Activity 2 – Data quality and Data cleaning – Group 2

Alexandre Baptista
Informatics Department
Faculdade de Ciências da Universidade de Lisboa
Lisbon Portugal
fc64506@alunos.fc.ul.pt

Vram Davtyan
Informatics Department
Faculdade de Ciências da Universidade de Lisboa
Lisbon Portugal
fc64691@alunos.fc.ul.pt

André Pires
Informatics Department
Faculdade de Ciências da Universidade de Lisboa
Lisbon Portugal
fc64347@alunos.fc.ul.pt

INTRODUCTION

The purpose of this project is to explore and enhance the understanding of data quality and data cleaning methodologies. By defining a research question to guide the project, selecting a relevant dataset, and possibly employing both manual and automated techniques, this project aims to identify and resolve key issues related to incomplete, missing, or inconsistent data. The goal is to apply advanced data processing strategies to refine the dataset and ensure its suitability for insightful analysis.

This projects seeks to bridge theoretical principles with practical application, leveraging Python and open-source tools, namely Tkinter, which it is the standard Python interface to the GUI toolkit to dialog and return the opened file object, in this case the CSV file containing the dataset[1], Pandas, which it is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool[2], Summarytools, which is used to generate standardized and comprehensive summary of dataframe in Jupyter Notebooks and its primary goal is to provide Exploratory Data Analysis (EDA), and matplotlib, to create the visualizations.

CONTRIBUTIONS

Each student contributed an estimated total of 9 hours to the project, with time allocated across various key tasks. Initially, 1 hour was dedicated to exploratory data analysis (EDA) to understand the dataset[3]. Subsequently, 2 hours were spent identifying and listing problems related to missing values and data quality. Another hour was allocated to discussing potential solutions to address these issues. The implementation of strategies for handling missing values and data quality problems involved 1 hour of Python scripting, while an additional hour was devoted to scripting the solution to the core research question. Students also invested 1 hour in discussing and evaluating the results and finally spent 2 hours collaboratively writing the report to document the process and findings comprehensively.

QUESTION N1

This project seeks to uncover insights into the automotive market by answering the question, “What is the type of car with the biggest sales between 2004 and 2024 in the USA?” This exploration aims to analyze trends and patterns in the automotive industry over two decades, using data to shed light on consumer preferences and market dynamics. Through this inquiry, we aim to identify the most dominant selling car type during this period.

The dataset selected for this project is titled "Used Cars Dataset" and was created by Austin Reese. This dataset is derived from Craigslist, the world's largest online platform for buying and selling used vehicles. The data is scraped and updated quarterly, ensuring its timeliness and relevance for ongoing studies, and being publicly available on Kaggle[4], the dataset is particularly valuable due to its structured format and inclusion of critical attributes, making it well-suited for exploring questions.

The dataset contains attributes that provide a robust foundation for analyzing trends and patterns in the used car market with 426 thousand advertisements and 26 attributes that describe either the car details or selling advertisements. The most important attributes are: 1) Id which is an numeric attribute (continuous data) with unique values that is the reference of the advertisement in the website; 2) region which is an text attribute (nominal data) that describes the region of each advertisement; 3) year which is an numeric attribute (continuous data) that describes the year the car was built; 4) manufacturer which is an text attribute (nominal data) that describes the brand of the car; 5) model which is an text attribute (nominal data) that describes the model of the car; 6) VIN which is an text attribute (nominal data) that describes the vehicle identification number; 7) type which is an text attribute (nominal data) that describes category of the car;
state: this an text attribute (nominal data) that describes the state of each advertisement.

columns. It has a quite interesting value in it, because it allows us to compare the information in the other columns with it. Generally when analyzing textual data and trying to extract information from it will lead to approaches such as using regular expressions. But with the help of currently available Llms it is possible to create prompts and extract all of the desired information. To accomplish this task we will use Ollama a , all the required python libraries and installations will be included in the Readme.md file on this Github repository. The key idea is to use the description and fill any missing element and in case of unequal values, ones from description will be favoured and replaced.

DATA CLEANING STRATEGY

To analyze the dataset effectively, a comprehensive Exploratory Data Analysis (EDA) was performed, incorporating multiple steps to ensure a deep understanding of the data and its nuances. The process began with identifying patterns and trends across the dataset, emphasizing key attributes such as type, manufacturer, and model, identifying the localization measures of

the numeric attributes, mainly to verify the existence of outliers in the attribute year.

