The AI Crystal Ball: Unleashing the Magic of House Prediction

The AI Crystal Ball

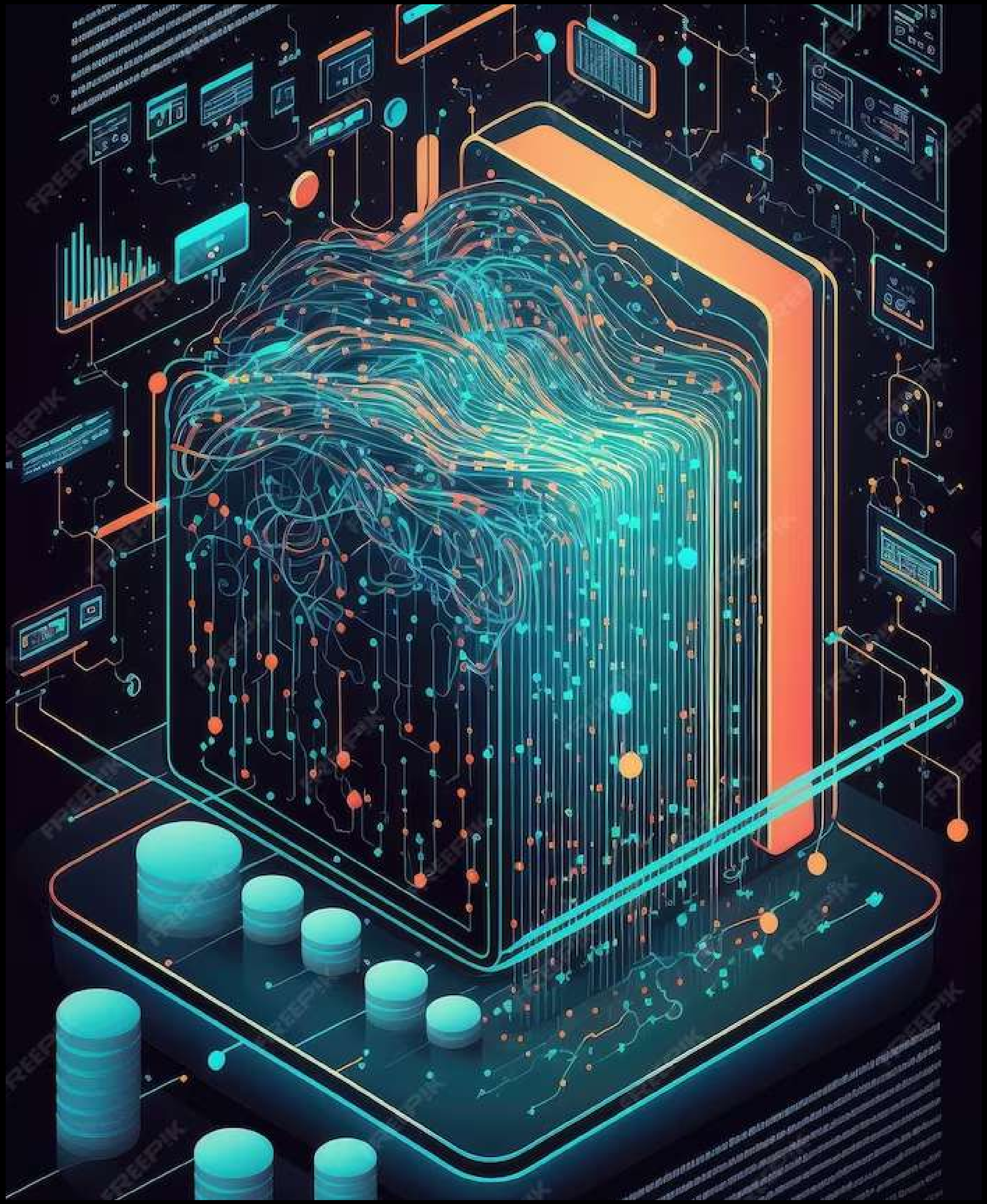
Unleashing the Magic of House Prediction

Introduction

Welcome to the world of *AI House Prediction*! Discover how artificial intelligence can **forecast** house prices and trends with incredible accuracy. Get ready for a journey into the future of real estate!

Understanding House Prediction

Learn the *fundamentals* of house prediction and how AI algorithms analyze **data patterns** to make accurate forecasts. Explore the key factors influencing house prices and how AI can uncover hidden insights.



