GT datafiles - Cleaning & Standardization

Jyothi

2025-03-09

```
#Loading the libraries
library(openxlsx)
library(dplyr)
library(janitor)
library(tidyverse)
library(ggplot2)
library(stringr)

#Reading both the datasets
gt_1 <- read.xlsx("Copy of G&T Results 2018-19 Responses - Sheet1.xlsx")
gt_2 <- read.xlsx("Copy of G&T Results 2017-18 (Responses) - Form Responses 1 (1).xlsx")
```

clean_names command helps in converting all column names into lower case, with underscores between words. It is part of the Janitor package. glimpse is an excellent command to view the column names and the first few values.

```
#Viewing the gt_1 dataset and cleaning the column names
gt_1_clean <- clean_names(gt_1)

glimpse(gt_1_clean)
```

```
## Rows: 100
## Columns: 14
## $ timestamp                  <chr> "42821", "43186", "43186", "43186", "43186"~
## $ entering_grade_level       <chr> "1", "1", "1", "1", "first", "1", "1", "1",~
## $ district                   <dbl> 13, 1, 2, 2, 3, 2, 2, 2, 2, 2, 2, 2, 3, 2, ~
## $ birth_month                <chr> "August", "March", "February", "January", "~
## $ olsat_verbal_score         <chr> "30", "26", "26", "25", "28", "25", "27", "~
## $ olsat_verbal_percentile    <dbl> 99, 99, 99, 99, 98, 99, 99, 99, 99, 98, 99,~
## $ nnat_non_verbal_raw_score  <chr> "46", "47", "47", "45", "45", NA, "43", "42~
## $ nnat_non_verbal_percentile <dbl> 99.00, 99.00, 99.00, 99.00, 0.00, 99.00, 99~
## $ overall_score              <dbl> 99, 99, 99, 99, 99, 99, 99, 99, 99, 99, 99,~
## $ school_preferences         <chr> "Anderson?", "1111", NA, "NEST+M", "NEST,An~
## $ school_assigned            <chr> NA, NA, NA, NA, "TAG", NA, NA, NA, "TAG", N~
## $ will_you_enroll_there      <chr> "at LL now, has a sib ent K w/ 99", NA, NA,~
## $ x13                        <chr> "N", NA, NA, "Yes", "home", NA, NA, NA, "at~
## $ x14                        <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, "Yes at~
```

as.date and as.numeric are in base R. as.numeric helps in converting the class to numeric form. as.date helps in changing the variable to the date format, using the excel date origin at 1899-12-30. head command shows the first ten rows of the data. select command is used to remove the timestamp variable from the dataset.

```
#changing "timestamp" variable into data format
gt_1_clean <- gt_1_clean %>%
   mutate(date = as.Date(as.numeric(timestamp), origin = "1899-12-30"))

head(gt_1_clean$date)
```

```
## [1] "2017-03-27" "2018-03-27" "2018-03-27" "2018-03-27" "2018-03-27"
## [6] "2018-03-27"
```

```
gt_1_clean <- gt_1_clean %>% select(-timestamp)
```

Using case_when from dplyr package, the values are standardized for grade variable. The dataset had inconsistent entries in terms of format. The variable is of character class.

```
#Standardizing grade levels
gt_1_clean <- gt_1_clean %>% mutate(entering_grade_level=
                                      case_when(
                                        entering_grade_level == "1" ~ "first",
                                        entering_grade_level == "2" ~ "second",
                                        entering_grade_level %in%  c("K","k") ~ "kindergarten",
                                        TRUE ~ entering_grade_level))
```

Using case_when from dplyr package, birth month is standardized. This column had inconsistent values with numeric and character formats.

```
##  [1] "August"    "March"     "February"  "January"   "May"       "July"
##  [7] "11"        "April"     "June"      "September" "October "  "December"
## [13] "November"  "February " "2"         "Feb"       "January "  NA
## [19] "Febrauary" "september" "12"
```

```
gt_1_clean <- gt_1_clean %>% mutate(birth_month=
                                      case_when(
                                        birth_month %in% c("Febrauary", "Feb", "2") ~ "February",
                                        birth_month == "september" ~ "September",
                                        birth_month == "11" ~ "November",
                                        birth_month == "12" ~ "December",
                                        TRUE ~ birth_month))
```

str command gives us the class of the variable. unique command shows the distinct values in the column. The olsat scores were entered in different formats. the highest possible score is 30. For all the values entered in the form of a fraction, the numerator is considered. For all the values entered in the form of decimal number, it is converted to whole number, by taking 30 as the highest possible number. The values are then converted to numeric class using as.numeric from base R.

