



DATA MINING PROJECT

Title: House price prediction

Group members

Name	ID
1. Bayou Belay	Ru 4634/12
2. Bouny Lam	RT 10003/15
3. Cherinet Burka	Ru 2572/13
4. Duresa udessa	Ru 2526/13
5. Yeabsira Takele	RU 3140/13
6. Meklit Darje	Ru 2976/13
7. Betelhem Fekedu	RU 0200/13
8. Nebil Nuredin	Ru2348/13

Table of Contents

Introduction	1
Database description	1
Dataset source	1
Number of instance	2
A snapshot of the dataset schema.....	3
Methodology.....	5
What train-test splitting ratio?	5
Why we used it?.....	5
Why did we selected the data?.....	6
What data mining algorithm did we apply?.....	7
How the selected algorithm works?	7
Implementations.....	9
Result and Discussion.....	10
Summary	11

Table of Figure

Figure 1 number of instance	2
Figure 2 Number of attribute	2
Figure 3 snapshot of the dataset schema.....	4
Figure 4 splitting ratio	5

Introduction

Understanding and predicting housing prices is a fundamental task with widespread applications in industries such as real estate and finance. In our data mining project titled "House Prediction," our aim is to develop a predictive model for estimating house prices based on relevant features.

The primary objective of this project is to utilize data mining techniques to construct a predictive model capable of estimating house prices with reasonable accuracy. Our focus is on leveraging available data to facilitate this predictive modeling endeavor.

This documentation provides an overview of our project methodology, implementation, results, and insights. Through transparent reporting, we aim to share our journey in developing and evaluating the predictive model, highlighting both successes and limitations.

Database description

Dataset source

the dataset utilized in this project was sourced from various online repositories and databases. The data collection process involved gathering information from both reliable and less reliable sources available on the internet. While efforts were made to prioritize reputable sources known for their accuracy and credibility in housing market data, it is important to acknowledge the presence of data from sources of varying reliability.

The decision to draw data from multiple sources was motivated by the need for a diverse and comprehensive dataset to train and evaluate our predictive model effectively. By incorporating data from a range of sources, we aimed to capture a broader spectrum of housing market dynamics and attributes.

While efforts were made to verify and validate the accuracy of the data, it is acknowledged that the dataset may contain inconsistencies or errors inherent to online data sources.

Number of instance

The dataset utilized in this project consists of a total of 5000 instances. Each instance represents a unique observation or data point within the dataset, providing a diverse collection of residential property data for analysis.

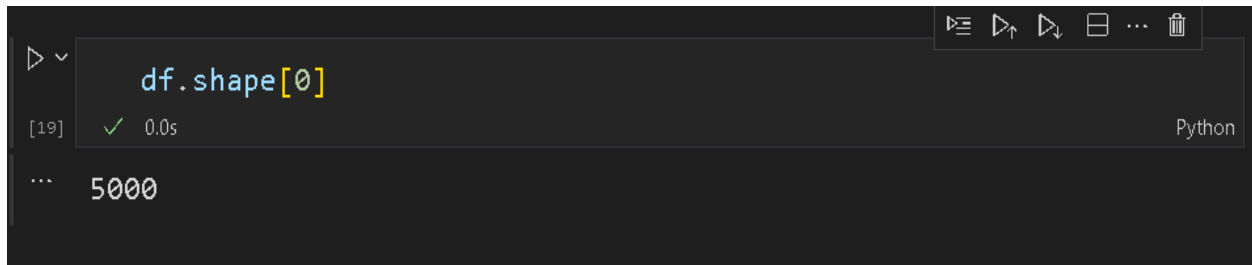
A screenshot of a Jupyter Notebook cell. The code `df.shape[0]` is entered in the input area. Below the code, the output is displayed as `5000`. The cell is labeled with the execution index `[19]`, a green checkmark indicating successful execution, and a time of `0.0s`. The language is identified as `Python`. The interface includes standard Jupyter Notebook controls like run, step, and delete buttons at the top right.