The Magic of AI

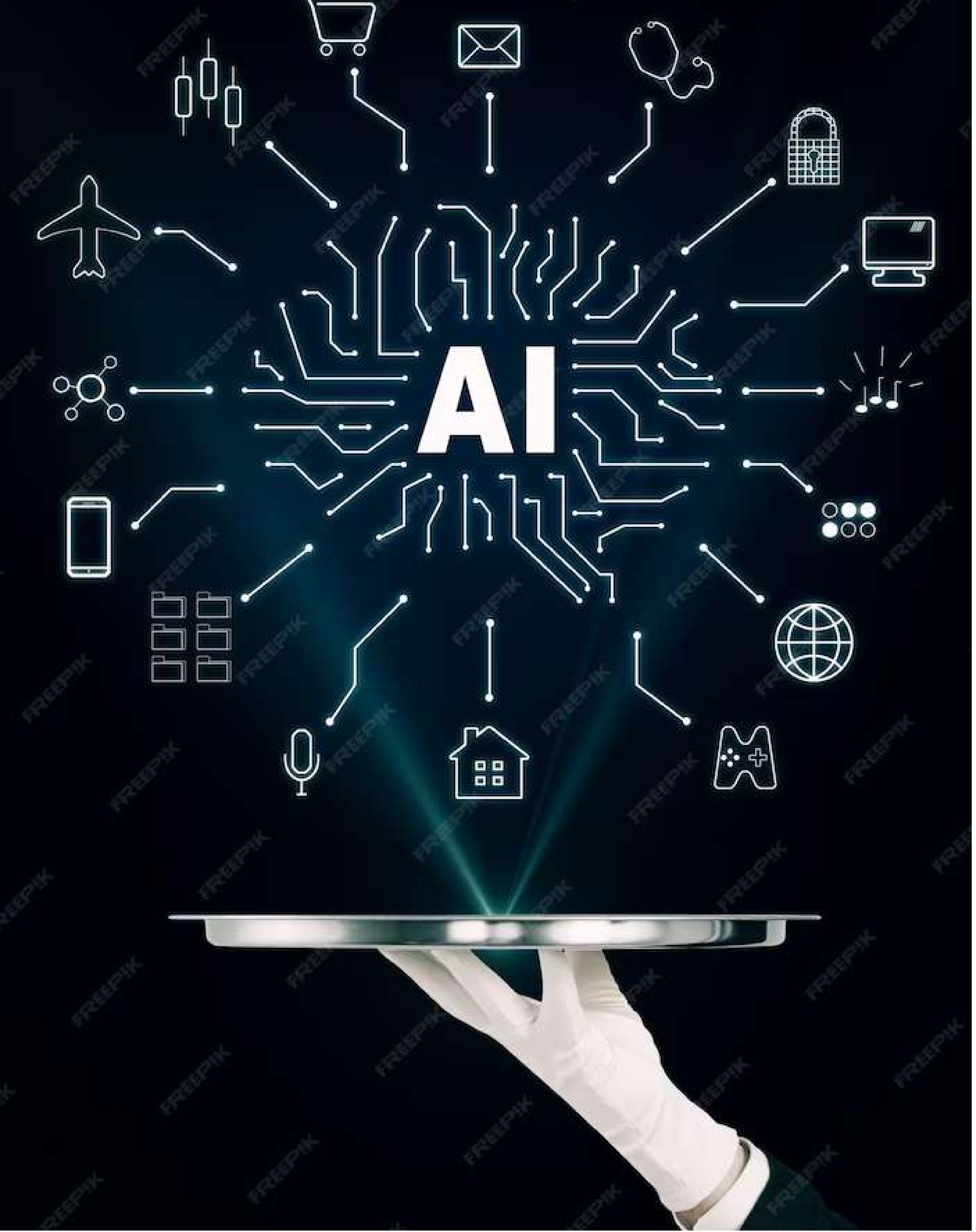
Discover the **magic** behind AI-powered house prediction. Witness how machine learning and deep neural networks can unlock the hidden potential of data, enabling us to predict future house trends with unprecedented accuracy.

Data Collection & Preprocessing

Explore the **data journey** in house prediction. Understand the importance of collecting comprehensive and relevant data, and how preprocessing techniques such as feature scaling and outlier removal enhance the accuracy of predictions.

Feature Engineering

Unleash the power of **feature engineering** in house prediction. Learn how AI algorithms transform raw data into meaningful features, extracting valuable insights that influence house prices. Discover the art of feature selection and creation.



AI Algorithms for House Prediction

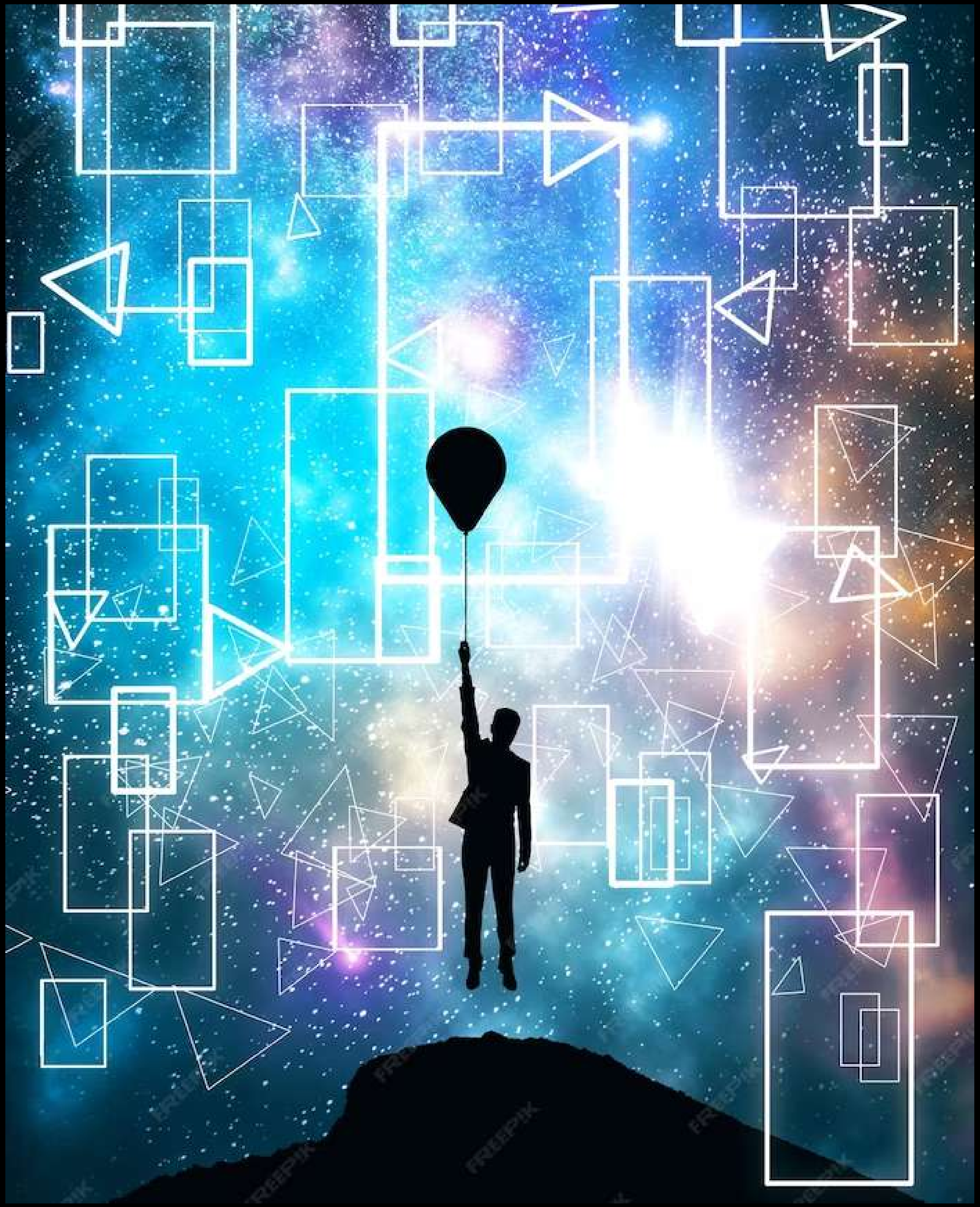
Dive into the world of AI algorithms for house prediction. Explore the strengths and weaknesses of regression algorithms, decision trees, and ensemble methods. Understand how these algorithms make accurate predictions based on historical data.

