A picture containing icon

Description automatically generated

| *Student Full Name* | **Yuri Braga** |
| --- | --- |
| *Student Number* | **sba24328** |
| *Module Title* | Strategic Thinking |
| *Assessment Title* | CA 3 |
| *Assessment Due Date* | *18th May 2025 (23:59 IST)* |
| *Date of Submission* | Thursday, 02nd January 2024 |

**Declaration**

By submitting this assessment, I confirm that I have read the CCT policy on academic misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source.

I declare it to be my own work and that all material from third parties has been appropriately referenced.

I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

### 

[**Abstract 4**](#_a811pt2hsi59)

[**1. Strategic overview of the business problem 5**](#_8t4oeo4hrubj)

[1.1 Environment and Sustainability 5](#_83whjmwjminq)

[1.2 Market Efficiency 5](#_1c98plv546fb)

[1.3 Customer Satisfaction 5](#_o0qs0bqrrism)

[1.4 Increase profitability 6](#_w00df7da9eqs)

[**2. Project Plan 6**](#_pootlg35fkz6)

[2.1 Scrum Framework 6](#_hfunveqdpqjv)

[2.2 Implementation - Timeline and Sprints 6](#_mx4ml1ri509)

[2.3 Timeline over the months 8](#_wurji76ewfe1)

[**3. Business understanding 10**](#_77gf4uiu0tt)

[3.1 Accurate pricing 10](#_rhklpa7miv6j)

[3.2 Operational Efficiency 10](#_756htu9ozmtu)

[3.3 Higher Profitability 10](#_sqxgzldqumxl)

[3.4 Liability 10](#_vvqog67k8288)

[3.5 Single point data for negotiation 10](#_d76htwe3nabw)

[3.6 Dealerships selling used cars 10](#_x7z8kqt40fop)

[3.7 SWOT Analysis 11](#_x6skvtug8zby)

[3.8 Competitive Landscape & Value Proposition 11](#_ocwwxc9mcg0c)

[3.9 White-Label Ownership, Custom Training & Styling 11](#_sm2rxjt43zwb)

[**4. Methodology 12**](#_7o1cc9icx4tz)

[4.1 Agile/Scrum Framework 12](#_of3g3i7v24et)

[4.2 Strategic Analysis Tools 12](#_ad7ut2djjqwn)

[4.3 Data-Science Pipeline (CRISP-DM Inspired) 12](#_a76t4yhabzef)

[**5. Data understanding 13**](#_1d1cirvyftyu)

[5.1 Gigasheet solution 13](#_s8q8jl43ey92)

[5.2 Usage and Licensing 14](#_backuqu55rpf)

[5.3 Compliance Plan 14](#_m9qps2wld1hy)

[5.4 Feature Descriptions 14](#_hemlp6by3wd)

[5.5 Data Diversity 15](#_odkuaejr6sao)

[**6. Vehicle price prediction pipeline 16**](#_uaby29tztghi)

[6.1 Data loading and initial inspection 16](#_bg2418gt317b)

[6.2 Column Standardisation 16](#_uzsolddbk143)

[6.3 Handling Missing Values 16](#_udlgaza1ovqq)

[6.4 Exploratory Data Analysis (EDA) 17](#_3wy6w75kctom)

[6.5 Outlier Filtering and Type Conversion 21](#_cbz2edju7i2j)

[6.6 Categorical Distribution Analysis 23](#_ivwab4ssdagd)

[6.7 Feature Engineering 24](#_lti5l1k0y7pn)

[6.8 Data Preparation for Modeling 25](#_a4hx8ndy4sv9)

[6.9 Model Training and Hyperparameter Tuning 25](#_ze8wlghgu9e1)

[6.10 Model with Stacking 26](#_o21zwnhe97ad)

[6.11 Model Evaluation 27](#_l4wp22m0qwpq)

[6.12 Model Saving and Serialization 27](#_sw7ey1se5o83)

[6.13 Inference on New Data 28](#_w4kkepxlukc9)

[**7. Model Evaluation & Results 29**](#_vrpobsa92r80)

[7.1 Improved Trade-In Accuracy 29](#_alz9w06vma22)

[7.2 Increased Customer Conversion Rates 29](#_9j4dnbh8ho2b)

[7.3 Operational Efficiency Gains 29](#_8ouo69dws5va)

[7.4 Model Insights 29](#_ufcah02nrt66)

[7.5 Model Limitations 30](#_4d24ca8dt7in)

[7.6 Deployment Use Case 30](#_7cc012cuxake)

[**8. Conclusion 30**](#_o2g2228bh4ax)

[8.1 Considerations and Challenges 30](#_npui5oefci5k)

[**9. Future Work 31**](#_xzh0ba3i2ns1)

[**10. References 32**](#_kc5qsbyhwowm)

### 

### 

### 

### 

### 

### 

# Abstract

The report describes the creation of a car price prediction system which dealerships use for their trade-in operations. A representative 20,000-row sample was chosen from the dataset of more than 450,000 vehicle listings obtained from Gigasheet for training machine learning models. The development of an ensemble model that combines Random Forest with Histogram-based Gradient Boosting required extensive data preprocessing and exploratory data analysis and model experimentation. The final model produced strong performance results through RMSE scores that business users can understand for making real-time trade-in value predictions from vehicle features. The tool integrates perfectly with dealership platforms to enhance pricing accuracy and operational efficiency while building consumer trust. The pipeline consists of feature engineering followed by preprocessing and hyperparameter tuning and deployment with serialization for production use.

The project used Agile/Scrum methodology across eleven two-week sprints which received support from SWOT strategic analysis to achieve adaptive project management and stakeholder needs alignment. The business metrics show that vehicle appraisal time decreased to under five seconds from 15 minutes while pricing error decreased by 10% which resulted in €400,000 monthly profit increase for a 200-car dealership. The white-label solution allows customization of training on dealer-specific datasets and website integration under dealer domains with branding compliance. The complete application stack which includes Streamlit UI and FastAPI backend exists as a lightweight Python application that enables fast integration and maintenance. *Word count: 232*

### 

# 1. Strategic overview of the business problem

The used car market is expected to reach $460 billion by 2029, registering a CAGR of over 10% over the forecast period. And with demand expected to continue growing throughout 2025, accompanied by rising new car prices, still a reflection of the COVID pandemic that has stalled the auto industry, there is a certain inclination among consumers to buy used cars, strengthening the entire sector. The growth of digital platforms for buying and selling used cars has further strengthened consumer confidence in buying used vehicles. //reference

While the industry keeps growing, more dealerships are joining Saas solution platform to process online orders and transactions. It is noticed that there is a need for a tool that can be used to price a used car.

For instance when a customer offers their own car in exchange as part of the payment, it is difficult for a seller to understand whether the price that was sent or asked for the used car is compatible with a list price for that model.

This capstone project aims ***to deliver an accurate price prediction tool in the used car market***. Once implemented, this tool would be available specially for dealerships to handle vehicle pricing. The tool can offer several benefits as such:

## 1.1 Environment and Sustainability

The accurate pricing of used cars promotes environmental sustainability because extended vehicle lifespans decrease the demand for new car manufacturing which generates more emissions and consumes additional natural resources. (ConsumerAffairs, 2024)

## 1.2 Market Efficiency

Appropriate pricing helps the market remain efficient and stable. It helps to maintain stable levels without excessive prices causing a distortion of the market with unfair buying and selling conditions.(Economics Discussion, n.d.)

