第 5 章 MATLAB 数据分析与多项式计算

Lecturer: 白煌

杭州师范大学
信息科学与技术学院

2022.11.18

本章要点

- MATLAB 数据统计处理
- MATLAB 数据插值
- MATLAB 曲线拟合
- MATLAB 多项式计算

目录

5.2.1 一维数据插值

在 MATLAB 中，实现插值的函数是 interp1，其调用格式为

Y1=interp1(X,Y,X1,method)

函数根据 X、Y 的值，计算函数在 X1 处的值。X、Y 是两个等长的已知向量，分别描述采样点和样本值，X1 是一个向量或标量，描述欲插值的点，Y1 是一个与 X1 等长的插值结果。method 是插值方法，允许的取值有线性插值 'linear'、最近点插值 'nearest'、分段 3 次埃尔米特插值 'pchip'、3 次样条插值 'spline'。

5.2.1 一维数据插值

在 MATLAB 中，实现插值的函数是 interp1，其调用格式为

Y1=interp1(X,Y,X1,method)

函数根据 X、Y 的值，计算函数在 X1 处的值。X、Y 是两个等长的已知向量，分别描述采样点和样本值，X1 是一个向量或标量，描述欲插值的点，Y1 是一个与 X1 等长的插值结果。method 是插值方法，允许的取值有线性插值 'linear'、最近点插值 'nearest'、分段 3 次埃尔米特插值 'pchip'、3 次样条插值 'spline'。

MATLAB 中有专门的 3 次埃尔米特插值函数 pchip(X,Y,X1) 和 3 次样条插值函数 spline(X,Y,X1)，其功能及使用方法与函数 interp1(X,Y,X1,'pchip') 和 interp1(X,Y,X1,'spline') 相同。

5.2.1 一维数据插值

例 5-10：用不同的插值方法计算 $\sin x$ 在 $\pi/2$ 点的值。

5.2.1 一维数据插值

例 5-10：用不同的插值方法计算 $\sin x$ 在 $\pi/2$ 点的值。

```
>> X=0:0.2:pi; Y=sin(X);            % 给出 X、Y
>> interp1(X,Y,pi/2)                % 用默认方法计算 sin(π/2)
>> interp1(X,Y,pi/2,'nearest')      % 用最近点插值方法计算
>> interp1(X,Y,pi/2,'linear')       % 用线性插值方法计算
>> interp1(X,Y,pi/2,'pchip')        % 用 3 次 Hermite 插值方法计算
>> interp1(X,Y,pi/2,'spline')       % 用 3 次样条插值方法计算
```

5.2.1 一维数据插值

例 5-11：某观测站测得某日 6:00 时至 18:00 时之间每隔两小时的室内外温度（°C），用 3 次样条插值分别求得该日室内外 6:30 时至 17:30 时之间每隔两小时各点的近似温度（°C）。

5.2.1 一维数据插值

例 5-11：某观测站测得某日 6:00 时至 18:00 时之间每隔两小时的室内外温度（°C），用 3 次样条插值分别求得该日室内外 6:30 时至 17:30 时之间每隔两小时各点的近似温度（°C）。

设时间变量 h 为一行向量，温度变量 t 为一个两列矩阵，其中第一列存放室内温度，第二列存放室外温度。命令如下

```
>> h=6:2:18;
>> t=[18,20,22,25,30,28,24;15,19,24,28,34,32,30]';
>> XI=6.5:2:17.5
>> YI=interp1(h,t,XI,'spline')    % 用 3 次样条插值计算
```

5.2.2 二维数据插值

在 MATLAB 中，提供了解决二维插值问题的函数 interp2，其调用格式为

Z1=interp2(X,Y,Z,X1,Y1,method)

其中，X、Y 是两个向量，分别描述两个参数的采样点，Z 是与参数采样点对应的函数值，X1、Y1 是两个向量或标量，描述欲插值的点。Z1 是根据相应的插值方法得到的插值结果。method 的取值与一维插值函数相同，但不支持 'pchip'。X、Y、Z 也可以是矩阵形式。