Evaluating Model Performance

Learn how to evaluate the performance of AI models in house prediction. Explore metrics such as **mean squared error** and **R-squared** to measure accuracy. Discover techniques like cross-validation to ensure reliable predictions.

Real-Life Applications

Witness the real-life applications of AI in house prediction. From assisting homebuyers in making informed decisions to helping real estate agents optimize pricing strategies, AI is revolutionizing the way we navigate the housing market.



Challenges & Ethical Considerations

Explore the challenges and ethical considerations in AI house prediction. Discuss issues like data privacy, algorithmic biases, and potential societal impacts. Understand the importance of responsible AI implementation in the real estate industry.

Future Trends

Get a glimpse into the future of AI house prediction. Discover emerging trends like **predictive analytics**, **automated valuation models**, and **smart home integration**. Explore how AI will continue to shape the real estate landscape.

Benefits of AI in House Prediction

Uncover the benefits of AI-powered house prediction. From improved accuracy and faster decision-making to enhanced market transparency, AI empowers individuals and businesses alike, revolutionizing the way we predict and navigate the housing market.

Key Takeaways

Recap the key takeaways from this exciting journey into AI house prediction. Understand the potential of AI in forecasting house prices, the importance of data quality, and the ethical considerations in responsible AI implementation.

program
import pandas as pd
import numpy as np import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection
import train_test_split
from sklearn.preprocessing
import StandardScaler from sklearn.metrics
importr2_score,mean_absolute_error,mean_square
d_error

- from sklearn.linear_model
- import LinearRegression
- from sklearn.linear_model import Lasso
- from sklearn.ensemble
- import RandomForestRegressor
- from sklearn.svm import SVR import xgboost as

Data set

```
dataset =pd.read_csv('USA_Housing.csv')
dataset.head(10)
```

	Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price	Address
0	79545.458574	5.682861	7.009188	4.09	23086.800503	1.059034e+06	208 Michael Ferry Apt. 674\nLaurabury, NE 3701...
1	79248.642455	6.002900	6.730821	3.09	40173.072174	1.505891e+06	188 Johnson Views Suite 079\nLake Kathleen, CA...
2	61287.067179	5.865890	8.512727	5.13	36882.159400	1.058988e+06	9127 Elizabeth Stravenue\nDanieltown, WI 06482...
3	63345.240046	7.188236	5.586729	3.26	34310.242831	1.260617e+06	USS Barnett\nFPO AP 44820
4	59982.197226	5.040555	7.839388	4.23	26354.109472	6.309435e+05	USNS Raymond\nFPO AE 09386
5	80175.754159	4.988408	6.104512	4.04	26748.428425	1.068138e+06	06039 Jennifer Islands Apt. 443\nTracyport, KS...
6	64698.463428	6.025336	8.147760	3.41	60828.249085	1.502056e+06	4759 Daniel Shoals Suite 442\nNguyenburgh, CO ...
7	78394.339278	6.989780	6.620478	2.42	36516.358972	1.573937e+06	972 Joyce Viaduct\nLake William, TN 17778-6483
8	59927.660813	5.362126	6.393121	2.30	29387.396003	7.988695e+05	USS Gilbert\nFPO AA 20957

USA_housing.shape

(5000, 7)

USA_housing.info()