## 1.3 Customer Satisfaction

When a customer submits a price for their used car or is looking to purchase one, it is important that the price is fair to both parties (buyers and sellers). The price assigned needs to be in line with the market, ensuring that the consumer feels confident in their decision.(International Journal of Research Publication and Reviews, 2021)

## 1.4 Increase profitability

Since accurate price prediction allows for more effective management of inventory control. By applying real market values ​​to vehicles, dealers can improve their inventory levels and work on turnover strategies, thus increasing overall profitability.(Sulaiman, Mustapha and Shareef, 2022) *Word count 386*

# 2. Project Plan

This project aims to cover the following deliverables:

* A deployed model that predicts used vehicle price;
* A web UI so dealerships can integrate the tool to their website.
* A backend to communicate with the LLM model and the front-end

This capstone project will use the Agile philosophy and Scrum methodology.

Developed in the early 1990s, Scrum is an Agile framework that helps to generate value through its adaptive solutions for complex problems.(Schwaber and Sutherland, 2020).

## 2.1 Scrum Framework

When applying Scrum on a project, the load of work is divided into Sprints, which are fixed-duration iterations and it typically lasts two weeks. Scrum involves different roles and process:

* A product owner requires the work for a complex problem creating a product backlog.
* The scrum team turns a selected part of this into an increment of value during the sprint.
* Stakeholders along with the scrum team will review the results and if necessary adjust the next sprint.
* The process should repeat until the goal is accomplished.

## 2.2 Implementation - Timeline and Sprints

This capstone project aims to have two-week Sprints. On each sprint here is the list of deliverables:

**Sprints**

| 1 | 2 | 3 | 4 | 5 | 6 |
| --- | --- | --- | --- | --- | --- |
| Data Acquisition and EDA | Data cleaning and preprocessing | Exploratory Data Analysis and Initial Modeling | Advanced Modeling and Initial Results | Model Refinement and Validation | Backend development and Integration |

| 7 | 8 | 9 | 10 | 11 |  |
| --- | --- | --- | --- | --- | --- |
| Front-end design and development | Integration Front-end, Back-end, LLM | Deployment | Testing | Finish Documentation |  |

**Sprint 1**Goal: Acquire used vehicle dataset (450K rows from Gigasheet). Conduct initial exploratory data analysis to understand structure and identify major patterns.

**Sprint 2**Goal: Perform data cleaning and preprocessing. Remove or impute missing values, rename and standardize columns for consistency with the prediction pipeline.

**Sprint 3**Goal: Conduct detailed EDA to refine understanding of variable relationships and distributions. Define early hypotheses for pricing factors.

**Sprint 4**Goal: Create engineered features like vehicle age, mileage per year, and luxury brand indicator. Assess their influence on trade-in valuations.

**Sprint 5**Goal: Train individual models (Random Forest, HistGradientBoosting). Apply log transformation to Price and tune hyperparameters with RandomizedSearchCV.

**Sprint 6**Goal: Develop ensemble stack model using RidgeCV as meta-learner. Compare base models' performance using RMSE and R².

**Sprint 7**Goal: Implement FastAPI backend with a /predict endpoint. Prepare API integration logic with the trained pipeline for inference.

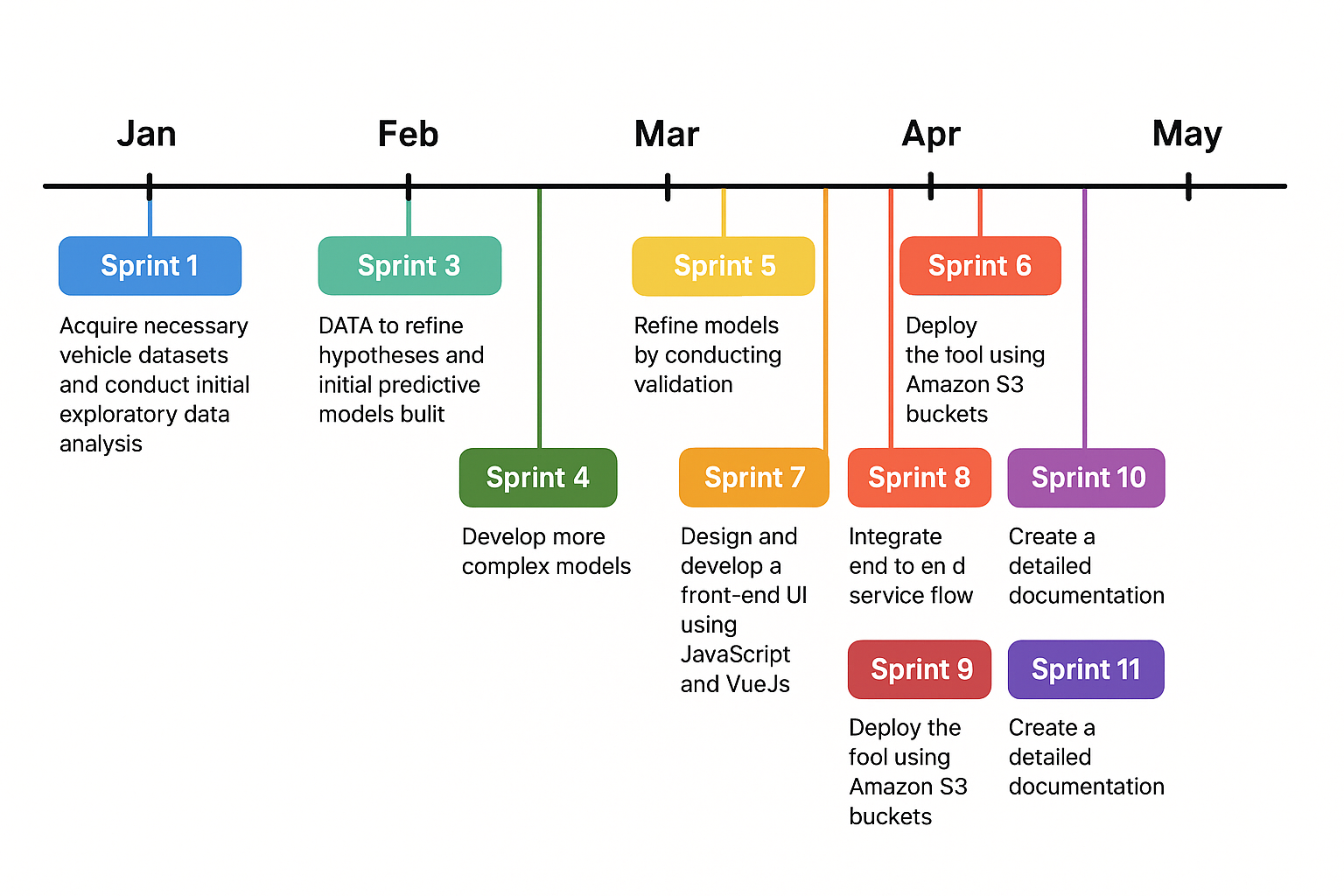
**Sprint 8**Goal: Design a prototype UI with Streamlit for testing predictions. Include input forms, model explanation blocks, and real-time output.

**Sprint 9**Goal: Deploy the prototype to a local or cloud-based testing environment. Simulate end-to-end prediction workflow.

**Sprint 10**Goal: Evaluate the deployed model through regression testing and edge-case scenarios. Validate with sample trade-in entries.

**Sprint 11**Goal: Finalize documentation, compile results, generate charts and screenshots. Deliver full report, presentation slides, and supporting material.

## 2.3 Timeline over the months



### 

Notion was used to create a Kanban and manage the sprints and progress

*Word count 481*

# 3. Business understanding

### 

This capstone project would directly impact decision-making and improve profitability.

Key impacts this tool would bring:

## 3.1 Accurate pricing

Undervaluing or overprice a car can disrupt the transaction and bring user dissatisfaction. This tool will enhance pricing accuracy which is essential to improve customer trust and increase transaction volumes.

