data cleaning

2023-10-21

housing = read.csv('data1.csv')

dim(housing)

## [1] 18396 22

# Assuming 'housing' is a data frame in R  
summary\_data <- summary(housing)  
summary\_data <- t(summary\_data)  
print(summary\_data)

##   
## X Min. : 1 1st Qu.: 5937 Median :11820   
## Suburb Length:18396 Class :character Mode :character   
## Address Length:18396 Class :character Mode :character   
## Rooms Min. : 1.000 1st Qu.: 2.000 Median : 3.000   
## Type Length:18396 Class :character Mode :character   
## Price Min. : 85000 1st Qu.: 633000 Median : 880000   
## Method Length:18396 Class :character Mode :character   
## SellerG Length:18396 Class :character Mode :character   
## Date Length:18396 Class :character Mode :character   
## Distance Min. : 0.00 1st Qu.: 6.30 Median : 9.70   
## Postcode Min. :3000 1st Qu.:3046 Median :3085   
## Bedroom2 Min. : 0.000 1st Qu.: 2.000 Median : 3.000   
## Bathroom Min. :0.000 1st Qu.:1.000 Median :1.000   
## Car Min. : 0.000 1st Qu.: 1.000 Median : 2.000   
## Landsize Min. : 0.0 1st Qu.: 176.5 Median : 440.0   
## BuildingArea Min. : 0.0 1st Qu.: 93.0 Median : 126.0   
## YearBuilt Min. :1196 1st Qu.:1950 Median :1970   
## CouncilArea Length:18396 Class :character Mode :character   
## Lattitude Min. :-38.18 1st Qu.:-37.86 Median :-37.80   
## Longtitude Min. :144.4 1st Qu.:144.9 Median :145.0   
## Regionname Length:18396 Class :character Mode :character   
## Propertycount Min. : 249 1st Qu.: 4294 Median : 6567   
##   
## X Mean :11827 3rd Qu.:17734 Max. :23546   
## Suburb   
## Address   
## Rooms Mean : 2.935 3rd Qu.: 3.000 Max. :12.000   
## Type   
## Price Mean :1056697 3rd Qu.:1302000 Max. :9000000   
## Method   
## SellerG   
## Date   
## Distance Mean :10.39 3rd Qu.:13.30 Max. :48.10   
## Postcode Mean :3107 3rd Qu.:3149 Max. :3978   
## Bedroom2 Mean : 2.913 3rd Qu.: 3.000 Max. :20.000   
## Bathroom Mean :1.538 3rd Qu.:2.000 Max. :8.000   
## Car Mean : 1.616 3rd Qu.: 2.000 Max. :10.000   
## Landsize Mean : 558.1 3rd Qu.: 651.0 Max. :433014.0   
## BuildingArea Mean : 151.2 3rd Qu.: 174.0 Max. :44515.0   
## YearBuilt Mean :1966 3rd Qu.:2000 Max. :2018   
## CouncilArea   
## Lattitude Mean :-37.81 3rd Qu.:-37.76 Max. :-37.41   
## Longtitude Mean :145.0 3rd Qu.:145.1 Max. :145.5   
## Regionname   
## Propertycount Mean : 7518 3rd Qu.:10331 Max. :21650   
##   
## X   
## Suburb   
## Address   
## Rooms   
## Type   
## Price   
## Method   
## SellerG   
## Date   
## Distance NA's :1   
## Postcode NA's :1   
## Bedroom2 NA's :3469   
## Bathroom NA's :3471   
## Car NA's :3576   
## Landsize NA's :4793   
## BuildingArea NA's :10634   
## YearBuilt NA's :9438   
## CouncilArea   
## Lattitude NA's :3332   
## Longtitude NA's :3332   
## Regionname   
## Propertycount NA's :1

