Town Recommendation System and Report on an Individual Data Science Project

Sandesh Sapkota

Softwarica College of IT and E-Commerce, Coventry University

ST5014CEM Data Science for Developers

Siddhartha Neupane

August 2024

Table of Contents

Table of Figures

Introduction

This assignment shows the process and outcomes of using data analysis to recommend the appropriate city in the United Kingdom to purchase a property in the United Kingdom and particularly in Cornwall or Bristol counties. Each city has their own advantages but buying a house requires detailed analysis of several factors influencing property valuation and quality of life. The primary factors that are usually considered while buying a property include the housing price, crime rate in the neighborhood, internet connectivity, education and so on. This project utilizes the data made available by the government of United Kingdom and public sources to facilitate buyers to make informed decisions.

The objective of this project is to develop a recommendation system that uses data to analyze and compare the cities of Bristol and Cornwall based on several factors. The system provides a score to each city of the two counties based on the given criteria and rank the towns accordingly. The final goal is to suggest the top three cities to purchase a house to the friend in the context. This report outlines the process of obtaining the data, cleaning and preprocessing, performing exploratory data analysis and linear modeling to develop the recommendation system.

Cleaning Data

Data cleaning is an important step in the data science lifecycle which comes right after obtaining the datasets. This step ensures that the datasets are accurate and consistent and prepares them appropriately to be used for analysis. The datasets obtained from the UK government and other public institutions also had several inconsistencies and inaccuracies. Each dataset were carefully cleaned and prepared to be used for the further analysis and development of the recommendation system.

Cleaning Housing Data

The housing price data from 2020 and 2023 were combined into one using the bind_rows() function in R. Then the data was filtered to include the housing data of Cornwall and Bristol only. A new column was added named “Year” by extracting the year from the existing column “Transaction_Date”. After that only the required columns were selected. Finally, the null values and redundant entries were removed using na.omit() and distinct() respectively. Now, the dataset was saved using the write_csv() function.

Cleaning Broadband Speed

The broadband speed dataset was loaded using read_csv() function selecting the relevant columns. The the columns were renamed to simpler names for easier operation. For example, “Median download speed (Mbit/s)” was renamed to “MedianDownSpeed”. The final columns were ‘Postcode’, ‘MedianDownSpeed’, ‘MedianUpSpeed’, ‘AvgUpSpeed’, ‘MaxUpSpeed’, ‘AvgDownSpeed’ and ‘MaxDownSpeed’. Then the null values were removed using the na.omit() function. After that, the broadband data was merged with housing dataset using inner_join() function in R. The two datasets were joined based on the common field “Postcode”. Finally, the redundant rows were removed using the distinct() function. In this way, the

broadband speed dataset was cleaned and processed for further analysis. The cleaned dataset was then saved as a CSV using the write_csv() function.

Cleaning Crime Data

The crime datasets from Bristol and Devon & Cornwall in the years 2020 to 2024 were loaded using the read_csv(). Then all the datasets were combined together using the rbind() functions and converted into a tibble. Since the crime dataset did not include postcodes but includes the LSOA codes, another dataset postcodes to LSOA was also loaded and cleaned to join with the crime dataset. Only the required columns were selected from the LSOA and crime data. From the combined crime dataset, month, LSOA code, crime type, and falls within columns were selected. The columns were renamed for simplicity in further processing. Similarly, from the LSOA to postcode dataset, only 'lsoa11cd', 'lsoa11nm', 'ladnm', 'pcds' were selected. The data frame was then filtered to include the counties Bristol and Cornwall only. The duplicate values for LSOA code were checked and removed from both crime and lsoa datasets. Finally, the selected crime dataset was merged with the lsoa dataset using left join by the column LSOA code common in both datasets. Then two new columns 'Year' and 'Month' were created using the mutate function from the original Month column by trimming from 1 to 4 and 6 to 7 respectively. Now, the population data was also merged using left join. Finally, distinct() and na.omit() functions were used to remove the redundant rows and null values to clean the final data. Now, the dataset was saved using the write_csv() function.

Cleaning School Data

The performance school datasets from the academic years 2021 to 2022 and 2022 to 2023 were loaded using the read_csv() function from the readR library. Then the datasets were filtered to only include the relevant columns 'SCHNAME', 'PCODE', 'ATT8SCR', 'TOWN'. The

columns ‘Year’ and ‘County’ were added to each dataset using the mutate() function. Then the data from both academic years were combined for each county using the rbind() function. The combined dataset was now cleaned by filtering out the non-numeric values ‘NE’ and ‘SUPP’ in the ATT8SCR column. The null values and redundant rows were removed using the na.omit() and distinct() functions respectively. The final dataset was then saved as comma separated values (CSV) using the write_csv() function.

Exploratory Data Analysis

Exploratory Data Analysis (EDA) is an approach in data analysis which uses graphical representation and data visualization to summarize the features of data. After cleaning and preprocessing the datasets, the trends were analyzed by visualizing the data. Several visualization techniques such as box plots, line graph, radar chart, bar chart and pie chart were used for better understanding of distribution, relationships and main features of the data.

Visualization of Housing Price

Bar Chart: Average House Price by Town in 2023

The data was filtered to only include the year 2023 then the average housing price is calculated using the summarize() method. Then the data is visualized in a bar chart using the ggplot2 library. The cities were plotted in the x-axis and the average price was plotted in the y-axis in the bar chart. The bar chart shows the average housing price in various towns in Bristol and Cornwall for the year 2023. Boscastle town is seen to have a significantly higher average housing price among the cities. Boscastle is followed by Port Isaac, Padstow, and Waderbridge as the most expensive cities to own a house. Camborne, Callington, Redruth are some of the least expensive cities to consider.

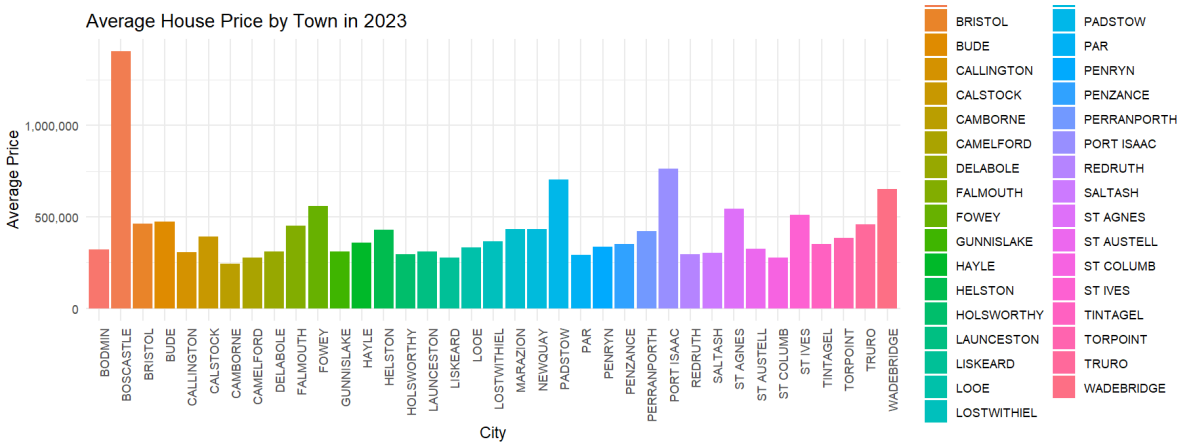


Figure 1: Barchart: Average House Price by Town

Bar Chart: Average House Price in 2023

In this chart, the average house prices in 2023 were visualized by county wise in a bar chart. The data was first filtered to include only from the year 2023 and then grouped by County using group_by() function. Then the average house price was summarized by calculating the mean of 'Price column'. The data was then visualized in a bar chart using ggplot2 to visualize the County to the x-axis and average house price to the y-axis. The chart shows that average house prices are noticeably higher in Bristol compared to Cornwall.

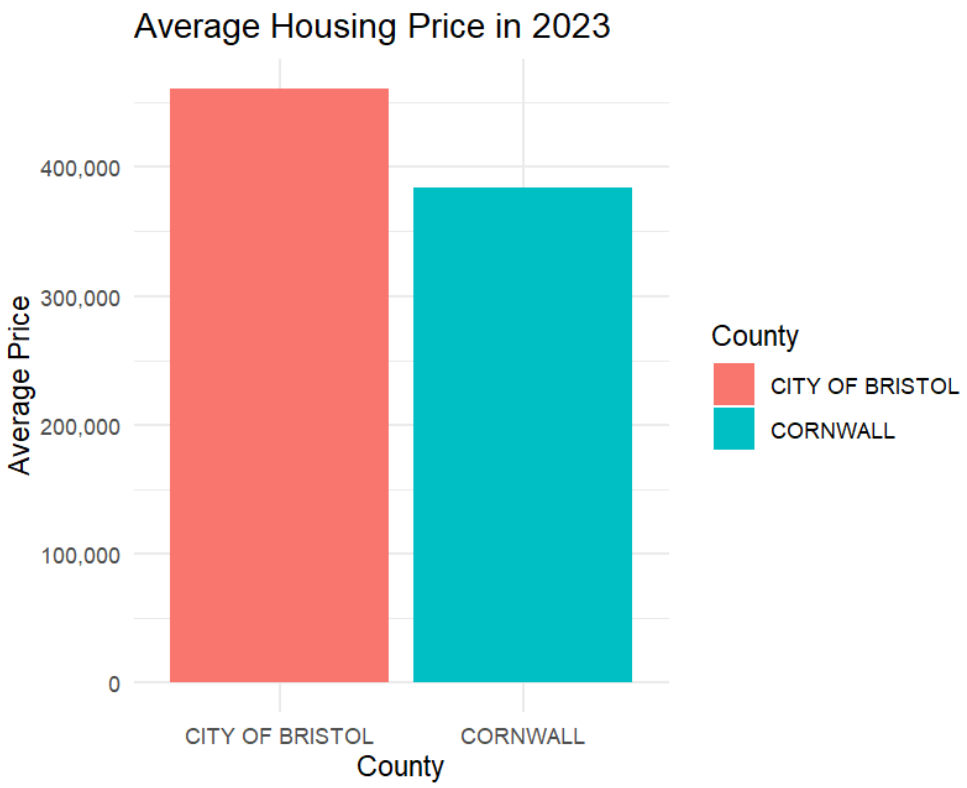


Figure 2: Bar Chart: Average House Price in 2023

Line Chart: Average House Prices from 2020 to 2023

The data was filtered so that the year lies between 2020 to 2023 and grouped by Year and County. Then the average price was calculated using summarise() method. Then the line chart was plotted using the ggplot2 library with year on the x-axis and average price price on the y-axis. Looking at the line chart, the city of Bristol has consistently had higher prices compared to Cornwall throughout the year. After 2022, the housing market in Cornwall has seen more stability while the gap between the two counties has widened more in recent years.

