表格

描述已自动生成

使用matlab建模，代码如下：

data = [

1 90.9 596.7 0.7167;

2 97.4 637.7 0.7277;

3 113.5 691.1 0.7436;

4 125.7 756.0 0.7676;

5 122.8 799.0 0.7906;

6 133.3 873.4 0.8254;

7 149.3 944.0 0.8679;

8 144.2 992.7 0.9145;

9 166.4 1077.6 0.9601;

10 195.0 1185.9 1.0000;

11 229.8 1326.4 1.0575;

12 228.7 1434.2 1.1508;

13 206.1 1549.2 1.2579;

14 257.9 1718.0 1.3234;

15 324.1 1918.3 1.4005;

16 386.6 2163.9 1.5042;

17 423.0 2417.8 1.6342;

18 401.9 2631.7 1.7842;

19 474.9 2954.7 1.9514;

20 424.5 3073.0 2.0688

];

investment = data(:, 2);

GNP = data(:, 3);

PI = data(:, 4);

[acf\_lags, acf\_values] = autocorr(investment);

subplot(2, 1, 1);

stem(acf\_lags, acf\_values);

title('自相关函数（ACF）');

xlabel('滞后阶数');

ylabel('自相关系数');

[pacf\_lags, pacf\_values] = parcorr(investment);

subplot(2, 1, 2);

stem(pacf\_lags, pacf\_values);

title('偏自相关函数（PACF）');

xlabel('滞后阶数');

ylabel('偏自相关系数');

model = arima(1, 1, 1);

[estimated\_model, ~, logL] = estimate(model, investment);

diff\_investment = diff(investment);

% 查看diff\_investment维度

[current\_size\_diff\_investment] = size(diff\_investment);

if numel(current\_size\_diff\_investment) > 1 && min(current\_size\_diff\_investment) > 1

% 如果是二维及以上矩阵，尝试将其转换为列向量（这里简单按列方向取元素拼接成列向量，具体依数据逻辑定）

if size(diff\_investment, 2) == 1

diff\_investment = diff\_investment(:);

else

diff\_investment = reshape(diff\_investment', [], 1);

end

end

X = [ones(size(GNP)) GNP PI];

% 查看X维度

[current\_size\_X] = size(X);

% 获取diff\_investment的元素个数（行数）

num\_rows\_diff\_investment = size(diff\_investment, 1);

if current\_size\_X(1) > num\_rows\_diff\_investment

% 如果X行数多于diff\_investment元素个数，删减X多余的行

X = X(1:num\_rows\_diff\_investment, :);

elseif current\_size\_X(1) < num\_rows\_diff\_investment

% 如果X行数少于diff\_investment元素个数，需要检查数据提取等环节是否有遗漏

% 这里假设是GNP或PI数据提取不完整导致，重新正确提取数据来补齐X（示例，具体依实际情况定）

% 假设原始GNP和PI数据分别存储在gnp和pi变量中，你需要根据实际代码中的变量名替换

correct\_gnp = gnp(1:num\_rows\_diff\_investment);

correct\_pi = pi(1:num\_rows\_diff\_investment);

X = [ones(num\_rows\_diff\_investment, 1) correct\_gnp correct\_pi];

end

[b, bint, r, rint, stats] = regress(diff\_investment, X);

disp('回归系数：');

disp(b);

disp('回归系数的置信区间：');

disp(bint);

disp('模型统计信息（R^2值、F值、p值等）：');

disp(stats);

future\_GNP = [3200, 3300];

future\_PI = [2.1, 2.2];

future\_X = [ones(length(future\_GNP), 1), future\_GNP', future\_PI'];

future\_diff\_investment\_pred = future\_X \* b;

initial\_investment = investment(end);

future\_investment\_pred = zeros(size(future\_diff\_investment\_pred));

for i = 1:length(future\_diff\_investment\_pred)

future\_investment\_pred(i) = initial\_investment + sum(future\_diff\_investment\_pred(1:i));

end

disp('未来投资额预测值：');

disp(future\_investment\_pred);

**最后得到结果：**

ARIMA(1,1,1) Model (Gaussian Distribution):

Value StandardError TStatistic PValue

\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_

Constant 19.904 10.848 1.8349 0.066523

AR{1} -0.94067 0.26663 -3.528 0.00041873

MA{1} 1 0.11018 9.076 1.1265e-19

Variance 1011.2 409.64 2.4686 0.013565

回归系数：

-222.1587

-0.3290

623.5225

回归系数的置信区间：

1.0e+03 \*

-0.5059 0.0615

-0.0007 0.0001

-0.1160 1.3630

模型统计信息（R^2值、F值、p值等）：

0.1675 1.6096 0.2307 969.8020

未来投资额预测值：

458.7823

522.5120