Exploratory Data Analysis on the automobile.txt Dataset

Report

This report was written by Sarennah Longworth-Cook

10 August 2025

Introduction

The automobile.txt dataset contains information about cars, including price, size and engine-related parameters as well as categories such as make and body style.

The original data comprises 205 rows and 26 columns.

There is no metadata associated with the dataset, which restricts interpretation and limits the application of findings due to the possibility of causal factors external to the dataset.

The methods used to clean the data prior to analysis are set out in Appendix A: Data cleansing approach and results.

The methods used in this analysis are set out in Appendix B: Methods and explanatory notes.

Background

Analysis of vehicles to suggest a range of executive cars suitable for senior management that are aligned with the new company policy of aggressively reducing environmental impact and costs. A factual analysis is required to enable data-driven decision-making.

Resultant recommendation

From the dataset provided, the optimised executive cars are Chevrolet or Honda, based on fuel efficiency and economy. To take into account the status-symbol aspect of the company cars and the importance of driving characteristics for long journeys, the vehicles chosen should be sedans from the top of the Chevrolet and Honda ranges.

Relationships between variables

Vehicle model frequency in the dataset

The manufacturer with the most models in the dataset is Toyota (Figure 1) with over 30 models included. The manufacturer with the least models in the dataset is Mercury with only 1 model. This broadly reflective of market share, with Mercury being a lesser-known and presently discontinued brand of the Ford Motor Company that was only sold in North America. In comparison, Toyota is a world-leading car manufacturer and has been for many years[1].

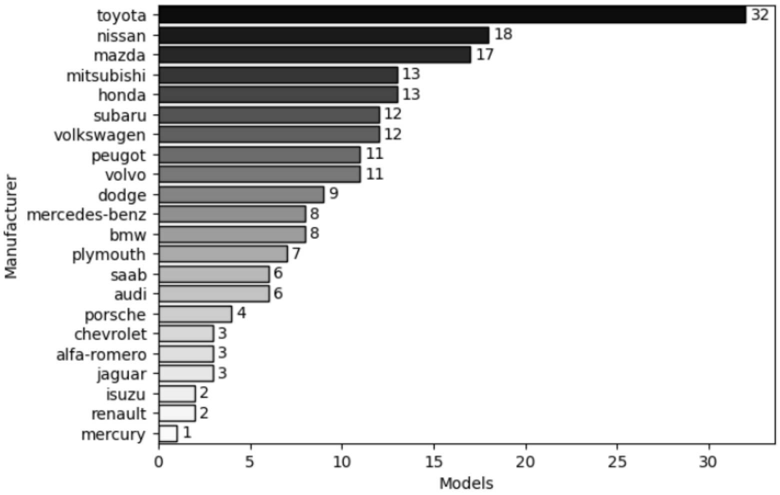


Figure 1 The number of models associated with each manufacturer in the dataset.

Cost and fuel efficiency

The dataset contains a large range of prices, from 5,000 – 45,000 curr[2] (Figure 2). The distribution is left-skewed with a small second peak at approximately 35,000 curr.

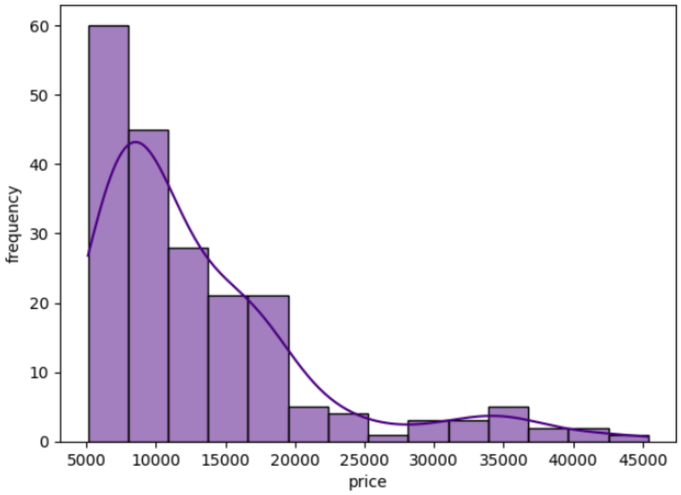


Figure 2 The frequency distribution of price in the dataset.

The most expensive car in the dataset is a Mercedes-Benz hardtop at 45,400 curr (Table 1). Among the five most expensive cars, the engines are large, with high horsepower and high peak RPMs, suggesting vehicles with lots of power that will be pleasing to drive for motoring enthusiasts. However, the miles per gallon (MPG) of these cars on the highway is low.

[1] https://en.wikipedia.org/wiki/List_of_automotive_manufacturers_by_production

[2] In lieu of metadata to identify the unit of currency, I will use “curr” to signify the unknown unit of currency.

Table 1 Characteristics of the five most expensive cars in the dataset. All the cars in the table are gas-fuelled and rear-wheel drive with multi-point fuel injection (mpfi).

Make (index)	Price curr	Doors	Body Style	Cylinders	Engine size	Horsepower	Peak RPM	Highway MPG
Mercedes-Benz (74)	45,400	2	Hardtop	8	304	184	4500	16
BMW (16)	41,315	2	Sedan	6	209	182	5400	22
Mercedes-Benz (73)	40,960	4	Sedan	8	308	184	4500	16
Porsche (128)	37,028	2	Convertible	6	194	207	5900	25
BMW (17)	36,880	4	Sedan	6	209	182	5400	20

The fuel efficiency of the five cheapest cars, five mid-range cars and the five most expensive cars is shown in Figure 3 and a grouped bar chart showing the fuel efficiency of the cheapest, median and most expensive cars is shown in Figure 4. There is almost no overlap between the points for each price group, supporting the assertion that the most expensive cars have the lowest fuel efficiency.

