Town Recommendation System and Report on an Individual Data Science Project

Bipasha Lamsal

BSc. (Hons.) Computing, Softwarica College of IT and E-commerce, Coventry University

Data Science for Developers, ST5014CEM

Siddhartha Neupane

August 19, 2024

Table of Contents

Table of Figures

Introduction

This project focuses on analyzing data to find the best towns or cities for living or investing. We looked at important factors like house prices, internet speed, school grades, and crime rates. By putting all these factors together, we aimed to identify the top places that offer a good balance of affordability, quality of life, and safety.

We used different datasets, each containing information about these factors in various towns and cities. The data was cleaned, combined, and then analyzed to create an overall score for each location. This score helps in ranking the towns and cities based on how well they meet the criteria we set.

The purpose of this document is to explain the steps we took, the results we found, and the recommendations we can make based on this analysis. We also discuss any challenges we faced and how this analysis could be improved in the future.

Cleaning Data

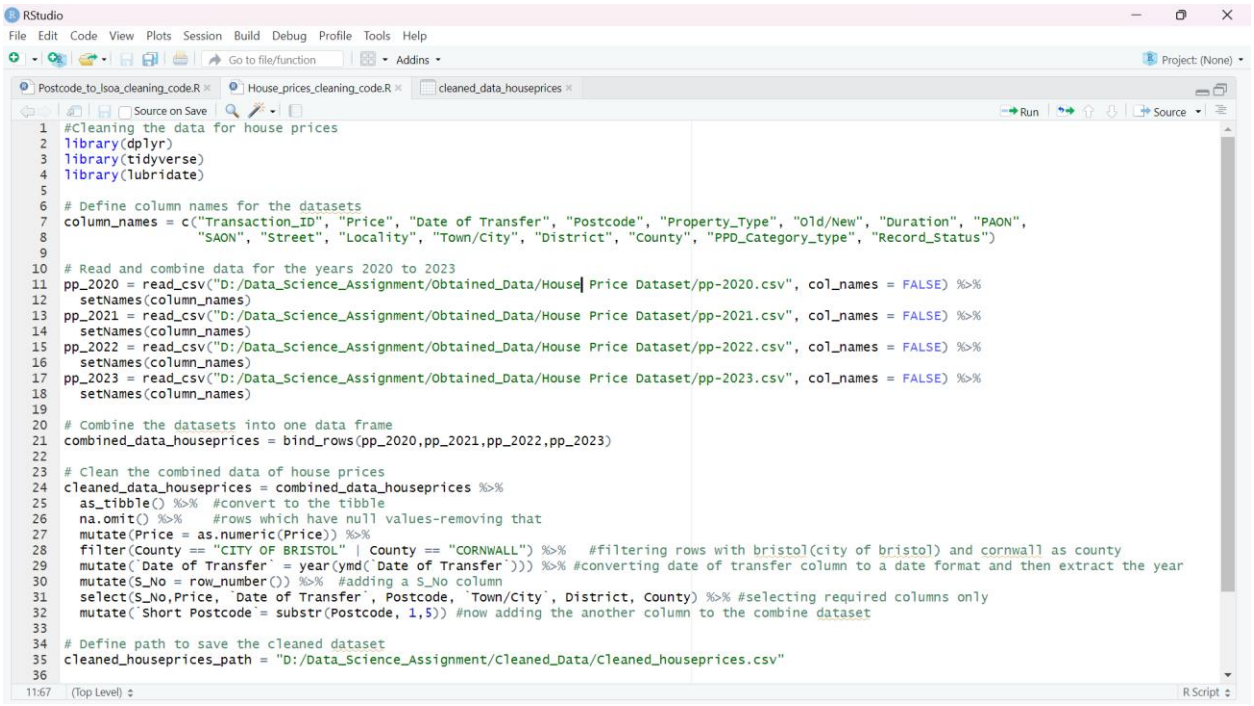
House Price

The first step involved reading and combining datasets from 2020 to 2023 into a single data frame with consistent column names. The Price column was then converted to a numeric format, and any rows with missing values were eliminated in order to clean up the data. In order to concentrate on particular areas, the dataset was filtered so that it only contained records from "CITY OF BRISTOL" and "CORNWALL." A new column called Short Postcode was created by taking the first five characters out of the Postcode and simplifying the Date of Transfer to just the year. This preparation guarantees that the dataset is appropriate for the analysis and manageable. (Government, n.d.)

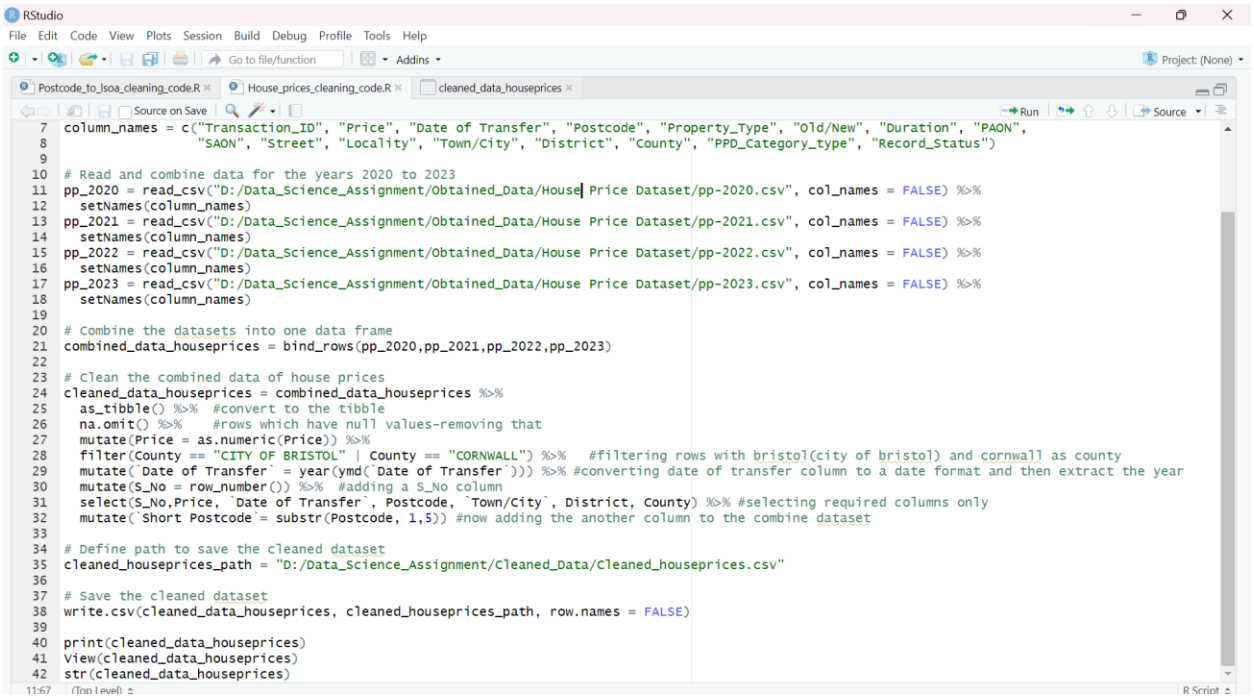
After the dataset was cleansed, just the important columns—S_No, Price, Date of Transfer, Postcode, Town/City, District, and County—were left. This approach makes the dataset more usable and less difficult. After processing, the data was put into a CSV file for further analysis. (GeeksforGeeks, n.d.)

Figure 1

House Prices Cleaning



```r
#Cleaning the data for house prices
library(dplyr)
library(tidyverse)
library(lubridate)

# Define column names for the datasets
column_names = c("Transaction_ID", "Price", "Date of Transfer", "Postcode", "Property_Type", "Old/New", "Duration", "PAON",
                 "SAON", "Street", "Locality", "Town/City", "District", "County", "PPD_Category_type", "Record_Status")

# Read and combine data for the years 2020 to 2023
pp_2020 = read_csv("D:/Data_Science_Assignment/Obtained_Data/House Price Dataset/pp-2020.csv", col_names = FALSE) %>%
  setNames(column_names)
pp_2021 = read_csv("D:/Data_Science_Assignment/Obtained_Data/House Price Dataset/pp-2021.csv", col_names = FALSE) %>%
  setNames(column_names)
pp_2022 = read_csv("D:/Data_Science_Assignment/Obtained_Data/House Price Dataset/pp-2022.csv", col_names = FALSE) %>%
  setNames(column_names)
pp_2023 = read_csv("D:/Data_Science_Assignment/Obtained_Data/House Price Dataset/pp-2023.csv", col_names = FALSE) %>%
  setNames(column_names)

# Combine the datasets into one data frame
combined_data_houseprices = bind_rows(pp_2020,pp_2021,pp_2022,pp_2023)

# Clean the combined data of house prices
cleaned_data_houseprices = combined_data_houseprices %>%
  as_tibble() %>%  #convert to the tibble
  na.omit() %>%    #rows which have null values-removing that
  mutate(Price = as.numeric(Price)) %>%
  filter(County == "CITY OF BRISTOL" | County == "CORNWALL") %>%   #filtering rows with bristol(city of bristol) and cornwall as county
  mutate(`Date of Transfer` = year(ymd(`Date of Transfer`))) %>% #converting date of transfer column to a date format and then extract the year
  mutate(S_No = row_number()) %>%  #adding a S_No column
  select(S_No,Price, `Date of Transfer`, Postcode, `Town/City`, District, County) %>% #selecting required columns only
  mutate(`Short Postcode`= substr(Postcode, 1,5)) #now adding the another column to the combine dataset

# Define path to save the cleaned dataset
cleaned_houseprices_path = "D:/Data_Science_Assignment/Cleaned_Data/Cleaned_houseprices.csv"
```



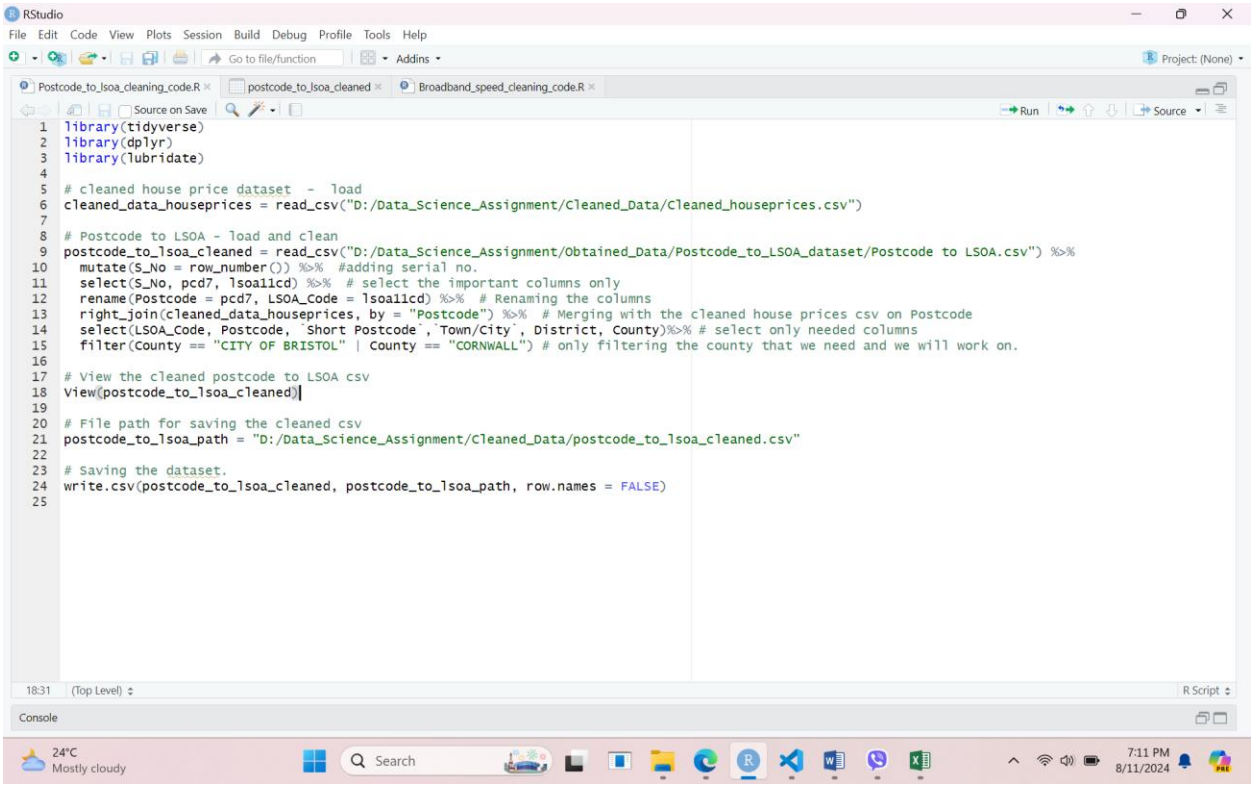
```r
column_names = c("Transaction_ID", "Price", "Date of Transfer", "Postcode", "Property_Type", "Old/New", "Duration", "PAON",
                 "SAON", "Street", "Locality", "Town/City", "District", "County", "PPD_Category_type", "Record_Status")

# Read and combine data for the years 2020 to 2023
pp_2020 = read_csv("D:/Data_Science_Assignment/Obtained_Data/House Price Dataset/pp-2020.csv", col_names = FALSE) %>%
  setNames(column_names)
pp_2021 = read_csv("D:/Data_Science_Assignment/Obtained_Data/House Price Dataset/pp-2021.csv", col_names = FALSE) %>%
  setNames(column_names)
pp_2022 = read_csv("D:/Data_Science_Assignment/Obtained_Data/House Price Dataset/pp-2022.csv", col_names = FALSE) %>%
  setNames(column_names)
pp_2023 = read_csv("D:/Data_Science_Assignment/Obtained_Data/House Price Dataset/pp-2023.csv", col_names = FALSE) %>%
  setNames(column_names)

# Combine the datasets into one data frame
combined_data_houseprices = bind_rows(pp_2020,pp_2021,pp_2022,pp_2023)

# Clean the combined data of house prices
cleaned_data_houseprices = combined_data_houseprices %>%
  as_tibble() %>%  #convert to the tibble
  na.omit() %>%    #rows which have null values-removing that
  mutate(Price = as.numeric(Price)) %>%
  filter(County == "CITY OF BRISTOL" | County == "CORNWALL") %>%   #filtering rows with bristol(city of bristol) and cornwall as county
  mutate(`Date of Transfer` = year(ymd(`Date of Transfer`))) %>% #converting date of transfer column to a date format and then extract the year
  mutate(S_No = row_number()) %>%  #adding a S_No column
  select(S_No,Price, `Date of Transfer`, Postcode, `Town/City`, District, County) %>% #selecting required columns only
  mutate(`Short Postcode`= substr(Postcode, 1,5)) #now adding the another column to the combine dataset

# Define path to save the cleaned dataset
cleaned_houseprices_path = "D:/Data_Science_Assignment/Cleaned_Data/Cleaned_houseprices.csv"

# Save the cleaned dataset
write.csv(cleaned_data_houseprices, cleaned_houseprices_path, row.names = FALSE)

print(cleaned_data_houseprices)
View(cleaned_data_houseprices)
str(cleaned_data_houseprices)
```

Towns and Postcodes

In order to align with the house price data, the postcode to LSOA dataset underwent preprocessing and cleaning. For distinct identity, a serial number column (S_No) was introduced. The only columns that were kept and renamed to comply with the naming conventions of the home price dataset were Postcode and LSOA_Code. This data was combined with the house price information using a right join based on the Postcode column. The two subsets of the combined dataset that were retained were "CITY OF BRISTOL" and "CORNWALL." Effective integration and clarity were ensured by saving the final cleaned dataset as a CSV file for additional analysis.

Figure 2

Postcode to Lsoa Cleaning



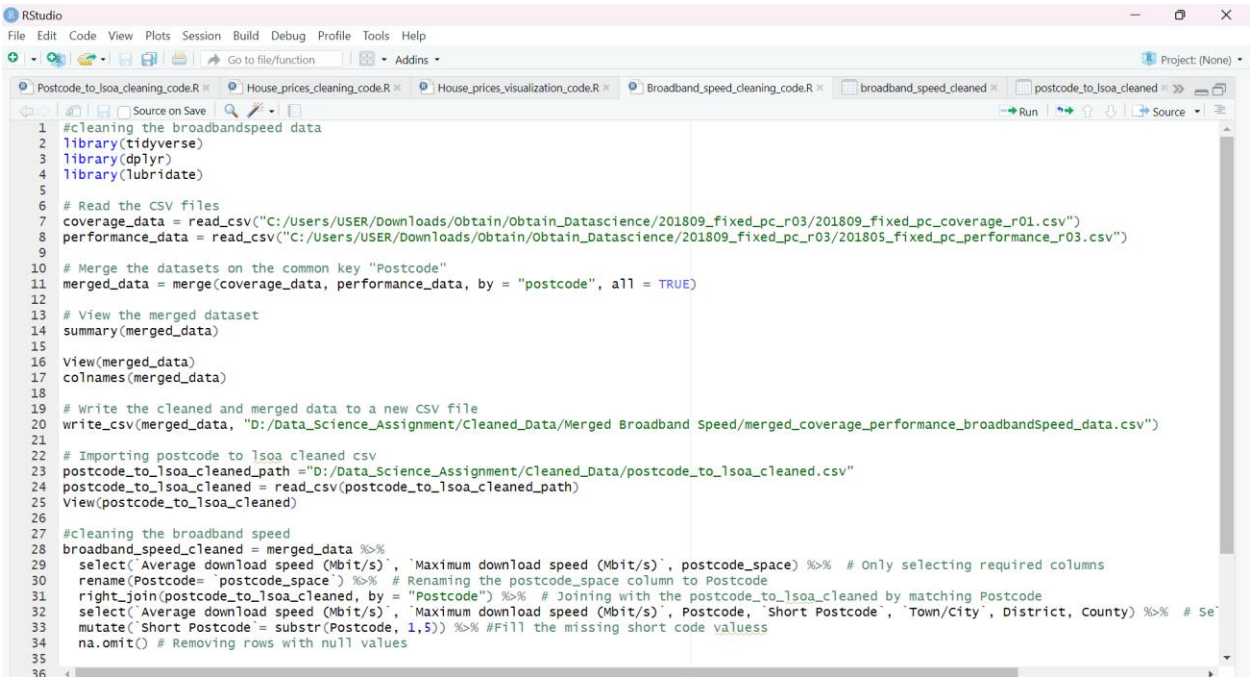
Broadband Speed

The broadband speed data required going through a number of steps of cleaning and preprocessing in order for it to be accurate and consistent with other datasets. Initially, the CSV files containing the coverage_data and performance_data were read and combined using the shared postcode key. Coverage and performance data were merged into a single dataset during this process. After merging, the dataset was reviewed to understand its structure and contents.

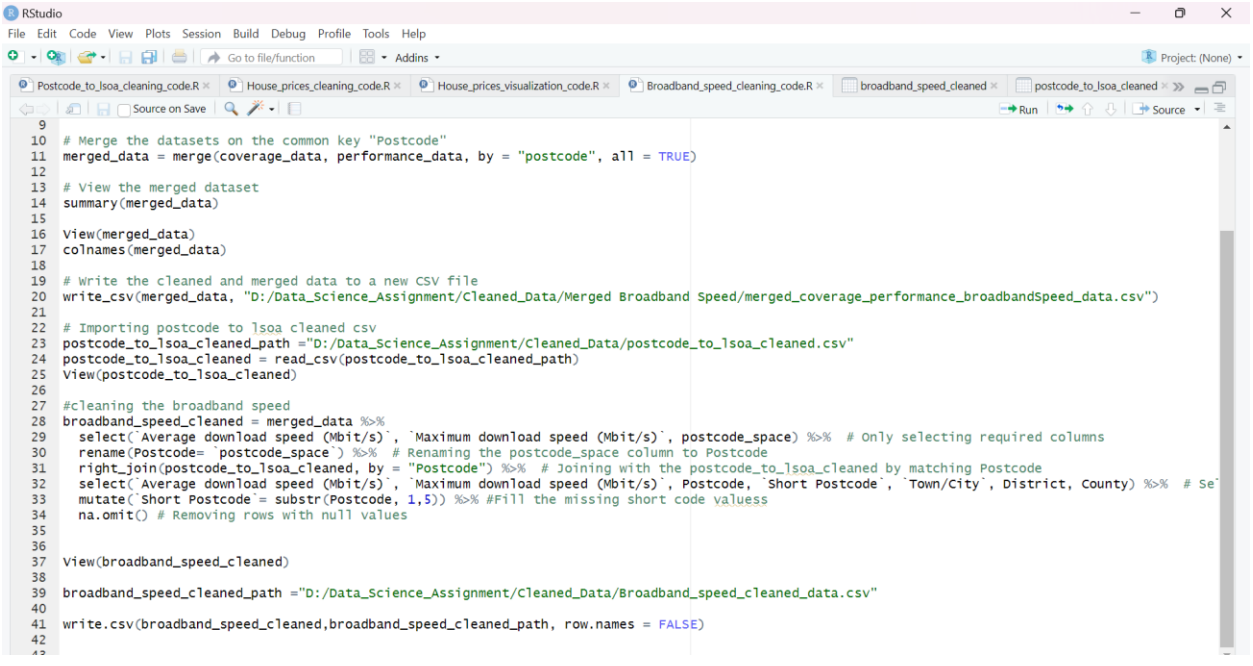
After that, the combined dataset was cleaned by selecting only important columns, such as postcode_space, Maximum download speed (Mbit/s), and Average download speed (Mbit/s). For uniformity, the postcode_space column was renamed to Postcode. After the postcode was cleaned, a right join was done with the LSOA dataset to match location data with broadband speed data. A Short Postcode column was added to the dataset, and any rows with missing values were eliminated. Ultimately, the dataset was cleaned and stored in a CSV file for later analysis. (Ofcom, n.d.)

Figure 3

Broadband Speed Cleaning



```
#cleaning the broadbandspeed data
library(tidyverse)
library(dplyr)
library(lubridate)

# Read the CSV files
coverage_data = read_csv("C:/Users/USER/Downloads/Obtain/Obtain_Datascience/201809_fixed_pc_r03/201809_fixed_pc_coverage_r01.csv")
performance_data = read_csv("C:/Users/USER/Downloads/Obtain/Obtain_Datascience/201809_fixed_pc_r03/201805_fixed_pc_performance_r03.csv")

# Merge the datasets on the common key "Postcode"
merged_data = merge(coverage_data, performance_data, by = "postcode", all = TRUE)

# View the merged dataset
summary(merged_data)

View(merged_data)
colnames(merged_data)

# Write the cleaned and merged data to a new CSV file
write_csv(merged_data, "D:/Data_Science_Assignment/Cleaned_Data/Merged Broadband Speed/merged_coverage_performance_broadbandSpeed_data.csv")

# Importing postcode to lsoa cleaned csv
postcode_to_lsoa_cleaned_path ="D:/Data_Science_Assignment/Cleaned_Data/postcode_to_lsoa_cleaned.csv"
postcode_to_lsoa_cleaned = read_csv(postcode_to_lsoa_cleaned_path)
View(postcode_to_lsoa_cleaned)

#cleaning the broadband speed
broadband_speed_cleaned = merged_data %>%
  select(`Average download speed (Mbit/s)`, `Maximum download speed (Mbit/s)`, postcode_space) %>%  # Only selecting required columns
  rename(Postcode= `postcode_space`) %>%  # Renaming the postcode_space column to Postcode
  right_join(postcode_to_lsoa_cleaned, by = "Postcode") %>%  # Joining with the postcode_to_lsoa_cleaned by matching Postcode
  select(`Average download speed (Mbit/s)`, `Maximum download speed (Mbit/s)`, Postcode, `Short Postcode`, `Town/City`, District, County) %>%  # Se
  mutate(`Short Postcode`= substr(Postcode, 1,5)) %>% #Fill the missing short code valuess
  na.omit() # Removing rows with null values
```



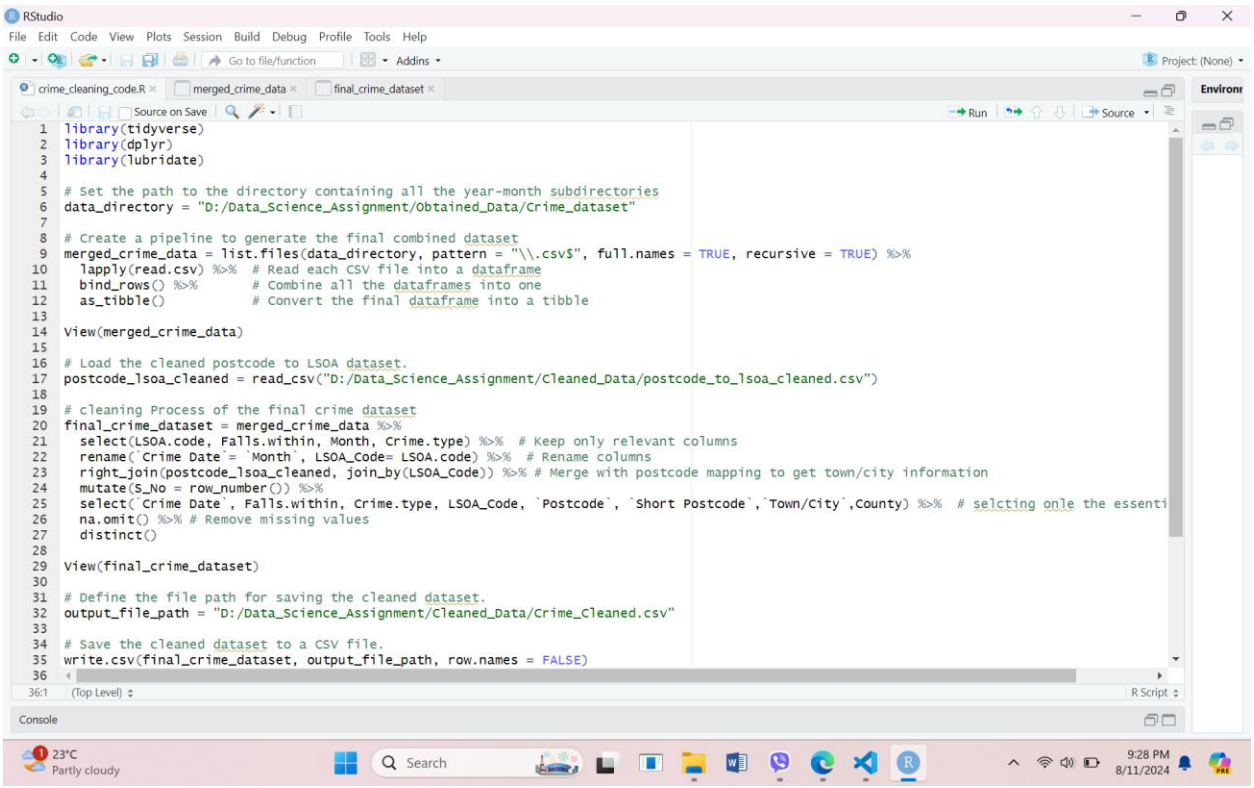
```
# Merge the datasets on the common key "Postcode"
merged_data = merge(coverage_data, performance_data, by = "postcode", all = TRUE)

# View the merged dataset
summary(merged_data)

View(merged_data)
colnames(merged_data)

# Write the cleaned and merged data to a new CSV file
write_csv(merged_data, "D:/Data_Science_Assignment/Cleaned_Data/Merged Broadband Speed/merged_coverage_performance_broadbandSpeed_data.csv")

# Importing postcode to lsoa cleaned csv
postcode_to_lsoa_cleaned_path ="D:/Data_Science_Assignment/Cleaned_Data/postcode_to_lsoa_cleaned.csv"
postcode_to_lsoa_cleaned = read_csv(postcode_to_lsoa_cleaned_path)
View(postcode_to_lsoa_cleaned)

#cleaning the broadband speed
broadband_speed_cleaned = merged_data %>%
  select(`Average download speed (Mbit/s)`, `Maximum download speed (Mbit/s)`, postcode_space) %>%  # Only selecting required columns
  rename(Postcode= `postcode_space`) %>%  # Renaming the postcode_space column to Postcode
  right_join(postcode_to_lsoa_cleaned, by = "Postcode") %>%  # Joining with the postcode_to_lsoa_cleaned by matching Postcode
  select(`Average download speed (Mbit/s)`, `Maximum download speed (Mbit/s)`, Postcode, `Short Postcode`, `Town/City`, District, County) %>%  # Se
  mutate(`Short Postcode`= substr(Postcode, 1,5)) %>% #Fill the missing short code valuess
  na.omit() # Removing rows with null values


View(broadband_speed_cleaned)

broadband_speed_cleaned_path ="D:/Data_Science_Assignment/Cleaned_Data/Broadband_speed_cleaned_data.csv"

write.csv(broadband_speed_cleaned,broadband_speed_cleaned_path, row.names = FALSE)
```

Crime

Using list.files() and bind_rows(), CSV files from several year-month subdirectories were combined into a single dataframe to clean and preprocess the crime dataset. After that, the data was transformed into a tibble for simpler management. For consistency, important columns like LSOA.code, Falls.within, Month, and Crime.type were kept but given new names (Month became Crime Date, for example). To include town/city information, the dataset was combined with postcode data that had been cleansed earlier and added to LSOA. Missing values were eliminated, unnecessary columns were eliminated, and a serial number (S_No) was inserted. For additional study, the cleaned dataset was stored as a CSV file. (Police, n.d.)

