ML - Project 2

2023-08-07

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## corrplot 0.92 loaded
##
##
## Attaching package: 'zoo'
##
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

Probability Practice

Part A Overall probability of someone answering "Yes" is 65%, so P(Yes) = 0.65

Probability of being a random clicker as P(Random) = 0.3

Probability of a random clicker choosing "Yes" is 0.5

According to the rule of total probability - P(Yes) = P(Yes,Truthful) + P(Yes,Random)

=> 0.65 = P(Yes,Truthful) + [P(Random) * P(Yes | Random)]

=> 0.65 = P(Yes,Truthful) + [0.3 * 0.5]

Now, solving for P(Yes,Truthful):

=> P(Yes,Truthful) = 0.65 - [0.3 * 0.5]

=> P(Yes,Truthful) = 0.65 - 0.15

=> **P(Yes,Truthful) = 0.5**

Approximately 50% of truthful clickers answered "Yes" to the survey.

Wrangling the Billboard Top 100

Part A

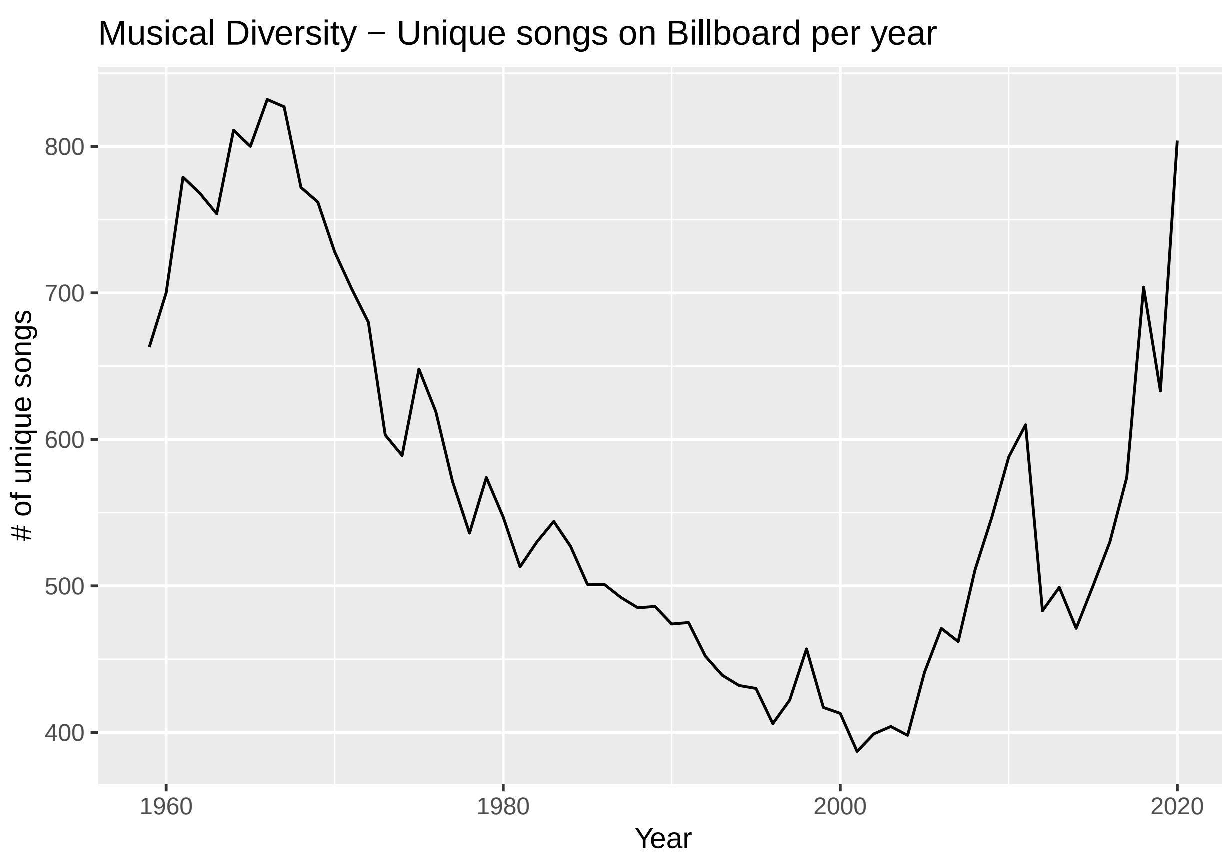
```
## `summarise()` has grouped output by 'song'. You can override using the
## `.groups` argument.
```

```
## [1] "Top 10 most popular songs since 1958 from Billboard"
```

```
## # A tibble: 10 x 3
## # Groups:   song [10]
##    song                                performer                  count_instances
##    <chr>                               <chr>                                <int>
##  1 Radioactive                         Imagine Dragons                         87
##  2 Sail                                AWOLNATION                              79
##  3 Blinding Lights                     The Weeknd                              76
##  4 I'm Yours                           Jason Mraz                              76
##  5 How Do I Live                       LeAnn Rimes                             69
##  6 Counting Stars                      OneRepublic                             68
##  7 Party Rock Anthem                   LMFAO Featuring Lauren B~               68
##  8 Foolish Games/You Were Meant For Me Jewel                                   65
##  9 Rolling In The Deep                 Adele                                   65
## 10 Before He Cheats                    Carrie Underwood                        64
```

