MCST1043 RESEARCH DESIGN AND ANALYSIS IN DATA SCIENCE

Big data driven: Forecast of global real estate market ups and downs in some regions

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

CANDIDATE : Yang Mu(MCS241045)
SUPERVISOR : DR. Mohd Shahizan Othman
DATE : 17/06/2025

FACULTY OF COMPUTING,
UNIVERSITI TEKNOLOGI MALAYSIA
www.utm.my
innovative • entrepreneurial • global

univteknologimalaysia utm_my utmofficial

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

Introduction

Methodology

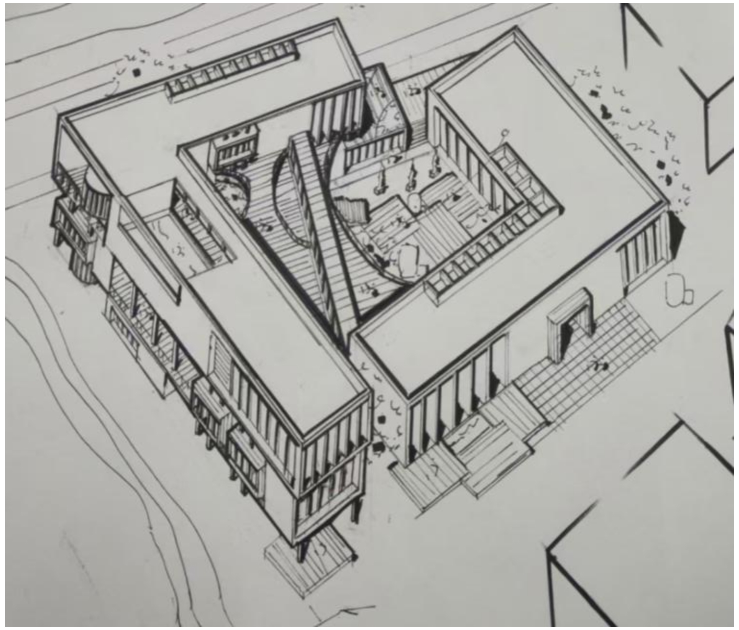
Model Design

Model improvement

Conclusion

INTRODUCTION

Real estate accounts for approximately 60% of people's assets, and the ups and downs of the real estate market also have a profound impact on people's lives



Project Overview

Question: Identify the factors influencing housing prices and design a better model for prediction

Solution: Use regression for factor judgment and select several models for comparison

- Dataset: Six-month changes in housing prices in a certain area (approximately 600,000 entries), as well as some datasets of other areas

Methodology: Currently, there are three models in total, which have undergone comprehensive preprocessing and optimization

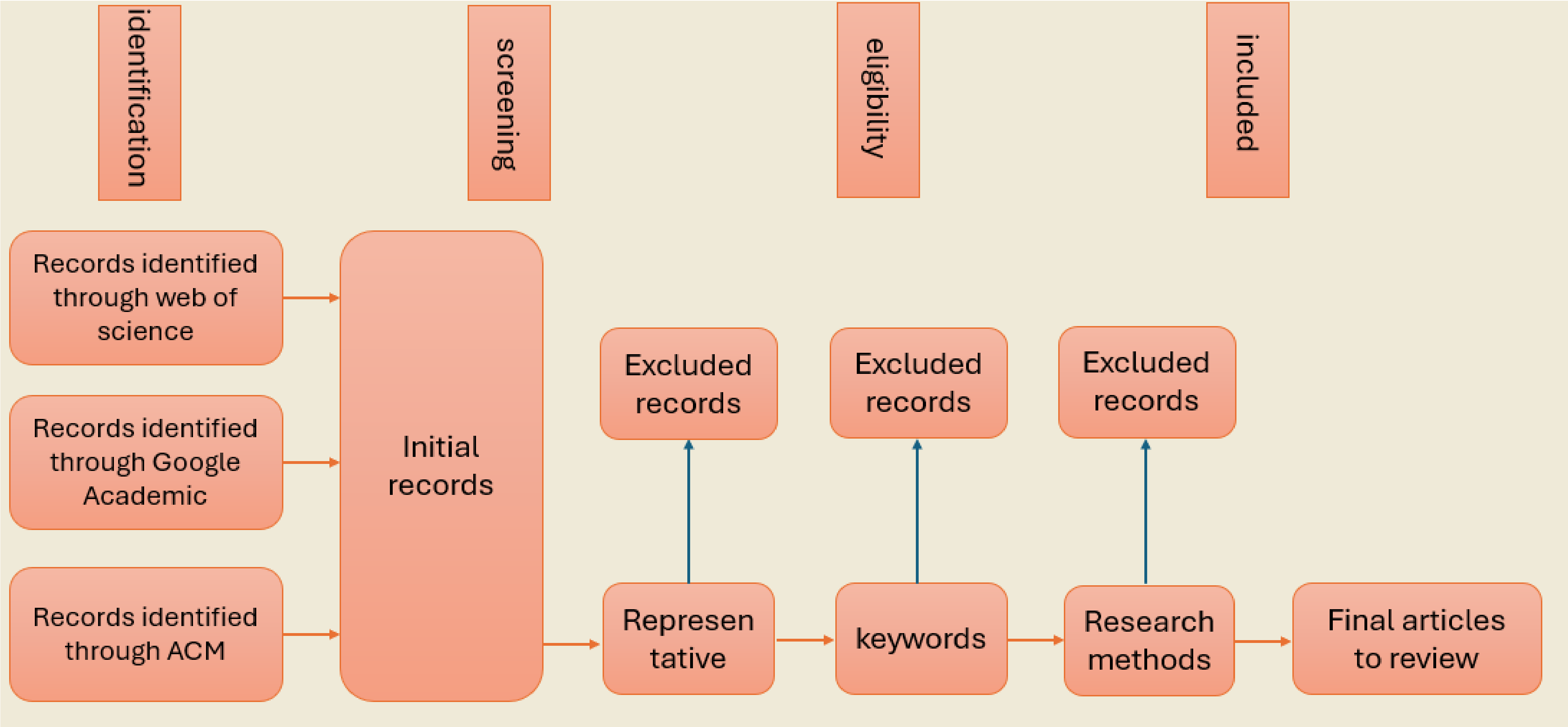
Overall Goal

By identifying the key influencing factors and comparing the three models, an optimized housing price prediction model is developed.

Sub-Goal	Description
OBJ1	Identify influential factors in housing price fluctuations using regression analysis. *Output: Ranked list of key drivers*
OBJ2	Optimize data preprocessing for the primary dataset (600k entries) and supplementary regional datasets.
OBJ3	Develop and train three distinct prediction models with hyperparameter optimization. *Output: 3 benchmarked models*
OBJ4	Validate model generalizability using datasets from secondary areas. *Output: Cross-area performance metrics (RMSE, R^2).*
OBJ5	Select the highest-accuracy model for deployment. *Output: Final model with documented performance superiority.*

INTRODUCTION

Literature



identification

screening

eligibility

included

Records identified through web of science

Records identified through Google Academic

Records identified through ACM

Initial records

Represen tative

keywords

Research methods

Final articles to review

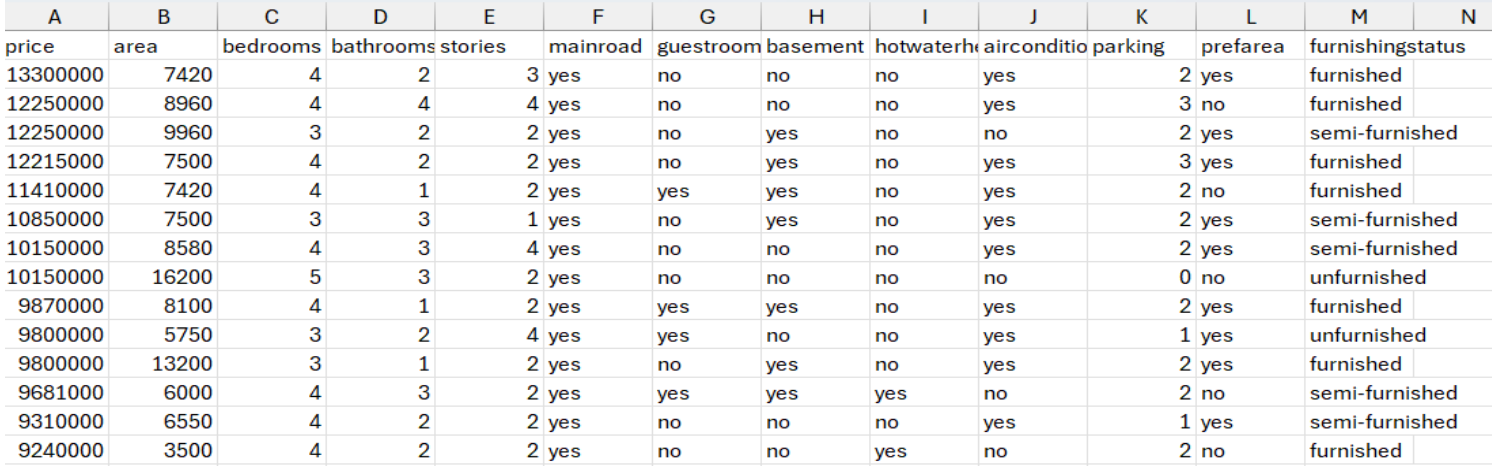
Excluded records

Excluded records

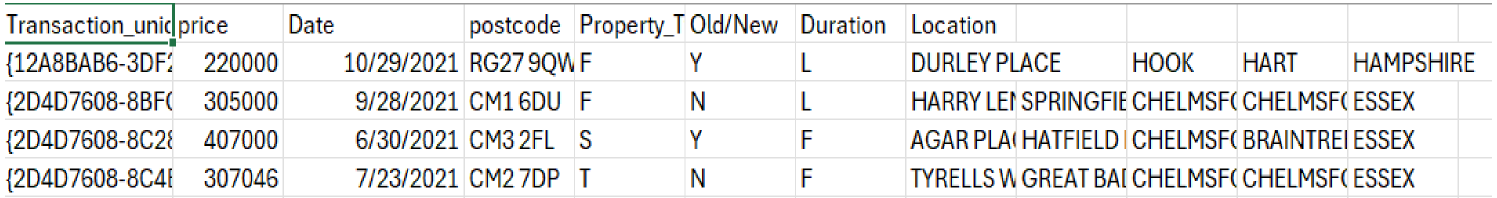
Excluded records

INTRODUCTION

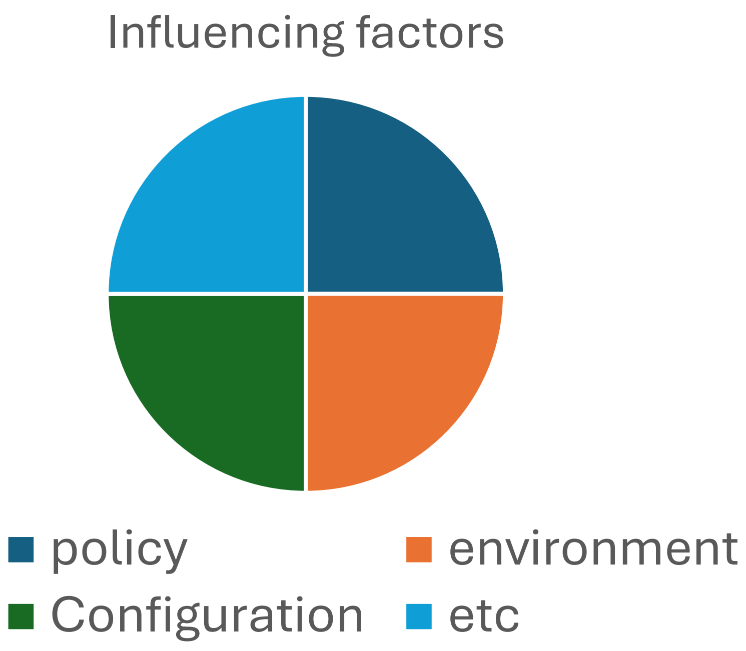
Dataset



A	B	C	D	E	F	G	H	I	J	K	L	M	N
price	area	bedrooms	bathrooms	stories	mainroad	guestroom	basement	hotwaterh	airconditio	parking	prefarea	furnishingstatus	
13300000	7420	4	2	3	yes	no	no	no	yes	2	yes	furnished	
12250000	8960	4	4	4	yes	no	no	no	yes	3	no	furnished	
12250000	9960	3	2	2	yes	no	yes	no	no	2	yes	semi-furnished	
12215000	7500	4	2	2	yes	no	yes	no	yes	3	yes	furnished	
11410000	7420	4	1	2	yes	yes	yes	no	yes	2	no	furnished	
10850000	7500	3	3	1	yes	no	yes	no	yes	2	yes	semi-furnished	
10150000	8580	4	3	4	yes	no	no	no	yes	2	yes	semi-furnished	
10150000	16200	5	3	2	yes	no	no	no	no	0	no	unfurnished	
9870000	8100	4	1	2	yes	yes	yes	no	yes	2	yes	furnished	
9800000	5750	3	2	4	yes	yes	no	no	yes	1	yes	unfurnished	
9800000	13200	3	1	2	yes	no	yes	no	yes	2	yes	furnished	
9681000	6000	4	3	2	yes	yes	yes	yes	no	2	no	semi-furnished	
9310000	6550	4	2	2	yes	no	no	no	yes	1	yes	semi-furnished	
9240000	3500	4	2	2	yes	no	no	yes	no	2	no	furnished	