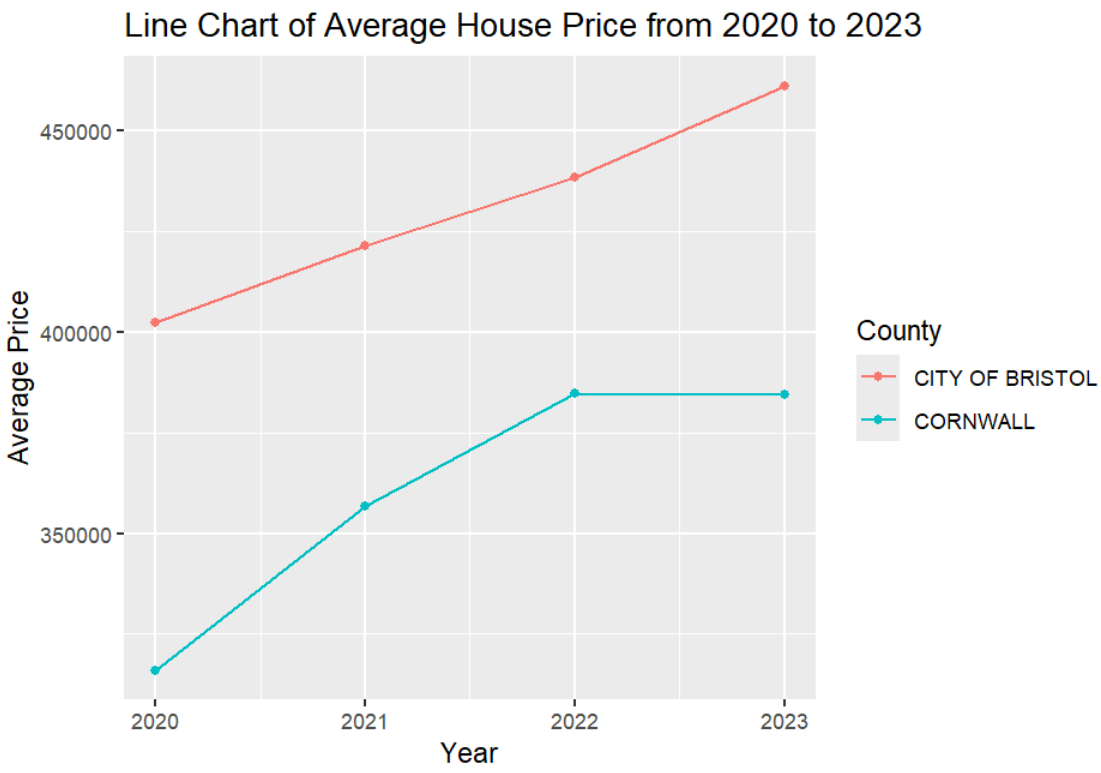


Figure 3: Line Chart of Average House Price

Visualization of Broadband Speed Data

Box plot: Average Download Speed by County

The cleaned broadband data is loaded from the csv file. Then the box plot is created using ggplot2 by grouping the average download speed and county. The box plot then visualizes the distribution of download speeds across the two counties. The box plot shows that Bristol has higher and more consistent download speeds with median around 80-90 Mbit/s while Cornwall has lower speed with median of 50 Mbits/s. The outliers denote the areas with significantly higher or lower speeds in both counties.

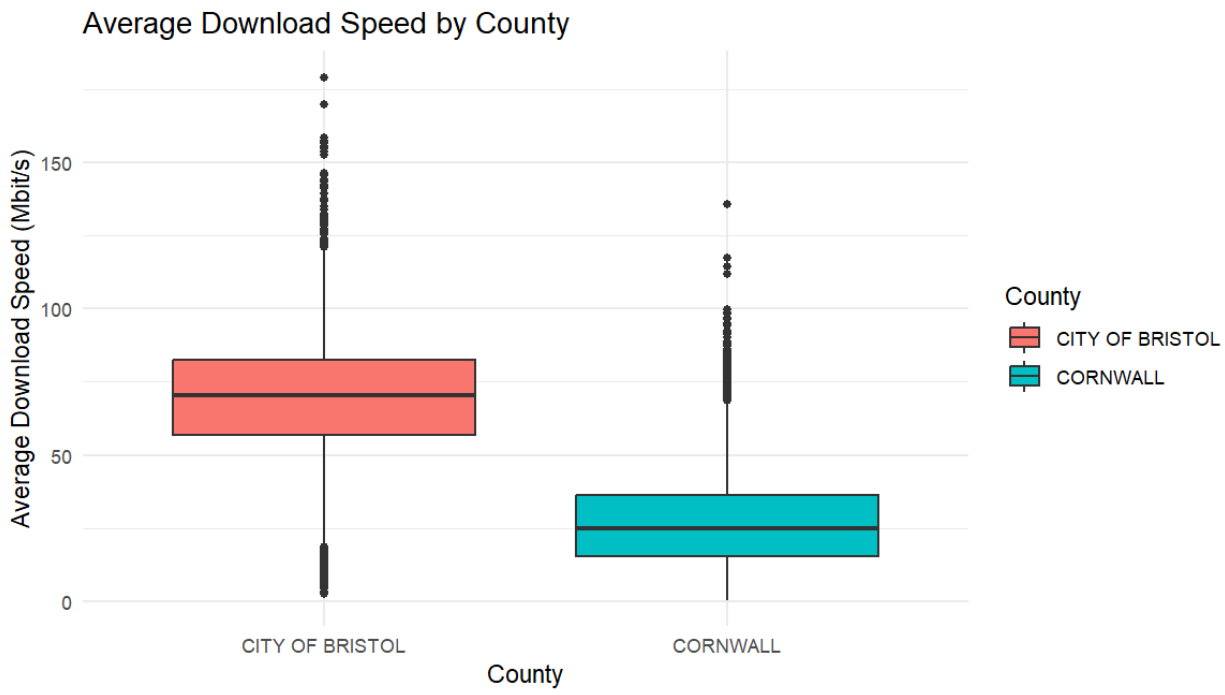


Figure 4: Boxplot: Average Download Speed by County

Bar Chart: Average and Maximum Download Speeds in Cornwall

The data was filtered to only include the county Cornwall and grouped by Town/City. Then the average and maximum download speeds were calculated using the summarize() method. The bar chart was plotted using ggplot2 which reveals significant differences in internet quality across Cornwall. The towns Saltash and St Austell have very high maximum speeds while the average among most towns is moderate.

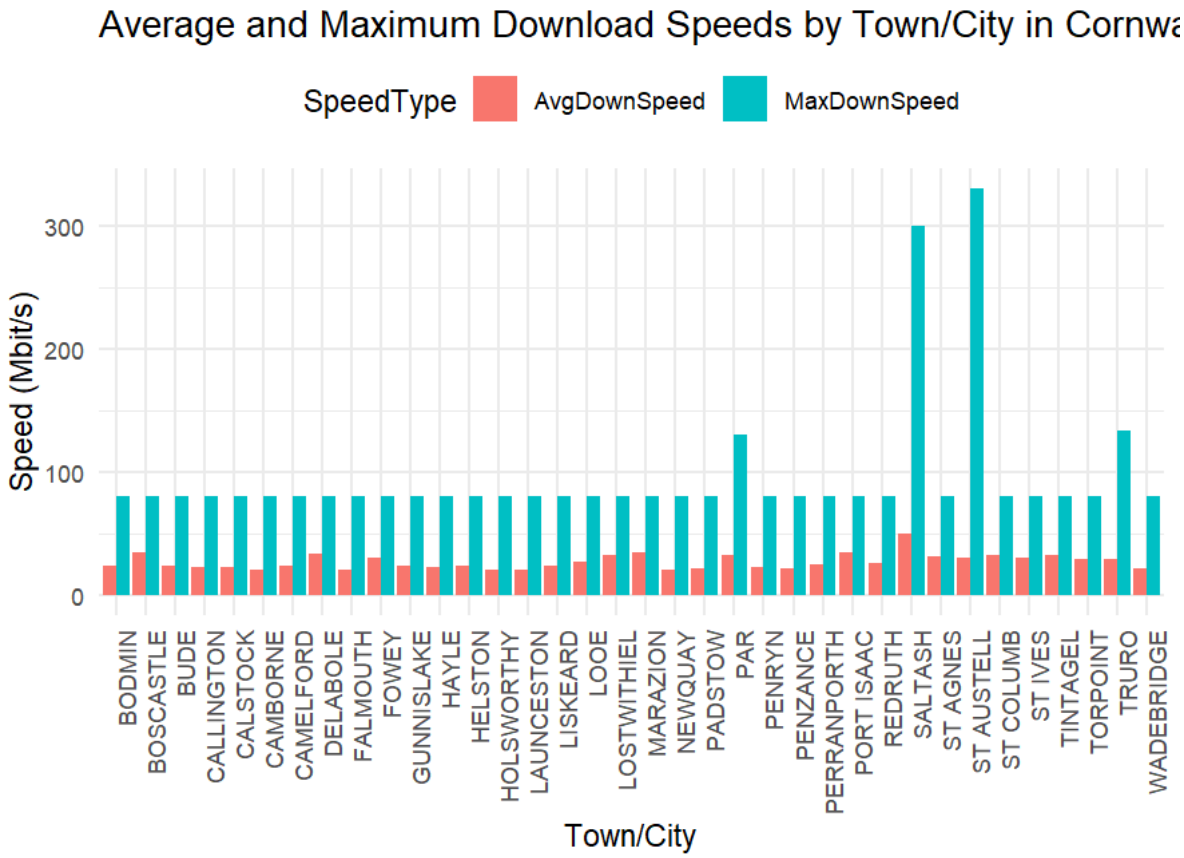


Figure 5: Barchart of Average and Max Download Speed in Cornwall

Bar Chart: Average and Maximum Download Speeds Bristol

The data was filtered to include the data of Bristol only and grouped by Town/City. Then the average and maximum speeds were calculated using summarize() function before visualizing the bar chart using ggplot2. From the bar chart, the city's internet service is not uniformly distributed as there is a big gap between the maximum possible speed and the average speed most users get in the City of Bristol.