Part B

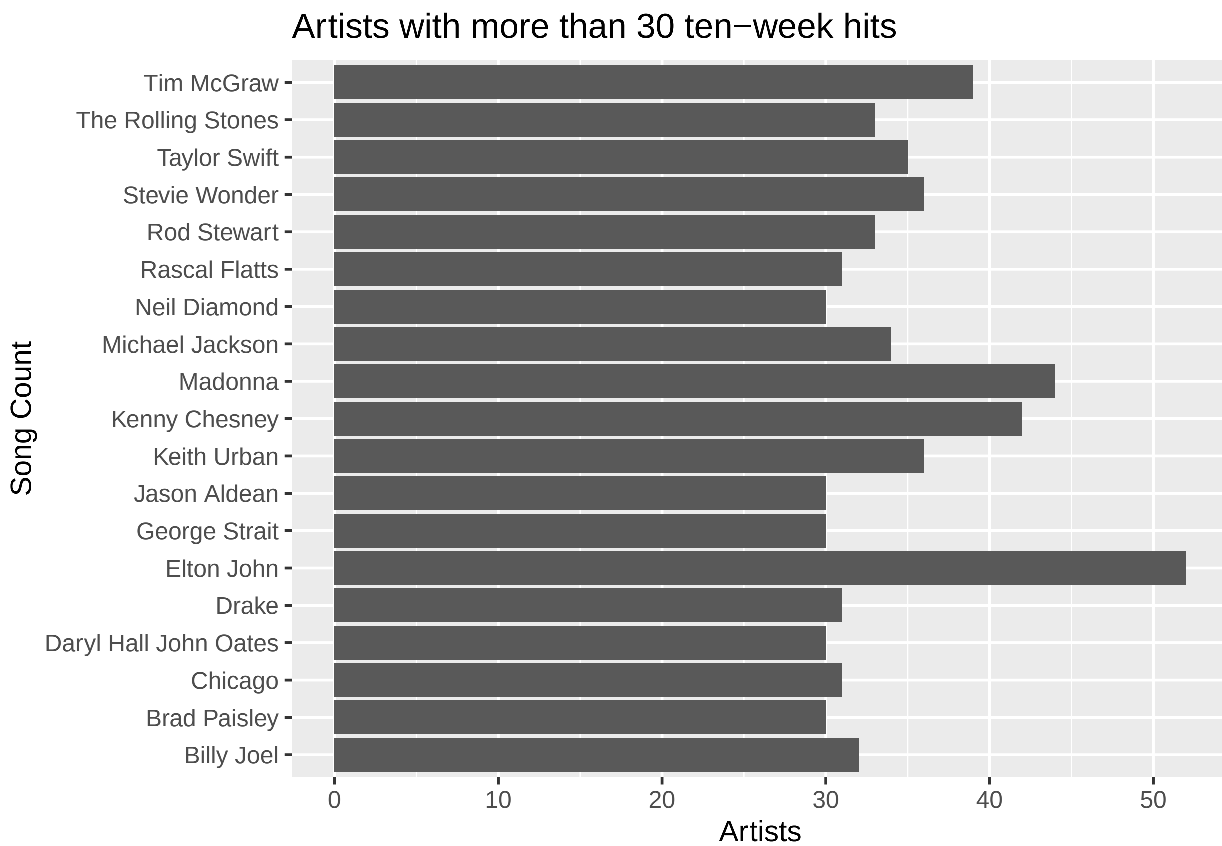
```
## `summarise()` has grouped output by 'year', 'song'. You can override using the
## `.groups` argument.
```



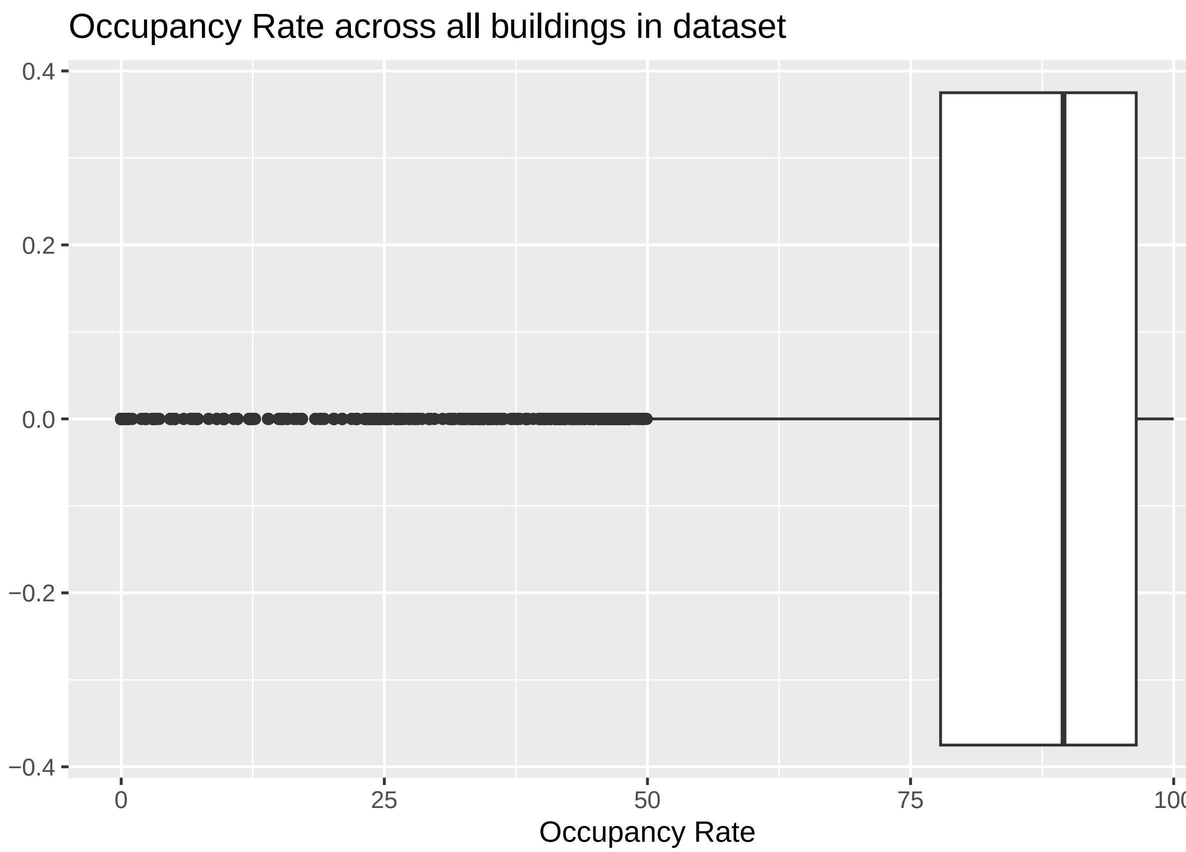
The musical diversity peaked in the mid 1960s over 800 unique songs, but took a hit and kept dropping till right after 2000 where it hit it's least unique songs and started increasing to match it's peak in a span of 20 years