In case of olsat percentile, geom_point from ggplot2 package is used to plot the values. seq_along helps in generating a standard numeric values on y axis. The plot helps in identifying any outliers in the column. Values with decimal points are converted to whole numbers by multiplying with 100. This decision is based on the trends observed in other values of the column.

```
##  chr [1:100] "30" "26" "26" "25" "28" "25" "27" "25" "25/30" "23" "25" "29" ...
```

```
##  [1] "30"    "26"    "25"    "28"    "27"    "25/30" "23"    "29"    "0.83"
## [10] "24"    NA      " "     "1"     "22"    "21"    "20"    "19"    "18"
## [19] "16"    "15"    "10"    "13"
```

```
gt_1_clean <- gt_1_clean %>% mutate(olsat_verbal_score=
                                      case_when(
                                        olsat_verbal_score =="25/30" ~ "25",
                                       olsat_verbal_score == "0.83" ~ "25",
                                        olsat_verbal_score == " " ~ "NA",
                                        TRUE ~ olsat_verbal_score))

gt_1_clean <- gt_1_clean %>% mutate(olsat_verbal_score = as.numeric(olsat_verbal_score))
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `olsat_verbal_score = as.numeric(olsat_verbal_score)`.
## Caused by warning:
## ! NAs introduced by coercion
```

```
str(gt_1_clean$olsat_verbal_percentile)
```

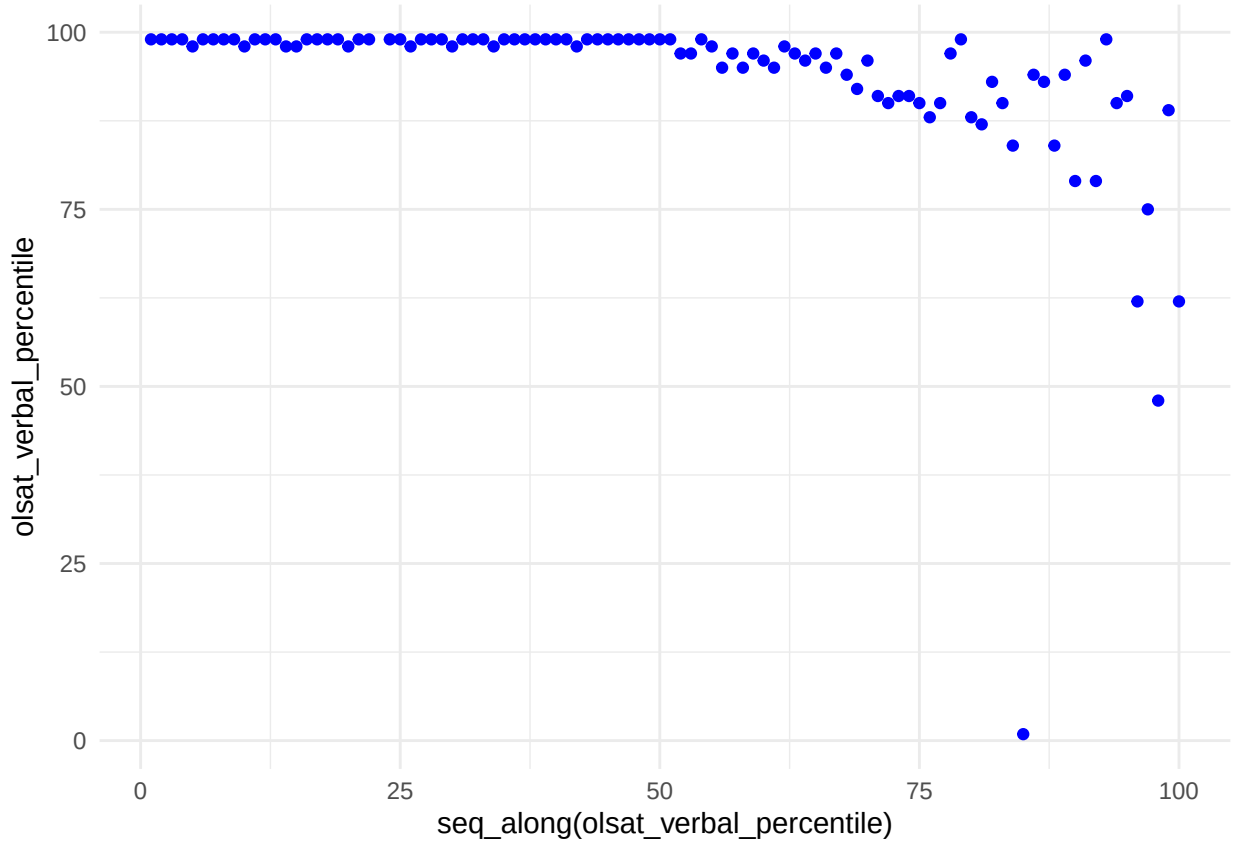
```
##  num [1:100] 99 99 99 99 98 99 99 99 99 98 ...
```

```
summary(gt_1_clean$olsat_verbal_percentile)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.91   93.50   98.00   93.68   99.00   99.00       1
```

```
ggplot(data=gt_1_clean, aes(x= seq_along(olsat_verbal_percentile), y= olsat_verbal_percentile))+
  geom_point(color="blue")+
  theme_minimal()
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_point()`).
```




```
gt_values <- gt_1_clean %>% filter(olsat_verbal_percentile < 10)
gt_values
```

```
##   entering_grade_level district birth_month olsat_verbal_score
## 1                first        2        June                 20
##   olsat_verbal_percentile nnat_non_verbal_raw_score nnat_non_verbal_percentile
## 1                    0.91                        34                         97
##   overall_score school_preferences school_assigned
## 1            95 Anderson/NEST only            <NA>
##                                    will_you_enroll_there                x13
## 1 Will not enroll if get PS 165/163. Maybe 166 but doubtful No. We should have!
##    x14       date
## 1 <NA> 2018-03-27
```

```
gt_1_clean <- gt_1_clean %>% mutate(olsat_verbal_percentile=
                                      case_when(olsat_verbal_percentile==0.91 ~ 91,
                                                TRUE ~ olsat_verbal_percentile))

gt_values <- gt_1_clean %>% filter(olsat_verbal_percentile < 10)
gt_values
```

```
##  [1] entering_grade_level       district
##  [3] birth_month                olsat_verbal_score
##  [5] olsat_verbal_percentile    nnat_non_verbal_raw_score
```

```
##  [7] nnat_non_verbal_percentile overall_score
##  [9] school_preferences         school_assigned
## [11] will_you_enroll_there      x13
## [13] x14                        date
## <0 rows> (or 0-length row.names)
```