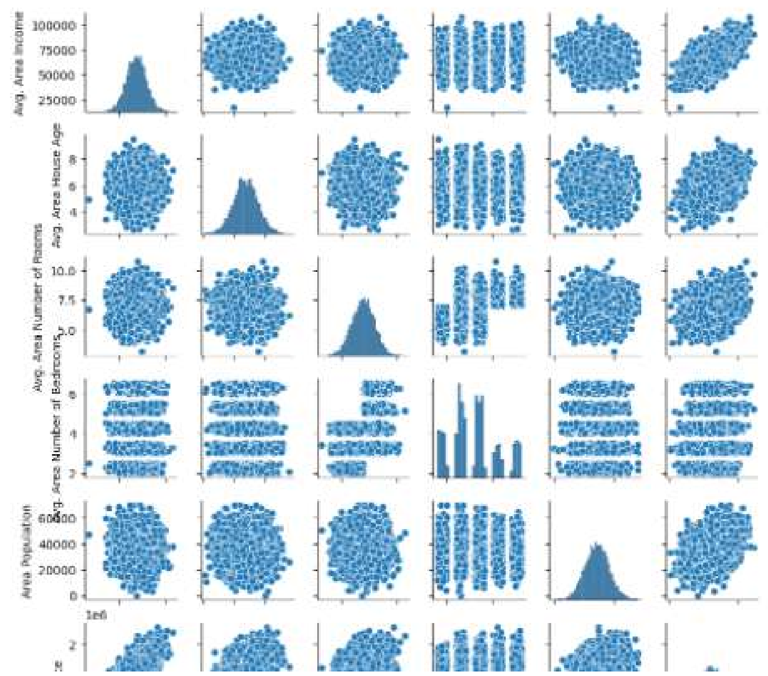
```
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   Avg. Area Income              5000 non-null   float64
 1   Avg. Area House Age           5000 non-null   float64
 2   Avg. Area Number of Rooms     5000 non-null   float64
 3   Avg. Area Number of Bedrooms  5000 non-null   float64
 4   Area Population               5000 non-null   float64
 5   Price                         5000 non-null   float64
 6   Address                       5000 non-null   object
dtypes: float64(6), object(1)
memory usage: 273.6+ KB

In [5]:
```

USA_housing.describe()

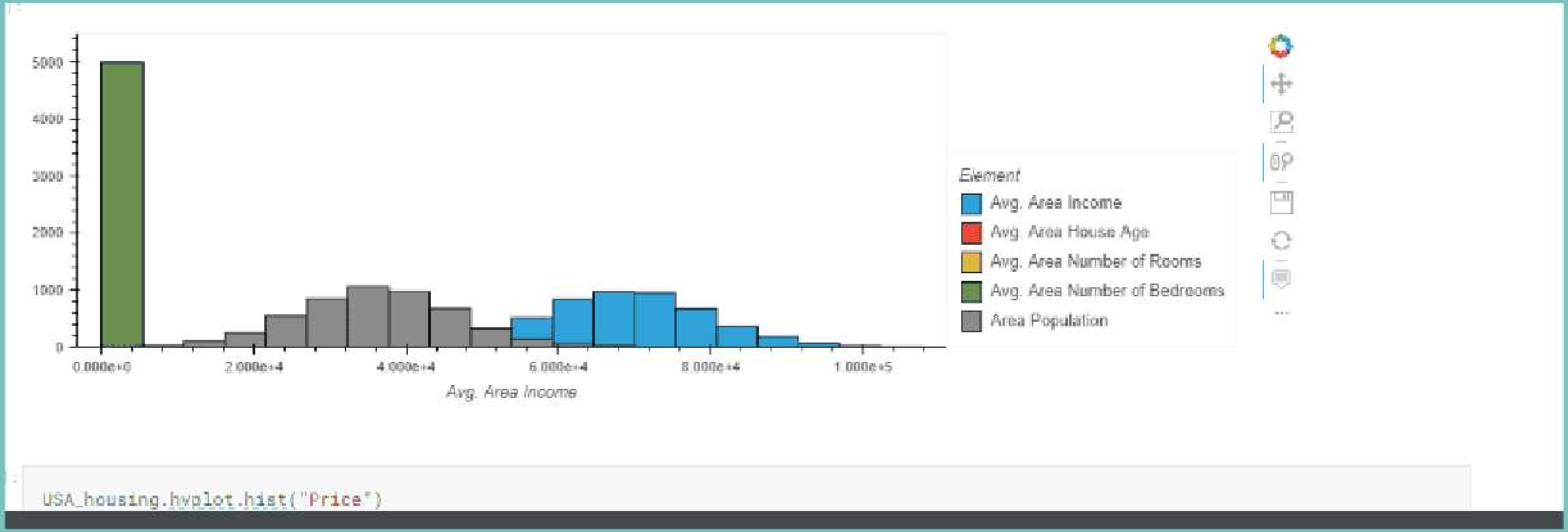
5]:

	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

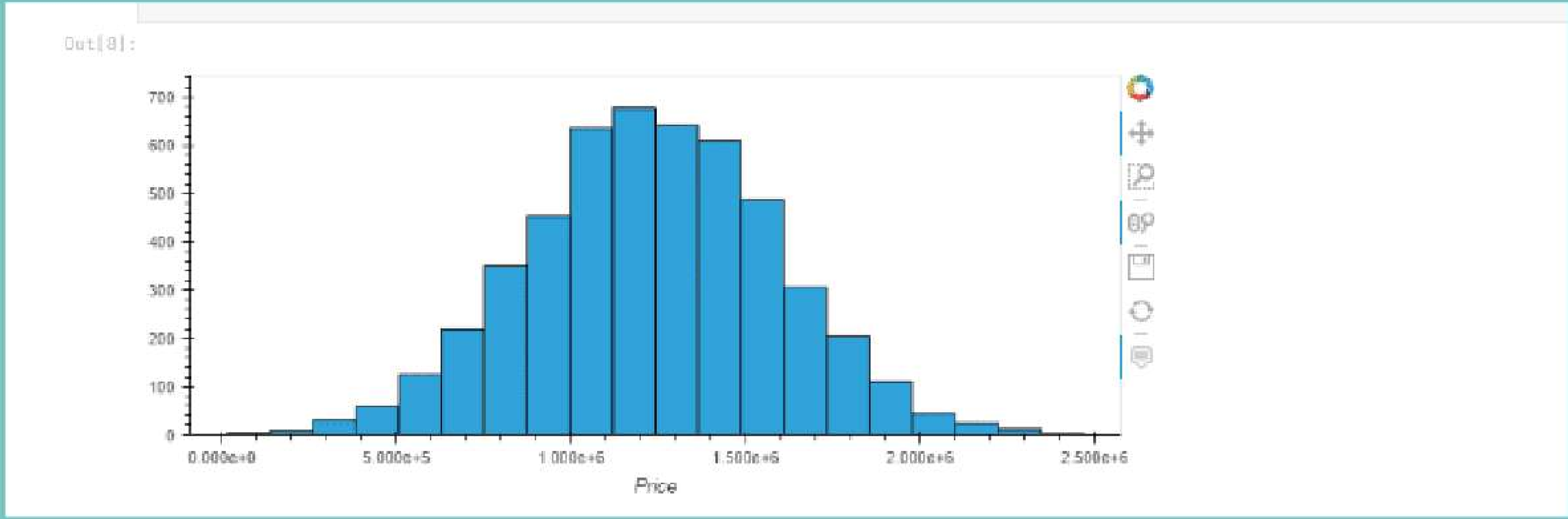
sns.pairplot(USA_housing, height = 1.5)