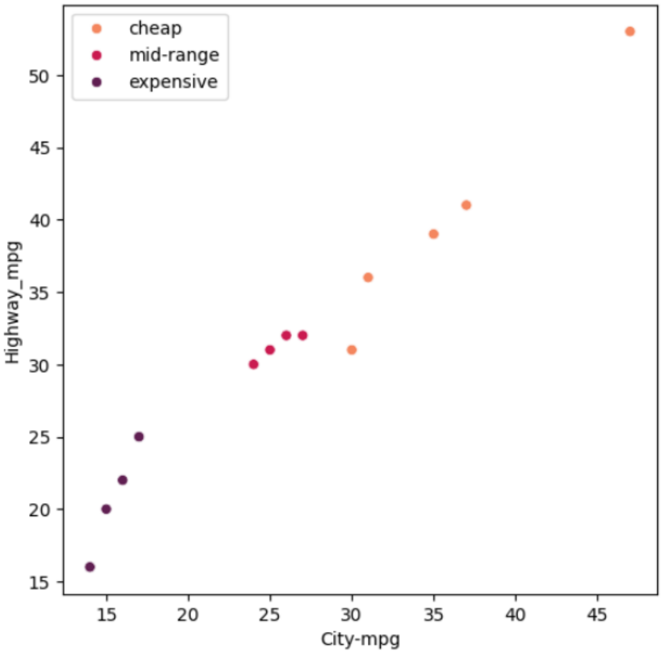


Figure 3 Scatterplot of fuel efficiency measured in miles per gallon (MPG) in the city and on the highway for the five cheapest cars, five mid-range cars and the five most expensive cars. The five mid-range cars are centred on the median price.

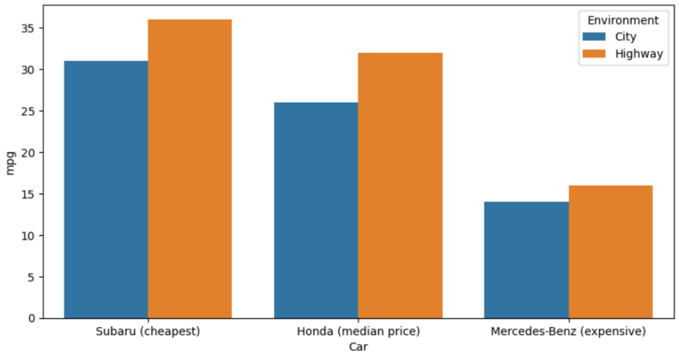


Figure 4 Fuel efficiency in city and highway environments for the cheapest car, the median-priced car and the most expensive car in the dataset.

The distribution of fuel efficiency as a function of car price for all the data reveals an inverse relationship between fuel efficiency and price (Figure 5). In this dataset, the highest priced cars have the lowest fuel efficiencies and the lowest priced cars have the highest fuel efficiencies. If a prospective buyer prioritizes their budget, both at time of purchase and during their use of the vehicle, above all other considerations, they will choose a lower priced car.

However, cars are more than a mode of transport for most people: they are visible indicators of status and personal choice. It is clear from the data that the most expensive cars are not optimised for fuel efficiency but for other factors.

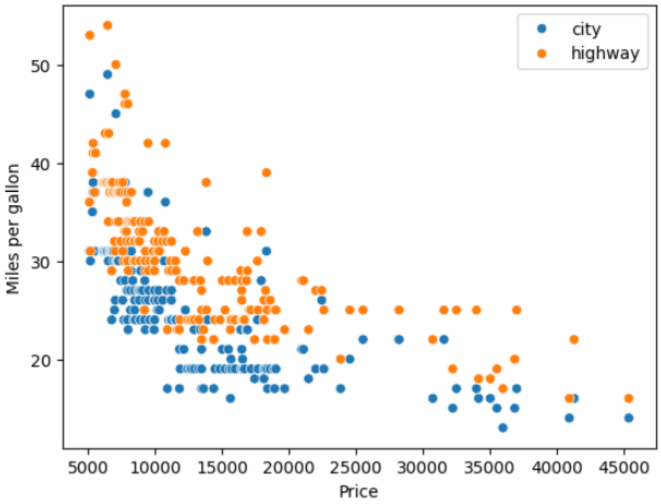
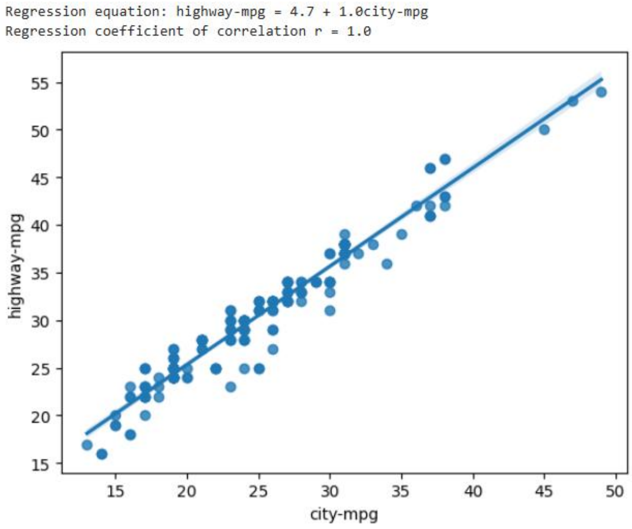


Figure 5 Scatterplot of fuel efficiency in city and highway environments and relating those figures to the price of the car.

City and highway fuel efficiency

A linear relationship between city and highway MPGs is suggested by Figure 3 and Figure 5. Expanding the scatterplot to include all the data shows this is indeed the case (Figure 6). The regression equation is highway-mpg = 4.7 + city-mpg and the regression coefficient of correlation r is 1.0, indicating a very good fit. With an r-value of 1.0, it is possible the model is overfitted to this dataset, so further testing is needed before applying to other datasets.



```
Regression equation: highway-mpg = 4.7 + 1.0city-mpg
Regression coefficient of correlation r = 1.0
```

Figure 6 Scatterplot of fuel efficiency measured in miles per gallon (MPG) in the city and on the highway.

