from*      6 191         1.00      0.00         1
## date_report_period_to*        7 191         5.02      1.61         4
## contributor_code*             8 191         1.92      0.28         2
## contributor_last*             9 191        74.36     47.69        77
## contributor_first*           10 191        58.20     37.29        58
## address_city_contributor*    11 191        19.74      9.10        26
## address_state_contributor*   12 191         1.93      0.30         2
## address_zip_contributor      13 175     92079.62    469.28     92106
## contributor_emp*             14 191        51.70     38.36        51
## contributor_occ*             15 191        43.45     29.60        44
## date_contribution*           16 191        20.17     12.82        18
## contribution_amount          17 191       550.67   1667.81       250
## contribution_annual          18 175       511.75   1129.20       250
## contribution_desc*           19 191         1.14      0.48         1
## contributor_id               20   0          NaN        NA        NA
## intermediary_last*           21 191         1.69      0.91         1
## intermediary_first           22   0          NaN        NA        NA
## address_city_intermediary*   23 191         1.69      0.91         1
## address_state_intermediary*  24 191         1.69      0.91         1
```

```
## address_zip_intermediary      25  58      2144.00      0.00      2144
## intermediary_emp              26   0          NaN        NA        NA
## intermediary_occ              27   0          NaN        NA        NA
## filing_id                     28 191 200383576.40 760610.43 200489948
## year_report                   29 191      2021.00      0.00      2021
##                                  trimmed      mad       min       max   range
## form                              460.00     0.00       460       460       0
## schedule*                           1.00     0.00         1         5       4
## schedule_description*               2.00     0.00         1         5       4
## recipient_id                  1417547.24  1731.68   1280768   1431354  150586
## recipient_name*                     5.69     2.97         1         8       7
## date_report_period_from*            1.00     0.00         1         1       0
## date_report_period_to*              5.10     1.48         1         7       6
## contributor_code*                   2.00     0.00         1         2       1
## contributor_last*                  74.33    62.27         1       155     154
## contributor_first*                 57.81    47.44         1       127     126
## address_city_contributor*          20.99     0.00         1        33      32
## address_state_contributor*          2.00     0.00         1         3       2
## address_zip_contributor         92089.09    35.58     89135     95818    6683
## contributor_emp*                   49.92    47.44         1       125     124
## contributor_occ*                   43.09    44.48         1        96      95
## date_contribution*                 19.69    13.34         1        50      49
## contribution_amount               313.24   222.39         0     20000   20000
## contribution_annual               326.26   222.39         0     10000   10000
## contribution_desc*                  1.01     0.00         1         4       3
## contributor_id                       NaN       NA       Inf      -Inf    -Inf
## intermediary_last*                  1.61     0.00         1         3       2
## intermediary_first                   NaN       NA       Inf      -Inf    -Inf
## address_city_intermediary*          1.61     0.00         1         3       2
## address_state_intermediary*         1.61     0.00         1         3       2
## address_zip_intermediary         2144.00     0.00      2144      2144       0
## intermediary_emp                     NaN       NA       Inf      -Inf    -Inf
## intermediary_occ                     NaN       NA       Inf      -Inf    -Inf
## filing_id                    200506713.27 45810.86 195770416 200573475 4803059
## year_report                      2021.00     0.00      2021      2021       0
##                                   skew kurtosis       se
## form                               NaN      NaN     0.00
## schedule*                         3.00     7.37     0.07
## schedule_description*             3.08     8.31     0.05
## recipient_id                     -5.58    30.58  1616.58
## recipient_name*                  -0.78     0.15     0.12
## date_report_period_from*           NaN      NaN     0.00
## date_report_period_to*           -0.06    -0.83     0.12
## contributor_code*                -2.98     6.92     0.02
## contributor_last*                -0.03    -1.25     3.45
## contributor_first*                0.05    -1.18     2.70
## address_city_contributor*        -0.92    -0.62     0.66
## address_state_contributor*       -2.00     6.49     0.02
## address_zip_contributor          -0.29    39.86    35.47
## contributor_emp*                  0.21    -1.22     2.78
## contributor_occ*                  0.06    -1.43     2.14
## date_contribution*                0.39    -0.76     0.93
## contribution_amount               9.44    99.84   120.68
## contribution_annual               6.72    47.63    85.36
```

```
## contribution_desc*           4.16    19.40     0.03
## contributor_id                 NA       NA       NA
## intermediary_last*           0.64    -1.48     0.07
## intermediary_first             NA       NA       NA
## address_city_intermediary*   0.64    -1.48     0.07
## address_state_intermediary*  0.64    -1.48     0.07
## address_zip_intermediary      NaN      NaN     0.00
## intermediary_emp               NA       NA       NA
## intermediary_occ               NA       NA       NA
## filing_id                   -5.83    32.42 55035.84
## year_report                   NaN      NaN     0.00
```

