ST5014CEM

Data Science for Developers

Coursework: Individual Report

Nishan Bista

230348

Submitted To

Siddhartha Neupane

Table of Contents

Table of Figures

Introduction

This assignment presents the process and outcomes of using data analysis to recommend the most suitable location to purchase a property in the United Kingdom, specifically comparing South Yorkshire and West Yorkshire. Both regions offer their own benefits, but making a property investment requires a detailed analysis of several factors that affect housing value and quality of life. Key aspects considered in this project include house prices, crime rates, broadband connectivity, education, and access to public amenities.

The main goal of this project is to build a simple recommendation system that uses publicly available UK government data to evaluate and compare towns in South and West Yorkshire. Based on selected criteria, each town receives a score which helps identify the most promising areas for investment. This report outlines the steps followed throughout the project, including data collection, cleaning, integration, exploratory data analysis, and linear modeling to support the final recommendation.

Cleaning Data

Data cleaning is a crucial step in the data science lifecycle that comes right after collecting the datasets. It helps ensure the data is accurate, consistent, and ready for analysis. The datasets used in this project, which were sourced from UK government portals and public sources, contained various inconsistencies such as missing values, formatting issues, and mismatched fields.

Each dataset was carefully cleaned and standardized to make sure they could be merged and analyzed together. This included removing duplicates, handling missing or invalid entries, formatting columns like postcodes, and filtering the data to include only relevant information from South Yorkshire and West Yorkshire. Cleaning the data properly was important to ensure reliable results and support the development of the final recommendation system.

House Data

The house price data from 2021 to 2024 was combined into a single dataset using the bind_rows() function in R. Since the original files did not have headers, column names were manually assigned based on the UK Price Paid Data structure. The postcode column was cleaned by trimming white spaces and converting all values to uppercase for consistency. Then, the dataset was filtered to include records only from South Yorkshire and West Yorkshire. Finally, the cleaned dataset was saved using the write_csv() function for further analysis.

Town and Postcodes

To map postcodes to their corresponding counties and LSOA codes, two datasets were used: one containing postcode-to-LSOA mappings and another with county information from the UK geoportal. These datasets were merged using the inner_join() function in R based on the local authority district code. The merged data was then cleaned by removing unnecessary columns and formatting the postcode column by removing spaces. Finally, the dataset was filtered to include only entries from South Yorkshire and West Yorkshire, and the result was saved as a CSV file for later use.

Broadband Data

Two separate broadband datasets were used, one for performance and one for coverage. These were cleaned by removing missing or invalid postcodes and standardizing them to uppercase with no spaces. The two datasets were then joined using the postcode field to combine performance and coverage information.

To link broadband data with location, a postcode_to_LSOA mapping data was used. After standardizing the postcodes in both datasets, they were merged to include LSOA codes. Further, the dataset was joined with a postcode-county mapping to isolate data for South Yorkshire and West Yorkshire. Irrelevant columns were removed to keep only the necessary features for analysis. Finally, the cleaned and filtered broadband dataset was saved for further use.

Crime Data

The crime data was collected from multiple CSV files stored in different folders, each representing a specific month and county. These files were combined into one dataset using a custom function that extracted metadata such as the year, month, and county name from the file names. Only the necessary columns were selected, including crime type, location, and LSOA code.

To enrich the dataset, a postcode-to-LSOA mapping was used to assign a postcode to each LSOA. Then, population data was added by matching postcodes, allowing crime rates to later be normalized by population. The final cleaned dataset includes crime, location, and population data for South Yorkshire and West Yorkshire, and was saved as a CSV file for further analysis.

School Data

School performance data from 2021 to 2024 was collected and combined into a single dataset. For each year, only the relevant columns were selected, including school name, postcode, town, and the Attainment 8 score. A new column was added to indicate the academic year.

The Attainment 8 score column was cleaned by removing non-numeric values such as "NE" and "SUPP", and then converted to numeric for analysis. Postcodes were trimmed to ensure consistency, and the dataset was joined with a postcode-county mapping to add county information. Finally, the data was filtered to include only schools located in South Yorkshire and West Yorkshire, and saved as a cleaned CSV for further use.

Exploratory Data Analysis

Exploratory Data Analysis (EDA) is an important step in understanding the underlying patterns and trends in a dataset. After cleaning and preparing the data, different visualization techniques were used to explore the key features across housing prices, crime rates, broadband availability, and school performance in South Yorkshire and West Yorkshire. Graphs such as bar charts, line plots, and box plots were used to help compare distributions, identify relationships, and highlight differences between the two regions. These insights supported the development of a data-driven recommendation system.

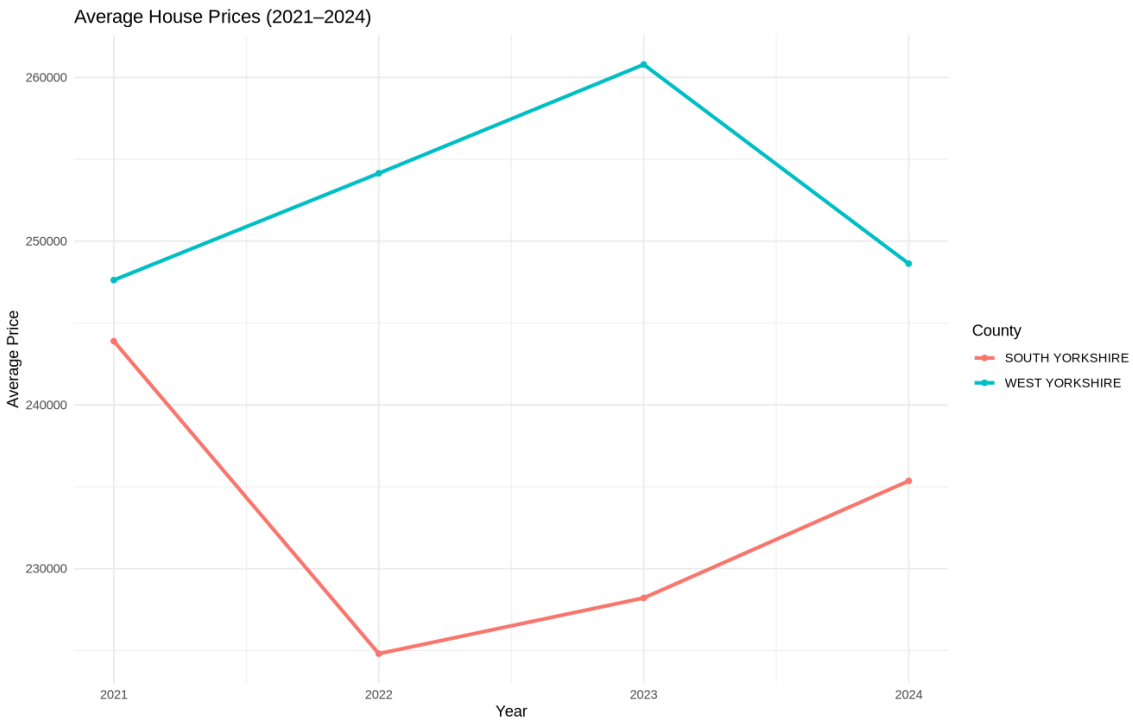
Visualization for house prices

Line Chart: Average House Prices from 2021 to 2024

To understand how house prices have changed over time, the dataset was first filtered to include the years from 2021 to 2024. The records were then grouped by year and county using the group_by() function, and the average house price was calculated with summarise(). The line chart was created using the ggplot2 library where the x-axis represents the year, and the y-axis shows the average house price.

From the line chart, we can clearly see how prices have shifted in West Yorkshire and South Yorkshire during these years. In general, South Yorkshire house prices have increased gradually from the past 2021 to 2024 . Whereas West Yorkshire house prices have decreased from 2023. However, the rate of growth in South Yorkshire appears to be more stable, while West Yorkshire shows slight fluctuations across 2023 to 2024.

Figure 1, Average house prices Line chart

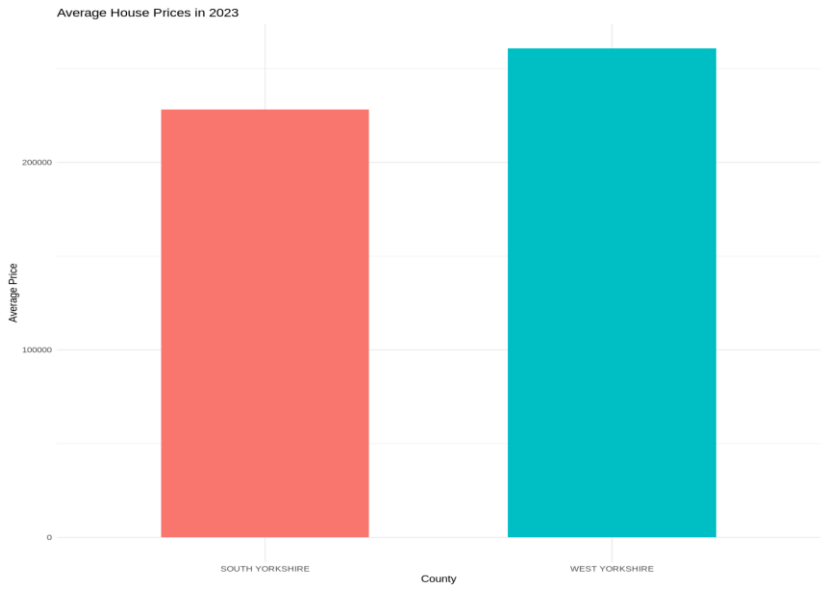


Bar Chart: Average House Price in 2023

For this chart, the dataset was filtered to include only transactions from the year 2023. Then, using group_by() and summarise(),. A bar chart was created using the ggplot2 package with the counties on the x-axis and the average price on the y-axis.

This bar chart gives a quick comparison of house prices in West Yorkshire and South Yorkshire for 2023. The chart shows that West Yorkshire had a slightly higher average price compared to South Yorkshire that year. This could be due to higher property values in certain towns within West Yorkshire or possibly due to different property types being more common in one area over the other.

Figure 2, Average House price Bar chart



Boxplots: Distribution of House Prices by County (District)

Boxplots was used to explore the spread and variation in house prices for both West Yorkshire and South Yorkshire. The dataset includes all years, and the prices were grouped by county. Separate boxplots were created for each county to better understand the distribution of housing prices.

The boxplots show a clear comparison of price ranges in both counties via district. South Yorkshire's prices seem to have a wider range, with some high outliers, suggesting that there might be certain towns or property types that are significantly more expensive. In contrast, West Yorkshire’s distribution looks more compact, with most prices clustering around near the Q3 suggesting that the majority of house prices are relatively low compared to South Yorkshire, with only a few scattered outliers at much higher values. These visualizations help us understand not just the averages but also the variability and extremities in the housing market across both counties.

Figure 3, House by districts west Box plot

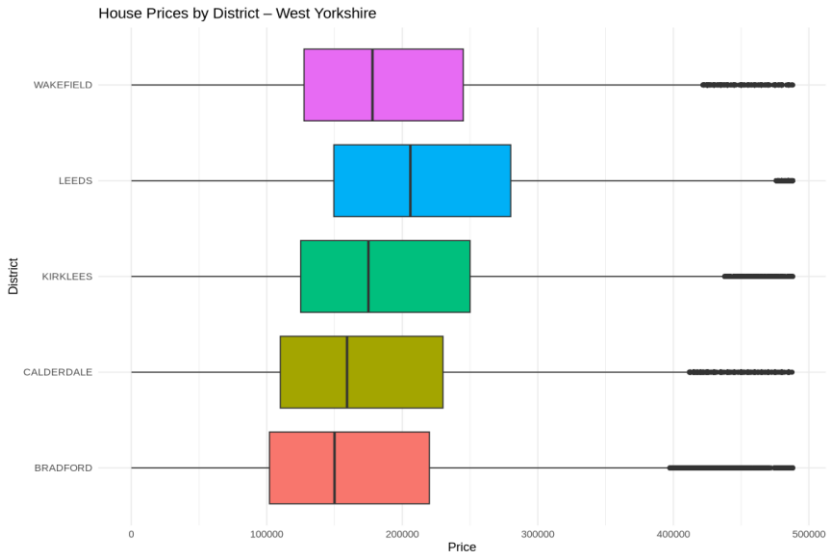
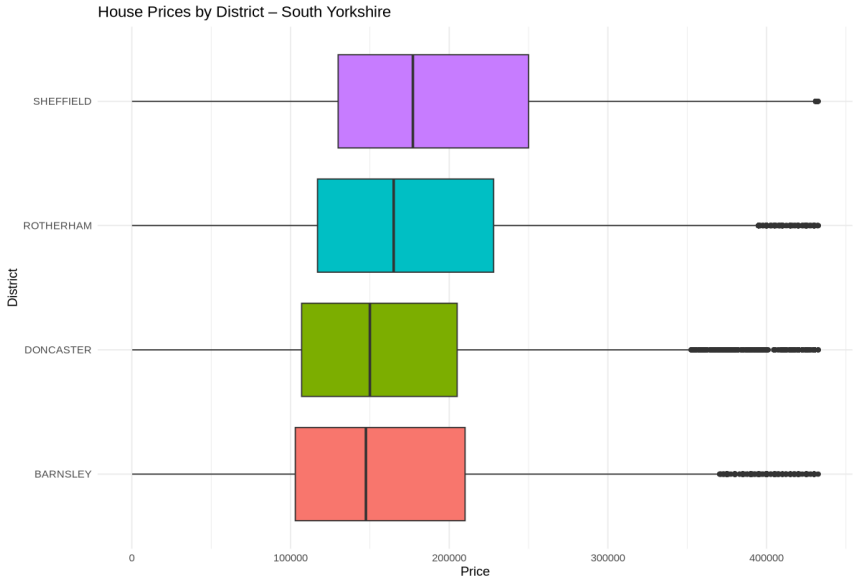


Figure 4, House by districts South Box plot



Visualisation for Broadband speed data

Boxplots: Average download speed by county

The clean broadband data is loaded from the csv file two boxplots are generated to visualize the variation in average download speeds across different districts within West Yorkshire and South Yorkshire. First, it filters the dataset for each county separately using the filter() function. Then, ggplot() is used to set up the plots, geom_boxplot() creates the boxplot. This visualization helps in identifying differences in internet speed distribution among districts within each county.

In West Yorkshire, All districts display a wide range of download speeds with numerous outliers, indicating the presence of faster connections in some areas.

In South Yorkshire, The median speeds appear similar across districts, all districts show a long right tail with numerous outliers indicating higher speeds, reaching up to 200 Mbps.