The frequency distributions for city and highway fuel efficiency have differences in the proportions of the quartiles, reinforcing the need to apply the above model to other datasets with care (Figure 7).

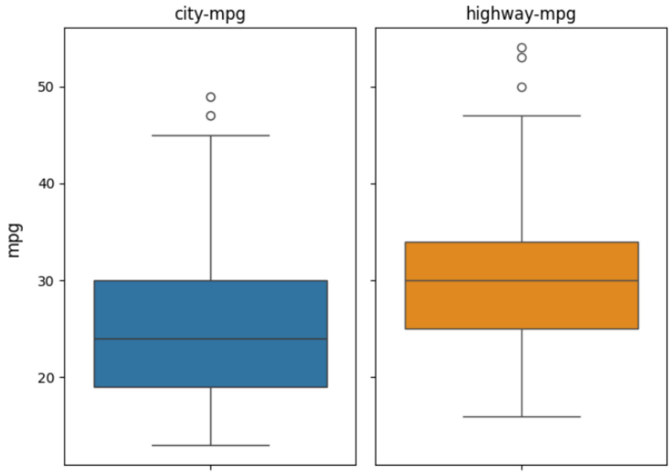


Figure 7 Boxplots showing the distributions of city and highway fuel efficiency measured in miles per gallon (MPG).

Manufacturers of fuel-efficient vehicles

The most fuel-efficient car in the dataset is a Honda and the least fuel efficient is a Jaguar (Table 2). The Jaguar is over five times more expensive than the Honda, consistent with the findings above that the more expensive a car is, the less fuel efficient it is.

Table 2 Characteristics of the most and least fuel-efficient cars in the dataset. Both cars are gas-fuelled with two doors.

Make (index)	City MPG	Highway MPG	Price curr	Body style	Engine size	Horsepower
Honda (30)	49	54	6479	Hatchback	92	58
Jaguar (49)	13	17	36000	Sedan	326	262

The mean MPG figures by car manufacturer are shown in Figure 8. The manufacturer with the most fuel efficient range of vehicles is Chevrolet, and it is clearly ahead of second-placed

Honda. However, there are only three datapoints for Chevrolet, while there are 13 for Honda (Figure 1) and the mean value is strongly influenced by outliers. More Chevrolet datapoints are required to support the assertion that Chevrolet produces the most fuel efficient cars in general, outside this dataset.

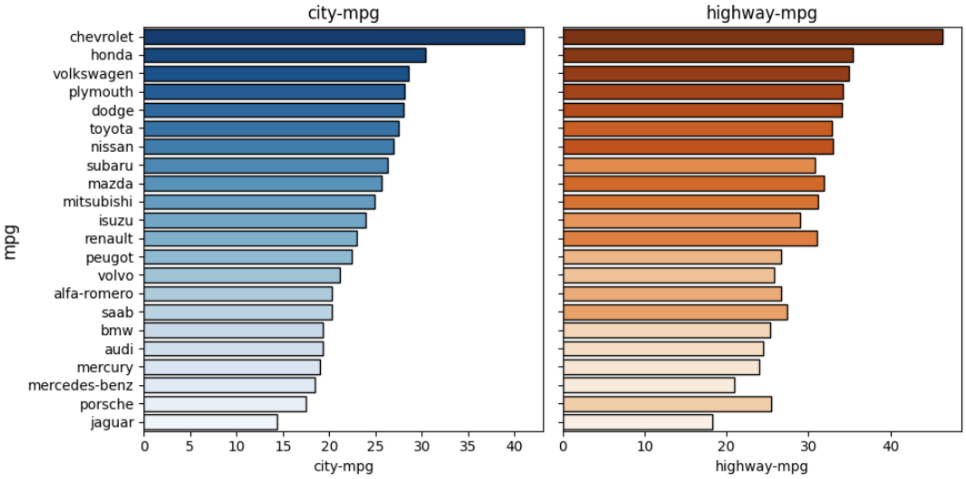


Figure 8 Mean fuel efficiency measured in miles per gallon (MPG) in city and highway environments for each manufacturer in the dataset.

The positions for each manufacturer are broadly consistent in both environments, however in the middle-order of this ranking there is some movement of 2-3 positions. The most notable shift is Porsche, which has the second lowest city MPG but the tenth lowest highway MPG. If a consumer is prioritising fuel efficiency when considering their car, they would be well-advised to consider a Chevrolet or Honda model.

Fuel efficiency and fuel type

Table 3 Fuel types within the dataset.

Fuel type	Frequency
Gas (petrol)	181
Diesel	20

Among the general European public diesel cars are perceived to be more fuel efficient than gas (petrol) cars. In this dataset, only 20 cars are fuelled by diesel (**Error! Reference source not found.**), however even with this restricted amount of data, city fuel efficiency appears to be generally higher for diesel cars (**Error! Reference source not found.**).

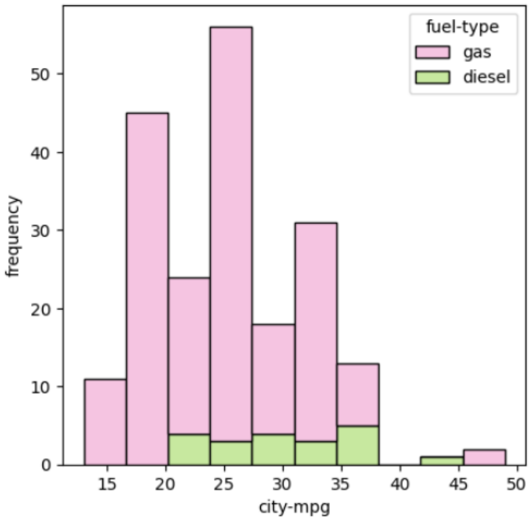


Figure 9 City fuel efficiency measured in miles per gallon (MPG) by fuel type.

