

计量经济学期末论文

地区内房地产行业发展影响因素分析 ——以山东省为例

18数据科学 赵呈亮

201800820179

一、引言

众所周知，房地产行业已经成为我国经济发展的关键，房地产行业更是关系到万千普通人的生活，因此，房地产行业的发展需要系统性的研究，本文首先确定研究所需数据，之后进行数据收集。在完成数据收集与分析后，进行建模，并对结果进行描述性统计检验包括：多重共线性检验、异方差性检验、自相关性检验，最终得到结论。

二、数据与理论分析

由于所面临的问题较为宽泛，需要将所需的问题进行数学建模。首先，定义一指标作为地区内房地产发展。这在之后将做为预测变量。而为了研究更有针对性，所以选取山东省做为分析的样本。如果以全国范围内进行分析，可能会因为地区差异带来许多隐藏的未知变量，从而造成误差。在建模时，我们采用多元线性回归模型进行回归分析，进而分析相关系数，判断各个变量对最终结果有无影响。同时也要排除变量间相互影响，最后，进行统计检验，最终得到结论。

数据及符号 符号

数据内容

 Y_t

山东省七月份住宅商品房销售额累计值

 X_{1t}

山东省七月份房地产业投资累计值

 X_{2t}

山东省七月份地方债务余额

 X_{3t}

山东省七月份固定投资完成额

 X_{4t}

山东省七月份商品零售价格指数增长率

数据来源为国家统计局，注意的是， Y_t 与 X_{1t} 原始数据为累计值，所以当我们需要做一阶差分。又 ΔY_t 为我们的预测变量，为每月山东省内住宅商品房销售额， ΔX_{1t} 为山东省各月份房地产业投资额，而 X_2 为地方债，由于房地产开发成本中很大一部分为土地出让金，而这一部分为地方政府收入来源 ~~共~~^{重要} 部分，所以地方债务是促使地方政府发展房地产业“卖地”的动机，所以引入这一变量，研究其对房地产业发展的影响。同理，引入 X_{3t} 固定投资额代表经济发展程度，商品价格指数表示经济发展情况。纵上数据自 2019 年 1 月至 2021 年 11 月。

三、数学模型

$$\Delta Y_t = \beta_0 + \beta_1 \Delta X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + u_t$$

首先，对模型及数据进行 OLS 分析，统计检验结果如下：

拟合优度检验： $R^2 = 0.642$ ，拟合优度尚可。F 检验：在给定显著水平 $\alpha = 0.05$ 下，F 检验的 P 值为 $0.003 < 0.025$ ， $<$ 表示 F 检验

十. 检验. 在给定显著水平 $\alpha = 0.05$ 下, 根据 t 分布表可知 $\frac{ta}{2} = 2.052$, 而
样本中 t 值均大于分位值: ~~有相关性~~ 可检验相关性
1. 多重共线性检验

$$\text{通过计算完善膨胀因子 } VIF_i = \frac{1}{1-R_i^2}$$

得到结果为

X_1	1.47
X_2	1.25
X_3	1.37
X_4	1.45

现各个变量的 VIF 并未 > 5 , 所以可以判定不存在多重共线性.
实际上.

2. 异方差性检验.

对原数据的多元线性回归结果进行 heteroscedasticity 检验. 结果中 white 检验值
为 19.98 P-value 为 0.13 似以量可知, 存在异方差性, 为了修正异方差性,
我们对其进行最小加权二乘. (权重 w_i 为 $\frac{1}{X_i}$) 修正后,
模型的异方差性再对其进行 heteroscedasticity 检验. 模型的异方差性
有显著改善.

3. 自相关性检验.

对原数据进行 DW 自相关性检验. 其 DW 值为 1.997 其 D.L. 为
0.0278, 0.08. 故存在自相关性. 为了消除这一影响.
我们进行了 1 阶差分法对解释变量进行操作, 之后
重新进行多元线性回归.

