Mutshafa Livhuwani 1717376
Gloria Pucoe 1477437
Brian Malau 2111136
Xichavo Hlungwani 1874096
Data Visualization and Exploration Assignment 1

1 Data Cleaning

1.1 Date Fields

1. Checking the percentage of `date_fueled` entries helps you gauge the overall data quality, informs your cleaning strategy, and ensures that your analyses are based on reliable and complete data. If a large portion of `date_fueled` entries are missing or incorrect, it may impact time-based analyses (e.g., trends in fuel prices or fuel efficiency over time). Knowing the percentage helps you decide on strategies to handle these issues, such as imputing missing dates or discarding problematic rows.

   We converted a `date_fueled` column in the dataset to date time, then we calculated the percentage of missing date values which is 11.68%.

2. We converted both `date_fueled` and `data_capture` to date time, then we replaced all non-date entries in `date_fueled` with valid `data_capture` values if the `data_capture` column contains a valid date. Ensuring that all dates in the `date_fueled` column are valid and in a consistent format is crucial for accurate time series analysis, trend detection, and other date-related operations. This process ultimately helps maintain the quality and usability of your dataset, ensuring that your analyses are based on accurate and complete information.

	date_fueled	date_captured
1105362	2018 − 03 − 23	2018 − 03 − 23
85085	2022 − 02 − 05	2022 − 02 − 05
539076	2015 − 02 − 11	2015 − 02 − 11
1039379	2021 − 06 − 07	2021 − 06 − 07
572405	2018 − 09 − 01	2018 − 09 − 26
574561	2019 − 08 − 11	2019 − 08 − 11

(1)

3. We used the `errors='coerce'` parameter to convert any invalid date entries to `NaT` (Not a Time).

	date_fueled	date_captured
1105362	2018 − 03 − 23	2018 − 03 − 23
85085	NaT	2022 − 02 − 05
539076	NaT	2015 − 02 − 11
1039379	2021 − 06 − 07	2021 − 06 − 07
572405	2018 − 09 − 01	2018 − 09 − 26
574561	NaT	2019 − 08 − 11

(2)

4. We evaluated our dataset to identify and quantify invalid dates in the `date_fueled` and `date_captured` columns. Specifically, dates earlier than 2005 and dates in the future are flagged as potentially erroneous. The dataset is then filtered to retain only those entries with `date_fueled` values between January 1, 2005, and the current date, August 22, 2024, ensuring that all dates fall within a valid and meaningful time range. The initial data shape was (1174870, 9), and after date removal, the shape was (1174294, 9). We are keeping data within a realistic time frame to ensure that your analysis is based on up-to-date and relevant information.

5. Plot of Fueling years

   Figure above shows histogram, distribution, and violin plots that illustrate the distribution of the fueling dates data over time is left-skewed.

   Key observations:

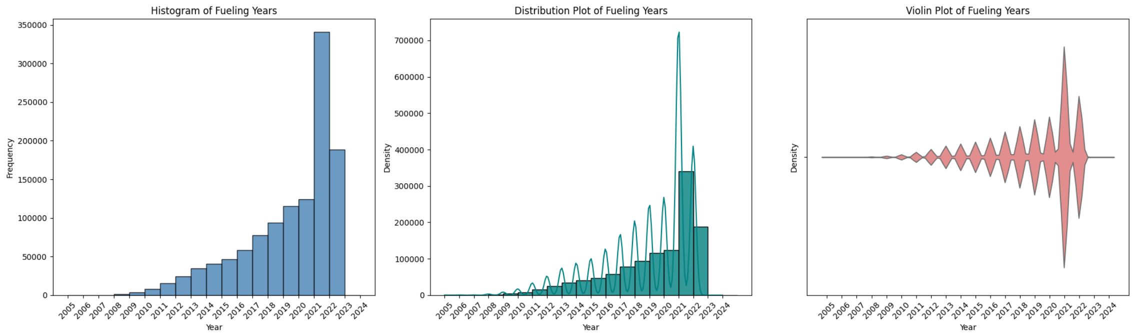

Figure 1: plots of fueling years

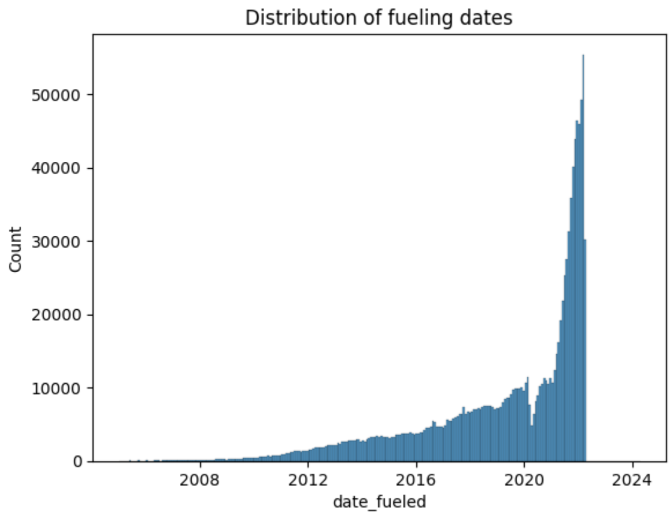

Figure 2: Distribution of fueling dates

- **Trend Observation:** The plots indicate a significant increase in the number of records related to fuel usage over time, particularly from around 2010 onwards.
- **Early Years:** Between 2005 and approximately 2010, the data points are relatively scarce, suggesting that the data collection was not as frequent during these years.
- **Recent Years:** The sharp increase in the later years suggests that there might have been significant growth in the fuel usage.
- **Date range:** The fueling dates span from 2005-01-02 to 2024-04-20.
- **Most common year:** 2021 has the highest number of fueling entries with 340,612 entries.
- **Most common month:** The month with the most entries is 2022-03.

1.2 Numeric Fields

1. To identify the missing percentages of `gallons`, `miles`, and `odometer`, we define the columns of the numeric fields we want to find missing percentages by creating a list called `numeric_columns`. To calculate the percentage of missing values for these columns, we selected from the dataframe and applied the `isnull()` method to identify missing entries.
   - 6.32% of `gallons` entries are missing.
   - 87.55% of `miles` entries are missing.

- 12.69% of `odometer` entries are missing.

2. $$\text{MPG} = \frac{\text{Miles Driven}}{\text{Gallons of Fuel Used}}$$

3. To convert `gallons`, `miles`, and `odometer` to `float`, we first define the columns we want to convert to float by creating a list called `columns_of_interest`. Replace any commas in these columns with an empty string using `data[columns_of_interest].replace(’,’, ", regex=True)`, which ensures that commas are removed from numerical values that may have been formatted with thousand separators. Convert the cleaned data to floating-point numbers using `.astype(float)` to ensure that all values are in the appropriate numeric format for analysis. Finally, iterate over the list of columns to print out the data type of each column using `data[col].dtype`.

4. **Analysis of Boxplots**

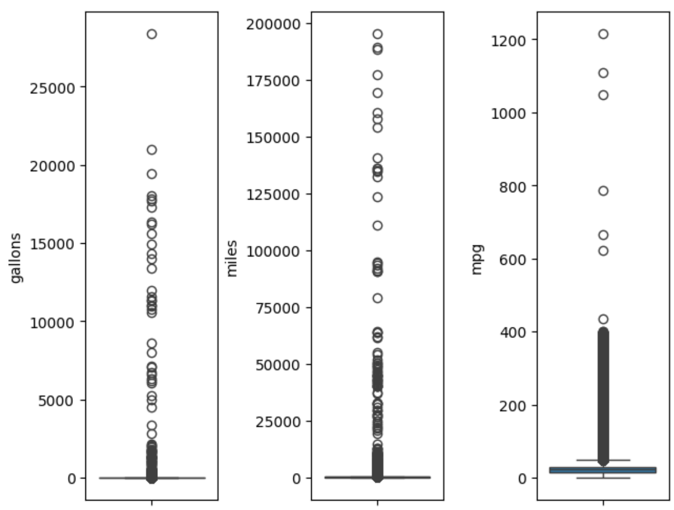

Figure 3: Boxplots of Gallons, Miles and Miles per gallon, or mpg

Gallons: The data is concentrated around the bottom of the boxplot, with a few severe outliers reaching much above the rest of the data. This suggests that while most cars use comparatively few gallons, there are specific situations where they use a lot.