Figure 4

Crime Cleaning

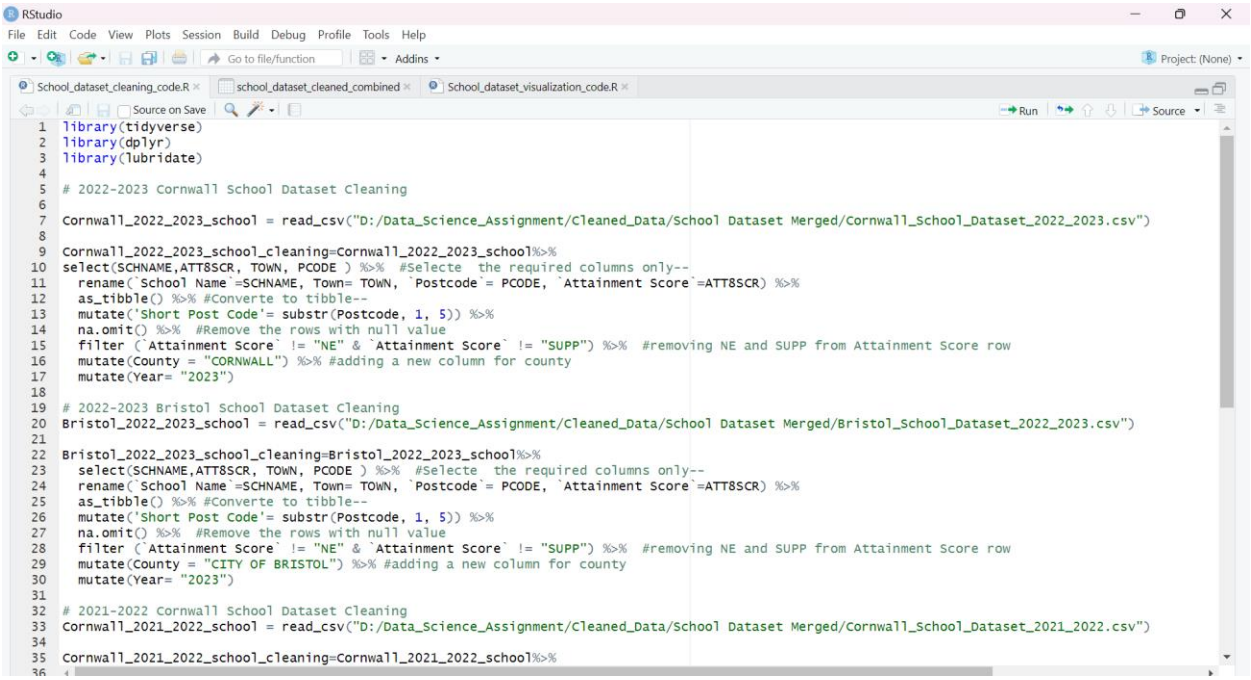


School

The school datasets for 2021–2022 and 2022–2023 were cleaned to ensure consistency between Bristol and Cornwall. Key columns (SCHNAME, ATT8SCR, TOWN, and PCODE) were selected, and column names were standardized. A new column, Short Post Code, was created from the first five characters of the Postcode. Rows with null values or "NE" or "SUPP" in the Attainment Score column were removed. The datasets were then combined into a single tibble, adding County and Year columns for context. The cleaned dataset was saved as a CSV file for further analysis. (School dataset, UK Government Compare School Performance Service, n.d.)

Figure 5

School Cleaning



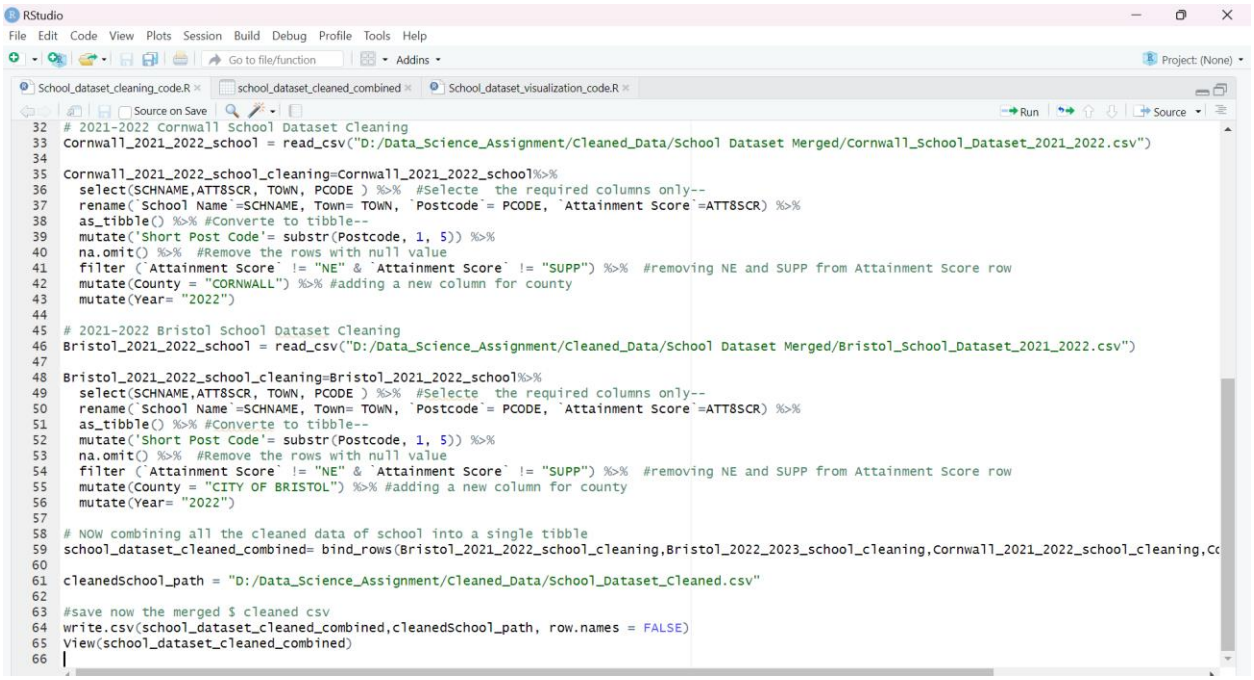
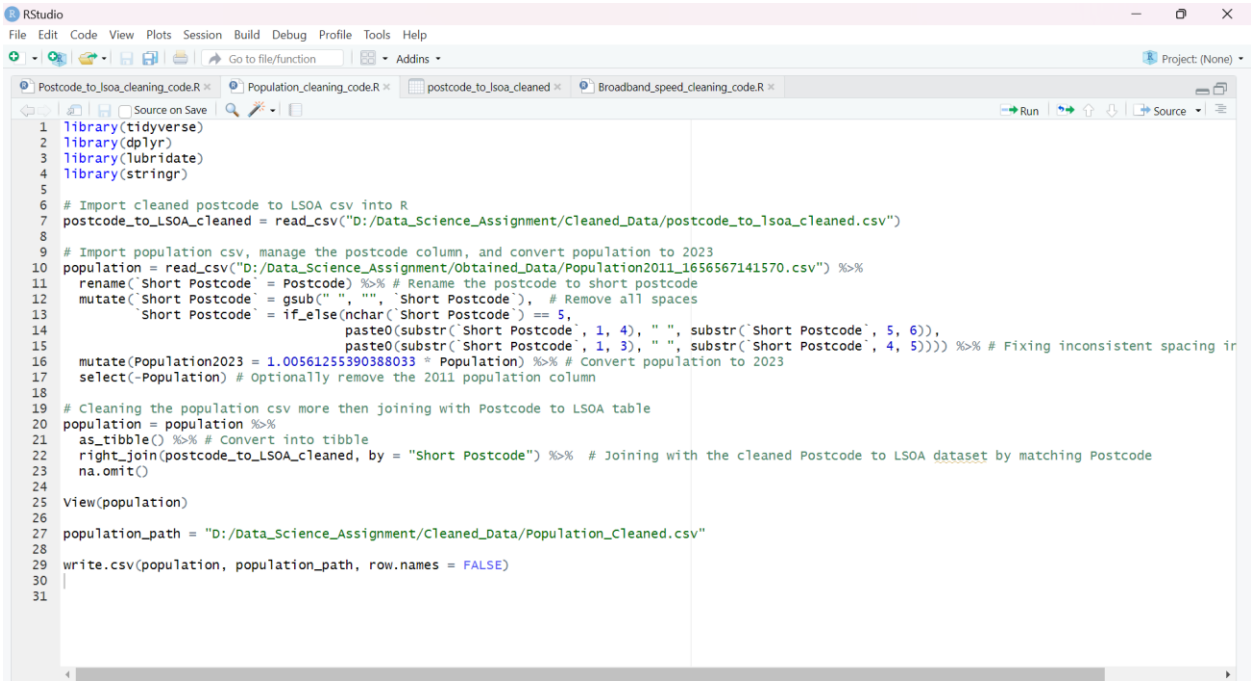


Figure 6

Population Cleaning



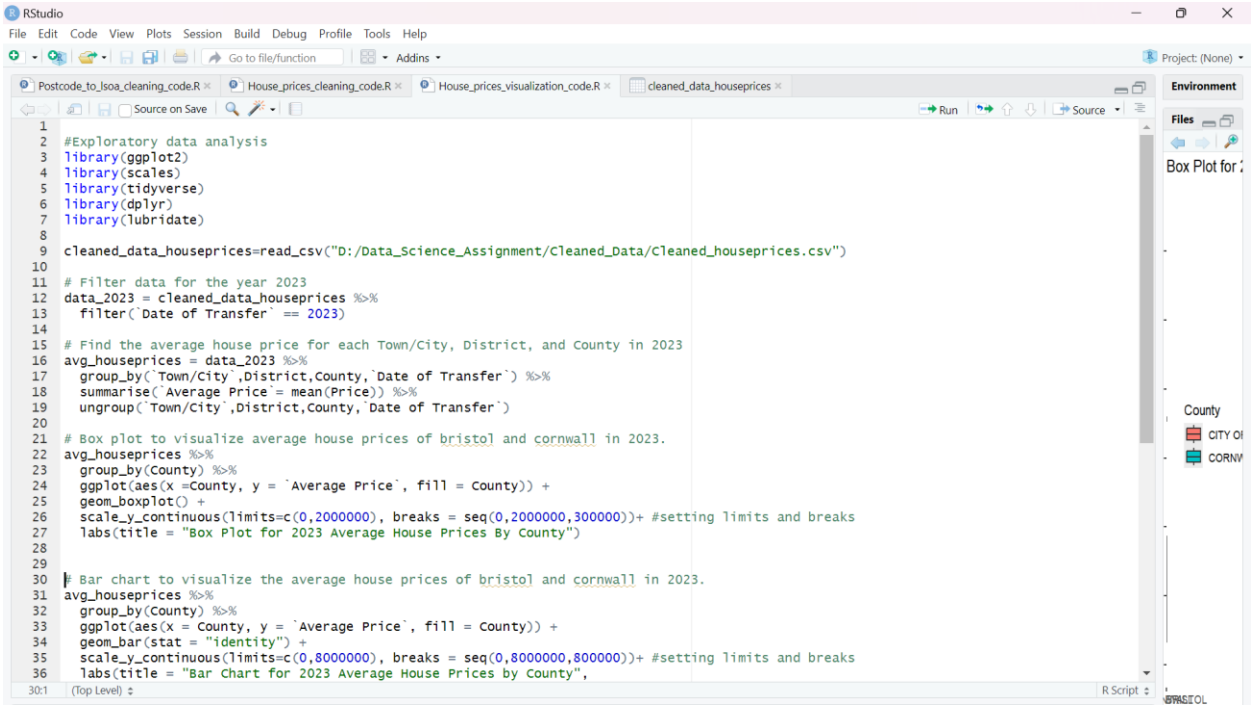
Exploratory Analysis

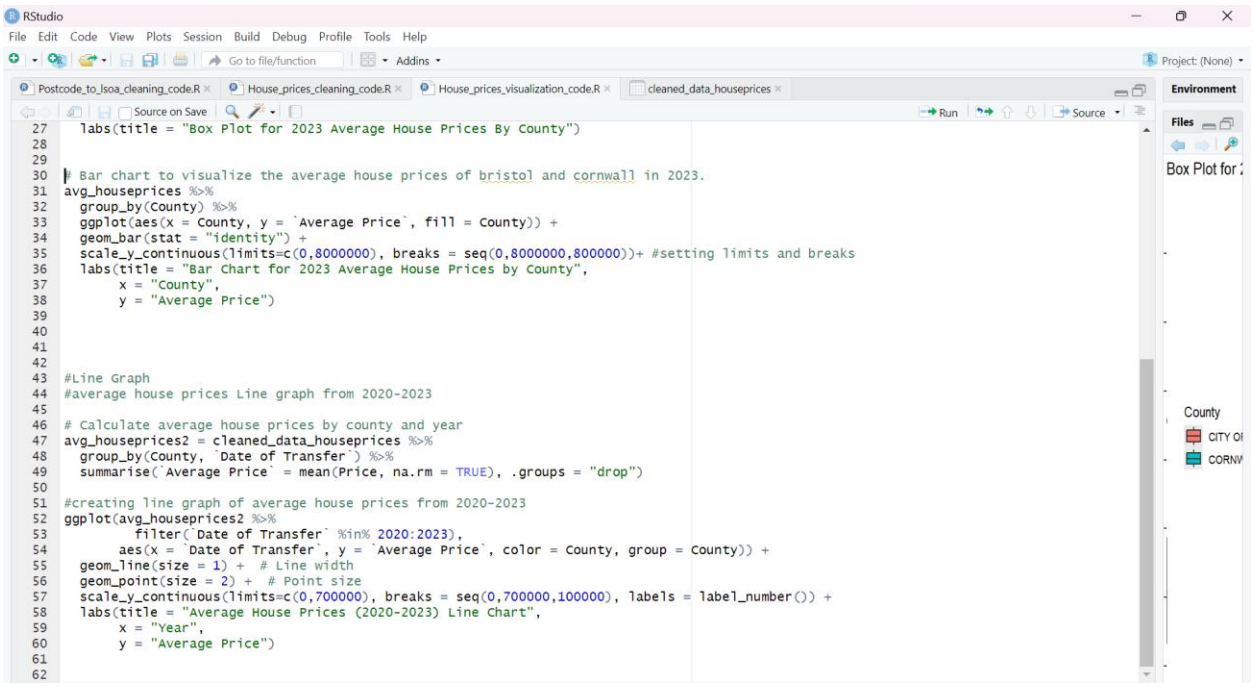
House Prices

In the EDA of the house prices dataset, we used several visualizations and summary statistics. A box plot was created to show the range of average house prices for Bristol and Cornwall in 2023, highlighting typical price ranges and outliers. A bar chart compared average prices between the two counties, and a line graph tracked price trends from 2020-2023. By summarizing prices across towns, districts, and counties, we identified key patterns and trends in the data. (Wickham, n.d.)

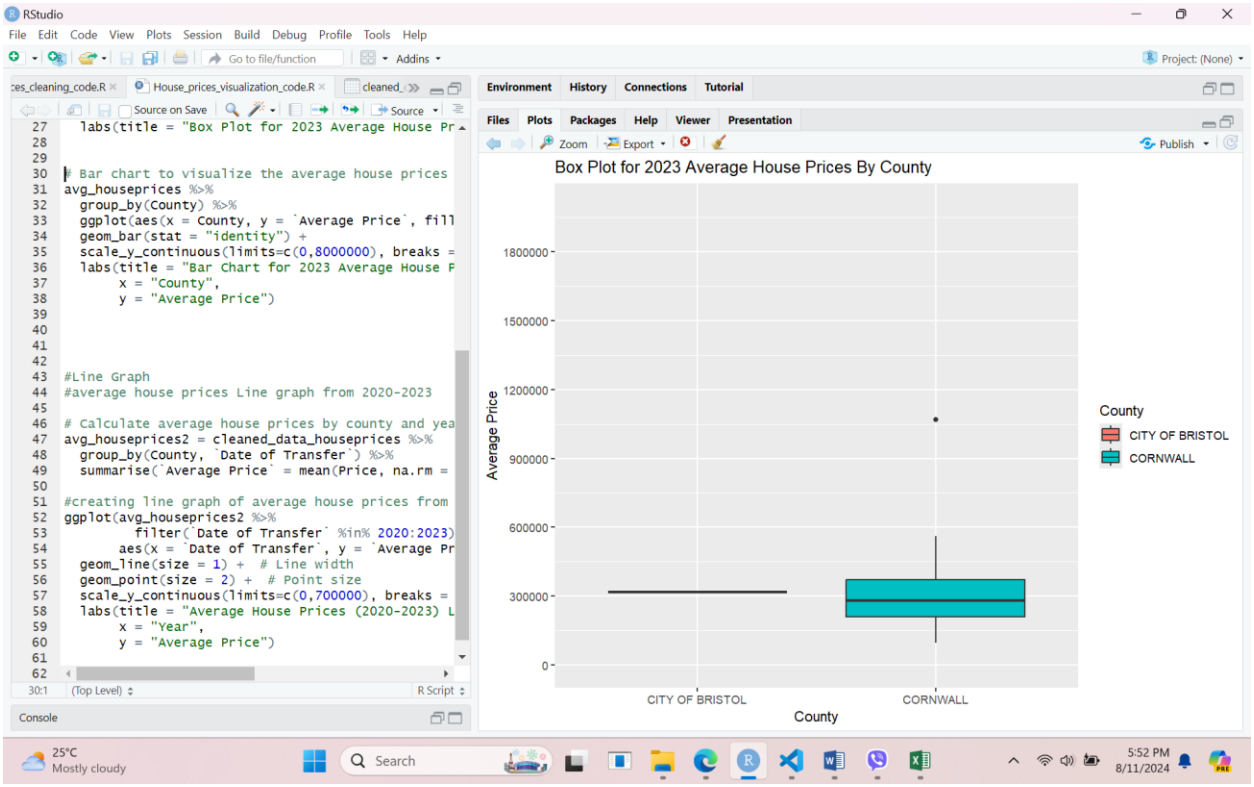
Figure 7

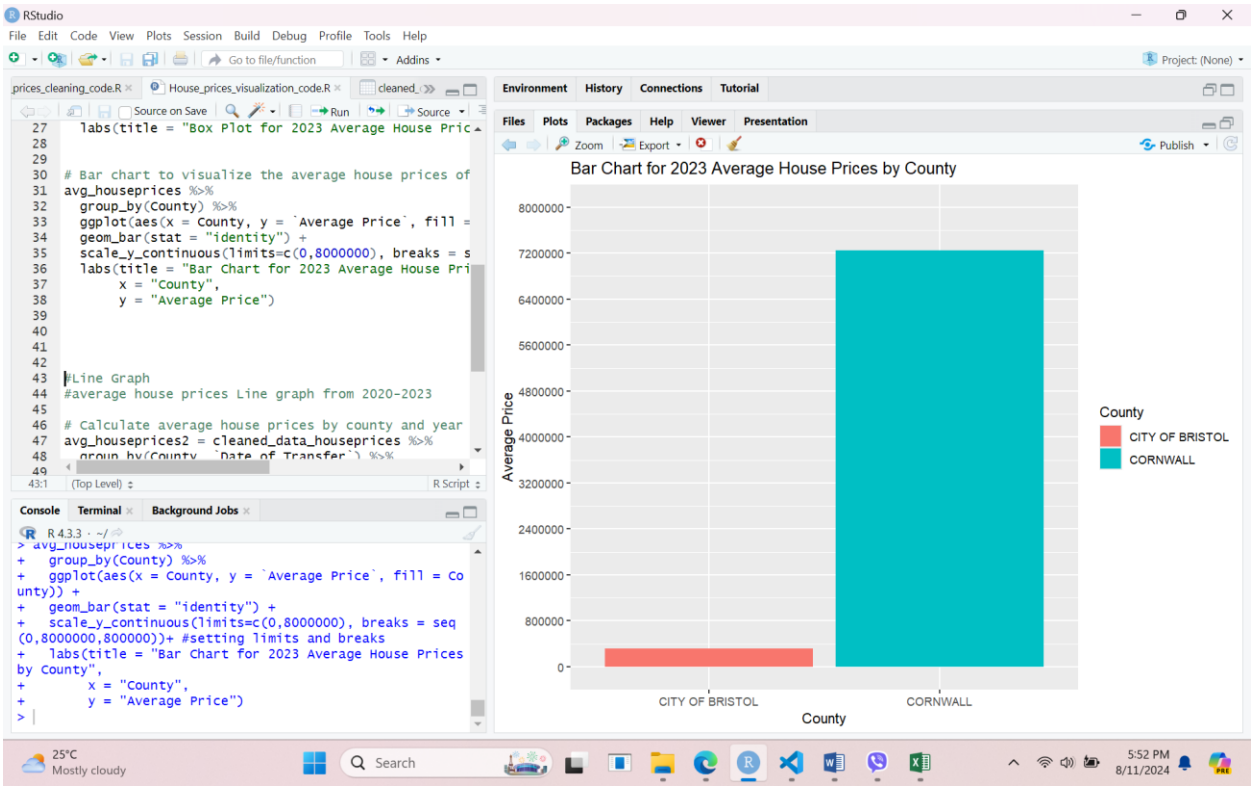
House Price Visualization

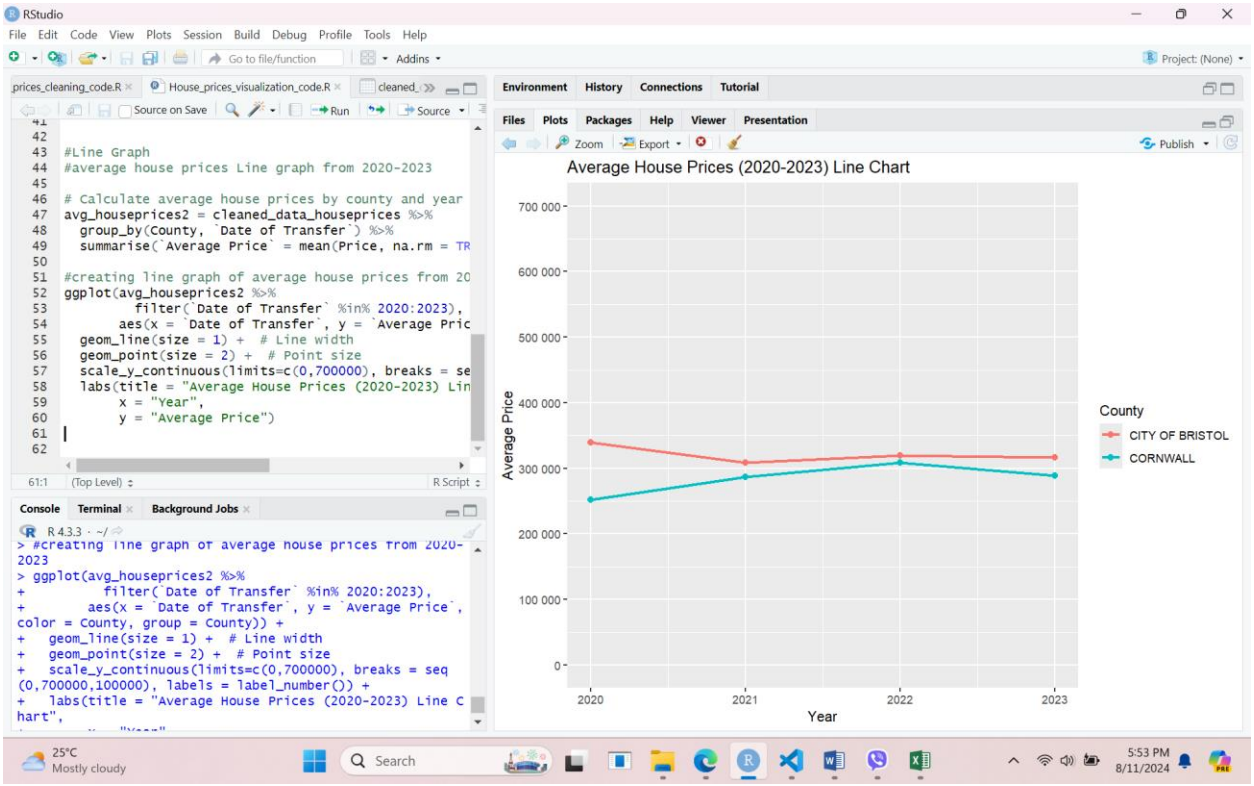




```
  labs(title = "Box Plot for 2023 Average House Prices By County")


# Bar chart to visualize the average house prices of bristol and cornwall in 2023.
avg_houseprices %>%
  group_by(County) %>%
  ggplot(aes(x = County, y = `Average Price`, fill = County)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(limits=c(0,8000000), breaks = seq(0,8000000,800000))+ #setting limits and breaks
  labs(title = "Bar Chart for 2023 Average House Prices by County",
       x = "County",
       y = "Average Price")



#Line Graph
#average house prices Line graph from 2020-2023

# Calculate average house prices by county and year
avg_houseprices2 = cleaned_data_houseprices %>%
  group_by(County, `Date of Transfer`) %>%
  summarise(`Average Price` = mean(Price, na.rm = TRUE), .groups = "drop")

#creating line graph of average house prices from 2020-2023
ggplot(avg_houseprices2 %>%
         filter(`Date of Transfer` %in% 2020:2023),
       aes(x = `Date of Transfer`, y = `Average Price`, color = County, group = County)) +
  geom_line(size = 1) +  # Line width
  geom_point(size = 2) +  # Point size
  scale_y_continuous(limits=c(0,700000), breaks = seq(0,700000,100000), labels = label_number()) +
  labs(title = "Average House Prices (2020-2023) Line Chart",
       x = "Year",
       y = "Average Price")
```




RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Bar Chart for 2023 Average House Prices by County
Average Price
8000000
7200000
6400000
5600000
4800000
4000000
3200000
2400000
1600000
800000
0
CITY OF BRISTOL
CORNWALL
County
County
CITY OF BRISTOL
CORNWALL
25°C
Mostly cloudy
Search
5:52 PM
8/11/2024

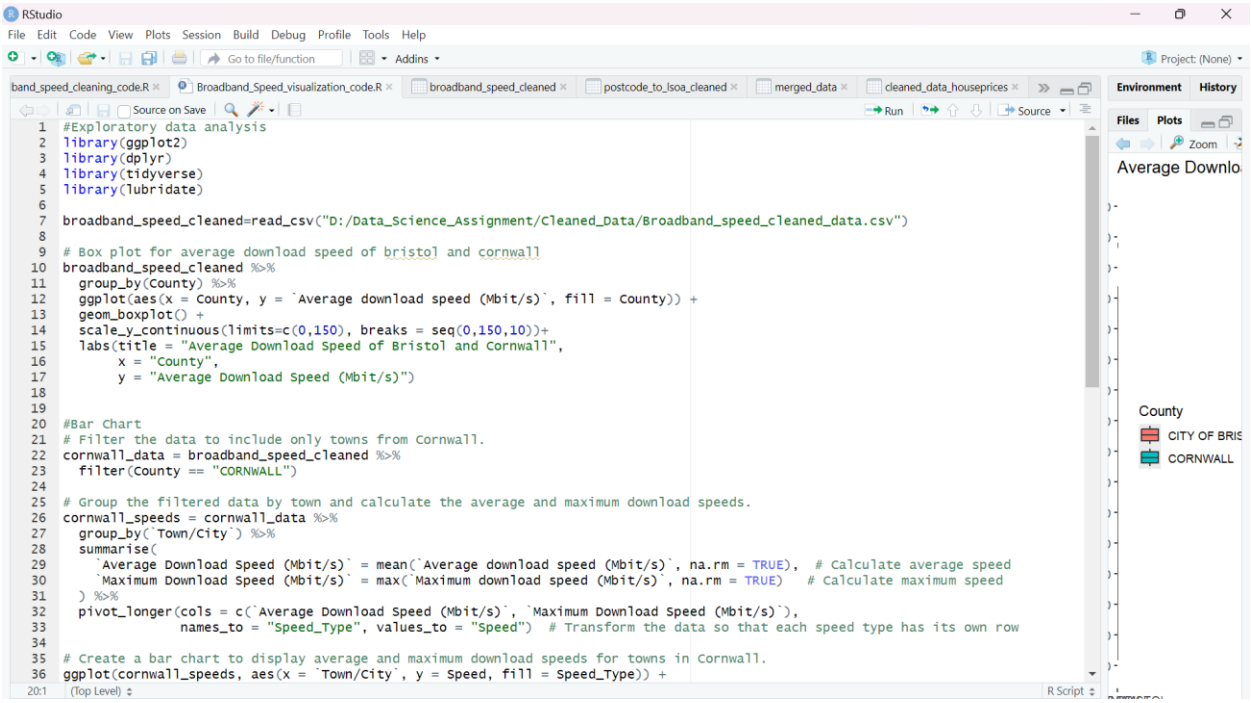

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Average House Prices (2020-2023) Line Chart
Average Price
700 000
600 000
500 000
400 000
300 000
200 000
100 000
0
2020
2021
2022
2023
Year
County
CITY OF BRISTOL
CORNWALL
25°C
Mostly cloudy
Search
5:53 PM
8/11/2024

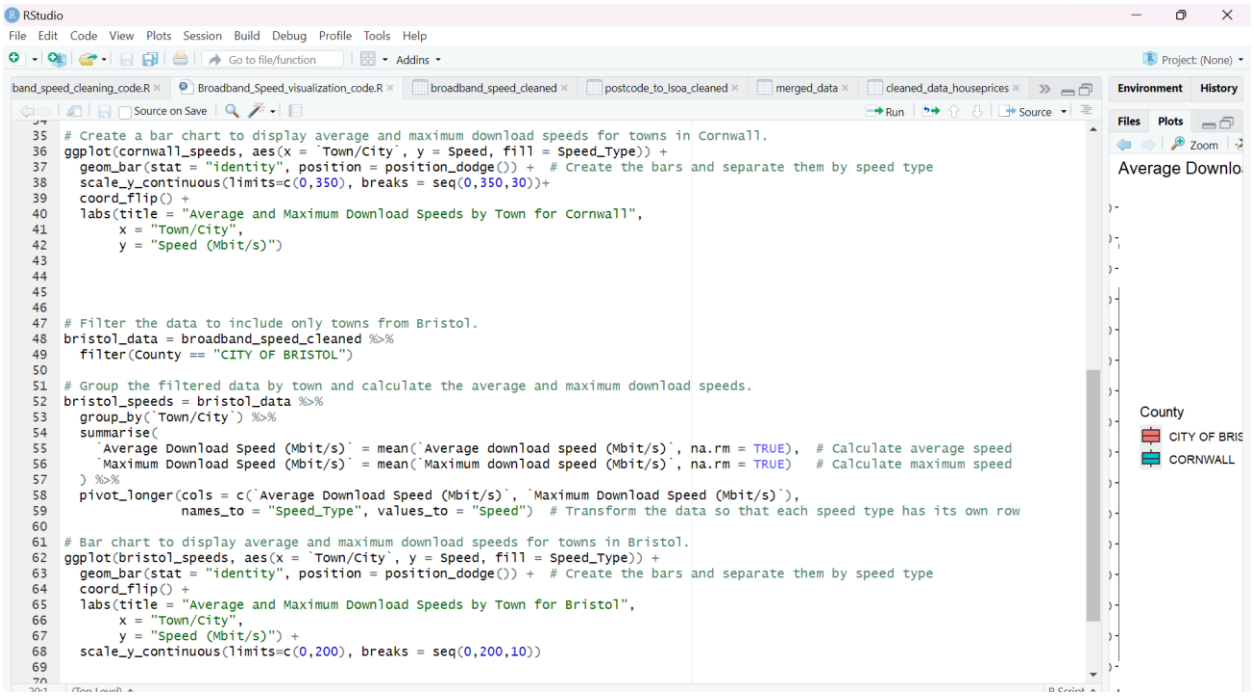
Broadband Speeds

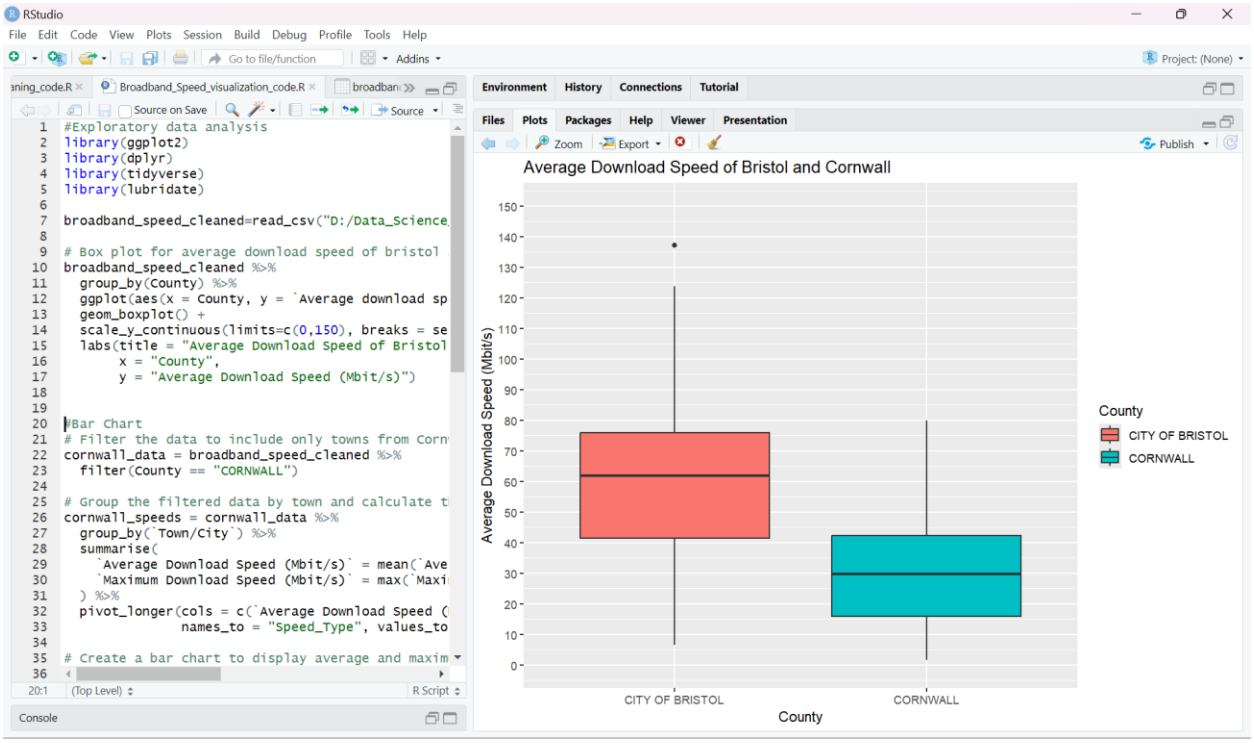
In the exploratory data analysis of broadband speeds for Bristol and Cornwall, a box plot was created to compare average download speeds, highlighting variations and outliers between the two counties. Bar charts were also made to compare average and maximum download speeds for towns within each county, revealing which towns have the best and worst broadband performance. These visualizations provided insights into broadband connectivity across Bristol and Cornwall.

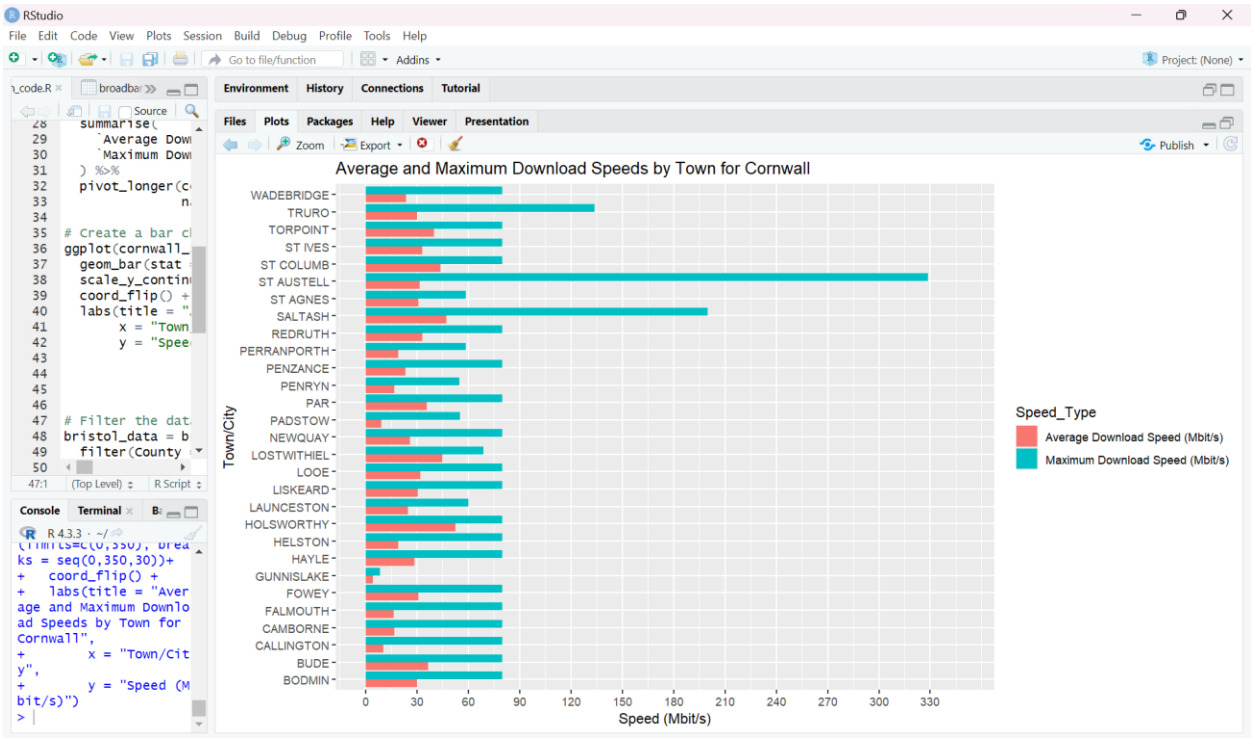
Figure 8

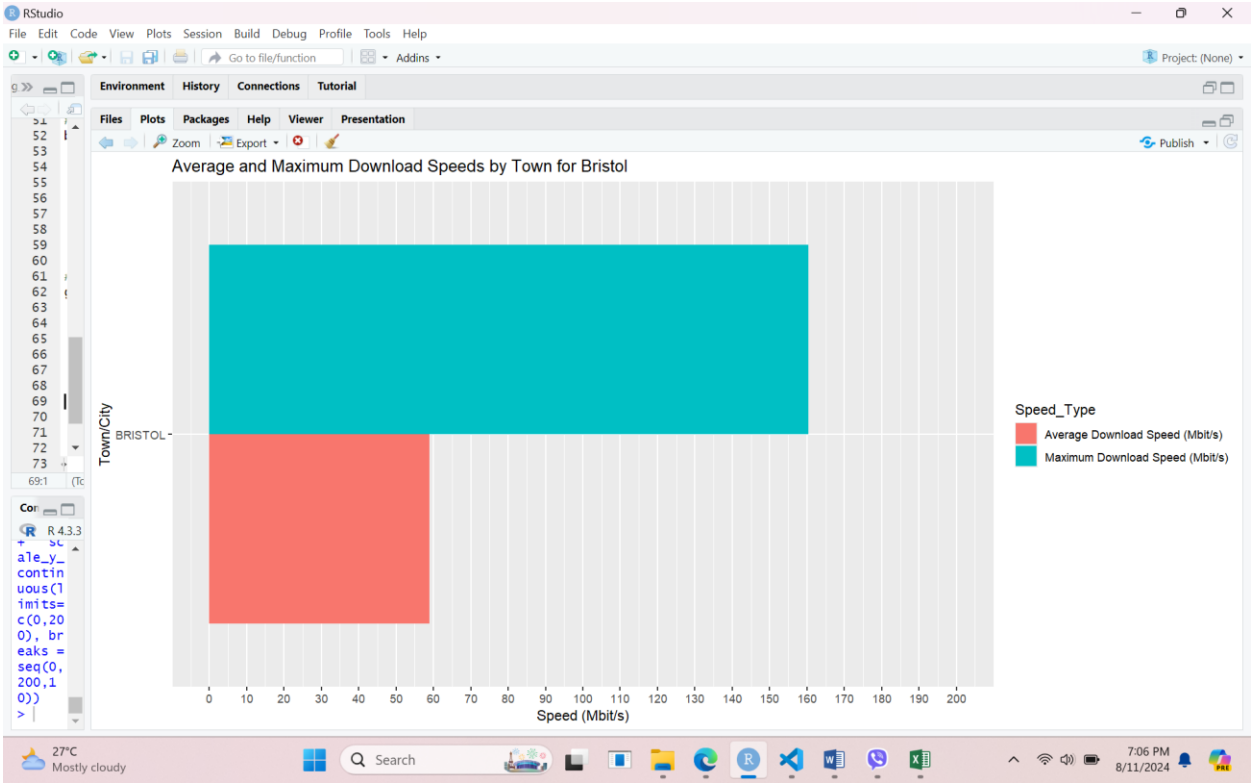
Broadband Speed Visualization




RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
band_speed_cleaning_code.R
Broadband_Speed_visualization_code.R
broadband_speed_cleaned
postcode_to_lsoa_cleaned
merged_data
cleaned_data_houseprices
Environment
History
Files
Plots
Average Downlo
County
CITY OF BRIS
CORNWALL
Create a bar chart to display average and maximum download speeds for towns in Cornwall.
ggplot(cornwall_speeds, aes(x = `Town/City`, y = Speed, fill = Speed_Type)) +
geom_bar(stat = "identity", position = position_dodge()) + # Create the bars and separate them by speed type
scale_y_continuous(limits=c(0,350), breaks = seq(0,350,30))+
coord_flip() +
labs(title = "Average and Maximum Download Speeds by Town for Cornwall",
x = "Town/City",
y = "Speed (Mbit/s)")
# Filter the data to include only towns from Bristol.
bristol_data = broadband_speed_cleaned %>%
filter(County == "CITY OF BRISTOL")
# Group the filtered data by town and calculate the average and maximum download speeds.
bristol_speeds = bristol_data %>%
group_by(`Town/City`) %>%
summarise(
`Average Download Speed (Mbit/s)` = mean(`Average download speed (Mbit/s)`, na.rm = TRUE), # Calculate average speed
`Maximum Download Speed (Mbit/s)` = mean(`Maximum download speed (Mbit/s)`, na.rm = TRUE) # Calculate maximum speed
) %>%
pivot_longer(cols = c(`Average Download Speed (Mbit/s)`, `Maximum Download Speed (Mbit/s)`),
names_to = "Speed_Type", values_to = "Speed") # Transform the data so that each speed type has its own row
# Bar chart to display average and maximum download speeds for towns in Bristol.
ggplot(bristol_speeds, aes(x = `Town/City`, y = Speed, fill = Speed_Type)) +
geom_bar(stat = "identity", position = position_dodge()) + # Create the bars and separate them by speed type
coord_flip() +
labs(title = "Average and Maximum Download Speeds by Town for Bristol",
x = "Town/City",
y = "Speed (Mbit/s)") +
scale_y_continuous(limits=c(0,200), breaks = seq(0,200,10))


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Broadband_Speed_visualization_code.R
Environment
History
Connections
Tutorial
Files
Plots
Packages
Help
Viewer
Presentation
Average Download Speed of Bristol and Cornwall
Average Download Speed (Mbit/s)
County
CITY OF BRISTOL
CORNWALL
#Exploratory data analysis
library(ggplot2)
library(dplyr)
library(tidyverse)
library(lubridate)
broadband_speed_cleaned=read_csv("D:/Data_Science
# Box plot for average download speed of bristol
broadband_speed_cleaned %>%
group_by(County) %>%
ggplot(aes(x = County, y = `Average download sp
geom_boxplot() +
scale_y_continuous(limits=c(0,150), breaks = se
labs(title = "Average Download Speed of Bristol
x = "County",
y = "Average Download Speed (Mbit/s)")
#Bar Chart
# Filter the data to include only towns from Corn
cornwall_data = broadband_speed_cleaned %>%
filter(County == "CORNWALL")
# Group the filtered data by town and calculate t
cornwall_speeds = cornwall_data %>%
group_by(`Town/City`) %>%
summarise(
`Average Download Speed (Mbit/s)` = mean(`Ave
`Maximum Download Speed (Mbit/s)` = max(`Maxi
) %>%
pivot_longer(cols = c(`Average Download Speed (
names_to = "Speed_Type", values_to
# Create a bar chart to display average and maxim
Console


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Environment History Connections Tutorial
Files Plots Packages Help Viewer Presentation
Average and Maximum Download Speeds by Town for Cornwall
WADEBRIDGE
TRURO
TORPOINT
ST IVES
ST COLUMB
ST AUSTELL
ST AGNES
SALTASH
REDRUTH
PERRANPORTH
PENZANCE
PENRYN
PAR
PADSTOW
NEWQUAY
LOSTWITHIEL
LOOE
LISKEARD
LAUNCESTON
HOLSWORTHY
HELSTON
HAYLE
GUNNISLAKE
FOWEY
FALMOUTH
CAMBORNE
CALLINGTON
BUDE
BODMIN
Town/City
0 30 60 90 120 150 180 210 240 270 300 330
Speed (Mbit/s)
Speed_Type
Average Download Speed (Mbit/s)
Maximum Download Speed (Mbit/s)

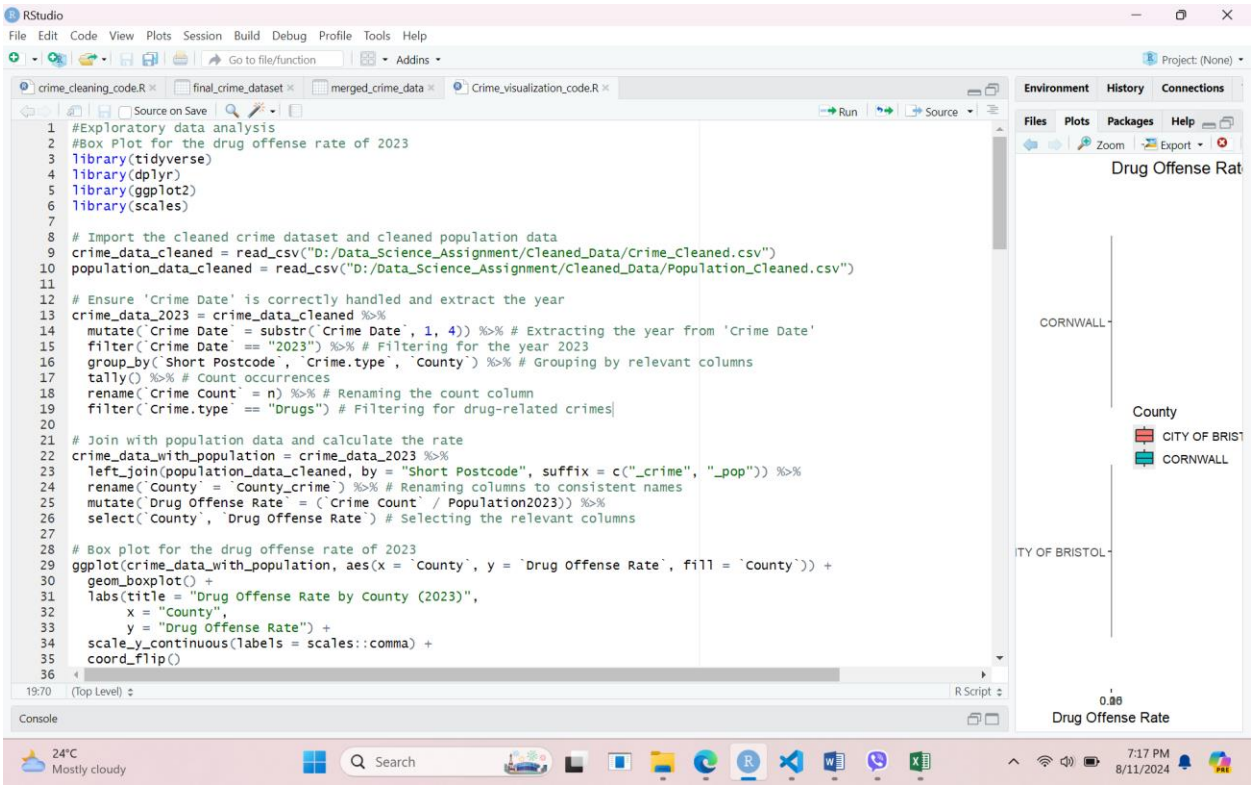

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Environment History Connections Tutorial
Files Plots Packages Help Viewer Presentation
Average and Maximum Download Speeds by Town for Bristol
BRISTOL
Town/City
0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180 190 200
Speed (Mbit/s)
Speed_Type
Average Download Speed (Mbit/s)
Maximum Download Speed (Mbit/s)
27°C
Mostly cloudy
Search
7:06 PM
8/11/2024

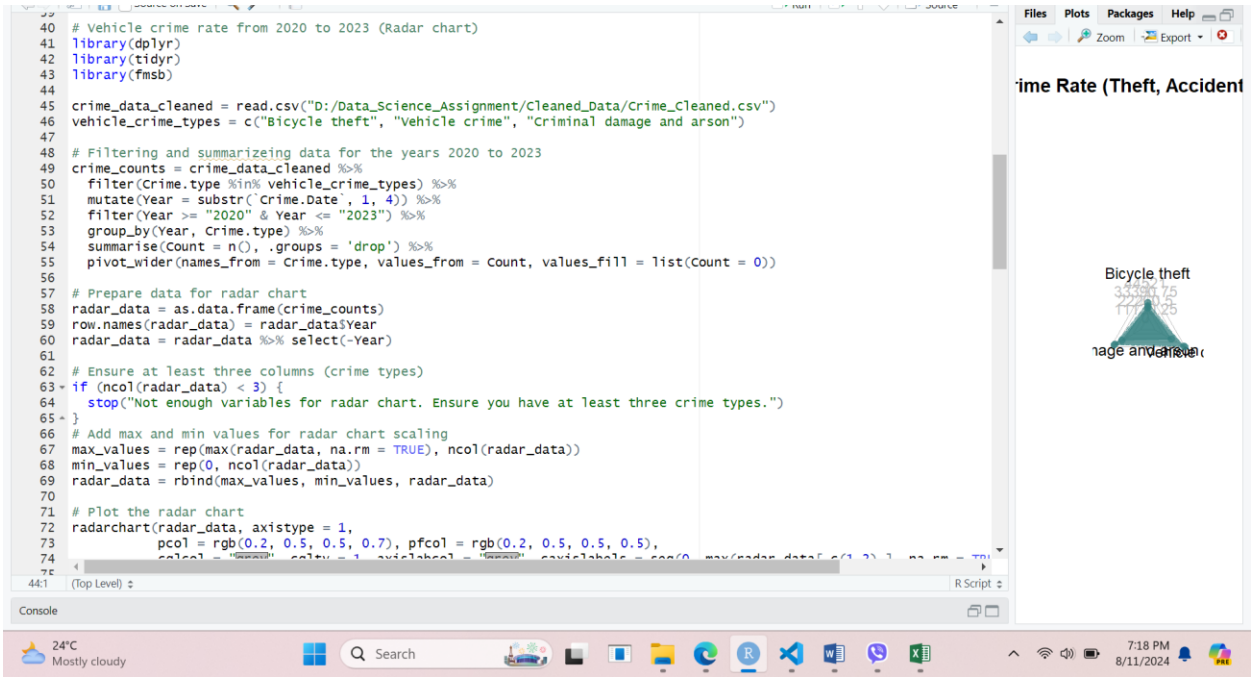
Crime Rate

The analysis first focused on drug offenses in 2023, using a box plot to compare drug offense rates by county, revealing variations and outliers. A radar chart was then used to analyze vehicle crime data from 2020 to 2023, showing different crime types over time. A pie chart illustrated robbery rates by county for a specific month in 2023. Finally, a line chart was plotted to track trends in drug crime rates over time.