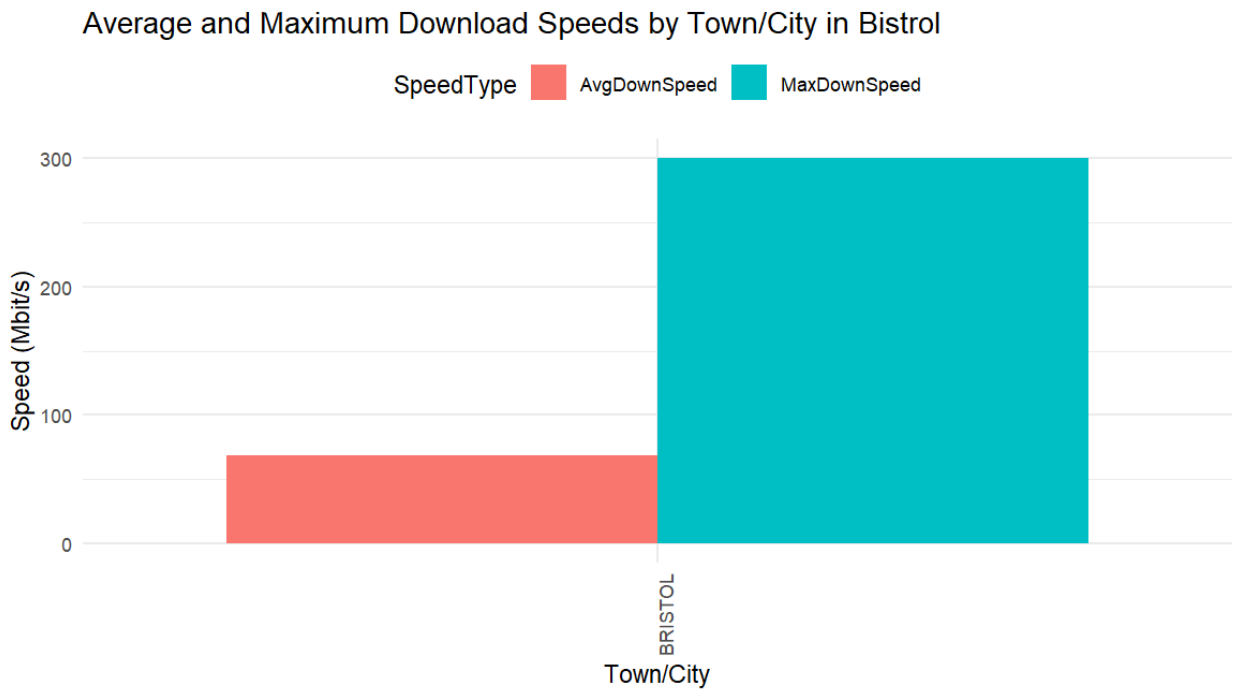


Figure 6: Barchart of Average and Max Downlaod Speed in Bristol

Visualization of Crime Data

Radar Chart: Vehicle Crime Rate per 10,000 people

The cleaned dataset is filtered to only include vehicle crimes and grouped by Year. Then the total crime is calculated by using the summarise() method. Then the data frame is prepared for the radar chart and plotted using the radarchart() function from 'fmsb' library in R. The chart shows the vehicle crimes were consistent from 2021 to 2024 with a few variations. The highest was in 2021 and the lowest was in 2023. Overall, the crime rates were stable across the years.

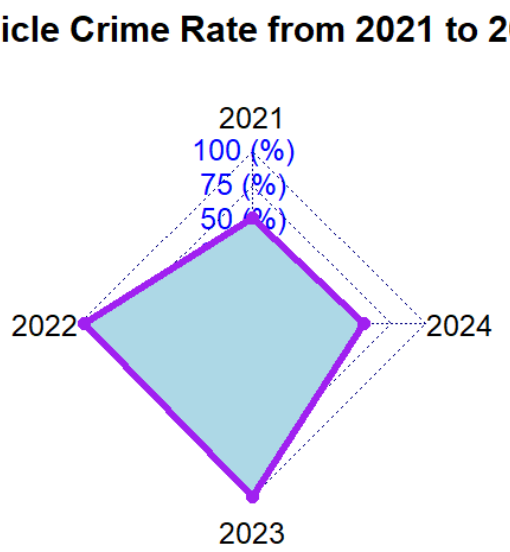


Figure 7: Radar Chart of Vehicle Crime Rate

Pie Chart: Robbery in 2023 by Month

The data was filtered and summarized for robbery rates in each month in 2023. The data is then visualized in a pie chart using ggplot2. The chart shows that there was highest percentage of robberies at 13.9% in January, while April and September had the lowest at 5.1%. Other than that, the rate of robberies is even throughout the year.

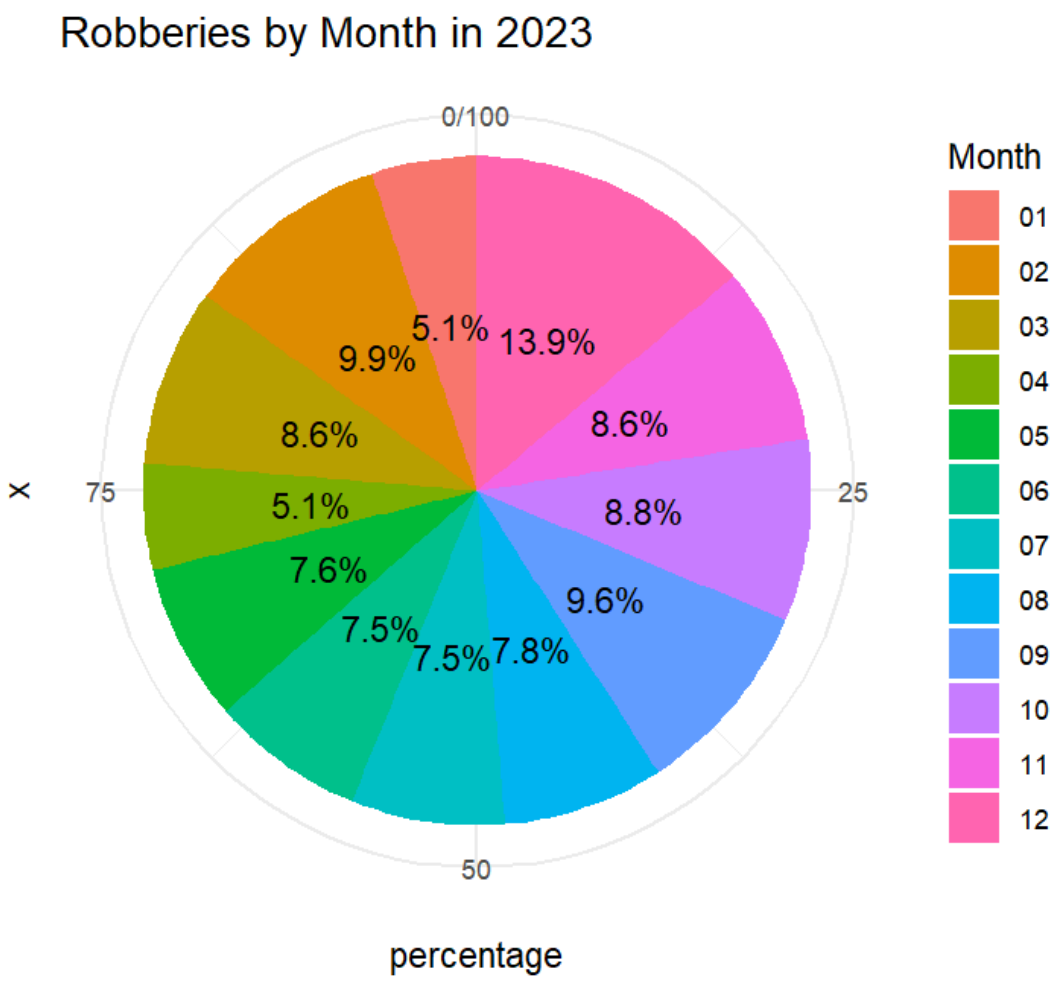


Figure 8: Pie Chart: Robberies by Month in 2023

Boxplot: Drug Offence Rate per 10,000 People in 2023

The boxplot visualizes the distribution of drug related crimes in 2023 per 10,000 people in Bristol and Cornwall. The boxplot shows that the drug rates are higher in Cornwall compared to Bristol. Similarly, Cornwall has a wider spread indicating differences in offences in different regions while Bristol shows uniform distribution of data.

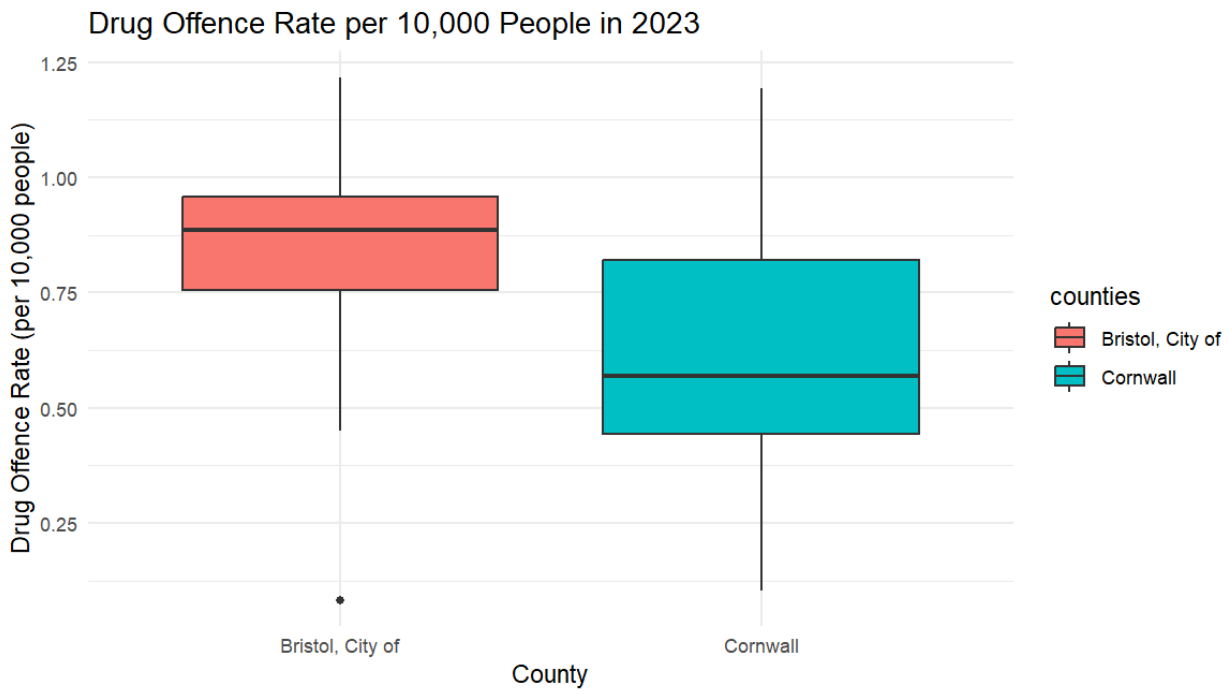


Figure 9: Boxplot : Drug Offence per 10000

Visualization of School Data

Line Graph: Average Attainment 8 scores for 2021-2022 in Cornwall

Attainment 8 scores is a way to measure the average performance of school students in the UK. The line line graph is created to show the average attainment 8 scores for different towns in Cornwall in academic year 2021-2022. Town is taken in x-axis and scores is taken in y-axis of the graph. It shows a significant variation across the towns. Towns like Cornwall, St Ives performed well while others like Truro have performed extremely poor. The others had an above average performance.