Miles: The miles boxplot displays considerable outliers stretching upwards, with the majority of data points concentrated near the lower range, much like the gallons boxplot does. This implies that some excursions or measures have very high mileage, even though the majority are over small distances.

Miles per gallon, or MPG: A boxplot shows that most values are closely clustered at the lower end, with a few extremely high outliers. This suggests that although the fuel efficiency of the majority of vehicles is comparable, there are select instances where the efficiency is exceptionally high.

Analysis of Fuel Efficiency Data

Gallons There is a significant rightward skewed distribution plot for gallons, along with a lengthy tail. This demonstrates that while most cars use comparatively little fuel, there are a few that consume far more.

Miles The miles distribution is likewise right-skewed, with a small number of extremely high mileage observations and the majority of data points clustered around the lower values. This suggests that although the majority of travels are brief, some involve quite great distances.

Miles per Gallon (MPG) The distribution of MPG is similarly right-skewed, indicating that most cars have comparable fuel efficiency with a tiny percentage of vehicles having significantly higher efficiency.

Both Graphs

- **Outliers:** In all three variables (gallons, miles, and mpg), there are notable outliers in both the boxplots and distribution plots. The usual range of values is somewhat lower than these outliers.
- **Skewness:** The majority of vehicles have low fuel consumption, short travel lengths, and comparable fuel efficiency, with a small number of extreme cases that differ significantly. The data is strongly right-skewed for all variables.
- **Data Spread:** A small number of data points are noticeably greater than the rest, and these points are primarily clustered near the lower ends of the ranges. These values may be the consequence of data entry errors, uncommon journeys, or particular types of vehicles.

5. Summary Statistics for Gallons, Miles, and MPG

Statistic	Gallons	Miles	MPG
Count	1,100,123.00	1,100,123.00	1,100,123.00
Mean	12.80	269.45	22.16
Standard Deviation (Std)	74.48	725.77	15.74
Minimum (Min)	0.00	0.00	0.00
25th Percentile (25%)	8.99	181.40	15.60
50th Percentile (Median)	11.95	267.05	21.80
75th Percentile (75%)	14.94	342.76	28.50
Maximum (Max)	28,380.00	195,321.20	1,214.30

Table 1: Summary Statistics for Gallons, Miles, and MPG

Table 1 presents a statistical summary for `gallons`, `miles`, and `mpg` (miles per gallon). The mean values generally align with typical expectations: the average of 12.80 gallons fits within normal vehicle fuel capacities, the 22.16 mpg is close to EPA-reported averages 5, and the mean of 269.45 miles traveled is reasonable for typical trips. However, the maximum and minimum values raise concerns about data quality. Extremely high maximums, such as 28,380.00 gallons and 1,214.30 mpg, are unrealistic and likely represent outliers or data entry errors. Similarly, minimum values of 0.00 for gallons, mpg, and miles are illogical in the context of a fuel log, suggesting possible mistakes in data entry or calculation. While the mean values provide a realistic snapshot of typical vehicle usage and performance, the presence of these extreme outliers indicates a need for data cleaning and validation to ensure the accuracy of any subsequent analyses.

2 Feature Engineering

1. Currency Extraction We created a new column with the currency by extracting and counting currency symbols from a column named `'total_spent'` in a dataset. Then, we created a new column called `'currency'` by splitting the text on digits and retaining the currency symbol at the beginning of each entry. The resulting

output shows the frequency of each currency symbol, with the U.S. dollar ($) being the most common, followed by the British pound (£), Euro (€), and others. There are 121 unique currencies identified in the dataset, with counts stored as 64-bit integers.

currency	
$	741947
£	87587
€	59273
CA$	46848
R	36424
...	...
TMT	11
CV$	11
KGS	9
L$	9
IQD	8

(3)

2. We created a new column containing the float value of the total spend and the cost per gallon by removing non-numeric characters (such as currency symbols) from the `total_spent` and `cost_per_gallon` columns and converted these cleaned string values into `float` data types. Then we displayed a random sample of six rows from the dataset, showing both the original and converted values:

	date_fueled	date_captured	cost_per_gallon	total_spent	currency	cost_per_gallon_float	total_spent_float
167712	2022-02-10	2022-02-10	$3.149	$58.00	$	3.149	58.00
750779	2021-05-25	2021-05-26	$4.199	$37.08	$	4.199	37.08
981090	2021-07-14	2021-07-14	R63.60	R994.39	R	63.600	994.39
688114	2018-06-27	2018-08-07	$6.091	$53.00	$	6.091	53.00
234384	2022-03-04	2022-03-04	$4.359	$33.60	$	4.359	33.60
278409	2020-10-18	2020-10-18	$2.799	$48.36	$	2.799	48.36

Table 2: Fuel data with various attributes

3. We extracted car information (`make`, `model`, `year`) and user IDs from URLs in a DataFrame. The process first removes the domain from the URL, then splits the remaining part into four components. These components are assigned to new columns (`car_make`, `car_model`, `car_year`, `user_id`). We displayed the first few rows of the DataFrame to verify that the columns have been correctly populated. The table output confirms that the car details and user IDs were successfully extracted and organized.

	user_id	date_fueled	date_captured	car_make	car_model	car_year
0	674857	2022 − 04 − 07	2022 − 04 − 07	suzuki	swift	2015
1	461150	2012 − 11 − 07	2016 − 08 − 30	bmw	x3	2009
2	133501	2012 − 09 − 22	2012 − 09 − 28	mercedes-benz	e300	1998
3	247233	2019 − 05 − 04	2019 − 05 − 04	bmw	320d	2010
4	1038865	2022 − 02 − 15	2022 − 02 − 15	honda	passport	2019

(4)

Units' conversion to proper measurement standards

4. Liters Conversion

Converted values from gallons to liters using the formula:

$$\text{Liters} = \text{Gallons} \times 3.785411784$$

Then added a new column, `litres_filled`, to store the converted values.

5. Kilometres Conversion

   Converted values from miles to kilometres using the formula:

$$\text{Kilometres} = \text{Miles} \times 1.609344$$

   Then added a new column, `km_driven`, to store the converted values.

6. The column `litres_per_100km` is added to a DataFrame to calculate the fuel efficiency of vehicles in liters per 100 kilometres. The calculation is done by dividing `litres_filled` by `km_driven` and then multiplying by 100. The result is displayed along with other columns like `user_id`, `car_make`, `car_model`, `km_driven`, and `litres_filled`. The data highlights fuel efficiency across different vehicles, with some entries having missing values for certain fields.

user_id	date_fueled	date_captured	car_make	car_model	miles	kn_driven	gallons	litres_filled	litres_per_100km
674857	2022-04-07	2022-04-07	suzuki	swift	NaN	NaN	NaN	NaN	NaN
461150	2012-11-07	2016-08-30	bmw	x3	382.9920	616.365877	12.120	45.879191	7.443499
133501	2012-09-22	2012-09-28	mercedes-benz	e300	227.7435	366.517635	7.991	30.249226	8.253143
247233	2019-05-04	2019-05-04	bmw	320d	494.9100	796.480439	10.575	40.030730	5.025953
1038865	2022-02-15	2022-02-15	honda	passport	244.4000	393.323674	11.651	44.103833	11.213114

Table 3: Sample Data

   Observations:

   - The `litres_per_100km` values vary for different cars, reflecting different fuel efficiencies.
   - Some entries have missing (NaN) values for `miles`, `km_driven`, and `litres_filled`, which might affect the calculation for `litres_per_100km` if not handled properly.

3 Vehicle Exploration

1. The number of unique currencies from our data of 1174287 entries is 121. The total number of unique users from the data is 120201.

