Phase-2

Student Name: Jagan.s

Register Number: 511523205020

Institution: P.T.Lee Chengalvaraya Naicker College Of Engineering & Technology

Department: Information Technology

Date Of Submission: 08-05-2025

Github Link [https://github.com/tamil9843/tamilselvan-phase-2.git]

Project Title Forecasting House Prices Accurately Using Smart Regression Techniques in Data Science

1. Problem Statement Accurately predicting house prices is a long-standing challenge in the real estate industry and is crucial for buyers, sellers, agents, and policymakers. This project aims to forecast housing prices based on a range of features including location, structure, and sale timelines. The task is framed as a regression problem, with the target variable being the house price (continuous numeric). Early and reliable predictions can assist in better investment decisions, pricing strategies, and market analysis.
2. Project Objectives

* Develop robust regression models to accurately forecast house prices.
* Identify key predictors that influence property value.
* Analyze the impact of geographic, structural, and temporal features.
* Ensure the model is interpretable and practical for real-world applications.
* Deploy a simple Gradio-based interface for user testing.
* Refine the approach to emphasize impactful variables such as number of bedrooms, property type, and location.

1. Flowchart of the Project Workflow (Insert or describe the workflow here: Data Collection → Preprocessing → EDA → Feature Engineering → Modeling → Evaluation → Deployment)
2. Data Description

* Dataset Name: Australian Property Sales Dataset
* Source: Public Real Estate Data – Australia
* Type: Structured tabular data
* Records & Features:
  + raw\_sales.csv: 29,580 records, 5 features
  + ma\_lga\_12345.csv: 347 records, 4 features
* Target Variable: price
* Static or Dynamic: Static dataset

Attributes Covered:

* Structural: Number of bedrooms, property type
* Location: Postcode
* Temporal: Sale date

1. Data Preprocessing

* Verified dataset for missing/null values.
* Removed irrelevant or redundant features.
* Handled missing values using appropriate imputations.
* Encoded categorical features using label encoding.
* Normalized numerical features using MinMaxScaler.
* Detected and addressed outliers using z-scores and boxplots.

1. Exploratory Data Analysis (EDA) Univariate Analysis:

* Histogram of house prices to examine distribution.
* Bar plots for property types and bedroom counts.

Bivariate & Multivariate Analysis:

* Correlation analysis between price and number of bedrooms.
* Boxplots showing price variations across postcodes and property types.

Key Insights:

* Higher bedroom count generally corresponds to higher prices.
* Certain postcodes consistently show premium pricing.
* Property type also plays a role in price variation.

1. Feature Engineering

* Created binary and ratio features based on bedroom count and price per unit.
* Parsed and transformed date features into useful components.
* Encoded categorical data efficiently to improve model performance.
* Scaled all numeric features for uniformity.

1. Model Building Algorithms Used:

* Linear Regression (Baseline)
* Random Forest Regressor
* XGBoost Regressor (advanced ensemble model)

Model Selection Rationale:

* Linear Regression: Serves as a baseline.
* Random Forest: Handles non-linear relationships and outliers.
* XGBoost: Provides high accuracy with regularization.

Train-Test Split:

* 80% training, 20% testing using train\_test\_split

Evaluation Metrics:

* MAE (Mean Absolute Error)
* RMSE (Root Mean Squared Error)
* R² Score (Explained variance)

1. Visualization of Results & Model Insights

* Feature Importance: Visualized using Random Forest and XGBoost.
* Model Comparison: Performance compared using MAE, RMSE, and R².
* Residual Analysis: Visualized actual vs predicted values.
* Deployment Interface: Simple Gradio app for testing house price predictions.

1. Tools and Technologies Used

* Language: Python 3
* Platform: Google Colab / Jupyter Notebook
* Libraries:
  + pandas, numpy (data handling)
  + matplotlib, seaborn, plotly (EDA & visualization)
  + scikit-learn, XGBoost (modeling)
  + Gradio (deployment)

1. Team Members and Contributions

* Ragulgandhi.K - Data Cleaning & Preprocessing
* Tamilselvan.J - EDA & Visualization
* Sakthivel.R - Model Building & Evaluation
* Jagan.S - Visualization & Documentation