Data Wrangling (Data Preprocessing)

Code

Practical assessment 2

Wong Yi Wei

26/5/2023

Setup

Hide

```
library(kableExtra)
library(magrittr)
library(readxl)
library(readr)
library(magrittr)
library(tidyverse)
library(tidyr)
library(lubridate)
library(stringr)
library(knitr)
library(zoo)
library(outliers)
library(MVN)
library(forecast)
```

Student names, numbers and percentage of contributions

Group information

Student name	Student number	Percentage of contribution
Wong Yi Wei	s3966890	50%
Adeyinka Kayode Freeman	s3960988	50%

Executive Summary

The aim of this analysis is to examine the relationship between crime rates and median rental values in various local government areas (LGA) in Melbourne, Victoria, Australia.

- **Get**
  - There are 2 data sets collected for this analysis. Firstly, the crime statistic data table from the DataVic website, which contains recorded offences in Victoria by LGA (DataVic 2023). The second data set comes from the Department of Families, Fairness and Housing (DFFH) website, which contains quarterly median rents around Victoria by LGA (DFFH 2023).
- **Understand**
  - There are a total of 870 observations and 6 variables in the crime dataset, and 94 observations and 194 variables in the rent dataset.
  - Since the crime data only contains the years 2013 to 2022, we further subset the rent dataset to these years for easier analysis.

    - The structure of the variables is checked and formatted.
- **Tidy and Manipulate**
    - We have removed unnecessary columns in the crime dataset.
    - There are only 79 LGA in the state of Victoria, hence we subsetted non-LGA's from both the data sets (VEC n.d.).
    - Following the tidy data principles, we have manipulated both data sets accordingly.
    - Both data sets are merged for further analysis.
    - New variables are also created from existing variables.
- **Scan**
    - Here missing values and obvious errors are accounted for and dealt with by using mean from their respective groups.
    - Through box plot graph mapping and other techniques, outliers are identified and managed with different approaches.
- **Transform**
    - Data transformations are applied in this step for incidents and median rent data.
    - For each skewed distributions, data normalising method is introduced to centre the data for easier understanding.

Data

There is a common conception that a higher crime rate at a suburb would lead to a lower rental price in the same suburb. To rectify this conception, we chose the crime statistic data set and the median rental prices data set by Local Government Area (LGA).

The first data used is the Crime Statistics Agency Data Tables - Criminal Incidents from the Crime Statistics Agency (CSA) (DataVic 2023). This data captures the number and rate of recorded offences in Victoria. The data format is in ".XLSX" extension and its for the period ending Dec 2022. The dataset from CSA was sourced from the URL: [https://discover.data.vic.gov.au/dataset/criminal-incident (https://discover.data.vic.gov.au/dataset/criminal-incident)]. There are 5 tables in this dataset. However,the table we will use is 'Table 01' as it excludes criminal incidents where geographic location is unknown. There are 6 variables and 870 observations for this data set. The variables of this data set is as follows (CSA n.d.):

- Year: The year of the recorded criminal incident
- Year ending: The year ending of the recorded criminal incident
- Police Region: The geographical area defined by Victoria Police for operational purposes.
- Local Government Area: The geographical area under the responsibility of the incorporated local government council.
- Incidents Recorded: The number of criminal incidents recorded.
- Rate per 100,000 population: The offence rate per 100,000 population for the criminal incident reference period and the most recent Estimated Resident Population (ERP).

Our second data source is from the March Quarterly Rental report for 2023 from the Department of Families, Fairness and Housing (DFFH) which empowers communities to build a fairer and safer Victoria (DFFH 2023). We are looking at the rental report on the private rental market in Victoria. The data format is in ".XLSX" extension and its for the first quarter of 2023. The dataset from DFFH was sourced from the URL: [https://www.dffh.vic.gov.au/publications/rental-report (https://www.dffh.vic.gov.au/publications/rental-report)]. There are 7 tables in this data set. However, the table we will use is

‘All Properties’ as it includes the median rent of all properties. There are 94 observations and 194 variables in this data. Due to the quarterly years spreading out from 1999 to 2023, there is a high volume of variables, but there are only 2 main variables in this data which are as follows:

- Count: Count of properties in the LGA
- Median: Median rental price of properties in the LGA

Through the R package readxl, we import the first dataset, crime statistics and label it as ‘crime’ by using the read_xlsx() function. Since we only need the third sheet in the data set, we add the argument, sheet=3 to import it. Then, using the head() function and argument of 3, the first 3 observations are printed.

Next, we import the second dataset, median rental price and label it as ‘rent’ using the read_xlsx() function. In this case, we need the seventh sheet in the data set, hence we add the argument, sheet=7 to import it. Using the head() function and argument of 3, we can view the first 3 observations of the data.

However, looking at the head of the data, there are multiple inconsistencies due to untidy data. Since we only need the median rent of the data from 2013 to 2022, we can then subset the data into a new dataset labelled ‘rent2013to2022’. Due to the nature of the data set having no variable name, we have to subset the data manually. Using the head() function and argument of 3, we can then view the first 3 observations of the new data set.

Hide

```
#Importing crime dataset
crime<-read_xlsx("C:/Users/ethan/Desktop/Practical Assessment 2/Data_Tables_LGA_Criminal_Incidents_Year_Ending_December_2022.xlsx",sheet=3)
head(crime,3)
```

Year <dbl>	Year ending <chr>	Police Region <chr>	Local Government Area <chr>	Incidents Recorded <dbl>
2022	December	1 North West Metro	Banyule	5387
2022	December	1 North West Metro	Brimbank	12156
2022	December	1 North West Metro	Darebin	9321

3 rows | 1-5 of 6 columns

Hide

```
#Importing median rent dataset
rent<-read_xlsx("C:/Users/ethan/Desktop/Practical Assessment 2/Quarterly median rents by local government area - March quarter 2023.xlsx",sheet=7)
```

```
New names:
```

Hide

```
head(rent,3)
```

Quarterly median rents by LGA <chr>	...2 <chr>	...3 <chr>	...4 <chr>	...5 <chr>	...6 <chr>	...7 <chr>	...8 <chr>
All properties	*NA*	Jun 1999	*NA*	Sep 1999	*NA*	Dec 1999	*NA*
NA	*NA*	Count	Median	Count	Median	Count	Median
Barwon South West	Colac-Otway	99	115	80	113	82	115

3 rows | 1-8 of 194 columns

Hide

```
#Subset to new dataset
rent2013to2022<-rent[3:90,c(2,120,128,136,144,152,160,168,176,184,192)]
head(rent2013to2022,3)
```

...2 <chr>	...120 <chr>	...128 <chr>	...136 <chr>	...144 <chr>	...152 <chr>	...160 <chr>	...168 <chr>	...176 <chr>	...184 <chr>
Colac-Otway	250	258	260	260	285	300	305	325	380
Corangamite	210	230	235	245	240	250	260	290	343
Glenelg	225	220	210	223	228	250	270	310	350

3 rows | 1-10 of 11 columns

Understand

Hide

```
colnames(crime)<-c("Year","Ending","Region","LGA","Incidents","Rate per 100,000")
str(crime)
```

```
tibble [870 × 6] (S3: tbl_df/tbl/data.frame)
 $ Year            : num [1:870] 2022 2022 2022 2022 2022 ...
 $ Ending          : chr [1:870] "December" "December" "December" "December" ...
 $ Region          : chr [1:870] "1 North West Metro" "1 North West Metro" "1 North West Metro" "1 Nort
h West Metro" ...
 $ LGA             : chr [1:870] "Banyule" "Brimbank" "Darebin" "Hobsons Bay" ...
 $ Incidents       : num [1:870] 5387 12156 9321 4664 12753 ...
 $ Rate per 100,000: num [1:870] 4264 6248 6246 5089 5066 ...
```

Hide

```
dim(crime)
```

```
[1] 870   6
```

Hide

```
unique(crime$Year)
```

```
 [1] 2022 2021 2020 2019 2018 2017 2016 2015 2014 2013
```

Hide

```
crime$Year<-factor(crime$Year,levels=c(2013,2014,2015,2016,2017,2018,2019,2020,2021,2022),ordered = TRU
E)

unique(crime$Region)
```

```
[1] "1 North West Metro"                                "2 Eastern"
"3 Southern Metro"
[4] "4 Western"                                         "Justice Institutions and Immigration Facilities"
"Unincorporated Vic"
```

Hide

```
crime$Region<-factor(crime$Region,levels=c("1 North West Metro","2 Eastern","3 Southern Metro","4 Weste
rn","Justice Institutions and Immigration Facilities","Unincorporated Vic"),labels=c("North West Metr
o","Eastern","Southern Metro","Western","Justice Institutions and Immigration Facilities","Unincorporat
ed Vic"))

crime$LGA<-as.factor(crime$LGA)

colnames(rent2013to2022)<-c("LGA","2013","2014","2015","2016","2017","2018","2019","2020","2021","202
2")
str(rent2013to2022)
```

```
tibble [88 × 11] (S3: tbl_df/tbl/data.frame)
 $ LGA : chr [1:88] "Colac-Otway" "Corangamite" "Glenelg" "Greater Geelong" ...
 $ 2013: chr [1:88] "250" "210" "225" "300" ...
 $ 2014: chr [1:88] "258" "230" "220" "310" ...
 $ 2015: chr [1:88] "260" "235" "210" "315" ...
 $ 2016: chr [1:88] "260" "245" "223" "330" ...
 $ 2017: chr [1:88] "285" "240" "228" "345" ...
 $ 2018: chr [1:88] "300" "250" "250" "360" ...
 $ 2019: chr [1:88] "305" "260" "270" "375" ...
 $ 2020: chr [1:88] "325" "290" "310" "380" ...
 $ 2021: chr [1:88] "380" "343" "350" "415" ...
 $ 2022: chr [1:88] "398" "360" "390" "450" ...
```

