CONSTRUCTOR UNIVERSITY

CO-586 Data Management and Analytics in Industry 4.0

Prof. Dr. -Ing. Hendro Wicaksono

Final report

Dataset 3: Car acceptability

Written by:

Joelle Karadsheh

Table of Contents

1 Dataset 3: Car Acceptability

1.1 Data description

1. What is the dataset all about?

This dataset is structured specifically for classification tasks to predict the acceptability of a car. It enables the modeling of relationships between a car's features and its market acceptability, which is pivotal for manufacturers and dealers to understand market trends and buyer preferences. The nominal nature of the data supports classification algorithms that handle categorical input to determine the potential classification of car acceptability.

2. Data dictionary

Table 11: Data dictionary for Car Acceptability

Column Name	Definition	Data Type	Possible Values	Required?
Buying_Price	The buying price category of the car	Nominal	vhigh, high, med, low	Yes
Maintenance_Price	The maintenance price category of the car	Nominal	vhigh, high, med, low	Yes

No_of_Doors	The number of doors in the car	Nominal	2, 3, 4, 5more	Yes
Person_Capacity	The capacity of the car in terms of the number of persons it can carry	Nominal	2, 4, more	Yes
Size_of_Luggage	The size of the luggage compartment	Nominal	small, med, big	Yes
Safety	The safety level of the car	Nominal	low, med, high	Yes
Car_Acceptability	The acceptability rating of the car	Nominal	unacc, acc, vgood, good	Yes

More details; Buying_Price: This variable categorizes the initial cost to purchase the car. It's an important factor for buyers as it directly affects their budget.

Maintenance_Price: This variable categorizes the ongoing cost required to maintain the car. This can influence a buyer's decision by reflecting potential future expenses after the purchase.

Price can also influence the perceived value of a product. For example, if a product has a lower price compared to similar products on the market, consumers may perceive it as an attractive offer. This can drive purchases, as consumers feel they are getting good value for their money. (Team, *How price influences the perception and behavior of consumers* 2023)

1.2 Exploratory data analysis

1.2.1 Data visualization and exploration

Understanding data types for car acceptability is crucial not only for graphing and analysis but also for data preprocessing tasks. It ensures that appropriate methods and techniques are applied to handle different types of variables, leading to more accurate and reliable results.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1728 entries, 0 to 1727
Data columns (total 7 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Buying_Price       1728 non-null   object
 1   Maintenance_Price  1728 non-null   object
 2   No_of_Doors        1728 non-null   object
 3   Person_Capacity    1728 non-null   object
 4   Size_of_Luggage    1728 non-null   object
 5   Safety             1728 non-null   object
 6   Car_Acceptability  1728 non-null   object
dtypes: object(7)
memory usage: 94.6+ KB
```

Figure 37: Data visulization

The dataset presented has a dimension of (1728, 7), indicating that it comprises 1728 entries across 7 distinct variables. Each entry corresponds to a single observation of a car's attributes, and each of the 7 variables represents a specific characteristic of the car that is relevant to its acceptability.