Figure 5, Average Download speed West Box plot

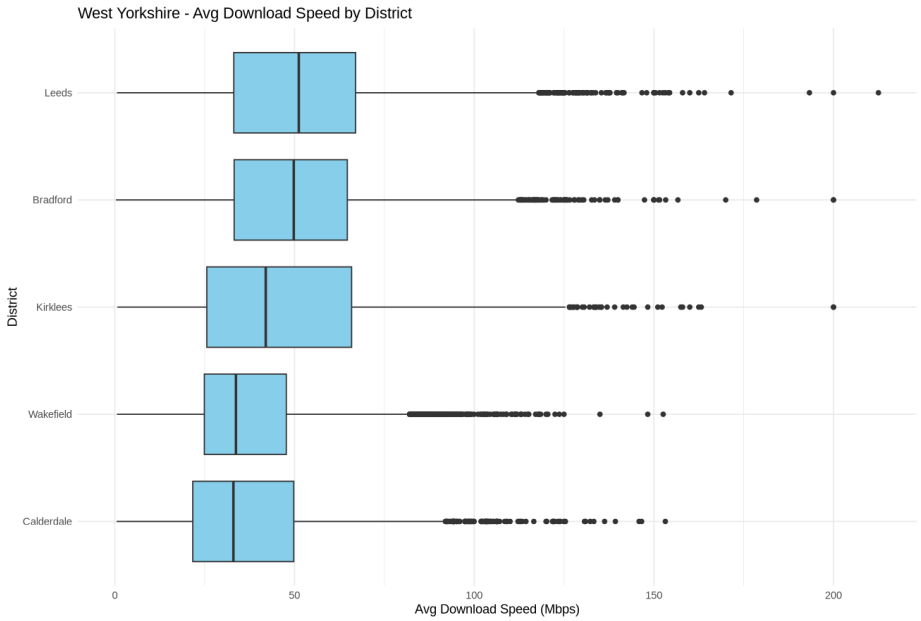
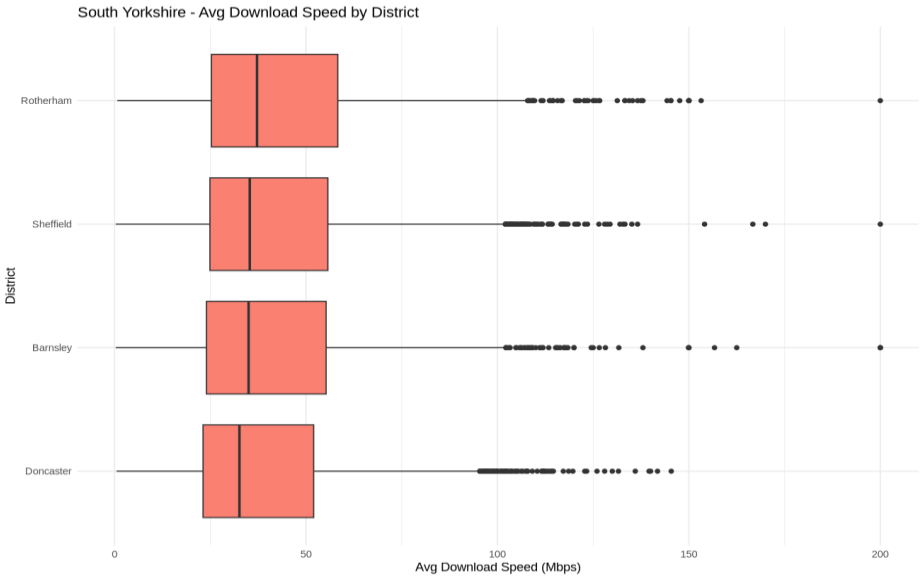


Figure 6, Average Download speed South box plot



Bar charts: Town vs Average Speed

This helps us to compare the cities in West Yorkshire and South Yorkshire based on average download speed. The data is grouped by district name (ladnm) using group_by(), and the mean download speed is calculated with summarise(), excluding any missing values. The results are sorted in descending order with arrange(). The ggplot() function then constructs a bar chart, where towns are reordered by average speed for clarity. These visuals help quickly identify which towns offer faster internet speeds within each region.

In West Yorkshire, Leeds is the town with the highest average broadband speed with around 52 Mbps followed by Bradford and Kirklees simultaneously.

In South Yorkshire, Rotherham Town has the highest average broadband speed with around 45 mbps. It is followed by Sheffield and Barnsley respectively with around average broadband speed of 42 Mbps

Figure 7, West Towns by Average Download speed

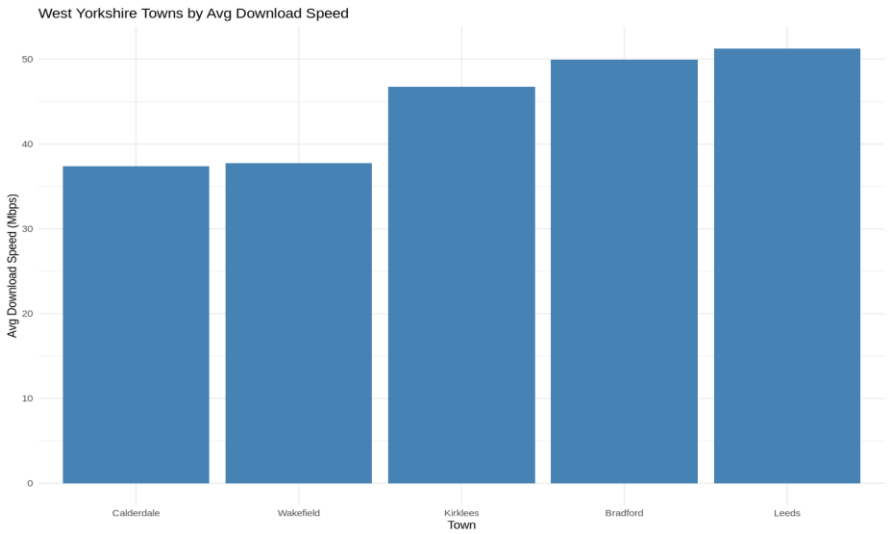
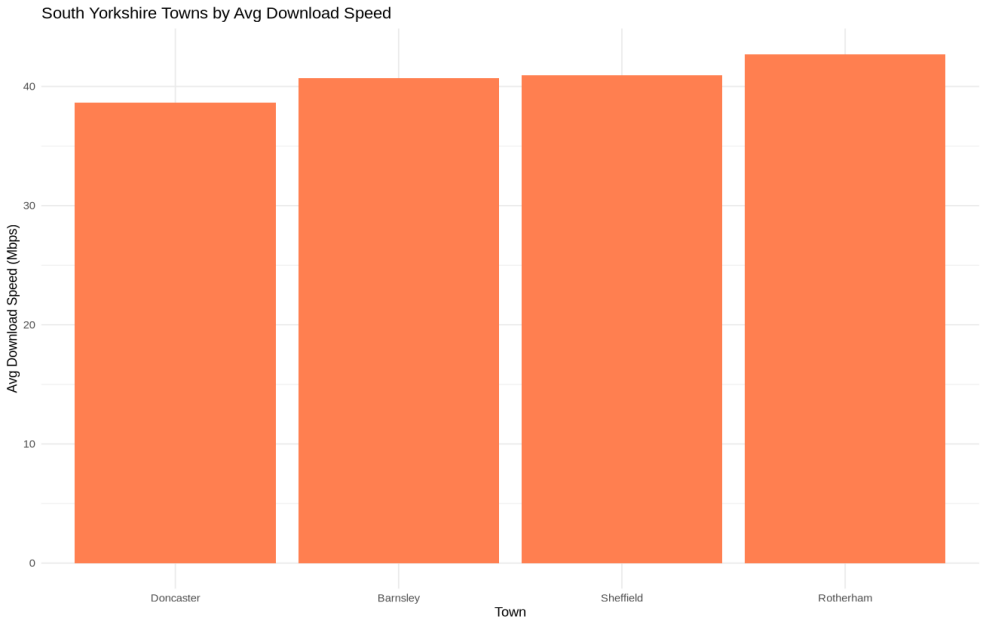


Figure 8, South Towns by Average Download speed



Visualization for crime data

Boxplot: Drug offense rate in the district of both counties

This boxplot compares drug offense rates in South and West Yorkshire. In South Yorkshire, Sheffield has the highest and most variable rates, while Barnsley shows the lowest and most stable. Rotherham and Doncaster fall in between. In West Yorkshire, Bradford leads with the highest and widest spread, followed by Leeds with a slightly more consistent rate. Wakefield stands out with the lowest rates, and Kirklees and Calderdale sit somewhere in the middle. Overall, Sheffield and Bradford seem to be the most affected, while Barnsley and Wakefield are the least.

Figure 9, Drug offense Rate South Districts

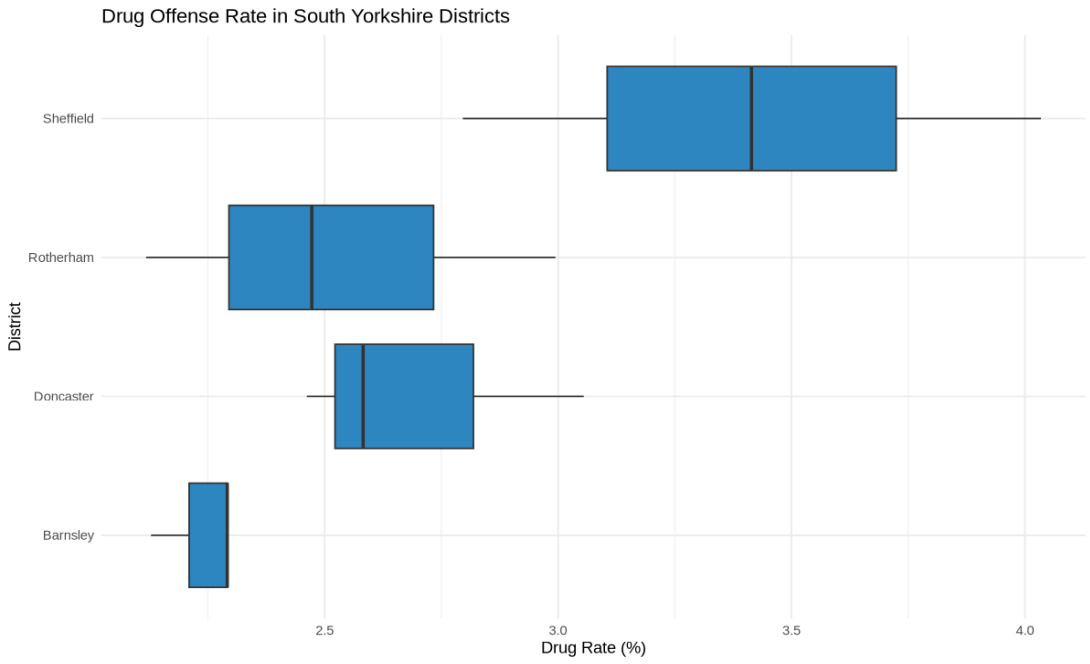
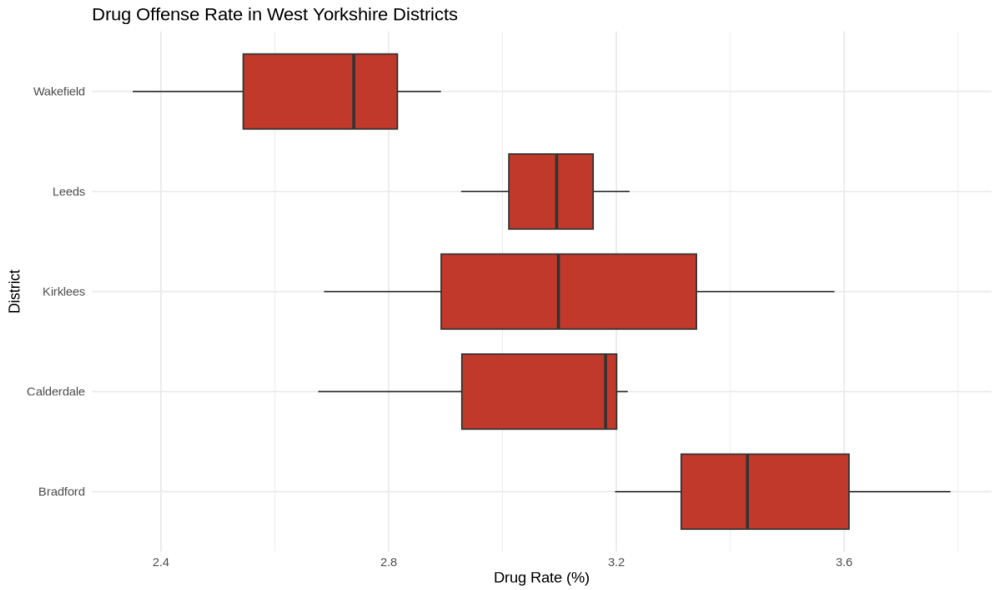


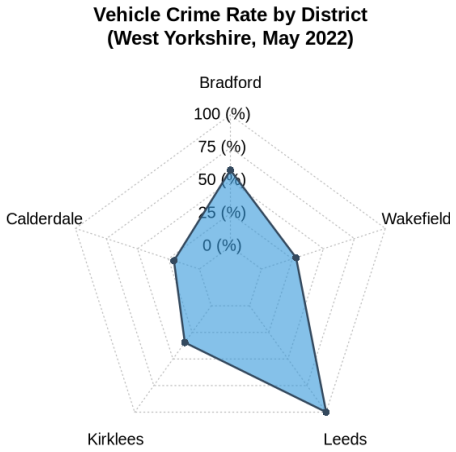
Figure 10, Drug offense Rate West districts



Radar chart: for Vehicle crime rate for West Yorkshire for May 2022

This radar chart visualizes vehicle crime rates across five districts in West Yorkshire for May 2022. Bradford and Leeds show the highest rates, indicating greater levels of vehicle-related crimes in these areas. In contrast, Kirklees, Calderdale, and Wakefield have noticeably lower rates, with Calderdale being the least affected. The chart highlights the regional disparity in vehicle crime, with Bradford standing out as the most impacted district.

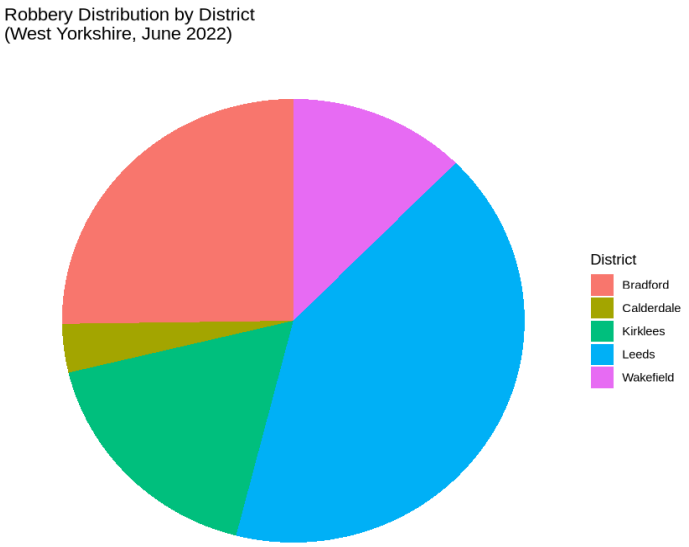
Figure 11, Vehicle Crime Radar chart



Pie chart: for Robbery rate for any one of two counties (for any specific month and year)

This pie chart shows how robbery cases were distributed across West Yorkshire districts in June 2022. Leeds had the largest share, followed by Bradford and Kirklees. Wakefield contributed a smaller portion, and Calderdale had the lowest share of robberies. The chart clearly highlights Leeds as the most affected district during this period.

