信陽師範學院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

第13章 MATLAB预测分析

讲授人：牛言涛

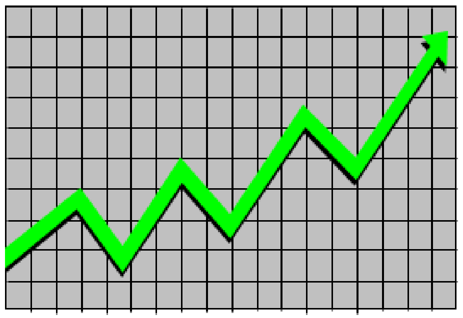
日期：2020年5月13日

目录
CONTENTS

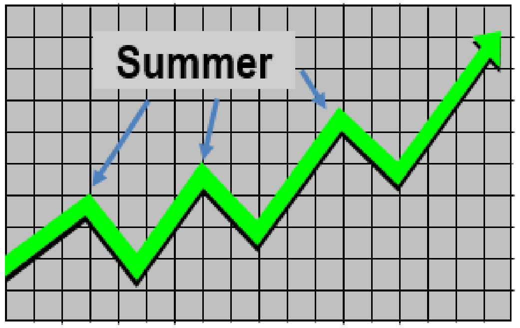
时间序列的构成

- <u>时间序列（或称动态数列）是指将同一统计指标的数值按其发生的时间先后顺序排列而成的数列。</u>时间序列分析的主要目的是根据已有的历史数据对未来进行预测。经济数据中大多数以时间序列的形式给出。根据观察时间的不同，时间序列中的时间可以是年份、季度、月份或其他任何时间形式
- <u>趋势成分</u>：数据随着时间的变化表现出一种趋向(由于人口、技术等原因)。它按某种规则稳步地上升或下降，或停留在某一水平。
- <u>季节成分</u>：在一年里按通常的频率围绕趋势作上下有规则的波动(由于天气、顾客等原因) 。
- <u>周期成分</u>：在较长的时间里（一年以上）围绕趋势作有规则的上下波动。这种波动常被称作经济周期。
- <u>随机成分</u>：由很多不可控因素引起的、没有规则的波动。

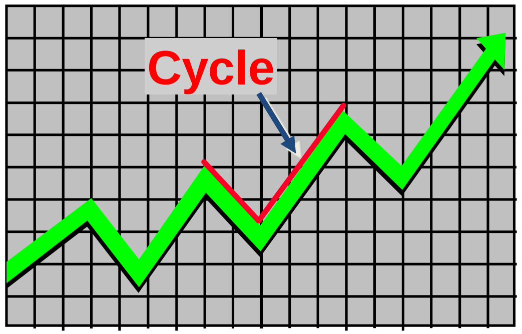
Response



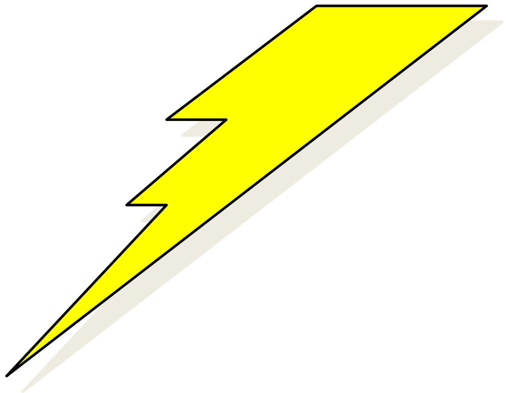
Mo., Qtr., Yr.



Mo., Qtr.

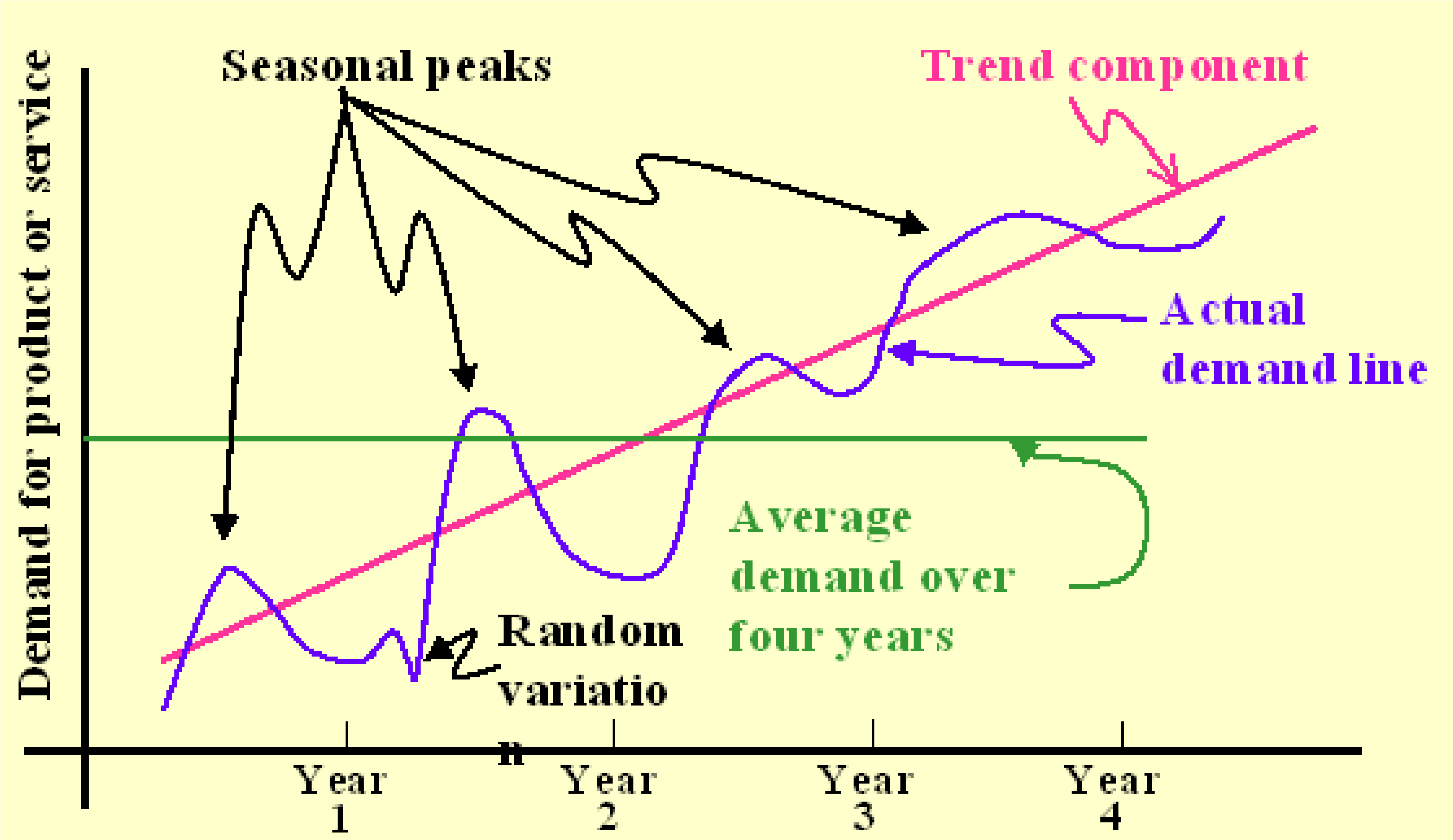


Mo., Qtr., Yr.



时间序列的构成

时间序列曲线及其构成示例



由于随机成分的影响而导致需求偏离平均水平时，应用时间序列平滑模型，通过对多期观测数据平均的办法，可以有效地消除或减少随机成分的影响，以使预测结果较好地反映平均需求水平。

一. 移动平均

- 设观测序列为$y_1, y_2, \cdots, y_T$，取移动平均的项数$N < T$。一次移动平均值计算公式为:

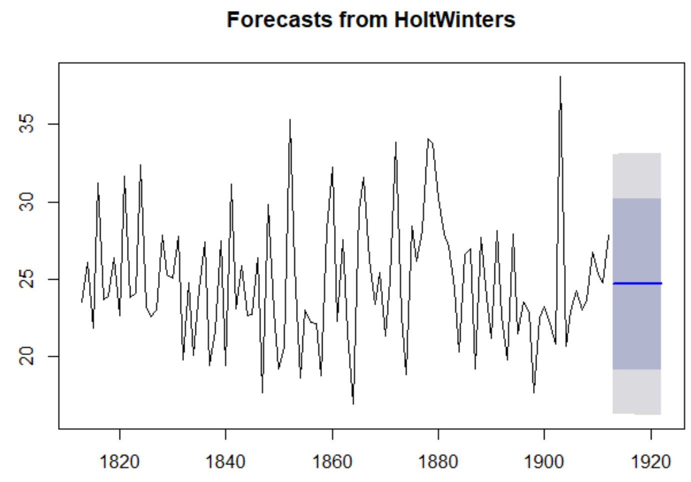
$$M_t^{(1)} = \frac{1}{N}\left(y_t + y_{t-1} + \cdots + y_{t-N+1}\right) = \frac{1}{N}\left(y_{t-1} + \cdots + y_{t-N}\right) + \frac{1}{N}\left(y_t - y_{t-N}\right) = M_{t-1}^{(1)} + \frac{1}{N}\left(y_t - y_{t-N}\right)$$

- 二次移动平均值计算公式为: $M_t^{(2)} = \frac{1}{N}\left(M_t^{(1)} + M_{t-1}^{(1)} + \cdots + M_{t-N+1}^{(1)}\right) = M_{t-1}^{(2)} + \frac{1}{N}\left(M_t^{(1)} - M_{t-N}^{(1)}\right)$

- 当预测目标的基本趋势在某一水平上下波动时，可用一次移动平均方法建立预测模型，即

$$\hat{y}_{t+1} = M_t^{(1)} = \frac{1}{N}\left(y_t + y_{t-1} + \cdots + y_{t-N+1}\right),\ t = N, N+1, \cdots, T$$

- 其预测标准误差为: $S = \sqrt{\dfrac{\sum_{t=N+1}^{T}\left(\hat{y}_t - y_t\right)^2}{T - N}}$

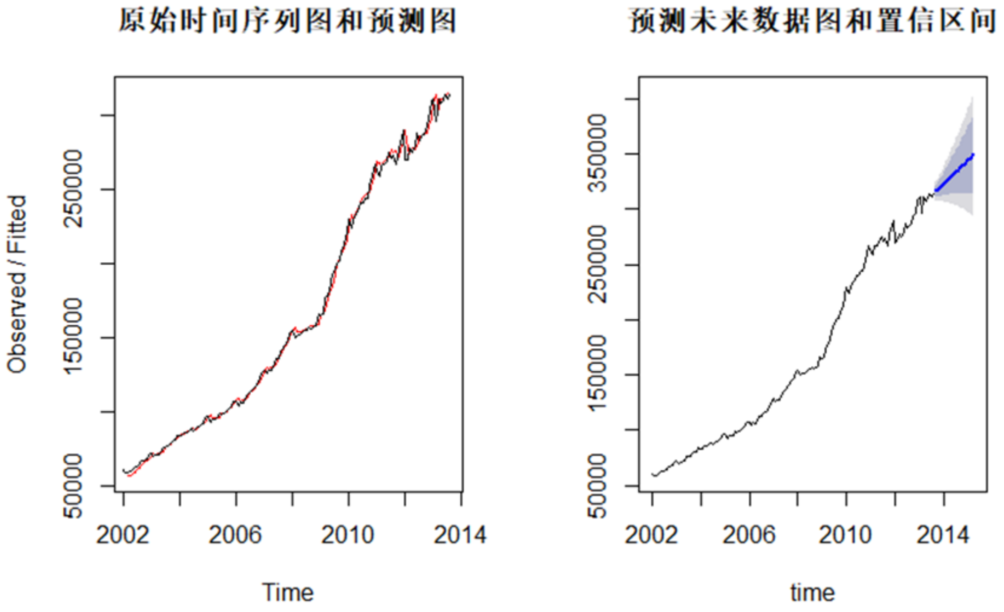


1. 移动平均原理

- 最近N期序列值的平均值作为未来各期的预测结果。一般N取值范围$5 \le N \le 200$。当历史序列的基本趋势变化不大且序列中随即变动成分较多时，N的取值应该较大一些，否则N的取值应小一些。在有确定的季节变动周期的资料中，移动平均的项数应取周期长度。选择最佳N值的一个有效方法是，比较若干模型的预测误差，预测误差最小者为好。
- 当预测目标的基本趋势与某一线性模型相吻合时，常用二次移动平均法，但序列同时存在线性趋势与周期波动时，可用趋势移动平均法建立预测模型：

$$\hat{y}_{T+m} = a_T + b_T m, \quad m = 1, 2, \cdots$$

$$\begin{cases} a_T = 2M_T^{(1)} - M_T^{(2)} \\ b_T = \dfrac{2}{N-1}\left(M_T^{(1)} - M_T^{(2)}\right) \end{cases}$$