Engine size

The engine size distributions by body style are shown in Figure 10. The largest engines are in sedan cars and the smallest in hatchbacks. In this dataset, the size of engines in wagons appears relatively small, given that wagons are typically larger vehicles for transporting more passengers or cargo. Therefore, these wagons may be considered underpowered. Alternatively, if the wagons have an appropriate engine size for their physical size and weight, it highlights the overpowered nature of the sedans with the largest engine sizes.

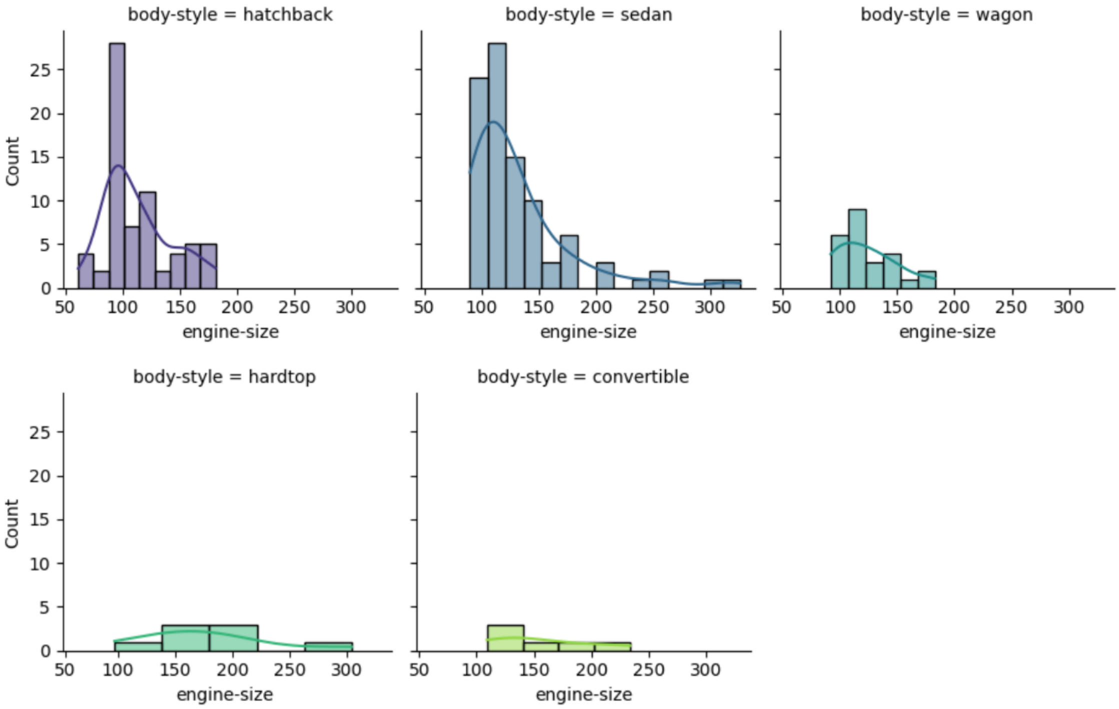


Figure 10 Engine size frequency distributions by body style.

The mean engine size by body style shows a different trend, with the largest mean engine size being hardtop and convertible sports cars (Figure 11).

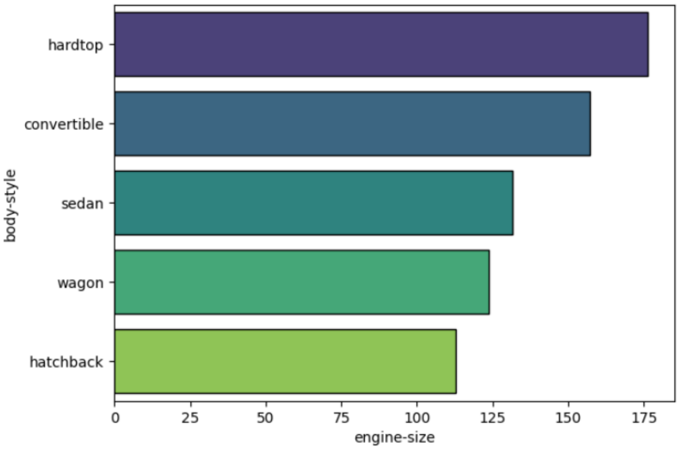


Figure 11 Mean engine size by body style.

The mean engine size by manufacturer has almost exactly the reverse ranking of fuel efficiency by manufacturer (compare Figure 12 with Figure 8).

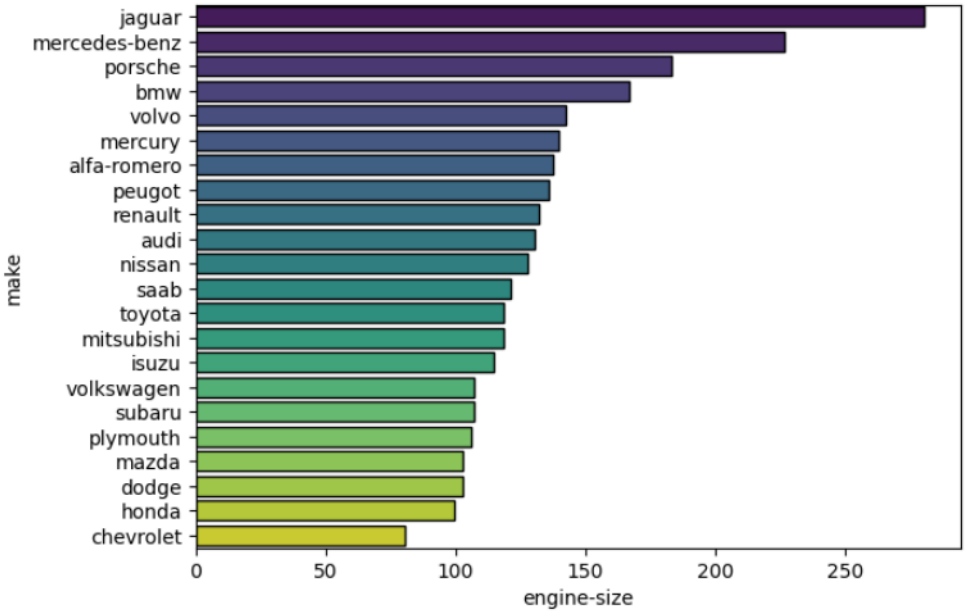


Figure 12 Mean engine size by manufacturer.