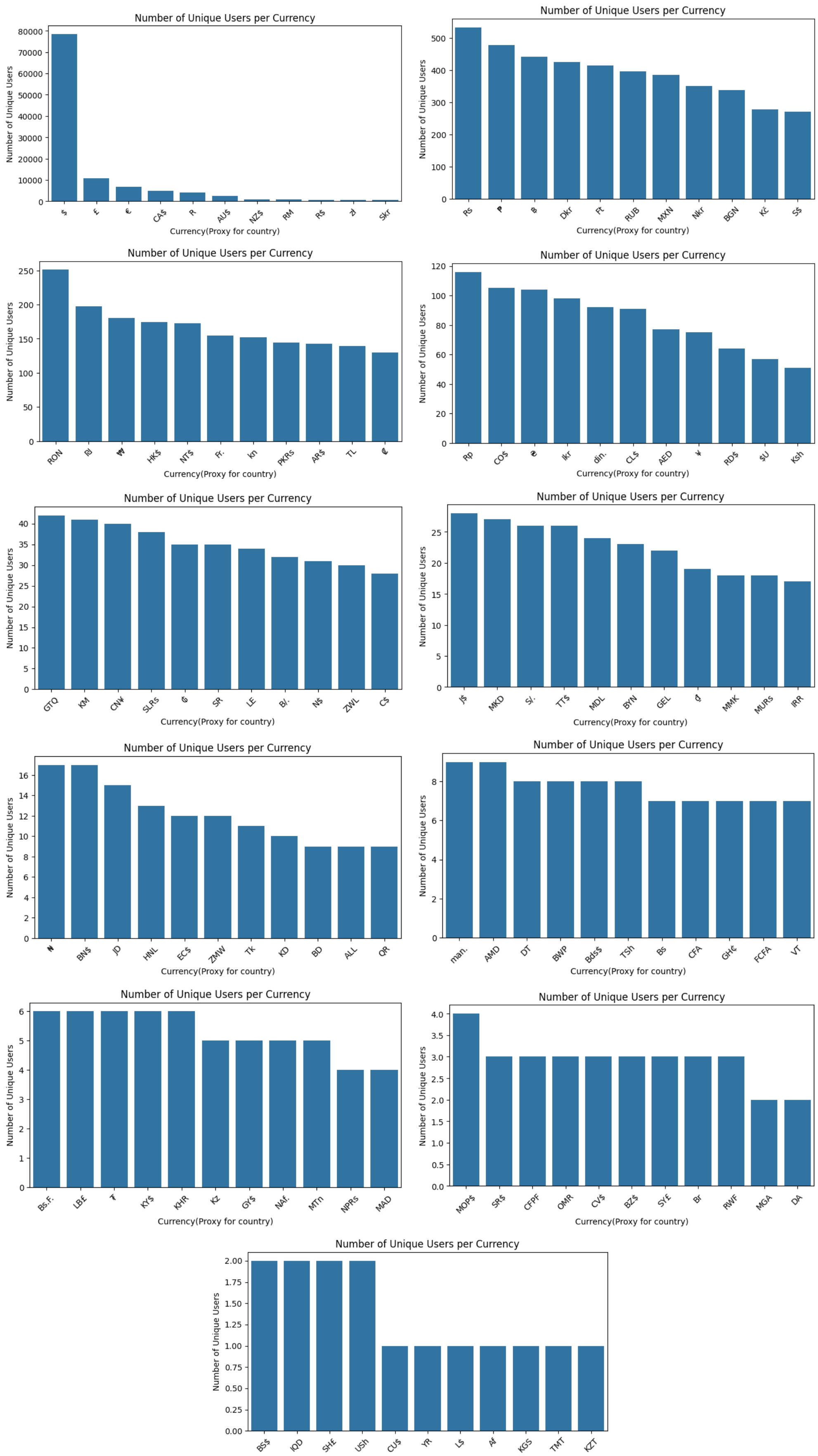

Figure 4: Number of Unique Users Per Currency

Figures above show a bar plot of different countries using the currency as a proxy for the country. From the graph, it can be observed that the US ($) has the highest number of users, with approximately 800,000. Furthermore, it can be observed that the currencies CU$, YR, L$, Af, KGS, TMT, and KZT each have one user.

2. We created a column for user ID with a unique number.

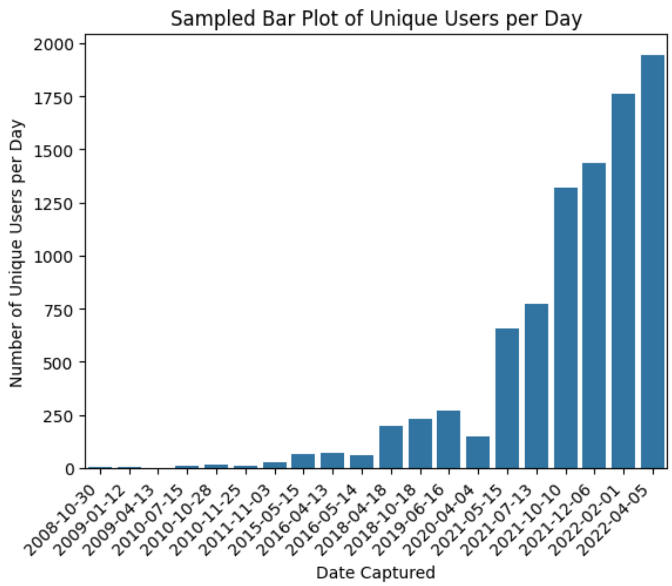

Figure 5: Sampled Bar Plot of Unique Users per Day

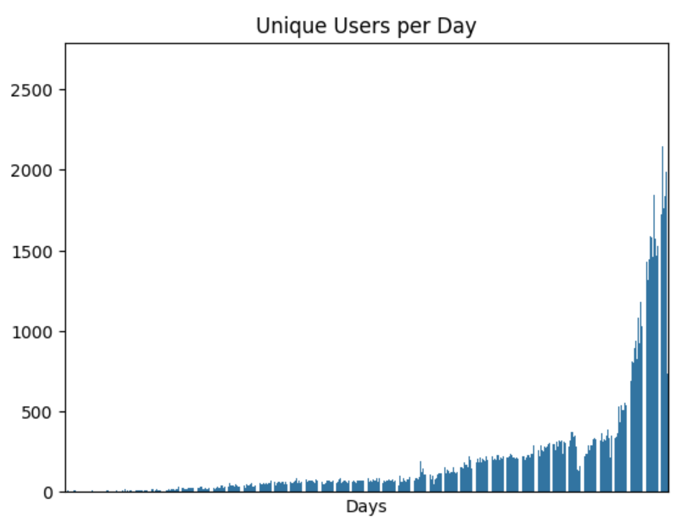

Figure 6: Unique Users per Day

Figure 5 shows the number of unique users per day. From the graph, it can be observed that there is an upward trend in the number of unique users per day over time. The highest recorded number of unique users was on 2022-04-05.