```
In [7]: df.describe()
```

Out[7]:

	Buying_Price	Maintenance_Price	No_of_Doors	Person_Capacity	Size_of_Luggage	Safety	Car_Acceptability
count	1728	1728	1728	1728	1728	1728	1728
unique	4	4	4	3	3	3	4
top	vhigh	vhigh	2	2	small	low	unacc
freq	432	432	432	576	576	576	1210

Figure 38: Data Statistics

The descriptive statistics highlight:

- The dataset is complete with 1728 entries across all variables, indicating no missing data.
- 'Vhigh' is the top category for both buying and maintenance prices, occurring 432 times each.
- A large portion of the dataset consists of vehicles with '2' doors and '2' person capacity.
- The 'small' size of luggage and 'low' safety are common, each category appearing 576 times.
- The car acceptability variable is predominantly 'unacc', representing 1210 of the entries, suggesting a dataset skewed towards cars that are not acceptable based on the set criteria.

1. Histogram where car acceptability is on the x-axis and the count is on the y-axis

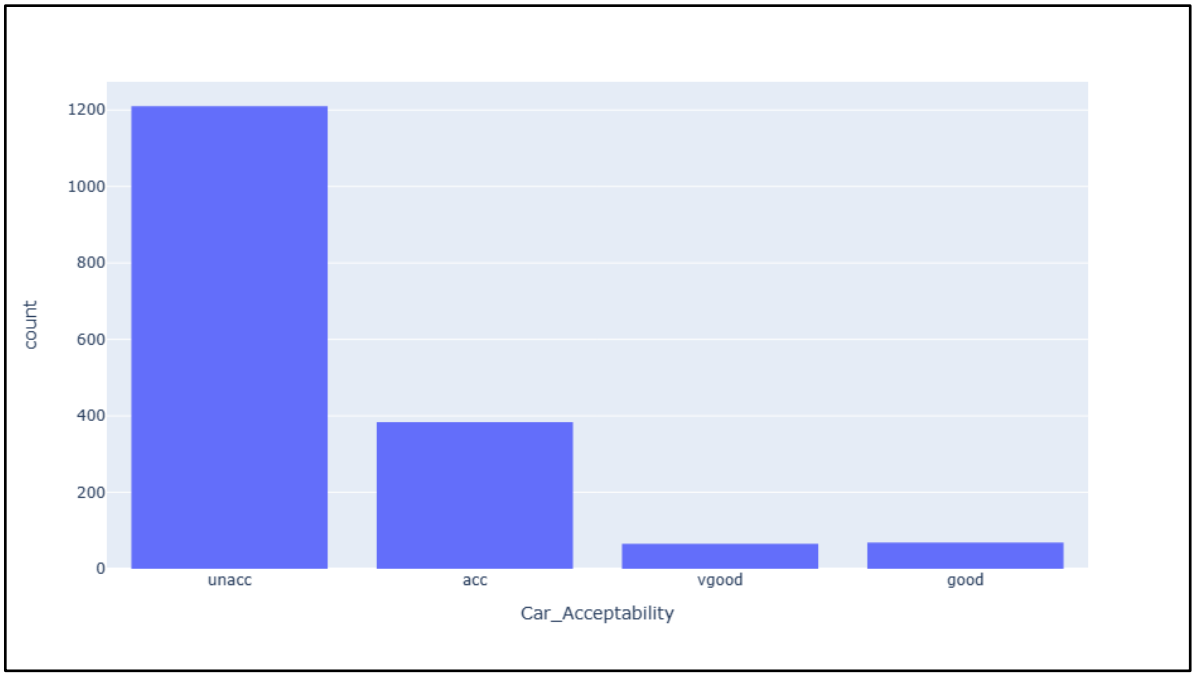


Figure 39: Histogram of car acceptability and count

The histogram displays the frequency of car acceptability ratings within the dataset. It shows a striking predominance of cars classified as 'unacc' (unacceptable), with this category vastly outnumbering the others. In contrast, 'acc' (acceptable), 'vgood' (very good), and 'good' categories have significantly fewer cars, with 'vgood' and 'good' particularly sparse. The data indicates that a considerable number of cars don't meet certain criteria set for acceptability. This could be influenced by factors such as safety ratings, cost, and capacity.

2. Box plot where car acceptability and buying price are being compared

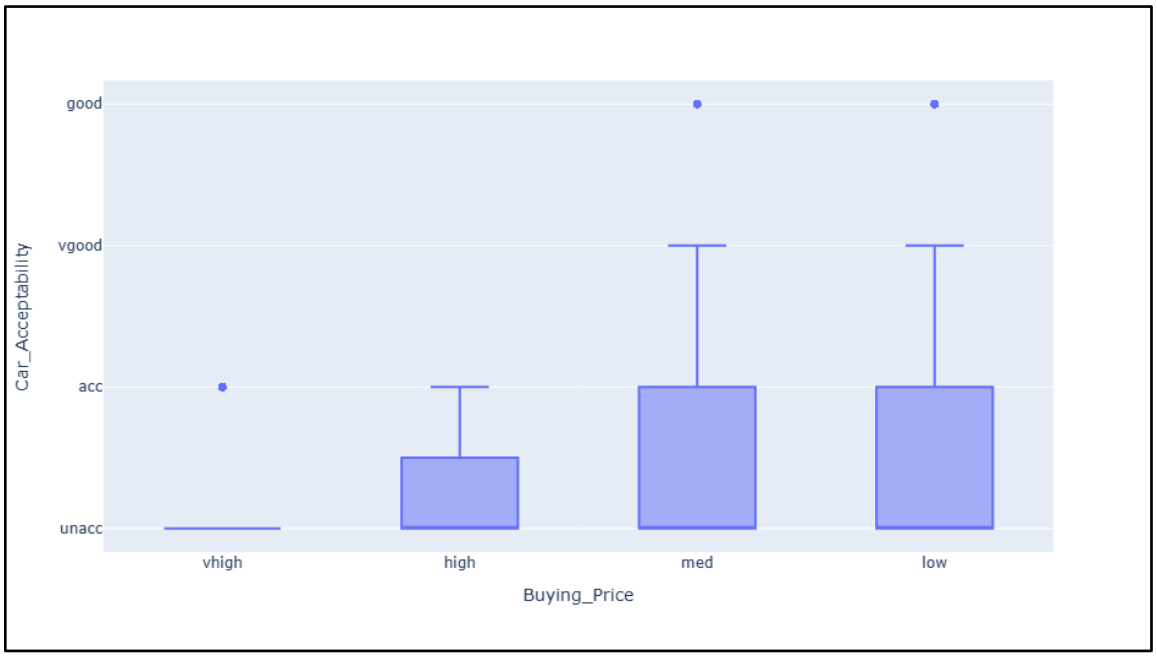
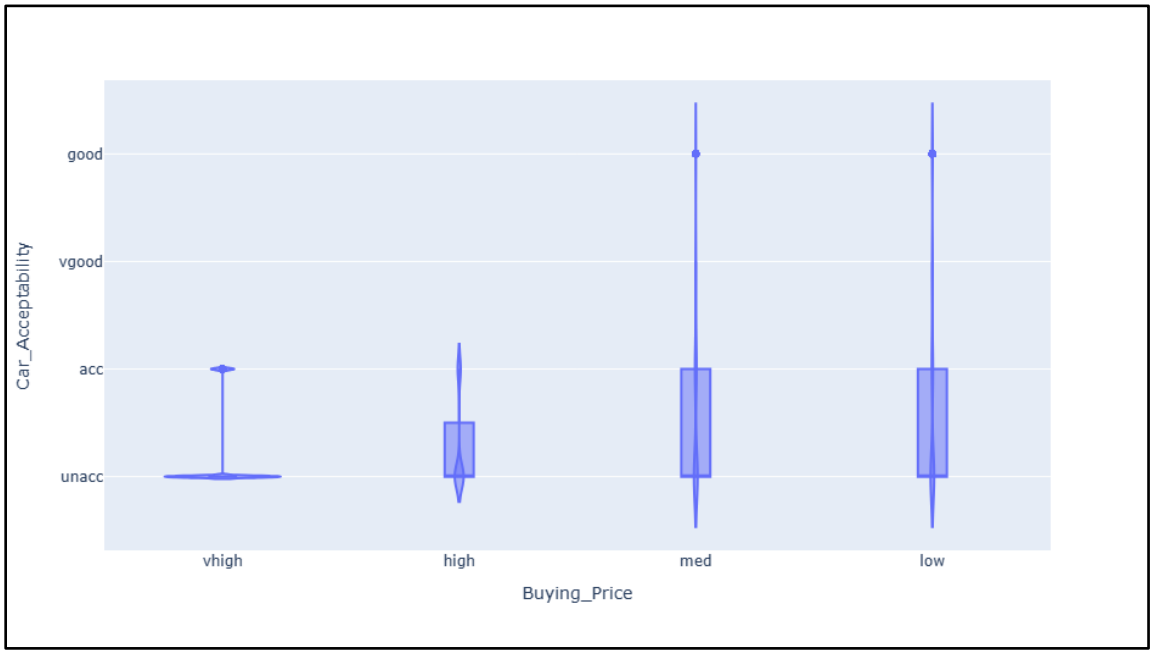


Figure 40: Box plot for buying price and car acceptability

The 'unacc' (unacceptable) category spans across all buying price ranges, with a particular concentration in the 'vhigh' and 'high' price categories, suggesting that higher prices do not guarantee acceptability. The 'acc' (acceptable) rating occurs within the 'high' to 'low' price ranges, with no 'acc' cars in the 'vhigh' category. This might indicate that excessively high buying prices negatively impact a car's acceptability. Ratings of 'good' and 'vgood' appear exclusively in the 'med' and 'low' price ranges. This suggests that more affordable cars are more likely to be rated favorably in terms of acceptability.



