**Data Analysis and Visualization Principles**

**Report**

**Crown Prosecution Service Case Outcomes by Principal Offence Category**

**1. Introduction and Executive Summary**

This project's data is obtained by downloading information from data.gov.uk website (link: [https://data.gov.uk/dataset/89d0aef9-e2f9-4d1a-b779-5a33707c5f2c/crown-prosecution-service-case-outcomes-by-principal-offence-category-data)](https://data.gov.uk/dataset/89d0aef9-e2f9-4d1a-b779-5a33707c5f2c/crown-prosecution-service-case-outcomes-by-principal-offence-category-data).%C2%A0). The CPS is responsible for prosecuting criminal cases investigated by law enforcement agencies in England and Wales. It operates independently, making decisions separate from the police and government. The dataset used spans from January 2014 to December 2018.

Convictions include cases where the defendant pleads guilty, convictions following a trial, and cases proven in the absence of the defendant. Unsuccessful outcomes encompass all results other than convictions, such as discontinuances, withdrawals, discharged committals, dismissals, acquittals, and administrative finalizations. Administrative finalizations occur when a case cannot proceed due to an outstanding arrest warrant, the defendant being untraceable for a summons, the defendant's death, or the defendant being deemed unfit to plead.

The offenses are divided into several categories, including homicide, offenses against the person, sexual offenses, burglary, robbery, theft and handling, fraud and forgery, criminal damage, drugs offenses, public order, motoring offenses, and all other offenses excluding motoring. The report does not specify the principal offense for cases resulting in administrative finalizations; instead, they are presented as a separate category.

This analysis is based on a diverse range of monthly data collected from various years. Although some of the appended months are not included in the existing archive, they have still been taken into consideration during the analysis.

This report provides a comprehensive outline for processing, cleaning, and analyzing a crime dataset. The data pre-processing and cleaning steps encompass several actions such as dropping percentage columns, adding and sorting by date, shifting columns, removing special characters, converting to integers, and handling missing data. The report also offers descriptive analytics of the cleaned dataset, examining attributes such as county, year, month, region, and various types of crimes including homicide, sexual offenses, burglary, and more. Additionally, the report includes code snippets to demonstrate the implementation of the pre-processing and cleaning procedures.

Moreover, the report delves into predictive analytics. It encompasses linear regression, clustering, and classification. In the section on linear regression, the report covers hypotheses, dataset summaries, model summaries, accuracy matrices, predicted versus actual plots, and evaluation statistics. It also provides code snippets for data splitting and the execution of linear and multiple regression.

In terms of clustering, the report explores K-means, featuring hypotheses, dataset summaries, and summaries. Code snippets are also included to guide the removal of non-numeric columns and the implementation of K-means.

Furthermore, the report examines SVM for classification purposes. It presents hypotheses, dataset summaries, model summaries, classification reports, model ROC curves, and code snippets for SVM.

**2. Data Overview**

The report makes use of the Crown Prosecution Service Case Outcomes by Principal Offense Category (POC) dataset, sourced from the data.gov.uk website. This dataset comprises information on outcomes from the CPS, classified into convictions and unsuccessful verdicts. It covers a time span from 2014 to 2018, with data collected on a monthly basis. The dataset pertains to forty-two counties across England, where applicable.

Convictions in the dataset encompass a range of scenarios, including guilty pleas, convictions resulting from trials, and verdicts issued in the absence of the defendant appearing in court. Incomplete outcomes encompass all other categories, including discontinuances, withdrawals, discharged committals, dismissals, acquittals, and administrative finalizations.

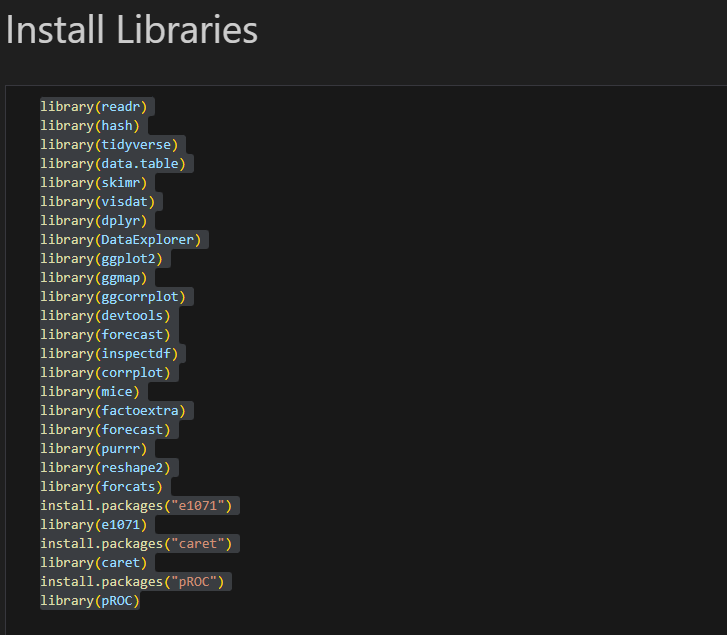
The recorded offenses in the dataset include homicide, violations against individuals like sexual assault, burglary, robbery, theft, fraud or forgery, and criminal damage to public places and vehicles. All offenses, except those related to motoring, are grouped within this category.

**3. Install Libraries**

**3.1 Report on R libraries used**

1. Library (readr): The 'readr' library offers a quick and effective method for reading structured data files into R. It provides tools for reading fixed-width formatted files, CSV files, and delimited text files. It makes importing data easier and guarantees the accuracy and consistency of the data.
2. Library (hash): A hash table implementation in R is offered via the 'hash' library. It enables effective key-value pair storage and retrieval, which is helpful for operations like data lookup, caching, and developing distinctive IDs. The library provides several collision-resolution techniques and hash algorithms.
3. Library (tidyverse): The 'tidyverse' is a group of R packages, such as 'dplyr', 'ggplot2', 'tidyr', and others, that cooperate to offer a standardised and effective method for manipulating and displaying data. It offers a potent set of tools for data manipulation and exploration and encourages the application of tidy data concepts.
4. Library (data.table): The 'data.table' module improves the data manipulation framework and expands R's data.frame features. It enables data aggregation, filtering, and combining operations that are quick and memory-efficient on huge datasets.
5. Library (skimr): A useful method for producing summary statistics and visualisations for data exploration is offered by the "skimr" library. It provides functions to calculate distributions, missing value data, and descriptive statistics for each variable in a dataset. The library facilitates rapid comprehension of the data's structure and content.
6. Library (visdat): The 'visdat' library enables visualizing missing values in a dataset. It provides tools to create informative visualizations that highlight the presence and patterns of missing data. This can aid in identifying potential issues and deciding on appropriate strategies for handling missing values.
7. Library (dplyr): The 'dplyr' library is a fundamental package within the tidyverse that provides a set of verbs for data manipulation. It offers functions for filtering rows, selecting columns, mutating variables, grouping data, and performing various summarization operations. The library follows a consistent grammar that makes it easy to express complex data manipulations concisely.
8. Library (DataExplorer): The 'DataExplorer' library facilitates exploratory data analysis by automating common data exploration tasks. It provides functions to generate comprehensive data summaries, visualizations, and interactive reports. The library helps in understanding the structure, relationships, and quality of the data.
9. library(ggplot2): The 'ggplot2' library is a popular data visualization package in R. It follows the grammar of graphics paradigm and offers a flexible and layered approach to creating visualizations. It provides a wide range of customization options, supports various plot types, and allows for the creation of publication-quality graphics.
10. library(ggmap): The 'ggmap' library extends the functionality of 'ggplot2' by incorporating maps into visualizations. It allows for the visualization of spatial data and facilitates the integration of geographic information with other data variables. The library provides functions to retrieve map data from different sources, overlay data onto maps, and customize map aesthetics.
11. library(ggcorrplot): The 'ggcorrplot' library specializes in creating correlation matrices and correlation plots. It offers functions to compute and visualize correlation coefficients using various methods. The library supports customization of correlation plots, including color schemes, size, and annotation options.
12. library(devtools): The 'devtools' library is a development toolkit that simplifies the process of building, testing, and sharing R packages. It provides functions for package installation, package development workflow, and version control integration. The library streamlines package development tasks and facilitates collaboration among developers.
13. library(forecast): The 'forecast' library is focused on time series forecasting. It offers a wide range of forecasting models, including exponential smoothing, ARIMA, state space models, and more. The library provides functions for automatic model selection, model diagnostics, and prediction intervals.
14. library(inspectdf): The 'inspectdf' library helps in exploring and profiling data frames. It provides functions to generate descriptive statistics, data quality metrics, and variable distributions. The library also allows for interactive exploration of data frames to identify potential issues and gain insights into the data.
15. library(corrplot): The 'corrplot' library specializes in creating correlation matrices and correlation plots. It offers various visualization methods for correlation matrices, including color-coded matrices, scatterplots, and network plots. The library supports customization of correlation plots and provides options for highlighting significant correlations.
16. library(mice): The 'mice' library is used for multiple imputation of missing data. It provides functions to impute missing values using various imputation methods, including predictive mean matching, random forests, and linear regression. The library supports imputation for both continuous and categorical variables.
17. library(factoextra): The 'factoextra' library complements the 'factoMineR' package and facilitates the interpretation of multivariate analysis results. It provides functions for visualizing and extracting information from principal component analysis (PCA), clustering, and factor analysis results. The library supports the creation of informative plots and tables.
18. library(purrr): The 'purrr' library enhances functional programming in R by providing a consistent set of tools for working with functions and vectors. It offers functions for iteration, mapping, filtering, and other operations on lists, vectors, and data frames. The library promotes a concise and expressive coding style.
19. library(reshape2): The 'reshape2' library provides functions for reshaping data between wide and long formats. It facilitates the restructuring of data frames, allowing for easier manipulation and analysis. The library supports various data aggregation and transformation operations.
20. library(forcats): The 'forcats' library focuses on working with categorical variables. It offers functions for managing and modifying factor levels, reordering factor levels based on custom criteria, and converting between factors and character variables. The library simplifies categorical data handling and preparation.
21. library(e1071): The 'e1071' library provides functions for statistical modeling and machine learning. It offers implementations of various algorithms, including support vector machines (SVM), naive Bayes, and clustering methods. The library supports classification, regression, and clustering tasks.
22. library(caret): The 'caret' (Classification And REgression Training) library is a comprehensive toolkit for machine learning in R. It provides a unified interface for building and evaluating predictive models. The library offers functions for data preprocessing, model training, feature selection, and performance evaluation.
23. library(pROC): The 'pROC' library focuses on analyzing and visualizing receiver operating characteristic (ROC) curves. It provides functions to compute ROC curves, calculate area under the curve (AUC) values, and compare different classification models. The library supports customization of ROC plots and provides tools for statistical inference.

In addition to the provided libraries, the code also includes the installation of the 'e1071', 'caret', and 'pROC' packages to ensure they are available for use.



Top of Form

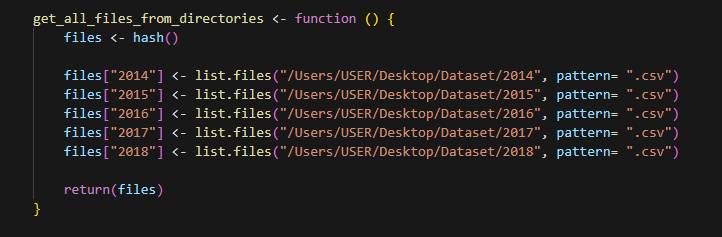
**4. Dataset integration**

Each year's dataset was dispersed in the form of a directory, with each directory including all data that could be found for that year's parent directory. The selected approach was to first construct a general function that reads all potential files within the directories and creates a hashmap as a response to the dataset structure. Then, another function reads the hashmap and merges all the files into a single dataframe, creating columns for year and month that are derived from the directory name and the file name, respectively, in the process.

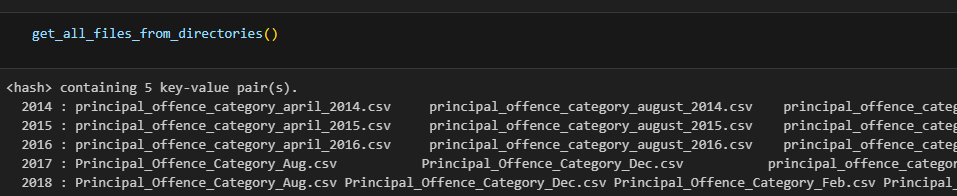
The code utilized for the dataset integration are:

**4.1 Get\_all\_files\_from\_directories**

This function is used to fetch all the files listed under specified directories and return a hash containing directories and filenames.

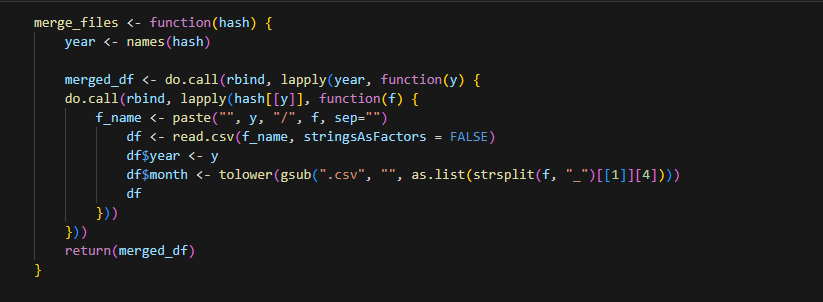


Below is the imge of the fetched dataset from the directory:

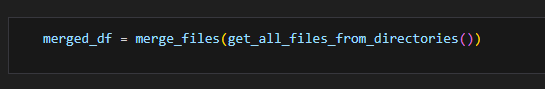


**4.2 Merge\_files**

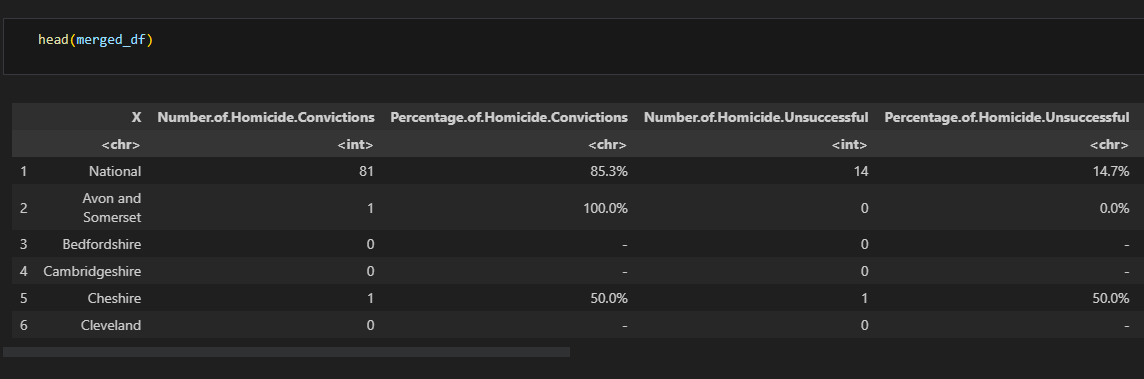
The provided function processes a hashmap as input. It proceeds by reading multiple files and combining them into a unified dataframe. During this process, the code generates appropriate columns to differentiate the data and maintain their individual characteristics.



merge\_files is used to create a dataframe which is merged\_df. See image below.



Below is the head(merged\_df) showing the first 6 rows. See image below

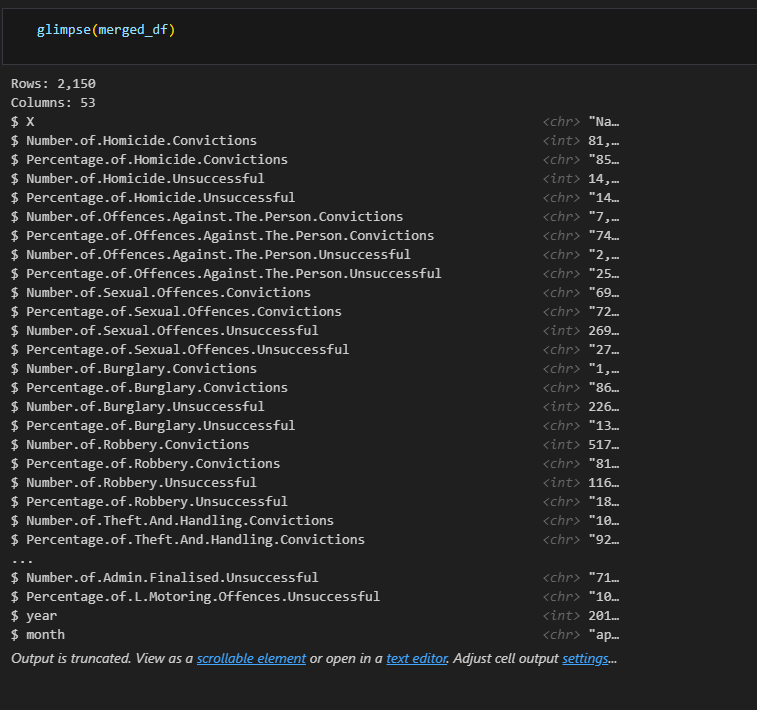


With the write.csv code, merged\_df becomes a csv file.



**4.3 Shape of the integrated dataset**

After the dataset was integrated, it took a shape. With the use of glimpse(merged\_df), the dataset takes the shape below:



**5. Data Preparation**

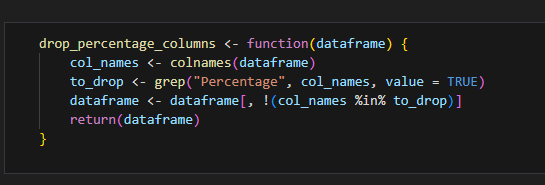
Data preparation, also known as data pre-processing, is a crucial step in data analysis and machine learning workflows. It involves transforming raw data into a format that is suitable for analysis, modelling, or feeding into machine learning algorithms. The goal of data preparation is to ensure data quality, improve the efficiency and effectiveness of subsequent analysis, and enhance the performance of machine learning models.

**5.1 Data Cleaning**

This involves handling missing values, correcting data errors, dealing with outliers, and resolving inconsistencies in the data. Missing values can be imputed using appropriate techniques, such as mean imputation or interpolation. Outliers can be identified and either removed or treated using statistical methods. The sections below contain each operation performed with reasoning and impact.

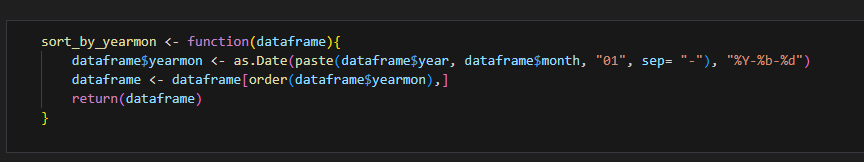
**5.1.1 Drop percentage columns**

By dropping percentage columns, the data set is made more manageable and the analysis can be focused on the most important variables. Additionally, percentage columns can be dropped when the data is skewed, which can cause problems in modeling and can lead to inaccurate results. Moreover, if required they can be re-generated using other attributes. One of the justification for dropping these columns is the redundancy of the columns. The functions drops all the columns that keyword “Percentage” in their column name from the dataframe. See image below:



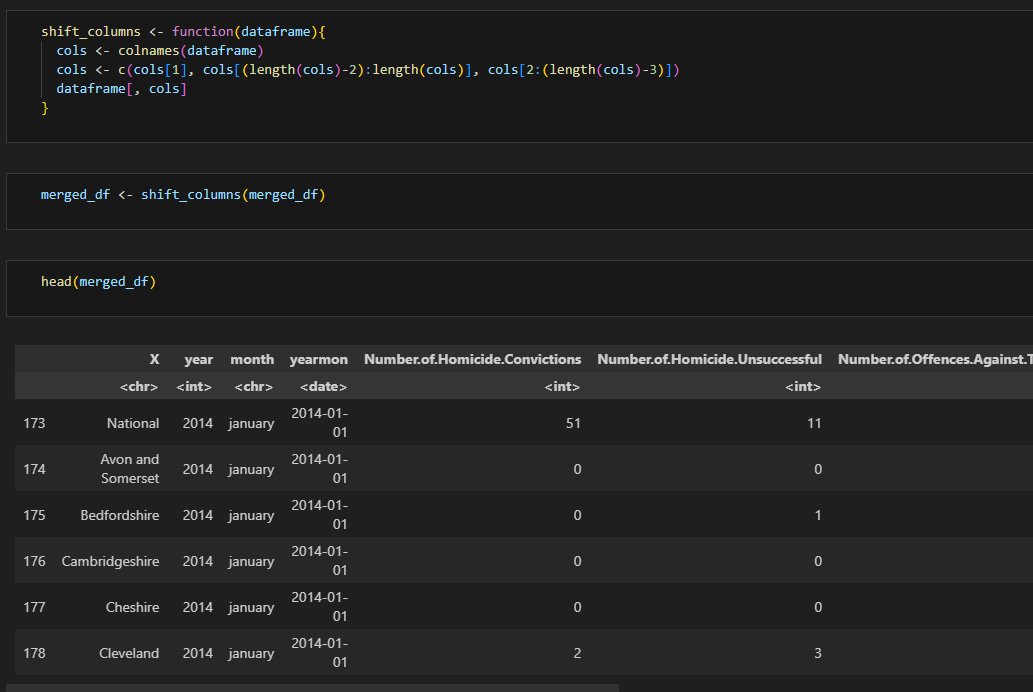
**5.1.2 Sort by date**

During the data cleaning process, a new column is introduced to the dataset by merging the year and month columns, thereby creating a date field. This inclusion of a date field serves a valuable purpose in sorting the dataset chronologically, facilitating the analysis of data trends over time. The effect of this operation is two-fold: it enhances the dataset's readability and usability for analysis by introducing a date field, and it enables sorting the data based on this field, simplifying the tracking of patterns and trends over time. The underlying rationale behind this approach is to enhance data usability, enabling more efficient analysis and facilitating the identification of temporal patterns and trends.



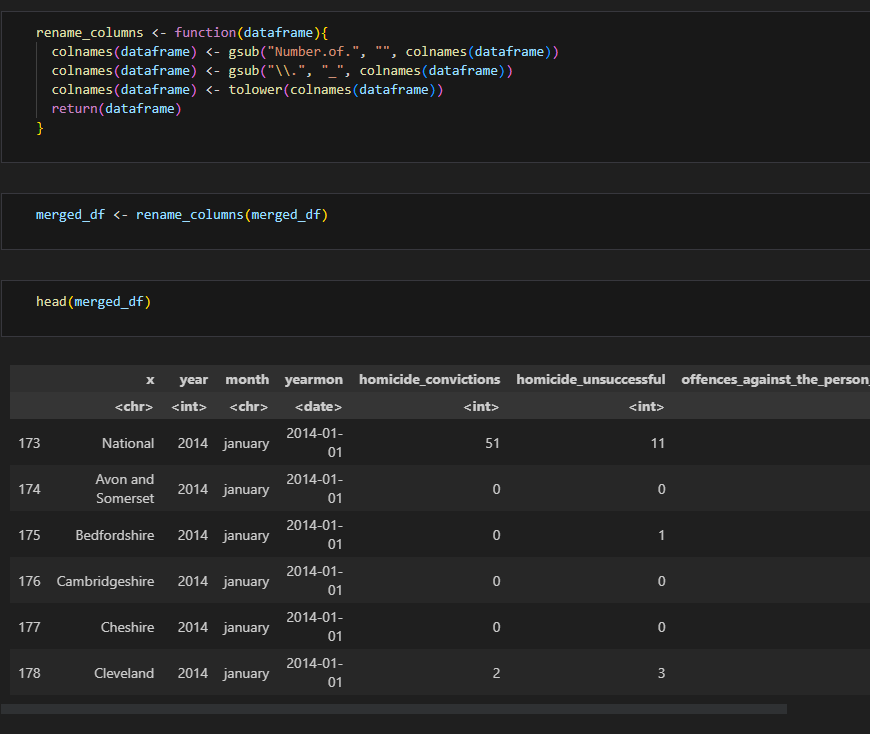
**5.1.3. Shift Columns**

This operation aims to restructure the dataset, enhancing its navigability, readability, comprehension, interpretation, and utility. The function shifts the last two columns after the first column.



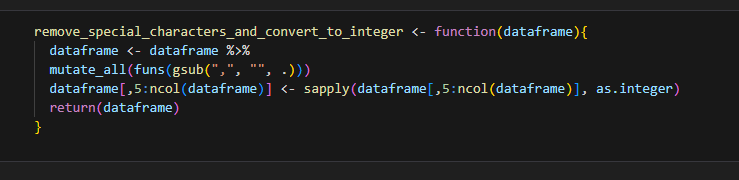
5.1.4 Rename Columns

The purpose of this operation is to eliminate redundant text from column names, resulting in shorter, more readable, and user-friendly names. By removing repetitive keywords such as "Number.of." from the column names, this function streamlines their readability and facilitates visualization during different analyses. The operation effectively renames the columns to enhance their clarity and ease of use.