Hide

```
dim(rent2013to2022)
```

```
[1] 88 11
```

Hide

```
unique(rent2013to2022$LGA)
```

```
 [1] "Colac-Otway"        "Corangamite"        "Glenelg"            "Greater Geelong"    "Moyne"
"Queenscliffe"       "Southern Grampians"
 [8] "Surf Coast"         "Warrnambool"        "Group Total"        "Ararat"             "Ballarat"
"Golden Plains"      "Hepburn"
[15] "Hindmarsh"          "Horsham"            "Moorabool"          "Northern Grampians" "Pyrenees"
"West Wimmera"       "Yarriambiack"
[22] "Buloke"             "Campaspe"           "Central Goldfields" "Gannawarra"         "Greater Bendi
go"    "Loddon"             "Macedon Ranges"
[29] "Mildura"            "Mount Alexander"    "Swan Hill"          "Alpine"             "Benalla"
"Greater Shepparton" "Indigo"
[36] "Mansfield"          "Mitchell"           "Moira"              "Murrindindi"        "Strathbogie"
"Towong"             "Wangaratta"
[43] "Wodonga"            "Bass Coast"         "Baw Baw"            "East Gippsland"     "Latrobe"
"South Gippsland"    "Wellington"
[50] "Banyule"            "Brimbank"           "Darebin"            "Hobsons Bay"        "Hume"
"Maribyrnong"        "Melbourne"
[57] "Melton"             "Merri-bek"          "Moonee Valley"      "Nillumbik"          "Whittlesea"
"Wyndham"            "Yarra"
[64] "Boroondara"         "Knox"               "Manningham"         "Maroondah"          "Monash"
"Whitehorse"         "Yarra Ranges"
[71] "Bayside"            "Cardinia"           "Casey"              "Frankston"          "Glen Eira"
"Greater Dandenong"  "Kingston"
[78] "Mornington Penin'a" "Port Phillip"       "Stonnington"        "Victoria"
```

Hide

```
rent2013to2022$LGA<-factor(rent2013to2022$LGA,levels=c("Colac-Otway","Corangamite","Glenelg","Greater G
eelong","Moyne","Queenscliffe","Southern Grampians","Surf Coast","Warrnambool","Group Total","Arara
t","Ballarat","Golden Plains","Hepburn","Hindmarsh","Horsham","Moorabool","Northern Grampians","Pyrenee
s","West Wimmera", "Yarriambiack","Buloke","Campaspe","Central Goldfields","Gannawarra","Greater Bendig
o","Loddon","Macedon Ranges","Mildura","Mount Alexander","Swan Hill","Alpine", "Benalla","Greater Shepp
arton","Indigo","Mansfield","Mitchell","Moira",          "Murrindindi","Strathbogie","Towong","Wangarat
ta","Wodonga","Bass Coast","Baw Baw", "East Gippsland","Latrobe","South Gippsland","Wellington","Banyul
e","Brimbank",    "Darebin","Hobsons Bay","Hume","Maribyrnong","Melbourne","Melton",            "Merri
-bek","Moonee Valley","Nillumbik","Whittlesea","Wyndham","Yarra","Boroondara","Knox","Manningham","Maro
ondah","Monash","Whitehorse","Yarra Ranges","Bayside",   "Cardinia","Casey","Frankston","Glen Eira","Gr
eater Dandenong","Kingston","Mornington Penin'a","Port Phillip","Stonnington","Victoria"),labels=c("Col
ac-Otway","Corangamite","Glenelg","Greater Geelong","Moyne","Queenscliffe","Southern Grampians","Surf C
oast","Warrnambool","Group Total","Ararat","Ballarat","Golden Plains","Hepburn","Hindmarsh","Horsha
m","Moorabool","Northern Grampians","Pyrenees","West Wimmera", "Yarriambiack","Buloke","Campaspe","Cent
ral Goldfields","Gannawarra","Greater Bendigo","Loddon","Macedon Ranges","Mildura","Mount Alexander","S
wan Hill","Alpine", "Benalla","Greater Shepparton","Indigo","Mansfield","Mitchell","Moira","Murrindind
i","Strathbogie","Towong","Wangaratta","Wodonga","Bass Coast","Baw Baw", "East Gippsland","Latrobe","So
uth Gippsland","Wellington","Banyule","Brimbank","Darebin","Hobsons Bay","Hume","Maribyrnong","Melbourn
e","Melton","Merri-bek","Moonee Valley","Nillumbik","Whittlesea","Wyndham","Yarra","Boroondara","Kno
x","Manningham","Maroondah","Monash","Whitehorse","Yarra Ranges","Bayside","Cardinia","Casey","Franksto
n","Glen Eira","Greater Dandenong","Kingston","Mornington Peninsula","Port Phillip","Stonnington","Vict
oria"))

rent2013to2022$'2013'<-as.numeric(rent2013to2022$'2013')
rent2013to2022$'2014'<-as.numeric(rent2013to2022$'2014')
rent2013to2022$'2015'<-as.numeric(rent2013to2022$'2015')
rent2013to2022$'2016'<-as.numeric(rent2013to2022$'2016')
```

```
Warning: NAs introduced by coercion
```

Hide

```
rent2013to2022$'2017'<-as.numeric(rent2013to2022$'2017')
```

```
Warning: NAs introduced by coercion
```

Hide

```
rent2013to2022$'2018'<-as.numeric(rent2013to2022$'2018')
```

```
Warning: NAs introduced by coercion
```

Hide

```
rent2013to2022$'2019'<-as.numeric(rent2013to2022$'2019')
```

```
Warning: NAs introduced by coercion
```

Hide

```
rent2013to2022$'2020'<-as.numeric(rent2013to2022$'2020')
rent2013to2022$'2021'<-as.numeric(rent2013to2022$'2021')
rent2013to2022$'2022'<-as.numeric(rent2013to2022$'2022')
```

```
Warning: NAs introduced by coercion
```

Hide

```
str(crime)
```

```
tibble [870 × 6] (S3: tbl_df/tbl/data.frame)
 $ Year            : Ord.factor w/ 10 levels "2013"<"2014"<..: 10 10 10 10 10 10 10 10 10 10 ...
 $ Ending          : chr [1:870] "December" "December" "December" "December" ...
 $ Region          : Factor w/ 6 levels "North West Metro",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ LGA             : Factor w/ 82 levels "Alpine","Ararat",..: 4 10 18 31 33 43 45 46 47 52 ...
 $ Incidents       : num [1:870] 5387 12156 9321 4664 12753 ...
 $ Rate per 100,000: num [1:870] 4264 6248 6246 5089 5066 ...
```

Hide

```
str(rent2013to2022)
```

```
tibble [88 × 11] (S3: tbl_df/tbl/data.frame)
 $ LGA : Factor w/ 81 levels "Colac-Otway",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ 2013: num [1:88] 250 210 225 300 260 325 220 375 280 295 ...
 $ 2014: num [1:88] 258 230 220 310 270 275 230 400 280 300 ...
 $ 2015: num [1:88] 260 235 210 315 280 340 230 410 280 300 ...
 $ 2016: num [1:88] 260 245 223 330 300 363 220 450 285 310 ...
 $ 2017: num [1:88] 285 240 228 345 283 NA 240 450 290 330 ...
 $ 2018: num [1:88] 300 250 250 360 300 390 240 465 318 350 ...
 $ 2019: num [1:88] 305 260 270 375 340 405 260 450 330 370 ...
 $ 2020: num [1:88] 325 290 310 380 363 450 280 470 350 380 ...
 $ 2021: num [1:88] 380 343 350 415 360 385 305 500 400 410 ...
 $ 2022: num [1:88] 398 360 390 450 400 NA 310 540 420 440 ...
```

- Firstly, we change the column names of the crime data set to allow for easier manipulation of the data by using the colnames() function. We also check the structure and dimension of the crime data set. From the structure of the data set, we can see that year, region, and LGA has to be changed to a factor variable. By using the dim() function, we can see that there are 870 observations and 6 variables.
- Next, we use the unique() function to see any unique values in the Year variable and we can factor it accordingly with order. Similarly in the Region variable, we can the factor it and relabel the observations for easier understanding of the data. Next, LGA is change to a factor type by just using the as.factor() function as no other relabeling is necessary. Lastly, we can verify the changes made to the crime table by using the str() function again.
- Similarly to the first step in the crime data set, we change the column names accordingly by year using colnames() function and we can also check the structure and dimension of the median rent data set. From the str() function, we can see that all the variables are in character format and we have to change the LGA variable to factor, and the rest as numerical data type.
- Using the unique() function, we can check the unique values in the LGA variable and we can see that the name for mornington penin'a needs to be renamed for consistency. Using the factor() function, we can change the data type and relabel the incorrect names. Next, the rest of the variables can be converted into numeric data type by using the as.numeric() function.
- Lastly, we can verify changes made to both tables by using the str() function. We can see here that the minimum requirements 2-4 are satisfied as there are multiple data types, data types are converted accordingly, and there are at least one factor variable.