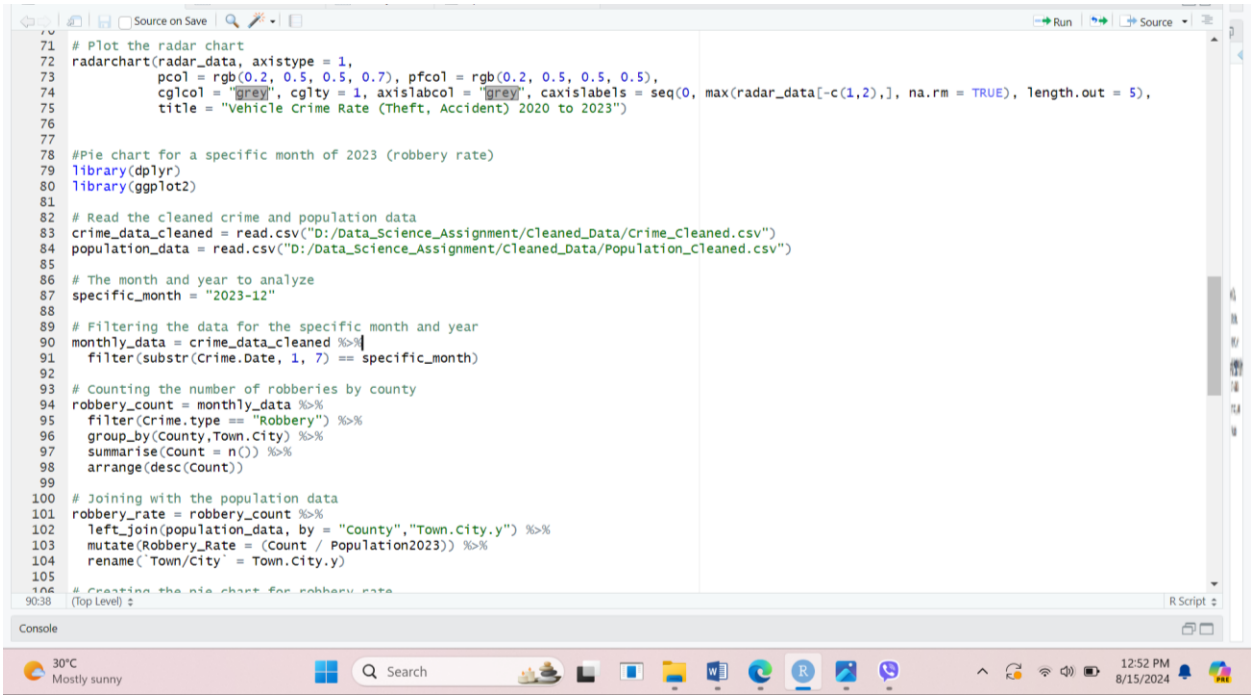
Figure 9

Crime Visualization

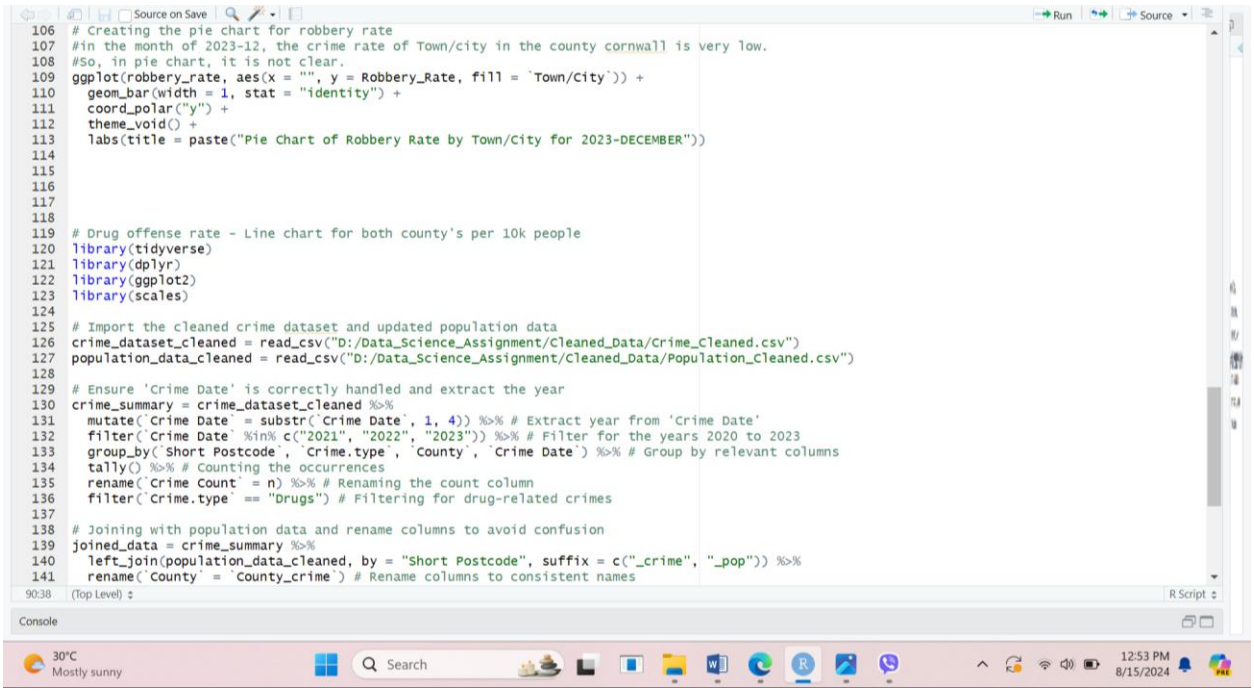




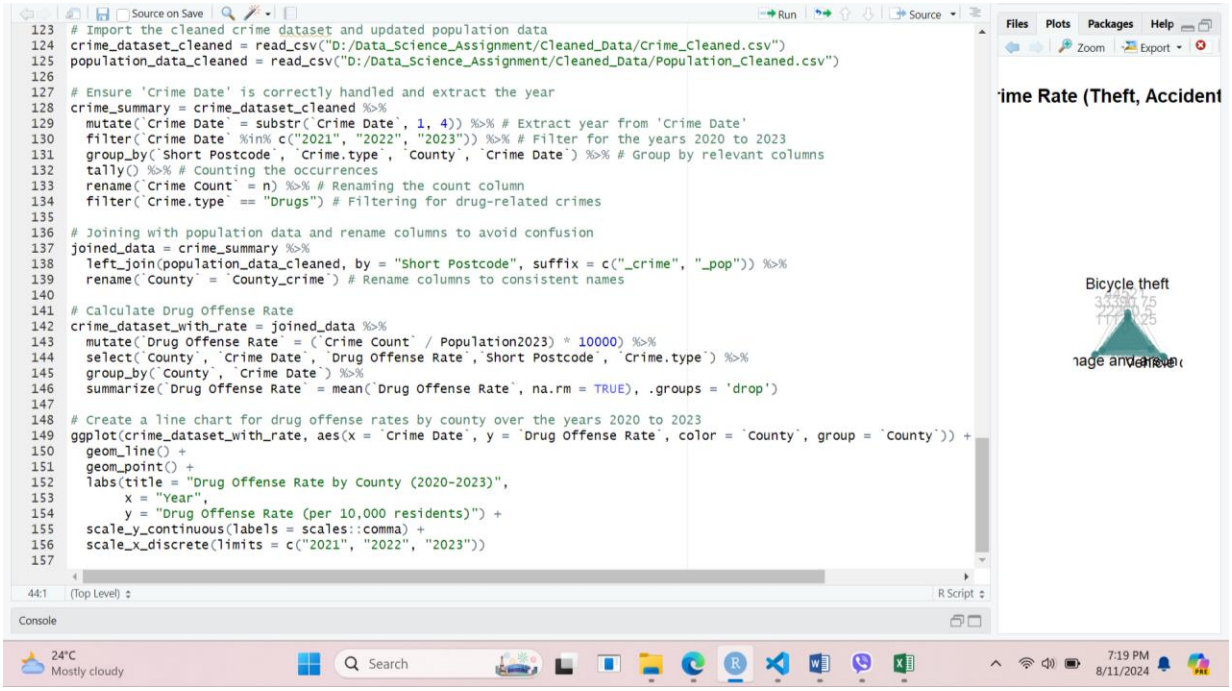
```r
# Vehicle crime rate from 2020 to 2023 (Radar chart)
library(dplyr)
library(tidyr)
library(fmsb)

crime_data_cleaned = read.csv("D:/Data_Science_Assignment/Cleaned_Data/Crime_Cleaned.csv")
vehicle_crime_types = c("Bicycle theft", "Vehicle crime", "Criminal damage and arson")

# Filtering and summarizeing data for the years 2020 to 2023
crime_counts = crime_data_cleaned %>%
  filter(Crime.type %in% vehicle_crime_types) %>%
  mutate(Year = substr(`Crime.Date`, 1, 4)) %>%
  filter(Year >= "2020" & Year <= "2023") %>%
  group_by(Year, Crime.type) %>%
  summarise(Count = n(), .groups = 'drop') %>%
  pivot_wider(names_from = Crime.type, values_from = Count, values_fill = list(Count = 0))

# Prepare data for radar chart
radar_data = as.data.frame(crime_counts)
row.names(radar_data) = radar_data$Year
radar_data = radar_data %>% select(-Year)

# Ensure at least three columns (crime types)
if (ncol(radar_data) < 3) {
  stop("Not enough variables for radar chart. Ensure you have at least three crime types.")
}
# Add max and min values for radar chart scaling
max_values = rep(max(radar_data, na.rm = TRUE), ncol(radar_data))
min_values = rep(0, ncol(radar_data))
radar_data = rbind(max_values, min_values, radar_data)

# Plot the radar chart
radarchart(radar_data, axistype = 1,
           pcol = rgb(0.2, 0.5, 0.5, 0.7), pfcol = rgb(0.2, 0.5, 0.5, 0.5),
```

```r
# Plot the radar chart
radarchart(radar_data, axistype = 1,
           pcol = rgb(0.2, 0.5, 0.5, 0.7), pfcol = rgb(0.2, 0.5, 0.5, 0.5),
           cglcol = "grey", cglty = 1, axislabcol = "grey", caxislabels = seq(0, max(radar_data[-c(1,2),], na.rm = TRUE), length.out = 5),
           title = "Vehicle Crime Rate (Theft, Accident) 2020 to 2023")


#Pie chart for a specific month of 2023 (robbery rate)
library(dplyr)
library(ggplot2)

# Read the cleaned crime and population data
crime_data_cleaned = read.csv("D:/Data_Science_Assignment/Cleaned_Data/Crime_Cleaned.csv")
population_data = read.csv("D:/Data_Science_Assignment/Cleaned_Data/Population_Cleaned.csv")

# The month and year to analyze
specific_month = "2023-12"

# Filtering the data for the specific month and year
monthly_data = crime_data_cleaned %>%
  filter(substr(Crime.Date, 1, 7) == specific_month)

# Counting the number of robberies by county
robbery_count = monthly_data %>%
  filter(Crime.type == "Robbery") %>%
  group_by(County,Town.City) %>%
  summarise(Count = n()) %>%
  arrange(desc(Count))

# Joining with the population data
robbery_rate = robbery_count %>%
  left_join(population_data, by = "County","Town.City.y") %>%
  mutate(Robbery_Rate = (Count / Population2023)) %>%
  rename(`Town/City` = Town.City.y)

# Creating the pie chart for robbery rate
```

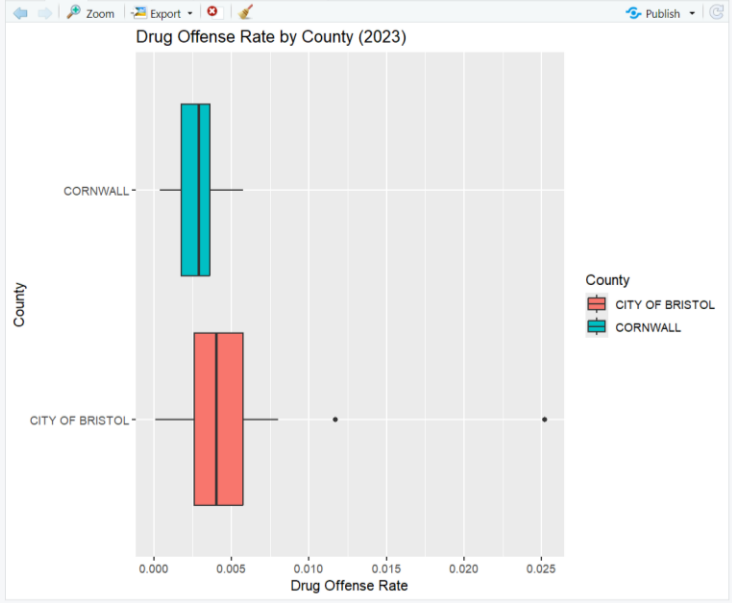


```r
# Creating the pie chart for robbery rate
#in the month of 2023-12, the crime rate of Town/city in the county cornwall is very low.
#So, in pie chart, it is not clear.
ggplot(robbery_rate, aes(x = "", y = Robbery_Rate, fill = `Town/City`)) +
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y") +
  theme_void() +
  labs(title = paste("Pie Chart of Robbery Rate by Town/City for 2023-DECEMBER"))



# Drug offense rate - Line chart for both county's per 10k people
library(tidyverse)
library(dplyr)
library(ggplot2)
library(scales)

# Import the cleaned crime dataset and updated population data
crime_dataset_cleaned = read_csv("D:/Data_Science_Assignment/Cleaned_Data/Crime_Cleaned.csv")
population_data_cleaned = read_csv("D:/Data_Science_Assignment/Cleaned_Data/Population_Cleaned.csv")

# Ensure 'Crime Date' is correctly handled and extract the year
crime_summary = crime_dataset_cleaned %>%
  mutate(`Crime Date` = substr(`Crime Date`, 1, 4)) %>% # Extract year from 'Crime Date'
  filter(`Crime Date` %in% c("2021", "2022", "2023")) %>% # Filter for the years 2020 to 2023
  group_by(`Short Postcode`, `Crime.type`, `County`, `Crime Date`) %>% # Group by relevant columns
  tally() %>% # Counting the occurrences
  rename(`Crime Count` = n) %>% # Renaming the count column
  filter(`Crime.type` == "Drugs") # Filtering for drug-related crimes

# Joining with population data and rename columns to avoid confusion
joined_data = crime_summary %>%
  left_join(population_data_cleaned, by = "Short Postcode", suffix = c("_crime", "_pop")) %>%
  rename(`County` = `County_crime`) # Rename columns to consistent names
```



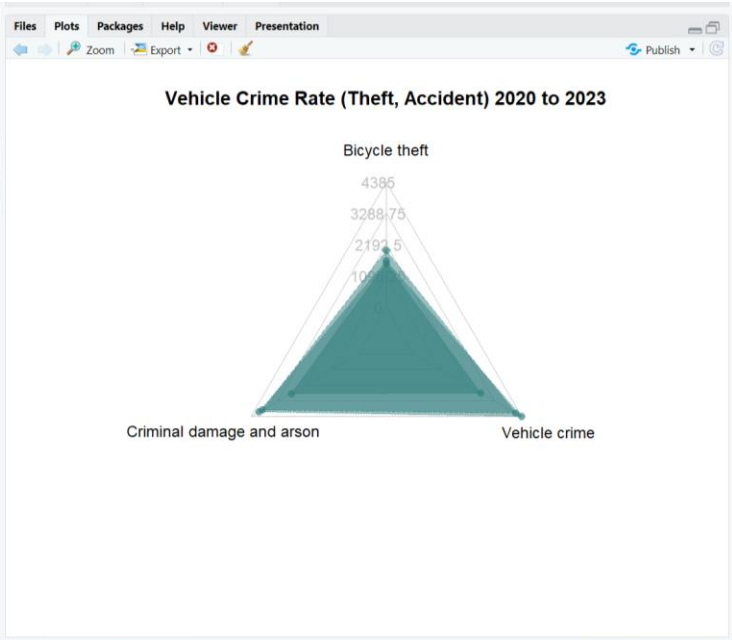
```r
# Import the cleaned crime dataset and updated population data
crime_dataset_cleaned = read_csv("D:/Data_Science_Assignment/Cleaned_Data/Crime_Cleaned.csv")
population_data_cleaned = read_csv("D:/Data_Science_Assignment/Cleaned_Data/Population_Cleaned.csv")

# Ensure 'Crime Date' is correctly handled and extract the year
crime_summary = crime_dataset_cleaned %>%
  mutate(`Crime Date` = substr(`Crime Date`, 1, 4)) %>% # Extract year from 'Crime Date'
  filter(`Crime Date` %in% c("2021", "2022", "2023")) %>% # Filter for the years 2020 to 2023
  group_by(`Short Postcode`, `Crime.type`, `County`, `Crime Date`) %>% # Group by relevant columns
  tally() %>% # Counting the occurrences
  rename(`Crime Count` = n) %>% # Renaming the count column
  filter(`Crime.type` == "Drugs") # Filtering for drug-related crimes

# Joining with population data and rename columns to avoid confusion
joined_data = crime_summary %>%
  left_join(population_data_cleaned, by = "Short Postcode", suffix = c("_crime", "_pop")) %>%
  rename(`County` = `County_crime`) # Rename columns to consistent names

# Calculate Drug Offense Rate
crime_dataset_with_rate = joined_data %>%
  mutate(`Drug Offense Rate` = (`Crime Count` / Population2023) * 10000) %>%
  select(`County`, `Crime Date`, `Drug Offense Rate`, `Short Postcode`, `Crime.type`) %>%
  group_by(`County`, `Crime Date`) %>%
  summarize(`Drug Offense Rate` = mean(`Drug Offense Rate`, na.rm = TRUE), .groups = 'drop')

# Create a line chart for drug offense rates by county over the years 2020 to 2023
ggplot(crime_dataset_with_rate, aes(x = `Crime Date`, y = `Drug Offense Rate`, color = `County`, group = `County`)) +
  geom_line() +
  geom_point() +
  labs(title = "Drug Offense Rate by County (2020-2023)",
       x = "Year",
       y = "Drug Offense Rate (per 10,000 residents)") +
  scale_y_continuous(labels = scales::comma) +
  scale_x_discrete(limits = c("2021", "2022", "2023"))
```

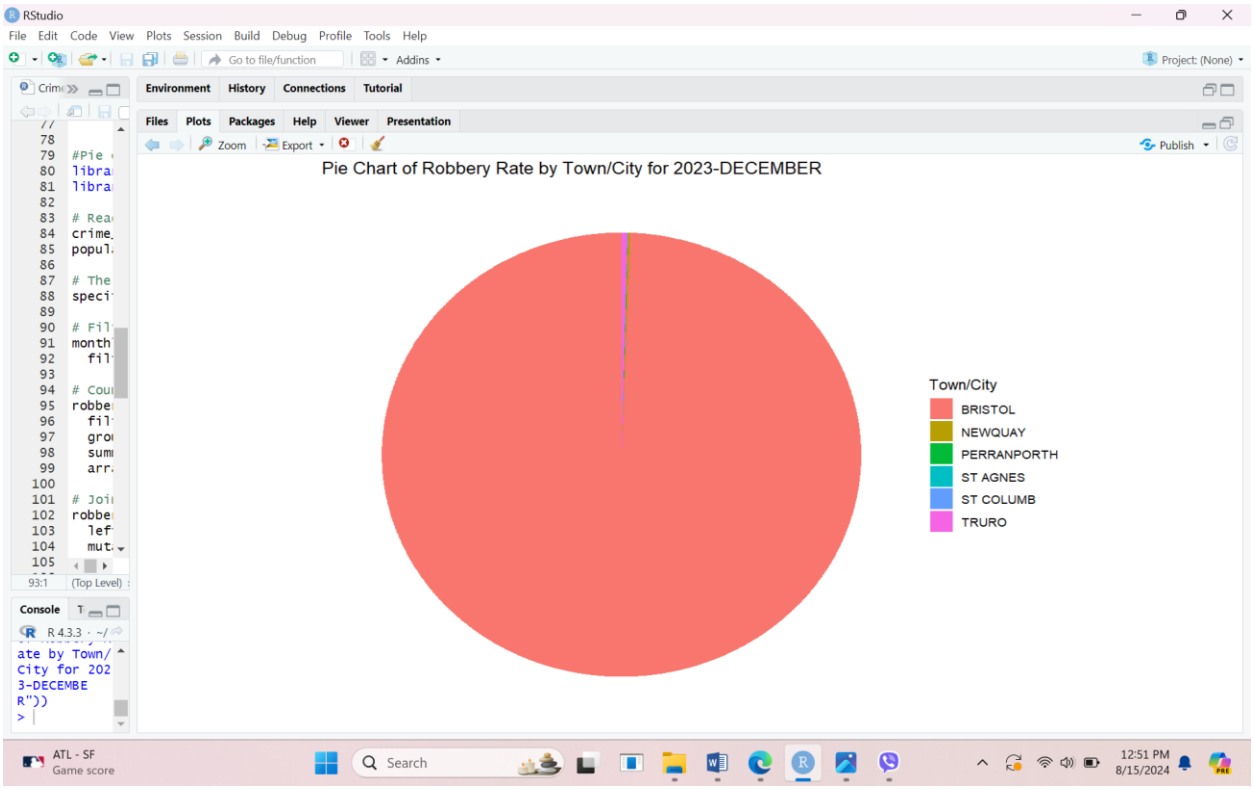
```r
#Exploratory data analysis
#Box Plot for the drug offense rate of 2023
library(tidyverse)
library(dplyr)
library(ggplot2)
library(scales)

# Import the cleaned crime dataset and cleaned populat
crime_data_cleaned = read_csv("D:/Data_Science_Assignm
population_data_cleaned = read_csv("D:/Data_Science_As

# Ensure 'Crime Date' is correctly handled and extract
crime_data_2023 = crime_data_cleaned %>%
  mutate(`Crime Date` = substr(`Crime Date`, 1, 4)) %>
  filter(`Crime Date` == "2023") %>% # Filtering for t
  group_by(`Short Postcode`, `Crime.type`, `County`) %
  tally() %>% # Count occurrences
  rename(`Crime Count` = n) %>% # Renaming the count c
  filter(`Crime.type` == "Drugs") # Filtering for drug

# Join with population data and calculate the rate
crime_data_with_population = crime_data_2023 %>%
  left_join(population_data_cleaned, by = "Short Postc
  rename(`County` = `County_crime`) %>% # Renaming col
  mutate(`Drug Offense Rate` = (`Crime Count` / Popula
  select(`County`, `Drug Offense Rate`) # Selecting th

# Box plot for the drug offense rate of 2023
ggplot(crime_data_with_population, aes(x = `County`, y
  geom_boxplot() +
  labs(title = "Drug Offense Rate by County (2023)",
       x = "County",
       y = "Drug Offense Rate") +
  scale_y_continuous(labels = scales::comma) +
  coord_flip()
```

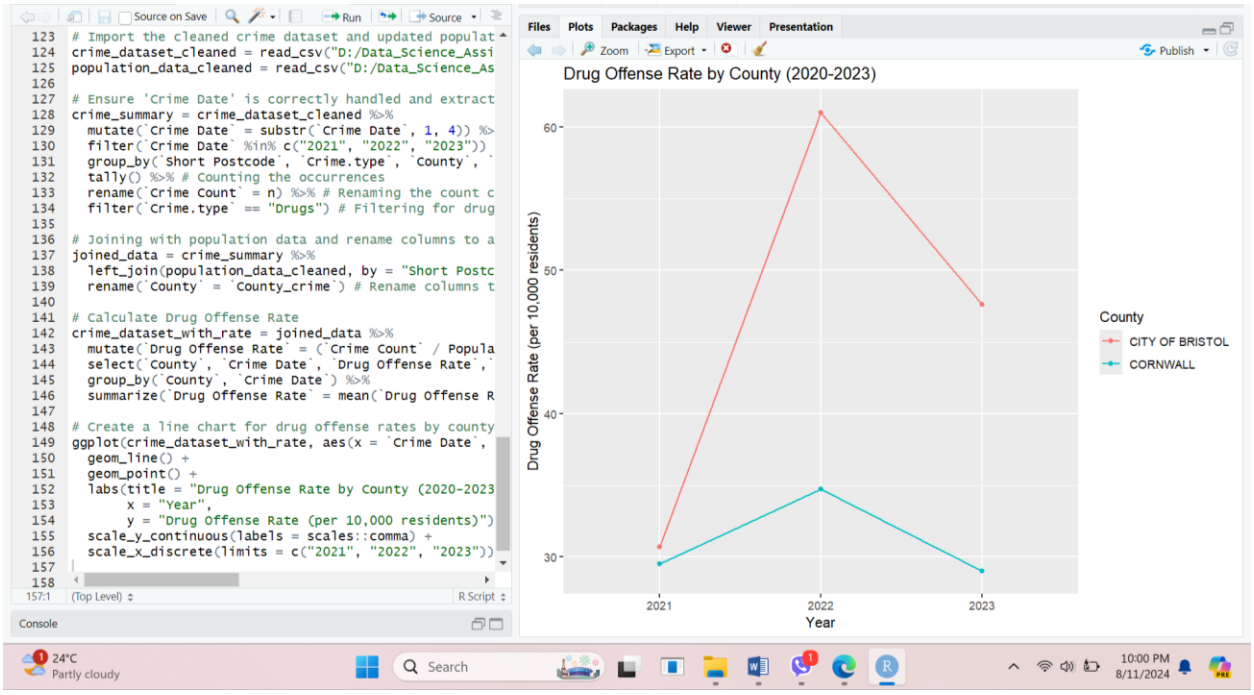



```r
#Pie chart for a specific month of 2023 (robbery rate)
library(dplyr)
library(ggplot2)
```

```
+   pivot_wider(names_from = Crime.type, values_from = Coun
t, values_fill = list(Count = 0))
> # Prepare data for radar chart
> radar_data = as.data.frame(crime_counts)
> row.names(radar_data) = radar_data$Year
> radar_data = radar_data %>% select(-Year)
> # Ensure at least three columns (crime types)
> if (ncol(radar_data) < 3) {
+   stop("Not enough variables for radar chart. Ensure you h
ave at least three crime types.")
+ }
> # Add max and min values for radar chart scaling
> max_values = rep(max(radar_data, na.rm = TRUE), ncol(radar
_data))
> min_values = rep(0, ncol(radar_data))
> radar_data = rbind(max_values, min_values, radar_data)
> # Plot the radar chart
> radarchart(radar_data, axistype = 1,
+            pcol = rgb(0.2, 0.5, 0.5, 0.7), pfcol = rgb(0.
2, 0.5, 0.5, 0.5),
+            cglcol = "grey", cglty = 1, axislabcol = "gre
y", caxislabels = seq(0, max(radar_data[-c(1,2),], na.rm = T
RUE), length.out = 5),
+            title = "Vehicle Crime Rate (Theft, Accident) 2
020 to 2023")
>
```




RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Environment History Connections Tutorial
Files Plots Packages Help Viewer Presentation
Zoom Export Publish
Pie Chart of Robbery Rate by Town/City for 2023-DECEMBER
Town/City
BRISTOL
NEWQUAY
PERRANPORTH
ST AGNES
ST COLUMB
TRURO
Console
ATL - SF
Game score
Search
12:51 PM
8/15/2024