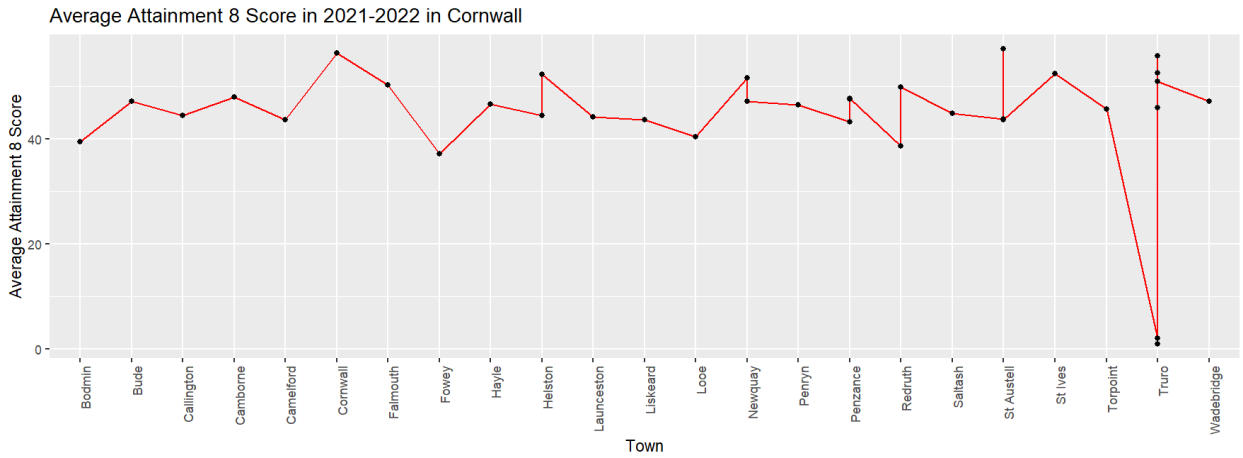


Figure 10: Linegraph of Attainment score in Cornwall

Line Graph: Average Attainment 8 scores for 2021-2022 in Bristol

For Bristol, the visualization for attainment 8 scores was done using ggplot2 based on Schools as there weren't significant data for the towns. In the line graph, x-axis represented Schools and y-axis contained their attainment scores. The graph shows schools like Montpelier High School peak the graph while schools like Venturers Academy show lower scores. This shows high academic disparities across the Bristol County.

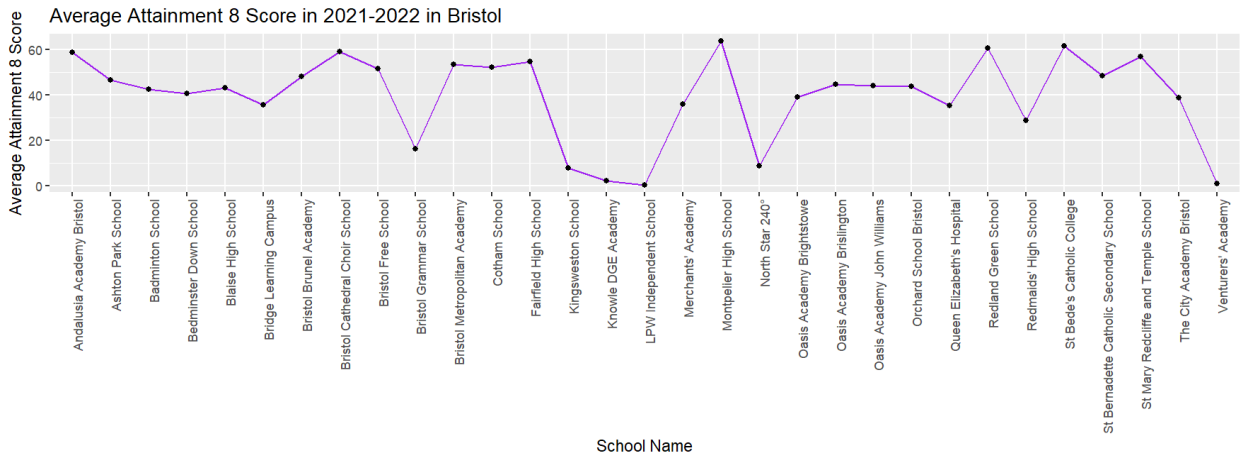


Figure 11: Linegraph of Attainment score in Bristol

Linear Modeling

Linear Modeling is a statistical method used in data analysis to analyze the the relationship between two or more variables. It shows the linear relationship between a dependent variable and and one or more independent variables expressed in the form of a linear equation. This approach was used to identify the trends, patterns and make predictions based on the data of Cornwall and Bristol.

Housing Price Vs. Average Download Speed

The shows a positive correlation between house price and average download speed. This suggests that higher the internet bandwidth, higher will be the housing price. This is likely because people with nicer house often want a good internet connection. However, there is a lot of outliers especially at the lower end of the house prices.

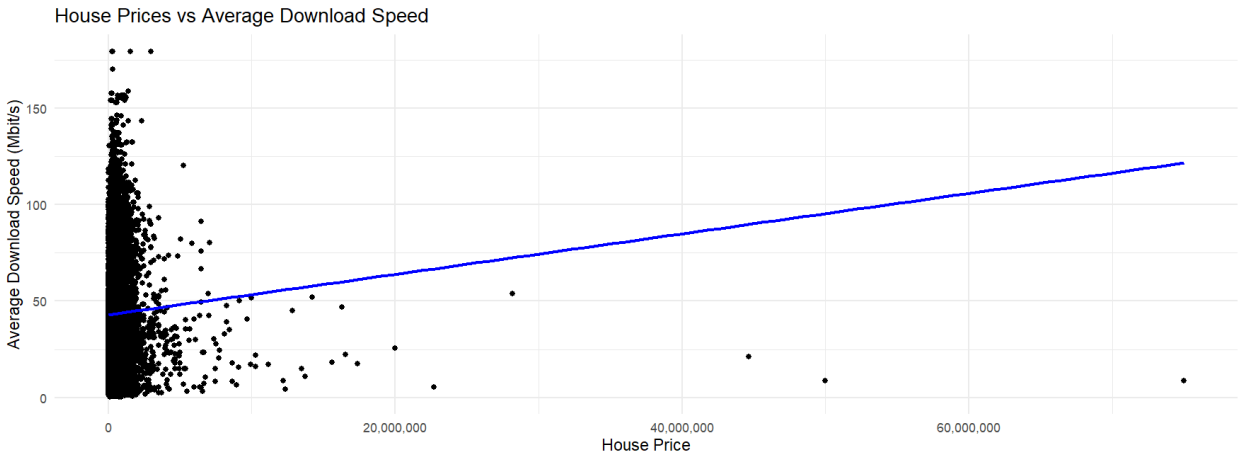


Figure 12: House Price Vs Average Download Speed

Average Attainment 8 Score Vs Housing Price

The scatterplot shows a weak positive relationship between the attainment 8 score and housing prices in the two counties. This shows that the areas with good educational performance

have higher housing price. The relationship is weak because the data points are widely scattered indicating that attainment 8 score might not have significant impact on the price.

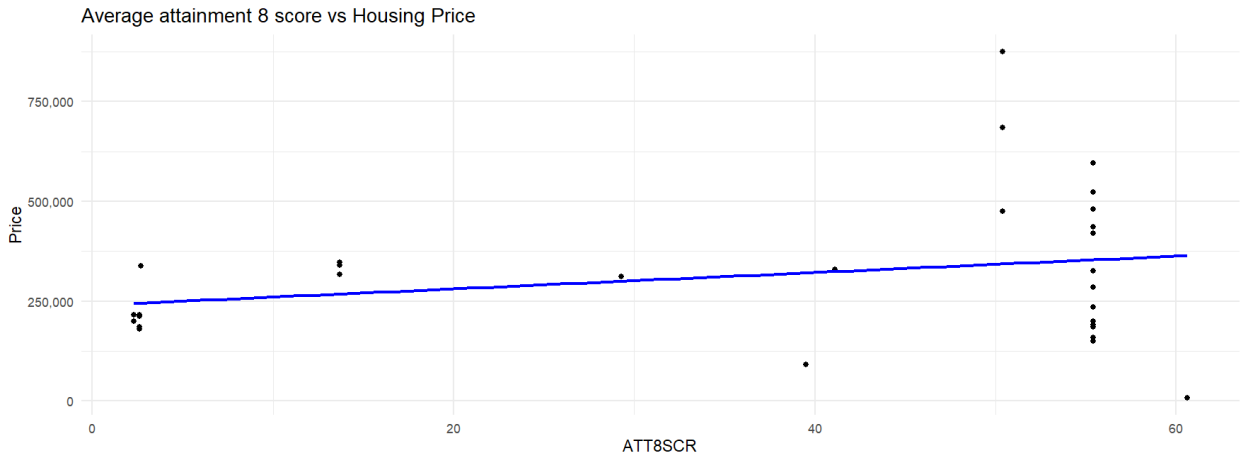


Figure 13: Average Attainment Vs House price

Housing Price Vs Drug Rates

The plot shows a negative correlation between the housing prices and drug crime rates in the area. This suggests that the areas with expensive houses have lower drug offence rate while cheaper neighborhoods have higher drug crime rates. This might be because social factors like this affect the real estate market.

