Toronto Apartment Analysis Report

Hsin Fang Hu

2023-02-17

Toronto's evaluation results for RentSafeTO registered apartment buildings enhance access to information for existing and prospective tenants, promoting transparency and accountability in the housing market. Our statistical analysis report provides a concise understanding of the dataset, enabling tenants to make informed rental choices by comprehending safety scores across different areas. It also assists building owners in identifying construction benchmarks and alerts law enforcement officers to areas requiring attention. By bridging the dataset with its implications, our report empowers stakeholders, fosters accountability, and contributes to a safer housing landscape through informed decision-making and improved transparency in Toronto.

Data Import

Read & Filter

```
library(tidyverse)
Apdata <- read_csv("./apartments_toronto.csv")
str(Apdata)
Apdata2 <- Apdata %>% filter(WARD %in% c("6"))
str(Apdata2)
```

Using the str() function, we can see that this dataset has a total of 40 columns and 11,651 rows. And there are three data types:

- Character(5): property_type, wardname, site_address, results_of_score, grid
- Date(1): evaluation_completed_on
- Numeric(34):remaining variables

In order to conduct a more detailed analysis, we filter out *York Center* from the ward column, and explore this area first.

Dealing with NA data

```
anyNA(Apdata2)
```

```
## [1] TRUE
```

```
nrow(Apdata2[!complete.cases(Apdata2),])
```

```
## [1] 743
```

```
Apdata2[!complete.cases(Apdata2),]
```

```
## # A tibble: 743 × 40
##      `_id`     RSN YEAR_…¹ YEAR_…² YEAR_…³ PROPE…⁴  WARD WARDN…⁵ SITE_…⁶ CONFI…⁷
##      <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <chr>   <dbl> <chr>   <chr>     <dbl>
##  1 2417935 4154459    2017      NA    1966 PRIVATE     6 York C… 41-51 …       4
##  2 2417957 4154439      NA      NA    1954 PRIVATE     6 York C… 2988 K…       3
##  3 2417972 4154438    2018      NA    1952 PRIVATE     6 York C… 2990 K…       3
##  4 2417973 4154377    2017      NA    1960 PRIVATE     6 York C… 933-93…       3
##  5 2417983 4154378    2017      NA    1960 PRIVATE     6 York C… 923-92…       3
##  6 2418257 4154602    2017      NA    1950 PRIVATE     6 York C… 4110 B…       3
##  7 2418258 4154601    2017      NA    1955 PRIVATE     6 York C… 4114 B…       4
##  8 2418259 4154606    2017      NA    1950 PRIVATE     6 York C… 4234 B…       3
##  9 2418308 4154605    2017      NA    1956 PRIVATE     6 York C… 4238 B…       3
## 10 2418350 4154600    2017      NA    1960 PRIVATE     6 York C… 3908 B…       4
## # … with 733 more rows, 30 more variables: CONFIRMED_UNITS <dbl>,
## #   EVALUATION_COMPLETED_ON <date>, SCORE <dbl>, RESULTS_OF_SCORE <chr>,
## #   NO_OF_AREAS_EVALUATED <dbl>, ENTRANCE_LOBBY <dbl>,
## #   ENTRANCE_DOORS_WINDOWS <dbl>, SECURITY <dbl>, STAIRWELLS <dbl>,
## #   LAUNDRY_ROOMS <dbl>, INTERNAL_GUARDS_HANDRAILS <dbl>,
## #   GARBAGE_CHUTE_ROOMS <dbl>, GARBAGE_BIN_STORAGE_AREA <dbl>, ELEVATORS <dbl>,
## #   STORAGE_AREAS_LOCKERS <dbl>, INTERIOR_WALL_CEILING_FLOOR <dbl>, …
```