2. 移动平均MATLAB编码实现

信阳师范学院
数学与统计学院

moving_average.m

```
function ma = moving_average(X,N,preT)
% moving_average实现移动平均法，X输入列向量，N为移动步长向量
% preT是预测未来期，ma为输出变量结构体，包含预测值和均方误差。

    %% 数据的预处理
    X = rmmissing(X); %删除原数据中的NAN值
    n = length(X); %数据的数量
    m = length(N); %移动步长向量数
    if max(N) >= n
        warning('移动步长N不能超过数据量，试用默认N = length(X)/10');
        N = round(n/10);
    end

    %% 一次移动平均
    Yhat = cell(1,m);
    Err = zeros(1,m);
    RelMeanErr1 = zeros(1,m);
    for i = 1:m
        for j = 1:n-N(i)+1
            Yhat{i}(j) = mean(X(j:j+N(i)-1)); %移动平均
        end
        Err(i) = sqrt(mean((Yhat{i}-X(N(i):end)).^2)); %均方根误差
        RelMeanErr1(i) = mean(abs(Yhat{i}-X(N(i):end))./X(N(i):end));
    end

    %% 二次移动平均
    Yhat2 = cell(1,m);
    at = cell(1,m);
    bt = cell(1,m);
    Err2 = zeros(1,m);
    RelMeanErr2 = zeros(1,m);
    for i = 1:m
        ni = length(Yhat{i});
        for j = 1:ni-N(i)+1
            Yhat2{i}(j) = mean(Yhat{i}(j:j+N(i)-1));
            at{i}(j) = 2*Yhat{i}(j+N(i)-1) - Yhat2{i}(j);
            bt{i}(j) = 2/(N(i)-1)*(Yhat{i}(j+N(i)-1) - Yhat2{i}(j));
        end
        Err2(i) = sqrt(mean((Yhat2{i}-X(2*N(i)-1:end)).^2));
        RelMeanErr2(i) = mean(abs(Yhat2{i}-X(2*N(i)-1:end))...
            ./X(2*N(i)-1:end));
    end
    %% 预测未来期
    Ypre = cell(1,m);
    for i = 1:m
        for t = 1:preT
            Ypre{i}(t) = at{i}(end) + bt{i}(end)*t;
        end
    end

    %% 可视化
    plot(X);
    hold on
    leg = cell(1,m);
    leg{1} = 'OriginalData';
    for i = 1:m
        plot(N(i):n,Yhat{i})
        leg{i+1} = strcat('MA_N = ',num2str(N(i)));
    end
```

Ypre{i}(t) = at{i}(end - N(i) + 1) + bt{i}(end- N(i) + 1)*t;

2. 移动平均MATLAB编码实现

```
    legend(leg,'Location','best')
    legend('boxoff')
    xlabel('Time');ylabel('Value'); hold off
    title('一次移动平均预测法')

    %% 二次移动平均可视化
    figure
    plot(X);hold on
    leg = cell(1,m);
    leg{1} = 'OriginalData';
    for i = 1:m
        plot(2*N(i)-1:n,Yhat2{i})
        leg{i+1} = strcat('MA_N = ',num2str(N(i)));
    end
    legend(leg,'Location','best');legend('boxoff')
    xlabel('Time');ylabel('Value'); hold off
    title('二次移动平均预测法')

    %% 结构体的组合
    ma.Yhat1 = Yhat;
    ma.Err1 = Err;
    ma.RelMeanErr1 = RelMeanErr1;
    ma.Yhat2 = Yhat2;
    ma.at = at;
    ma.bt = bt;
    ma.Err2 = Err2;
    ma.RelMeanErr2 = RelMeanErr2;
    ma.Ypre = Ypre;
end
```

- 代码忽略了残差的检验问题。
- 数项n的数值，要根据时间序列的特点而定。
- 不宜过大或过小，n过大会降低移动平均的敏感性，影响预测的准确性。
- n过小，移动平均数易受随机变动的影响，难以反映实际趋势。
- 一般取n的大小能包含季节变动和周期变动的时期为好，这样可以消除它们的影响。

3. 移动平均案例分析

例1：消费者物价指数（Consumer Price Index），英文缩写为CPI，也称消费价格指数，是反映与居民生活有关的产品及劳务价格统计出来的物价变动指标，通常作为观察通货膨胀水平的重要指标，是与人民生活密切相关的参考指标。

城市居民食品零售价格是消费者物价指数的重要组成部分，权威机构研究认为粮食生产、流通成本上涨一定会带动农产品价格总体上涨。数据是2010-3-5——2011-3-25的城市居民食品零售价格数据，共有42种食品。根据数据建立数学模型，预测2011年4、5月的城市居民食品零售价格走势。

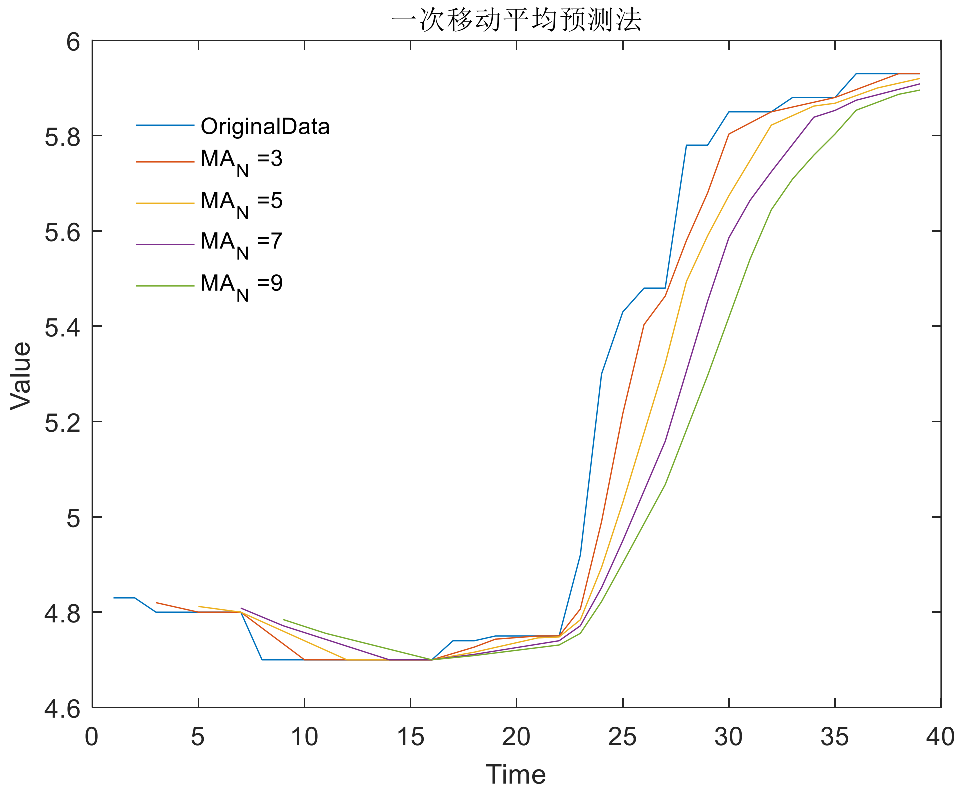
商品名称	类型	单价	均价(10-3-5)	均价(10-3-15)	均价(10-3-25)	均价(10-4-5)	均价(10-4-15)	均价(10-4-25)	…
菜籽油	散装	元/500克	4.83	4.83	4.8	4.8	4.8	4.8	…
大豆油	散装	元/500克	4.38	4.35	4.33	4.33	4.33	4.33	…
花生油	一级桶装	元/5升	95.65	95.65	95.6	95.6	95.6	95.6	…
大豆调和油	一级桶装	元/5升	50.38	50.63	50.48	50.48	50	50.48	…
鲜猪肉	精瘦肉	元/500克	10.93	10.6	10.35	10.35	10.35	11.23	…
鲜猪肉	肋条肉	元/500克	8	7.84	7.55	6.9	6.45	6.53	…
…	…	…	…	…	…	…	…	…	…

3. 移动平均案例分析

```
>> [food,label] = xlsread('foodprice.xlsx');
>> f1 = food(1,:);
>> ma = moving_average(f1',[3,5,7,9],6)
```

```
ma =
  包含以下字段的 struct:

        Yhat1: {[1×37 double]  [1×35 double]  [1×33 double]  [1×31 double]}
         Err1: [0.0775 0.1353 0.1933 0.2507]
   RelMeanErr1: [0.0068 0.0141 0.0221 0.0303]
        Yhat2: {[1×35 double]  [1×31 double]  [1×27 double]  [1×23 double]}
           at: {[1×35 double]  [1×31 double]  [1×27 double]  [1×23 double]}
           bt: {[1×35 double]  [1×31 double]  [1×27 double]  [1×23 double]}
         Err2: [0.1398 0.2575 0.3782 0.5061]
   RelMeanErr2: [0.0141 0.0304 0.0487 0.0714]
         Ypre: {1×4 cell}
```



随着移动步长n的增大，曲线越光滑，但同时误差也越大，n越小，越容易受随机波动的影响。这里选择n = 3，表示一个月的数据波动情况。

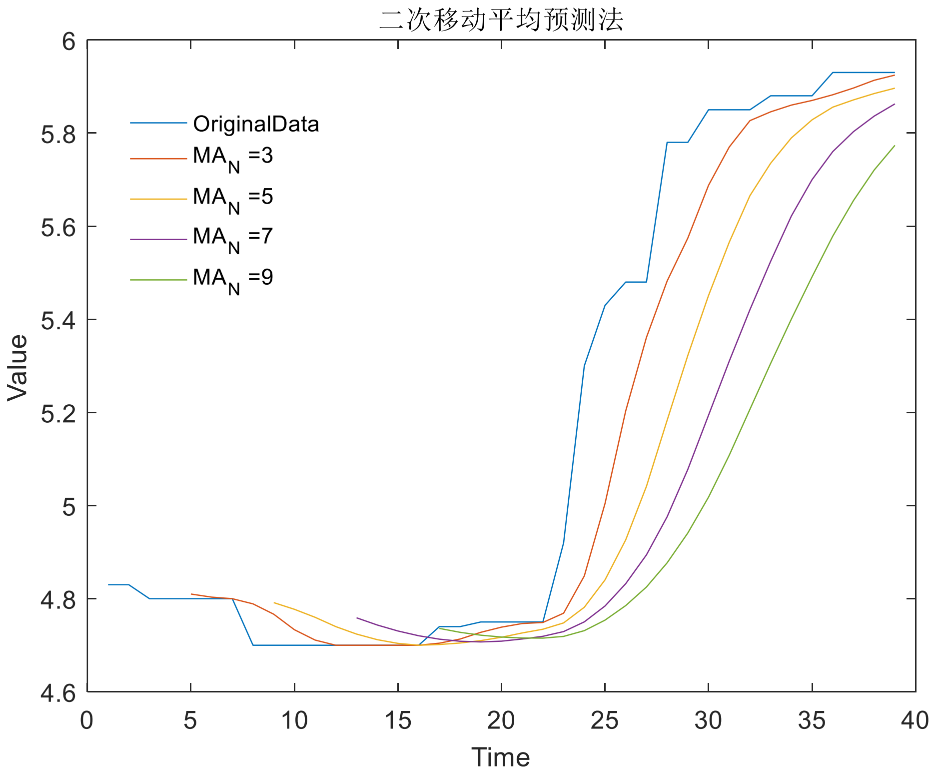
3. 移动平均案例分析

```
>> at = ma.at{1}(end)
at =
    5.9356
>> bt = ma.bt{1}(end)
bt =
    0.0056
```