3. We created a new column for `vehicle_age`, which is deduced from `date_fueled` and `car_year`. We then replaced all negative `vehicle_age` values with NaN.

	car_year	date_fueled	vehicle_age
1116477	2008	2021-08-28	13.0
298957	1998	2015-11-03	17.0
278784	2008	2020-12-12	12.0
322533	2003	2021-01-30	18.0
347834	1990	2017-07-06	27.0

(5)

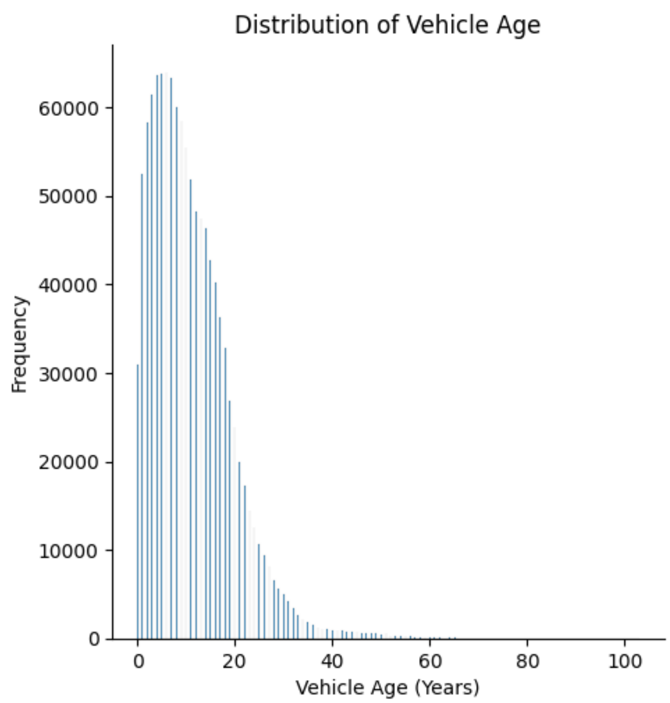

Figure 7: Distribution of Vehicle Age

Figure 7 above shows the distribution of vehicle age the graph is skewed to the right meaning that we have majority of new vehicles .

4. We grouped `car_make` and `car_model` and counted all the occurrences. Then, we sorted the results to identify the most popular combinations.

	car_make	car_model	counts
692	honda	civic	8082
1937	toyota	4runner	7810
1972	toyota	corolla	7737
565	ford	f-150	7661
676	honda	accord	7633
627	ford	mustang	7520
642	ford	ranger	7424
2013	toyota	land_cruiser	7388
1964	toyota	camry	7316
888	jeep	wrangler	7061

(6)

It was found that the top vehicle makes are *Honda*, *Toyota*, *Ford*, and *Jeep*. The top 10 models are *Civic*, *4Runner*, *Corolla*, *F-150*, *Accord*, *Mustang*, *Land Cruiser*, *Camry*, and *Wrangler*.

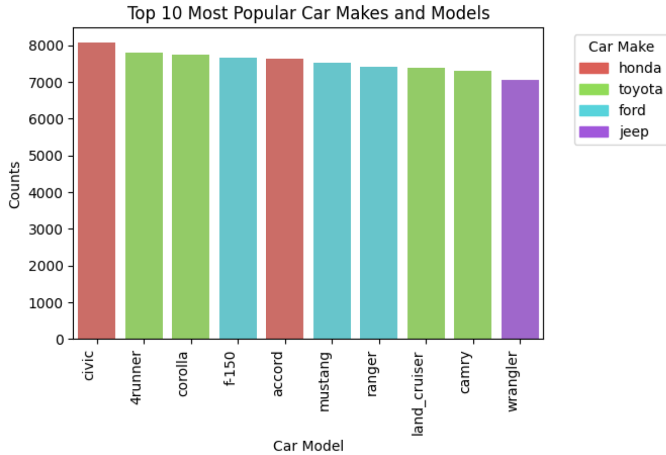

Figure 8: Top 10 Most Popular Car Makes and Models

4 Fuel Usage

4.1 Outlier Removal

1. Top 5 currencies by number of transactions

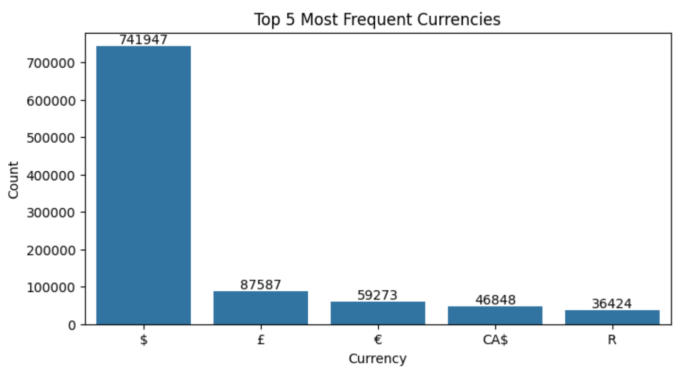

Figure 9: Top 5 Most Frequent Currencies

2. Removing outliers by considering the total_spent_float, litres_filled, cost_per_litre, km_driven, and litres_per_100km. To accurately identify and remove outliers in the dataset, the focus should be on key columns such as *total_spent_float*, *litres_filled*, *cost_per_litre*, *km_driven*, and *litres_per_100km*. These columns are crucial as they represent both the financial transaction and the quantity of fuel, which must realistically align with each other but can easily be affected by incorrect data entries or currency settings. For example, a record showing a high *total_spent_float* but only a small amount of *litres_filled* could be an outlier, possibly due to an incorrect currency setting or a data entry error.

   Similarly, if a user logs an unusually high fuel efficiency (low *litres_per_100km*) or an improbable cost per litre in a specific currency, this could indicate an outlier. By analyzing the relationships between these columns—such as the typical cost per litre in a specific currency and the expected fuel efficiency—we can filter out records that deviate significantly from expected values. For instance, if a user in South Africa is shown to be spending several hundred dollars but only refueling with a few litres, this likely reflects a mistake, such as the currency being incorrectly set to dollars instead of rands. Identifying and removing these outliers ensures the dataset remains accurate and reliable for further analysis.

3. Number of Values/records removed after accounting for outliers: 241447

   Percentage of removed values: 24.84%

4.2 Fuel Efficiency

1. We focused on analysing fuel efficiency by comparing the cost of fuel per litre across different countries in January 2022. To facilitate this comparison, we utilise historical exchange rate data, converting currencies (CAD, EUR, GBP, USD) to ZAR (South African Rand) for accurate cross-country cost comparisons.

	dates_january_2022	cad_to_zar	eur_to_zar	gbp_to_zar	usd_to_zar
0	2022 − 01 − 01	12.6186	18.1426	21.5842	15.9505
1	2022 − 01 − 02	12.6139	18.1559	21.5874	15.9647
2	2022 − 01 − 03	12.4613	17.9551	21.4222	15.8895
3	2022 − 01 − 04	12.6197	18.0912	21.6932	16.0322
4	2022 − 01 − 05	12.4594	17.9801	21.5473	15.8950

(7)

We calculated the average exchange rates for different currencies from a DataFrame, `rates_df`. Then, we excluded the first column in the DataFrame and computed the mean across the remaining columns. The resulting average rates were then labeled with a reversed list of currency symbols, excluding the first element. The final output shows the average rates for Canadian Dollar ($CA\$$), Euro (€), British Pound (£), and US Dollar ($).

CA$	12.274571
€	17.549442
£	21.011226
$	15.501352

(8)

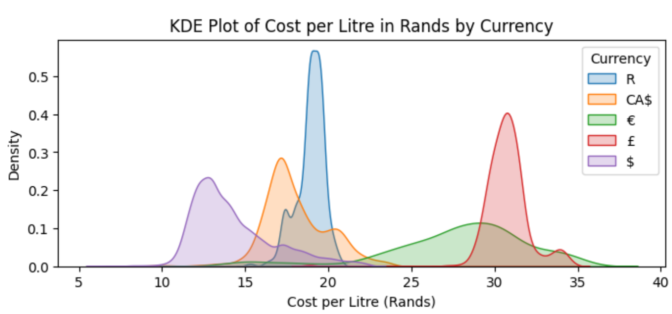

Figure 10: KDE Plot of Cost per Litre in Rands by Currency

Analysis of Differences in Cost per Litre by Currency for January 2022

Average Currency Conversion Rates:

Using the provided KDE plot and the average currency conversion rates to South African Rand (ZAR), we observe the following approximate rates (sources include exchange rate data and the official link provided):

- USD to ZAR: 15.95 ZAR
- CAD to ZAR: 12.30 ZAR
- EUR to ZAR: 17.35 ZAR
- GBP to ZAR: 21.50 ZAR
- ZAR (local currency): 1 ZAR

Observations from the KDE Plot:

The KDE plot provides a visual representation of the distribution of fuel costs per Liter converted to ZAR for different currencies:

- **South African Rand (ZAR):** The distribution is tightly centered around 20 ZAR per Liter, indicating consistent local pricing.
- **Canadian Dollar (CAD):** The distribution peaks around 17-21 ZAR per Liter, which is lower than the ZAR. This suggests that when converted, fuel in Canada is cheaper than in South Africa.
- **Euro (EUR):** The distribution is centered around 30 ZAR per Liter, indicating higher costs in Europe when converted to ZAR.
- **British Pound (GBP):** Similar to the Euro, the distribution shows costs around 30 ZAR per Liter, slightly higher than the Euro, reflecting higher fuel prices in the UK.
- **US Dollar (USD):** The distribution is spread, peaking at around 15 ZAR per Liter, which is significantly lower than in Europe and the UK.