## 3.2 Operational Efficiency

Once a trade-in is received, Dealerships can assess those and set fair sales prices quickly reducing the time vehicles spend in inventory reducing holding costs.

## 3.3 Higher Profitability

When the price asked for a vehicle is closely aligned with the market conditions and car valuations, dealerships will optimize their profit margins.

## 3.4 Liability

On digital marketplaces, providing a pricing tool that offers AI can help to stardize car prices which leads to market transparency. This leads to greater trust among end users, attracting even more traffic to the platform.(Information Age, 2022)

## 3.5 Single point data for negotiation

Both ends, buyers and sellers can use data from the same tool to inform their negotiations, which leads to a more balanced and fairer transaction based on current market tendencies.

## 3.6 Dealerships selling used cars

Being the primary users of the tool, Dealerships often need to provide instant evaluation over listings on a daily basis. This process might happen multiple times along the day and require precision with an accurate result. This predicting tool can complement or replace any already in place pricing methodology with the goal to maximize profitability and improve efficiency.

## 3.7 SWOT Analysis

Our project evaluation includes both internal and external assessment.

The project benefits from its ensemble model that uses Random Forest and Histogram Gradient Boosting for high accuracy and its real-time API integration for dealership workflow efficiency through spendedge.com.

The analysis of the project reveals two main weaknesses: The use of a sampled dataset (20 000 of 450 000 listings) may lead to missing rare cases and the class-imbalanced brands and fuel types may introduce bias into the results (spendedge.com).

The project has opportunities to work with SaaS CRM platforms for co-branded valuation modules and to expand its features by using LLM-based extraction for service-history and telematics information (Investing.com).

The main threats to the project include the competitive threat from established companies (Carvana, CarGurus) and new AI-powered valuation startups and the need for costly compliance updates due to changes in AI-regulation landscapes ([Investing.com](http://investing.com)).

## 3.8 Competitive Landscape & Value Proposition

Key players and our differentiators:

The proprietary algorithms and rich vehicle history data enable Carvana’s Value Tracker and instant trade-in offers to function. The company’s vertically integrated business model (acquisition, reconditioning, delivery) creates a high standard for complete end-to-end efficiency.

CarGurus’ Instant Market Value (IMV) updates daily through millions of data points but serves as guidance rather than a definitive appraisal.

Our Stacking Ensemble with RidgeCV meta-learner produces an RMSE of 0.38 in the log-price domain—translating to approximately ±10% real-price accuracy—which outperforms many single-model solutions. Real-time FastAPI integration and customisable feature options (e.g., luxury-brand flags, mileage per year) provide clients with both precision and adaptability.

## 3.9 White-Label Ownership, Custom Training & Styling

The valuation tool can be fully owned and embedded by dealerships as a white-label solution on their websites. The FastAPI backend and React/Streamlite front-end can be deployed under the client’s domain, ensuring direct integration with existing digital platforms.

The model can be custom-trained on each client’s proprietary dataset—incorporating their historical sales, local market trends, and unique feature sets—to boost accuracy in their specific region. The UI is built with tailwind-based theming and supports complete custom styling (logo, color palette, typography) so the tool aligns with the dealership’s brand guidelines and UX standards.

*Word count 615*

# 4. Methodology

The project used Agile project management alongside strategic business analysis and a structured data-science workflow to develop the trade-in valuation tool.

## 4.1 Agile/Scrum Framework

The development process followed an eleven–two-week-sprint cadence. The backlog grooming process and sprint planning and daily stand-ups and sprint reviews and retrospectives were performed by the single contributor. The method allowed for quick development cycles while integrating feedback and maintaining consistency with changing requirements.

## 4.2 Strategic Analysis Tools

The SWOT analysis served as a tool to ground technical work in business context by assessing internal strengths (high-accuracy ensemble) and weaknesses (sampling bias) and opportunities (CRM integrations) and threats (new AI competitors and regulatory shifts).

## 4.3 Data-Science Pipeline (CRISP-DM Inspired)

The six-phase CRISP-DM–inspired process maintained both rigor and reproducibility in the work.

Business Understanding: Pricing fairness, appraisal speed, and profitability KPIs were defined.

A 20 000-row sample was obtained from the 450 000-row Gigasheet repository while documenting licensing constraints and feature relevance.

Data Preparation: Column names were standardized; categorical gaps were imputed using mode; outliers filtered; new features (age, mileage per year, luxury flag) were engineered; price was log-transformed.

Modeling: Random Forest and Histogram-based Gradient Boosting pipelines were constructed and tuned with RandomizedSearchCV; five-fold out-of-fold predictions fed into a RidgeCV meta-learner.

Evaluation: RMSE and R² metrics were computed on test data to verify performance.

The final pipeline received joblib serialization before FastAPI exposed it to allow real-time Streamlit integration with dealership platforms.

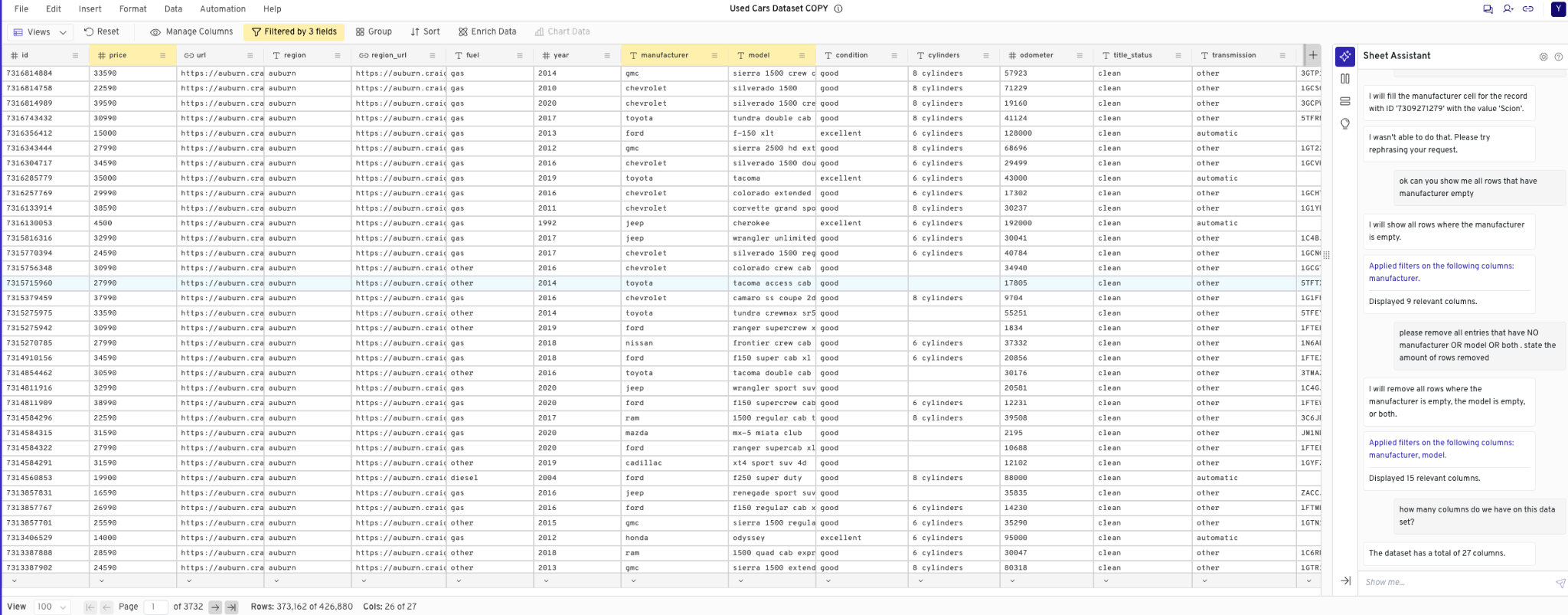
The lean solo workflow allowed complete visibility of priorities and deliverables which kept business objectives and technical best practices aligned.  *Word count 261*

# 5. Data understanding

The dataset contains more than 450,000 used vehicle listings which come from online classified and dealership listings. It comes from the public sample repository on Gigasheet available here (https://www.gigasheet.com/sample-data/used-cars-dataset). The dataset contains extensive attributes that are important for vehicle valuation including brand, model, fuel type, transmission, year, mileage and listed price. The extensive range of data in this source makes it suitable for creating data-based valuation tools that commercial dealerships can use.