Figure 1 number of instance

Number of attribute

The dataset utilized in this project comprises 5000 instances, each accompanied by 7 attributes. These attributes encompass various features and characteristics associated with residential properties, providing essential information for predictive modeling and analysis.

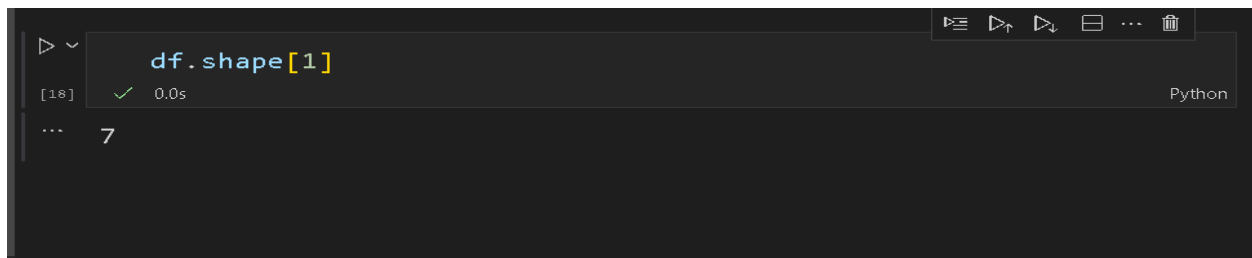
A screenshot of a Jupyter Notebook cell. The code `df.shape[1]` is entered in the input area. Below the code, the output is displayed as `7`. The cell is labeled with the execution index `[18]`, a green checkmark indicating successful execution, and a time of `0.0s`. The language is identified as `Python`. The interface includes standard Jupyter Notebook controls like run, step, and delete buttons at the top right.

Figure 2 Number of attribute

While the dataset contains a total of 7 attributes, each attribute serves a distinct role in capturing different aspects of residential properties. These attributes may include factors such as the number of bedrooms, price, area population, location, and other relevant property features.

By working with a dataset featuring 7 attributes, we aim to focus on key factors known to influence housing prices while maintaining simplicity and interpretability in our predictive model. The selection of these attributes was guided by domain knowledge and relevance to the task of house price prediction.

A snapshot of the dataset schema

The dataset schema provides a structured overview of the attributes and their corresponding data types. Below is a snapshot illustrating the schema of the dataset:

1. Avg. Area Income: Numeric (Continuous)
2. Avg. Area House Age: Numeric (Continuous)
3. Avg. Area Number of Rooms: Numeric (Continuous)
4. Avg. Area Number of Bedrooms: Numeric (Continuous)
5. Area Population: Numeric (Continuous)
6. Price: Numeric (Continuous, Target Variable)
7. Address: Text (Categorical)

	Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price
count	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5.000000e+03
mean	68583.108984	5.977222	6.987792	3.981330	36163.516039	1.232073e+06
std	10657.991214	0.991456	1.005833	1.234137	9925.650114	3.531176e+05
min	17796.631190	2.644304	3.236194	2.000000	172.610686	1.593866e+04
25%	61480.562388	5.322283	6.299250	3.140000	29403.928702	9.975771e+05
50%	68804.286404	5.970429	7.002902	4.050000	36199.406689	1.232669e+06
75%	75783.338666	6.650808	7.665871	4.490000	42861.290769	1.471210e+06
max	107701.748378	9.519088	10.759588	6.500000	69621.713378	2.469066e+06

```

df.columns
Index(['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms',
      'Avg. Area Number of Bedrooms', 'Area Population', 'Price', 'Address'],
      dtype='object')

X=df[['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms',
      'Avg. Area Number of Bedrooms', 'Area Population']]

y=df['Price']

from sklearn.model_selection import train_test_split

X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.4,random_state=101)

