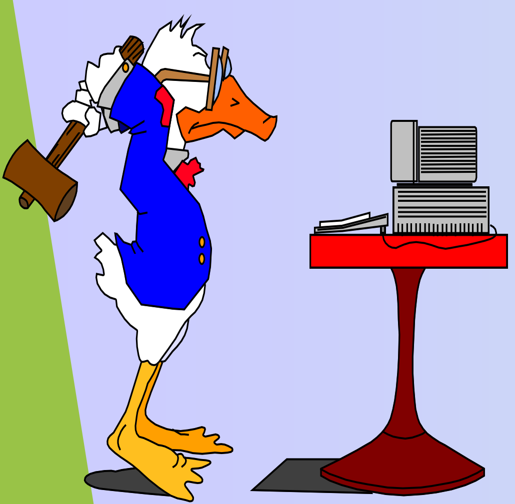
数学建模与数学实验

拟合



应用数学系

陈六新

若$f(x)$是由实验或观测得到的，则其函数通常由函数表$(x_i, f(x_i))\ (i=1,2,\dots,m)$给出. 插值法是找到一个简单且便于计算的公式，利用它可近似地计算给定区间上的函数值.

但有**问题**:

数据误差

(1)高次多项式会出现龙格现象；

(2)用插值条件来确定函数关系不合理.

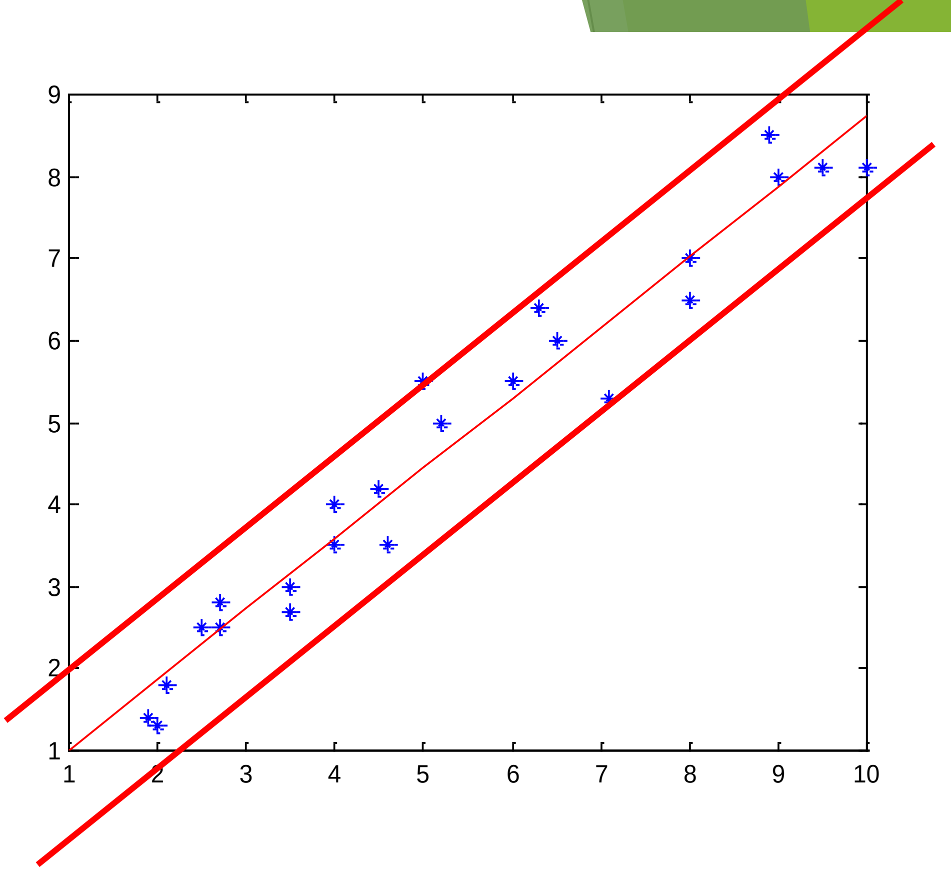
解决办法：曲线拟合的最小二乘法.