First, we know that the data has missing values through the anyNA() function. Then through the nrow() function, we know that there are 743 rows in which the data is missing. If we directly print out the table with missing values, there will be too much data. Therefore, by calculating the missing rate of each variable, we will know the overall situation of the data.

```
colMeans(is.na(Apdata2))
```

```
##                         _id                         RSN
##                 0.000000000                 0.000000000
##             YEAR_REGISTERED               YEAR_EVALUATED
##                 0.037084399                 0.121483376
##                  YEAR_BUILT               PROPERTY_TYPE
##                 0.000000000                 0.000000000
##                        WARD                    WARDNAME
##                 0.000000000                 0.000000000
##                SITE_ADDRESS           CONFIRMED_STOREYS
##                 0.000000000                 0.000000000
##             CONFIRMED_UNITS     EVALUATION_COMPLETED_ON
##                 0.000000000                 0.000000000
##                       SCORE             RESULTS_OF_SCORE
##                 0.000000000                 0.000000000
##       NO_OF_AREAS_EVALUATED               ENTRANCE_LOBBY
##                 0.000000000                 0.000000000
##      ENTRANCE_DOORS_WINDOWS                    SECURITY
##                 0.000000000                 0.000000000
##                  STAIRWELLS               LAUNDRY_ROOMS
##                 0.000000000                 0.024296675
##   INTERNAL_GUARDS_HANDRAILS         GARBAGE_CHUTE_ROOMS
##                 0.000000000                 0.585677749
##    GARBAGE_BIN_STORAGE_AREA                   ELEVATORS
##                 0.001278772                 0.478260870
##       STORAGE_AREAS_LOCKERS INTERIOR_WALL_CEILING_FLOOR
##                 0.517902813                 0.000000000
##    INTERIOR_LIGHTING_LEVELS                    GRAFFITI
##                 0.000000000                 0.000000000
##            EXTERIOR_CLADDING            EXTERIOR_GROUNDS
##                 0.000000000                 0.000000000
##            EXTERIOR_WALKWAYS              BALCONY_GUARDS
##                 0.000000000                 0.346547315
## WATER_PEN_EXT_BLDG_ELEMENTS                PARKING_AREA
##                 0.000000000                 0.003836317
##             OTHER_FACILITIES                        GRID
##                 0.860613811                 0.000000000
##                    LATITUDE                   LONGITUDE
##                 0.008951407                 0.008951407
##                            X                           Y
##                 0.007672634                 0.007672634
```

Missing data can be divided into 3 categories:
1. Year registered & Year evaluated: The displayed information is the year when the building was first registered and the year of the building evaluation scores. The missing building information is all built before 1970, so it may be too old to be recorded.
2. Facilities:Some buildings may not contain these facilities, such as car parks, elevators, laundry rooms, etc., so there is no registration information.
3. Latitude, Longitude, X, Y: These variables represent the location of buildings, perhaps because the orientation of these buildings is difficult to locate or cut, resulting in missing values.

Exploring the dataset

Handling dates

```
str(Apdata2$EVALUATION_COMPLETED_ON)
```

```
##  Date[1:782], format: "2022-12-18" "2022-12-15" "2022-12-15" "2022-12-15" "2022-12
-14" ...
```

```
birthmonth <- Apdata2 %>%
  mutate(month = month(EVALUATION_COMPLETED_ON)) %>%
  filter(month %in% c(4)) %>%
  group_by(month)
birthmonth
```

```
## # A tibble: 26 × 41
## # Groups:   month [1]
##      `_id`      RSN YEAR_…¹ YEAR_…² YEAR_…³ PROPE…⁴  WARD WARDN…⁵ SITE_…⁶ CONFI…⁷
##      <dbl>    <dbl>   <dbl>   <dbl>   <dbl> <chr>   <dbl> <chr>   <chr>     <dbl>
##  1 2420762 4154602    2017    2021    1950 PRIVATE     6 York C… 4110 B…       3
##  2 2420778 4154639    2017    2021    1969 PRIVATE     6 York C… 601 FI…      14
##  3 2420788 4154611    2017    2021    1989 TCHC        6 York C… 2 FAYW…       3
##  4 2420793 4154436    2017    2021    1970 PRIVATE     6 York C… 2425 J…      20
##  5 2420798 4154408    2017    2021    1996 SOCIAL…     6 York C… 1206 W…       5
##  6 2420822 4154700    2017    2021    1980 TCHC        6 York C… 4455 B…      14
##  7 2420823 4154723    2017    2021    1972 TCHC        6 York C… 6250 B…      14
##  8 2420833 4154427    2018    2021    1968 PRIVATE     6 York C… 2900 K…       3
##  9 2420836 4227299    2017    2021    1962 PRIVATE     6 York C… 4100 B…       5
## 10 2420837 4154603    2017    2021    1964 PRIVATE     6 York C… 4160 B…       7
## # … with 16 more rows, 31 more variables: CONFIRMED_UNITS <dbl>,
## #   EVALUATION_COMPLETED_ON <date>, SCORE <dbl>, RESULTS_OF_SCORE <chr>,
## #   NO_OF_AREAS_EVALUATED <dbl>, ENTRANCE_LOBBY <dbl>,
## #   ENTRANCE_DOORS_WINDOWS <dbl>, SECURITY <dbl>, STAIRWELLS <dbl>,
## #   LAUNDRY_ROOMS <dbl>, INTERNAL_GUARDS_HANDRAILS <dbl>,
## #   GARBAGE_CHUTE_ROOMS <dbl>, GARBAGE_BIN_STORAGE_AREA <dbl>, ELEVATORS <dbl>,
## #   STORAGE_AREAS_LOCKERS <dbl>, INTERIOR_WALL_CEILING_FLOOR <dbl>, …
```

From the function of str() above, we can see that only the variable “evaluation_completed_on” is in date format. So we can filter further.
We take out the value of the evaluation completed in April every year, which is my birth month, and we can see that 26 buildings are evaluated in April.