5.2.2 二维数据插值

例 5-12：设 $z = x^2 + y^2$，对 z 函数在 [0,1]×[0,2] 区域内进行插值。

5.2.2 二维数据插值

例 5-12：设 $z = x^2 + y^2$，对 z 函数在 [0,1]×[0,2] 区域内进行插值。

```
>> x=0:0.1:1; y=0:0.2:2;
>> [X,Y]=meshgrid(x,y);                % 产生自变量网格坐标
>> Z=X.^2+Y.^2;                         % 求对应的函数值
>> interp2(x,y,Z,0.5,0.5)               % 在 (0.5,0.5) 点插值
>> interp2(x,y,Z,[0.5 0.6],0.4)         % 在 (0.5,0.4) 和 (0.6,0.4) 插值
>> interp2(x,y,Z,[0.5 0.6],[0.4 0.5])   % 在 (0.5,0.4) 和 (0.6,0.5) 插值
>> interp2(x,y,Z,[0.5 0.6]',[0.4 0.5])
>> interp2(x,y,Z,[0.5 0.6]',[0.4 0.5],'spline')
```

5.2.2 二维数据插值

例 5-13：某实验对一根长 10 米的钢轨进行热源的温度传播测试。用 x 表示测量点 0:2.5:10（米），用 h 表示测量时间 0:30:60（秒），用 T 表示测试所得各点的温度（°C）。试用线性插值求出在一分钟内每隔 20 秒、钢轨每隔 1 米处的温度 TI。

5.2.2 二维数据插值

例 5-13：某实验对一根长 10 米的钢轨进行热源的温度传播测试。用 x 表示测量点 0:2.5:10（米），用 h 表示测量时间 0:30:60（秒），用 T 表示测试所得各点的温度（°C）。试用线性插值求出在一分钟内每隔 20 秒、钢轨每隔 1 米处的温度 TI。

```
>> x=0:2.5:10;
>> h=[0:30:60]';
>> T=[95,14,0,0,0;88,48,32,12,6;67,64,54,48,41];
>> xi=[0:10];
>> hi=[0:20:60]';
>> TI=interp2(x,h,T,xi,hi)
```

5.3.1 曲线拟合原理

MATLAB 曲线拟合的最优标准是采用常见的最小二乘原理，所构造的 g(x) 是一个次数小于插值节点个数的多项式。

5.3.2 曲线拟合的实现

采用最小二乘法进行曲线拟合时，实际上是求一个系数向量，该系数向量是一个多项式的系数。在 MATLAB 中，用 polyfit 函数来求得最小二乘拟合多项式的系数，再用 polyval 函数按所得的多项式计算所给出的点上的函数近似值。polyfit 函数的调用格式为

[P,S]=polyfit(X,Y,m)

函数根据采样点 X 和采样点函数值 Y，产生一个 m 次多项式 P 及其在采样点的误差向量 S。其中 X、Y 是两个等长的向量，P 是一个长度为 m+1 的向量，P 的元素为多项式系数。

polyval 函数的功能是按多项式的系数计算 x 点多项式的值，将在 5.4.3 节中详细介绍。

5.3.2 曲线拟合的实现

例 5-14：已知数据表 [t,y]，试求 2 次拟合多项式 p(t)，然后求 ti=1，1.5，2，2.5，$\cdots$，9.5，10 各点的函数近似值。

5.3.2 曲线拟合的实现

例 5-14：已知数据表 [t,y]，试求 2 次拟合多项式 p(t)，然后求 ti=1，1.5，2，2.5，···，9.5，10 各点的函数近似值。

```
>> t=1:10;
>> y=[9.6,4.1,1.3,0.4,0.05,0.1,0.7,1.8,3.8,9.0];
>> p=polyfit(t,y,2)      % 计算 2 次拟合多项式的系数
>> ti=1:0.5:10;
>> yi=polyval(p,ti)
>> plot(t,y,':o',ti,yi,'-*')
```