最后 我们得到. 原方程.

$$Y = -0.0842 X_1 + 0.0103 X_2 + -4.3729 X_3 \\ + -17.052 X_4 - 92.829$$

四、结论

房地产发展以每月的住宅销售量为指标，可以发现其与地方债务的余额呈现正相关。这解释不了地方政府为了增加收入，其次，房地产销售量与商品价格指数成负相关，可以解释为当物价上涨时，人民将有限的生活资金用于生活，所以可支配收入减少，从而房地产住宅的潜在客户减少，销售量下降。

五、进一步工作

1. 扩大数据量，使统计结果更具有代表性
2. 建立更复杂的模型。

```

import scipy.stats as stats
import numpy as np
import statsmodels.api as sm
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn import model_selection

from patsy import dmatrices

```

```
data=pd.read_csv("final.csv")
```

```
from statsmodels.formula.api import ols
```

```

data=data[ [ "Y" , "X1" , "X2" , "X3" , "X5" ] ]
data[ 'X5' ]= -1*data[ 'X5' ]

```

```

fit = ols('Y~ X1 + X2 + X3 + X5', data=data).fit()

fit.summary()

```

OLS Regression Results

Dep. Variable:	Y	R-squared:	0.632
Model:	OLS	Adj. R-squared:	0.534
Method:	Least Squares	F-statistic:	6.440
Date:	Tue, 28 Dec 2021	Prob (F-statistic):	0.00318
Time:	01:36:36	Log-Likelihood:	-83.228
No. Observations:	20	AIC:	176.5
Df Residuals:	15	BIC:	181.4
Df Model:	4		
Covariance Type:	nonrobust		

	coef	std err	t	P> t 	[0.025	0.975]
Intercept	-92.8290	46.135	-2.012	0.063	-191.163	5.505
X1	-0.0842	0.039	-2.150	0.048	-0.168	-0.001
X2	0.0103	0.003	3.652	0.002	0.004	0.016
X3	-4.3729	2.015	-2.170	0.046	-8.668	-0.078
X5	-17.0542	6.009	-2.838	0.012	-29.861	-4.247

Omnibus:	4.789	Durbin-Watson:	1.544
Prob(Omnibus):	0.091	Jarque-Bera (JB):	2.975
Skew:	0.924	Prob(JB):	0.226
Kurtosis:	3.392	Cond. No.	1.73e+05

Warnings:

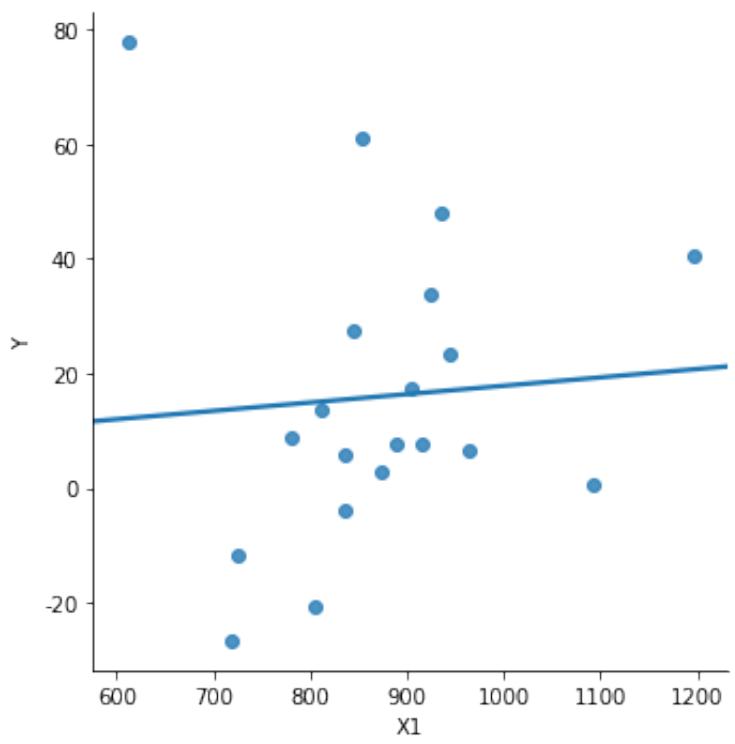
- [1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
- [2] The condition number is large, 1.73e+05. This might indicate that there are strong multicollinearity or other numerical problems.