The dataset contains several numerical data columns related to the dimensions of the vehicles, engine parameters and fuel efficiency. Several of these are related variables: there is an established relationship between city-mpg and highway-mpg (Figure 6) and curb-weight, length, width, wheelbase and height can all be considered as proxies for ‘size’. Figure 13 shows the correlation heatmap between the remaining unrelated variables (city-mpg, curb-weight, length and width were removed).

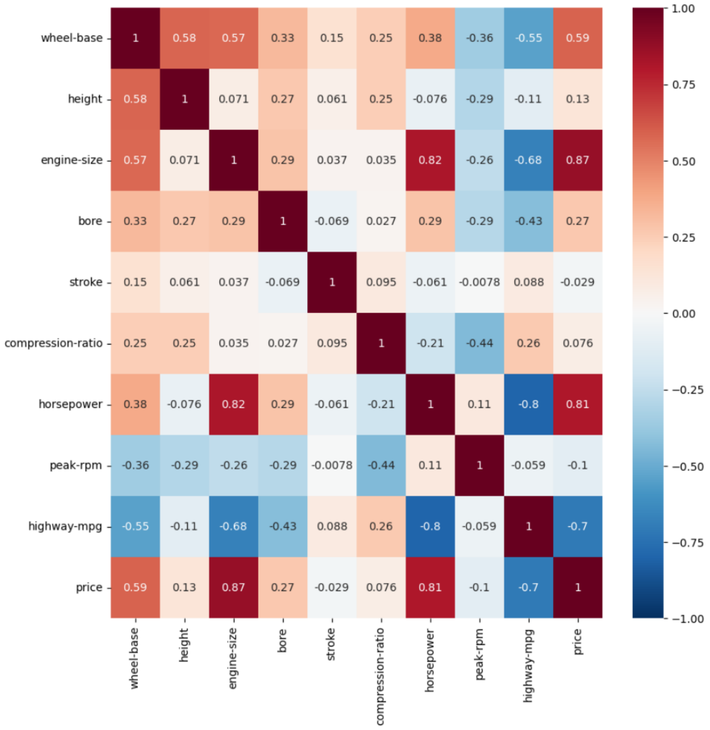


Figure 13 Correlation heatmap for the dataset.

The heatmap shows pale colors on the engine size row, indicating weak or null relationships, for height, stroke, bore, compression-ratio and peak-rpm. In contrast, the colors for wheelbase, horsepower, highway-mpg and price have higher intensity indicating stronger relationships between these variables and engine-size.

Error! Reference source not found. shows relationships between the numeric parameters m ost closely correlated with engine size. The histograms on the diagonal show the distributions of each parameter. The upper triangle shows relationships between pairs of parameters as a scatterplot. The lower triangle shows the same relationships in a KDE plot.