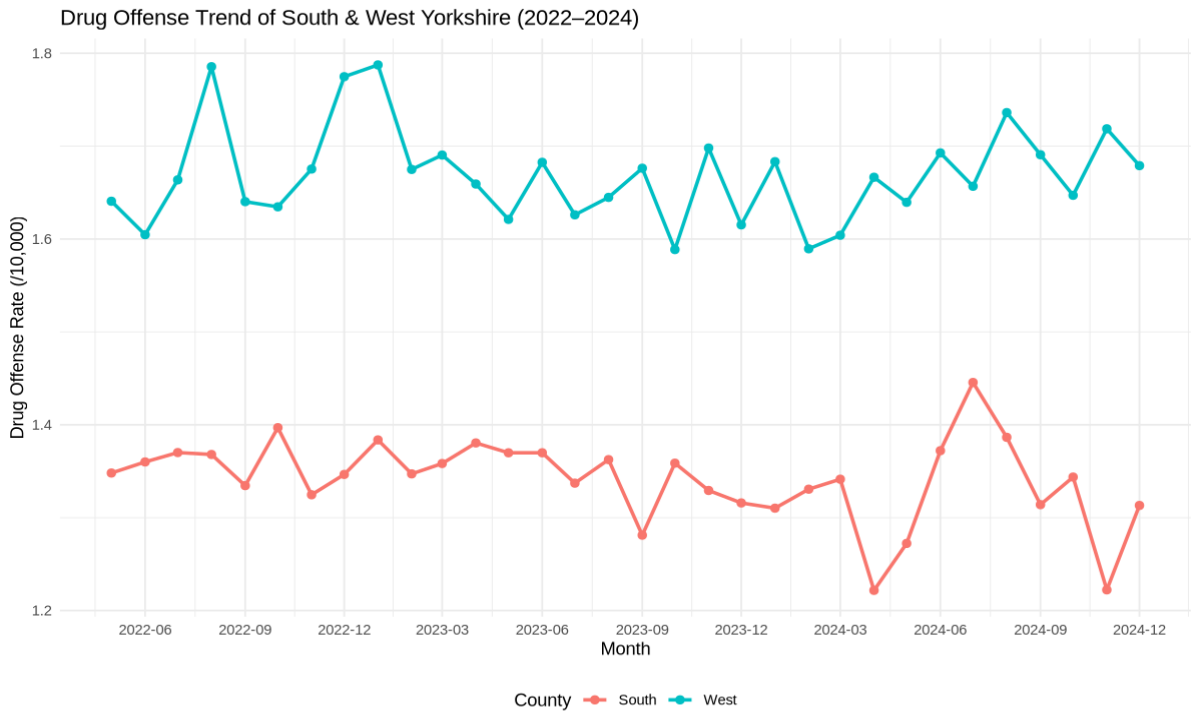
Figure 12, Robbery Distribution Pie chart



Line chart for Drug offense rates per 10,000 people for both counties for all years.

This line chart shows the monthly drug offense rates in South and West Yorkshire from 2022 to 2024. West Yorkshire consistently had higher rates than South Yorkshire throughout the period. While both regions experienced some fluctuations, West Yorkshire's trend remained more stable, whereas South Yorkshire showed more noticeable ups and downs over time.

Figure 13, Drug offense trend Counties



Visualization for school data

Boxplot: Average attainment 8 score 2022

The boxplot shows the distribution of the *Average Attainment 8 Score* for schools in 2022. This score reflects student performance across 8 key subjects. This will help us to know which district has better attainment scores in both counties.

Figure 14, Attainment 8 score South Box plot

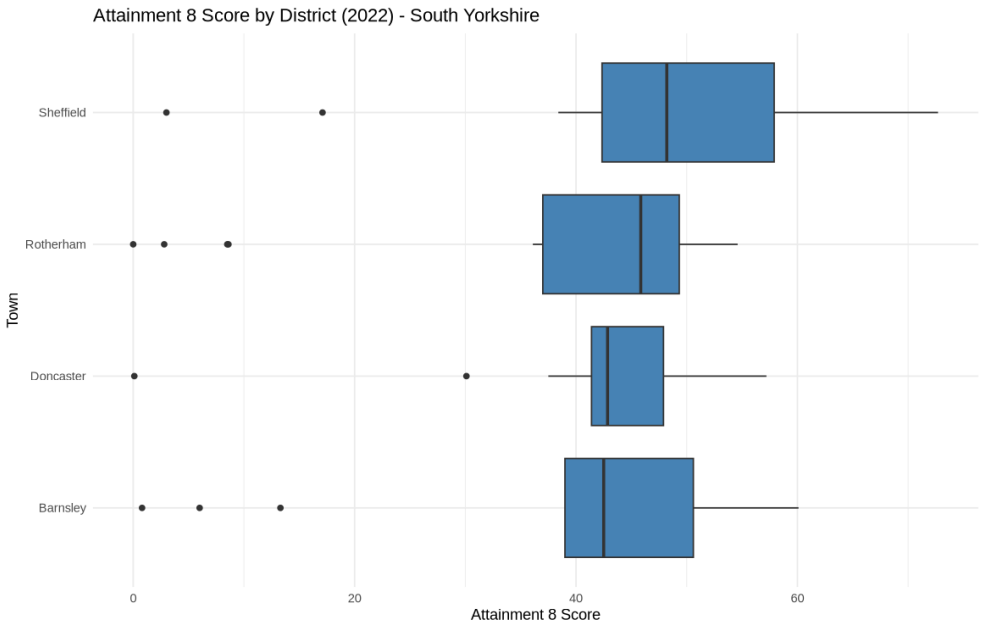
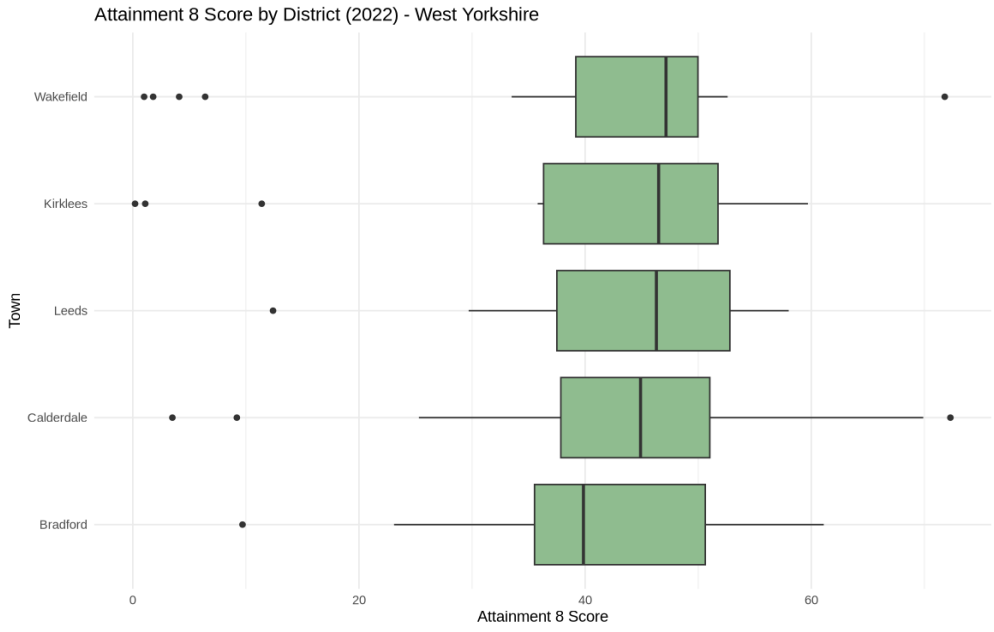


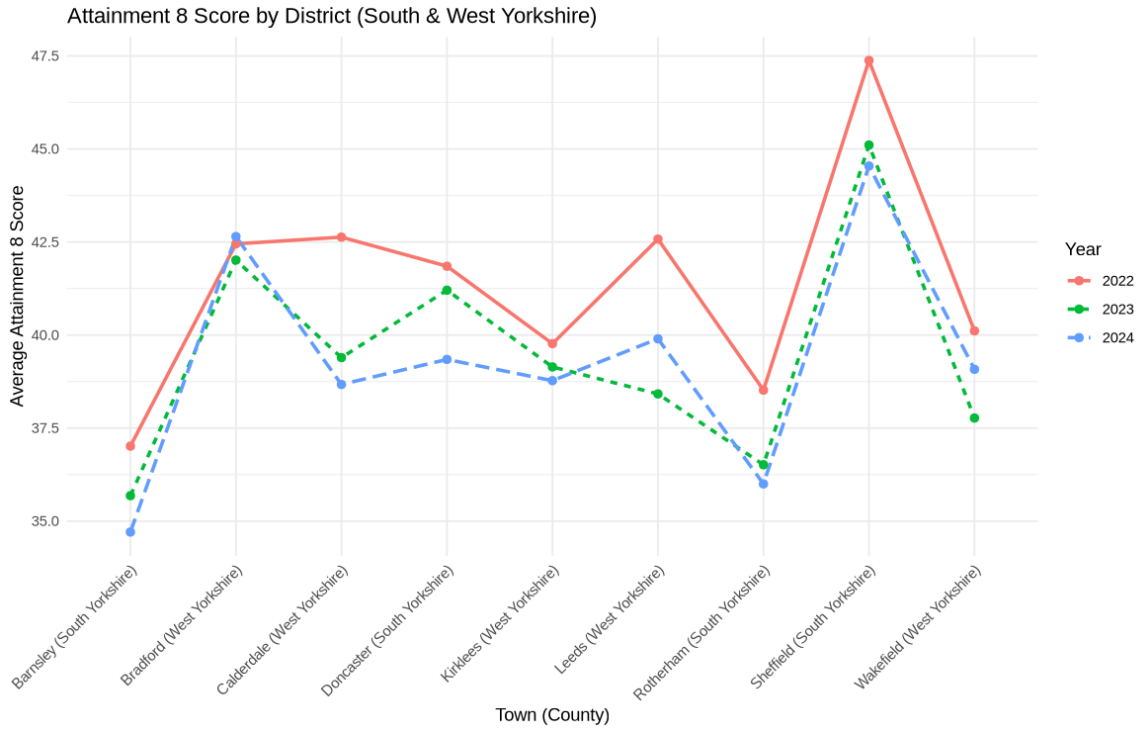
Figure 15, Attainment 8 score West Box plot



Line Graph: Relationship between attainment 8 score and years over multiple districts in South and West Yorkshire

From the line graph below, We can analyze that in three years Sheffield has a high attainment score whereas Barnsley has significantly low attainment score over the year. Sheffield also has the more consistent attainment score among other districts

Figure 16, Attainment 8 score South and West Line chart



Linear Modeling

Linear modeling is an easy to use yet potent statistical tool used for investigating the relationships between two or more variables. In this project, it was used to demonstrate trends and patterns between house price, broadband speed, crime rates, and school performance in both South Yorkshire and West Yorkshire.

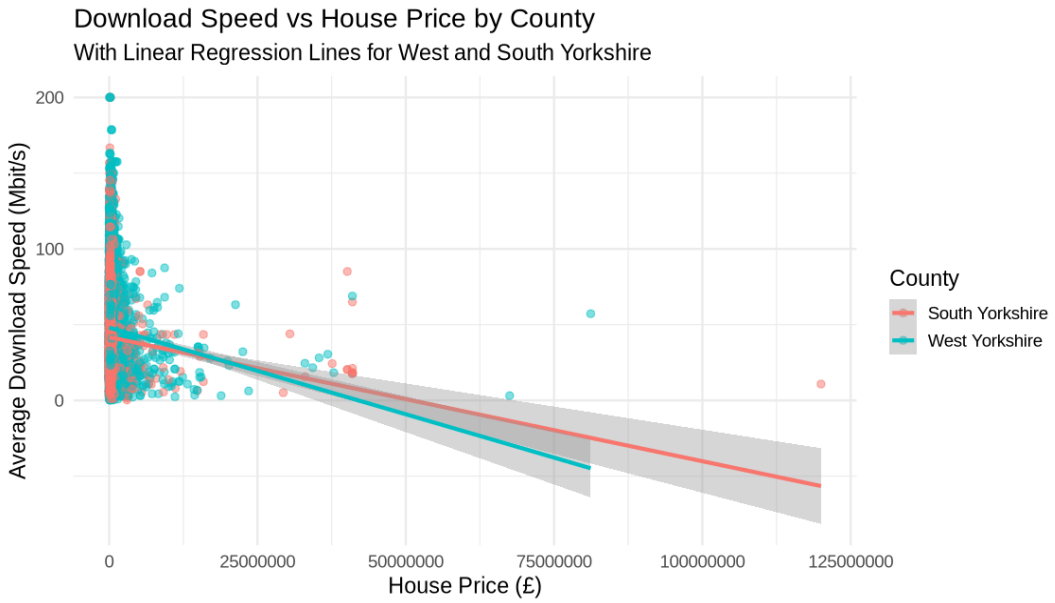
By Using a linear regression analysis allowed us to look at the shift when one variable (house price) changes with another (like average broadband speed or crime rate). For example, we assessed if areas with higher drug offense had lower house prices, or if the value of housing was linked with better school or college performance. We were able to go through these models and assess which were correlated and/or able to compare districts in an informed way, so we could address our top line conclusion on which district was a good place to buy a residential property.

House Price vs Download Speed for both Counties

Linear regression analysis carried out to analyze the relationship between house prices, and average download speed of broadband in West and South Yorkshire. The correlation in both regions was very weak and negative (−0.026 in West Yorkshire and −0.027 in South Yorkshire), which indicates there is almost no relationship at all.

While the regression models were statistically significant ($p < 0.001$), the effect size was negligible, download speed dropped by less than 0.000001 Mbps for every £1 increase in house price. The R-squared values were close to 0, suggesting that house prices explain almost no variation in download speed.

