Theta Technolabs

Stage 2
Technical Evaluation Task

Submission Instructions:

- Write **clean, well-documented code**.
- **Test and execute** your programs.
- Submit your **code** in a shared document or repository (Publicly accessible).

🔴 **Important:** Please **save a local copy of your files** as well. If anything goes wrong, we can evaluate based on that.

📌 **Deadline: 4 days from the assignment date, you must submit it by 19-12-2025 (Monday) morning.**

If you have any questions, feel free to reach out. **All the best!** 🚀

AI/ML Task: Predicting the Price of Used Cars in the Indian Market

1. Problem Statement:

The Indian used car market is vast. Accurately pricing a used car is crucial. The goal is to build a machine learning model to predict the selling price of a used car based on its key features.

2. Business Objective:

- For an online platform: Provide a basic price estimate for sellers and buyers.
- For individuals: Understand primary factors influencing car prices.

3. Dataset:

- **Example Dataset:**
  - **Car_Name**: Name of the car model (e.g., "ritz", "sx4", "ciaz")
  - **Year**: Year of manufacture
  - **Selling_Price**: The price the car was sold for (target variable, in Lakhs INR)

- **Present_Price**: Showroom price of a new car (in Lakhs INR)
- **Kms_Driven**: Kilometers driven
- **Fuel_Type**: Petrol, Diesel, CNG
- **Seller_Type**: Dealer, Individual
- **Transmission**: Manual, Automatic
- **Owner**: Number of previous owners (0 for first, 1 for second, etc.)

- **Indian Market Considerations:**
  - Consider the prevalence of popular Indian brands (Maruti Suzuki, Hyundai, etc.).
  - Note how fuel type preferences might influence price.

4. Detailed Task Breakdown:

- Data Exploration and Basic Feature Engineering

- **Data Loading & Initial Inspection:**
  - Load the dataset using Pandas.
  - Use df.info(), df.describe(), df.head(), df.isnull().sum().
- **Exploratory Data Analysis (EDA):**
- **Univariate Analysis:**
    - Distribution of Selling_Price - check for skewness.
    - Distribution of numerical Year, Present_Price, Kms_Driven.
  - Frequency counts for Fuel_Type, Seller_Type, Transmission, Owner.
- **Bivariate Analysis:**
    - Scatter plots: Selling_Price vs. Year, Present_Price, Kms_Driven.
    - Box plots: Selling_Price vs. Fuel_Type, Seller_Type, Transmission.
  - Correlation heatmap for numerical features.
- **Data Cleaning:**
  - Handle missing values (if any – often this dataset is clean).
  - Check for extreme outliers in Kms_Driven or Present_Price (e.g., visually or using IQR) and consider if capping/removal is necessary.
- **Feature Engineering (Crucial for this problem):**
  - Create Car_Age = 2024 - Year (assuming 2024 as the current year for simplicity).
  - **Decision on** for this simplified task, we will **initially drop the** to avoid complexity with brand extraction or high cardinality encoding. We will focus on the other provided features.
- **Data Splitting:** Split data into training (80%) and testing (20%) sets. Use random_state for reproducibility.

- Preprocessing & Baseline Model Building

- **Feature Preprocessing (fit on training data, transform on train & test)**
- **Categorical Encoding:**
  - Use One-Hot Encoding (e.g., pd.get_dummies) for Fuel_Type, Seller_Type, Transmission, and Owner.
- **Numerical Scaling (Optional but good practice for some models):**
  - Consider StandardScaler for Present_Price, Kms_Driven, Car_Age. Tree-based models are less sensitive, so this is not strictly mandatory but good practice.
- **Target Variable Transformation (if skewed):**
  - If Selling_Price is skewed, apply np.log1p. Remember to inverse transform predictions later using np.expm1.
- **Model Selection & Training (start with simpler models):**
  - **Linear Regression:** As a simple baseline.
  - **Random Forest Regressor:** A powerful ensemble model that often works well.
- **Initial Evaluation:**
- Use metrics like:
    - **Mean Absolute Error (MAE)**
    - **Root Mean Squared Error (RMSE)**
  - **R-squared (R^2) Score**
  - Compare performance on training and testing sets to check for overfitting.

- Model Evaluation and Interpretation

- **Feature Importance:** For the Random Forest Regressor, extract and visualize feature importances.
- **Final Model Selection:** Based on test set performance (R^2, RMSE, MAE), choose the better performing model (likely Random Forest).
- **Interpretation & Insights:**
  - Which features are most important in predicting car prices (e.g., Car_Age, Present_Price, Kms_Driven)?
  - Briefly discuss how features like Fuel_Type or Transmission appear to affect the price based on model insights or EDA.
- Write a summary covering:
    - The problem.
    - Key EDA findings.
    - Preprocessing steps.
    - Which model performed best and its performance (MAE, RMSE, R^2).
    - Key insights from feature importances.

- Limitations and potential next steps.

5. Success Criteria / Evaluation Metrics:

- **Primary Metric:** Aim for a good R^2 score (e.g., > 0.80).
- **Secondary Metrics:** Reasonable MAE/RMSE values.
- Clear demonstration of the EDA and preprocessing pipeline.
- Meaningful interpretation of feature importances.