故模型为：$\hat{y}_{T+m} = 5.9356 + 0.0056m$，$m = 1, 2, \cdots$

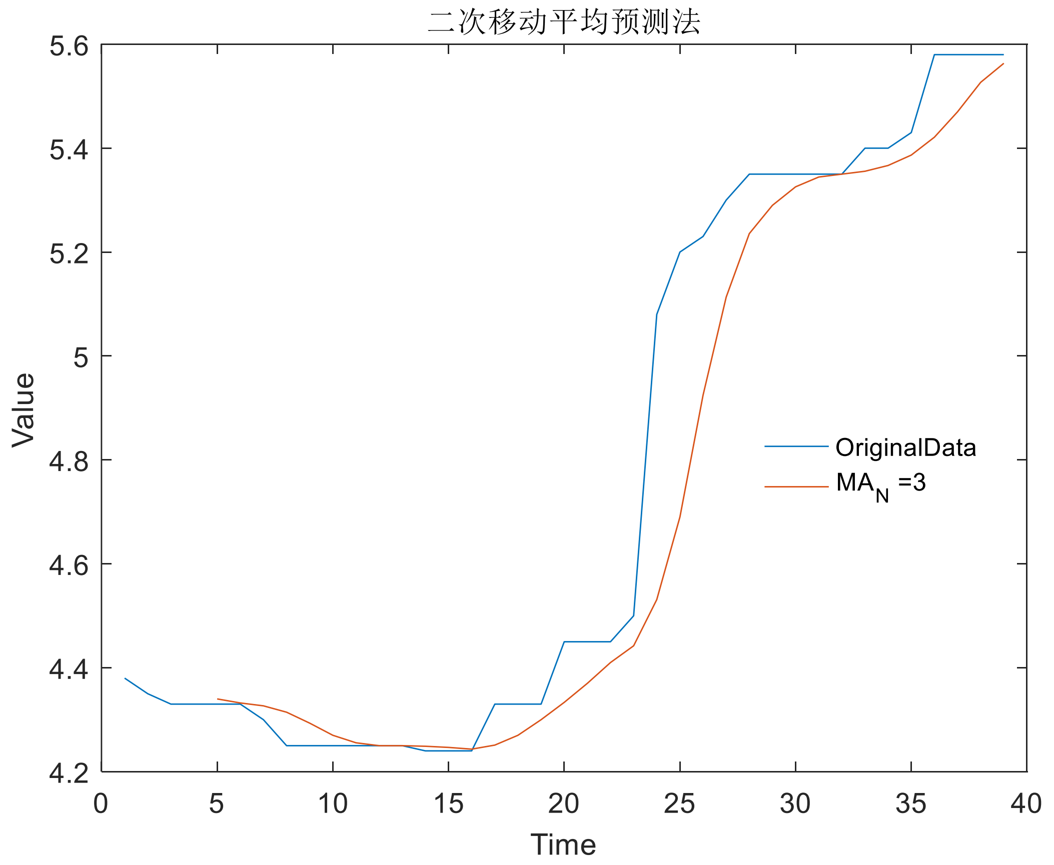
2011年4、5月的城市居民食品零售价格走势如下：

```
>> N3 = ma.Ypre{1}
N3 =
    5.9411    5.9467    5.9522    5.9578    5.9633    5.9689
```



3. 移动平均案例分析

```
>> f2 = food(2,:);   %大豆油
>> ma = moving_average(f2,[3],6)
ma =
  包含以下字段的 struct:
       Yhat1: {[1×37 double]}
         Err1: 0.0871
   RelMeanErr1: 0.0079
        Yhat2: {[1×35 double]}
           at: {[1×35 double]}
           bt: {[1×35 double]}
         Err2: 0.1508
   RelMeanErr2: 0.0163
         Ypre: {[5.6133 5.6300 5.6467 5.6633 5.6800 5.6967]}
```



3. 移动平均案例分析


```
n = size(food,1);
PreData = cell(n,7);
for i = 1:n
    close all
    f = food(i,:);
    ma = moving_average(f,[3],6);
    PreData(i,1) = label(i+1,1);
    PreData(i,2:end) = num2cell(ma.Ypre{1});
end
xlswrite('foodprice.xlsx',PreData,'price','A1');
```

42种食品预测，部分数据如右表所示

	A	B	C	D	E	F	G
1	菜籽油	5.941111111	5.946666667	5.952222222	5.957777778	5.963333333	5.968888889
2	大豆油	5.613333333	5.63	5.646666667	5.663333333	5.68	5.696666667
3	花生油	116.85	116.85	116.85	116.85	116.85	116.85
4	大豆调和油	63.06666667	63.4	63.73333333	64.06666667	64.4	64.73333333
5	鲜猪肉	13.29	13.3	13.31	13.32	13.33	13.34
6	鲜猪肉	9.39	9.196666667	9.003333333	8.81	8.616666667	8.423333333
7	鲜牛肉	18.84444444	19.01666667	19.18888889	19.36111111	19.53333333	19.70555556
8	鲜羊肉	21.69666667	21.77	21.84333333	21.91666667	21.99	22.06333333
9	鲜羊肉	19.2	19.2	19.2	19.2	19.2	19.2
10	活鸡	11.93333333	11.92	11.90666667	11.89333333	11.88	11.86666667
11	鸡肉	8.033333333	7.886666667	7.74	7.593333333	7.446666667	7.3
12	鸡蛋	4.133333333	4.05	3.966666667	3.883333333	3.8	3.716666667
13	带鱼	11.26111111	11.52333333	11.78555556	12.04777778	12.31	12.57222222
14	草鱼	4.826666667	4.58	4.333333333	4.086666667	3.84	3.593333333
15	鲤鱼	5.773333333	5.716666667	5.66	5.603333333	5.546666667	5.49
16	芹菜	2.764444444	2.663333333	2.562222222	2.461111111	2.36	2.258888889
17	大白菜	0.747777778	0.77	0.792222222	0.814444444	0.836666667	0.858888889
18	油菜	1.498888889	1.496666667	1.494444444	1.492222222	1.49	1.487777778
19	黄瓜	2.89	2.846666667	2.803333333	2.76	2.716666667	2.673333333
20	萝卜	0.99	1.04	1.09	1.14	1.19	1.24
21	茄子	3.275555556	3.11	2.944444444	2.778888889	2.613333333	2.447777778
22	西红柿	3.118888889	3.086666667	3.054444444	3.022222222	2.99	2.957777778
23	土豆	2.192222222	2.11	2.027777778	1.945555556	1.863333333	1.781111111
24	胡萝卜	1.825555556	1.73	1.634444444	1.538888889	1.443333333	1.347777778
25	青椒	4.764444444	5.05	5.335555556	5.621111111	5.906666667	6.192222222
26	尖椒	6.958888889	7.123333333	7.287777778	7.452222222	7.616666667	7.781111111
27	圆白菜	1.185555556	1.156666667	1.127777778	1.098888889	1.07	1.041111111
28	豆角	6.841111111	6.95	7.058888889	7.167777778	7.276666667	7.385555556
29	蒜苔	5.223333333	4.706666667	4.19	3.673333333	3.156666667	2.64

Sheet1 price

3. 移动平均案例分析

例2：Box & Jenkins航空公司1949-1960年每月国际航线乘客数，季节性数据。

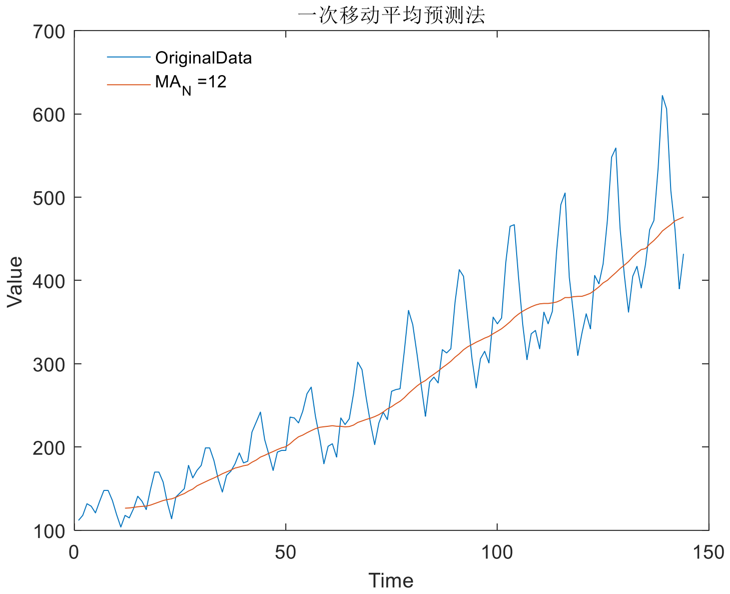
Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1949	112	118	132	129	121	135	148	148	136	119	104	118
1950	115	126	141	135	125	149	170	170	158	133	114	140
1951	145	150	178	163	172	178	199	199	184	162	146	166
1952	171	180	193	181	183	218	230	242	209	191	172	194
1953	196	196	236	235	229	243	264	272	237	211	180	201
1954	204	188	235	227	234	264	302	293	259	229	203	229
1955	242	233	267	269	270	315	364	347	312	274	237	278
1956	284	277	317	313	318	374	413	405	355	306	271	306
1957	315	301	356	348	355	422	465	467	404	347	305	336
1958	340	318	362	348	363	435	491	505	404	359	310	337
1959	360	342	406	396	420	472	548	559	463	407	362	405
1960	417	391	419	461	472	535	622	606	508	461	390	432

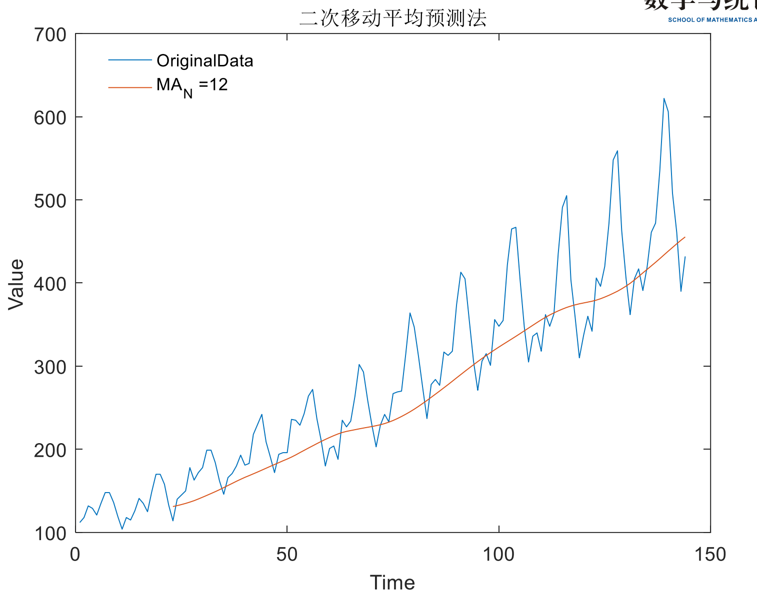
3. 移动平均案例分析


```
>> ap = xlsread('AirPassengers.xlsx');
>> ap(:,1) = [];
>> ap = ap';
>> X = ap(:)';
>> ma = moving_average(X,12,12)
```