The analysis then turned to missing values, revealing that the type attribute is missing in 22% of the observations. Since this attribute plays an important role in answering the project question, it was determined that a strategy must be developed to address these missing values rather than simply excluding them from the dataset.

type [object]	1. nan	92,858 (21.8%)
	2. sedan	87,056 (20.4%)
	3. SUV	77,284 (18.1%)
	4. pickup	43,510 (10.2%)
	5. truck	35,279 (8.3%)
	6. other	22,110 (5.2%)
	7. coupe	19,204 (4.5%)
	8. hatchback	16,598 (3.9%)
	9. wagon	10,751 (2.5%)
	10. van	8,548 (2.0%)
	11. other	13,682 (3.2%)

Figure 1: EDA of type attribute

Finally, an analysis of duplicates was conducted, leading to the conclusion that while no duplicated observations exist in the dataset, but there are multiple ads sharing the same Vehicle Identification Number (VIN). A deeper investigation into these cases revealed that the same car is often listed for sale in multiple regions across the United States. This insight highlights the importance of accounting for such cases to avoid over-representing certain cars in the dataset.

VIN [object]	1. nan	161,042 (37.7%)
	2. 1FMJU1JT1HEA52352	261 (0.1%)
	3. 3C6JR6DT3KG560649	235 (0.1%)
	4. 1FTER1EH1LLA36301	231 (0.1%)
	5. 5TFTX4CN3EX042751	227 (0.1%)
	6. 1GCHTCE37G1186784	214 (0.1%)
	7. 1GTN1TEH5EZ273019	207 (0.0%)
	8. 3VWF17AT1FM655022	199 (0.0%)
	9. JN1AZ4EH8KM420880	198 (0.0%)
	10. 1FTMF1CP3GKD62143	195 (0.0%)
	11. other	263,871 (61.8%)

Figure 2: EDA of VIN attribute

Following the analyses previously mentioned, several automations have been implemented in the Python script to enhance the quality and consistency of the dataset. These automations were designed to address common issues such as inconsistent formatting, missing data, and duplicates, ensuring that the dataset is both reliable and ready for further analysis. Each step in the automation process has been carefully crafted to minimize errors and maintain the integrity of the data.

First, the text fields have been normalized by converting all text to lowercase, ensuring uniformity and avoiding mismatches caused by variations in capitalization. For instance, entries such as "Bmw" "bmw," and "BMW" would be treated as the same value after normalization. This helps streamline downstream tasks such as categorization, matching, and filtering, which rely on consistent data.

Second, handling missing data for car type involved a systematic approach to imputation based on related attributes. Selling ads with missing car type information were supplemented using details from other ads that matched the same manufacturer and model. This imputation process leverages the logical assumption that cars of the same make and model are likely to share similar characteristics, including type. For models associated with multiple car types (e.g., hatchback, sedan), the automation resolves ambiguity by assigning the most frequently occurring car type for that specific model. This follows the principle that common patterns in the data are more likely to reflect reality. In cases where no clear resolution could be achieved, for example, due to insufficient data or conflicting information, the observations were excluded to maintain the integrity and reliability of the dataset.

Third, the problem of duplicate values was addressed by identifying the VIN (Vehicle Identification Number) as the main attribute for detecting repeated ads. The VIN is a unique identifier for vehicles and serves as a robust basis for identifying duplicates, even if other details in the ads differ slightly. For ads with the same VIN but differing year values, the year attribute was harmonized by taking the median year value for that VIN. This approach ensures that minor discrepancies in reporting do not lead to unnecessary exclusions while still maintaining a high level of data accuracy. For cases where ads with the same VIN had conflicting manufacturer or model information, these entries were excluded, as it was impossible to verify their authenticity. With this approach, duplicate ads posted across different regions were identified and consolidated to prevent overcounting.

By implementing these automations, the dataset has been significantly improved in terms of quality, consistency, and usability. The process not only addresses common data issues but also ensures that the final dataset reflects reality as closely as possible.

RESULTS QN1

After implementing these strategies make the dataset a more reliable foundation for subsequent analysis the project question can be answered. Based on the analysis, the type of car most commonly advertised between 2004 and 2024 in the USA is the SUV. This is evident from the peak in the number of cars ads, which significantly surpasses all other car types, particularly between 2010 and 2018. SUVs consistently dominate during this time frame, possible due to their immense popularity among consumers. While sedans maintain a strong presence, their numbers do not reach the peaks achieved by SUVs. Thus, SUVs can be identified as the car type with most selling advertisements during this 20-year period in the USA.

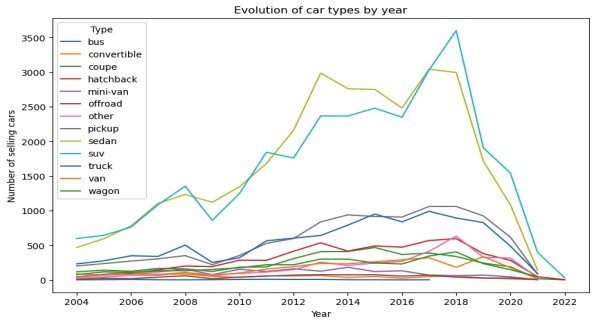


Figure 3: Evolution of car types by year

QUESTION N2

This Question arose from the Data itself. The ['description'] column holds almost all of the information from other columns. Our Goal is to use this information to see if there are mismatches between those two. Therefore any rows containing empty description will be dropped. For further data visualization some of the columns which do not present a particular interest will also be removed.