3. Violin plot that compares the buying price and car acceptability

Figure 41: Violin plot that compares the buying price and car acceptability

The 'unacc' (unacceptable) rating is concentrated in the 'vhigh' buying price category and is represented by a line, indicating no variability; all 'vhigh' cars are rated as unacceptable. The 'acc' (acceptable) ratings appear primarily in the 'high' buying price category, with some variability in acceptability shown by the length of the whiskers. The 'good' and 'vgood' ratings are not observed in the 'vhigh' price category and are distributed across 'high', 'med', and 'low' categories. The variability is noticeable for cars in the 'med' price range, less so in the 'high' and 'low' ranges.

4. Scatter plot taking Buying price, Car acceptability and Safety into account

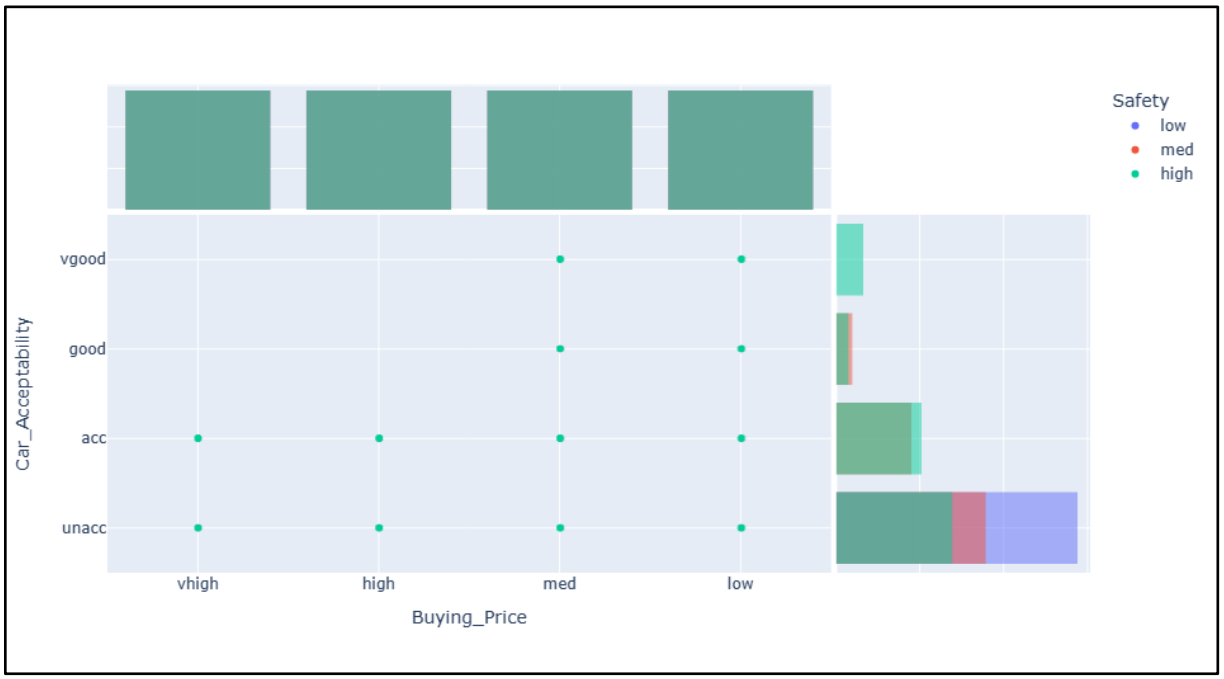


Figure 42: Scatter plot taking Buying price, Car acceptability and Safety into account

The 'unacc' (unacceptable) rating spans all buying price categories, with cars rated as 'unacc' regardless of safety rating. The 'acc' (acceptable) rating is present across various price categories but seems more prevalent in cars with 'high' and 'med' safety ratings. Cars rated as 'good' and 'vgood' for acceptability are mostly found in the 'med' and 'low' buying price categories and predominantly have 'high' safety ratings.

Note: since the dataset does not have any numerical values I couldnt use the pairplot graph or the heatmap graph

However I did an extra plot called catplot that counts the categorical variables and plots them in a graph.

5. Catplot of the car acceptability/safety and their count.

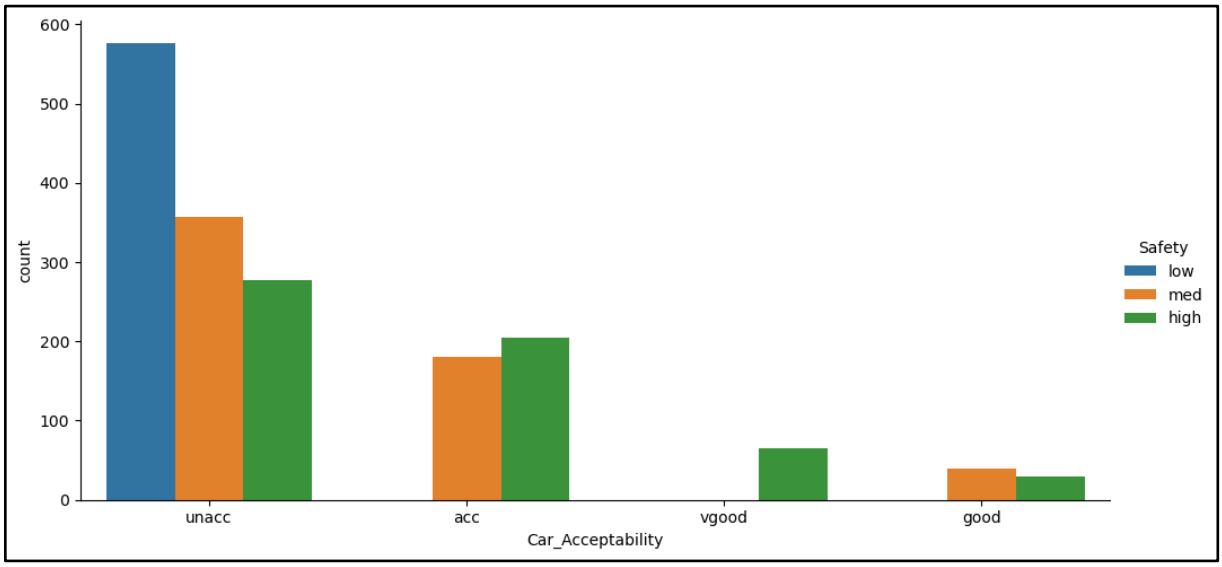


Figure 43: Catplot of the car acceptability/safety and their count.

