# 第 15 章　预测方法习题解答

15.1　**解**　**（一）插值法**

**要估计在任意时刻（包括水泵灌水期间）** $t$ **居民的用水速度和日总用水量，分如下三步。**

**（1）水塔中水的体积的计算**

**计算水的流量，首先需要计算出水塔中水的体积**

$$V=\frac{p}{4}D^2h\text{，}$$

**式中，** $D$ **为水塔的直径，** $h$ **为水塔中的水位高度。**

**（2）水塔中水流速度的估计**

**居民的用水速度就是水塔中的水流速度，水流速度应该是水塔中水的体积对时间的导数，但由于没有每一时刻水体积的具体数学表达式，只能用差商近似导数。**

**由于在两个时段，水泵向水塔供水，无法确定水位的高度，因此在计算水塔中水流速度时要分三段计算。第一段从 0h 到** 8.97h**，第二时段，从 10.95h 到** 20.84h**，第三段，从** 23.88h **到** 25.91h。

**上面计算仅给出流速的离散值，如果需要得到流速的连续型曲线，需要作插值处理，这里可以使用三次样条插值。**

**（3）日总用水量的计算**

**日用水是对水流速度作积分，其积分区间是**[0,24]**，可以采用数值积分的方法计算。**

**用** Matlab **软件计算时，首先把原始数据粘贴到纯文本文件** data151 **中，并且把“-”替换为数值-1。计算的 Matlab 程序如下**

```
clc, clear
a=load('data151.txt');
t0=a([1:2:end],:); t0=t0'; t0=t0(:); %提出时间数据，并展开成列向量
h0=a([2:2:end],:); h0=h0'; h0=h0(:); %提出高度数据，并展开成列向量
D=17.4;
V=pi/4*D^2*h0; %计算各时刻的体积
dv=gradient(V,t0); %计算各时刻的数值导数（导数近似值）
no1=find(h0==-1) %找出原始无效数据的地址
no2=[no1(1)-1:no1(2)+1,no1(3)-1:no1(4)+1] %找出导数数据的无效地址
t=t0; t(no2)=[]; %删除导数数据无效地址对应的时间
dv2=-dv; dv2(no2)=[]; %给出各时刻的流速
plot(t,dv2,'*') %画出流速的散点图
pp=csape(t,dv2); %对流速进行插值
tt=0:0.1:t(end); %给出插值点
fdv=ppval(pp,tt); %计算各插值点的流速值
hold on, plot(tt,fdv) %画出插值曲线
I=trapz(tt(1:241),fdv(1:241)) %计算 24 小时内总流量的数值积分
```

**画出的流速图见图** 15.1。**求得的日用水总量为** 1248.3$m^3$。

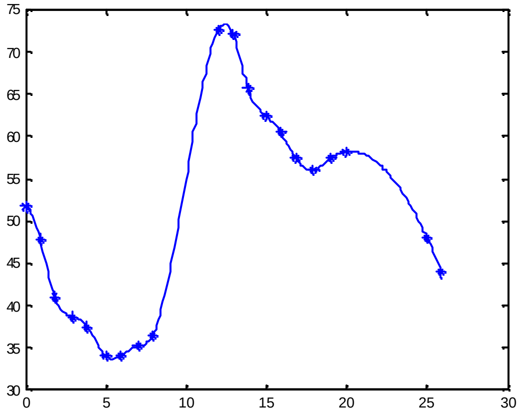

图 15.1 流速的散点图和样条插值函数图

（二）拟合法

要估计在任意时刻（包括水泵灌水期间）$t$ 居民的用水速度和日总用水量，分如下三步。

（1）水塔中水的体积的计算

计算水的流量，首先需要计算出水塔中水的体积

$$V=\frac{p}{4}D^2h,$$

式中，$D$ 为水塔的直径，$h$ 为水塔中的水位高度。

（2）水塔中水流速度的估计

居民的用水速度就是水塔中的水流速度，水流速度应该是水塔中水的体积对时间的导数，但由于没有每一时刻水体积的具体数学表达式，只能用差商近似导数。

由于在两个时段，水泵向水塔供水，无法确定水位的高度，因此在计算水塔中水流速度时要分三段计算。第一段从 0h 到 8.97h，第二时段，从 10.95h 到 20.84h，第三段，从 23.88h 到 25.91h。

上面计算仅给出流速的离散值，流速的散点图见图 15.2 中的“*”点。如果需要得到流速的连续型曲线，可以拟合多项式曲线，原始数据总共有 28 个观测值，其中 4 个无效数据。图 15.2 中总共有 20 个数据点，这里我们分三段进行三次多项式拟合，应用前 6 个数据点拟合三次多项式，即在时间区间[0, 4.98]上拟合三次多项式；应用第 6 个数据点到第 10 个数据点，即在时间区间[4.98, 12.03]，拟合第二个三次多项式；应用第 10 个数据点到第 20 个数据点，总共 11 数据点，即在时间区间[12.03,25.91]，拟合第三个三次多项式。拟合得到的分段三次多项式曲线见图 15.2。