Building observation

```
str(Apdata2$WARD)
```

```
##  num [1:782] 6 6 6 6 6 6 6 6 6 6 ...
```

In str() above, we can see that the variable “Ward” is of numeric type. But according to the meaning of the data, it should be changed to character type. Because it is a categorical variable.

```
median(Apdata2$CONFIRMED_STOREYS)
```

```
## [1] 4
```

```
mean(Apdata2$CONFIRMED_STOREYS)
```

```
## [1] 6.189258
```

Regarding the variable “confirmed stores”, the median number is 4 and the mean number is 6.19. The reason behind this is that the median is less sensitive to extreme values, while the mean is more affected by them.

```
Apdata2 %>%
    group_by(RESULTS_OF_SCORE) %>%
    summarise(percent = 100 * n() / nrow(Apdata2))
```

```
## # A tibble: 4 × 2
##   RESULTS_OF_SCORE                              percent
##   <chr>                                           <dbl>
## 1 Building Audit                                   1.28
## 2 Evaluation needs to be conducted in 1 year      23.9
## 3 Evaluation needs to be conducted in 2 years     66.4
## 4 Evaluation needs to be conducted in 3 years      8.44
```

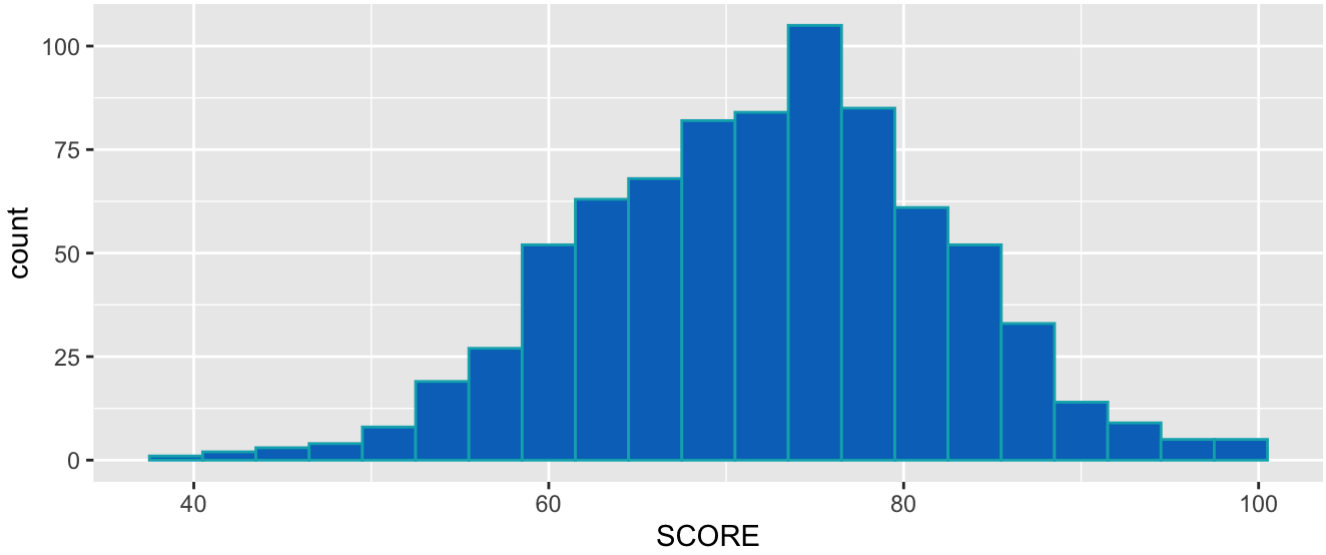
```
oldest <- Apdata2 %>% filter(YEAR_BUILT %in% c("1945"))
mean(oldest$SCORE)
```

```
## [1] 67.83333
```

8.44% of the buildings in York Center received a result of “Evaluation needs to be conducted in 3 years”.Through the data, we can find that the building in 1945 is the oldest. For buildings built in this year, the overall evaluation score was 67.83.

