Data Cleansing using R

Name

Institution of Affiliation

Date

**Introduction**

It is believed that about 90% of the data in the world has been generated within 10 to 15 years ago. This is because the use of internet technology in businesses and organizations around the world has been adopted in the same period. The advancement in statistics and machine learning has been applied to big data with the aim of mining pieces of information that are necessary for strategic decision making in business. Currently, many companies utilize information from historical data in determining how the business has been performing, as section of the company that demonstrated weaknesses and others that showed strength. Data has been used to predict market trends and thus help the management in planning and budgeting for the same. However, information from thee data can be misleading, especially when the data used is of poor quality. In other words, Ridzuan & Zainon (2019) affirm that data has to be cleansed first, for the results of its analysis to be dependable.

**Data cleansing**

In this activity, R statistical software was used to import the raw data for the check-up of inconsistencies and errors. Inconsistency and errors emerge during data entry and data collection where wrong values and strings can be entered, duplication of records, or omission of values (Ridzuan & Zainon, 2019). The following cleansing strategies were applied to the dataset.

**Acquisition of raw data**

The CSV dataset was imported into R using the read\_scv () function. The first five rows of the dataset were retrieved using the head () function for quick scanning if the dataset could be containing some string inconsistencies for qualitative variables and missing values for date and quantitative variables. It was identified that the data had missing values and could contain whitespaces in strings. Also, extreme values of quantitative variables suggested the greater possibility of outliers’ existence in the dataset. Outliers are discouraged as they may affect the accuracy of analysis results.

**Frequency tables**

Frequency tables of neighborbood\_group and room\_type variables were generated using the freq () function found in summarytools package in R. The tables showed frequencies and cumulative frequencies for the categories. The aim was to find the same neighborhood groups or room type that could have been represented as different due to spelling errors—also, the task aimed at identifying the missing categories. Neither missing category nor the same categories represented in different spelling were identified.

**Univariate functions**

The mean (), median (), var () functions are the univariate functions in R that were used to output the descriptive summary statistics for the quantitative univariate variables price, and availability\_365. The mean and median measures the central tendency of variables, while variance explains about the scatteredness of the values from the mean. The mean of price and availability\_365 were 158.4125 and 159.5886, medians were 125, and 166 respective and standard deviations were 145.036 and 136.0674, respectively.

**Outliers detection and treatment**

According to Osborne (2013), in data science and machine learning, most classification and algorithms are affected by outliers, and hence the accuracy of their results is compromised. Histogram and boxplots were used to show the presence of outliers in quantitative variables. The outliers were replaced with medians of their respective variable through boxplot. Stats () function.

**Missing values and duplicated records**

The sum of the missing values in the dataset was obtained using the sum (is.na ()) function in R. The dataset was found to contain 156 missing values, which were removed using the na.omit () function. Their removal reduced the number of rows from 1743 to 1587. After removing the missing values, the dataset was checked for duplicated rows based on the id column, and none was found.

**Whitespace removal**

Whitespaces in string variables are sometimes difficult to notice, but they make the variables inconsistent. The whitespaces were trimmed using the str\_trim () function, if any, were present in the qualitative variables.

**Strengths and weaknesses**

Knowing the strategies explained above to cleanse data, is my strength as the results of the analysis from the dataset can be applied with greater confidence in decision making. Also, the skills enable me to clean large datasets within a short time, to provide a correct and consistent dataset for different purposes. Another strength is that I am capable of appending and merging different datasets from various sources into one dataset and perform cleansing at once, which saves time. The only weakness I have in cleanup is that when replacing outliers with median or mean and removing missing values, it may sometime affect the generalization accuracy for the whole population.

**Conclusion**

Clean data is essential for providing accurate and compelling results for strategic decision making. For this task, the freq () function was used to explore the distribution of categorical variables. In contrast, the univariate functions, that are mean (), median (), and var (), were used to show the distribution of quantitative services. Boxplot and histogram were used to visualize outliers and then boxplot.stats () was used to locate them and replace them with the median. All the missing values were removed using the na.omit (), while the whitespaces in the string were removed by str\_trim () function. The skills are my strength for manipulating large datasets and provide consistent dataset for analysis.

References

Osborne, J. W. (2013). One: Why Data Cleaning is Important: Debunking the Myth of Robustness. *Best Practices in Data Cleaning: A Complete Guide to Everything You Need to Do Before and After Collecting Your Data*, 1-17.

Ridzuan, F., & Zainon, W. M. N. W. (2019). A Review on Data Cleansing Methods for Big Data. *Procedia Computer Science*, *161*, 731-738.