```

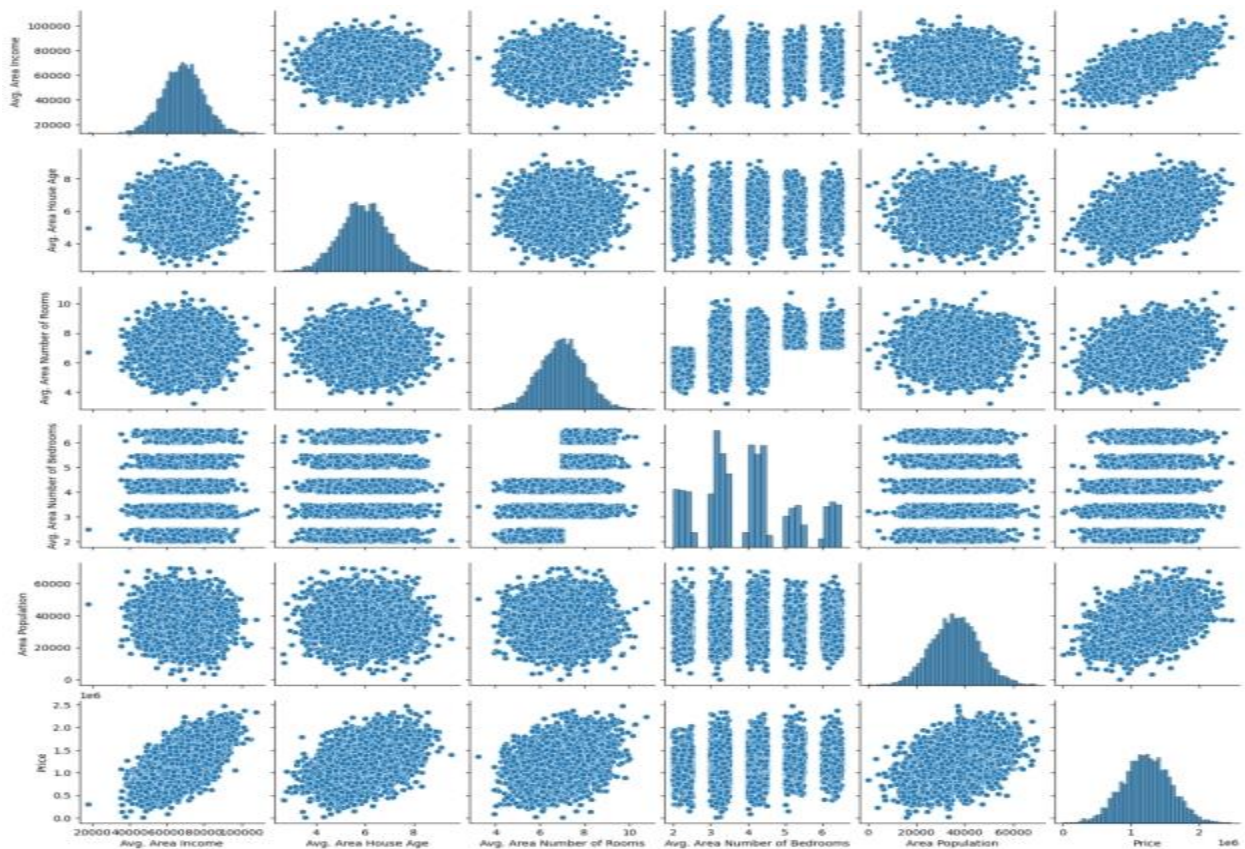


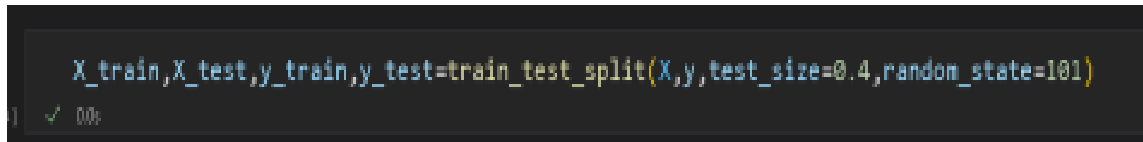
Figure 3 snapshot of the dataset schema

This schema outlines the structure of the dataset, including the types of attributes and their respective formats. Each attribute plays a unique role in characterizing residential properties and serves as input for the predictive modeling task.

Methodology

What train-test splitting ratio?