USA_housing.hvplot.hist(by='Price', subplots=False, width=1000)



USA_housing.hvplot.hist("Price")



```
import pandas as pd
import numpy as np
import pickle

#visualisation
import plotly.express as px
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import seaborn as sns

#sns.set_style('whitegrid')

# Model
from sklearn.svm import SVR
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression, SGDRegressor
from sklearn.linear_model import ElasticNet

#perfomance
from sklearn.metrics import mean_squared_error,r2_score

# Load the dataset
df= pd.read_csv("USA_Housing.csv")
```

```
df.head()
```

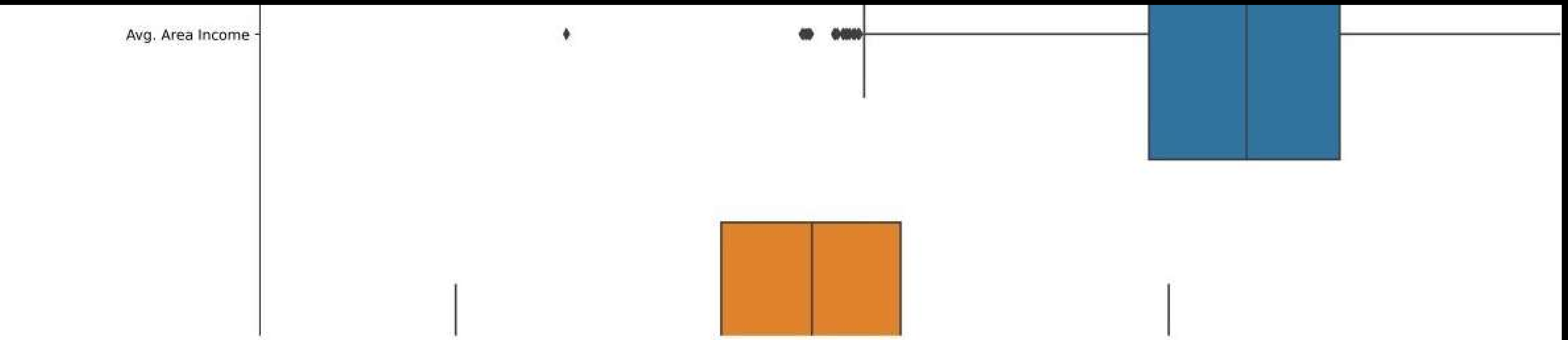
	Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price	
0	79545.458574	5.682861	7.009188	4.09	23086.800503	1.059034e+06	208 M
1	79248.642455	6.002900	6.730821	3.09	40173.072174	1.505891e+06	188 John
2	61287.067179	5.865890	8.512727	5.13	36882.159400	1.058988e+06	9127

```
df.drop('Address',axis=1,inplace=True)
```

```
df.isna().sum()
```

```
Avg. Area Income                0
Avg. Area House Age             0
Avg. Area Number of Rooms       0
Avg. Area Number of Bedrooms    0
Area Population                 0
Price                           0
dtype: int64
```

```
plt.figure(figsize=(20,8),dpi=400)
sns.boxplot(data=df[[ 'Avg. Area Income', 'Area Population']],orient='h')
plt.show()
```




```
# Split the data into training and testing sets
X = df.drop("Price", axis=1)  # Features
y = df["Price"]  # Target variable

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

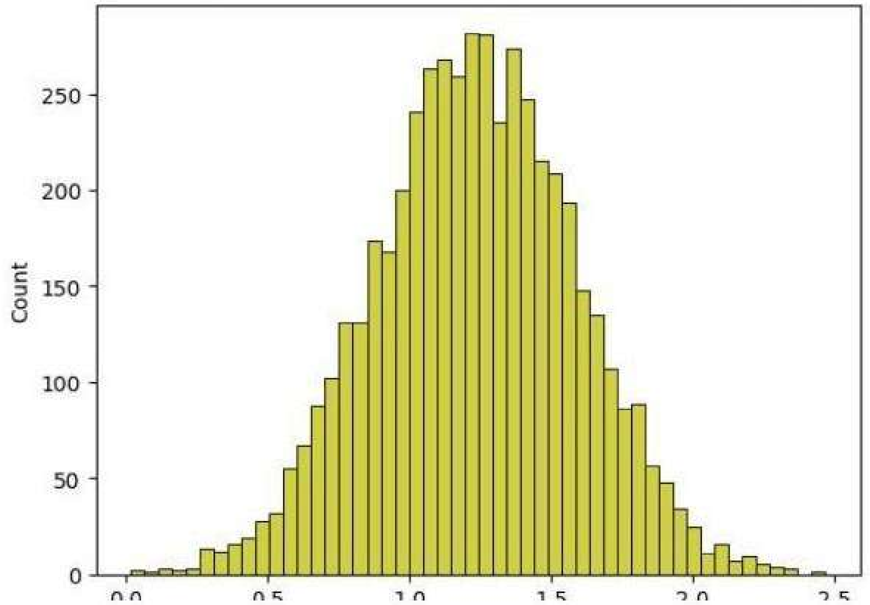
```
X = df.drop(['Price'],axis=1)
y = df['Price']
print(X.shape)
```