Figure 17, Download speed vs House Price



Linear Model summary & correlation

Figure 18, LM 1 West Summary

```
Call:
lm(formula = download_speed ~ price, data = west)

Residuals:
    Min      1Q  Median      3Q     Max
-47.702 -17.848  -1.575  16.136 152.173

Coefficients:
                Estimate    Std. Error t value            Pr(>|t|)
(Intercept) 48.0619745271  0.0679648863 707.159 <0.0000000000000002 ***
price       -0.0000011440  0.0000001212  -9.442 <0.0000000000000002 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 22.46 on 131414 degrees of freedom
Multiple R-squared:  0.0006779, Adjusted R-squared:  0.0006703
F-statistic: 89.15 on 1 and 131414 DF,  p-value: < 0.00000000000000022

> cat("Correlation (West Yorkshire):", cor_west, "\n")
Correlation (West Yorkshire): -0.02603654
```

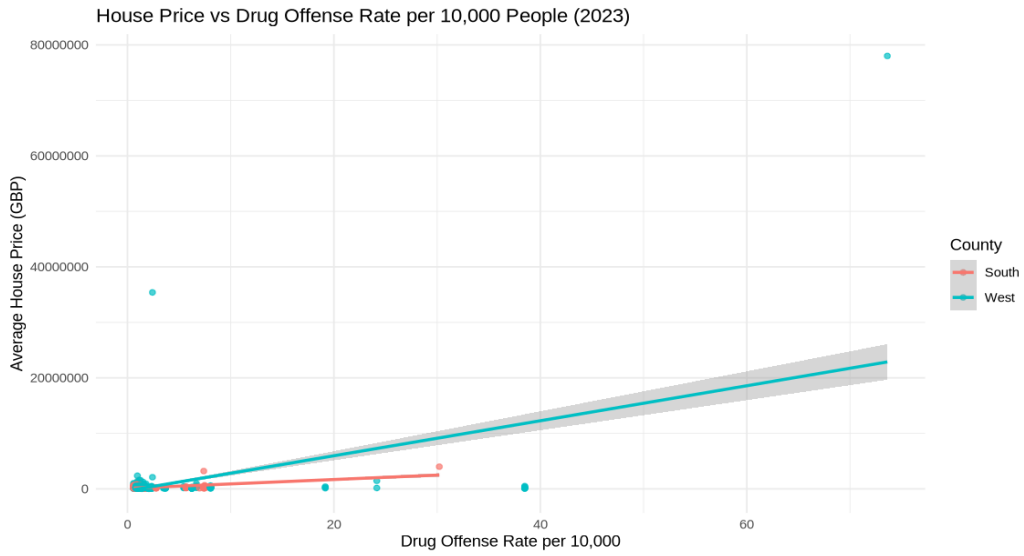
Figure 19, LM 1 South Summary

```
Call:
lm(formula = download_speed ~ price, data = south)

Residuals:
    Min      1Q  Median      3Q     Max
-41.011 -15.936  -6.118  15.616 158.438

Coefficients:
                Estimate    Std. Error t value             Pr(>|t|)
(Intercept) 41.7479820895  0.0794921759 525.184 < 0.0000000000000002 ***
price       -0.0000008189  0.0000001064  -7.695   0.0000000000000143 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 21.34 on 78406 degrees of freedom
Multiple R-squared:  0.0007547, Adjusted R-squared:  0.0007419
F-statistic: 59.22 on 1 and 78406 DF,  p-value: 0.00000000000001429

> cat("Correlation (South Yorkshire):", cor_south, "\n")
Correlation (South Yorkshire): -0.02747158
```

House price vs Drug rates (2023) per 10000 people for both counties

The line chart illustrates the relationship between average house prices and drug crime rates per 10,000 people between districts in West and South Yorkshire for 2023. A moderate positive correlation was observed ($r \approx 0.48$), suggesting that areas with higher drug crime rates may also have higher average house prices. The linear regression model further supports this, with a statistically significant positive coefficient for drug rate ($p < 0.001$). However, the R^2 value of 0.23 indicates that drug rates explain only 23% of the variation in house prices, suggesting other factors also play a major role.

Figure 20, House price vs Drug Offense



Linear Model summary and correlation

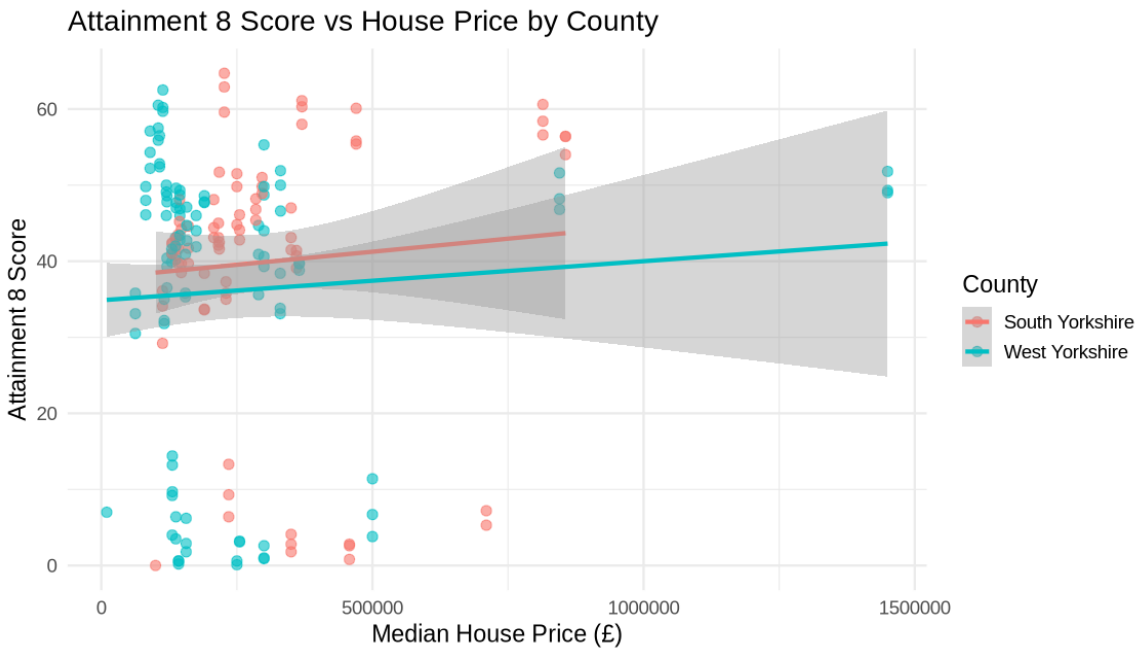
Figure 21, LM 2 Summary

```
Call:
lm(formula = AveragePrice ~ RatePer10k, data = joined_data)

Residuals:
      Min        1Q    Median        3Q       Max
-10999066    -39040    104531    214628  56565648

Coefficients:
            Estimate Std. Error t value             Pr(>|t|)
(Intercept)  -273653      88962  -3.076              0.00216 **
RatePer10k    294885      17646  16.711 < 0.0000000000000002 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2459000 on 933 degrees of freedom
Multiple R-squared:  0.2304,    Adjusted R-squared:  0.2295
F-statistic: 279.3 on 1 and 933 DF,  p-value: < 0.00000000000000022

> cat("Correlation between Drug Rate and House Price:", correlation, "\n")
Correlation between Drug Rate and House Price: 0.4799685
```

Attainment 8 score vs House Price for both counties

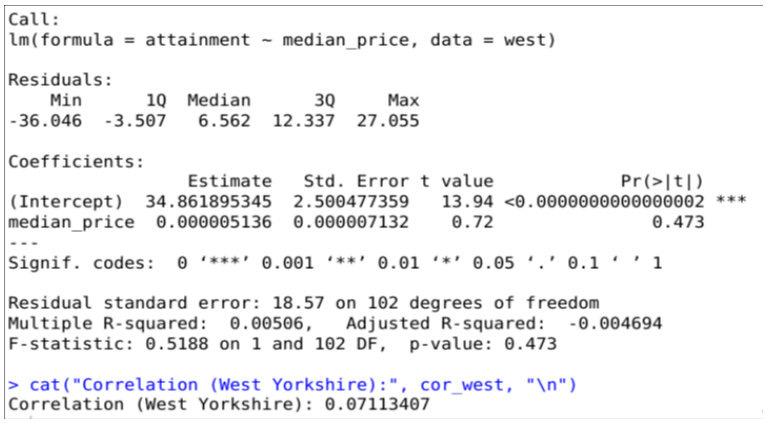
The analysis explores the relationship between median house prices and average Attainment 8 scores between West and South Yorkshire. In both counties, the correlation is very weak ($r \approx 0.07–0.08$), and linear regression models showed no significant relationship ($p > 0.47$). This suggests that house prices in these regions do not have a meaningful impact on academic performance as measured by Attainment 8 scores.

Figure 22, Attainment 8 vs House price by county



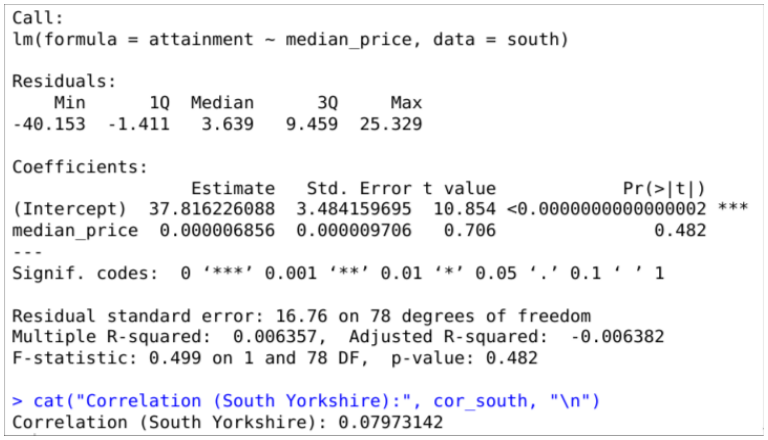
Linear model summary and correlation

Figure 23, LM 3 West Summary



```
Call:
lm(formula = attainment ~ median_price, data = west)

Residuals:
    Min      1Q  Median      3Q     Max
-36.046  -3.507   6.562  12.337  27.055

Coefficients:
                Estimate   Std. Error t value            Pr(>|t|)
(Intercept)  34.861895345  2.500477359   13.94 <0.0000000000000002 ***
median_price  0.000005136  0.000007132    0.72               0.473
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 18.57 on 102 degrees of freedom
Multiple R-squared:  0.00506,   Adjusted R-squared:  -0.004694
F-statistic: 0.5188 on 1 and 102 DF,  p-value: 0.473

> cat("Correlation (West Yorkshire):", cor_west, "\n")
Correlation (West Yorkshire): 0.07113407
```

Figure 24, LM 3 South Summary

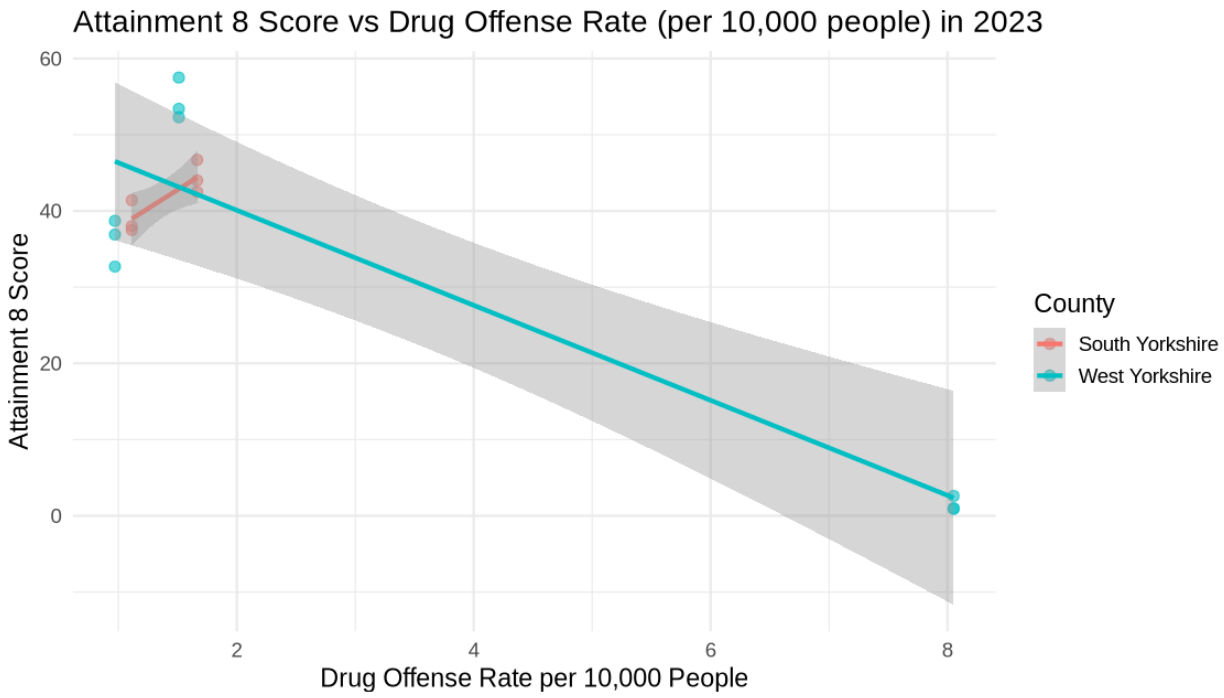


```
Call:
lm(formula = attainment ~ median_price, data = south)

Residuals:
    Min      1Q  Median      3Q     Max
-40.153  -1.411   3.639   9.459  25.329

Coefficients:
                Estimate   Std. Error t value            Pr(>|t|)
(Intercept)  37.816226088  3.484159695  10.854 <0.0000000000000002 ***
median_price  0.000006856  0.000009706   0.706               0.482
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 16.76 on 78 degrees of freedom
Multiple R-squared:  0.006357,  Adjusted R-squared:  -0.006382
F-statistic: 0.499 on 1 and 78 DF,  p-value: 0.482

> cat("Correlation (South Yorkshire):", cor_south, "\n")
Correlation (South Yorkshire): 0.07973142
```

Attainment 8 scores vs Drug Offense rates per 10000 people in 2023 for both counties

The analysis shows a strong negative correlation in West Yorkshire (r = -0.91), indicating that areas with lower Attainment 8 score has higher drug offense rates. The regression model supports this with a statistically significant negative relationship ($p < 0.001$). In contrast, South Yorkshire shows a strong positive correlation (r = 0.84), and the model also finds a significant positive relationship ($p = 0.035$). This suggests that the relationship between education outcomes and drug offense rates varies notably between South and West Yorkshire.

