Machine Learning Assignment Documentation

Table of Contents

---

1. Overview

This project predicts house prices using **Random Forest Regression**. The dataset consists of **545 rows and 13 columns**, including numerical and categorical features like price, area, bedrooms, and furnishing status.

Objective:

- Develop an accurate model to predict house prices.
- Improve accuracy using feature engineering and data preprocessing.
- Evaluate the effectiveness of different modeling techniques.

---

2. Dependencies

Libraries Used:

- **`numpy`**: Numerical computations.
- **`pandas`**: Data manipulation.
- **`matplotlib`**, `seaborn`: Data visualization.

- **`sklearn`**: Machine learning model and preprocessing tools.
- **`joblib`**: Model serialization for saving trained models.

Dataset:

- **house-price.csv** containing features such as:
  - **`Price`**
  - **`Area`**
  - **`Bedrooms`**, **`Bathrooms`**
  - **`Furnishing Status`**, **`Parking`**
  - Other categorical features

---

3. Data Preprocessing

Steps:

1. **Load Data:** Read CSV into a DataFrame.
2. **Check Missing Values:**
   - **`df.isnull().sum()`** shows no missing values.
3. **Handle Outliers:**
   - Visualized using **`boxplots.`**
   - Removed extreme values using the **Interquartile Range (IQR) method**.
4. **Encoding Categorical Variables:**
   - Used **`OneHotEncoder`** for categorical features.
5. **Scaling Numerical Features:**
   - Applied **`StandardScaler`** for normalization.
6. **Feature Engineering:**
   - Created new variables like **`Price per sqft`** to enhance prediction accuracy.

---

4. Exploratory Data Analysis (EDA)

Data Exploration:

- **Dataset Shape:** `(545, 13)`
- **Data Types:** Mixed numerical and categorical features.
- **Correlation Analysis:**
  - **`sns.heatmap(df.corr())`** to visualize correlations.
  - `Price` highly correlated with `Area` and `Bedrooms`.

Key Insights:

- Bigger houses tend to be more expensive.
- Categorical variables (e.g., `Furnishing Status`) influence pricing.
- The dataset appears clean with no missing values.

---

5. Feature Engineering

Feature engineering helps in improving the predictive power of the model by transforming raw data into meaningful features.

Added Features:

- **Price per Square Foot: `df['price_per_sqft'] = df['price'] / df['area']`**
- **Total Bathrooms:** Combining `bathrooms` and additional bathrooms if applicable.
- **House Age:** Estimated based on available historical data.
- **Neighborhood Categorization:** Based on average pricing trends.

These features can provide additional insights and enhance model performance.

---

6. Model Training

Train-Test Split:

- **Split data into 80% training, 20% testing using `train_test_split()`.**

Preprocessing Pipeline:

- **Numerical Features:** Standardized with `StandardScaler()`.
- **Categorical Features:** Encoded using `OneHotEncoder()`.

Model Selection:

- **Random Forest Regressor** chosen due to its high accuracy and robustness.
- **Baseline Model:** Predicted average house price as a comparison.

Training:

- **`model.fit(X_train, y_train)`** to train Random Forest.

---

7. Model Evaluation

Performance Metrics:

- **Mean Absolute Error (MAE):** Measures average error.
- **Mean Squared Error (MSE):** Penalizes large errors.
- **R^2 Score:** Measures variance explained by the model.

The Random Forest model significantly outperforms the baseline.

---

8. Hyperparameter Tuning

Grid Search:

- Optimized hyperparameters like **`n_estimators`**, **`max_depth`**.
- Used **`GridSearchCV`** for tuning.

Best Parameters:

```
{'n_estimators': 200, 'max_depth': 20, 'min_samples_split': 5}
```

Tuning improved accuracy by reducing overfitting and optimizing model complexity.

---

9. Visualizations and Insights

Scatter Plot:

- **Price vs. Area:** Shows strong positive correlation.

Box Plots:

- **Categorical Variables vs. Price:** Shows pricing trends across categories.

Residual Analysis:

- Residuals randomly distributed → Model is well-fitted.

Feature Importance:

- **Random Forest provides feature importance metrics**:

```
importances = model.feature_importances_
```

- `Area`, `Bedrooms`, and `Bathrooms` were the most important features.

---

10. Future Work & Conclusion

Summary:

- Successfully built an accurate **Random Forest Regression Model**.
- Optimized hyperparameters improved performance.
- Model **outperformed the baseline**, proving its predictive ability.

Future Improvements:

- **Try alternative models like XGBoost and Gradient Boosting** for better results.
- **Collect more data** to enhance generalization.
- **Feature Selection Techniques**: Experiment with PCA and Recursive Feature Elimination.

The model shows strong predictive power, and further refinements can enhance performance.

Thank You!