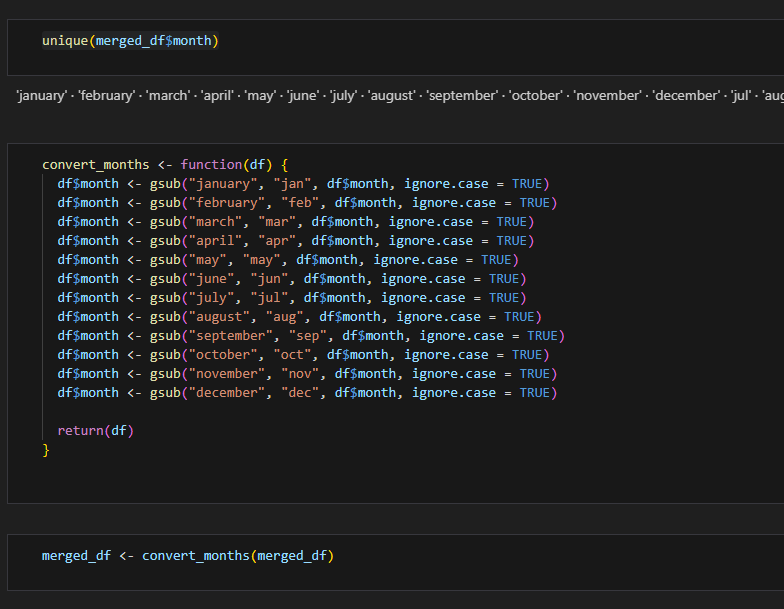
**5.1.5 Remove special characters and convert to integer**

This operation focuses on eliminating unnecessary text, such as commas, from column values and converting them to the integer data type. By doing so, the data becomes more suitable for seamless data analysis. Many models and algorithms perform optimally with integer inputs, making this conversion beneficial. The primary objective is to ensure that the columns in the data frame are primarily represented as integers. Additionally, the function is employed to remove special characters and convert most of the columns in the data frame to the integer data type.



**5.1.6 Rename month values**

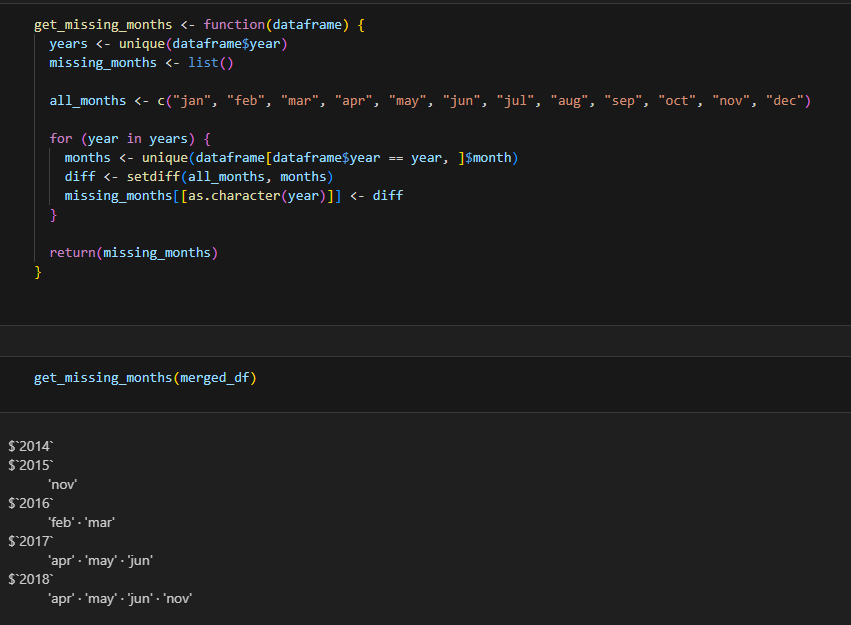
In the given figure, it is observed that there are multiple representations of the same month within the dataframe, leading to inconsistencies. To address this issue and eliminate noise, this operation standardizes all the month representations into a unified format. The code specifically renames the month values to ensure a consistent and uniform format throughout the dataset. By doing so, it resolves the discrepancy and ensures accurate and coherent month representation across the dataframe. The first step is to use unique(merged\_df$month) to get the unique month values.



**5.1.7. Missing data in year column**

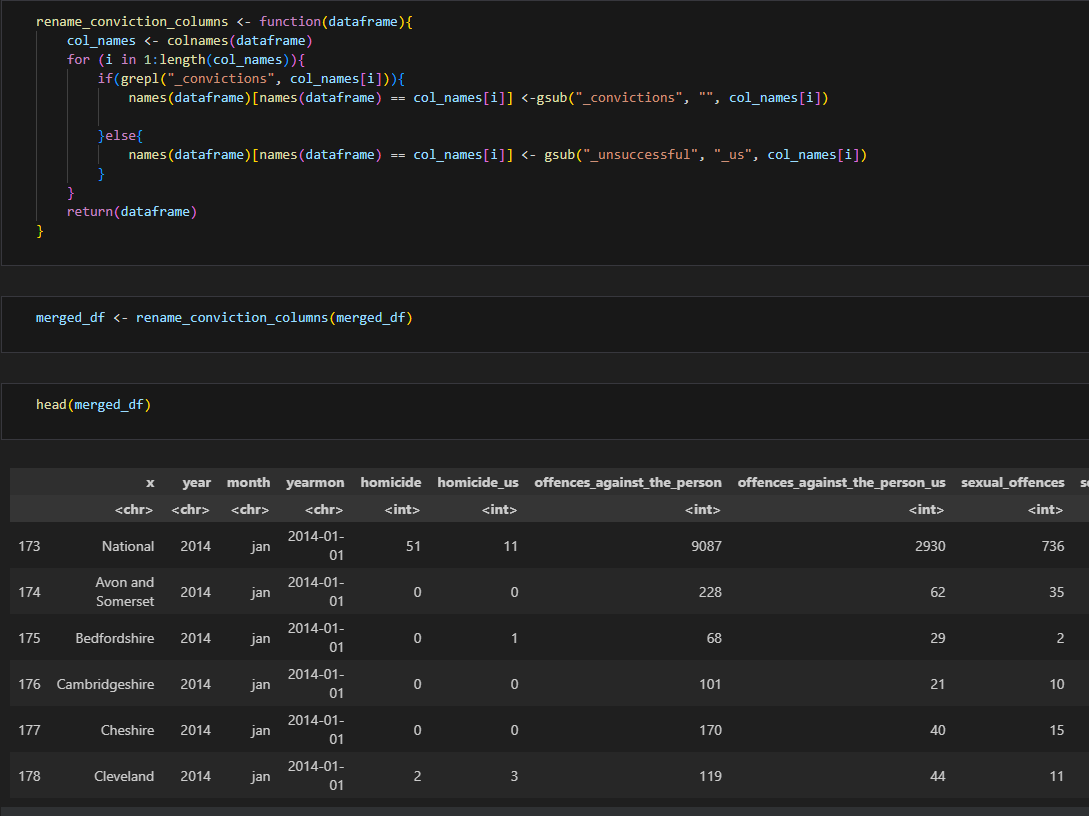
The purpose of this column is to visually represent the missing months within each respective year in the dataset. In this particular analysis, the missing data was not imputed initially. This approach is followed in scenarios where the missing data is not missing at random or when strong assumptions about its generation are lacking. Imputing the missing data in such cases can introduce bias and potentially lead to inaccurate analysis results.

During the initial stages of analysis, the missing data is retained in its original state without imputation. However, as the analysis progresses and when necessary, the missing data can be imputed at a later stage. The function provided returns all the missing months' data under their respective years, highlighting the gaps in the dataset. This enables further investigation and potential imputation of missing values as needed, taking into account the specific requirements of the analysis.



**5.1.8 Rename conviction column**

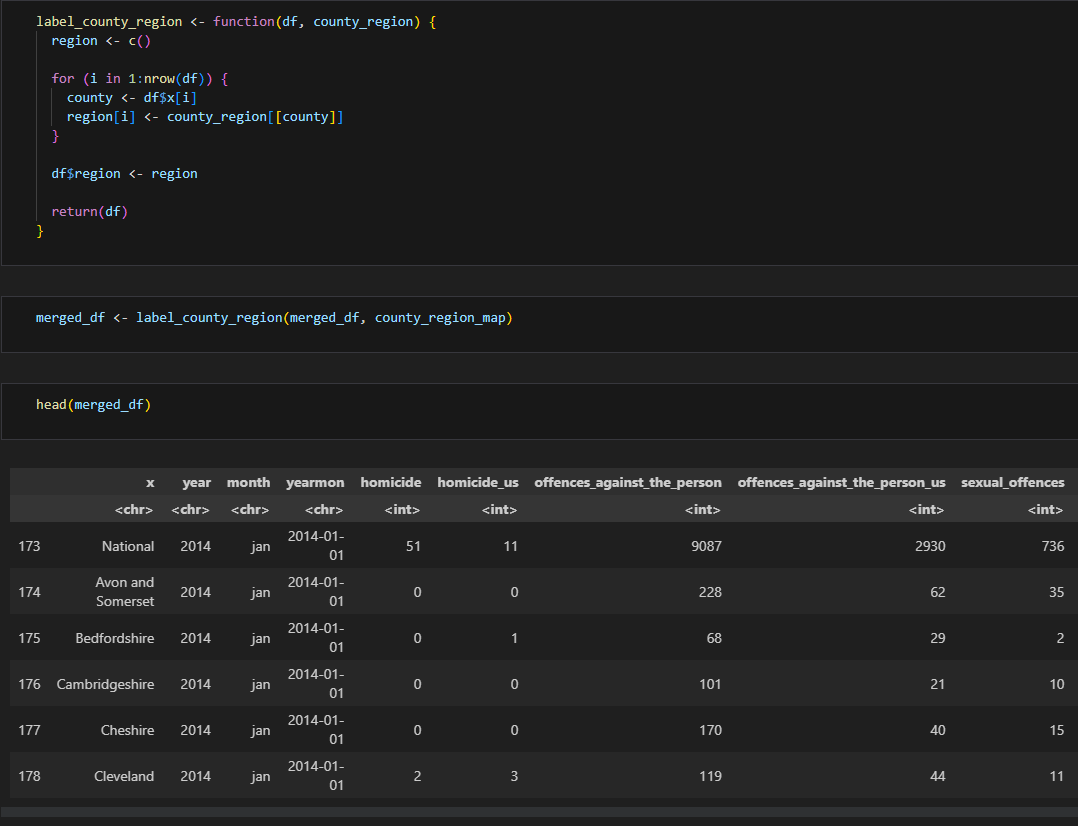
The purpose of this operation was to further shorten column names by eliminating the term "convictions" from crime column names and abbreviating "unsuccessful" as "us" in the column names. This function is responsible for renaming the columns in a dataframe, removing the "\_convictions" and "\_unsuccessful" substrings from the column names. If the latter substring exists, it is replaced with "\_us".



**5.1.9. Label region based on map**

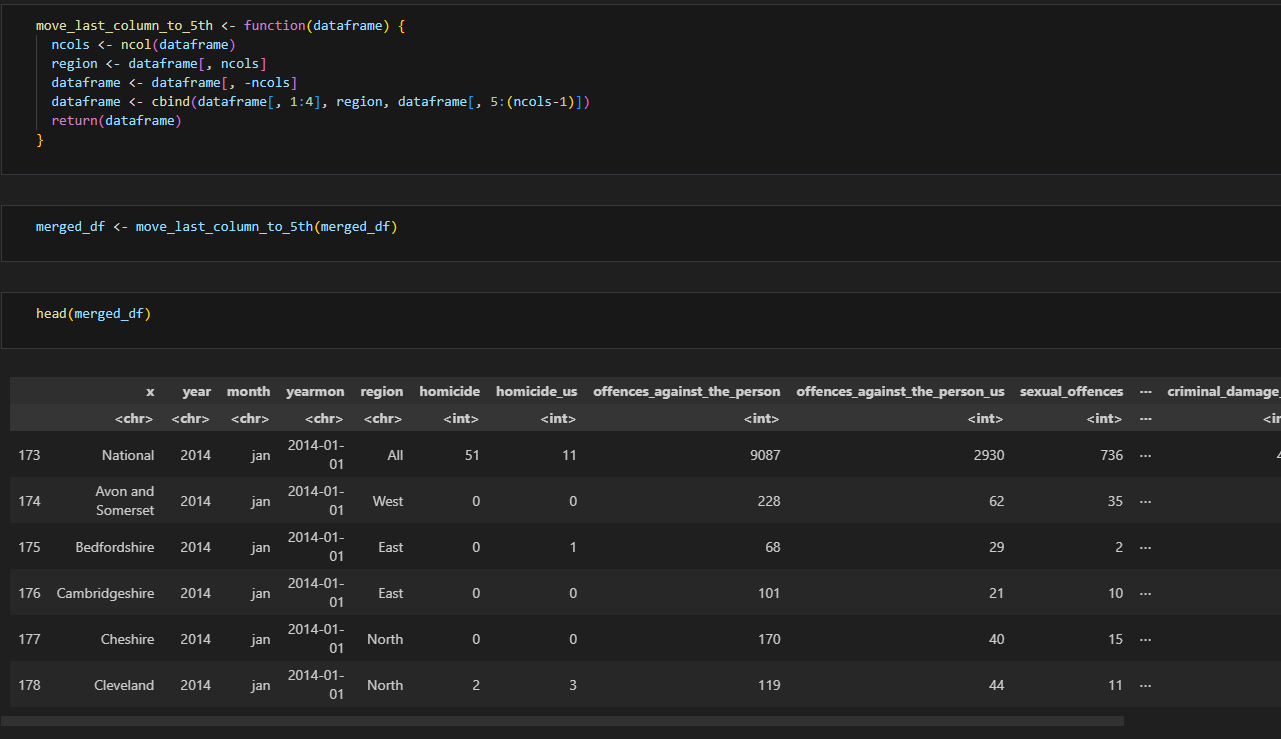
This operation involves the creation of a new column in the dataframe called "region." The values for this column are determined by referencing a lookup table called county\_region\_map, which maps county values to corresponding region values. This process is beneficial for grouping and analyzing data at the regional level, rather than individually by counties. It also facilitates the creation of aggregate statistics for regions, leading to a more organized and comprehensible dataset, which in turn enables more efficient analysis. By dividing the county data into regions, this operation introduces a new level of analysis that enhances the usability of the data. It aids in tracking patterns and trends specific to regions, which can be more meaningful for certain types of analysis. The function responsible for this operation creates the new column "region" and assigns labels to its values based on the input map and the existing values in the county column.





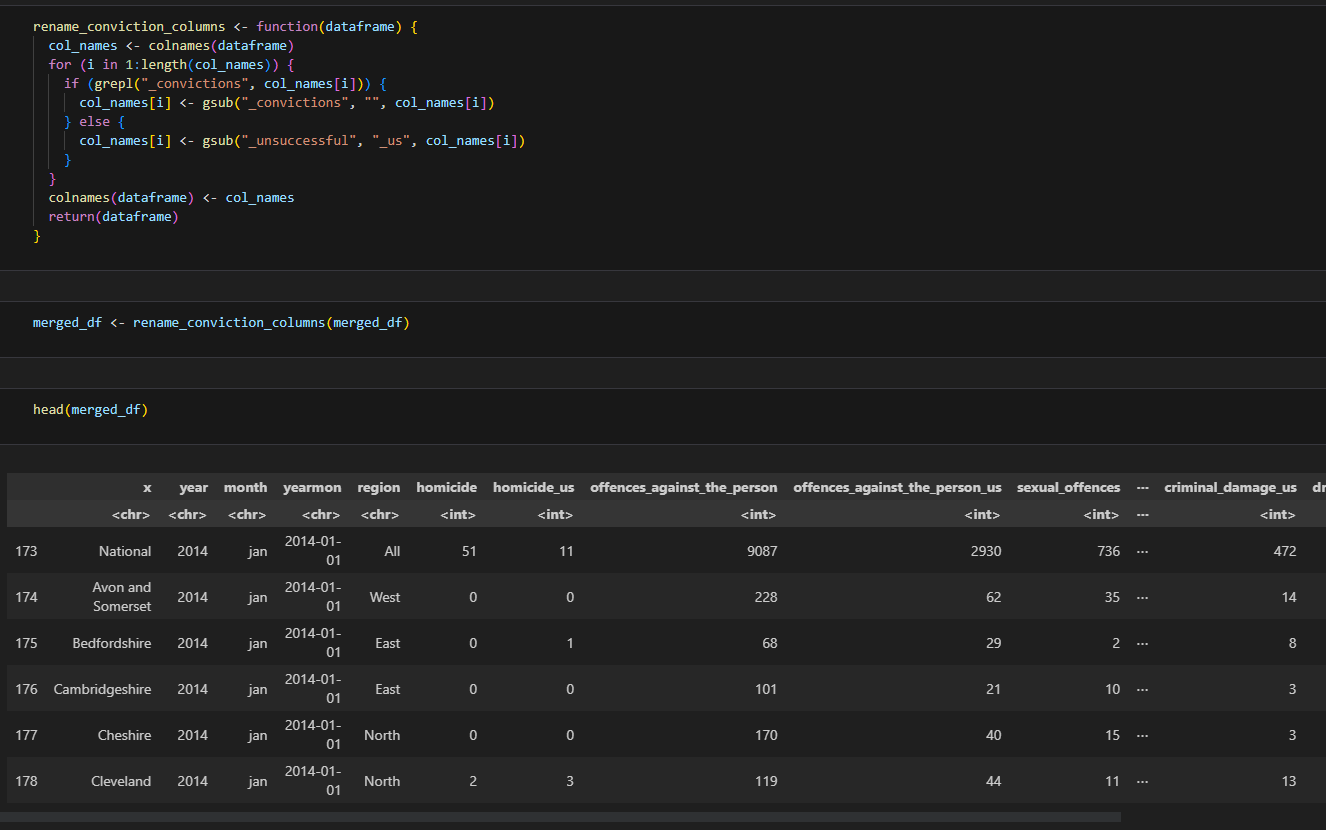
**5.1.10. Move last column to 5th column**

The purpose of this function is to rearrange the order of columns in a dataset, enhancing its manageability and usability. When given a dataframe as input, the function moves the last column to the 5th position and then returns the modified dataframe. By repositioning the columns in this manner, the function aims to improve the organization and facilitate more efficient utilization of the dataset.



**5.1.11. Rename conviction column**

This operation aimed to further abbreviate column names for increased conciseness. It involved removing the term "convictions" from crime column names and replacing "unsuccessful" with the shorter form "us" in the column names. The function responsible for this task renames the columns of a dataframe by eliminating the "\_convictions" and "\_unsuccessful" substrings from the column names. If the latter substring exists, it is replaced with "\_us". The objective is to create more compact and streamlined column names, improving readability and potentially enhancing data analysis efficiency.



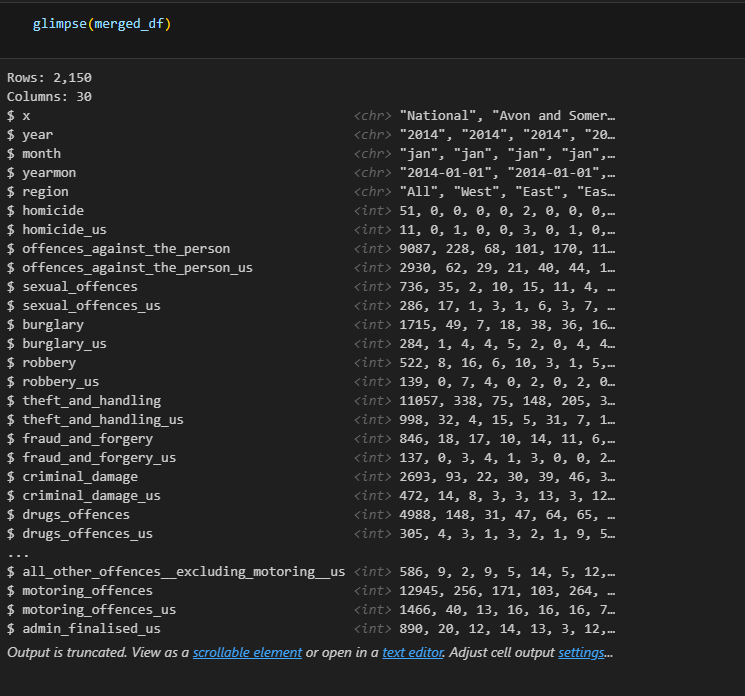
**5.2. Dataset new shape**

After the cleaning had been done, the dataset takes a new shape as visualized below.

**5.2.1 Glimpse**

The "glimpse" function is a data exploration tool in R, commonly used with the "dplyr" package. It provides a concise summary of a dataset by displaying a glimpse of the data. When applied to a dataframe or a tibble object, the "glimpse" function returns a compact overview of the dataset, showing the column names, data types, and a few sample rows.

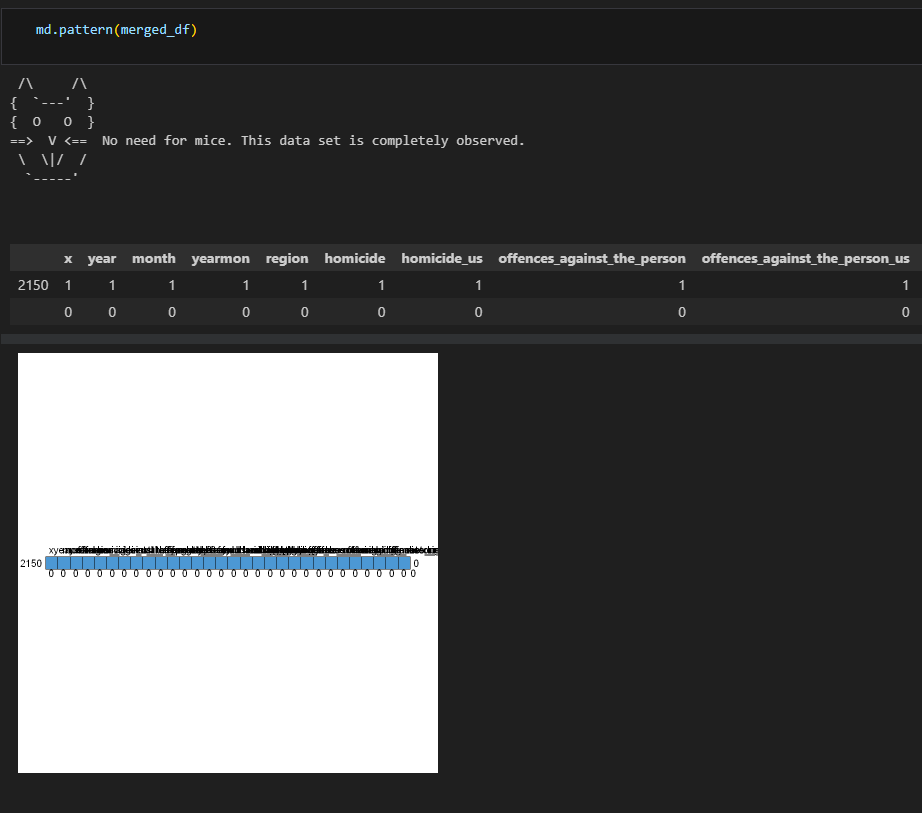
The "glimpse" function is similar to the "str" function in R but offers a more condensed and user-friendly output. It is particularly useful for quickly understanding the structure and content of a dataset, especially when dealing with large datasets with numerous columns. By displaying a concise summary, it allows users to get a quick overview of the data and make informed decisions about further data manipulation or analysis.



**5.2.2. Visualizing missing data**

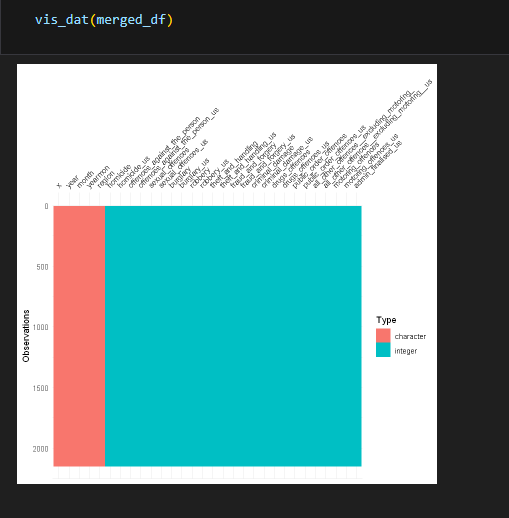
Visualization of missing data was done with vis\_miss and md.pattern. All the rows have complete data and there are no null values.





**5.2.3. Visualizing the different data types**

With the use of vis\_dat, I was able to visualize the different type of data. The successful and unsuccessful crime columns have been successfully converted to Integer types, accounting for approximately 80% of the dataframe. The remaining 20% consists of other columns unrelated to crime. County (which is x), year, month, yearmon, region remains as character.



**5.3. Splitting the dataset**.

It is imperative to state that all the data present in the dataset given are being used in this project. This project is not limited to 24 months. All the data present in 2014 to 2018 are being used in the analysis.