Part C

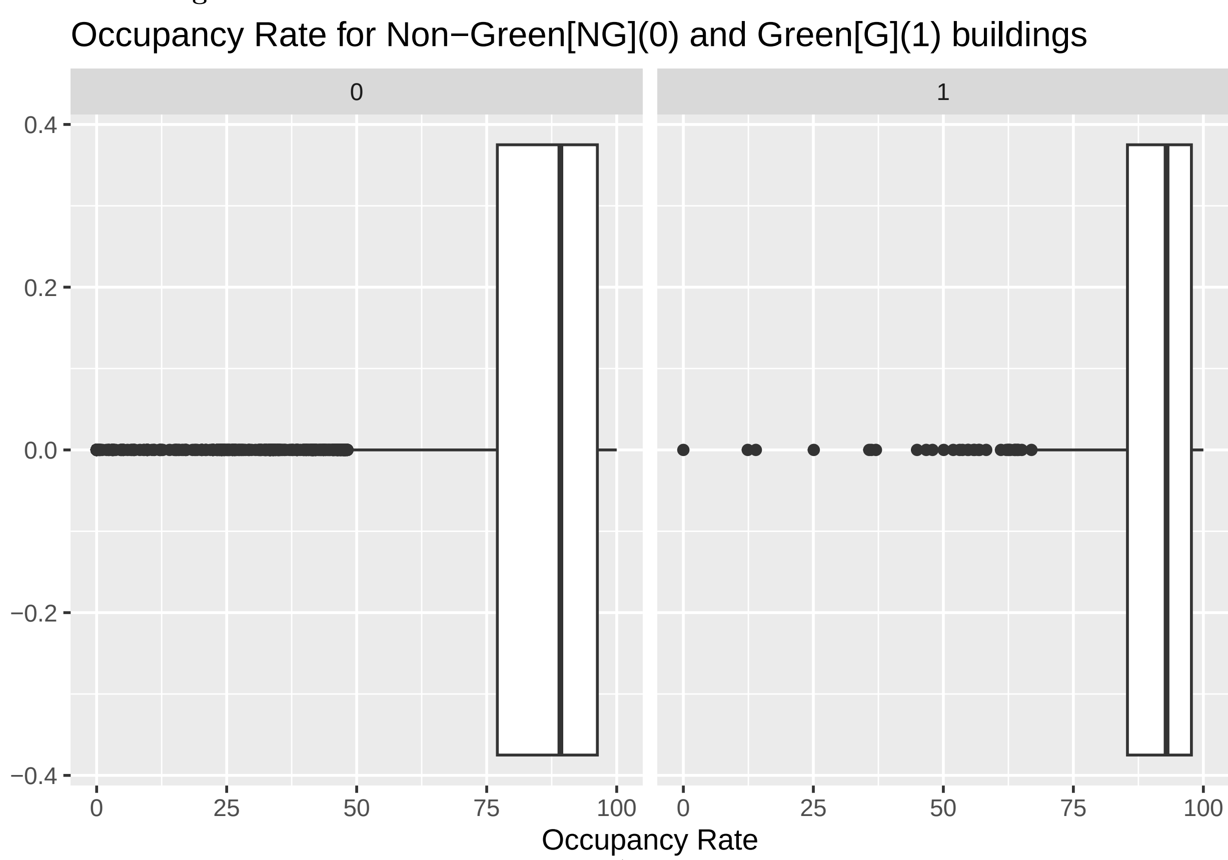
```
## ‘summarise()‘ has grouped output by ’song’. You can override using the
## ‘.groups‘ argument.
```

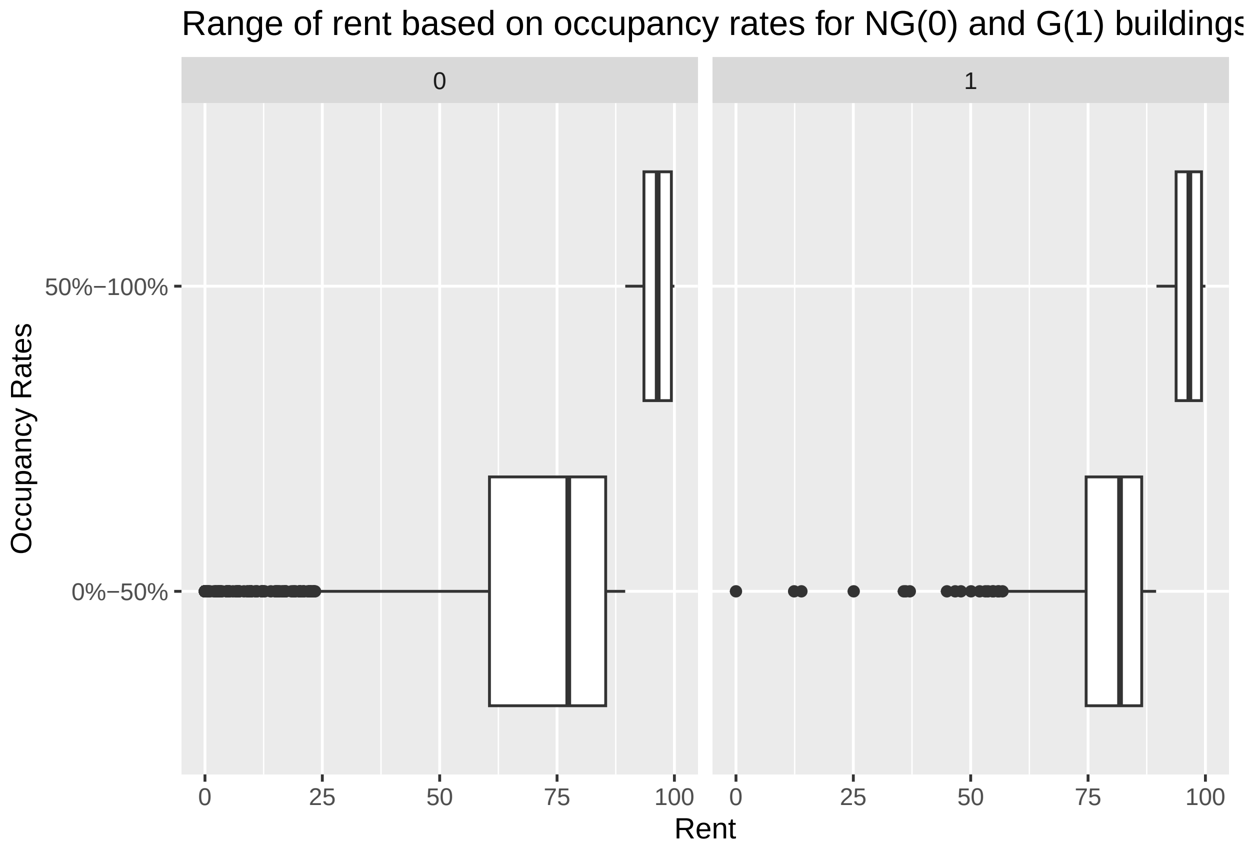


Visual Story telling Part 1: Green Buildings



Outlier marking:

