数学建模第四次作业

问题分析

作业——投资额预测

北京郵電大學 Beijing University of Posts and Telecommunications

- 建立投资额模型，研究某地区实际投资额与国民生产总值（GNP）及物价指数（PI）的关系并根据对未来GNP及PI的估计，预测未来投资额

该地区连续20年的统计数据

年份序号	投资额	国民生产总值	物价指数	年份序号	投资额	国民生产总值	物价指数
1	90.9	596.7	0.7167	11	229.8	1326.4	1.0575
2	97.4	637.7	0.7277	12	228.7	1434.2	1.1508
3	113.5	691.1	0.7436	13	206.1	1549.2	1.2579
4	125.7	756.0	0.7676	14	257.9	1718.0	1.3234
5	122.8	799.0	0.7906	15	324.1	1918.3	1.4005
6	133.3	873.4	0.8254	16	386.6	2163.9	1.5042
7	149.3	944.0	0.8679	17	423.0	2417.8	1.6342
8	144.2	992.7	0.9145	18	401.9	2631.7	1.7842
9	166.4	1077.6	0.9601	19	474.9	2954.7	1.9514
10	195.0	1185.9	1.0000	20	424.5	3073.0	2.0688

题目分析

1. **目标**：建立一个回归模型，通过分析历史数据中的投资额、国民生产总值（GNP）和物价指数（PI），找到它们之间的关系。
2. **预测要求**：根据历史数据和模型关系，对未来的GNP和PI估计后，预测未来的投资额。

解题思路

1. **构建基础模型**：
   - 使用GNP和PI作为自变量，投资额作为因变量，建立多元回归模型。
   - 检验模型的拟合效果（如R^2）以及各系数的显著性，初步分析投资额与GNP、PI之间的线性关系。

2. **考虑时间序列特性**：
    - 由于这是时间序列数据，可能存在自相关性，需要对模型进行自相关性检验。
    - 如果发现自相关性问题，可以尝试引入滞后项，构建自回归模型（AR模型），以消除自相关性，提高模型的预测准确性。
3. **模型改进--也就是建立自回归模型**：
    - 在基础模型的基础上，引入**一阶自回归项**，改进模型，使其能够更好地解释投资额的变化趋势。
4. **模型检验与对比--下面我会给出可视化对照图**：
    - 对改进后的模型进行残差分析，比较改进前后的模型效果。
    - 使用检验结果（如DW统计量）判断改进后的模型是否消除了自相关性
5. **预测预测未来的投资额**：利用**基本模型和改进的自回归模型**得出预测结果

建立基本回归模型

基本回归模型分析：

分析模型结构

基本回归模型的形式为：

$$y_t = \beta_0 + \beta_1 x_{1t} + \beta_2 x_{2t} + \epsilon_t$$

其中：

- y_t 表示投资额
- x_{1t} 表示国民生产总值（GNP）
- x_{2t} 表示物价指数（PI）
- $beta_0$ $beta_1$ $beta_2$ 分别是模型的回归系数

参数估计结果

模型的估计结果如下：

- 常数项 $beta_0$ =**322.7250**
- GNP的回归系数 $beta_1$ = **0.6185**
- 物价指数的回归系数 $beta_2$ = **-859.4790**

参数的置信区间如下：

- $beta_0$ 的置信区间为 [224.3386, 421.1114]

- $beta_1$ 的置信区间为 [0.4773, 0.7596]
- $beta_2$ 的置信区间为 [-1121.4757, -597.4823]

思考一下模型拟合度

- R方 = **0.9908**：说明模型拟合优度较高，GNP和物价指数能够解释投资额的99.08%的变异性。
- F统计量 F = **919.8529**，表明模型整体显著。

让我们评估一下模型优缺点

- **模型优点**：
  - 拟合效果好，R方较高，模型能够较好地解释投资额的变动。
  - 各回归系数显著，说明GNP和物价指数在统计上对投资额有显著影响。

- **模型缺点**：
  - **滞后性影响**：没有考虑时间序列数据的滞后性，时间序列数据通常存在滞后效应，忽视这一点可能会导致模型不准确。
  - **自相关性**：模型未检验随机误差项的自相关性，时间序列数据往往存在自相关现象，如果误差项存在自相关性，模型可能会产生不良后果。

因此我们需要改进建议

可以通过构建一阶自回归模型（AR(1)模型），引入滞后项，消除自相关性，进一步提升模型的预测能力和稳定性。

建立自回归模型

一阶自回归模型（AR(1)）分析：

原模型自相关性检验

- Durbin-Watson统计量 $DW_{old} = 0.8754$
- 在样本容量 n = 20、回归变量数量 k = 3、显著性水平 $alpha = 0.05$ 下查表得出临界值： d = 1.10， dU = 1.54。
- 因为 $DW_{old} < d_L$，说明原模型存在正自相关性。
- 计算自相关系数 -- 0.5623。