```
print(head(train_x03_df01))
```

```
##       sample station depth_m date_sample        time project parameter
## 1 2001018683     S11      NA  2020-01-01 9:18:00 PST    SBOO    ENTERO
## 2 2001018683     S11      NA  2020-01-01 9:18:00 PST    SBOO     TOTAL
## 3 2001018683     S11      NA  2020-01-01 9:18:00 PST    SBOO     FECAL
## 4 2001018680      S4      NA  2020-01-01                SBOO     TOTAL
## 5 2001018680      S4      NA  2020-01-01                SBOO     FECAL
## 6 2001018682      S6      NA  2020-01-01 9:31:00 PST    SBOO     FECAL
##   qualifier value      units
## 1         e   220 CFU/100 mL
## 2        NR    NA CFU/100 mL
## 3        NR    NA CFU/100 mL
## 4        NS    NA CFU/100 mL
## 5        NS    NA CFU/100 mL
## 6        NR    NA CFU/100 mL
```

```
describe(train_x03_df01)
```

```
##              vars     n         mean          sd       median      trimmed
## sample          1 70163 2.056056e+09 50138313.89 2.012275e+09 2.055917e+09
## station*        2 70163 4.412000e+01       35.21 4.800000e+01 4.304000e+01
## depth_m         3 63330 1.297000e+01       18.18 9.000000e+00 8.680000e+00
## date_sample*    4 70163 2.033200e+02      106.33 2.030000e+02 2.068600e+02
## time*           5 70163 1.873000e+02      132.02 1.500000e+02 1.866100e+02
## project*        6 70163 1.480000e+00        0.50 1.000000e+00 1.470000e+00
## parameter*      7 70163 5.490000e+00        2.83 5.000000e+00 5.490000e+00
## qualifier*      8 70163 1.530000e+00        0.95 1.000000e+00 1.290000e+00
## value           9 70078 2.162000e+02     1480.17 8.340000e+00 1.674000e+01
## units*         10 70163 4.140000e+00        2.10 3.000000e+00 4.050000e+00
##                    mad        min        max     range  skew kurtosis
## sample       16673679.87 2001018680 2112291250 111272570  0.02    -1.98
## station*           56.34          1        104       103  0.04    -1.62
## depth_m            10.38          1         98        97  2.82     8.15
## date_sample*      137.88          1        369       368 -0.20    -1.15
## time*             186.81          1        384       383  0.07    -1.65
## project*            0.00          1          2         1  0.10    -1.99
## parameter*          2.97          1         10         9  0.03    -1.19
## qualifier*          0.00          1          8         7  1.84     2.33
## value               9.40          0      16000     16000  8.78    80.13
## units*              1.48          1          8         7  0.44    -0.91
##                    se
## sample       189284.76
```

```
## station*          0.13
## depth_m           0.07
## date_sample*      0.40
## time*             0.50
## project*          0.00
## parameter*        0.01
## qualifier*        0.00
## value             5.59
## units*            0.01
```

```
train_x01_df01_ss <- train_x01_df01 %>%
  group_by(sample_source, date_sample) %>%
  summarise(Count = n())
```

```
## `summarise()` has grouped output by 'sample_source'. You can override using the
## `.groups` argument.
```

```
train_x01_df01_ay <- train_x01_df01 %>%
  group_by(analyte) %>%
  summarise(Count = n())

train_x01_df01_date <- train_x01_df01 %>%
  group_by(date_sample) %>%
  summarise(Count = n())

train_x01_df01_full <- train_x01_df01 %>%
  group_by(date_sample, sample_source, analyte) %>%
  summarise(Total = sum(analyte_value))
```

```
## `summarise()` has grouped output by 'date_sample', 'sample_source'. You can
## override using the `.groups` argument.
```

```
print(head(train_x01_df01_ss))
```

```
## # A tibble: 6 x 3
## # Groups:   sample_source [1]
##   sample_source date_sample Count
##   <chr>         <chr>       <int>
## 1 128 SYS       2016-01-05      3
## 2 128 SYS       2016-01-12      3
## 3 128 SYS       2016-01-19      3
## 4 128 SYS       2016-01-26      3
## 5 128 SYS       2016-02-02      3
## 6 128 SYS       2016-02-09      3
```

```
print(head(train_x01_df01_ay))
```

```
## # A tibble: 4 x 2
##   analyte   Count
##   <chr>     <int>
## 1 COLOR      5850
## 2 FLUORIDE   2489
## 3 TON        5826
## 4 TURBIDITY  6158
```