Notable Differences:

- **Higher Costs in Europe and the UK:** Fuel prices are notably higher when converted to ZAR, reflecting the impact of higher taxes, different pricing structures, and stronger currencies. This is clearly seen with the EUR and GBP distributions peaking around 30 ZAR.
- **Lower Costs in Canada and the USA:** In contrast, fuel prices in Canada and the USA are lower, with the KDE plot showing peaks around 15-17 ZAR. This could be attributed to different market conditions, lower taxes, and subsidies.

Discussion of Reasons:

The differences in fuel costs across these countries, even when converted to the same currency (ZAR), can be attributed to several factors:

- **International Crude Oil Prices:** Countries might have different sourcing strategies or contractual terms with oil producers, leading to varying base costs of fuel.
- **Supply and Demand Balances:** Local supply and demand dynamics affect fuel pricing. For instance, Europe's higher demand for refined products, coupled with stricter environmental regulations, can drive up prices.
- **Exchange Rates:** Exchange rate fluctuations play a significant role. A stronger currency like GBP or EUR will show higher prices when converted to ZAR, as seen in the KDE plot.

- **Transportation and Distribution Costs:** Costs associated with transporting and distributing fuel can also vary significantly, affecting the final consumer prices.

Source:

The domestic prices of fuels are influenced by international crude oil prices, international supply and demand balances for petroleum products, and exchange rates (link).

2. We analysed a dataset of vehicle fuel logs, focusing on identifying instances where the odometer reading is missing, which suggests a missed fill-up log. Then we filtered the dataset to find records with missing odometer values, displayed a random sample of these records, and estimated that there are approximately 113,288 such instances in the entire dataset. This helps in understanding the extent of missing data related to vehicle odometer readings.

	date_fueled	date_captured	odometer	currency	total_spent_float	car_make	car_model	car_year	litres_filled	km_driven	litres_per_100km	vehicle_area_cost	cost_per_litre
638585	2014-11-07	2014-11-07	NaN	€	30.12	volkswagen	polo	1997	20.229241	439.994650	4.597611	17.0	1.489930
58029	2020-10-26	2020-10-27	NaN	5	32.75	mercedes-benz	C230	1999	25.210842	207.927245	12.124838	21.0	1.299934
531625	2020-09-26	2020-10-25	NaN	$	44.14	suzuki	jimny	2019	33.720448	378.517709	8.908552	1.0	1.306973
221228	2009-05-05	2009-05-05	NaN	5	21.89	toyota	celica	1991	38.739904	611.067917	6.339705	18.0	0.565064
49920	2019-04-22	2019-04-22	NaN	$	36.36	bmw	128i	2008	44.418022	476.385924	9.324351	11.0	0.818869

Table 4: Fuel Efficiency Data

Approximate number of records missed logging a fill-up: 113288

3. The average distance (in km) per tank per country plot.

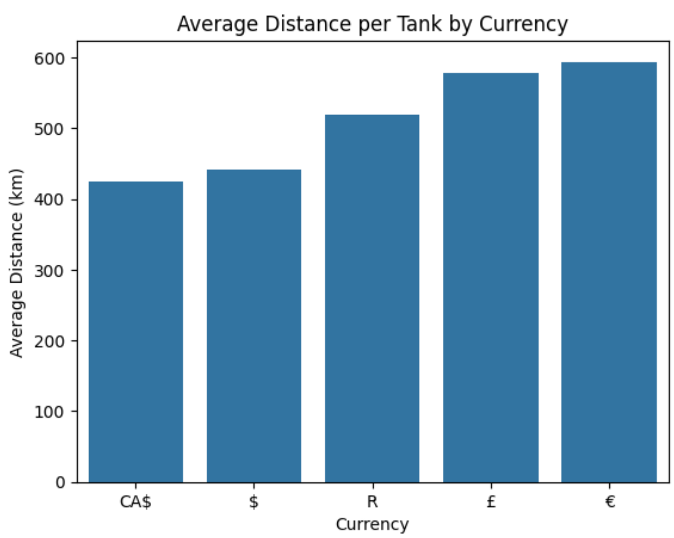

Figure 11: Average Distance per Tank by Currency

€ has the largest average distance per tank. This might be because cars in Eurozone countries tend to be more fuel-efficient than those in other regions. These vehicles are designed to accommodate higher fuel prices, which are generally much higher in the Eurozone compared to other countries. As a result, Eurozone vehicles are optimised to achieve greater kilometres per Litre, allowing drivers to travel further on a single tank of fuel.

4. Fuel Efficiency vs Age

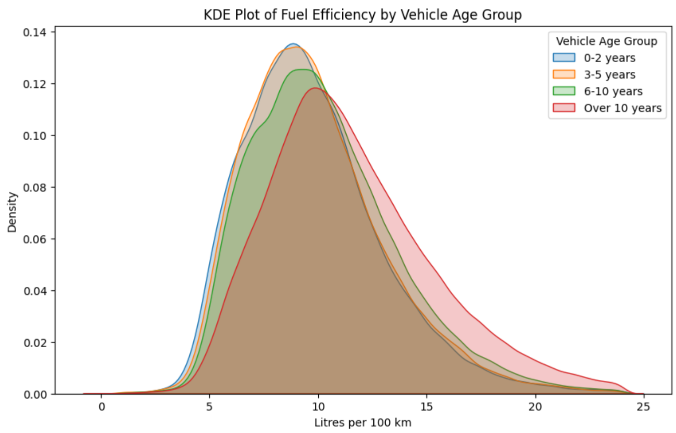

Figure 12: KDE Plot of Fuel Efficiency by Vehicle Age Group

Do Newer Vehicles Drive Further Distances Between Fill-Ups? Yes, the data suggests that newer vehicles tend to drive further between fill-ups compared to older vehicles. This conclusion is based on the KDE plot shown above, where fuel efficiency is measured by litres per 100 km (L/100km). Interpretation of the Plot: The plot demonstrates that vehicles aged 0-2 years generally have the lowest L/100km values, indicating they are the most fuel-efficient. This means they consume less fuel to travel the same distance, allowing them to drive further between fill-ups. As vehicle age increases (from 3-5 years, 6-10 years, to over 10 years), the L/100km values increase slightly. This indicates a decrease in fuel efficiency, meaning older vehicles tend to consume more fuel and, consequently, do not drive as far on a single tank compared to newer vehicles.