# Assuming 'housing' is a data frame in R  
str(housing)

## 'data.frame': 18396 obs. of 22 variables:  
## $ X : int 1 2 4 5 6 10 11 14 15 16 ...  
## $ Suburb : chr "Abbotsford" "Abbotsford" "Abbotsford" "Abbotsford" ...  
## $ Address : chr "85 Turner St" "25 Bloomburg St" "5 Charles St" "40 Federation La" ...  
## $ Rooms : int 2 2 3 3 4 2 3 2 3 2 ...  
## $ Type : chr "h" "h" "h" "h" ...  
## $ Price : num 1480000 1035000 1465000 850000 1600000 ...  
## $ Method : chr "S" "S" "SP" "PI" ...  
## $ SellerG : chr "Biggin" "Biggin" "Biggin" "Biggin" ...  
## $ Date : chr "3/12/2016" "4/02/2016" "4/03/2017" "4/03/2017" ...  
## $ Distance : num 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 ...  
## $ Postcode : num 3067 3067 3067 3067 3067 ...  
## $ Bedroom2 : num 2 2 3 3 3 2 4 2 NA NA ...  
## $ Bathroom : num 1 1 2 2 1 1 2 1 NA NA ...  
## $ Car : num 1 0 0 1 2 0 0 2 NA NA ...  
## $ Landsize : num 202 156 134 94 120 181 245 256 NA NA ...  
## $ BuildingArea : num NA 79 150 NA 142 NA 210 107 NA NA ...  
## $ YearBuilt : num NA 1900 1900 NA 2014 ...  
## $ CouncilArea : chr "Yarra" "Yarra" "Yarra" "Yarra" ...  
## $ Lattitude : num -37.8 -37.8 -37.8 -37.8 -37.8 ...  
## $ Longtitude : num 145 145 145 145 145 ...  
## $ Regionname : chr "Northern Metropolitan" "Northern Metropolitan" "Northern Metropolitan" "Northern Metropolitan" ...  
## $ Propertycount: num 4019 4019 4019 4019 4019 ...

Converting Datatype Variables I will need to convert the datatypes for the above datatypes to their appropriate classification. Some columns such as SALE PRICE is considered an object when it should be an float. And other columns such as NEIGHBORHOOD should be a categorical datatype rather than an object.

This dataset only contains object and integer dtypes.

# Assuming 'housing' is a data frame in R  
object\_columns <- sapply(housing, class) == "character"  
print(names(housing[object\_columns]))

## [1] "Suburb" "Address" "Type" "Method" "SellerG"   
## [6] "Date" "CouncilArea" "Regionname"

# Assuming 'housing' is a data frame in R  
integer\_columns <- sapply(housing, class) == "integer"  
print(names(housing[integer\_columns]))

## [1] "X" "Rooms"

# Assuming 'housing' is a data frame in R  
str(housing) # Provides information about data types and a summary of the data

## 'data.frame': 18396 obs. of 22 variables:  
## $ X : int 1 2 4 5 6 10 11 14 15 16 ...  
## $ Suburb : chr "Abbotsford" "Abbotsford" "Abbotsford" "Abbotsford" ...  
## $ Address : chr "85 Turner St" "25 Bloomburg St" "5 Charles St" "40 Federation La" ...  
## $ Rooms : int 2 2 3 3 4 2 3 2 3 2 ...  
## $ Type : chr "h" "h" "h" "h" ...  
## $ Price : num 1480000 1035000 1465000 850000 1600000 ...  
## $ Method : chr "S" "S" "SP" "PI" ...  
## $ SellerG : chr "Biggin" "Biggin" "Biggin" "Biggin" ...  
## $ Date : chr "3/12/2016" "4/02/2016" "4/03/2017" "4/03/2017" ...  
## $ Distance : num 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 ...  
## $ Postcode : num 3067 3067 3067 3067 3067 ...  
## $ Bedroom2 : num 2 2 3 3 3 2 4 2 NA NA ...  
## $ Bathroom : num 1 1 2 2 1 1 2 1 NA NA ...  
## $ Car : num 1 0 0 1 2 0 0 2 NA NA ...  
## $ Landsize : num 202 156 134 94 120 181 245 256 NA NA ...  
## $ BuildingArea : num NA 79 150 NA 142 NA 210 107 NA NA ...  
## $ YearBuilt : num NA 1900 1900 NA 2014 ...  
## $ CouncilArea : chr "Yarra" "Yarra" "Yarra" "Yarra" ...  
## $ Lattitude : num -37.8 -37.8 -37.8 -37.8 -37.8 ...  
## $ Longtitude : num 145 145 145 145 145 ...  
## $ Regionname : chr "Northern Metropolitan" "Northern Metropolitan" "Northern Metropolitan" "Northern Metropolitan" ...  
## $ Propertycount: num 4019 4019 4019 4019 4019 ...