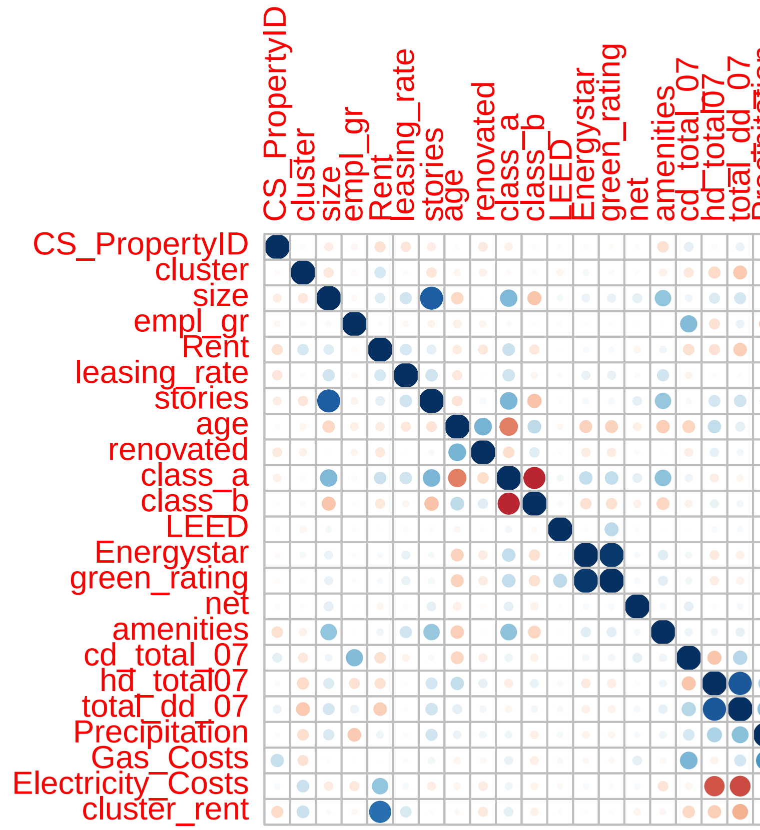




Findings:

- The occupancy rates of the buildings in the dataset fall within 0 to 100, but the quantile range of 25 and 75 fall between 78% to 96% occupancy
- When we look at green and non-green buildings separately, the green buildings had only a few buildings that had a low occupancy rate but vice versa for non-green buildings
- Looking at rent for these occupancy rates between NG (non-green) and G(green) buildings, we see that the rent for non-green buildings with a lower occupancy rate was higher than green buildings

Since there is an impact of occupancy rate on green buildings as well as the rent, it would be better to now mark any outliers based on this variable as of now, but to proceed with the given dataset as it is.



Finding variables that may impact rent

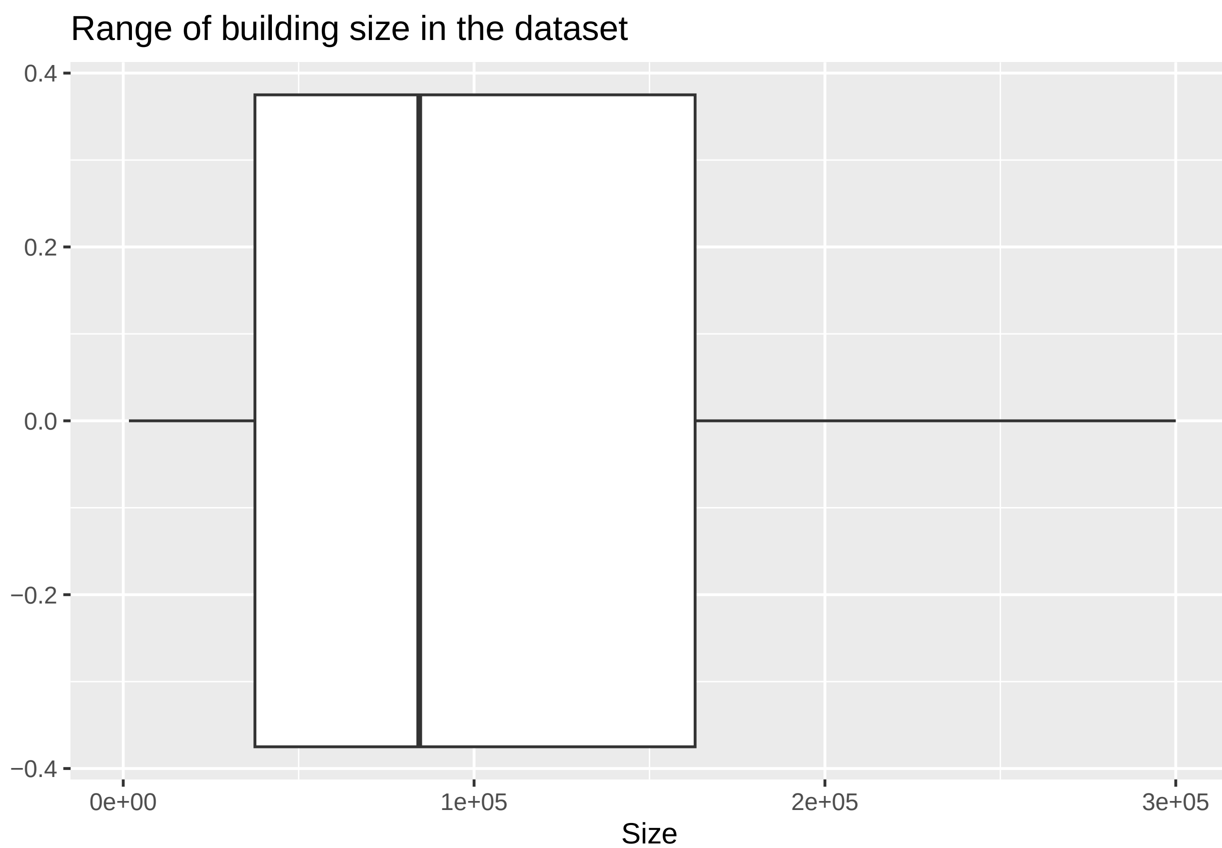
Findings:

- From the correlation plot we can see that cluster, size, occupancy rate, stories, class_a, electricity_costs and cluster rent were positively correlated with rent
- Age, total number of degree days, class_b, renovated were negatively correlated with rent

Considering the information we have about the building - **size, age, stories, class and occupancy rate** were relevant to filter for - so that it is similar to the case of the building we are going for

Filtering the dataset to get buildings similar to the specifications of the building to be constructed