Tidy & Manipulate Data I

Hide

```
crime<-crime %>% select(-Ending)
crime<-subset(crime,crime$LGA != "Total" & crime$LGA != "Justice Institutions and Immigration Facilitie
s" & crime$LGA !="Unincorporated Vic")
crime$`Rate per 100,000`<-round(crime$`Rate per 100,000`,digits=2)

rent2013to2022<-rent2013to2022 %>% pivot_longer(names_to="Year",values_to="Median",cols=2:11)

rent2013to2022<-subset(rent2013to2022,rent2013to2022$LGA != "Group Total" & rent2013to2022$LGA != "Vict
oria")

crimerent<-inner_join(crime,rent2013to2022,by=c("Year","LGA"))
colnames(crimerent)<-c("Year","Region","LGA", "Incidents","Rate per 100,000","Median Rent")
crimerent<-arrange(crimerent,LGA)
head(crimerent,3)
```

Year	Region	LGA	Incidents	Rate per 100,000	Median Rent
<chr>	<fctr>	<fctr>	<dbl>	<dbl>	<dbl>
2022	Eastern	Alpine	349	2635.29	420
2021	Eastern	Alpine	386	2934.02	350
2020	Eastern	Alpine	413	3170.10	350

3 rows

- Firstly, we can remove the year ending column from the crime table as the data are all recorded in year ending December and it is unnecessary. Furthermore, we can subset the unrelated LGA's in the LGA column from crime data as there are only 79 LGAs in Victoria. We also round the Rate per 100,000 variable to 2 decimal places.

- Next, by applying the tidy data principle of each variable must have it's own column in the rent2013to2022 data, we can use the pivot_longer() function as the years are values instead of variables. Then, We also subset unrelated LGA's from the rent2013to2022 data for consistency.
- By using the inner_join function, we can join and merge both crime and rent2013to2022 data using the Year and LGA variable as a new dataset 'crimerent'. Then, using colnames() function we rename the columns appropriately. We also arrange the crimerent table by LGA using the arrange() function.
- The head() function shows us the first 3 observations in the crimerent data.

Tidy & Manipulate Data II

Hide

```
crimerent<-crimerent %>% group_by(LGA) %>% arrange(Year,.by_group = TRUE) %>% mutate("Percentage Change
(Incidents)" = round((((lag(Incidents)-Incidents)/lag(Incidents))*100), digits=2)) %>% relocate("Percen
tage Change (Incidents)",.after=Incidents)
```

- Here we have created a new variable of percentage change in incidents grouped by LGA. We can use the formula of percentage change [old value - new value/old value]*100 and use the round() function to round it to 2 decimal places. We also used the relocate() function to move it after Incidents.
- Due to the NA values in the median rent data, we are unable to create a percentage change variable for median rent. However, this issue is rectified in the next section.

Scan I

Hide

```
colSums(is.na(crimerent))
```

```
                        Year                       Region                          LGA
Incidents Percentage Change (Incidents)
                           0                            0                            0
0                           79
             Rate per 100,000                  Median Rent
                           0                            5
```

Hide

```
crimerent$`Percentage Change (Incidents)`[is.na(crimerent$`Percentage Change (Incidents)`)]<-0

crimerent<-crimerent %>% group_by(LGA) %>% arrange(Year,.by_group = TRUE) %>% mutate_at(vars('Median Re
nt'), ~replace_na(.,mean(.,na.rm=TRUE)))

crimerent$`Median Rent`<-round(crimerent$`Median Rent`,digits=0)

crimerent<-crimerent %>% group_by(LGA) %>% arrange(Year,.by_group = TRUE) %>% mutate("Percentage Change
(Rent)" = round((((lag(`Median Rent`)-`Median Rent`)/lag(`Median Rent`))*100),digits=2))

crimerent$`Percentage Change (Rent)`[is.na(crimerent$`Percentage Change (Rent)`)]<-0

colSums(is.na(crimerent))
```

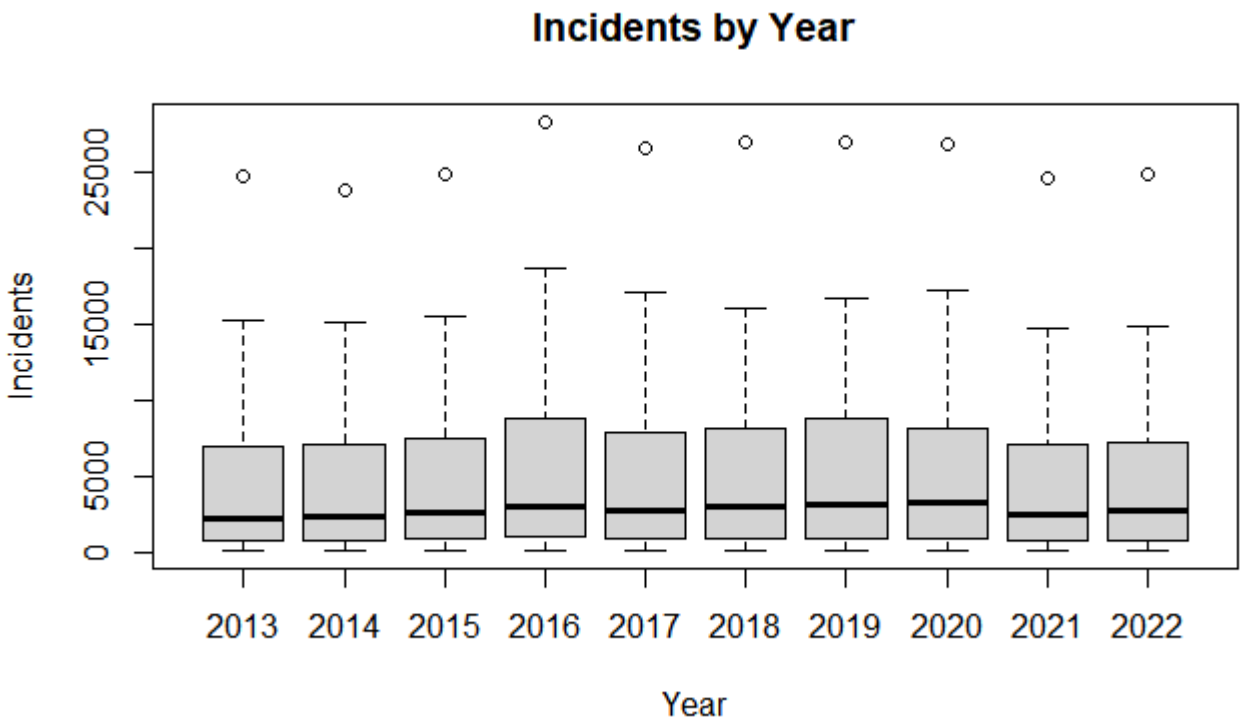
```
                        Year                       Region                          LGA
Incidents Percentage Change (Incidents)
                           0                            0                            0
0                            0
             Rate per 100,000                Median Rent      Percentage Change (Rent)
                           0                            0                            0
```

- Firstly, by using the is.na() and colSums() function, we can identify the number of NAs in the crimerent data. Here we can see that there are 79 NAs in Percentage Change (Incidents) and 5 in Median Rent.
- Since percentage change starts from 2013 to 2022, it make sense for 2013 to have a NA as it should be 0 as there are no changes. We have then changed all NA values in this variable to 0. For the median rent NAs, we replaced all NAs with the mean of its group by year and round it to 0 decimal places. We used the mean of its group by year as we can use the average of its group to fill out the missing NAs.
- Since there are no NAs in the median rent table, we can then create the percentage change column for median rent. Similarly to Percentage Change (Incidents), we have used the same methods to change the NAs to 0 value.
- Lastly, we can use the colSums() and is.na() functions again to verify that there are no further missing values.

Scan II

Hide

```
boxplot(crimerent$Incidents ~ crimerent$Year,main="Incidents by Year",ylab="Incidents",xlab="Year")
```



Hide

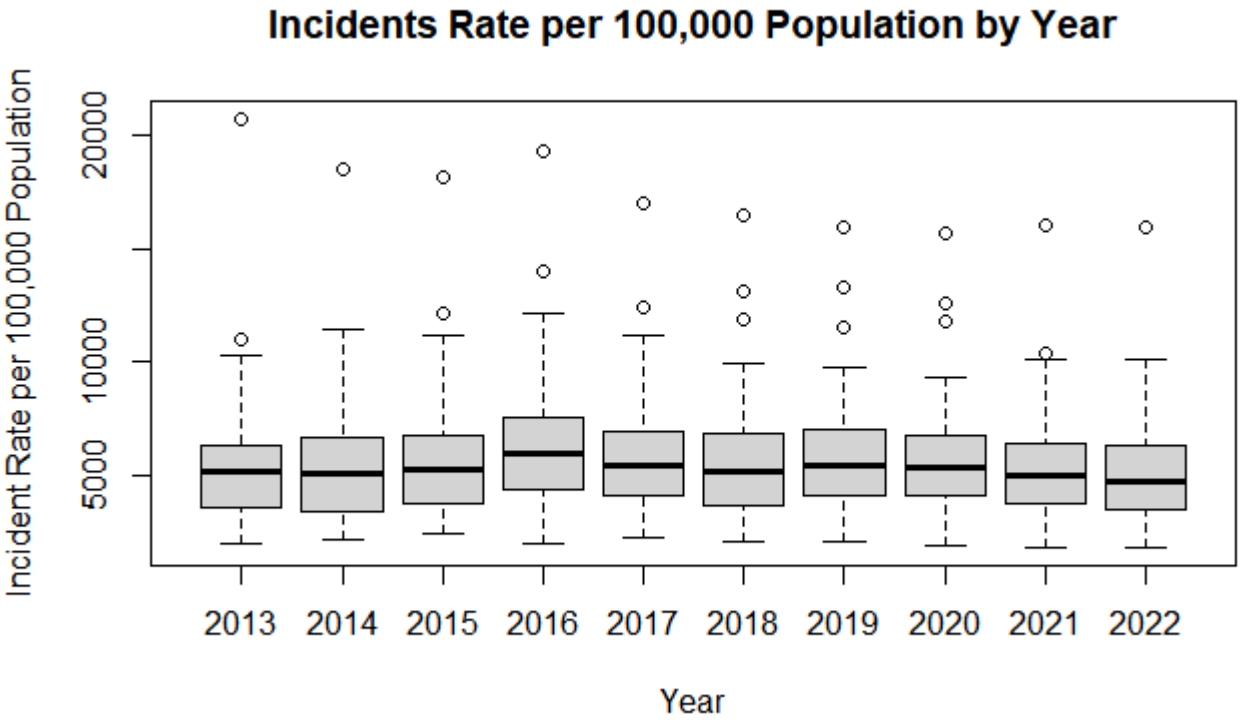
```
crimerent[which(abs(crimerent$Incidents) %>% scores(type="z")>3),c(1,2,3,4)]
```