Transaction_unic	price	Date	postcode	Property_T	Old/New	Duration	Location				
{12A8BAB6-3DF2	220000	10/29/2021	RG27 9QW	F	Y	L	DURLEY PLACE		HOOK	HART	HAMPSHIRE
{2D4D7608-8BF(	305000	9/28/2021	CM1 6DU	F	N	L	HARRY LEI	SPRINGFIE	CHELMSF(	CHELMSF(	ESSEX
{2D4D7608-8C28	407000	6/30/2021	CM3 2FL	S	Y	F	AGAR PLA(	HATFIELD I	CHELMSF(	BRAINTREI	ESSEX
{2D4D7608-8C4I	307046	7/23/2021	CM2 7DP	T	N	F	TYRELLS W	GREAT BAI	CHELMSF(	CHELMSF(	ESSEX

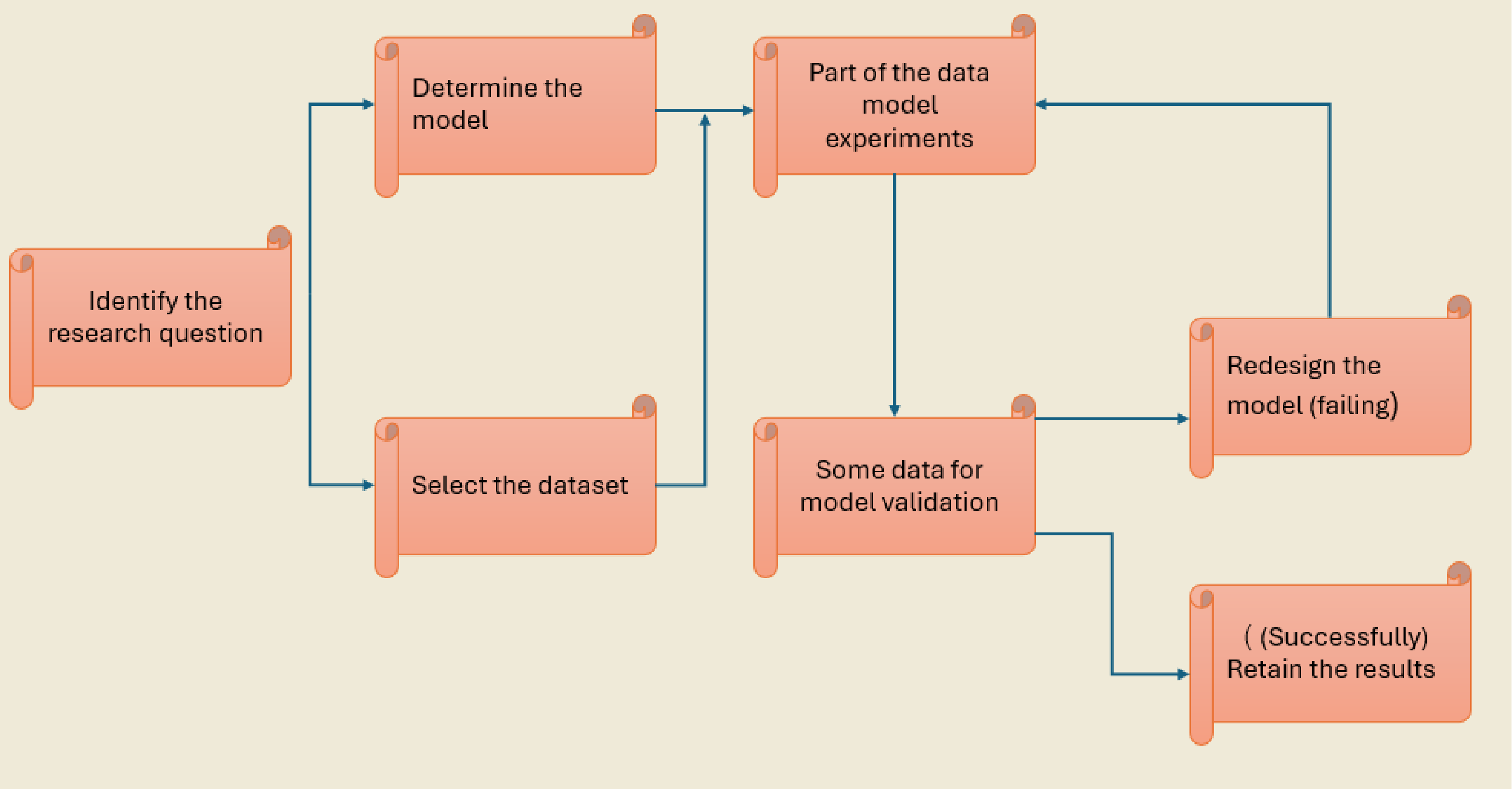


IMDB MOVIE REVIEWS DATASET	Details
Source	Kaggle
Size	600k rows
Length variation	1 to 50 words
variate	20

Methodology

INTRODUCTION

Executive Summary



Identify the research question

Determine the model

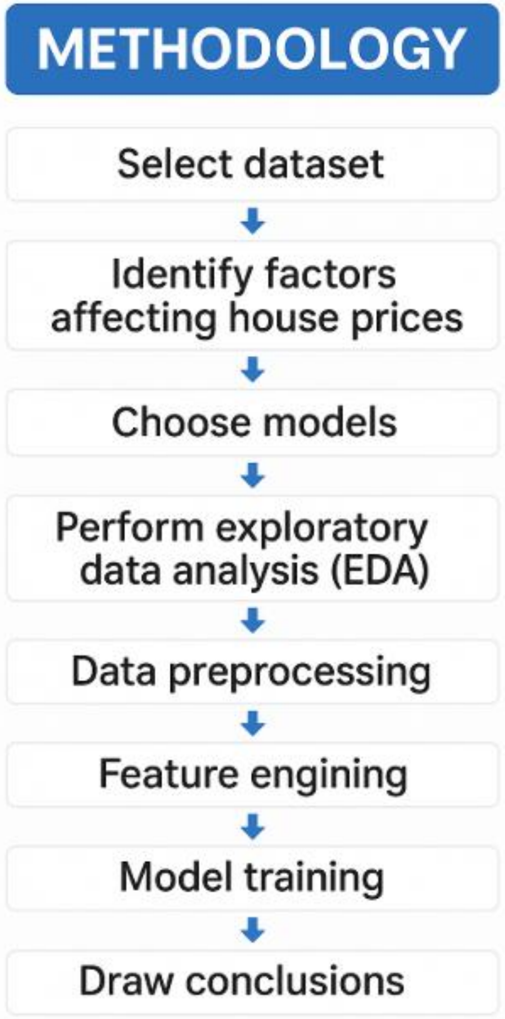
Select the dataset

Part of the data model experiments

Some data for model validation

Redesign the model (failing)

((Successfully) Retain the results



METHODOLOGY

Select dataset

Identify factors affecting house prices

Choose models

Perform exploratory data analysis (EDA)

Data preprocessing

Feature engining

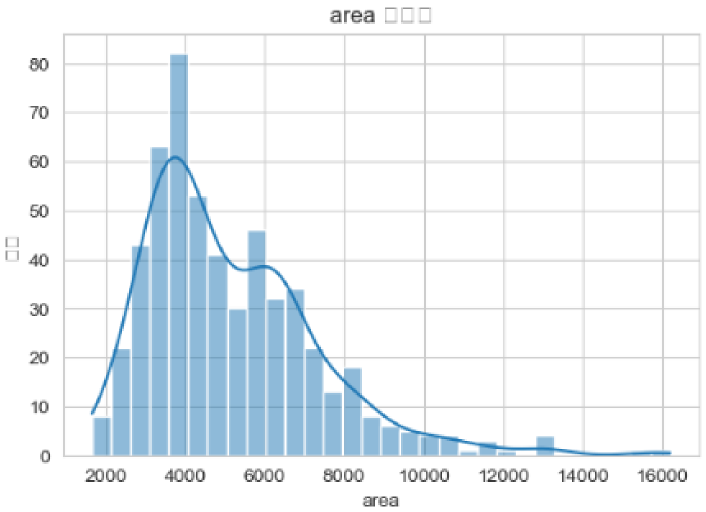
Model training

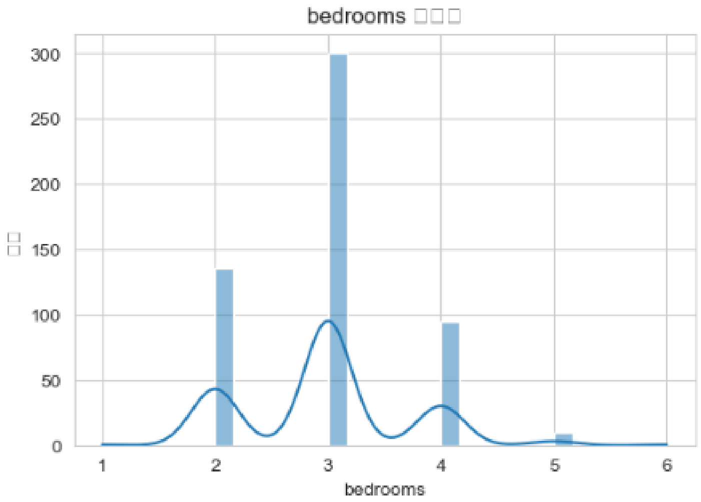
Draw conclusions

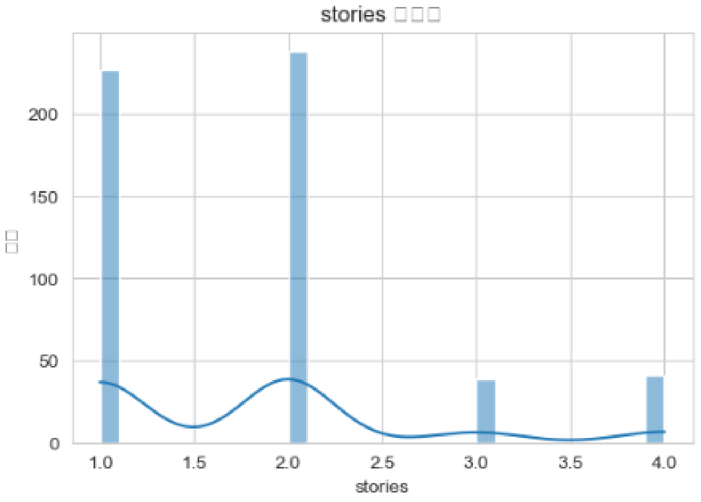
Judgment of influencing factors
Screen out strong relevant factors → Retain
Filter out independent factors → Remove