# To get a summary of the data frame  
summary(housing) # Provides summary statistics for numeric columns

## X Suburb Address Rooms   
## Min. : 1 Length:18396 Length:18396 Min. : 1.000   
## 1st Qu.: 5937 Class :character Class :character 1st Qu.: 2.000   
## Median :11820 Mode :character Mode :character Median : 3.000   
## Mean :11827 Mean : 2.935   
## 3rd Qu.:17734 3rd Qu.: 3.000   
## Max. :23546 Max. :12.000   
##   
## Type Price Method SellerG   
## Length:18396 Min. : 85000 Length:18396 Length:18396   
## Class :character 1st Qu.: 633000 Class :character Class :character   
## Mode :character Median : 880000 Mode :character Mode :character   
## Mean :1056697   
## 3rd Qu.:1302000   
## Max. :9000000   
##   
## Date Distance Postcode Bedroom2   
## Length:18396 Min. : 0.00 Min. :3000 Min. : 0.000   
## Class :character 1st Qu.: 6.30 1st Qu.:3046 1st Qu.: 2.000   
## Mode :character Median : 9.70 Median :3085 Median : 3.000   
## Mean :10.39 Mean :3107 Mean : 2.913   
## 3rd Qu.:13.30 3rd Qu.:3149 3rd Qu.: 3.000   
## Max. :48.10 Max. :3978 Max. :20.000   
## NA's :1 NA's :1 NA's :3469   
## Bathroom Car Landsize BuildingArea   
## Min. :0.000 Min. : 0.000 Min. : 0.0 Min. : 0.0   
## 1st Qu.:1.000 1st Qu.: 1.000 1st Qu.: 176.5 1st Qu.: 93.0   
## Median :1.000 Median : 2.000 Median : 440.0 Median : 126.0   
## Mean :1.538 Mean : 1.616 Mean : 558.1 Mean : 151.2   
## 3rd Qu.:2.000 3rd Qu.: 2.000 3rd Qu.: 651.0 3rd Qu.: 174.0   
## Max. :8.000 Max. :10.000 Max. :433014.0 Max. :44515.0   
## NA's :3471 NA's :3576 NA's :4793 NA's :10634   
## YearBuilt CouncilArea Lattitude Longtitude   
## Min. :1196 Length:18396 Min. :-38.18 Min. :144.4   
## 1st Qu.:1950 Class :character 1st Qu.:-37.86 1st Qu.:144.9   
## Median :1970 Mode :character Median :-37.80 Median :145.0   
## Mean :1966 Mean :-37.81 Mean :145.0   
## 3rd Qu.:2000 3rd Qu.:-37.76 3rd Qu.:145.1   
## Max. :2018 Max. :-37.41 Max. :145.5   
## NA's :9438 NA's :3332 NA's :3332   
## Regionname Propertycount   
## Length:18396 Min. : 249   
## Class :character 1st Qu.: 4294   
## Mode :character Median : 6567   
## Mean : 7518   
## 3rd Qu.:10331   
## Max. :21650   
## NA's :1

# To count non-missing values in each column  
colSums(!is.na(housing))

## X Suburb Address Rooms Type   
## 18396 18396 18396 18396 18396   
## Price Method SellerG Date Distance   
## 18396 18396 18396 18396 18395   
## Postcode Bedroom2 Bathroom Car Landsize   
## 18395 14927 14925 14820 13603   
## BuildingArea YearBuilt CouncilArea Lattitude Longtitude   
## 7762 8958 18396 15064 15064   
## Regionname Propertycount   
## 18396 18395

head(housing)