The 'unacc' category has the highest count, with a significant number of cars rated as having 'low' safety. In the 'acc' category, cars with 'med' and 'high' safety ratings are more prevalent than those with 'low' safety. The 'vgood' and 'good' categories show cars only with 'med' and 'high' safety ratings. Notably, no cars with a 'low' safety rating fall into these categories. The 'good' category, while having fewer cars overall, exhibits a count of cars with 'high' safety equal to those with 'med' safety.

1.2.2 Brainstorming and discussions

Hypothesis 1:

The hypothesis could be that the standards used to evaluate car acceptability—likely related to attributes like safety, cost, and capacity—are not met by most cars in the dataset.

Findings for Hypothesis 1:

- The vast majority of cars are rated as 'unacc' (unacceptable), indicating that the evaluation standards, which may emphasize safety, cost-effectiveness, capacity, and other attributes, are not met by these vehicles. (Paul Fink aut, 2019)

- Only a small fraction of cars are considered 'acc' (acceptable), 'vgood' (very good), or 'good'. This suggests that positive evaluations are reserved for cars that excel in areas such as cost-value balance, safety features, and perhaps capacity, which are less commonly observed in the dataset.
- The stark disparity between the number of 'unacc' cars and those in the other categories suggests key areas where cars can improve to meet the standards, or it may suggest a reconsideration of the standards if they are disproportionately skewed toward factors that are challenging for most car models to achieve.

Hypothesis 2:

The hypothesis could be that cars with lower buying prices are more likely to be rated as acceptable or better, whereas very high buying prices are associated with a car being rated as unacceptable.

Findings for hypothesis 2:

- Cars with moderate to low buying prices tend to have higher acceptability ratings, possibly due to better perceived value or affordability.
- There is a notable absence of 'acceptable' cars in the very high price category, indicating that buyers might have higher expectations for very expensive cars, which could be not met as frequently.
- The spread and outliers in 'acc', 'good', and 'vgood' categories suggest some variability in how cars are rated within these price brackets, warranting further investigation into other factors that might influence these acceptability ratings beyond price alone.