Appendix

DATA CLEANSING MARKDOWN

## importing data, cheking structure and dimension

data <- read.csv(file.choose(), header = T)  
View(head(data, n = 5))  
str(data)

## 'data.frame': 1743 obs. of 16 variables:  
## $ id : int 2539 2595 3647 3831 5022 5099 5121 5178 5203 5238 ...  
## $ name : Factor w/ 1721 levels " 1 Bed Apt in Utopic Williamsburg ",..: 482 1378 1597 579 690 944 325 947 574 626 ...  
## $ host\_id : int 2787 2845 4632 4869 7192 7322 7356 8967 7490 7549 ...  
## $ host\_name : Factor w/ 1034 levels "","A.B.","Aaron",..: 458 435 263 582 552 167 321 896 650 100 ...  
## $ neighbourhood\_group : Factor w/ 5 levels "Bronx","Brooklyn",..: 2 3 3 2 3 3 2 3 3 3 ...  
## $ neighbourhood : Factor w/ 115 levels "Allerton","Arrochar",..: 63 72 57 24 37 76 8 58 107 20 ...  
## $ latitude : num 40.6 40.8 40.8 40.7 40.8 ...  
## $ longitude : num -74 -74 -73.9 -74 -73.9 ...  
## $ room\_type : Factor w/ 3 levels "Entire home/apt",..: 2 1 2 1 1 1 2 2 2 1 ...  
## $ price : int 149 225 150 89 80 200 60 79 79 150 ...  
## $ minimum\_nights : int 1 1 3 1 10 3 45 2 2 1 ...  
## $ number\_of\_reviews : int 9 45 0 270 9 74 49 430 118 160 ...  
## $ last\_review : Factor w/ 592 levels "","1/1/2013",..: 66 329 1 466 120 397 92 403 449 433 ...  
## $ reviews\_per\_month : num 0.21 0.38 NA 4.64 0.1 0.59 0.4 3.47 0.99 1.33 ...  
## $ calculated\_host\_listings\_count: int 6 2 1 1 1 1 1 1 1 4 ...  
## $ availability\_365 : int 365 355 365 194 0 129 0 220 0 188 ...

dim(data)

## [1] 1743 16

## Frequency of qualitative variables

library(summarytools)

## Warning: package 'summarytools' was built under R version 3.6.3

## Registered S3 method overwritten by 'pryr':  
## method from  
## print.bytes Rcpp

## For best results, restart R session and update pander using devtools:: or remotes::install\_github('rapporter/pander')

library(lifecycle)

## Warning: package 'lifecycle' was built under R version 3.6.3

summarytools::freq(data$neighbourhood\_group, order = "freq")

## Frequencies   
## data$neighbourhood\_group   
## Type: Factor   
##   
## Freq % Valid % Valid Cum. % Total % Total Cum.  
## ------------------- ------ --------- -------------- --------- --------------  
## Manhattan 805 46.18 46.18 46.18 46.18  
## Brooklyn 793 45.50 91.68 45.50 91.68  
## Queens 103 5.91 97.59 5.91 97.59  
## Bronx 22 1.26 98.85 1.26 98.85  
## Staten Island 20 1.15 100.00 1.15 100.00  
## <NA> 0 0.00 100.00  
## Total 1743 100.00 100.00 100.00 100.00

summarytools::freq(data$room\_type, order = "freq")

## Frequencies   
## data$room\_type   
## Type: Factor   
##   
## Freq % Valid % Valid Cum. % Total % Total Cum.  
## --------------------- ------ --------- -------------- --------- --------------  
## Entire home/apt 1052 60.36 60.36 60.36 60.36  
## Private room 674 38.67 99.02 38.67 99.02  
## Shared room 17 0.98 100.00 0.98 100.00  
## <NA> 0 0.00 100.00  
## Total 1743 100.00 100.00 100.00 100.00

### Univeriate functions

mean(data$price)

## [1] 158.4125

median(data$price)

## [1] 125

var(data$price)

## [1] 21035.43

sqrt(var(data$price))

## [1] 145.036

mean(data$availability\_365)

## [1] 159.5886

median(data$availability\_365)

## [1] 166

var(data$availability\_365)

## [1] 18514.33

sqrt(var(data$availability\_365))