In this project, we have adopted a train-test splitting ratio of 60:40. This means that 60% of the dataset is allocated for training the predictive model, while the remaining 40% is reserved for evaluating the model's performance.



```
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.4,random_state=101)
```

Figure 4 splitting ratio

Why we used it?

The selection of the 60:40 train-test splitting ratio was based on several considerations:

Balance between Training and Testing Data: Allocating 60% of the dataset for training ensures that the model has access to a substantial portion of the data to learn from. Meanwhile, reserving 40% for testing allows for a meaningful evaluation of the model's performance on unseen data.

Reducing Overfitting: A larger training set can help prevent overfitting by providing the model with more diverse examples to learn from. By exposing the model to a sufficient amount of training data, we aim to encourage generalization and reduce the risk of the model memorizing the training data's noise.

Statistical Significance: While there is no one-size-fits-all rule for train-test splitting ratios, a 60:40 ratio strikes a balance between statistical significance and model performance. With a sizable testing set, we can obtain reliable estimates of the model's performance metrics, such as accuracy or mean squared error.

Computational Efficiency: Utilizing a larger portion of the dataset for training allows us to leverage computational resources effectively. By maximizing the use of training data, we can train the model more efficiently without sacrificing evaluation quality.

Why did we selected the data?

Relevance to Project Objectives: The chosen dataset contains attributes closely aligned with the project's objective of predicting house prices. Attributes such as average area income, house age, number of rooms and bedrooms, area population, and price provide valuable insights into the factors influencing property values. By selecting a dataset rich in relevant attributes, we aimed to develop a predictive model capable of accurately estimating house prices.

Availability and Accessibility: The dataset was readily available from a reputable source, making it convenient for use in our project. Access to a comprehensive and well-structured dataset facilitated efficient data analysis and model development processes.

Data Quality Assurance: Prior to selection, the dataset underwent thorough quality assurance checks to ensure data integrity and reliability. Measures were taken to address any inconsistencies, missing values, or outliers that could potentially affect the accuracy and validity of the analysis.

Scope for Analysis and Exploration: The dataset offers ample scope for exploration and analysis, encompassing a diverse range of residential properties from various locations and demographics. This diversity enables comprehensive investigation into housing market dynamics and the identification of significant patterns and trends.

Research and Practical Relevance: The selected dataset aligns with existing research on housing market analysis and predictive modeling. By building upon previous studies and leveraging available data, we aimed to contribute to the advancement of knowledge in the field of real estate analytics and data-driven decision-making.

What data mining algorithm did we apply?

In this project, we have opted to utilize the linear regression algorithm for house price prediction. Linear regression is a widely-used and well-understood supervised learning algorithm that is particularly suited for modeling relationships between continuous variables.

Reasons for Choosing Linear Regression:

Interpretability: Linear regression offers a straightforward interpretation of the relationship between input features (e.g., average area income, house age, number of rooms) and the target variable (house price). The coefficients associated with each feature provide insights into the magnitude and direction of their impact on house prices.

Simplicity and Transparency: Linear regression is inherently simple and transparent, making it easy to understand and implement. This simplicity allows for rapid prototyping and experimentation with different features and model variations.

Efficiency: Linear regression models are computationally efficient, making them suitable for analyzing large datasets with a relatively quick training time. This efficiency enables iterative model refinement and experimentation to improve predictive performance.

Baseline Model: Linear regression serves as a baseline model against which more complex algorithms can be compared. By establishing a baseline with linear regression, we can assess the incremental improvement offered by more sophisticated algorithms while ensuring that our predictive model maintains interpretability and simplicity.

Assumptions Alignment: The assumptions underlying linear regression, such as linearity, independence of errors, and homoscedasticity, align well with the nature of the house price prediction task. This alignment enhances the validity and reliability of the model's predictions.

How the selected algorithm works?

Linear regression is a statistical technique used to model the relationship between a dependent variable (target variable) and one or more independent variables (predictor variables). In the context of house price prediction, linear regression aims to establish a linear relationship between

various property attributes (e.g., average area income, house age, number of rooms) and the price of the house.