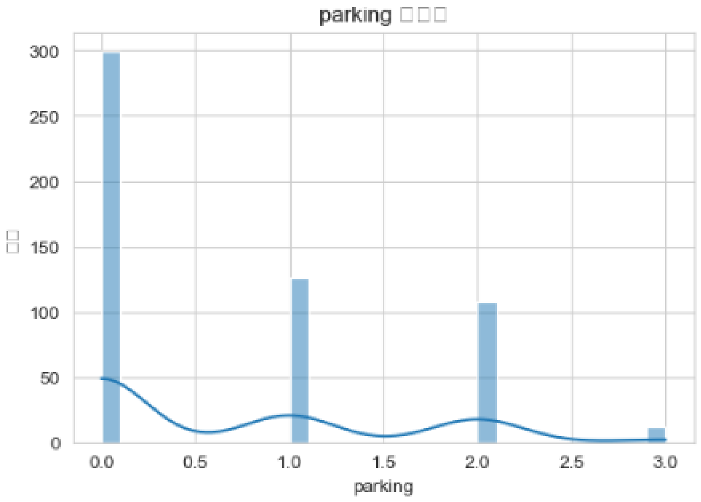
Methodology

EDA

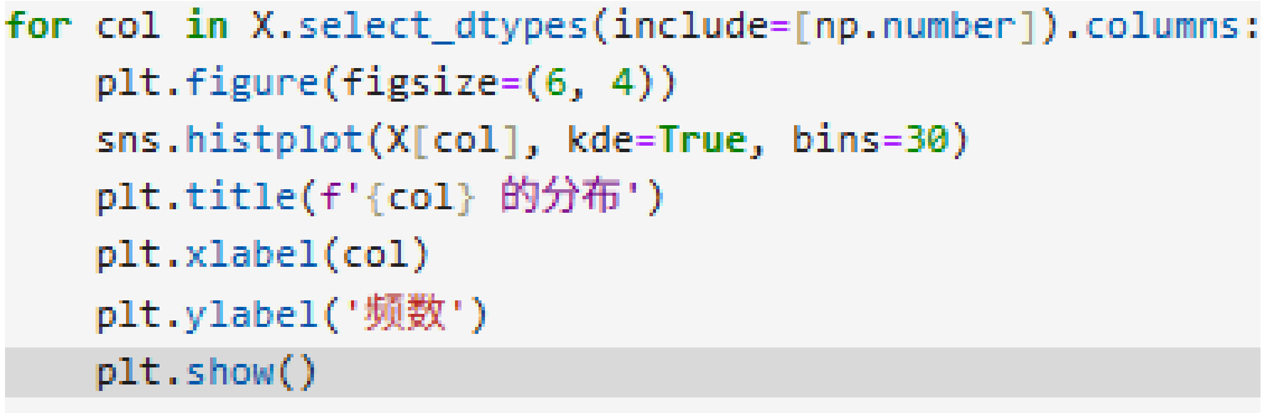








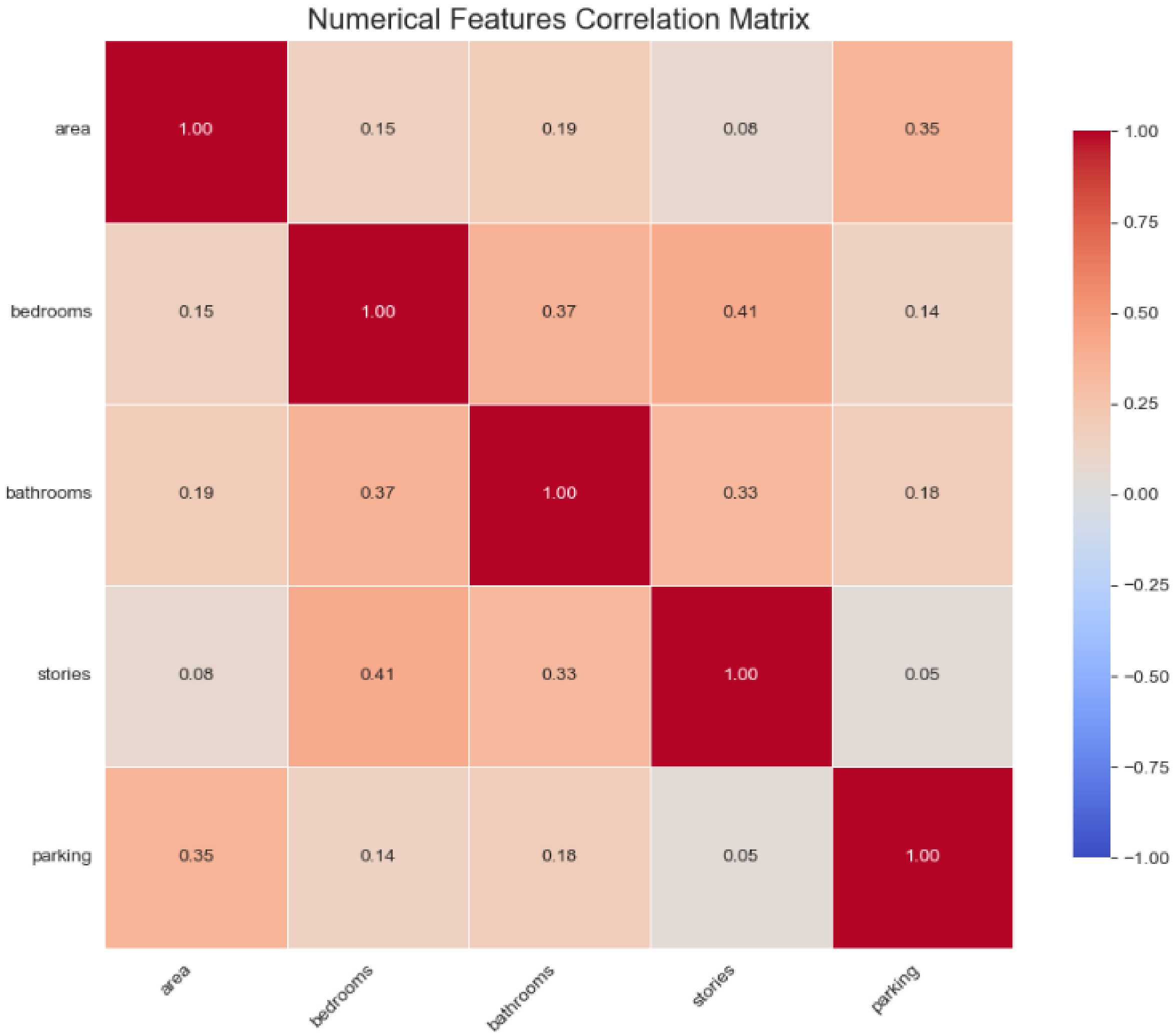
From this, the internal condition distribution of the data set can be seen

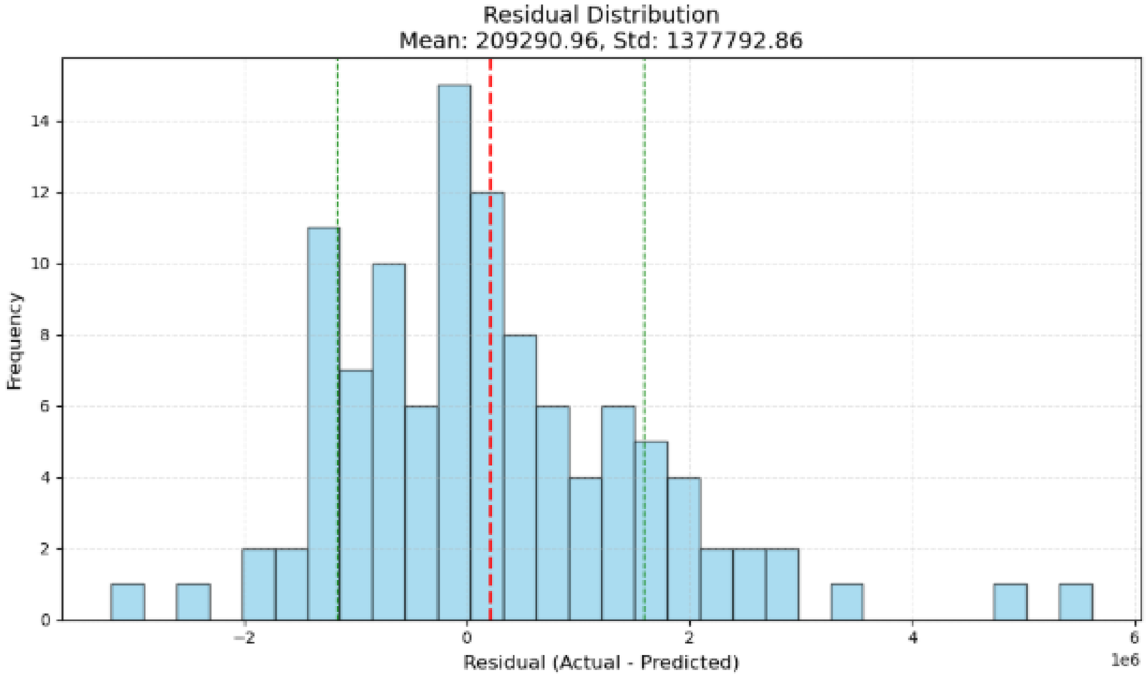


```
for col in X.select_dtypes(include=[np.number]).columns:
    plt.figure(figsize=(6, 4))
    sns.histplot(X[col], kde=True, bins=30)
    plt.title(f'{col} 的分布')
    plt.xlabel(col)
    plt.ylabel('频数')
    plt.show()
```