Figure 25, Attainment 8 vs Drug offense rate



Linear Model summary and correlation

Figure 26, LM 4 West Summary

```
Call:
lm(formula = attainment ~ rate_per_10k, data = west)

Residuals:
    Min      1Q  Median      3Q     Max
-13.803  -7.803  -1.358   9.161  14.361

Coefficients:
             Estimate Std. Error t value  Pr(>|t|)
(Intercept)    52.557      5.083  10.340 0.0000172 ***
rate_per_10k   -6.235      1.067  -5.842  0.000636 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.3 on 7 degrees of freedom
Multiple R-squared:  0.8298,    Adjusted R-squared:  0.8055
F-statistic: 34.13 on 1 and 7 DF,  p-value: 0.0006358

> cat("Correlation (West Yorkshire):", cor_west, "\n")
Correlation (West Yorkshire): -0.9109349
```

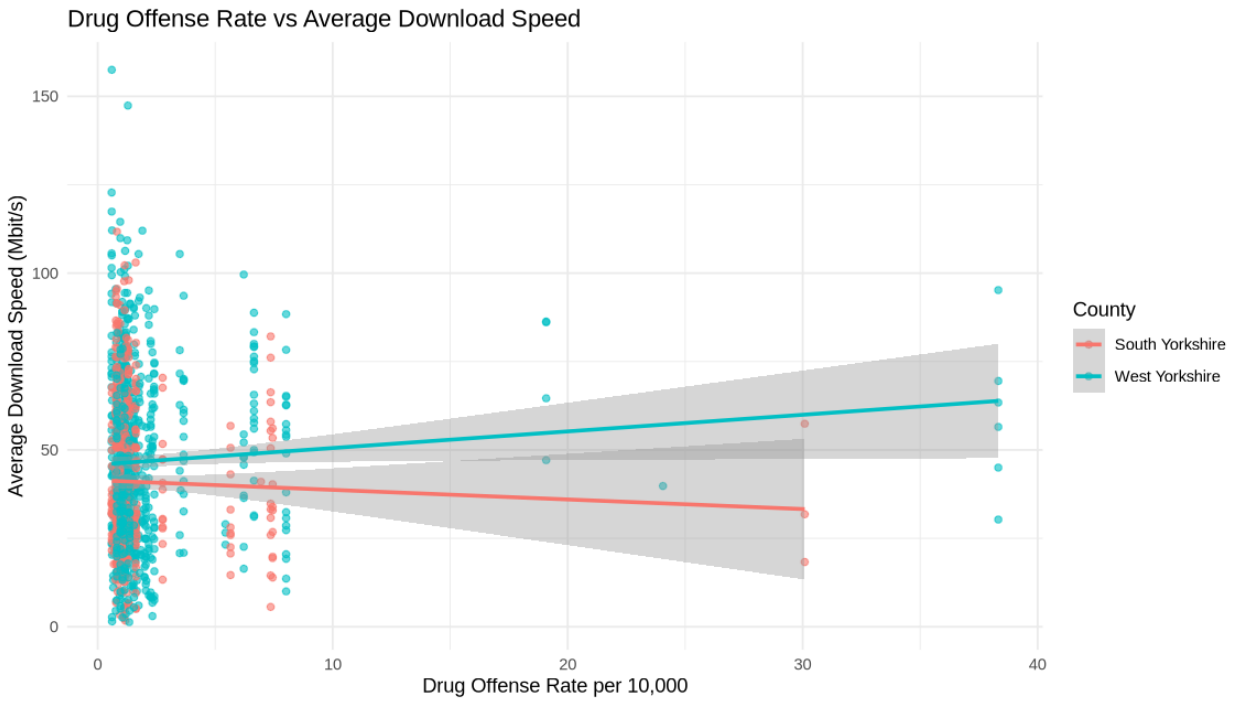
Figure 27, LM 4 South Summary

```
Call:
lm(formula = attainment ~ rate_per_10k, data = south)

Residuals:
      1       2       3       4       5       6
 2.4333  2.3000 -0.9667 -1.9000 -1.4667 -0.4000

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    27.990      4.459   6.278  0.00329 **
rate_per_10k    9.855      3.147   3.131  0.03515 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.125 on 4 degrees of freedom
Multiple R-squared:  0.7102,    Adjusted R-squared:  0.6378
F-statistic: 9.804 on 1 and 4 DF,  p-value: 0.03515

> cat("Correlation (South Yorkshire):", cor_south, "\n")
Correlation (South Yorkshire): 0.8427515
```

Average Download speed vs Drug Offense Rate per 10000 people for both counties

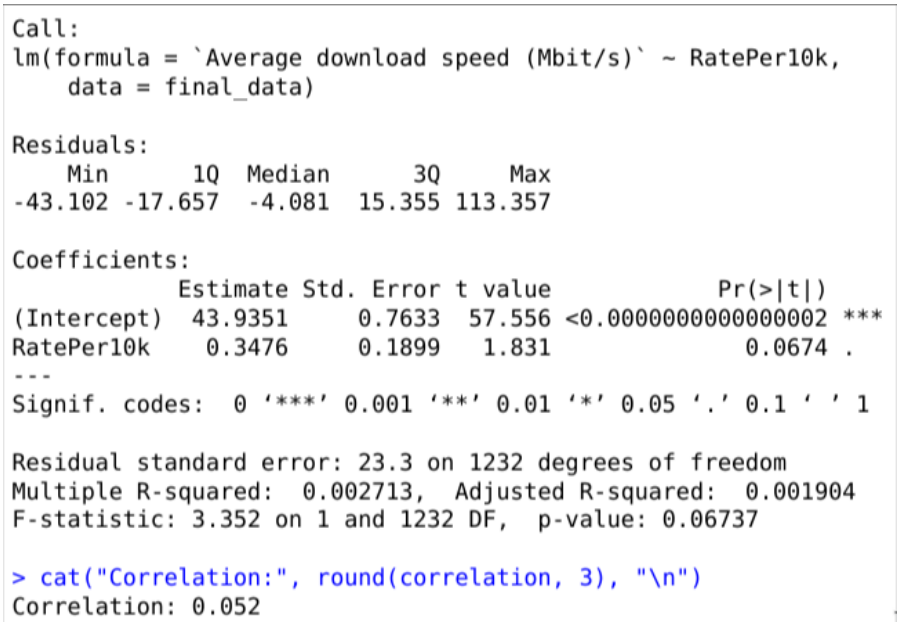
There is a very weak positive correlation (r = 0.052) between drug offense rate and average download speed, indicating almost no linear relationship. The regression model also supports this, with a small positive coefficient (0.35) and a p-value of 0.067, which is slightly above the conventional significance threshold of 0.05. This suggests that the relationship is not statistically significant, and drug offense rates do not meaningfully predict download speeds across the regions studied.

Figure 28, Drug Offense vs Average Download Speed



Linear Model summary and correlation

Figure 29, LM 5 Summary



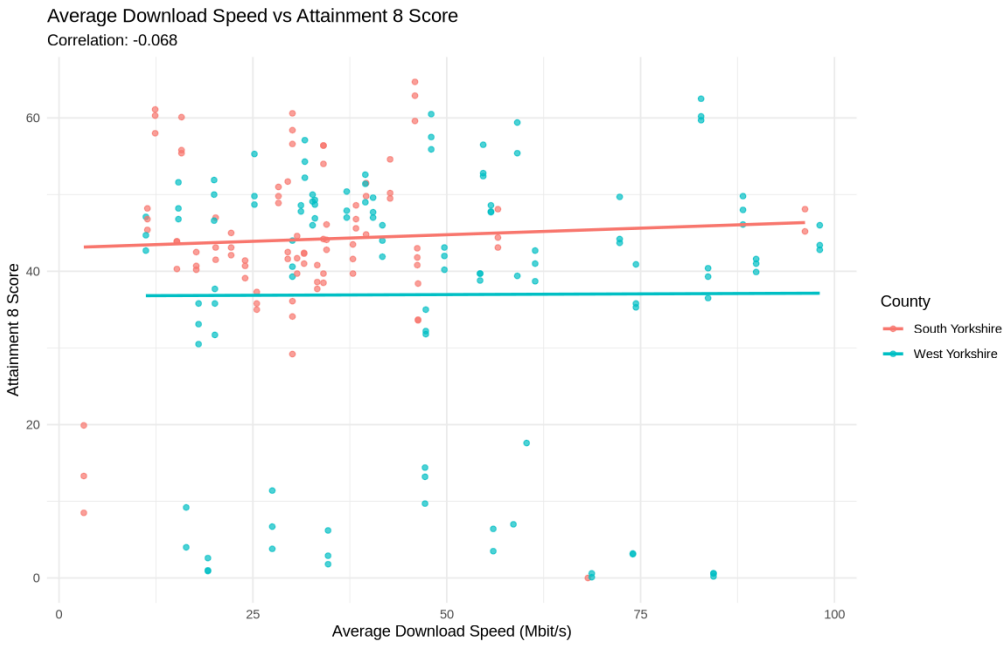
```
Call:
lm(formula = `Average download speed (Mbit/s)` ~ RatePer10k,
    data = final_data)

Residuals:
    Min      1Q  Median      3Q     Max
-43.102 -17.657  -4.081  15.355 113.357

Coefficients:
            Estimate Std. Error t value            Pr(>|t|)
(Intercept)  43.9351     0.7633  57.556 <0.0000000000000002 ***
RatePer10k    0.3476     0.1899   1.831              0.0674 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 23.3 on 1232 degrees of freedom
Multiple R-squared:  0.002713,  Adjusted R-squared:  0.001904
F-statistic: 3.352 on 1 and 1232 DF,  p-value: 0.06737

> cat("Correlation:", round(correlation, 3), "\n")
Correlation: 0.052
```

Average download speed vs Attainment 8 score for both counties

The correlation between Attainment 8 and Average Download Speed score was very weak and negative (correlation = -0.068), suggesting no meaningful linear relationship.

This was confirmed by Linear regression model. The slope of the model was -0.048 ($p = 0.335$), indicating the association is not statistically significant. The model suggests almost none of the variation in Attainment 8 scores ($R^2 \approx 0.005$), meaning download speed is not a good predictor of academic performance in this dataset.

Figure 30, Average Download Speed vs Attainment 8



Linear Model summary and correlation

Figure 31, LM 6 Summary

```
Call:
lm(formula = ATT8SCR ~ Average.download.speed..Mbit.s., data = filtered_data)

Residuals:
    Min      1Q  Median      3Q     Max
-40.227  -1.845   3.400   8.963  24.864

Coefficients:
                             Estimate Std. Error t value            Pr(>|t|)
(Intercept)                  42.05521    2.33897  17.980 <0.0000000000000002 ***
Average.download.speed..Mbit.s. -0.04835    0.05001  -0.967               0.335
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 15.76 on 199 degrees of freedom
Multiple R-squared:  0.004675,  Adjusted R-squared:  -0.0003264
F-statistic: 0.9347 on 1 and 199 DF,  p-value: 0.3348

> cat("Correlation: ", correlation, "\n")
Correlation:  -0.06837595
```

Recommendation System

Overview

The recommendation system was developed to evaluate and compare towns in South Yorkshire and West Yorkshire based on multiple key factors: house prices, broadband speed, crime rates, and school performance. Each factor was cleaned, standardized, and analyzed to generate comparable scores across towns.

The system uses a weighted scoring method where each town receives a normalized score for each criterion. These scores are then combined to calculate an overall score, helping to identify the most attractive locations for property investment. By integrating data from multiple domains, the system provides a data-driven and balanced approach to support decision-making for potential home buyers or investors.

Results

Based on Broadband speed

Figure 32, Broadband speed result

	TOWN	avg_down_speed	normDownSpeed
1	LEEDS	53.21395	10.000000
2	HUDDERSFIELD	52.17439	9.653007
3	ELLAND	52.04573	9.610060
4	BRADFORD	50.48136	9.087893
5	MEXBOROUGH	49.03646	8.605603
6	BRIGHOUSE	46.63752	7.804864
7	BARNSLEY	43.45569	6.742809
8	ROTHERHAM	42.95196	6.574668
9	HALIFAX	41.22150	5.997061
10	NORMANTON	40.54146	5.770073

Based on School Attainment score

Figure 33, School Result

	TOWN	avgAtt8	normAtt8
1	HOLMFIRTH	53.38333	10.000000
2	HEBDEN BRIDGE	48.90000	7.787222
3	ELLAND	46.20000	6.454620
4	NORMANTON	44.13333	5.434604
5	SHEFFIELD	44.09559	5.415974
6	HALIFAX	43.51905	5.131419
7	MEXBOROUGH	42.66667	4.710721
8	BRADFORD	42.52875	4.642651
9	SOWERBY BRIDGE	41.24444	4.008774
10	DONCASTER	40.88431	3.831030

Based on House Prices

Figure 34, House prices Result

	TOWN	avgPrice	normHousingPrice
1	MEXBOROUGH	156509.8	10.000000
2	BRADFORD	182448.5	8.517212
3	ROTHERHAM	193302.4	7.896743
4	BARNSLEY	193977.6	7.858147
5	HALIFAX	198690.6	7.588729
6	DONCASTER	200276.1	7.498092
7	BRIGHOUSE	204026.7	7.283688
8	PONTEFRACT	222419.6	6.232257
9	OSSETT	222863.5	6.206877
10	SOWERBY BRIDGE	229973.1	5.800456

Based on Crime Data (safest)

Figure 35, Crime Result

	TOWN	normCrimeRate	crimeno
1	SHEFFIELD	10.000000	37
2	ROTHERHAM	9.994925	49
3	HUDDERSFIELD	9.983930	75
4	DONCASTER	9.983084	77
5	HOLMFIRTH	9.982661	78
6	LEEDS	9.981392	81
7	BRIGHOUSE	9.975472	95
8	BRADFORD	9.974626	97
9	WAKEFIELD	9.965745	118
10	MEXBOROUGH	9.965322	119

Overall Score

Figure 36, Overall Score - Final Results

	TOWN	avgPrice	crimeno	avgAtt8	avg_down_speed	finalPoints
1	MEXBOROUGH	156509.8	119	42.66667	49.03646	8.320412
2	BRADFORD	182448.5	97	42.52875	50.48136	8.055596
3	ELLAND	272270.0	330	46.20000	52.04573	7.330829
4	HALIFAX	198690.6	159	43.51905	41.22150	7.166404
5	HUDDERSFIELD	233966.0	75	37.15610	52.17439	6.800021
6	ROTHERHAM	193302.4	49	37.85957	42.95196	6.701122
7	DONCASTER	200276.1	77	40.88431	38.10030	6.566862
8	HOLMFIRTH	308118.4	78	53.38333	36.49605	6.433922
9	BRIGHOUSE	204026.7	95	33.12222	46.63752	6.266006
10	BARNSLEY	193977.6	121	33.61818	43.45569	6.202554

After examining and ranking towns on house price, broadband speed, frequency and rate of crime, and school ratings I am recommending and found the three best towns to buy in are Mexborough, Bradford, and Elland. Mexborough was ranked highest on affordability and broadband reliability, Bradford was also ranked very high due to its excellent infrastructure and connectivity and Elland is a good option due to the quality of schools and lower crime. These three towns offer a good balance of price, livability and buying a house with potential upside from a long-term ownership perspective.

Reflection

This project provided valuable hands-on experience in applying data science techniques to solve a real-world problem. Through data cleaning, integration, analysis, and modeling, I developed a stronger understanding of the entire data science workflow. One key takeaway was the importance of high-quality, well-structured data. A significant portion of the time was spent cleaning and preparing datasets, which proved essential for producing reliable insights.

I additionally learned how different variables interact and how to use linear models to understand a trend or directionality even when the relationships were weak or contrary to expectations. Creating the recommendation system based on multiple weighted factors allowed me to include the analytical side of thinking with the praxis of decision-making. Overall, this assignment reinforced my confidence in using R for data analysis and helped bolster my ability to effectively communicate results visually and descriptively through statistical summaries.

Legal and Ethical Issues

This project relied entirely on publicly available datasets from UK government and open data portals. All data used was collected, processed, and presented in compliance with open data licenses. Personal information or sensitive data was not accessed or used at any point.

From an ethical standpoint, we ensured that we conducted an impartial analysis, with no bias. The recommendation system was based exclusively on a data-screening process and no personal beliefs. Likewise, while some of the neighborhoods might look unfavorable based on their higher crime rates or lower school scores, the report will not objectify a neighborhood, but rather will emphasize only measurable indicators that will vet property investment choices.

Conclusion

This analysis used house prices, broadband speed, crime, and school data to compare locations in South Yorkshire and West Yorkshire. Using data science methods and a scoring system, Mexborough, Bradford, and Elland were identified as the best locations to buy properties, as all provided an affordable, livable, and quality infrastructure combinations. Although not all factors provided strong linear relationships, the multi-criteria assessment provided a more holistic understanding of what makes certain place viable. Ultimately, this project illustrates how data science can contribute to evidence-based decision making and future buyers and investor understanding of the location.