To optimize our dataset analysis, we classified the integer columns into two main categories: Crimes and Unsuccessful crimes. This division allows us to eliminate redundant filtering operations and provides a focused foundation for forming hypotheses and conducting various analysis tasks. By splitting the dataset based on these two distinct types, we streamline the analytical process and enhance the efficiency of subsequent operations.

Explained below are the reasons for splitting the dataset:

Splitting the dataset in this project is an essential step that helps in building reliable and effective models. It involves dividing the available data into different subsets for training, validation, and testing purposes. Here are some reasons why it is important to split the dataset:

1. Model Training: The primary reason for splitting the dataset is to train a machine learning or statistical model. By using a portion of the data for training, the model can learn patterns, relationships, and trends in the data, enabling it to make predictions or classifications accurately.

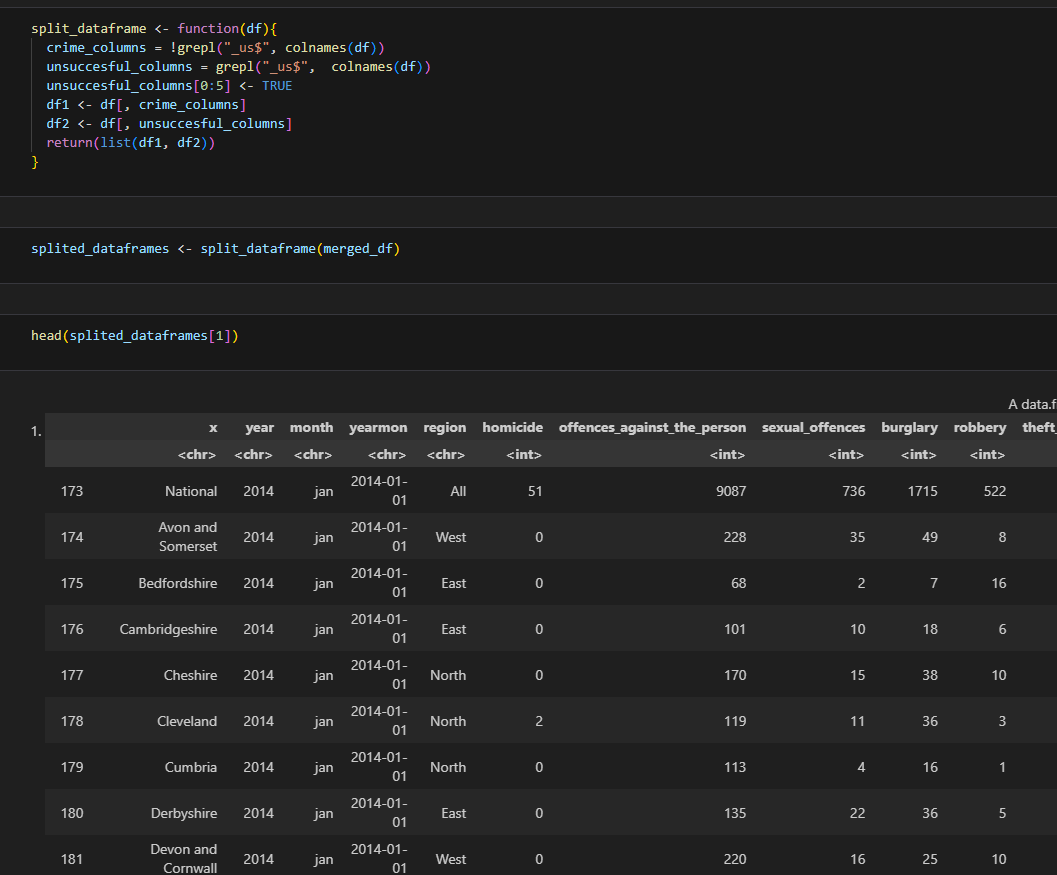
2. Model Evaluation: After training the model, it is crucial to evaluate its performance. Splitting the dataset allows you to assess how well the model generalizes to unseen data. By using a separate validation set, you can tune the model's hyperparameters and make adjustments to improve its performance.

3. Preventing Overfitting: Overfitting occurs when a model performs exceptionally well on the training data but fails to generalize to new, unseen data. By splitting the data, you can identify if the model is overfitting. If the model performs well on the training set but poorly on the validation or test set, it suggests overfitting. Adjustments can then be made to the model to reduce overfitting, such as regularization techniques.

4. Unbiased Performance Estimates: Splitting the dataset into separate training, validation, and test sets provides unbiased estimates of the model's performance. The model is trained on the training set, optimized using the validation set, and ultimately evaluated on the test set, which represents unseen data. This approach ensures that the model's performance metrics, such as accuracy or error rate, are reliable indicators of how it would perform in real-world scenarios.

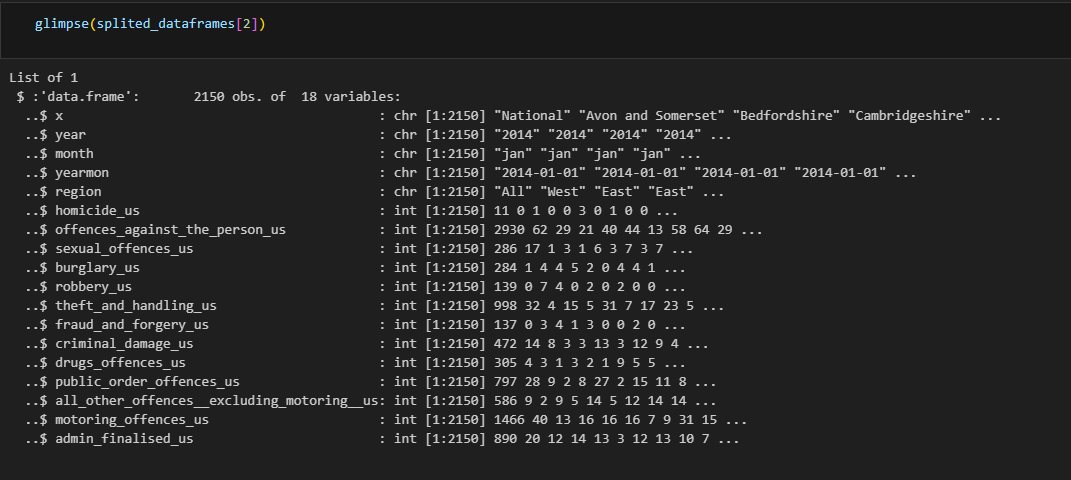
**5.3.1. Crime dataframe dataset split**

There bare a number of process involved in the plitting of the crime\_df dataset. This section will show the a series of tables/images that shows the steps involved in splitting the dataframe and the result after the glimpse() was applied.



**5.3.2. Unsuccessful Crime Dataset Glimpse**

The provided dataset offers a preview of rows, columns, and a few initial values from the Unsuccessful Crime dataset. This dataset focuses on cases where users were neither admitted nor found guilty of committing the respective crime. The columns represent various attributes related to these unsuccessful cases.



**6. Descriptive Analytics**

Descriptive analytics is a branch of data analytics that focuses on summarizing and interpreting historical data to gain insights and understand patterns, trends, and characteristics of a dataset. It involves the use of statistical and visualization techniques to describe and present data in a meaningful way. The primary objective of descriptive analytics is to provide a clear and comprehensive understanding of past events or phenomena.

Here are some key aspects of descriptive analytics:

1. Data Summary: Descriptive analytics begins with summarizing the dataset by calculating measures such as mean, median, mode, range, standard deviation, and quartiles. These summary statistics provide a concise overview of the data distribution and central tendencies.

2. Data Visualization: Visualizations play a crucial role in descriptive analytics as they help in presenting data in a visually appealing and understandable manner. Common visualization techniques include histograms, bar charts, line graphs, scatter plots, and pie charts. These visuals aid in identifying patterns, trends, and relationships within the data.

3. Exploratory Data Analysis (EDA): EDA is a key component of descriptive analytics. It involves examining the dataset to discover insights and patterns that may not be immediately apparent. Techniques such as grouping, filtering, and sorting the data are used to explore relationships, distributions, and outliers. EDA helps in formulating hypotheses and generating initial insights before moving on to more advanced analytics.

4. Data Interpretation: Descriptive analytics aims to interpret the findings obtained from analyzing the dataset. It involves extracting meaningful information, identifying important patterns, and drawing conclusions based on the observed data. Interpretation often involves making comparisons, identifying trends over time, or investigating relationships between variables.

5. Data Reporting: Descriptive analytics typically culminates in the creation of reports or presentations that convey the key findings and insights from the analysis. These reports are often shared with stakeholders, decision-makers, or other relevant parties to facilitate understanding and support data-driven decision-making processes.

Some examples of descriptive analysis include:

* Summarizing a dataset using measures of central tendency and dispersion
* Creating a frequency distribution table or a histogram to show the distribution of a variable
* Creating a box plot to show the distribution of a variable and identify outliers
* Creating a scatter plot to show the relationship between two variables.

Descriptive analytics provides a foundational understanding of the dataset, enabling organizations and individuals to gain insights, identify trends, and make informed decisions based on historical data. It serves as a crucial starting point for further advanced analytics, such as predictive or prescriptive analytics, which involve using historical data to make predictions or optimize future outcomes.

**6.1. Attribute Analysis**

Attribute analysis, also known as feature analysis or variable analysis, is a process in data analysis where the attributes or variables of a dataset are examined to understand their characteristics, significance, and impact on the data. It involves assessing the individual attributes in terms of their descriptive statistics, distributions, relationships with other variables, and relevance to the analysis objectives. Attribute analysis plays a vital role in data exploration, feature engineering, and modeling tasks.

Here are some key aspects of attribute analysis:

1. Descriptive Statistics: Attribute analysis starts by calculating basic descriptive statistics for each attribute, such as mean, median, mode, range, standard deviation, and quartiles. These statistics provide insights into the central tendency, spread, and variability of the attribute values.

2. Distribution Analysis: Analyzing the distribution of attribute values helps understand their frequency and spread. Histograms, density plots, and box plots are commonly used to visualize and analyze attribute distributions. Skewness, kurtosis, and normality tests may also be applied to assess the shape and characteristics of the distributions.

3. Correlation and Relationships: Attribute analysis involves examining the relationships between attributes. Correlation analysis, scatter plots, and correlation matrices are employed to identify correlations and dependencies between attributes. This helps understand how attributes influence each other and whether any attribute redundancy exists.

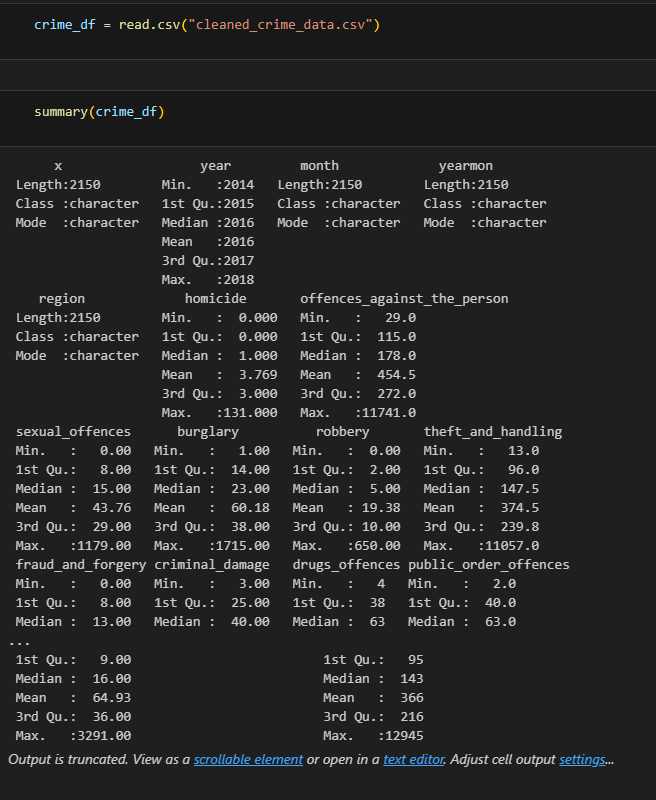
4. Missing Data and Outliers: Analyzing attributes also involves investigating missing data and outliers. Missing data analysis helps identify the extent and patterns of missing values, while outlier detection techniques (e.g., z-score, box plots, or clustering methods) help identify extreme or unusual observations.

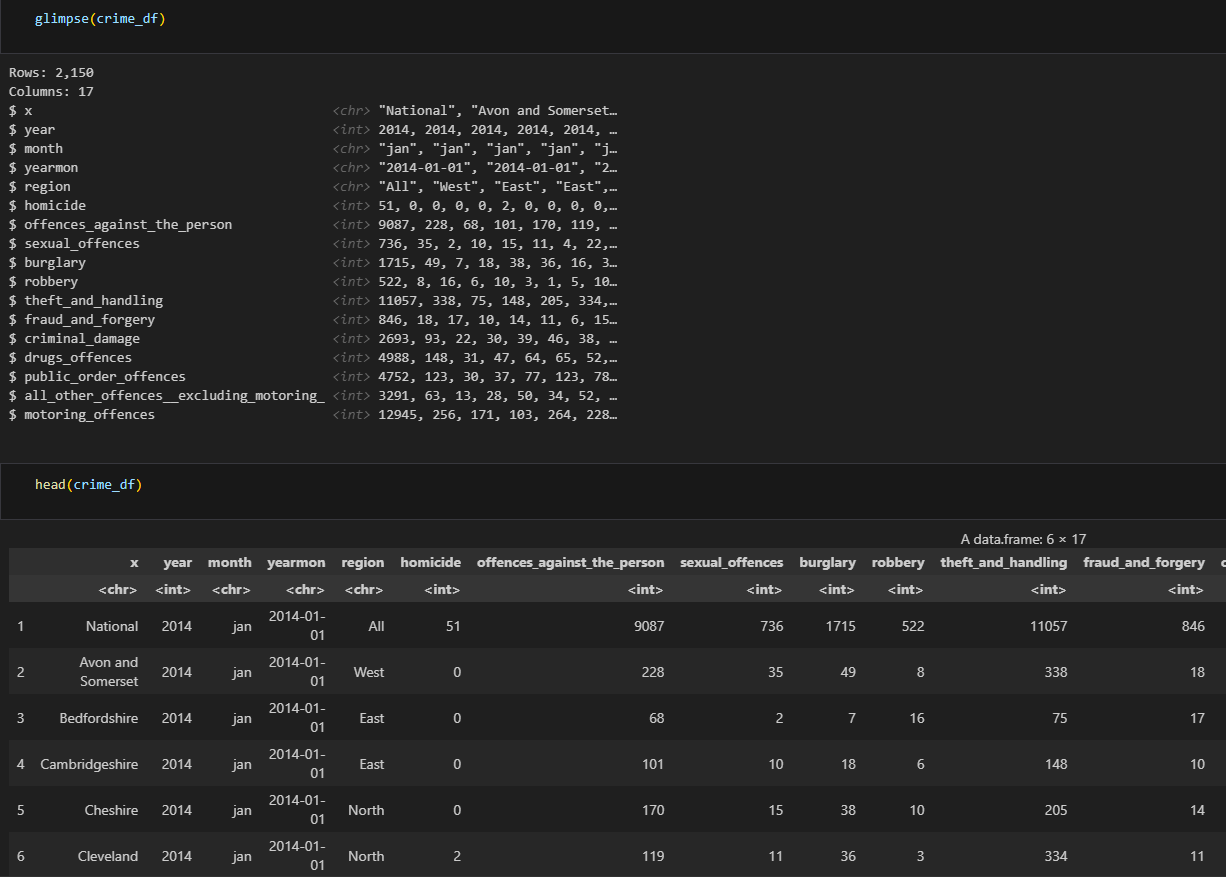
5. Attribute Importance: Determining attribute importance is crucial for feature selection and engineering. Various techniques such as information gain, chi-square tests, or statistical models like regression or decision trees can be used to assess the relevance and predictive power of attributes for the analysis objectives.

6. Visualization: Visualizing attribute relationships and distributions is an important aspect of attribute analysis. It helps identify trends, patterns, and potential insights that may not be apparent through numerical analysis alone. Visualizations such as scatter plots, bar charts, heatmaps, or parallel coordinate plots aid in understanding attribute characteristics.

Attribute analysis provides a comprehensive understanding of the dataset's attributes, enabling data scientists, analysts, and researchers to make informed decisions regarding data pre-processing, feature selection, model development, and further analysis. It helps identify influential variables, assess data quality, identify outliers, and reveal potential relationships or patterns that may contribute to achieving the analysis objectives.

**6.1.1 Crime dataframe**





The county column has all possible county values from England. The Year column has 5 values because we are using a dataset from 5 years which are 2014, 2015, 2016, 2017, 2018. The month column containsmonths ranging from Jnauary to December.

The yearmon column contains combined value of year and months while the region column contain all 5 possible values which are All, East, West, North and South.

The Homicide column shows that the minimum value of the variable is 0. 25% of the values are below 0, the median value is 1, the mean value is 3.798. 75% of the values are below 3 and the maximum value is 131. It also indicates that the variable has some outliers as the max value is quite high as compared to the mean and median.

For offences\_against\_the\_person, the variable exhibits a range of statistical measures. The minimum value is recorded as 29, signifying the lowest data point. Additionally, 25% of the values fall below 115, indicating a lower quartile boundary. The median value stands at 179, representing the middle value when the data is sorted in ascending order. The mean value is calculated as 454.9, providing an average measure of the variable. Furthermore, 75% of the values are below 272, marking the upper quartile boundary. Notably, the maximum value reaches 11741, which is considerably higher than both the mean and median values, suggesting the presence of outliers within the dataset.

For sexual\_offences, the variable under consideration showcases several key statistical measures. The minimum value is recorded as 0, denoting the lowest data point. Additionally, 25% of the values lie below 8, indicating the lower quartile boundary. The median value stands at 15, representing the middle value when the data is arranged in ascending order. The mean value is calculated as 43.78, providing an average measure of the variable. Furthermore, 75% of the values fall below 29, marking the upper quartile boundary. Notably, the maximum value reaches 1179, which is significantly higher than both the mean and median values, suggesting the presence of outliers within the dataset.

For burglary, the variable in question demonstrates various statistical characteristics. The minimum value is observed as 1, indicating the lowest data point. Additionally, 25% of the values fall below 14, representing the lower quartile boundary. The median value is determined as 23, signifying the middle value when the data is arranged in ascending order. The mean value is calculated as 60.09, providing an average representation of the variable. Furthermore, 75% of the values lie below 38, denoting the upper quartile boundary. Notably, the maximum value is recorded as 1715, which is notably higher than both the mean and median values, suggesting the existence of outliers within the dataset.

For robbery, the variable reflects a range of statistical measures. The minimum value is recorded as 0, representing the lowest data point. Additionally, 25% of the values fall below 2, indicating the lower quartile boundary. The median value is determined to be 5, representing the middle value when the data is arranged in ascending order. The mean value is calculated as 19.33, providing an average representation of the variable. Furthermore, 75% of the values lie below 10, denoting the upper quartile boundary. Notably, the maximum value reaches 650, which is significantly higher than both the mean and median values, indicating the presence of outliers within the dataset.

For theft\_and\_handling, the variable showcases several statistical measures. The minimum value is recorded as 13, representing the lowest data point. Additionally, 25% of the values fall below 95, indicating the lower quartile boundary. The median value is determined to be 147, representing the middle value when the data is arranged in ascending order. The mean value is calculated as 373.1, providing an average representation of the variable. Furthermore, 75% of the values lie below 237, denoting the upper quartile boundary. Notably, the maximum value reaches 11057, which is significantly higher than both the mean and median values, suggesting the presence of outliers within the dataset.

Fraud\_and\_forgery, the variable illustrates various statistical measures. The minimum value is recorded as 0, signifying the lowest data point. Additionally, 25% of the values fall below 8, representing the lower quartile boundary. The median value is determined as 13, representing the middle value when the data is arranged in ascending order. The mean value is calculated as 38.57, providing an average representation of the variable. Furthermore, 75% of the values lie below 21, denoting the upper quartile boundary. Notably, the maximum value reaches 1075, which is significantly higher than both the mean and median values, indicating the presence of outliers within the dataset.

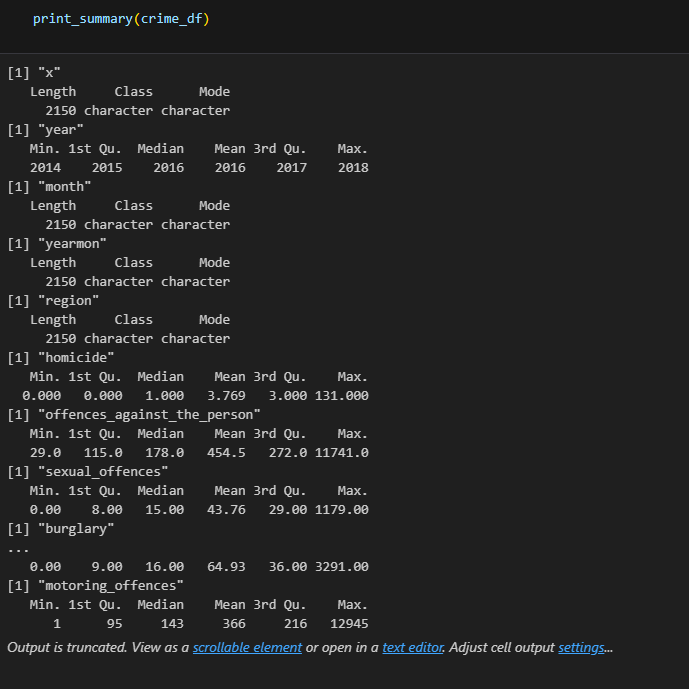
Criminal\_damage, the data indicates that the variable has a minimum value of 3, with 25% of the values falling below 25. The median value is recorded as 40, representing the middle value when the data is arranged in ascending order. The mean value is calculated to be 95.82, providing an average representation of the variable. Furthermore, 75% of the values are below 59, indicating the upper quartile boundary. Notably, the maximum value is 2693, which is considerably higher than both the mean and median values. This suggests the presence of outliers in the dataset, as the maximum value deviates significantly from the central tendency represented by the mean and median.

Drugs\_offences, The data indicates that the variable has a minimum value of 4, with 25% of the values falling below 38. The median value is recorded as 63, representing the middle value when the data is arranged in ascending order. The mean value is calculated to be 186.6, providing an average representation of the variable. Furthermore, 75% of the values are below 100, indicating the upper quartile boundary. Notably, the maximum value is 4988, which is significantly higher than both the mean and median values. This suggests the presence of outliers in the dataset, as the maximum value deviates significantly from the central tendency represented by the mean and median.

Public\_order\_offenced, the data reveals that the variable has a minimum value of 2, with 25% of the values falling below 39. The median value is recorded as 63, representing the middle value when the data is arranged in ascending order. The mean value is calculated to be 162.4, providing an average representation of the variable. Furthermore, 75% of the values are below 100, indicating the upper quartile boundary. Notably, the maximum value is 4752, which is significantly higher than both the mean and median values. This suggests the presence of outliers in the dataset, as the maximum value deviates considerably from the central tendency represented by the mean and median.

All\_other\_offences\_excluding\_monitorring, he provided information illustrates the characteristics of a variable. The minimum value of the variable is recorded as 0, with 25% of the values falling below 9. The median value is determined to be 16, representing the middle value when the data is arranged in ascending order. The mean value is calculated as 64.34, indicating the average value of the variable. Additionally, 75% of the values lie below 35, denoting the upper quartile boundary. Notably, the maximum value reaches 3291, which is significantly higher than both the mean and median values. This suggests the presence of outliers within the dataset, as the maximum value deviates notably from the central tendency represented by the mean and median.

Motoring\_offences, this provided information highlights the characteristics of a variable. The variable's minimum value is reported as 1, with 25% of the values falling below 95. The median value is determined to be 143, representing the middle value when the data is sorted in ascending order. The mean value is calculated as 365.5, indicating the average value of the variable. Furthermore, 75% of the values lie below 216, marking the upper quartile boundary. Notably, the maximum value is recorded as 12945, which is significantly higher than both the mean and median values. This suggests the presence of outliers within the dataset, as the maximum value deviates significantly from the central tendency represented by the mean and median.