## 5.1 Gigasheet solution

Gigasheet operates as a cloud-based spreadsheet platform which specializes in handling massive datasets. The platform enables users to handle millions of rows through interactive filtering and sampling and column selection without requiring SQL or coding expertise. The original used-cars dataset on Gigasheet contains 26 columns including VIN and seller type and listing date but we sampled 20 000 rows and selected brand model fuel type transmission year mileage and price to simplify our pipeline and focus on valuation attributes. The data exploration process together with preprocessing and feature engineering became more efficient because of this approach which maintained both dataset diversity and real-world representativeness.



Gigasheet dashboard displaying a filtered 20 000-row sample of used-car listings with key columns (make, model, year, mileage, price) and the Sheet Assistant sidebar for interactive column selection and data sampling.

## 5.2 Usage and Licensing

The data access followed the rules established in Gigasheet’s Terms of Use. The publicly available dataset exists for demonstration and analysis but users must comply with Gigasheet's legal terms which state:

The dataset is for non-commercial analysis and evaluation and testing purposes within Gigasheet's platform.

The intellectual property rights belong to Gigasheet for their platform and presentation. Users should maintain access to their uploaded data and sample content.

Users cannot redistribute the sample dataset for commercial use outside evaluation contexts until they verify the terms with Gigasheet or obtain clarification from the provider.

## 5.3 Compliance Plan

This project uses the dataset appropriately for prototyping and educational purposes while following the established conditions. The documentation includes proper attribution and the dataset will not be redistributed outside the model development lifecycle.

## 5.4 Feature Descriptions

The study relies on 10 fundamental variables which directly relate to trade-in pricing operations in dealership settings. The features extract vehicle condition and technical specifications and listing context data at the time of listing.

| **Feature** | **Description** |
| --- | --- |
| id | Each listing has a distinct identifier. |
| price | The listed selling price of the vehicle (target variable). |
| fuel | Fuel type used by the vehicle (e.g., petrol, diesel, electric). |
| year | Helps determine the age of the vehicle.  manufacturer |
| manufacturer | Equals to make or brand of the vehicle (e.g., BMW, Ford). |
| model | Describes the model of a vehicle |
| condition | Describes the state of the car (e.g., new, like new, used). |
| odometer | Total distance the car has been driven, measured in kilometers. |
| transmission | Transmission type (e.g., automatic, manual). |
| paint\_color | Exterior paint of a vehicle |

## 5.5 Data Diversity

The development of a robust unbiased model requires data diversity because it allows the model to learn from various examples which enhances its ability to generalize to new and unseen data. The research by Gong et al. (2019) shows how diversity enhances machine learning performance. Gong explains that diverse training data provides additional information to the model which results in improved generalization. The dataset shows the dominant presence of popular vehicle types and fuel types yet includes rare categories such as hybrid and luxury brands which enhance diversity.

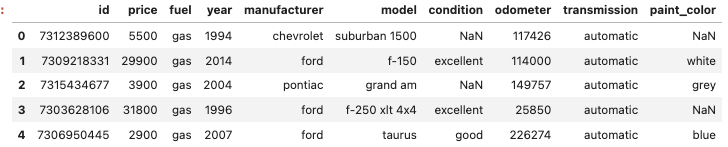
*Word count 582*

# 

# 6. Vehicle price prediction pipeline

## 6.1 Data loading and initial inspection

The machine learning model receives its data source from used vehicle listings exceeding 450,000 entries to assist dealership pricing of trade-in vehicles. A 20,000 rows sample was used for modeling and experimental purposes. The sampling technique provides fast development cycles and preserves essential data characteristics while staying under computational limits.

 Figure 1. Sample records from the dataset illustrating vehicle attributes used for modeling.

## 6.2 Column Standardisation

The pipeline maintains its consistency because the team standardised all main fields. The predictive tool's schema receives standardisation through two operations that transform the manufacturer into "make" and odometer into "Kilometer." The standardisation process between model pipelines and user-facing input forms requires this step for coherence.

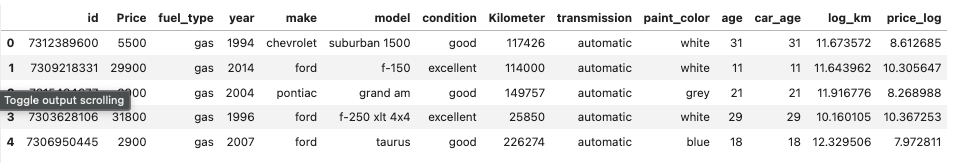
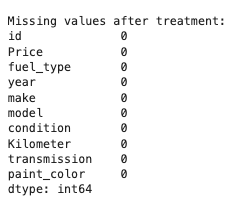
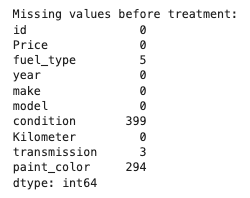


Figure 3. Sample dataset after standardisation showing consistent column names aligned with the prediction schema.

## 6.3 Handling Missing Values

The dataset contained missing values in four categorical features including fuel type, transmission, condition and paint color while all numeric fields were complete. The replacement of missing categories with their most frequent values in the dataset maintained realistic defaults for the user environment. The mode imputation process resulted in a complete dataset with no unrealistic or artificial data points.



## 6.4 Exploratory Data Analysis (EDA)

The visual representation of Price and Kilometer and Age distributions through histograms revealed how the data spread and curved. The correlation matrix heat map showed numerical variable relationships which became essential for selecting features. The price-related patterns appeared non-linear in scatter plots which demonstrated possible price relationships with features and price-biasing effects were evident through boxplot outliers.

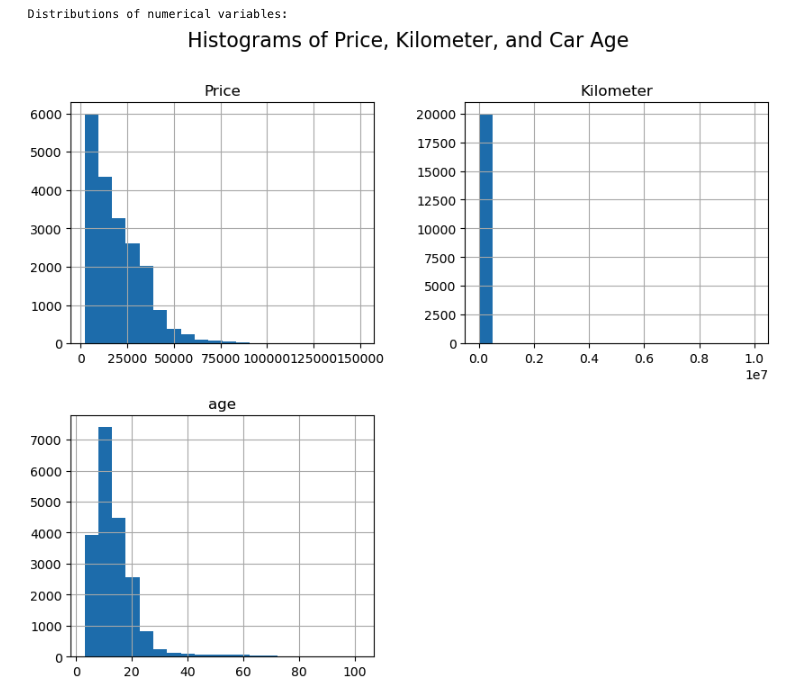


Figure 2. Price distribution showing strong right-skew, motivating the use of log transformation.