Achievement Display

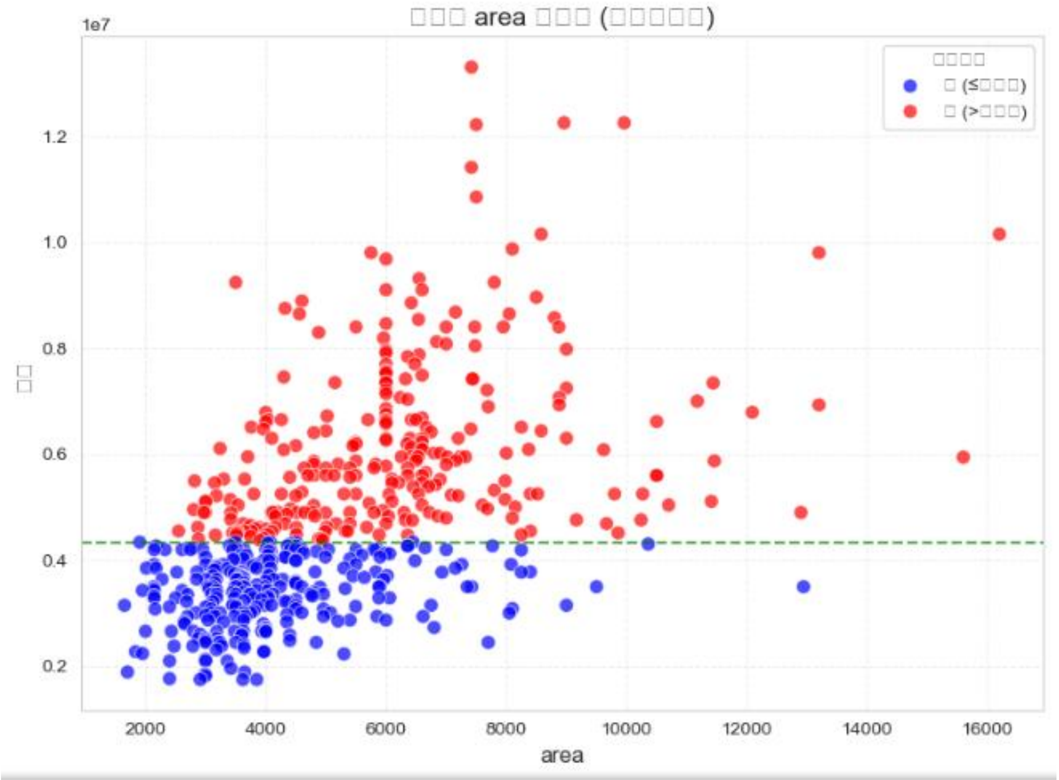
EDA

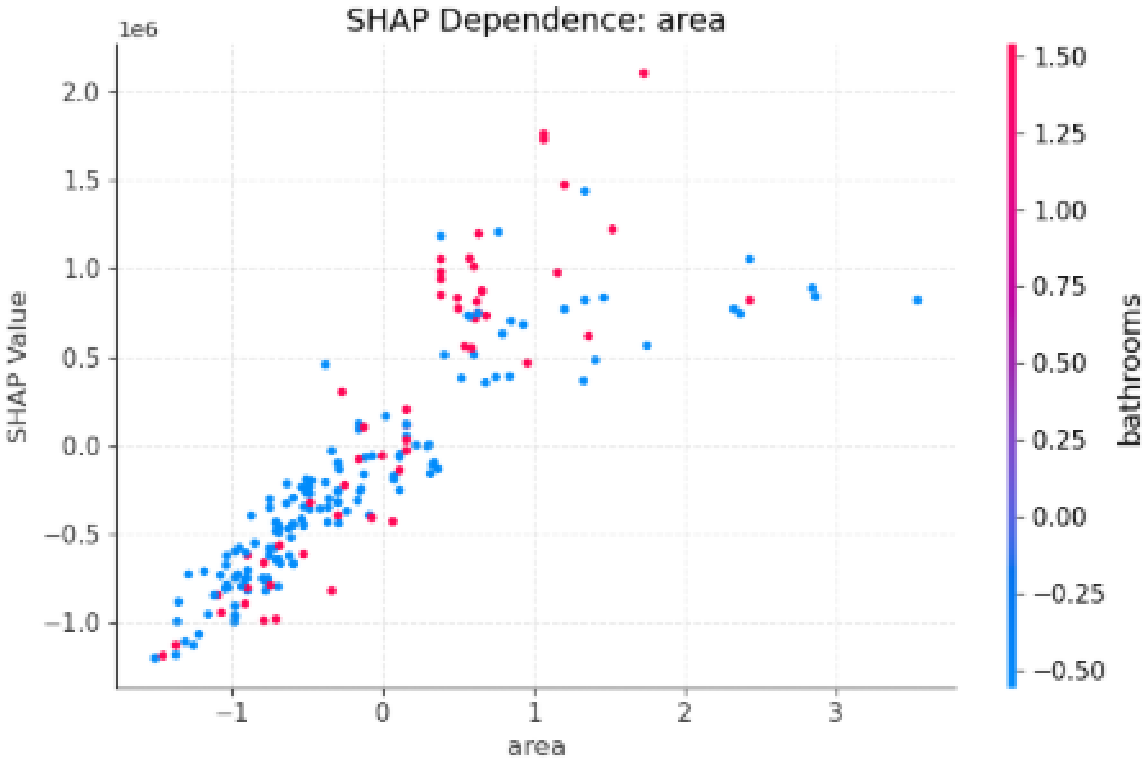




Methodology

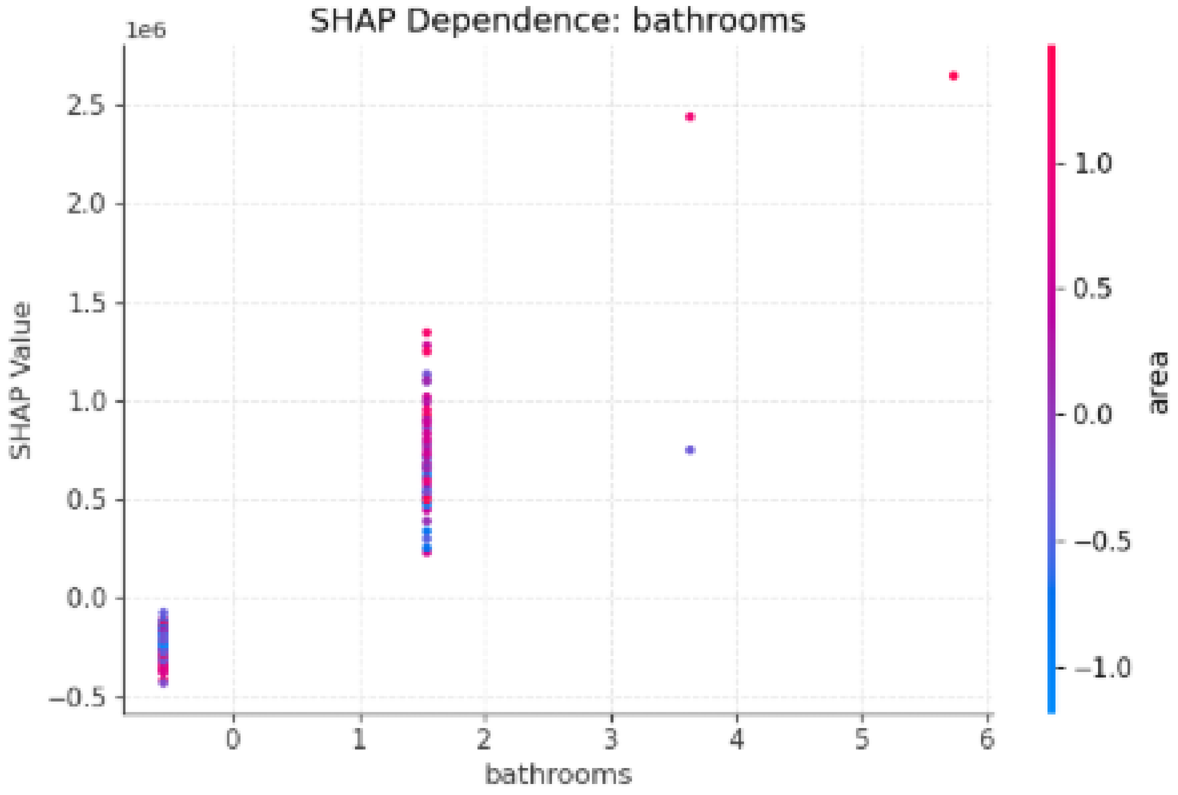
EDA

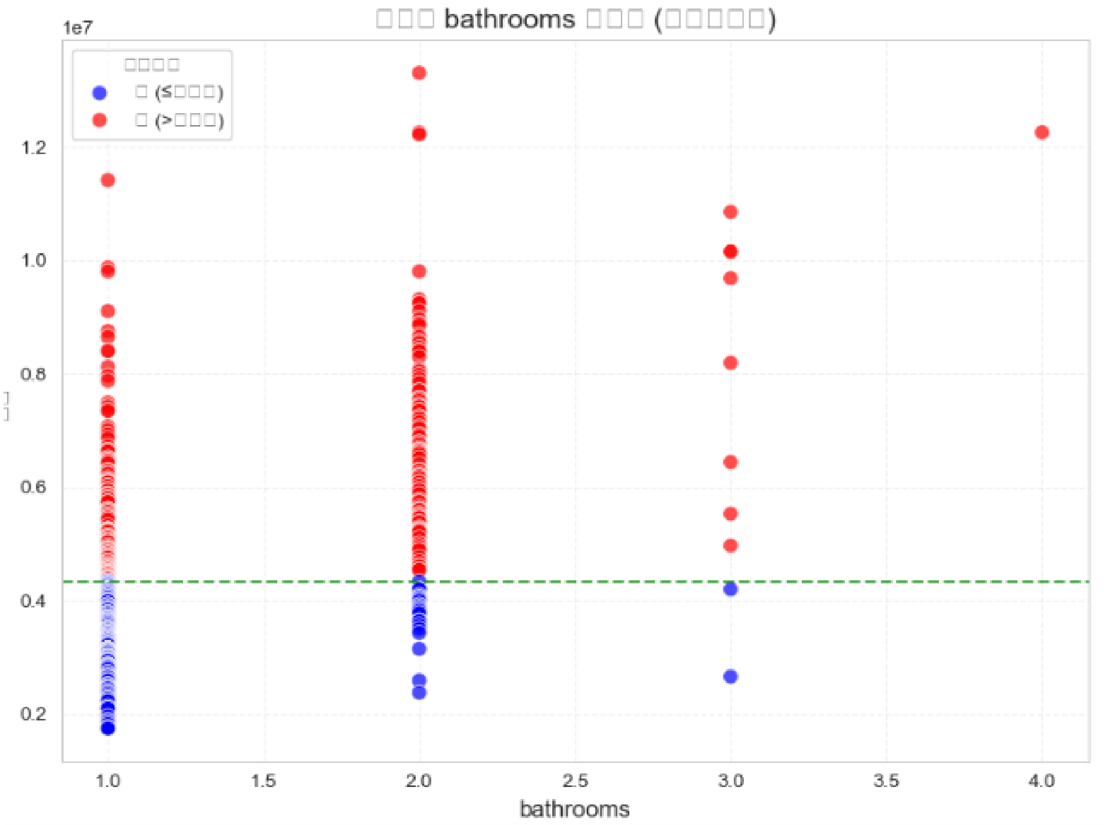




The relationship between area and house price is positively correlated and close

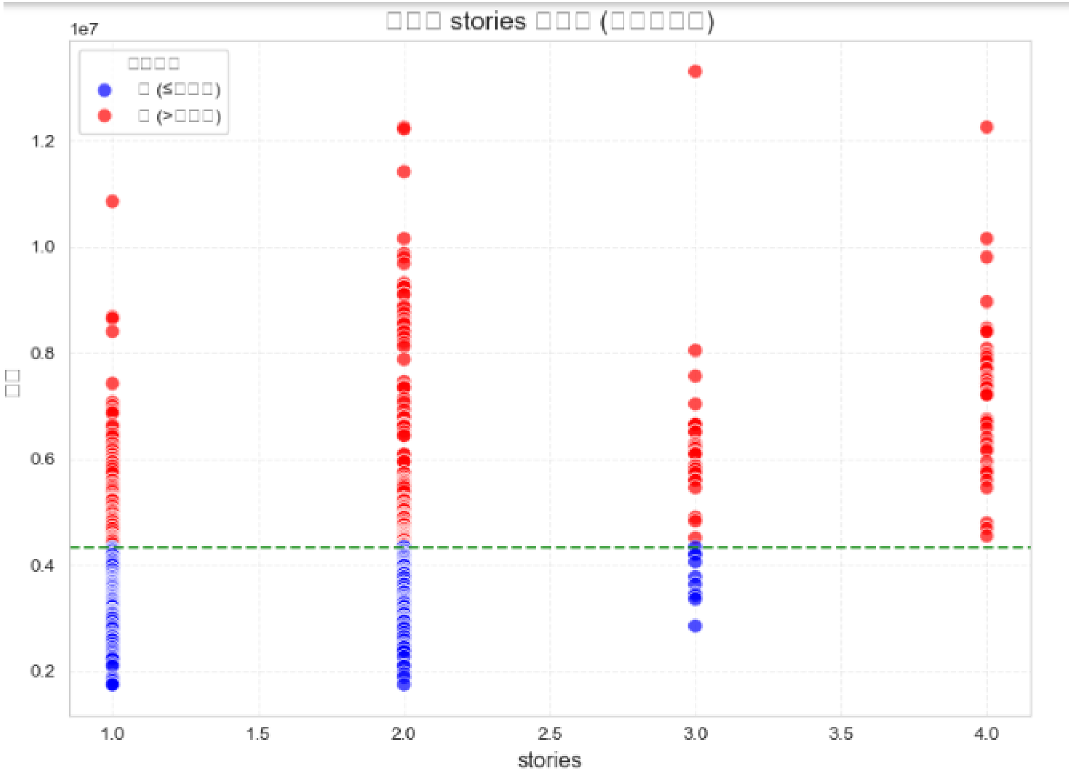
The preferred region is independent of the area, indicating that it is an independent premium factor


SHAP Dependence: bathrooms
1e6
SHAP Value
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
0
1
2
3
4
5
6
bathrooms
area
1.0
0.5
0.0
−0.5
−1.0

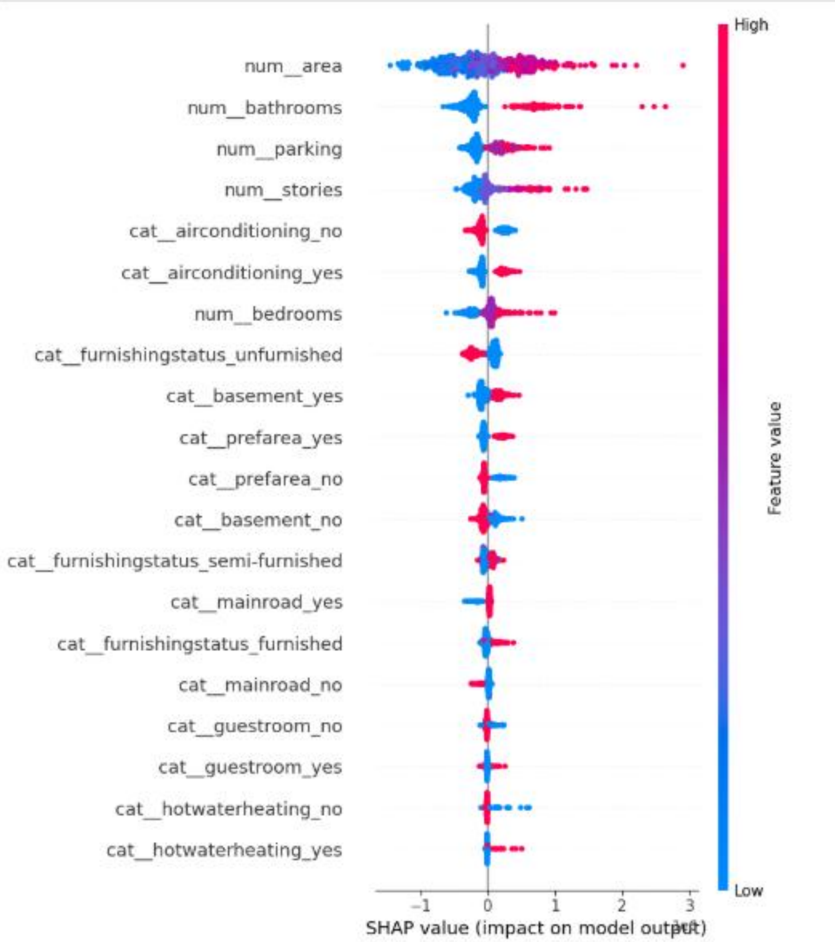

□□□ bathrooms □□□ (□□□□□)
1e7
□□□□
□ (≤□□□)
□ (>□□□)
1.2
1.0
0.8
0.6
0.4
0.2
1.0
1.5
2.0
2.5
3.0
3.5
4.0
bathrooms