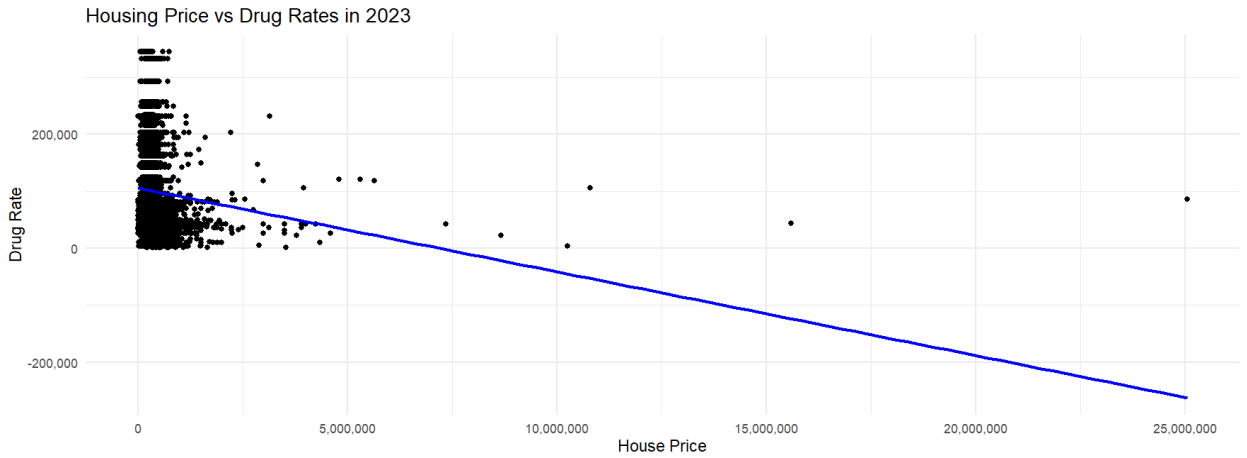


Figure 14: House Price vs Drug Rate

Average Download Speed vs Attainment 8 scores

A weak positive correlation is seen between the average download speed and attainment 8 scores of a region. Even though the scattered data points suggest the relationship was not too strong, it could be said that the better internet has a slight impact in the academic performance of schools.

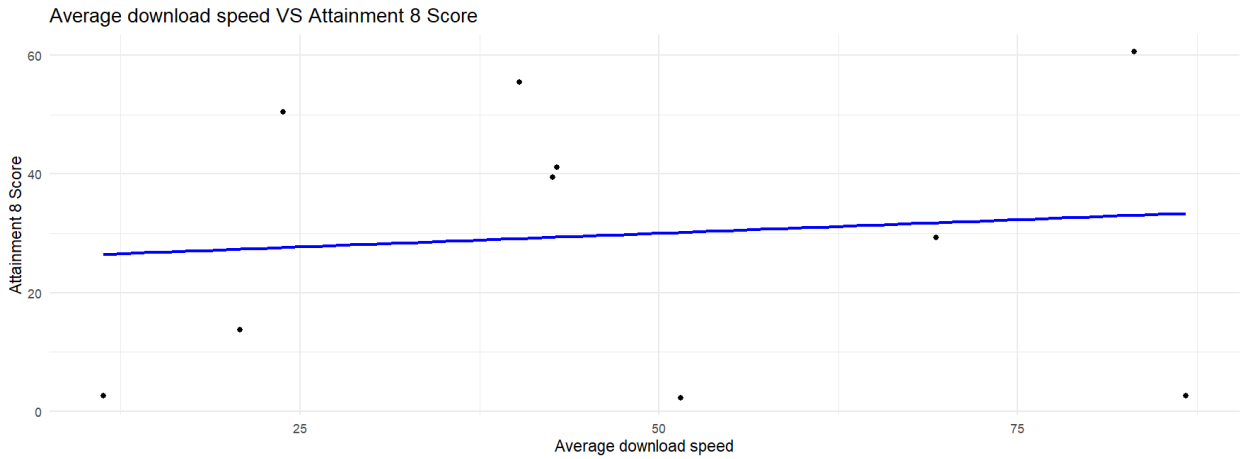


Figure 15: Average Downlaod Speed Vs Attainment 8

Town Recommendation System

Overview

After cleaning the obtained datasets, the data were prepared and visualized for exploratory data analysis, prepared linear models to compare the relationships between the different parameters in deciding the house purchase. Now, the task was to identify the most suitable town to purchase the house considering affordable housing, low crime rates, good internet connectivity, and good school performance measured by the attainment 8 scores. The preprocessed datasets were loaded and filtered and processed to calculate average for each factors by town. The data was then merged into a single data frame and normalized to ensure they can be compared later on. Finally, the towns are ranked based on the scores. Higher scores indicate more favorable to purchase a house or property in that location.

Results

Based on Housing Price Ranking

	TOWN	avgPrice	normHousingPrice
1	CAMBORNE	244939.2	1.0000000
2	LISKEARD	276915.0	0.9214842
3	CAMELFORD	277289.9	0.9205635
4	REDRUTH	295858.8	0.8749681
5	SALTASH	301707.2	0.8606076
6	CALLINGTON	305634.0	0.8509653
7	LAUNCESTON	310922.9	0.8379786
8	BODMIN	323267.0	0.8076681
9	ST AUSTELL	325660.4	0.8017912
10	LOOE	333198.0	0.7832828

Figure 16: Top 10 towns based on House Price

Based on Crime Rates Ranking

	TOWN	normCrimeRate	crimerate
1	FOWEY	1.0000000	98
2	CAMELFORD	0.9913297	199
3	LOOE	0.9800841	330
4	CALLINGTON	0.9683235	467
5	WADEBRIDGE	0.9663490	490
6	TORPOINT	0.9634303	524
7	PENRYN	0.9592240	573
8	ST IVES	0.9545884	627
9	HAYLE	0.9478067	706
10	BUDE	0.9420551	773

Figure 17: Top 10 Towns based on Crime Rates

Based on School Performance

	TOWN	avgAtt8	normAtt8
1	ST IVES	49.90000	1.000000000
2	PENRYN	47.75000	0.816112567
3	HELSTON	47.47500	0.792592082
4	ST AUSTELL	47.26667	0.774773532
5	WADEBRIDGE	46.60000	0.717754173
6	BUDE	46.45000	0.704924817
7	FALMOUTH	46.45000	0.704924817
8	CAMBORNE	46.25000	0.687819010
9	LAUNCESTON	44.65000	0.550972548
10	PENZANCE	44.31667	0.522462868

Figure 18:Top 10 based on Schools

Based on Broadband

	TOWN	avg_down_speed	normDownSpeed
1	BRISTOL	68.48118	1.000000000
2	SALTASH	49.86904	0.609798505
3	FOWEY	30.87476	0.211585278
4	ST IVES	30.63361	0.206529673
5	ST AUSTELL	30.58067	0.205419759
6	TORPOINT	29.01877	0.172674778
7	LOOE	27.68960	0.144808782
8	REDRUTH	26.06246	0.110696055
9	LISKEARD	24.47380	0.077389865
10	HELSTON	24.44340	0.076752519

Figure 19 Top 10 based on Broadband

Overall Ranking

	TOWN	avgPrice	crimerate	avgAtt8	avg_down_speed	finalPoints
1	SALTASH	301707.2	836	42.65000	49.86904	3.786967
2	ST AUSTELL	325660.4	2744	47.26667	30.58067	3.531017
3	ST IVES	510935.4	627	49.90000	30.63361	3.463236
4	TORPOINT	385197.4	524	43.60000	29.01877	3.093698
5	LOOE	333198.0	330	40.80000	27.68960	2.843493
6	HELSTON	431010.9	1361	47.47500	24.44340	2.830370
7	REDRUTH	295858.8	2542	42.87500	26.06246	2.783760
8	PENRYN	335319.3	573	47.75000	23.28255	2.728404
9	BUDE	475156.9	773	46.45000	24.13062	2.684772
10	LISKEARD	276915.0	1110	42.25000	24.47380	2.675947

Figure 20 Overall Top 10

Ergo, after analyzing the towns on different parameters, the towns SALTASH, ST AUSTELL and ST IVES can be recommended for purchasing a house since these cities provide the most balanced parameters optimal housing price, less crime rates, good internet and good education.

Reflection

The relevant datasets were obtained from reliable sources of the official websites UK government and other public organizations to maintain the credibility of the data. Then the analysis was done using the R programming language to clean, process, visualize, construct linear models and finally rank the towns based on different parameters. The data cleaning included filtering required columns, removing null values and redundant fields. Then the data were visualized using different visualization methods such as box plots, bar charts, line graphs

and pie charts to find the patterns and trends. Similarly, linear modeling was done to analyse and compare the relationships between different parameters. This helped to classify the counties based on their performance in various factors. Finally, all the parameters were analyzed to identify the most favorable characteristics and recommendations were made on the basis of different comparative results.

Legal and Ethical Considerations

The project has been done by considering all the ethical and legal compliances practiced in the industry. The datasets were obtained from the official websites of the UK government that are publicly available and approved by the government. The data weren't manipulated to push a certain political ideology and are completely based on the UK government approved sources.

Conclusion

This project demonstrated the use of data analysis to develop a recommendation system to figure out the optimal town in the Bristol and Cornwall counties in the United Kingdom to purchase a house. By cleaning and preprocessing the datasets obtained from credible sources, the parameters were compared to identify Saltash, St Austell and St Ives as the top three most favorable cities to buy a house considering affordable pricing, low crime rates, good internet and strong academic performance of local schools. It was a great learning experience to explore the capacities of data analysis in recommendation systems.

References

Sisense. (2018, June 12). What is Data Cleaning? | Sisense. https://www.sisense.com/glossary/data-cleaning/

Stedman, C. (2022, January 28). data cleansing (data cleaning, data scrubbing). Data Management. https://www.techtarget.com/searchdatamanagement/definition/data-scrubbing

Data downloads | data.police.uk. (n.d.). https://data.police.uk/data/

Connected Nations 2018: Data downloads. (2019, January 2). www.ofcom.org.uk. https://www.ofcom.org.uk/phones-and-broadband/coverage-and-speeds/data-downloads

RPUBS - Introduction to Linear Modeling in R. (n.d.). https://rpubs.com/AnthonyCorbisieri/1100671

Radar. (n.d.). https://plotly.com/r/radar-chart/

Chatterjee, S. (2024, February 26). *5 Ethical aspects for data science professionals to consider*. Emeritus Online Courses. https://emeritus.org/blog/data-science-and-analytics-data-science-course-curriculum/

Appendix

Github Link: https://github.com/sandeshs0/data-science-assignment

Google Drive Link:
https://drive.google.com/drive/folders/11tJCS5Z5XRncfJ_cnh9a0_XBJJ31dKoX?usp=sharing