str and unique commands are used to identify the class and the distinct values in the nnat scores variable. The fractional values are converted to whole numbers, with the assumption that the maximum possible score is 50. Once again, ggplot2 package is generate a scatter plot to view the outliers in the column. All the values are converted to whole numbers.

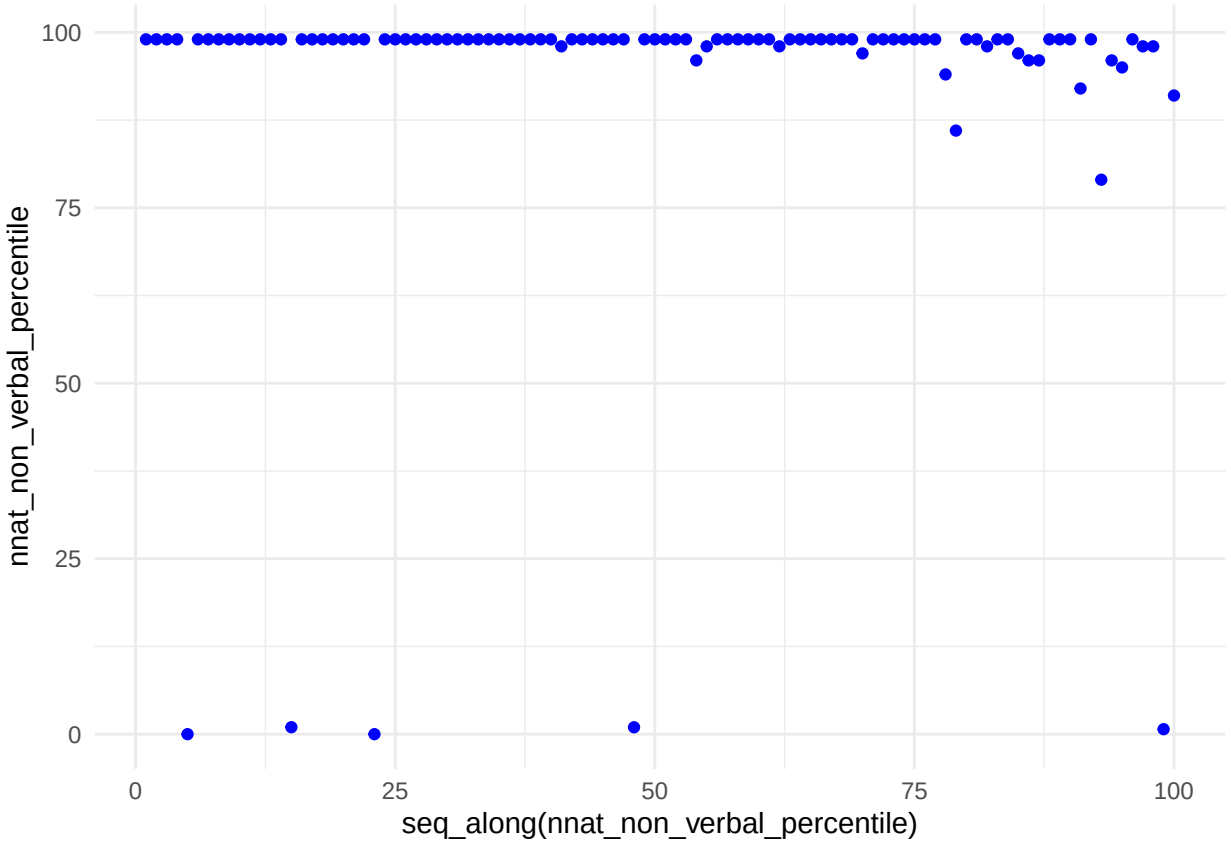
```
##  chr [1:100] "46" "47" "47" "45" "45" NA "43" "42" "40/50" "39" "38" "37" ...
```

```
##  [1] "46"    "47"    "45"    NA      "43"    "42"    "40/50" "39"    "38"
## [10] "37"    "36"    "40"    "41"    "44"    "4"     "48"    "34"    "31"
## [19] "33"    "35"    "26"    "32"    "24"    "29"
```

```
gt_1_clean <- gt_1_clean %>% mutate(nnat_non_verbal_raw_score = as.numeric(
                                      case_when(nnat_non_verbal_raw_score == "40/50" ~ "40",
                                                TRUE ~ nnat_non_verbal_raw_score)))


str(gt_1_clean$nnat_non_verbal_percentile)
```

```
##  num [1:100] 99 99 99 99 0 99 99 99 99 99 ...
```

```
ggplot(data = gt_1_clean, aes(x=seq_along(nnat_non_verbal_percentile), y= nnat_non_verbal_percentile))+
  geom_point(color = "blue")+
  theme_minimal()
```




```
gt_values <- gt_1_clean %>% filter(nnat_non_verbal_percentile < 10)
View(gt_values)

gt_1_clean <- gt_1_clean %>% mutate(nnat_non_verbal_percentile= ifelse(nnat_non_verbal_percentile < 10,
                                                                       nnat_non_verbal_percentile*100,
                                                                       nnat_non_verbal_percentile))
```

