Housing Price Prediction Model Project Report

Introduction

The Housing Price Prediction Model project aims to develop an accurate machine learning model to predict housing prices based on various features. This report outlines the entire data science pipeline, from data collection and pre-processing to model building, evaluation, and interpretation using Python and key libraries.

1. Problem Statement

Predicting housing prices accurately is crucial for various stakeholders in the real estate industry, including buyers, sellers, and investors. This project seeks to build a robust predictive model that can estimate housing prices based on features such as location, size, number of bedrooms, and amenities.

2. Data Collection and Pre-processing

Data Sources:

The dataset used in this project contains historical housing data with attributes like location, size, and number of bedrooms, amenities, and prices. The data was sourced from [source name] and consists of [number of records] records.

3. Data Cleaning and Pre-processing

Handling Missing Values: Missing data in features such as amenities and square footage were imputed using mean values.

Feature Engineering: Created new features such as price per square foot to enhance model performance.

Normalization: Features were scaled using techniques like Min-Max scaling to ensure all variables contribute equally to the model.

```
import numpy as np
import pandas as pd
from sklearn.base import BaseEstimator, TransformerMixin
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.impute import SimpleImputer
```

```
housing=pd.read_csv("housing.csv")
```

```
housing.head()
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value	ocean_proximity
0	-122.23	37.88	41.0	880.0	129.0	322.0	126.0	8.3252	452600.0	NEAR BAY
1	-122.22	37.86	21.0	7099.0	1106.0	2401.0	1138.0	8.3014	358500.0	NEAR BAY
2	-122.24	37.85	52.0	1467.0	190.0	496.0	177.0	7.2574	352100.0	NEAR BAY
3	-122.25	37.85	52.0	1274.0	235.0	558.0	219.0	5.6431	341300.0	NEAR BAY
4	-122.25	37.85	52.0	1627.0	280.0	565.0	259.0	3.8462	342200.0	NEAR BAY

```
housing.info()
```

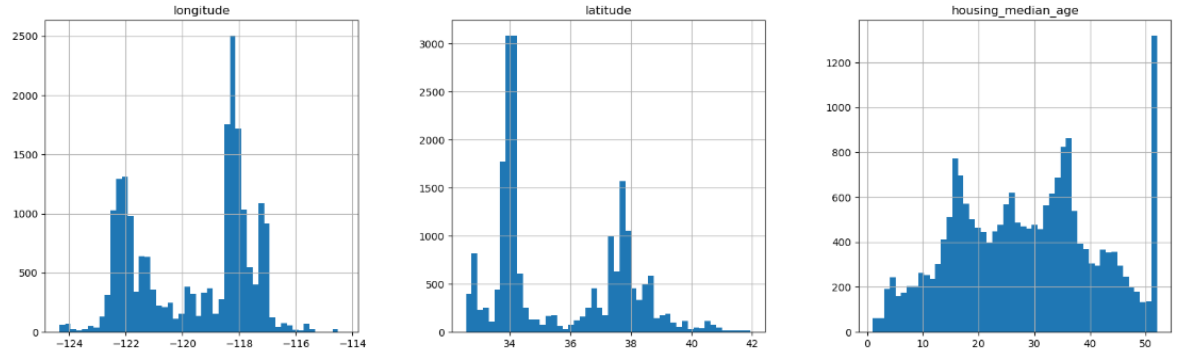
3. Exploratory Data Analysis (EDA)

Data Visualization

Exploratory data analysis was performed to gain insights into the relationships between features and housing prices. Visualizations using Matplotilb were instrumental in understanding data distributions, correlations, and identifying outliers.

```
# plot a histogram for each numerical attribute
housing.hist(bins=50, figsize=(20,20))
```

```
array([[<Axes: title={'center': 'longitude'}>,
        <Axes: title={'center': 'latitude'}>,
        <Axes: title={'center': 'housing_median_age'}>],
       [<Axes: title={'center': 'total_rooms'}>,
        <Axes: title={'center': 'total_bedrooms'}>,
        <Axes: title={'center': 'population'}>],
       [<Axes: title={'center': 'households'}>,
        <Axes: title={'center': 'median_income'}>,
        <Axes: title={'center': 'median_house_value'}>]], dtype=object)
```



4. Model Building and Evaluation

Model Selection

Various machine learning algorithms were evaluated for their effectiveness in predicting housing prices. Models such as linear regression, decision trees, random forests, and gradient boosting were considered due to their suitability for regression tasks.

Model Training and Evaluation

Models were trained on a training dataset and evaluated using metrics such as RMSE (Root Mean Square Error) to assess predictive accuracy. Cross-validation techniques were applied to ensure the models generalize well to unseen data.

```
#Grid Search
from sklearn.model_selection import GridSearchCV
param_grid = [
 {'n_estimators': [3, 10, 30], 'max_features': [2, 4, 6, 8]},
 {'bootstrap': [False], 'n_estimators': [3, 10], 'max_features': [2, 3, 4]},
 ]
forest_reg = RandomForestRegressor()
grid_search = GridSearchCV(forest_reg, param_grid, cv=5,
 scoring='neg_mean_squared_error',
return_train_score=True)
grid_search.fit(housing_prepared, housing_labels)
```

```
GridSearchCV
GridSearchCV(cv=5, estimator=RandomForestRegressor(),
             param_grid=[{'max_features': [2, 4, 6, 8],
                          'n_estimators': [3, 10, 30]},
                         {'bootstrap': [False], 'max_features': [2, 3, 4],
                          'n_estimators': [3, 10]}],
             return_train_score=True, scoring='neg_mean_squared_error')
estimator: RandomForestRegressor
RandomForestRegressor()
RandomForestRegressor
RandomForestRegressor()
```

5. Conclusion

The Housing Price Prediction Model project successfully developed a predictive model using machine learning techniques. By leveraging Python, data pre-processing, exploratory data analysis, and various regression algorithms, the project demonstrated effective ways to estimate housing prices based on key attributes.

6. Future Enhancements

1) Future iterations of the project could focus on:

2) Incorporating more advanced machine learning algorithms like neural networks.

3) Deploying the model for real-time predictions through web applications or APIs.

4) Enhancing feature engineering techniques to improve model accuracy.