```
print(head(train_x01_df01_date))
```

```
## # A tibble: 6 x 2
##   date_sample Count
##   <chr>       <int>
## 1 2016-01-01      1
## 2 2016-01-02      1
## 3 2016-01-03      1
## 4 2016-01-04      1
## 5 2016-01-05     55
## 6 2016-01-06      1
```

```
print(head(train_x01_df01_full))
```

```
## # A tibble: 6 x 4
## # Groups:   date_sample, sample_source [5]
##   date_sample sample_source analyte   Total
##   <chr>       <chr>         <chr>     <dbl>
## 1 2016-01-01  249A SYS      FLUORIDE  0.651
## 2 2016-01-02  281 SYS       FLUORIDE  0.666
## 3 2016-01-03  150 SYS       FLUORIDE  0.681
## 4 2016-01-04  259 SYS       FLUORIDE  0.675
## 5 2016-01-05  128 SYS       COLOR    NA
## 6 2016-01-05  128 SYS       TON       1
```

```
train_x03_df01_ss <- train_x03_df01 %>%
  group_by(station, date_sample) %>%
  summarise(Count = n())
```

```
## `summarise()` has grouped output by 'station'. You can override using the
## `.groups` argument.
```

```
train_x03_df01_ay <- train_x03_df01 %>%
  group_by(parameter) %>%
  summarise(Count = n())

train_x03_df01_date <- train_x03_df01 %>%
  group_by(date_sample) %>%
  summarise(Count = n())

train_x03_df01_full <- train_x03_df01 %>%
  group_by(date_sample, station, parameter) %>%
  summarise(Total = sum(value))
```

```
## `summarise()` has grouped output by 'date_sample', 'station'. You can override
## using the `.groups` argument.
```

```
print(head(train_x03_df01_ss))
```

```
## # A tibble: 6 x 3
## # Groups:   station [1]
##   station date_sample Count
##   <chr>   <chr>       <int>
## 1 A1      2020-01-02     30
## 2 A1      2020-01-07     30
```

```
## 3 A1      2020-01-13     30
## 4 A1      2020-01-21     30
## 5 A1      2020-01-28     30
## 6 A1      2020-02-05     30
```

```
print(head(train_x03_df01_ay))
```

```
## # A tibble: 6 x 2
##   parameter   Count
##   <chr>       <int>
## 1 CHLOROPHYLL  6645
## 2 DENSITY      6645
## 3 DO           6645
## 4 ENTERO       8680
## 5 FECAL        7489
## 6 PH           6645
```

```
print(head(train_x03_df01_date))
```

```
## # A tibble: 6 x 2
##   date_sample Count
##   <chr>       <int>
## 1 2020-01-01      8
## 2 2020-01-02    474
## 3 2020-01-03      4
## 4 2020-01-05      4
## 5 2020-01-07    480
## 6 2020-01-08     24
```

```
print(head(train_x03_df01_full))
```

```
## # A tibble: 6 x 4
## # Groups:   date_sample, station [3]
##   date_sample station parameter Total
##   <chr>       <chr>   <chr>     <dbl>
## 1 2020-01-01  S11     ENTERO      220
## 2 2020-01-01  S11     FECAL        NA
## 3 2020-01-01  S11     TOTAL        NA
## 4 2020-01-01  S4      FECAL        NA
## 5 2020-01-01  S4      TOTAL        NA
## 6 2020-01-01  S6      ENTERO      200
```

Run function to create comparative boxplots

```
x01_lst01 <- c()

x01_lst02 <- c("analyte_value")
x02_lst02 <- c("contribution_amount",
               "contribution_annual")
x03_lst02 <- c("value")

x01_lst03 <- c()

x01_lst04 <- c()
```

```r
x01_lst05 <- c()

x01_lst06 <- c()

x01_lst07 <- c()

x01_lst08 <- c()

x01_lst09 <- c()

x01_lst10 <- c()

x01_lst11 <- c()

x01_lst12 <- c()

x01_lst13 <- c()

x01_lst14 <- c()

x01_lst15 <- c()

x01_lst16 <- c()