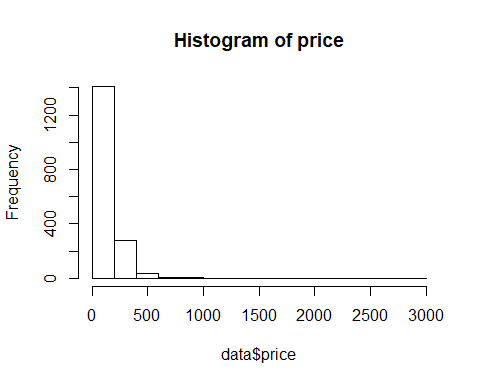
## [1] 136.0674

## triming white spaces.

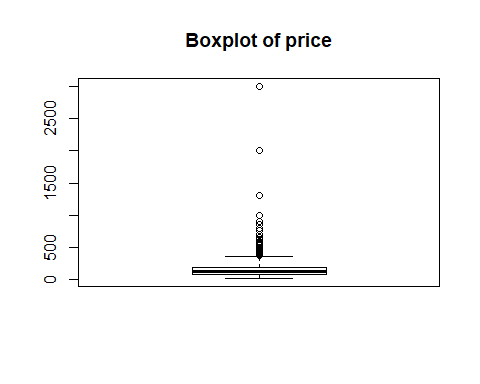
library(stringr)  
data$name <- str\_trim(data$name)  
data$host\_name <- str\_trim(data$host\_name)  
data$neighbourhood\_group <- str\_trim(data$neighbourhood\_group)  
data$neighbourhood <- str\_trim(data$neighbourhood)  
data$room\_type <- str\_trim(data$room\_type)  
data$last\_review <- str\_trim(data$last\_review)

##visualizations to detect outliers.

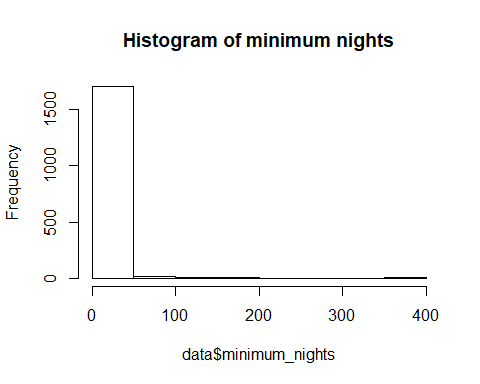
hist(data$price, main="Histogram of price")



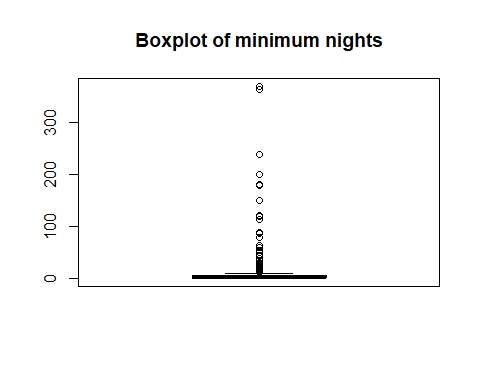
boxplot(data$price, main="Boxplot of price")



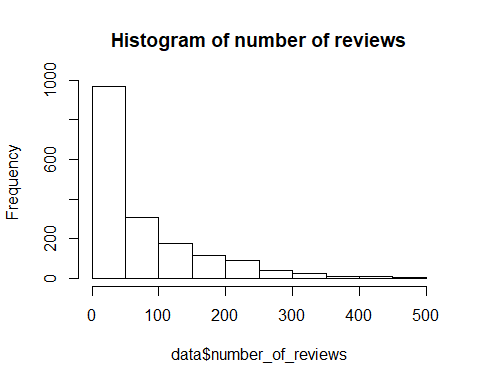
hist(data$minimum\_nights, main="Histogram of minimum nights")



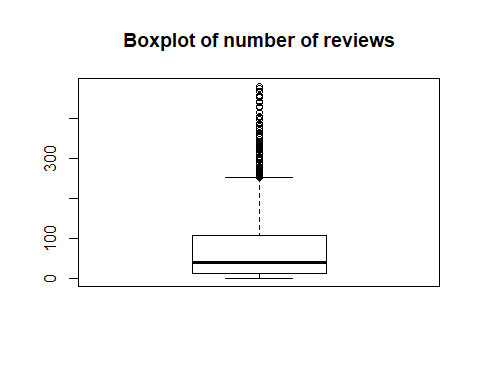
boxplot(data$minimum\_nights, main="Boxplot of minimum nights")



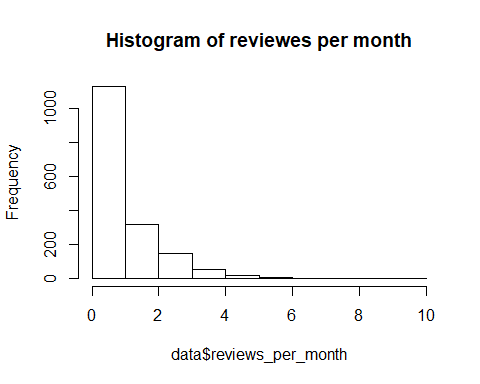
hist(data$number\_of\_reviews, main="Histogram of number of reviews")