Methodology

EDA

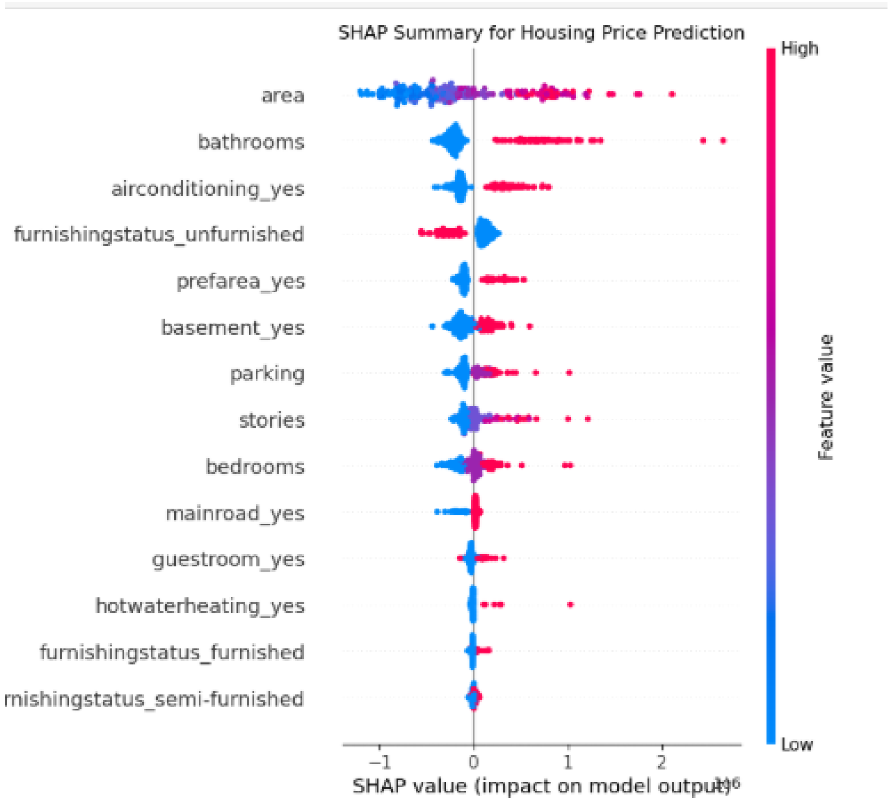


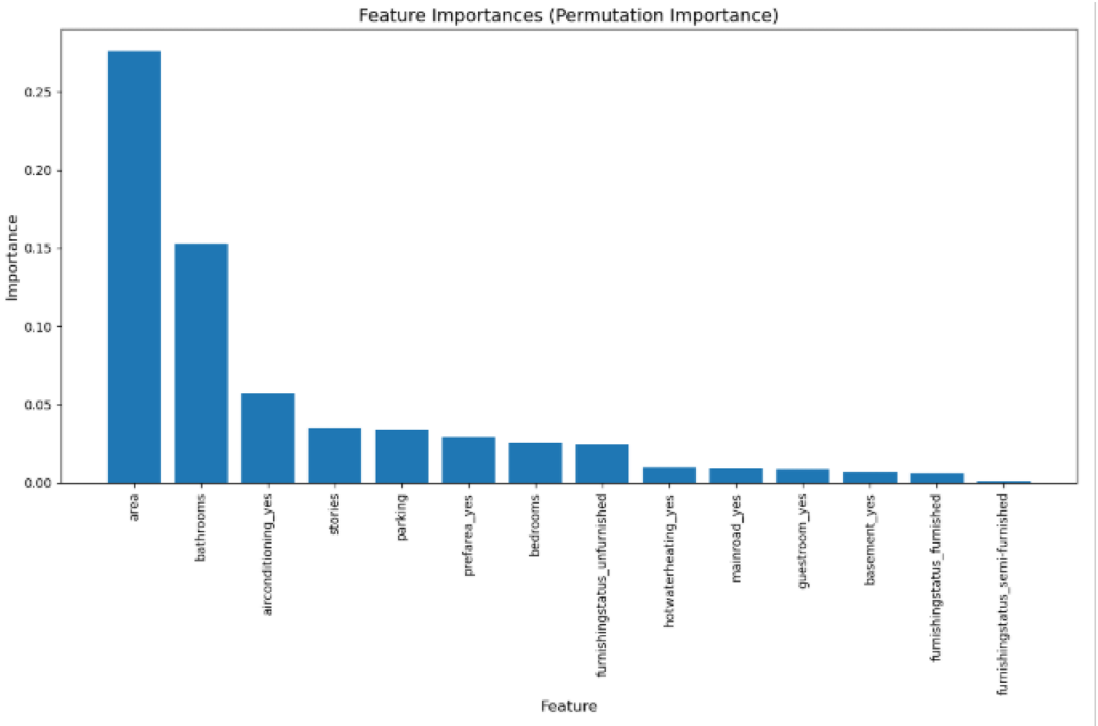
Similar to bathrooms



Methodology

EDA

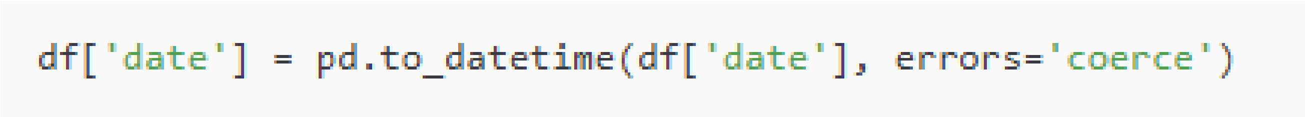




Methodology

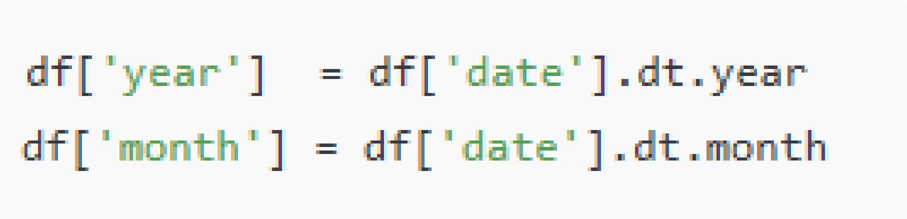
Preprocessing

- **Column selection & cleanup:** keep only price, date, postcode.
- **Date parsing:**



```
df['date'] = pd.to_datetime(df['date'], errors='coerce')
```

- **Drop bad rows:** any row missing date or price is removed.
- **Derive time fields:**

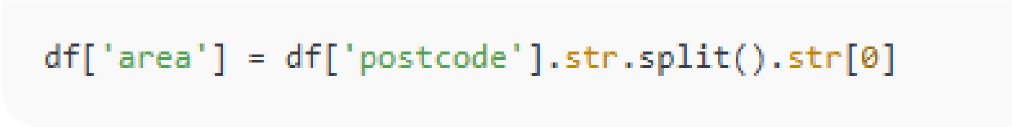


```
df['year']  = df['date'].dt.year
df['month'] = df['date'].dt.month
```

Pipeline Preprocessor:

- **Numeric features** (FEATURE_NUM) → StandardScaler()
- **Categorical feature** (area) → OneHotEncoder(handle_unknown='ignore')

Extract area code:



```
df['area'] = df['postcode'].str.split().str[0]
```

Methodology

Feature engineering

Step	**Code Snippet**	**Description**
1. Identify feature types	categorical_cols = X.columns[:3]	Select first 3 columns as categorical, others as numeric
2. One-Hot Encoding	("cat", OneHotEncoder(...), categorical_cols)	Convert categories to binary columns (0/1)
3. Standardization	("num", StandardScaler(), numeric_cols)	Scale numeric features to mean=0, std=1
4. Combine transformers	ColumnTransformer([...])	Apply different preprocessing to different column sets
5. Transform datasets	fit_transform(X_train) / transform(X_test)	Fit on train set, transform both train and test
6. **Macroeconomic characteristics**	merge(macros, on=['year','month'], how='left')	Incorporate macro data such as the GDP in the UK on a monthly basis
7. **Next period goals**	 monthly['up_next'] = (monthly['next_price'] > monthly['price']).astype(int)	Calculate the price for the next month and generate a binary rise and fall label

Model Design

Model plan

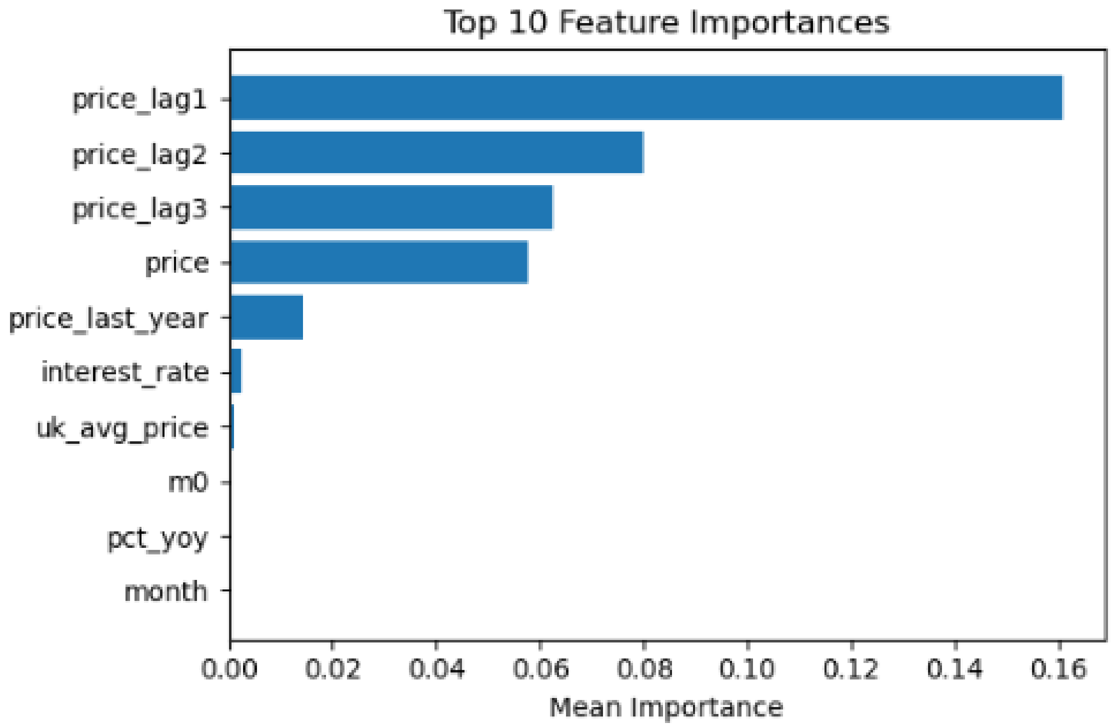
It is planned to select a five-month dataset for research, namely a dataset of three consecutive months, data of the same month but different years, and data of adjacent months of different years for study, in order to observe its periodicity.

