Project members: Jasper Suursild, Artjom Vassiljev, Hugo Arrak
Project repository: https://github.com/JSild/Car-Price-Prediction

Business understanding

Identifying Business Goals

Background

The used-car market in Estonia is dynamic, and car buyers and sellers often struggle to understand fair market value. Auto24 is the most widely used platform for car listings, containing useful information such as brand, model, year, mileage, fuel type, engine power, and asking price. However, the pricing varies a lot across listings, making it difficult to estimate whether a particular car is appropriately priced. Our project aims to build a machine learning–based price prediction system that uses scraped Auto24 data to estimate car values based on their attributes.

Business Goals

The primary goal is to support decision-making for individuals involved in used-car transactions. This includes helping buyers avoid overpaying and helping sellers set competitive, realistic prices. From a technical perspective, the goal is to build an automated pipeline that scrapes car listings, processes features, and trains a predictive model capable of estimating market value with reasonable accuracy.

Business success criteria

Success will be judged based on:

- Accuracy and practical usefulness of the predictions
- Reliability of the scraping process
- User satisfaction, measured informally through positive feedback from test users (e.g., friends or classmates evaluating predictions).
- Model performance, such as achieving an error small enough to be meaningful in real-world pricing

Assessing the situation

Inventory of resources

- Data source: Auto24 listings (about 15 000 listings).
- Jupyter Notebook/Google Colab + Python,
- BeautifulSoup/Selenium for scraping
- Pandas for working on the dataset
- Plotnine or similar for visualization
- Machine learning libraries

Requirements, assumptions, and constraints

Requirements:

- The dataset must contain enough variety to train a generalizable model.

Assumptions:

- Car listings on Auto24 are reasonably accurate and representative of the Estonian used-car market.
- Historical prices remain useful even as the market changes gradually.

Constraints:

- Auto24 may block scraping or change HTML structure.
- Some listings may contain missing, inconsistent, or noisy data.
- Time constraints for scraping sufficiently large datasets.

Risks and contingencies

- Scraping blocked: Implement rate limiting or user-agent rotation
- Poor data quality: Develop cleaning pipelines and feature engineering strategies.
- Weak model performance: Try more features, better preprocessing, or more advanced models.

Terminology

- Listing: A single vehicle advertisement on Auto24.
- Target variable: Car's asking price.
- Features: Car attributes such as brand, model, year, mileage, gearbox, fuel type, etc.
- MAE/MSE: Error metrics used to evaluate price prediction accuracy.

Costs and benefits

Costs:

- Development time, data scraping time, computational resources, and potential website access limitations.

Benefits:

- A reliable tool to assess car prices.
- Practical experience with scraping, cleaning, modeling, and evaluating.
- Reusable infrastructure for future car market analysis projects.

Data-mining goals

Data-mining goals

- Scrape and collect a clean dataset of used-car listings from Auto24. Build a regression model that predicts car prices based on available attributes.
- Identify the most influential factors affecting pricing.
- Ensure reproducibility (scripts for scraping, cleaning, and training).

Data-mining success criteria

- A working dataset with minimal missing or unusable values.
- A regression model achieving acceptable performance.
- Stable predictions when tested on new, unseen listings.

Data Understanding

Gathering Data

Outlining data requirements

To build an accurate car price prediction model, the project requires a dataset containing vehicle-specific attributes and their listed prices. The features needed must describe characteristics that influence car value. These include:

- Basic identifiers: brand, model, generation.
- Technical details: year of manufacture, mileage, engine size, engine power (kW), fuel type, drivetrain, gearbox.
- Listing information: asking price, additional equipment.

Each record must correspond to one listing on Auto24. The target variable is always the asking price.

Verifying data availability

After running some tests on a smaller set of pages we can confirm that Auto24 publicly displays all required features on each listing page. However, some attributes are not consistently present or labeled differently across sellers (addons).

The majority of the essential attributes—price, brand, model, year, mileage, fuel type, gearbox, power—are reliably available across most listings. Therefore, sufficient data exists to train a predictive model, although some optional fields may have high missingness and require dropping.

Selection Criteria

The project will include only:

- Passenger cars and SUV-s

Listings with obviously incorrect or incomplete data will be excluded.
Aiming for 5,000–15,000 listings across various brands and models to ensure sufficient feature variability.

NB! The rest of this task we cannot answer, because we have yet to finish the data scraping. Due to the variety of possible combinations of addons in the listings, the dataset would have thousands of columns, so we are working on setting up a structured scraper.

Planning the project

Task 1 Data Scraping (Team total: 20 hours)

- Jasper: 20 hours
- Hugo: 0 hours
- Artjom: 0 hours

Description: Create a web scraper for Auto24 listings using Python, BeautifulSoup/Selenium. Includes handling pagination, rate limiting, and HTML structure changes.
Important: Scraper reliability and that the data is read correctly to avoid too much work in the next task

Task 2 Data Cleaning and Preprocessing (Team total: 20 hours)

- Jasper: 0 h
- Hugo: 10 h
- Artjom: 10 h

Description: Normalize features, handle missing values, convert categorical variables, remove duplicates, and filter invalid listings.

Task 3 Exploratory Data Analysis (Team total: 10 hours)

- Jasper: 3 h
- Hugo: 4 h
- Artjom: 3 h

Description: Analyze distributions, detect outliers, compute correlations, visualize price relationships using Pandas and Matplotlib or similar.

Task 4 Model Development & Evaluation (Team total: 25 hours)

- Jasper: 5 h
- Hugo: 8 h
- Artjom: 12 h

Description: Train regression models and optimize hyperparameters

Task 5 Reporting & Presentation (Team total: 15 hours)

- Jasper: 2 h
- Hugo: 8 h
- Artjom: 5 h

Description: Compile results, prepare slides, document the methodology.

Methods and Tools

Python, BeautifulSoup/Selenium, Pandas, NumPy, Scikit-learn, Matplotlib, Git/GitHub, Google Docs/Slide, Chat GPT, Claude, Deep Seek, Gemini.