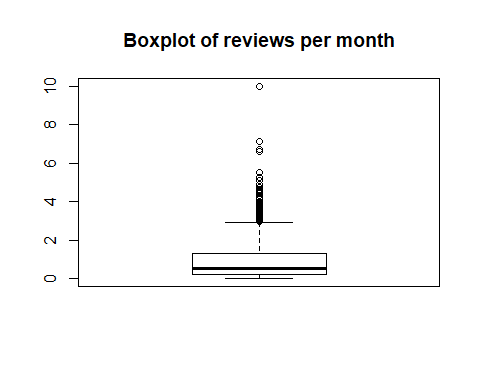
boxplot(data$number\_of\_reviews, main="Boxplot of number of reviews")



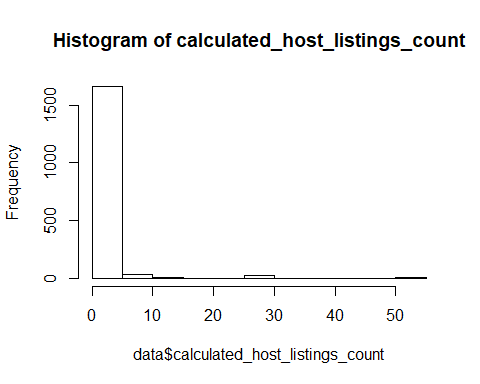
hist(data$reviews\_per\_month, main="Histogram of reviewes per month")



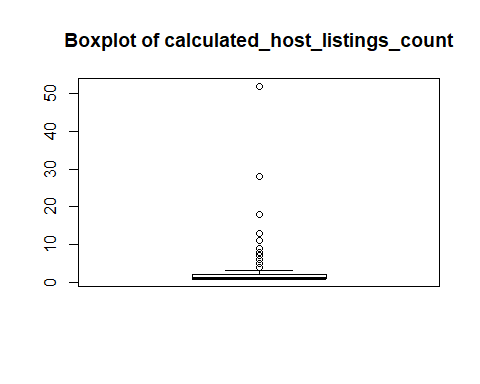
boxplot(data$reviews\_per\_month, main="Boxplot of reviews per month")



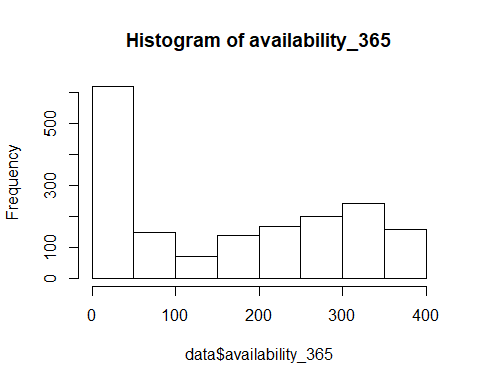
hist(data$calculated\_host\_listings\_count, main="Histogram of calculated\_host\_listings\_count")



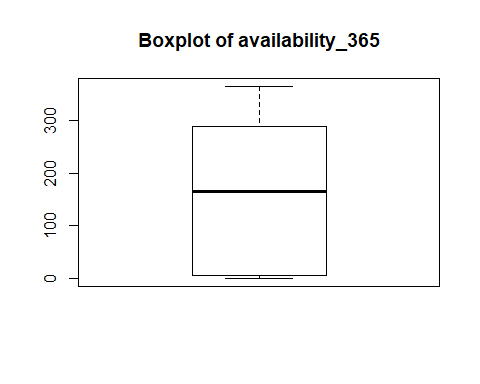
boxplot(data$calculated\_host\_listings\_count, main="Boxplot of calculated\_host\_listings\_count")



hist(data$availability\_365, main="Histogram of availability\_365")



boxplot(data$availability\_365, main="Boxplot of availability\_365")



### Replacing outliers in the data with median

column1<-boxplot.stats(data$price)$out;  
data$price[data$price %in% column1]<-median(data$price)  
  
column2<-boxplot.stats(data$minimum\_nights)$out;  
data$minimum\_nights[data$minimum\_nights %in% column2]<-median(data$minimum\_nights)  
  
column3<-boxplot.stats(data$number\_of\_reviews)$out;  
data$number\_of\_reviews[data$number\_of\_reviews %in% column3]<-median(data$number\_of\_reviews)  
  
column4<-boxplot.stats(data$reviews\_per\_month)$out;  
data$reviews\_per\_month[data$reviews\_per\_month %in% column4]<-median(data$reviews\_per\_month)  
  
column5<-boxplot.stats(data$calculated\_host\_listings\_count)$out;  
data$calculated\_host\_listings\_count[data$calculated\_host\_listings\_count %in% column5]<-median(data$calculated\_host\_listings\_count)  
  
column6<-boxplot.stats(data$availability\_365)$out;  
data$availability\_365[data$availability\_365 %in% column6]<-median(data$availability\_365)

### Missing values.

sum(is.na(data))

## [1] 156

data1 <-na.omit(data)  
dim(data1)

## [1] 1587 16

### Checking for duplicates

sum(duplicated(data1))

## [1] 0

table(duplicated(data1$id))

##   
## FALSE   
## 1587