5.4.1 多项式的四则运算

1. 多项式的加减运算

多项式的加减运算就是其所对应的系数向量的加减运算。

5.4.1 多项式的四则运算

2. 多项式乘法运算

函数 conv(P1,P2) 用于求多项式 P1 和 P2 的乘积。这里，P1、P2 是两个多项式系数向量。

5.4.1 多项式的四则运算

2．多项式乘法运算

函数 conv(P1,P2) 用于求多项式 P1 和 P2 的乘积。这里，P1、P2 是两个多项式系数向量。

例 5-15：求多项式 x^4+8x^3-10 与多项式 $2x^2-x+3$ 的乘积。

5.4.1 多项式的四则运算

2．多项式乘法运算

函数 conv(P1,P2) 用于求多项式 P1 和 P2 的乘积。这里，P1、P2 是两个多项式系数向量。

例 5-15：求多项式 x^4+8x^3-10 与多项式 $2x^2-x+3$ 的乘积。

```
>> A=[1,8,0,0,-10];
>> B=[2,-1,3];
>> C=conv(A,B)
```

5.4.1 多项式的四则运算

3. 多项式除法

5.4.1 多项式的四则运算

3．多项式除法

- 函数 [Q,r]=deconv(P1,P2) 用于对多项式 P1 和 P2 作除法运算。其中 Q 返回多项式 P1 除以 P2 的商式，r 返回 P1 除以 P2 的余式。这里，Q 和 r 仍是多项式系数向量。

5.4.1 多项式的四则运算

3. 多项式除法

- 函数 [Q,r]=deconv(P1,P2) 用于对多项式 P1 和 P2 作除法运算。其中 Q 返回多项式 P1 除以 P2 的商式，r 返回 P1 除以 P2 的余式。这里，Q 和 r 仍是多项式系数向量。
- deconv 是 conv 的逆函数，即有 P1=conv(P2,Q)+r。

5.4.1 多项式的四则运算

3．多项式除法

- 函数 [Q,r]=deconv(P1,P2) 用于对多项式 P1 和 P2 作除法运算。其中 Q 返回多项式 P1 除以 P2 的商式，r 返回 P1 除以 P2 的余式。这里，Q 和 r 仍是多项式系数向量。
- deconv 是 conv 的逆函数，即有 P1=conv(P2,Q)+r。

例 5-16：求多项式 x^4+8x^3-10 除以多项式 $2x^2-x+3$ 的结果。

5.4.1 多项式的四则运算

3. 多项式除法

- 函数 [Q,r]=deconv(P1,P2) 用于对多项式 P1 和 P2 作除法运算。其中 Q 返回多项式 P1 除以 P2 的商式，r 返回 P1 除以 P2 的余式。这里，Q 和 r 仍是多项式系数向量。
- deconv 是 conv 的逆函数，即有 P1=conv(P2,Q)+r。

例 5-16：求多项式 x^4+8x^3-10 除以多项式 $2x^2-x+3$ 的结果。

```
>> A=[1,8,0,0,-10];
>> B=[2,-1,3];
>> [P,r]=deconv(A,B)
```

5.4.2 多项式的导函数

对多项式求导数的函数是:

- p=polyder(P):求多项式 P 的导函数。
- p=polyder(P,Q):求 P·Q 的导函数。
- [p,q]=polyder(P,Q):求 P/Q 的导函数,导函数的分子存入 p,分母存入 q。

上述函数中,参数 P、Q 是多项式的向量表示,结果 p、q 也是多项式的向量表示。

5.4.2 多项式的导函数

例 5-17：求有理分式的导数。

$$f(x) = \frac{1}{x^2+5}$$

5.4.2 多项式的导函数

例 5-17：求有理分式的导数。

$$f(x) = \frac{1}{x^2 + 5}$$

```
>> P=[1];
>> Q=[1,0,5];
>> [p,q]=polyder(P,Q)
```

5.4.3 多项式的求值

MATLAB 提供了两种求多项式值的函数：polyval 与 polyvalm，它们的输入参数均为多项式系数向量 P 和自变量 x。两者的区别在于前者是代数多项式求值，而后者是矩阵多项式求值。