Hypothesis 3:

The hypothesis here could be that safety ratings play a significant role in a car's acceptability and may even be more influential than the buying price, especially for cars in the 'med' and 'low' price ranges.

Findings for hypothesis 3:

- A key finding is that cars with 'high' safety ratings tend to have better acceptability, particularly in the 'med' and 'low' price categories, suggesting that safety is a critical factor in the evaluation of car acceptability.
- The absence of 'good' and 'vgood' ratings in the 'vhigh' price category, even with 'high' safety ratings, might indicate that there are diminishing returns on acceptability as the price increases. (Paul Fink aut, 2019)
- The fact that there are no cars with 'low' safety ratings that are rated above 'unacc' reinforces the importance of safety in car acceptability.

1.3 Data preprocessing

1.3.1 Handling missing values

After conducting a thorough assessment of the dataset's completeness, it is evident that the dataset has no missing values. The analysis, which included key variables such as Buying_Price, Maintenance_Price, Number_of_Doors, Person_Capacity, Size_of_Luggage, Safety, and Car_Acceptability, yielded a missing ratio of 0.0 for each of these columns. This indicates that there are no missing entries in any of these fields across the entire dataset.

Out[36]:

	column_name	missing_ratio
Buying_Price	Buying_Price	0.0
Maintenance_Price	Maintenance_Price	0.0
No_of_Doors	No_of_Doors	0.0
Person_Capacity	Person_Capacity	0.0
Size_of_Luggage	Size_of_Luggage	0.0
Safety	Safety	0.0
Car_Acceptability	Car_Acceptability	0.0

Figure 44: Identifying missing values using pandas

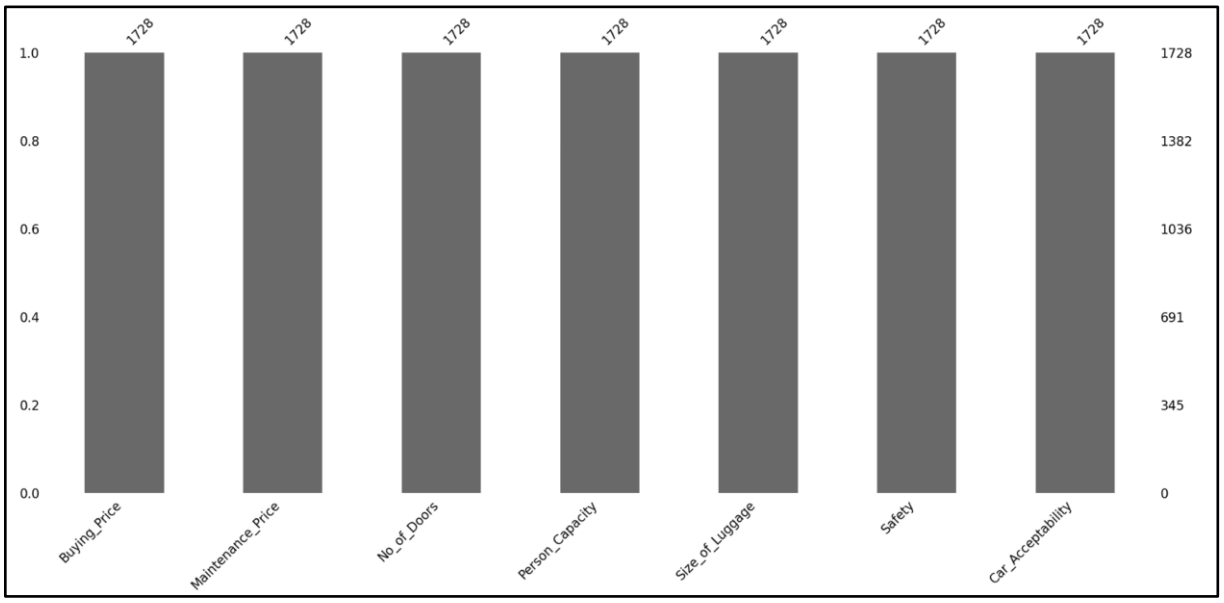


Figure 45: Missingno Bar

As shown in the figure all bars representing variables have the total of 1728 which is the dimension of the data set. Further confirming that there are no missing values. There is no need for a KNN Imputation.

1.3.2 Encoding categorical data

The dataset contains several categorical variables, such as 'Buying_Price', 'Maintenance_Price', 'No_of_Doors', 'Person_Capacity', 'Size_of_Luggage', 'Safety', and 'Car_Acceptability' as shown in the figure below.