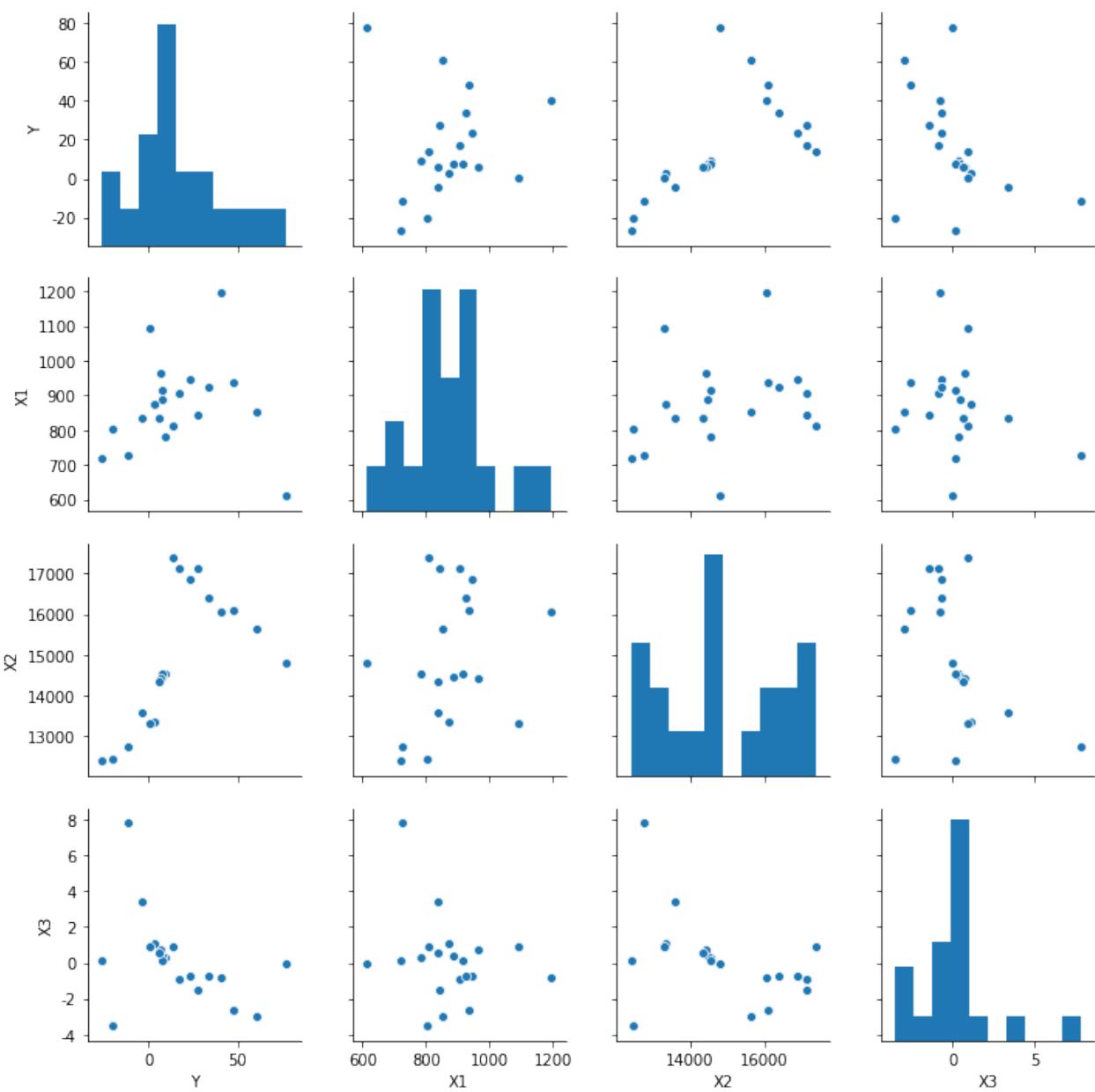
```

sns.lmplot(x='X1', y='Y', data=data, ci=None)
plt.rcParams['font.sans-serif'] = ['SimHei']
plt.rcParams['axes.unicode_minus'] = False
plt.show()
sns.pairplot(data.loc[:, ['Y', 'X1', 'X2', 'X3']])
# 显示图形
plt.show()

```

```
findfont: Font family ['sans-serif'] not found. Falling back to DejaVu Sans.
```





```

from statsmodels.stats.outliers_influence import variance_inflation_factor
def check_vif(df):
    df = df.drop(['Y'], axis=1) # 删除多余变量, 包括因变量
    df = df.dropna(axis=0, how='any') # 删除带有任何空值的行
    df['const'] = 1 # 添加常数项, 这个是重点
    x = np.array(df)
    vif_list = [variance_inflation_factor(x, i) for i in range(x.shape[1])]
    df_vif = pd.DataFrame({'variable': list(df.columns), 'vif': vif_list})
    df_vif = df_vif[df_vif['variable'] != 'const'] # 删除常数项
    print(df_vif)