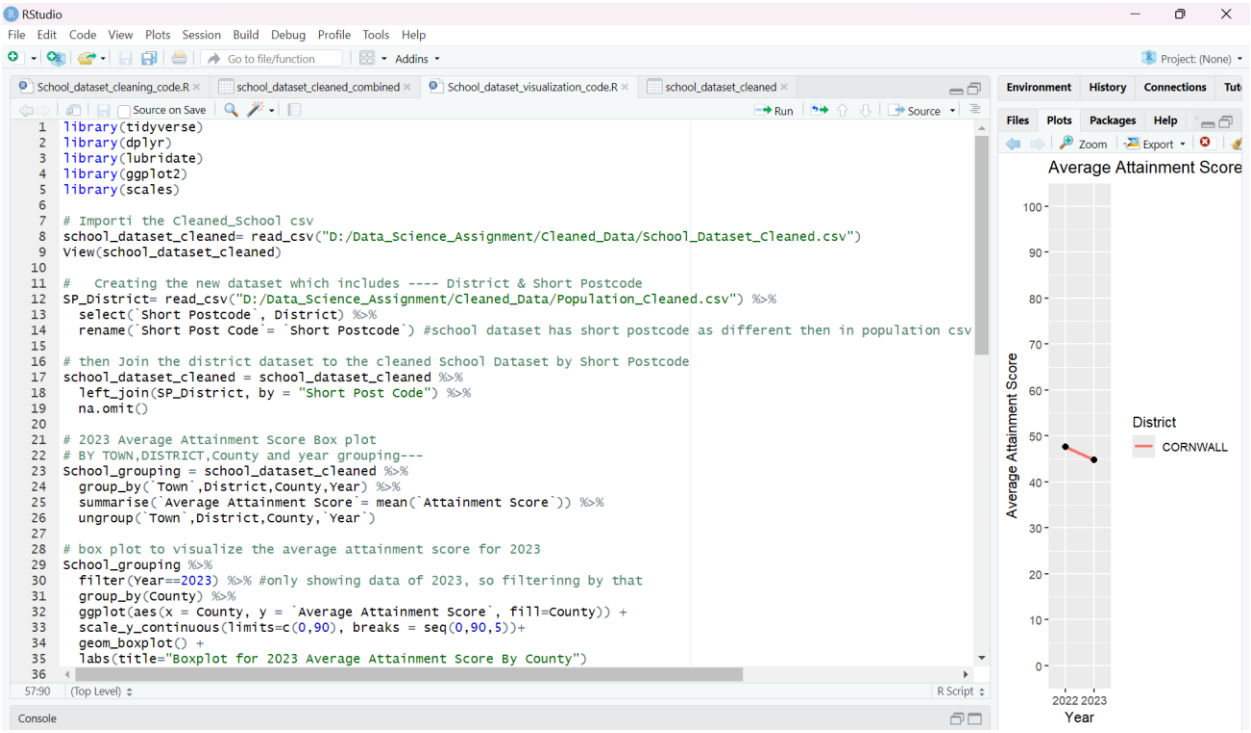

Source on Save
Run
Source
# Import the cleaned crime dataset and updated populat
crime_dataset_cleaned = read_csv("D:/Data_Science_Assi
population_data_cleaned = read_csv("D:/Data_Science_As
# Ensure 'Crime Date' is correctly handled and extract
crime_summary = crime_dataset_cleaned %>%
mutate(`Crime Date` = substr(`Crime Date`, 1, 4)) %>
filter(`Crime Date` %in% c("2021", "2022", "2023"))
group_by(`Short Postcode`, `Crime.type`, `County`,
tally() %>% # Counting the occurrences
rename(`Crime Count` = n) %>% # Renaming the count c
filter(`Crime.type` == "Drugs") # Filtering for drug
# Joining with population data and rename columns to a
joined_data = crime_summary %>%
left_join(population_data_cleaned, by = "Short Postc
rename(`County` = `County_crime`) # Rename columns t
# Calculate Drug Offense Rate
crime_dataset_with_rate = joined_data %>%
mutate(`Drug Offense Rate` = (`Crime Count` / Popula
select(`County`, `Crime Date`, `Drug Offense Rate`,
group_by(`County`, `Crime Date`) %>%
summarize(`Drug Offense Rate` = mean(`Drug Offense R
Create a line chart for drug offense rates by county
ggplot(crime_dataset_with_rate, aes(x = `Crime Date`,
geom_line() +
geom_point() +
labs(title = "Drug Offense Rate by County (2020-2023
x = "Year",
y = "Drug Offense Rate (per 10,000 residents)")
scale_y_continuous(labels = scales::comma) +
scale_x_discrete(limits = c("2021", "2022", "2023"))
Console
Files Plots Packages Help Viewer Presentation
Zoom Export Publish
Drug Offense Rate by County (2020-2023)
Drug Offense Rate (per 10,000 residents)
60
50
40
30
2021
2022
2023
Year
County
CITY OF BRISTOL
CORNWALL
24°C
Partly cloudy
Search
10:00 PM
8/11/2024

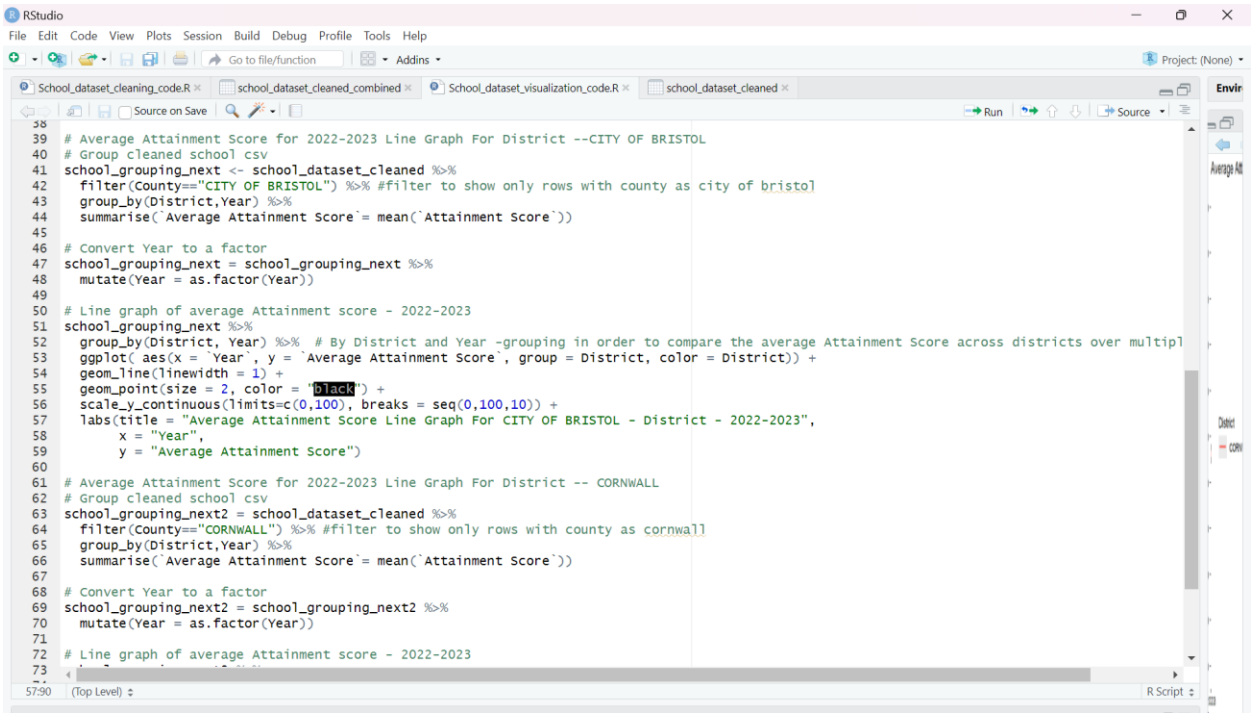
Schools

The cleaned school dataset was merged with a population dataset to include district information, with standardized column names and missing values removed. The analysis then visualized average attainment scores for 2023 using a box plot, grouped by Town, District, County, and Year. This revealed score distributions across counties. Additionally, line graphs were created to show trends in average attainment scores from 2022-2023 by district, highlighting performance changes over time.

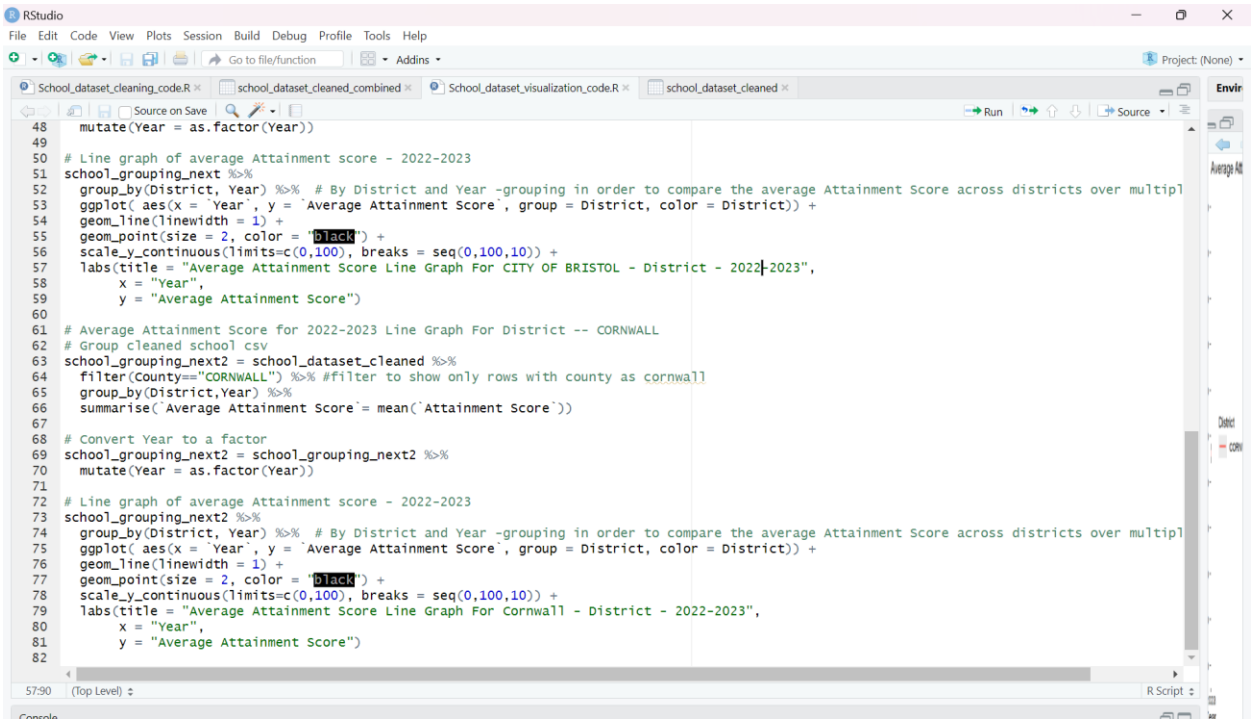
Figure 10

School Visualization

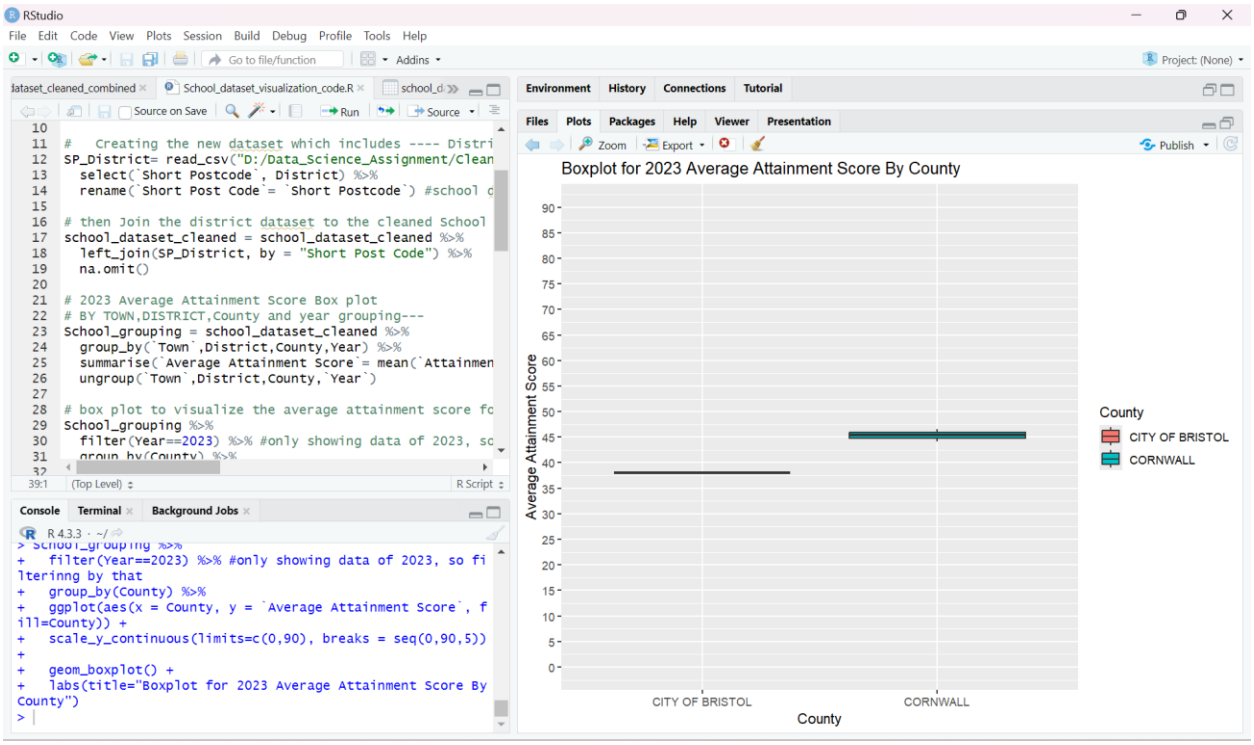


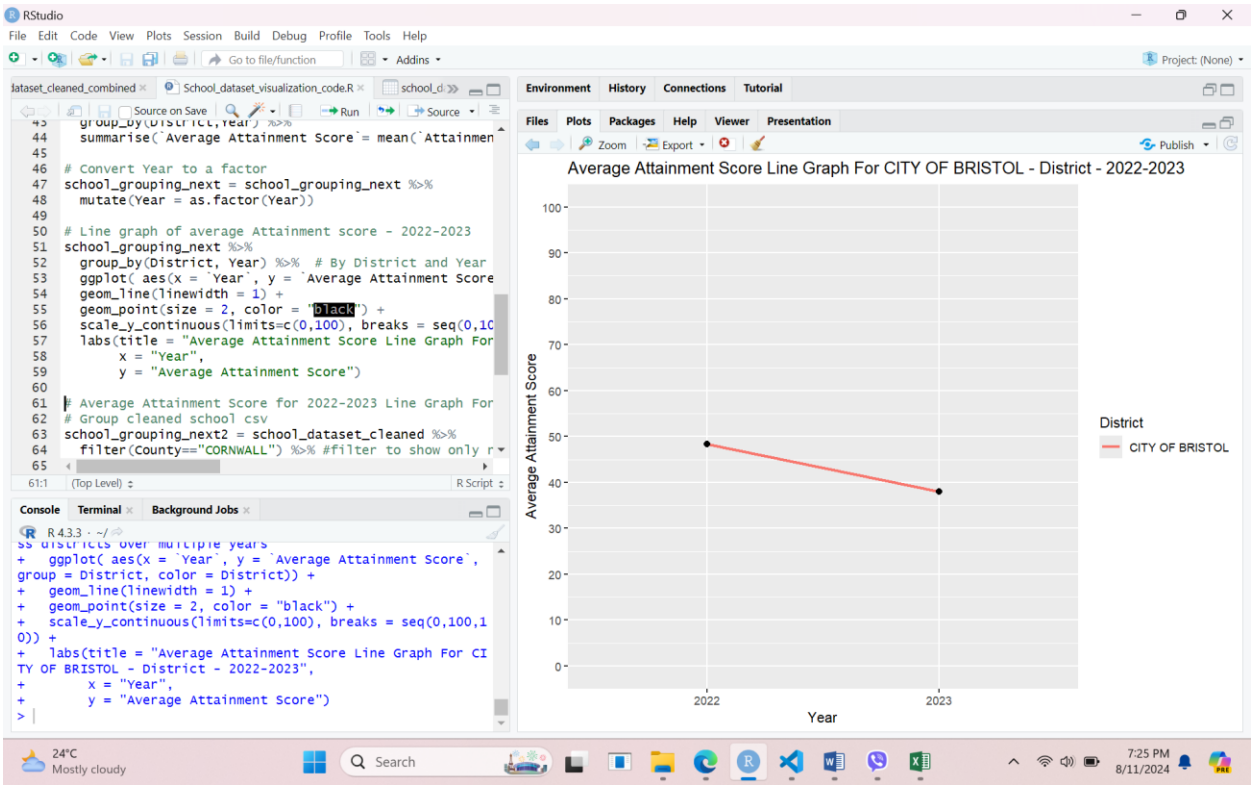


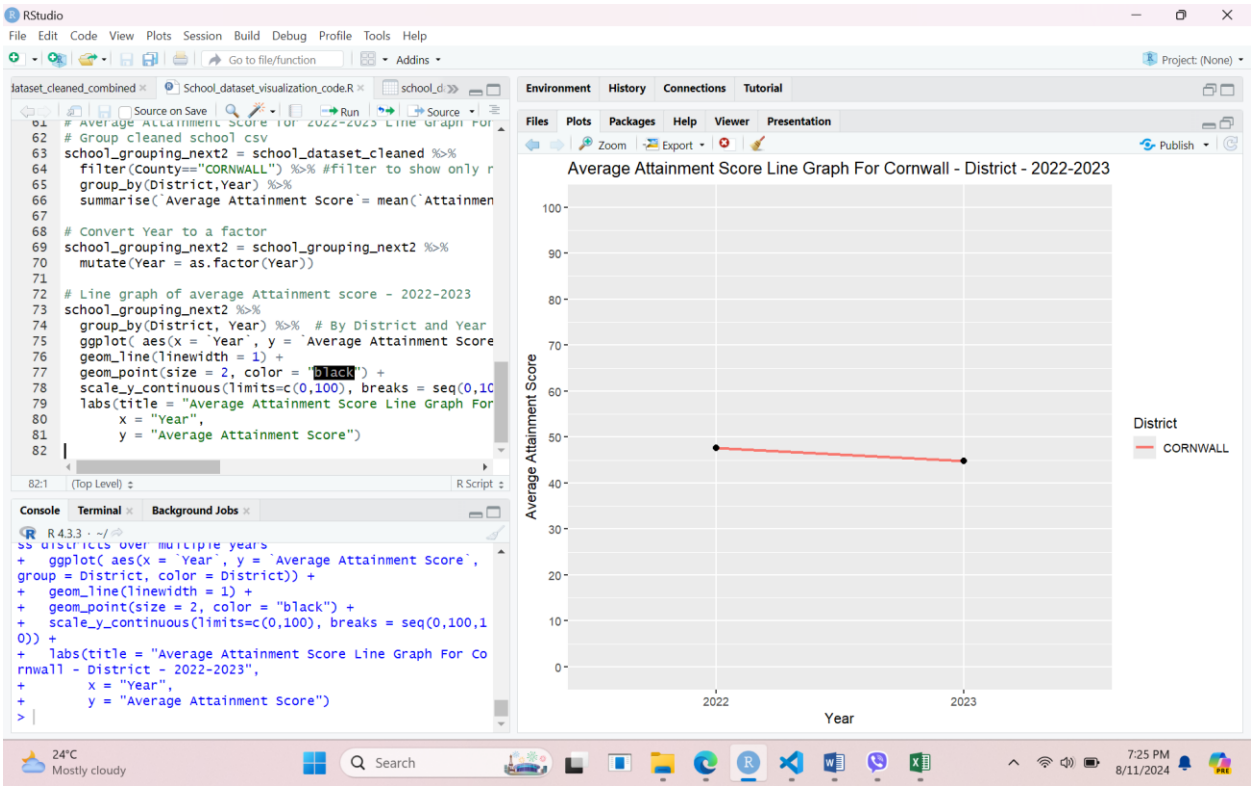
```
# Average Attainment Score for 2022-2023 Line Graph For District --CITY OF BRISTOL
# Group cleaned school csv
school_grouping_next <- school_dataset_cleaned %>%
  filter(County=="CITY OF BRISTOL") %>% #filter to show only rows with county as city of bristol
  group_by(District,Year) %>%
  summarise(`Average Attainment Score`= mean(`Attainment Score`))

# Convert Year to a factor
school_grouping_next = school_grouping_next %>%
  mutate(Year = as.factor(Year))

# Line graph of average Attainment score - 2022-2023
school_grouping_next %>%
  group_by(District, Year) %>%  # By District and Year -grouping in order to compare the average Attainment Score across districts over multipl
  ggplot( aes(x = `Year`, y = `Average Attainment Score`, group = District, color = District)) +
  geom_line(linewidth = 1) +
  geom_point(size = 2, color = "black") +
  scale_y_continuous(limits=c(0,100), breaks = seq(0,100,10)) +
  labs(title = "Average Attainment Score Line Graph For CITY OF BRISTOL - District - 2022-2023",
       x = "Year",
       y = "Average Attainment Score")

# Average Attainment Score for 2022-2023 Line Graph For District -- CORNWALL
# Group cleaned school csv
school_grouping_next2 = school_dataset_cleaned %>%
  filter(County=="CORNWALL") %>% #filter to show only rows with county as cornwall
  group_by(District,Year) %>%
  summarise(`Average Attainment Score`= mean(`Attainment Score`))

# Convert Year to a factor
school_grouping_next2 = school_grouping_next2 %>%
  mutate(Year = as.factor(Year))

# Line graph of average Attainment score - 2022-2023
```



```
  mutate(Year = as.factor(Year))

# Line graph of average Attainment score - 2022-2023
school_grouping_next %>%
  group_by(District, Year) %>%  # By District and Year -grouping in order to compare the average Attainment Score across districts over multipl
  ggplot( aes(x = `Year`, y = `Average Attainment Score`, group = District, color = District)) +
  geom_line(linewidth = 1) +
  geom_point(size = 2, color = "black") +
  scale_y_continuous(limits=c(0,100), breaks = seq(0,100,10)) +
  labs(title = "Average Attainment Score Line Graph For CITY OF BRISTOL - District - 2022-2023",
       x = "Year",
       y = "Average Attainment Score")

# Average Attainment Score for 2022-2023 Line Graph For District -- CORNWALL
# Group cleaned school csv
school_grouping_next2 = school_dataset_cleaned %>%
  filter(County=="CORNWALL") %>% #filter to show only rows with county as cornwall
  group_by(District,Year) %>%
  summarise(`Average Attainment Score`= mean(`Attainment Score`))

# Convert Year to a factor
school_grouping_next2 = school_grouping_next2 %>%
  mutate(Year = as.factor(Year))

# Line graph of average Attainment score - 2022-2023
school_grouping_next2 %>%
  group_by(District, Year) %>%  # By District and Year -grouping in order to compare the average Attainment Score across districts over multipl
  ggplot( aes(x = `Year`, y = `Average Attainment Score`, group = District, color = District)) +
  geom_line(linewidth = 1) +
  geom_point(size = 2, color = "black") +
  scale_y_continuous(limits=c(0,100), breaks = seq(0,100,10)) +
  labs(title = "Average Attainment Score Line Graph For Cornwall - District - 2022-2023",
       x = "Year",
       y = "Average Attainment Score")
```


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Boxplot for 2023 Average Attainment Score By County
Average Attainment Score
County
CITY OF BRISTOL
CORNWALL
Console Terminal Background Jobs
R 4.3.3 · ~/


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Average Attainment Score Line Graph For CITY OF BRISTOL - District - 2022-2023
Average Attainment Score
Year
2022
2023
District
CITY OF BRISTOL
Console Terminal Background Jobs
R 4.3.3 · ~/


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
dataset_cleaned_combined
School_dataset_visualization_code.R
school_d
Source on Save
Run
Source
# Average Attainment Score for 2022-2023 Line Graph For
# Group cleaned school csv
school_grouping_next2 = school_dataset_cleaned %>%
filter(County=="CORNWALL") %>% #filter to show only r
group_by(District,Year) %>%
summarise(`Average Attainment Score`= mean(`Attainmen
# Convert Year to a factor
school_grouping_next2 = school_grouping_next2 %>%
mutate(Year = as.factor(Year))
# Line graph of average Attainment score - 2022-2023
school_grouping_next2 %>%
group_by(District, Year) %>% # By District and Year
ggplot(aes(x = `Year`, y = `Average Attainment Score
geom_line(linewidth = 1) +
geom_point(size = 2, color = "black") +
scale_y_continuous(limits=c(0,100), breaks = seq(0,10
labs(title = "Average Attainment Score Line Graph For
x = "Year",
y = "Average Attainment Score")
82:1
(Top Level)
R Script
Console
Terminal
Background Jobs
R 4.3.3 · ~/
ss districts over multiple years
+ ggplot(aes(x = `Year`, y = `Average Attainment Score`,
group = District, color = District)) +
+ geom_line(linewidth = 1) +
+ geom_point(size = 2, color = "black") +
+ scale_y_continuous(limits=c(0,100), breaks = seq(0,100,1
0)) +
+ labs(title = "Average Attainment Score Line Graph For Co
rnwall - District - 2022-2023",
+ x = "Year",
+ y = "Average Attainment Score")
>
Environment
History
Connections
Tutorial
Files
Plots
Packages
Help
Viewer
Presentation
Zoom
Export
Publish
Average Attainment Score Line Graph For Cornwall - District - 2022-2023
Average Attainment Score
100
90
80
70
60
50
40
30
20
10
0
2022
2023
Year
District
CORNWALL
24°C
Mostly cloudy
Search
7:25 PM
8/11/2024