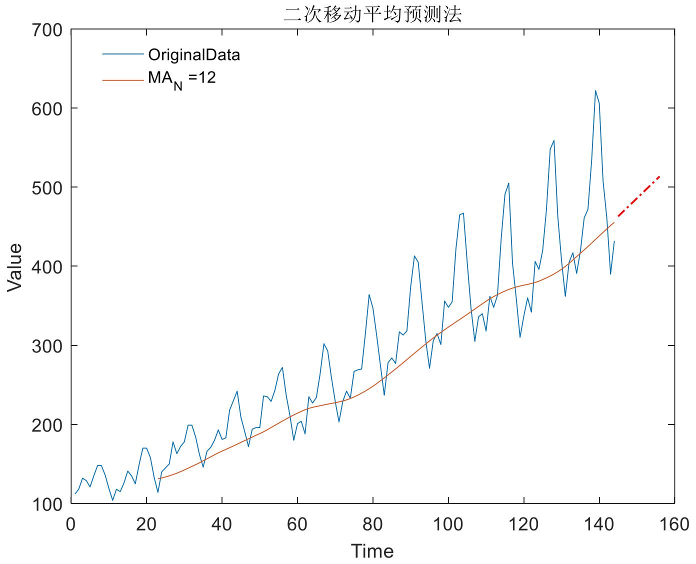
包含以下字段的 struct:

```
        Yhat1: {[1×133 double]}
         Err1: 48.0092
  RelMeanErr1: 0.1078
        Yhat2: {[1×122 double]}
           at: {[1×122 double]}
           bt: {[1×122 double]}
         Err2: 57.1708
  RelMeanErr2: 0.1256
         Ypre: {[1×12 double]}
```






```
at = ma.at{1}(end-11)
bt = ma.bt{1}(end-11)
for i = 1:12
    Y(i) = at + bt*i;
end
 plot(length(X)+1:length(X)+12,Y,'r-.','LineWidth',1)
```



4. 加权移动平均法

- 加权移动平均法，顾名思义，就是在简单移动平均法基础上进行加权计算。运用加权移动平均法时，一般计算的是奇数项加权移动数，各期权数以二项展开式为计算基础，是为了让中项时期指标值的权数最大，两边对称，而越靠边的项权值则越小。
- 当$k=3$时，项数拆分：$(a+b)^2=a^2+2ab+b^2$

  当$k=3$时，计算方式：$\overline{X}_t=\dfrac{X_{t-1}+2X_t+X_{t+1}}{4}$
- 当$k=5$时，项数拆分：$(a+b)^4=a^4+4a^3b+6a^2b^2+4ab^3+b^4$

  当$k=5$时，计算方式：$\overline{X}_t=\dfrac{X_{t-2}+4X_{t-1}+6X_t+4X_{t+1}+X_{t+2}}{16}$
- 移动平均法形成的派生数列的项数要比原来的时间数列的项数少，若按奇数时期项数移动，首尾各少$(k-1)/2$项数值；若按偶数时期项数移动，首尾各少$(k/2)$项数值。因此，移动平均法只用于求时间数列各期的趋势值和观察长期趋势，而很少直接根据派生数列来进行预测。

二. 指数平滑法

- 在做时序预测时，一个显然的思路是：认为离着预测点越近的点，作用越大。比如我这个月体重100斤，去年某个月120斤，显然对于预测下个月体重而言，这个月的数据影响力更大些。
- 假设随着时间变化权重以指数方式下降——最近为0.8，然后0.8^2，0.8^3...，最终年代久远的数据权重将接近于0。将权重按照指数级进行衰减，这就是指数平滑法的基本思想。
- 指数平滑法有几种不同形式：一次指数平滑法针对没有趋势和季节性的序列，二次指数平滑法针对有趋势但没有季节性的序列，三次指数平滑法针对有趋势也有季节性的序列。
- 所有的指数平滑法都要更新上一时间步长的计算结果，并使用当前时间步长的数据中包含的新信息。它们通过"混合"新信息和旧信息来实现，而相关的新旧信息的权重由一个可调整的参数来控制。
- 分类：布朗单参数指数平滑法（一次、二次和三次指数平滑），Holt双参数指数平滑法（alpha, beta），Holt-Winters三参数指数平滑法（alpha, beta , gamma）。

1. 一次指数平滑法

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

- 一次指数平滑法的递推关系如下：

$$s_i = \alpha x_i + (1-\alpha) s_{i-1}, \quad 0 \le \alpha \le 1$$

其中，s_i是时间步长i（理解为第i个时间点）上经过平滑后的值，x_i是这个时间步长上的实际数据，s_{i-1}是时间$i-1$的平滑值。

- α可以是0和1之间的任意值，它控制着新旧信息之间的平衡：当α接近1，就只保留当前数据点；当α接近0时，就只保留前面的平滑值(整个曲线都是平的)。当时间序列相对平稳时，可取较大的α；当时间序列波动较大时，应取较小的α，以不忽略远期实际值的影响。
- 其递推关系式：

$$\begin{aligned} s_i &= \alpha x_i + (1-\alpha) s_{i-1} = \alpha x_i + (1-\alpha)\left[\alpha x_{i-1} + (1-\alpha) s_{i-2}\right] \\ &= \alpha x_i + (1-\alpha)\left[\alpha x_{i-1} + (1-\alpha)\left[\alpha x_{i-2} + (1-\alpha) s_{i-3}\right]\right] \\ &= \alpha\left[x_i + (1-\alpha) x_{i-1} + (1-\alpha)^2 x_{i-2} + (1-\alpha)^3 x_{i-3}\right] = \cdots = \alpha \sum_{j=0}^{i} (1-\alpha)^j x_{i-j} \end{aligned}$$

1. 一次指数平滑法

显然有 $\alpha\sum_{j=0}^{\infty}(1-\alpha)^{j}=\frac{\alpha}{1-(1-\alpha)}=1$，由于加权系数符合指数规律，又具有平滑数据的功能，故称之为指数平滑。

- 可以看出，在指数平滑法中，所有先前的观测值都对当前的平滑值产生了影响，但它们所起的作用随着参数α的幂的增大而逐渐减小。那些相对较早的观测值所起的作用相对较小。同时，称α为记忆衰减因子可能更合适——因为α的值越大，模型对历史数据“遗忘”的就越快。
- 从某种程度来说，指数平滑法就像是拥有无限记忆（平滑窗口足够大）且权值呈指数级递减的移动平均法。一次指数平滑所得的计算结果可以在数据集及范围之外进行扩展，因此也就可以用来进行预测。
- 预测方式为：$x_{i+h}=s_i$，s_i是最后一个已经算出来的值，h等于1代表预测的下一个值。即预测的时间序列为一条直线，不能反映时间序列的趋势和季节性。

1. 一次指数平滑法

信阳师范学院
数学与统计学院

指数平滑预测模型是以时刻 t 为起点，综合历史序列的信息，对未来进行预测的。选择合适的加权系数α是提高预测精度的关键环节。根据实践经验，α的取值范围一般以0.1~0.3为宜。α值愈大，加权系数序列衰减速度愈快，所以实际上α取值大小起着控制参加平均的历史数据的个数的作用。α值愈大意味着采用的数据愈少。因此，可以得到选择α值的一些基本准则。

（1）如果序列的基本趋势比较稳，预测偏差由随机因素造成，则α值应取小一些，如0.1~0.5，以减少修正幅度，使预测模型能包含更多历史数据的信息。

（2）如果预测目标的基本趋势已发生系统地变化，则α值应取得大一些，如0.6~0.8。这样，可以偏重新数据的信息对原模型进行大幅度修正，以使预测模型适应预测目标的新变化。

2. 二次指数平滑法

一次指数平滑法虽然克服了移动平均法的缺点。但当时间序列的变动出现直线趋势时，用一次指数平滑法进行预测，仍存在明显的滞后偏差。因此，也必须加以修正。修正的方法与趋势移动平均法相同，即再作二次指数平滑，利用滞后偏差的规律建立直线趋势模型。这就是二次指数平滑法。其计算公式为

$$S_t^{(1)} = \alpha y_t + (1-\alpha) S_{t-1}^{(1)}$$

$$S_t^{(2)} = \alpha S_t^{(1)} + (1-\alpha) S_{t-1}^{(2)}$$

式中$S_t^{(1)}$为一次指数的平滑值； $S_t^{(2)}$为二次指数的平滑值。当时间序列$\{y_t\}$从某时期开始具有直线趋势时，类似趋势移动平均法，可用如下直线趋势模型进行预测。

$$\hat{y}_{t+T} = a_t + b_t T, \quad T = 1, 2, \cdots$$

$$\begin{cases} a_t = 2S_t^{(1)} - S_t^{(2)} \\ b_t = \dfrac{\alpha}{1-\alpha}\left(S_t^{(1)} - S_t^{(2)}\right) \end{cases}$$

3. 三次指数平滑法

当时间序列的变动表现为二次曲线趋势时，则需要用三次指数平滑法。三次指数平滑是在二次指数平滑的基础上，再进行一次平滑，其计算公式为

$$\begin{cases} S_t^{(1)} = \alpha y_t + (1-\alpha) S_{t-1}^{(1)} \\ S_t^{(2)} = \alpha S_t^{(1)} + (1-\alpha) S_{t-1}^{(2)} \\ S_t^{(3)} = \alpha S_t^{(2)} + (1-\alpha) S_{t-1}^{(3)} \end{cases}$$

，式中$S_t^{(3)}$为三次指数平滑值。

$$\hat{y}_{t+T} = a_t + b_t T + C_t T^2, \quad T = 1, 2, \cdots$$

三次指数平滑法的预测模型为

$$\begin{cases} a_t = 3S_t^{(1)} - 3S_t^{(2)} + S_t^{(3)} \\ b_t = \left(\alpha \big/ 2(1-\alpha)^2\right)\left[(6-5\alpha) S_t^{(1)} - 2(5-4\alpha) S_t^{(2)} + (4-3\alpha) S_t^{(3)}\right] \\ c_t = \left(\alpha^2 \big/ 2(1-\alpha)^2\right)\left[S_t^{(1)} - 2S_t^{(2)} + S_t^{(3)}\right] \end{cases}$$

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

4. MATLAB程序实现


```matlab
function esf =  ExpSF_model(X,t,cycle,alpha)
% ExpSF_model用于指数平滑预测法Exponential Smoothing Forecasts
% 输入参数X是时间序列，t是未来预期数，cycle为周期数，若不存在季节周期，则为1
% alpha是指定参数，缺省按照误差最小法则自动选择alpha值
% 输出参数esf是结构体，包括一次、二次、三次平滑值和预测值，以及误差等。

    n = length(X); %样本数量
    %% 根据是否指定alpha进行控制，如果指定，则采用指定的alpha，
    % 否则按照误差最小进行选择
    if nargin == 4
        [S1,err1,S2,Y2t,err2,S3,Y3t,err3,at2,bt2,at3,bt3,ct3] = ...
            expsf_method(X,alpha); %采用指定的alpha
    elseif nargin == 3
        k = 1;
        alpha = 0.1:0.05:1;
        Err3 = zeros(1,length(0.1:0.05:1));
        for ap = alpha
            [~,~,~,~,~,~,~,err3,~,~,~,~,~] = expsf_method(X,ap);
            Err3(k) = err3;
            k = k + 1;
        end
        [~,ind] = min(Err3);
        alpha = alpha(ind);
        [S1,err1,S2,Y2t,err2,S3,Y3t,err3,at2,bt2,at3,bt3,ct3] = ...
            expsf_method(X,alpha);
    elseif nargin == 2
        cycle = 1;
    else
        warning('输入参数数目不足！');
        return
    end

    %% 预测
    PreY2 = zeros(1,t);
    PreY3 = zeros(1,t);
    for k = 1:t
        if cycle  == 1
            PreY2(k) = at2(end-cycle+1) + bt2(end-cycle+1)*(k+1);
            PreY3(k) = at3(end-cycle+1) + bt3(end-cycle+1)*(k+1) + ...
                ct3(end-cycle+1)*(k+1)^2;
        elseif cycle > 1
            PreY3(k) = at3(end-cycle+1) + bt3(end-cycle+1)*(k+1) + ...
                ct3(end-cycle+1)*(k+1)^2;
        end
    end

    %% 可视化——一次指数平滑
    T = 1:n;
    subplot(2,2,1)
    plot(T,X,'LineWidth',1);
    hold on; grid on
    plot(T,S1,'LineWidth',1);
    leg = strcat('alpha = ',num2str(alpha));
    legend('OriginalData',leg,'Location','best')
    legend('boxoff')
    title('One time exponential smoothing forecasts');
    xlabel('Time'); ylabel('Value')

    %% 可视化——二次指数平滑
    subplot(2,2,2)
    plot(T,X,'LineWidth',1);
    hold on; grid on
    plot(T,Y2t,'LineWidth',1);
```