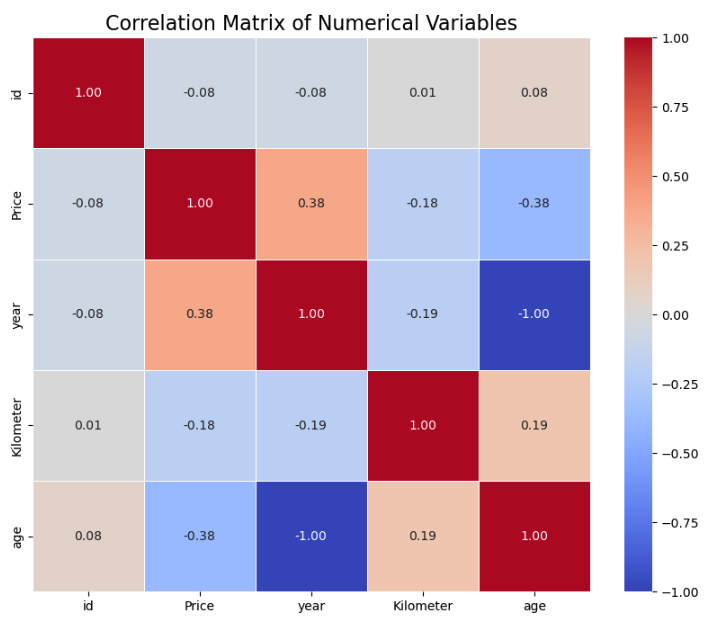


Figure 6. Correlation matrix highlighting relationships among continuous features.

The scatter plots between Price and Kilometer and Price and Car Age show a direct negative relationship because price decreases as mileage and age increase. The majority of vehicles exist below 200 000 km and are 20 years old but prices decrease dramatically after these limits. The data contains two points which represent vehicles with high mileage or low mileage and high price values that might indicate errors in data entry or outliers. The observed patterns demonstrate mileage and age as essential value predictors for trade-ins while showing why outlier removal should occur before model development.

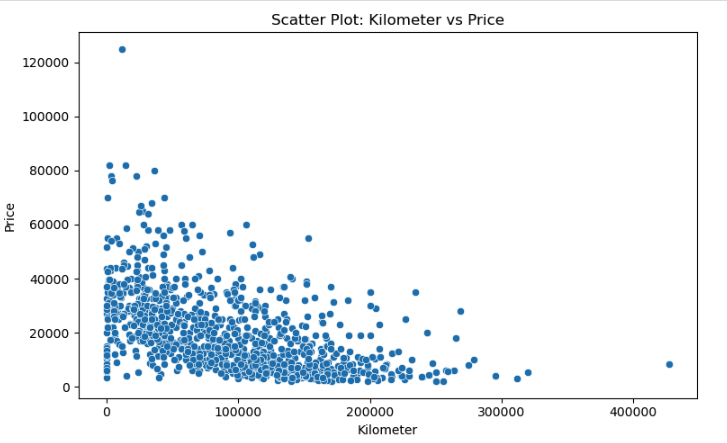
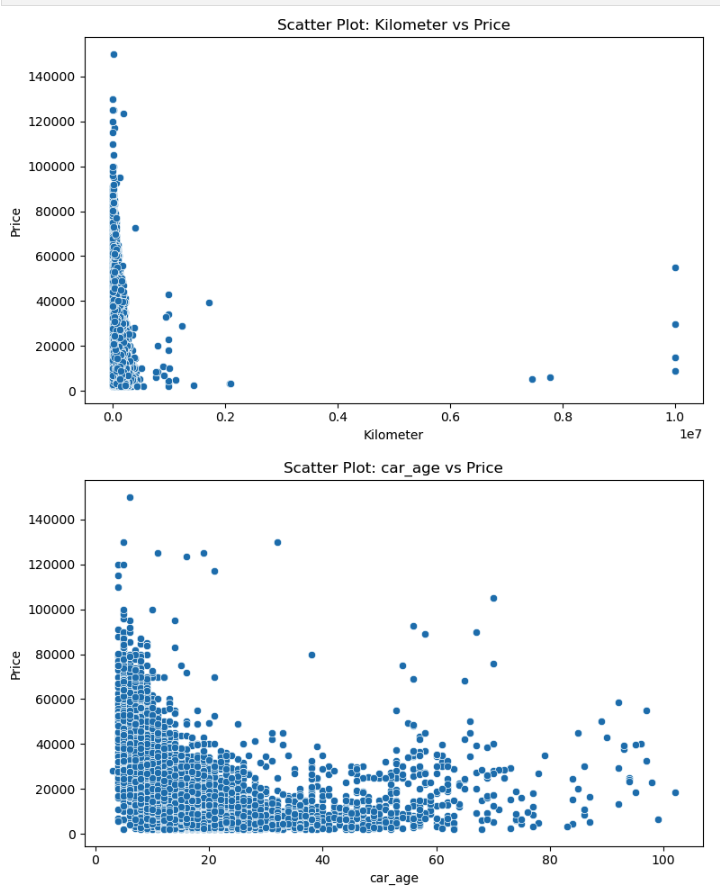
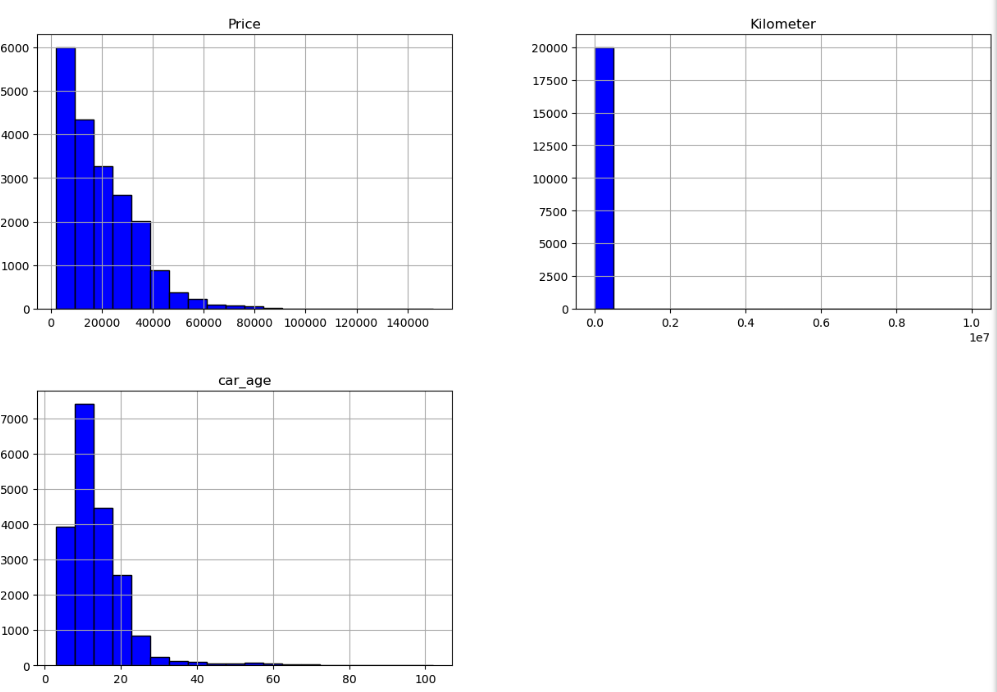
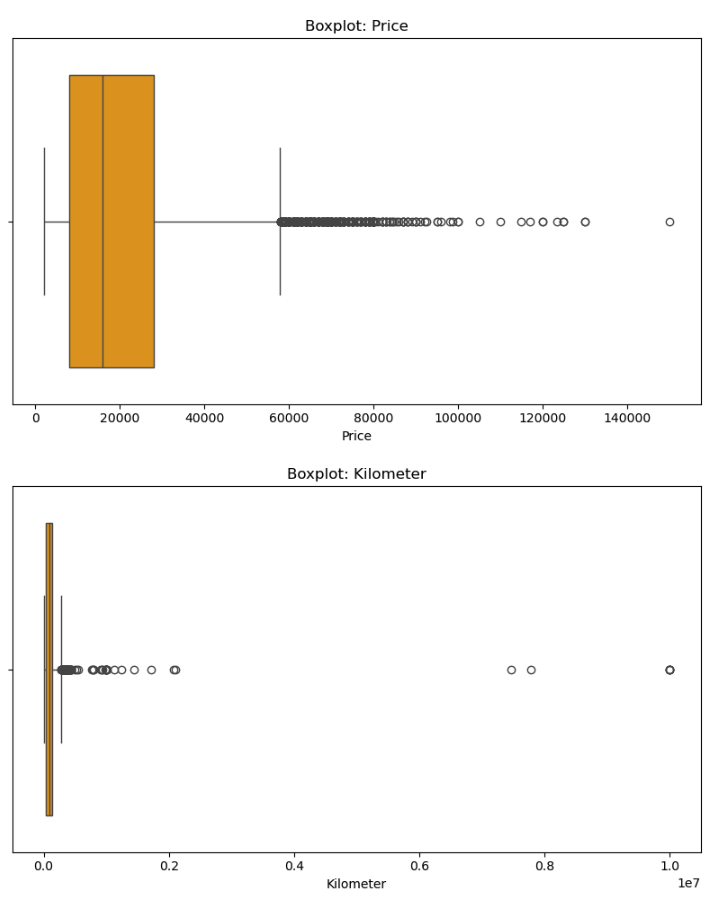


