EDAV Final Project NYC Tree Census

Ruizhi Zhang, Jie Zheng, Junnan Zhao, Yawen Han

December 8, 2018

INTRODUCTION

With the development of modern industry, the urban population has increased dramatically, the environment has deteriorated, and the ecology has been seriously threatened. Urban greening can improve the quality of life by purifying the air, reducing noise, and beautifying the environment. What is more, as one of the most crucial cities in the United States, New York's greening reflects not only the overall greening level of American cities, but also a good example for other cities. Therefore, our project is mainly to study the greening levels in New York, and trees' levels is an important indicator of urban greening among many greening indicators. After discussion, our research project has narrowed down to the study of trees' levels in NYC, in order to reflect the greening levels in New York.

Our data comes from New York Open Data resource which provides 2015 street trees information in New York City. Based on the data availability, our project aims to explore the following questions: 1) How is the survival situation and health condition of NYC street trees? 2) What factors impact the tree's living situation significantly? 3) Whether these relationships and conclusions vary across different boroughs?

In the analysis, we first analyzed the reliability of the data set, including missing values, whether data is biased, then cleaning up data based on previous analysis. After completing the data cleaning, we analyzed the distribution of trees status (alive, dead, stump), the health conditions of survived trees, as well as diameters of the trees. In the following analysis, we analyzed whether these properties and attribute are related to variables such as boroughs and tree guard situations etc. Finally, we discuss the limitations and future work.

Each team member evolved in all parts and contributes equally to this project. All members participated in main analysis part. Jie Zheng and Yawen Han took charge of interactive component, Ruizhi Zhang mainly focused on Data Analysis Quality part, Junnan Zhao was charge of Data Description part. In the end, we finished report together, including introduction and conclusion.

DESCRIPTION OF DATA

Our data is from 2015 Street Tree Census conducted by volunteers and staff organized by NYC Parks & Recreation (DPR) and partner organizations. Those data are accessible on https://data.cityofnewyork.us/Environment/2015-Street-Tree-Census-Tree-Data/uvpi-gqnh. This dataset is created on June 3, 2016 and has been updated on October 4, 2017 most recently. There are 683788 tree observations of 45 variables in the raw dataset. According to the research goals, we remain 14 related variables we are interested most, such as tree species, tree diameter, survival status, perception of health, etc. We use "tree species" as the variable name instead of "spc_common", "tree diameter" instead of "tree_dbh" and "data collector" instead of "user_type". The tree diameter is measured by Diameter at Breast Height (DBH) method, which refers to the tree diameter measured at 4.5 feet above the ground. Table 1 describes all the variables we are interested in our project in details.

Table 1: Description of Raw NYC Street Tree Variables

https://github.com/RuizhiZhang/EDAVfinalprojectNYCtrees/blob/master/Table%201.pdf

ANALYSIS OF DATA QUALITY

Missing Patterns

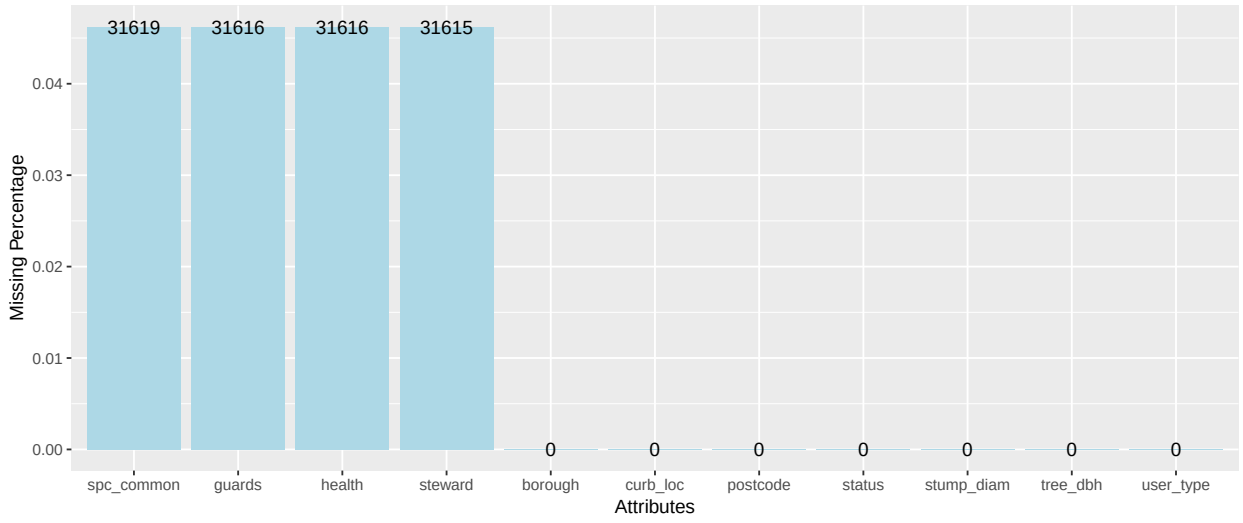
Firstly, the dataset is loaded here, and a subset as mentioned in data description part is extracted into **Tree** variable.

```
library(tidyverse)
library(ggthemes)
library(dplyr)

# loading data
Tree<-read.csv("Tree_Data.csv")

# create a subset for tree dataset w/ demanded attributes
Tree<-Tree[,c('tree_dbh','stump_diam','curb_loc','status','health','spc_common','steward',
              'guards','user_type','postcode','borough')]
```

Missing Value Bar Chart

```
# standardization
Tree[Tree == '']<-NA

TreeMiss<-colSums(is.na(Tree))
TreeMissPercent<-TreeMiss/length(Tree$tree_dbh)
TreeValue<-names(TreeMissPercent)
TreePercent<-unname(TreeMissPercent)
TreeMissPer<-data.frame(TreeValue,TreePercent)
ggplot(TreeMissPer, aes(x=reorder(TreeValue,-TreePercent),y=TreePercent)) +geom_bar(stat="identity",fil
  "Fig 1. Bar Chart: Percentage/Count of Missing values by Attributes")+labs(x="Attributes",
                                                                             y="Missing Percentage")
```

Fig 1. Bar Chart: Percentage/Count of Missing values by Attributes



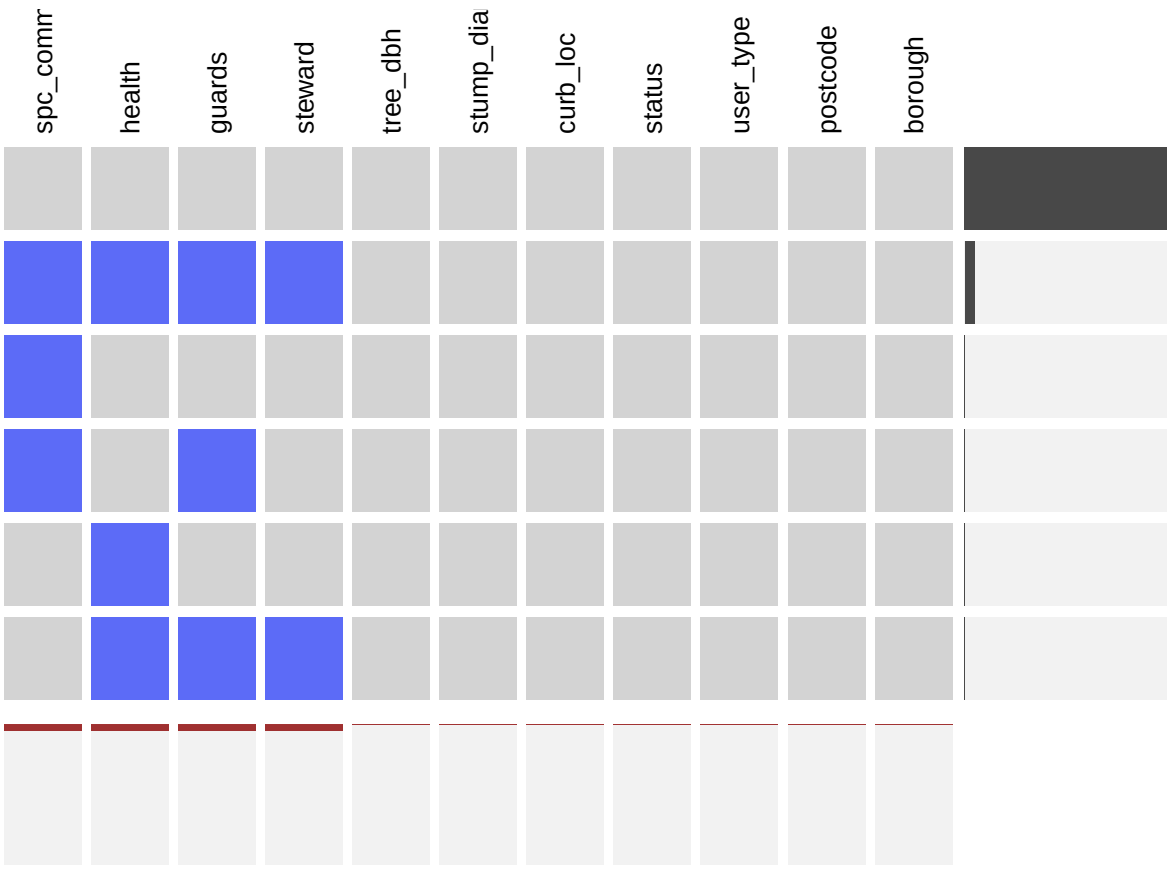
According to the bar chart of the missing value, the missing values were distributing in spc_common, guards, health, and steward with value around 31620 for each, occupying more than 4.5% of the total observations. Then, the specific missing pattern was shown below.

Missing Value Heatmap

```
library(extracat)
cat("Fig 2. Heatmap: Missing Value")
```

```
## Fig 2. Heatmap: Missing Value
```

```
visna(Tree,sort="b")
```



According to the missing value heatmap, there were in total 5 missing patterns:

1) All four spc_common, guards, health, and steward are missing, which was the largest portion of missing except no missing obs.;
2) Only spc_common missing;
3) spc_common and guards missing;
4) Only health missing;
5) health, guards, and steward missing;

Compared to the full-set observations, the number of missing for each pattern was too tiny to observable.

There were some reasons resulting in the distribution of missing values. One important reason was the varibales in status. For both stump and dead tree, it was resonable that it was hard for the recorders to provided info for those trees.

```
cat("Table 2. Correlation of Missing Value against Status")
```

```
## Table 2. Correlation of Missing Value against Status
```

```
Tree %>%
  group_by(status) %>%
  summarize(tree_spc_num_na = sum(is.na(`spc_common`))/n(),
            tree_health_num_na = sum(is.na(`health`))/n(),
            tree_guards_num_na = sum(is.na(`guards`))/n(),
            tree_steward_num_na = sum(is.na(`steward`))/n())
```

```
## # A tibble: 3 x 5
##   status tree_spc_num_na tree_health_num~ tree_guards_num~ tree_steward_n~
##   <fct>            <dbl>            <dbl>            <dbl>           <dbl>
## 1 Alive       0.00000767       0.00000153       0.00000153               0
## 2 Dead        1.000            1                1                        1
## 3 Stump       1                1                1                        1
```

Then, a brief analysis of stump and dead trees was shown here. Firstly, the sum of the all stump and dead trees was calculated as below.

```
# #of both Stump and Dead trees
sum(Tree$status == "Stump") + sum(Tree$status == "Dead")
```

```
## [1] 31615
```

```
# percentage of total observations
cat('The percentage of the stump and dead trees is around',
    round(31615/length(Tree$status)*100), '% of the total recorded trees.')
```

```
## The percentage of the stump and dead trees is around 5 % of the total recorded trees.
```

Detailed Statistic Features

In this part, we will see detailed features of the dataset. Firstly, there were 683,788 observations and 10 features for each observation.

```
dim(Tree)
```

```
## [1] 683788     11
```

For 10 features, their names were shown below.

```
names(Tree)
```

```
##  [1] "tree_dbh"   "stump_diam" "curb_loc"   "status"     "health"
##  [6] "spc_common" "steward"    "guards"     "user_type"  "postcode"
## [11] "borough"
```

For the given 9 attributes,

Categorical: curb_loc, status, health, spc_common, steward, guards, user_type, postcode, borough

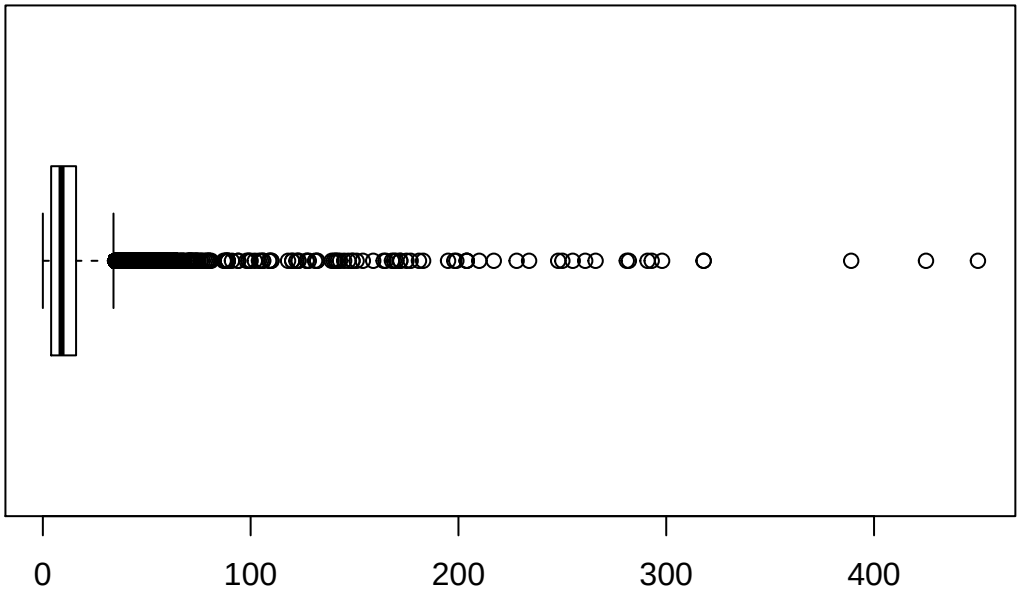
Discrete: tree_dbh, stump_diam

Discrete Variable Outliers

Since there was only two features, tree_dbh and stump_diam, being discrete variables. Firstly, as a first shot of the outliers of tree dbh, a boxplot was plotted.

```
boxplot(Tree$tree_dbh, horizontal = TRUE, main="Fig 3. Boxplot: tree_dbh")
```

Fig 3. Boxplot: tree_dbh



According to the boxplot shown above, the dbh were heavely spreaded from 0 to more than 400, where the main distribution was concentrated within 30. By zooming in, a 5-value summary was calculated as below:
5-value summary:

```
boxplot.stats(Tree$tree_dbh)[1]
```

```
## $stats
## [1]  0  4  9 16 34
```

For all dbh of trees, the min value was 0, the lower hinge was 4, the median was 9, the upper hinge was 16, and the max value was 34. However, according to the data description, the dbh of stumps were set to 0, redo the statistics for non-stump trees.

```
boxplot.stats(Tree$tree_dbh[Tree$status != "Stump"])[1]
```

```
## $stats
## [1]  0  5 10 16 32
```

Therefore, the dbh larger than 16+(16-5)*1.5=32.5 were outliers.

```
# # of outliers
sum(Tree$tree_dbh>32.5)
```

```
## [1] 15405
```

```
# percentage of outliers
sum(Tree$tree_dbh>32.5)/length(Tree$tree_dbh)
```

```
## [1] 0.02252891
```

On the other hand, there were some concerns about the stump and non-stump diameters.

```
mean(Tree$stump_diam[Tree$status == "Stump"])
```

```
## [1] 16.75048
```

```
mean(Tree$tree_dbh[Tree$status != "Stump"])
```

```
## [1] 11.57873
```

The mean of the stump was fairly larger than the mean of the non-stump trees. There might be two hypothesis could be considered. The reason why the stumps were lefted rather than removed from roots might result from the concerning of diameters of the trees. The thicker the stumps were, the harder the trees could be totally removed. Moreover, the stumped trees might be cut because they were too thick that they became the obstacles in human living senarios.