str and summary commands are used to verify the class and trends of the overall score variable.

```
##  num [1:100] 99 99 99 99 99 99 99 99 99 99 ...
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   79.00   96.00   99.00   97.18   99.00   99.00
```

str and unique are used to identify the class and the distinct values for the school preferences. The columns had inconsistent values . The goal was to streamline the values. I converted all the values that imply as not having any preference or unsure into - none. I then split the column into two - first preferred school, and other preferred schools. This categorization will help in identifying clearly the preferred school and alternatives option. It also helps to clearly group those subjects with no preference. I used ifelse and str functions to acheive these tasks. Head funstion helps in viewing the first few rows for the specific variable. The parent column is then deleted.

```
# Standardizing school preferences
```

```r
str(gt_1_clean$school_preferences)

unique(gt_1_clean$school_preferences)
```

```r
gt_1_clean <- gt_1_clean %>%
  mutate(school_preferences = ifelse(
    school_preferences %in% c(NA, ":-(", "Not sure", "No idea! ", "Likely none","None", "N/A stay distri
    "none",
    school_preferences
  ))
```

```r
# Checking unique values after replacement
unique(gt_1_clean$school_preferences)
```

```r
gt_1_clean1 <- gt_1_clean %>%
  mutate(
    school_preferences = str_trim(school_preferences),
    school_preferences = str_replace_all(school_preferences, "\\?", ""),
    school_preferences = str_replace_all(school_preferences, "/", ", "),
    first_choice = word(school_preferences, 1, sep = fixed(",")),
    remaining_choices = str_remove(school_preferences, paste0("^", first_choice, ",?\\s*"))
  )

# Viewing updated results
head(gt_1_clean1[, c("school_preferences", "first_choice", "remaining_choices")])
```

```
##           school_preferences first_choice     remaining_choices
## 1                   Anderson     Anderson
## 2                       1111         1111
## 3                       none         none
## 4                     NEST+M       NEST+M                NEST+M
## 5 NEST,Anderson,TAG,Q300,BSI         NEST Anderson,TAG,Q300,BSI
## 6                     NEST+M       NEST+M                NEST+M
```

```r
gt_1_clean1 <- gt_1_clean1 %>% mutate(remaining_choices=
                                        ifelse(remaining_choices == first_choice,"none",
                                               ifelse(remaining_choices == "", "none",
                                                      remaining_choices)))


head(gt_1_clean1[, c("school_preferences", "first_choice", "remaining_choices")])
```

```
##           school_preferences first_choice     remaining_choices
## 1                   Anderson     Anderson                  none
## 2                       1111         1111                  none
## 3                       none         none                  none
## 4                     NEST+M       NEST+M                  none
## 5 NEST,Anderson,TAG,Q300,BSI         NEST Anderson,TAG,Q300,BSI
## 6                     NEST+M       NEST+M                  none
```

```
gt_1_clean1 <- gt_1_clean1 %>% select(-school_preferences)
```