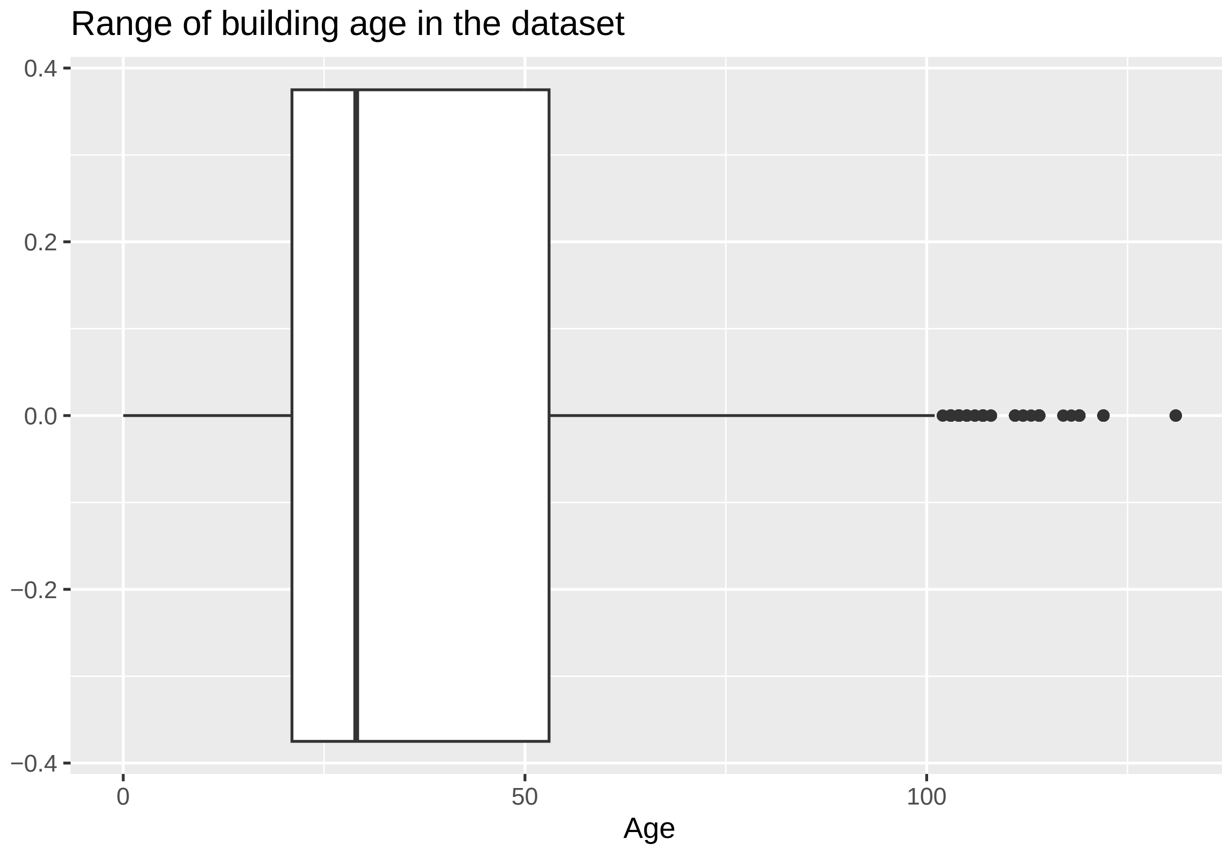
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1624   50891  128838  234638  294212 3781045
```




```
## ‘summarise()‘ has grouped output by ’green_rating’. You can override using the
## ‘.groups‘ argument.
```

```
## # A tibble: 6 x 4
## # Groups:   green_rating [2]
##   green_rating size_groups       median_rent     n
##          <int> <chr>                   <dbl> <int>
## 1            0 0-50th Quantile          24    3765
## 2            0 50-75th Quantile         27    1744
## 3            0 75-100th Quantile        24.8  1700
## 4            1 0-50th Quantile          28.2   177
## 5            1 50-75th Quantile         28.7   234
## 6            1 75-100th Quantile        26     274
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   21.00   29.00   41.78   53.00  131.00
```

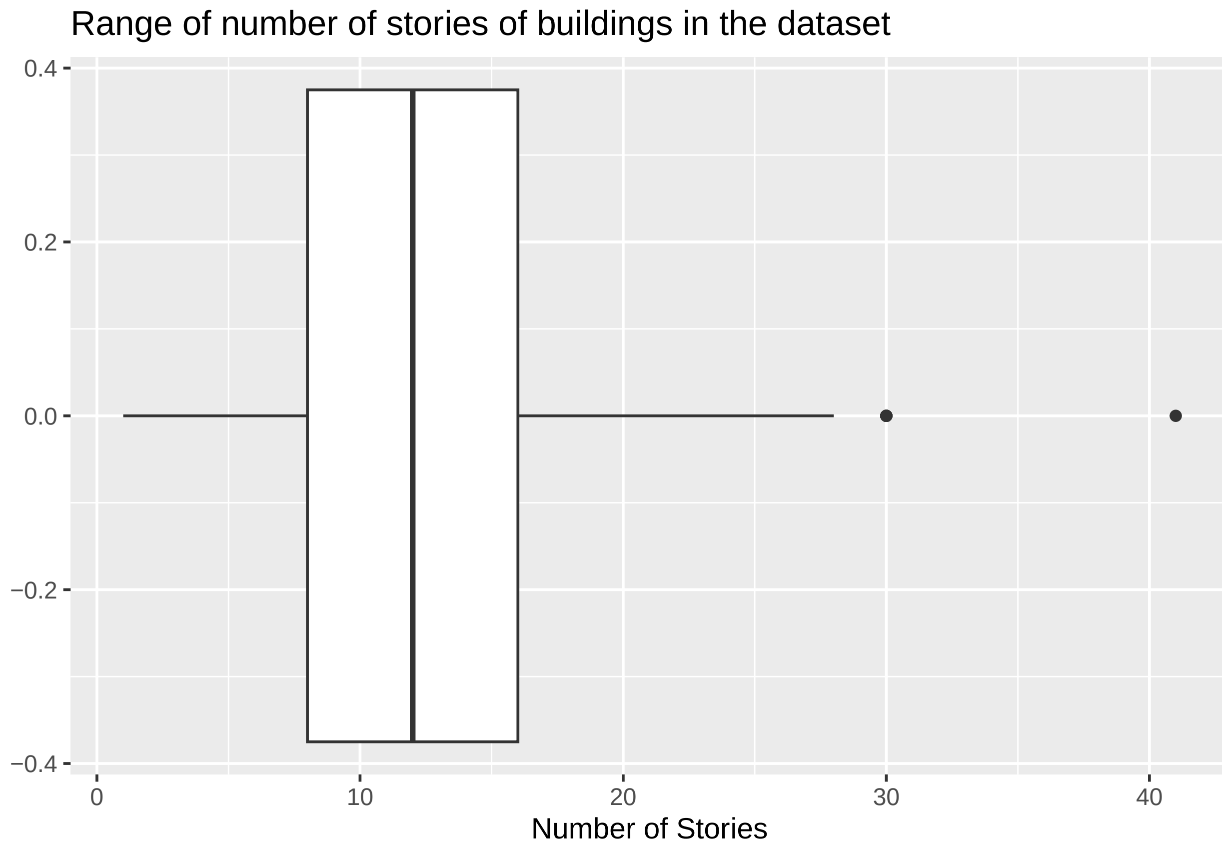