1、最小二乘法原理

引例　考察某种纤维的强度与其拉伸倍数的关系,下表是实际测定的24个纤维样品的强度与相应的拉伸倍数的记录:

编　号	拉伸倍数x_i	强　度y_i	编　号	拉伸倍数x_i	强　度y_i
1	1.9	1.4	13	5	5.5
2	2	1.3	14	5.2	5
3	2.1	1.8	15	6	5.5
4	2.5	2.5	16	6.3	6.4
5	2.7	2.8	17	6.5	6
6	2.7	2.5	18	7.1	5.3
7	3.5	3	19	8	6.5
8	3.5	2.7	20	8	7
9	4	4	21	8.9	8.5
10	4	3.5	22	9	8
11	4.5	4.2	23	9.5	8.1
12	4.6	3.5	24	10	8.1

纤维强度随拉伸倍数增加而增加；并且24个点大致分布在一条直线附近.



因此可以认为强度y与拉伸倍数x的主要关系应是线性关系 $y(x)=\beta_0+\beta_1 x$ 其中β_0,β_1为待定参数

我们希望$y(x)=\beta_0+\beta_1 x$与所有的数据点(样本点)(x_i,y_i)越接近越好

必须找到一种度量标准来衡量什么曲线最接近所有数据点.

对所找出的曲线 $y=y(x)$:

令 $\delta_i = y(x_i) - y_i \quad (i=1,2,\ldots,m)$ 残差P.39

一般使用 $\|\delta\|_2^2 = \sum_{i=1}^{m} \delta_i^2 = \sum_{i=1}^{m} (y(x_i) - y_i)^2$

作为衡量 $y(x)$ 与数据点 (x_i, y_i) 偏离程度大小

称为平方误差.

度量标准: $\min \sum_{i=1}^{m} \delta_i^2$. **称在这个要求下的拟合**

拟合问题

为曲线拟合的最小二乘法.

一般地，求一条曲线，使数据点均在离此曲线的上方或下方不远处，所求的曲线称为拟合曲线.

它既能反映数据的总体分布,又不至于出现局部较大的波动,更能反映被逼近函数的特性,使求得的逼近函数与已知函数从总体上来说其偏差按某种方法度量达到最小。

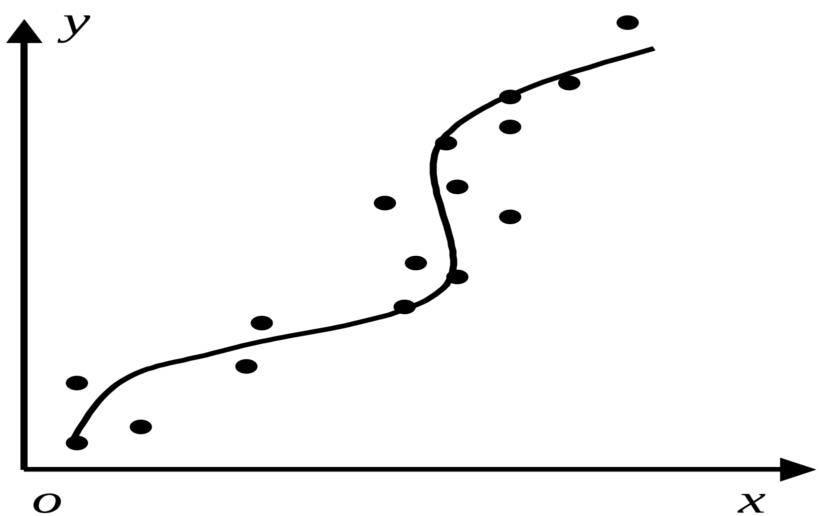


图1 曲线拟合示意图

注:

与函数插值问题不同,曲线拟合不要求曲线通过所有已知点,而是要求得到的近似函数能反映数据的基本关系.在某种意义上,曲线拟合更有实用价值.

在对给出的实验(或观测)数据作曲线拟合时,怎样才算拟合得最好呢? 一般希望各实验(或观测)数据与拟合曲线的误差为最小,即距离为最小(不同的度量意义).