train_x01_df01_cols01 <- colnames(train_x01_df01)
print(train_x01_df01_cols01)
```

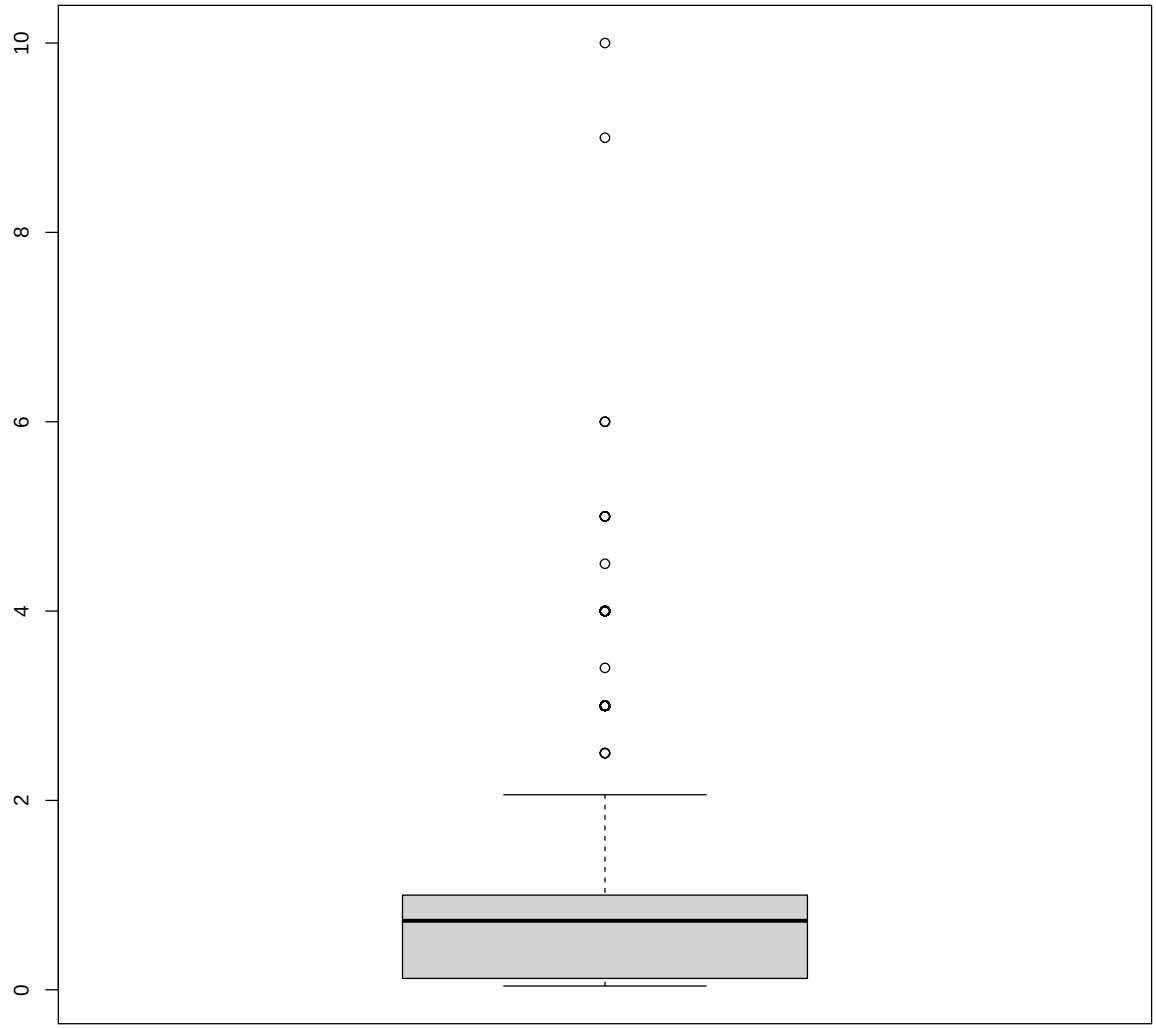
```
## [1] "date_sample"        "sample_source"      "sample_id"
## [4] "analyte"            "value_qualifier"    "analyte_value"
## [7] "value_units"        "source_description"
```

```r
#train_x01_df01_metrics <- box_comp(xcol = train_x01_df01_cols01, df = train_x01_df01)
#train_x01_df01_metrics
#write.csv(train_x01_df01_metrics, "../outputs/demos.csv", row.names = FALSE)

train_x01_df03 <- subset(x = train_x01_df01, select = x01_lst02)
train_x01_df03 <- na.omit(train_x01_df03)
print(head(train_x01_df03))
```

```
##   analyte_value
## 1         0.469
## 2         0.438
## 3         0.478
## 5         1.000
## 6         0.100
## 8         2.000
```