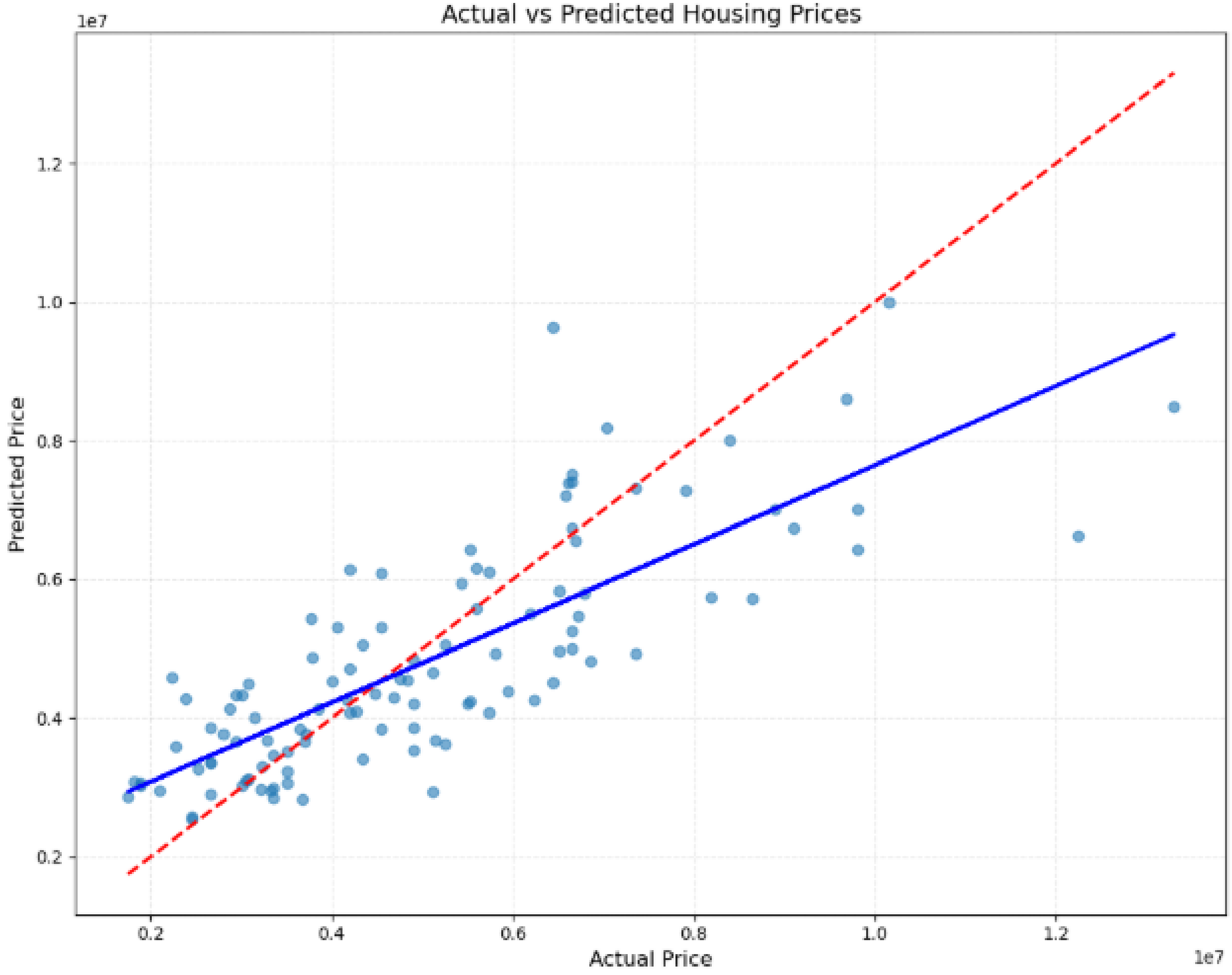
Model	Core Purpose	Data Requirements	Causal Inference Power	Typical Application Area
Random Forest	Ensemble prediction via aggregating multiple decorrelated decision trees	Handles mixed data types ,robust to noise and missing values	Limited	redit scoring, disease diagnosis, feature selection
Regression Model	Statistical analysis of variable relationships	Cross-sectional / time series data	Medium (requires strict assumptions)	Interdisciplinary/general-purpose analysis
Machine Learning Model	High-accuracy prediction	Big data, complex features	Weak	Business forecasting, image recognition

Model Design

Regression

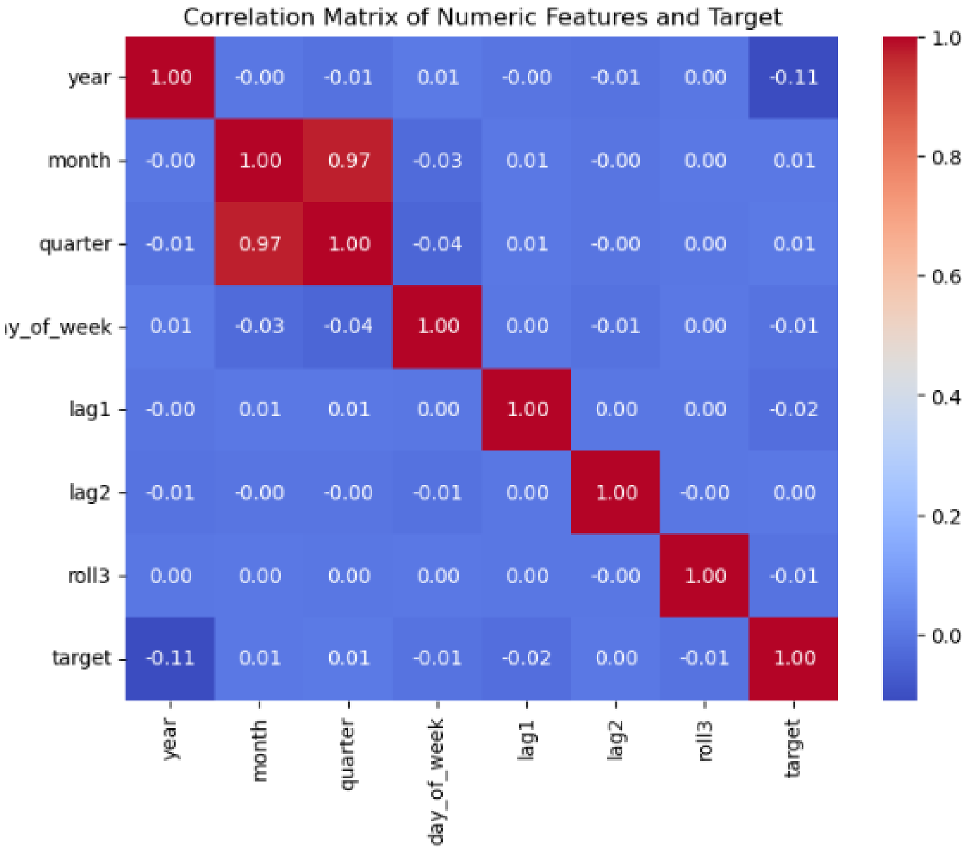
The one-time regression model is difficult to cope with the multivariate nonlinear relationship, and the value is only 0.56

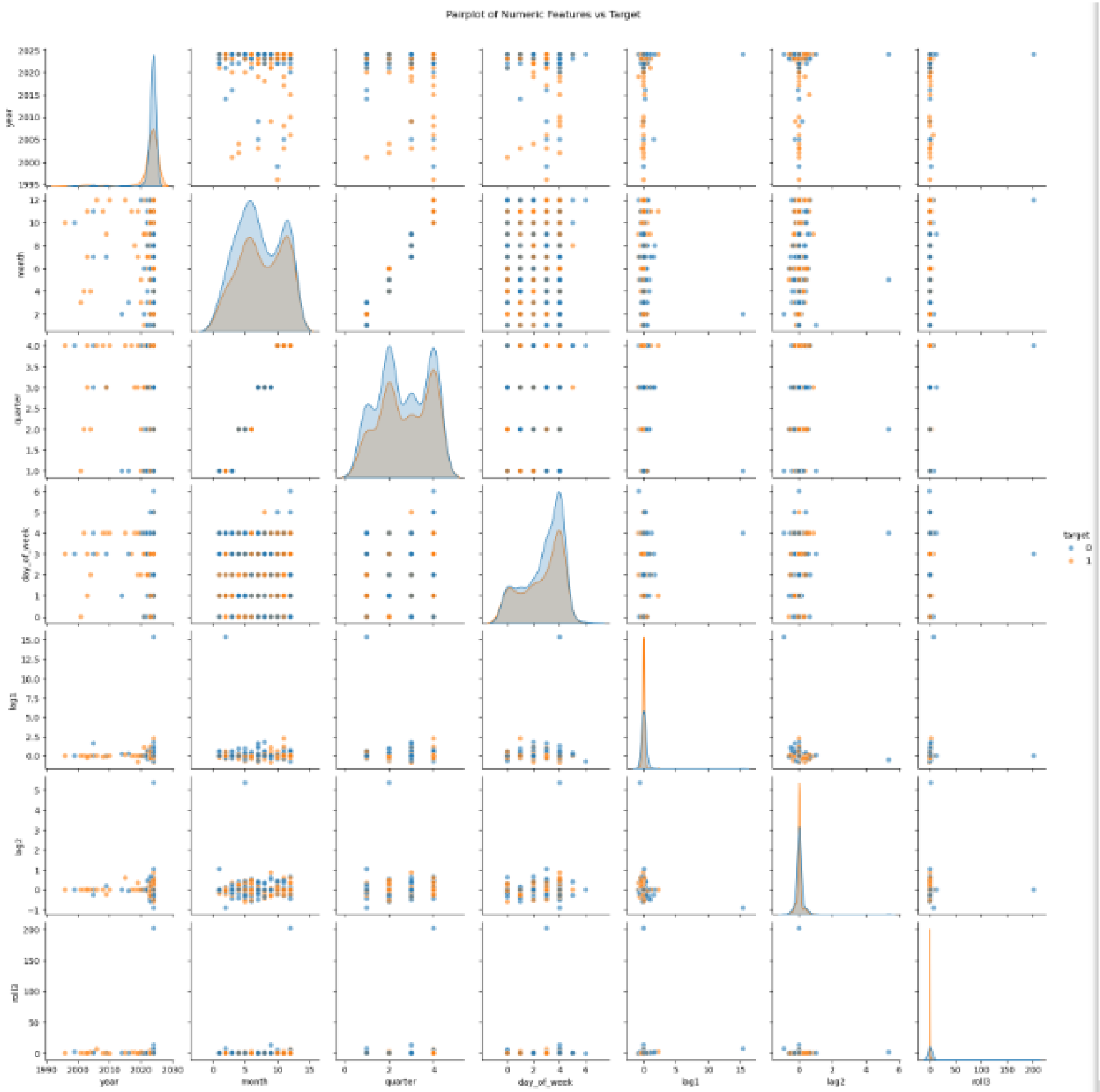




Model Design

Random forest





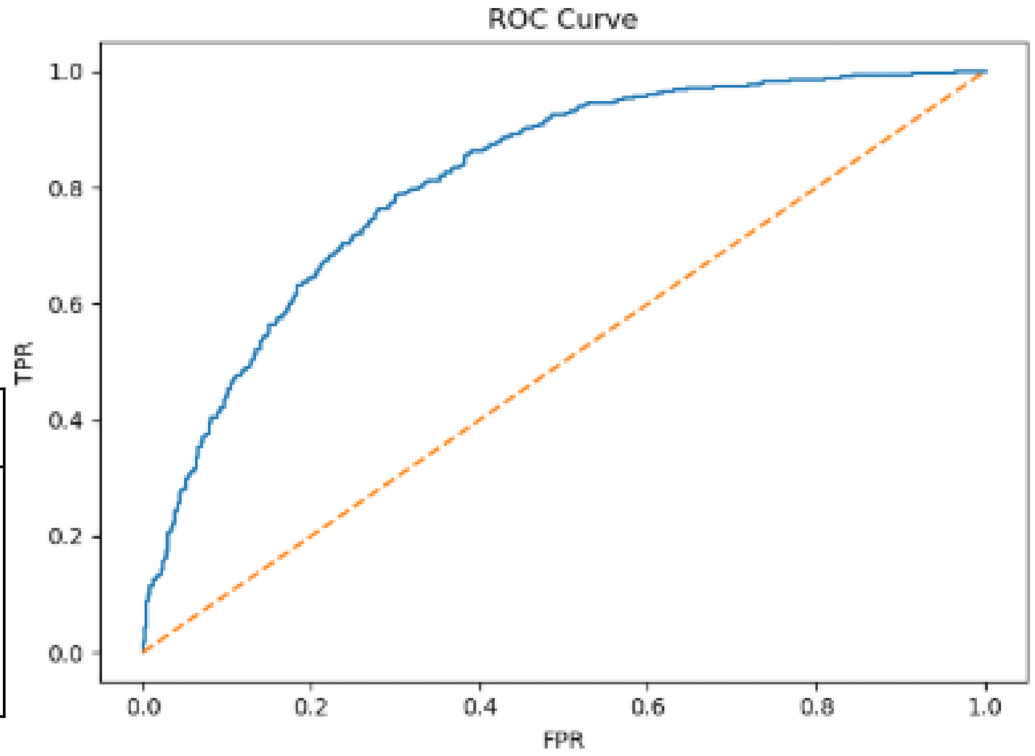
The analysis of the variable relationship is reasonable, but the values still need to be adjusted