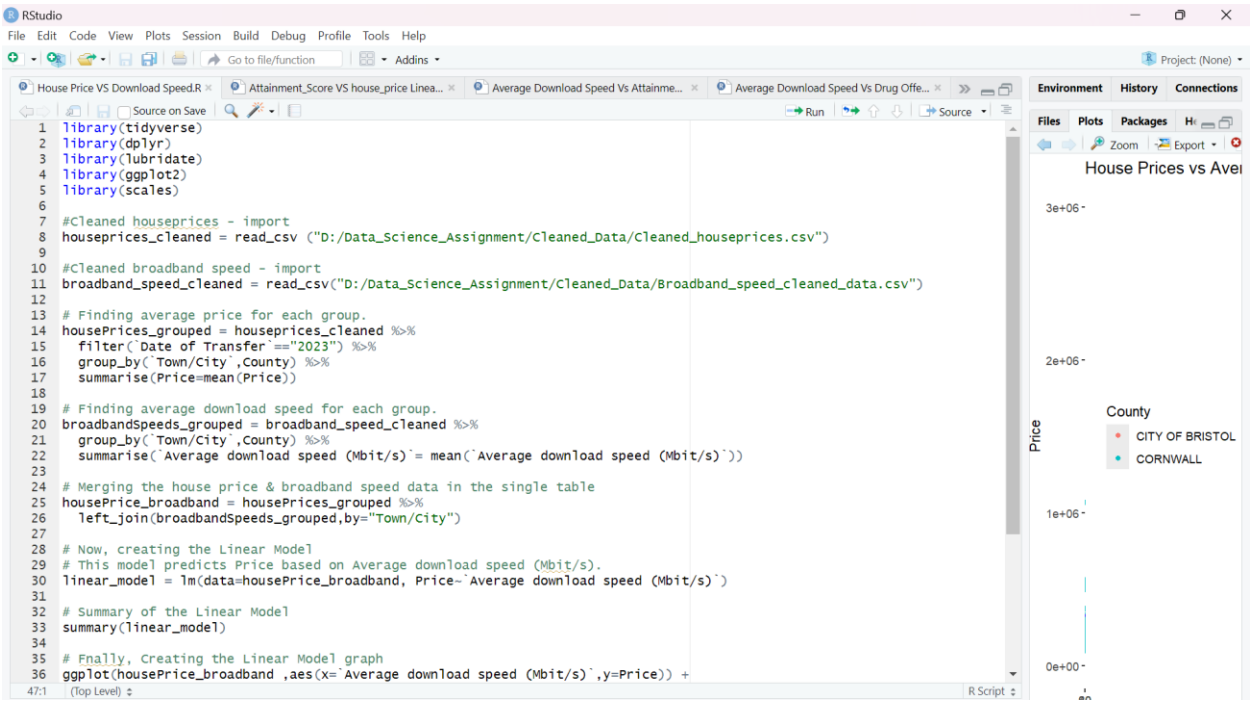
Linear Modeling

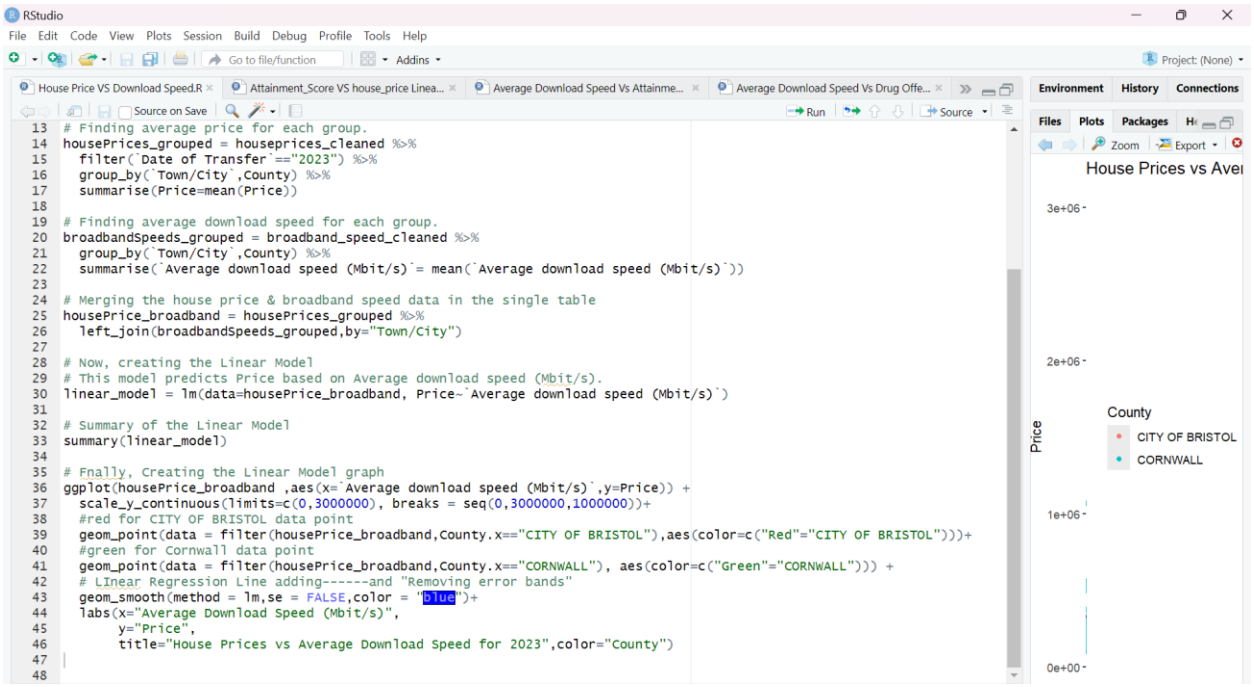
1. House prices vs Download Speed

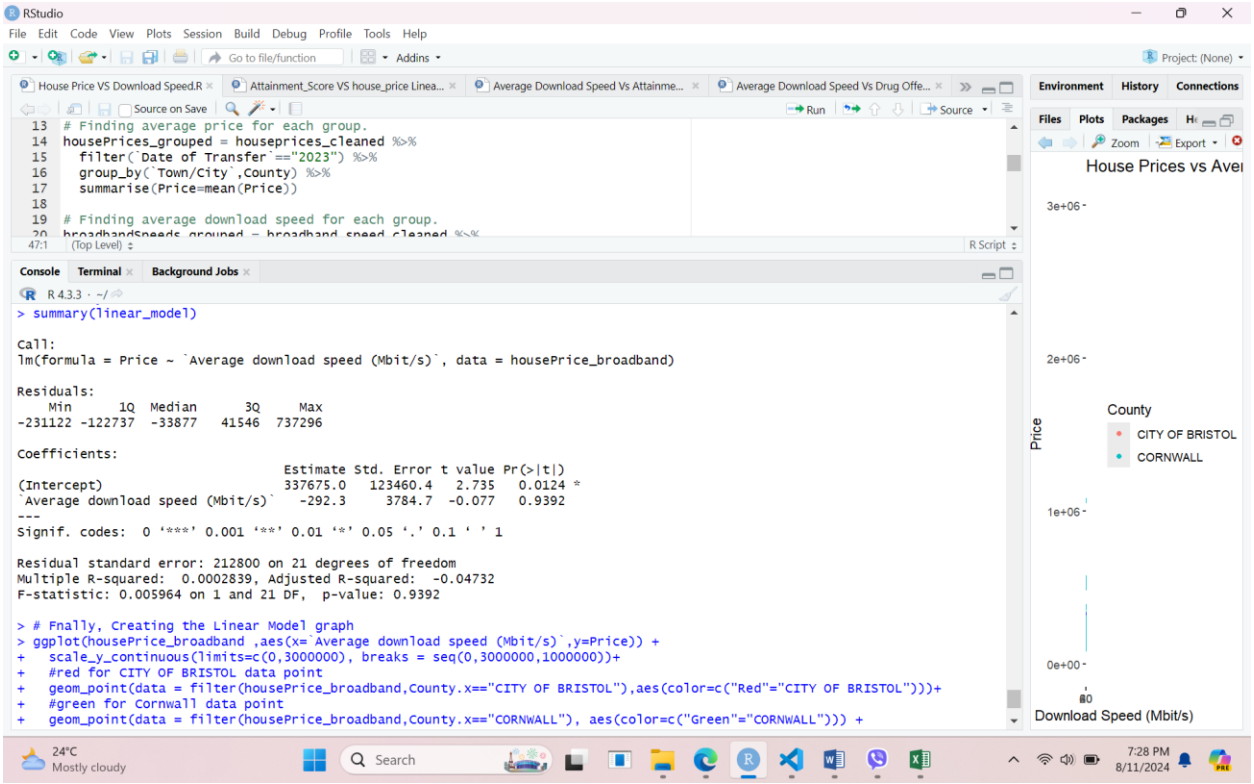
The analysis shows no statistically significant relationship between average download speed and house prices in the dataset. The p-value for the average download speed is very high (0.9392), and the R-squared value is near zero, indicating that average download speed does not explain the variation in house prices. The flat trend line in the plot further supports the conclusion that there is no strong relationship between house prices and average download speed in the data provided.

Figure 11

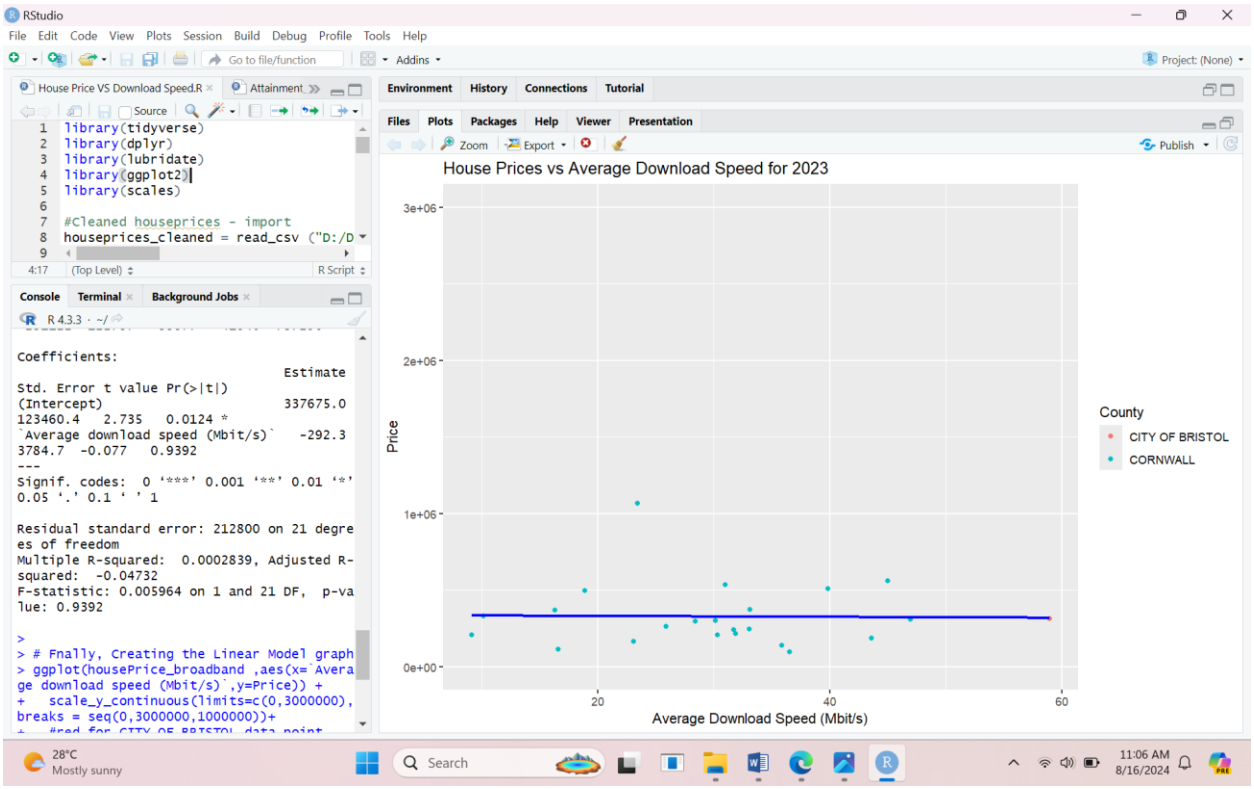
House prices vs Download Speed




RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
House Price VS Download Speed.R
Attainment_Score VS house_price Linea...
Average Download Speed Vs Attainme...
Average Download Speed Vs Drug Offe...
Environment History Connections
Files Plots Packages
House Prices vs Avei
3e+06
2e+06
1e+06
0e+00
Price
County
CITY OF BRISTOL
CORNWALL
13 # Finding average price for each group.
14 housePrices_grouped = houseprices_cleaned %>%
15 filter(`Date of Transfer`=="2023") %>%
16 group_by(`Town/City`,County) %>%
17 summarise(Price=mean(Price))
18
19 # Finding average download speed for each group.
20 broadbandSpeeds_grouped = broadband_speed_cleaned %>%
21 group_by(`Town/City`,County) %>%
22 summarise(`Average download speed (Mbit/s)`= mean(`Average download speed (Mbit/s)`))
23
24 # Merging the house price & broadband speed data in the single table
25 housePrice_broadband = housePrices_grouped %>%
26 left_join(broadbandSpeeds_grouped,by="Town/City")
27
28 # Now, creating the Linear Model
29 # This model predicts Price based on Average download speed (Mbit/s).
30 linear_model = lm(data=housePrice_broadband, Price~`Average download speed (Mbit/s)`)
31
32 # Summary of the Linear Model
33 summary(linear_model)
34
35 # Fnally, Creating the Linear Model graph
36 ggplot(housePrice_broadband ,aes(x=`Average download speed (Mbit/s)`,y=Price)) +
37 scale_y_continuous(limits=c(0,3000000), breaks = seq(0,3000000,1000000))+
38 #red for CITY OF BRISTOL data point
39 geom_point(data = filter(housePrice_broadband,County.x=="CITY OF BRISTOL"),aes(color=c("Red"="CITY OF BRISTOL")))+
40 #green for Cornwall data point
41 geom_point(data = filter(housePrice_broadband,County.x=="CORNWALL"), aes(color=c("Green"="CORNWALL"))) +
42 # LInear Regression Line adding------and "Removing error bands"
43 geom_smooth(method = lm,se = FALSE,color = "blue")+
44 labs(x="Average Download Speed (Mbit/s)",
45 y="Price",
46 title="House Prices vs Average Download Speed for 2023",color="County")
47
48


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
House Price VS Download Speed.R
Attainment_Score VS house_price Linea...
Average Download Speed Vs Attainme...
Average Download Speed Vs Drug Offe...
Environment History Connections
Files Plots Packages
House Prices vs Avei
3e+06
2e+06
1e+06
0e+00
Price
County
CITY OF BRISTOL
CORNWALL
Download Speed (Mbit/s)
13 # Finding average price for each group.
14 housePrices_grouped = houseprices_cleaned %>%
15 filter(`Date of Transfer`=="2023") %>%
16 group_by(`Town/City`,County) %>%
17 summarise(Price=mean(Price))
18
19 # Finding average download speed for each group.
47:1 (Top Level)
R Script
Console Terminal Background Jobs
R 4.3.3 · ~/
> summary(linear_model)
Call:
lm(formula = Price ~ `Average download speed (Mbit/s)`, data = housePrice_broadband)
Residuals:
Min 1Q Median 3Q Max
-231122 -122737 -33877 41546 737296
Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 337675.0 123460.4 2.735 0.0124 *
`Average download speed (Mbit/s)` -292.3 3784.7 -0.077 0.9392

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 212800 on 21 degrees of freedom
Multiple R-squared: 0.0002839, Adjusted R-squared: -0.04732
F-statistic: 0.005964 on 1 and 21 DF, p-value: 0.9392
> # Fnally, Creating the Linear Model graph
> ggplot(housePrice_broadband ,aes(x=`Average download speed (Mbit/s)`,y=Price)) +
+ scale_y_continuous(limits=c(0,3000000), breaks = seq(0,3000000,1000000))+
+ #red for CITY OF BRISTOL data point
+ geom_point(data = filter(housePrice_broadband,County.x=="CITY OF BRISTOL"),aes(color=c("Red"="CITY OF BRISTOL")))+
+ #green for Cornwall data point
+ geom_point(data = filter(housePrice_broadband,County.x=="CORNWALL"), aes(color=c("Green"="CORNWALL"))) +
24°C
Mostly cloudy
Search
7:28 PM
8/11/2024


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
House Price VS Download Speed.R
Attainment
1 library(tidyverse)
2 library(dplyr)
3 library(lubridate)
4 library(ggplot2)
5 library(scales)
7 #Cleaned houseprices - import
8 houseprices_cleaned = read_csv ("D:/D
Console Terminal Background Jobs
Coefficients:
Estimate
Std. Error t value Pr(>|t|)
(Intercept) 337675.0
123460.4 2.735 0.0124 *
`Average download speed (Mbit/s)` -292.3
3784.7 -0.077 0.9392

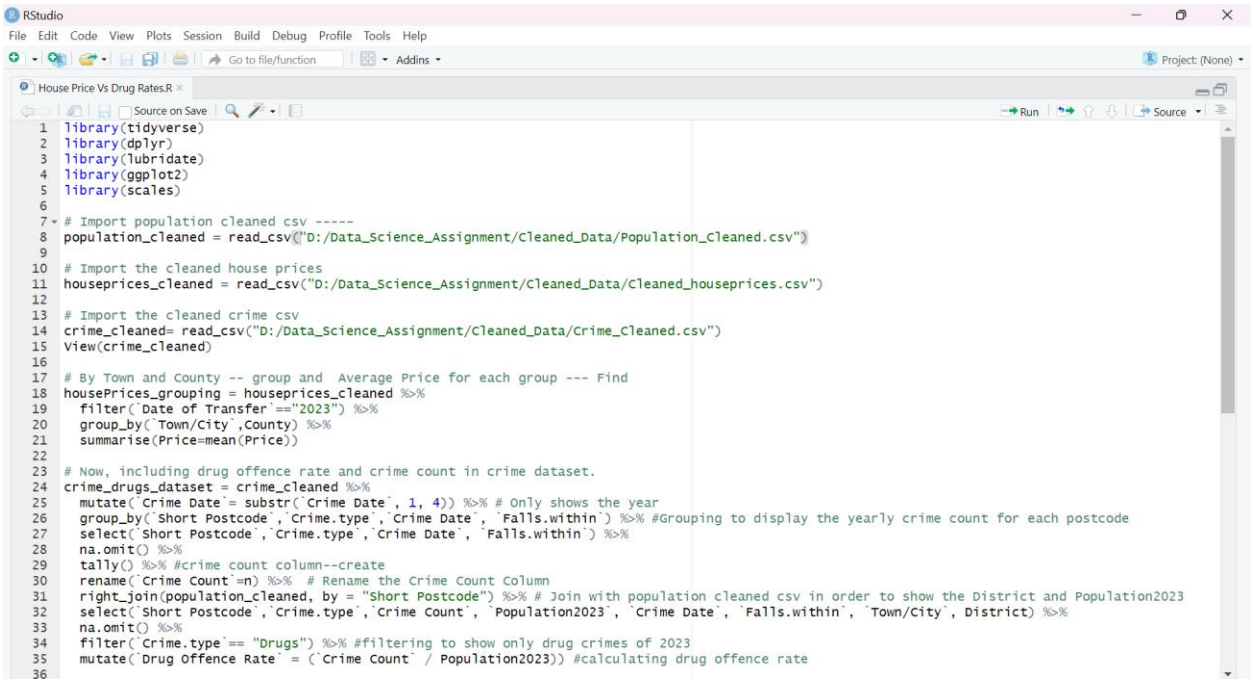
Signif. codes: 0 '***' 0.001 '**' 0.01 '*'
0.05 '.' 0.1 ' ' 1
Residual standard error: 212800 on 21 degre
es of freedom
Multiple R-squared: 0.0002839, Adjusted R-
squared: -0.04732
F-statistic: 0.005964 on 1 and 21 DF, p-va
lue: 0.9392
> # Fnally, Creating the Linear Model graph
> ggplot(housePrice_broadband ,aes(x=`Avera
ge download speed (Mbit/s)`,y=Price)) +
+ scale_y_continuous(limits=c(0,3000000),
breaks = seq(0,3000000,1000000))+
Environment History Connections Tutorial
Files Plots Packages Help Viewer Presentation
House Prices vs Average Download Speed for 2023
Price
3e+06
2e+06
1e+06
0e+00
20
40
60
Average Download Speed (Mbit/s)
County
CITY OF BRISTOL
CORNWALL
28°C
Mostly sunny
Search
11:06 AM
8/16/2024

2. House prices vs drug rates

The linear model graph displays the correlation between drug offense rates and home prices in Cornwall and the City of Bristol, two UK counties. According to the graph, there is a very weak positive link between drug offense rates and home prices, which means that when drug offense rates rise, house values typically rise as well. But the correlation is so weak that it is essentially insignificant.

Figure 12

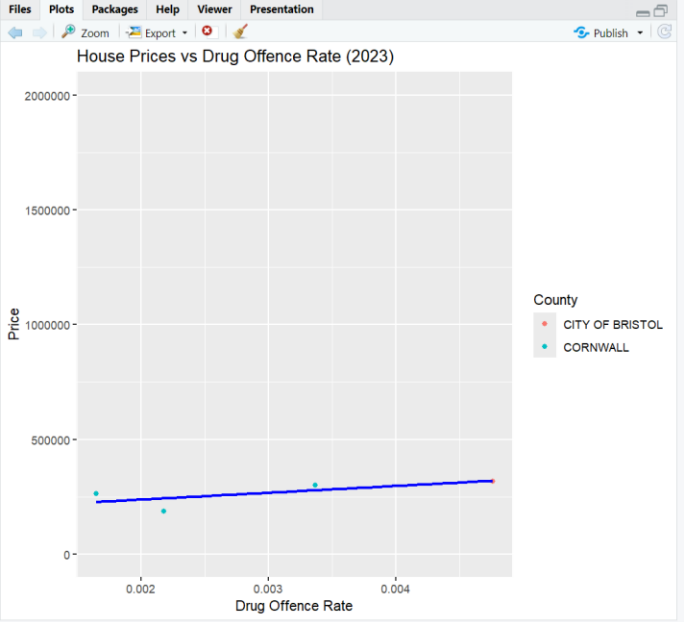
House prices vs Drug Offence Rates



```
  filter(`Crime.type`== "Drugs") %>% #filtering to show only drug crimes of 2023
  mutate(`Drug Offence Rate` = (`Crime Count` / Population2023)) #calculating drug offence rate

# By county and town grouping the crime and drug csv & Showing the rate for each group for the year 2023
drug_crime_grouping = crime_drugs_dataset %>%
  filter(`Crime Date`=="2023") %>%
  group_by(`Falls.within`,`Town/City`) %>%
  summarise(`Drug Offence Rate`= mean(`Drug Offence Rate`))

# House price data and drug crime rate data ----JOINING---in a single table
housePrice_drugCrime = housePrices_grouping %>%
  left_join(drug_crime_grouping,by="Town/City") %>%
  na.omit

# Create a Linear Model
# This model predicts House Price based on Drug Rates.
linear_model = lm(data = housePrice_drugCrime , Price~`Drug Offence Rate`)

# Show Summary of Linear Model
summary(linear_model)

# Create Linear Model Graph
ggplot(housePrice_drugCrime,aes(x=`Drug Offence Rate`,y=Price)) +
  scale_y_continuous(limits=c(0,2000000), breaks = seq(0,2000000,500000))+
  scale_x_continuous(labels = label_number(scale = 1, big.mark = ",")) + # Format the x-axis labels
  # Red as City of Bristol data point.
  geom_point(data = filter(housePrice_drugCrime,County=="CITY OF BRISTOL"),aes(color=c("Red"="CITY OF BRISTOL")))+
  # Green for Cornwall data point.
  geom_point(data = filter(housePrice_drugCrime,County=="CORNWALL"), aes(color=c("Green"="CORNWALL"))) +
  # LInear Regression Line adding------and "Removing error bands"
  geom_smooth(method = lm , se = FALSE , color = "blue")+
  labs(x="Drug Offence Rate",
       y="Price",
       title="House Prices vs Drug Offence Rate (2023) ",color="County")
```

```
  na.omit

# Create a Linear Model
# This model predicts House Price based on Drug Rates.
linear_model = lm(data = housePrice_drugCrime , Price~`Drug O
```

```
> linear_model = lm(data = housePrice_drugCrime , Price~`Drug Offe
nce Rate`)
>
> # Show Summary of Linear Model
> summary(linear_model)

Call:
lm(formula = Price ~ `Drug Offence Rate`, data = housePrice_drugCr
ime)

Residuals:
     1      2      3      4
 -3223  37144 -57397  23476

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)         178624      68736   2.599    0.122
`Drug Offence Rate` 29607103   21349191   1.387    0.300

Residual standard error: 51160 on 2 degrees of freedom
Multiple R-squared:  0.4902,    Adjusted R-squared:  0.2353
F-statistic: 1.923 on 1 and 2 DF,  p-value: 0.2998

>
> # Create Linear Model Graph
> ggplot(housePrice_drugCrime,aes(x=`Drug Offence Rate`,y=Price))
+
+   scale_y_continuous(limits=c(0,2000000), breaks = seq(0,200000
```



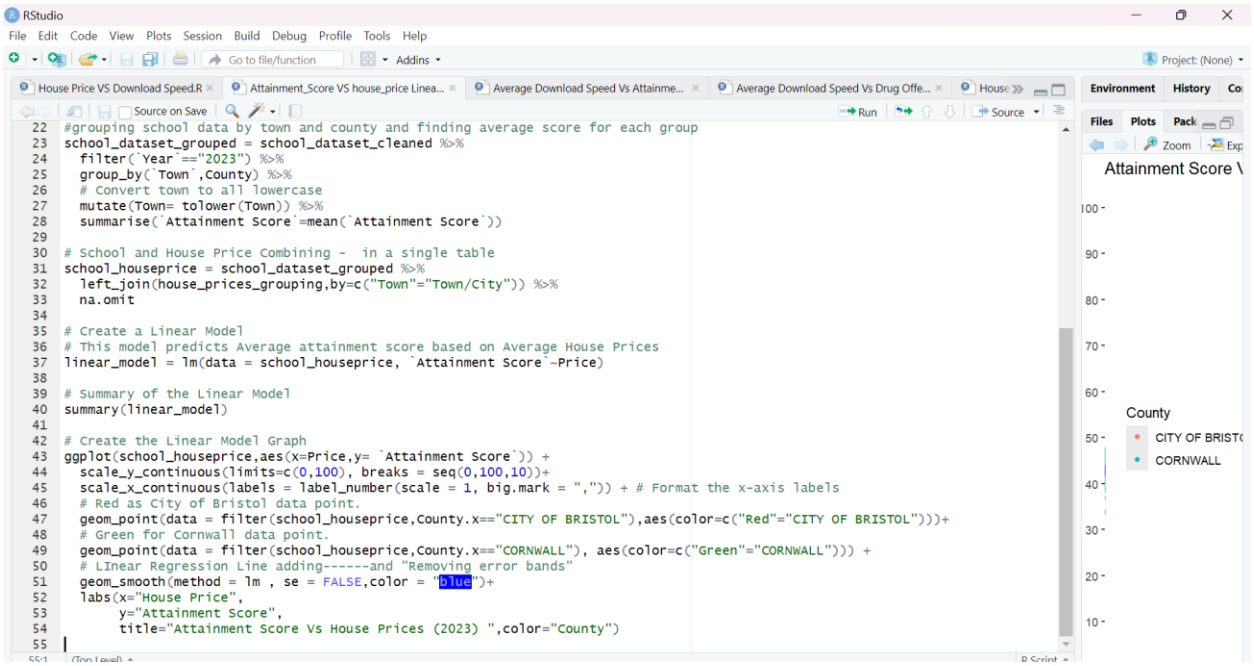
3. Attainment 8 Score vs house price

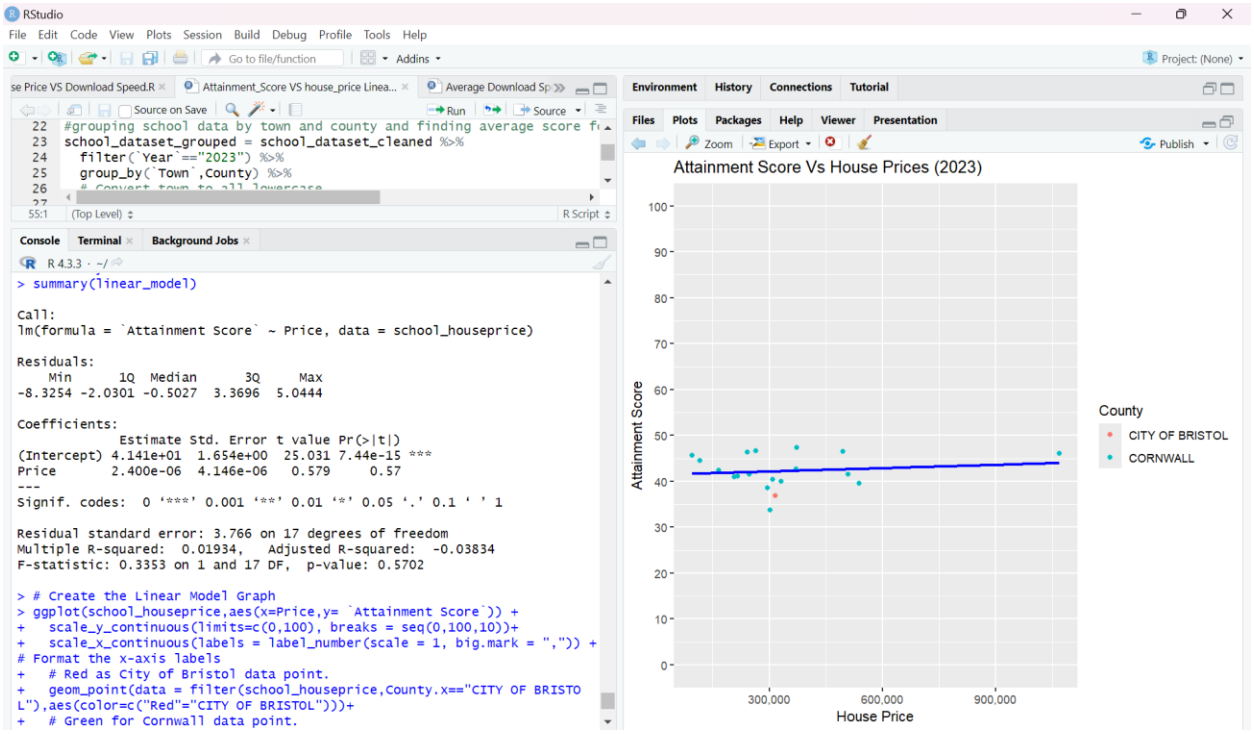
Most data points are clustered around lower house prices (below 500,000) and attainment scores between 30 and 50. The trend line is nearly flat, suggesting that there is little to no significant correlation between house prices and attainment scores in the data provided. This means that higher house prices do not necessarily correlate with higher or lower attainment scores.