两种逼近概念:

插值: 在节点处函数值相同

拟合: 在数据点处误差最小

用最小二乘解决实际问题基本步骤如下：

(1) 确定近似函数类,即确定近似函数 $y=\varphi(x)$ 的形式.

这并非单纯的数学问题,与其它各领域的专门知识有关.
数学上,通常根据在坐标纸上所描点的情况来选择 $\varphi(x)$ 的形式.

(2) 求最小二乘解.

求使残差的平方和 $\|\delta\|_2^2=\sum_{i=1}^{m}\delta_i^2$ 达到最小的

$\varphi(x)$ 中的待定参数.

2 最小二乘法的求法

一、多项式拟合

取 $\varphi_j(x)=x^j\ (j=0,1,\cdots,n)$，则

$$\varphi(x)=a_0+a_1x+\cdots+a_nx^n$$

$$\begin{cases} ma_0+a_1\sum_{i=1}^{m}x_i+\cdots+a_n\sum_{i=1}^{m}x_i^n=\sum_{i=1}^{m}y_i \\ a_0\sum_{i=1}^{m}x_i+a_1\sum_{i=1}^{m}x_i^2+\cdots+a_n\sum_{i=1}^{m}x_i^{n+1}=\sum_{i=1}^{m}y_ix_i \\ \cdots\quad\cdots\quad\cdots \\ a_0\sum_{i=1}^{m}x_i^n+a_1\sum_{i=1}^{m}x_i^{n+1}+\cdots+a_n\sum_{i=1}^{m}x_i^{2n}=\sum_{i=1}^{m}y_ix_i^n \end{cases} \quad (3)$$

方程特点？

解出 $a_0,a_1,\cdots,a_n$，就得到拟合公式 $y=a_0+a_1x+\cdots+a_nx^n$.

所以，曲线拟合的最小二乘法要解决的问题，实际上就是求以下超定方程组的最小二乘解的问题。

$$Ra=y \qquad (3)$$

其中 $$R=\begin{bmatrix} r_1(x_1)\cdots r_m(x_1) \\ \cdots\cdots \\ r_1(x_n)\cdots r_m(x_n) \end{bmatrix},\ a=\begin{bmatrix} a_1 \\ \vdots \\ a_m \end{bmatrix},\ y=\begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

定理：当R^TR可逆时，超定方程组（3）存在最小二乘解，且即为方程组

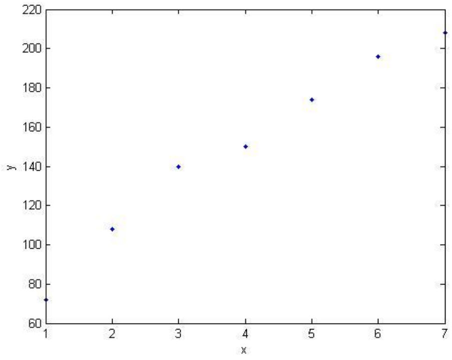
$$R^TRa=R^Ty$$

的解：$a=(R^TR)^{-1}R^Ty$

例 1 已知某单位 2001—2007 年的利润(万元)为

时间	2001	2002	2003	2004	2005	2006	2007
利润	72	108	140	150	174	196	208

试预测 2008 年的利润. (P.40)



解 作出散点图 取拟合函数为 $y=a+bt$,

法方程 $\begin{cases} 7\boldsymbol{a}+28\boldsymbol{b}=1048 \\ 28\boldsymbol{a}+140\boldsymbol{b}=4810 \end{cases}$