4. MATLAB程序实现


```
    if cycle == 1
        plot(n+1:n+t,PreY2,'r-')
        legend('Original Data',leg,'Forecatst Value','Location','best')
    else
        legend('Original Data',leg,'Location','best')
    end
    legend('boxoff')
    title('Quadratic exponential smoothing forecasts');
    xlabel('Time'); ylabel('Value')

    %% 可视化——三次指数平滑
    subplot(2,2,3)
    plot(T,X,'LineWidth',1);
    hold on; grid on
    plot(T,Y3t,'LineWidth',1);
    plot(n+1:n+t,PreY3,'r-')
    legend('Original Data',leg,'Forecatst Value','Location','best')
    legend('boxoff')
    title('Triple exponential smoothing forecasts');
    xlabel('Time'); ylabel('Value')

    %% 三次平滑对比分析
    subplot(2,2,4)
    plot(T,X,'k-','LineWidth',1);
    hold on; grid on
    plot(1:n,S1,'r-',1:n,Y2t,'b-',1:n,Y3t,'g-');
    legend('OriginalData','Onetime','Quadratic','Triple','Location','best')
    legend('boxoff')
    title(strcat('alpha = ',num2str(alpha),...
        ' exponential smoothing forecasts'));
    xlabel('Time'); ylabel('Value')
```

```
    %% 输出结构体组合
    esf.S1 = S1;
    esf.Error1 = err1;
    esf.S2 = S2;
    esf.Y2t = Y2t;
    esf.Error2 = err2;
    esf.S3 = S3;
    esf.Y3t = Y3t;
    esf.Error3 = err3;
    esf.PreY2 = PreY2;
    esf.PreY3 = PreY3;
    esf.at2 = at2;
    esf.bt2 = bt2;
    esf.at3 = at3;
    esf.bt3 = bt3;
    esf.ct3 = ct3;

    %% 实现一次、二次和三次指数平滑
    function [S1,err1,S2,Y2t,err2,S3,Y3t,err3,at2,bt2,at3,bt3,ct3] ...
            = expsf_method(X,alpha)
        %% 一次指数平滑
        S1 = zeros(n,1);
        S0 = mean(X(1:3));%初始化为三个值的均值
        S1(1) = alpha*X(1) + (1-alpha)*S0;
        for i = 2:n
            S1(i) = alpha*X(i-1) + (1-alpha)*S1(i-1);
        end
        err1 = sqrt(mean((S1 - X).^2));

        %% 二次指数平滑
        S2 = zeros(n,1);
```

4. MATLAB程序实现

信阳师范学院
数学与统计学院

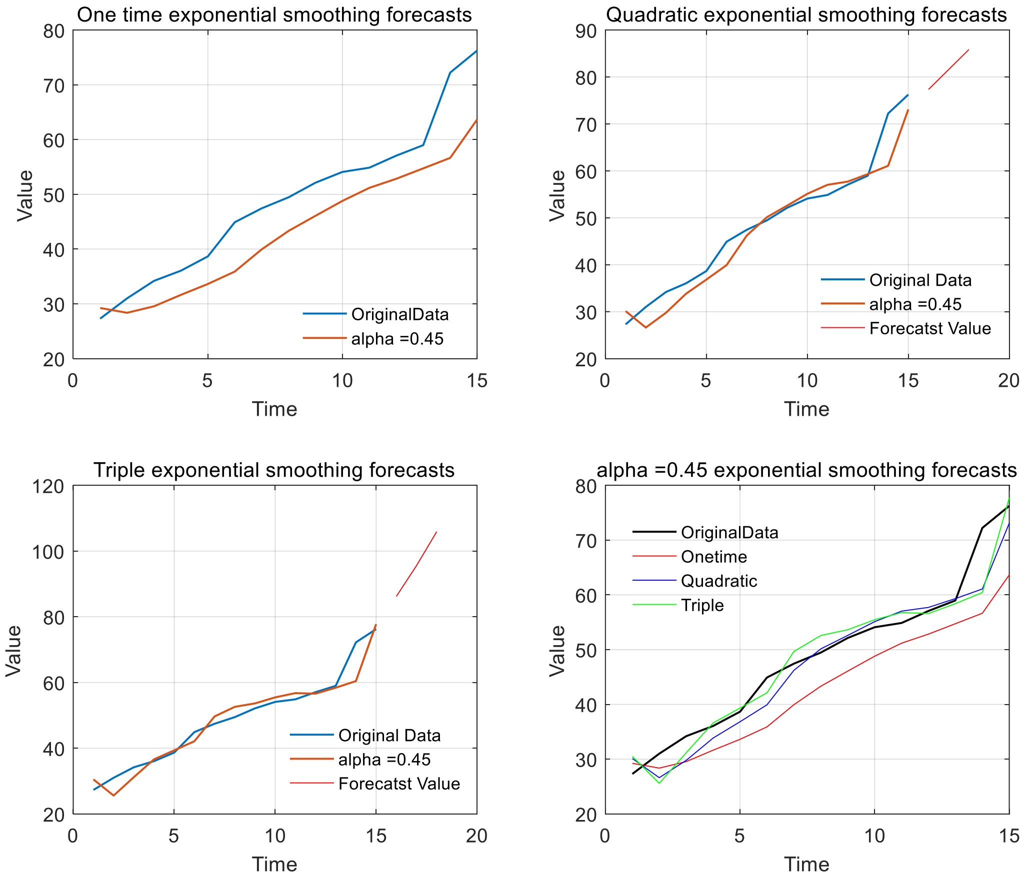
```
        at2 = zeros(n,1);
        bt2 = zeros(n,1);
        Y2t = zeros(n,1);
        S2(1) = alpha*S1(1)+(1-alpha)*S0;
        Y2t(1) = S2(1);
        for i = 2:n
            S2(i) = alpha*S1(i) + (1-alpha)*S2(i-1);
            at2(i) = 2*S1(i) - S2(i);
            bt2(i) = alpha/(1-alpha)*(S1(i) - S2(i));
            Y2t(i) = at2(i) + bt2(i);
        end
        err2 = sqrt(mean((Y2t - X).^2));

        %% 三次指数平滑
        S3 = zeros(n,1);
        Y3t = zeros(n,1);
        at3 = zeros(n,1);
        bt3 = zeros(n,1);
        ct3 = zeros(n,1);
        S3(1) = alpha*S2(1)+(1-alpha)*S0;
        Y3t(1) = S3(1);
        for i = 2:n
            S3(i) = alpha*S2(i) + (1-alpha)*S3(i-1);
            at3(i) = 3*S1(i) - 3*S2(i) + S3(i);
            bt3(i) = alpha/(2*(1-alpha)^2)*(((6-5*alpha)*S1(i) - ...
                    2*(5-4*alpha)*S2(i))+(4-3*alpha)*S3(i));
            ct3(i) = alpha^2/(2*(1-alpha)^2)*(S1(i) - 2*S2(i) + S3(i));
            Y3t(i) = at3(i) + bt3(i) + ct3(i);
        end
        err3 = sqrt(mean((Y3t - X).^2));
    end
end
```

X1 = [76.2464 72.2225 58.9812 57.0639 54.8689 54.0919 52.1303 49.4609 47.4415 44.9042 38.6658 36.0590 34.1970 31.0037 27.3002]; %程序测试

X1 = fliplr(X1);

esf = ExpSF_model(X',3,1) %历年硕士招生人数（国家数据网）



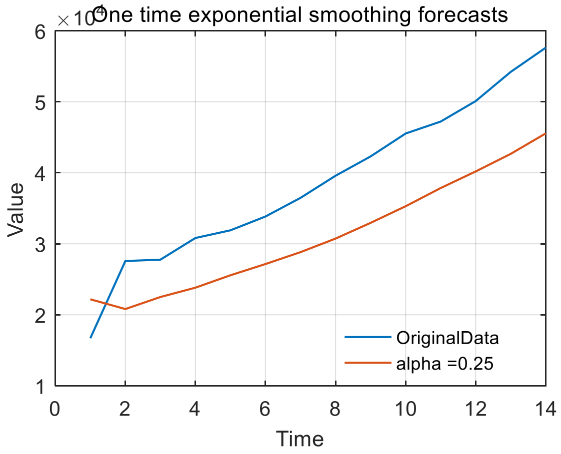
4. MATLAB程序实现

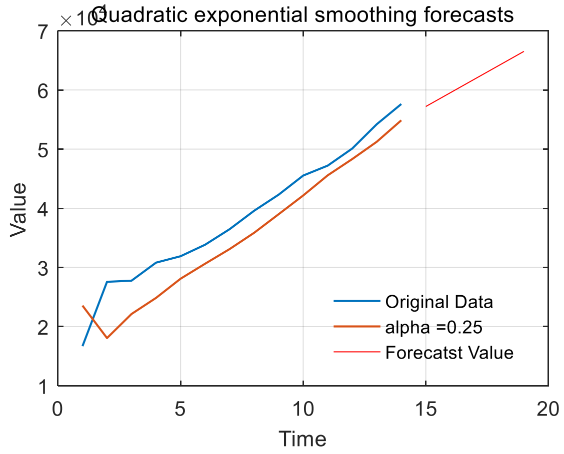
% 国家数据网，生活能源消费量(万吨标准煤)，从2000年到2017年

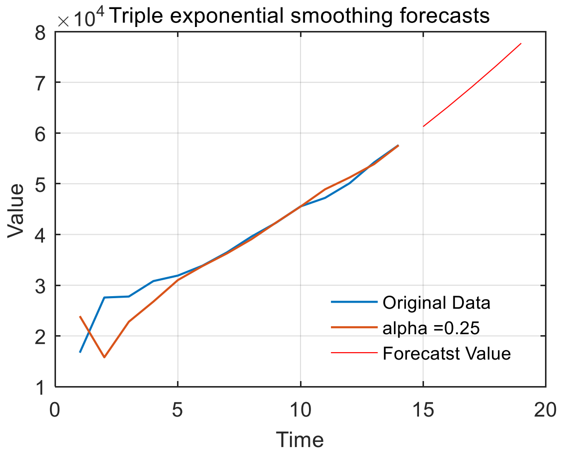
```
>> X = [57620 54209 50099 47212 45531 42306 39584 36470 33843 31898 30814 27765 27573 16695];
>> X = fliplr(X);
>> esf =  ExpSF_model(X',5,1)
```

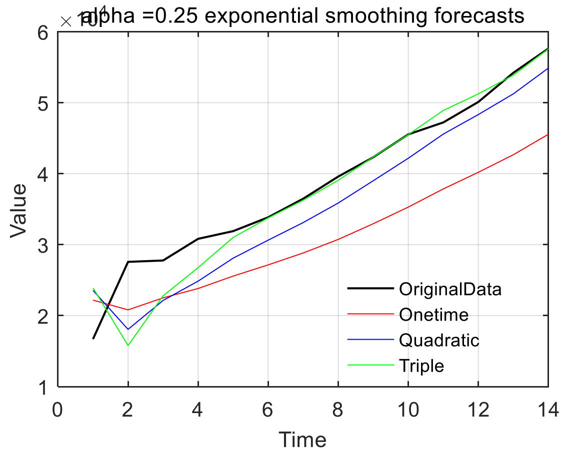
```
esf =
  包含以下字段的 struct:

        S1: [14×1 double]
    Error1: 8.5837e+03
        S2: [14×1 double]
       Y2t: [14×1 double]
    Error2: 4.6213e+03
        S3: [14×1 double]
       Y3t: [14×1 double]
    Error3: 4.1181e+03
     PreY2: [5.7200e+04  5.9530e+04  6.1859e+04  6.4189e+04  6.6519e+04]
     PreY3: [6.1271e+04  6.5127e+04  6.9152e+04  7.3347e+04  7.7712e+04]
       at2: [14×1 double]
       bt2: [14×1 double]
       at3: [14×1 double]
       bt3: [14×1 double]
       ct3: [14×1 double]
```