## X Suburb Address Rooms Type Price Method SellerG Date  
## 1 1 Abbotsford 85 Turner St 2 h 1480000 S Biggin 3/12/2016  
## 2 2 Abbotsford 25 Bloomburg St 2 h 1035000 S Biggin 4/02/2016  
## 3 4 Abbotsford 5 Charles St 3 h 1465000 SP Biggin 4/03/2017  
## 4 5 Abbotsford 40 Federation La 3 h 850000 PI Biggin 4/03/2017  
## 5 6 Abbotsford 55a Park St 4 h 1600000 VB Nelson 4/06/2016  
## 6 10 Abbotsford 129 Charles St 2 h 941000 S Jellis 7/05/2016  
## Distance Postcode Bedroom2 Bathroom Car Landsize BuildingArea YearBuilt  
## 1 2.5 3067 2 1 1 202 NA NA  
## 2 2.5 3067 2 1 0 156 79 1900  
## 3 2.5 3067 3 2 0 134 150 1900  
## 4 2.5 3067 3 2 1 94 NA NA  
## 5 2.5 3067 3 1 2 120 142 2014  
## 6 2.5 3067 2 1 0 181 NA NA  
## CouncilArea Lattitude Longtitude Regionname Propertycount  
## 1 Yarra -37.7996 144.9984 Northern Metropolitan 4019  
## 2 Yarra -37.8079 144.9934 Northern Metropolitan 4019  
## 3 Yarra -37.8093 144.9944 Northern Metropolitan 4019  
## 4 Yarra -37.7969 144.9969 Northern Metropolitan 4019  
## 5 Yarra -37.8072 144.9941 Northern Metropolitan 4019  
## 6 Yarra -37.8041 144.9953 Northern Metropolitan 4019

str(housing)

## 'data.frame': 18396 obs. of 22 variables:  
## $ X : int 1 2 4 5 6 10 11 14 15 16 ...  
## $ Suburb : chr "Abbotsford" "Abbotsford" "Abbotsford" "Abbotsford" ...  
## $ Address : chr "85 Turner St" "25 Bloomburg St" "5 Charles St" "40 Federation La" ...  
## $ Rooms : int 2 2 3 3 4 2 3 2 3 2 ...  
## $ Type : chr "h" "h" "h" "h" ...  
## $ Price : num 1480000 1035000 1465000 850000 1600000 ...  
## $ Method : chr "S" "S" "SP" "PI" ...  
## $ SellerG : chr "Biggin" "Biggin" "Biggin" "Biggin" ...  
## $ Date : chr "3/12/2016" "4/02/2016" "4/03/2017" "4/03/2017" ...  
## $ Distance : num 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 ...  
## $ Postcode : num 3067 3067 3067 3067 3067 ...  
## $ Bedroom2 : num 2 2 3 3 3 2 4 2 NA NA ...  
## $ Bathroom : num 1 1 2 2 1 1 2 1 NA NA ...  
## $ Car : num 1 0 0 1 2 0 0 2 NA NA ...  
## $ Landsize : num 202 156 134 94 120 181 245 256 NA NA ...  
## $ BuildingArea : num NA 79 150 NA 142 NA 210 107 NA NA ...  
## $ YearBuilt : num NA 1900 1900 NA 2014 ...  
## $ CouncilArea : chr "Yarra" "Yarra" "Yarra" "Yarra" ...  
## $ Lattitude : num -37.8 -37.8 -37.8 -37.8 -37.8 ...  
## $ Longtitude : num 145 145 145 145 145 ...  
## $ Regionname : chr "Northern Metropolitan" "Northern Metropolitan" "Northern Metropolitan" "Northern Metropolitan" ...  
## $ Propertycount: num 4019 4019 4019 4019 4019 ...

missing value

# Assuming 'housing' is a data frame in R  
# Use the apply function with is.na to check for missing values in each column  
missing\_counts <- apply(housing, 2, function(x) sum(is.na(x)))  
  
