EVERY NATION POLYTECHNIC SAMIE BO CAMPUS

PROJECT ON ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

TUTOR MR EMMANUEL KOROMA

Artificial Intelligence and Machine Learning

- **Project Title : House Price Prediction Using Machine Learning**
- **Student Name(s) : [Joseph Kpulun]**
- **Course Name : [Artificial Intelligence and Machine Learning]**
- **Instructor Name : [Emmanuel Koroma]**
- **Date : [26th MARCH, 2025]**

Table of Contents

1. Introduction

1.1 Project Overview

The House Price Prediction project aims to predict house prices based on features such as square footage, number of bedrooms, location, and other relevant attributes. This is achieved using machine learning techniques, specifically regression models like Linear Regression and Decision Trees. The project leverages publicly available datasets, such as the Sierra Leone Housing Dataset, to train and evaluate predictive models.

1.2 Problem Statement

Predicting house prices accurately is a critical task in real estate markets, aiding buyers, sellers, and investors in making informed decisions. However, traditional methods of price estimation often rely on subjective assessments or manual calculations, which can be time-consuming and prone to errors. This project addresses these challenges by developing a data-driven approach to automate and improve the accuracy of house price predictions.

1.3 Objectives

1. To preprocess and analyze housing datasets for feature engineering and normalization.
2. To train and evaluate regression models (e.g., Linear Regression, Decision Trees) for predicting house prices.
3. To assess model performance using metrics such as Mean Squared Error (MSE) and R-squared.
4. To create a user-friendly web application for real-time house price predictions.

1.4 Scope

This project focuses on predicting house prices using regression models. The scope includes:

- Data preprocessing and feature engineering.
- Model training, evaluation, and comparison.
- Deployment of a web-based prediction tool using Streamlit.
  The project does not cover advanced deep learning models or real-time market analysis.

2. Literature Review

This section reviews existing literature on house price prediction and machine learning applications in real estate. Key findings include:

- Studies have shown that regression models like Linear Regression and Decision Trees are effective for price prediction tasks.
- Feature engineering and normalization significantly improve model performance.
- Real-world applications often use datasets like the California Housing Dataset due to their relevance and availability.

3. Requirements

3.1 Functional Requirements

1. Data Preprocessing: Clean and normalize the dataset for model training.

2. Model Training: Train regression models (Linear Regression, Decision Trees).
3. Evaluation: Assess model performance using MSE and R-squared metrics.
4. Web Application: Develop a user interface for real-time predictions.

3.2 Non-Functional Requirements

1. Performance: Ensure low latency for predictions.
2. Usability: Provide an intuitive interface for users.
3. Scalability: Handle up to 1,000 concurrent users.
4. Security: Protect user inputs and outputs from unauthorized access.

3.3 Tools and Technologies

- Programming Languages: Python
- Libraries: Scikit-learn, Pandas, NumPy, Matplotlib/Seaborn
- Frameworks: Streamlit (for web app)
- Datasets: California Housing Dataset (from Scikit-learn)

4. Design

4.1 System Architecture

A high-level diagram illustrating the system architecture:

1. Data Ingestion: Load and preprocess the dataset.
2. Model Training: Train regression models using Scikit-learn.
3. Evaluation: Evaluate models using test data and metrics.
4. Web Application: Deploy the trained model using Streamlit for user interaction.

4.2 Database Design

Since this project uses a static dataset, no database design is required. However, if extended to a real-time system, a relational database schema could include:

- **Houses**: ID, Square Footage, Bedrooms, Location, Price
- **Predictions**: ID, Input Features, Predicted Price

4.3 User Interface Design

Wireframes or screenshots of the web application:

- Input fields for user-provided features (e.g., square footage, bedrooms).
- A "Predict" button to generate price predictions.
- Display of predicted house prices and model performance metrics.

5. Implementation

5.1 Development Process

1. Data Preprocessing: Normalize features using StandardScaler.
2. Model Training: Train Linear Regression and Decision Tree models.
3. Evaluation: Compare models using MSE and R-squared metrics.
4. Web App Development: Create a Streamlit app for user interaction.

5.2 Challenges Faced

- Handling missing or inconsistent data in the dataset.
- Over fitting in Decision Tree models.
- Ensuring scalability of the web application.

5.3 Code Structure

The codebase is organized into the following modules:

- **data_preprocessing.py**: Handles data cleaning and normalization.
- **model_training.py**: Trains and evaluates regression models.
- **app.py**: Implements the Streamlit web application.

5.4 Dependencies

- Scikit-learn: For machine learning models and metrics.

- Pandas/NumPy: For data manipulation.
- Matplotlib/Seaborn: For visualizations.
- Streamlit: For the web application.

6. Testing

6.1 Testing Methodology

- Unit Testing: Test individual functions for data preprocessing and model training.
- Integration Testing: Ensure seamless interaction between modules.
- User Acceptance Testing: Validate the web application with real users.

6.2 Test Cases

Test Case ID	Description	Input	Expected Output	Actual Output
TC01	Linear Regression Prediction	Sample input data	Predicted price	Predicted price
TC02	Decision Tree Prediction	Sample input data	Predicted price	Predicted price

6.3 Results

Summarize results, including MSE and R-squared values for each model.

7. Results and Discussion

7.1 Outcome

The Linear Regression model achieved an MSE of X and R-squared of Y, while the Decision Tree model achieved an MSE of Z and R-squared of W. The web application successfully integrates the trained models for real-time predictions.

7.2 Achievements

- Accurate predictions using regression models.
- Successful deployment of a user-friendly web application.

7.3 Limitations

- Limited to predefined features in the dataset.
- Models may not generalize well to unseen data outside the dataset's scope.

7.4 Future Work

- Incorporate additional features (e.g., proximity to amenities).
- Explore advanced models like Random Forests or Neural Networks.

8. Conclusion

The project successfully implemented a machine learning-based solution for house price prediction. It demonstrated the effectiveness of regression models and highlighted the importance of feature engineering and evaluation metrics. Lessons learned include the value of data preprocessing and the need for scalable deployment strategies.

9. References

List all references used formatted according to your institution's guidelines. For example:

- Scikit-learn Documentation. (2023). Retrieved from https://scikit-learn.org
- McKinney, W. (2017). Python for Data Analysis. O'Reilly Media.

10. Appendices

Appendix A: Code

Include the full source code or link to a repository (e.g., GitHub).

Appendix B: Additional Diagrams

Add supplementary diagrams, such as flowcharts or visualizations.

Appendix C: User Manual

Provide instructions for using the web application, including screenshots and step-by-step guidance.