$\Rightarrow a=\dfrac{430}{7},\ b=\dfrac{309}{14}$

$y=\dfrac{430}{7}+\dfrac{309}{14}t$

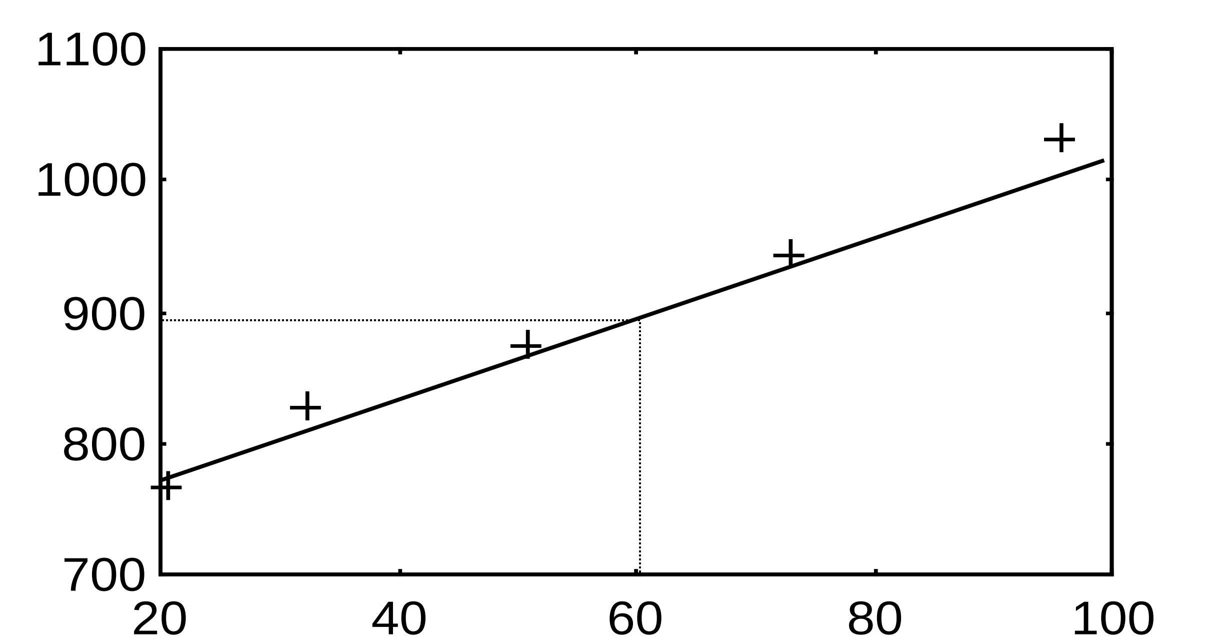
利润为 $y=\dfrac{430}{7}+\dfrac{309}{14}\times 8=\dfrac{1666}{7}=238$（万元）.

拟合问题 例1

已知热敏电阻数据：

温度t(^{0}C)	20.5	32.7	51.0	73.0	95.7
电阻R(Ω)	765	826	873	942	1032

求60^{0}C时的电阻R。



设 R=at+b

a,b为待定系数

拟合问题 例2

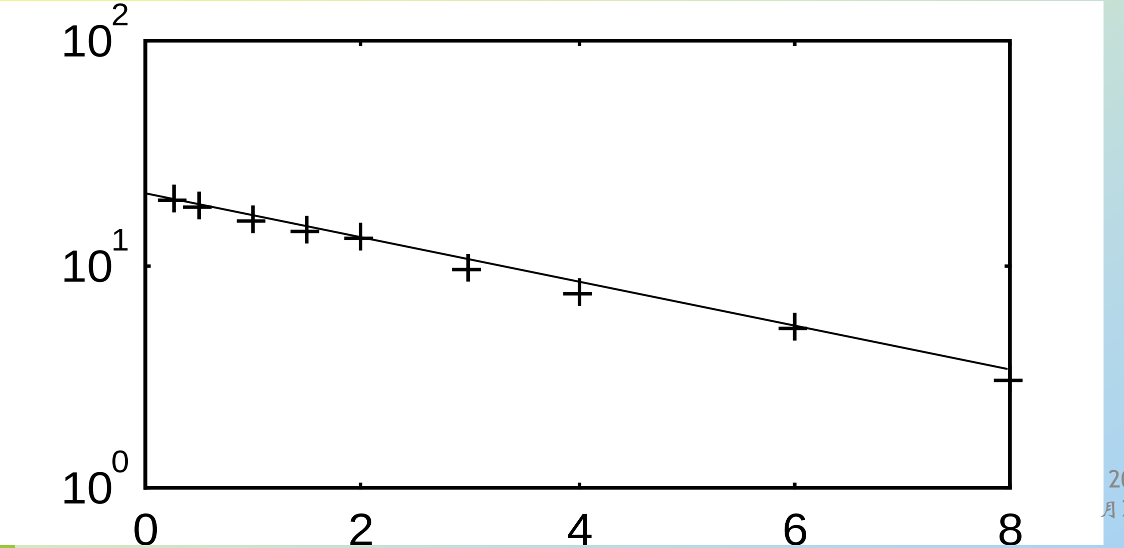
已知一室模型a静脉注射下的血药浓度数据(t=0注射300mg)