```r
box_comp(xcol = x01_lst02, df = subset(x = train_x01_df03, select = x01_lst02), rtn_met =
↪  TRUE)
```



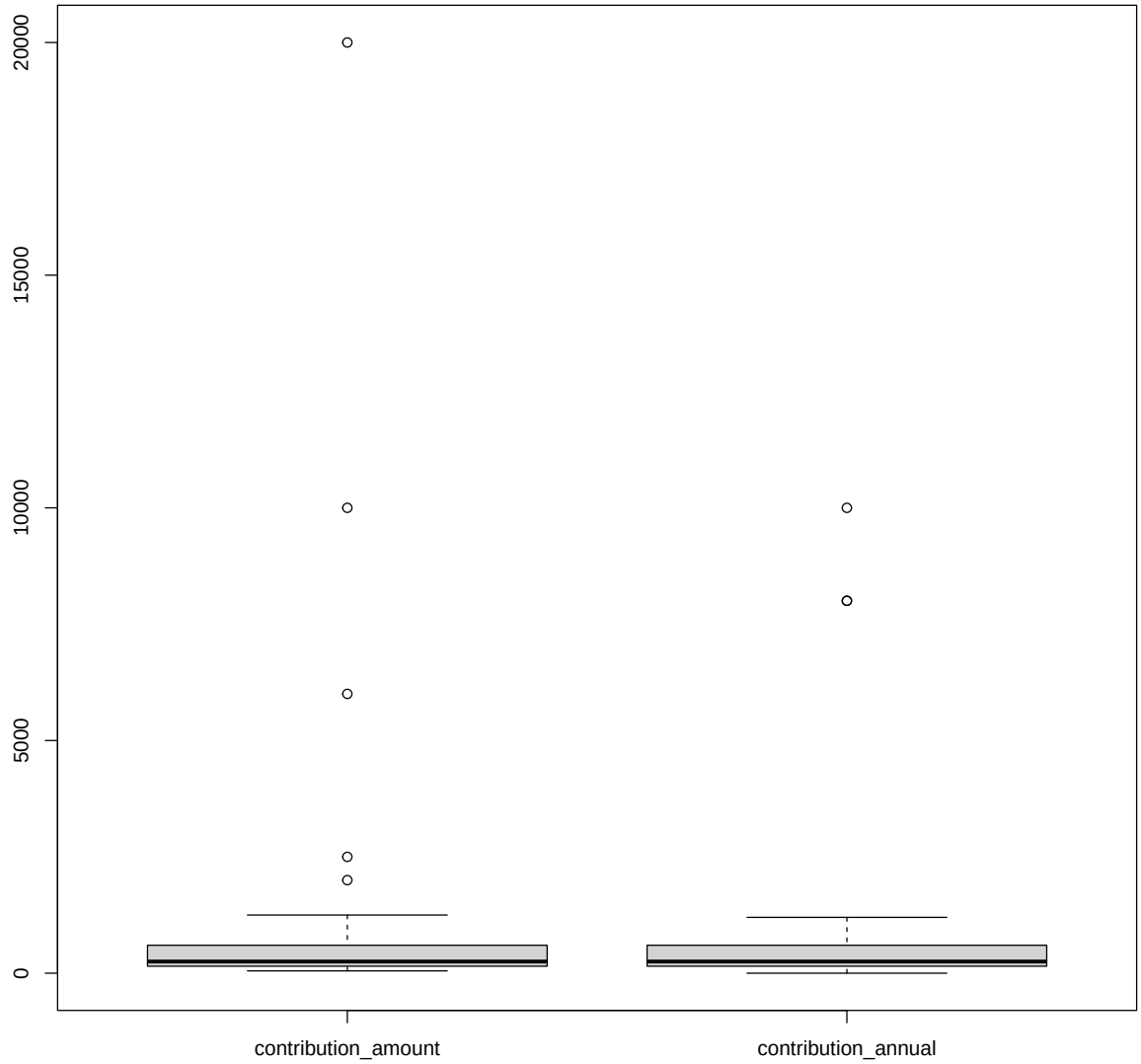
	metric	V2
1		Variable: analyte_value
2	Total N:	
3	Count	15443
4	NA Count	0
5	Mean	0.735
6	Median	0.729
7	Standard Deviation	0.618
8	Variance	0.382
9	Range	9.96
10	Min	0.04
11	Max	10
12	25th Percentile	0.12
13	75th Percentile	1

```
## 14 Subset w/o Outliers:
## 15                 Count                   15291
## 16                     %                     99%
## 17             Outlier %                      1%
## 18              NA Count                       0
## 19                  Mean                   0.709
## 20                Median                   0.706
## 21   Standard Deviation                   0.552
## 22              Variance                   0.305
## 23                 Range                    2.46
## 24                   Min                    0.04
## 25                   Max                     2.5
```

```
train_x02_df03 <- subset(x = train_x02_df01, select = x02_lst02)
train_x02_df03 <- na.omit(train_x02_df03)
print(head(train_x02_df03))
```

```
##   contribution_amount contribution_annual
## 1                2500                   0
## 2               20000                   0
## 3                1250                   0
## 4                 150                 150
## 5                 150                 150
## 6                 150                 150
```