```
(5000, 5)
```

** DATA VISUALIZATION**

```
sns.histplot(df, x='Price', bins=50, color='y')
```

```
<Axes: xlabel='Price', ylabel='Count'>
```



Dividing Dataset in to features and target variable

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

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=101)
```

```
Y_train.head()
```

```
3413    1.305210e+06
1610    1.400961e+06
3459    1.048640e+06
4293    1.231157e+06
1039    1.391233e+06
Name: Price, dtype: float64
```

Standardizing the data

```
sc = StandardScaler()
X_train_scal = sc.fit_transform(X_train)
X_test_scal = sc.fit_transform(X_test)
```

Model Building and Evaluation

```
model_lr=LinearRegression()
```

```
model_lr.fit(X_train_scal, Y_train)
```

```
▾ LinearRegression
LinearRegression()
```

Predicting Prices

```
Prediction1 = model_lr.predict(X_test_scal)
```

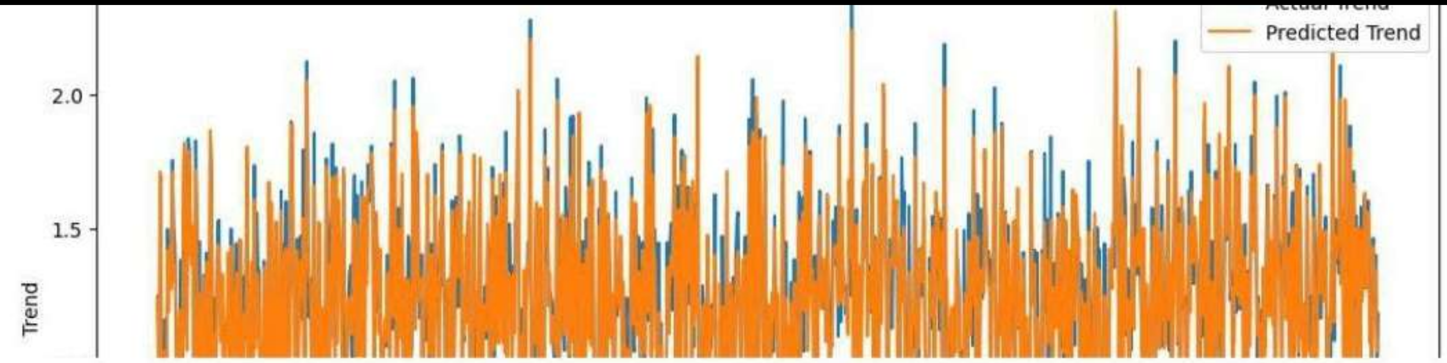
```
model_svr = SVR()
```

```
model_svr.fit(X_train_scal, Y_train)
```

```
Prediction2 = model_svr.predict(X_test_scal)
```

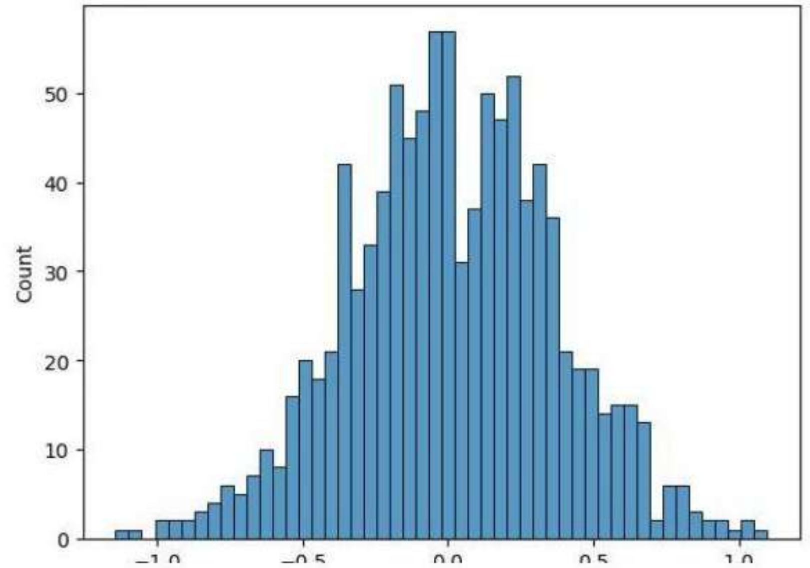
Evaluation of Predicted Data

```
plt.figure(figsize=(12,6))
plt.plot(np.arange(len(Y_test)), Y_test, label='Actual Trend')
plt.plot(np.arange(len(Y_test)), Prediction1, label='Predicted Trend')
plt.xlabel('Data')
plt.ylabel('Trend')
plt.legend()
plt.title('Actual vs Predicted')
```




```
sns.histplot((Y_test-Prediction2), bins=50)
```