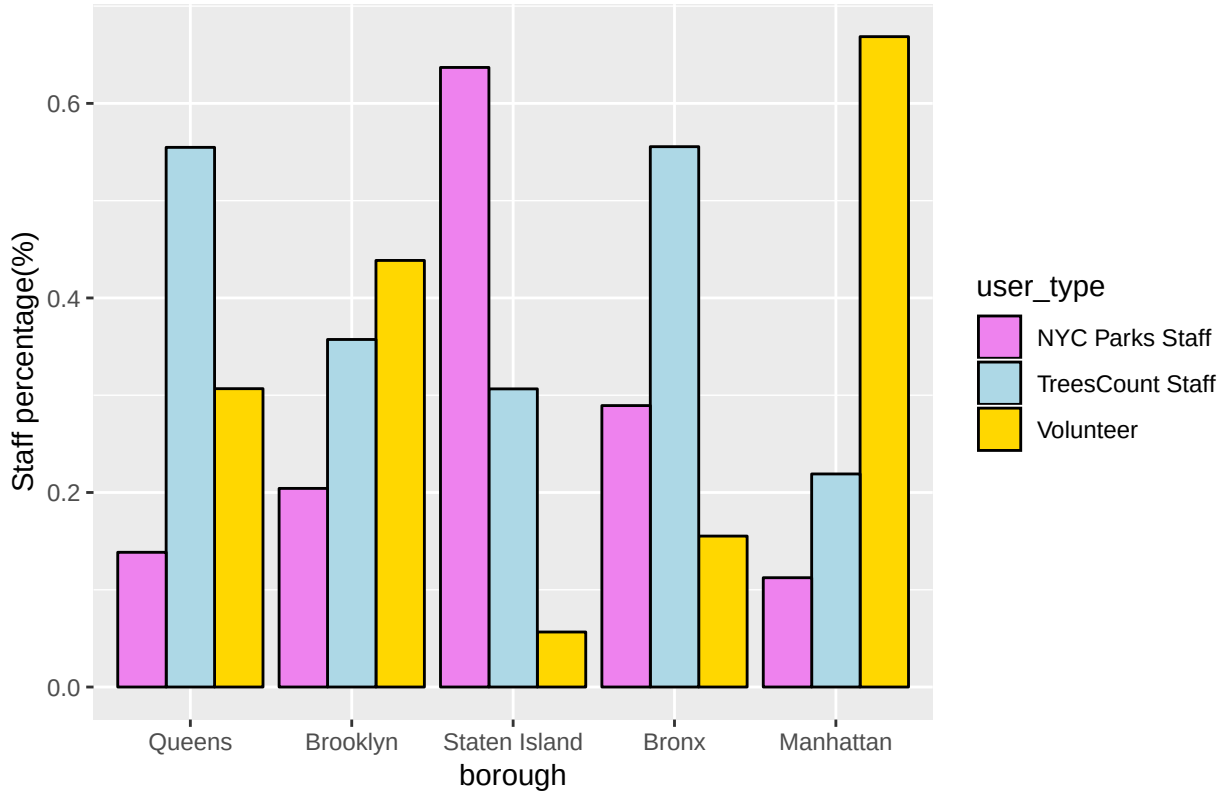
Data Biases Check

We may consider if the "user_type" (who count the trees) will influence the count result? In comnnon sense, the "Tree count staff" and "NYC Parks staff" are more professional and provide more reliable count results, while the "Volunteer" may provide a higher error rate result.

Plot the grouped bar plot to explore the distribution of street trees of each user_type based on borough. "User_type" describes the category of user who collected this tree point's data. The order of bough in each group is based on the descending order of total count numbers obtatined in the above section.

```
Tree$borough<-factor(Tree$borough,levels=c("Queens", "Brooklyn",
                                           "Staten Island", "Bronx","Manhattan"))
ggplot(Tree, aes(x = borough,fill=user_type)) +
   geom_bar(aes( y=..count../tapply(..count.., ..x.. ,sum)[..x..]),
            position="dodge" ,color="black") +
  scale_fill_manual(values = c("violet","lightblue","gold"))+ylab(
    "Staff percentage(%)") + ggtitle("Fig 4. Bar Chart: User_type distribution per borough")
```

Fig 4. Bar Chart: User_type distribution per borough



From the grouped bar plot above, the distribution of stree trees of different user_type based on borough are compared as follows:

1)The user_type has different staff percentage reatio for five borough, which implies that the user_type ratio might have a relationship with the final TreeCount results;

2)"Queens", the borough with most street trees counts, has its biggest user_type as "TreesCount Staff";

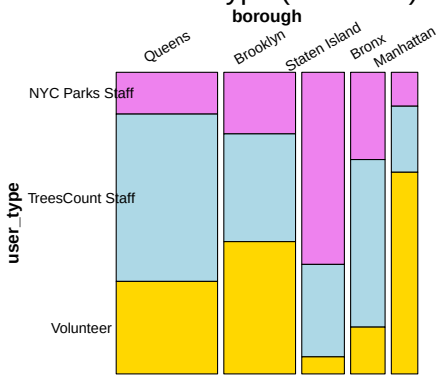
3)"Manhatten", the borough with the least street trees, has its smallest user_type as "Volunteer".

According to the observations above, the user_type did show somewhat relationship with the final TreeCount result. To investigate the possible relationship between two categorical variables, "user_type" and "TreeCount", the mosaic plot was generated below:

```
library(vcd)
library(grid) # needed for gpar
fillcolors <- c("violet","lightblue","gold")
Tree$borough<-factor(Tree$borough,levels=c("Queens", "Brooklyn", "Staten Island",
                                           "Bronx","Manhattan"))
vcd::mosaic( user_type ~ borough, Tree, gp = gpar(fill = fillcolors),
             main="Fig 5. Mosaic Plot: User Type (Recorder) against Borough",
             direction = c("v", "h"),tl_labels = c(TRUE, TRUE),
             labeling = labeling_border(gp_labels = gpar(fontsize = 10),
                                        gp_varnames = gpar(fontsize = 12,fontface = 2),
                                        rot_labels = c(30,0, 0, 0),
                                        rot_varnames = c(0,0,0,90),
                                        offset_varnames = c(0.7,0,0,3.0),
                                        offset_labels=c(0.5,0,0,1),
                                        pos_labels = c("center", "center",
```

```
                                                    "left", "center")))
```

Fig 5. Mosaic Plot: User Type (Recorder) against Borough



The mosaic plot above show the user_type ratio for each borough, and it was in the descending order of the TreeCount. In the above grouped bar plot, we can reconfirm our observations above:“Queens”, the borough with most street trees counts, has its biggest user_type as “TreesCount Staff”; “Manhatten”, the borough with the least street trees, has its smallest user_type as “Volunteer”. It’s also observed that “Manhattan” had the most percentage ratio of “Volunteer” user_type, which might as a result of the volunteers’ perference to count trees in Manhattan.

Therefore, we can conclued that

1)User_type, the category of user who collected this tree point’s data, had different percentage ratio for each borough;

2)Different user_type ratio had influence on the final TreeCount result, as “NYC Parks Staff” and “TreesCount Staff” with more professional knowledge would provide a more reliable data, while “Volunteer” with less experience would provide more mistakes.

Above all, different user_type had different recording criteria, and the five boroughs did not have the identical user_type percentage ratio, which implies that occurance of bias in the given NYC Census dataset.

MAIN ANALYSIS

Data Cleaning Process

In terms of the data cleaning, the two main processes were applied. Firstly, since there were 45 features in original dataset, which was overloaded for our data exploratory, we subsetted this original dataset into 12 columns with our own concerned features mentioned in data description part.

After subsetting data, as mentioned in data quality part, the info for those stumped and dead trees were despretely lost where we already set them as NA. The percentage of these trees were about 5%, which was tolarable for our analysis. Therefore, these portions of data was removed as demanded for some parts use. This value will be applied to the next few parts.

```
TreeNonNA <- subset(Tree, !(Tree$status %in% "Stump") & !(Tree$status %in% "Dead"))
```

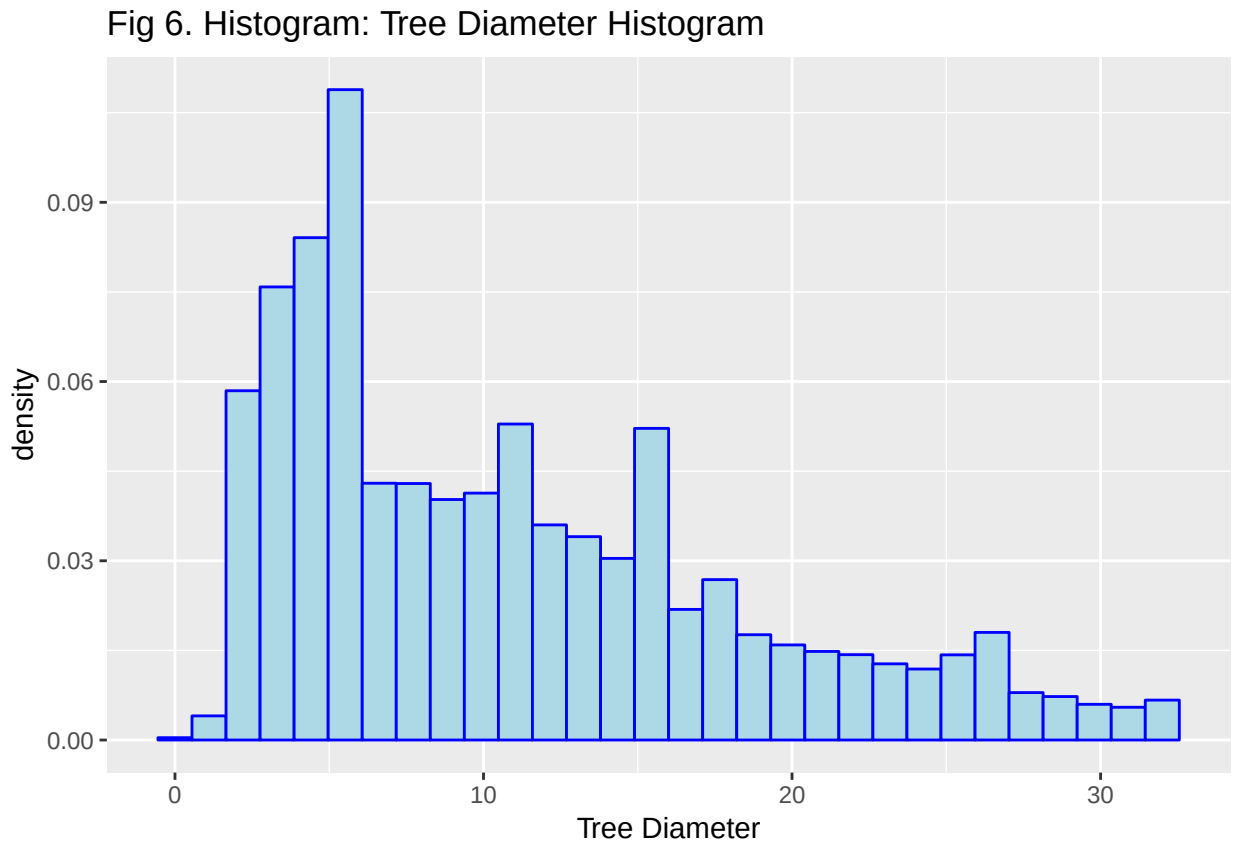
Specifically for tree diameters, the outliers could be removed due to the low proportion and trivial concerns. This value will be applied to the next few parts.

```
TreeNoOut <- subset(Tree, Tree$tree_dbh <= 32.5)
```

Tree Properties Data Analysis

Tree Diameters

```
ggplot(subset(TreeNoOut,!(status %in% "Stump")))+geom_histogram(
  aes(x=tree_dbh,y=..density..),
  fill="lightblue",color="blue")+xlab("Tree Diameter")+labs(
    title="Fig 6. Histogram: Tree Diameter Histogram")
```

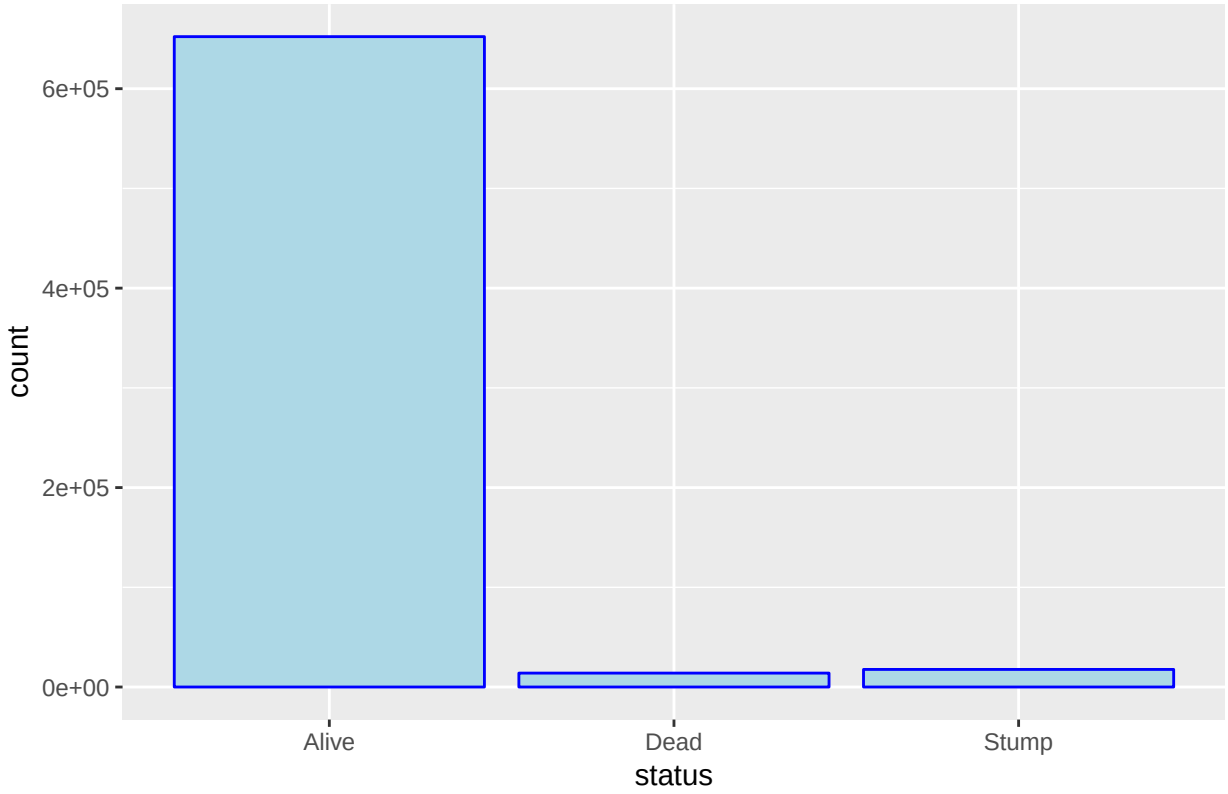


According to the histogram, we could see that the most diameters fall in small sizes. The largest count value is 5-6, about 10%. The distribution appears to be positive skewed.

Status

```
ggplot(Tree, aes(status)) +
geom_histogram(stat='count',fill="lightblue",color="blue") +
ggtitle("Fig 7. Bar Chart: Three status(alive,dead,stump) counts for all trees")
```

Fig 7. Bar Chart: Three status(alive,dead,stump) counts for all trees



We can see the count number of alive trees is much higher than dead trees and trees with stump conditions. What is more, the count number of dead tree is roughly equals to trees with stump condition.