```
box_comp(xcol = x02_lst02, df = subset(x = train_x02_df03, select = x02_lst02), rtn_met =
↪  TRUE)
```



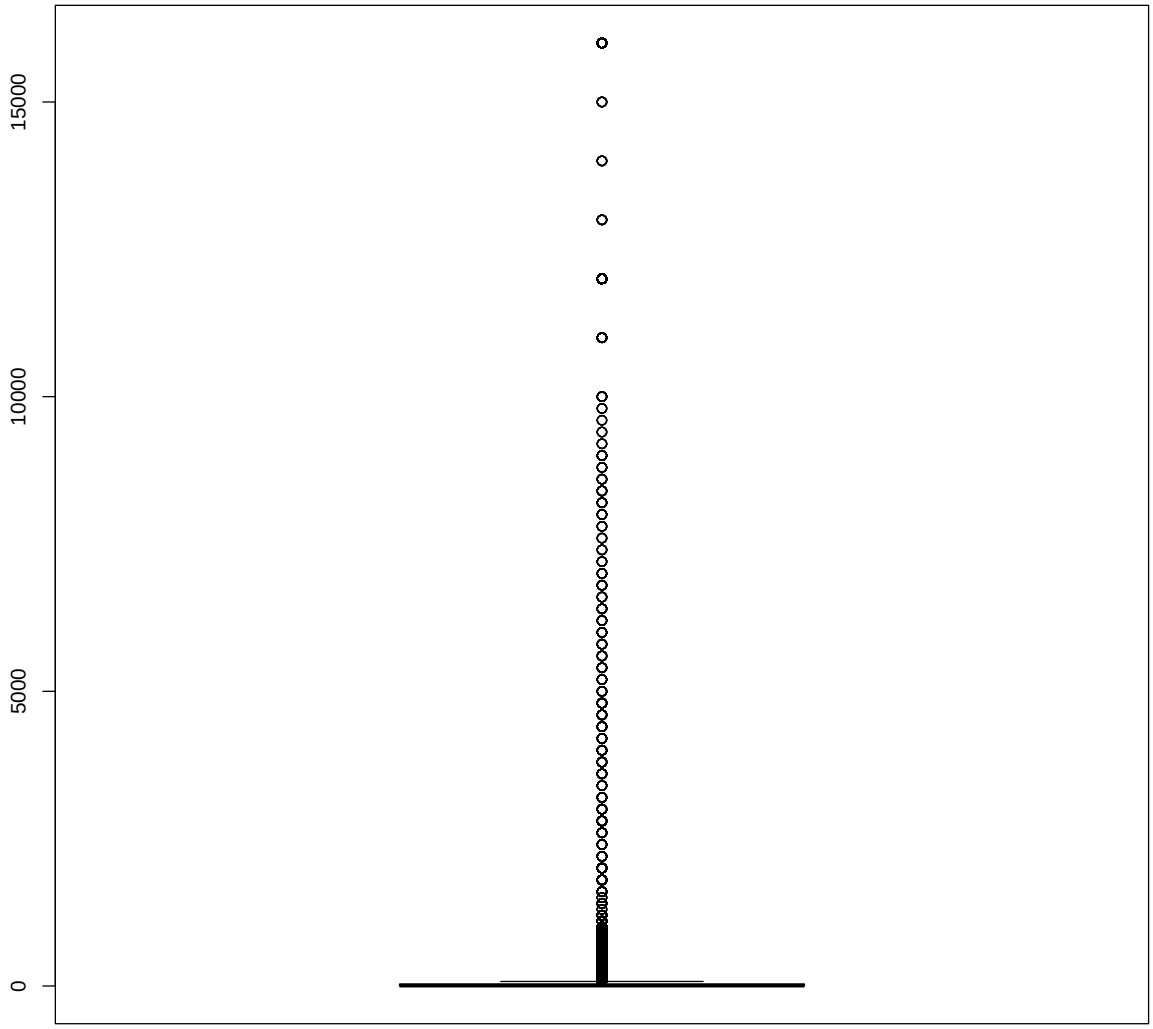
	metric	V2
1	Variable: contribution_amount	
2	Total N:	
3	Count	175
4	NA Count	0
5	Mean	595.996
6	Median	250
7	Standard Deviation	1735.362
8	Variance	3011480.237
9	Range	19950
10	Min	50
11	Max	20000
12	25th Percentile	150
13	75th Percentile	600