# The missing\_counts variable now contains the count of missing values for each column  
print(missing\_counts)

## X Suburb Address Rooms Type   
## 0 0 0 0 0   
## Price Method SellerG Date Distance   
## 0 0 0 0 1   
## Postcode Bedroom2 Bathroom Car Landsize   
## 1 3469 3471 3576 4793   
## BuildingArea YearBuilt CouncilArea Lattitude Longtitude   
## 10634 9438 0 3332 3332   
## Regionname Propertycount   
## 0 1

# Assuming 'housing' is a data frame in R  
# Use the apply function with is.na to check for missing values in each column  
missing\_counts <- apply(housing, 2, function(x) sum(is.na(x)))  
  
# Calculate the percentage of missing values for each column  
percentage\_missing <- (missing\_counts / nrow(housing)) \* 100  
  
# The percentage\_missing variable now contains the percentage of missing values for each column  
print(percentage\_missing)

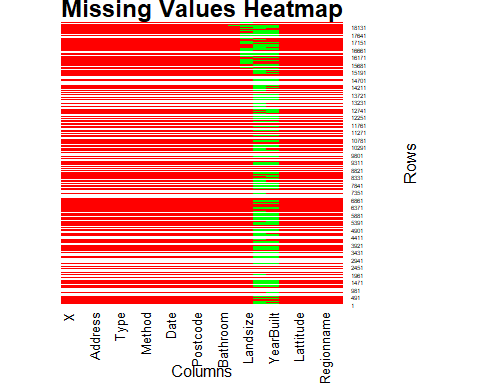
## X Suburb Address Rooms Type   
## 0.000000000 0.000000000 0.000000000 0.000000000 0.000000000   
## Price Method SellerG Date Distance   
## 0.000000000 0.000000000 0.000000000 0.000000000 0.005435964   
## Postcode Bedroom2 Bathroom Car Landsize   
## 0.005435964 18.857360296 18.868232224 19.439008480 26.054577082   
## BuildingArea YearBuilt CouncilArea Lattitude Longtitude   
## 57.806044792 51.304631442 0.000000000 18.112633181 18.112633181   
## Regionname Propertycount   
## 0.000000000 0.005435964

head(housing)

## X Suburb Address Rooms Type Price Method SellerG Date  
## 1 1 Abbotsford 85 Turner St 2 h 1480000 S Biggin 3/12/2016  
## 2 2 Abbotsford 25 Bloomburg St 2 h 1035000 S Biggin 4/02/2016  
## 3 4 Abbotsford 5 Charles St 3 h 1465000 SP Biggin 4/03/2017  
## 4 5 Abbotsford 40 Federation La 3 h 850000 PI Biggin 4/03/2017  
## 5 6 Abbotsford 55a Park St 4 h 1600000 VB Nelson 4/06/2016  
## 6 10 Abbotsford 129 Charles St 2 h 941000 S Jellis 7/05/2016  
## Distance Postcode Bedroom2 Bathroom Car Landsize BuildingArea YearBuilt  
## 1 2.5 3067 2 1 1 202 NA NA  
## 2 2.5 3067 2 1 0 156 79 1900  
## 3 2.5 3067 3 2 0 134 150 1900  
## 4 2.5 3067 3 2 1 94 NA NA  
## 5 2.5 3067 3 1 2 120 142 2014  
## 6 2.5 3067 2 1 0 181 NA NA  
## CouncilArea Lattitude Longtitude Regionname Propertycount  
## 1 Yarra -37.7996 144.9984 Northern Metropolitan 4019  
## 2 Yarra -37.8079 144.9934 Northern Metropolitan 4019  
## 3 Yarra -37.8093 144.9944 Northern Metropolitan 4019  
## 4 Yarra -37.7969 144.9969 Northern Metropolitan 4019  
## 5 Yarra -37.8072 144.9941 Northern Metropolitan 4019  
## 6 Yarra -37.8041 144.9953 Northern Metropolitan 4019

theres no data in EASE.MENT AND APARTMENT.NUMBER colomns so we drop this

# Assuming 'housing' is a data frame in R  
# Load the graphics package  
library(graphics)  
  
# Create a matrix of missing values (1 for missing, 0 for not missing)  
missing\_matrix <- ifelse(is.na(housing), 1, 0)  
  
# Set up the plot  
heatmap(missing\_matrix,   
 Colv = NA,   
 Rowv = NA,   
 col = c("red", 'green'), # Specify colors for missing (grey) and non-missing (white)  
 xlab = "Columns",   
 ylab = "Rows",   
 main = "Missing Values Heatmap")

 remove missing value

# Assuming your dataset is named 'data'  
housing <- na.omit(housing)

str(housing)