**6.2. Regional Analysis**

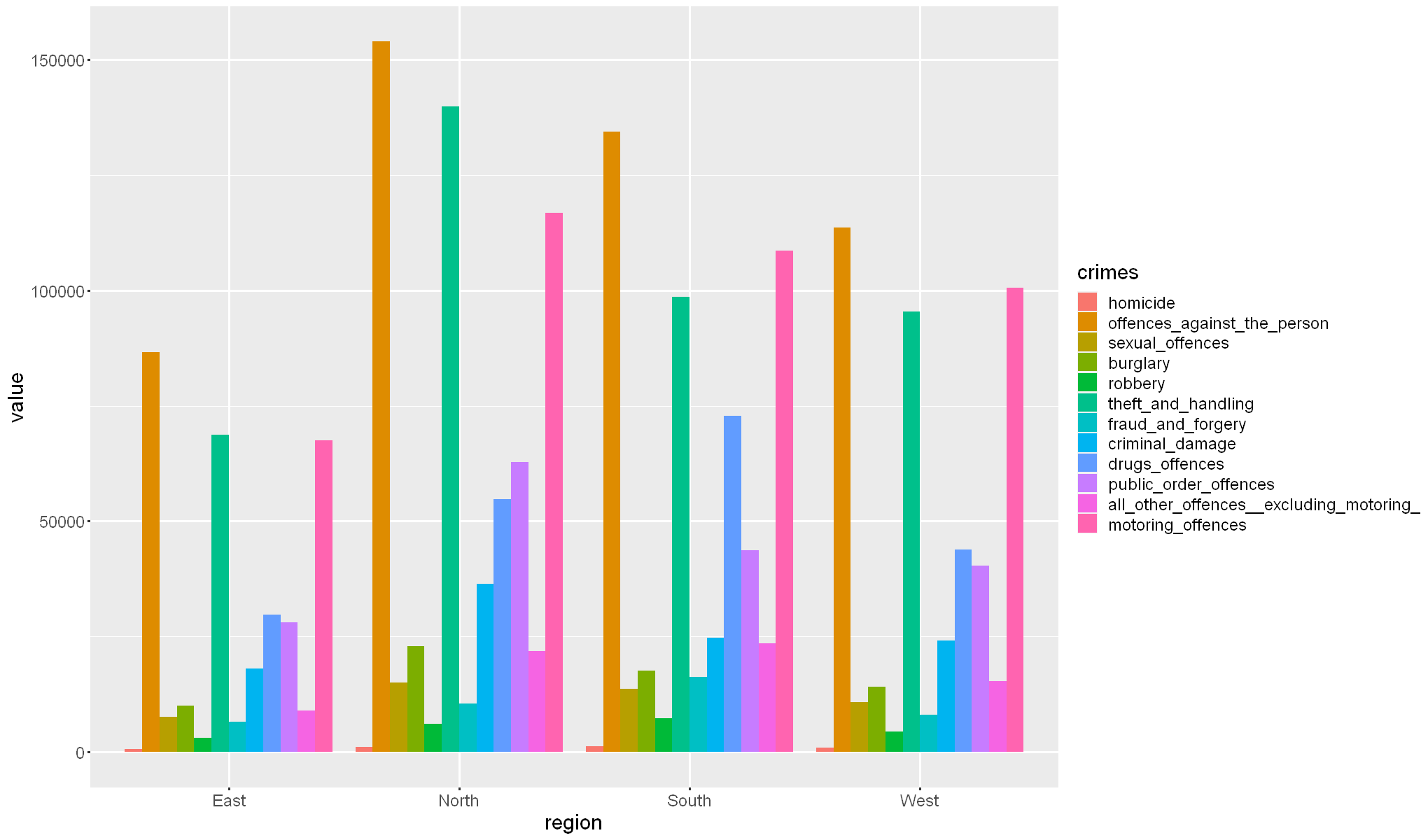
In this analytical section, our objective is to examine the relationships between crime occurrences at different times and their corresponding regions in England. We employ graph visualization and statistical analysis techniques to gain insights into these associations. By visualizing the data graphically and conducting statistical examinations, we aim to comprehend the patterns and connections between crime occurrences and specific time periods across various regions in England.

**6.2.1. Region and all types of crime**

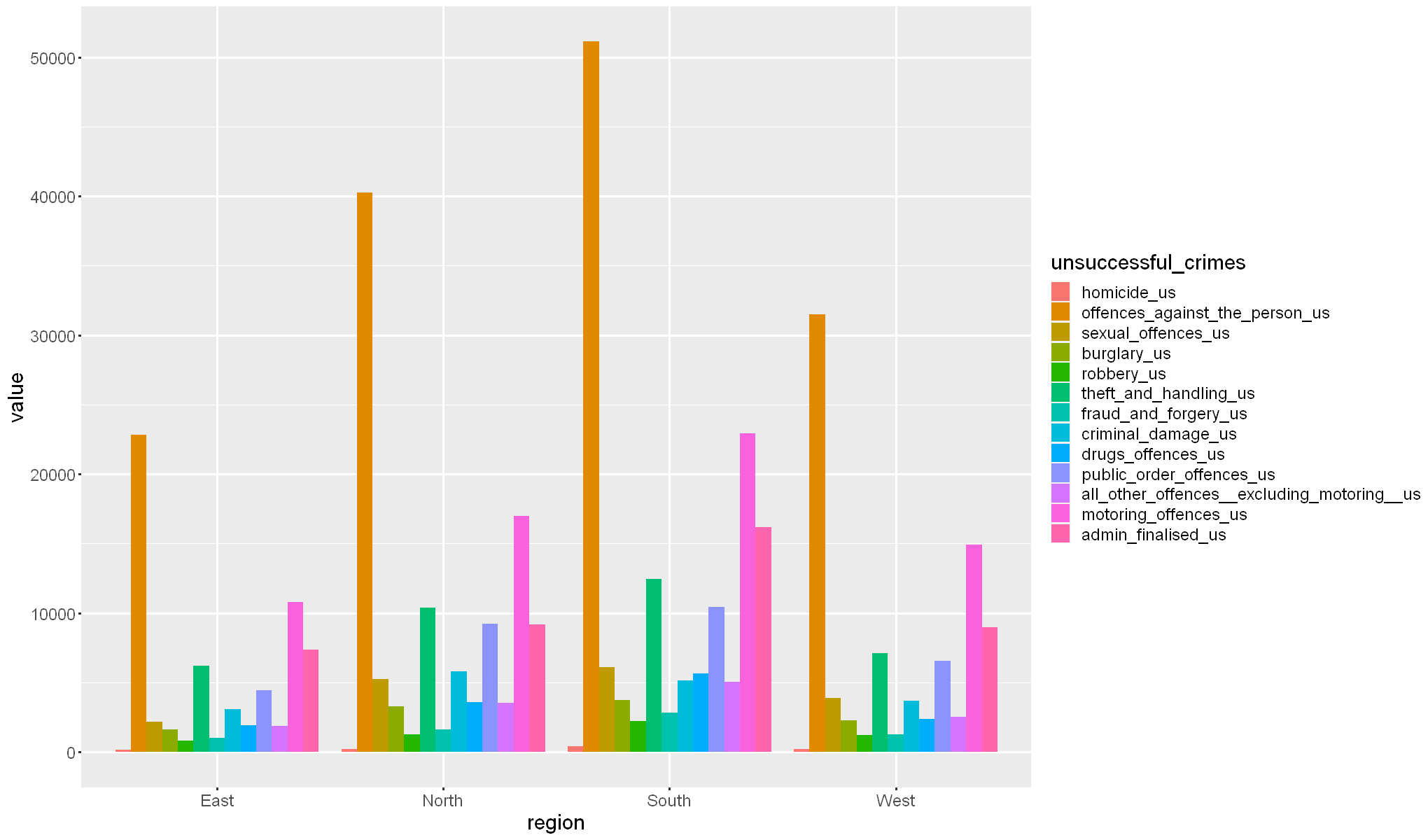
In our analysis, we consider two datasets: the crimes dataset and the unsuccessful crimes dataset. We proceed by visualizing the occurrence of these crimes, grouped by crime types, across different regions of England. By examining the data in this manner, we aim to gain insights into the distribution and patterns of both successful and unsuccessful crimes, allowing us to understand the relationship between crime types and their prevalence in specific regions of England.

**6.2.1.1. Crimes**

The graph provided examines crime cases and reveals that offenses against individuals are the most prevalent across all regions of England, particularly in the North. Robbery ranks as the second-highest crime category and maintains this trend consistently across all regions, with the North exhibiting the highest incidence. Motoring offenses follow as the third highest category in all regions, with the remaining crime types displaying a similar pattern. Overall, the North region exhibits the highest peaks in terms of crime rates, while the East region can be observed to have the lowest incidence.

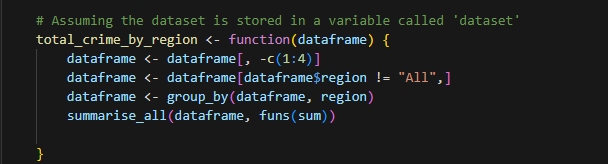


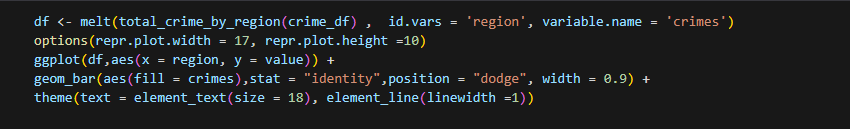
**6.2.1.2. Unsuccessful crime**

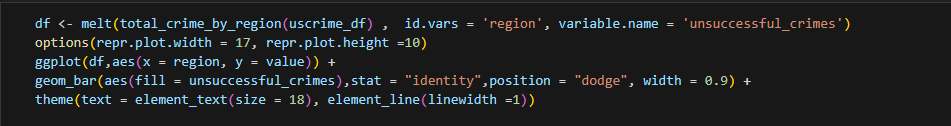


The graph provided examines unsuccessful cases and reveals that unsuccessful offenses against individuals are the most prevalent across all regions of England, particularly in the South. Unsuccessful Motoring Offenses rank as the second-highest category and maintain this trend consistently across all regions, with the South region exhibiting the highest incidence. Unsuccessful Admin Finalized cases follow as the third highest category in all regions, with the remaining unsuccessful crimes displaying a similar pattern. Generally, the South region has the highest peaks of unsuccessful cases, while the East region can be observed to have the lowest incidence.

Below are the codes used for both crime and unsuccessful crime.



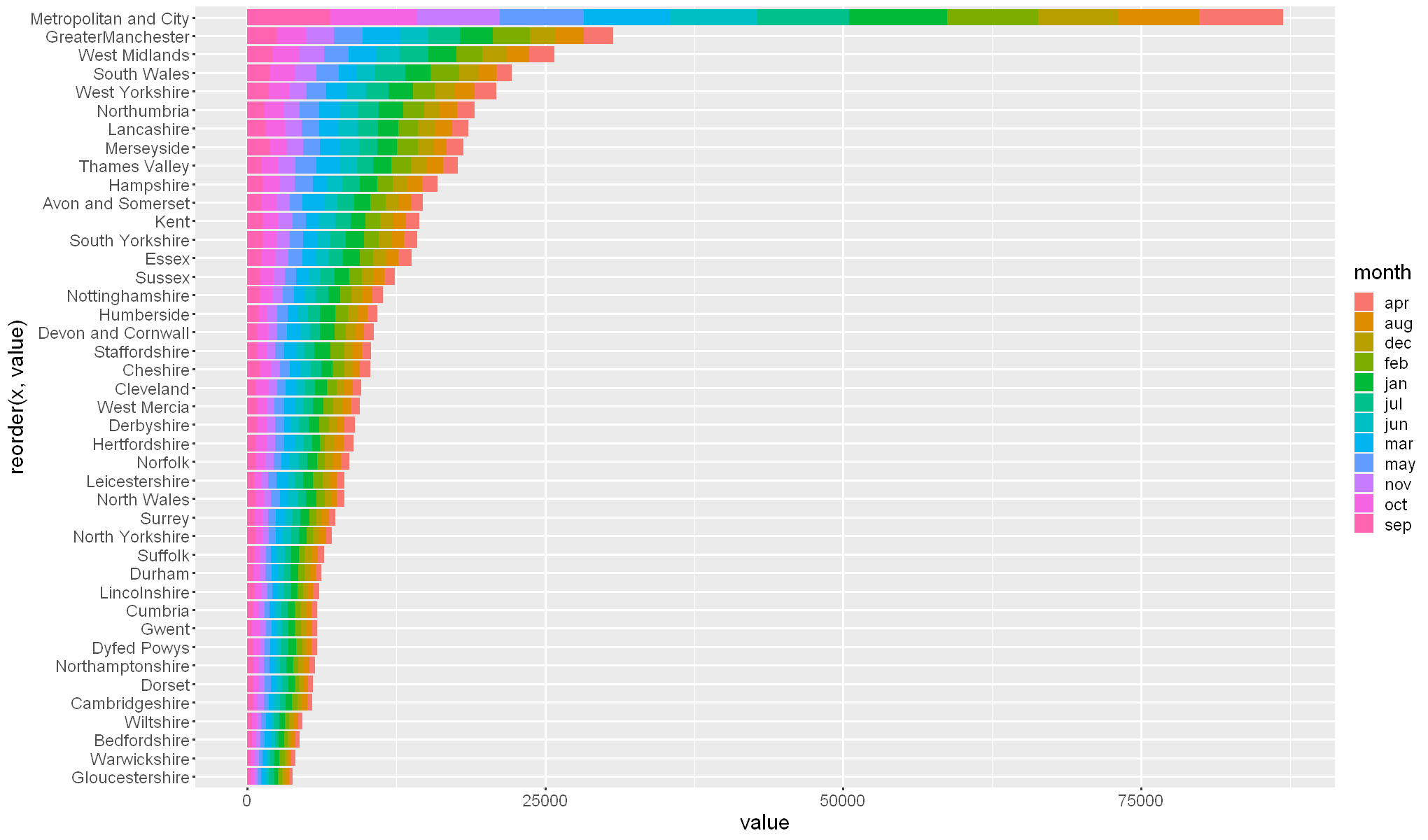




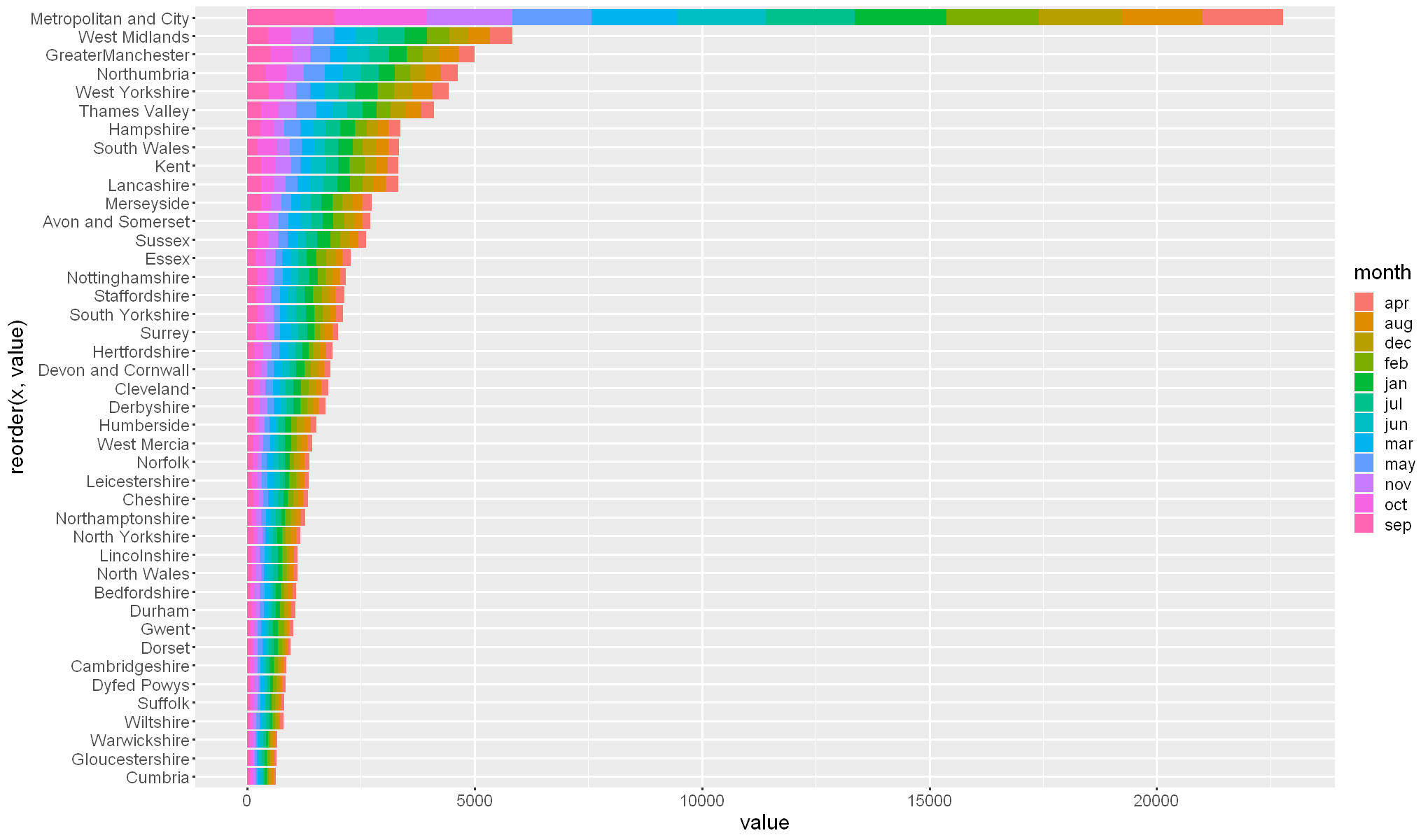
**6.3. Analysis dependent on the variables of years and months**

**In this analytical section, focus is to examine the relationship between successful and unsuccessful crime occurrences in different counties of England over the course of years and months. I will utilize graph visualization techniques to gain insights into these associations. By visualizing the data graphically, we aim to understand the patterns and connections between successful and unsuccessful crime incidents in various counties of England, specifically considering the temporal factors of years and months.**