Year <chr>	Region <fctr>	LGA <fctr>	Incidents <dbl>
2013	North West Metro	Melbourne	24650
2014	North West Metro	Melbourne	23745
2015	North West Metro	Melbourne	24868
2016	North West Metro	Melbourne	28281
2017	North West Metro	Melbourne	26537
2018	North West Metro	Melbourne	26922
2019	North West Metro	Melbourne	26989
2020	North West Metro	Melbourne	26773
2021	North West Metro	Melbourne	24631
2022	North West Metro	Melbourne	24785

1-10 of 10 rows

Hide

```
boxplot(crimerent$`Rate per 100,000` ~ crimerent$Year,main="Incidents Rate per 100,000 Population by Ye
ar",ylab="Incident Rate per 100,000 Population",xlab="Year")
```



Hide

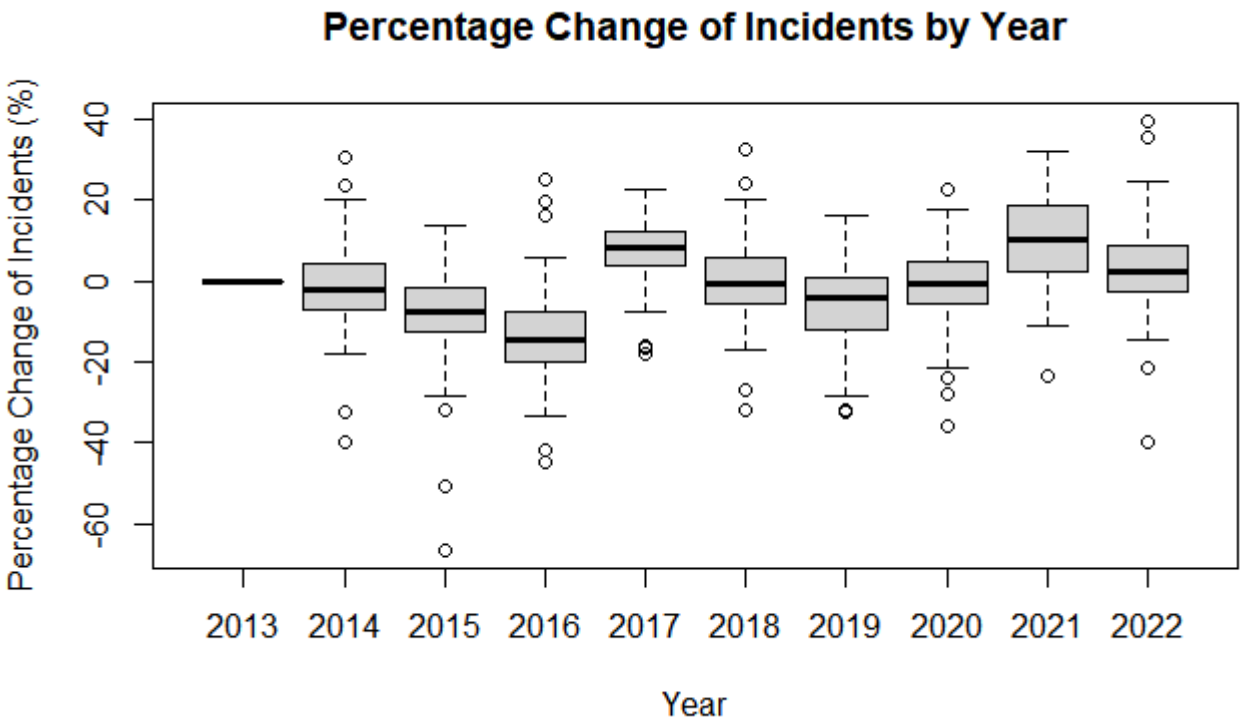
```
crimerent[which(abs(crimerent$`Rate per 100,000`) %>% scores(type="z")>3),c(1,2,3,6) ]
```

Year <chr>	Region <fctr>	LGA <fctr>	Rate per 100,000 <dbl>
2016	Eastern	Latrobe	14002.57
2018	Eastern	Latrobe	13119.60
2019	Eastern	Latrobe	13284.93
2013	North West Metro	Melbourne	20764.71
2014	North West Metro	Melbourne	18554.41
2015	North West Metro	Melbourne	18168.80
2016	North West Metro	Melbourne	19357.82
2017	North West Metro	Melbourne	17010.90
2018	North West Metro	Melbourne	16469.88
2019	North West Metro	Melbourne	15958.30

1-10 of 13 rows Previous **1** 2 Next

Hide

```
boxplot(crimerent$`Percentage Change (Incidents)` ~ crimerent$Year,main="Percentage Change of Incidents by Year",ylab="Percentage Change of Incidents (%)",xlab="Year")
```



Hide

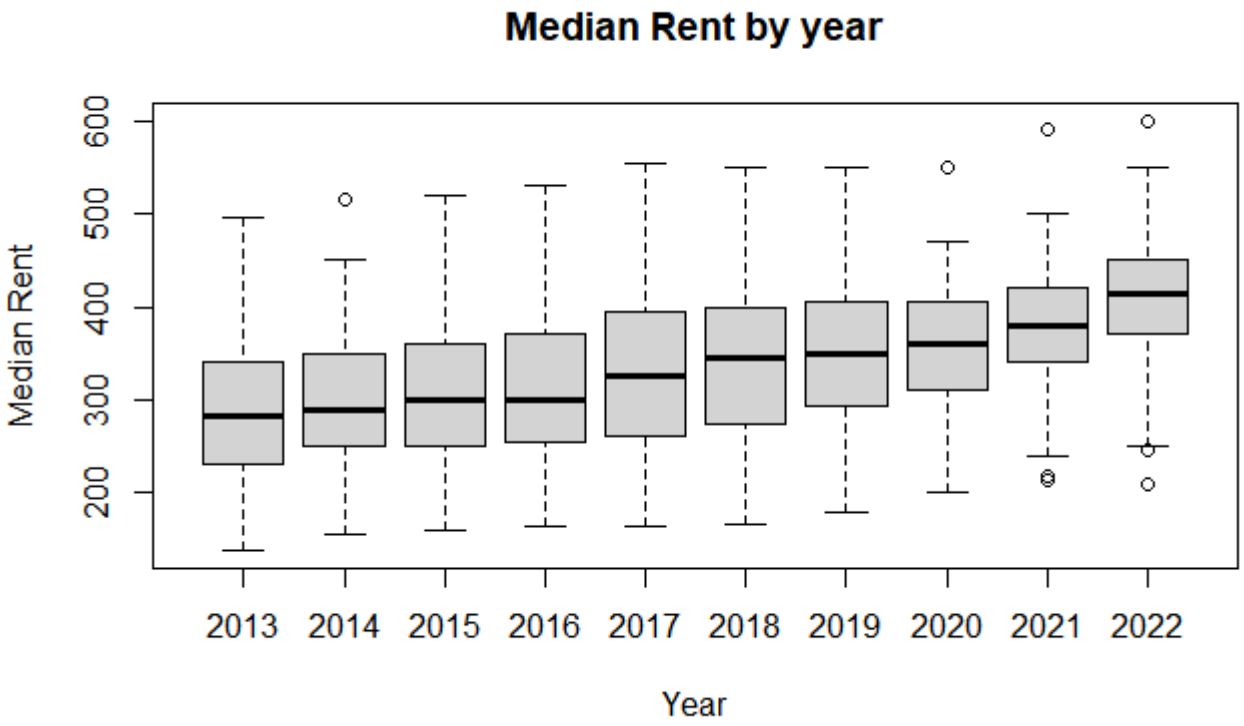
```
crimerent[which(abs(crimerent$`Percentage Change (Incidents)`) %>% scores(type="z")>3),c(1,2,3,5) ]
```

Year <chr>	Region <fctr>	LGA <fctr>	Percentage Change (Incidents) <dbl>
2016	Western	Corangamite	-44.86
2016	Western	Hindmarsh	-45.06
2020	Western	Hindmarsh	-35.81
2022	Western	Hindmarsh	35.39
2022	Eastern	Indigo	-39.75
2016	Eastern	Mansfield	-41.89
2014	Western	Queenscliffe	-39.73
2015	Western	Queenscliffe	-50.98
2022	Western	West Wimmera	39.51
2015	Western	Yarriambiack	-66.67