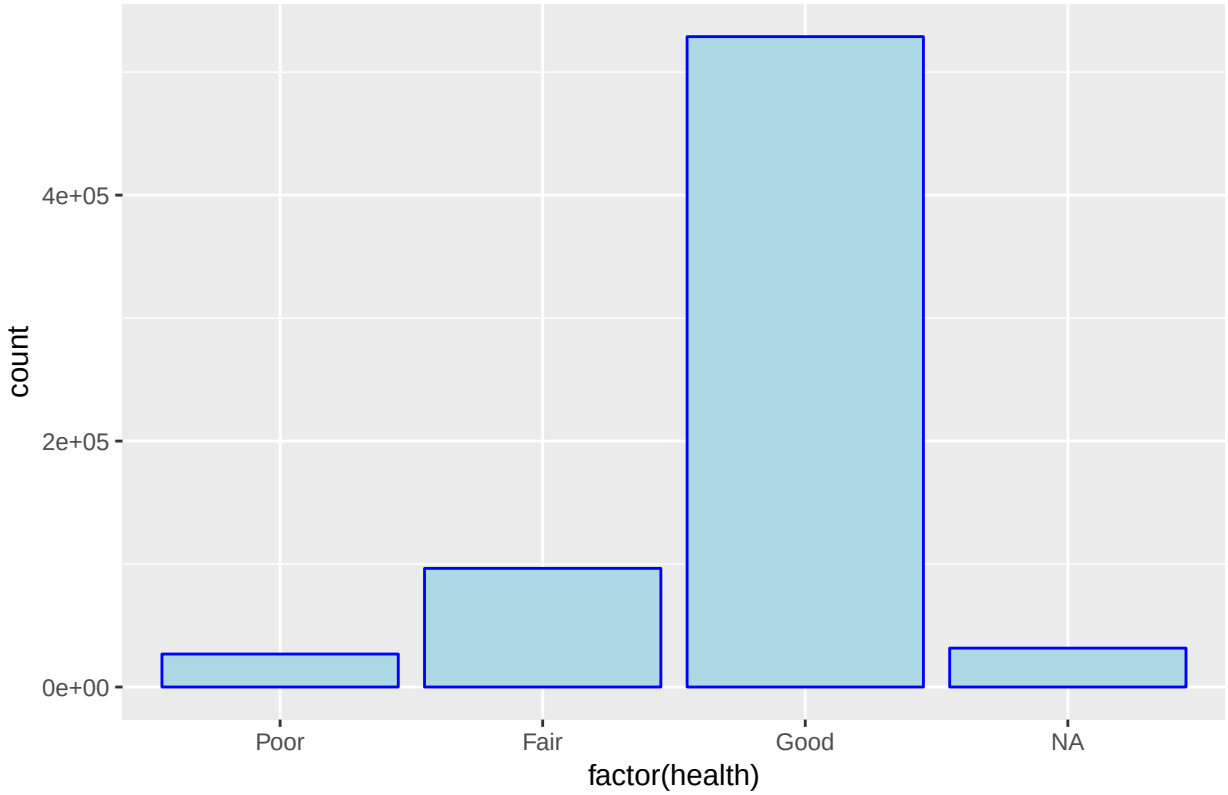
Health

```
Tree$health <- factor(Tree$health, ordered = TRUE, levels <- c("Poor", "Fair", "Good"))
summary(Tree$health)
```

```
##   Poor   Fair   Good   NA's
##  26818  96504 528850  31616
```

```
g_health <- ggplot(Tree, aes(x = factor(health)), width=0.5) +
              geom_bar(fill="lightblue",color="blue") +
              ggtitle("Fig 8. Bar Chart:  NYC Street Tree Health Condition") +
              theme_get()
g_health
```

Fig 8. Bar Chart: NYC Street Tree Health Condition



As figure 1 shows, the trees in good condition is about 81.1% of all the alive trees, fair condition trees is about 14.8% and poor condition is about 4.1%. The trees without any health condition data are dead (NAs). The dead and poor condition trees are about the same amount.

Species

```
summary(Tree$spc_common)
```

```
##       London planetree            honeylocust           Callery pear
##                  87014                  64264                  58931
##               pin oak          Norway maple      littleleaf linden
##                  53185                  34189                  29742
##                cherry       Japanese zelkova                 ginkgo
##                  29279                  29258                  21024
##               Sophora             red maple             green ash
##                  19338                  17246                  16251
##        American linden          silver maple              sweetgum
##                  13530                  12277                  10657
##       northern red oak         silver linden          American elm
##                   8400                   7995                   7975
##                 maple       purple-leaf plum        swamp white oak
##                   7080                   6879                   6598
##     crimson king maple           Chinese elm 'Schubert' chokecherry
##                   5923                   5345                   4888
##    Japanese tree lilac         eastern redbud        golden raintree
```

```
##                 4568                 3801                 3719
##           crab apple  Kentucky coffeetree           willow oak
##                 3527                 3364                 3184
##         dawn redwood             hawthorn         sugar maple
##                 3020                 2988                 2844
##       sycamore maple                  ash         hedge maple
##                 2731                 2609                 2550
##     common hackberry         sawtooth oak         Amur maackia
##                 2382                 2244                 2197
##    European hornbeam           Amur maple         serviceberry
##                 2099                 2049                 2032
##         black locust           white oak         English oak
##                 1784                 1686                 1644
##         Siberian elm   flowering dogwood   American hornbeam
##                 1595                 1552                 1517
##       Schumard's oak         scarlet oak           black oak
##                 1487                 1465                 1327
##         bald cypress            mulberry       Japanese maple
##                 1261                 1157                 1130
##           white ash     eastern redcedar       horse chestnut
##                 1121                 1101                 1096
##  American hophornbeam         tulip-poplar     Cornelian cherry
##                 1081                 1076                 1066
##         shingle oak   hardy rubber tree         katsura tree
##                 1049                  915                  911
##       tree of heaven            magnolia         black cherry
##                  756                  699                  607
##         river birch             catalpa          paper birch
##                  565                  551                  535
##             bur oak  Kentucky yellowwood   Chinese tree lilac
##                  515                  479                  462
##         crepe myrtle   Japanese hornbeam   Japanese snowbell
##                  442                  426                  392
##  Atlantic white cedar       Norway spruce   cockspur hawthorn
##                  355                  355                  341
##           arborvitae     Turkish hazelnut       kousa dogwood
##                  328                  317                  302
##         silver birch        black walnut                 pine
##                  300                  295                  289
##            blackgum      weeping willow      pagoda dogwood
##                  288                  282                  280
##     Persian ironwood  eastern cottonwood      American beech
##                  277                  276                  273
##         empress tree  Chinese fringetree two-winged silverbell
##                  245                  234                  221
##      paperbark maple      Oklahoma redbud               spruce
##                  220                  219                  202
##           white pine       Amur cork tree              (Other)
##                  202                  183                 3259
##                 NA's
##                31619
```

We find the number of species more than 100, and the range of species from highest count number to the lowest count number is 86,831.Therefore, we want to subset of the species which total number of counts with

those species has large proportion on our total species count number. Then we select top 10 species without choosing species is blank(missing data with species).In the following analysis, for tree species varaible, we just focus on top ten tree species.

```
Tree$species<-Tree$spc_common
for (i in levels(Tree$spc_common)){
  if ((i %in% c("London planetree", "honeylocust","Callery pear",
                "pin oak","Norway maple","littleleaf linden",
  "cherry","Japanese zelkova","ginkgo","Sophora"))==FALSE){
    Tree<-Tree %>%
      mutate(species = fct_recode(species,OTHER=i))
  }
}
trees<-subset(Tree, species != "OTHER")
trees$species<-factor(trees$species)
```

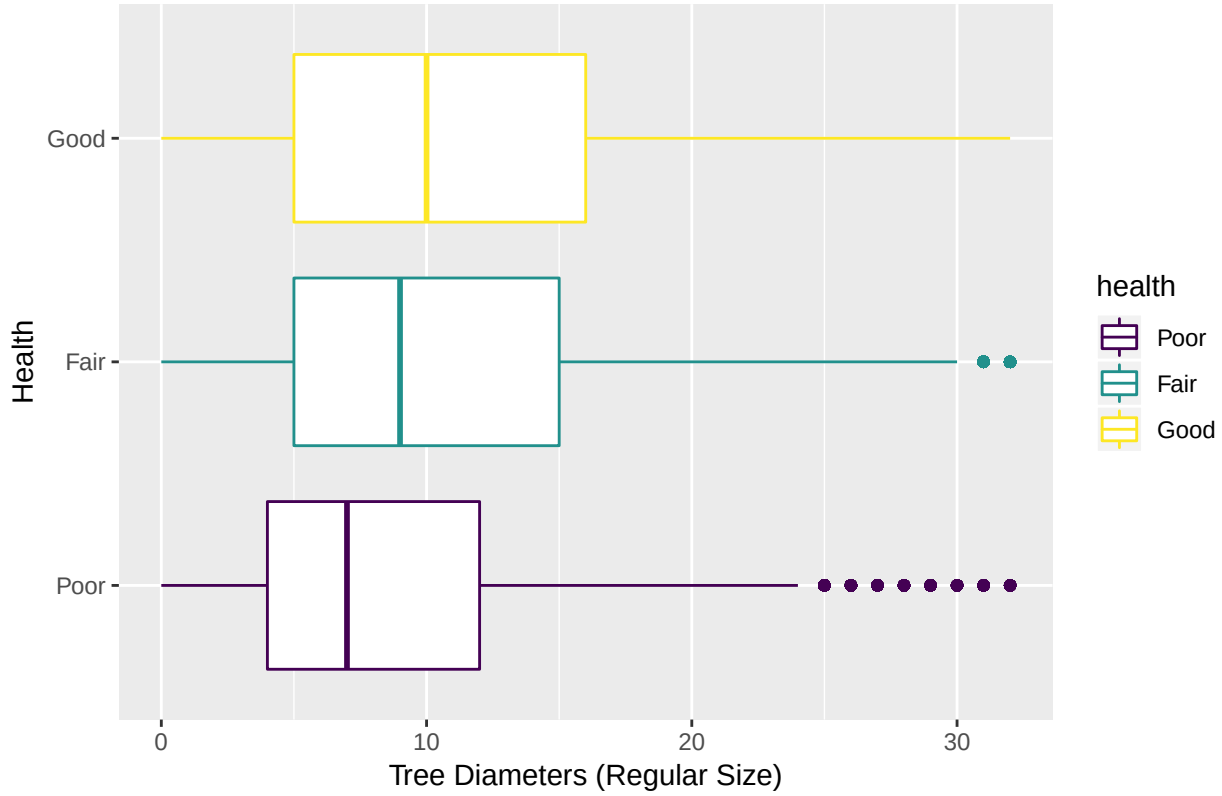
Tree Properties Dependencies

Health vs. Tree DBH

We draw the boxplot in terms of the health status. As a common sense, the larger the diameters are, the stronger the trees are. Therefore, the trees with good and fair status have the larger diameters.

```
healthbox<-ggplot(subset(Tree,tree_dbh <= 32.5 & !(health %in% NA) ))+geom_boxplot(aes(x=reorder(health
                                                                                          y=tree_dbh,group=
  coord_flip() + xlab("Health") +ylab(
    "Tree Diameters (Regular Size)")+labs(title="Fig 9. Boxplot: Tree Diameter vs. Health")
healthbox
```

Fig 9. Boxplot: Tree Diameter vs. Health



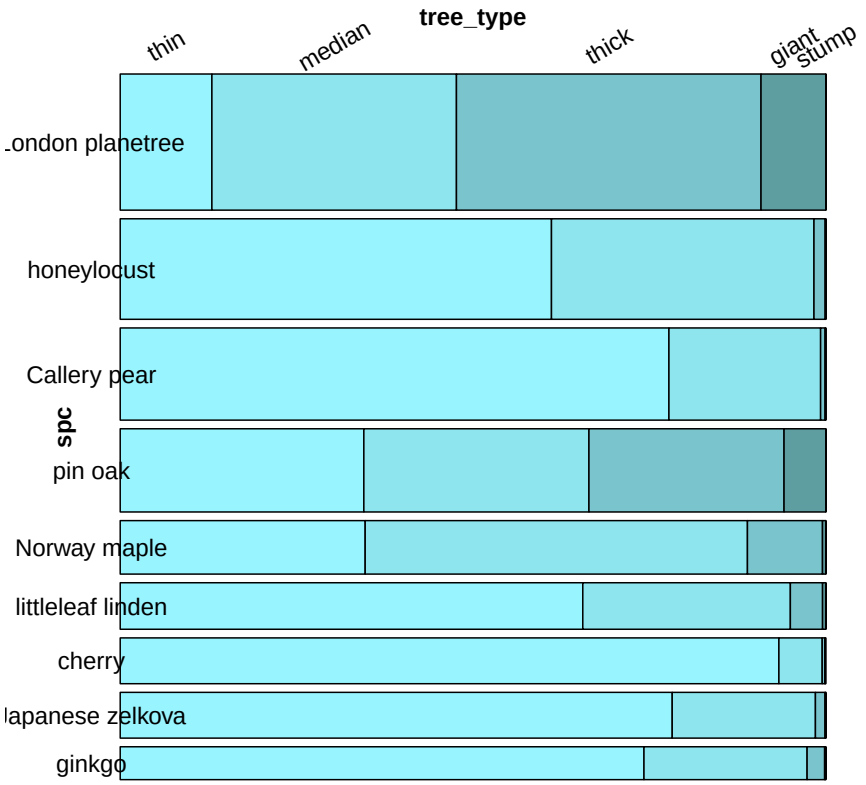
Species vs. Tree DBH

```
library(vcd)
library(dplyr)
group<-Tree %>%
      group_by(spc_common) %>%
      summarise(count=n())
group[order(-group$count),]
```

```
## # A tibble: 133 x 2
##    spc_common        count
##    <fct>             <int>
##  1 London planetree  87014
##  2 honeylocust       64264
##  3 Callery pear      58931
##  4 pin oak           53185
##  5 Norway maple      34189
##  6 <NA>              31619
##  7 littleleaf linden 29742
##  8 cherry            29279
##  9 Japanese zelkova  29258
## 10 ginkgo            21024
## # ... with 123 more rows
```

```
Tree[['spc']]<-Tree$spc_common
for (i in levels(Tree$spc_common)){
  if ((i %in% c("London planetree", "honeylocust","Callery pear",
                "pin oak","Norway maple","littleleaf linden","cherry","Japanese zelkova","ginkgo"))==FA
      Tree<-Tree %>%
          mutate(spc = fct_recode(spc,OTHER=i))
  }
}
Tree$tree_type <- Tree$tree_dbh
Tree$tree_type <- ifelse(Tree$tree_dbh<=11&Tree$tree_dbh>0,"thin",Tree$tree_type)
Tree$tree_type <- ifelse(Tree$tree_dbh>11&Tree$tree_dbh<22,"median",Tree$tree_type)
Tree$tree_type <- ifelse(Tree$tree_dbh>=22&Tree$tree_dbh<=32.5,"thick",Tree$tree_type)
Tree$tree_type <- ifelse(Tree$tree_dbh>32.5,"giant",Tree$tree_type)
Tree$tree_type <- ifelse(Tree$tree_dbh==0,"stump",Tree$tree_type)
# mosaic(tree_type ~ spc, Tree,rot_labels = c(30, 0, 0, 0))
Tree2<- Tree %>% filter(`spc` %in% c("London planetree",
                                     "honeylocust","Callery pear",
                                     "pin oak","Norway maple",
                                     "littleleaf linden","cherry",
                                     "Japanese zelkova","ginkgo"))
Tree2$spc <- factor(Tree2$spc,
                    levels = c("London planetree", "honeylocust",
                               "Callery pear","pin oak","Norway maple",
                               "littleleaf linden","cherry",
                               "Japanese zelkova","ginkgo"))
Tree2$tree_type <- factor(Tree2$tree_type,
                    levels = c("thin", "median","thick","giant","stump"))
fillcolors=c("cadetblue1","cadetblue2","cadetblue3","cadetblue","cadetblue4")
mosaic(tree_type ~ spc, Tree2,rot_labels = c(30, 0, 0, 0),gp=gpar(fill=fillcolors),
       main="Fig 10. Mosaic Plot: Species vs. Tree Diameter")
```