Cleaning.R

```r
# libraries
library(dplyr)
library(readr)
library(tidyverse)
# Defining column names
column_names <- c("Transaction_ID", "Price", "Transaction_Date", "Postcode",
                  "Property_Type", "Old_New", "Duration", "PAON", "SAON",
                  "Street", "Locality", "Town_City", "District", "County",
                  "PPD_Category_Type", "Record_Status")
# Loading the datasets with column names
housing_2020 <- read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Housing/pp-2020.csv", col_names = column_na
housing_2021 <- read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Housing/pp-2021.csv", col_names = column_na
housing_2022 <- read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Housing/pp-2022.csv", col_names = column_na
housing_2023 <- read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Housing/pp-2023.csv", col_names = column_na
# Combining the datasets
combined_housing = bind_rows(housing_2020, housing_2021, housing_2022, housing_2023)
#Data Cleaning
cleaned_housing <- combined_housing %>%
  filter(County %in% c('CORNWALL', 'CITY OF BRISTOL')) %>%
  mutate(Year = year(Transaction_Date)) %>%
  select(Price, Postcode,Year,Town_City,County) %>%
  na.omit() %>%
  distinct()
write_csv(cleaned_housing, 'D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/housing_cleaned.csv')
```

```r
#----------------CLEANING BROADBAND SPEED DATASET--------
broadband <- read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Broadband Speed/201805_fixed_pc_performance_r0
#Selecting Relevant Columns
broadband_selected <- broadband %>%
  select("postcode_space",
         "Median download speed (Mbit/s)",
         "Median upload speed (Mbit/s)",
         "Average upload speed (Mbit/s)",
         "Maximum upload speed (Mbit/s)",
         "Average download speed (Mbit/s)",
         "Maximum download speed (Mbit/s)",
  )
#Checking Null Values
na_summary <- sapply(broadband_selected, function(x) sum(is.na(x)))
print(na_summary)
#Cleaning
broadband_clean <-broadband_selected %>%
  rename(
    Postcode = postcode_space,
    MedianDownSpeed = `Median download speed (Mbit/s)`,
    MedianUpSpeed = `Median upload speed (Mbit/s)`,
    AvgUpSpeed = `Average upload speed (Mbit/s)`,
    MaxUpSpeed = `Maximum upload speed (Mbit/s)`,
    AvgDownSpeed = `Average download speed (Mbit/s)`,
    MaxDownSpeed = `Maximum download speed (Mbit/s)`
  ) %>%
  na.omit() %>%
  distinct()
```

```r
  distinct()
#Checking for null values after Cleaning
na_summary_clean <- sapply(broadband_clean, function(x) sum(is.na(x)))
print(na_summary_clean)
#Selecting only 3 rows from housing
housing_selected <- cleaned_housing %>%
  select(Postcode,Town_City,County)

#Using Inner Join to Merge the broadband with housind data
broadband_W_housing = inner_join(housing_selected,broadband_clean,by="Postcode");
View(broadband_W_housing)

na_merged <- sapply(broadband_W_housing, function(x) sum(is.na(x)))
print(na_merged)
dim(broadband_W_housing)


#Removing redundent rows
broadband_final <- broadband_W_housing %>%
  distinct()
dim(broadband_final)
View(broadband_final)

#Saving the dataset
write_csv(broadband_final, "D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/broadband_cleaned.csv")
```

```r
bist31=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Crime/2024-01/2024-01-avon-and-somerset-street.csv")
bist32=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Crime/2024-02/2024-02-avon-and-somerset-street.csv")
bist33=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Crime/2024-03/2024-03-avon-and-somerset-street.csv")
bist34=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Crime/2024-04/2024-04-avon-and-somerset-street.csv")
bist35=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Crime/2024-05/2024-05-avon-and-somerset-street.csv")
bist36=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Crime/2024-06/2024-06-avon-and-somerset-street.csv")

#Combining the Datasets
crime_combined=rbind(
  bist1,bist2,bist3,bist4,bist5,bist6,bist7,bist8,bist9,bist10,bist11,bist12,bist13,bist14,bist15,bist16,bist17,bist18,bist19,bist20,bist21,
  bist22,bist23,bist24,bist25,bist26,bist27,bist28,bist29,bist30,bist31,bist32,bist33,bist34,bist35,bist36,

  corn1,corn2,corn3,corn4,corn5,corn6,corn7,corn8,corn9,corn10,corn11,corn12,corn13,corn14,corn15,corn16,corn17,corn18,corn19,corn20,corn21,
  corn23,corn24,corn25,corn26,corn27,corn28,corn29,corn30,corn31,corn32,corn33,corn34,corn35,corn36
)

head(crime_combined)
View(crime_combined)
dim(crime_combined)

#Converting it to a tibble
crime_combined <- crime_combined %>%
  as_tibble()


#POST CODE TO LSOA

postcode_to_lsoa <- read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Postcode to LSOA.csv")
population <- read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Population.csv")
#Selecting relevant columns
selected_lsoa <- postcode_to_lsoa %>%
```

```r
postcode_to_lsoa <- read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Postcode to LSOA.csv")
population <- read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/Population.csv")
#Selecting relevant columns
selected_lsoa <- postcode_to_lsoa %>%
  select(`lsoa11cd`, `lsoa11nm`, `ladnm`, `pcds`)

selected_crime <- crime_combined %>%
  select(Month, `LSOA code`, `Crime type`, `Falls within`)

#Renaming the columns
colnames(selected_lsoa) = c('LSOA code', 'street', 'counties', "postcode")
colnames(population) = c("postcode", "population")
View(selected_crime)
View(selected_lsoa)
#Cleaning LSOA
clean_lsoa <- selected_lsoa %>%
  filter(counties %in% c("Bristol, City of","Cornwall")) %>%
  mutate(postcode=str_trim((substring(postcode,1,6))))

#Checking for duplicates of lsoa codes
any(duplicated(selected_crime$`LSOA code`))
any(duplicated(clean_lsoa$`LSOA code`))

#Since the duplicates are there, removing them
clean_lsoa=unique(clean_lsoa,by="LSOA code")
selected_crime=unique(selected_crime,by="LSOA code")

#Final Cleaning and merging
finalCrime= selected_crime %>%
  left_join(clean_lsoa, by=("LSOA code"),relationship = "many-to-many") %>%
  mutate(Year=str_trim(substring(Month,1,4))) %>%
```

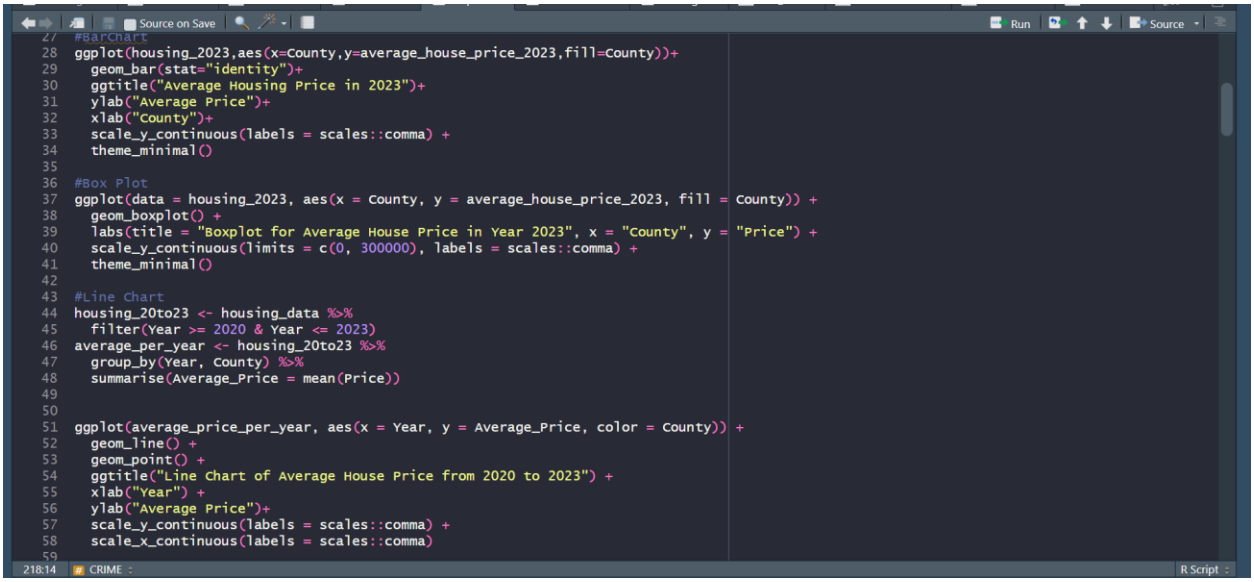
```r
#Final Cleaning and merging
finalCrime= selected_crime %>%
  left_join(clean_lsoa, by=("LSOA code"),relationship = "many-to-many") %>%
  mutate(Year=str_trim(substring(Month,1,4))) %>%
  mutate(Month=str_trim(substring(Month,6,7))) %>%
  left_join(population,by="postcode") %>%
  distinct() %>%
  na.omit()

dim(finalCrime)
dim(clean_lsoa)
View(finalCrime)
dim
write_csv(finalCrime, "D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/crime_cleaned.csv")

#-------------------------------SCHOOL---------------------------
bSchool21=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/School/Bristol/2021-2022/801_ks4final.csv")
bSchool22=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/School/Bristol/2022-2023/801_ks4final.csv")
cSchool21=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/School/Cornwall/2021-2022/908_ks4final.csv")
cSchool22=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Obtained Datasets/School/Cornwall/2022-2023/908_ks4final.csv")

view(bSchool21)
head(bSchool21)
view(cSchool21)

bSchool21 <- bSchool21 %>%
  select(SCHNAME,PCODE,ATT8SCR,TOWN) %>%
  mutate(YEAR=2021,COUNTY="Bristol")

bSchool22 <- bSchool22 %>%
  select(SCHNAME,PCODE,ATT8SCR,TOWN) %>%
```

```r
bSchool21 <- bSchool21 %>%
  select(SCHNAME,PCODE,ATT8SCR,TOWN) %>%
  mutate(YEAR=2021,COUNTY="Bristol")

bSchool22 <- bSchool22 %>%
  select(SCHNAME,PCODE,ATT8SCR,TOWN) %>%
  mutate(YEAR=2022,COUNTY="Bristol")

cSchool21 <- cSchool21 %>%
  select(SCHNAME,PCODE,ATT8SCR,TOWN) %>%
  mutate(YEAR=2021,COUNTY="Cornwall")

cSchool22 <- cSchool22 %>%
  select(SCHNAME,PCODE,ATT8SCR,TOWN) %>%
  mutate(YEAR=2022,COUNTY="Cornwall")

#Combining
bCombined=rbind(bSchool21,bSchool22)
cCombined=rbind(cSchool21,cSchool22)

View(bCombined)
View(cCombined)

schoolCombined=rbind(bCombined,cCombined)
View(schoolCombined)

schoolCombined <- schoolCombined %>%
  filter(ATT8SCR!="NE"&ATT8SCR!="SUPP") %>%
  na.omit() %>%
  distinct()
```

```r
  mutate(YEAR=2022,COUNTY="Cornwall")

#Combining
bCombined=rbind(bSchool21,bSchool22)
cCombined=rbind(cSchool21,cSchool22)

View(bCombined)
View(cCombined)

schoolCombined=rbind(bCombined,cCombined)
View(schoolCombined)

schoolCombined <- schoolCombined %>%
  filter(ATT8SCR!="NE"&ATT8SCR!="SUPP") %>%
  na.omit() %>%
  distinct()

dim(schoolCombined)
View(schoolCombined)

write_csv(schoolCombined,"D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/schoolcleaned.csv")
```