Scoring results observation

```
scoreplot <- ggplot(data=Apdata2)+ geom_histogram(aes(x=SCORE),
                                  binwidth = 3,
                                  fill="#0073C2FF",
                                  color="#00AFBB")
scoreplot
```



In this Score histogram, we can see that the variables are bell-shaped and the frequency distribution is symmetrical. Data tend to cluster around the mean, with relatively few tail observations.

```
scoreplot + facet_wrap(~RESULTS_OF_SCORE)+ theme(strip.background = element_rect(fill
="orange"))
```

As can be seen from this chart, buildings with scores below 50 are subject to comprehensive examination.As the score increases, the years required for reassessment can last longer.Buildings with higher assessment scores can receive another assessment later.

Score of Prorerty Type

```
average_scores <- Apdata2 %>%
  group_by(PROPERTY_TYPE) %>%
  summarize(avg_score = mean(SCORE, na.rm = TRUE))
average_scores
```

```
## # A tibble: 3 × 2
##   PROPERTY_TYPE  avg_score
##   <chr>              <dbl>
## 1 PRIVATE             72.2
## 2 SOCIAL HOUSING      73.9
## 3 TCHC                72.6
```

```
library(ggplot2)
ggplot(Apdata2, aes(x = PROPERTY_TYPE, y = SCORE)) +
  geom_boxplot() +
  xlab("PROPERTY_TYPE") +
  ylab("SCORE") +
  ggtitle("Boxplot of SCORE by PROPERTY_TYPE")
```

We can see that the average scores of the three buildings which are private, by Toronto Community Housing Corporation (TCHC) or another assisted, social or supportive housing provider, are comparable. Only the ratings of private buildings are scattered.

Different variable's average score

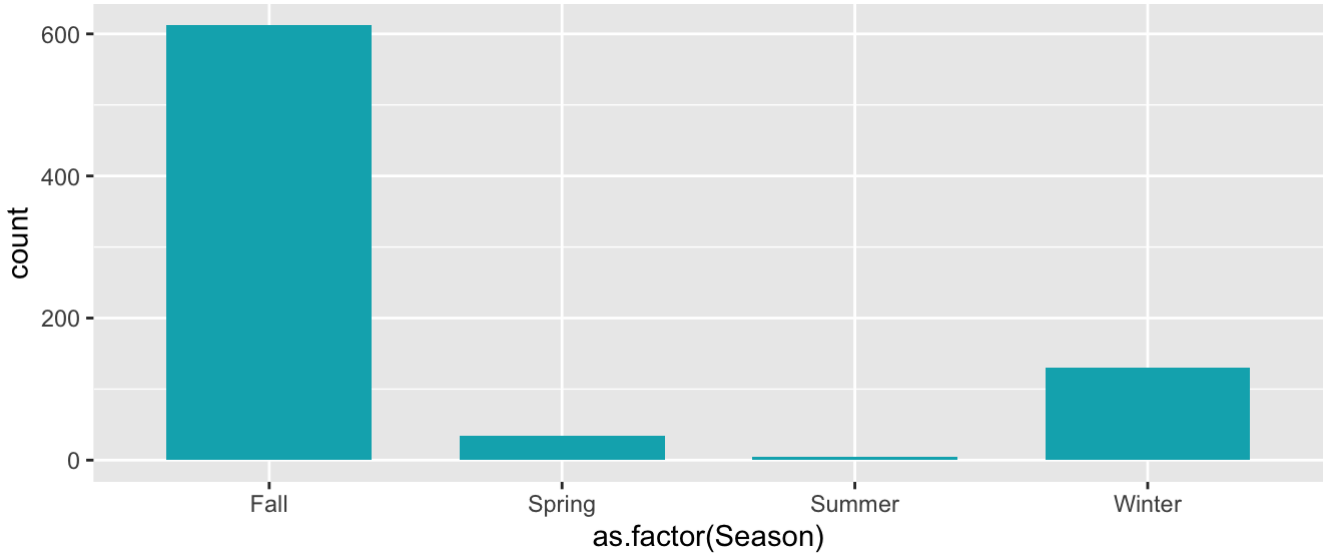
```
selected_vars <- c("ENTRANCE_LOBBY", "ENTRANCE_DOORS_WINDOWS", "SECURITY", "STAIRWELL
S", "LAUNDRY_ROOMS","INTERNAL_GUARDS_HANDRAILS", "GARBAGE_CHUTE_ROOMS", "GARBAGE_BIN_
STORAGE_AREA", "ELEVATORS","STORAGE_AREAS_LOCKERS", "INTERIOR_WALL_CEILING_FLOOR", "I
NTERIOR_LIGHTING_LEVELS", "GRAFFITI","EXTERIOR_CLADDING", "EXTERIOR_GROUNDS", "EXTERI
OR_WALKWAYS", "BALCONY_GUARDS","WATER_PEN_EXT_BLDG_ELEMENTS", "PARKING_AREA", "OTHER_
FACILITIES")

average_scores <- Apdata2 %>%
  group_by(PROPERTY_TYPE) %>%
  summarise(across(all_of(selected_vars), mean, na.rm = TRUE)) %>%
  pivot_longer(cols = -PROPERTY_TYPE, names_to = "Variable", values_to = "AverageScor
e") %>%
  pivot_wider(names_from = PROPERTY_TYPE, values_from = AverageScore)
average_scores
```

```
## # A tibble: 20 × 4
##    Variable                    PRIVATE `SOCIAL HOUSING`  TCHC
##    <chr>                         <dbl>            <dbl> <dbl>
##  1 ENTRANCE_LOBBY                 3.61             3.84  3.73
##  2 ENTRANCE_DOORS_WINDOWS         3.60             3.72  3.45
##  3 SECURITY                       3.94             4.2   3.91
##  4 STAIRWELLS                     3.4              3.6   3.41
##  5 LAUNDRY_ROOMS                  3.43             3.52  3.5
##  6 INTERNAL_GUARDS_HANDRAILS      3.63             3.68  3.5
##  7 GARBAGE_CHUTE_ROOMS            3.49             3.52  3.43
##  8 GARBAGE_BIN_STORAGE_AREA       3.56             3.64  3.55
##  9 ELEVATORS                      3.79             3.62  3.72
## 10 STORAGE_AREAS_LOCKERS          3.38             3.71  3
## 11 INTERIOR_WALL_CEILING_FLOOR    3.43             3.48  3.5
## 12 INTERIOR_LIGHTING_LEVELS       3.55             3.72  3.64
## 13 GRAFFITI                       4.72             4.88  4.82
## 14 EXTERIOR_CLADDING              3.48             3.48  3.45
## 15 EXTERIOR_GROUNDS               3.56             3.48  3.5
## 16 EXTERIOR_WALKWAYS              3.52             3.8   3.59
## 17 BALCONY_GUARDS                 3.62             3.53  3.36
## 18 WATER_PEN_EXT_BLDG_ELEMENTS    3.59             3.52  3.59
## 19 PARKING_AREA                   3.34             3.28  3.41
## 20 OTHER_FACILITIES               3.70             3.8   3.75
```