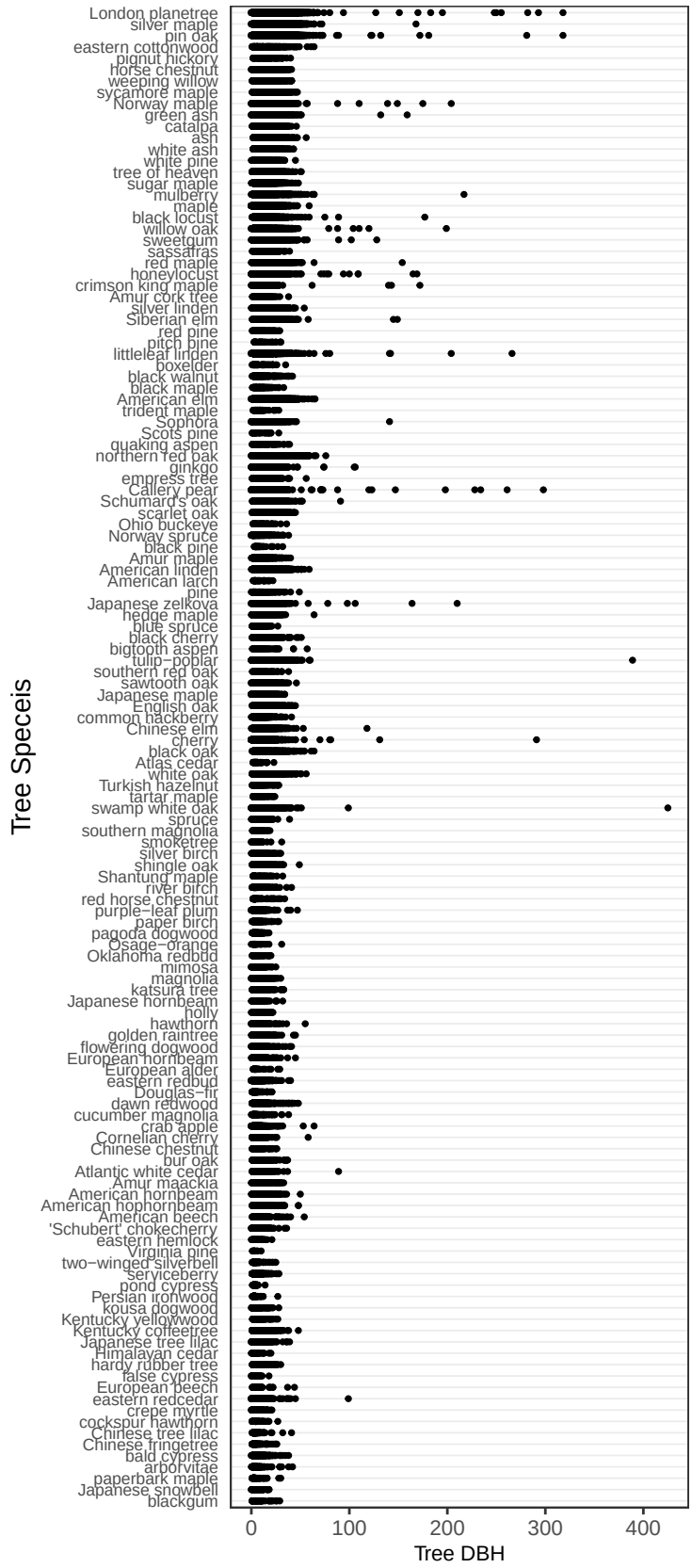
Fig 10. Mosaic Plot: Species vs. Tree Diameter



According to the mosaic plot above, it was reasonable to observe that the diameters of the trees were highly depending on the tree species. London planetree, ocuppying the highest portion of the street trees, tended to be larger in diameters. While the second large amount of the tree species, honeylocust, were concentrating in thick category.

```
theme_dotplot <- theme_bw(18) +
    theme(
          axis.text.y = element_text(size = rel(.75)),
          axis.ticks.y = element_blank(),
          axis.title.x = element_text(size = rel(.75)),
          panel.grid.major.x = element_blank(),
          panel.grid.major.y = element_line(size = 0.5),
          panel.grid.minor.x = element_blank())

ggplot(subset(Tree,!(spc_common %in% NA) & !(status %in% "Stump")))+geom_point(aes(x=tree_dbh,y=fct_reo
  spc_common,tree_dbh)))+theme_dotplot+ggtitle(
    "Fig 11. Cleveland Dotplot: Tree Species vs. Tree Diameter")+labs(
      x="Tree DBH", y="Tree Speceis")
```

Fig 11. Cleveland Dotplot: Tree S



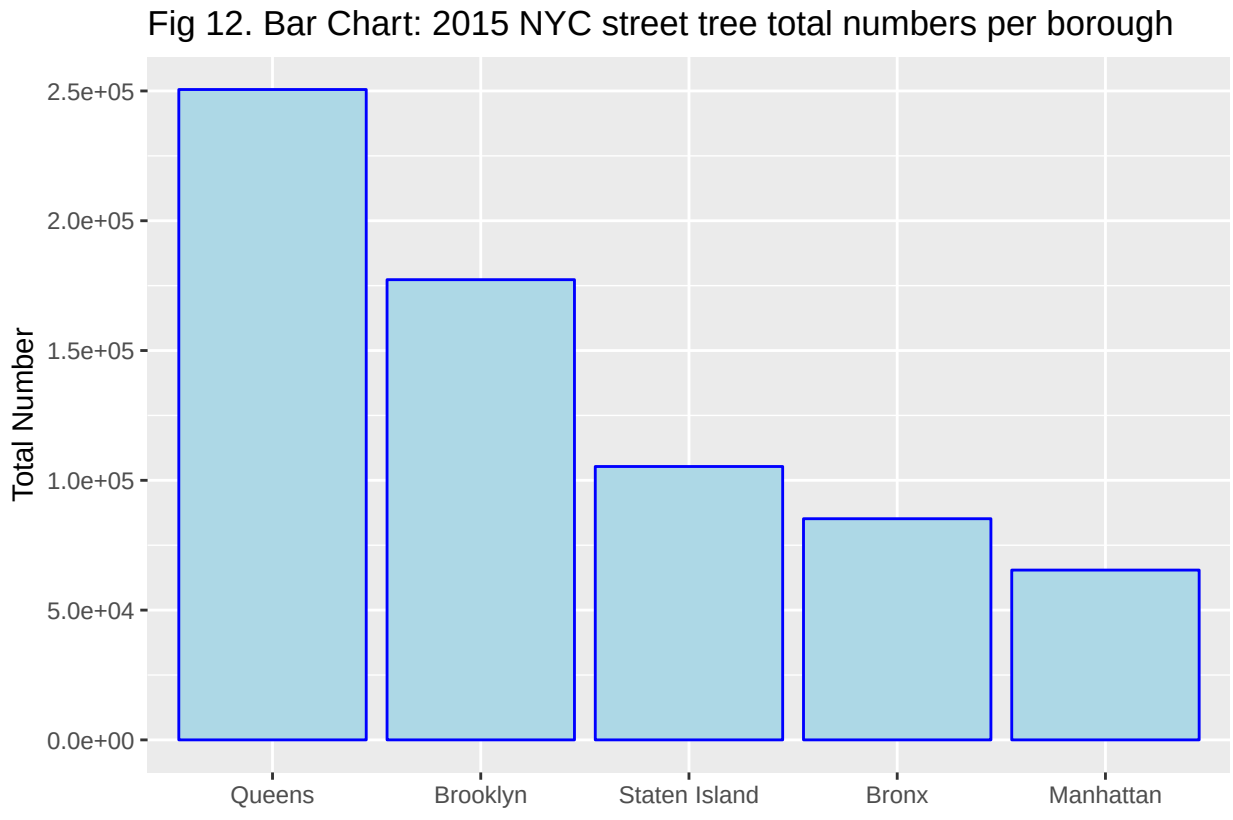
According to the dotplot shown above, the trees distributions for each species was relatively spreaded. For those species with large amount, there were very large spreading distributs, eg. London planetree, ranging from lowest to the highest values.

Data Attributes Dependencies

Tree Geographic Counts

First, to generate a overview of NYC street tree counts in each borough, the barchart was plotted with y-axis show the total count of street trees for each borough in NYC 2015 and the x-axis show the borough name. For better comparision of street tree counts in different borough, the plot was shown in decreasing order of total counts.

```
library(scales)

Tree$borough<-factor(Tree$borough,levels=c("Queens", "Brooklyn",
                                           "Staten Island", "Bronx","Manhattan"))
b<-Tree %>% group_by(borough) %>% summarise(n=n())
# plot the barchart of total tree numbers by borough with a decreasing order
ggplot(b,aes(x=fct_reorder(borough,n,.desc=TRUE),y=n))+
  geom_bar(color="blue",fill="lightblue",stat="identity")+
  scale_y_continuous(labels = scientific)+
  ggtitle("Fig 12. Bar Chart: 2015 NYC street tree total numbers per borough")+labs(
    x="",y="Total Number")
```



According to the bar chart above, the total number order of NYC borough is 'Queens', 'Brooklyn', 'Staten

Island', 'Bronx' and 'Manhattan'. In year 2015, among all five boroughs, "Queens" had the largest number of street trees around 250,000, while "Manhattan" has the smallest number of street trees around 70,000, which is less than 1/3 of "Queens". Thus, the barchart above demonstrated the distinct differeence of street trees counts between five boroughs.

In Figure 1 above, it demonstrated the distinct differeence of street trees counts between five boroughs. Thus, it's interesting to furthur explore what cause the difference between the five boroughs, and the two most obvious are "Land Area" and "Population". Consideirng the difference of "Land Area" and "Population" between five boroughs, we will try to recognize whether there exist relationships between these three variables: "borough", "Land Area", and "Population".

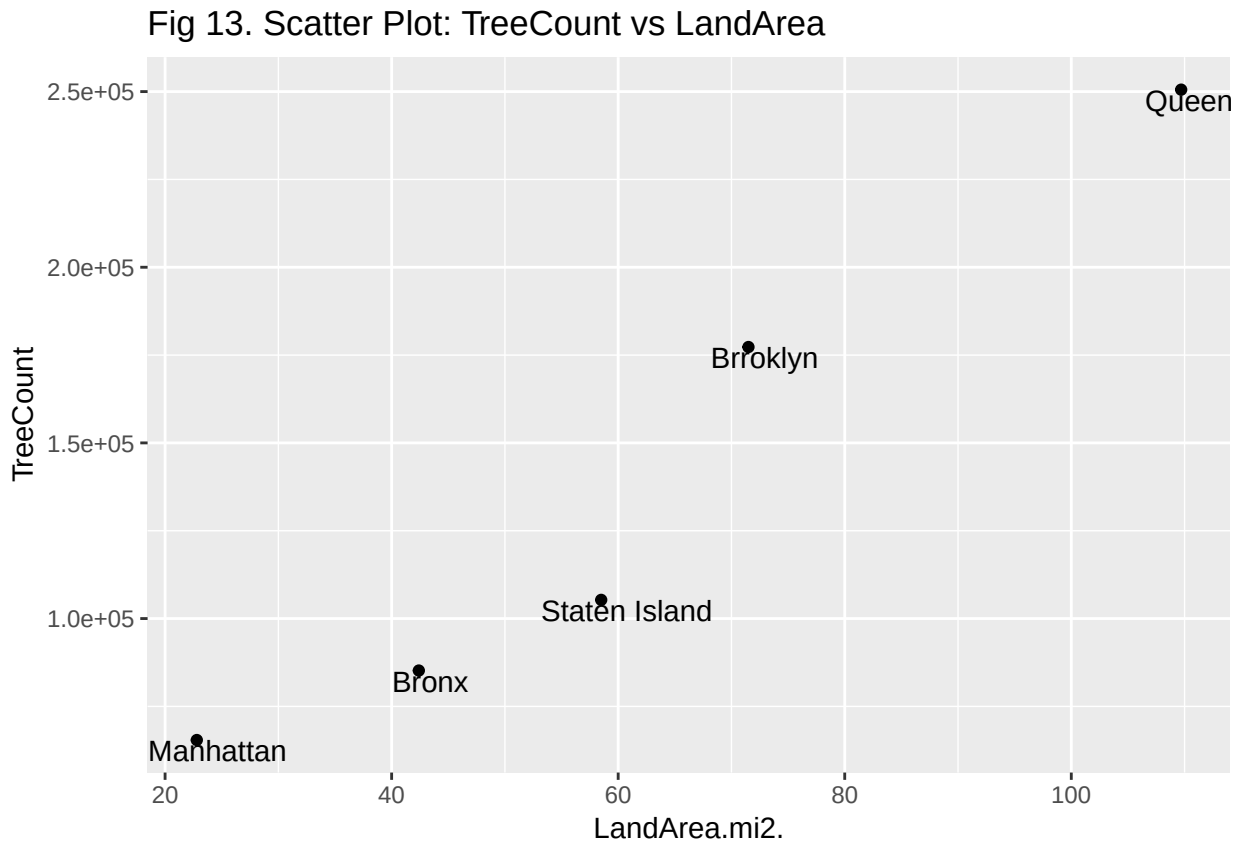
The data of "borough", "Land Area", and "Population" is shown in the table below:

```
borough_info <- data.frame("Borough" = c("Queens","Brroklyn",
                                         "Staten Island","Bronx",
                                         "Manhattan"), "LandArea(mi2)" =
                             c(109.7,71.5,58.5,42.4,22.8), "Population" = c(2230722,2504700,468730,1385
borough_info$LandArea.mi2.=as.numeric(borough_info$LandArea.mi2.)
borough_info$Population=as.numeric(borough_info$Population)
borough_info$TreeCount=b$n
borough_info
```

```
##         Borough LandArea.mi2. Population TreeCount
## 1        Queens         109.7    2230722    250551
## 2      Brroklyn          71.5    2504700    177293
## 3 Staten Island          58.5     468730    105318
## 4         Bronx          42.4    1385108     85203
## 5     Manhattan          22.8    1585873     65423
```

To investigate the possible relationship between variables "TreeCount", "Land Area", and "Population" based on borough, two scatter plot "TreeCount vs LandArea" and "TreeCount vs Population" were generated below:

```
ggplot(borough_info, aes(x=LandArea.mi2., y=TreeCount)) +scale_y_continuous(
  labels = scientific)+ geom_point()+geom_text(label=borough_info$Borough,vjust = 1,hjust=0.35)+ggtitle
```

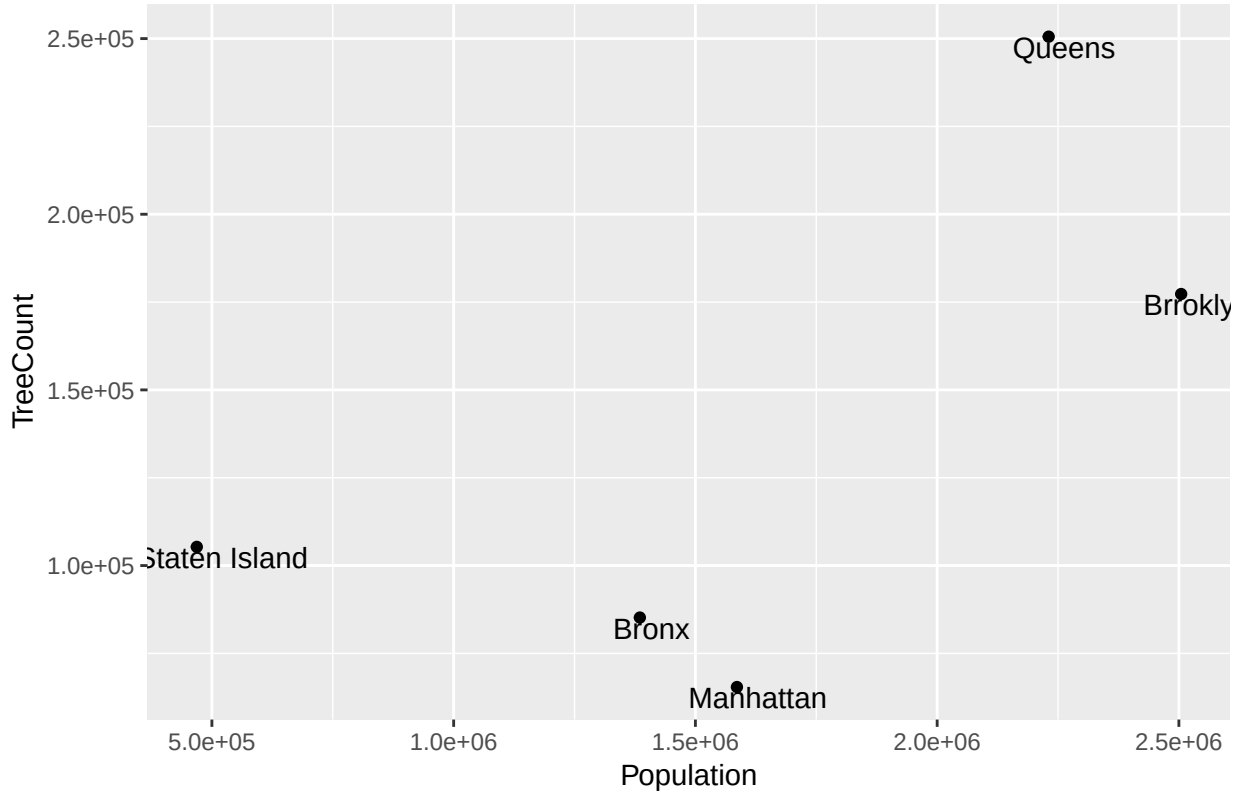


In Figure 2 above, the data points make a straight line going from the origin out to high x- and y-values, theerfore the two variables are said to have a positive correlation. It's reasonable to concluded that the TreeCount difference between five boroughs are affected by the their LandArea.