```
## [1] 29
```

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

```
## # A tibble: 4 x 4
## # Groups:   green_rating [2]
##   green_rating new_old median_rent     n
##          <int> <chr>         <dbl> <int>
## 1            0 New            30.2   788
## 2            0 Old            25     956
## 3            1 New            28.6   188
## 4            1 Old            29.9    46
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    8.00   12.00   12.45   16.00   41.00
```




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

## [1] "\n"

## [1] "25th Quantile: 8"

## [1] "50th Quantile: 12"

## [1] "75th Quantile: 16"

## [1] "90th Quantile: 22"

## # A tibble: 10 x 4
## # Groups:   green_rating [2]
##    green_rating stories_groups    median_rent     n
##           <int> <chr>                   <dbl> <int>
##  1            0 0-25th Quantile          25.4   141
##  2            0 25-50th Quantile         35.4   207
##  3            0 50-75th Quantile         35.5   213
##  4            0 75-90th Quantile         25.6   130
##  5            0 90-100th Quantile        25.6    97
##  6            1 0-25th Quantile          25.6    58
##  7            1 25-50th Quantile         31.5    62
```

```
##  8            1 50-75th Quantile         31.8    41
##  9            1 75-90th Quantile         28.5    25
## 10            1 90-100th Quantile        21       2
```

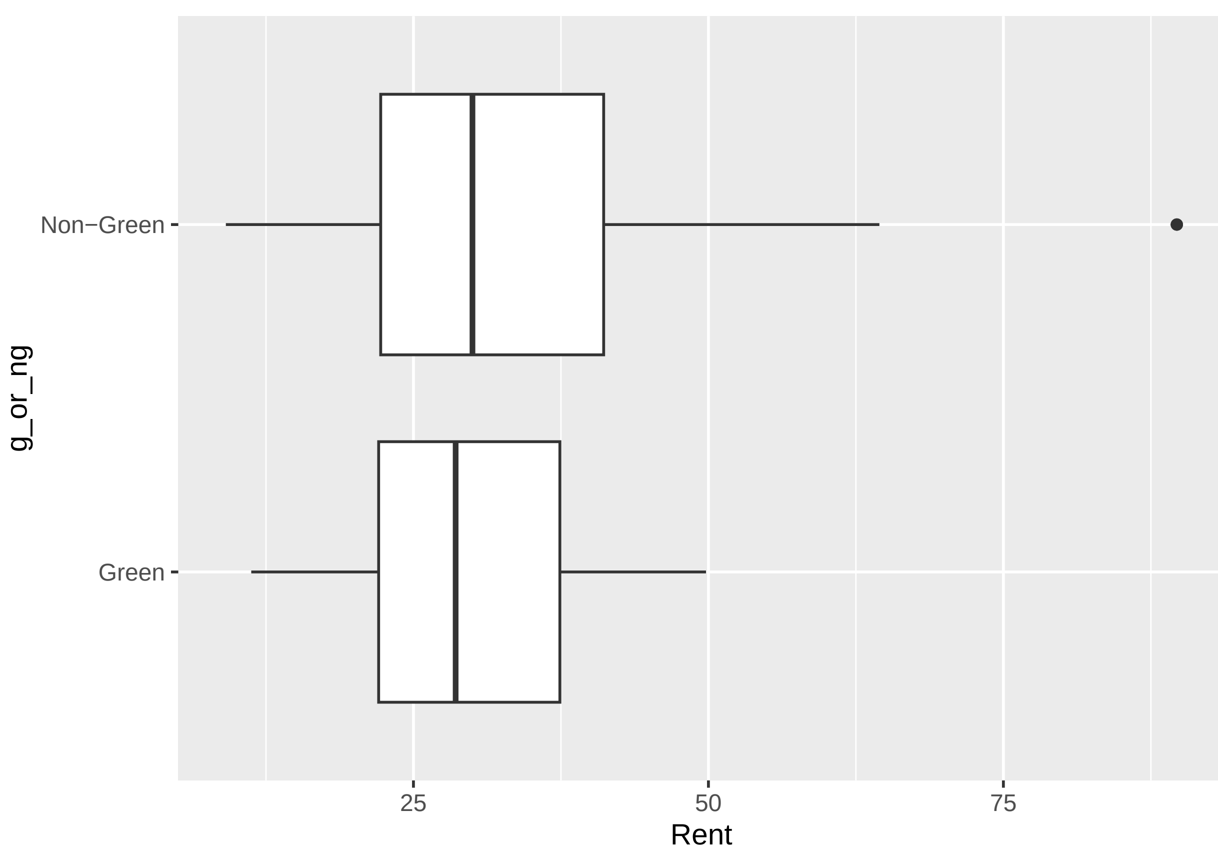
Findings:

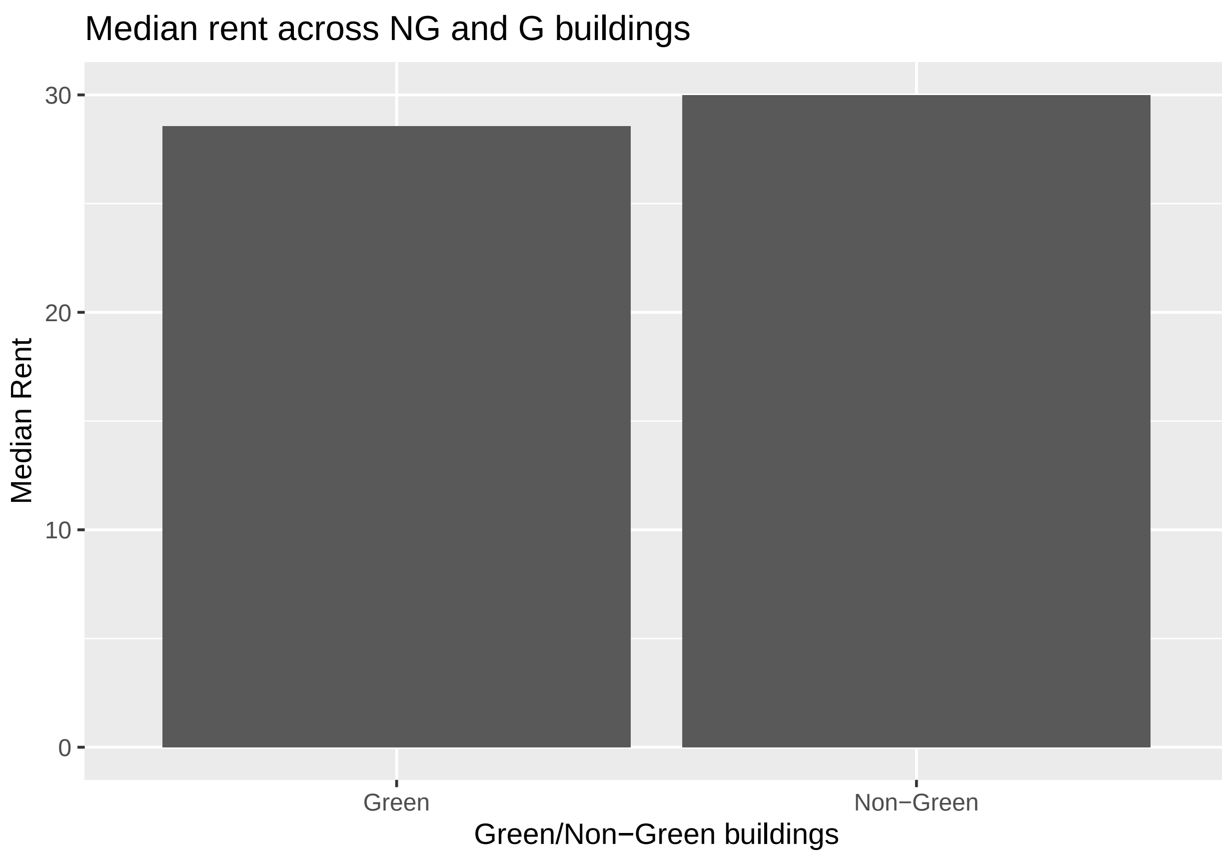
- The dataset had a very high range in terms of size and also affected rent, so filtered the dataset to keep it within the limits of the 50th quantile and 75th quantile range [128838 sq.ft to 294212 sq.ft] given that the building under consideration is estimated to be 250000 sq.ft
- It also had a long range in terms of the age of the building, which also affected rent, so filtered the dataset to keep relatively new buildings below the median age of all buildings (29 years)
- The dataset had a range of buildings with 1 story to 41 stories, which affected rent as well, so filtered to keep buildings that have 12 to 21 stories pertaining to the 50th and 90th quantiles respectively