There are two unnamed columns in the dataset. The second column had only one values which seemed like an error, entered into the second column instead of the first column. Once again, the goal is to stream line and group the values accurately. All the values are grouped into yes, no and missing. The values are related to the test preparation. The column had more than 10% of the missing data. The data is left as it is as this is not considered as one of the primary variables of interest. It should be noted that the rows with missing values for this variable, also had missing values for other variables - enrollment and assigned school. This shows that the values are not missing by chance by clear lack of data collected for these subjects. The chi-square test shows a significant association between the missing values in the "school assigned" variable and the enrollment variable.The deeper analysis of the data collected and data sites is required to understand if data entry is missing from specifics data collection sites.

```
gt_1_clean1 <- gt_1_clean1 %>% mutate(x13= ifelse(!is.na(x14), x14, x13)) %>% select(-x14)


gt_1_clean1 <- gt_1_clean1 %>%
  mutate(
    testprep = case_when(
      str_detect(x13, regex("\\b(Yes|yes|Y|y)\\b", ignore_case = TRUE)) ~ "Yes",
      str_detect(x13, regex("\\b(No|no|N|n)\\b", ignore_case = TRUE)) ~ "No",
      str_detect(x13, regex("\\b(home|Home|testing|Testingmom.com|Minimally|mom)\\b", ignore_case = TRU
      str_detect(x13, regex("\\b(Missed)\\b", ignore_case = TRUE)) ~ "No",
      TRUE ~ x13
    )
  ) %>% select(-x13)
```

```
gt_1_clean1 <- gt_1_clean1 %>%
  mutate(enrollment = case_when(
    will_you_enroll_there %in% c("no", "No", "NO") ~ "No",
    is.na(will_you_enroll_there) ~ "",
    will_you_enroll_there == "Not sure" ~ "Unsure",
    TRUE ~ "Yes"
  )) %>% select(-will_you_enroll_there)

# Creating a contingency table
missing_table <- table(gt_1_clean1$enrollment, is.na(gt_1_clean1$school_assigned))

# Performing a Chi-Square Test
chisq_test <- chisq.test(missing_table)
print(chisq_test)
```

```
##
##  Pearson's Chi-squared test
##
## data:  missing_table
## X-squared = 20.192, df = 3, p-value = 0.0001549
```

```
missing_table <- table(gt_1_clean1$testprep, is.na(gt_1_clean1$enrollment))

# Performing a Chi-Square Test
chisq_test <- chisq.test(missing_table)
print(chisq_test)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  missing_table
## X-squared = 11.792, df = 1, p-value = 0.0005947
```

```
glimpse(gt_1_clean1)
```

```
## Rows: 100
## Columns: 14
## $ entering_grade_level       <chr> "first", "first", "first", "first", "first"~
## $ district                   <dbl> 13, 1, 2, 2, 3, 2, 2, 2, 2, 2, 2, 2, 3, 2, ~
## $ birth_month                <chr> "August", "March", "February", "January", "~
## $ olsat_verbal_score         <dbl> 30, 26, 26, 25, 28, 25, 27, 25, 25, 23, 25,~
## $ olsat_verbal_percentile    <dbl> 99, 99, 99, 99, 98, 99, 99, 99, 99, 98, 99,~
## $ nnat_non_verbal_raw_score  <dbl> 46, 47, 47, 45, 45, NA, 43, 42, 40, 39, 38,~
## $ nnat_non_verbal_percentile <dbl> 99, 99, 99, 99, 0, 99, 99, 99, 99, 99, 99, ~
## $ overall_score              <dbl> 99, 99, 99, 99, 99, 99, 99, 99, 99, 99, 99,~
## $ school_assigned            <chr> NA, NA, NA, NA, "TAG", NA, NA, NA, "TAG", N~
## $ date                       <date> 2017-03-27, 2018-03-27, 2018-03-27, 2018-0~
## $ first_choice               <chr> "Anderson", "1111", "none", "NEST+M", "NEST~
## $ remaining_choices          <chr> "none", "none", "none", "none", "Anderson,T~
## $ testprep                   <chr> "No", NA, NA, "Yes", "Yes", NA, NA, NA, "Ye~
## $ enrollment                 <chr> "Yes", "", "", "Yes", "", "", "", "", "Yes"~
```

Similar strategies were used in cleanign and standardizing the variabel sin the gt_2 dataset, before combining both the datasets. The gt_2 dataset did not have the test preparation column. The rows from this dataset will not have any value for this variable.

```
#Standardizing the column names for gt_2 dataset

gt_2_clean <- clean_names(gt_2)

glimpse(gt_2_clean)
```

```
## Rows: 117
## Columns: 12
## $ timestamp                  <dbl> 42833.28, 42832.44, 42832.45, 42832.45, 428~
## $ entering_grade_level       <chr> "1", "K", "1", "K", "K", "K", "K", "K", "fi~
## $ district                   <chr> "6", NA, NA, NA, "22", NA, "Anderson", NA, ~
## $ birth_month                <chr> "September", "August", "March", "September"~
## $ olsat_verbal_score         <chr> "28/30", "25", "27", "23", "2", "24", "26",~
## $ olsat_verbal_percentile    <chr> "99", "99", "96", "97", "98", "97", "99", "~
## $ nnat_non_verbal_raw_score  <chr> "45/50", "39", "42", "40", "38", "36", "42"~
## $ nnat_non_verbal_percentile <dbl> 99.00, 99.00, 99.00, 99.00, 99.00, 98.00, 9~
## $ overall_score              <dbl> 99, 99, 98, 98, 99, 0, 99, 99, 95, 99, 94, ~
## $ school_preferences         <chr> "NEST+m, TAG, Anderson, Q300", "Anderson, N~
## $ school_assigned            <chr> "NEST", NA, NA, NA, "Currently - local Broo~
## $ will_you_enroll_there      <chr> "YES", "Maybe", "Maybe", NA, "Maybe", NA, N~
```

```
#Changing the timestamp variable to date format
str(gt_2_clean$timestamp)
```

```
##  num [1:117] 42833 42832 42832 42832 42835 ...
```

```
gt_2_clean <- gt_2_clean %>%
  mutate(date = as.Date(timestamp, origin = "1899-12-30")) %>% select(-timestamp)

head(gt_2_clean$date)
```

```
## [1] "2017-04-08" "2017-04-07" "2017-04-07" "2017-04-07" "2017-04-10"
## [6] "2017-04-07"
```

```
#Standardizing the grade levels
gt_2_clean <- gt_2_clean %>% mutate(entering_grade_level=
                                      case_when(
                                        entering_grade_level == "1" ~ "first",
                                        entering_grade_level == "2" ~ "second",
                                        entering_grade_level %in%  c("K","k","kinder") ~ "kindergarten"
                                        entering_grade_level == "3" ~ "third",
                                        TRUE ~ entering_grade_level))
```

```
#Standardizing the birth month

gt_2_clean <- gt_2_clean %>% mutate(birth_month=
                                      case_when(
                                        birth_month %in% c("Febrauary", "Feb", "2") ~ "February",
                                        birth_month == "8" ~ "August",
                                        birth_month == "11" ~ "November",
                                        birth_month == "12" ~ "December",
                                        TRUE ~ birth_month))
```

```
#Standardizing olsat scores and percentiles

str(gt_2_clean$olsat_verbal_score)
```

```
##  chr [1:117] "28/30" "25" "27" "23" "2" "24" "26" "24" "23" "29" "17" "23" ...
```

```
gt_2_clean <- gt_2_clean %>% mutate(olsat_verbal_score=
                                      case_when(
                                        olsat_verbal_score =="28/30" ~ "28",
                                        olsat_verbal_score == "19/30" ~ "19",
                                        olsat_verbal_score == "23/30" ~ "23",
                                        olsat_verbal_score == "0.83" ~ "25",
                                        olsat_verbal_score %in% c("", "**","---","-","Fill out later.")
                                        TRUE ~ olsat_verbal_score))

gt_2_clean <- gt_2_clean %>% mutate(olsat_verbal_score = as.numeric(olsat_verbal_score))
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `olsat_verbal_score = as.numeric(olsat_verbal_score)`.
## Caused by warning:
## ! NAs introduced by coercion
```

```r
summary(gt_2_clean$olsat_verbal_score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.97   23.00   26.00   27.75   28.00   99.00       8
```