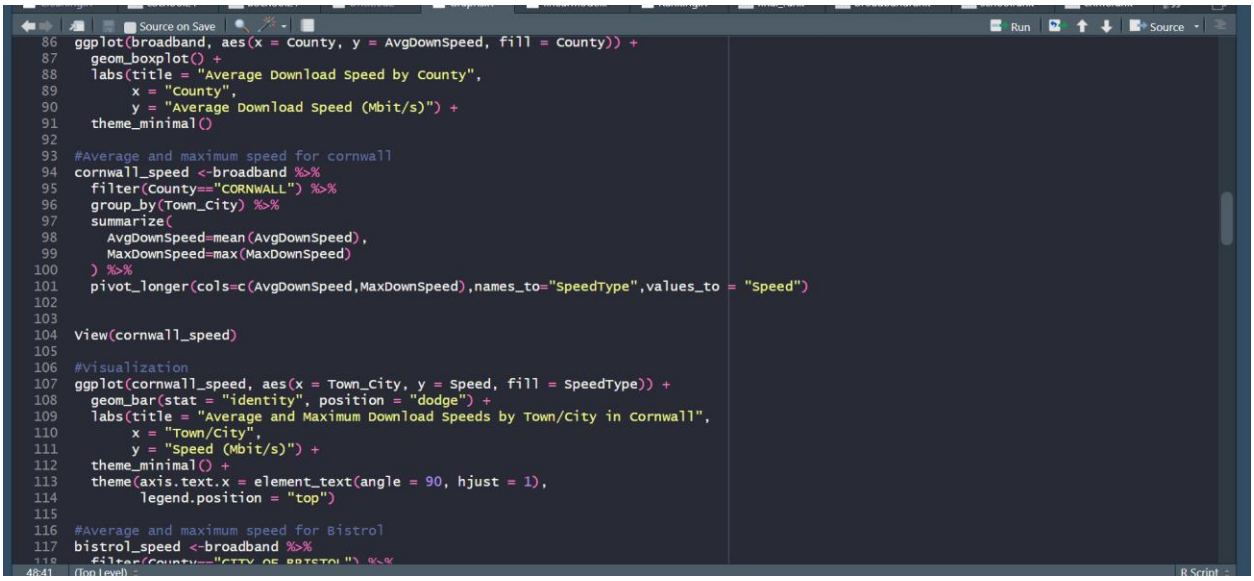
Graphs.R

```r
library(tidyverse)
library(ggplot2)
library(fmsb)
library(scales)

#Housing
housing_data=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/housing_cleaned.csv")

#For Average House Price of 2023 for Bristol and Cornwall

housing_2023<- housing_data %>%
  filter(Year==2023) %>%
  group_by(County) %>%
  summarise(average_house_price_2023=mean(Price))

colnames(housing_data)

#Avaeage house price for each county for 2020-2023
housing_20_23=housing_data %>%
  filter(Year>=2020) %>%
  group_by(Year,County) %>%
  summarise(avg_house_price=mean(Price))

head(housing_2023)
head(housing_20_23)

#BarChart
ggplot(housing_2023,aes(x=County,y=average_house_price_2023,fill=County))+
  geom_bar(stat="identity")+
  ggtitle("Average Housing Price in 2023")+
  ylab("Average Price")+
  xlab("County")+
```



```r
#BarChart
ggplot(housing_2023,aes(x=County,y=average_house_price_2023,fill=County))+
  geom_bar(stat="identity")+
  ggtitle("Average Housing Price in 2023")+
  ylab("Average Price")+
  xlab("County")+
  scale_y_continuous(labels = scales::comma) +
  theme_minimal()

#Box Plot
ggplot(data = housing_2023, aes(x = County, y = average_house_price_2023, fill = County)) +
  geom_boxplot() +
  labs(title = "Boxplot for Average House Price in Year 2023", x = "County", y = "Price") +
  scale_y_continuous(limits = c(0, 300000), labels = scales::comma) +
  theme_minimal()

#Line Chart
housing_20to23 <- housing_data %>%
  filter(Year >= 2020 & Year <= 2023)
average_per_year <- housing_20to23 %>%
  group_by(Year, County) %>%
  summarise(Average_Price = mean(Price))


ggplot(average_price_per_year, aes(x = Year, y = Average_Price, color = County)) +
  geom_line() +
  geom_point() +
  ggtitle("Line Chart of Average House Price from 2020 to 2023") +
  xlab("Year") +
  ylab("Average Price")+
  scale_y_continuous(labels = scales::comma) +
  scale_x_continuous(labels = scales::comma)
```

```r

#housing by years
housing_by_year = housing_data %>%
  filter(Year >= 2020) %>%
  group_by(Year, Town_City) %>%
  summarise(avg_price = mean(Price))

# Filtering data for 2023
housing_data_2023 = housing_by_year %>%
  filter(Year == 2023)

ggplot(housing_data_2023, aes(x = Town_City, y = avg_price, fill = Town_City)) +
  geom_bar(stat = "identity") +
  ggtitle("Average House Price by Town in 2023") +
  ylab("Average Price") +
  xlab("City") +
  scale_y_continuous(labels = scales::comma) +
  theme_minimal()+
  theme(axis.text.x = element_text(angle = 90, hjust = 1))


#----------Broadband Data
broadband=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/broadband_cleaned.csv")

colnames(broadband)
view(broadband)
#Average download speed
ggplot(broadband, aes(x = County, y = AvgDownSpeed, fill = County)) +
  geom_boxplot() +
  labs(title = "Average Download Speed by County",
       x = "County",
       y = "Average Download Speed (Mbit/s)") +
```



```r
ggplot(broadband, aes(x = County, y = AvgDownSpeed, fill = County)) +
  geom_boxplot() +
  labs(title = "Average Download Speed by County",
       x = "County",
       y = "Average Download Speed (Mbit/s)") +
  theme_minimal()

#Average and maximum speed for cornwall
cornwall_speed <-broadband %>%
  filter(County=="CORNWALL") %>%
  group_by(Town_City) %>%
  summarize(
    AvgDownSpeed=mean(AvgDownSpeed),
    MaxDownSpeed=max(MaxDownSpeed)
  ) %>%
  pivot_longer(cols=c(AvgDownSpeed,MaxDownSpeed),names_to="SpeedType",values_to = "Speed")


view(cornwall_speed)

#Visualization
ggplot(cornwall_speed, aes(x = Town_City, y = Speed, fill = SpeedType)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Average and Maximum Download Speeds by Town/City in Cornwall",
       x = "Town/City",
       y = "Speed (Mbit/s)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1),
        legend.position = "top")

#Average and maximum speed for Bistrol
bistrol_speed <-broadband %>%
```

```r
#Average and maximum speed for Bistrol
bistrol_speed <-broadband %>%
  filter(County=="CITY OF BRISTOL") %>%
  group_by(Town_City) %>%
  summarize(
    AvgDownSpeed=mean(AvgDownSpeed),
    MaxDownSpeed=max(MaxDownSpeed)
  ) %>%
  pivot_longer(cols=c(AvgDownSpeed,MaxDownSpeed),names_to="SpeedType",values_to = "Speed")

view(bistrol_speed)

#Visualization
ggplot(bistrol_speed, aes(x = Town_City, y = Speed, fill = SpeedType)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Average and Maximum Download Speeds by Town/City in Bistrol",
       x = "Town/City",
       y = "Speed (Mbit/s)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1),
        legend.position = "top")

#---------------------CRIME-------------------------------------------
crime_data = read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/crime_cleaned.csv")
View(crime_data)
colnames(crime_data)
dim(crime_data)

# Vehicle Crime per 10000 people in April 2022
vehicleCrime <- crime_data %>%
  filter(`Crime type` == "Vehicle crime")
colnames(vehicleCrime)
```

```r
vehicle_sum <- vehicleCrime %>%
  group_by(Year) %>%
  summarise(total_crime = sum(population, na.rm = TRUE))

view(vehicle_sum)

for_radar = as.data.frame(t(vehicle_sum$total_crime))
colnames(for_radar) <- vehicle_sum$Year
view(for_radar)

for_radar <- rbind(rep(max(vehicle_sum$total_crime), length(years)),
                   rep(0, length(years)),
                   for_radar)

par(mar = c(2, 2, 2, 2))

#Plotting Radar Chart for Vehicle Crime
radarchart(for_radar,
           axistype = 1,
           pcol = "purple",
           pfcol = "lightblue",
           plwd = 4,
           title = "Vehicle Crime Rate from 2021 to 2024"
)


#Pie Chart for Robbery in 2023 by month

robbery_data =crime_data %>%
  filter(`Crime type`=="Robbery" & Year=="2023") %>%
  group_by(Month) %>%
  summarise(count=n()) %>%
```

```r
#Pie Chart for Robbery in 2023 by month

robbery_data =crime_data %>%
  filter(`Crime type`=="Robbery" & Year=="2023") %>%
  group_by(Month) %>%
  summarise(count=n()) %>%
  mutate(percentage=count/sum(count)*100)

#Plotting the Pie Chart
ggplot(robbery_data,aes(x = "", y = percentage, fill = as.factor(Month))) +
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y") +
  geom_text(aes(label = paste0(round(percentage, 1), "%")),
            position = position_stack(vjust = 0.5)) +
  labs(title = "Robberies by Month in 2023", fill = "Month") +
  theme_minimal()


#Drugs

drugCrime <- subset(crime, `Crime type` == "Drugs" & Year == 2023 & counties %in% c("Bristol, City of", "Cornwall"))

# Create the boxplot

ggplot(drugCrime, aes(x = counties, y = population / 10000, fill = counties)) +
  geom_boxplot() +
  labs(title = "Drug Offence Rate per 10,000 People in 2023",
       x = "County",
       y = "Drug Offence Rate (per 10,000 people)")+
  theme_minimal()
```

```r
# Filtering data for drug offences
drug2023 = crime %>%
  filter(`Crime type` == "Drugs" & Year == "2023")

# For Cornwall
cornDrug <- drugCrime %>%
  filter(counties == "Cornwall") %>%
  distinct(`LSOA code`, .keep_all = TRUE) %>%
  summarise(total_population = sum(population),
            total_drug_offences = n())
cornwall_data


# Summarize data for Bristol
bristolDrug <- drugCrime %>%
  filter(counties == "Bristol, City of") %>%
  distinct(`LSOA code`, .keep_all = TRUE) %>%
  summarise(total_population = sum(population),
            total_drug_offences = n())

cornDrug <- cornDrug %>%
  mutate(offence_rate = (total_drug_offences / total_population) * 10000)

bristolDrug <- bristolDrug %>%
  mutate(offence_rate = (total_drug_offences / total_population) * 10000)


combined_data <- bind_rows(
  cornDrug %>% mutate(county = "Cornwall"),
  bristolDrug %>% mutate(county = "Bristol, City of")
)
```

```r

ggplot(combined_data, aes(x = county, y = offence_rate, fill = county)) +
  geom_boxplot() +
  labs(title = "Distribution of Drug Offence Rates (2023)",
       x = "Location",
       y = "Offence Rate (per 10,000)") +
  theme_minimal()


#_____________________Schools

school_data= read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/schoolcleaned.csv")

head(school_data)
school_filtered= school_data %>%
  filter(YEAR==2022)

#BOXPLOT: Average Attainment 8 scores by County in 2022
ggplot(school_filtered,aes(x = COUNTY, y = ATT8SCR, fill = COUNTY)) +
  geom_boxplot() +
  labs(title = "Average Attainment 8 Scores by County in 2022",
       x = "County",
       y = "Attainment 8 Score") +
  theme_minimal()

#Line Graph : Average Attainment 8 Score in Academic Year 2021-2022 in BISTROL
bristol_21=school_data %>%
  filter(YEAR==2021) %>%
  filter(COUNTY=="Bristol")

ggplot(bristol_21, aes(x = SCHNAME, y = ATT8SCR, group = 1)) +
  geom_line(color = "purple") +
```

```r
  theme_minimal()

#Line Graph : Average Attainment 8 Score in Academic Year 2021-2022 in BISTROL
bristol_21=school_data %>%
  filter(YEAR==2021) %>%
  filter(COUNTY=="Bristol")

ggplot(bristol_21, aes(x = SCHNAME, y = ATT8SCR, group = 1)) +
  geom_line(color = "purple") +
  geom_point() +
  labs(title = "Average Attainment 8 Score in 2021-2022 in Bristol",
       x = "School Name",
       y = "Average Attainment 8 Score") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))

#Line Graph : Average Attainment 8 Score in Academic Year 2021-2022 in CORNWALL
cornwall_21=school_data %>%
  filter(YEAR==2021) %>%
  filter(COUNTY=="Cornwall")
colnames(cornwall_21)
head(cornwall_21)

ggplot(cornwall_21, aes(x = TOWN, y = ATT8SCR, group = 1)) +
  geom_line(color = "red") +
  geom_point() +
  labs(title = "Average Attainment 8 Score in 2021-2022 in Cornwall",
       x = "Town",
       y = "Average Attainment 8 Score") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

LinearModel.R