GENERAL DESCRIPTION

For the question N2 project includes a python executable which handles the data and does all the necessary cleaning and manipulation and also a jupyter notebook to showcase the end result with some charts. Because of the rather large size of the csv file (around 1.35 GB) simply using pd.read_csv() is not the best approach so converting the file into a parquet file and loading it instead is prefered, this will reduce the file size to (434MB) and will significantly improve the loading speed.

OUTPUT FROM THE PROMPT

To gather the desired output we iterate through each row of the dataframe and the Llm gets the description of the car with columns as keys and creates a corresponding dictionary based on that. Some techniques are used to make sure that the output is in the correct format but there is still room for improvement due to some unexpected outputs that occurred many times. This part is explained with simpler example in the jupyter notebook.

POTENTIAL ERRORS AND ROOM FOR IMPROVEMENT

Due to the nature of the Llm the answer it gives is quite inconsistent at times and improvements in the prompt should be made with a better error handling system. Further, due to the immense size of the file even after dropping a significant amount of data running the script with the intended output in mind was not possible on local machine. After solving the corresponding issues an updated version will be pushed to Github Repo and will be continuously updated. To make it easier to follow, no branches in Repo would be made, it will only include the latest version.

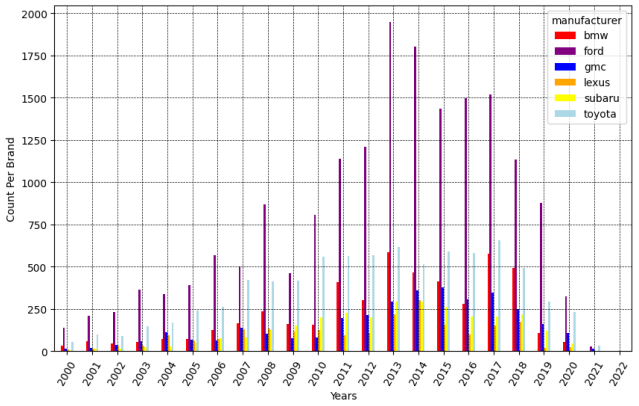
OPEN SOURCE TOOLS AND LIBRARIES

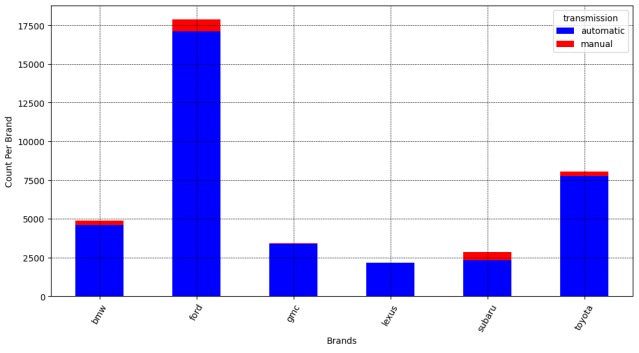
Everything used in this project is described in the Github Repo and is included in requirements.txt file. Which include Ollama, Langchain-Ollama, Pandas, Matplotlib, Pyarrow.

RESULTS QN2

Those charts showcase some of the preselected brands using the criteria by mentioning year, transmission, condition and odometer's reading, and correlation between transmission types. The picture below represents the idea that was in our mind, here you can see the difference between two values. Ideally if the url for corresponding car is correct it would be possible to check which value is true. But for visualization purposes the value from description is favoured. Therefore older value will be replaced. As you can see in odometer an integer is replaced with a string. With a different prompt and changing the dtypes this issue can be resolved.

E	F	G	H	I	J	K	L
price	year	manufacturer	model	condition	cylinders	fuel	odometer
4500	1992	jeep	cheroke	excellent	6 cylinders	gas	192000
29990, Desc : $26990	2018, Desc : 2022	alfa-romeo, D	romeo	good		other	26978, Desc : 26k miles
2100, Desc : 2100	2006, Desc : 2006	subaru, Desc :	impreza	fair, Desc :	4 cylinders	gas	97000
80	2004	honda		excellent	6 cylinders	gas	94020





LINKS

Github
dataset

REFERENCES

[1] https://docs.python.org/3/library/dialog.html
[2] https://pandas.pydata.org
[3] https://pypi.org/project/summarytools/
[4] https://www.kaggle.com/datasets/austinreese/craigslist-carstrucks-data