新模型的构建

为了消除自相关性，进行了如下转换：

1. 投资额的转换：$y_t^* = y_t - 0.5623y_{t-1}$
2. 自变量的转换：$x_{it}^* = x_{it} - 0.5623x_{i,t-1}$，其中 (i = 1, 2)。

新模型的形式为：

$$y_t^* = \beta_0^* + \beta_1 x_{1t}^* + \beta_2 x_{2t}^* + u_t$$

通过该模型重新估计参数，得到：

- $beta_0^* = 163.4905$
- $beta_1 = 0.6990$
- $beta_2 = -1009.0333$
- 新模型的 R方= **0.9772** ，，残差标准差 = **9.8277** ，相较于原模型的残差标准差 (**12.7164**) 有所下降。

新模型的表达式

回归方程为：

$$haty_t = 163.4905 + 0.699x_{1t} - 1009.033x_{2t} + \epsilon_t$$

还原为原始变量的表达式：

$$haty_t = 163.4905 + 0.5623y_{t-1} + 0.699x_{1t} - 0.3930x_{1,t-1} - 1009.033x_{2t} + 567.3794x_{2,t-1}$$

py/matlab展示PI和GNP关系和可视化对比

代码展示：

```
import numpy as np
import pandas as pd
import statsmodels.api as sm
from statsmodels.stats.stattools import durbin_watson
import matplotlib.pyplot as plt

# 数据输入
data = pd.DataFrame({
    '年份': np.arange(1, 21),
    '投资额': [90.9, 97.4, 113.5, 125.7, 122.8, 133.3, 149.3, 144.2, 166.4,
195.0,
```

```
            229.8, 228.7, 206.1, 257.9, 324.1, 386.6, 423.0, 401.9, 474.9,
424.5],
    'GNP': [596.7, 637.7, 691.1, 756.0, 799.0, 873.4, 944.0, 992.7, 1077.6,
1185.9,
            1326.4, 1434.2, 1549.2, 1718.0, 1918.3, 2163.9, 2417.8, 2631.7,
2954.7, 3073.0],
    '物价指数': [0.7167, 0.7277, 0.7436, 0.7676, 0.7906, 0.8254, 0.8679,
0.9145, 0.9601, 1.0000,
             1.0575, 1.1508, 1.2579, 1.3234, 1.4005, 1.5042, 1.6342, 1.7842,
1.9514, 2.0688]
})

# 基础模型
X_basic = data[['GNP', '物价指数']]
X_basic = sm.add_constant(X_basic)
y = data['投资额']
model_basic = sm.OLS(y, X_basic).fit()

# Durbin-Watson检验基础模型的自相关性
dw_basic = durbin_watson(model_basic.resid)

# 创建一阶自回归模型
data['投资额滞后'] = data['投资额'].shift(1)
data['GNP滞后'] = data['GNP'].shift(1)
data['物价指数滞后'] = data['物价指数'].shift(1)
data.dropna(inplace=True)  # 删除缺失值

# 转换后的新模型
X_new = data[['GNP', '物价指数', '投资额滞后', 'GNP滞后', '物价指数滞后']]
X_new = sm.add_constant(X_new)
y_new = data['投资额']
model_new = sm.OLS(y_new, X_new).fit()

# Durbin-Watson检验新模型的自相关性
dw_new = durbin_watson(model_new.resid)

# 打印模型结果
print("基础回归模型结果：")
print(model_basic.summary())
print(f"Durbin-Watson统计量（基础模型）：{dw_basic}\n")

print("一阶自回归模型结果：")
print(model_new.summary())
print(f"Durbin-Watson统计量（一阶自回归模型）：{dw_new}\n")

# 绘制残差对比图
plt.rcParams['font.sans-serif'] = ['SimHei']  # 黑体
```

```python
plt.rcParams['axes.unicode_minus'] = False    # 解决负号显示问题
plt.figure(figsize=(12, 5))
plt.subplot(1, 2, 1)
plt.plot(model_basic.resid, label="基础模型残差--2023211075", marker='o')
plt.legend()
plt.title("基础模型残差图")

plt.subplot(1, 2, 2)
plt.plot(model_new.resid, label="一阶自回归模型残差--by魏生辉", marker='o',
color='orange')
plt.legend()
plt.title("一阶自回归模型残差图")

plt.tight_layout()
plt.show()

# 绘制拟合对比图
plt.figure(figsize=(10, 5))
plt.plot(data['年份'], y_new, label="实际值", marker='o')
plt.plot(data['年份'], model_basic.predict(X_basic), label="基础模型预测值",
linestyle='--')
plt.plot(data['年份'], model_new.predict(X_new), label="一阶自回归模型预测值",
linestyle='--', color='orange')
plt.legend()
plt.title("拟合值对比")
plt.show(block=True)
```