Figure 13

Attainment 8 Score vs House Price


```r
#Attainment 8 Score VS house price (both counties combined)
library(tidyverse)
library(dplyr)
library(lubridate)
library(ggplot2)
library(scales)

# Import Cleaned House Prices
houseprices_cleaned = read_csv("D:/Data_Science_Assignment/Cleaned_Data/Cleaned_houseprices.csv")

# Import Cleaned School csv
school_dataset_cleaned= read_csv("D:/Data_Science_Assignment/Cleaned_Data/School_Dataset_Cleaned.csv")

# By Town and County grouping the House Prices & Average Price --- Finding for each group
house_prices_grouping = houseprices_cleaned %>%
  filter(`Date of Transfer`=="2023") %>%
  group_by(`Town/City`,County) %>%
  # From uppercase to all lowercase to the Town column
  mutate(`Town/City` = tolower(`Town/City`)) %>%
  summarise(Price=mean(Price))

#grouping school data by town and county and finding average score for each group
school_dataset_grouped = school_dataset_cleaned %>%
  filter(`Year`=="2023") %>%
  group_by(`Town`,County) %>%
  # Convert town to all lowercase
  mutate(Town= tolower(Town)) %>%
  summarise(`Attainment Score`=mean(`Attainment Score`))

# School and House Price Combining -  in a single table
school_houseprice = school_dataset_grouped %>%
  left_join(house_prices_grouping,by=c("Town"="Town/City")) %>%
  na.omit

# Create a Linear Model
# This model predicts Average attainment score based on Average House Prices
```


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
House Price VS Download Speed.R
Attainment_Score VS house_price Linea...
Average Download Speed Vs Attainme...
Average Download Speed Vs Drug Offe...
#grouping school data by town and county and finding average score for each group
school_dataset_grouped = school_dataset_cleaned %>%
filter(`Year`=="2023") %>%
group_by(`Town`,County) %>%
Convert town to all lowercase
mutate(Town= tolower(Town)) %>%
summarise(`Attainment Score`=mean(`Attainment Score`))
# School and House Price Combining - in a single table
school_houseprice = school_dataset_grouped %>%
left_join(house_prices_grouping,by=c("Town"="Town/City")) %>%
na.omit
# Create a Linear Model
# This model predicts Average attainment score based on Average House Prices
linear_model = lm(data = school_houseprice, `Attainment Score`~Price)
# Summary of the Linear Model
summary(linear_model)
# Create the Linear Model Graph
ggplot(school_houseprice,aes(x=Price,y= `Attainment Score`)) +
scale_y_continuous(limits=c(0,100), breaks = seq(0,100,10))+
scale_x_continuous(labels = label_number(scale = 1, big.mark = ",")) + # Format the x-axis labels
Red as City of Bristol data point.
geom_point(data = filter(school_houseprice,County.x=="CITY OF BRISTOL"),aes(color=c("Red"="CITY OF BRISTOL")))+
Green for Cornwall data point.
geom_point(data = filter(school_houseprice,County.x=="CORNWALL"), aes(color=c("Green"="CORNWALL"))) +
LInear Regression Line adding------and "Removing error bands"
geom_smooth(method = lm , se = FALSE,color = "blue")+
labs(x="House Price",
y="Attainment Score",
title="Attainment Score Vs House Prices (2023) ",color="County")
Environment History
Files Plots
Attainment Score V
County
CITY OF BRISTO
CORNWALL


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Environment History Connections Tutorial
Files Plots Packages Help Viewer Presentation
Attainment Score Vs House Prices (2023)
Attainment Score
House Price
300,000
600,000
900,000
County
CITY OF BRISTOL
CORNWALL
Console Terminal Background Jobs
R 4.3.3 · ~/
> summary(linear_model)
Call:
lm(formula = `Attainment Score` ~ Price, data = school_houseprice)
Residuals:
Min 1Q Median 3Q Max
-8.3254 -2.0301 -0.5027 3.3696 5.0444
Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 4.141e+01 1.654e+00 25.031 7.44e-15 ***
Price 2.400e-06 4.146e-06 0.579 0.57

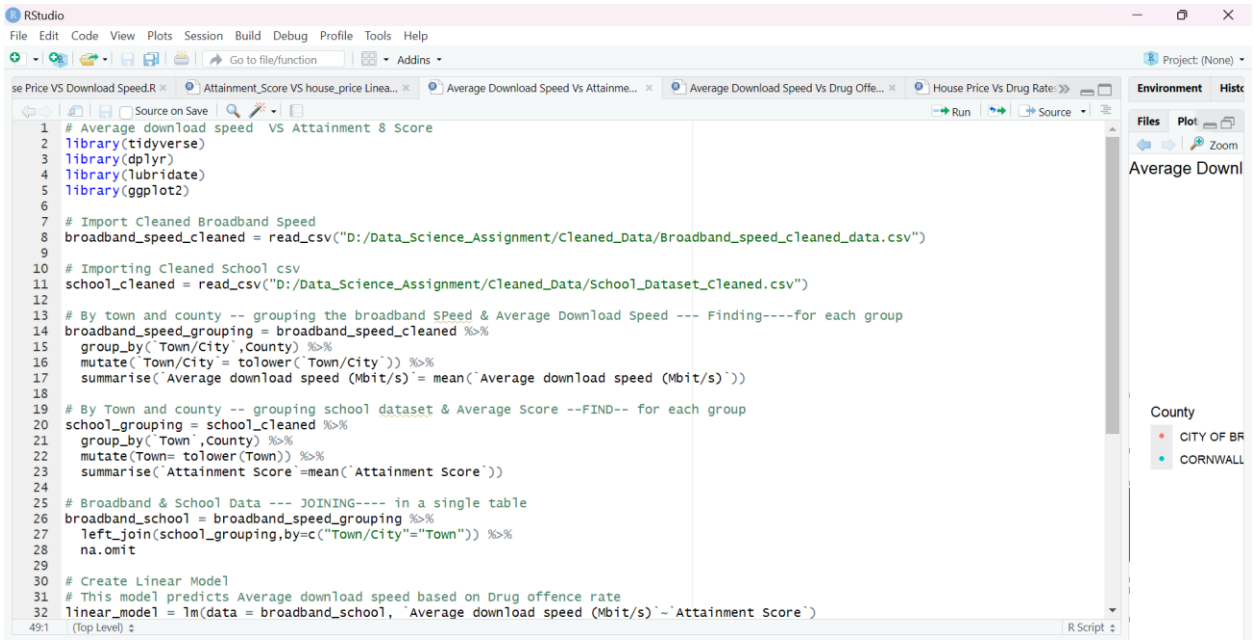
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 3.766 on 17 degrees of freedom
Multiple R-squared: 0.01934, Adjusted R-squared: -0.03834
F-statistic: 0.3353 on 1 and 17 DF, p-value: 0.5702
> # Create the Linear Model Graph
> ggplot(school_houseprice,aes(x=Price,y= `Attainment Score`)) +
+ scale_y_continuous(limits=c(0,100), breaks = seq(0,100,10))+
+ scale_x_continuous(labels = label_number(scale = 1, big.mark = ",")) +
# Format the x-axis labels
+ # Red as City of Bristol data point.
+ geom_point(data = filter(school_houseprice,County.x=="CITY OF BRISTO
L"),aes(color=c("Red"="CITY OF BRISTOL")))+
+ # Green for Cornwall data point.

4. Average download speed vs Attainment 8 Score

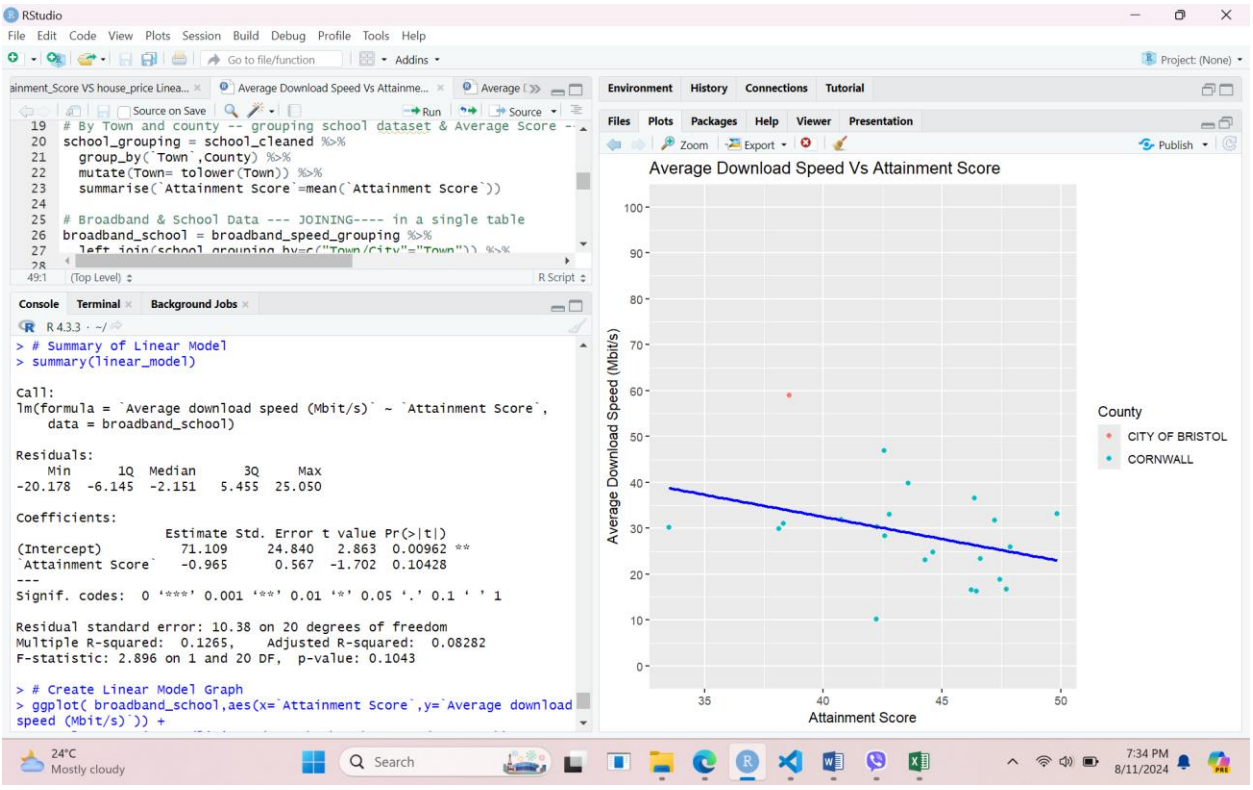
City of Bristol has a relatively low attainment score but the highest average download speed among all data points. Cornwall shows a wider spread of attainment scores, mostly around 40-45, but the download speeds are consistently lower than the City of Bristol. The slight negative slope of the trend line suggests that, on average, regions with higher attainment scores tend to have lower average download speeds.

Figure 14

Average download speed vs Attainment 8 Score



```
# By Town and county -- grouping school dataset & Average Score --FIND-- for each group
school_grouping = school_cleaned %>%
  group_by(`Town`,County) %>%
  mutate(Town= tolower(Town)) %>%
  summarise(`Attainment Score`=mean(`Attainment Score`))

# Broadband & School Data --- JOINING---- in a single table
broadband_school = broadband_speed_grouping %>%
  left_join(school_grouping,by=c("Town/City"="Town")) %>%
  na.omit

# Create Linear Model
# This model predicts Average download speed based on Drug offence rate
linear_model = lm(data = broadband_school, `Average download speed (Mbit/s)`~`Attainment Score`)

# Summary of Linear Model
summary(linear_model)

# Create Linear Model Graph
ggplot( broadband_school,aes(x=`Attainment Score`,y=`Average download speed (Mbit/s)`)) +
  scale_y_continuous(limits=c(0,100), breaks = seq(0,100,10))+
  # Red as City of Bristol data point.
  geom_point(data = filter(broadband_school , County.x=="CITY OF BRISTOL"),aes(color=c("Red"="CITY OF BRISTOL")))+
  # Green for Cornwall data point
  geom_point(data = filter(broadband_school , County.x=="CORNWALL"), aes(color=c("Green"="CORNWALL"))) +
  # LInear Regression Line adding------and "Removing error bands"
  geom_smooth(method = lm , se = FALSE ,color = "blue")+
  labs(x="Attainment Score",
       y="Average Download Speed (Mbit/s)",
       title="Average Download Speed Vs Attainment Score",color="County")
```



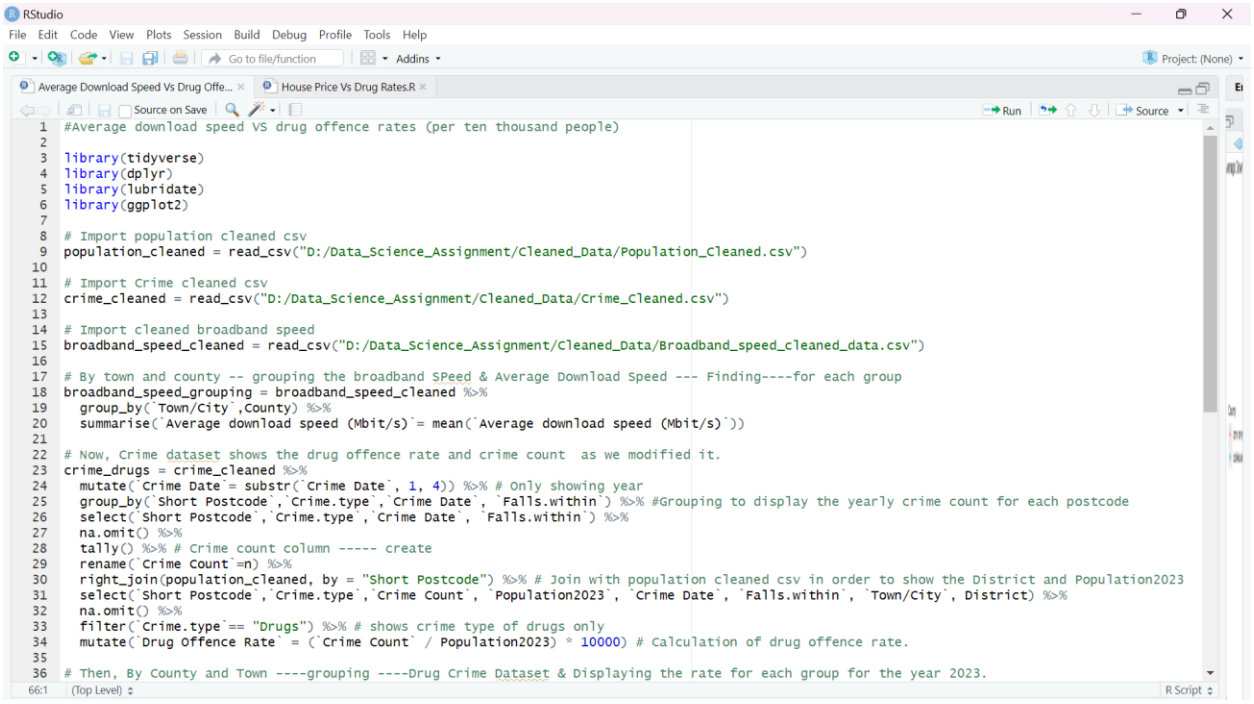
```
> # Summary of Linear Model
> summary(linear_model)

Call:
lm(formula = `Average download speed (Mbit/s)` ~ `Attainment Score`,
    data = broadband_school)

Residuals:
    Min      1Q  Median      3Q     Max
-20.178  -6.145  -2.151   5.455  25.050

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)        71.109     24.840   2.863  0.00962 **
`Attainment Score`  -0.965      0.567  -1.702  0.10428
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.38 on 20 degrees of freedom
Multiple R-squared:  0.1265,    Adjusted R-squared:  0.08282
F-statistic: 2.896 on 1 and 20 DF,  p-value: 0.1043

> # Create Linear Model Graph
> ggplot( broadband_school,aes(x=`Attainment Score`,y=`Average download
speed (Mbit/s)`)) +
```

5. Average download speed vs drug offence rates (per ten thousand people)

City of Bristol has the highest drug offense rate and the highest average download speed among the two counties. Cornwall has lower drug offense rates and lower average download speeds in comparison. The positive slope of the trend line suggests a potential relationship where higher drug offense rates are associated with higher average download speeds.

Figure 15

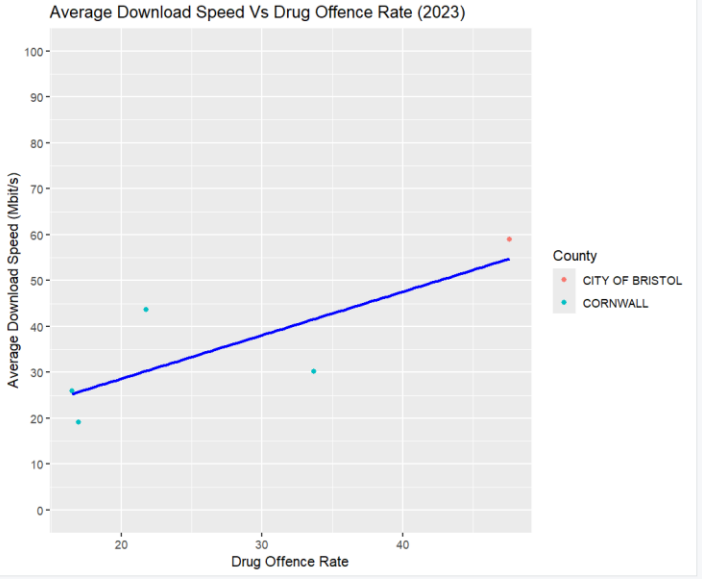
Average download speed vs drug offence rates (per ten thousand people)



```r
  mutate(`Drug Offence Rate` = (`Crime Count` / Population2023) * 10000) # Calculation of drug offence rate.

# Then, By County and Town ----grouping ----Drug Crime Dataset & Displaying the rate for each group for the year 2023.
drug_crime_grouping = crime_drugs %>%
  filter(`Crime Date`=="2023") %>%
  group_by(`Falls.within`,`Town/City`) %>%
  summarise(`Drug Offence Rate`= mean(`Drug Offence Rate`))

# Broadband and Drug crime rate data -- JOIN----in a single table
broadband_crime = broadband_speed_grouping %>%
  left_join(drug_crime_grouping,by="Town/City") %>%
  na.omit

# Create a Linear Model
# This model predicts Average download speed based on Drug offence rate
linear_model = lm(data = broadband_crime, `Average download speed (Mbit/s)`~`Drug Offence Rate`)

# Summary of Linear Model
summary(linear_model)

# Create Linear Model Graph
ggplot(broadband_crime,aes(x=`Drug Offence Rate`,y=`Average download speed (Mbit/s)`)) +
  scale_y_continuous(limits = c(0,100), breaks = seq(0,100,10))+
  # Red as City of Bristol data point.
  geom_point(data = filter(broadband_crime , County=="CITY OF BRISTOL"),aes(color=c("Red"="CITY OF BRISTOL")))+
  # Green for Cornwall data point
  geom_point(data = filter(broadband_crime , County=="CORNWALL"), aes(color=c("Green"="CORNWALL"))) +
  # LInear Regression Line adding------and "Removing error bands"
  geom_smooth(method = lm,se = FALSE,color = "blue")+
  labs(x="Drug Offence Rate",
       y="Average Download Speed (Mbit/s)",
       title="Average Download Speed Vs Drug Offence Rate (2023) ",color="County")
```

```r
  labs(x="Drug Offence Rate",
       y="Average Download Speed (Mbit/s)",
       title="Average Download Speed Vs Drug Offence Rate
```

```
> # Summary of Linear Model
> summary(linear_model)

Call:
lm(formula = `Average download speed (Mbit/s)` ~ `Drug Offence R
ate`,
    data = broadband_crime)

Residuals:
      1       2       3       4       5
 4.1644  0.6078 -6.6595 13.3250 -11.4378

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)         9.6858    12.4389   0.779    0.493
`Drug Offence Rate`   0.9472     0.4177   2.268    0.108

Residual standard error: 11.11 on 3 degrees of freedom
Multiple R-squared:  0.6316,    Adjusted R-squared:  0.5087
F-statistic: 5.142 on 1 and 3 DF,  p-value: 0.1082

>
> # Create Linear Model Graph
> ggplot(broadband_crime,aes(x=`Drug Offence Rate`,y=`Average do
wnload speed (Mbit/s)`)) +
+   scale_y_continuous(limits = c(0,100), breaks = seq(0,100,1
0))+
```



Recommendation System

Overview

In this documentation, we have developed a comprehensive recommendation system to guide potential homebuyers and investors in Bristol and Cornwall. Our analysis integrates multiple key factors including house prices, broadband speeds, school grades, and crime rates to identify the most favorable towns and cities. By normalizing and ranking each factor—where lower house prices and lower crime rates receive higher scores, and higher broadband speeds and better school grades contribute positively—we have created a unified scoring system. This system enables us to objectively assess and compare various locations based on critical quality-of-life and investment criteria.

The final recommendations are based on the combined scores from our detailed analysis. We present the top ten recommended towns or cities in Bristol and Cornwall, each selected for their exceptional attributes, to assist you in making an informed decision for your future home and investment opportunities.

Results

In our analysis, we assessed various towns and cities across City of Bristol and Cornwall based on four key factors: house pricing, broadband speed, school grades, and crime rates. The goal was to identify the towns and cities with the best overall living conditions by normalizing and combining these factors into a single score.

Top Ten Recommended Towns/Cities

Based on the combined scores, the top ten towns/cities are on the County - Cornwall:

1. Bude

House Pricing Score: 9.04

Broadband Speed Score: 3.66

School Grades Score: 4.57

Crime Score: 0

Overall Score: 4.32

Recommendation: Bude is a great place to live because it has very good house prices, excellent broadband speed, and good schools. Plus, it has no significant crime problems. Overall, it's a top choice for a comfortable and safe living environment.

2. Saltash

House Pricing Score: 4.70

Broadband Speed Score: 4.70

School Grades Score: 4.04

Crime Score: 0

Overall Score: 3.91

Recommendation: Saltash offers a good balance of affordable house prices and strong broadband speeds. It also has decent school grades and no major crime issues. It’s a solid choice for those looking for a well-rounded living environment.

3. St Austell

House Pricing Score: 7.58

Broadband Speed Score: 3.17

School Grades Score: 4.63

Crime Score: 0

Overall Score: 3.85

Recommendation: St Austell provides good house prices and school grades. The broadband speed is lower compared to the other top towns, but it still has no significant crime problems. It’s a good option if you’re looking for affordable housing and quality schools.

4. Looe

House Pricing Score: 7.84

Broadband Speed Score: 3.19

School Grades Score: 4.12

Crime Score: 0

Overall Score: 3.79

Recommendation: Looe offers excellent house pricing and school grades. The broadband speed is slightly lower compared to some other towns, but it remains favorable. With no crime issues, Looe is a strong choice for affordable living and good educational opportunities.

5. Liskeard

House Pricing Score: 7.92

Broadband Speed Score: 3.03

School Grades Score: 4.09

Crime Score: 0

Overall Score: 3.76

Recommendation: Liskeard features strong house pricing and solid school grades. While the broadband speed is somewhat lower, the absence of crime makes it an attractive location for those prioritizing affordability and education over internet speed.

6. Redruth

House Pricing Score: 7.53

Broadband Speed Score: 3.31

School Grades Score: 4.15

Crime Score: 0

Overall Score: 3.75

Recommendation: Redruth stands out with competitive house prices and good school grades. Although broadband speed is modest, the town's lack of crime and overall value make it a practical choice for residents seeking a balance between cost and quality.

7. Camborne

House Pricing Score: 8.84

Broadband Speed Score: 1.66

School Grades Score: 4.45

Crime Score: 0

Overall Score: 3.74

Recommendation: Camborne excels in house pricing and school grades, though broadband speed is significantly lower. Despite this, its zero crime rate and strong property value make it a convincing option for those who prioritize housing affordability and educational standards.

8. Penzance

House Pricing Score: 8.34

Broadband Speed Score: 2.31

School Grades Score: 4.24

Crime Score: 0

Overall Score: 3.72

Recommendation: Penzance offers very competitive house pricing and school grades, with broadband speed slightly below average. The town's low crime rate enhances its appeal for those seeking a mix of good property value and quality education.

9. Newquay

House Pricing Score: 7.35

Broadband Speed Score: 2.59

School Grades Score: 4.66

Crime Score: 0.23

Overall Score: 3.59

Recommendation: Newquay features solid house pricing and school grades with a relatively lower broadband speed. Although it has a minor crime score, the overall quality of life, including strong educational performance, makes it a suitable option.

10. St Ives

House Pricing Score: 6.27

Broadband Speed Score: 3.31

School Grades Score: 4.74

Crime Score: 0

Overall Score: 3.58

Recommendation: St Ives offers very good school grades and reasonable broadband speed. Despite its lower house pricing score, its zero crime rate and strong educational environment make it an attractive choice for families looking for a safe and educationally robust community.