Here you can see the average scores for different variables for different categories of buildings. Generally speaking, the scores of each variable are about 3.5, and only the scores of graffiti are above 4 for all three types, which means they are all not so much graffiti. But in the parking area, the three types are all lower than 3.5 points, which reminds the relevant units to pay special attention to the safety of the parking lot.

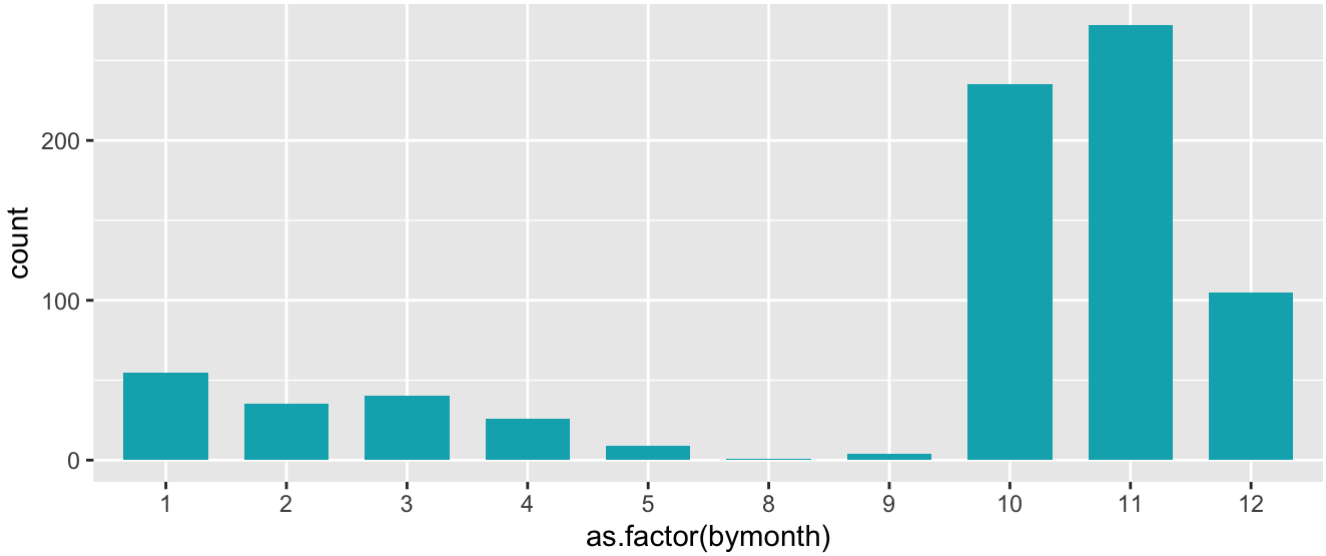
Season classification

```
Apdata3 <- Apdata2 %>%
  mutate(Season=quarter(EVALUATION_COMPLETED_ON)) %>%
  group_by(Season) %>%
  mutate(Season = recode(Season, '1' = 'Winter', '2' = 'Spring',
                         '3' = 'Summer', '4' = 'Fall'))
library(ggplot2)
ggplot(Apdata3, aes(x=as.factor(Season))) + geom_bar(fill = "#00AFBB", width = 0.7)
```