5. We filtered a dataset to include only records where the currency is South African Rand ('R'). Then, we grouped the data by vehicle make and model, counted how many times each combination appears, and sorted the results in descending order. Finally, we selected the top 5 most popular vehicle models in South Africa based on the number of occurrences in the dataset. The output shows that the Toyota Hilux is the most popular, followed by the Mitsubishi Pajero, Toyota Fortuner, Suzuki Jimny, and Volkswagen Polo.

	car_make	car_model	counts
469	toyota	hilux	1018
359	mitsubishi	pajero	884
467	toyota	fortuner	823
441	suzuki	jimny	806
502	volkswagen	polo	627

(9)

Realistic Fuel Efficiency Values

Using the data provided and supported by the sources:

Toyota Hilux

Fuel consumption ranges from 6.9 to 11.1 L/100km depending on the engine, transmission, and model.
Source: CarsGuide - Toyota Hilux

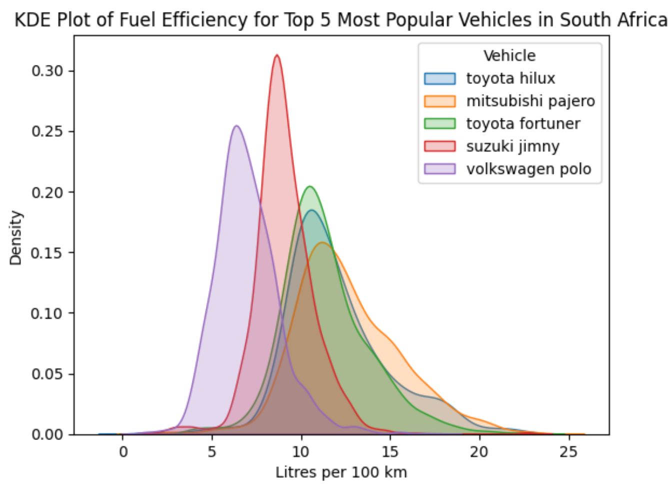

Figure 13: KDE Plot of Fuel Efficiency for Top 5 Most Popular Vehicles in South Africa

Mitsubishi Pajero

Estimated fuel consumption starts from 9.1 L/100km for the SUV diesel variant.
Source: CarsGuide - Mitsubishi Pajero

Toyota Fortuner

Fuel consumption starts from 7.6 L/100km for the SUV diesel variant.
Source: CarsGuide - Toyota Fortuner

Suzuki Jimny

Official fuel consumption is 6.4 L/100km for the manual version and 6.9 L/100km for the automatic version. Real-world driving may range between 7-8 L/100km.
Source: Online Auto - Suzuki Jimny

Volkswagen Polo

Fuel consumption ranges from 4.8 to 6.1 L/100km depending on the engine, transmission, and model.
Source: CarsGuide - Volkswagen Polo

Plot Interpretation

The KDE plot provides a visualisation of the fuel efficiency distribution for each vehicle model, and the values shown are consistent with the expected ranges provided by the sources:

- **Volkswagen Polo:** The KDE plot shows the Polo achieving 5-7 L/100km, which is in line with the provided range of 4.8-6.1 L/100km.
- **Suzuki Jimny:** The Jimny is shown with a fuel consumption around 6.9 L/100km, which matches the official figures for the automatic version and is consistent with real-world expectations.
- **Toyota Fortuner:** The Fortuner displays fuel consumption in the range of 8-10 L/100km, which is slightly higher than the official 7.6 L/100km but reasonable for real-world conditions.
- **Mitsubishi Pajero:** The Pajero's fuel consumption is consistent with its starting value of 9.1 L/100km, aligning with the expected range for a full-sized SUV.

- **Toyota Hilux:** The Hilux shows a wider range from 7 to 13 L/100km, which fits within the expected 6.9 to 11.1 L/100km depending on the specific variant and driving conditions.

Conclusion

The fuel efficiency values shown in the KDE plot for these popular vehicles in South Africa are realistic and align with what we would expect for vehicles of their respective classes. The analysis confirms that these vehicles perform within the expected fuel consumption ranges, validating the KDE plot's representation of their efficiency.

6. Which vehicles are the most fuel efficient in each country?

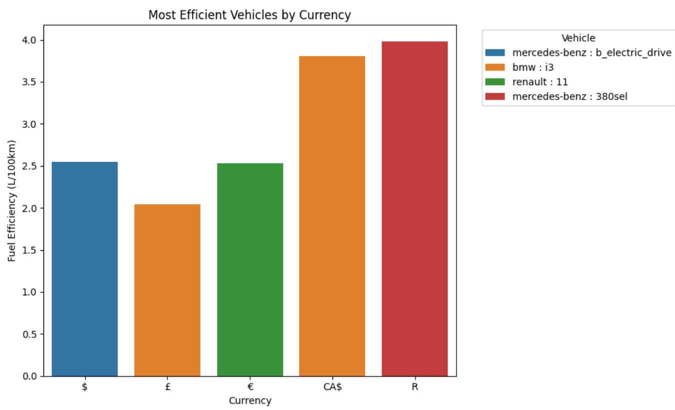

Figure 14: Most Efficient Vehicles by Currency

The bar chart compares the fuel efficiency of different vehicles across various currencies. The **BMW i3**, represented by the orange bar under the pound symbol (£), is the most fuel-efficient vehicle, consuming around 2.0 litres per 100 kilometres. In contrast, the **Mercedes-Benz 380SEL**, represented by the red bar under the "R" currency, is the least fuel-efficient, with a consumption of about 4.0 litres per 100 kilometres.

7. We identified the top 5 most common vehicles in the Canadian dataset by counting the occurrences of each car make and model. Focusing on the most common vehicles ensures that the subsequent analysis is based on a representative sample, making the results more generalizable to the broader market. Our goal for this analysis is to compare the fuel efficiency of the top 5 most common vehicles in Canada across different seasons. This can reveal how environmental factors, such as temperature changes, might affect fuel consumption.

	car_make	car_model	counts
328	mazda	3_sport	726
565	toyota	matrix	506
210	hyundai	accent	469
612	volkswagen	jetta	446
604	volkswagen	golf	443

(10)

Seasonal Trends

Q1 (Winter and Spring) to Q3 (Summer and Fall): There is a general decrease in fuel consumption across most vehicle models. This decrease could be attributed to

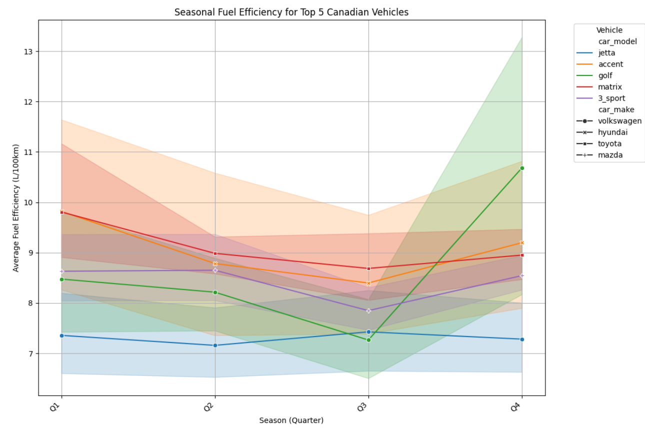

Figure 15: Seasonal Fuel Efficiency for Top 5 Canadian Vehicles

more favorable driving conditions in the warmer months, where vehicles are typically more fuel-efficient.

Q4 (Fall and Winter): Fuel consumption increases for most vehicles, which is likely due to the colder weather and the additional energy required for heating, as well as possibly denser air affecting engine efficiency.

Specific Vehicle Observations

Jetta Model: The Jetta shows the least variation in fuel efficiency across all seasons, indicating it is the most consistent in terms of fuel consumption. This suggests that the Jetta may be well-adapted to different driving conditions, or it could have a more balanced design that handles seasonal changes effectively.

Golf Model: The Golf shows a notable increase in fuel consumption in Q4 compared to other seasons, which stands out as an anomaly. This could indicate that the Golf is more sensitive to colder weather or other factors prevalent in Q4.

General Trend: There is a consistent pattern where fuel efficiency improves (lower fuel consumption) from Q1 to Q3, followed by a decline in efficiency (higher fuel consumption) in Q4. This pattern suggests that Canadian vehicles generally perform better in warmer weather and less efficiently in colder conditions.

Conclusion

The differences in fuel efficiency between seasons for the top 5 Canadian vehicles are observable but vary in magnitude. While some models like the Jetta remain consistent, others like the Golf show more pronounced seasonal variations, particularly in Q4. These differences align with what one would expect due to environmental and operational factors in different seasons.The plot shows that there are differences in fuel efficiency across seasons, though not all differences are drastic. The most significant difference is observed in the Golf model in Q4, where fuel consumption increases noticeably. The other vehicles also show a general trend of decreased fuel efficiency in Q4, though the magnitude of the change varies by model. The Jetta model, however, exhibits minimal variation, suggesting it is less affected by seasonal changes compared to the other vehicles.