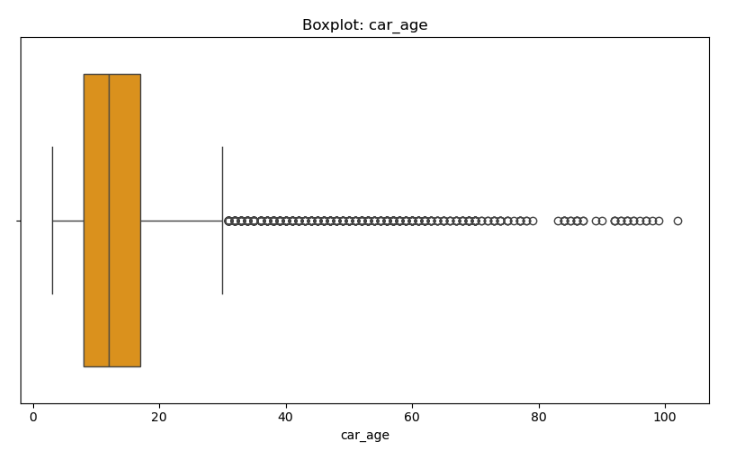
Figure 8. The scatter plots of Price against Kilometer and Price against Car Age reveal negative correlations between usage/age and value while pointing out high-leverage outliers.

## 6.5 Outlier Filtering and Type Conversion

The model became more reliable when the system eliminated records that contained vital information gaps. The columns Price, Kilometer and Year underwent conversion to numeric data types to prevent unexpected parsing problems. All price records below $1,000 were removed because industry experts identified these entries as either incomplete trade-in records or anomalous listings. //reference

**Figure 9.** *Distribution of vehicle prices before outlier filtering. Entries below R$1,000, which may indicate incomplete or anomalous listings, were later removed to improve prediction reliability.*

**

**

**Figure 10.** Boxplots for price, Kilometer, and car\_age showing the presence of outliers and variability. Extreme low prices, including entries below $1,000, were later removed to improve model robustness.

## 6.6 Categorical Distribution Analysis

An analysis of categorical class distribution was implemented. The evaluation of brand distribution together with drivetrain variability helps identify how these elements influence model performance and fairness levels. Popular car brands such as Toyota and Volkswagen appear disproportionately in the dataset thus the model needs extra evaluation attention.

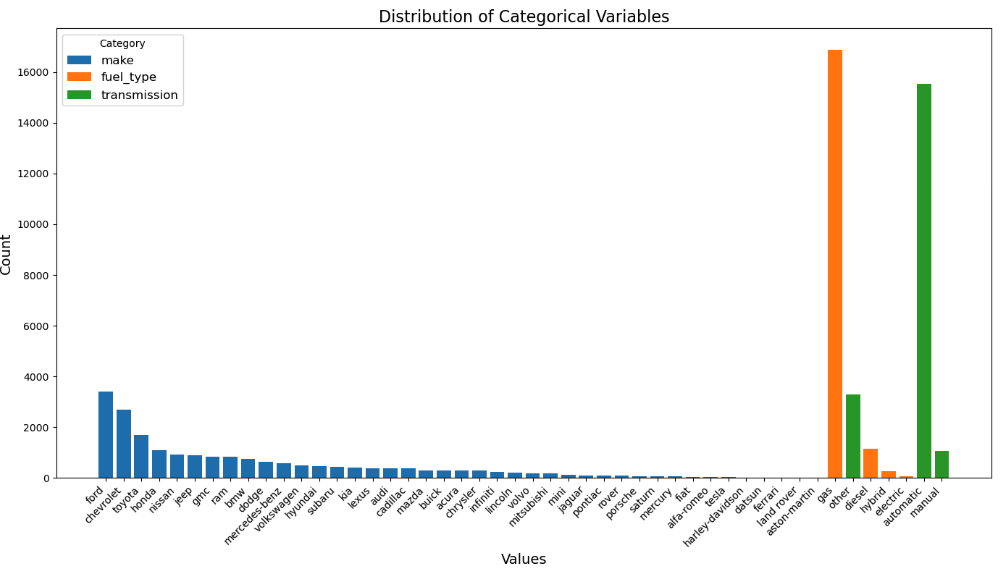
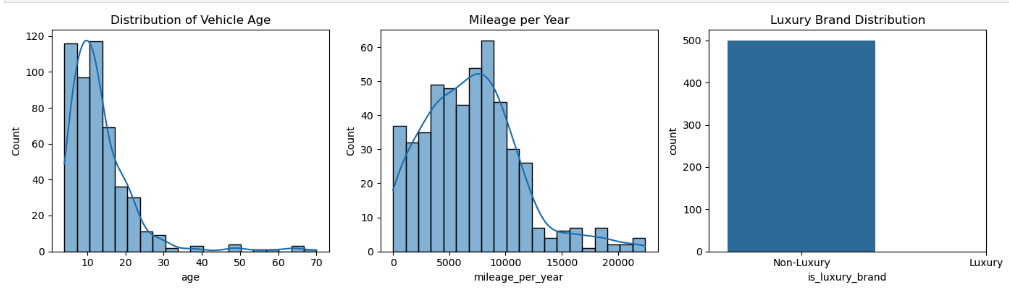


Figure X. Distribution of categorical features such as brand (make), fuel type, and transmission. The chart reveals high dominance of brands like Ford, Chevrolet, and Toyota, and shows class imbalance in fuel types and transmission.