1-10 of 10 rows

Hide

```
boxplot(crimerent$`Median Rent` ~ crimerent$Year, main="Median Rent by year",ylab="Median Rent",xlab="Year")
```



Hide

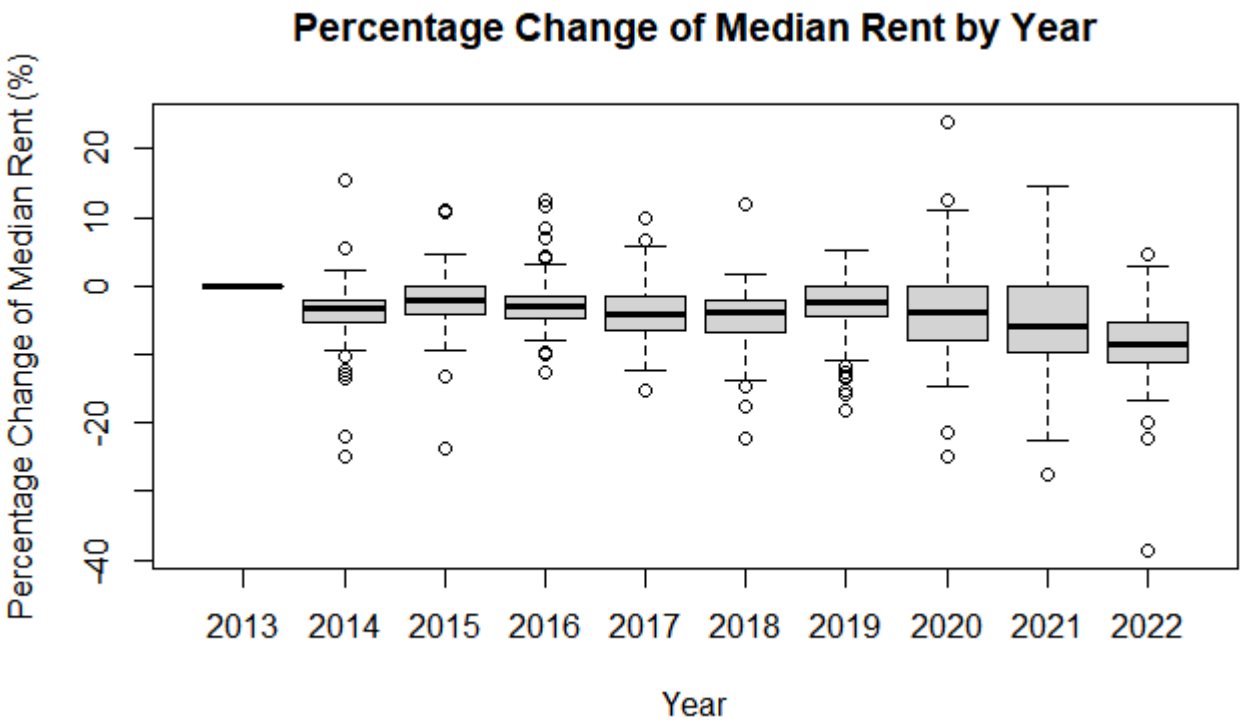
```
crimerent[which(abs(crimerent$`Median Rent`) %>% scores(type="z")>3),c(1,2,3,7) ]
```

Year	Region	LGA	Median Rent
<chr>	<fctr>	<fctr>	<dbl>
2021	Southern Metro	Bayside	590
2022	Southern Metro	Bayside	600

2 rows

Hide

```
boxplot(crimerent$`Percentage Change (Rent)` ~ crimerent$Year, main="Percentage Change of Median Rent b
y Year",ylab="Percentage Change of Median Rent (%)",xlab="Year")
```



Hide

```
crimerent[which(abs(crimerent$`Percentage Change (Rent)`) %>% scores(type="z")>3), c(1,2,3,8)]
```

Year	Region	LGA	Percentage Change (Rent)
<chr>	<fctr>	<fctr>	<dbl>
2022	Eastern	Alpine	-20.00
2021	Eastern	Bass Coast	-20.00
2021	Western	Hindmarsh	-27.50
2021	Eastern	Indigo	-22.58
2021	Eastern	Latrobe	-19.30
2014	Western	Loddon	-25.00
2020	North West Metro	Melbourne	24.00
2022	North West Metro	Melbourne	-38.67
2018	Western	Northern Grampians	-22.22
2014	Western	Pyrenees	-21.95

1-10 of 16 rows

Previous **1** 2 Next

Hide

```
cap <- function(x){
    quantiles <- quantile( x, c(.05,.95))
    x[ x < quantiles[1] ] <- quantiles[1]
    x[ x > quantiles[2] ] <- quantiles[2]
    x
}

crimerentsub<-crimerent %>% dplyr::select(LGA,`Rate per 100,000`,`Percentage Change (Incidents)`,`Perce
ntage Change (Rent)`)

crimerentcapped<-sapply(crimerentsub[2:4],FUN=cap)

summary(crimerentsub)
```

```
         LGA      Rate per 100,000 Percentage Change (Incidents) Percentage Change (Rent)
 Alpine    : 10   Min.   : 1764    Min.   :-66.670               Min.   :-38.670
 Ararat    : 10   1st Qu.: 3769    1st Qu.: -7.630               1st Qu.: -6.550
 Ballarat  : 10   Median : 5222    Median :  0.000               Median : -3.030
 Banyule   : 10   Mean   : 5563    Mean   : -1.127               Mean   : -3.837
 Bass Coast: 10   3rd Qu.: 6760    3rd Qu.:  5.600               3rd Qu.:  0.000
 Baw Baw   : 10   Max.   :20765    Max.   : 39.510               Max.   : 24.000
 (Other)   :730
```

Hide

```
summary(crimerentcapped)
```

```
 Rate per 100,000 Percentage Change (Incidents) Percentage Change (Rent)
 Min.   :2558     Min.   :-21.4725              Min.   :-13.133
 1st Qu.:3769     1st Qu.: -7.6300              1st Qu.: -6.550
 Median :5222     Median :  0.0000              Median : -3.030
 Mean   :5413     Mean   : -0.8919              Mean   : -3.806
 3rd Qu.:6760     3rd Qu.:  5.6000              3rd Qu.:  0.000
 Max.   :9283     Max.   : 18.8440              Max.   :  3.171
```

- Firstly, we use the boxplot() function on Incidents variable to check for outliers. In this plot, we can see that there are 10 outliers in the data. By using the which() and abs() function we can view which variables are outliers, defined by its z-scores absolute value greater than 3. We can see here that all these outliers are from Melbourne. However, it make sense as these incidents are recorded in the city and can be left out to be investigated further.
- Using the boxplot() function on Incidents per 100,000 we can see that there are multiple outliers in this variable. Using the which() and abs() function we can see that these data converge around Latrobe and Melbourne. Interestingly, since Latrobe is further away from the city there should have lesser criminal incidents. However, this can be dealt with in the later steps.
- Using boxplot() function on Percentage Change of Incidents shows multiple outliers detected in the variable. Using which() and abs() function shows that these incidents shows multiple outliers converging around different LGAs. This can also be dealt with in the later steps.
- Using boxplot() function on Median Rent shows the outliers detected in the variable. Looking in depth by using which() and abs() function, we can see that interestingly, only 2 observations have a z-score absolute value greater than 3. Since Bayside is closer to the beach, it make sense on them having a high median rent, hence this can be left out and be investigated further.
- Using boxplot() function on Percentage Change of Median Rent shows multiple outliers detected in the variable. Using which() and abs() function shows that these incidents shows multiple outliers converging around different LGAs. This can also be dealt with in the later steps.

- After analyzing, we can further deduce that outliers in Incidents per 100,000, Percentage Change of Incidents, and Percentage Change of Median Rent can be dealt with. By using the capping or winsorising technique, we can replace the outliers with the nearest neighbours that are not outliers.
- To implement this technique, we can first create a function called cap which replaces the values with the 5th percentile for lower limit and 95th percentile for upper limit. We can then use this function on the selected variables as mentioned in the previous point.
- After applying the capping technique, we can see that by using the summary() function, the original and capped values have very different min, mean, and max values. This shows us that changes has been made and the values have been capped.