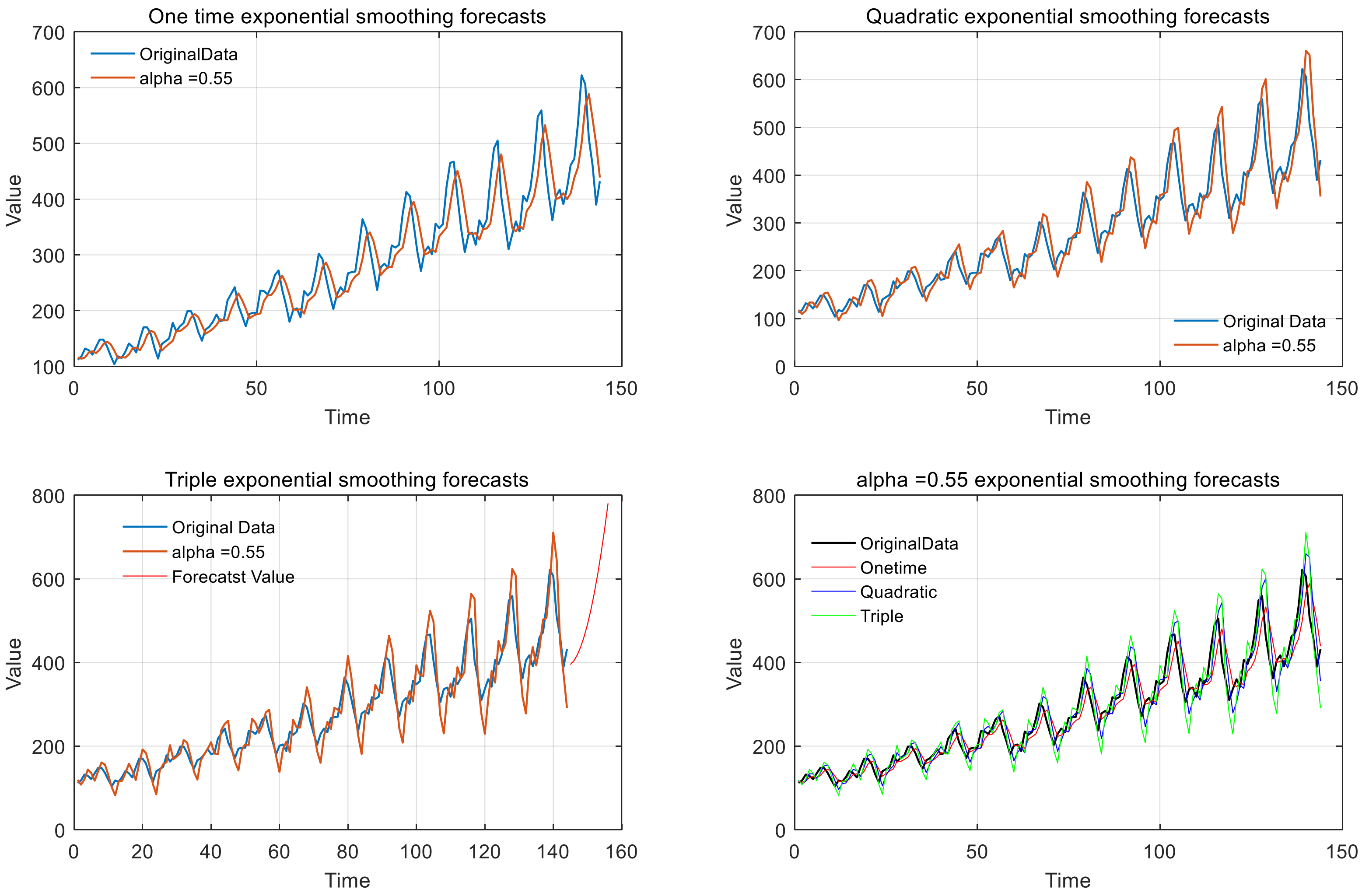


4. MATLAB程序实现


```
>> ap = xlsread('AirPassengers.xlsx');
>> ap(:,1) = [];
>> ap = ap';
>> X = ap(:);
>> esf =  ExpSF_model(X,12,12)
```

```
esf =
  包含以下字段的 struct:

        S1: [144×1 double]
    Error1: 40.5545
        S2: [144×1 double]
       Y2t: [144×1 double]
    Error2: 40.3625
        S3: [144×1 double]
       Y3t: [144×1 double]
    Error3: 44.5475
     PreY2: [0 0 0 0 0 0 0 0 0 0 0 0]
     PreY3: [395.0960 403.8287 417.8252 437.
       at2: [144×1 double]
       bt2: [144×1 double]
       at3: [144×1 double]
       bt3: [144×1 double]
       ct3: [144×1 double]
```



按照最小误差，参数alpha选取0.55，未来一年12个月的预测。

5. Holt-Winters指数平滑法

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

- 趋势或者说斜率的定义：$b=\Delta y/\Delta x$，其中Δx为两点在x坐标轴的变化值，所以对于一个序列而言，相邻两个点的$\Delta x=1$，因此$b=\Delta y=y(x)-y(x-1)$。 除了用点的增长量表示，也可以用二者的比值表示趋势。比如可以说一个物品比另一个贵20块钱，等价地也可以说贵了5%，前者称为可加的（addtive），后者称为可乘的（multiplicative）。在实际应用中，可乘的模型预测稳定性更佳。
- 指数平滑考虑的是数据的baseline，二次指数平滑在此基础上将趋势作为一个额外考量，保留了趋势的详细信息。即保留并更新两个量的状态：平滑后的信号和平滑后的趋势。公式如下：

$$s_i=\alpha x_i+(1-\alpha)(s_{i-1}+t_{i-1}),\quad t_i=\beta(s_i-s_{i-1})+(1-\beta)t_{i-1}$$

第二个等式描述了平滑后的趋势。当前趋势的未平滑“值”（t_i）是当前平滑值（s_i）和上一个平滑值（s_{i-1}）的差；也就是说，当前趋势告诉我们在上一个时间步长里平滑信号改变了多少。要想使趋势平滑，我们用一次指数平滑法对趋势进行处理，并使用参数β（理解：对t_i的处理类似于一次平滑指数法中的s_i，即对趋势也需要做一个平滑，临近的趋势权重大）。

5. Holt-Winters指数平滑法

- 为获得平滑信号，像上次那样进行一次混合，但要同时考虑到上一个平滑信号及趋势。假设单个步长时间内保持着上一个趋势，那么第一个等式的最后那项就可以对当前平滑信号进行估计。若要利用该计算结果进行预测，就取最后那个平滑值，然后每增加一个时间步长就在该平滑值上增加一次最后那个平滑趋势：$x_{i+h} = s_i + ht_i$，二次指数平滑的预测结果是一条斜的直线。
- 三次指数平滑就是Holt-Winters方法。当一个序列在每个固定的时间间隔中都出现某种重复的模式，就称之具有季节性特征，而这样的一个时间间隔称为一个季节。一个季节的长度k为它所包含的序列点个数。二次指数平滑考虑了序列的baseline和趋势，三次就是在此基础上增加了一个季节分量。类似于趋势分量，对季节分量也要做指数平滑。比如预测下一个季节第3个点的季节分量时，需要指数平滑地考虑当前季节第3个点的季节分量、上个季节第3个点的季节分量...等等。

5. Holt-Winters指数平滑法

计算公式(左为累加法，右为累乘法)，其中， p_i是指“周期性”部分：

$$\begin{cases} s_i = \alpha\left(x_i - p_{i-k}\right) + (1-\alpha)\left(s_{i-1} + t_{i-1}\right) \\ t_i = \beta\left(s_i - s_{i-1}\right) + (1-\beta)t_{i-1} \\ p_i = \gamma\left(x_i - s_i\right) + (1-\gamma)p_{i-k} \end{cases}$$

$$\begin{cases} s_i = \dfrac{\alpha x_i}{p_{i-k}} + (1-\alpha)\left(s_{i-1} + t_{i-1}\right) \\ t_i = \beta\left(s_i - s_{i-1}\right) + (1-\beta)t_{i-1} \\ p_i = \dfrac{\gamma x_i}{s_i} + (1-\gamma)p_{i-k} \end{cases}$$

预测公式 $\begin{cases} \text{累加：} & x_{i+h} = s_i + ht_i + p_{i-k+(h\bmod k)} \\ \text{累乘：} & x_{i+h} = \left(s_i + ht_i\right)p_{i-k+(h\bmod k)} \end{cases}$ ， k是这个周期的长度。

- α,β,γ的值都位于[0,1]之间，可以多试验几次以达到最佳效果。当然，一些寻优方法，比如贝叶斯调参，网格调参可用于调整参数。
- s,t,p 初始值的选取对于算法整体的影响不是特别大，通常的取值$s_0 = x_0$，$t_0 = x_1 - x_0$，累加时 $p = 0$，累乘时 $p = 1$ 。

6. Holt-Winters指数平滑法MATLAB代码实现

TriExpSF_model.m

```matlab
function esf =  TriExpSF_model(X,k,fc,alpha,beta,gamma)
% TriExpSF_model实现三次指数平滑模型，适用于季节周期性数据
% 输入参数：X是时间序列数据，行向量形式；alpha为数据平滑因子
% beta为趋势平滑因子，gamma为季节改变平滑因子
% k为周期数，如4个季节，12个月等，fc为预测期数
% 输出参数esf是一个结构体，组合各种计算结果。

    if size(X,1) > size(X,2)
        X = X'; %行向量形式
    end
    n = length(X);
    if nargin == 3  % 按照误差最小选择参数
        [alpha,beta,gamma] = expsf(X);
    end
    XM = X; %用作累乘

    %% 初始化第一个值且建模预测，累加形式
    s = zeros(1,n+fc);
    t = zeros(1,n+fc);
    s(1) = mean(X(1:k)); %一个周期内均值
    t(1) = (sum(X(k+1:2*k))-sum(X(1:k)))/k^2; %趋势估计值
    p = X - s(1);
    preYA = zeros(1,n+fc);
    preYA(1) = s(1) + t(1) + p(1);
    for i = 1:n+fc
        if i == length(X)
            %X(i+1) = s(end) + t(end) + p(end-k+1);
            X(i+1) = s(i-1) + t(i-1) + p(i-1-k+1); %未来期平滑值
        end
        s(i+1) = alpha*(X(i)-p(i)) + (1-alpha)*(s(i)+t(i));
        t(i+1) = beta*(s(i+1)-s(i)) + (1-beta)*t(i);%趋势
        p(i+1) = gamma*(X(i)-s(i) - t(i)) + (1-gamma)*p(i);%周期
        preYA(i+1) = s(i+1) + t(i+1) + p(i+1); %模拟值
    end
    SqMErrorA = sqrt(mean((preYA(1:n) - X(1:n)).^2));

    %% 建模预测，累乘形式
    sM = zeros(1,n+fc);
    tM = zeros(1,n+fc);
    sM(1) = mean(XM(1:k)); %一个周期内均值
    tM(1) = (sum(XM(k+1:2*k))-sum(XM(1:k)))/k^2; %趋势估计值
    pM = XM/s(1);
    preYM = zeros(1,n+fc);
    preYM(1) = (sM(1) + tM(1)) * pM(1);
    for i = 1:n+fc
        if i == length(XM)
            XM(i+1) = (sM(i-1) + tM(i-1)) * pM(i-1-k+1); %未来期平滑值
        end
        sM(i+1) = alpha*XM(i)/pM(i) + (1-alpha)*(sM(i)+tM(i));
        tM(i+1) = beta*(sM(i+1)-sM(i)) + (1-beta)*tM(i);%趋势
        pM(i+1) = gamma*XM(i)/(sM(i)+tM(i)) + (1-gamma)*pM(i);%周期
        preYM(i+1) = (sM(i+1) + tM(i+1)) * pM(i+1); %模拟值
    end
    SqMErrorM = sqrt(mean((preYM(1:n) - XM(1:n)).^2));

    %% 可视化
    subplot(2,1,1)
    plot(1:n,X(1:n),'k-','LineWidth',1);
    hold on; grid on
    plot(1:n,preYA(1:n),'b-','LineWidth',1);
```