Finding the rent of green and non-green buildings in the new filtered dataset

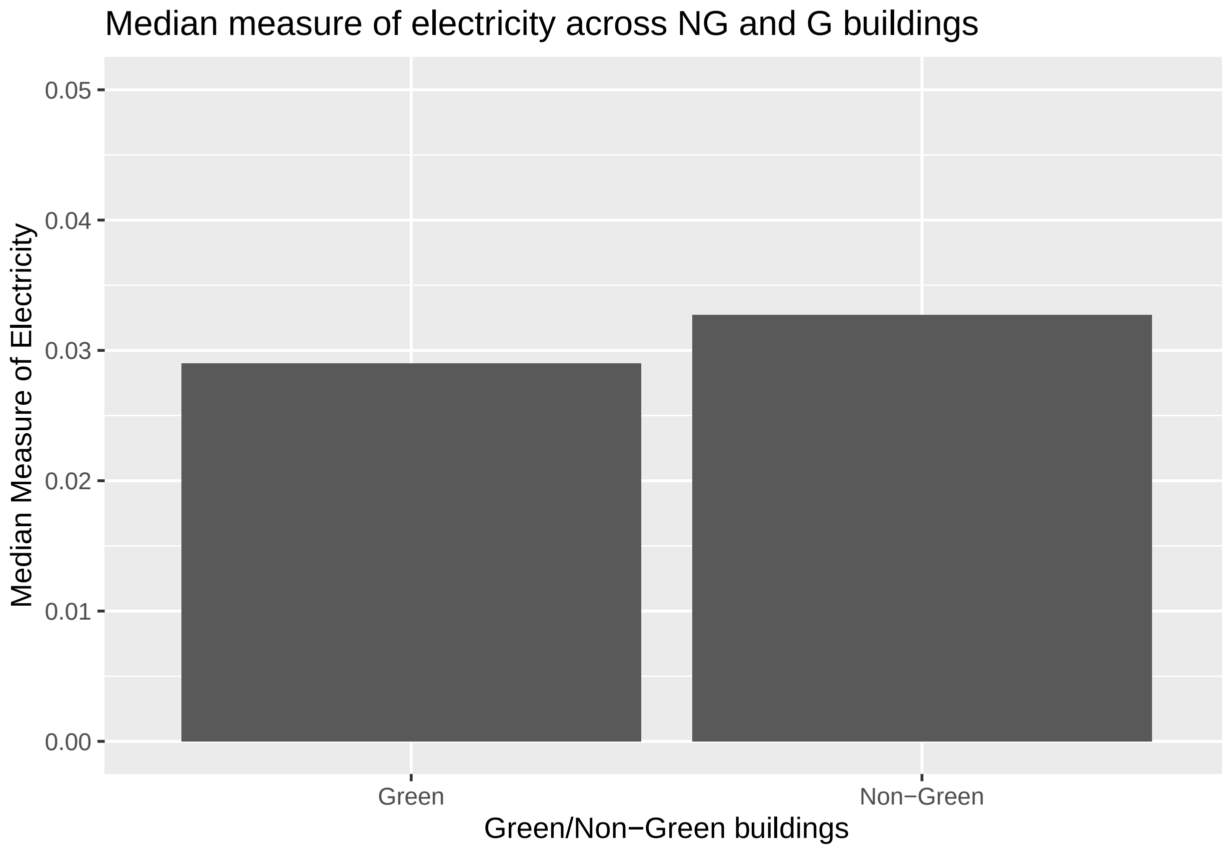
```
## # A tibble: 2 x 3
##   g_or_ng   median_rent     n
##   <chr>           <dbl> <int>
## 1 Green            28.6    66
## 2 Non-Green        30     343
```

```
## [1] "Loss in rent per year = 350000"
```