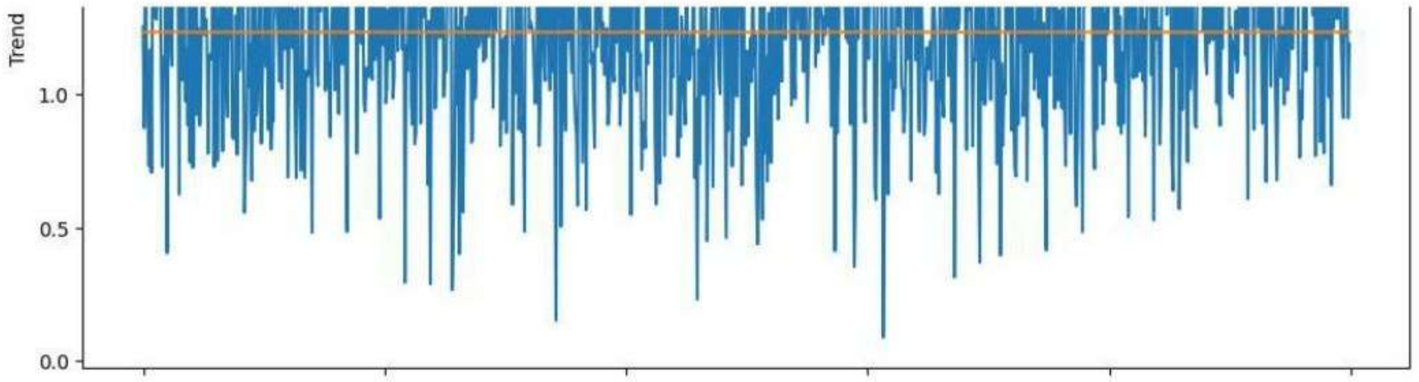
```
<Axes: xlabel='Price', ylabel='Count'>
```

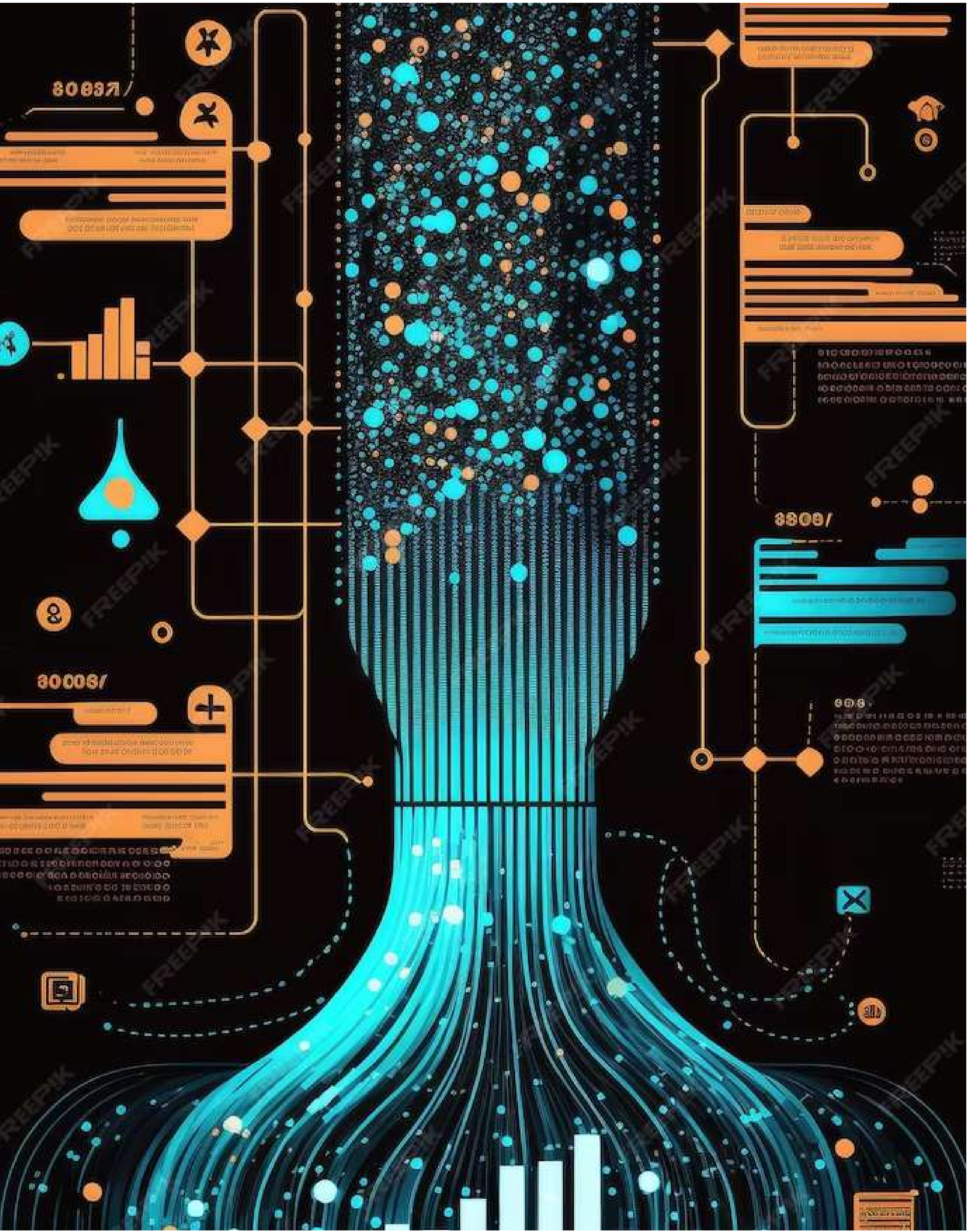



```
plt.figure(figsize=(12,6))
plt.plot(np.arange(len(Y_test)), Y_test, label='Actual Trend')
plt.plot(np.arange(len(Y_test)), Prediction2, label='Predicted Trend')
plt.xlabel('Data')
plt.ylabel('Trend')
plt.legend()
plt.title('Actual vs Predicted')
```


```
print(r2_score(Y_test, Prediction2))
print(mean_squared_error(Y_test, Prediction2))
print(mean_squared_error(Y_test, Prediction2))
```