6. Holt-Winters指数平滑法MATLAB代码实现

```
plot(n+1:n+fc,X(n+1:n+fc),'c-','LineWidth',1);
plot(n+1:n+fc,preYA(n+1:n+fc),'r-','LineWidth',1)
legend('OriginalData','ForecastCurrentData','Future smoothing value',...
    'ForecastValue','Location','best')
legend('boxoff')
title('Accumulation——Triple exponential smoothing forecasts');
xlabel('Time'); ylabel('Value')

subplot(2,1,2)
plot(1:n,XM(1:n),'k-','LineWidth',1);
hold on; grid on
plot(1:n,preYM(1:n),'b-','LineWidth',1);
plot(n+1:n+fc,XM(n+1:n+fc),'c-','LineWidth',1);
plot(n+1:n+fc,preYM(n+1:n+fc),'r-','LineWidth',1)
legend('OriginalData','ForecastCurrentData','Future smoothing value',...
    'ForecastValue','Location','best')
legend('boxoff')
title('Multiplicative——Triple exponential smoothing forecasts');
xlabel('Time'); ylabel('Value')

%% 结构体组合
esf.AlphaBetaGamma = [alpha,beta,gamma];
esf.S = [s;sM];
esf.T = [t;tM];
esf.P = [p;pM];
esf.Accumulation_preY = preYA;
esf.SqMErrorA = SqMErrorA;
esf.ErrorA = preYA(1:n) - X(1:n);
esf.Multiplicative_preY = preYM;
esf.SqMErrorM = SqMErrorM;
esf.ErrorM = preYM(1:n) - XM(1:n);
```

```
%% 按照最小误差选择参数
function [alpha,beta,gamma] = expsf(X)
    XA = X;
    alpha = 0.1:0.1:1;
    beta = 0.1:0.1:1;
    gamma = 0.1:0.1:1;
    ir = 1;
    Error = zeros(1,3*length(alpha));
    sa = zeros(1,n+fc);
    ta = zeros(1,n+fc);
    for a = alpha
        for b = beta
            for g = gamma
                %% 初始化第一个值
                sa(1) = mean(XA(1:k)); %一个周期内均值
                %趋势估计值
                ta(1) = (sum(XA(k+1:2*k))-sum(XA(1:k)))/k^2;
                pa = XA - sa(1);
                preY = zeros(1,n+fc);
                preY(1) = sa(1) + ta(1) + pa(1);

                %% 建模预测，累加形式
                for j = 1:n+fc
                    if j == length(XA)
                        %未来期平滑值
                        XA(j+1) = sa(j-1) + ta(j-1) + pa(j-1-k+1);
                    end
```

信阳师范学院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

信阳师范学院
数学与统计学院

6. Holt-Winters指数平滑法MATLAB代码实现

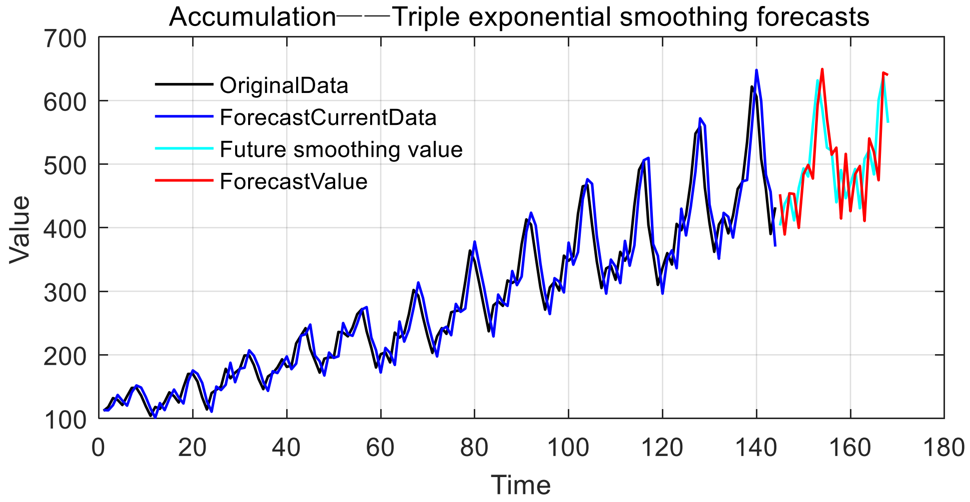
```
                        sa(j+1) = a*(XA(j)-pa(j)) + (1-a)*(sa(j)+ta(j));
                        ta(j+1) = b*(sa(j+1)-sa(j)) + (1-b)*ta(j);%趋势
                        pa(j+1) = g*(XA(j)-sa(j) - ta(j)) + (1-g)*pa(j);%周期
                        preY(j+1) = sa(j+1) + ta(j+1) + pa(j+1); %模拟值
                    end
                    ErrorA = sqrt(mean((preY(1:n) - XA(1:n)).^2));
                    Error(ir) = ErrorA;
                    ir = ir + 1;
                end
            end
        end
        %% 按照最小误差分离参数
        [~,ind] = min(Error);
        if floor(ind/100) == 0 %如56
            alpha = alpha(1);
            if floor(ind/10) == 0 %如5
                beta = beta(1);
                gamma = gamma(ind);
            else
                beta = beta(floor(ind/10));
                if mod(ind,10) == 0  %如50
                    gamma = gamma(1);
                else
                    gamma = gamma(mod(ind,10));
                end
            end
            return;
        end
        inda = floor(ind/100); %取alpha所在索引
        alpha = alpha(inda);
        indb = floor(mod(ind,100)/10); %取beta所在索引
        if mod(ind,100) == 0  %如400
            beta = beta(1);
            gamma = gamma(1);
        elseif mod(ind,10) == 0  %如420
            beta = beta(indb);
            gamma = gamma(end);
        else  %如425
            beta = beta(indb);
            indc = mod(ind,10); %取gamma所在索引
            gamma = gamma(indc);
        end
    end
end
```

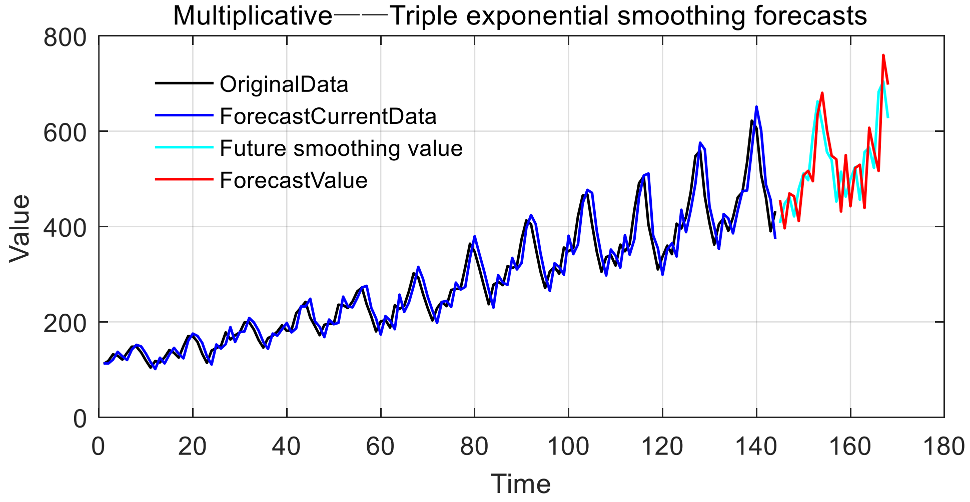
7. 案例分析

```
>> ap = xlsread('AirPassengers.xlsx');
>> ap(:,1) = [];
>> ap = ap';
>> X = ap(:);
>> esf =  TriExpSF_model(X,12,24)
```

```
esf =
  包含以下字段的 struct:

          AlphaBetaGamma: [0.3000 0.1000 1]
                       S: [2×169 double]
                       T: [2×169 double]
                       P: [2×169 double]
       Accumulation_preY: [1×169 double]
               SqMErrorA: 31.8924
                  ErrorA: [1×144 double]
     Multiplicative_preY: [1×169 double]
               SqMErrorM: 32.0466
                  ErrorM: [1×144 double]
```





7. 案例分析

例3：纽约每月出生人口数量是在夏季有峰值、 冬季有低谷的时间序列。

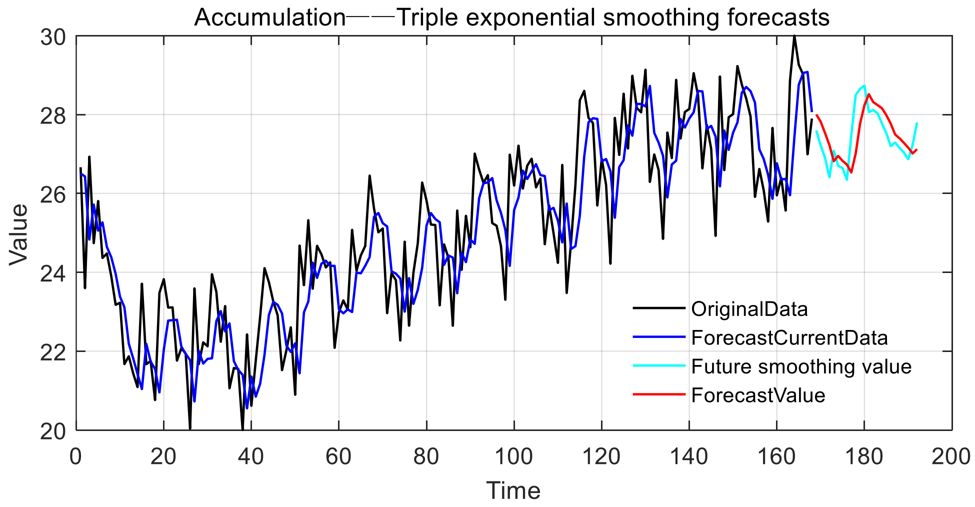
year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1946	26.663	23.598	26.931	24.74	25.806	24.364	24.477	23.901	23.175	23.227	21.672	21.87
1947	21.439	21.089	23.709	21.669	21.752	20.761	23.479	23.824	23.105	23.11	21.759	22.073
1948	21.937	20.035	23.59	21.672	22.222	22.123	23.95	23.504	22.238	23.142	21.059	21.573
1949	21.548	20	22.424	20.615	21.761	22.874	24.104	23.748	23.262	22.907	21.519	22.025
1950	22.604	20.894	24.677	23.673	25.32	23.583	24.671	24.454	24.122	24.252	22.084	22.991
1951	23.287	23.049	25.076	24.037	24.43	24.667	26.451	25.618	25.014	25.11	22.964	23.981
1952	23.798	22.27	24.775	22.646	23.988	24.737	26.276	25.816	25.21	25.199	23.162	24.707
1953	24.364	22.644	25.565	24.062	25.431	24.635	27.009	26.606	26.268	26.462	25.246	25.18
1954	24.657	23.304	26.982	26.199	27.21	26.122	26.706	26.878	26.152	26.379	24.712	25.688
1955	24.99	24.239	26.721	23.475	24.767	26.219	28.361	28.599	27.914	27.784	25.693	26.881
1956	26.217	24.218	27.914	26.975	28.527	27.139	28.982	28.169	28.056	29.136	26.291	26.987
1957	26.589	24.848	27.543	26.896	28.878	27.39	28.065	28.141	29.048	28.484	26.634	27.735
1958	27.132	24.924	28.963	26.589	27.931	28.009	29.229	28.759	28.405	27.945	25.912	26.619
1959	26.076	25.286	27.66	25.951	26.398	25.565	28.865	30	29.261	29.012	26.992	27.897