First we want to split the evaluation date into different quarters according to the date of the variable “evaluation_completed_on”. And according to different quarters, renamed as spring, summer, autumn, winter. After converting the seasons to a bar chart, it can be seen that the period with the most evaluations completed is mostly in the fall, followed by winter. But in fact, we cannot draw conclusions from quarters alone, we need more detailed observations.

```
ByMonth <- Apdata2 %>%
  mutate(bymonth=month(EVALUATION_COMPLETED_ON))
ggplot(ByMonth, aes(x=as.factor(bymonth))) + geom_bar(fill = "#00AFBB", width = 0.7)
```



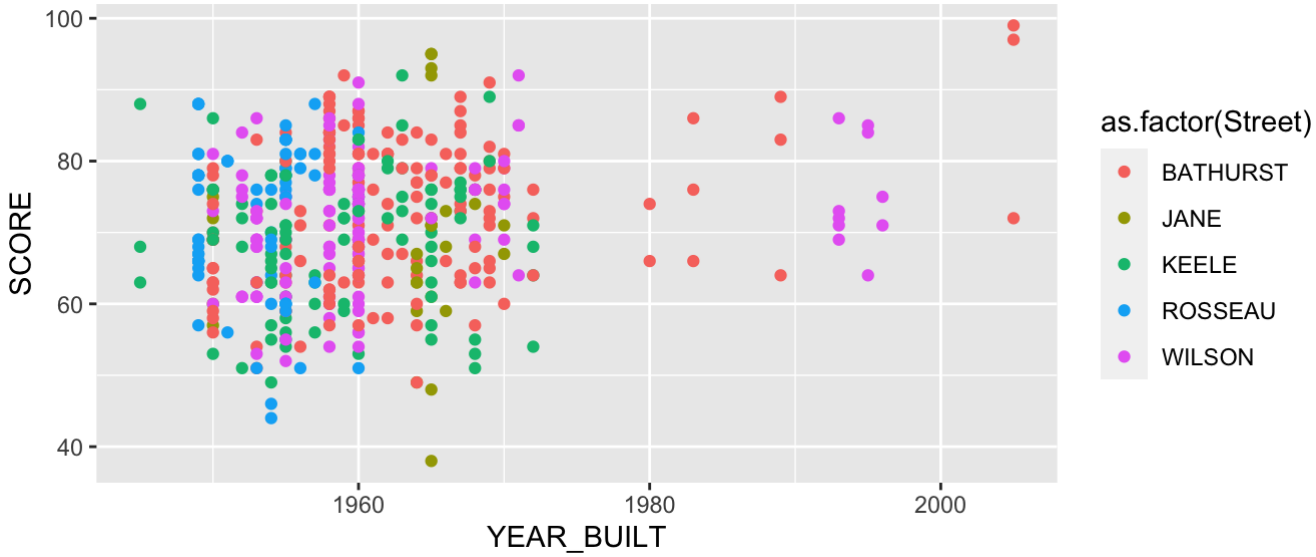
By converting to months, we find that most of the evaluators start to complete their evaluations in October. We can understand that the evaluators hope to complete the evaluation before the end of this year (perhaps involving their performance appraisal), so the date is mostly at the end of the year.