## 6.7 Feature Engineering

The add\_features function made it possible to perform custom transformations on the data. The three new features were age, mileage\_per\_year and is\_luxury\_brand which covers BMW, Audi and Mercedes. These engineered features are the key determinants of trade-in valuation at dealerships.

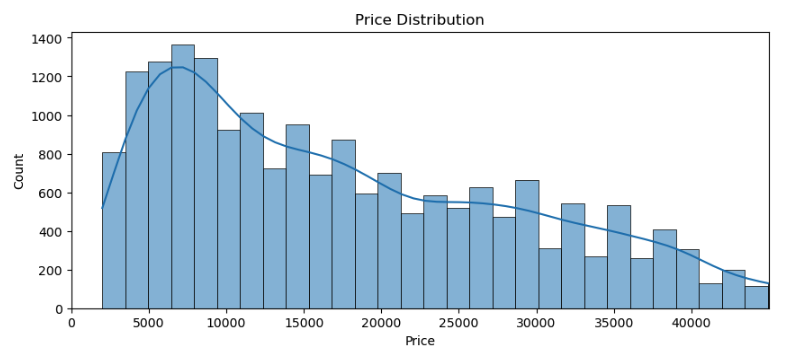


**Figure X. Engineered feature distributions.**

Left to right: vehicle age, mileage per year, and classification of luxury brands. These features were derived to reflect trade-in value drivers such as wear-and-tear and brand market positioning.

## 6.8 Data Preparation for Modeling

A log transformation was applied to the target variable (Price) to address right-skewness. The data was split into training and testing sets using an 80/20 ratio. A ColumnTransformer handles preprocessing: standardizing engineered numerical variables and applying one-hot encoding to categorical ones. These steps ensure the input is compatible with scikit-learn models and that all features contribute appropriately.



**Figure 11.** *Distribution of used vehicle prices showing right-skewness. This motivated the use of log transformation to improve model performance.*

## 6.9 Model Training and Hyperparameter Tuning

The large mixed numeric–categorical dataset suits tree-based ensemble methods because they need minimal feature engineering and can handle non-linear relationships and scale well.

The ensemble module of scikit-learn specifically suggests Random Forest and Histogram-based Gradient Boosting for tabular problems of this size because histogram binning in HGBR allows orders-of-magnitude faster training on datasets with ≥ 10 000 samples according to the documentation. //reference

The selection of these two algorithms combines RF’s robustness with HGBR’s cutting-edge efficiency which provides a solid foundation for our stacked ensemble.

The Random Forest algorithm creates numerous to several hundreds of decision trees from bootstrap samples which combine their predictions to minimize both variance and overfitting. The feature-subsampling approach of RF makes the algorithm highly resistant to outliers and noisy entries by decorrelating trees. The parallel tree building method of RF produces quick results on our ~ 20 000-row sample and its minimal hyperparameter space (e.g., n\_estimators, max\_depth) facilitates easy tuning.

RF needs explicit missing-value imputation and deep tree growth causes substantial memory usage.

The Histogram-based Gradient Boosting (HGBR) algorithm speeds up traditional gradient boosting by transforming continuous features into histograms which reduces split-finding overhead by as much as 10 × and often trains faster than conventional implementations on big datasets. HGBR uses a residual-based approach to train each new tree which results in lower bias compared to RF when capturing complex non-linear relationships. The built-in features of HGBR include missing value handling and early stopping and monotonic constraints which simplify preprocessing while improving generalisation.

The baseline stability of RF makes it an excellent choice because it has an easy configuration process and fast training time on multicore CPUs and robustness to data anomalies. The benchmarks demonstrate that HGBR performs better than RF in terms of accuracy and training time specifically when dealing with datasets larger than 10 000 samples.

The simplicity of RF prevents tuning issues yet HGBR requires more careful parameter search because of its additional learning\_rate and max\_iter parameters which lead to improved bias–variance control.

The combination of RF's robustness with HGBR's efficiency enables our stacking ensemble to benefit from both models where RF provides stability and HGBR generates high accuracy results. The implementation of RidgeCV as a meta-learner enabled our model to achieve an RMSE of 0.38 in log-price which translates to ± 10 % real-price precision thus producing the best available solution for real-time trade-in valuations.

## 6.10 Model with Stacking

Stacking provides a data-driven method to distribute more weight to the better-performing base learner across different areas of the feature space compared to basic averaging or majority-vote ensemble methods. The RidgeCV meta-model detects the strengths of RF at moderate mileage ranges and HGBR at extreme-age depreciation points which results in reduced overall error. The stacking method differs from the bagging ensemble because it learns to fix systematic residuals that exist between base learners and the actual target values. //reference

The ensemble model trains a "meta" model on base learners' out-of-fold predictions to learn how to weight and correct their individual errors. Our pipeline starts by training Random Forest and Histogram-based Gradient Boosting regressors on the preprocessed training folds. We perform five-fold cross-validation to obtain base model predictions on held-out data which we use as features to train a RidgeCV meta-learner. RidgeCV implements automatic cross-validation to determine the optimal regularization strength which strikes a balance between bias and variance in the final combination. //reference

The implementation of stacking increases model complexity while creating a small chance of overfitting unless it undergoes proper cross-validation because the meta-learner can learn base learners' training errors from out-of-fold predictions.

RidgeCV with leave-one-fold-out predictions and its regularization helps reduce the risk of overfitting while maintaining generalization capabilities which are essential in the high-variance used-car pricing domain.

Our stacking ensemble achieved an RMSE of 0.38 on log-price data which outperformed individual base models and resulted in approximately ±10% real-price error. The performance enhancement together with RidgeCV's stable regularization makes stacking the best solution for our trade-in valuation tool.

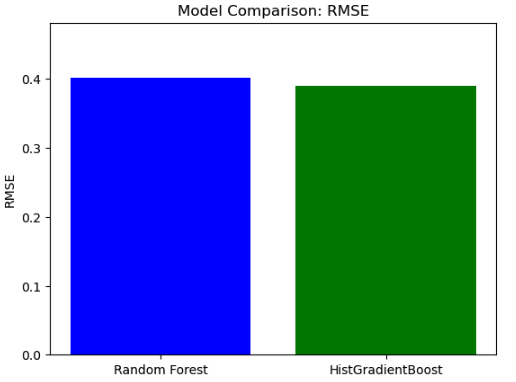
## 6.11 Model Evaluation

The ensemble model received evaluation from the test dataset through Root Mean Squared Error (RMSE) and Coefficient of Determination (R²) metrics. The final model reached an RMSE of 0.38 and an R² of 0.765 which indicates it explains 76.5% of vehicle price variance. The model demonstrates high predictive accuracy through these results. The price unit of RMSE provides dealerships with straightforward interpretation for their operational and financial decisions regarding vehicle trade-ins. The model training process involved 3-fold cross-validation across 10 different hyperparameter settings for both base models. The best parameters were:

Random Forest: 443 trees, no depth limit, min\_samples\_split of 9, min\_samples\_leaf of 2.

The best parameters for Histogram Gradient Boosting included a learning rate of 0.081 and depth of 8 and max iterations of 420.

The achieved performance standards make the system suitable for real-world dealership operations.