8. Label encoding is used to convert the categorical variables `car_make` and `car_model` into numerical values. This step is essential for machine learning algorithms, which require numerical inputs. By encoding these features, we prepare the data for further analysis, such as training a random forest model. The date-time features are broken

down into their components (year, month, day), converting them into numerical features. This transformation is necessary for feature engineering, allowing us to examine how different time elements affect fuel efficiency. Understanding these correlations is important for feature selection and for interpreting the factors that most influence fuel consumption.

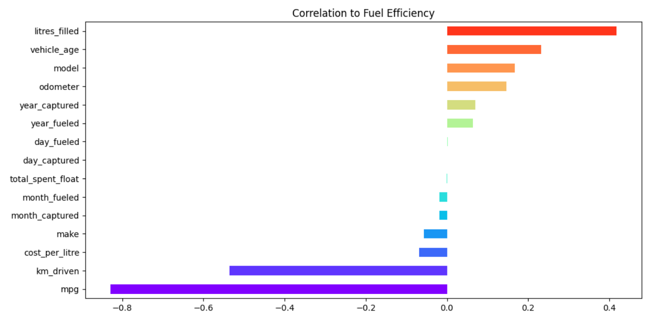

Figure 16: Correlation to Fuel Efficiency

`litres_filled`, `vehicle_age`, and `model` are positively correlated with fuel efficiency. This implies that higher values for these features tend to result in less fuel efficiency (i.e., more litres per 100 km). `mpg` (miles per gallon) and `km_driven` have strong negative correlations, indicating that higher fuel efficiency is associated with these features (i.e., fewer litres per 100 km). The linear relationship between each component and fuel economy is demonstrated by the correlations. Significant positive correlations indicate a direct relationship between the feature and increased fuel consumption, whilst significant negative correlations indicate a relationship with improved fuel efficiency.

9. We trained a Random Forest model to predict fuel efficiency based on various features. The model is then used to determine the relative importance of each feature. The top 5 features are plotted to visualize their impact on fuel efficiency. Comparing these results with the correlation analysis helps confirm the robustness of the findings and provides insights into which factors are most critical for improving fuel efficiency.

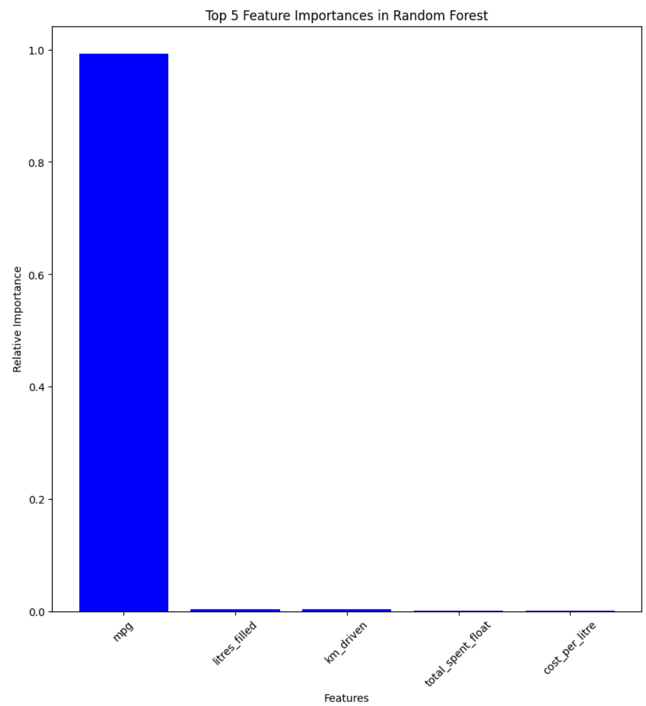

Figure 17: Top 5 Feature Importances in Random Forest

1. Random Forest Feature Importance

The feature `mpg` dominates the feature importance in the Random Forest model, indicating that it is the most critical predictor for fuel efficiency. Other features like `litres_filled`, `km_driven`, `vehicle_age`, and `total_spent` have significantly lower importance scores.

2. Comparison

The feature `litres_filled`, which shows a strong correlation with fuel efficiency, also appears as important in the Random Forest, but not as dominant as `mpg`. `Vehicle_age` and `km_driven`, which had noticeable correlations, are less important in the Random Forest model, suggesting that while they correlate with fuel efficiency, they may not be as predictive when combined with other features in the model. `mpg` shows a significant negative correlation and is also the most critical feature according to the Random Forest, confirming its strong influence on predicting fuel efficiency.

4.3 Fuel Usage in SA

1. We filtered the dataset to include only the records related to South African drivers by selecting rows where the currency is `'R'` (South African Rand). The `.copy()` method is used to create a new DataFrame, ensuring that modifications made to `sa_drivers_dataset` do not affect the original `final_cleaned_data`. Filtering the data is crucial for analyzing specific regions or groups. By focusing on South African

drivers, the analysis becomes more relevant to the local context, leading to more accurate insights about fuel usage patterns in South Africa.

2. A line plot is generated to visualize the trend of fuel prices over time in South Africa. The x-axis represents the date when the fuel was purchased (`date_fueled`), and the y-axis represents the cost per liter of fuel (`cost_per_litre`). Visualizing fuel price trends over time helps in understanding how external factors like economic conditions or global oil prices impact fuel costs in South Africa. It also aids in identifying periods of significant price changes, which could influence driver behavior and fuel consumption patterns.

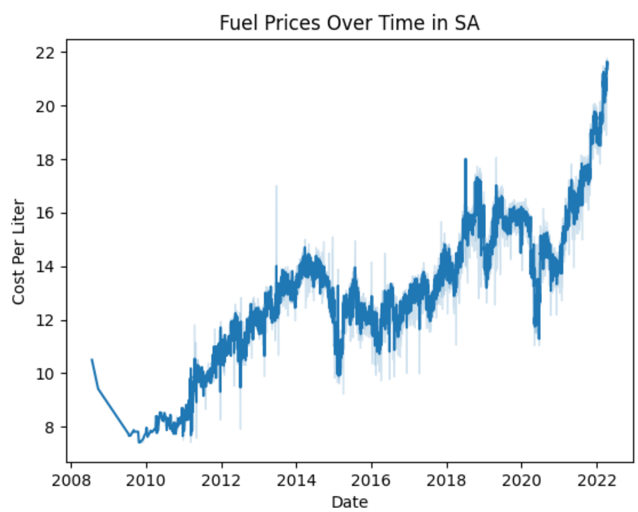

Figure 18: Fuel Prices Over Time in SA

Fuel costs may rise sharply after 2020 due to external causes such as shifts in the global market or local economic circumstances. The upward trend points to a steady rise in living and travel expenses over time, which is important information for budgeting and policy-making.

3. We calculated the day of the week for each `date_fueled`, where Monday is 0 and Sunday is 6. We seek to understand which days drivers are more likely to refuel, as this provides insights into consumer behavior. Highlighting Tuesday allows for a deeper investigation into why this day might be different, possibly due to pricing strategies or consumer habits.

Tuesday is indicated in orange on the bar plot, which shows the total number of refueling on each day of the week. Compared to other days of the week, Tuesday sees the most refueling. This may indicate that drivers prefer to refuel on Tuesdays for a variety of reasons, including pricing differences, promotions, or habitual behaviour. The least amount of refueling occur on Saturday, maybe as a result of fewer people driving or other weekend commitments.