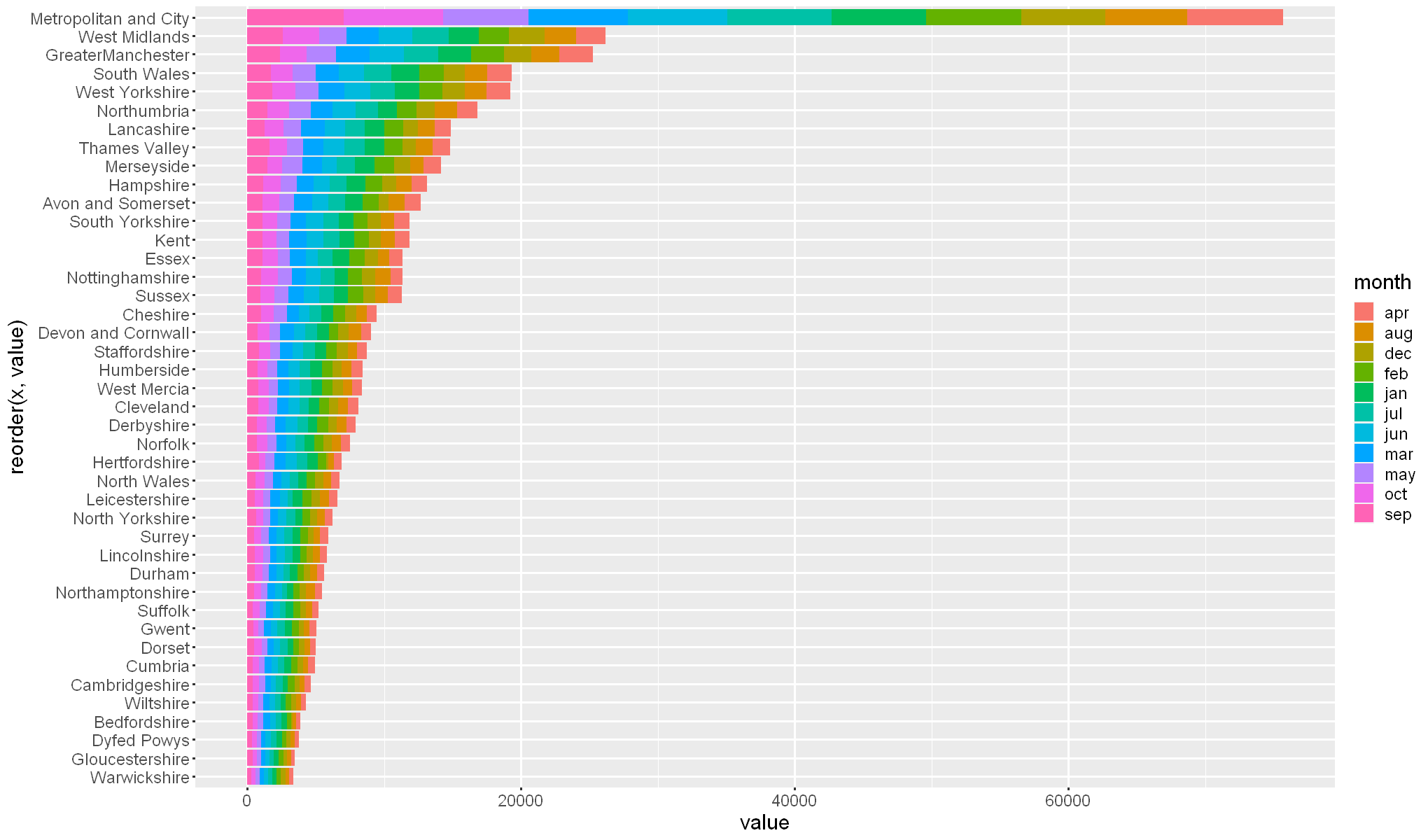
**6.3.1. 2014 Crimes**



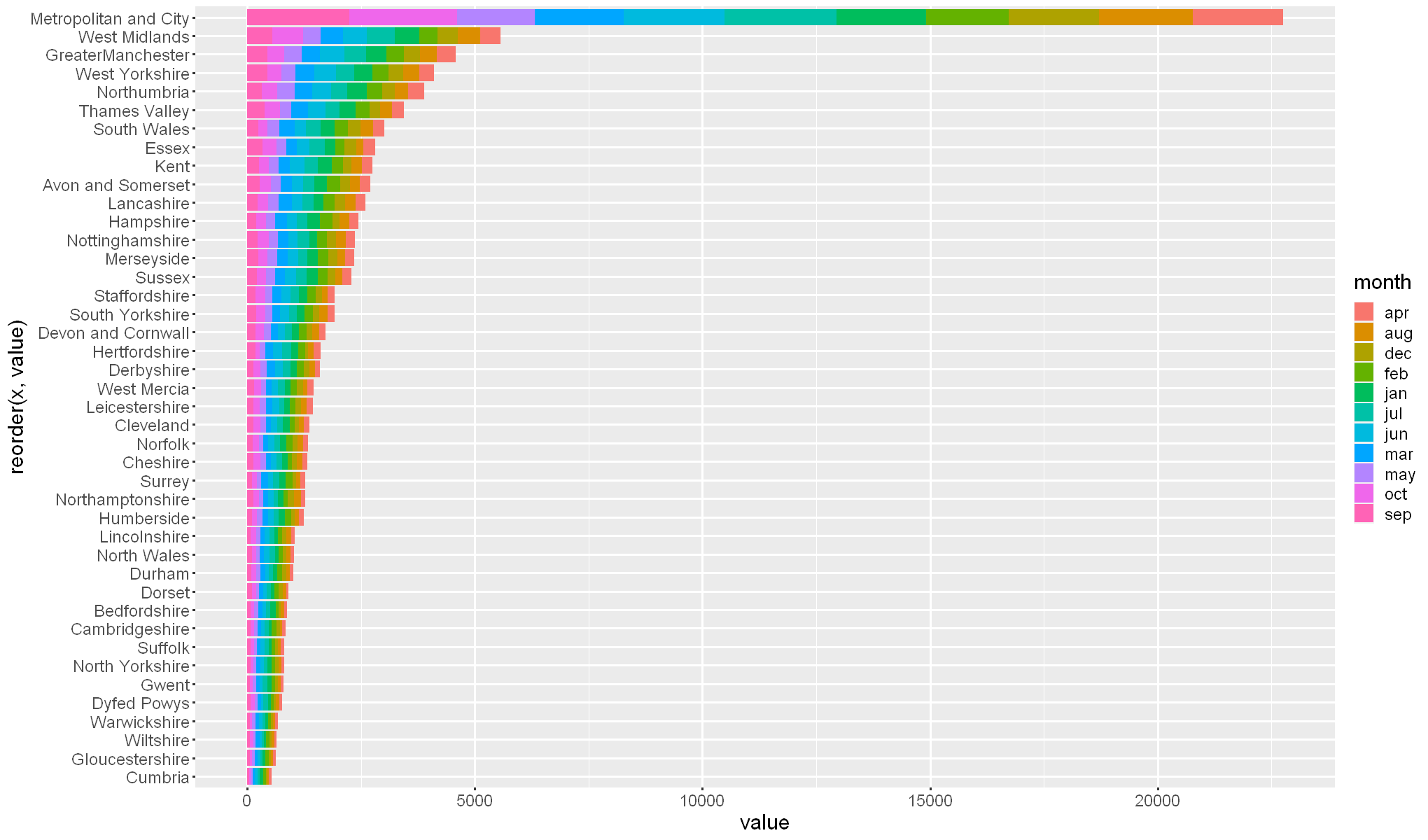
**6.3.1.1. 2014 Unsuccessful crimes**



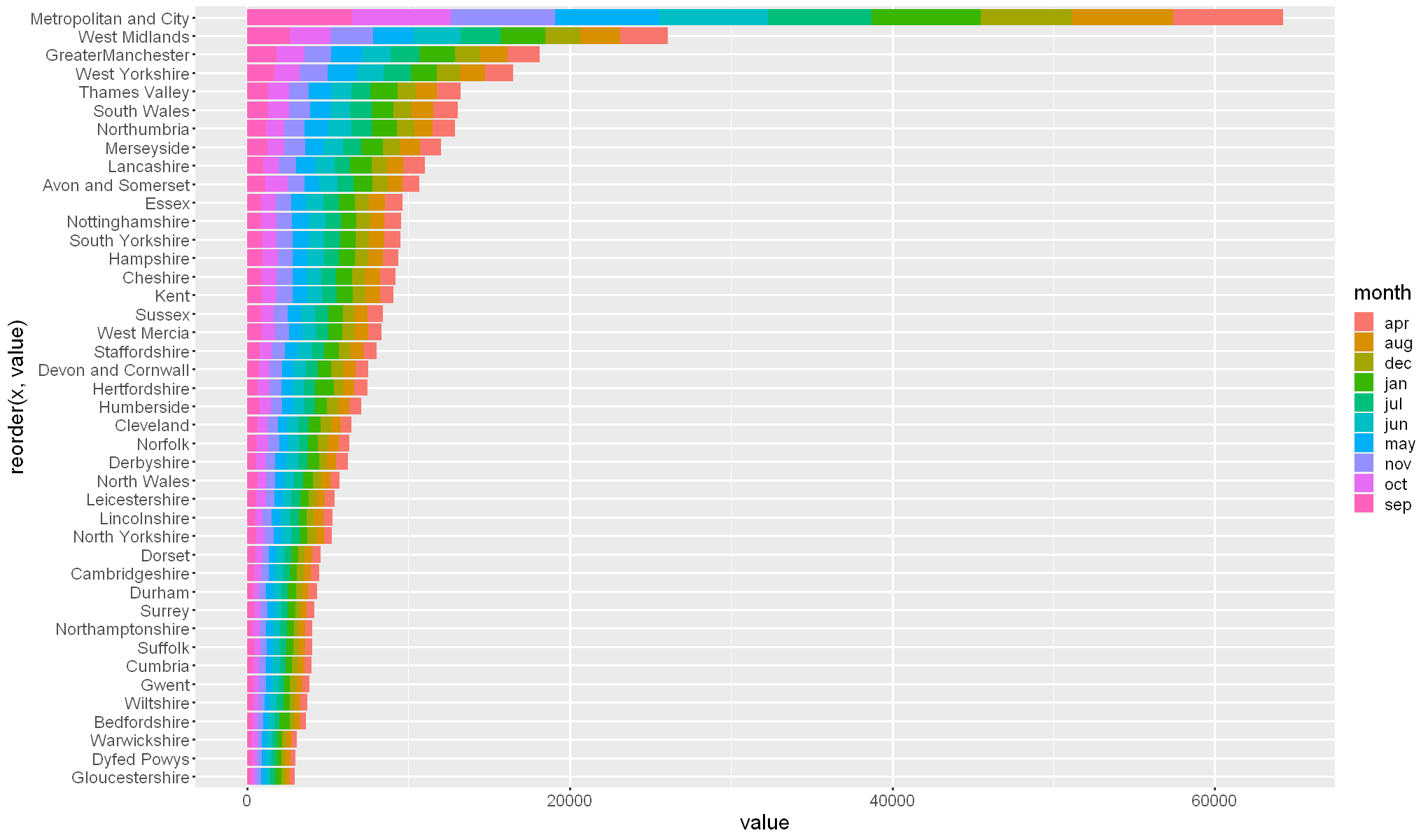
6.3.2. 2015 crimes



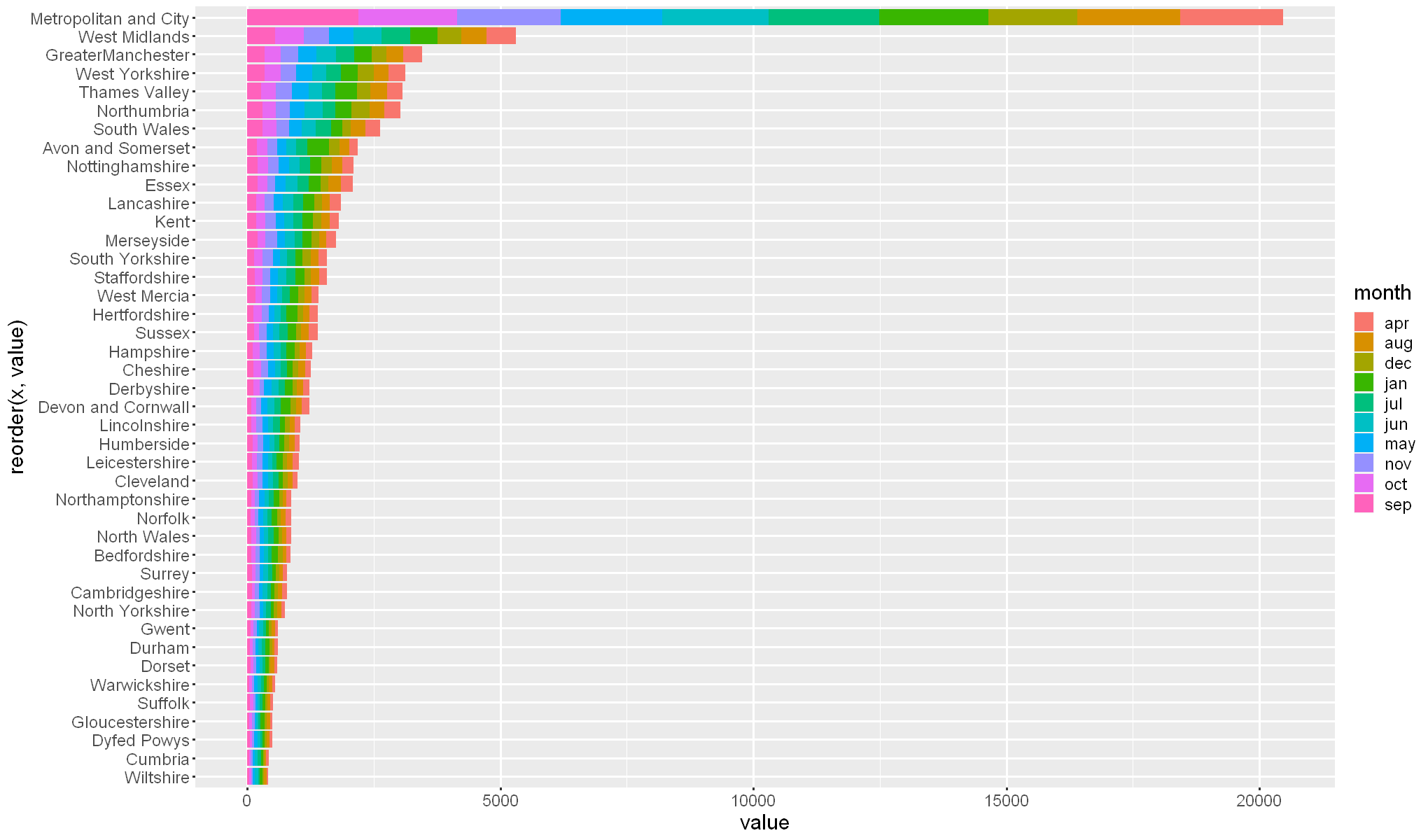
6.3.2.1. 2015 Unsuccessful crime



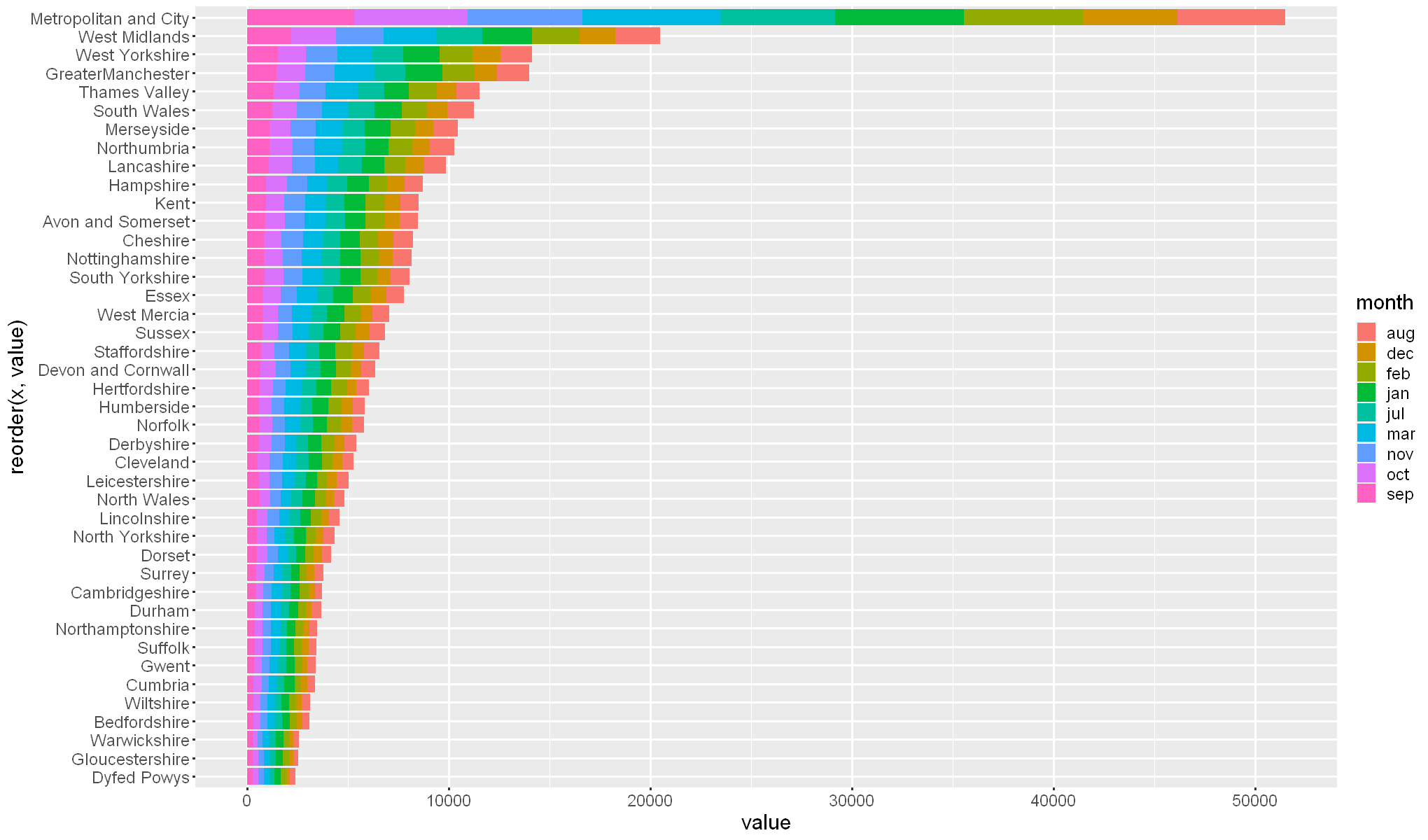
6.3.3. 2016 crime



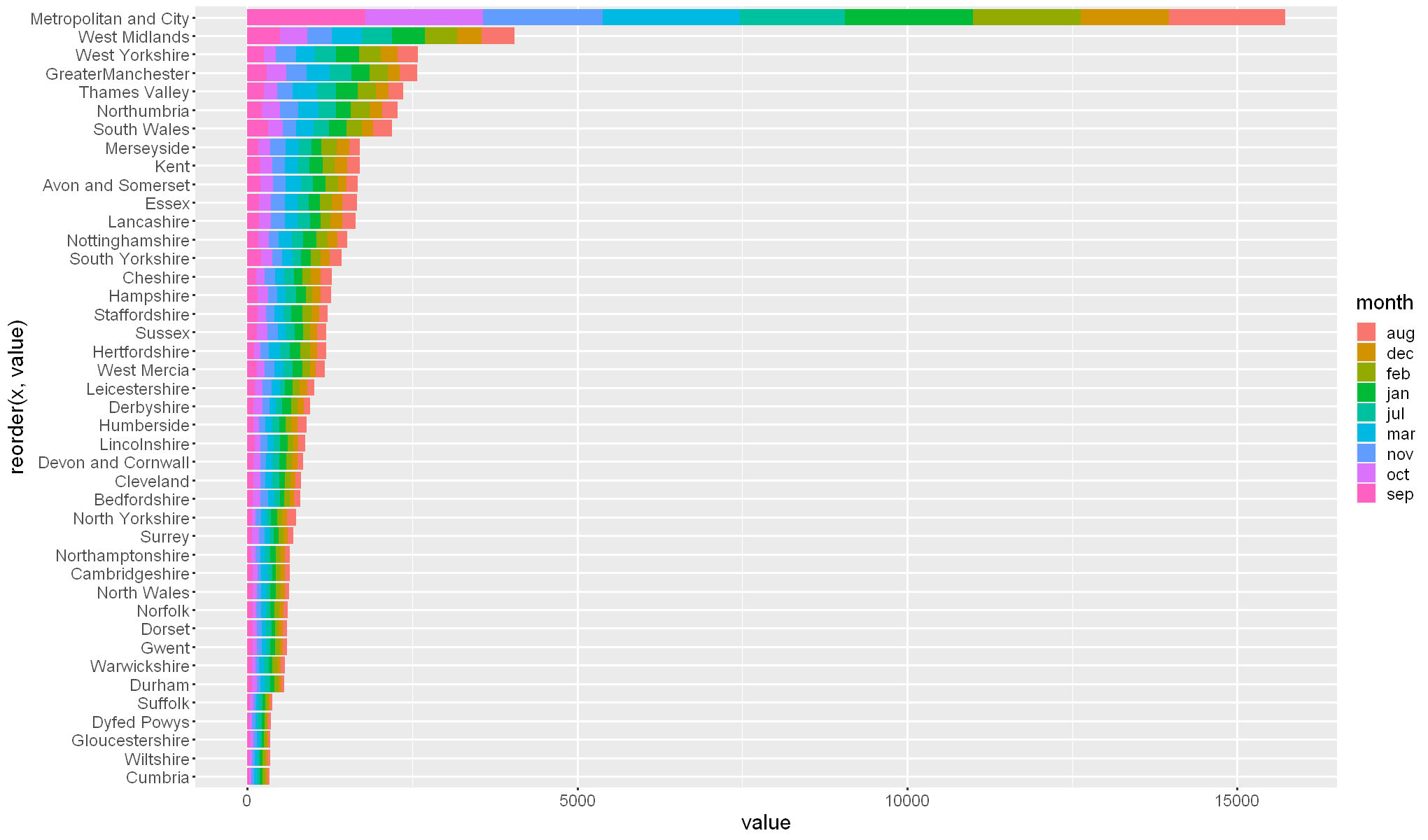
6.3.3.1. 2016 unsuccessful crime



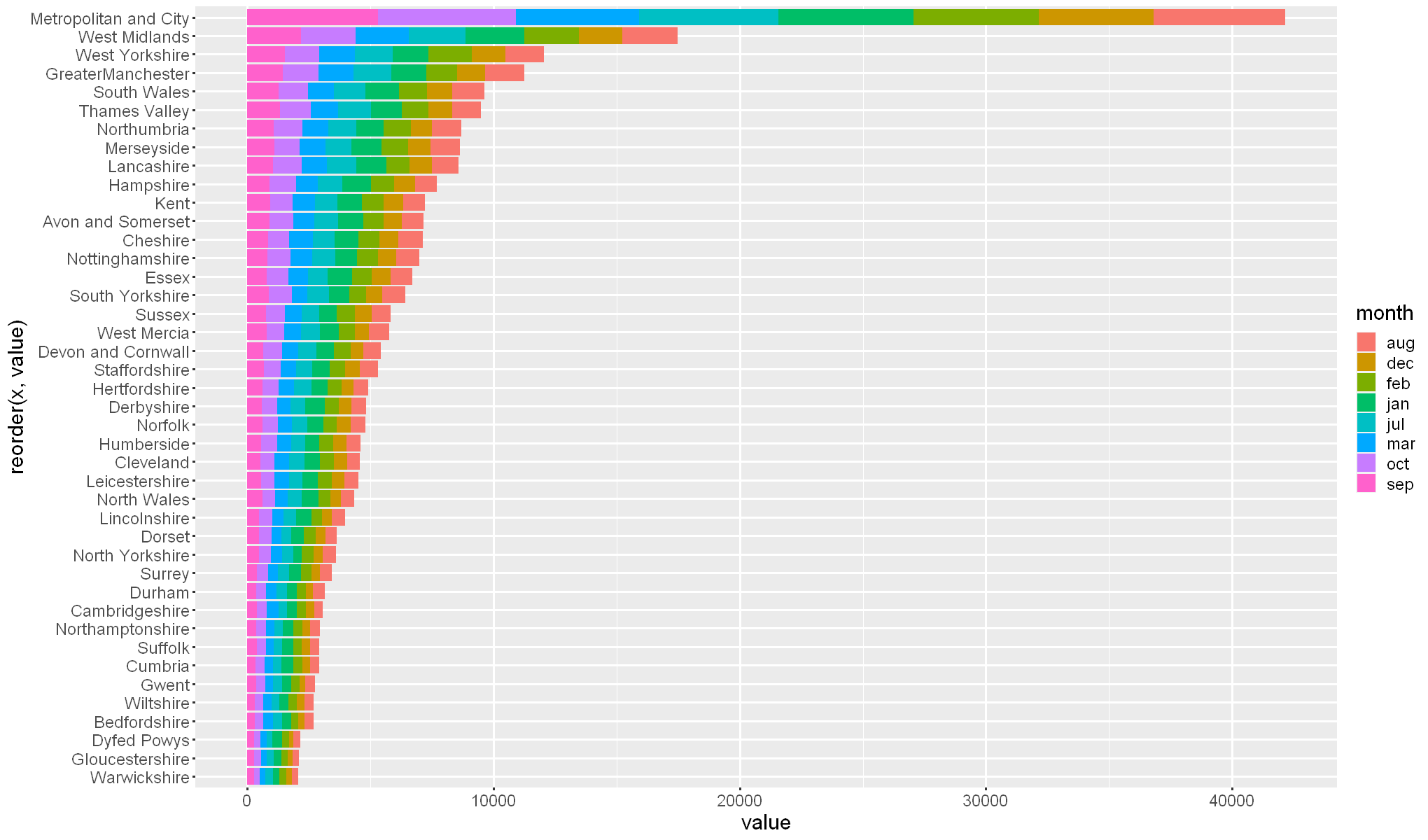
6.3.4. 2017 crime



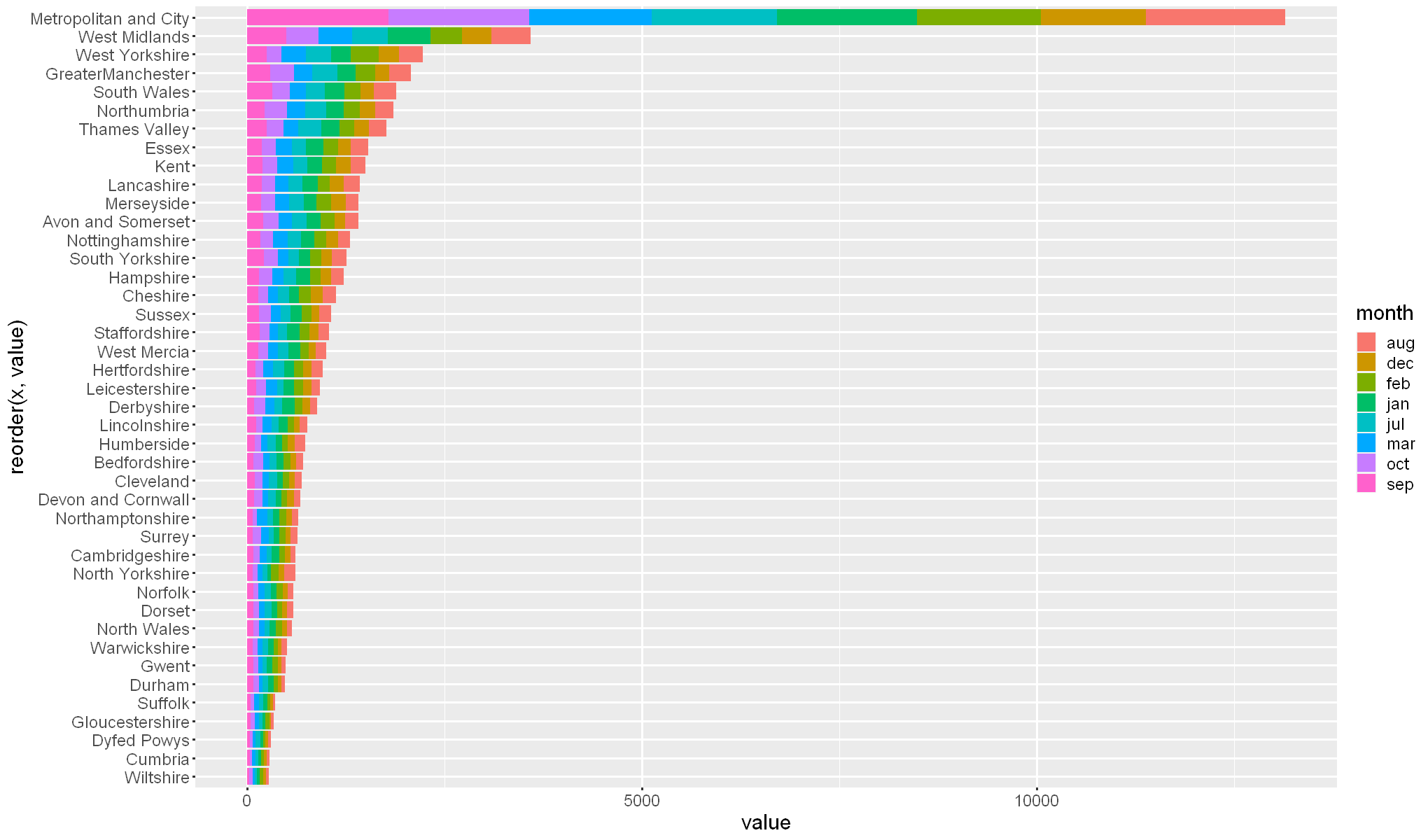
6.3.4.1. 2017 unsuccessful crime

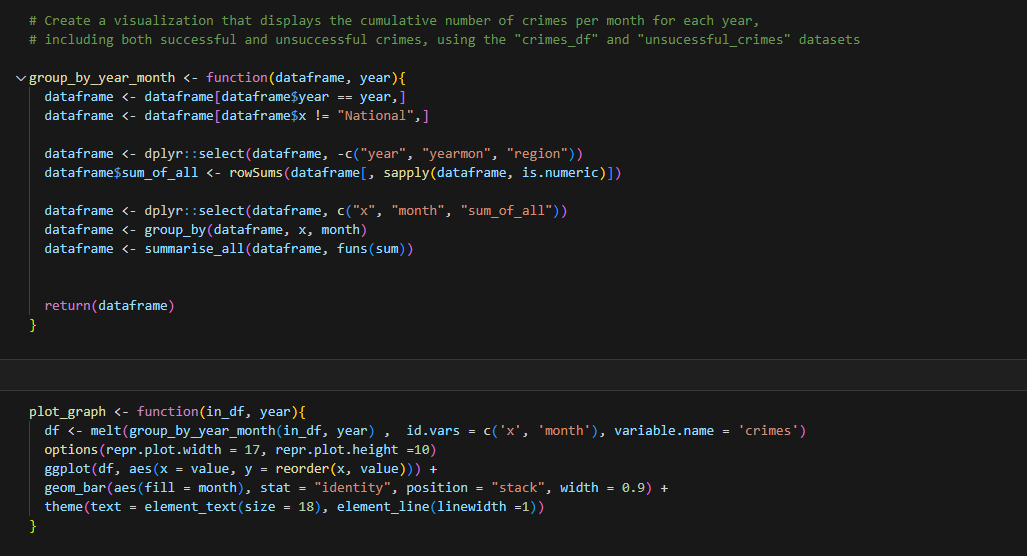


6.3.5. 2018 crime



6.3.5.1. 2018 unsuccessful crime





**6.3.6. Group by year and month**

This function accepts a dataframe and a year as inputs. It performs the following operations on the dataframe:

* It groups the data by county and month.
* It filters the dataframe to include only rows where the year column matches the input year.
* It removes any rows where the county column is labeled as "National".
* It eliminates the year, yearmon, and region columns from the dataframe.
* It creates a new column named "sum\_of\_all" that contains the sum of all numeric columns.
* It retains only the "county", "month", and "sum\_of\_all" columns.
* It groups the data by the "county" and "month" columns.
* Finally, it applies the sum function to all columns in the dataframe and returns the summarized dataframe.