References

Open Geography Portal. (n.d.). Open Geography Portal. https://geoportal.statistics.gov.uk/

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

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

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

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-sciencecourse-curriculum/

Compare school and college performance in England. (n.d.). Compare School and College Performance in England. https://www.compare-school-performance.service.gov.uk/download-data

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

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

Appendix

GitHub Link: https://github.com/niii-san/Datascience-Assignment

Cleaning.R

broadband_cleaning.R

```r
library(tidyverse)

# Reading Datasets
broadband_perf <- read_csv("201805_fixed_pc_performance_r03.csv")
broadband_cov <- read_csv("201809_fixed_pc_coverage_r01.csv")

# Standardizing and cleaning postcodes in broadband data sets
broadband_perf <- broadband_perf %>%
  mutate(postcode = toupper(str_replace_all(postcode, " ", ""))) %>%
  filter(!is.na(postcode) & postcode != "")
broadband_cov <- broadband_cov %>%
  mutate(postcode = toupper(str_replace_all(postcode, " ", ""))) %>%
  filter(!is.na(postcode) & postcode != "")

# Joining broadband performance and coverage data on postcode
broadband_combined <- left_join(broadband_perf, broadband_cov, by = "postcode")

# Reading postcode to LSOA data set
postcode_to_lsoa <- read_csv("PostcodeToLSOA.csv")
View(postcode_to_lsoa)

postcode_to_lsoa_clean <- postcode_to_lsoa %>%
  mutate(pcds = toupper(str_replace_all(pcds, " ", ""))) %>%
  filter(!is.na(pcds) & pcds != "")

broadband_with_lsoa <- broadband_combined %>%
  left_join(postcode_to_lsoa_clean, by = c("postcode" = "pcds"))

broadband_final <- broadband_with_lsoa %>%
  filter(!is.na(lsoa11cd))

write_csv(broadband_final, "cleaned_broadband_speed_with_lsoa.csv")
View(broadband_final)

colnames(broadband_final)

postcode_lsoa_county = read.csv("postcode_lsoa_county.csv")
View(postcode_lsoa_county)

broadband_final <- broadband_final %>%
  mutate(pcds = str_remove_all(postcode, "\\s+"))

postcode_lsoa_county <- postcode_lsoa_county %>%
  mutate(pcds = str_remove_all(pcds, "\\s+"))

broadband_with_county <- broadband_final %>%
  inner_join(
    postcode_lsoa_county %>% select(pcds, County),
    by = "pcds"
  )
colnames(broadband_with_county)

# Filtering only South Yorkshire, West Yorkshire and needed columns
broadband_filtered <- broadband_with_county %>%
  filter(County %in% c("South Yorkshire", "West Yorkshire")) %>%
  select(-"% of premises unable to receive 2Mbit/s", -"% of premises unable to receive 5Mbit/s",
         -"% of premises unable to receive 10Mbit/s", -"% of premises unable to receive 30Mbit/s",
         -"% of premises unable meet USO", -"% of premises able to receive decent broadband from FWA",
         -"% of premises able to receive SFBB from FWA", -"% of premises able to receive NGA",
         -"Average data usage (GB) for lines < 10Mbit/s",-"UFBB availability (% premises)", -"SFBB availability (% premises)",
         -"FTTP availability (% premises)",-"Number of connections < 2 Mbit/s (number of lines)",
         -"Number of connections 2<5 Mbit/s (number of lines)",-"Number of connections 5<10 Mbit/s (number of lines)",
         -"Number of connections 10<30 Mbit/s (number of lines)",-"Number of connections 30<300 Mbit/s (number of lines)",
         -"Number of connections >= 300 Mbit/s (number of lines)",-"Number of connections >= 30 Mbit/s (number of lines)",
         -"Average data usage (GB) for lines < 10Mbit/s",-"Average data usage (GB) for Basic BB lines",
         -"Average data usage (GB) for SFBB lines",-"Average data usage (GB) for UFBB lines",
         ,-"All Premises",-"All Matched Premises",-"doterm")

View(broadband_filtered)
write_csv(broadband_filtered, "broadband_filtered_yorkshire.csv")
```

crime_data_cleaning.R

Source on Save

```r
library(tidyverse)

base_path <- "~/ds_assignment/obtained_data/crime_dataset/all_together"

# Getting all CSV files inside sub folders
csv_files <- list.files(base_path, pattern = "*.csv", full.names = TRUE, recursive = TRUE)

# Cleaning function
clean_crime_file <- function(file_path) {
  df <- read_csv(file_path, show_col_types = FALSE)

  filename <- basename(file_path)
  split_name <- str_split(filename, "-", simplify = TRUE)
  year_month <- paste0(split_name[1], "-", split_name[2])
  county <- str_to_title(split_name[3])  # "South" or "West"

  # Adding metadata and select useful columns only
  df <- df %>%
    mutate(
      County = county,
      YearMonth = year_month,
      District = word(`LSOA name`, 1)
    ) %>%
    select(
      YearMonth,
      Month,
      County,
      District,
      `LSOA code`,
      `Crime type`,
      Longitude,
      Latitude
    )

  return(df)
}

# Combining all cleaned crime files
crime_df <- map_dfr(csv_files, clean_crime_file)

# Save to CSV if needed
write_csv(crime_df, "cleaned_crime_dataset.csv")

cleaned_crime_dataset.csv = read_csv("cleaned_crime_dataset.csv")
View(cleaned_crime_dataset.csv)

########################## Crime dataset combining with population  ####################
crime_dataset = read.csv("cleaned_crime_dataset.csv")
postcode_lsoa_county_filtered = read.csv("postcode_lsoa_county_filtered.csv")
View(crime_dataset)
View(postcode_lsoa_county_filtered)

# Triming all whitespace in 'pcds'
postcode_lsoa_county_filtered$pcds <- gsub("\\s.*", "", postcode_lsoa_county_filtered$pcd8

# Keep only one postcode per LSOA (choose the first for now)
postcode_unique <- postcode_lsoa_county_filtered %>%
  select(lsoa11cd, pcds) %>%
  distinct(lsoa11cd, .keep_all = TRUE)

# Join using deduplicated postcode dataset
combined_dataset <- crime_dataset %>%
  left_join(postcode_unique, by = c("LSOA.code" = "lsoa11cd"))

# combining crime dataset with population ###################
population = read_csv("population_combined_distinct.csv")

# Triming numbers after white space in Postcode column
population <- population %>%
  mutate(Postcode = gsub("\\s.*", "", Postcode)) %>%     # Keep only part before space
  distinct(Postcode, .keep_all = TRUE)

# combining population with crime dataset
combined_with_population <- combined_dataset %>%
  left_join(population, by = c("pcds" = "Postcode"))

View(combined_with_population)

write_csv(combined_with_population,"cleaned_crime_combined_population.csv")
```

house_price_cleaning.R

Source on Save

```r
library(tidyverse)


# Defining the column names based on UK Price Paid Data structure
col_names <- c(
  "transaction_id", "price", "date_of_transfer", "postcode",
  "property_type", "new_build_flag", "tenure_type", "primary_address",
  "secondary_address", "street", "locality", "town_city",
  "district", "county", "ppd_category_type", "record_status"
)

# Loading the datasets without headers
pp2021 <- read_csv("pp-2021.csv", col_names = FALSE)
pp2022 <- read_csv("pp-2022.csv", col_names = FALSE)
pp2023 <- read_csv("pp-2023.csv", col_names = FALSE)
pp2024 <- read_csv("pp-2024.csv", col_names = FALSE)

# Assigning column names to each dataset
colnames(pp2021) <- col_names
colnames(pp2022) <- col_names
colnames(pp2023) <- col_names
colnames(pp2024) <- col_names

View(pp2022)

# Combining all datasets into one
house_data <- bind_rows(pp2021, pp2022, pp2023, pp2024)

# Cleaning the postcode column: trim white space and convert to uppercase
house_data <- house_data %>%
  mutate(postcode = str_trim(str_to_upper(postcode)))

# only filtering south yorkshire and west yorkshire
filtered_data <- house_data %>%
  filter(county %in% c("WEST YORKSHIRE", "SOUTH YORKSHIRE"))
View(filtered_data)

write_csv(filtered_data, "cleaned_house_prices.csv")
```

school_data_cleaning.R

Source on Save

```r
library(tidyverse)

ks4final_2021_2022 = read_csv("england_ks4final_2021_2022.csv")
ks4final_2022_2023 = read_csv("england_ks4final_2022_2023.csv")
ks4final_2023_2024 = read_csv("england_ks4final_2023_2024.csv")

postcode_lsoa_county = read_csv("postcode_lsoa_county_filtered.csv")

test = ks4final_2021_2022 %>%
  distinct(TOWN , .keep_all =  TRUE)

View(ks4final_2021_2022)
View(ks4final_2022_2023)
View(ks4final_2023_2024)

# Adding year and selecting relevant columns
ks4_21_22 <- ks4final_2021_2022 %>%
  select(URN, SCHNAME, PCODE, TOWN, ATT8SCR) %>%
  mutate(Year = "2022")
ks4_22_23 <- ks4final_2022_2023 %>%
  select(URN, SCHNAME, PCODE, TOWN, ATT8SCR) %>%
  mutate(Year = "2023")
ks4_23_24 <- ks4final_2023_2024 %>%
  select(URN, SCHNAME, PCODE, TOWN, ATT8SCR) %>%
  mutate(Year = "2024")

# Combining all three years ks4 dataset
ks4_combined <- bind_rows(ks4_21_22, ks4_22_23, ks4_23_24)

# Cleaning ATT8SCR by removing 'NE', 'SUPP', and non-numeric values, then remove NAs
ks4_clean <- ks4_combined %>%
  filter(!ATT8SCR %in% c("NE", "SUPP")) %>%
  filter(!is.na(ATT8SCR)) %>%
  filter(str_detect(ATT8SCR, "^[0-9.]+$"))

# Converting ATT8SCR to numeric for analysis/visualization
ks4_clean <- ks4_clean %>%
  mutate(ATT8SCR = as.numeric(ATT8SCR))

postcode_lsoa_county = read_csv("postcode_lsoa_county.csv")

# Triming whitespace in postcode columns
ks4_clean <- ks4_clean %>%
  mutate(PCODE = str_trim(PCODE))
postcode_lsoa_county <- postcode_lsoa_county %>%
  mutate(pcds = str_trim(pcds))

# Joining to add 'County'
ks4_with_county <- ks4_clean %>%
  left_join(postcode_lsoa_county %>% select(pcds, County), by = c("PCODE" = "pcds"))

View(ks4_with_county)

# Filtering only West Yorkshire and South Yorkshire
ks4_filtered <- ks4_with_county %>%
  filter(County %in% c("West Yorkshire", "South Yorkshire"))

View(ks4_filtered)

# Joining ladnm to school data set
school_data <- inner_join(
  school_data,
  postcode_lsoa_county %>% select(pcd7, ladnm),
  by = c("PCODE" = "pcd7")
)

write.csv(ks4_filtered, "cleaned_filtered_school_dataset.csv", row.names = FALSE)
```

population_cleaning.R

Source on Save

```r
library(tidyverse)

# Read datasets
population <- read_csv("Population2011_1656567141570.csv")

# Clean population dataset
population_clean <- population %>%
  mutate(Postcode = str_replace_all(str_trim(Postcode), " +", " "),
         Postcode = toupper(Postcode))

# Save the cleaned dataset
write_csv(population_clean, "cleaned_population.csv")

PopulationData = population_clean

View(PopulationData)
PopulationData$Postcode <- gsub("\\s.*", "", PopulationData$Postcode)

PopulationData = PopulationData %>%
  distinct(Postcode, .keep_all = TRUE)


PopulationData$Population <- as.numeric(PopulationData$Population)


CombinedPopulationData = PopulationData %>%
  group_by(Postcode) %>%
  summarise_at(vars(Population), list(Population2011 = ~sum(., na.rm = TRUE))) %>%
  mutate(Population2012 = 1.00695353132322269 * Population2011) %>%
  mutate(Population2013 = 1.00669740535540783 * Population2012) %>%
  mutate(Population2014 = 1.00736463978721671 * Population2013) %>%
  mutate(Population2015 = 1.00792367505802859 * Population2014) %>%
  mutate(Population2016 = 1.00757874492811929 * Population2015) %>%
  mutate(Population2017 = 1.00679374473924223 * Population2016) %>%
  mutate(Population2018 = 1.00605929132212552 * Population2017) %>%
  mutate(Population2019 = 1.00561255390388033 * Population2018) %>%
  mutate(Population2020 = 1.00561255390388033 * Population2019) %>%
  mutate(Population2021 = 1.005425 * Population2020) %>%
  mutate(Population2022 = 1.004920 * Population2021) %>%
  mutate(Population2023 = 1.004510 * Population2022) %>%
  mutate(Population2024 = 1.004220 * Population2023) %>%
  select(Postcode, Population2020, Population2021, Population2022, Population2023, Population2024)



View(CombinedPopulationData)

write_csv(CombinedPopulationData,"population_combined_distinct.csv")
```

postcode_lsoa_cleaning_filtering.R

Source on Save

```r
library(tidyverse)

# joining postcode_lsoa with county
postcode_to_lsoa = read.csv("PostcodetoLSOA.csv")
View(postcode_to_lsoa)
lad_counties = read.csv("lad_counties.csv") # source https://geoportal.statistics.gov.uk/
View(lad_counties)

postcode_with_county <- postcode_to_lsoa %>%
  inner_join(lad_counties, by = c("ladcd" = "LAD24CD")) %>%
  rename(County = CTY24NM)

View(postcode_with_county)

postcode_with_county <- postcode_with_county %>% select(-LAD24NM)

postcode_lsoa_county <- postcode_with_county

View(postcode_lsoa_county)

postcode_lsoa_county <- postcode_lsoa_county %>%
  mutate(pcds = str_replace_all(pcds, "\\s+", "")) %>%  # Remove all spaces in pcds column
  filter(County %in% c("South Yorkshire", "West Yorkshire"))

View(postcode_yorkshire)

# saving filtered data to CSV
write.csv(postcode lsoa county, "postcode lsoa county filtered.csv", row.names = FALSE)
```