分析PI和GNP的关系

①**GNP 的增长**往往对 PI 产生正向影响，但这种影响可能存在滞后性。

②**PI 的上升**会增加生产成本，抑制投资，进而可能对未来的 GNP 产生负向影响。

③**模型结果**表明，在预测投资额时，GNP 和 PI 是两个关键因素，分别在推动和抑制投资方面起到重要作用。

下面是代码分析结果

```
基础回归模型结果:
                            OLS Regression Results
==============================================================================
Dep. Variable:                   投资额   R-squared:                       0.991
Model:                            OLS   Adj. R-squared:                  0.990
Method:                 Least Squares   F-statistic:                     919.9
Date:                Mon, 11 Nov 2024   Prob (F-statistic):           4.73e-18
Time:                        23:51:44   Log-Likelihood:                -77.611
No. Observations:                  20   AIC:                             161.2
Df Residuals:                      17   BIC:                             164.2
Df Model:                           2
```

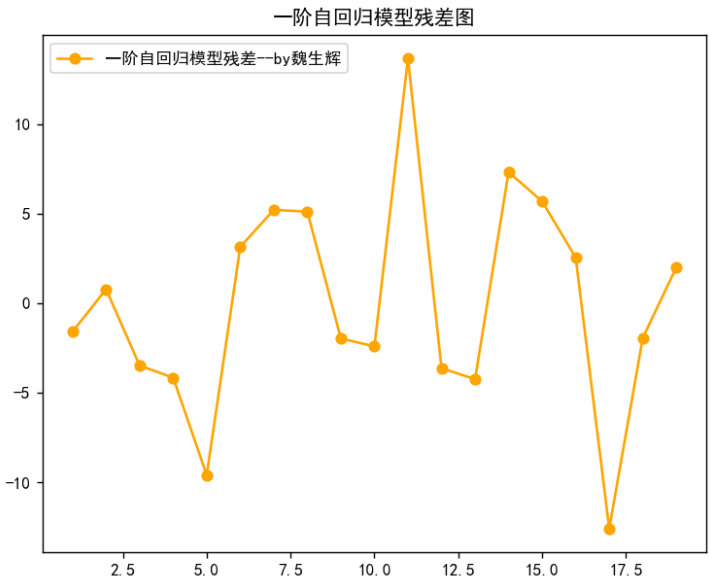
```
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const        322.7250     46.633      6.921      0.000     224.339     421.111
GNP            0.6185      0.067      9.242      0.000       0.477       0.760
物价指数        -859.4790    124.180     -6.921      0.000   -1121.476    -597.482
==============================================================================
Omnibus:                        1.151   Durbin-Watson:                   0.802
Prob(Omnibus):                  0.563   Jarque-Bera (JB):                0.867
Skew:                          -0.192   Prob(JB):                        0.648
Kurtosis:                       2.056   Cond. No.                     7.81e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 7.81e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
Durbin-Watson统计量（基础模型）: 0.8024793257864277

一阶自回归模型结果:
                            OLS Regression Results
==============================================================================
Dep. Variable:                    投资额   R-squared:                       0.998
Model:                            OLS   Adj. R-squared:                  0.997
Method:                 Least Squares   F-statistic:                     1074.
Date:                Mon, 11 Nov 2024   Prob (F-statistic):           1.58e-16
Time:                        23:51:44   Log-Likelihood:                -60.845
No. Observations:                  19   AIC:                             133.7
Df Residuals:                      13   BIC:                             139.4
D:\code\python\venv\Lib\site-packages\scipy\stats\_axis_nan_policy.py:418: UserWarning: `kur
  return hypotest_fun_in(*args, **kwds)
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         89.7191     57.856      1.551      0.145     -35.271     214.709
GNP            0.6981      0.044     15.901      0.000       0.603       0.793
物价指数        -657.2761    128.865     -5.100      0.000    -935.673    -378.879
投资额滞后          0.4351      0.164      2.658      0.020       0.082       0.789
GNP滞后        -0.5015      0.100     -4.997      0.000      -0.718      -0.285
物价指数滞后       421.5188    156.435      2.695      0.018      83.562     759.476
==============================================================================
Omnibus:                        0.633   Durbin-Watson:                   1.978
Prob(Omnibus):                  0.729   Jarque-Bera (JB):                0.035
Skew:                           0.080   Prob(JB):                        0.983
Kurtosis:                       3.136   Cond. No.                     2.46e+05
==============================================================================
```

```
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.46e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
```

可视化展示



预测投资额

投资额预测分析

1. **基础回归模型预测值：**

$$haty_t = 322.725 + 0.6185 \cdot 3312 - 859.479 \cdot 2.1938 = 485.6720$$

2. **一阶自回归模型预测值：**

$$haty_t = 163.4905 + 0.5623 \cdot 424.5 + 0.699 \cdot 3312 - 0.3930 \cdot 3073.0 - 1009.033 \cdot 2.1938 + 567.3794 \cdot 2.0688 = 469.7638$$

预测结果对比

- **基础回归模型预测结果： 485.6720**
- **一阶自回归模型预测结果：469.7638**