Five most common streets

```
library(tidyr)
Apdata3 <- separate(Apdata3, SITE_ADDRESS, c("Number", "Street"), " ")
Apdata4 <- Apdata3 %>% filter(Street %in% names(sort(table(Apdata3$Street), decreasin
g = TRUE)[1:5]))
unique(pull(Apdata4, Street))
```

```
## [1] "KEELE"    "WILSON"   "BATHURST" "ROSSEAU"  "JANE"
```

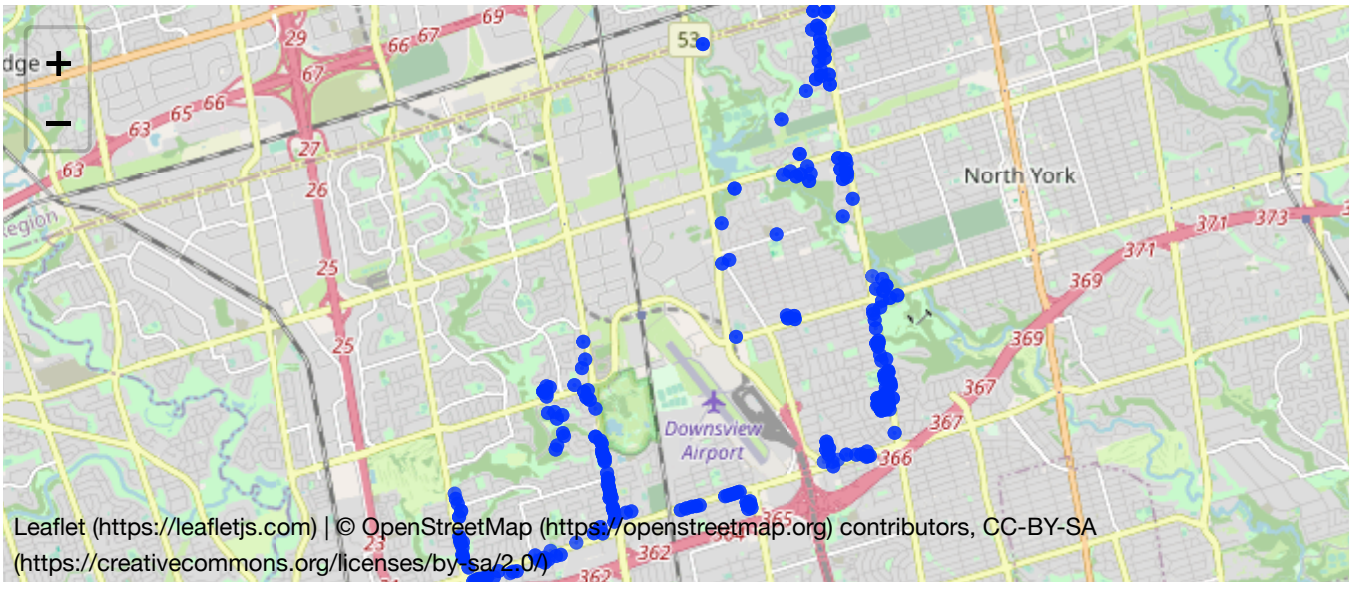
```
ggplot(data = Apdata4) +geom_point(aes(x=YEAR_BUILT, y=SCORE, color=as.factor(Stree
t)))
```



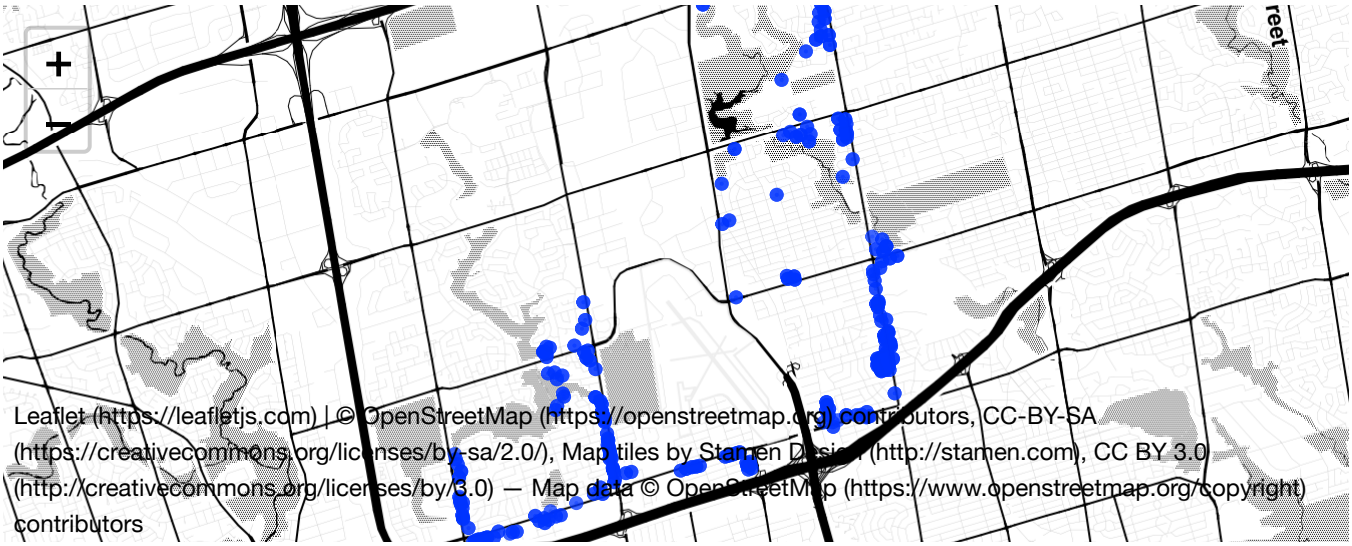
Next, we screened out the five most common streets in the evaluation of the York Center area. As can be seen from the dot scatter plot, there are newer buildings on Bathurst and Wilson streets. However, on these five streets, the scores of buildings do not seem to have much correlation with the year of construction.