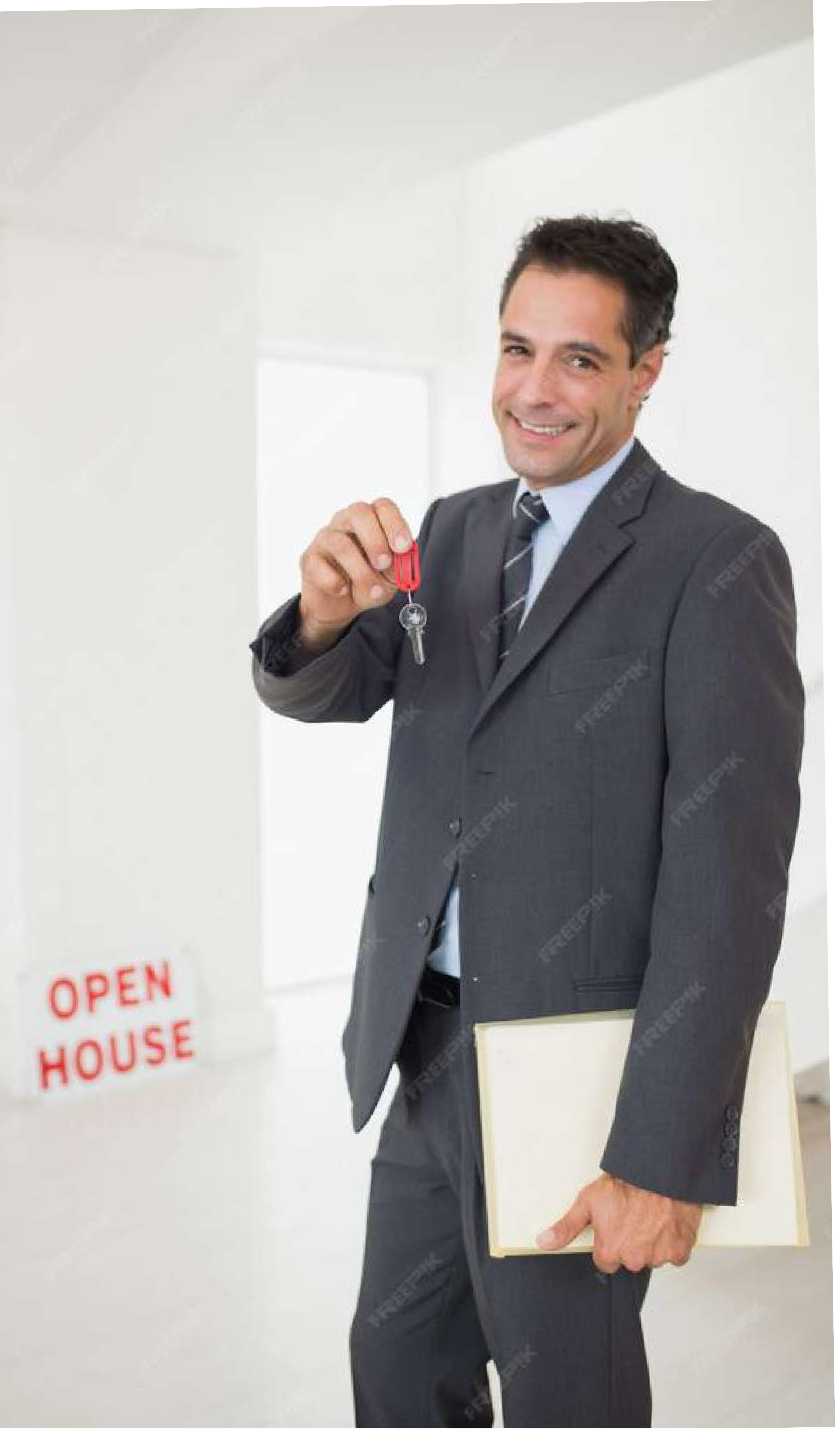
```
-0.0006222175925689744
128209033251.4034
128209033251.4034
```





Benefits of AI in House Prediction

Artificial Intelligence offers several advantages in house prediction. It can **analyze vast amounts of data** and identify patterns that humans might miss. AI algorithms can **improve accuracy** by considering numerous factors and historical trends. Additionally, AI-powered systems can **increase efficiency** by automating repetitive tasks, saving time and resources.



Real-world Applications

AI-powered house prediction has diverse applications. It helps **real estate agents** in estimating property values accurately, enabling informed decisions. **Homebuyers** can leverage AI to assess fair prices and negotiate better deals. Financial institutions utilize AI for **risk assessment** during mortgage approvals. Overall, AI enhances decision-making for all stakeholders.

Future Possibilities

The future of AI-powered house prediction is promising. Advancements in **machine learning** and **data analysis** techniques will further enhance accuracy. Integration with **Internet of Things** (IoT) devices can provide real-time data for better predictions. As AI continues to evolve, we can expect more sophisticated models and improved outcomes.

Conclusion

In conclusion, AI is transforming the world of house prediction, unlocking the magic of accurate forecasts. Embrace the power of AI, and let it guide you through the enchanting realm of real estate.