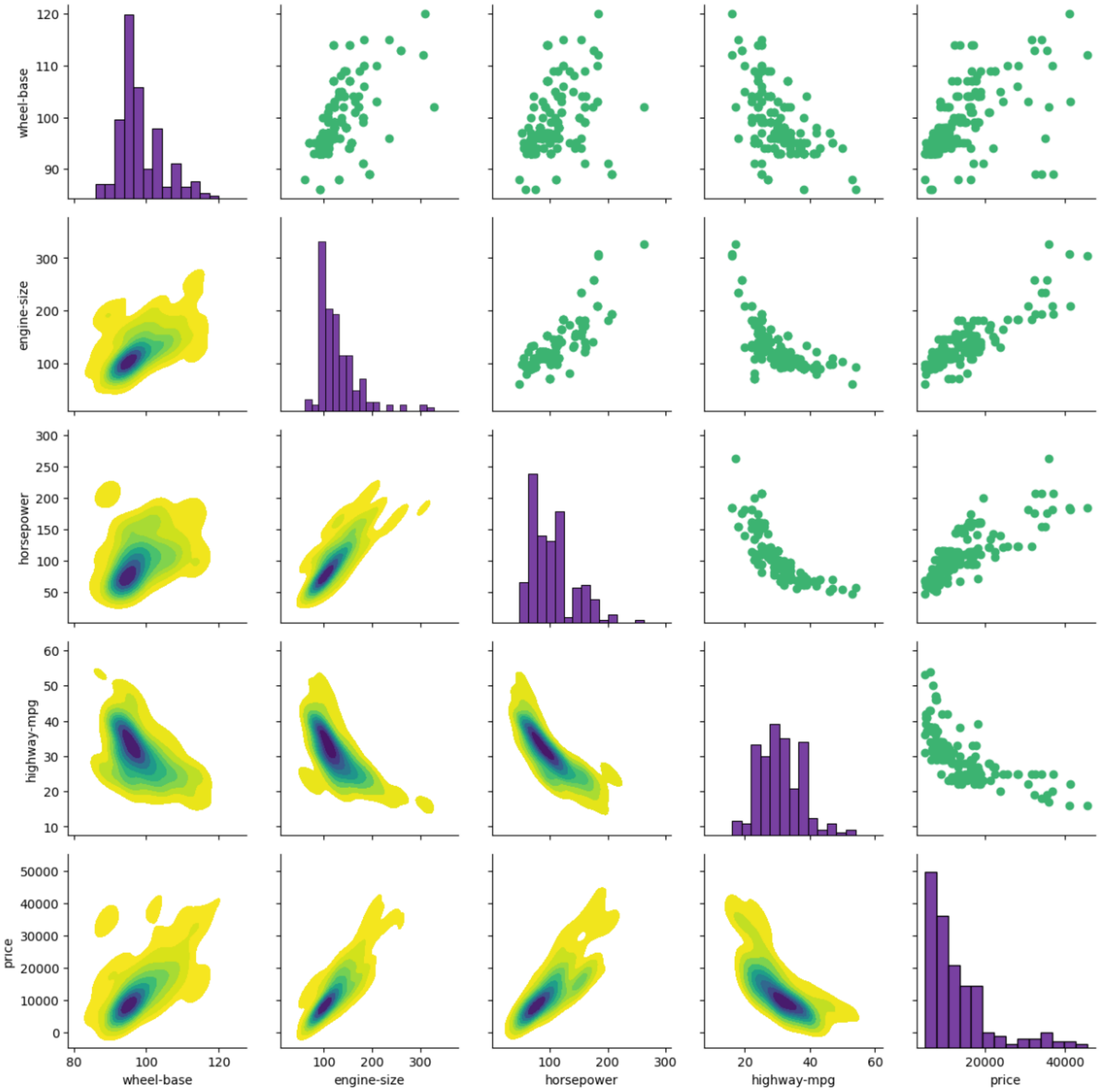


Figure 14 Relationships between the numerical parameters most strongly correlated with engine size.

All the KDE plots show high probability density in the bottom left corner of the plots, consistent with the left skew in engine size, horsepower and price distributions. With the possible exception of fuel efficiency-price KDE plot, all of them contain a single mode. In the case of fuel efficiency-price there is a hint of a second mode of high price vehicles: these potentially represent luxury cars.

The strongest, clearest relationships are between price and horsepower and engine-size. The KDE plots show a narrowly defined and positively correlated probability density function. It is clear that as engine size increases, horsepower and price also increase.

The inverse relationship between fuel efficiency and engine size identified previously, is also visible here.

The relationship between engine size and wheelbase is the weakest shown. The contours of the KDE plot show a weakly positive correlation.

Overall these distributions follow the expected trends from previous analysis.

Conclusions

- Toyota has the most models in the dataset
- Mercedes-Benz hardtop is the most expensive
- Price is strongly negatively correlated with fuel efficiency: the most expensive cars are the least fuel efficient
- Honda and Chevrolet ranges generally contain more fuel efficient models
- The largest engines are found in hardtop and convertible sports cars and certain sedans

Resultant recommendation

From the dataset provided, the optimised executive cars are Chevrolet or Honda, based on fuel efficiency and economy. To take into account the status-symbol aspect of the company cars and the importance of driving characteristics for long journeys, the vehicles chosen should be sedans from the top of the Chevrolet and Honda ranges.

Appendix A: Data cleansing approach and results

Clean the data

The initial data has 205 rows and 26 columns.

The columns ‘normalized-losses’ and ‘symboling’ were identified as irrelevant to the subsequent analysis and were removed using the Pandas drop() method, reducing the column count to 24.

Remove any duplicate rows

Duplicate rows were searched for and removed using the Pandas drop_duplicates() method. No duplicates were found. However, some rows are near duplicates, such as the first two rows, where the only difference is the price. Without metadata to consult, I left both these rows in the dataset, so as not to lose information unnecessarily. An alternative approach would be to replace the two rows with a single row and set the price to the mean value.

Remove rows with missing data

Missing values were searched for using Pandas isnull().sum() method. No missing (ie null) values were found in the dataset.

Inspection of the unique values in each column using the Pandas unique() method found ‘?’ appears in the data to indicate missing information. Such entries are not recognised by isnull(). There are 18 instances of ‘?’ in 14 unique rows (Table 4): 7% of the 205 rows in the dataset are affected. Of the affected columns, only price is a key variable in the subsequent analysis. To preserve as many rows as possible the 4 rows with ‘?’ in the price column were dropped using Pandas drop.() method.

Table 4 Columns with missing values indicated by '?' and actions taken to address them

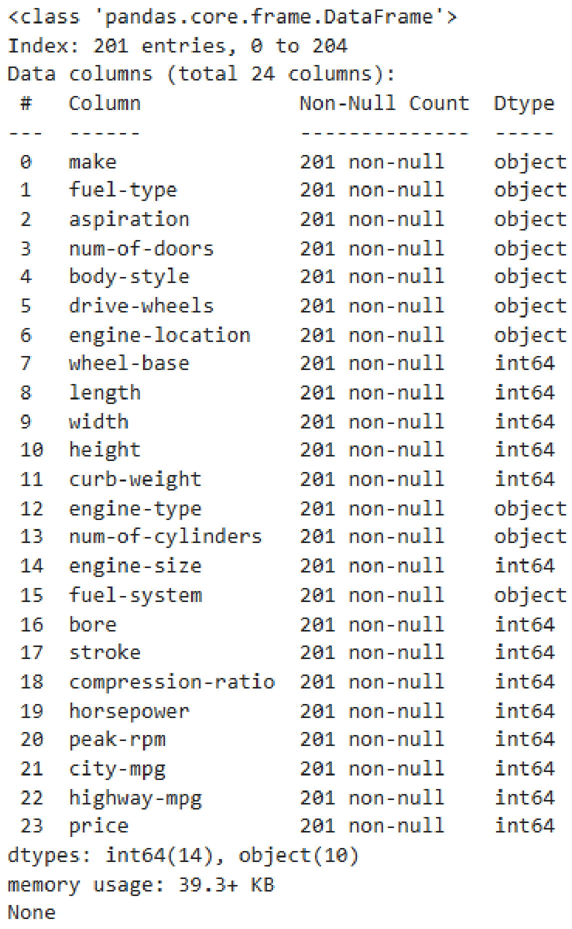
Column	Rows with ‘?’	Affected row indices	Action
num-of-doors	2	[27, 63]	Categorical data: substitute with value from previous row
bore	4	[55, 56, 57, 58]	Numerical data: substitute with mode value, 3.62
stroke	4	[55, 56, 57, 58]	Numerical data: substitute with mode value, 3.40
horsepower	2	[130,131]	Numerical data: substitute with mode value, 68
peak-rpm	2	[130,131]	Numerical data: no clear mode, two values share the maximum frequency, substitute the mean of those two values, 5150
price	4	[9, 44, 45, 129]	Remove using drop()

Convert all numerical data to int64

The columns listed in Table 4 were interpreted by Pandas as objects due to the presence of the ‘?’ entries. The data in these columns was cast from the string data type to the float64 datatype using the numpy astype(‘float64’) method. The columns containing floats were listed using Pandas .select_dtypes(include=[np.float64]) and cast to float64 using numpy astype(‘int64’) method.