Map

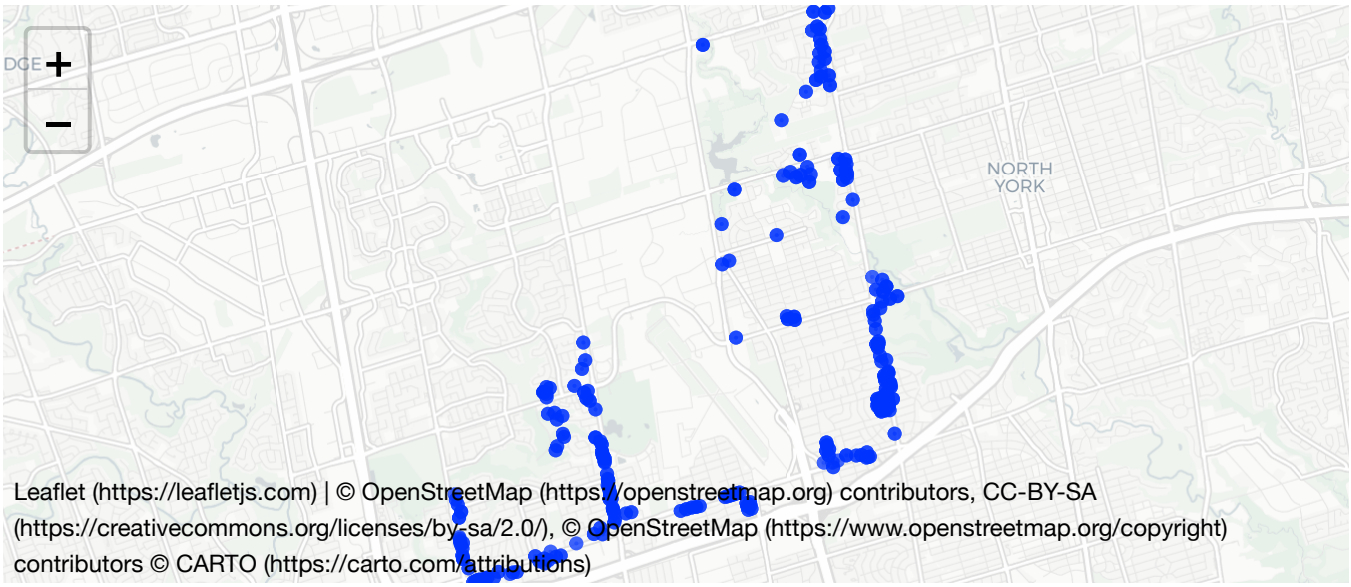
```
library(leaflet)
m <- leaflet() %>% addTiles() %>% addCircles(lng = Apdata3$LONGITUDE, lat = Apdata3$L
ATITUDE)
m
```




```
m2 <- leaflet() %>% addTiles() %>% addCircles(lng = Apdata3$LONGITUDE, lat = Apdata3
$LATITUDE) %>% addProviderTiles(providers$Stamen.Toner)
m2
```




```
m3 <- leaflet() %>% addTiles() %>% addCircles(lng = Apdata3$LONGITUDE, lat = Apdata3
$LATITUDE) %>% addProviderTiles(providers$CartoDB.Positron)
m3
```



Finally, we used the function in the leaflet package to print out all the evaluated buildings in the York Center area. Interestingly, at this time we can find that Downsview Airport is in this area. In addition, we can print out maps of different styles through the addProviderTiles() function. Still, my favorite is the original version, because it's shown in more detail.