Model Design

HistGradientBoostingClassifier

It is suitable for prediction, with the highest value, but the ideal effect has not been achieved

Model	accuracy	precision	recall	f1	auc
HistGradientBoostingClassifier	0.710702	0.677249	0.939794	0.787208	0.804984



Model improvement

Model improvement

Check dataset

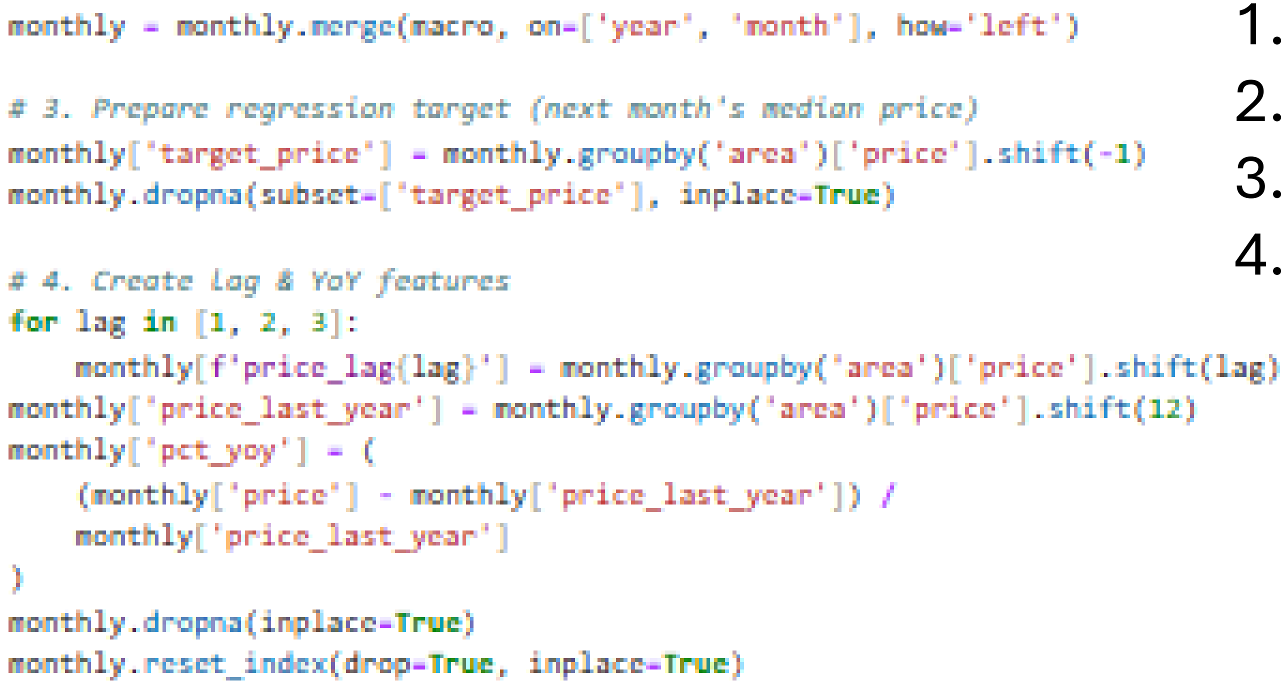
Month	F1-score
1月	0.716157
2月	0.771281
3月	0.500000
9月	0.677520
10月	0.723596
11月	0.679905
12月	0.671561

Region code	F1-score
SW1X	0.0
SW17	0.0
SW19	0.0
SW1V	0.0
EC1V	0.0
…	…
SM4	1.0
TN3	1.0
WF2	1.0
SY14	1.0
CW3	1.0

Through dataset inspection, the datasets of appropriate months were selected and the locations were filtered

Model improvement

Model adjustment



```
monthly = monthly.merge(macro, on=['year', 'month'], how='left')

# 3. Prepare regression target (next month's median price)
monthly['target_price'] = monthly.groupby('area')['price'].shift(-1)
monthly.dropna(subset=['target_price'], inplace=True)

# 4. Create lag & YoY features
for lag in [1, 2, 3]:
    monthly[f'price_lag{lag}'] = monthly.groupby('area')['price'].shift(lag)
monthly['price_last_year'] = monthly.groupby('area')['price'].shift(12)
monthly['pct_yoy'] = (
    (monthly['price'] - monthly['price_last_year']) /
    monthly['price_last_year']
)
monthly.dropna(inplace=True)
monthly.reset_index(drop=True, inplace=True)
```

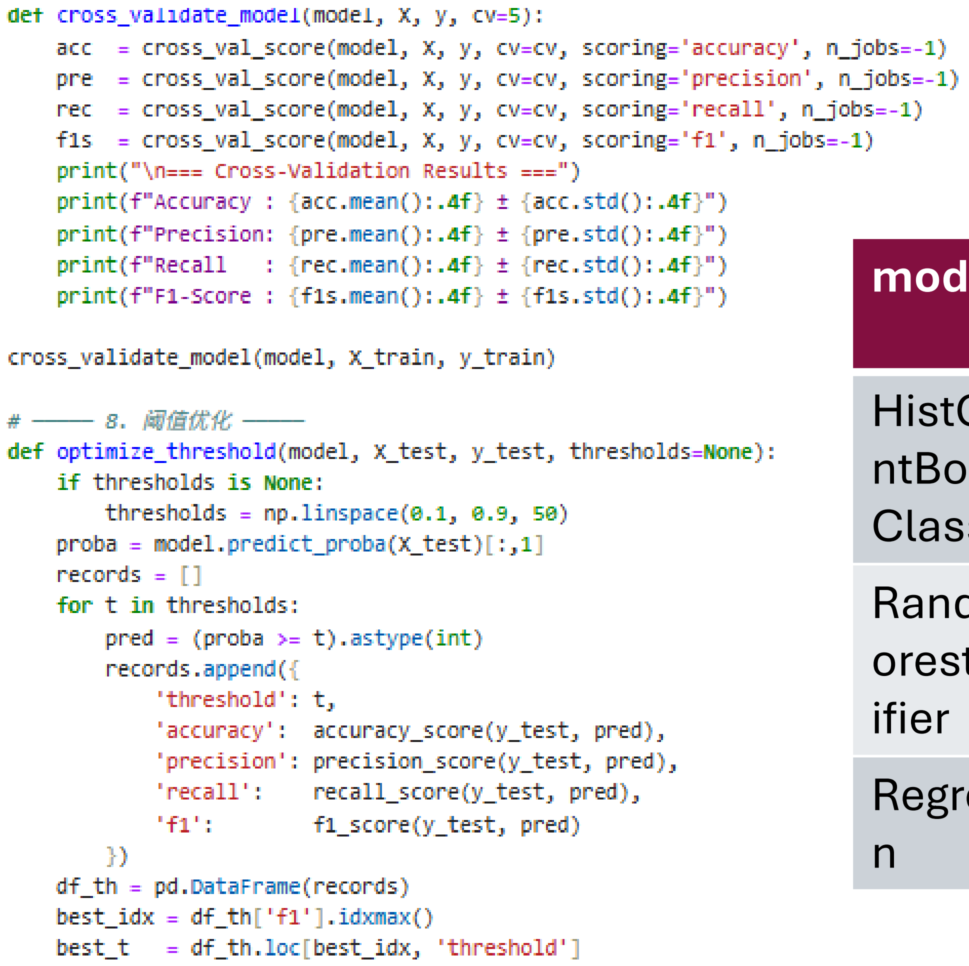
1. Increase macroeconomic factors
2. Adjust the feature engineering
3. Adjust the architecture of some models
4. Dual evaluation system

mean_squared_error(), mean_absolute_error(),r2_score()
accuracy_score(), precision_score(), roc_curve(