Eda.R

broadbandspeed.R

Source on Save

```r
library(tidyverse)

broadband = read.csv("broadband_filtered_yorkshire.csv")
colnames(broadband)

# Remove rows with missing or invalid average download speed
broadband <- broadband %>%
  filter(!is.na(Average.download.speed..Mbit.s.))


# Boxplot of Avg Download Speed by District
# West Yorkshire boxplot
broadband %>%
  filter(County == "West Yorkshire") %>%
  ggplot(aes(x = fct_reorder(ladnm.x, Average.download.speed..Mbit.s., .fun = median),
             y = Average.download.speed..Mbit.s.)) +
  geom_boxplot(fill = "skyblue") +
  coord_flip() +
  labs(title = "West Yorkshire - Avg Download Speed by District",
       x = "District", y = "Avg Download Speed (Mbps)") +
  theme_minimal()

# South Yorkshire boxplot
broadband %>%
  filter(County == "South Yorkshire") %>%
  ggplot(aes(x = fct_reorder(ladnm.x, Average.download.speed..Mbit.s., .fun = median),
             y = Average.download.speed..Mbit.s.)) +
  geom_boxplot(fill = "salmon") +
  coord_flip() +
  labs(title = "South Yorkshire - Avg Download Speed by District",
       x = "District", y = "Avg Download Speed (Mbps)") +
  theme_minimal()

#  Bar Chart of Avg Download Speed by Town (from lsoa11nm)
# West Yorkshire
broadband %>%
  filter(County == "West Yorkshire") %>%
  group_by(ladnm.x) %>%
  summarise(avg_speed = mean(Average.download.speed..Mbit.s., na.rm = TRUE)) %>%
  arrange(desc(avg_speed)) %>%
  slice_head(n = 15) %>%
  ggplot(aes(x = fct_reorder(ladnm.x, avg_speed), y = avg_speed)) +
  geom_col(fill = "steelblue") +
  labs(title = "West Yorkshire Towns by Avg Download Speed",
       x = "Town", y = "Avg Download Speed (Mbps)") +
  theme_minimal()

# South Yorkshire
broadband %>%
  filter(County == "South Yorkshire") %>%
  group_by(ladnm.x) %>%
  summarise(avg_speed = mean(Average.download.speed..Mbit.s., na.rm = TRUE)) %>%
  arrange(desc(avg_speed)) %>%
  slice_head(n = 15) %>%
  ggplot(aes(x = fct_reorder(ladnm.x, avg_speed), y = avg_speed)) +
  geom_col(fill = "coral") +
  labs(title = "South Yorkshire Towns by Avg Download Speed",
       x = "Town", y = "Avg Download Speed (Mbps)") +
  theme_minimal()
```

crime.R

Source on Save

```r
library(tidyverse)
library(fmsb)

# Read data
crime_data <- read.csv("cleaned_crime_combined_population.csv") %>%
  filter(Year != 2025) %>%
  filter(!(County == "South" & District == "Leeds")) %>%
  filter(!(County == "West" & District == "Barnsley")) %>%
  distinct()

# ---- 1. Boxplot for Drug Offense Rate by District (2 graphs) ----
# Calculate drug rate per 100 total crimes per District-Year
drug_data <- crime_data %>%
  filter(Crime.type == "Drugs") %>%
  group_by(County, District, Year) %>%
  summarise(drug_count = n(), .groups = "drop")

total_data <- crime_data %>%
  group_by(County, District, Year) %>%
  summarise(total_crimes = n(), .groups = "drop")
drug_rate <- drug_data %>%
  left_join(total_data, by = c("County", "District", "Year")) %>%
  mutate(rate_percent = (drug_count / total_crimes) * 100)

# Separate data
drug_south <- drug_rate %>% filter(County == "South")
drug_west  <- drug_rate %>% filter(County == "West")

# Plot 1 - South
ggplot(drug_south, aes(x = District, y = rate_percent)) +
  geom_boxplot(fill = "#2E86C1") +
  labs(title = "Drug Offense Rate in South Yorkshire Districts",
       y = "Drug Rate (%)", x = "District") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  coord_flip() + theme_minimal()

# Plot 2 - West
ggplot(drug_west, aes(x = District, y = rate_percent)) +
  geom_boxplot(fill = "#C0392B") +
  labs(title = "Drug Offense Rate in West Yorkshire Districts",
       y = "Drug Rate (%)", x = "District") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  coord_flip() + theme_minimal()

# ---- 2. Radar Chart for Vehicle Crime Rate (West, May 2022) ----
radar_data <- crime_data %>%
  filter(County == "West", Year == 2022, Month == 5, Crime.type == "Vehicle crime") %>%
  group_by(District) %>%
  summarise(vehicle_crime = n(), .groups = "drop")
# Prepare for radar
radar_plot_data <- radar_data %>%
  column_to_rownames("District") %>%
  t() %>%
  as.data.frame()

# Add max-min rows required by fmsb
radar_plot_data <- rbind(rep(max(radar_plot_data), ncol(radar_plot_data)),
                         rep(0, ncol(radar_plot_data)),
                         radar_plot_data)
# Plot radar
radarchart(radar_plot_data, axistype = 1,
           pcol = "#34495E", pfcol = scales::alpha("#3498DB", 0.6), plwd = 2,
           cglcol = "grey", axislabcol = "black",
           title = "Vehicle Crime Rate by District\n(West Yorkshire, May 2022)")

# ---- 3. Pie Chart for Robbery (South, June 2022) ----
robbery_data <- crime_data %>%
  filter(County == "West", Year == 2022, Month == 6, Crime.type == "Robbery") %>%
  group_by(District) %>%
  summarise(robbery_count = n(), .groups = "drop")
```

```r
# Plot pie chart
ggplot(robbery_data, aes(x = "", y = robbery_count, fill = District)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y") +
  labs(title = "Robbery Distribution by District\n(West Yorkshire, June 2022)") +
  theme_void()

# ---- 4. Line Chart for Drug Offense Rate per 10,000 People ----
# Filter for drug crimes only
drug_data <- crime_data %>%
  filter(Crime.type == "Drugs")
# Ensure Year is numeric if not already
drug_data$Year <- as.numeric(drug_data$Year)
# Add population per year
drug_data <- drug_data %>%
  mutate(Population = case_when(
    Year == 2020 ~ Population2020,
    Year == 2021 ~ Population2021,
    Year == 2022 ~ Population2022,
    Year == 2023 ~ Population2023,
    Year == 2024 ~ Population2024,
    TRUE ~ NA_real_
  ))

# Group by County, Year, and Month
district_rate_yearly <- drug_data %>%
  group_by(County, Year, Month) %>%
  summarise(
    drug_offense_count = n(),
    Population = sum(Population, na.rm = TRUE),
    .groups = "drop"
  ) %>%
  mutate(
    drug_rate_per_10k = (drug_offense_count / Population) * 10000,
    date = as.Date(paste(Year, Month, "01", sep = "-"))
  )

# Plot the monthly drug offense rate with proper date formatting
ggplot(district_rate_yearly, aes(x = date, y = drug_rate_per_10k, color = County)) +
  geom_line(size = 1) +
  geom_point(size = 2) +
  scale_x_date(
    date_breaks = "3 months",
    date_labels = "%Y-%m"
  ) +
  labs(
    title = "Drug Offense Trend of South & West Yorkshire (2022–2024)",
    x = "Month",
    y = "Drug Offense Rate (/10,000)",
    color = "County"
  ) +
  theme_minimal() +
  theme(
    legend.position = "bottom"  )
```

house_prices.R

Source on Save

```r
library(tidyverse)

house_prices = read_csv("cleaned_house_prices.csv")
colnames(house_prices)

# Extract year
house_prices <- house_prices %>%
  mutate(year = year(as.Date(date_of_transfer)))
# Filter years and counties
filtered_hp <- house_prices %>%
  filter(year %in% 2021:2024,
         county %in% c("WEST YORKSHIRE", "SOUTH YORKSHIRE"))
# Group and summarize
avg_prices_by_year_county <- filtered_hp %>%
  group_by(year, county) %>%
  summarise(avg_price = mean(price, na.rm = TRUE), .groups = "drop")
# Plot line graph
ggplot(avg_prices_by_year_county, aes(x = year, y = avg_price, color = county)) +
  geom_line(size = 1.2) +
  geom_point() +
  labs(
    title = "Average House Prices (2021–2024)",
    x = "Year",
    y = "Average Price",
    color = "County"
  ) + theme_minimal()

# bar chart
# Filter for 2023 only
avg_2023 <- filtered_hp %>%
  filter(year == 2023) %>%
  group_by(county) %>%
  summarise(avg_price = mean(price, na.rm = TRUE), .groups = "drop")

# Plot bar chart
ggplot(avg_2023, aes(x = county, y = avg_price, fill = county)) +
  geom_col(width = 0.6) +
  labs(
    title = "Average House Prices in 2023",
    x = "County",
    y = "Average Price"
  ) +
  theme_minimal() +
  theme(legend.position = "none")

# Boxplot for average house prices for both counties in separate diagrams (take
# variables Price and District)
# Calculate IQR and filter out outliers
filtered_hp_clean <- filtered_hp %>%
  group_by(county) %>%
  mutate(
    Q1 = quantile(price, 0.25, na.rm = TRUE),
    Q3 = quantile(price, 0.75, na.rm = TRUE),
    IQR = Q3 - Q1,
    lower_bound = Q1 - 1.5 * IQR,
    upper_bound = Q3 + 1.5 * IQR
  ) %>%
  filter(price >= lower_bound & price <= upper_bound) %>%
  ungroup()

# Split datasets
west_yorkshire <- filtered_hp_clean %>% filter(county == "WEST YORKSHIRE")
south_yorkshire <- filtered_hp_clean %>% filter(county == "SOUTH YORKSHIRE")

# Plot for West Yorkshire
ggplot(west_yorkshire, aes(x = district, y = price, fill = district)) +
  geom_boxplot() +
  labs(
    title = "House Prices by District – West Yorkshire",
    x = "District",
```

```
    x = "District",
    y = "Price"
  ) +
  theme_minimal() +
  coord_flip() +
  theme(
     legend.position = "none"
  )

# Plot for South Yorkshire
ggplot(south_yorkshire, aes(x = district, y = price, fill = district)) +
  geom_boxplot() +
  labs(
    title = "House Prices by District – South Yorkshire",
    x = "District",
    y = "Price"
  ) +
  theme_minimal() +
  coord_flip() +
  theme(
    legend.position = "none"
  )
```

school.R

Source on Save

```r
library(tidyverse)

school_data = read.csv("cleaned_filtered_school_dataset.csv")
View(school_data)
colnames(school_data)

#  Boxplot: South Yorkshire – Attainment 8 Score by District (TOWN)
# South Yorkshire boxplot
school_data %>%
  filter(County == "South Yorkshire", Year == 2022, !is.na(ladnm)) %>%
  ggplot(aes(x = fct_reorder(ladnm, ATT8SCR), y = ATT8SCR)) +
  geom_boxplot(fill = "steelblue") +
  labs(
    title = "Attainment 8 Score by District (2022) - South Yorkshire",
    x = "Town", y = "Attainment 8 Score"
  ) +
  theme_minimal() +
  coord_flip()

# Boxplot for West Yorkshire schools (2022 only)
school_data %>%
  filter(County == "West Yorkshire", Year == 2022, !is.na(ladnm)) %>%
  ggplot(aes(x = fct_reorder(ladnm, ATT8SCR), y = ATT8SCR)) +
  geom_boxplot(fill = "darkseagreen") +
  labs(
    title = "Attainment 8 Score by District (2022) - West Yorkshire",
    x = "Town", y = "Attainment 8 Score"
  ) +
  theme_minimal() +
  coord_flip()

# Line Graph to show the relationship between attainment 8 score and years over
# multiple districts in South Yorkshire and west Yorkshire
# West Yorkshire Line Graph
school_data %>%
  filter(County %in% c("South Yorkshire", "West Yorkshire"), !is.na(ladnm)) %>%
  group_by(County, ladnm, Year) %>%
  summarise(avg_attainment = mean(ATT8SCR, na.rm = TRUE), .groups = "drop") %>%
  mutate(ladnm_full = paste0(ladnm, " (", County, ")")) %>%
  ggplot(aes(x = ladnm_full, y = avg_attainment, group = Year, color = as.factor(Year))) +
  geom_line(aes(linetype = as.factor(Year)), size = 1) +
  geom_point(size = 2) +
  labs(
    title = "Attainment 8 Score by District (South & West Yorkshire)",
    x = "Town (County)",
    y = "Average Attainment 8 Score",
    color = "Year",
    linetype = "Year"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

LinearModeling.R

LM_1_houseprice_downloadspeed.R

Source on Save

```r
library(tidyverse)

house_data = read_csv("cleaned_house_prices.csv")
broadband_data = read_csv("broadband_filtered_yorkshire.csv")

colnames(house_data)
colnames(broadband_data)

# Preprocess Data
house_trimmed <- house_data %>%
  select(postcode, price, county = county) %>%
  mutate(postcode = str_remove_all(postcode, " "))  # Remove spaces

broadband_trimmed <- broadband_data %>%
  select(postcode = `postcode.x`, download_speed = `Average download speed (Mbit/s)`, County) %>%
  mutate(postcode = str_remove_all(postcode, " "))

# Merge Both Datasets by Postcode
merged_data <- inner_join(house_trimmed, broadband_trimmed, by = "postcode") %>%
  filter(!is.na(download_speed) & !is.na(price))  # Remove missing values

# Visualize Scatter Plot + Linear Models
ggplot(merged_data, aes(x = price, y = download_speed, color = County)) +
  geom_point(alpha = 0.5, size = 2) +
  geom_smooth(method = "lm", se = TRUE, size = 1.2) +
  labs(
    title = "Download Speed vs House Price by County",
    subtitle = "With Linear Regression Lines for West and South Yorkshire",
    x = "House Price (£)",
    y = "Average Download Speed (Mbit/s)",
    color = "County"
  ) +
  theme_minimal(base_size = 14)

# Linear Model Summary per County
# i. West Yorkshire
west <- merged_data %>% filter(County == "West Yorkshire")
model_west <- lm(download_speed ~ price, data = west)
summary(model_west)

# ii. South Yorkshire
south <- merged_data %>% filter(County == "South Yorkshire")
model_south <- lm(download_speed ~ price, data = south)
summary(model_south)

# Correlation Calculation
cor_west <- cor(west$download_speed, west$price, use = "complete.obs")
cor_south <- cor(south$download_speed, south$price, use = "complete.obs")

cat("Correlation (West Yorkshire):", cor_west, "\n")
cat("Correlation (South Yorkshire):", cor_south, "\n")

```

LM_2_houseprice_drugrates.R

Source on Save

```r
library(tidyverse)

house_data = read_csv("cleaned_house_prices.csv")
drug_data = read_csv("cleaned_crime_combined_population.csv") %>%
  filter(Crime.type == "Drugs")

colnames(house_data)
colnames(drug_data)

# Fix postcode formatting: remove spaces and uppercase for joining
house_data <- house_data %>%
  mutate(Postcode = gsub(" ", "", toupper(postcode)))

# Compute average house price per postcode
avg_house_price <- house_data %>%
  group_by(Postcode, county) %>%
  summarise(AveragePrice = mean(price, na.rm = TRUE), .groups = "drop")

#Preprocess Drug Data
# Filter only 2023 drug crimes
drug_2023 <- drug_data %>%
  filter(Crime.type == "Drugs", Year == 2023)

# Remove spaces and uppercase for joining
drug_2023 <- drug_2023 %>%
  mutate(Postcode = gsub(" ", "", toupper(Postcode)))

# Calculate drug offense rate per 10,000 people
drug_rates_2023 <- drug_2023 %>%
  group_by(Postcode, County) %>%
  summarise(
    DrugOffenseCount = n(),
    Population = sum(Population2023, na.rm = TRUE),
    RatePer10k = (DrugOffenseCount / Population) * 10000,
    .groups = "drop"
  )

# Join Both Datasets
# Inner join on postcode
joined_data <- inner_join(drug_rates_2023, avg_house_price, by = "Postcode")

# Plotting with Linear Model and Correlation
# Correlation
correlation <- cor(joined_data$RatePer10k, joined_data$AveragePrice, use = "complete.obs")
cat("Correlation between Drug Rate and House Price:", correlation, "\n")

# Linear model
lm_model <- lm(AveragePrice ~ RatePer10k, data = joined_data)
summary(lm_model)

# Plot
ggplot(joined_data, aes(x = RatePer10k, y = AveragePrice, color = County)) +
  geom_point(alpha = 0.7) +
  geom_smooth(method = "lm", se = TRUE) +
  labs(
    title = "House Price vs Drug Offense Rate per 10,000 People (2023)",
    x = "Drug Offense Rate per 10,000",
    y = "Average House Price (GBP)"
  ) +
  theme_minimal()
```

LM_3_att8_houseprice.R

Source on Save