```
## 14 Subset w/o Outliers:
## 15                 Count                          172
## 16                     %                        98.3%
## 17             Outlier %                         1.7%
## 18              NA Count                            0
## 19                  Mean                      397.076
## 20                Median                          250
## 21   Standard Deviation                      366.845
## 22              Variance                   134575.544
## 23                 Range                         2450
## 24                   Min                           50
## 25                   Max                         2500
##                                V3
## 1  Variable: contribution_annual
## 2
## 3                             175
## 4                               0
## 5                         511.752
## 6                             250
## 7                        1129.197
## 8                     1275086.517
## 9                           10000
## 10                              0
## 11                          10000
## 12                            150
## 13                            600
## 14
## 15                            172
## 16                          98.3%
## 17                           1.7%
## 18                              0
## 19                        369.488
## 20                            250
## 21                        307.216
## 22                      94381.807
## 23                           1200
## 24                              0
## 25                           1200
```

```
train_x03_df03 <- subset(x = train_x03_df01, select = x03_lst02)
train_x03_df03 <- na.omit(train_x03_df03)
print(head(train_x03_df03))
```

```
##     value
## 1  220.00
## 7  200.00
## 9   15.01
## 10   7.69
## 11   8.10
## 12   1.70
```

```
box_comp(xcol = x03_lst02, df = subset(x = train_x03_df03, select = x03_lst02), rtn_met =
↪  TRUE)
```



	metric	V2
1		Variable: value
2	Total N:	
3	Count	70078
4	NA Count	0
5	Mean	216.197
6	Median	8.34
7	Standard Deviation	1480.169
8	Variance	2190900.281
9	Range	16000
10	Min	0
11	Max	16000
12	25th Percentile	2
13	75th Percentile	32

```
## 14 Subset w/o Outliers:
## 15                 Count           69069
## 16                     %           98.6%
## 17             Outlier %            1.4%
## 18              NA Count               0
## 19                  Mean          47.973
## 20                Median            8.24
## 21   Standard Deviation          252.85
## 22              Variance       63933.175
## 23                 Range            4600
## 24                   Min               0
## 25                   Max            4600
```

```
#box_comp(xcol = x01_lst03, df = subset(x = train_x01_df01, select = x01_lst03), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst04, df = subset(x = train_x01_df01, select = x01_lst04), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst05, df = subset(x = train_x01_df01, select = x01_lst05), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst06, df = subset(x = train_x01_df01, select = x01_lst06), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst07, df = subset(x = train_x01_df01, select = x01_lst07), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst08, df = subset(x = train_x01_df01, select = x01_lst08), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst09, df = subset(x = train_x01_df01, select = x01_lst09), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst10, df = subset(x = train_x01_df01, select = x01_lst10), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst11, df = subset(x = train_x01_df01, select = x01_lst11), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst12, df = subset(x = train_x01_df01, select = x01_lst12), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst13, df = subset(x = train_x01_df01, select = x01_lst13), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst14, df = subset(x = train_x01_df01, select = x01_lst14), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst15, df = subset(x = train_x01_df01, select = x01_lst15), rtn_met
↪  = FALSE)
#box_comp(xcol = x01_lst16, df = subset(x = train_x01_df01, select = x01_lst16), rtn_met
↪  = FALSE)
```

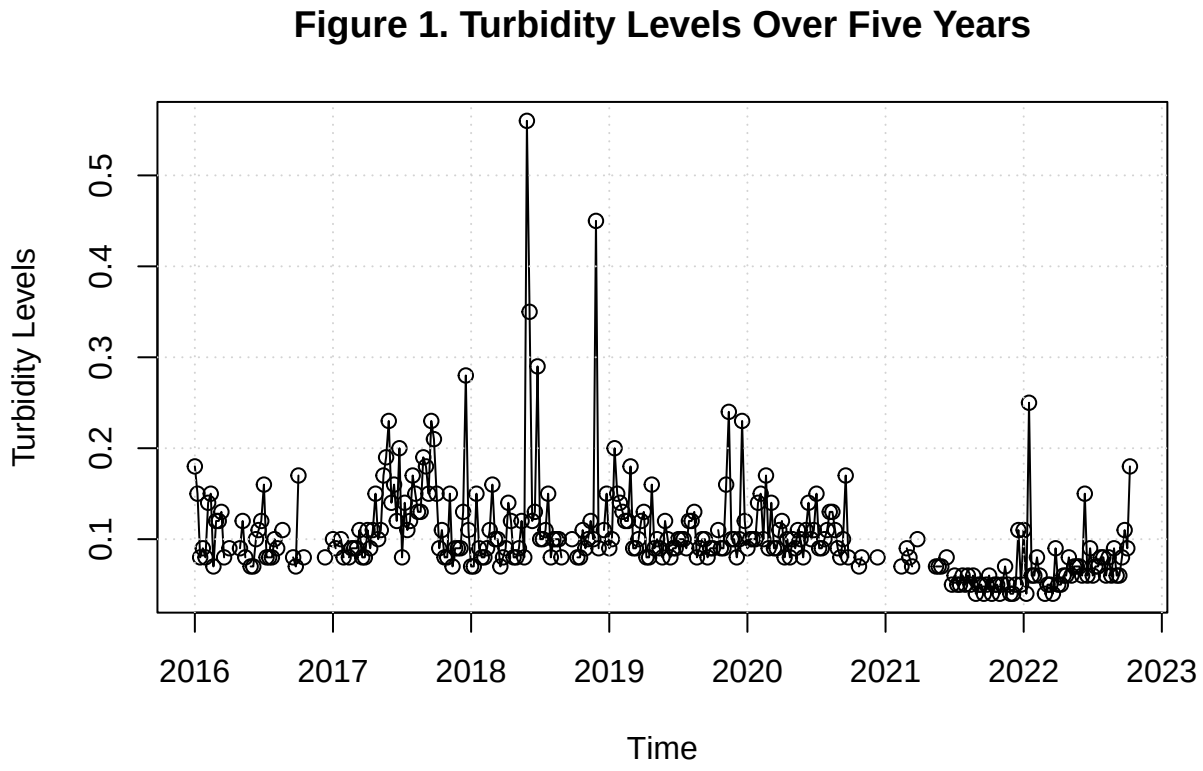
```
print(head(train_x01_df01_full))
```