**6.3.7. Plot graph**

This function takes a dataframe and year as inputs and utilizes the ggplot2 library to create a stacked bar chart. The steps involved are as follows:

* It applies the function "group\_by\_year\_month" to the input dataframe and year, resulting in a new dataframe grouped by county and month.
* The "melt" function is then used to reshape the dataframe, consolidating each county and month into a single row with corresponding columns for the types of crimes and their respective values.
* Using the ggplot2 library, the function proceeds to plot the data. The 'value' column is assigned to the x-axis, the 'county' column is assigned to the y-axis (reordered by 'value'), and the 'month' column is used for the fill aesthetic.
* Additionally, the function adjusts the plot size, text size, and line width from their default settings to enhance the visual presentation.

**6.4. Correlation**

In this analytical section, the aim is to explore and comprehend the correlation between successful and unsuccessful crime incidents. The goal is to seek to investigate the relationship and statistical measures of association between these two variables. By conducting this analysis, I can gain insights into the extent and nature of the connection between successful and unsuccessful crime occurrences.

**6.4.1. About correlation**

Correlation is a statistical measure that quantifies the strength and direction of the relationship between two variables. It assesses how closely the values of one variable are associated with the values of another variable. Correlation is typically represented by a correlation coefficient, which ranges from -1 to 1.

A positive correlation indicates that as one variable increases, the other variable also tends to increase. In contrast, a negative correlation implies that as one variable increases, the other variable tends to decrease. A correlation coefficient of 0 suggests no linear relationship between the variables.

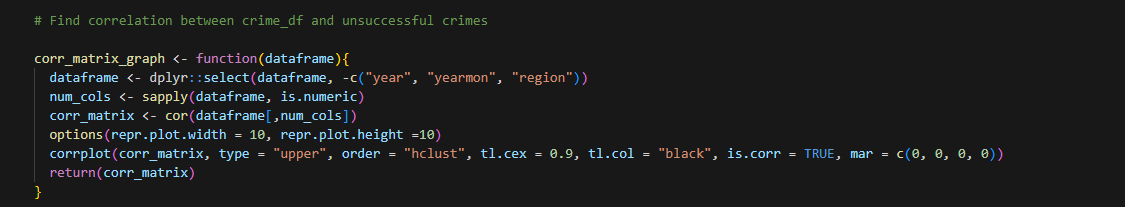
Correlation is a valuable tool in data analysis as it helps identify patterns, dependencies, and connections between variables. It provides insights into how changes in one variable may impact another variable. However, it is important to note that correlation does not imply causation, as there may be other factors influencing the relationship between variables.

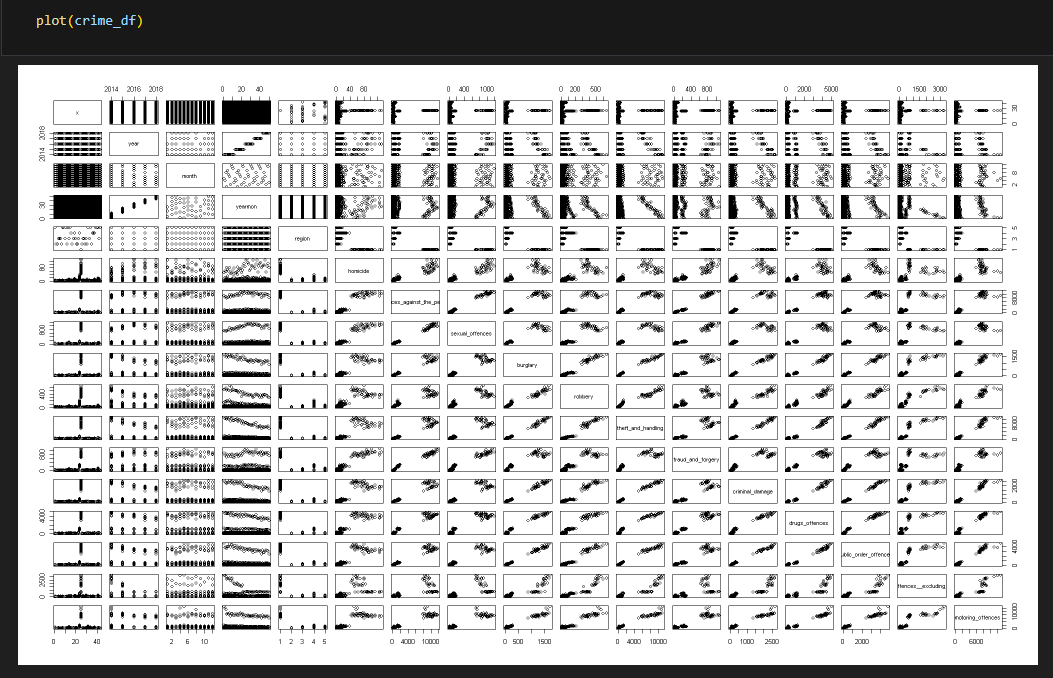
For example, in the context of crime, a positive correlation between the number of burglaries and the number of thefts would indicate that an increase in burglaries is associated with an increase in thefts, and vice versa. A negative correlation between the number of burglaries and the number of drug offenses, on the other hand, would indicate that an increase in burglaries is associated with a decrease in drug offenses, and vice versa.

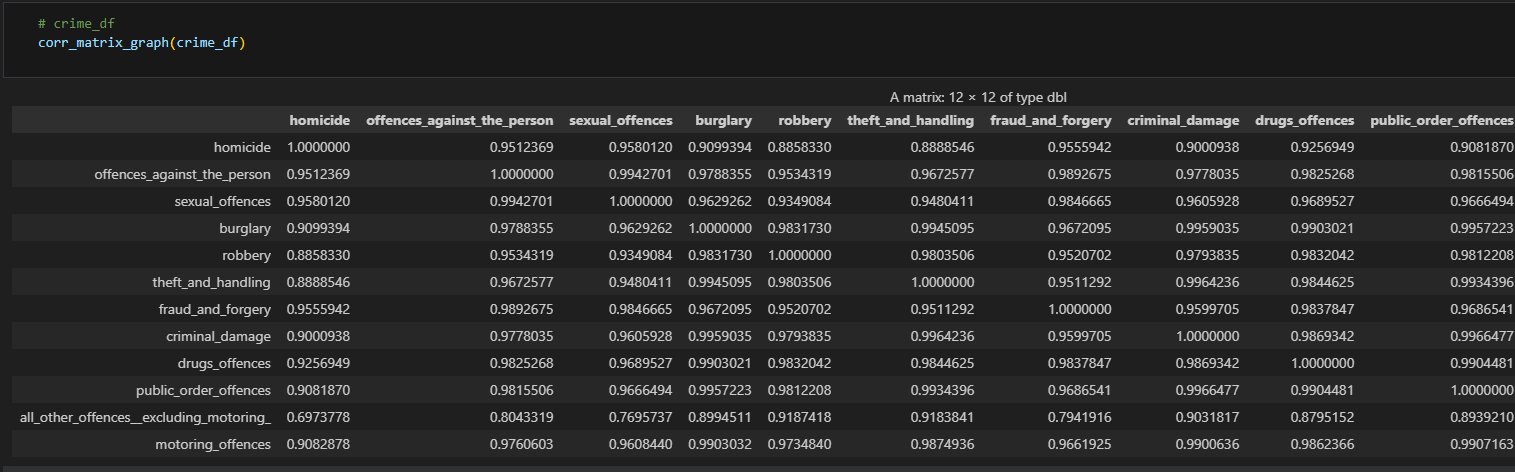
Correlation is typically represented as a numerical value, such as a Pearson correlation coefficient, and can be used in conjunction with other statistical measures, such as regression analysis, to better understand the relationship between variables.

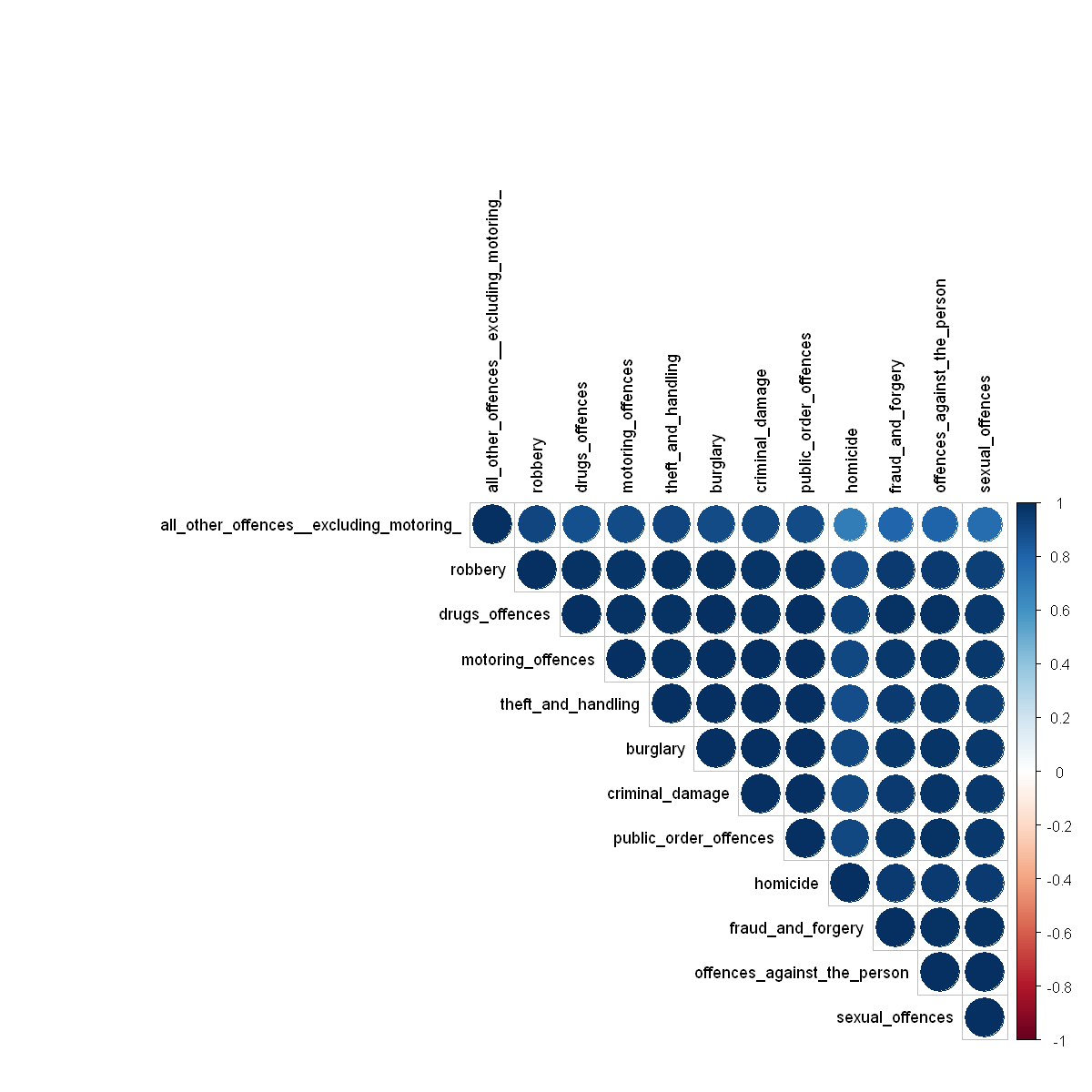
It's worth mentioning that correlation does not indicate causality, it just indicates that two variables are related and in which direction the relationship is.

**6.4.1.1. Crimes**









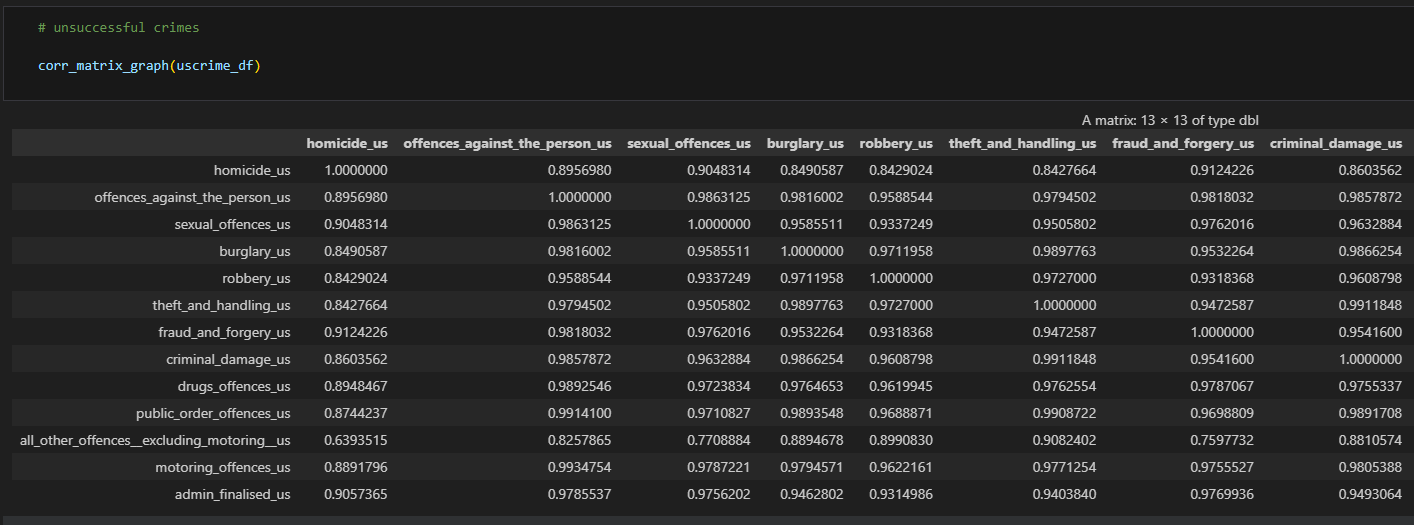
This correlation matrix illustrates the correlation coefficients between different categories of crimes. Each cell within the matrix represents the correlation coefficient between two specific types of crimes, corresponding to the row and column of that cell. The correlation coefficient ranges from -1 to 1, where 1 indicates a perfect positive correlation, -1 indicates a perfect negative correlation, and 0 indicates no correlation.

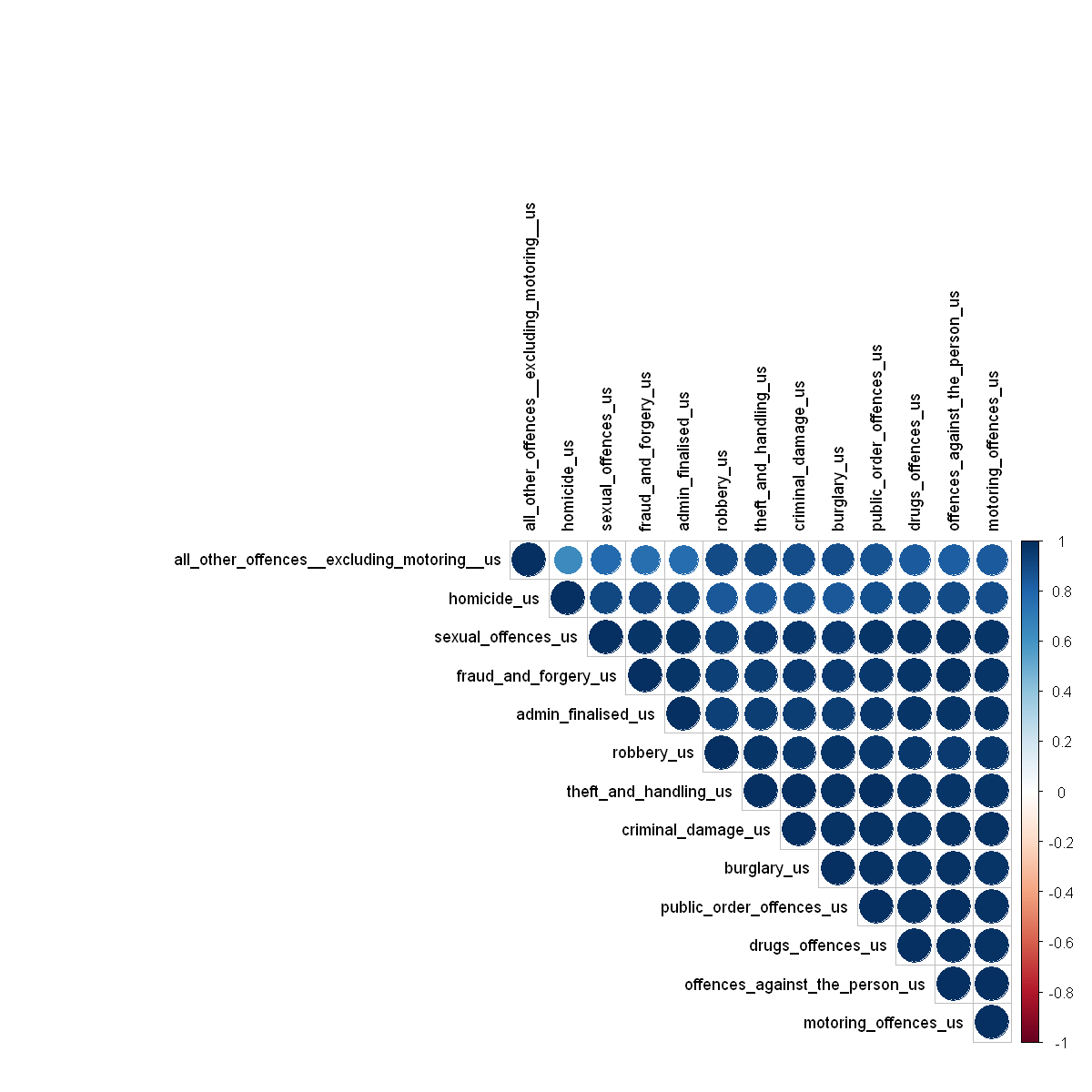
The correlation matrix serves to examine the relationships among various types of crimes. For instance, it reveals that homicide exhibits a strong correlation with offences against the person (0.95), sexual offences (0.96), burglary (0.91), and robbery (0.89). This indicates a tendency for these crimes to occur together.

Conversely, the lowest correlations are observed between motoring offences and other crimes. For instance, the correlation between motoring offences and homicide is 0.91, between motoring offences and offences against the person is 0.98, and between motoring offences and sexual offences is 0.96. These values are lower compared to the correlations observed among other types of crimes. For instance, homicide, offences against the person, and sexual offences exhibit correlations of 0.95, 0.99, and 0.99, respectively.

These findings suggest that motoring offences are not as strongly associated with other types of crimes as those crimes are with each other. This could indicate that individuals who commit motoring offences differ from those who engage in other types of crimes, and thus distinct intervention and prevention strategies may be required to address motoring offences.

**6.4.1.2. Unsuccessful crimes**



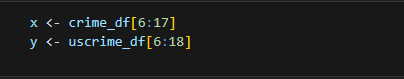


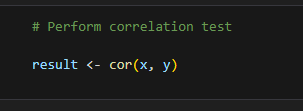
The provided correlation matrix displays the correlation between various types of unsuccessful crimes. The values within the matrix range from -1 to 1, where 1 signifies a perfect positive correlation, -1 denotes a perfect negative correlation, and 0 indicates no correlation. The correlation coefficient measures the strength and direction of the linear relationship between two variables.

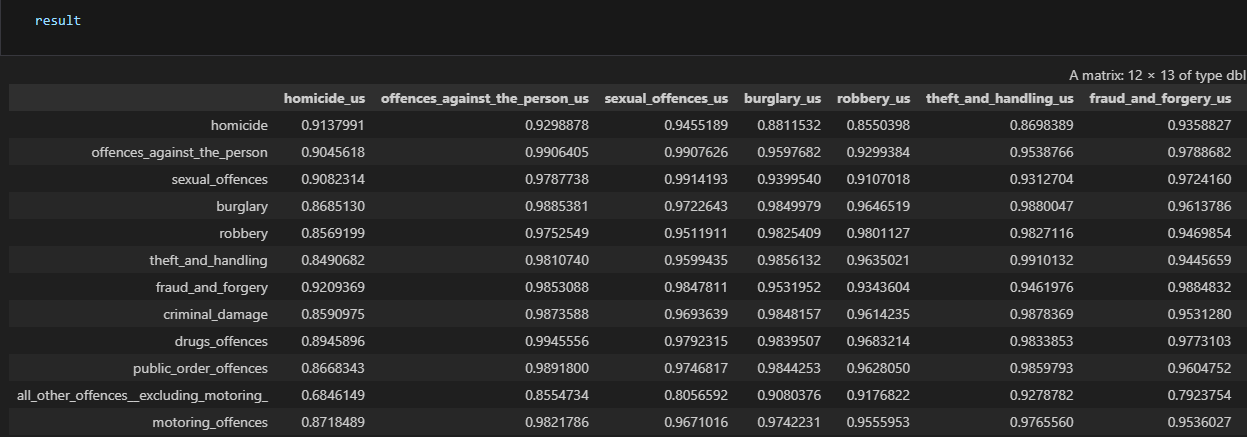
Upon analyzing the correlation matrix, we can observe that certain unsuccessful crimes exhibit higher correlations compared to others. For instance, unsuccessful homicide demonstrates a strong positive correlation with unsuccessful offences against the person (0.89), unsuccessful sexual offences (0.90), and unsuccessful burglary (0.85). These crimes are likely to occur together or share some form of association. Conversely, crimes such as unsuccessful homicide and unsuccessful motoring offences have a lower correlation (0.91), implying that they are less prone to occurring together.

Furthermore, the matrix reveals that certain unsuccessful crimes exhibit low correlations with other types of unsuccessful crimes. For instance, there is a correlation of 0.91 between unsuccessful homicide and unsuccessful motoring offences, which is lower than the correlations observed among other crimes. This suggests that these particular unsuccessful crimes may have less association or are less likely to transpire together.

**6.4.1.3. Perform Correlation Test**







**6.4.1.4. Correlation test explained**

Looking at the row "homicide," there is correlation coefficients with other crime types such as "offences\_against\_the\_person\_us," "sexual\_offences\_us," "burglary\_us," etc. The correlation coefficients range from approximately 0.85 to 0.94.

A positive correlation coefficient close to 1 suggests a strong positive relationship, indicating that as one type of crime increases, the other type tends to increase as well. Conversely, a negative correlation coefficient close to -1 suggests a strong negative relationship, indicating that as one type of crime increases, the other type tends to decrease.

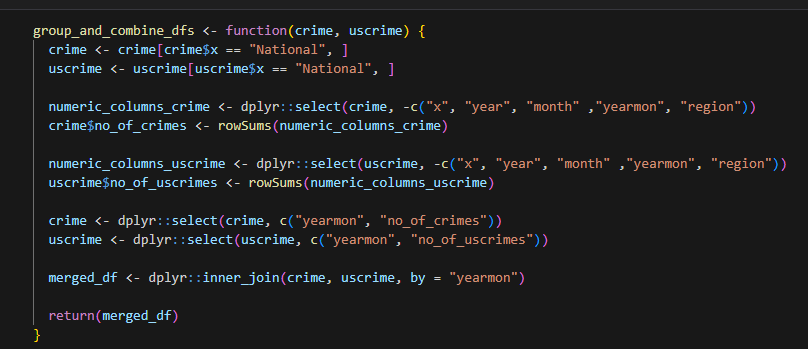
Note that correlation does not imply causation. These correlation coefficients indicate the degree of linear association between crime types but do not provide information about the cause-and-effect relationship between them.