```
for col in df.select_dtypes(exclude = 'number').columns:
    print(col)
```

```
Buying_Price
Maintenance_Price
No_of_Doors
Person_Capacity
Size_of_Luggage
Safety
Car_Acceptability
```

Figure 46: Categorical columns

```
df
```

[30]:

	Buying_Price	Maintenance_Price	No_of_Doors	Person_Capacity	Size_of_Luggage	Safety	Car_Acceptability
0	3	3	2	2	0	0	0
1	3	3	2	2	0	1	0
2	3	3	2	2	0	2	0
3	3	3	2	2	1	0	0
4	3	3	2	2	1	1	0
...	...	...	...	...	...	...	...
4835	0	0	3	5	1	2	3
4836	1	1	2	5	2	2	3
4837	0	2	5	4	1	2	3
4838	0	2	5	4	2	2	3
4839	1	0	5	4	1	2	3

4840 rows × 7 columns

Figure 47: Encoded data

In our analysis of the Car Acceptability dataset, we have switched from using one-hot encoding to label/ordinal encoding for organizing categorical variables. This switch is important because our data categories (like 'unacc' = 0, 'acc' = 1, 'good' = 2, 'vgood' = 3) have a specific order, and label/ordinal encoding helps maintain this order while reducing the number of features in our model. This simplifies the model and helps it run faster and more effectively. Additionally, we revised how we measure the model's performance to better fit situations where there are multiple categories (unacc=0, acc=1, good=2, vgood=3) instead of just binary (0 and 1), using measures like accuracy and F1-score that are appropriate for models predicting multiple categories. This gives us a clearer view of how well our model is working.

1.3.3 Feature scaling

```
for col in df.select_dtypes(include = 'number').columns:
    print(col)
```

```
Buying_Price_high
Buying_Price_low
Buying_Price_med
Buying_Price_vhigh
Maintenance_Price_high
Maintenance_Price_low
Maintenance_Price_med
Maintenance_Price_vhigh
No_of_Doors_2
No_of_Doors_3
No_of_Doors_4
No_of_Doors_5more
Person_Capacity_2
Person_Capacity_4
Person_Capacity_more
Size_of_Luggage_big
Size_of_Luggage_med
Size_of_Luggage_small
Safety_high
Safety_low
Safety_med
Car_Acceptability_acc
Car_Acceptability_good
Car_Acceptability_unacc
Car_Acceptability_vgood
```

Figure 48: Non-numerical columns

The information provided indicates that the dataset consists exclusively of categorical variables that have been one-hot encoded, resulting in binary columns with values of 0 or 1. Since feature scaling, such as normalization or standardization, is typically applied to numerical data to bring all variables to the same scale, it is not applicable in this context

1.4 Development of predictive models

1.4.1 Dataset splitting

Since we have around 7200 entries the possible range would be from 0.1-0.2. A 20/80 split offers a practical balance. It ensures that the majority of the data is used to train our model, maximizing the learning from available examples, while still reserving a meaningful portion for testing to ensure that our findings are reliable and the model's performance is robust when faced with new data. This approach is commonly adopted in the field and is especially suitable for datasets of this size.

As for the Y value we chose the only dependent variable which is car acceptability and specifically chose the encoded column Car_acceptability_acc which shows the accepted cars.

	Buying_Price	Maintenance_Price	No_of_Doors	Person_Capacity	Size_of_Luggage	Safety
0	3	3	2	2	0	0
1	3	3	2	2	0	1
2	3	3	2	2	0	2
3	3	3	2	2	1	0
4	3	3	2	2	1	1
...	...	...	...	...	...	...
4835	0	2	3	5	1	2
4836	1	1	4	4	1	2
4837	0	1	5	4	2	2
4838	0	2	2	4	2	2
4839	0	2	3	5	2	2

	Car_Acceptability
0	0
1	0
2	0
3	0
4	0
...	...
4835	3
4836	3
4837	3
4838	3
4839	3

4840 rows × 1 columns

Figure 49: Data set split