Figure 16

Top Ten Towns

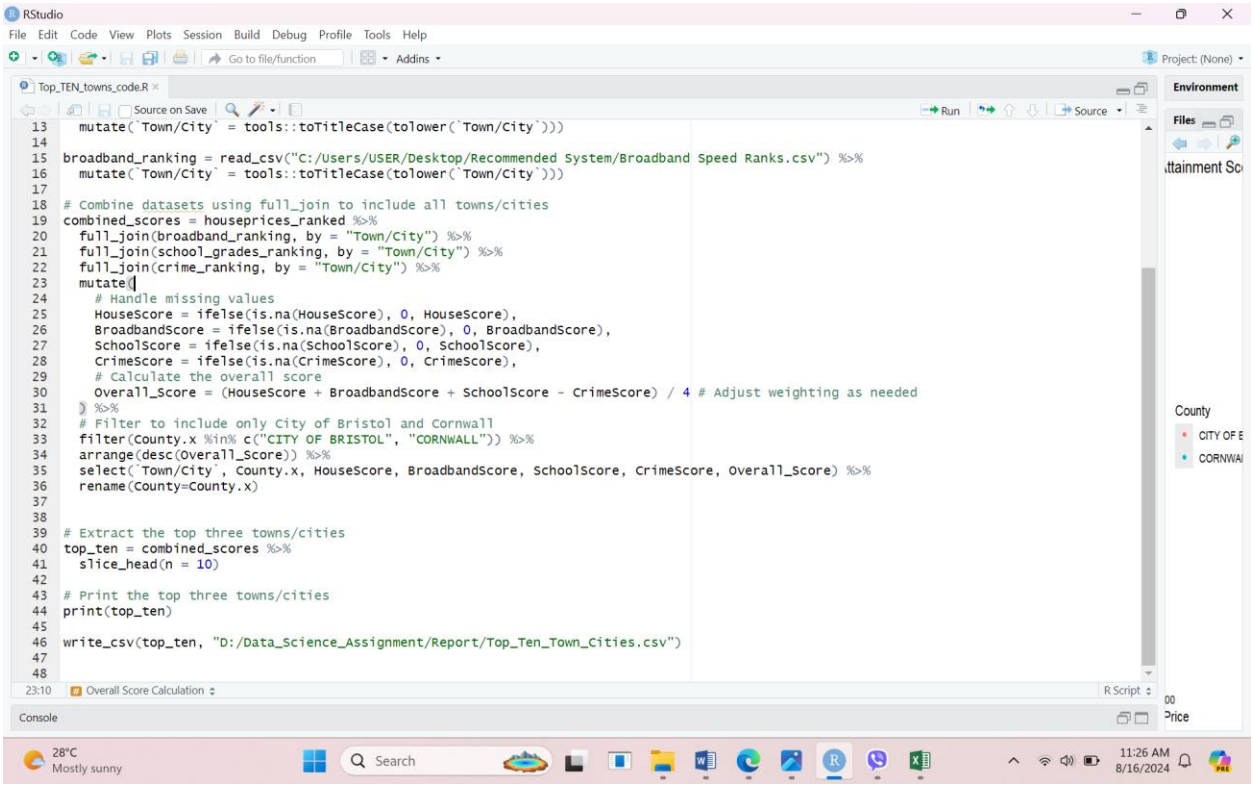


Figure 17

Top Ten Towns Output

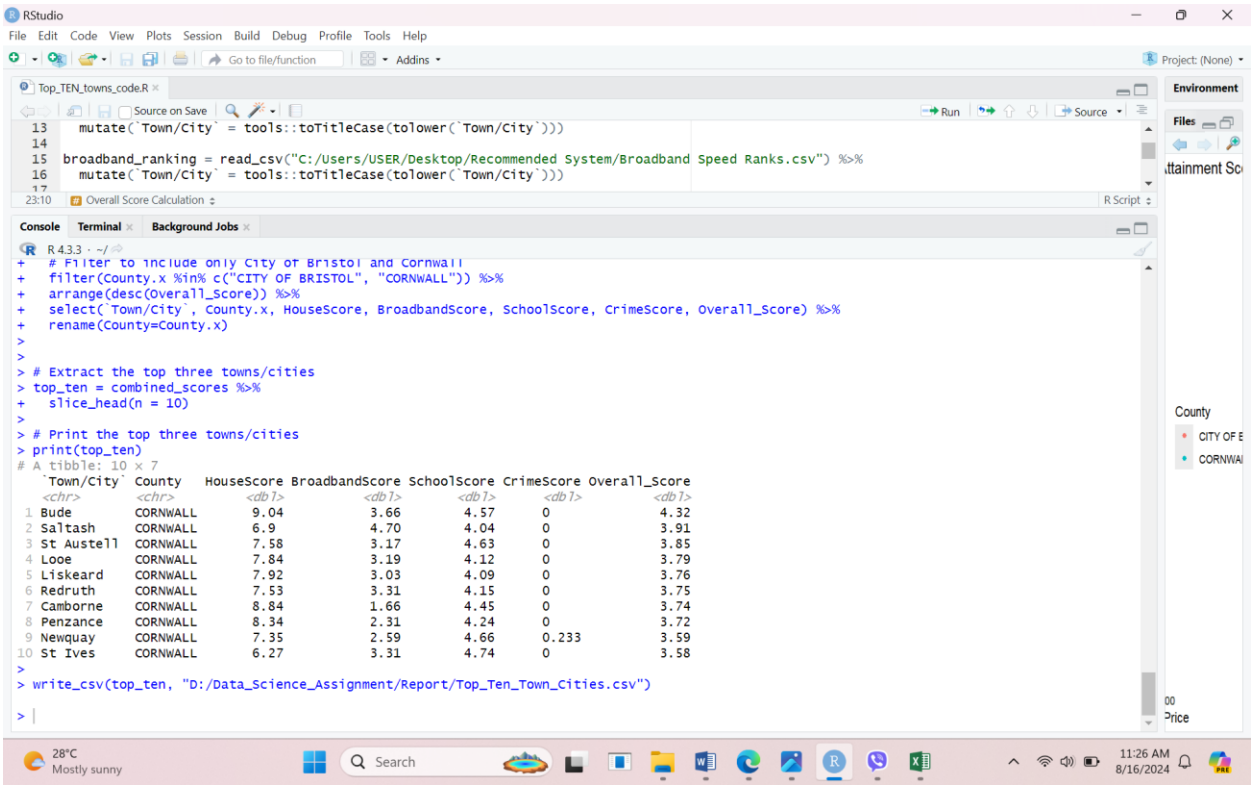
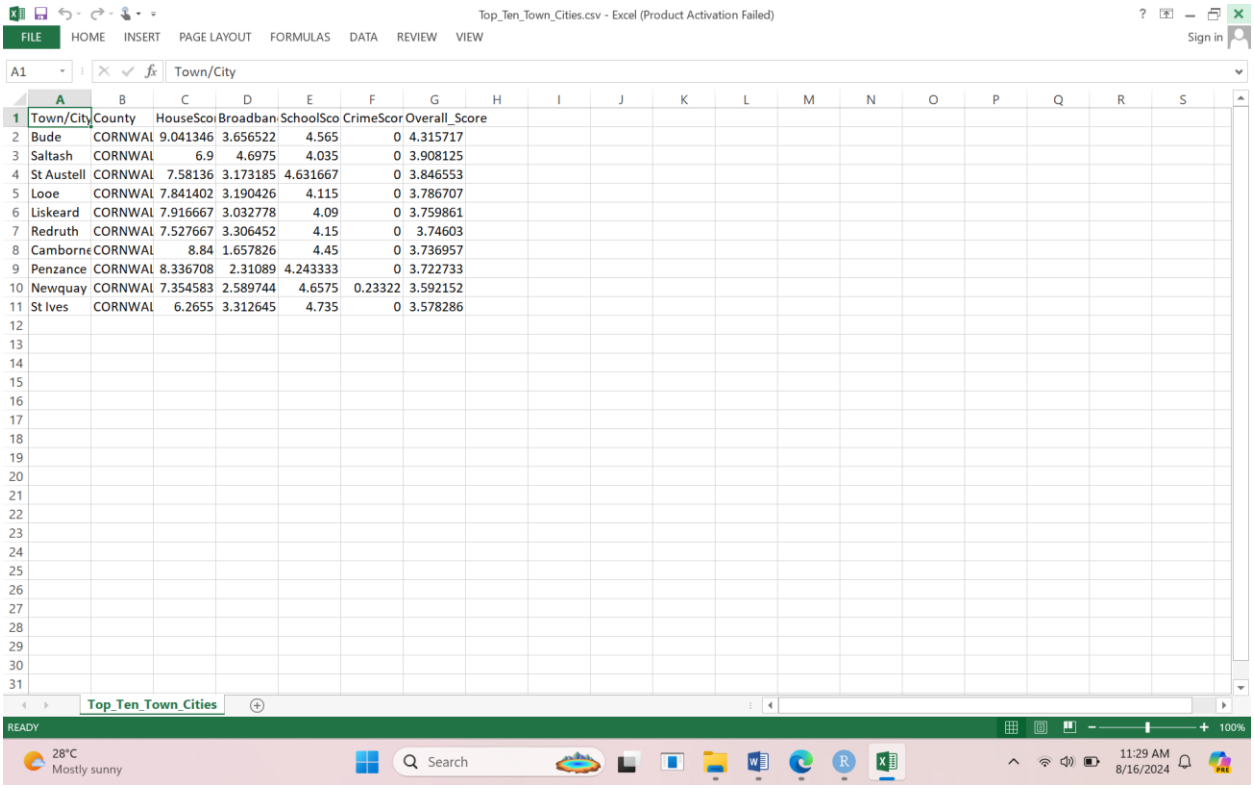


Figure 18

Top Ten Towns Excel



	A	B	C	D	E	F	G
1	Town/City	County	HouseSco	Broadban	SchoolSco	CrimeScor	Overall_Score
2	Bude	CORNWAL	9.041346	3.656522	4.565	0	4.315717
3	Saltash	CORNWAL	6.9	4.6975	4.035	0	3.908125
4	St Austell	CORNWAL	7.58136	3.173185	4.631667	0	3.846553
5	Looe	CORNWAL	7.841402	3.190426	4.115	0	3.786707
6	Liskeard	CORNWAL	7.916667	3.032778	4.09	0	3.759861
7	Redruth	CORNWAL	7.527667	3.306452	4.15	0	3.74603
8	Camborne	CORNWAL	8.84	1.657826	4.45	0	3.736957
9	Penzance	CORNWAL	8.336708	2.31089	4.243333	0	3.722733
10	Newquay	CORNWAL	7.354583	2.589744	4.6575	0.23322	3.592152
11	St Ives	CORNWAL	6.2655	3.312645	4.735	0	3.578286

Reflection

This analysis aimed to identify the top towns and cities for potential relocation or investment by evaluating key factors such as house prices, broadband speed, school grades, and crime rates. By combining these factors into a single overall score, we were able to highlight the best options according to our criteria. The insights gained are as below:

Top Choices Identified: The towns of Bude, Saltash, St Austell, Looe, Liskeard, Redruth, Camborne, Penzance, Newquay and St Ives emerged as the top ten recommendations. Bude scored the highest overall due to its strong performance in house pricing, broadband speed, and school grades, combined with a lack of significant crime issues and St Ives showed the lowest score overall.

Balanced Criteria: The use of a combined score allowed us to weigh each factor equally, offering a balanced view of each town's strengths and weaknesses.

Limitations:

Data Completeness: The analysis is limited by the data available. For instance, if some towns or cities were missing from any of the datasets because of LSOA_code being NA, then the towns associated with that Lsoa_code were also NA. So, they could not be evaluated which lead to towns/city not being present in the merged data. Overall, this analysis provides a useful starting point for evaluating potential relocation or investment options.

Overall Score

The Overall Score is a combined measure that reflects the desirability of each town or city based on four key factors: house prices, broadband speed, school grades, and crime rates. Here's how the score is calculated:

House Pricing Score: This reflects how affordable or expensive houses are in the town. Higher scores indicate better value for money.

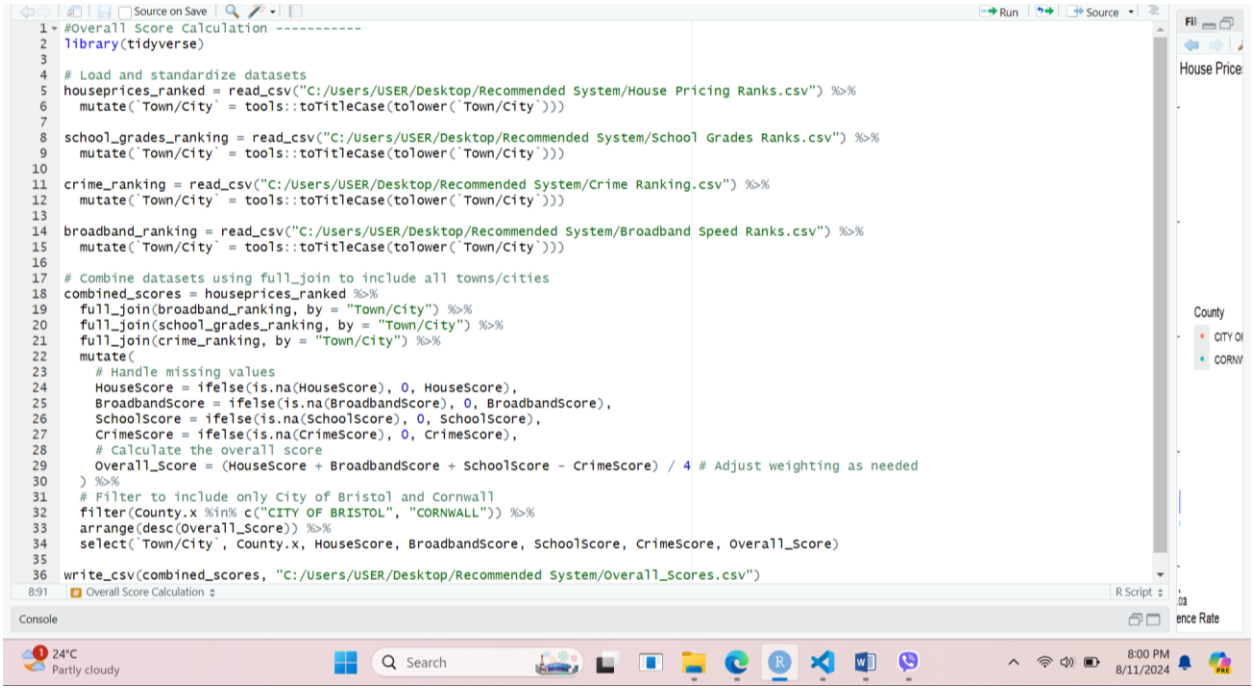
Broadband Speed Score: This measures the quality of internet service. Higher scores indicate faster and more reliable broadband.

School Grades Score: This shows the quality of education in the area. Higher scores represent better school performance.

Crime Score: This reflects the safety of the town. Higher scores indicate higher crime rates, which is treated as a negative factor. We subtract this score to account for the impact of crime on the overall desirability.

Figure 19

Overall Score



```
#Overall Score Calculation -----------
library(tidyverse)

# Load and standardize datasets
houseprices_ranked = read_csv("C:/Users/USER/Desktop/Recommended System/House Pricing Ranks.csv") %>%
  mutate(`Town/City` = tools::toTitleCase(tolower(`Town/City`)))

school_grades_ranking = read_csv("C:/Users/USER/Desktop/Recommended System/School Grades Ranks.csv") %>%
  mutate(`Town/City` = tools::toTitleCase(tolower(`Town/City`)))

crime_ranking = read_csv("C:/Users/USER/Desktop/Recommended System/Crime Ranking.csv") %>%
  mutate(`Town/City` = tools::toTitleCase(tolower(`Town/City`)))

broadband_ranking = read_csv("C:/Users/USER/Desktop/Recommended System/Broadband Speed Ranks.csv") %>%
  mutate(`Town/City` = tools::toTitleCase(tolower(`Town/City`)))

# Combine datasets using full_join to include all towns/cities
combined_scores = houseprices_ranked %>%
  full_join(broadband_ranking, by = "Town/City") %>%
  full_join(school_grades_ranking, by = "Town/City") %>%
  full_join(crime_ranking, by = "Town/City") %>%
  mutate(
    # Handle missing values
    HouseScore = ifelse(is.na(HouseScore), 0, HouseScore),
    BroadbandScore = ifelse(is.na(BroadbandScore), 0, BroadbandScore),
    SchoolScore = ifelse(is.na(SchoolScore), 0, SchoolScore),
    CrimeScore = ifelse(is.na(CrimeScore), 0, CrimeScore),
    # Calculate the overall score
    Overall_Score = (HouseScore + BroadbandScore + SchoolScore - CrimeScore) / 4 # Adjust weighting as needed
  ) %>%
  # Filter to include only City of Bristol and Cornwall
  filter(County.x %in% c("CITY OF BRISTOL", "CORNWALL")) %>%
  arrange(desc(Overall_Score)) %>%
  select(`Town/City`, County.x, HouseScore, BroadbandScore, SchoolScore, CrimeScore, Overall_Score)

write_csv(combined_scores, "C:/Users/USER/Desktop/Recommended System/Overall_Scores.csv")
```

Legal and Ethical Issues

In conducting this analysis, it is essential to address both legal and ethical considerations. The datasets used for this study are publicly available, which means they have been provided by governmental or authorized entities for public access and use. Utilizing publicly available datasets ensures compliance with legal requirements concerning data access and use.

Conclusion

This analysis used the data mining lifecycle to assess towns and cities based on house prices, broadband speed, school grades, and crime rates. Bude, Saltash, St Austell, Looe, Liskeard, Redruth, Camborne, Penzance, Newquay and St Ives as top recommendations due to their overall favorable scores by integrating these factors into a combined score. The analysis was based on publicly available datasets, ensuring relevance and accessibility.

References

GeeksforGeeks. (n.d.). *Data Cleaning in R - GeeksforGeeks*. Retrieved from https://www.geeksforgeeks.org/data-cleaning-in-r/

Government, U. (n.d.). *House pricing dataset. UK Government Statistical Data Sets*. Retrieved from https://www.gov.uk/government/statistical-data-sets/price-paid-data-downloads

Ofcom. (n.d.). *Broadband speed data*. Retrieved from https://www.ofcom.org.uk/research-and-data/multi-sector-research/infrastructure-research/connected-nations-2018/data-downloads

Police, U. (n.d.). *Crime dataset*. Retrieved from https://data.police.uk/data/

School dataset, UK Government Compare School Performance Service. (n.d.). Retrieved from https://www.compare-school-performance.service.gov.uk/download-data?currentstep=datatypes&regiontype=all&la=0&downloadYear=2018-2019&datatypes=ks5

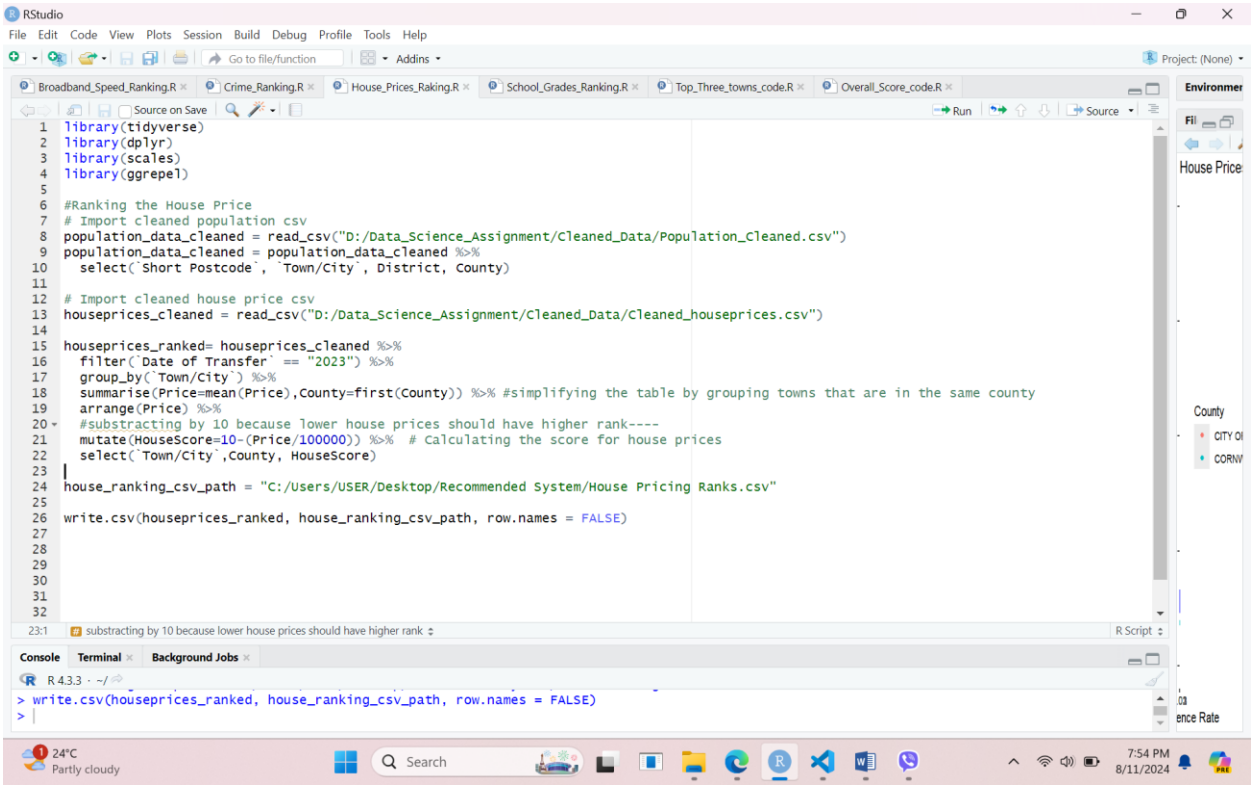
Wickham, H. (n.d.). *Exploratory Data analysis. R for Data Science*. Retrieved from https://r4ds.had.co.nz/exploratory-data-analysis.html

Appendix

1. House Prices Ranking

Figure 20

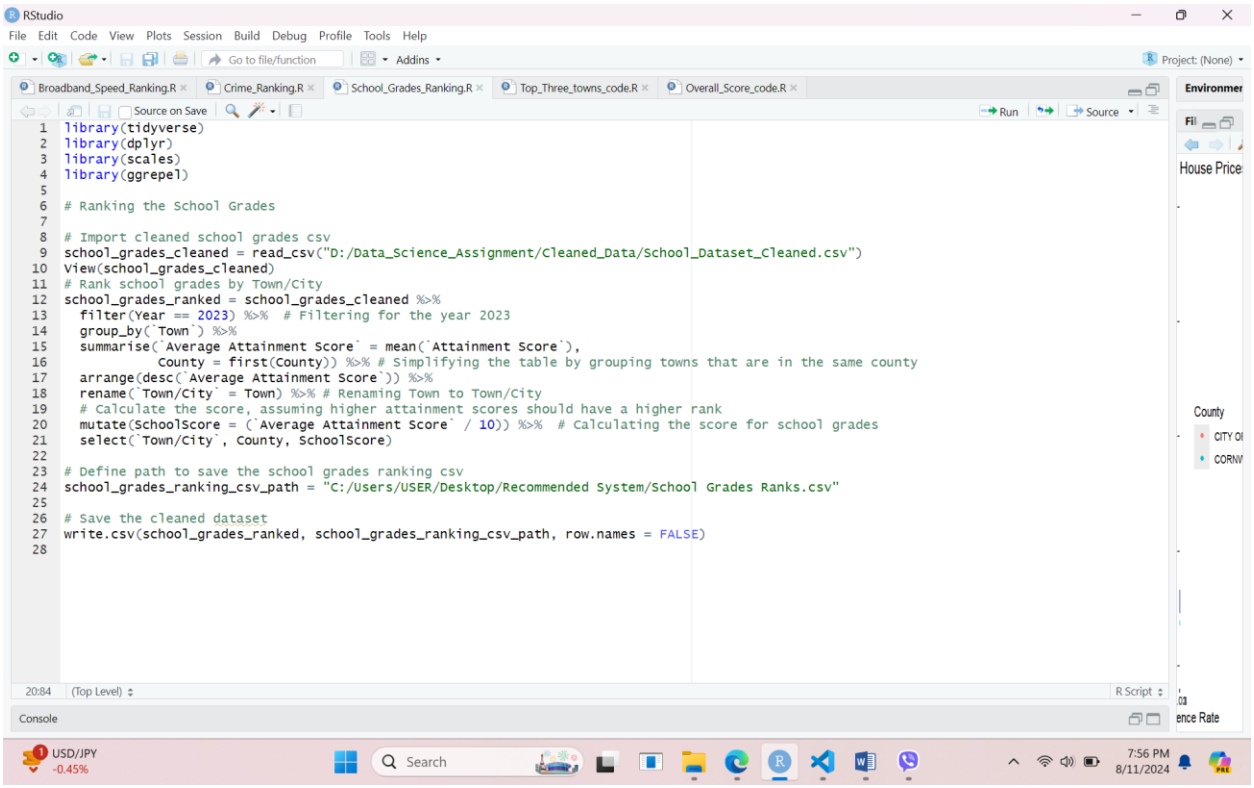
House Prices Ranking



2. School Grades Ranking

Figure 21

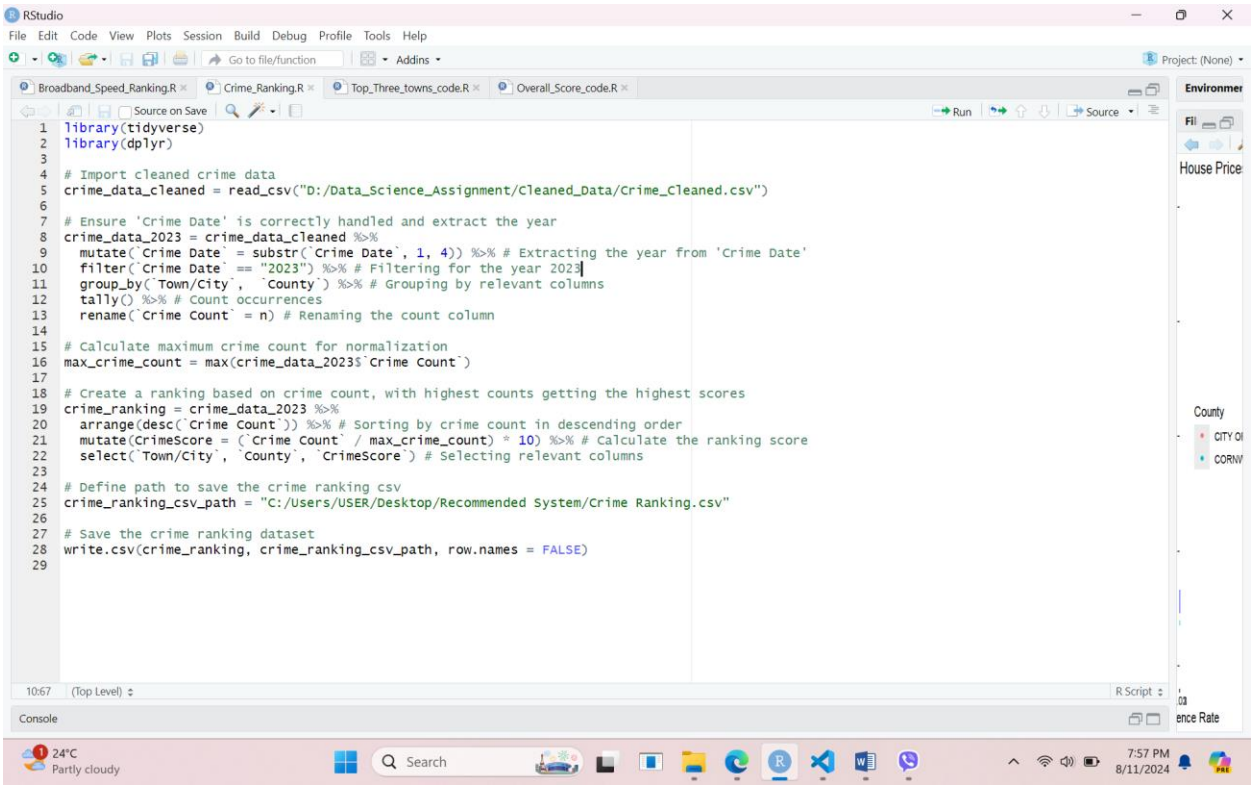
School Grades Ranking



3. Crime Ranking

Figure 22

Crime Ranking



4. Broadband Ranking

Figure 23

Broadband Speed Ranking