5.4.3 多项式的求值

1．代数多项式求值

polyval 函数用来求代数多项式的值，其调用格式为

Y=polyval(P,x)

若 x 为一数值，则求多项式在该点的值；若 x 为向量或矩阵，则对向量或矩阵中的每个元素求其多项式的值。

5.4.3 多项式的求值

例 5-18： 已知多项式 $x^4 + 8x^3 - 10$，分别取 $x = 1.2$ 和一个 2×3 矩阵为自变量计算该多项式的值。

5.4.3 多项式的求值

例 5-18：已知多项式 x^4+8x^3-10，分别取 $x=1.2$ 和一个 2×3 矩阵为自变量计算该多项式的值。

```
>> A=[1,8,0,0,-10];                % 4 次多项式系数
>> x=1.2;                          % 取自变量为一数值
>> y1=polyval(A,x)
>> x=[-1,1.2,-1.4;2,-1.8,1.6]      % 给出一个矩阵 x
>> y2=polyval(A,x)   % 分别计算各元素为自变量的多项式之值
```

5.4.3 多项式的求值

2．矩阵多项式求值

polyvalm 函数用来求矩阵多项式的值，其调用格式与 polyval 相同，但含义不同。polyvalm 函数要求 x 为方阵，它以方阵为自变量求多项式的值。设 A 为方阵，P 代表多项式 $x^3 - 5x^2 + 8$，那么

- polyvalm(P,A) 的含义是
  A*A*A-5*A*A+8*eye(size(A))
- polyval(P,A) 的含义是
  A.*A.*A-5*A.*A+8*ones(size(A))

5.4.3 多项式的求值

例 5-19：仍以多项式 x^4+8x^3-10 为例，取一个 2×2 矩阵为自变量，分别用 polyval 和 polyvalm 计算该多项式的值。

5.4.3 多项式的求值

例 5-19：仍以多项式 x^4+8x^3-10 为例，取一个 2×2 矩阵为自变量，分别用 polyval 和 polyvalm 计算该多项式的值。

```
>> A=[1,8,0,0,-10];       % 多项式系数
>> x=[-1,1.2;2,-1.8]      % 给出一个矩阵 x
>> y1=polyval(A,x)        % 计算代数多项式的值
>> y2=polyvalm(A,x)       % 计算矩阵多项式的值
```

5.4.4 多项式求根

n 次多项式具有 n 个根，当然这些根可能是实根，也可能含有若干对共轭复根。MATLAB 提供的 roots 函数用于求多项式的全部根，其调用格式为

x=roots(P)

其中，P 为多项式的系数向量，求得的根赋给向量 x，即 x(1)、x(2)、···、x(n) 分别代表多项式的 n 个根。

5.4.4 多项式求根

例 5-20：求多项式 $x^4 + 8x^3 - 10$ 的根。

5.4.4 多项式求根

例 5-20：求多项式 $x^4 + 8x^3 - 10$ 的根。

```
>> A=[1,8,0,0,-10];
>> x=roots(A)
```

5.4.4 多项式求根

若已知多项式的全部根，则可以用 poly 函数建立起该多项式，其调用格式为

P=poly(x)

若 x 为具有 n 个元素的向量，则 poly(x) 建立以 x 为其根的多项式，且将该多项式的系数赋给向量 P。

5.4.4 多项式求根

例 5-21： 已知

$$f(x) = 3x^5 + 4x^3 - 5x^2 - 7.2x + 5$$

1. 计算 f(x)=0 的全部根。
2. 由方程 f(x)=0 的根构造一个多项式 g(x)，并与 f(x) 进行对比。

5.4.4 多项式求根

例 5-21： 已知

$$f(x) = 3x^5 + 4x^3 - 5x^2 - 7.2x + 5$$

1. 计算 f(x)=0 的全部根。
2. 由方程 f(x)=0 的根构造一个多项式 g(x)，并与 f(x) 进行对比。

```
>> P=[3,0,4,-5,-7.2,5];
>> X=roots(P)       % 求方程 f(x)=0 的根
>> G=poly(X)        % 求多项式 g(x)
```