```

```
check_vif(data)
```

```
variable      vif
0          X1  1.479286
1          X2  1.254423
2          X3  1.373096
3          X5  1.451434
```

```
sm.stats.diagnostic.het_white(fit.resid, exog = fit.model.exog)
```

```
(19.83662761966763,
 0.1353769006048792,
 43.36418340575183,
 0.0002903002498445005)
```

```
fix = sm.WLS(data.Y, data.loc[:, ['X1', 'X2', 'X3', 'X5']], weights=1. /
              (data.X1 ** 2)).fit()
print(fix.summary())
```

```
WLS Regression Results
=====
=====
Dep. Variable: Y R-squared (uncentered):
0.846
Model: WLS Adj. R-squared (uncentered):
0.799
Method: Least Squares F-statistic:
17.86
Date: Wed, 29 Dec 2021 Prob (F-statistic):
3.40e-05
Time: 01:53:06 Log-Likelihood:
-94.248
No. Observations: 17 AIC:
196.5
Df Residuals: 13 BIC:
199.8
Df Model: 4
Covariance Type: nonrobust
=====
coef std err t P>|t| [ 0.025 0.975]
=====
```

```

-----
X1          -0.0226    0.112     -0.201     0.844     -0.265     0.220
X2           0.0044    0.003      1.699     0.113     -0.001     0.010
X3           1.6421    1.063      1.545     0.146     -0.655     3.939
X5          -18.7238   2.301     -8.139     0.000     -23.694    -13.754
=====
Omnibus:                  7.761   Durbin-Watson:            1.994
Prob(Omnibus):             0.021   Jarque-Bera (JB):        5.288
Skew:                      -0.775   Prob(JB):                 0.0711
Kurtosis:                   5.251   Cond. No.                1.15e+03
=====
```

Warnings:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[2] The condition number is large, 1.15e+03. This might indicate that there are strong multicollinearity or other numerical problems.

#变量本身带有趋势

我们通常研究的时间序列自身就是带有一定趋势的本身就存在一定关系（此时可试试AR模型），此时就会出现异方差问题。

变量的之后效应。通常，一个变量对另一个变量的影响并不会在当期表现出来，往往需要之后一期或几期才会表现出来，如果忽视了这些之后效应，模型也可能出现自相关问题。

```

data = data.diff()
data = data.dropna()
x = np.array(data[['X1', 'X2', "X3", 'X5']])
y = np.array(data[['Y']])
x = sm.add_constant(x)
model = sm.OLS(y,x)
fit = model.fit()
print(fit.summary())
```

OLS Regression Results

```

=====
Dep. Variable:                  y      R-squared:           0.634
Model:                          OLS    Adj. R-squared:       0.502
Method:                         Least Squares   F-statistic:         4.773
Date:                          Wed, 29 Dec 2021   Prob (F-statistic):  0.0177
Time:                          01:53:55        Log-Likelihood:     -85.987
```

```

No. Observations: 16 AIC: 182.0
Df Residuals: 11 BIC: 185.8
Df Model: 4
Covariance Type: nonrobust
=====
            coef    std err          t      P>|t|      [0.025      0.975]
-----
const     -3.8818   16.010     -0.242     0.813    -39.120    31.357
x1        -0.0582   0.022     -2.659     0.022    -0.106    -0.010
x2        -0.0109   0.011     -0.957     0.359    -0.036     0.014
x3        -0.1630   1.493     -0.109     0.915    -3.448     3.122
x4       -29.9964   8.555     -3.506     0.005    -48.827   -11.166
=====
Omnibus: 4.679 Durbin-Watson: 2.642
Prob(Omnibus): 0.096 Jarque-Bera (JB): 2.470
Skew: 0.929 Prob(JB): 0.291
Kurtosis: 3.504 Cond. No. 1.77e+03
=====
```

Warnings:

```
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.

[2] The condition number is large, 1.77e+03. This might indicate that there
are
strong multicollinearity or other numerical problems.
```

```
/Users/zcl271828/opt/anaconda3/lib/python3.7/site-
packages/scipy/stats/stats.py:1450: UserWarning: kurtosistest only valid for
n>=20 ... continuing anyway, n=16
"anyway, n=%i" % int(n))
```