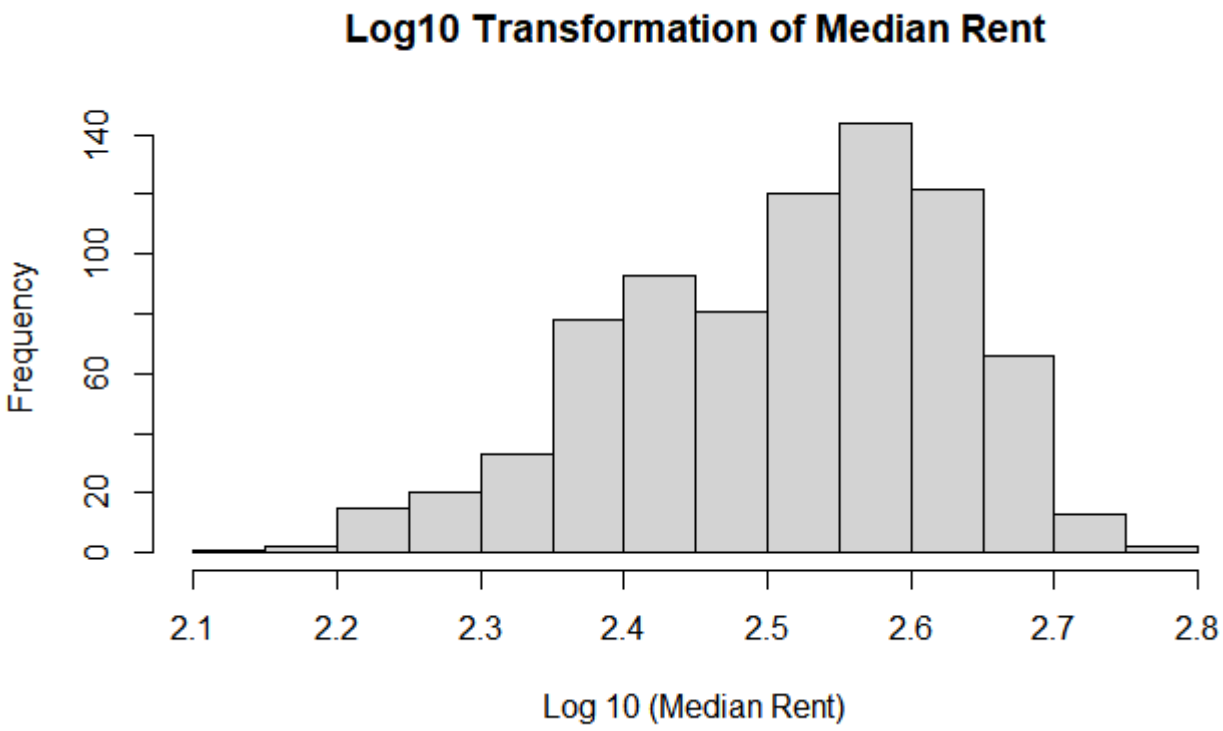
Transform

Hide

```
crimerent$`Median Rent` %>% hist(main="Histogram of Median Rent",xlab="Median Rent")
```

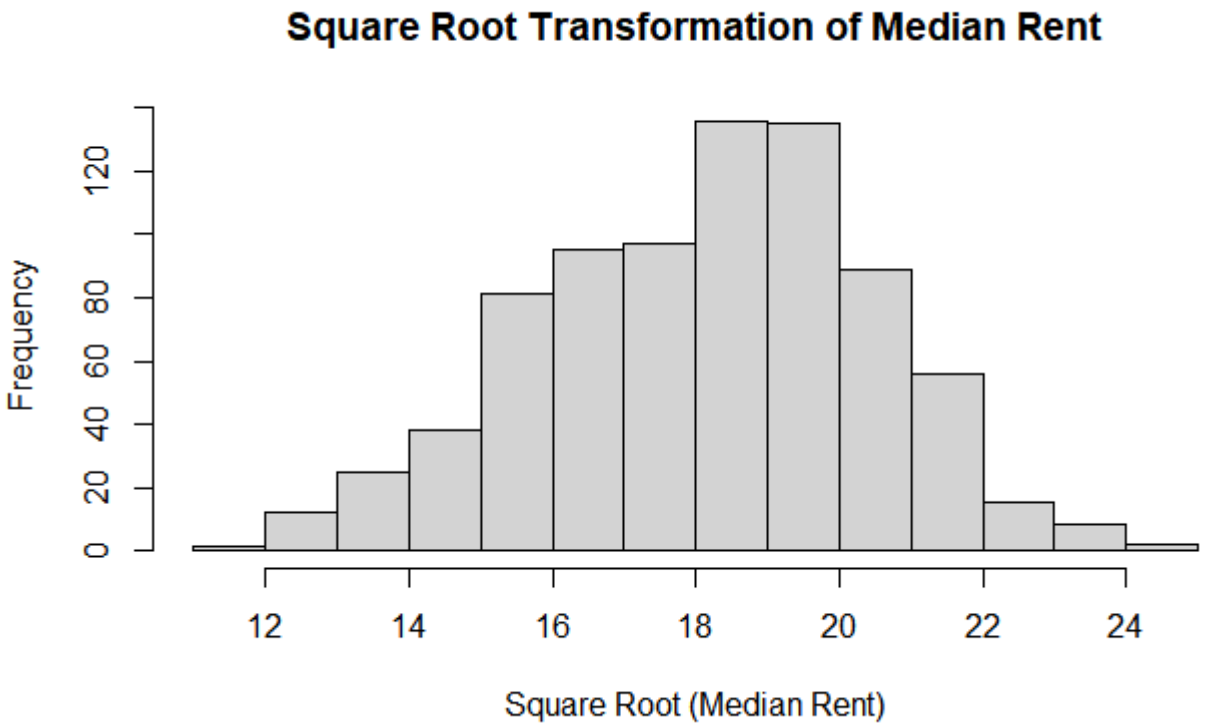
Hide

```
crimerent$`Median Rent`%>% log10() %>% hist(main="Log10 Transformation of Median Rent",xlab="Log 10 (Median Rent)")
```



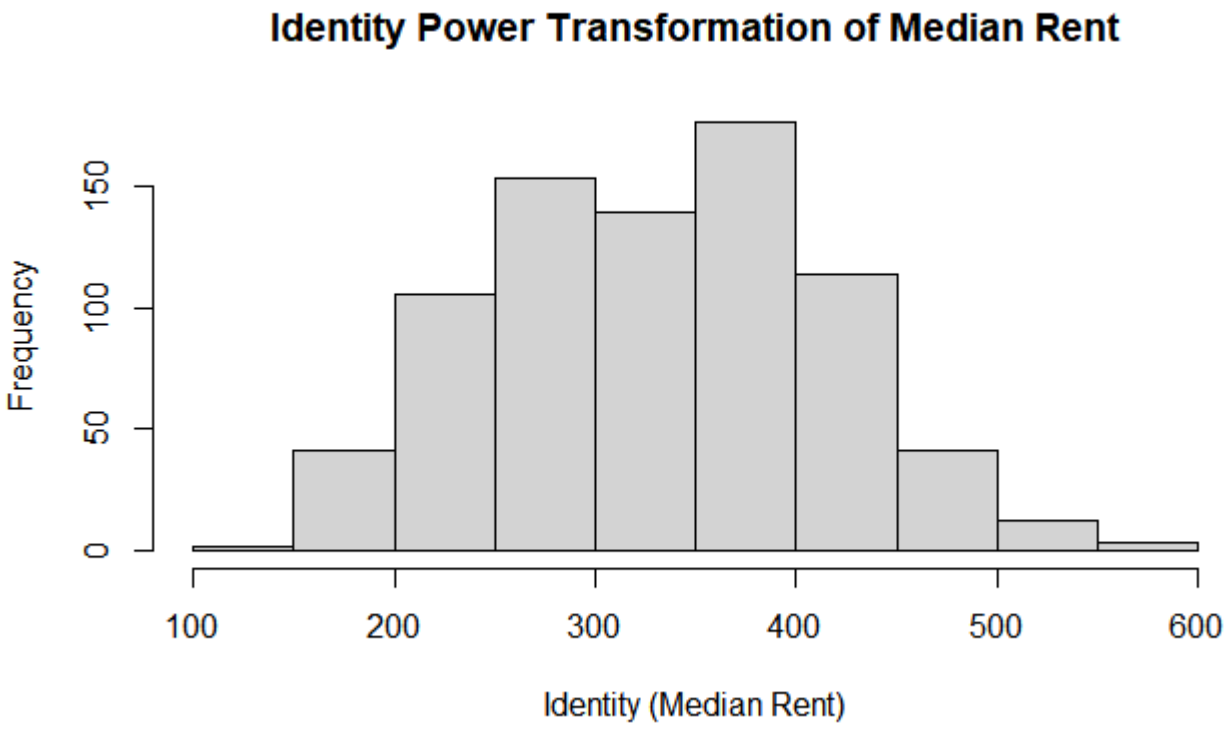
Hide

```
crimerent$`Median Rent` %>% sqrt() %>% hist(main="Square Root Transformation of Median Rent",xlab="Squa
re Root (Median Rent)")
```



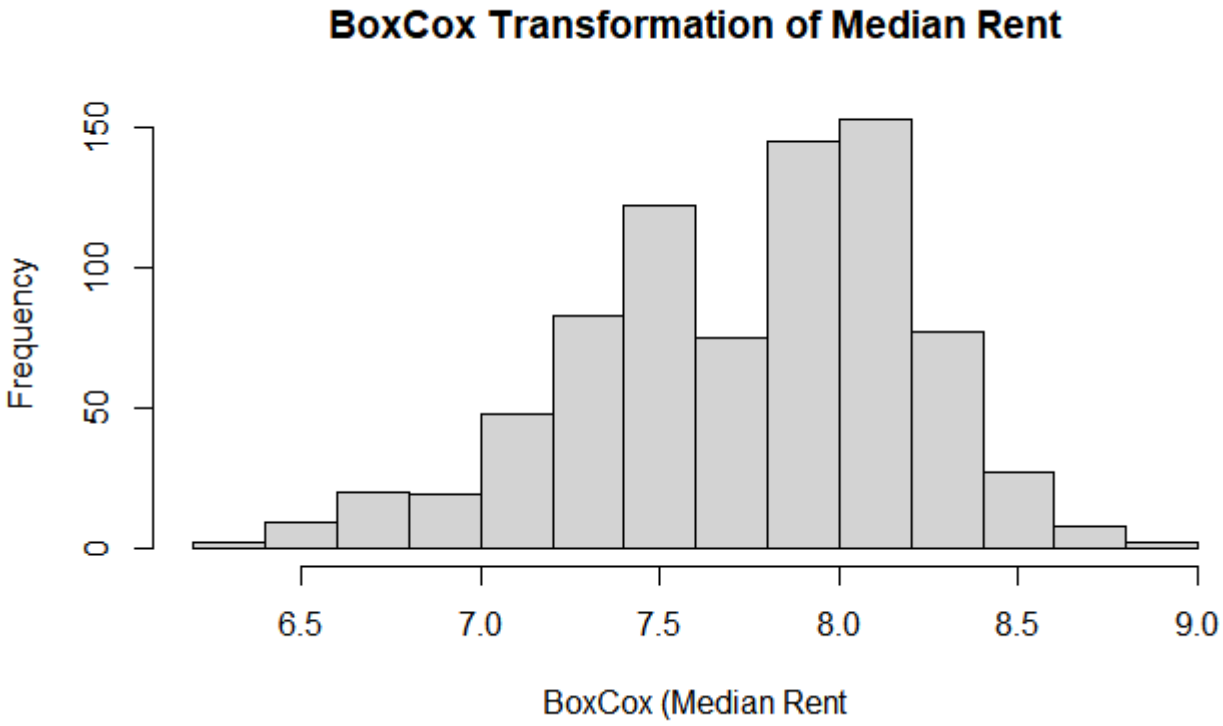
Hide

```
(crimerent$`Median Rent`)^1 %>% hist(xlab = "Identity (Median Rent)", main = "Identity Power Transforma
tion of Median Rent")
```