Next, the scatterplot between "TreeCount" and "Population" based on borough:

```
ggplot(borough_info, aes(x=Population, y=TreeCount)) +scale_y_continuous(
  labels = scientific)+ geom_point()+geom_text(label=borough_info$Borough,vjust = 1,hjust=0.35)+scale_x
    "Fig 14. Scatter Plot: TreeCount vs Population")
```

Fig 14. Scatter Plot: TreeCount vs Population



In Figure 3 above, the data points also clustered in a band running from lower left to upper right, indicating a positive correlation between "TreeCount" and "Population". However, as the data points had larger bias from a straight line compared to Figure 2, we can concluded that the "TreeCount"" of five boroughs had positive correlation with their "Population", but not as strong as "LandArea".

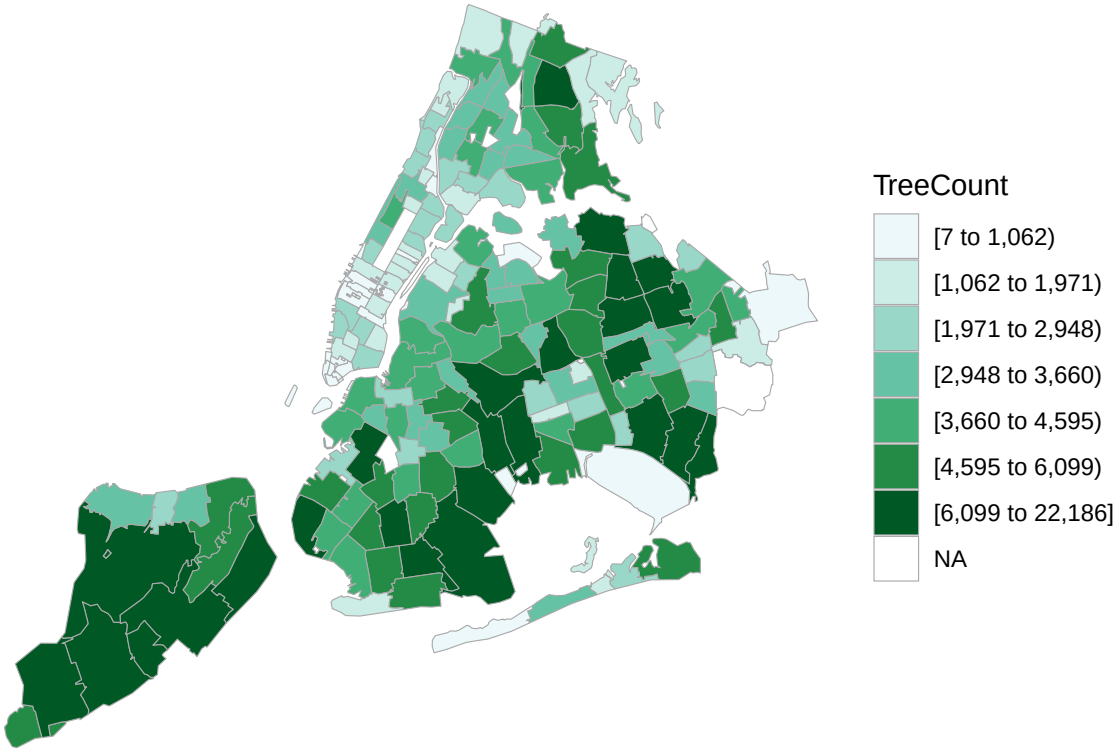
In the analysis above, we explore the TreeCount of NYC street trees based on borough, figuring out the difference of TreeCount in five boroughs, and the relationship between certain variables and borough. This provided a basic idea about the distribution of NYC street trees for five borough.

In this section, instead of using the borough as a border to demonstrate count data, "postcode" of each subarea was applied as boudaries in a choroplethic map.

```
library(devtools)
library(choroplethr)
library(choroplethrZip)
library(ggplot2)
data("zip.regions")
tree_summary<-Tree %>% group_by(postcode) %>% summarise(n=n())
tree_summary<-tree_summary[tree_summary$postcode %in% zip.regions$region,]
tree_summary<-rename(tree_summary,"region"="postcode","value"="n")
tree_summary$region<-as.character(tree_summary$region)
tree_summary$value<-as.numeric(tree_summary$value)
nyc_fips = c("36005", "36047", "36061", "36081", "36085")

choro = ZipChoropleth$new(tree_summary)
choro$title = "Fig 15. Choropleth: 2015 NYC Street TreeCount by ZipCode"
choro$ggplot_scale = scale_fill_brewer(name="TreeCount", palette=2, drop=FALSE)
choro$set_zoom_zip(state_zoom=NULL,county_zoom = nyc_fips, msa_zoom=NULL, zip_zoom=NULL)
```

```
choro$render()
```

Fig 15. Choropleth: 2015 NYC Street TreeCount by ZipCode



The choropleth map above provided an easy way to visualize how the TreeCount varies across the subregion in NYC by the zipcode. To provide a better perceptual intuition, the map used "green" to represent the TreeCount numbers, and the darker the green color the higher the TreeCount data.

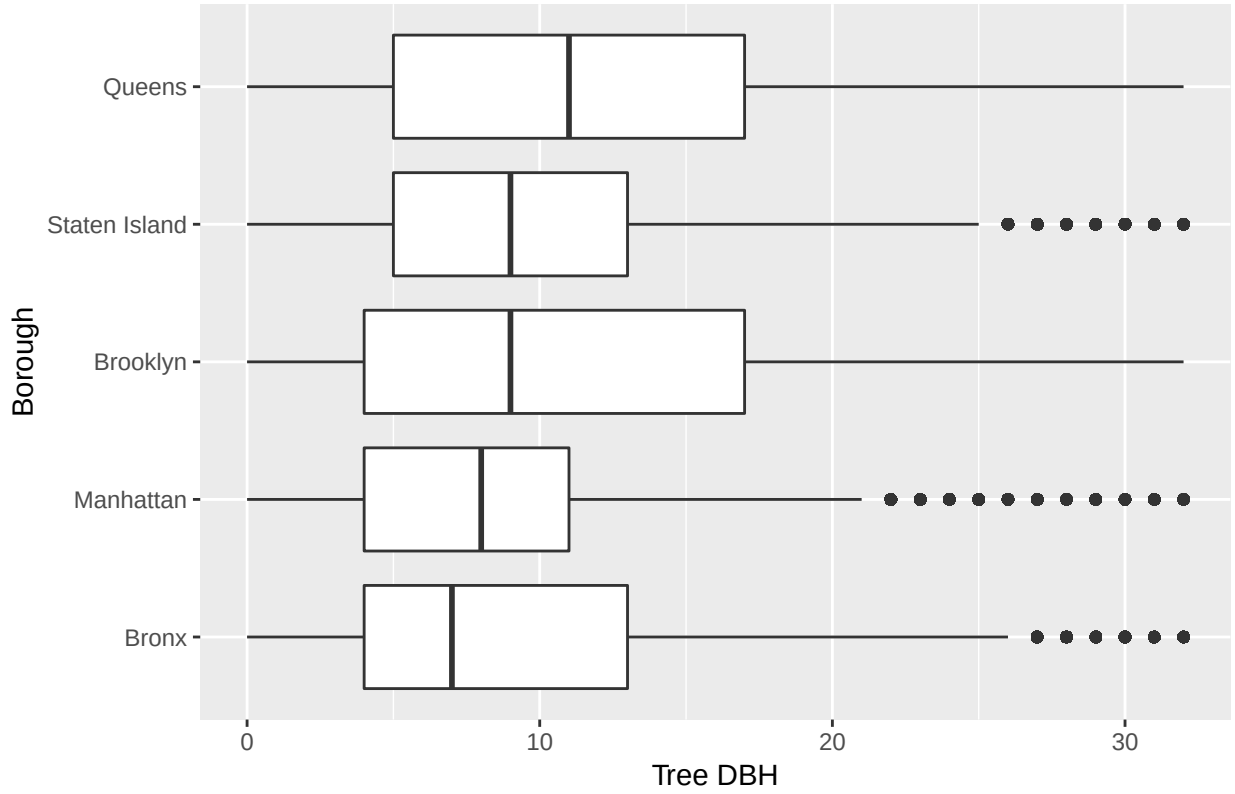
This choropleth map show the diistribution of street trees in in greater detail,and the visulization based on the map and color density provide a easier way to analysis the dataset.

Borough vs. Tree DBH

We draw the boxplot of the diameters faceted by the borough.

```
BoroughDia <- ggplot(aes(x=reorder(borough,tree_dbh,FUN=median),
                         y=tree_dbh,group=paste(borough)),
                     data=TreeNoOut)+geom_boxplot()+coord_flip()+labs(
                       y= "Tree DBH", x = "Borough")
BoroughDia+ggtitle("Fig 16. Boxplot: Borough vs Tree DBH")
```

Fig 16. Boxplot: Borough vs Tree DBH



According to the boxplot, Queens has the largest median of diameters. Queens and Brooklyn have a wider range of the diameters.Bronx and Manhattan have smallest diameters maybe because the urbanlization of the boroughs.

Next, we categorized the regular sized (<=32.5) trees into three categories: thin, median, thick. The rest are stump and outliers (as giant). Being consistent with the results we got before, even Manhattan and Bronx have less trees, they own a large amount of the thinner trees b/c of the urbanlization.

```
type<-Tree %>%
      group_by(tree_type) %>%
      summarise(count=n())
type[order(-type$count),]
```

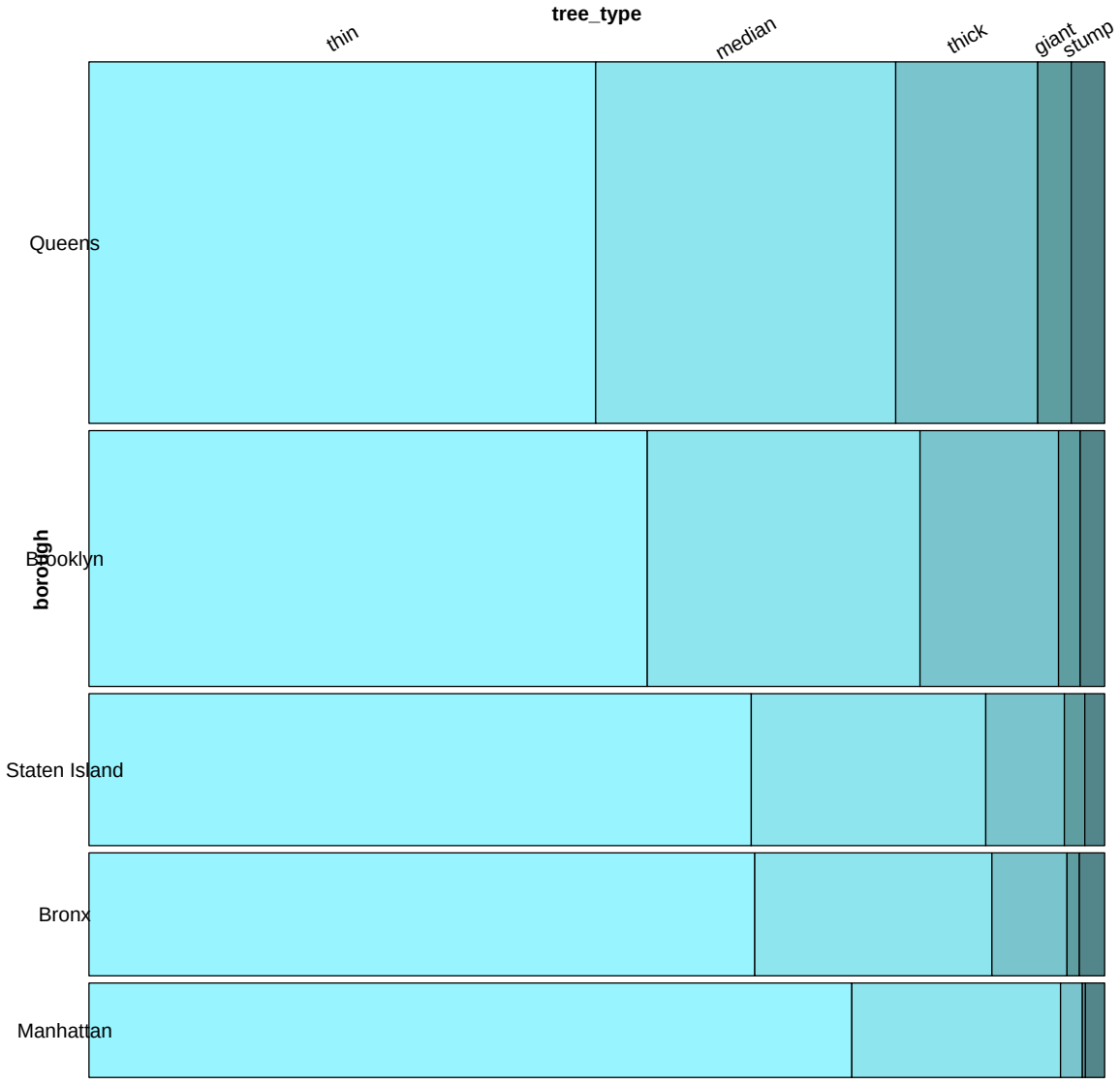
```
## # A tibble: 5 x 2
##   tree_type  count
##   <chr>      <int>
## 1 thin      396106
## 2 median    179279
## 3 thick      75066
## 4 stump      17932
## 5 giant      15405
```

```
Tree$tree_type <- factor(Tree$tree_type,
                         levels = c("thin", "median","thick","giant","stump"))
borough<-Tree %>%
      group_by(borough) %>%
      summarise(count=n())
borough[order(-borough$count),]
```

```
## # A tibble: 5 x 2
##   borough        count
##   <fct>          <int>
## 1 Queens        250551
## 2 Brooklyn      177293
## 3 Staten Island 105318
## 4 Bronx          85203
## 5 Manhattan      65423
```

```
Tree$borough <- factor(Tree$borough,
                       levels = c("Queens", "Brooklyn","Staten Island","Bronx","Manhattan"))
fillcolors=c("cadetblue1","cadetblue2","cadetblue3","cadetblue","cadetblue4")
mosaic(tree_type ~ borough, Tree,rot_labels = c(30, 0, 0, 0),gp=gpar(fill=fillcolors),
       main="Fig 17. Mosaic Plot: Borough vs. Tree Diameter")
```

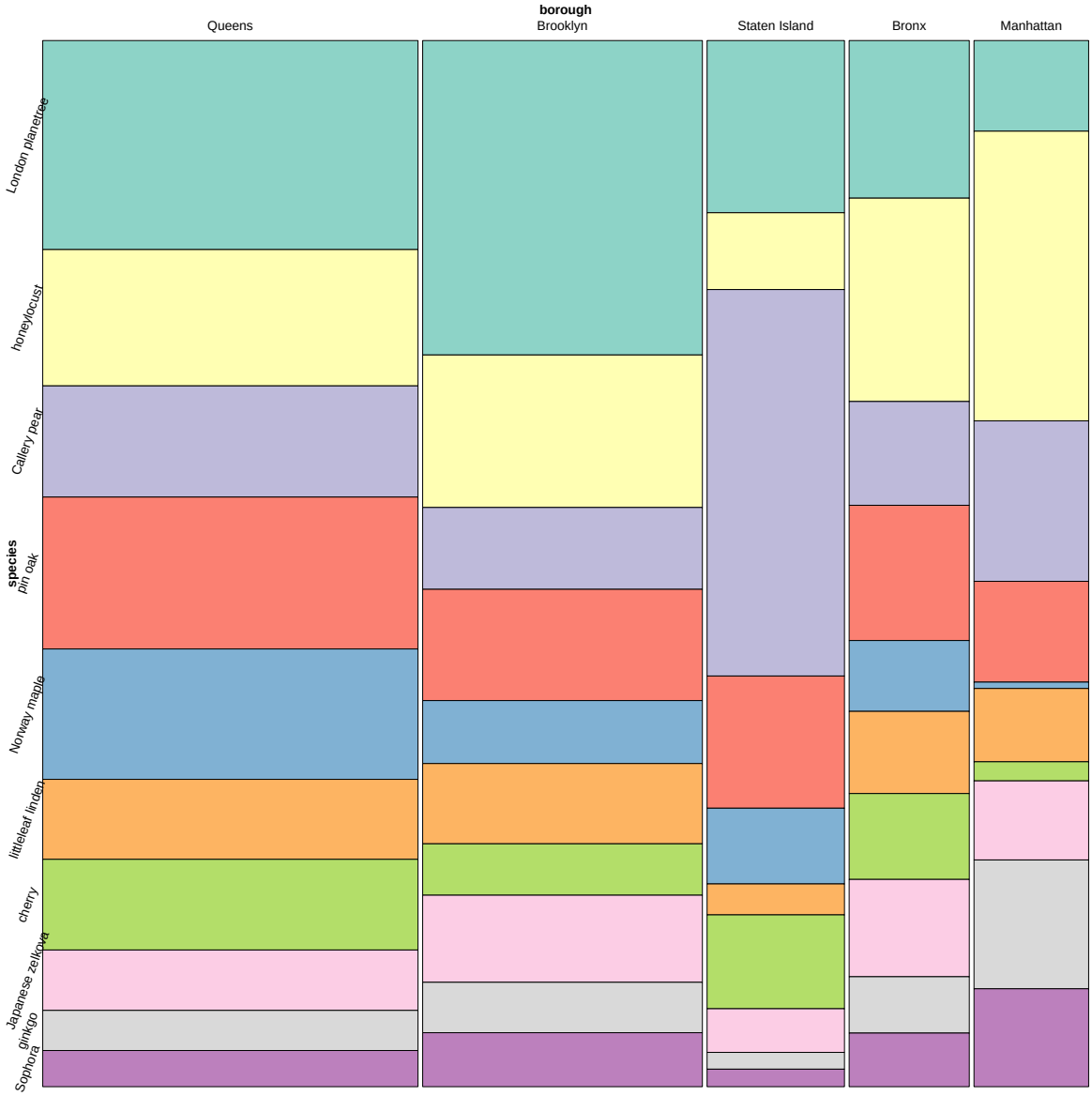
Fig 17. Mosaic Plot: Borough vs. Tree Diameter