```
my_test_size = 0.2
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=my_test_size, shuffle=True)
X_train.describe()
```

	Buying_Price	Maintenance_Price	Size_of_Luggage	Safety
count	3872.000000	3872.000000	3872.000000	3872.000000
mean	0.928202	1.053461	1.181043	1.435176
std	1.022359	1.037894	0.781154	0.696103
min	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.000000	1.000000	1.000000
50%	1.000000	1.000000	1.000000	2.000000
75%	1.000000	2.000000	2.000000	2.000000
max	3.000000	3.000000	2.000000	2.000000

Figure 50: Split dataset with a test size of 20%

1.4.2 Model selection and parameter settings

In addressing the task of classifying car acceptability based on categorical features such as Buying_Price, Maintenance_Price, Safety, and others, I have chosen four specific models: Decision Trees, Naive Bayes, Logistic Regression, and Random Forest. Each model has distinct characteristics making them suitable for handling categorical data in different ways.

1. Decision Trees: Ideal for handling categorical data, Decision Trees provide clear interpretability, which is crucial for understanding the factors influencing car acceptability. They work well with non-linear data distributions typical of categorical data and are flexible in capturing interactions between features without the need for feature scaling.
2. Random Forest: Random Forest is an ensemble machine learning algorithm that constructs multiple decision trees during training and aggregates their results for the final prediction, making it highly effective for datasets with complex patterns like ours, which involves predicting car acceptability from categorical variables. We chose Random Forest because it manages categorical data well, reduces overfitting through its ensemble nature, and provides important feature insights
3. Logistic Regression: Despite the categorical nature of the dataset, we chose Logistic Regression due to its robustness in binary classification problems. It can efficiently handle multi-class scenarios, providing a probabilistic understanding of class memberships. (It is a method for modeling relationships between variables and predicts a variable based on one or more other variables)
4. XGBoost: XGBoost (Extreme Gradient Boosting) is an advanced implementation of gradient boosting that excels in handling complex data patterns and interactions between features, making it ideal for our dataset composed entirely of categorical variables for predicting car acceptability. We chose XGBoost for its efficiency, robust handling of different data types, and its ensemble learning approach that significantly improves predictive accuracy.

Table 12. Parameter setting of Decision Trees

Parameter	**Value**
Criterion	Gini
Random state	0
Mini samples split	10
Max depth	30

- Criterion ('Gini'): Utilizes Gini impurity as the criterion for splitting nodes. This measure is faster to compute than entropy, making it suitable for larger datasets where computational efficiency is crucial.
- Random State (42): Sets the seed for the random number generator to '42' to ensure reproducibility. This helps in achieving consistent results across multiple runs, facilitating more reliable comparisons and validations of model performance.
- Min Samples Split (10): Specifies that a node must have at least 10 samples before it can be split. This parameter helps prevent the model from becoming overly complex and overfitting to the noise in the training data. By requiring more samples to justify each split, the model is encouraged to find more generalizable patterns in the data.
- Max Depth (30): Limits the maximum depth of the tree to 30 layers. A deeper tree can model more complex relationships by capturing more information about the data. However, setting a limit helps mitigate the risk of overfitting by not allowing the tree to become too deep, which might adapt too specifically to the training data. Testing with various depths (e.g., 10, 20, 30,

None) is a good practice to find an optimal balance between bias (underfitting) and variance (overfitting).

Table 13. Parameter setting of Logistic Regression

Parameter	**Value**
solver	´lbfgs`
max_iter	200
C	0.5
Penalty	I2

- Solver ('lbfgs'): This solver is recommended for small to medium-sized datasets and supports only L2 regularization. It's well-regarded for its robustness and efficiency in handling logistic regression problems.
- Penalty (L2): The L2 penalty helps prevent overfitting by penalizing the square of the coefficients, which encourages smaller, more diffuse coefficient values. Using L2 regularization with the 'lbfgs' solver is standard practice as it provides stable solutions.
- C (0.5): This is the inverse of regularization strength; a lower value indicates stronger regularization. Setting C to 0.5 helps control overfitting by imposing stronger regularization.
- Max Iter (200): The max_iter parameter specifies the maximum number of iterations the solver is allowed to run until convergence. Increasing this value to 200 is beneficial for ensuring that the model has sufficient opportunity to find the optimal solution, especially in cases where the dataset is large or features are many, which can extend the convergence time.

Table 14. Parameter setting of Random Forest

Parameter	**Value**
n estimators	300
max depth	10
min samples split	10
Random State	0

- n_estimators: We chose 300 trees in the forest to ensure that the model has enough variety in the trees to capture complex patterns and stabilize the prediction across different types of input data.