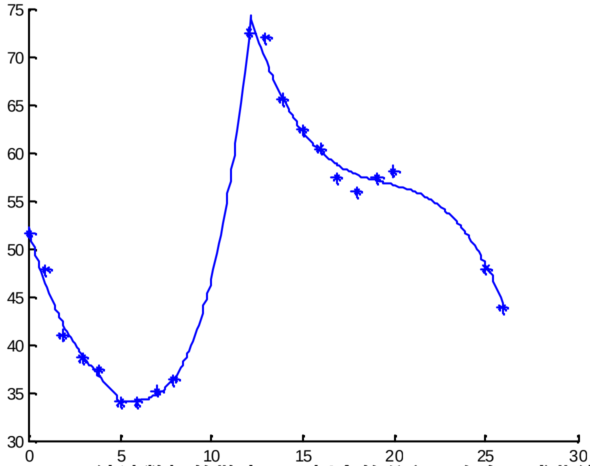


图 15.2　流速数据的散点图及拟合的分段三次多项式曲线

（3）日总用水量的计算

日用水是对水流速度作积分，其积分区间是[0,24]，可以采用数值积分的方法计算。这里求得的日总用水量为 1221.8 $m^3$。

计算的 Matlab 程序如下

```
clc, clear
a=load('data151.txt');
t0=a([1:2:end],:); t0=t0'; t0=t0(:); %提出时间数据，并展开成列向量
h0=a([2:2:end],:); h0=h0'; h0=h0(:); %提出高度数据，并展开成列向量
D=17.4;
V=pi/4*D^2*h0; %计算各时刻的体积
dv=gradient(V,t0); %计算各时刻的数值导数（导数近似值）
no1=find(h0==-1) %找出原始无效数据的地址
no2=[no1(1)-1:no1(2)+1,no1(3)-1:no1(4)+1] %找出导数数据的无效地址
t=t0; t(no2)=[]; %删除导数数据无效地址对应的时间
dv2=-dv; dv2(no2)=[]; %给出各时刻的流速
hold on, plot(t,dv2,'*') %画出流速的散点图
a1=polyfit(t(1:6),dv2(1:6),3); %拟合第一个多项式的系数
a2=polyfit(t(6:10),dv2(6:10),3); %拟合第二个多项式的系数
a3=polyfit(t(10:20),dv2(10:20),3); %拟合第三个多项式的系数
dvf1=polyval(a1,[t(1):0.1:t(6)]); %计算第一个多项式的函数值
dvf2=polyval(a2,[t(6):0.1:t(10)]); %计算第二个多项式的函数值
dvf3=polyval(a3,[t(10):0.1:t(end)]); %计算第三个多项式的函数值
tt=t(1):0.1:t(end); dvf=[dvf1,dvf2,dvf3];
plot(tt,dvf) %画出拟合的三个分段多项式曲线
I=trapz(tt(1:241),dvf(1:241)) %计算 24 小时内总流量的数值积分
```

15.2　解　微分方程对应的差分方程为

$$\begin{cases} x(k+1)-x(k)=-cx(k)y(k)-ax(k) \\ y(k+1)-y(k)=-dx(k)y(k)-by(k) \end{cases}, \quad k=1,2,\cdots,19,$$

可以改写成下列格式

$$\begin{bmatrix} -x(k)y(k) & -x(k) & 0 & 0 \\ 0 & 0 & -x(k)y(k) & -y(k) \end{bmatrix} \begin{bmatrix} c \\ a \\ d \\ b \end{bmatrix} = \begin{bmatrix} x(k+1)-x(k) \\ y(k+1)-y(k) \end{bmatrix}, k=1,2,\cdots,19 .$$

**上述所有的差分方程可以写成矩阵格式**

$$\begin{bmatrix} -x(1)y(1) & -x(1) & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ -x(19)y(19) & -x(19) & 0 & 0 \\ 0 & 0 & -x(1)y(1) & -y(1) \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & -x(19)y(19) & -y(19) \end{bmatrix} \begin{bmatrix} c \\ a \\ d \\ b \end{bmatrix} = \begin{bmatrix} x(2)-x(1) \\ \vdots \\ x(20)-x(19) \\ y(2)-y(1) \\ \vdots \\ y(20)-y(19) \end{bmatrix}.$$

**利用最小最小二乘法，求得**$c=0$**，**$a=0.0389$**，**$d=0$**，**$b=0.0921$**。**

**拟合的** Matlab **程序如下**

```
clc, clear
a=[1500	1400	1320	1100	1000	950	880	800	700	680
1200	1120	1080	1060	980	930	870	790	680	670
11	12	13	14	15	16	17	18	19	20
620	600	570	520	500	450	440	420	400	390
600	590	560	500	480	420	400	370	350	330];
x=a([1,4],:); x=x'; x=x(:);
y=a([2,5],:); y=y'; y=y(:);
dx=diff(x);   %求一阶向前差分
dy=diff(y);
temp=x(1:end-1).*y(1:end-1);   %构造分块矩阵的子矩阵
a=[-temp -x(1:end-1) zeros(19,2);zeros(19,2),-temp -y(1:end-1)];
b=[dx;dy];
solution=a\b
```