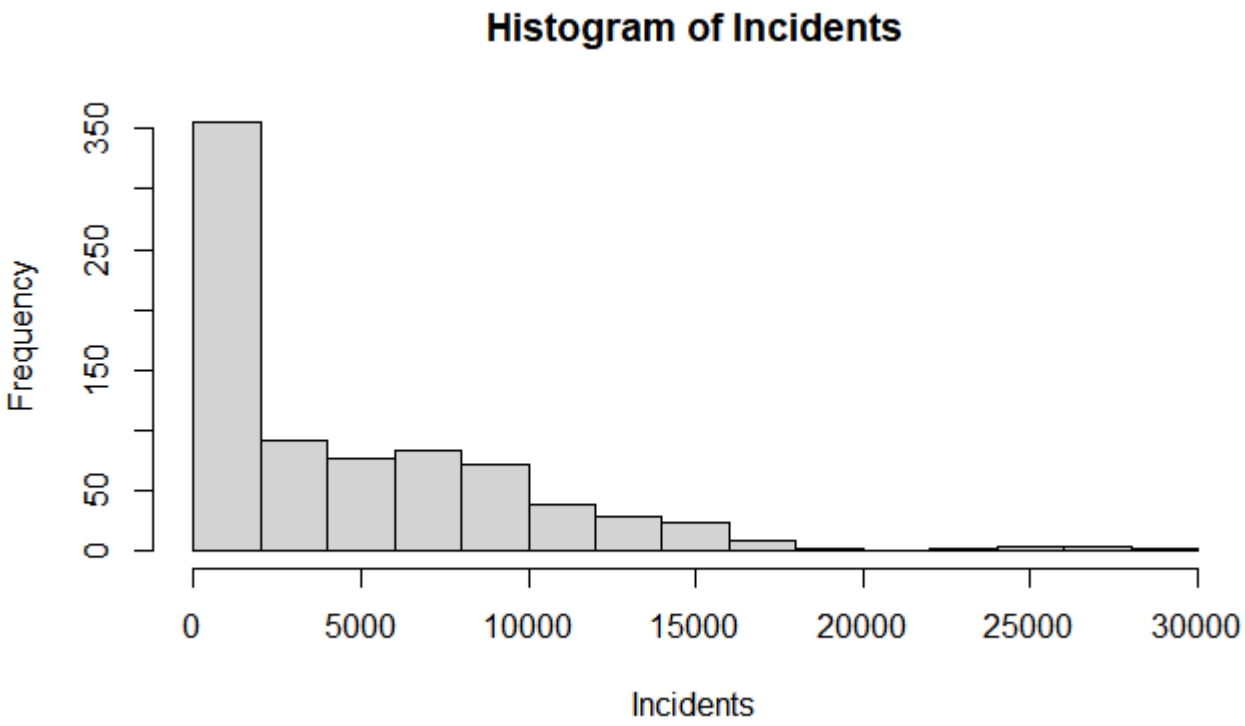
Hide

```
BoxCox(crimerent$`Median Rent`,lambda="auto") %>%hist(main="BoxCox Transformation of Median Rent",xlab
="BoxCox (Median Rent")
```



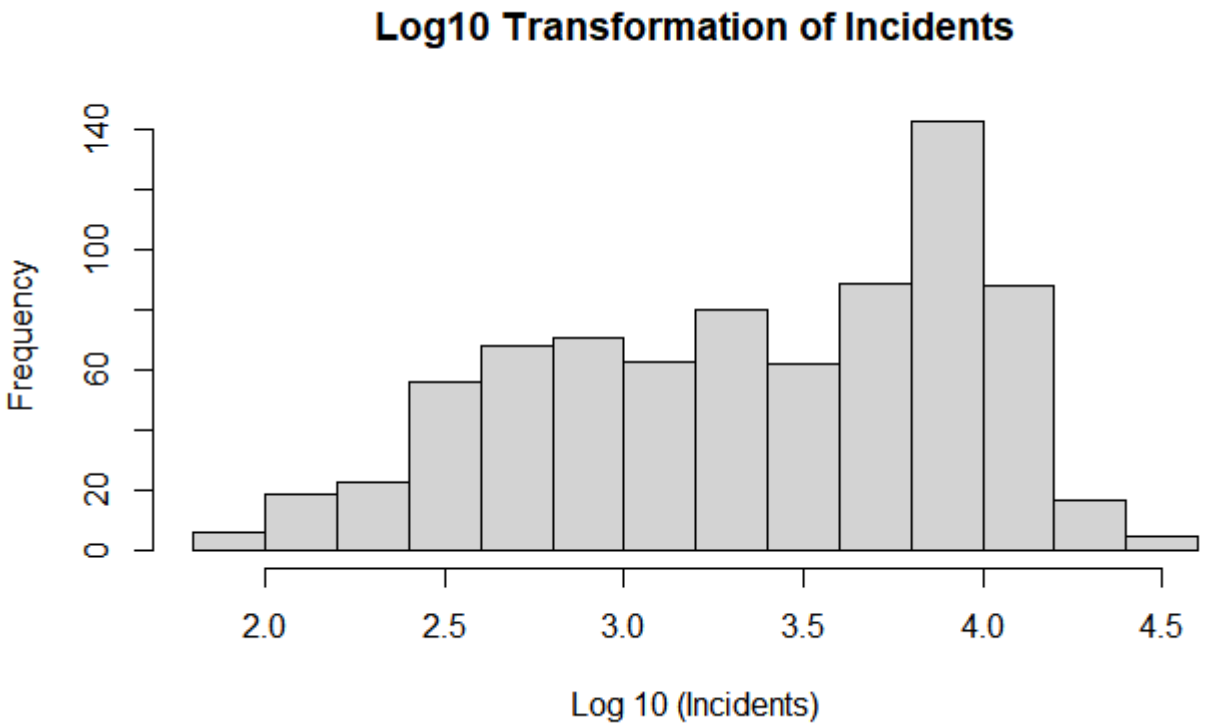
Hide

```
crimerent$Incidents %>% hist(main="Histogram of Incidents",xlab="Incidents")
```



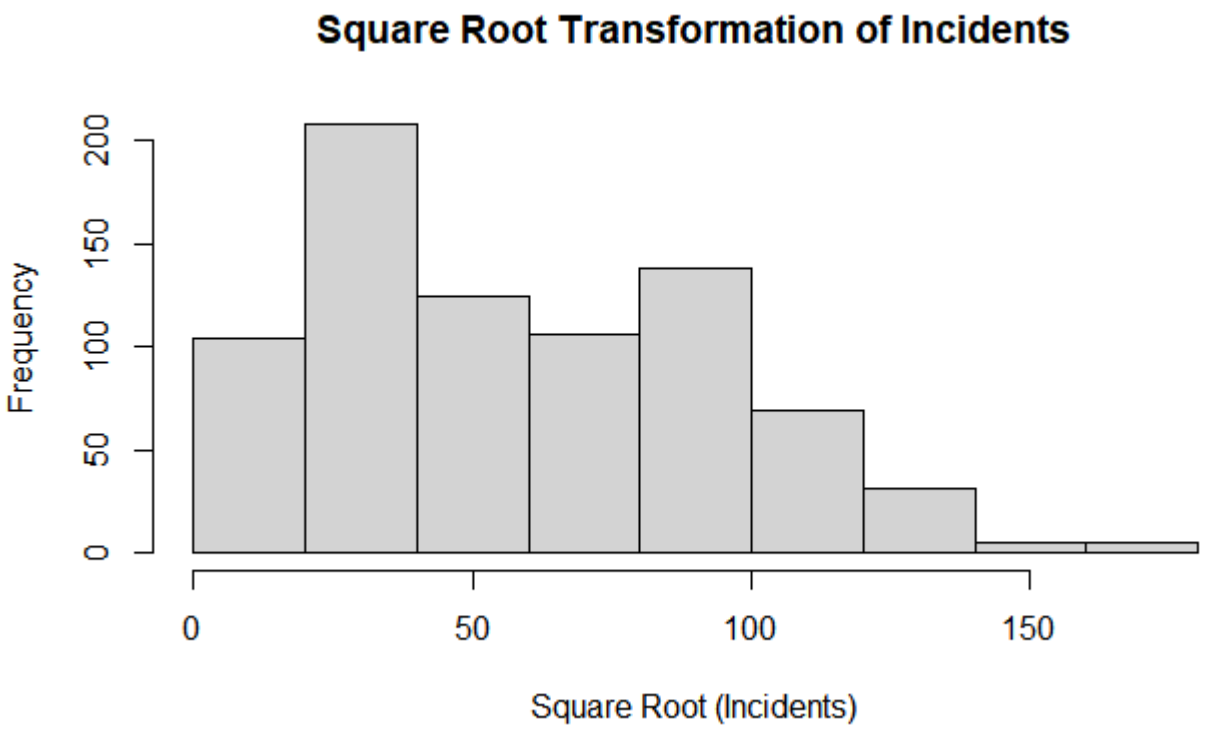
Hide

```
crimerent$Incidents %>% log10() %>% hist(main="Log10 Transformation of Incidents",xlab="Log 10 (Inciden
ts)")
```