Speceis vs. Borough

```
library(RColorBrewer)
fillcolors <- brewer.pal(10, "Set3")
orderlevel = c("Queens","Brooklyn","Staten Island","Bronx","Manhattan")
orderlevel1 = c("London planetree", "honeylocust","Callery pear",
                "pin oak","Norway maple",
                "littleleaf linden","cherry",
                "Japanese zelkova","ginkgo","Sophora")
trees$borough = factor(trees$borough, levels = orderlevel)
trees$species = factor(trees$species , levels = orderlevel1)
vcd::mosaic(species ~ borough, trees,
            direction = c("v", "h"),
            rot_labels=c(0,0,0,70),
            gp = gpar(fill = fillcolors),
main = "Fig 18. Mosaic Plot: Top 10 Count Number of Tree Species Depend on Different Boroughs")
```

Fig 18. Mosaic Plot: Top 10 Count Number of Tree Species Depend on Different Boroughs



We can see top ten count numbers of tree species' distribution depend on boroughs.

1)Top ten count numbers of tree species' distribution on Brooklyn and Queens are very similar. We consider the reason of this is that both two boroughs is living areas, so the damage rate from human to trees are very similar.

2)Top ten count numbers of tree species' distribution on Bronx and Manhattan are very similar. We consider the reason of this is that both two boroughs are very close, so the soil condition might be very similar .

3)Top ten count numbers of tree species' distribution on Staten Island is so different from other boroughs, the reason of this might be the island is isolated place, the damage rate from hunnam, soil condition and other conditions might be different from other boroughs.

Health vs. User Type

```
summary(Tree$user_type)
```

```
##  NYC Parks Staff TreesCount Staff        Volunteer
##           169986           296284           217518
```

```
healthcolors <- c("gray", "lightgreen", "darkgreen")
ggplot(Tree, aes(x = user_type, fill = health)) +
  geom_bar(position = "dodge") +
  scale_fill_manual(values = healthcolors, na.value = "black") +
  ggtitle("Fig 19. Group Bar Chart: Tree Count vs. Data Recorder
          Group by Health Status") + theme_gray()
```

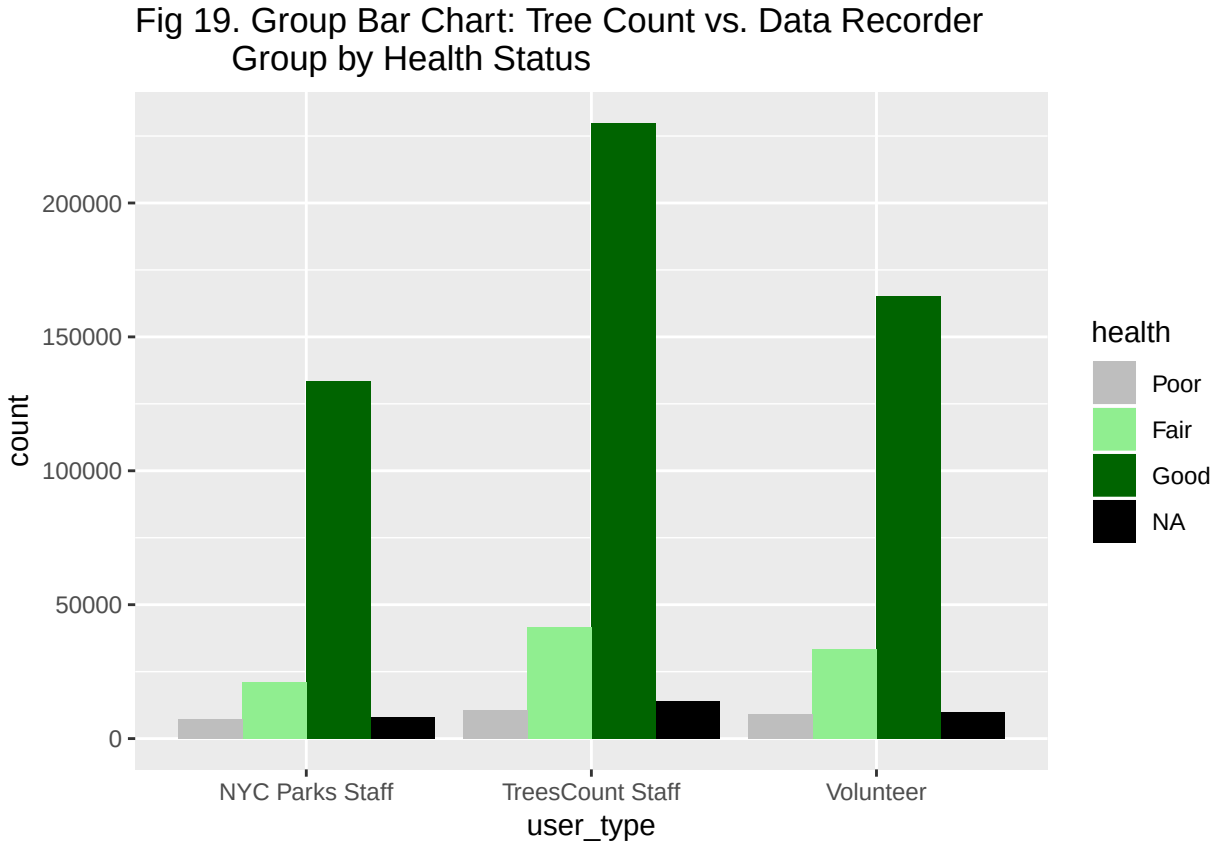


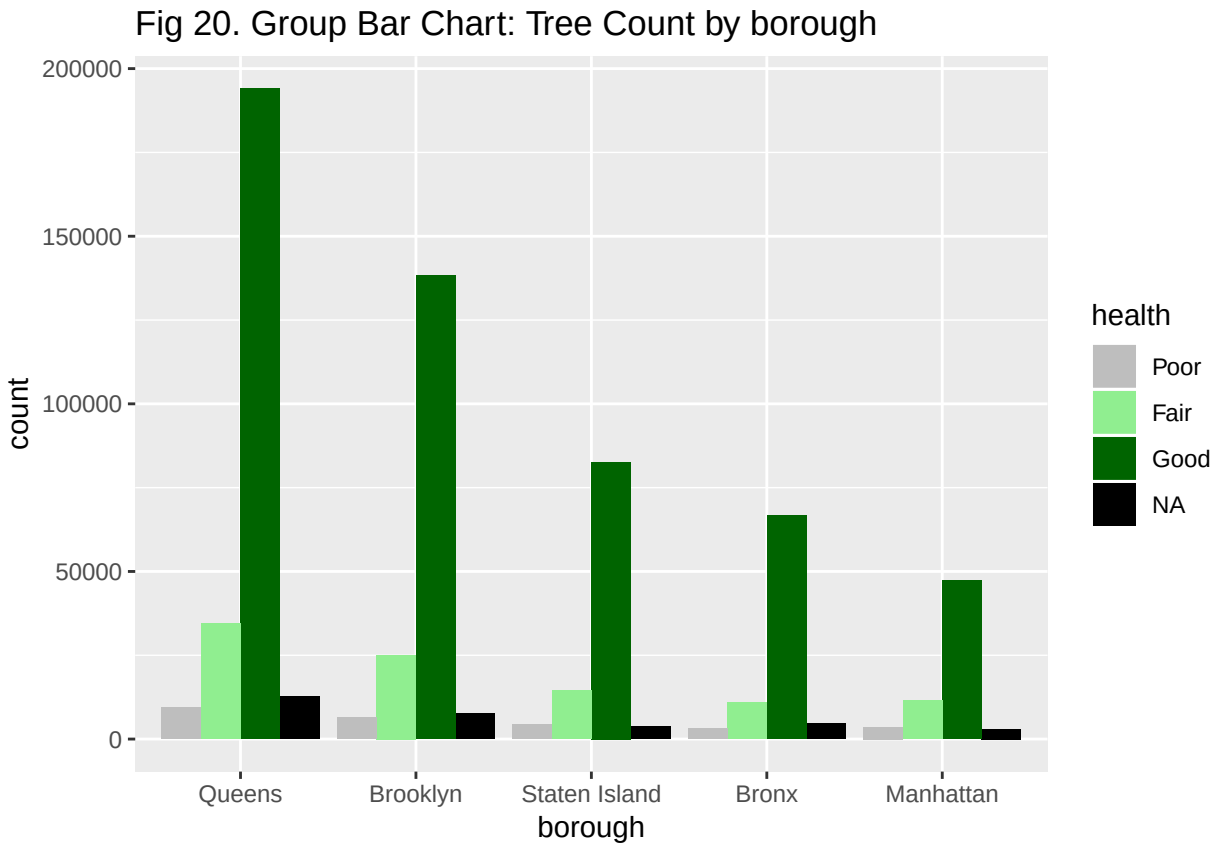
Figure 2 indicates the tree's health condition is not definitely related by the type of data collectors. Although the trees count staff do collect more good condition trees, it maybe because that trees count staff are assign more good condition trees to count.

Health vs. Borough

```
summary(Tree$borough)
```

```
##        Queens      Brooklyn Staten Island         Bronx     Manhattan
##        250551        177293        105318         85203         65423
```

```
healthcolors <- c("gray", "lightgreen", "darkgreen")
ggplot(Tree, aes(x = borough, fill = health)) +
  geom_bar(position = "dodge") +
  scale_fill_manual(values = healthcolors, na.value = "black") +
  ggtitle("Fig 20. Group Bar Chart: Tree Count by borough") + theme_gray()
```




```
HealthBorough<-ggplot(Tree, aes(x = borough, fill = health)) +
    geom_bar()+ scale_fill_manual(values = healthcolors,
                                  na.value = "black")+scale_y_continuous(labels=
                                                                           c("0","50","100","150","200","250

HealthBorough+ggtitle("Fig 21. Stack Bar Chart: Tree Count by Borough")
```

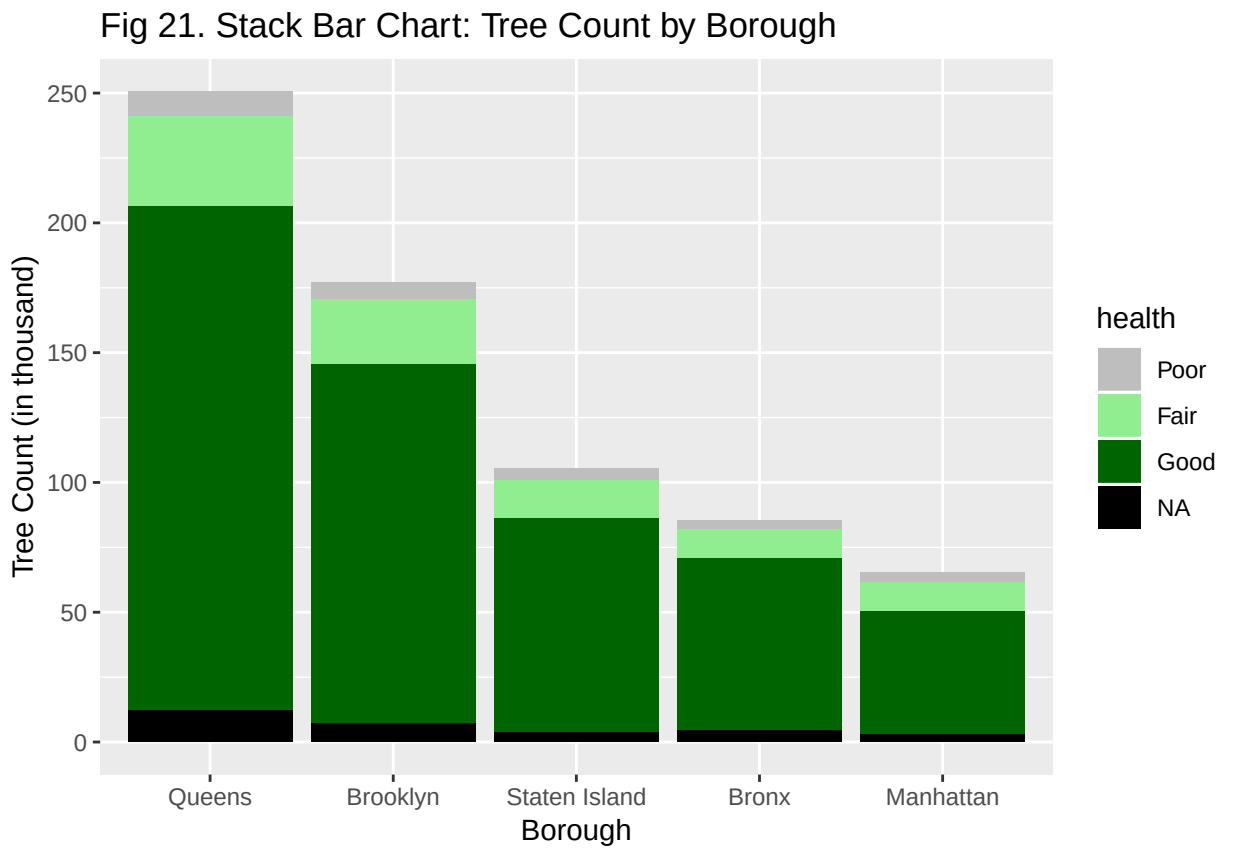


Figure 3 shows that Queens and Brooklyn have significantly more percentage trees in good conditions than the other three boroughs.

