Car Price Prediction with Machine Learning

Objective

The goal of this project is to develop a machine learning model that predicts car prices based on user input. Students will work with a real-world dataset from Kaggle, exploring data, building predictive models, and deploying an API.

Project Steps

1. **Dataset Exploration**
   - Go to Kaggle to learn more about the dataset via the link below: https://www.kaggle.com/datasets/sidharth178/car-prices-dataset
   - Load the dataset into a Jupyter Notebook.
   - Perform exploratory data analysis: check missing values, data distribution, correlations, and outliers.
   - Clean and preprocess the data.
   - Split the data into train, validation and test dataset.
2. **Baseline Model**
   - Define a simple baseline model.
   - Choose an appropriate performance metric.
   - Evaluate the baseline model.
3. **Model Comparison**
   - Experiment with multiple machine learning models.
   - Compare their performance using the chosen metric.
   - Interpret the results.
4. **Hyperparameter Tuning**
   - Use GridSearchCV to optimize model hyperparameters.
   - Get the best hyperparameters.
5. **Model Saving**
   - Select the best-performing model with the package pickle.
6. **API Development with FastAPI**
   - Create an API that allows users to input car features and output a predicted price.
   - Implement at least one endpoint for prediction.
   - Test the API locally with sample requests.
7. **Code Versioning with GitHub**
   - Organize the code structure and control the environment.
   - Use Git for version control and push the project to GitHub.
   - Write a README file explaining the project and its usage.
8. **Deployment on Render**
   - Deploy the FastAPI application on Render.
   - Ensure the API is accessible online.

Deliverables

- A Jupyter Notebook with data analysis, baseline model, model comparisons, and hyperparameter tuning.

- A FastAPI script for the prediction API in a GitHub repository with all project files.
- A deployed API link on Render.
- A short report (maximum 5 pages) explaining the workflow and results.

In the report you have to present:

- The context and the project,
- Why you use the metric of performance,
- The performance of your algorithm,
- The best hyperparameter,
- The explanation of how work your API,
- The link of the different resources of the project (notebook, github, url of the API on render).
- Conclusion of the project.

All deliverables must be submitted by March 3 at my email address
gautherotmorgan0@gmail.com

This project will serve as the final evaluation for the course, testing both technical skills and the ability to apply data science to a real-world problem. 🚀