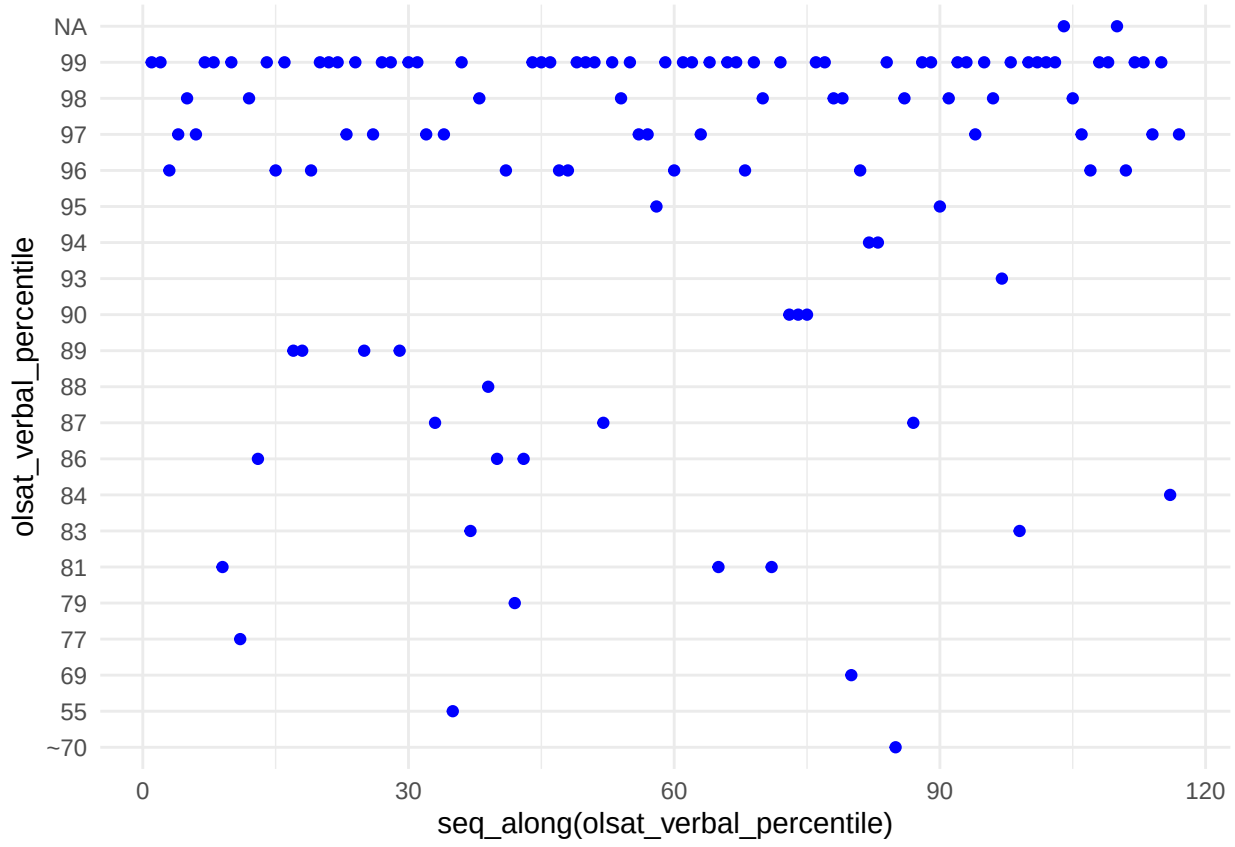
```r
gt_2_clean <- gt_2_clean %>% mutate(olsat_verbal_score=
                                      case_when(
                                        olsat_verbal_score < 10 | olsat_verbal_score >30  ~ NA,
                                        TRUE ~ olsat_verbal_score
                                      ))


str(gt_2_clean$olsat_verbal_percentile)
```

```
##  chr [1:117] "99" "99" "96" "97" "98" "97" "99" "99" "81" "99" "77" "98" ...
```

```r
summary(gt_2_clean$olsat_verbal_percentile)
```

```
##    Length     Class      Mode
##       117 character character
```

```r
ggplot(data=gt_2_clean, aes(x= seq_along(olsat_verbal_percentile), y= olsat_verbal_percentile))+
  geom_point(color="blue")+
  theme_minimal()
```



```
gt_values <- gt_2_clean %>% filter(olsat_verbal_percentile < 10)
gt_values
```

```
##   entering_grade_level district birth_month olsat_verbal_score
## 1         kindergarten       25      August                 NA
##   olsat_verbal_percentile nnat_non_verbal_raw_score nnat_non_verbal_percentile
## 1                     ~70           Fill out later.                         99
##   overall_score school_preferences school_assigned will_you_enroll_there
## 1            93               <NA>            <NA>                  <NA>
##         date
## 1 2017-05-08
```

```
gt_2_clean <- gt_2_clean %>% mutate(olsat_verbal_percentile=
                                      case_when(olsat_verbal_percentile== "~70" ~ "70",
                                                TRUE ~ olsat_verbal_percentile),olsat_verbal_percentile
```

```
#Standardizing nnat scores

str(gt_2_clean$nnat_non_verbal_raw_score)
```