Model improvement

Parametric adjustment

Use parameter adjustment and cross-validation

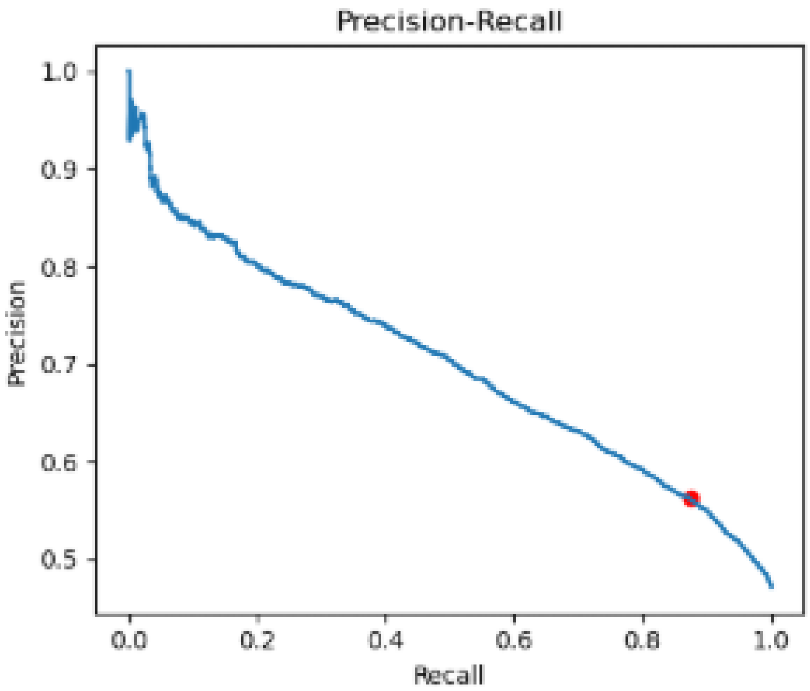


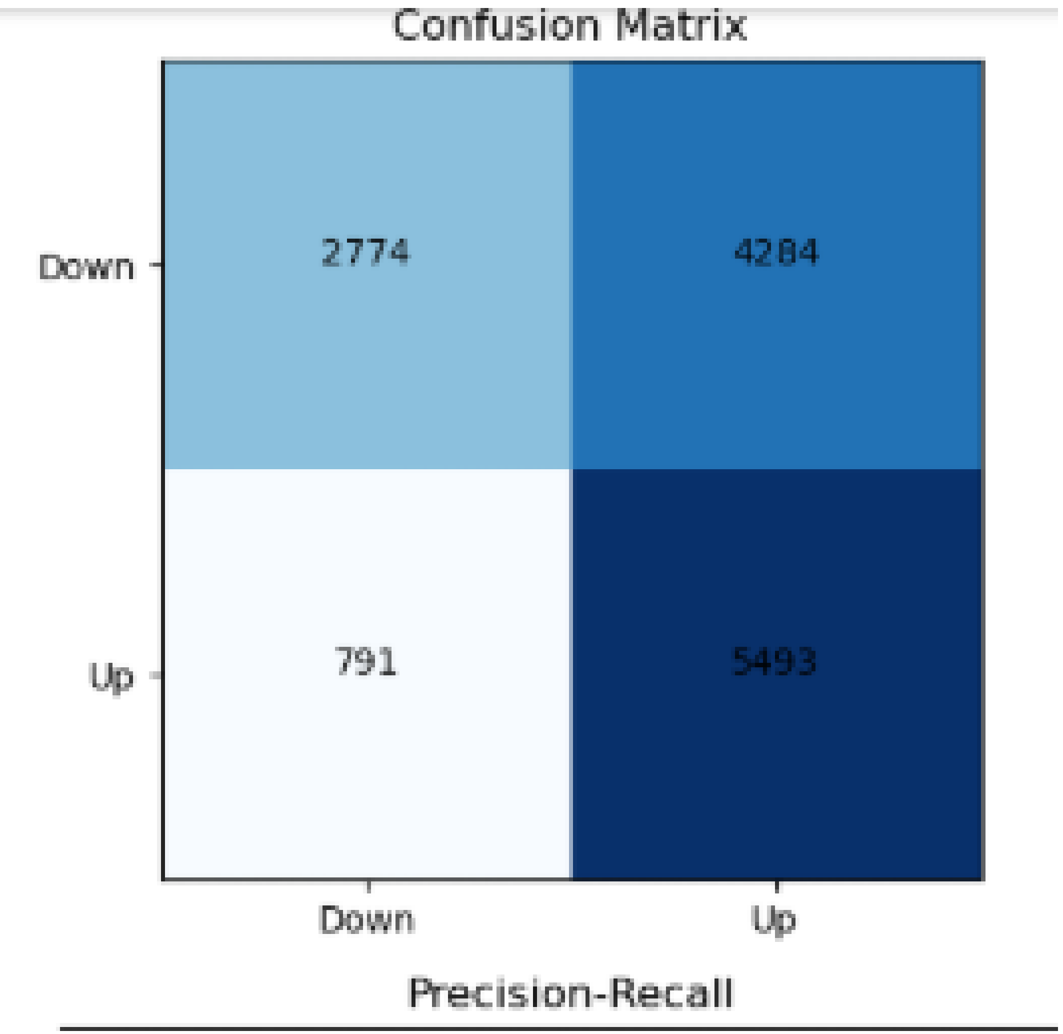
```
def cross_validate_model(model, X, y, cv=5):
    acc  = cross_val_score(model, X, y, cv=cv, scoring='accuracy', n_jobs=-1)
    pre  = cross_val_score(model, X, y, cv=cv, scoring='precision', n_jobs=-1)
    rec  = cross_val_score(model, X, y, cv=cv, scoring='recall', n_jobs=-1)
    f1s  = cross_val_score(model, X, y, cv=cv, scoring='f1', n_jobs=-1)
    print("\n=== Cross-Validation Results ===")
    print(f"Accuracy : {acc.mean():.4f} ± {acc.std():.4f}")
    print(f"Precision: {pre.mean():.4f} ± {pre.std():.4f}")
    print(f"Recall    : {rec.mean():.4f} ± {rec.std():.4f}")
    print(f"F1-Score : {f1s.mean():.4f} ± {f1s.std():.4f}")

cross_validate_model(model, X_train, y_train)

# ——— 8. 阈值优化 ———
def optimize_threshold(model, X_test, y_test, thresholds=None):
    if thresholds is None:
        thresholds = np.linspace(0.1, 0.9, 50)
    proba = model.predict_proba(X_test)[:,1]
    records = []
    for t in thresholds:
        pred = (proba >= t).astype(int)
        records.append({
            'threshold': t,
            'accuracy':  accuracy_score(y_test, pred),
            'precision': precision_score(y_test, pred),
            'recall':    recall_score(y_test, pred),
            'f1':        f1_score(y_test, pred)
        })
    df_th = pd.DataFrame(records)
    best_idx = df_th['f1'].idxmax()
    best_t   = df_th.loc[best_idx, 'threshold']
```

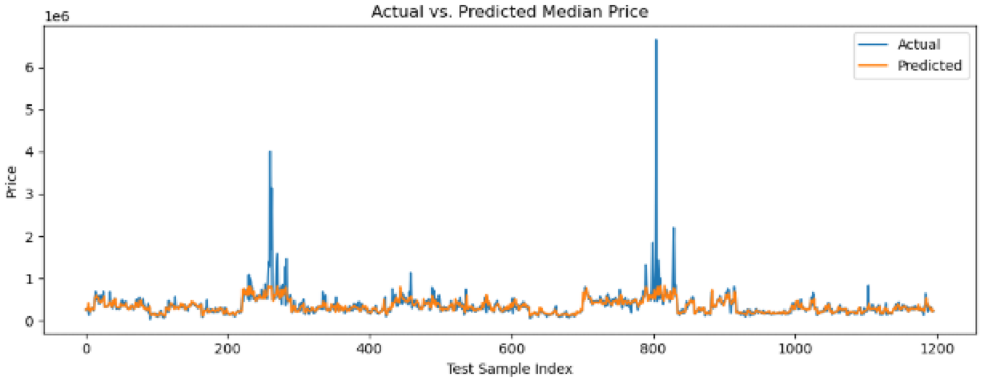
model	opt_threshold	accuracy	precision	recall	F1-score
HistGradientBoostingClassifier	0.418	0.746656	0.715262	0.922173	0.805645
RandomForestClassifier	0.520	0.739130	0.724180	0.875184	0.792553
Regression	3593.677	0.748328	0.731144	0.882526	0.799734

Model improvement

Achievement Display



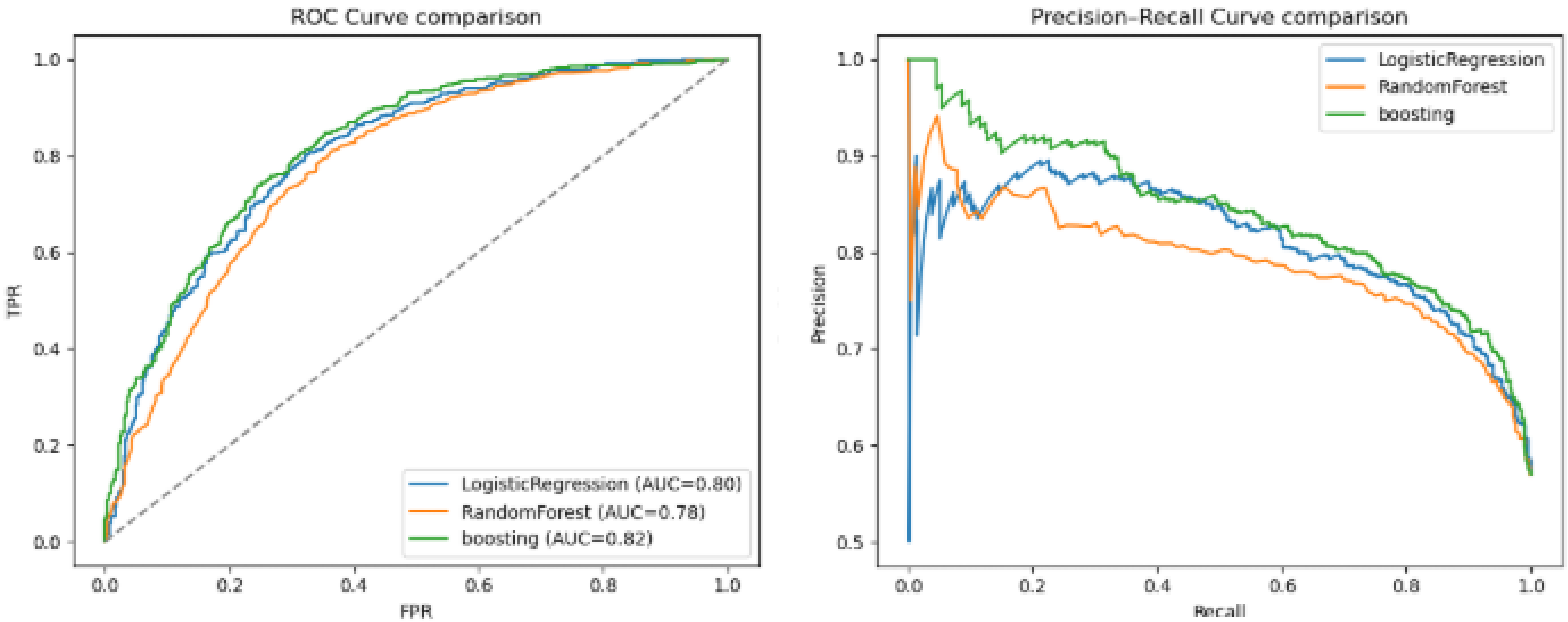




Model improvement

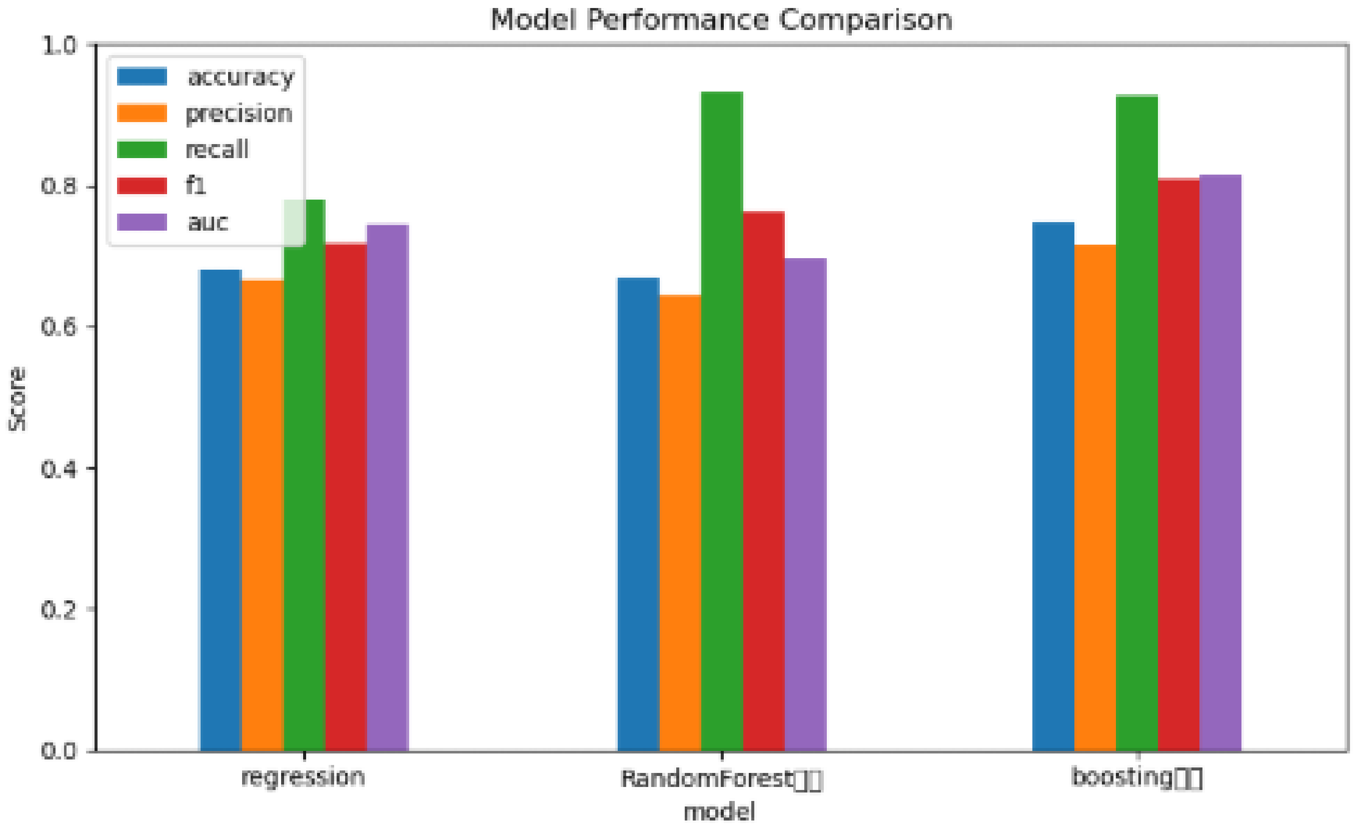
Parametric adjustment

After adjusting the parameters



Model improvement

Model Comparison



After comparison, found HistGradientBoostingClassifier value slightly higher than the other models of the model

Model Selection Guide:

Regression models are not good at handling nonlinear relationships.

The random forest model has a relatively high recall value, and its overall value is not inferior to that of machine learning models. However, the model takes too long to run, making it the preferred choice for offline batch analysis.

HistGradientBoostingClassifier is dealing with large data sets the pursuit of speed and efficiency of one of the powerful model.

Model improvement

matters need attention

1.Fix the information leakage
Calculate the percentile threshold using y_train Ensure that the test set is completely independent

2. Improve the threshold selection
Add the median as the reference point (50% percentile)
Use NaN instead of 0 to handle the division by zero problem to avoid false high values

3. Enhance the analysis function
Add positive and negative sample ratio analysis
Create more professional biaxial visualization

Model improvement

Future Work

1. Add more dataset, including data from different regions and different eras

2. Add more macro factors

3. Test and compare more models