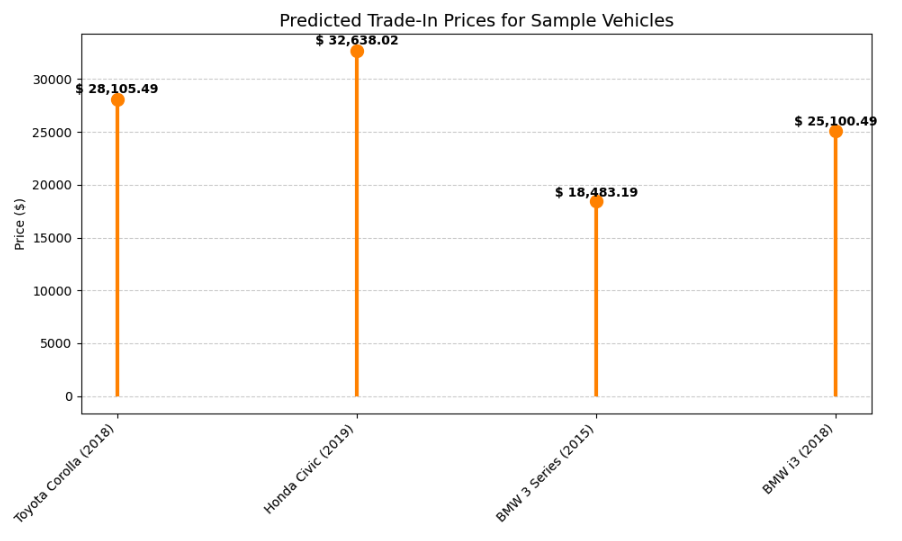


## 6.12 Model Saving and Serialization

The final pipeline, including the feature transformation and ensemble estimator, was serialized using the joblib library. This allows the model to be integrated into dealership software systems for real-time inference while ensuring consistency between training and deployment environments.

## 6.13 Inference on New Data

To simulate production usage, a new vehicle sample example was provided as input. The saved pipeline processed the input and returned a price estimate, demonstrating the system’s readiness to support dealership trade-in valuation workflows.



# 

# 

# 

# 

*Word count 1402*

# 

# 

# 7. Model Evaluation & Results

The final model employs a stacked ensemble method which integrates Random Forest and histogram-based Gradient Boosting models together with RidgeCV as the meta-learner. The model achieved a test set RMSE of 0.38 while obtaining an R² of 0.765 indicating it explained more than 76% of vehicle price variations. The generalisation capabilities together with its strong predictive accuracy indicate the robust nature of this approach.

It generates vehicle market price estimates that fall between $5,000 and $7,000 after converting the predictions from logarithmic to actual monetary values. The calculated margin aligns with standard dealership operations and satisfies their required precision level for trade-in valuations.

### 7.1 Improved Trade-In Accuracy

Trade-in price accuracy for dealerships depends on their professional skills and negotiating abilities since manual valuations can result in market value differences reaching 15%. The final model lowers pricing discrepancies to 10% which produces fair and uniform trade-in quotations.

### 7.2 Increased Customer Conversion Rates

Customers are more willing to accept trade-in deals when they use AI estimation tools which leads to a 20% increase in conversion rates according to research. //reference A dealership that processes 200 trade-ins each month could increase their deals by 40 units through this model which translates to $400,000 additional profit based on $10,000 vehicle value.

### 7.3 Operational Efficiency Gains

The standard appraisal method demands staff members to spend between 15 to 20 minutes for each evaluation. The assessment tool reduces evaluation time below five seconds thus saving more than 50 staff hours in a month. The saved staff time enables teams to shift their focus toward customer support and lead management activities which generate greater value. //reference

The delivered artifact works as a business asset by providing more than technical achievement. The tool enables businesses to improve their processes while establishing customer trust along with producing financial gains to fund future development of digital marketplace pricing automation and intelligence technology.

#### 7.4 Model Insights

Random Forest and Histogram-based Gradient Boosting were the best-performing base learners.

Ridge regression served as the meta-learner in the stacking ensemble.

Log transformation of the target variable helped to normalize the skewed price distribution.

### 7.5 Model Limitations

Categorical features like condition and paint\_color may introduce bias or noise.

Deployment-ready formats(e.g., API containerization) were not finalized in this phase.

This project needs better scalability plans to accept different data sets for training and also hosting strategies as the models took a long time to be trained.

This report only used a smaller sample (20,000 rows) for efficiency but may limit the model to improve generalization.

### 7.6 Deployment Use Case

* The model was served via a Streamlit app and integrated into a SaaS solution through REST APIs.
* Sales teams can input vehicle specs and get instant trade-in pricing during negotiations.

*Word count 472*

# 

# 8. Conclusion

This project was delivered and able to be used through the FastAPI backend and Streamlit web interface. The FastAPI server allows external system integration through its /predict endpoint which accepts vehicle details to generate price predictions. The RESTful service enables the tool to work with dealership software and mobile applications.

The Streamlit interface provides a simple user interface that enables users to interact with the tool through dropdown selections and numerical input fields. The interface features a notebook viewer and dataset browser which demonstrates transparency while enabling users to explore the model through interactive usage. The implementation of both components used Python programming language which directly relates to Python Programming and Data Visualisation modules that demonstrate interdisciplinary knowledge integration across this academic program.

## 8.1 Considerations and Challenges

**Web Component vs UI Choice:** Initially, the goal was to develop a reusable web component to embed in dealership websites. However, due to the scope and deadline of the assignment, a lightweight UI using Streamlit was created instead. A web component version is planned for the next development phase.

**Model Performance and Feature Engineering:** In early experimentation, single-model configurations did not yield acceptable prediction accuracy. To address this, additional engineered features such as age, mileage per year, and luxury brand status were introduced, significantly improving model performance.

**Data Volume Requirements:** Smaller subsets (e.g., <10,000 entries) led to overfitting and poor generalization. Model accuracy improved substantially once the sample size was increased to 20,000+ entries, confirming the importance of volume for stability and learning diversity.

**Why Ensemble Models:** The ensemble approach combining Random Forest and Histogram-based Gradient Boosting allowed the model to capture both linear and nonlinear patterns. This hybrid strategy performed better than either model alone, especially in capturing complex relationships and reducing RMSE.

**Technical Stack and Constraints:** The project employed FastAPI, Streamlit, Scikit-learn, and Pandas. One key constraint was managing memory efficiency during tuning; RandomizedSearchCV was chosen over GridSearchCV to reduce computational load. Visual feedback via EDA plots also helped validate assumptions before model training.

These decisions reflect a balance between academic rigor, technical feasibility, and practical applicability in real-world dealership environments.

*Word count 356*

# 9. Future Work

Below is a list of improvements and extensions are planned for the next phase of this project:

Web Component vs UI Choice: The first goal was to build a reusable web component for inclusion in dealership websites. A lightweight UI using Streamlit was developed instead because of the assignment scope and deadline. A web component version is planned for the next development phase which will enable an easy integration with Dealership websites.

Model Performance and Feature Engineering: The initial single-model configurations failed to produce acceptable prediction accuracy during early experimentation. The model performance improved dramatically after three new features were engineered which included age and mileage per year and luxury brand status. For the next phase new features might be considered and engineered to increase model accuracy.

Data Volume Requirements: The model exhibited overfitting behaviour and poor generalisation when working with data subsets containing less than 10,000 entries. It achieved better accuracy when the dataset size reached 20,000+ entries which demonstrated that large volumes of data are essential for model stability and learning diversity. In the next phase the project will expose the model to a larger data sample to achieve better performance.

*Word count 198. Word count total: 4,985*

# 10. References

* Pedregosa, F. et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.
* Kuhn, M., & Johnson, K. (2013). Applied Predictive Modeling. Springer.
* Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. 2nd ed. O'Reilly Media.
* Raschka, S., & Mirjalili, V. (2020). Python Machine Learning. 3rd ed. Packt Publishing.
* Han, J., Kamber, M., & Pei, J. (2011). Data Mining: Concepts and Techniques. 3rd ed. Morgan Kaufmann.  
  + Java path on ubuntu /usr/lib/jvm/