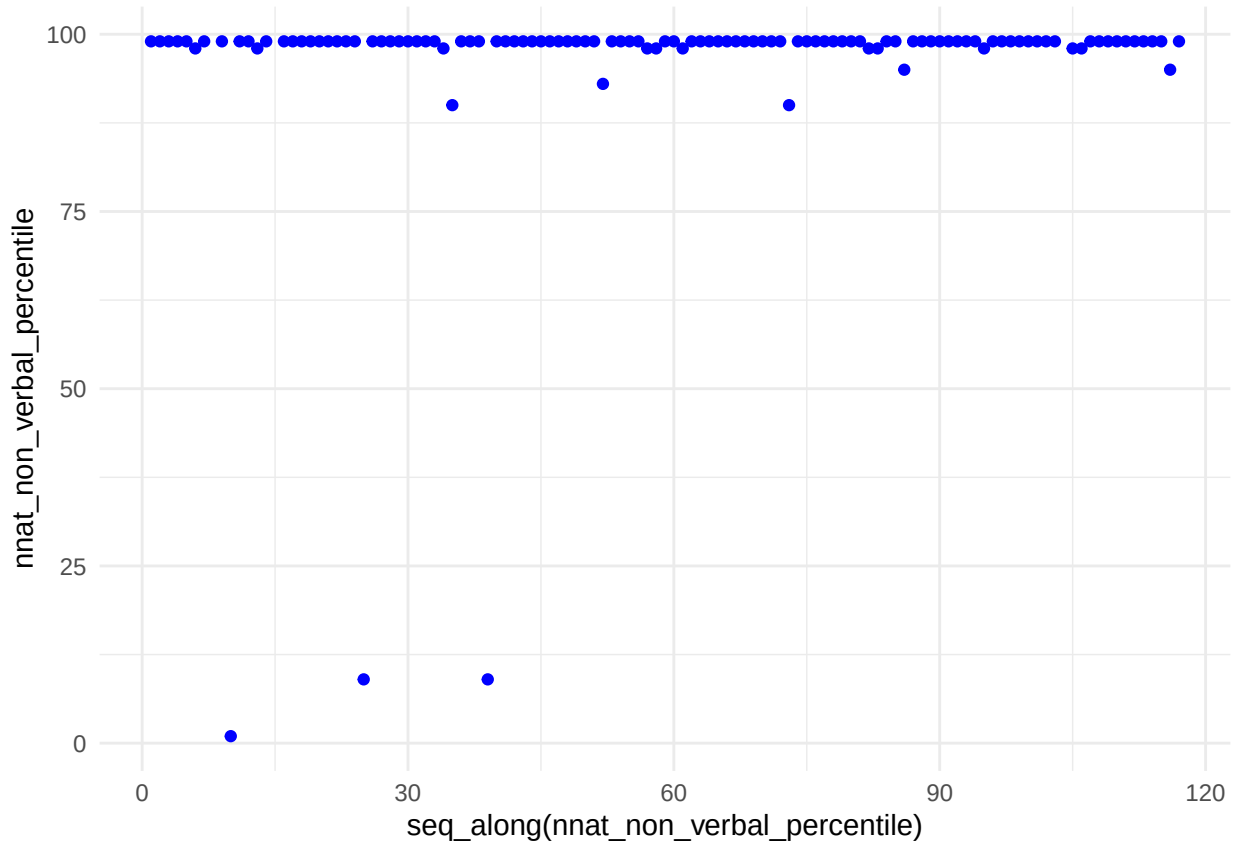
```
##  chr [1:117] "45/50" "39" "42" "40" "38" "36" "42" "42" "42" "44" "39" "45" ...
```

```
gt_2_clean <- gt_2_clean %>% mutate(nnat_non_verbal_raw_score = as.numeric(
  case_when(nnat_non_verbal_raw_score == "45/50" ~ "45",
            nnat_non_verbal_raw_score == "39/48" ~ "39",
            nnat_non_verbal_raw_score == "36/48" ~ "36",
            nnat_non_verbal_raw_score == "41/48" ~ "41",
            nnat_non_verbal_raw_score == "40/48" ~ "40",
            nnat_non_verbal_raw_score %in% c("-","Fill out later.","**","---") ~ "",
            nnat_non_verbal_raw_score >50 ~ "",
            TRUE ~ nnat_non_verbal_raw_score)),nnat_non_verbal_raw_score= as.numeric(nnat_non_verbal_ra


str(gt_2_clean$nnat_non_verbal_percentile)
```

```
##  num [1:117] 99 99 99 99 99 98 99 NA 99 0.99 ...
```

```
ggplot(data = gt_2_clean, aes(x=seq_along(nnat_non_verbal_percentile), y= nnat_non_verbal_percentile))+
  geom_point(color = "blue")+
  theme_minimal()
```

```
## Warning: Removed 3 rows containing missing values or values outside the scale range
## (`geom_point()`).
```




```
gt_values <- gt_2_clean %>% filter(nnat_non_verbal_percentile < 10)
View(gt_values)

gt_2_clean <- gt_2_clean %>% mutate(nnat_non_verbal_percentile= case_when(nnat_non_verbal_percentile <
                                                                          nnat_non_verbal_percentile == 9.
TRUE ~ nnat_non_verbal_percentile))
```

```
#Checking the overall score values
str(gt_2_clean$overall_score)
```

```
##  num [1:117] 99 99 98 98 99 0 99 99 95 99 ...
```

```
summary(gt_2_clean$overall_score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   97.00   99.00   95.07   99.00   99.00
```

```
#Standardizing school preferences

str(gt_2_clean$school_preferences)
```

```
##  chr [1:117] "NEST+m, TAG, Anderson, Q300" "Anderson, NEST+m" NA NA ...
```

```
gt_2_clean <- gt_2_clean %>%
  mutate(school_preferences = ifelse(
    school_preferences %in% c(NA,"na", "All CW", ":-(","Likely staying in zoned schools; D15 G&T options
                              "Not sure", "No idea! ", "Likely none","None", "N/A stay district g&t ",
                              "Brooklyn School of Inquiry" , "Any citywide or district 3 school", "Any
    "none",
    school_preferences
  ))

# Check unique values after replacement


gt_2_clean1 <- gt_2_clean %>%