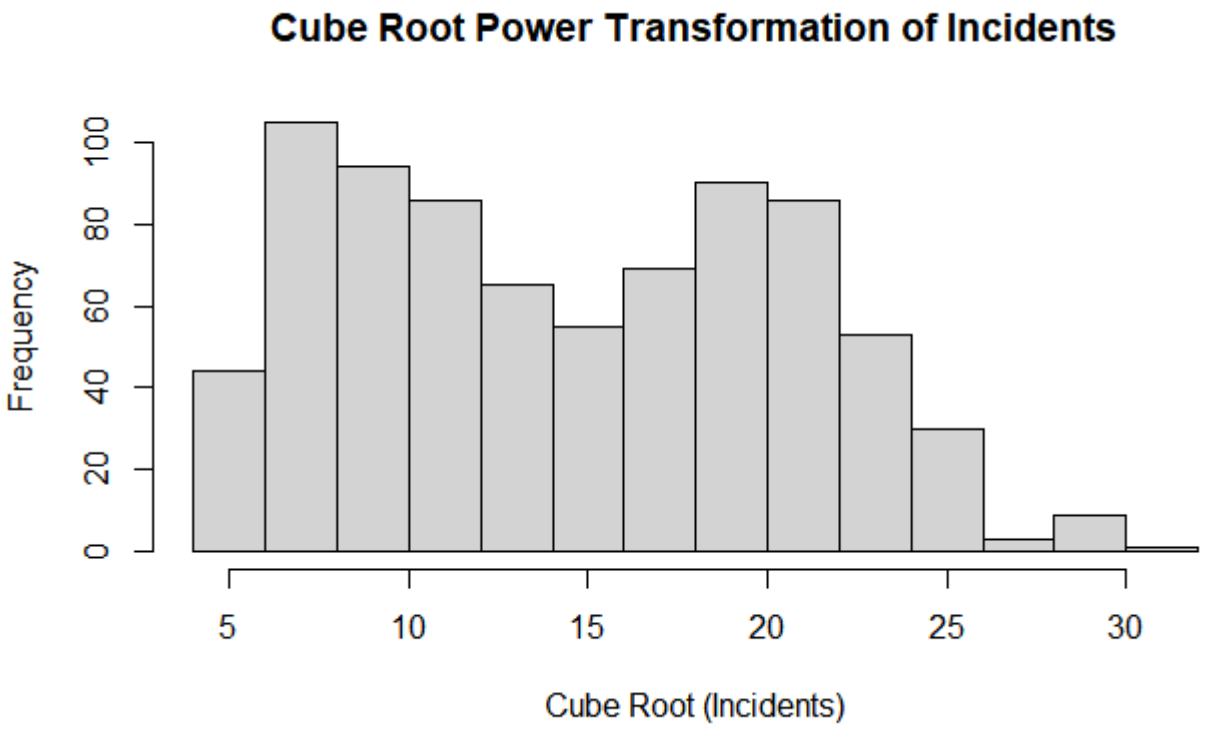
Hide

```
crimerent$Incidents %>% sqrt() %>% hist(main="Square Root Transformation of Incidents",xlab="Square Roo
t (Incidents)")
```



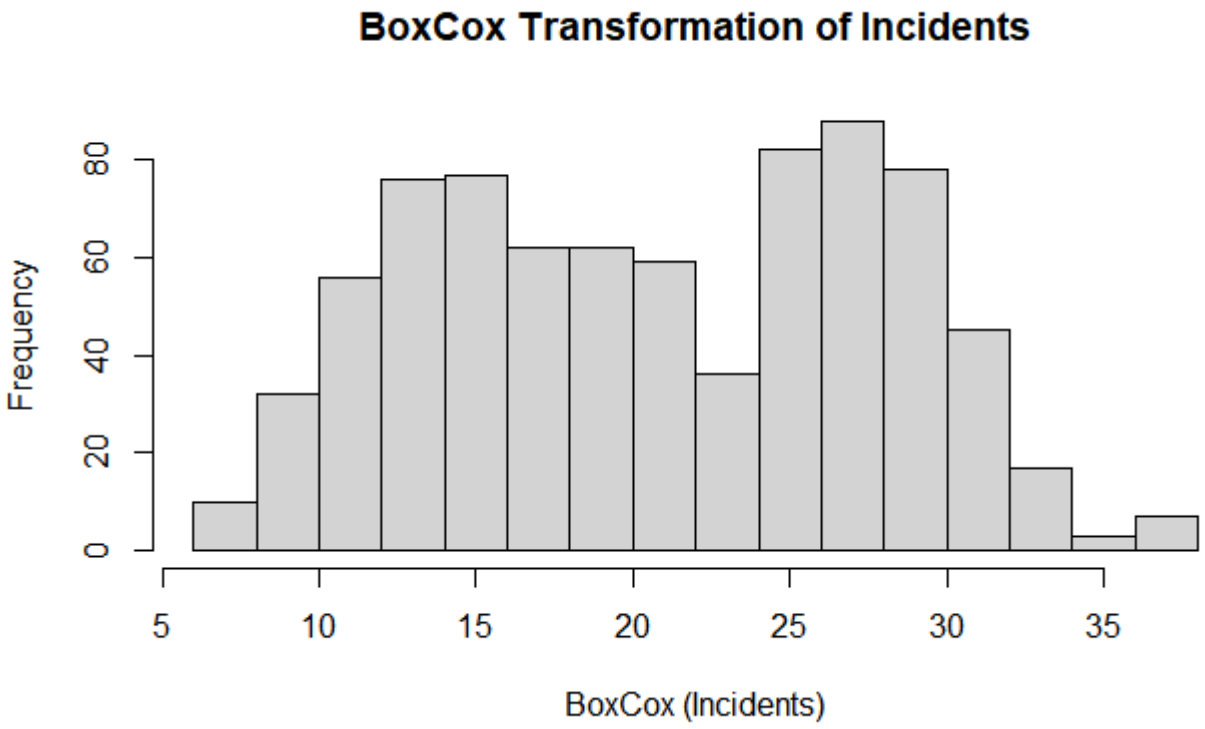
Hide

```
(crimerent$Incidents)^(1/3) %>% hist(main="Cube Root Power Transformation of Incidents",xlab="Cube Root
(Incidents)")
```



Hide

```
incidents<-BoxCox(crimerent$Incidents,lambda="auto") %>% hist(main="BoxCox Transformation of Incident
s",xlab="BoxCox (Incidents)")
```



- Firstly, we have created a histogram of Median Rent without any transformation as we can then compare it with other transformations. Applying the log10 transformation on median rent we can see that it is skewed to the left. Next, we can apply the square root transformation of median rent and we can see that it is slightly more centered. Then, by applying the identity power transformation, which is to the power of 1 we can see that it comforms more to a normally distributed graph. Lastly, by applying the BoxCox transformation it is not as centered than identity power distribution.
- Secondly, for Incidents, we have also created a histogram without any transformation to compare with other transformation. applying the log10 transformation on incidents we can see that it is skewed to the left. Next we apply the square root transformation on incidents and interestingly we can see it skewed to the right. Then, by applying the cube root transformation we can see that it is slightly more centered but not normally distributed. Lastly, applying the BoxCox transformation we can see that it is more centered.
- To sum up this section, after different transformations, the best transformation for median rent would be using the identity power transformation or by using no transformation at all as it is the most suitable choice among the rest. For Incidents, we can say that the best transformation is by using the BoxCox transformation as the distribution is the most normal out of all the other transformations.

Reference List

CSA (n.d.) *Glossary and Data dictionary*, Crime Statistics Agency website accessed 21 May 2023. https://www.crimestatistics.vic.gov.au/about-the-data/glossary-and-data-dictionary (https://www.crimestatistics.vic.gov.au/about-the-data/glossary-and-data-dictionary)

DataVic (2023) *Crime Statistics Agency Data Tables - Criminal Incidents* [data set], DataVic website, accessed 20 May 2023. https://discover.data.vic.gov.au/dataset/criminal-incident (https://discover.data.vic.gov.au/dataset/criminal-incident)

DFFH (2023) *Rental report* [data set], DFFH website, accessed 20 May 2023. https://www.dffh.vic.gov.au/publications/rental-report (https://www.dffh.vic.gov.au/publications/rental-report)

VEC (n.d.) *Local councils*, VEC website, accessed 21 May 2023. https://www.vec.vic.gov.au/electoral-boundaries/local-councils#:~:text=There%20are%2079%20local%20councils,local%20council%20for%20their%20area (https://www.vec.vic.gov.au/electoral-boundaries/local-councils#:~:text=There%20are%2079%20local%20councils,local%20council%20for%20their%20area)