- max_depth: Set to 10, this parameter limits the growth of the trees, preventing them from becoming overly complex and overfitting the training data. It strikes a balance between learning fine details and maintaining a general approach to unseen data.
- min_samples_split: We selected a value of 10, meaning a node will split only if it contains more than 10 samples. This avoids overly granular splits in smaller groups, reducing the noise influence in the model’s predictions.

Table 15. Parameter setting of XGBoost

Parameter	**Value**
max depth	4
learning rate	0,1
n_estimators	200

subsample	0.8
colsample_bytree	0.8

- max_depth: We set this to 4 to allow the model to explore the data sufficiently but not so deeply that it begins fitting to noise, balancing complexity and overfitting risks.
- learning_rate: We chose a value of 0.1 to ensure the model learns steadily but remains sensitive enough to adjust to nuances in the dataset without overshooting during updates.
- n_estimators: We used 200 trees in the ensemble, providing a robust framework for learning that improves accuracy by aggregating more decision paths, yet remains computationally feasible.
- subsample: Set at 0.8, this parameter helps in reducing overfitting by using only 80% of the data for each tree's training, thereby adding more diversity to the models.
- colsample_bytree: Also set to 0.8, allowing each tree to consider only 80% of features when making splits, which increases the model's ability to generalize by preventing it from relying too heavily on any single feature.

1.5 Models evaluation

Note: Our ROC curve was not working for some reason. The professor is aware of that :)

	Accuracy	Precision	Recall	F1 score
Logistic Regression	0.883264	0.883264	0.883264	0.883264
Random Forest	0.995868	0.995868	0.995868	0.995868
Decision Tree	0.995868	0.995868	0.995868	0.995868
XGBoost	0.995868	0.995868	0.995868	0.995868

Figure 51: Cross Validation Evaluation

It is apparent that Logistic regression is outperformed. while logistic regression achieves high accuracy, it falls short of the better scores of the other models and is more suitable for simpler, interpretable problems. Based on the metrics, Random Forest, Decision Tree, and XGBoost all achieve the highest scores in accuracy, precision, recall, and F1 score. Since their performance is identical, the choice among these models should be based on other considerations. From our research it was concluded that XGBoost is typically faster and can handle larger datasets compared to the other models. Given these considerations, XGBoost would be the best choice overall due to its robustness, efficiency, and excellent performance metrics.

1.6 Discussion

When integrating predictive models into a business sector, there are multiple implications and potential benefits. The effectiveness of these models can significantly transform operations, strategic decision-making, and competitive dynamics.

1. Improved Processes:

Predictive models can streamline several processes within the automotive industry, especially in:

- Inventory Management: Predicting car acceptability can help manufacturers and dealerships optimize their inventory by stocking models and variants that are more likely to be accepted by the market.
- Marketing and Sales Strategies: Insights from the model can guide targeted marketing strategies, focusing on the most appealing features and car models as determined by consumer preferences.
- Product Development: Understanding which features contribute to a car's acceptability can influence future design and development, focusing on what consumers value the most.

2. Beneficiaries:

- Manufacturers: Can refine production plans and enhance car features that drive acceptability.
- Dealerships: Can better tailor their sales strategies and stock vehicles that are predicted to be more desirable, thus potentially increasing sales efficiency.
- Consumers: Benefit indirectly through products that better meet their needs and preferences.
- Marketing Teams: Can develop more effective campaigns based on predicted popular features and models.

3. Potential Disruption:

The introduction of sophisticated predictive models like XGBoost and decision trees in the car industry could lead to significant disruptions:

- Shift in Market Power: Manufacturers who effectively use these models may gain a competitive edge, potentially leading to a shift in market dynamics.
- Changes in Employment: Roles that traditionally relied on intuition and experience, such as certain sales and marketing positions, might see a shift towards more data-driven positions.
- Dealer Operations: Dealerships that adapt to a data-informed stocking strategy might outperform competitors who do not use such analytics.

4. Other Implications:

- Customer Relationships: Enhanced predictive capabilities can lead to a more personalized shopping experience, as dealerships could predict and understand customer preferences better.
- Ethical Considerations: There could be concerns about data privacy and the ethical use of consumer data in prediction models.
- Market Prediction Accuracy: Over-reliance on predictive models might risk misinterpreting market dynamics if models are not regularly updated with new data or fail to account for external variables like economic shifts or regulatory changes.

Podolean, I. (2023, August 21). Predictive analytics in the automotive industry: Opportunities and challenges. Oneest. **Was referenced for this part**