  mutate(
    school_preferences = str_replace_all(str_trim(school_preferences), "/", ", "),
    first_choice = word(school_preferences, 1, sep = fixed(",")),
    remaining_choices = str_remove(school_preferences, paste0("\\b", fixed(first_choice), "\\b,?\\s*"))
    remaining_choices = trimws(mapply(gsub, first_choice, "", remaining_choices, MoreArgs = list(fixed =
    remaining_choices = ifelse(remaining_choices %in% c("", first_choice), "none", remaining_choices),
    remaining_choices = str_remove(remaining_choices, "^,")
  )

gt_2_clean1 <- gt_2_clean1 %>% select(-school_preferences)
```

```
#Standardizing enrollment options

gt_2_clean1 <- gt_2_clean1 %>%
  mutate(enrollment = case_when(
    will_you_enroll_there %in% c("no", "No", "NO") ~ "No",
    is.na(will_you_enroll_there) ~ "",
    will_you_enroll_there == "Maybe" ~ "Unsure",
    TRUE ~ "Yes"
  )) %>% select(-will_you_enroll_there)


glimpse(gt_2_clean1)
```

```
## Rows: 117
## Columns: 13
## $ entering_grade_level       <chr> "first", "kindergarten", "first", "kinderga~
## $ district                   <chr> "6", NA, NA, NA, "22", NA, "Anderson", NA, ~
## $ birth_month                <chr> "September", "August", "March", "September"~
## $ olsat_verbal_score         <dbl> 28, 25, 27, 23, NA, 24, 26, 24, 23, 29, 17,~
## $ olsat_verbal_percentile    <dbl> 99, 99, 96, 97, 98, 97, 99, 99, 81, 99, 77,~
## $ nnat_non_verbal_raw_score  <dbl> 45, 39, 42, 40, 38, 36, 42, 42, 42, 44, 39,~
## $ nnat_non_verbal_percentile <dbl> 99, 99, 99, 99, 99, 98, 99, NA, 99, 99, 99,~
## $ overall_score              <dbl> 99, 99, 98, 98, 99, 0, 99, 99, 95, 99, 94, ~
## $ school_assigned            <chr> "NEST", NA, NA, NA, "Currently - local Broo~
## $ date                       <date> 2017-04-08, 2017-04-07, 2017-04-07, 2017-0~
## $ first_choice               <chr> "NEST+m", "Anderson", "none", "none", "none~
## $ remaining_choices          <chr> " TAG, Anderson, Q300", "none", "none", "no~
## $ enrollment                 <chr> "Yes", "Unsure", "Unsure", "", "Unsure", ""~
```

```r
#Changing district variable to numeric class
gt_2_clean1 <- gt_2_clean1 %>% mutate(district=as.numeric(district))

#Combining the gt_1 and gt_2 datasets

combined_gt <- bind_rows(gt_1_clean1, gt_2_clean1)
write.csv(combined_gt, "combined_gt.csv", row.names = FALSE)
```