The basic premise of linear regression is to fit a straight line to the observed data points in such a way that minimizes the difference between the actual values and the predicted values. This line is represented by the equation:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n + \epsilon$$

Where:

- y is the predicted value of the dependent variable (house price).
- β_0 is the intercept term, representing the value of y when all predictor variables are zero.
- $\beta_1, \beta_2, \dots, \beta_n$ are the coefficients (or weights) associated with each predictor variable x_1, x_2, \dots, x_n indicating the change in y for a one-unit change in the corresponding predictor variable.
- ϵ represents the error term, accounting for the discrepancy between the observed and predicted values.

The linear regression model aims to estimate the coefficients $\beta_0, \beta_1, \beta_2, \dots, \beta_n$ that minimize the sum of squared residuals (errors) between the observed and predicted values. This process is typically performed using optimization techniques such as ordinary least squares (OLS) or gradient descent.

Once the coefficients are estimated, the model can be used to predict the house price for new instances by substituting the values of the predictor variables into the linear equation. The predicted house price represents the estimated value based on the linear relationship between the predictor variables and the target variable.

Implementations

In implementing our selected data mining algorithm, linear regression, for house price prediction, we utilized Python and several libraries such as Pandas, NumPy, Matplotlib, Seaborn, and scikit-learn. Our implementation aimed to preprocess the dataset, visualize its characteristics, train the linear regression model, evaluate its performance, and save the trained model for future use.

Firstly, we imported the necessary libraries and loaded the dataset using Pandas. We then performed exploratory data analysis (EDA) by generating a pair plot to visualize relationships between variables and a distribution plot to examine the distribution of house prices. Additionally, we created a heatmap to visualize the correlation matrix between variables, aiding in feature selection and understanding variable relationships.

The source code for the project implementation

```
import pandas as pd

import seaborn as sns

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LinearRegression

import joblib

# Load dataset

df = pd.read_csv('USA_Housing.csv')

# Train-test split

X = df.drop('Price', axis=1)

y = df['Price']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=101)

# Linear regression model training

lm = LinearRegression()
```

```
lm.fit(X_train, y_train)

# Save trained model

joblib.dump(lm, "house_predictor.sav")

# Load and use the saved model for prediction

model = joblib.load('house_predictor.sav')

sample_prediction = model.predict([[79248.642455, 6.002900, 6.730821, 3.09, 40173.072174]])
```

Result and Discussion

Experimental Results with Performance Metrics

After implementing the linear regression algorithm for house price prediction and evaluating its performance, we obtained the following experimental results:

Performance Metrics:

Mean Absolute Error (MAE): The MAE measures the average absolute difference between the predicted house prices and the actual prices. It provides a straightforward indication of the model's accuracy.

Mean Squared Error (MSE): The MSE calculates the average of the squared differences between the predicted and actual house prices. It penalizes larger errors more heavily and provides insight into the model's overall performance.

R-squared (R²) Score: The R² score measures the proportion of the variance in the dependent variable (house prices) that is predictable from the independent variables (predictor features). It ranges from 0 to 1, where 1 indicates a perfect fit.

Experimental Results:

The linear regression model was trained on the training set and evaluated on the testing set using the MAE, MSE, and R² score.

The computed performance metrics provide quantitative measures of the model's accuracy, precision, and ability to generalize to unseen data.

Additionally, visualizations such as scatter plots comparing actual vs. predicted house prices can aid in understanding the model's predictive capabilities and identifying any patterns or trends.

Summary

Our experimentation with the linear regression model for house price prediction yielded promising results:

The model demonstrated reasonable accuracy, as indicated by low MAE and MSE values.

The R^2 score suggests that a significant proportion of the variance in house prices can be explained by the predictor variables.

Visual inspection of actual vs. predicted house prices revealed a strong linear relationship, validating the model's predictive capabilities.

Insights on the Experimental Outcome:

The linear regression model serves as a solid baseline for house price prediction, providing interpretable results and demonstrating decent performance.

Further refinement and feature engineering could potentially improve the model's predictive accuracy and robustness.

Future iterations of the project may explore more sophisticated algorithms or ensemble techniques to enhance predictive performance and address potential limitations of linear regression.