t (h)	0.25	0.5	1	1.5	2	3	4	6	8
c (μg/ml)	19.21	18.15	15.36	14.10	12.89	9.32	7.45	5.24	3.01

求血药浓度随时间的变化规律c(t).

作半对数坐标系(semilogy)下的图形

MATLAB(aa1)



$$c(t) = c_0 e^{-kt}$$

c, k为待定系数

拟合与插值的关系

问题：给定一批数据点，需确定满足特定要求的曲线或曲面。

解决方案：

- 若要求所求曲线（面）通过所给所有数据点，**就是插值问题**；
- 若不要求曲线（面）通过所有数据点，而是要求它反映对象整体的变化趋势，这就是**数据拟合**，**又称曲线拟合或曲面拟合**。

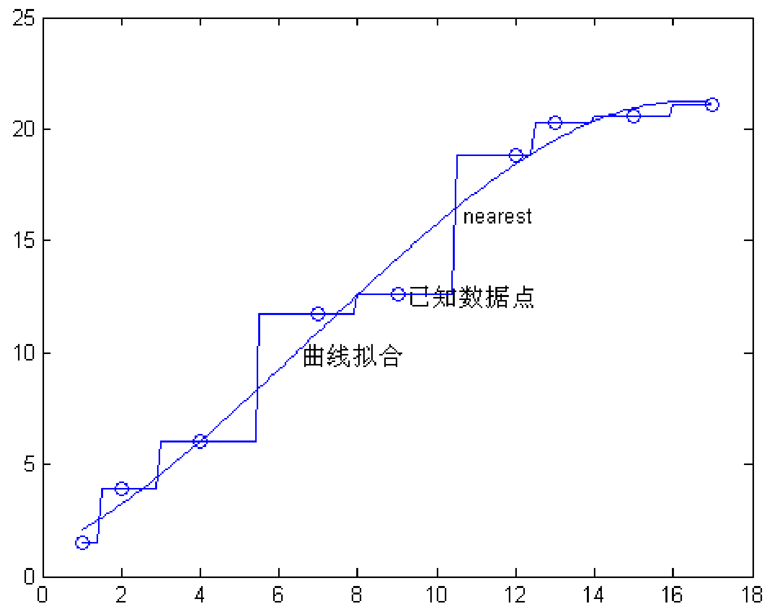
函数插值与曲线拟合都是要根据一组数据构造一个函数作为近似，由于近似的要求不同，二者的数学方法上是完全不同的。