```r
library(tidyverse)
library(ggplot2)

#Importing Datasets
crimeData=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/crime_cleaned.csv")
schoolData=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/schoolcleaned.csv")
housingData=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/housing_cleaned.csv")
broadbandData=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/broadband_cleaned.csv")

#Linear Model for : Housing Price vs Average Download Speed
housingPrice <- housingData %>%
  select(Postcode,Price)

downSpeed <- broadbandData %>%
  select(Postcode,AvgDownSpeed)

housing_speed <- merge(housingPrice, downSpeed, by="Postcode")

View(housing_speed)

ggplot(data = housing_speed, aes(x = Price, y = AvgDownSpeed)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(title = "House Prices vs Average Download Speed",
       x = "House Price",
       y = "Average Download Speed (Mbit/s)") +
  scale_y_continuous(labels = scales::comma) +
  scale_x_continuous(labels = scales::comma) +
  theme_minimal()

#Linear Model for Housing Price Vs Drug Offence
housingPrice23=housingData %>%
```

```r

#Linear Model for Housing Price Vs Drug Offence
housingPrice23=housingData %>%
  filter(Year==2023) %>%
  mutate(Postcode=str_trim((substring(Postcode,1,6)))) %>%
  distinct()

drug_offence_23 <-crimeData %>%
  filter(Year==2023, `Crime type`=="Drugs") %>%
  group_by(postcode) %>%
  summarise(DrugRate=sum(population)) %>%
  distinct() %>%
  na.omit()

housing_drugs <- housingPrice23 %>%
  left_join(drug_offence_23,by=c("Postcode"="postcode")) %>%
  distinct() %>%
  na.omit()

ggplot(data = housing_drugs, aes(x = Price, y = DrugRate)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(title = "Housing Price vs Drug Rates in 2023",
       x = "House Price",
       y = "Drug Rate") +
  scale_y_continuous(labels = scales::comma) +
  scale_x_continuous(labels = scales::comma) +
  theme_minimal()

#Linear Model for Average Download Speed Vs Attainment 8 Score
school <- schoolData %>%
  filter(YEAR==2022) %>%
```

```r
#Linear Model for Average Download Speed Vs Attainment 8 Score
school <- schoolData %>%
  filter(YEAR==2022) %>%
  select(PCODE, ATT8SCR) %>%
  rename(Postcode="PCODE")

school_broadband <-merge(downSpeed,school, by= "Postcode")

ggplot(data = school_broadband, aes(x = AvgDownSpeed, y = ATT8SCR)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(title = "Average download speed VS Attainment 8 Score",
       x = "Average download speed",
       y = "Attainment 8 Score") +
  scale_y_continuous(labels = scales::comma) +
  scale_x_continuous(labels = scales::comma) +
  theme_minimal()


#Linear Model for Attainment 8 Score Vs Housing Price
school_housing = merge(school, housingPrice, by = "Postcode")

ggplot(data = school_housing, aes(x = ATT8SCR, y = Price)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(title = "Average attainment 8 score vs Housing Price",
       x = "ATT8SCR",
       y = "Price") +
  scale_y_continuous(labels = scales::comma) +
  scale_x_continuous(labels = scales::comma) +
  theme_minimal()
```

Ranking.R

```r
library(tidyverse)

#Importing Libraries
crime=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/crime_cleaned.csv")
school=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/schoolcleaned.csv")
housingPrice=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/housing_cleaned.csv")
broadband=read_csv("D:/Academics/Fourth Semester/Data Science/Assignment/Cleaned Datasets/broadband_cleaned.csv")

dim(crime)

#Selecting Relevent Rows from Broadband
selected_broadband= broadband %>%
  group_by(Town_City) %>%
  summarise(
    avg_upl_speed = mean(AvgUpSpeed),
    avg_down_speed = mean(AvgDownSpeed)
  )%>%
  mutate(TOWN =  str_trim(toupper(Town_City)))%>%
  select(TOWN, avg_upl_speed, avg_down_speed)

selected_attainment8= school %>%
  group_by(TOWN) %>%
  summarise(avgAtt8 = mean(ATT8SCR))%>%
  select(TOWN, avgAtt8)%>%
  distinct()%>%
  mutate(TOWN= str_trim(toupper(TOWN)))

selected_house = housingPrice %>%
  filter(Year == 2023)%>%
  mutate(TOWN =  str_trim(toupper(Town_City)))%>%
  group_by(TOWN) %>%
  summarise(avgPrice = mean(Price))%>%
```

```r
selected_house = housingPrice %>%
  filter(Year == 2023)%>%
  mutate(TOWN =  str_trim(toupper(Town_City)))%>%
  group_by(TOWN) %>%
  summarise(avgPrice = mean(Price))%>%
  select(avgPrice, TOWN)%>%
  na.omit()%>%
  distinct()

town = housingPrice %>%
  mutate(postcode  = str_trim(substring(Postcode,  1, 6))) %>%
  mutate(TOWN =  str_trim(toupper(Town_City)))%>%
  select(postcode, TOWN)%>%
  distinct()

selected_crime = crime %>%
  filter(Year==2023)%>%
  group_by(postcode)%>%
  summarise(crimeno =n())%>%
  arrange(desc(crimeno))%>%
  select(postcode, crimeno)

final_crime = selected_crime%>%
  left_join(town, by= "postcode")%>%
  na.omit()%>%
  distinct()

final_crime = final_crime%>%
  group_by(TOWN)%>%
  summarise(crimerate = sum(crimeno))%>%
  select(TOWN, crimerate)
```

```r
ranking = selected_house %>%
  left_join(selected_attainment8, by = "TOWN") %>%
  left_join(selected_broadband, by = "TOWN") %>%
  left_join(final_crime, by = "TOWN") %>%
  na.omit()
view(ranking)


# Calculating the minimum and Maximum for each column
Extremes <- ranking_Points %>%
  summarise(
    minDownSpeed= min(avg_down_speed),
    maxDownSpeed = max(avg_down_speed),
    minUpSpeed = min(avg_upl_speed),
    maxUpSpeed = max(avg_upl_speed),
    minAtt8 = min(avgAtt8),
    maxAtt8 = max(avgAtt8),
    minHousingPrice = min(avgPrice),
    maxHousingPrice = max(avgPrice),
    minCrimeRate = min(crimerate),
    maxCrimeRate = max(crimerate)
  )

# Normalizing and calculating the final points
finalRanking <- ranking_Points %>%
  mutate(
    normDownSpeed = (avg_down_speed - Extremes$minDownSpeed) / (Extremes$maxDownSpeed - Extremes$minDownSpeed),
    normUpSpeed = (avg_upl_speed - Extremes$minUpSpeed) / (Extremes$maxUpSpeed - Extremes$minUpSpeed),
    normAtt8 = (avgAtt8 - Extremes$minAtt8) / (Extremes$maxAtt8 - Extremes$minAtt8),
    normHousingPrice = 1 - (avgPrice - Extremes$minHousingPrice) / (Extremes$maxHousingPrice - Extremes$minHousingPrice),
    normCrimeRate = 1 - (crimerate - Extremes$minCrimeRate) / (Extremes$maxCrimeRate - Extremes$minCrimeRate),
    finalPoints = normDownSpeed + normUpSpeed + normAtt8 + normHousingPrice + normCrimeRate
```

```r
# View the data with the final score
finalRanking= finalRanking%>%
  arrange(desc(finalPoints))

view(finalRanking)
write_csv(finalRanking, "D:/Academics/Fourth Semester/Data Science/Assignment/ranking.csv")

houserank = finalRanking %>%
  select(TOWN, avgPrice, normHousingPrice)%>%
  arrange(desc(normHousingPrice))

view(houserank)

crimerank = finalRanking %>%
  select(TOWN, normCrimeRate, crimerate) %>%
  arrange(desc(normCrimeRate))
view(crimerank)

schoolrank = finalRanking %>%
  select(TOWN, avgAtt8, normAtt8)%>%
  arrange(desc(normAtt8))
view(schoolrank)

broadbandrank = finalRanking %>%
  select(TOWN, avg_down_speed, normDownSpeed)%>%
  arrange(desc(normDownSpeed))

view(broadbandrank)

final_rank = finalRanking  %>%
  select(TOWN, avgPrice, crimerate, avgAtt8, avg_down_speed, finalPoints) %>%
  arrange(desc(finalPoints))
```