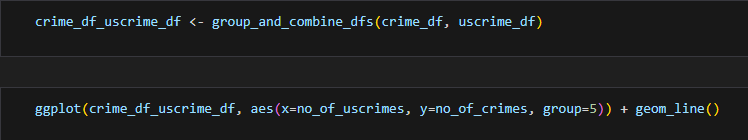
**6.5. Analyzing the trend of successful crime (crime\_df) and unsuccessful crime.**

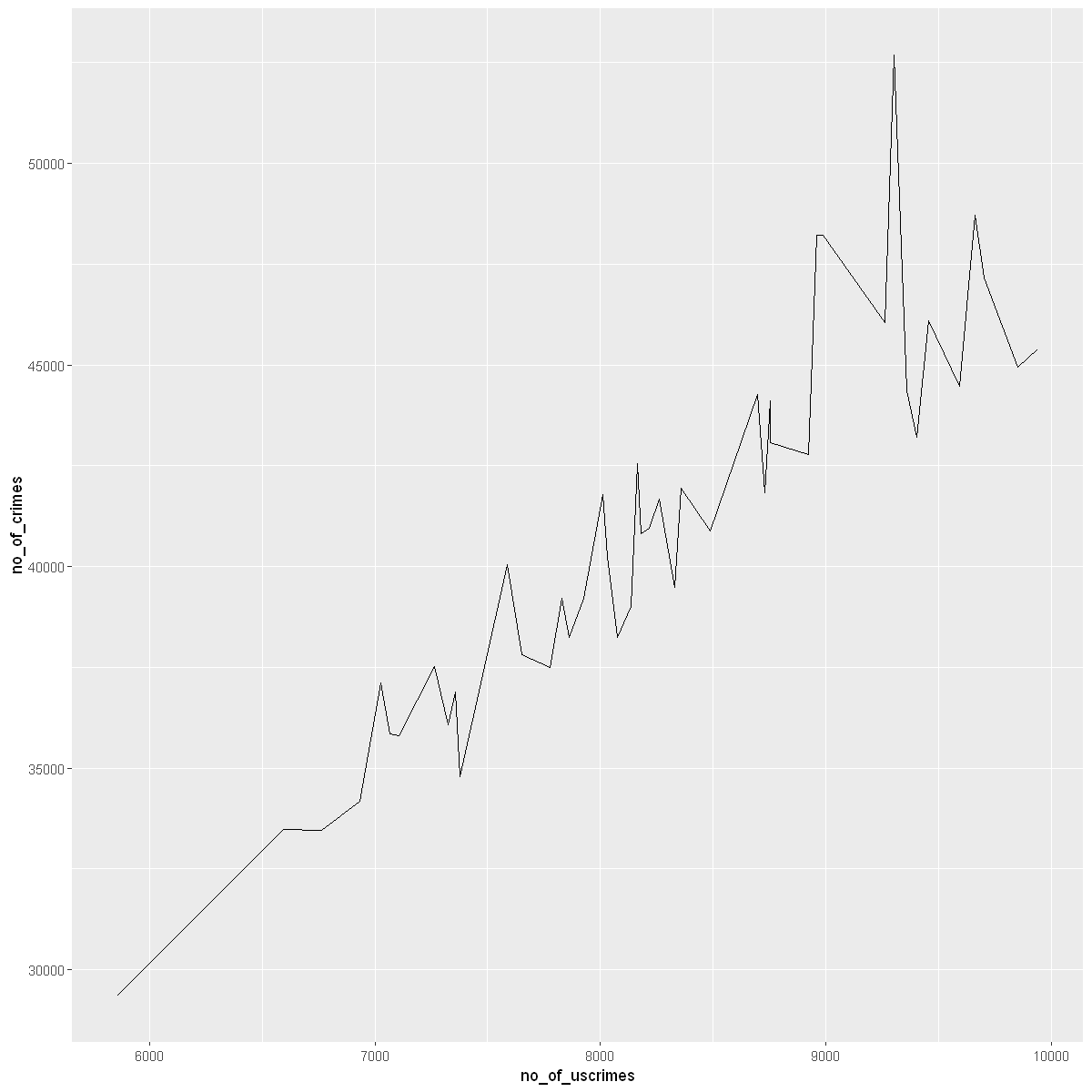
In this analysis section, the goal is to examine the trend between successful and unsuccessful crime occurrences. I aim to gain insights into the patterns and changes over time for both successful and unsuccessful crimes.

To accomplish this, I will analyze the data by considering the frequency or count of successful and unsuccessful crimes across different time periods. By visualizing and comparing these trends, I can identify any notable patterns or shifts in the occurrence of successful and unsuccessful crimes over time.

This analysis will provide me with valuable information about the overall trend in crime rates, the effectiveness of crime prevention measures, and potential areas that require further attention and intervention.







The visualization above depicting successful and unsuccessful crimes reveals a positive correlation between the number of overall crime cases and the number of unsuccessful crime cases.

There are advantages associated with the simultaneous increase in both successful and unsuccessful crime cases:

* It suggests that law enforcement efforts are effective in detecting and addressing criminal activity.
* It indicates that the criminal justice system is proficient in identifying and pursuing cases, even if they ultimately result in unsuccessful outcomes.

However, it is crucial to consider the potential drawbacks:

* + An escalation in unsuccessful crime cases may indicate resource constraints or backlogs within the criminal justice system, leading to delays and inefficiencies.
  + It could imply that law enforcement agencies face challenges in effectively investigating and prosecuting crimes.
  + The simultaneous increase in crime and unsuccessful crime cases may strain the criminal justice system, eroding public trust and confidence.
  + It may result in the prioritization of more serious crimes at the expense of addressing less severe offenses.
  + The justice system's failure to address the underlying causes of crime could perpetuate high crime rates in the long run.

**7. Predictive Analytics**

Predictive analytics is the practice of extracting information from data sets to identify patterns, trends, and relationships in order to make predictions about future events or outcomes. It involves the use of statistical algorithms, machine learning techniques, and data mining methods to analyze historical data and uncover insights that can be used to forecast future behavior.

Predictive analytics can be applied in various fields and industries, such as finance, marketing, healthcare, manufacturing, and more. It utilizes historical data, relevant variables, and mathematical models to generate predictions, probabilities, or trends that can guide decision-making processes.

With advancements in technology and the increasing availability of big data, predictive analytics has gained significant popularity and has the potential to revolutionize various aspects of our lives.

However, it's important to note that predictive analytics relies heavily on historical data, and its accuracy and reliability depend on the quality of the data, the assumptions made, and the models used. It is also subject to ethical considerations, as it may raise privacy concerns when dealing with personal data.

Predictive analytics and prediction models go hand in hand. Predictive analytics is the process of using data, statistical algorithms, and machine learning techniques to make predictions about future events or outcomes. Prediction models, on the other hand, are the mathematical representations or algorithms that are used to generate these predictions.

It's important to note that the accuracy of prediction models depends on the quality of the data, the selection of appropriate variables, the choice of the model, and the assumptions made during the modelling process. Regular monitoring and evaluation are essential to assess the model's performance and make necessary adjustments. I employ the following predictive techniques:

* Regression
* Clustering
* Classification

**7.1. Regression**

Regression is a statistical technique employed to examine the association between a dependent variable (also called the response or outcome variable) and one or more independent variables (also known as predictor or explanatory variables). The objective of regression is to determine the optimal line or curve that represents the relationship between the dependent variable and the independent variables. This line or curve is referred to as the regression line or model.

Various forms of regression exist, such as linear regression, multiple regression, logistic regression, and polynomial regression. In this scenario, we will be utilizing linear regression technique.

7.1.1. Linear Regression

Linear regression is a statistical technique used to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the observed data. It assumes that the relationship between the variables can be represented by a straight line.

In linear regression, the dependent variable is predicted or estimated based on the values of the independent variables. The goal is to find the best-fitting line that minimizes the difference between the observed data points and the predicted values on the line. This line is determined by estimating the coefficients or parameters of the linear equation.

The general form of a linear regression equation with one independent variable can be represented as:

y = β₀ + β₁x + ε

where:

- y is the dependent variable,

- x is the independent variable,

- β₀ is the y-intercept,

- β₁ is the slope or coefficient associated with the independent variable,

- ε is the error term or residual, representing the variability not explained by the linear relationship.

The coefficients β₀ and β₁ are estimated using various statistical techniques, such as the method of least squares, which minimizes the sum of the squared differences between the observed and predicted values. These coefficients provide information about the magnitude and direction of the relationship between the variables.

Linear regression can be extended to multiple regression, where multiple independent variables are considered simultaneously. The equation is then expanded to include additional coefficients for each independent variable.

Linear regression analysis is commonly used for prediction, forecasting, and understanding the relationships between variables in various fields, including economics, finance, social sciences, and engineering. It provides insights into how changes in the independent variables affect the dependent variable and can help in making predictions or estimating unknown values.

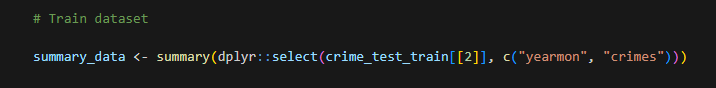
7.1.1.1. Hypothesis

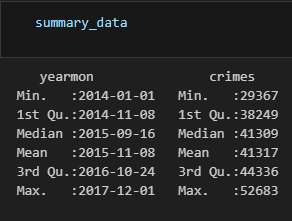
To forecast the quantity of crimes happening using only the date as a single predictor variable.

7.1.1.2. Dataset summary

The process involves reviewing the summaries of both the train and test datasets to train the model and evaluate the outcomes. The dataset in question comprises two variables: "yearmon" representing the date in the format of year and month, and "crimes" indicating the recorded number of crimes for each date.

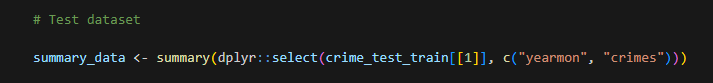
7.1.1.2.1. Train the dataset

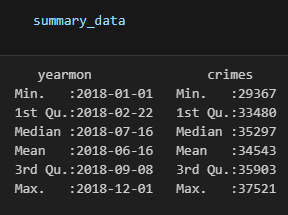




After analysis, it is observed that the earliest recorded date is January 2014, with a corresponding minimum of 29,367 crimes. The middle point or median date falls in September 2015, with an associated median count of 41,308 crimes. Finally, the latest recorded date is December 2017, with a maximum count of 52,683 crimes.

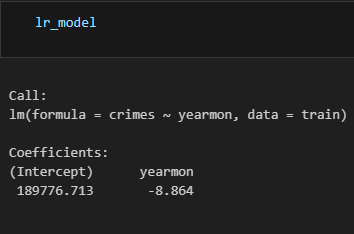
7.1.1.2.2. Test dataset

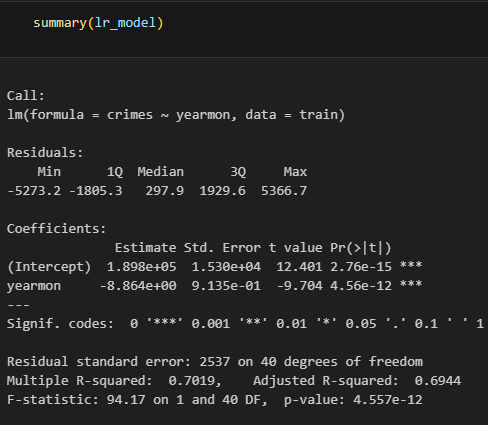




Upon examination, it is evident that the earliest date in the dataset is January 2018, accompanied by a minimum recorded count of 29,367 crimes. The median date, on the other hand, falls in August 2018, with a corresponding median count of 35,808 crimes. Lastly, the latest recorded date in the dataset is December 2018, showcasing a maximum count of 37,521 crimes.

7.1.1.3. Model summary





This is a summary of the linear regression model used for predicting the number of crimes (dependent variable) based on the year and month (independent variable). The summary provides information on the model's coefficients, significance, and goodness of fit.

The coefficients table displays the estimated parameters of the model. The first coefficient represents the y-intercept, which is the predicted value of the dependent variable when the independent variable is zero, and it is calculated as 189,800. The second coefficient corresponds to the slope of the line, which is determined to be -8.864.

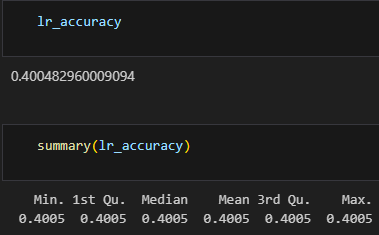
The t-value and p-value in the coefficients table indicate the significance of these coefficients. A small p-value (less than 0.05) suggests that the coefficient is statistically significant, and the t-value is utilized to test the null hypothesis of the coefficient being zero. Typically, a t-value greater than 2 (in absolute value) indicates statistical significance.

The Residuals section illustrates the distribution of the residuals, which are the differences between the observed values of the dependent variable and the predicted values. The values for Min, 1Q, Median, 3Q, and Max provide insights into the range of residuals and whether they are symmetrically distributed.

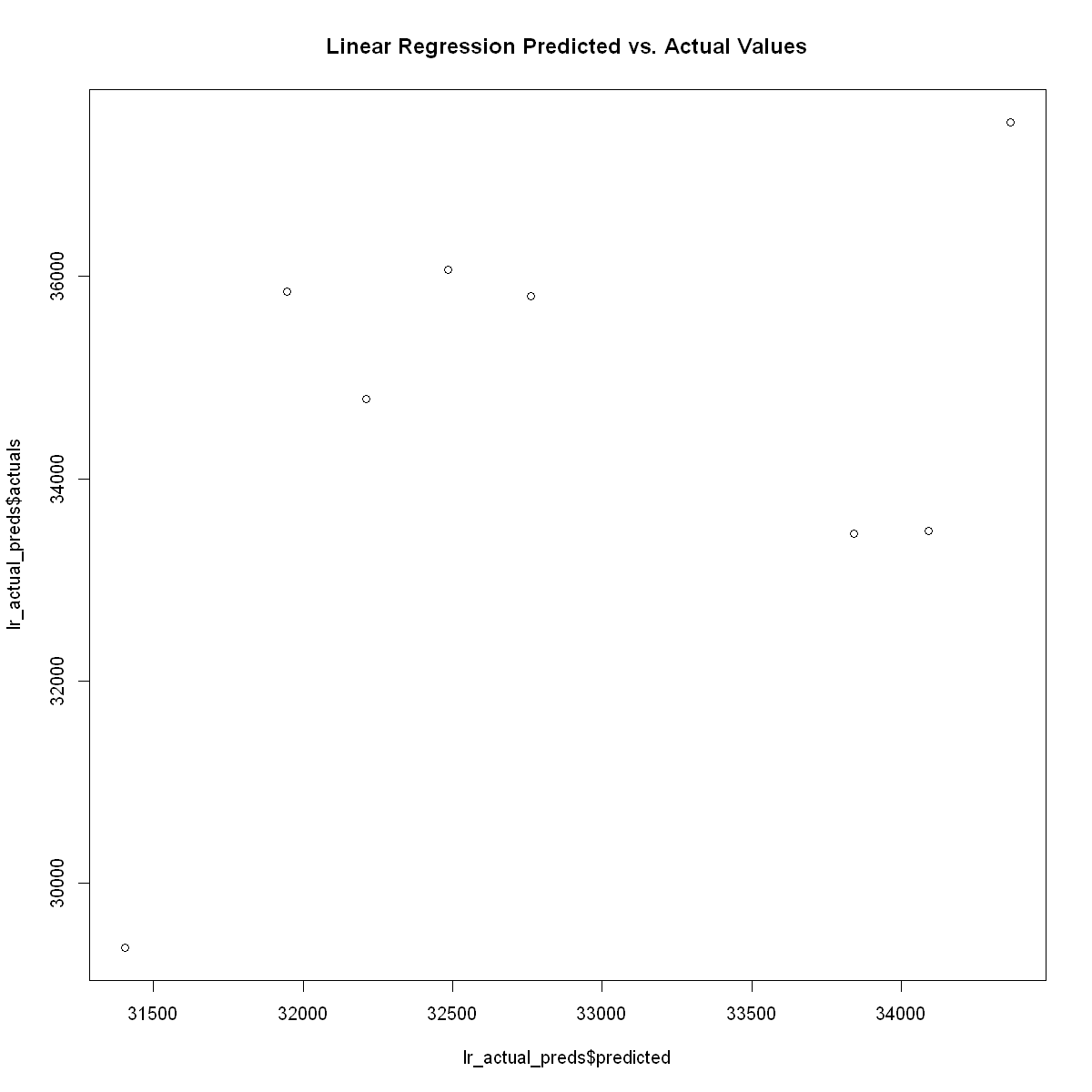
The Residual standard error, R-squared, and the F-statistic are measures of the model's goodness of fit. The residual standard error quantifies the average deviation of residuals from zero, R-squared indicates the proportion of variation in the dependent variable explained by the independent variable, and the F-statistic gauges the overall significance of the model.

In this instance, the R-squared value of 0.7019 and the low p-value of the F-statistic (4.557e-12) suggest that the model exhibits a strong fit and is capable of explaining 70.19% of the variance in the number of crimes.

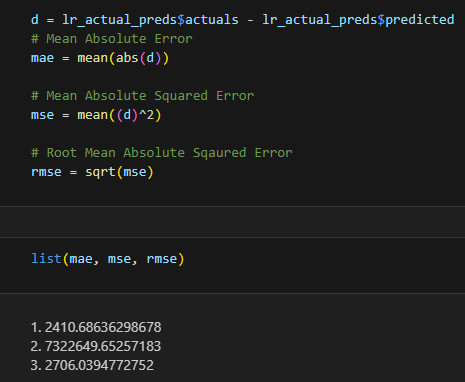
7.1.1.4. Accuracy Matrix



7.1.1.5. Predicted v Actual plot

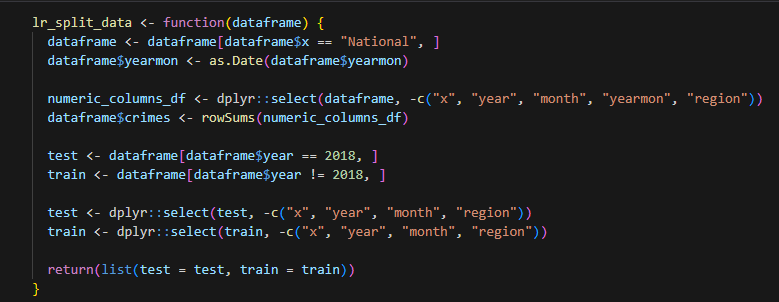


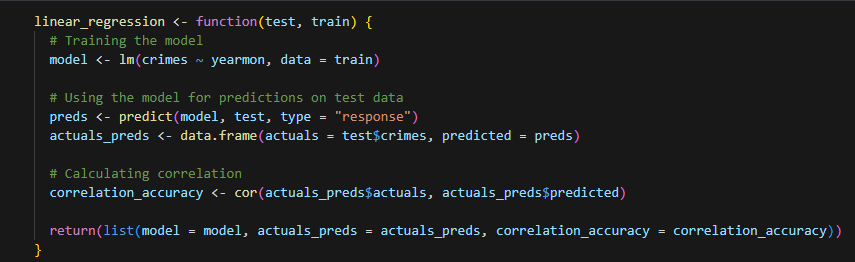
7.1.1.6. Model Evaluation



These values indicate the disparities between the predicted value and the actual value. The Mean Absolute Error (MAE) represents the average of the absolute errors, the Mean Squared Error (MSE) represents the average of the squared errors, and the Root Mean Squared Error (RMSE) corresponds to the square root of the MSE.

7.1.1.7. Linear Regression





This function utilizes two dataframes, namely 'test' and 'train', as inputs. Its purpose is to conduct a linear regression analysis. Initially, the function employs the lm() function, employing 'yearmon' as the predictor and 'crimes' as the response variable, to fit a linear model. Subsequently, the predict() function is employed to generate predictions on the test data using the previously built model. The function then generates a new dataframe that includes both the actual values from the test data and the predicted values. Lastly, the correlation between the actual and predicted values is computed using the cor() function. The function ultimately returns a list comprising the model, the dataframe containing actual and predicted values, and the correlation value.

**7.2. Clustering**

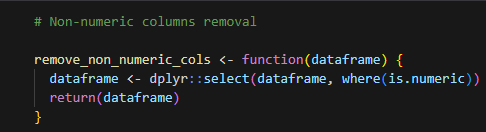
Clustering is an unsupervised machine learning approach employed to group similar data points. Its objective is to identify patterns or structure within the data by organizing comparable observations into clusters. Clustering algorithms accomplish this by partitioning a dataset into a predefined number of clusters based on the similarity among data points within each cluster. Clustering finds utility in various applications including market segmentation, image compression, and anomaly detection. Noteworthy clustering algorithms encompass K-means, Hierarchical clustering, and DBSCAN, among others. For our purposes, we will focus on working with K-Means.

7.2.1. K-Means

K-Means is an unsupervised machine learning algorithm utilized for clustering purposes. Its objective is to divide a collection of points (also referred to as "samples" or "observations") into K clusters, where each point is assigned to the cluster with the nearest mean. The algorithm functions by initially initializing K cluster centroids randomly. It then iteratively reassigns each point to the cluster associated with the closest centroid and adjusts the centroids' positions to reflect the mean of the points within the new clusters. This process is repeated until the centroids no longer move or a maximum number of iterations is reached. The user specifies the number of clusters, K, as a parameter. The outcome of the K-Means algorithm is a data partition into K clusters.

7.2.1.1. Hypothesis

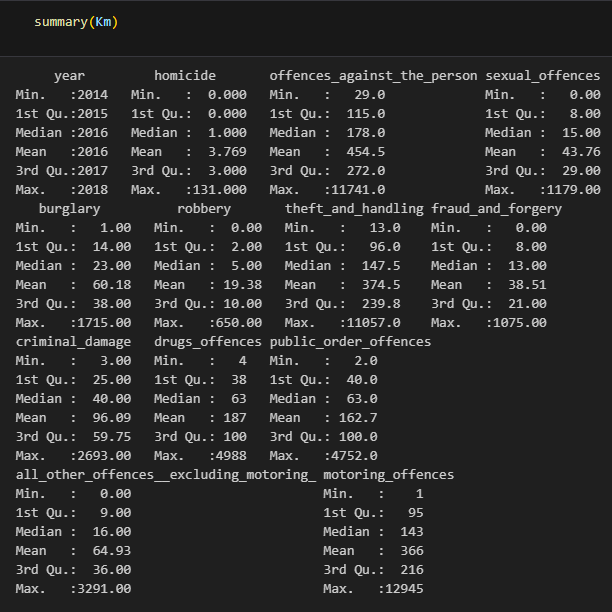
To be able to cluster all of the crime types into clusters