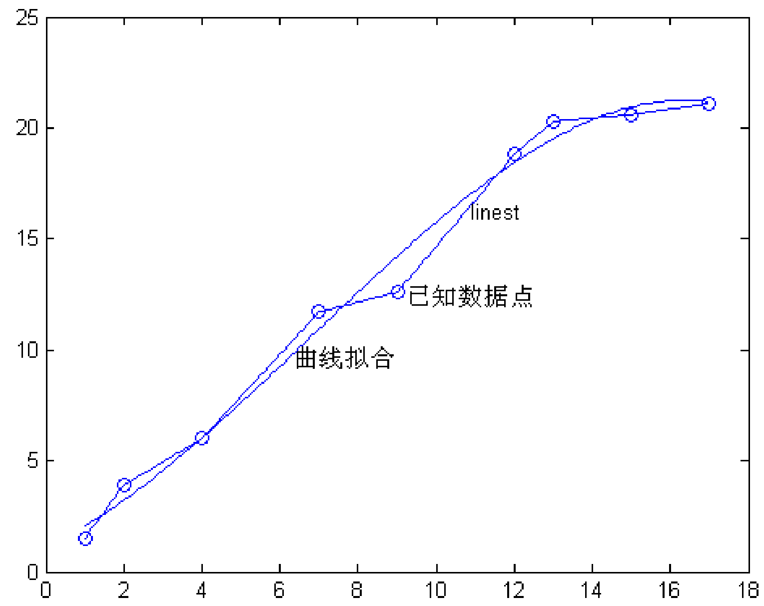
实例：下面数据是某次实验所得，希望得到X和 f之间的关系？

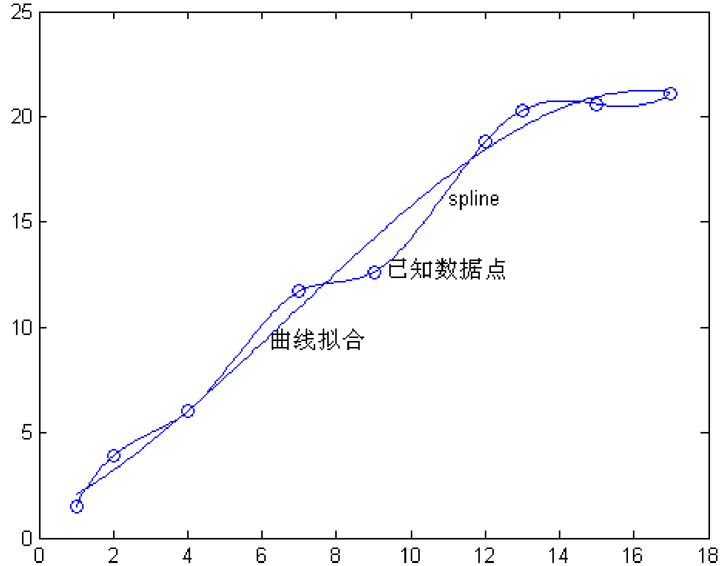
x	1	2	4	7	9	12	13	15	17
f	1.5	3.9	6.6	11.7	15.6	18.8	19.6	20.6	21.1

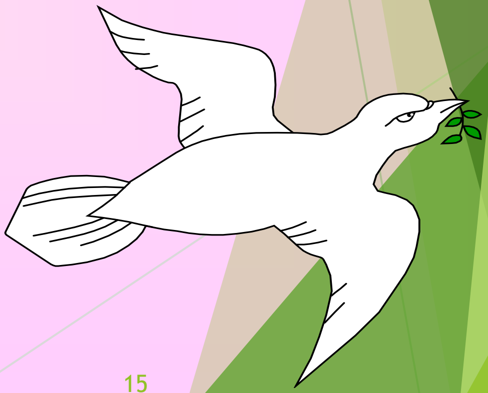
MATLAB(cn)