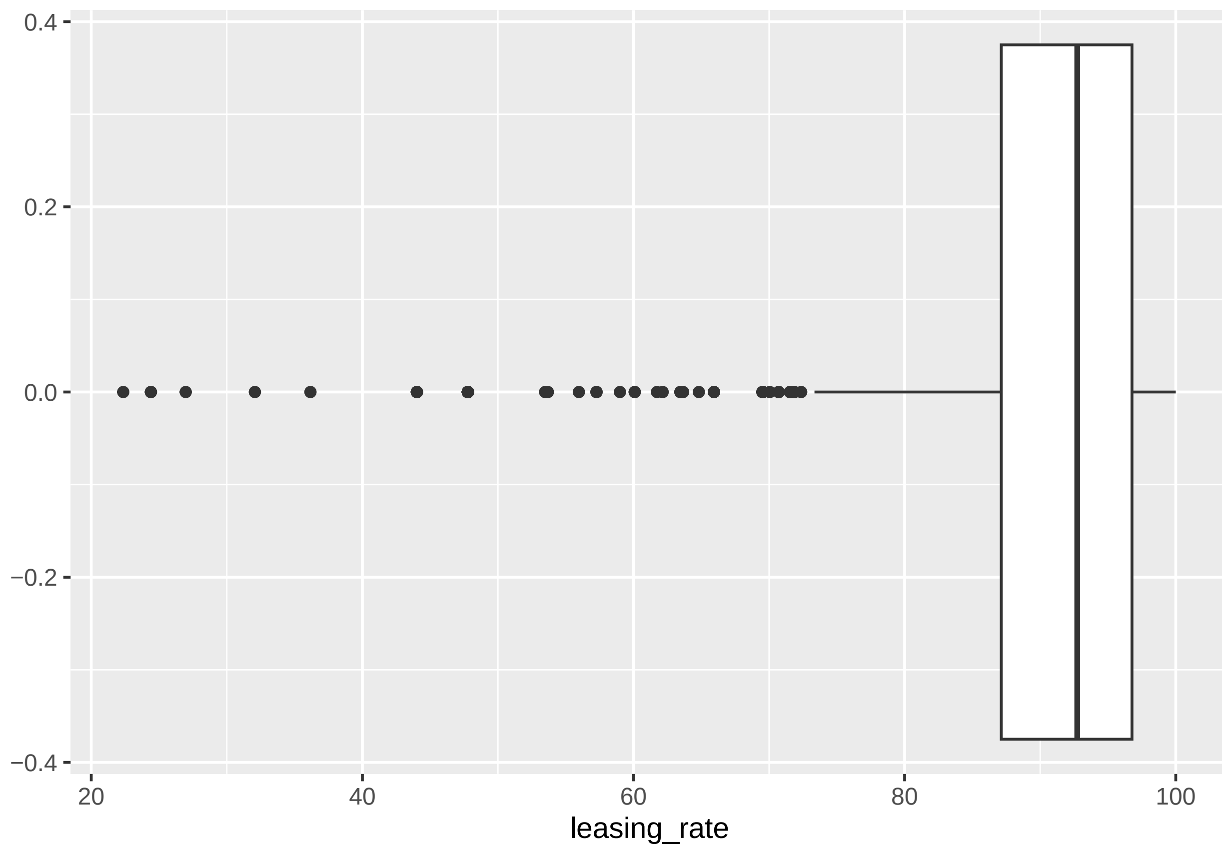

```
## # A tibble: 2 x 3
##   g_or_ng   median_elec     n
##   <chr>           <dbl> <int>
## 1 Green           0.029    66
## 2 Non-Green      0.0327   343
```




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

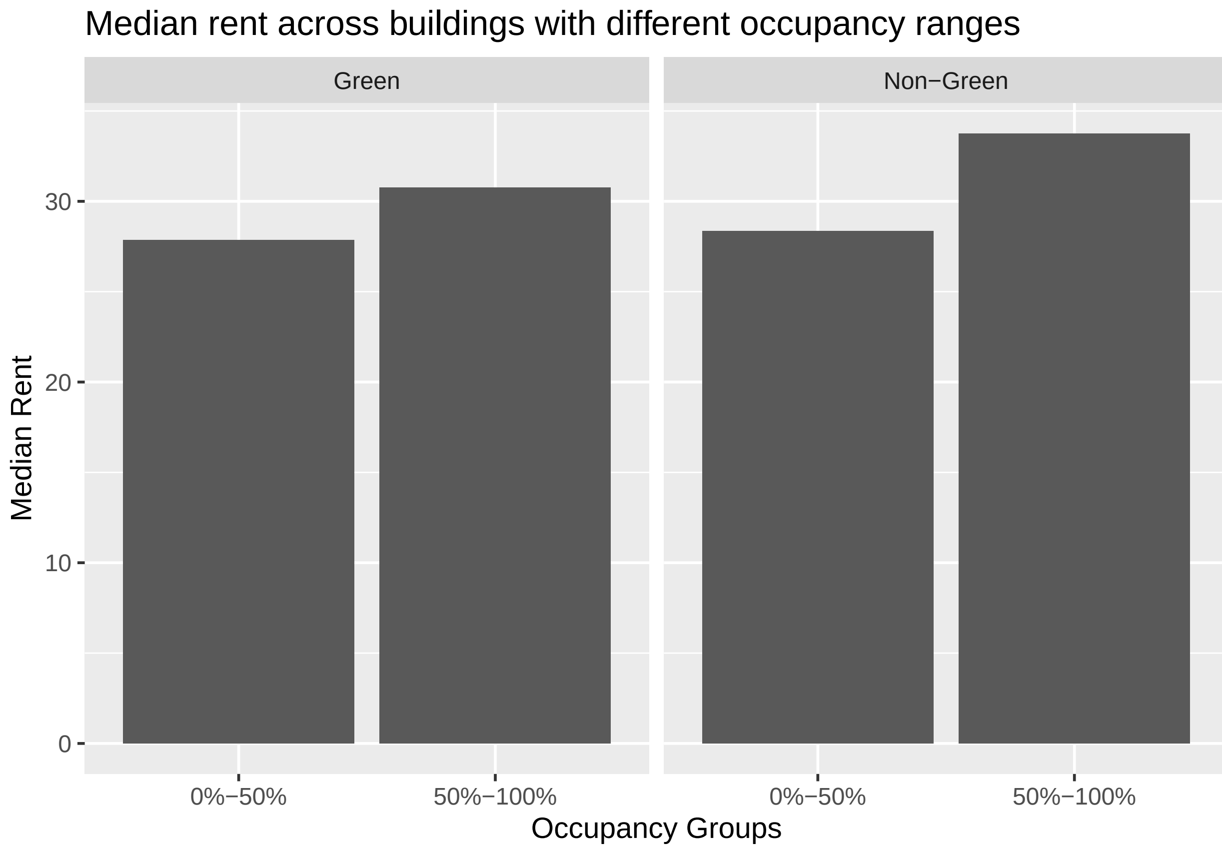
```
## # A tibble: 5 x 4
## # Groups:   g_or_ng [2]
##   g_or_ng   Classes median_rent     n
##   <chr>     <chr>         <dbl> <int>
## 1 Green     Class A        31.8    59
## 2 Green     Class B        23.6     7
## 3 Non-Green Class A        33.2   242
## 4 Non-Green Class B        28.0    98
## 5 Non-Green Class C        29       3
```


```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   22.36   87.13   92.73   88.28   96.77  100.00
```




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

```
## # A tibble: 4 x 4
## # Groups:   g_or_ng [2]
##   g_or_ng   occupancy_groups median_rent     n
##   <chr>     <chr>                  <dbl> <int>
## 1 Green     0%-50%                  27.9    36
## 2 Green     50%-100%                30.8    30
## 3 Non-Green 0%-50%                  28.4   168
## 4 Non-Green 50%-100%                33.8   175
```



Findings:

- Looking at the median value of rent across all green and non-green buildings, green buildings have a lesser rent value compared to non-green buildings
- When we look at the class and occupancy rates, we get similar results of green buildings having a lesser value than non-green buildings irrespective of the class or the occupancy rate

Recommendation: Though green buildings are looked at positively at an environment perspective, in an economical standpoint, building a green building would not only increase the construction costs, but also produce lesser rent compared to non-green buildings, leading to a loss of 5 million dollars during construction along with a loss in rent of 350,000 dollars per year. Therefore constructing a non-green better is going to yield more profits from an economic point of view