4. We selected refueling records that occurred on the 1st day of the month (`day_fueled == 1`) and were either on a Tuesday (`day_of_week == 1`) or Wednesday (`day_of_week == 2`). The filtered data is stored in `sa_drivers_dataset_tue_wed_1st`, and the first few rows are displayed to verify the selection. By focusing on the 1st of the month, particularly on Tuesdays

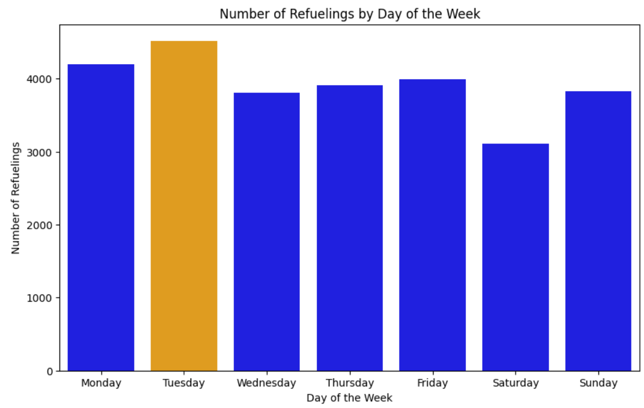

Figure 19: Number of Refuelings by Day of the Week

and Wednesdays, the analysis aims to explore specific patterns or anomalies in fuel prices and driver behavior.

	date_fueled	date_captured	day_fueled	day_of_week
703271	$2017-08-01$	$2017-08-01$	1	1
703309	$2016-06-01$	$2016-06-01$	1	2
703425	$2020-12-01$	$2020-12-01$	1	1
703750	$2018-08-01$	$2018-08-19$	1	2
704122	$2015-07-01$	$2016-04-07$	1	2

(11)

A filtered dataset with refuelings on Tuesdays and Wednesdays on the first day of the month is displayed in the table. The data that has been filtered indicates that there is a need to concentrate on identifying trends associated with the start of the month, when individuals may be refuelling following their monthly budget or after getting their salary.These days can be isolated so that you can examine particular patterns or behaviours that are particular to these periods.

5. We sorted the dataset by date_fueled to ensure that the price trends are calculated sequentially. The dataset is split into two subsets: one for Tuesdays and one for Wednesdays. A year_month column is created to group the data by month. The price trend is calculated by determining the difference in cost_per_litre between consecutive months, labeled as 'Up', 'Down', or 'No Change'. The mean cost_per_litre for each date is calculated and added as a new column to investigate daily price trends further. Analyzing price trends by day and month helps to identify patterns and anomalies in fuel pricing. Understanding these trends is vital for both consumers and businesses to optimize fuel purchasing strategies, and for policymakers to make informed decisions regarding fuel regulations and subsidies.

	date_fueled	day_fueled	day_of_week	cost_per_litre	price_trend
0	2010 − 12 − 01	1	2	8.361045	No Change
1	2011 − 03 − 01	1	1	9.539253	Up
2	2011 − 03 − 01	1	1	8.820705	Up
3	2011 − 03 − 01	1	1	9.420375	Up
4	2011 − 06 − 01	1	2	10.059672	Up
5	2011 − 06 − 01	1	2	9.998912	Up
6	2011 − 11 − 01	1	1	9.959286	Up
7	2011 − 11 − 01	1	1	10.569524	Up
8	2011 − 11 − 01	1	1	10.239309	Up
9	2011 − 11 − 01	1	1	10.239309	Up
10	2011 − 11 − 01	1	1	10.239309	Up
11	2012 − 02 − 01	1	2	10.881247	Up
12	2012 − 02 − 01	1	2	9.919661	Up
13	2012 − 05 − 01	1	1	11.251088	Up
14	2012 − 05 − 01	1	1	11.750373	Up
15	2012 − 05 − 01	1	1	11.779432	Up
16	2012 − 08 − 01	1	2	10.691043	Down
17	2012 − 08 − 01	1	2	10.680476	Down
18	2013 − 01 − 01	1	1	11.581303	Up
19	2013 − 05 − 01	1	2	11.869250	Up

(12)

6. The `value_counts()` function is used on the `price_trend` column to count the occurrences of each trend (`'Up'`, `'Down'`, `'No Change'`) on Wednesdays. The data is filtered using `.iloc[1:]` to exclude the first row (possibly to avoid any initial outlier or setup data) and a condition `sa_drivers_dataset_tue_wed_1st['day_of_week'] == 2` to specifically analyze Wednesdays. This graph sheds light on how fuel prices behave, particularly on Wednesdays.

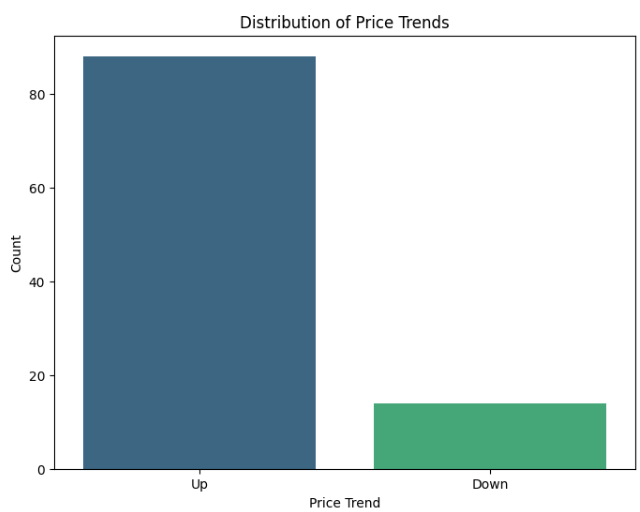

Figure 20: Distribution of Price Trends

When prices drop on the first Wednesday of each month, we see that fewer individuals refuel. This implies that price reductions may not always lead to increased refuelling on those particular days. The data suggests that refuelling behaviour on the first

Wednesday is not solely dependent on price fluctuations, and that refuelling activity does not rise in response to a lower trend in prices.

7. The `value_counts()` function is again used on the `price_trend` column, but this time filtered for Tuesdays using the condition `sa_drivers_dataset_tue_wed_1st['day_of_week'] == 1`. You can determine whether there are any notable variations in the pricing trends on Tuesdays and Wednesdays by comparing this plot with the one created in Step 5. Knowing these patterns can help businesses develop competitive pricing strategies and consumers make decisions. For example, customers may decide to refuel on a day when costs are often lower.

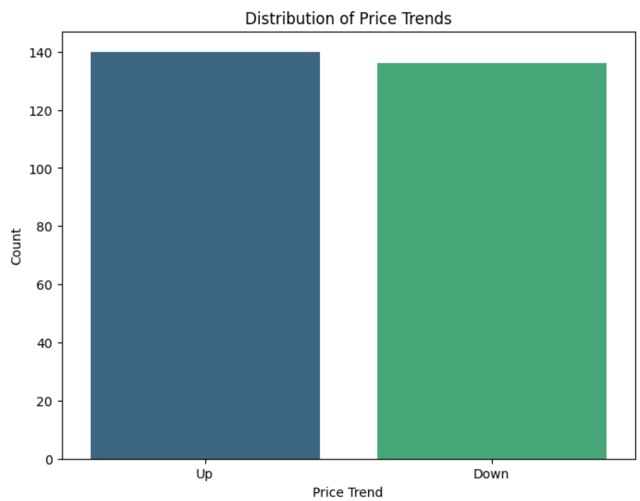

Figure 21: Distribution of Price Trends

With a minor trend for more people to refuel when prices rise, the distribution of refuelings on the first Tuesday of each month appears to be fairly balanced between when prices rise and when they fall. It appears that whether prices rise or fall, the quantity of refuelings on the first Tuesday of the month remains rather constant. The distribution of "Up" and "Down" pricing trends in the data is almost equal, indicating that price changes may not have a significant impact on refuelling behaviour on this particular day.

References

1. https://www.dmre.gov.za/energy-resources/energy-sources/pretoleum/fuel-price-structure#:~:text=This%20means%20that%20the%20domestic,Rand%2FUS%20Dollar%20exchange%20rate
2. https://www.carsguide.com.au/car-advice/what-is-average-fuel-consumption-88469#:~:text=However%2C%20as%20a%20rule%20of,100km%20in%20the%20real%20world.
3. https://www.carsguide.com.au/mitsubishi/pajero
4. https://www.carsguide.com.au/toyota/fortuner/wheel-size
5. https://www.epa.gov/system/files/documents/2022-12/420r22029.pdf