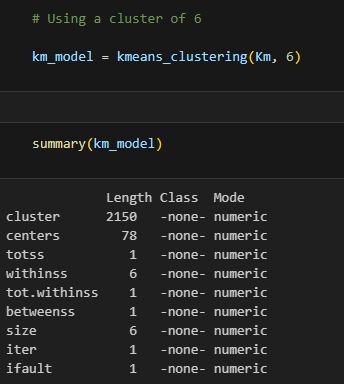


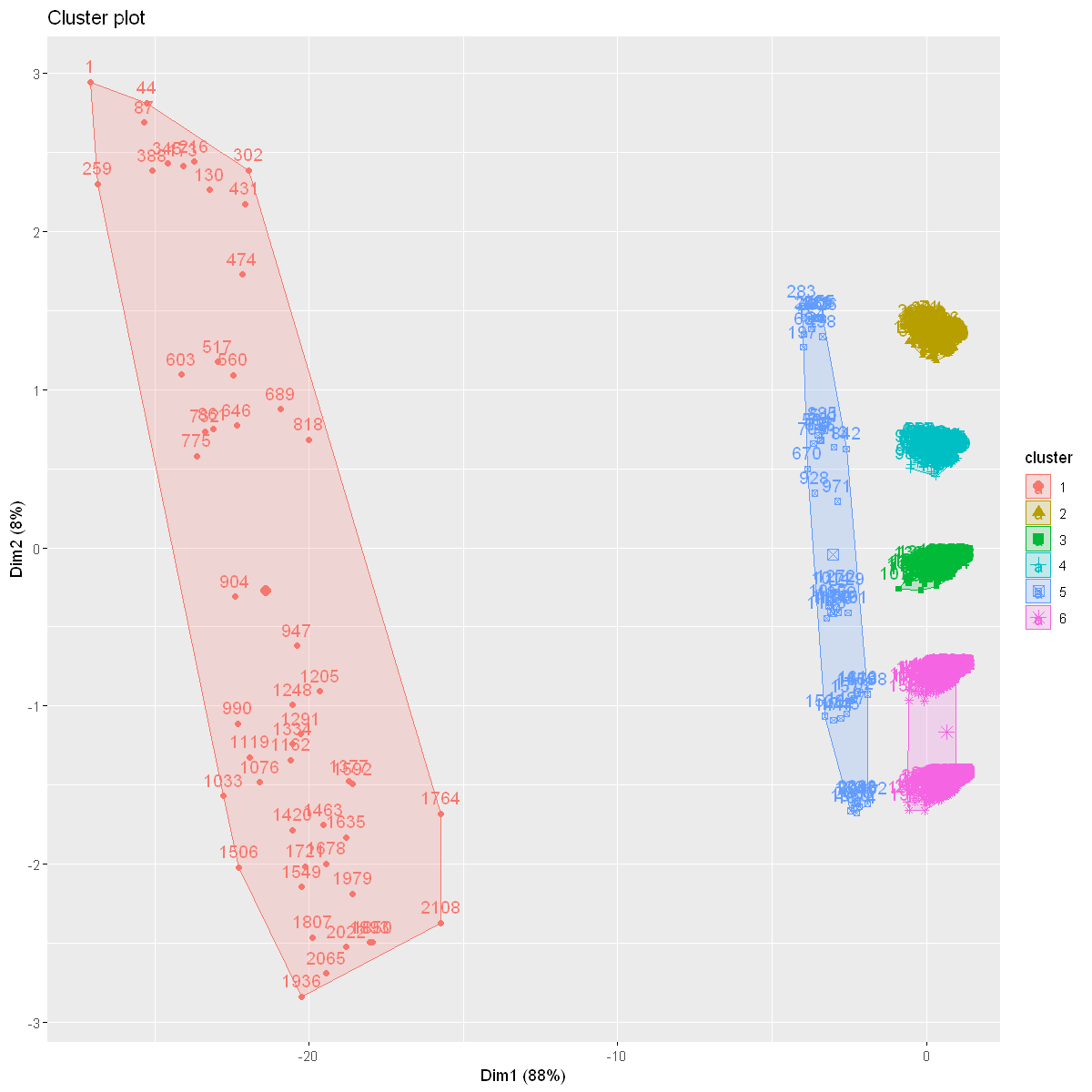


7.2.1.2. Dataset summary

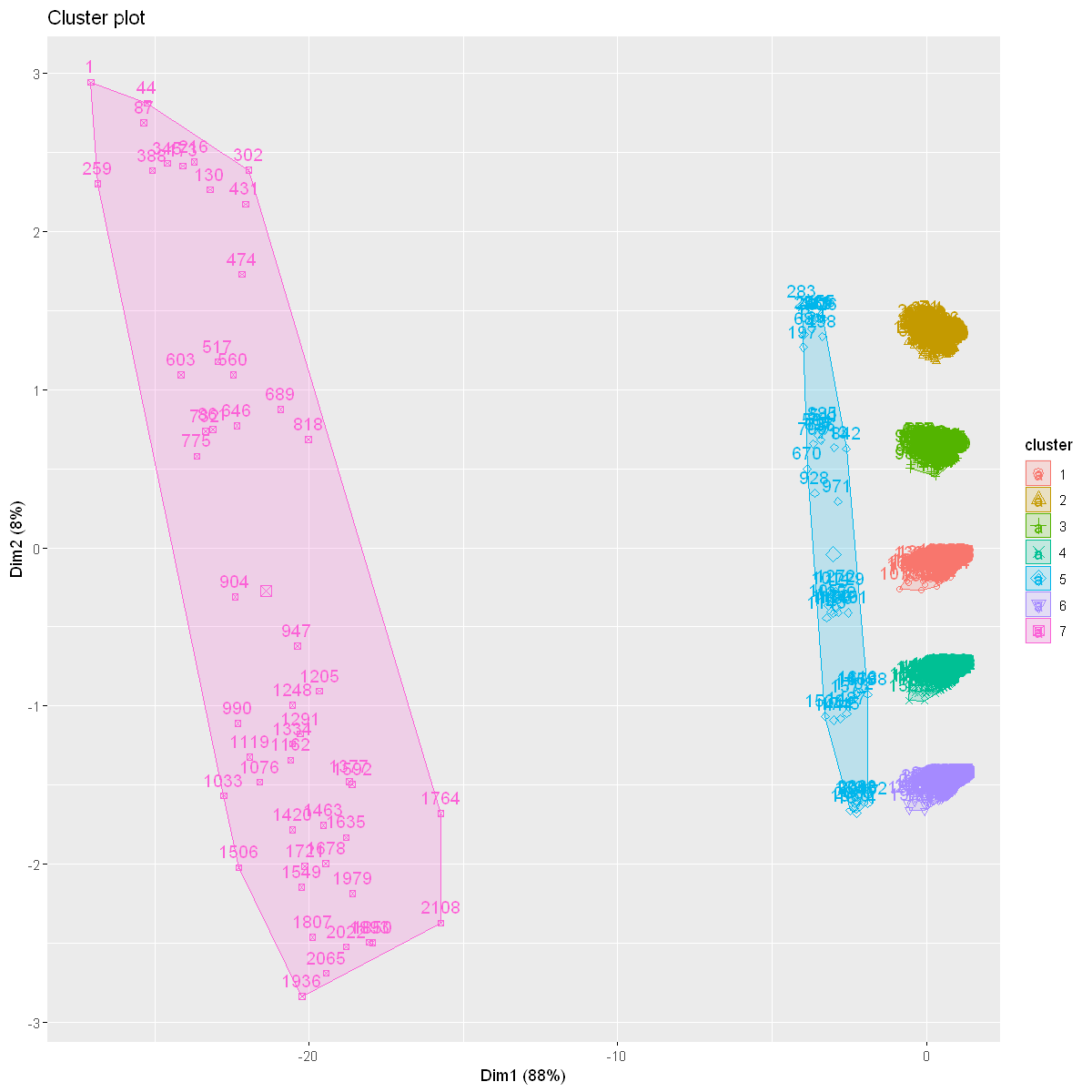


7.2.1.3. Using a cluster of 6

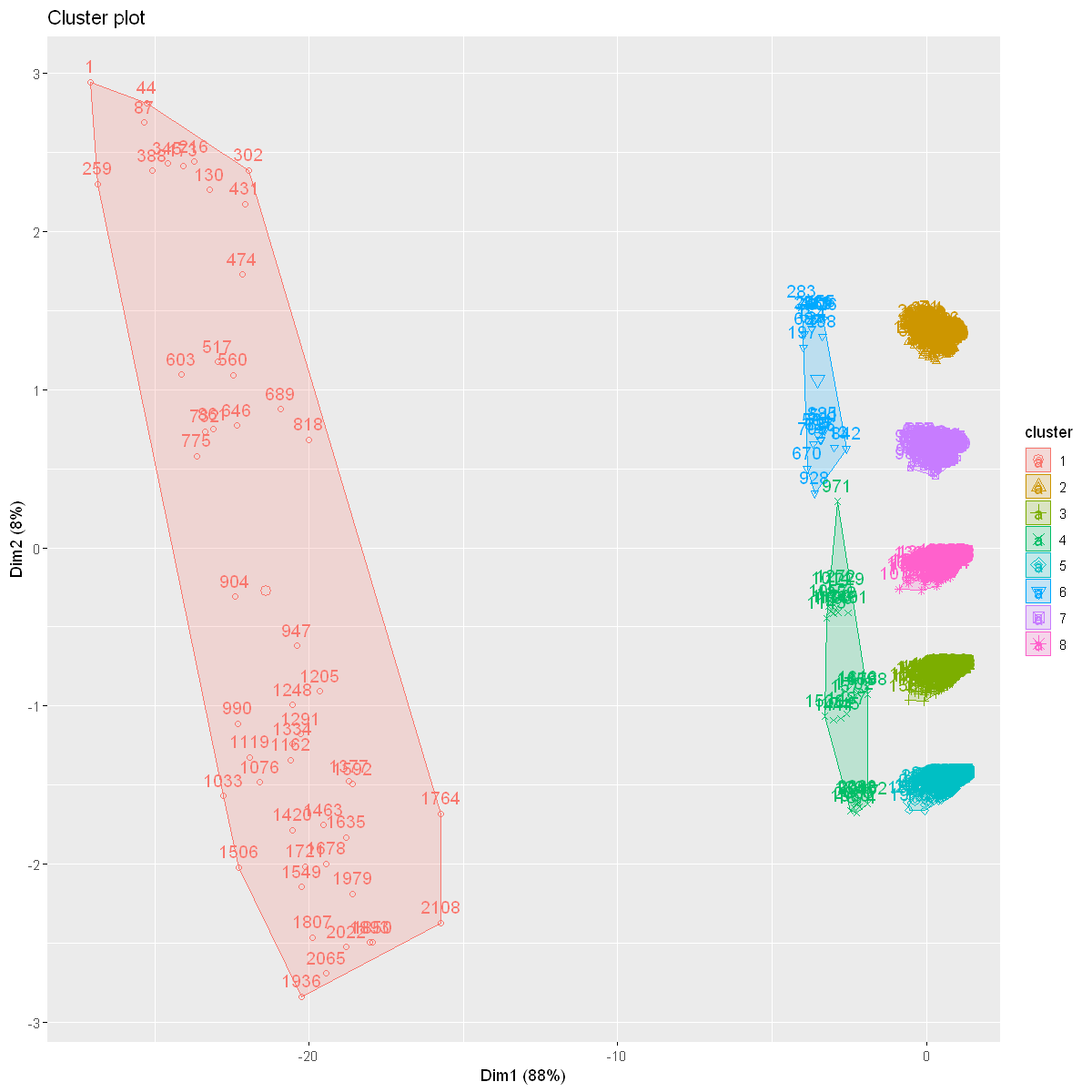




7.2.1.4. Using a cluster of 7



7.2.1.5. Using a cluster of 8



7.2.1.6. Summary

To ensure the successful functioning of the algorithms, the optimal number of clusters is determined to be 5. Increasing the number further results in the breakdown of clusters into smaller ones that are already in close proximity. Conversely, reducing the number of clusters leads to overly generalized clustering. Therefore, the model achieves success in effectively clustering all crime types.

7.3. Classification

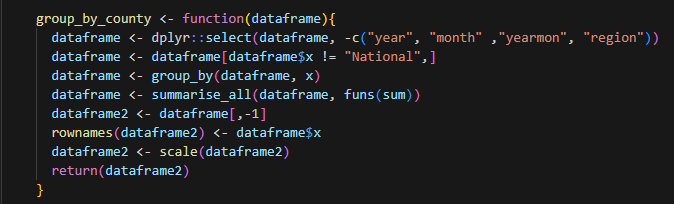
Classification is a supervised machine learning technique utilized to anticipate the class or category of an input sample based on a set of features. Its purpose is to assign data into predefined classes. The classification process commences with training a model using a labeled dataset where the target variable is already known. The model comprehends the correlation between the input features and the target variable, enabling it to predict the class of new, unseen data.

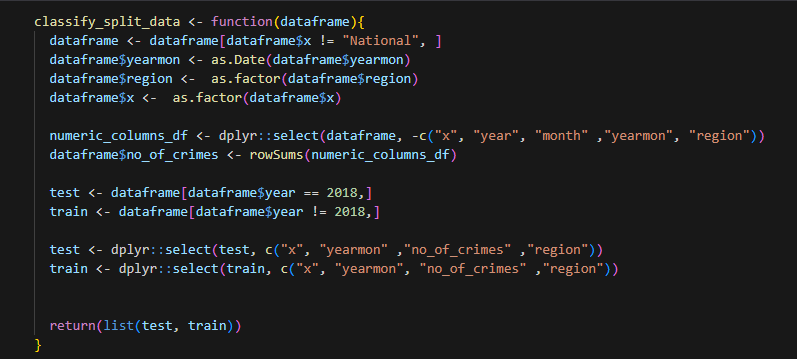
There are two primary types of classification: binary classification and multi-class classification.

Binary classification determines one of two possible outcomes. For instance, discerning whether an email is spam or not.

Multi-class classification ascertains one of numerous possible outcomes. For example, classifying images of handwritten digits as 0-9.

Various classification algorithms exist, including Logistic Regression, k-Nearest Neighbors (k-NN), Decision Trees, Random Forest, Naive Bayes, and Neural Networks. Each algorithm exhibits distinct strengths and weaknesses, with the selection of an algorithm dependent on the data characteristics and the specific problem at hand. In this case, SVM will be deployed.





7.3.1. SVM

Support Vector Machines (SVMs) are a supervised learning algorithm widely used for classification and regression tasks. Their objective is to identify the optimal boundary, or hyperplane, that effectively separates data into distinct classes. This boundary is carefully selected to maximize the margin, which refers to the distance between the boundary and the closest data points from each class, known as support vectors.

SVMs are versatile and can handle both linear and non-linear classification problems using a technique called the kernel trick. This technique enables SVMs to transform input data into a higher dimensional space where a linear boundary can effectively segregate the classes. Various kernels are commonly employed in SVMs, such as linear, polynomial, and radial basis function (RBF).

In real-world applications, SVMs have demonstrated considerable success, particularly in tasks like image and document classification, bioinformatics, and text classification. One of the key advantages of SVMs is their ability to handle high-dimensional data and their robustness against overfitting. However, the choice of kernel can impact their performance, and the results may also be influenced by the scaling of input features.

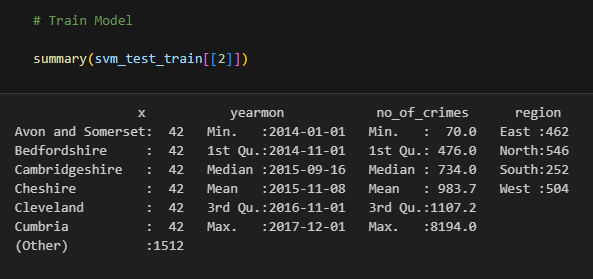
7.3.1.1. Hypothesis

To be able to successfully classify the region based on county, yearmon and sum of all crime values.

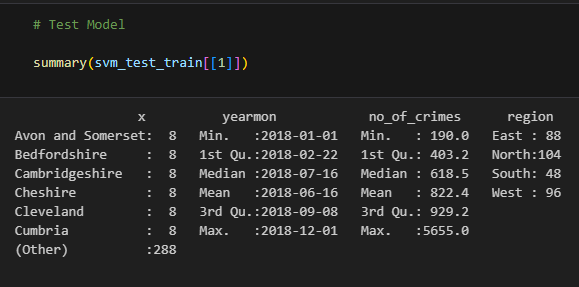
7.3.1.2. Dataset summary

The dataset is divided into two parts: the test portion and the train portion. In the test set, only data from the year 2018 is included, while the remaining years' data are utilized for training. Our primary task is to classify the four classes within the region.

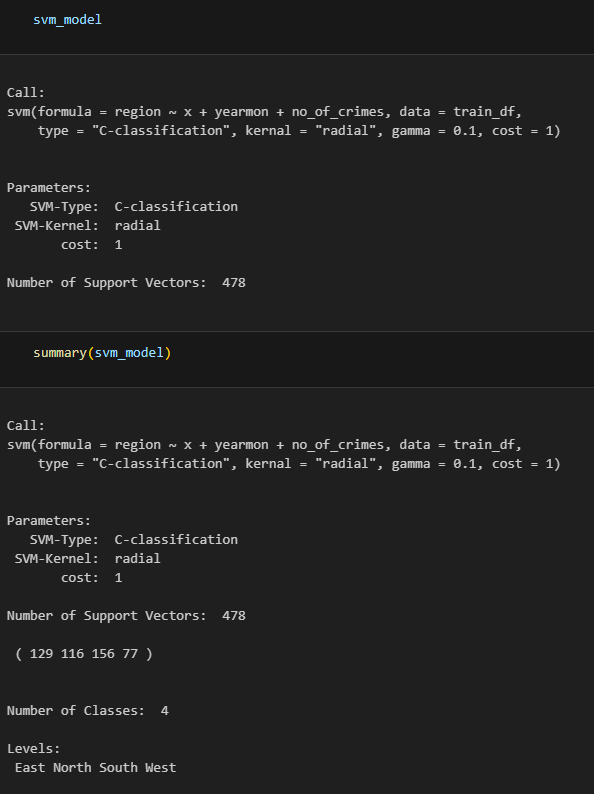
7.3.1.2.1. Train the dataset



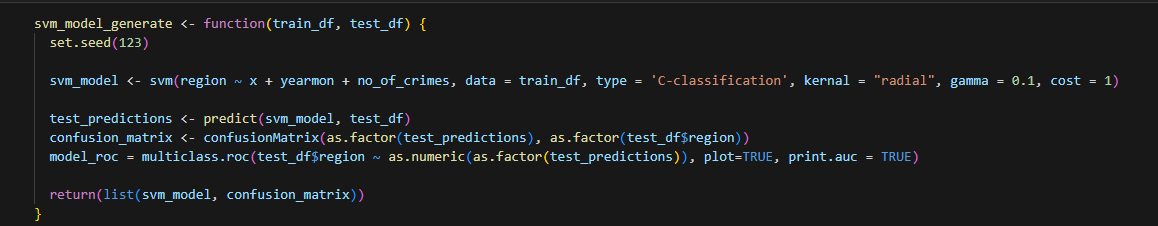
7.3.1.2.2. Test the dataset

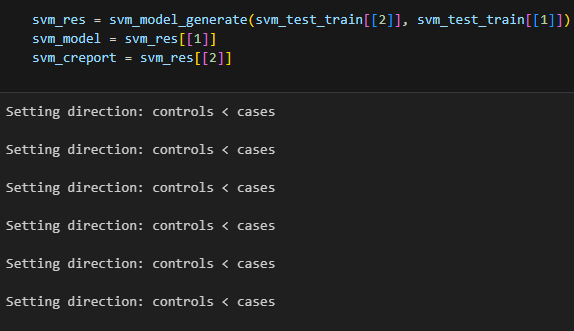


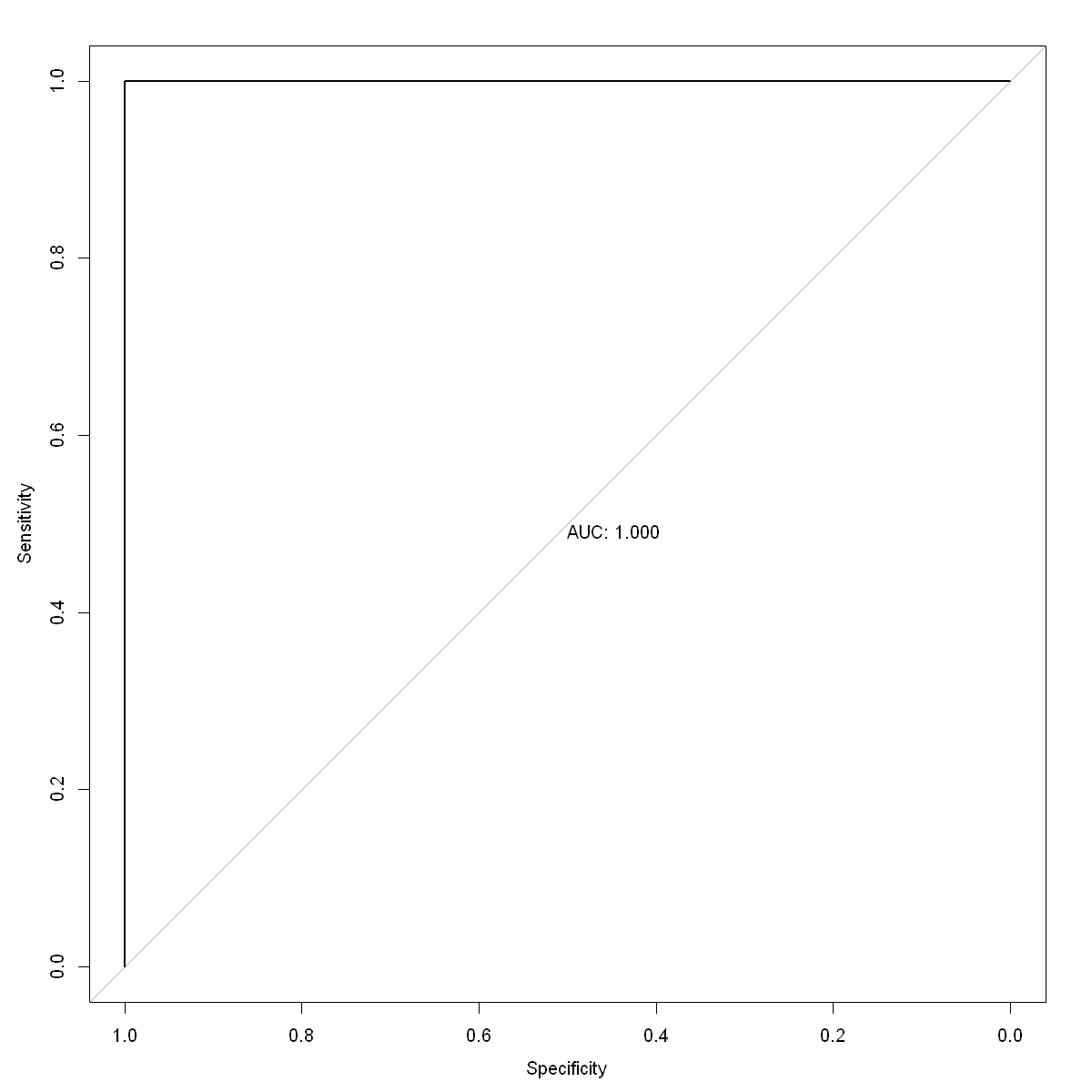
7.3.1.3. Model Summary

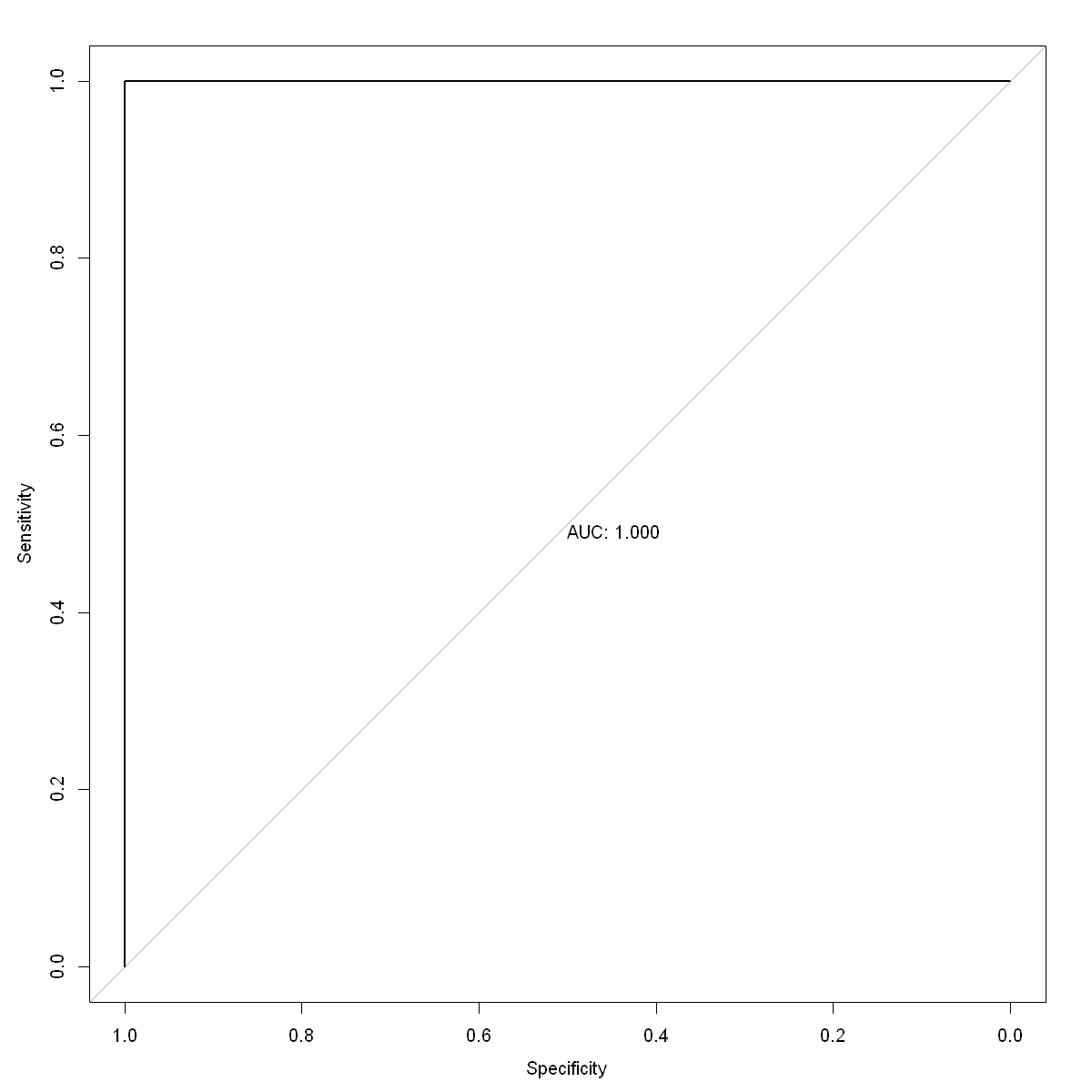


The summary reveals that the classification approach utilizes SVM with a radial kernel and a cost parameter of 1. The dataset consists of four distinct classes.









A perfect ROC for model training which means the model is well fitted.

In this case, the provided report presents the matrix and statistics for the SVM model. The model is specifically trained to classify directions as East, North, South, or West. The report evaluates the model's performance in terms of various metrics, including accuracy, kappa, sensitivity, specificity, Positive Predictive Value, Negative Predictive Value, and Balanced Accuracy.

The confusion matrix illustrates the counts of true positives, true negatives, false positives, and false negatives for each class. The model achieves an overall accuracy of 1, indicating perfect classification accuracy of 100%. The Kappa statistic is also 1, indicating complete agreement between the predicted and actual labels.

All other metrics such as Sensitivity, Specificity, Positive Predictive Value, and Negative Predictive Value are also 1, implying that the model accurately predicts all observations in the test dataset. The Detection Rate, Detection Prevalence, and Prevalence are likewise 1, indicating that the model successfully detects all observations in the test dataset.

Furthermore, the Balanced Accuracy is 1, meaning that the model classifies all observations in the test dataset with 100% accuracy. Additionally, the McNemar's Test P-Value is NA, indicating that the test is not applicable in this particular case.

Overall, the report demonstrates exceptional performance of the model, achieving a perfect 100% accuracy in accurately classifying the directions East, North, South, and West.