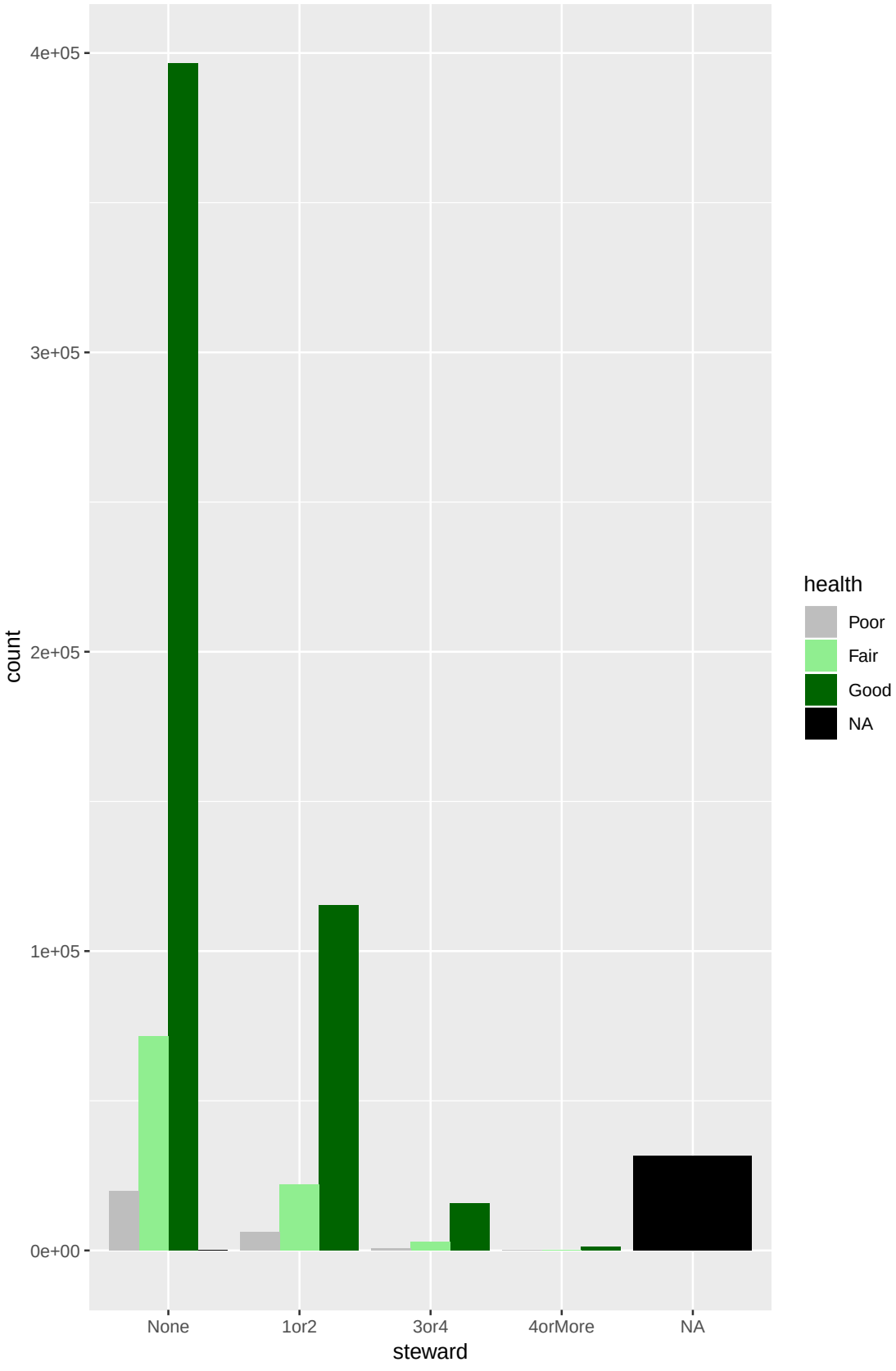
Health vs. Steward

```
summary(Tree$steward)
```

```
##            1or2    3or4 4orMore    None    NA's
##       0  143557   19183    1610  487823   31615
```

```
Tree$steward <- factor(Tree$steward, ordered = TRUE,
                       levels <- c("None", "1or2", "3or4", "4orMore"))

healthcolors <- c("gray", "lightgreen", "darkgreen")
ggplot(Tree, aes(x = steward, fill = health)) +
  geom_bar(position = "dodge") +
  scale_fill_manual(values = healthcolors, na.value = "black") +
  ggtitle("Fig 22. Group Bar Chart: Tree Count by steward") + theme_gray()
```

Fig 22. Group Bar Chart: Tree Count by steward



```r
library(grid)
vcd::mosaic(health~steward, Tree,
            direction = c("v", "h"), # <- order: steward ("v"), health ("h")
            gp = gpar(fill = healthcolors),
            rot_labels=c(90,0,0,90),offset_labels=c(1,0,0,0),
            labeling_args=list(gp_labels=gpar(fontsize=8)),
            main="Fig 23. Mosaic Plot: Health vs. Steward")
```

Fig 23. Mosaic Plot: Health vs. Steward

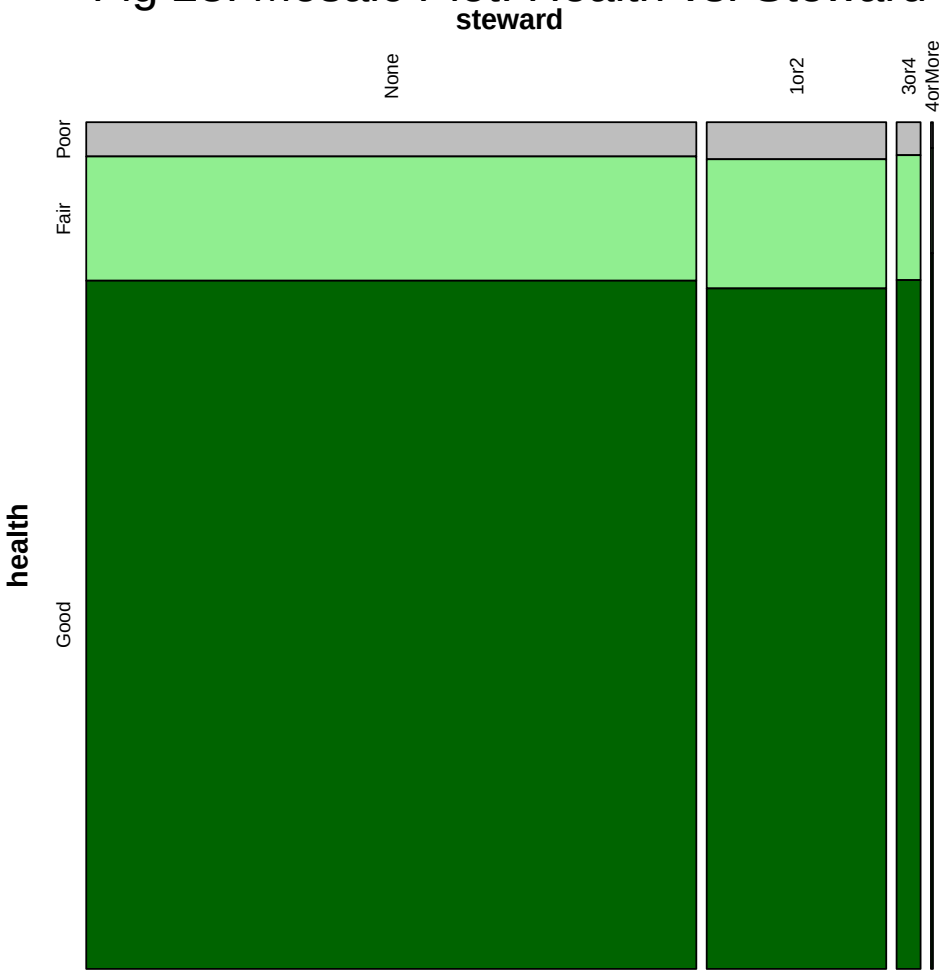
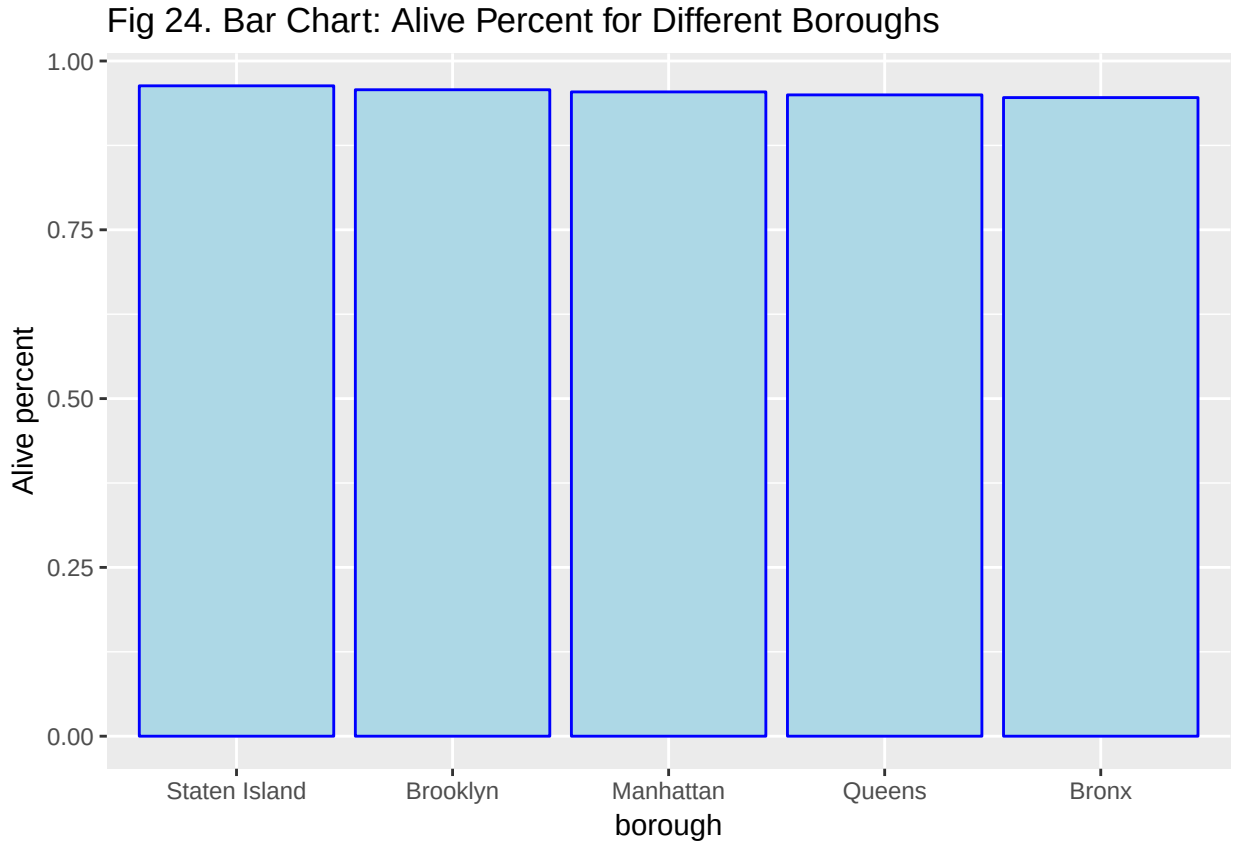


Figure 4 shows that when the number of steward increases, the percentage of good condition trees increase slightly as well. The trees need more stewards may be more difficult to be alive, so the positive impact of steward may be larger than what figure 4 demonstrates.

Status(Alive Percantage) vs. Borough

```
borough_percent<-Tree%>%group_by(borough)%>%
summarize(total = n(),alive_tree_borough = sum(status=='Alive'))%>%
mutate(alive_percent_borough = alive_tree_borough/total)
borough_percent$borough <- factor(borough_percent$borough, levels = borough_percent$borough[order(-borou
borough_plot_alive<-ggplot(borough_percent,aes(x = borough,
                                               y = alive_percent_borough))+
geom_bar(stat = "identity",fill="lightblue", color="blue")+
ylab("Alive percent")+
ggtitle("Fig 24. Bar Chart: Alive Percent for Different Boroughs")
borough_plot_alive
```



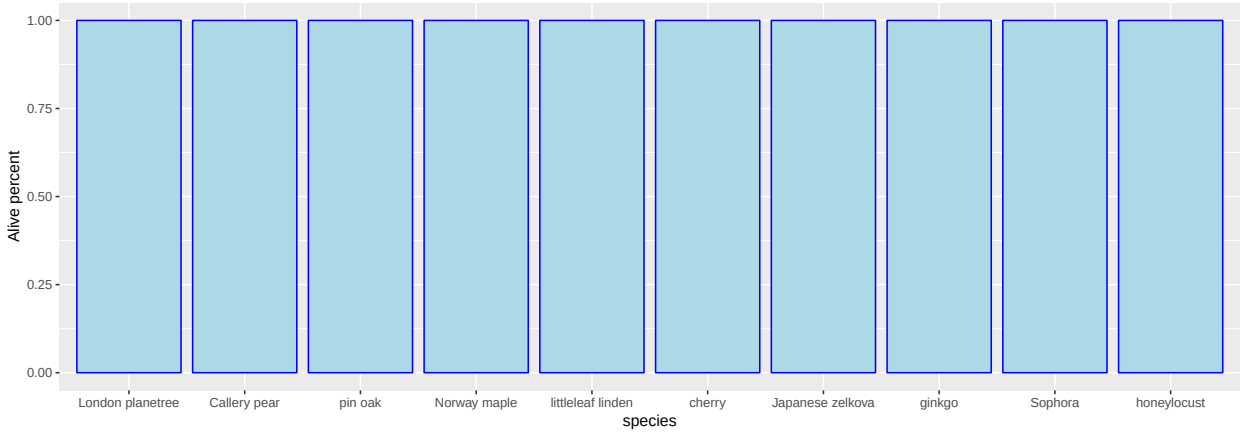
Alive percent is transformed from status variable, alive percent = number of alive tree/number of total alive/dead/stump trees.

Status(Alive Percantage) vs. Species

```
species_percent<-trees%>%group_by(species)%>%
summarize(total = n(),alive_species = sum(status=='Alive'))%>%
```

```
mutate(alive_percent_species = alive_species/total)
species_percent$species <- factor(species_percent$species, levels = species_percent$species[order(-speci
species_plot_alive<-ggplot(species_percent,aes(x = species,
                                               y = alive_percent_species))+
geom_bar(stat = "identity",fill="lightblue", color="blue")+
ylab("Alive percent")+
ggtitle("Fig 25. Bar Chart: Alive Percent for Different Species")
species_plot_alive
```

Fig 25. Bar Chart: Alive Percent for Different Species



From above two bar chart, we see status (alive percent) not depends on boroughs and top 10 tree species.

EXECUTIVE SUMMARY

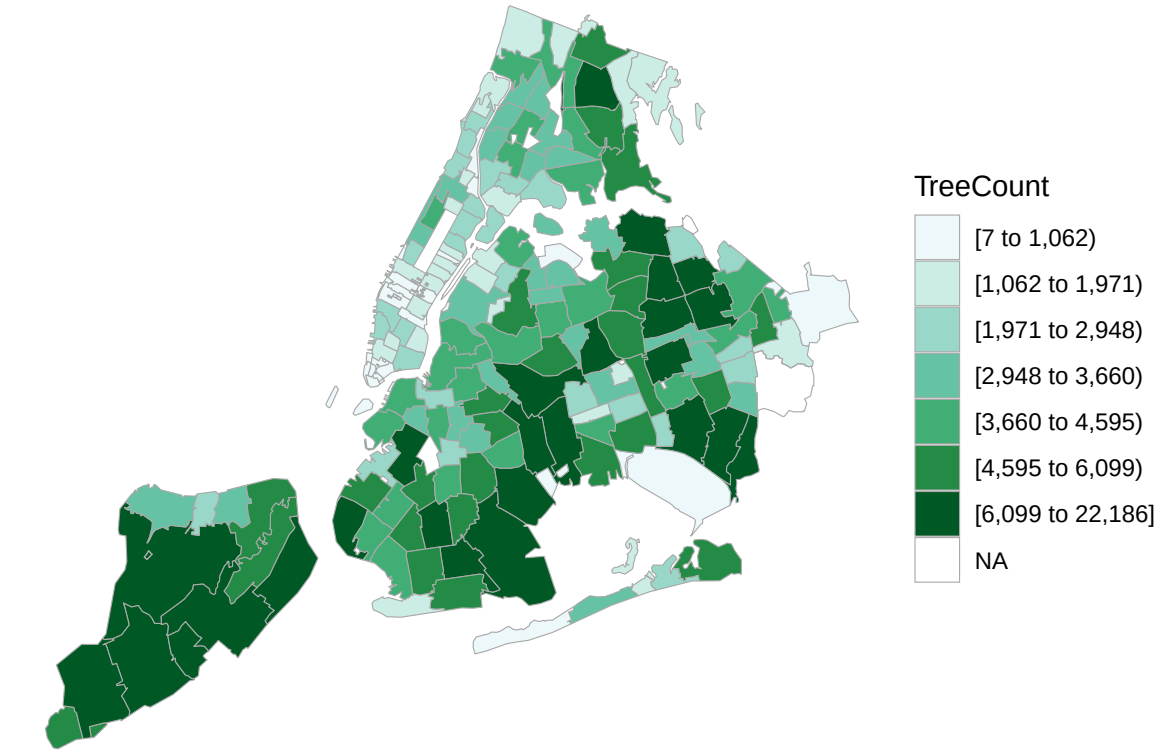
Our data comes from the New York Open Data, including 2015 trees census in NYC. The analysis of street tree census reflects the landscape of the city, which contributes to the human living senario. Based on the data, our project mainly wants to explore the situations of the tree's own properties (including the alive rate for trees, the distribution of the tree species, the diameter of the trees) and also the attributes related to these properties (whether those tree properties depend on boroughs, stewards, user types, population, and land area in NYC etc.).