```r
library(tidyverse)

school_dataset = read_csv("cleaned_filtered_school_dataset.csv")
house_dataset = read_csv("cleaned_house_prices.csv")
colnames(school_dataset)
colnames(house_dataset)


# Clean postcodes (remove spaces for consistent joining)
house_clean <- house_dataset %>%
  select(postcode, price, county) %>%
  mutate(postcode = str_remove_all(postcode, " "))  # e.g. "HX28SN"

school_clean <- school_dataset %>%
  select(PCODE, ATT8SCR, County) %>%
  rename(postcode = PCODE, attainment = ATT8SCR) %>%
  mutate(postcode = str_remove_all(postcode, " "))  # e.g. "S639EW"

# Aggregate House Prices by Postcode
house_avg <- house_clean %>%
  group_by(postcode, county) %>%
  summarise(median_price = median(price, na.rm = TRUE), .groups = "drop")


# Merge School & House Datasets
merged_data <- inner_join(school_clean, house_avg, by = "postcode") %>%
  filter(!is.na(attainment) & !is.na(median_price))

# Merge School & House Datasets
merged_data <- inner_join(school_clean, house_avg, by = "postcode") %>%
  filter(!is.na(attainment) & !is.na(median_price))

# Plot (House Price vs Attainment Score)
ggplot(merged_data, aes(x = median_price, y = attainment, color = County)) +
  geom_point(alpha = 0.6, size = 2.5) +
  geom_smooth(method = "lm", se = TRUE, size = 1.2) +
  labs(
    title = "Attainment 8 Score vs House Price by County",
    x = "Median House Price (£)",
    y = "Attainment 8 Score",
    color = "County"
  ) +
  theme_minimal(base_size = 14)

# Linear Model Summary (Per County)
# West Yorkshire model
west <- merged_data %>% filter(County == "West Yorkshire")
model_west <- lm(attainment ~ median_price, data = west)
summary(model_west)

# South Yorkshire model
south <- merged_data %>% filter(County == "South Yorkshire")
model_south <- lm(attainment ~ median_price, data = south)
summary(model_south)

# Correlation Coefficients
cor_west <- cor(west$median_price, west$attainment, use = "complete.obs")
cor_south <- cor(south$median_price, south$attainment, use = "complete.obs")

cat("Correlation (West Yorkshire):", cor_west, "\n")
cat("Correlation (South Yorkshire):", cor_south, "\n")
```

LM_4_att8_drugoffense.R

Source on Save

```r
library(tidyverse)

school_dataset = read_csv("cleaned_filtered_school_dataset.csv")
crime_data = read_csv("cleaned_crime_combined_population.csv")

# Filter for Drugs only and year 2023
drug_data <- crime_data %>%
  filter(Crime.type == "Drugs", Year == 2023) %>%
  distinct(LSOA.code, .keep_all = TRUE)


# Calculate Drug Offense Rate per 10,000 people
drug_rates <- drug_data %>%
  group_by(Postcode, Year, Month) %>%
  summarise(
    drug_crimes = n(),
    population = sum(Population2023, na.rm = TRUE),
    .groups = "drop"
  ) %>%
  mutate(rate_per_10k = (drug_crimes / population) * 10000)

# Clean & Prepare School Dataset
school_clean <- school_dataset %>%
  select(PCODE, ATT8SCR, County) %>%
  rename(attainment = ATT8SCR) %>%
  mutate(PCODE = str_remove_all(PCODE, " "))

colnames(school_clean)
colnames(drug_rates)


# Merge Drug Offense Rates with School Attainment
merged_data <- inner_join(school_clean, drug_rates, by = c("PCODE" = "Postcode")) %>%
  filter(!is.na(attainment) & !is.na(rate_per_10k))

# Plot: Drug Crime Rate vs Attainment Score
ggplot(merged_data, aes(x = rate_per_10k, y = attainment, color = County)) +
  geom_point(alpha = 0.6, size = 2.5) +
  geom_smooth(method = "lm", se = TRUE, size = 1.2) +
  labs(
    title = "Attainment 8 Score vs Drug Offense Rate (per 10,000 people) in 2023",
    x = "Drug Offense Rate per 10,000 People",
    y = "Attainment 8 Score",
    color = "County"
  ) +
  theme_minimal(base_size = 14)


# Linear Model Summary (Per County)
west <- merged_data %>% filter(County == "West Yorkshire")
south <- merged_data %>% filter(County == "South Yorkshire")

model_west <- lm(attainment ~ rate_per_10k, data = west)
model_south <- lm(attainment ~ rate_per_10k, data = south)

summary(model_west)
summary(model_south)

# Correlation
cor_west <- cor(west$rate_per_10k, west$attainment, use = "complete.obs")
cor_south <- cor(south$rate_per_10k, south$attainment, use = "complete.obs")

cat("Correlation (West Yorkshire):", cor_west, "\n")
cat("Correlation (South Yorkshire):", cor_south, "\n")
```

LM_5_avgdownloadspeed_drugoff...

Source on Save | Run

```r
library(tidyverse)

# 4. Average Download speed vs Drug Offense Rate per 10000 people for both counties
# in single diagram (include linear model summary report and correlation)

broadband_data = read_csv("broadband_filtered_yorkshire.csv")
drugs_data = read_csv("cleaned_crime_combined_population.csv")


# STEP 1: Filter drug data for only "Drugs" crime
drug_data_filtered <- drugs_data %>%
  filter(Crime.type == "Drugs")

# STEP 2: Aggregate drug offenses per postcode and count per 10,000
drug_rates <- drug_data_filtered %>%
  group_by(Postcode, County) %>%
  summarise(
    TotalCrimes = n(),
    AvgPopulation = sum(Population2024, na.rm = TRUE)
  ) %>%
  mutate(RatePer10k = (TotalCrimes / AvgPopulation) * 10000)

# STEP 3: Select relevant broadband columns
broadband_clean <- broadband_data %>%
  select(postcode = "postcode.x", County, `Average download speed (Mbit/s)`)

# STEP 4: Join the datasets by pcds
joined_data <- inner_join(drug_rates, broadband_clean, by = c("Postcode" = "postcode"), suffix = c("_drug", "_broadband"))

# STEP 5: Clean NA values
final_data <- joined_data %>%
  filter(!is.na(`Average download speed (Mbit/s)`), !is.na(RatePer10k))

# STEP 6: Correlation
correlation <- cor(final_data$`Average download speed (Mbit/s)`, final_data$RatePer10k)
cat("Correlation:", round(correlation, 3), "\n")


# STEP 7: Linear model
lm_model <- lm(`Average download speed (Mbit/s)` ~ RatePer10k, data = final_data)
summary(lm_model)

# STEP 8: Plot
ggplot(final_data, aes(x = RatePer10k, y = `Average download speed (Mbit/s)`, color = County_broadband)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = TRUE) +
  labs(
    title = "Drug Offense Rate vs Average Download Speed",
    x = "Drug Offense Rate per 10,000",
    y = "Average Download Speed (Mbit/s)",
    color = "County"
  ) +
  theme_minimal()
```

LM_6_avgdownloadspeed_att8.R

Source on Save

```r
library(tidyverse)

# 6. Average download speed vs Attainment 8 score for both counties in single
# diagram (include linear model summary report and correlation)

broadband_data = read_csv("broadband_filtered_yorkshire.csv")
school_data = read_csv("cleaned_filtered_school_dataset.csv")

colnames(broadband_data)
colnames(school_data)

# 1. Clean column names if needed
colnames(broadband_data) <- make.names(colnames(broadband_data))
colnames(school_data) <- make.names(colnames(school_data))

# 2. Merge broadband and school data using postcode (remove space to match formats)
broadband_data <- broadband_data %>%
  mutate(postcode_nospace = gsub(" ", "", postcode_space))

school_data <- school_data %>%
  mutate(postcode_nospace = gsub(" ", "", PCODE))

merged_data <- inner_join(school_data, broadband_data, by = "postcode_nospace")

# 3. Filter for West Yorkshire and South Yorkshire
filtered_data <- merged_data %>%
  filter(County.x %in% c("West Yorkshire", "South Yorkshire")) %>%
  select(SCHNAME, ATT8SCR, `Average.download.speed..Mbit.s.`, County = County.x)

# 4. Remove rows with missing values
filtered_data <- filtered_data %>%
  filter(!is.na(ATT8SCR), !is.na(Average.download.speed..Mbit.s.))

# 5. Correlation
correlation <- cor(filtered_data$Average.download.speed..Mbit.s., filtered_data$ATT8SCR, use = "complete.obs")
cat("Correlation: ", correlation, "\n")


# 6. Linear model
model <- lm(ATT8SCR ~ Average.download.speed..Mbit.s., data = filtered_data)
summary(model)

# 7. Plot
ggplot(filtered_data, aes(x = Average.download.speed..Mbit.s., y = ATT8SCR, color = County)) +
  geom_point(alpha = 0.7) +
  geom_smooth(method = "lm", se = TRUE) +
  labs(
    title = "Average Download Speed vs Attainment 8 Score",
    subtitle = paste("Correlation:", round(correlation, 3)),
    x = "Average Download Speed (Mbit/s)",
    y = "Attainment 8 Score"
  ) +
  theme_minimal()
```

RecommendationSystem.R

recommendation_system.R

Source on Save

```r
library(tidyverse)
library(lubridate)

house_data = read_csv("cleaned_house_prices.csv")
crime = read_csv("cleaned_crime_combined_population.csv")
broadband_data = read_csv("broadband_filtered_yorkshire.csv")
schoolData = read_csv("cleaned_filtered_school_dataset.csv")

# Check original data dimensions
cat("Original data dimensions:\n")
cat(sprintf("house_data: %d rows\n", nrow(house_data)))
cat(sprintf("crime: %d rows\n", nrow(crime)))
cat(sprintf("schoolData: %d rows\n", nrow(schoolData)))
cat(sprintf("broadband_data: %d rows\n", nrow(broadband_data)))

# Check unique towns in each dataset
cat("\nUnique towns count:\n")
cat(sprintf("house_data towns: %d\n", length(unique(house_data$town_city))))
cat(sprintf("crime towns: %d\n", length(unique(crime$town_city))))
cat(sprintf("schoolData towns: %d\n", length(unique(schoolData$TOWN))))
cat(sprintf("broadband_data towns: %d\n", length(unique(broadband_data$town_city))))


# Housing Data Processing
selected_house <- house_data %>%
  mutate(TOWN = str_trim(toupper(town_city))) %>%
  group_by(TOWN) %>%
  summarise(avgPrice = mean(price, na.rm = TRUE)) %>%
  select(avgPrice, TOWN) %>%
  na.omit() %>%
  distinct()

# School Data Processing
selected_attainment8 <- schoolData %>%
  mutate(
    TOWN = str_trim(toupper(TOWN)),
    ATT8SCR_clean = suppressWarnings(as.numeric(as.character(ATT8SCR)))
  ) %>%
  filter(!is.na(ATT8SCR_clean), !TOWN == "SOUTH YORKSHIRE", !TOWN == "WEST YORKSHIRE") %>%
  group_by(TOWN) %>%
  summarize(avgAtt8 = mean(ATT8SCR_clean, na.rm = TRUE)) %>%
  ungroup()


# Broadband Data Processing
selected_broadband <- broadband_data %>%
  group_by(town_city) %>%
  mutate(
    AvgUpSpeed = `Average upload speed (Mbit/s)`,
    AvgDownSpeed = `Average download speed (Mbit/s)`
  ) %>%
  summarise(
    avg_down_speed = mean(AvgDownSpeed, na.rm = TRUE)
  ) %>%
  mutate(TOWN = str_trim(toupper(town_city))) %>%
  select(TOWN, avg_down_speed)

# Crime Data processsing
selected_crime <- crime %>%
  group_by(postcode) %>%
  summarize(
    crimeno = n(),
    TOWN = first(town_city)  # or use unique(Town_City)[1] for safety
  ) %>%
  select(postcode, crimeno, TOWN) %>%
  arrange(desc(crimeno))

cat(sprintf("crime towns: %d\n", length(unique(selected_crime$TOWN))))
colnames(selected_crime)

```

```r
#Combine All Data (using left_join to keep as many as possible)
ranking <- selected_house %>%
  left_join(selected_attainment8, by = "TOWN") %>%
  left_join(selected_broadband, by = "TOWN") %>%
  left_join(selected_crime, by = "TOWN") %>%
  na.omit()


Extremes <- ranking %>%
  summarise(
    minDownSpeed = min(avg_down_speed, na.rm = TRUE),
    maxDownSpeed = max(avg_down_speed, na.rm = TRUE),
    minAtt8 = min(avgAtt8, na.rm = TRUE),
    maxAtt8 = max(avgAtt8, na.rm = TRUE),
    minHousingPrice = min(avgPrice, na.rm = TRUE),
    maxHousingPrice = max(avgPrice, na.rm = TRUE),
    minCrimeRate = min(crimeno, na.rm = TRUE),
    maxCrimeRate = max(crimeno, na.rm = TRUE)
  )

print(Extremes)



finalRanking <- ranking %>%
  mutate(
    normDownSpeed = 10 * (avg_down_speed - Extremes$minDownSpeed) / (Extremes$maxDownSpeed - Extremes$minDownSpeed),
    normAtt8 = 10 * (avgAtt8 - Extremes$minAtt8) / (Extremes$maxAtt8 - Extremes$minAtt8),
    normHousingPrice = 10 * (1 - (avgPrice - Extremes$minHousingPrice) / (Extremes$maxHousingPrice - Extremes$minHousingPrice)),
    normCrimeRate = 10 * (1 - (crimeno - Extremes$minCrimeRate) / (Extremes$maxCrimeRate - Extremes$minCrimeRate)),
    finalPoints = normDownSpeed  + normAtt8 + normHousingPrice + normCrimeRate
  )

colnames(finalRanking)


houserank <- finalRanking %>%
  select(TOWN, avgPrice, normHousingPrice) %>%
  arrange(desc(normHousingPrice))  %>%
  distinct(TOWN,.keep_all = TRUE) %>%
  slice_head(n=10)

crimerank <- finalRanking %>%
  select(TOWN, normCrimeRate, crimeno) %>%
  arrange(desc(normCrimeRate))  %>%
  distinct(TOWN,.keep_all = TRUE) %>%
  slice_head(n=10)

schoolrank <- finalRanking %>%
  select(TOWN, avgAtt8, normAtt8) %>%
  arrange(desc(normAtt8))  %>%
  distinct(TOWN,.keep_all = TRUE) %>%
  slice_head(n=10) 1

broadbandrank <- finalRanking %>%
  select(TOWN, avg_down_speed, normDownSpeed) %>%
  arrange(desc(normDownSpeed)) %>%
  distinct(TOWN,.keep_all = TRUE) %>%
  slice_head(n=10)


final_rank <- finalRanking %>%
  select(TOWN, avgPrice, crimeno, avgAtt8, avg_down_speed, finalPoints) %>%
  mutate(finalPoints = finalPoints / 4) %>%
  arrange(desc(finalPoints)) %>%
  distinct(TOWN,.keep_all = TRUE) %>%
  slice_head(n=10)
```