```
## # A tibble: 6 x 4
## # Groups:   date_sample, sample_source [5]
##   date_sample sample_source analyte   Total
##   <chr>       <chr>         <chr>     <dbl>
## 1 2016-01-01  249A SYS      FLUORIDE  0.651
## 2 2016-01-02  281 SYS       FLUORIDE  0.666
## 3 2016-01-03  150 SYS       FLUORIDE  0.681
## 4 2016-01-04  259 SYS       FLUORIDE  0.675
## 5 2016-01-05  128 SYS       COLOR    NA
## 6 2016-01-05  128 SYS       TON       1
```

```r
train_x01_df01_full02 <- train_x01_df01_full[train_x01_df01_full$analyte == "TURBIDITY" &
↪  train_x01_df01_full$sample_source == "128 SYS", ]
aps_df01_ts01 <- ts(train_x01_df01_full02$Total, start = c(2016, 1), freq = 52)
#print(aps_df01_ts01)

#ship_fore_avg <- tslm(aps_df01_ts01 ~ trend)
#ship_fore_trnd <- tslm(aps_df01_ts01 ~ trend + I(trend^2))

plot(aps_df01_ts01,
     xlab = "Time",
     ylab = "Turbidity Levels",
     type = "o",
     main = "Figure 1. Turbidity Levels Over Five Years")
grid()
```




```r
print(head(train_x03_df01_full))
```

```
## # A tibble: 6 x 4
## # Groups:   date_sample, station [3]
##   date_sample station parameter Total
##   <chr>       <chr>   <chr>     <dbl>
## 1 2020-01-01  S11     ENTERO      220
## 2 2020-01-01  S11     FECAL        NA
## 3 2020-01-01  S11     TOTAL        NA
## 4 2020-01-01  S4      FECAL        NA
## 5 2020-01-01  S4      TOTAL        NA
```

```
## 6 2020-01-01  S6      ENTERO      200
```

```
print(tail(train_x03_df01_full))
```

```
## # A tibble: 6 x 4
## # Groups:   date_sample, station [2]
##   date_sample station parameter Total
##   <chr>       <chr>   <chr>     <dbl>
## 1 2021-12-29  D8-B    ENTERO       14
## 2 2021-12-29  D8-B    FECAL         2
## 3 2021-12-29  D8-B    TOTAL       200
## 4 2021-12-29  D9      ENTERO        4
## 5 2021-12-29  D9      FECAL         2
## 6 2021-12-29  D9      TOTAL        20
```

```
train_x03_df01_full02 <- train_x03_df01_full[train_x03_df01_full$parameter == "ENTERO" &
↪  train_x03_df01_full$station == "A1", ]
aps_df01_ts01 <- ts(train_x03_df01_full02$Total, start = c(2020, 1), freq = 52)
#, freq = 52
#print(aps_df01_ts01)

#ship_fore_avg <- tslm(aps_df01_ts01 ~ trend)
#ship_fore_trnd <- tslm(aps_df01_ts01 ~ trend + I(trend^2))

plot(aps_df01_ts01,
     xlab = "Time",
     ylab = "Entero Levels",
     type = "o",
     main = "Figure 1. Entero Levels Over Five Years")
grid()
```

Figure 1. Entero Levels Over Five Years