All the entries in the dataset are now present as either object for categorical data or int64 for numerical data.

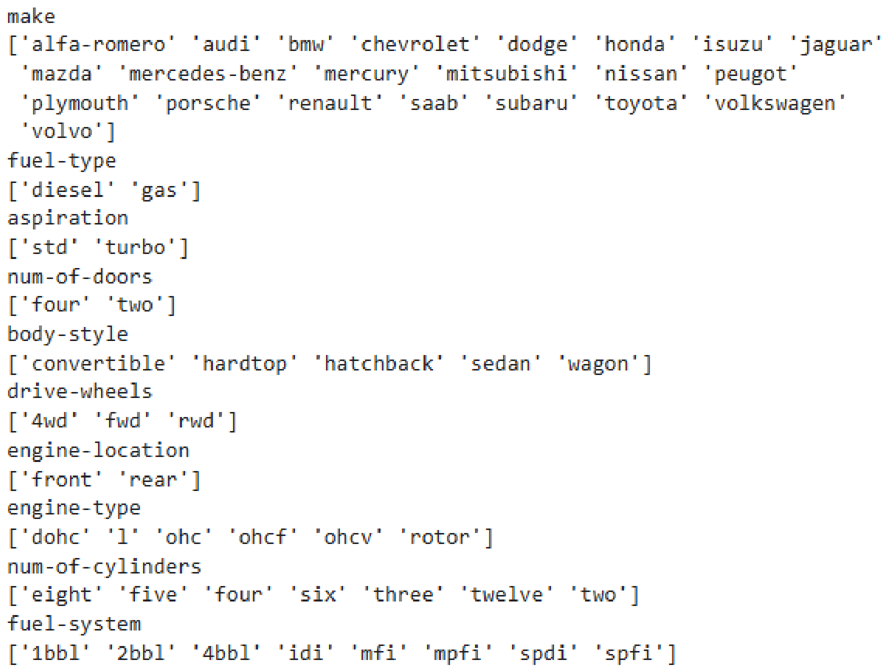
The cleaned dataset has 201 rows and 24 columns (Figure 15).



```
<class 'pandas.core.frame.DataFrame'>
Index: 201 entries, 0 to 204
Data columns (total 24 columns):
 #   Column             Non-Null Count  Dtype 
---  ------             --------------  ----- 
 0   make               201 non-null    object
 1   fuel-type          201 non-null    object
 2   aspiration         201 non-null    object
 3   num-of-doors       201 non-null    object
 4   body-style         201 non-null    object
 5   drive-wheels       201 non-null    object
 6   engine-location    201 non-null    object
 7   wheel-base         201 non-null    int64 
 8   length             201 non-null    int64 
 9   width              201 non-null    int64 
 10  height             201 non-null    int64 
 11  curb-weight        201 non-null    int64 
 12  engine-type        201 non-null    object
 13  num-of-cylinders   201 non-null    object
 14  engine-size        201 non-null    int64 
 15  fuel-system        201 non-null    object
 16  bore               201 non-null    int64 
 17  stroke             201 non-null    int64 
 18  compression-ratio  201 non-null    int64 
 19  horsepower         201 non-null    int64 
 20  peak-rpm           201 non-null    int64 
 21  city-mpg           201 non-null    int64 
 22  highway-mpg        201 non-null    int64 
 23  price              201 non-null    int64 
dtypes: int64(14), object(10)
memory usage: 39.3+ KB
None
```

Figure 15 List of column headings and data types

The unique entries in the categorical columns are shown in Figure 16.



```
make
['alfa-romero' 'audi' 'bmw' 'chevrolet' 'dodge' 'honda' 'isuzu' 'jaguar'
 'mazda' 'mercedes-benz' 'mercury' 'mitsubishi' 'nissan' 'peugot'
 'plymouth' 'porsche' 'renault' 'saab' 'subaru' 'toyota' 'volkswagen'
 'volvo']
fuel-type
['diesel' 'gas']
aspiration
['std' 'turbo']
num-of-doors
['four' 'two']
body-style
['convertible' 'hardtop' 'hatchback' 'sedan' 'wagon']
drive-wheels
['4wd' 'fwd' 'rwd']
engine-location
['front' 'rear']
engine-type
['dohc' 'l' 'ohc' 'ohcf' 'ohcv' 'rotor']
num-of-cylinders
['eight' 'five' 'four' 'six' 'three' 'twelve' 'two']
fuel-system
['1bbl' '2bbl' '4bbl' 'idi' 'mfi' 'mpfi' 'spdi' 'spfi']
```

Figure 16 The unique entries in each of the categorical data columns under each column heading.

Descriptive statistics for the dataset are shown in Table 5 Descriptive statistics for numerical data columns (1 of 2).Table 5 and Table 6.

Table 5 Descriptive statistics for numerical data columns (1 of 2).