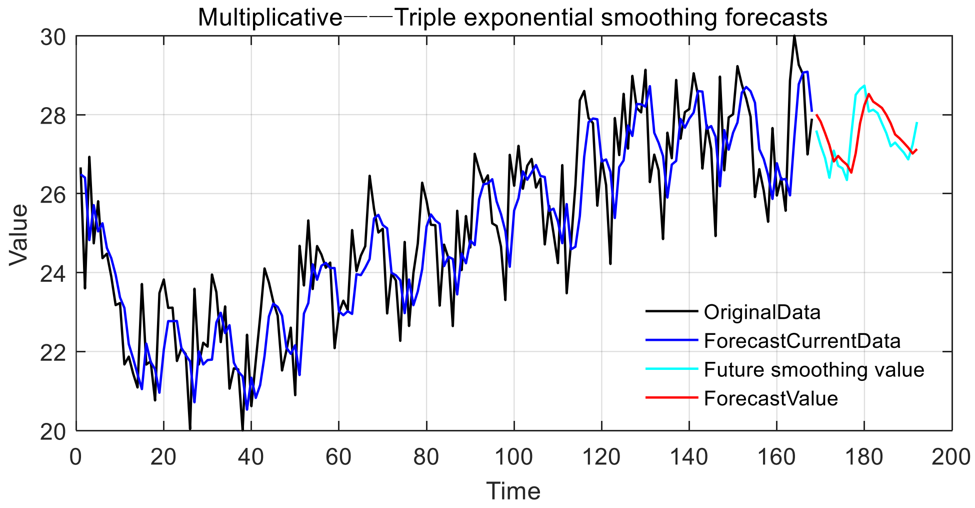
7. 案例分析

```
>> bt = xlsread('births.xlsx');
>> bt(:,1) = [];
>> bt = bt';
>> X = bt(:);
>> esf =  TriExpSF_model(X,12,24)
```

```
esf =
  包含以下字段的 struct:

            AlphaBetaGamma: [0.1000 0.1000 0.4000]
                         S: [2×193 double]
                         T: [2×193 double]
                         P: [2×193 double]
         Accumulation_preY: [1×193 double]
                 SqMErrorA: 1.3091
                    ErrorA: [1×168 double]
       Multiplicative_preY: [1×193 double]
                 SqMErrorM: 1.3096
                    ErrorM: [1×168 double]
```



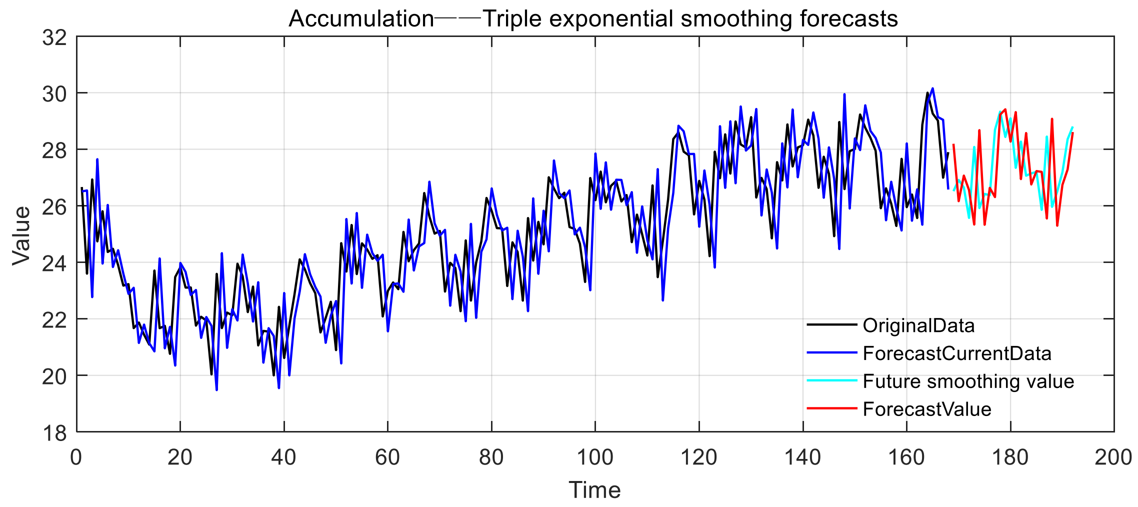


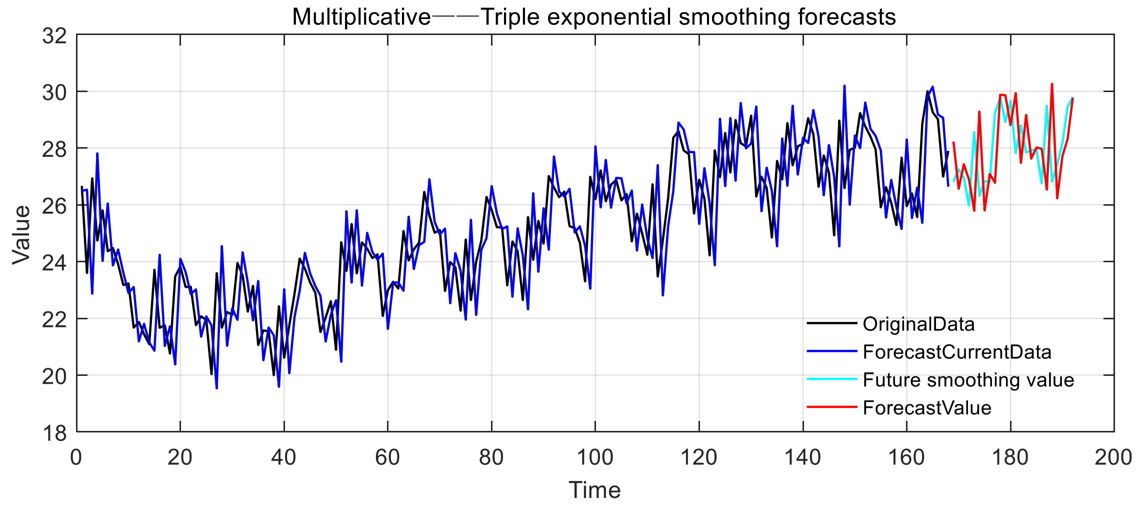
7. 案例分析


```
>> bt = xlsread('births.xlsx');
>> bt(:,1) = [];
>> bt = bt';
>> X = bt(:);
>> esf =  TriExpSF_model(X,12,24,0.3,0.1,0.9)
```

```
esf =
  包含以下字段的 struct:

          AlphaBetaGamma: [0.3000 0.1000 0.9000]
                       S: [2×193 double]
                       T: [2×193 double]
                       P: [2×193 double]
      Accumulation_preY: [1×193 double]
               SqMErrorA: 1.7259
                  ErrorA: [1×168 double]
    Multiplicative_preY: [1×193 double]
               SqMErrorM: 1.7323
                  ErrorM: [1×168 double]
```

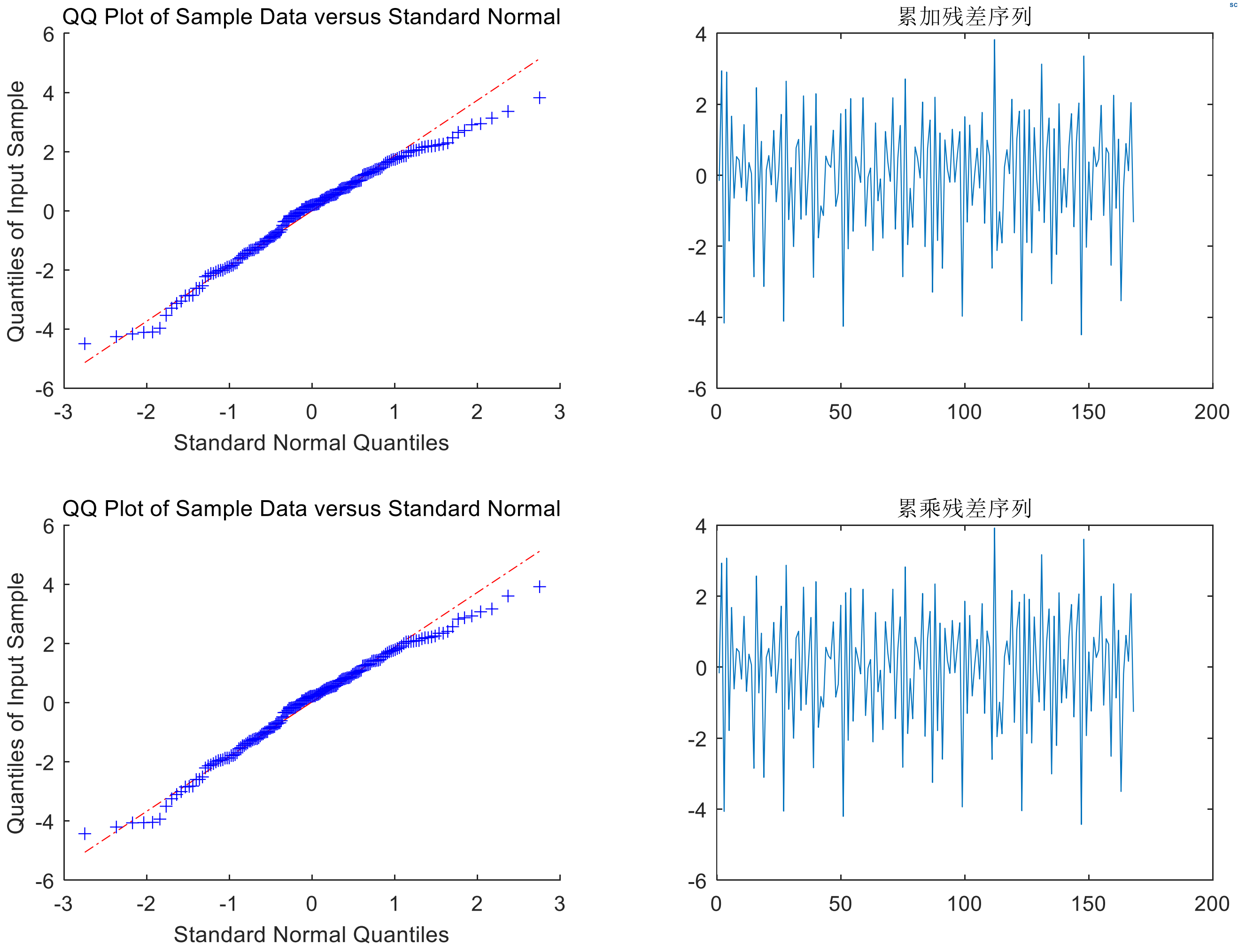




7. 案例分析

```
subplot(2,2,1); qqplot(esf.ErrorA)
subplot(2,2,2); plot(esf.ErrorA)
title('累加残差序列')
subplot(2,2,3); qqplot(esf.ErrorM)
subplot(2,2,4); plot(esf.ErrorM)
title('累乘残差序列')
>> [h,pValue,stat,cValue] = lbqtest(esf.ErrorA)
h = logical  1
pValue = 0
stat =  436.4591
cValue = 31.4104
```

从残差图可以看出，残差基本服从正态分布，且残差序列在零值附件波动，方差大致为常数，已无价值信息，独立。



检验残差存在自相关性，需要ARMA或ARIMA模型进行分析。


01
IDEA
02
LEARNING
03
CREATIVE
INSPIRATION
EDUCATION
BUSINESS
PHOTO BOOK

感谢聆听