After the data cleaning process mentioned in the preparation part, the tree's own properties including the **status**, **health**, **tree species**, and **diameters** were explored by analyzing their counts in histogram, bar chart, and table. Moreover, the dependencies between species and health against diameters were also explored. Then, the dependencies of their properties on the attributes were visualize, specifically on **borough**, **user type (recorder)**, and **steward**.

Below are takeaways we explored in terms of the visualizations implemented in the main analysis.

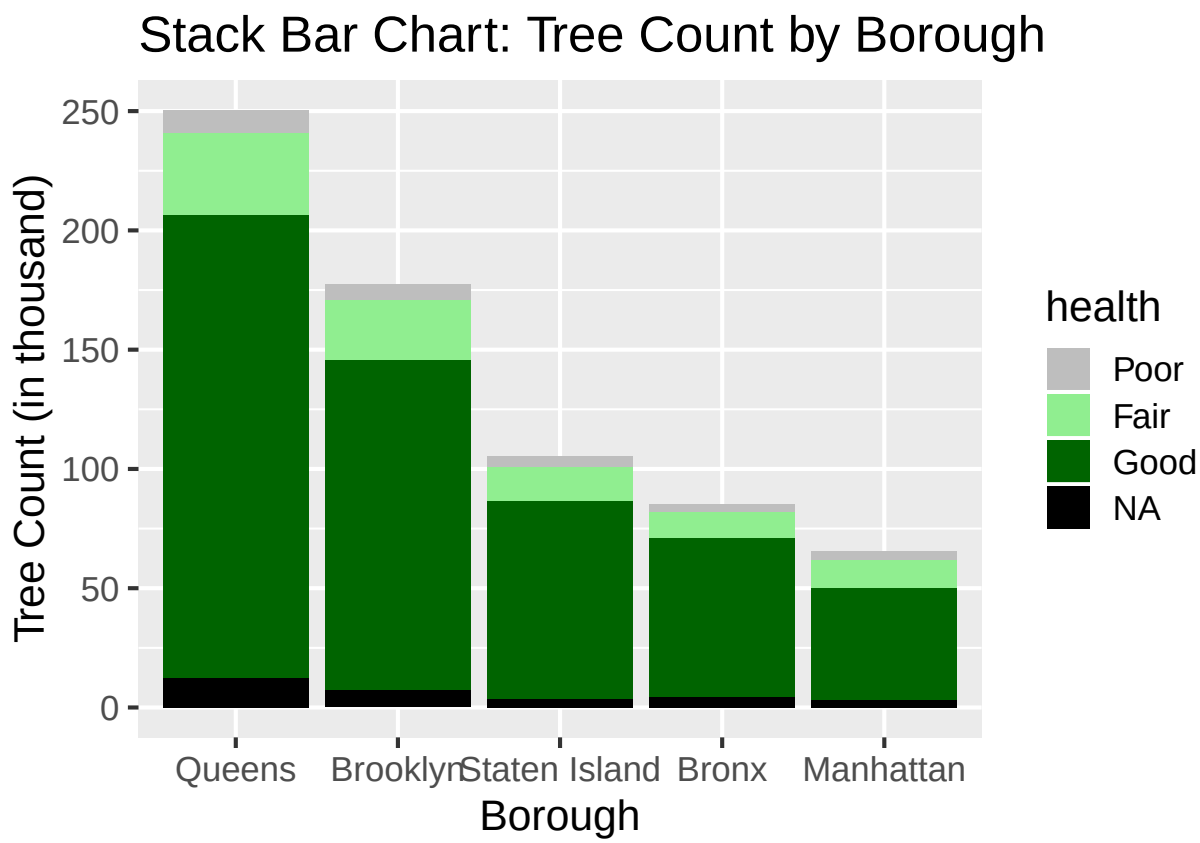
- Regarding to the tree's own properties, there are around 95% of trees classified as in alive condition whereas the rest 5% specified into the dead and stump status, which consists of most NA features in this dataset. Within the alive trees, the trees in good condition is about 81.1% of all the alive trees, fair condition trees is about 14.8% and poor condition is about 4.1%. On the other hand, the tree DBH(diameter at breast height) is centering at 5-6, with more than 100 species.
- Considering the properties dependencies, consistent with the common senses, the thicker the trees are, the better their health conditions are. Also, the diameters of the trees are highly depend on the tree species.
- In terms of the trees attributes upon their properties, borough, as a popular consideration, was frequently talked. Firstly, the map of the count of the trees upon the borough is plotted below, where Staten Island owns the highest greening proportion, while Manhattan hits the lowest one.

```
choro$title="Choropleth: 2015 NYC Street Tree Count by ZipCode"
choro$render()
```



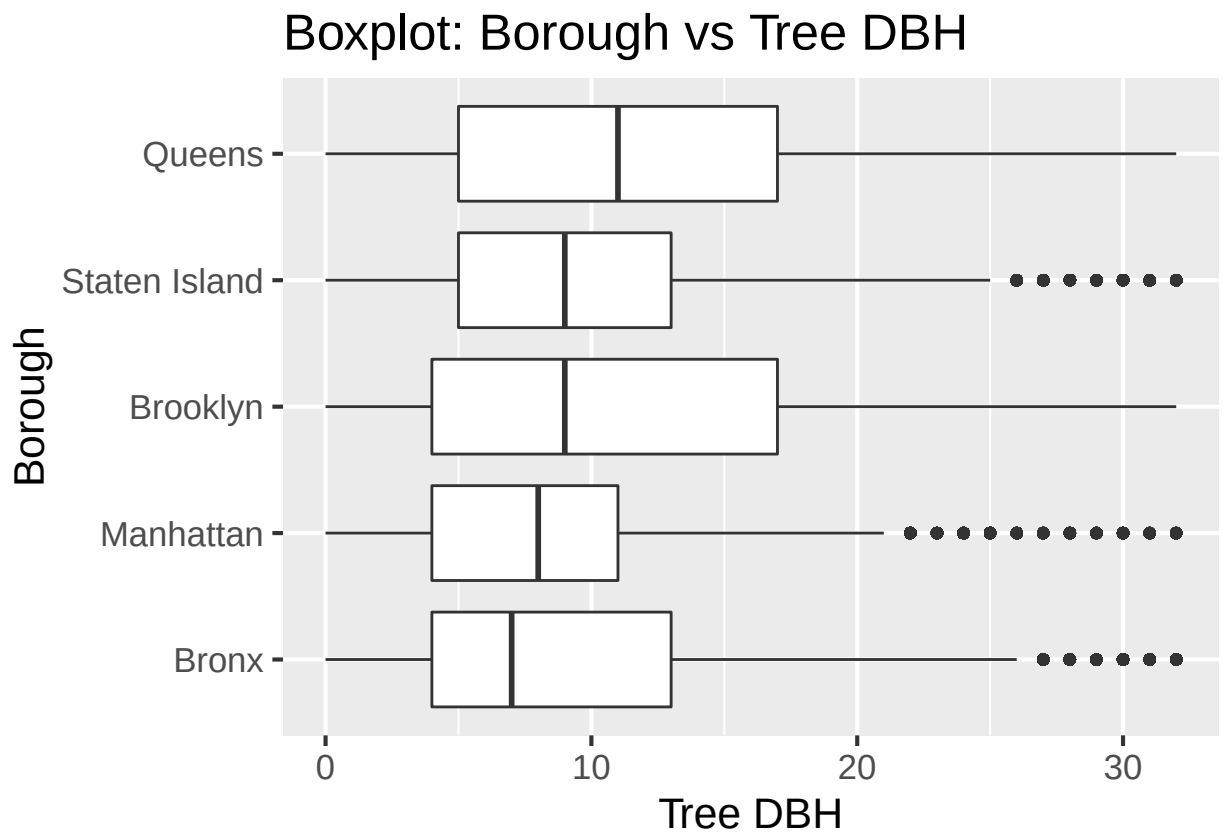
Second, be consistent with the trees count shown on the map, the health conditions upon the borough is shown here, where Queens and Brooklyn have significantly more percentage trees in good conditions than the other three boroughs.

```
HealthBorough+theme_grey(16)+
  ggtitle("Stack Bar Chart: Tree Count by Borough")
```



Then, there are also findings in diameters difference of the borough shown below. Queens has the largest median of diameters. Queens and Brooklyn have a wider range of the diameters.Bronx and Manhattan have smallest diameters maybe because the urbanlization of the boroughs.

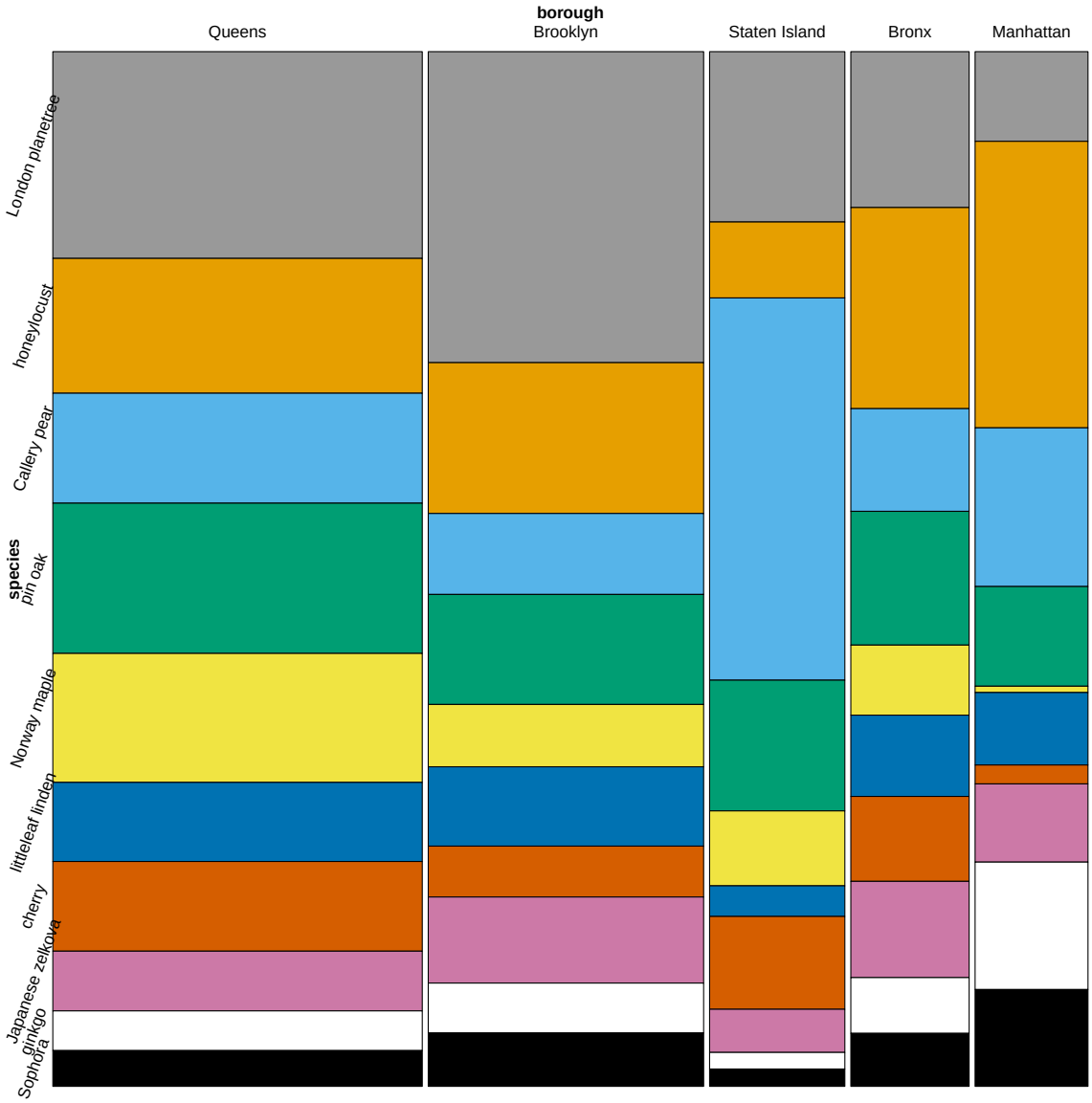
```
BoroughDia+theme_grey(16)+ggtitle(
                        "Boxplot: Borough vs Tree DBH")
```



Last, the tree species distributions in terms of the borough are plotted. According to it, the greening plan for every boroughs are extracted by considering the urbanlizations and soil conditions. We refill this plot with the color blind brewer as below:

```
fillcolors <- c("#999999", "#E69F00", "#56B4E9", "#009E73",
                "#F0E442", "#0072B2", "#D55E00", "#CC79A7","white","#000000")

orderlevel2 = c("Queens","Brooklyn","Staten Island","Bronx","Manhattan")
orderlevel3 = c("London planetree", "honeylocust","Callery pear",
                "pin oak","Norway maple","littleleaf linden","cherry","Japanese zelkova","ginkgo","Soph
trees$borough = factor(trees$borough, levels = orderlevel2)
trees$species = factor(trees$species , levels = orderlevel3)
mosaic_final<-vcd::mosaic(species ~ borough, trees,
            direction = c("v", "h"),
            rot_labels=c(0,0,0,70),
            gp = gpar(fill = fillcolors),
main = "Mosaic Plot: Top 10 Count Number of Tree Species Depend on Different Boroughs")
```

Mosaic Plot: Top 10 Count Number of Tree Species Depend on Different Boroughs



INTERACTIVE COMPONENT

The group developed a interactive app named "Explore Trees in NYC", including two parts: "Tree Species Distribution" and "Related Variables to Borough". This interactive app was developed by "Shiny", and was published on the web at https://edvtreeanalysis.shinyapps.io/application/

Remark: Because dataset is big, the time to open it a little long. Thanks for your patient!

In the Data Analysis part, we found that there are 133 different species of street trees spreaded in NYC, with the greatest of 87014 tree counts in "London planetree", and smallest of 183 tree counts in "Amur cork tree". It's interesting to see such a large range in tree species, and the significant difference in total tree counts between them, so we want to further explore the spatial distribution of different tree species. In this case, it's a good idea to show this information on the spatial map interactively. What's more, "borough" in our dataset is a crucial variable, because many other variables do have a relationship with it based on the above analysis.Therefore, it will be helpful for the user to analysis their relationship if we can show the mosaic plot of borough and these variables interactively based on users' choice.

The objective of the interactive component is to show a) the spatial distribution of different tree species on map; b) the relationship between "borough" and its related variables. In order to implement the app with a concise and user-friendly design, we choose "Shiny" as the development tool and deploy it on shinyapps.io. The app mainly included two parts: "Tree Species Distribution" and "Related Variables to Borough", corresponding to the two design objectives, respectively.

1)Trees Species Distribution part
In the analysis part, we found that the distribution of top 10 count number trees species' varied from borough to borough,so we want to show our audience, the distribution of the tree species that the user interested based on their choice. For this part, audiences can see a map of five boroughs in NYC, and each green circle represent a single tree with corresponding location in NYC on the map. For interaction, audiences can choose a single specie from a list we provided. Also, a checkbox named "Alive" would only return the chosen species trees in alive status.(Note: At each time, audiences can just choose one specie).

2)Related variables to Borough part The two related variables we choose are "species"(shown as "Tree_species") and "user_type"(shown as "Recorder_type"). In the data quality analysis part, it was found that the proportion of "user_type"(recorder types) is different in each borough, which might cause the occurrence of bias in the dataset(as different record groups have different preference when recording the data).Therefore, we want to shows our audience this information to pay attention to data quality.

For this part, audience can select their interested variable. After selecting variables, a mosaic plot between this specific variable and boroughs would be shown. This interactivity design can help the user observe the relationship between their interested variables and borough clearly.

During the development of the "Explore Trees in NYC" app, the main constraint in this process is the slow speed of execution, which as a result of the large scale of the dataset. With 683,788 observations and 45 variables in the dataset, it cost much longer running time than expected to demonstrate the distribution of the selected species in NYC when executing . To speed up this reaction process, only the top 10 species were shown in the drop-down selection bar. In the future, we would focus on improving the response running time in the "Trees Species Distribution part" in order to provide a better interaction experience for the app users.

CONCLUSION