Statistic	wheel-base	length	width	height	curb-weight	engine-size	bore
mean	98.3	173.7	65.4	53.3	2555.7	126.9	2.9
standard deviation	6.1	12.2	2.1	2.5	517.3	41.5	0.3
min	86.0	141.0	60.0	47.0	1488.0	61.0	2.0
25%	94.0	166.0	64.0	52.0	2169.0	98.0	3.0
50%	97.0	173.0	65.0	54.0	2414.0	120.0	3.0
75%	102.0	183.0	66.0	55.0	2926.0	141.0	3.0
max	120.0	208.0	72.0	59.0	4066.0	326.0	3.0

Table 6 Descriptive statistics for numerical data columns (2 of 2).

Statistic	stroke	compression-ratio	horsepower	peak-rpm	city-mpg	highway-mpg	price
mean	2.9	9.9	103.0	5117.9	25.2	30.7	13207.1
standard deviation	0.3	4.0	37.5	478.1	6.4	6.8	7947.1
min	2.0	7.0	48.0	4150.0	13.0	16.0	5118.0
25%	3.0	8.0	70.0	4800.0	19.0	25.0	7775.0
50%	3.0	9.0	95.0	5150.0	24.0	30.0	10295.0
75%	3.0	9.0	116.0	5500.0	30.0	34.0	16500.0
max	4.0	23.0	262.0	6600.0	49.0	54.0	45400.0

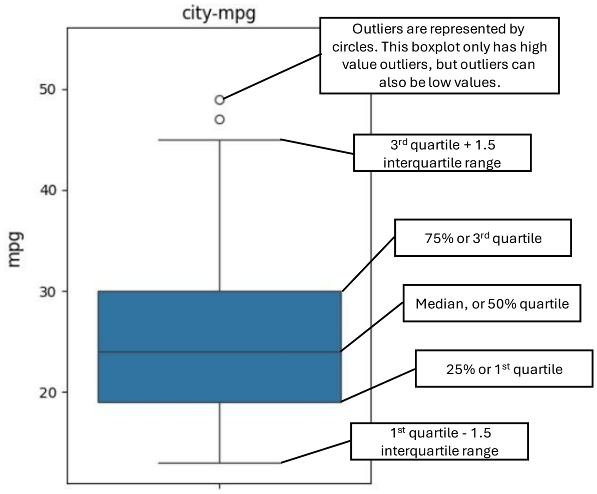
Appendix B: Methods and explanatory notes

Data preparation, cleaning, analysis and visualization was performed in a Jupyter Notebook using Python with numpy, pandas, matplotlib.pyplot, scipy and seaborn libraries.

The Jupyter Notebook is available on github.

Boxplots

Boxplots illustrate the distribution of values for a variable, including the quartiles and outliers[3]. Boxplots are drawn according to defined rules and conventions, so they can be easily read and compared.


city-mpg
mpg
50
40
30
20
Outliers are represented by circles. This boxplot only has high value outliers, but outliers can also be low values.
3rd quartile + 1.5 interquartile range
75% or 3rd quartile
Median, or 50% quartile
25% or 1st quartile
1st quartile - 1.5 interquartile range

Correlation

The Pearson correlation coefficient[4] takes values between +1 and -1, indicating the correlation between two variables[5]. A correlation coefficient of +1 indicates a strong positive relationship between the variables such that as either variable increases, the other increases, for example y = x. A correlation coefficient of -1 indicates a strong negative relationship between the variables such that as one variable increases, the other decreases, for example y = -x. A correlation coefficient of 0 is a null correlation, where there is no discernible relationship between the variables.

A correlation matrix contains the respective correlation coefficients between all possible pairs of variables in a data set. The correlation matrix can be shown graphically as a heatmap, traditionally with a scale from blue -1 through white 0 to red +1, where the color of each cell in the heatmap is derived from the correlation coefficient at that position in the correlation matrix.

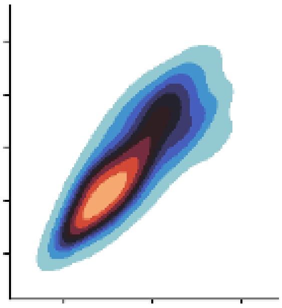
[3] For more about how seaborn defines a boxplot see https://seaborn.pydata.org/tutorial/categorical.html#categorical-tutorial . For the relationship between the boxplot and the frequency distribution see https://statisticsbyjim.com/graphs/box-plot/.

[4] For mathematical definition and formula see https://mathworld.wolfram.com/CorrelationCoefficient.html

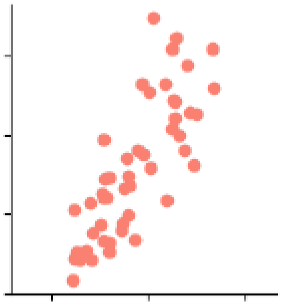
[5] For a guide to interpreting the correlation coefficient see https://www.scribbr.com/statistics/pearson-correlation-coefficient/

KDE plot

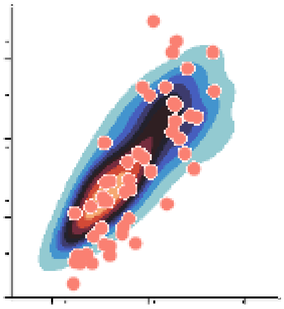
The KDE estimates the shape of a continuous distribution from the datapoints. The KDE has a predictive element to it. A KDE plot is a map of the probability density function[6]. Comparing the KDE plot[7] with a scatterplot highlights areas where concentrations of data points in the scatterplot overlap, potentially obscuring the true concentration. For example, compare this KDE plot:



With the corresponding scatterplot:



The KDE plot reveals a peak (ie in the probability density function) in the marked region, where the scatterplot shows some overlapping points. The shape of both plots is similarly positively correlated, but this is clearer in the KDE plot, as shown if the two plots are overlaid.



Further the KDE plot allows predictions to be made from the data.

[6] Details of the kernel density estimation including formulae https://en.wikipedia.org/wiki/Kernel_density_estimation

[7] For information about interpretation of KDE plots see https://mathisonian.github.io/kde/