## 'data.frame': 6830 obs. of 22 variables:  
## $ X : int 2 4 6 11 14 18 24 25 30 32 ...  
## $ Suburb : chr "Abbotsford" "Abbotsford" "Abbotsford" "Abbotsford" ...  
## $ Address : chr "25 Bloomburg St" "5 Charles St" "55a Park St" "124 Yarra St" ...  
## $ Rooms : int 2 3 4 3 2 2 3 2 2 3 ...  
## $ Type : chr "h" "h" "h" "h" ...  
## $ Price : num 1035000 1465000 1600000 1876000 1636000 ...  
## $ Method : chr "S" "SP" "VB" "S" ...  
## $ SellerG : chr "Biggin" "Biggin" "Nelson" "Nelson" ...  
## $ Date : chr "4/02/2016" "4/03/2017" "4/06/2016" "7/05/2016" ...  
## $ Distance : num 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 ...  
## $ Postcode : num 3067 3067 3067 3067 3067 ...  
## $ Bedroom2 : num 2 3 3 4 2 3 3 2 2 3 ...  
## $ Bathroom : num 1 2 1 2 1 1 2 2 1 2 ...  
## $ Car : num 0 0 2 0 2 2 2 1 2 1 ...  
## $ Landsize : num 156 134 120 245 256 220 214 0 238 113 ...  
## $ BuildingArea : num 79 150 142 210 107 75 190 94 97 110 ...  
## $ YearBuilt : num 1900 1900 2014 1910 1890 ...  
## $ CouncilArea : chr "Yarra" "Yarra" "Yarra" "Yarra" ...  
## $ Lattitude : num -37.8 -37.8 -37.8 -37.8 -37.8 ...  
## $ Longtitude : num 145 145 145 145 145 ...  
## $ Regionname : chr "Northern Metropolitan" "Northern Metropolitan" "Northern Metropolitan" "Northern Metropolitan" ...  
## $ Propertycount: num 4019 4019 4019 4019 4019 ...  
## - attr(\*, "na.action")= 'omit' Named int [1:11566] 1 4 6 9 10 11 13 14 15 16 ...  
## ..- attr(\*, "names")= chr [1:11566] "1" "4" "6" "9" ...

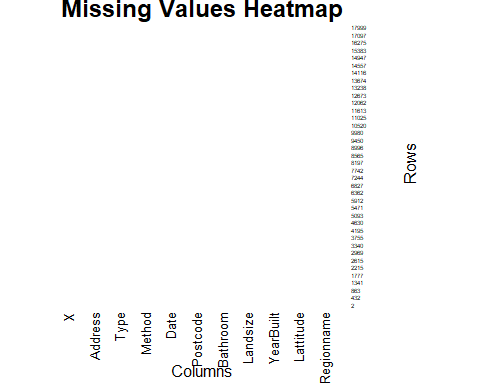
dim(housing)

## [1] 6830 22

# Assuming 'housing' is a data frame in R  
# Load the graphics package  
library(graphics)  
  
# Create a matrix of missing values (1 for missing, 0 for not missing)  
missing\_matrix <- ifelse(is.na(housing), 1, 0)  
  
# Set up the plot  
heatmap(missing\_matrix,   
 Colv = NA,   
 Rowv = NA,   
 col = c("red", 'green'), # Specify colors for missing (grey) and non-missing (white)  
 xlab = "Columns",   
 ylab = "Rows",   
 main = "Missing Values Heatmap")

## Warning in min(x): no non-missing arguments to min; returning Inf

## Warning in max(x): no non-missing arguments to max; returning -Inf



get cleaned dataset as data

write.csv(housing, file = "data.csv", row.names = FALSE)