最临近插值、线性插值、样条插值与曲线拟合结果：







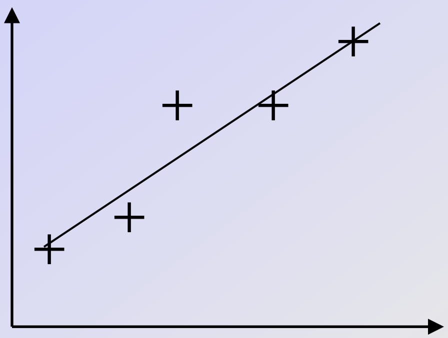


线性最小二乘拟合中函数$\{r_1(x), \cdots r_m(x)\}$的选取

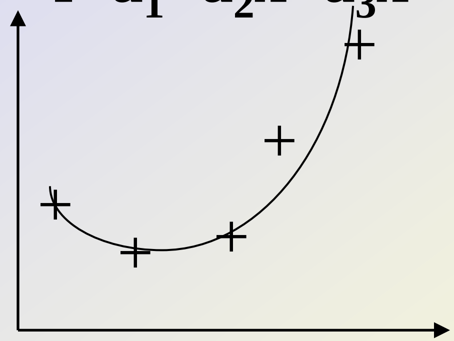
1. 通过机理分析建立数学模型来确定 $f(x)$；

2. 将数据 (x_i, y_i) $i=1, \cdots n$ 作图，通过直观判断确定 $f(x)$：

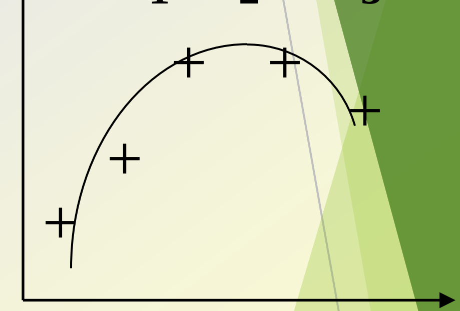
$f=a_1+a_2x$



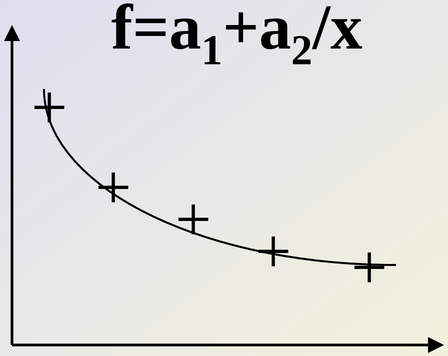
$f=a_1+a_2x+a_3x^2$



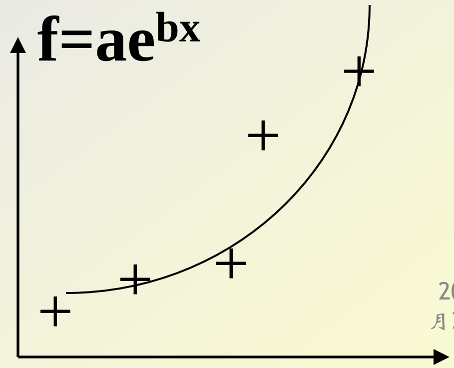
$f=a_1+a_2x+a_3x^2$



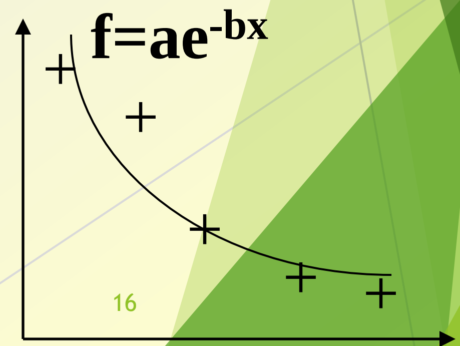
$f=a_1+a_2/x$



$f=ae^{bx}$



$f=ae^{-bx}$



二、用MATLAB解拟合问题

1.线性最小二乘拟合

2.非线性最小二乘拟合

1. 用MATLAB作线性最小二乘拟合

1) 作多项式$f(x)=a_1x^m+ \ldots +a_mx+a_{m+1}$拟合,可利用已有程序:

```
a=polyfit(x,y,m)
```

输出拟合多项式系数

$a=[a_1, \ldots a_m, a_{m+1}]$ (数组))

输入同长度的数组X，Y

拟合多项式次数

2) 对超定方程组 $R_{n\times m}a_{m\times 1}=y_{n\times 1}\ (m<n)$，用 $a=R\backslash y$

可得最小二乘意义下的解。

3)多项式在x处的值y可用以下命令计算:

y=polyval（a，x）

例 对下面一组数据作二次多项式拟合

xi	0.1	0.2	0.4	0.5	0.6	0.7	0.8	0.9	1
yi	1.978	3.28	6.16	7.34	7.66	9.58	9.48	9.30	11.2

即要求 出二次多项式:

$$f(x) = a_1 x^2 + a_2 x + a_3$$

中 的 $A = (a_1, a_2, a_3)$ **使得:**

$$\sum_{i=1}^{11} [f(x_i) - y_i]^2 \quad 最小$$

解法1．用解超定方程的方法

此时 $R=\begin{pmatrix} x_1^2 & x_1 & 1 \\ \cdots & \cdots & \cdots \\ x_{11}^2 & x_{11} & 1 \end{pmatrix}$

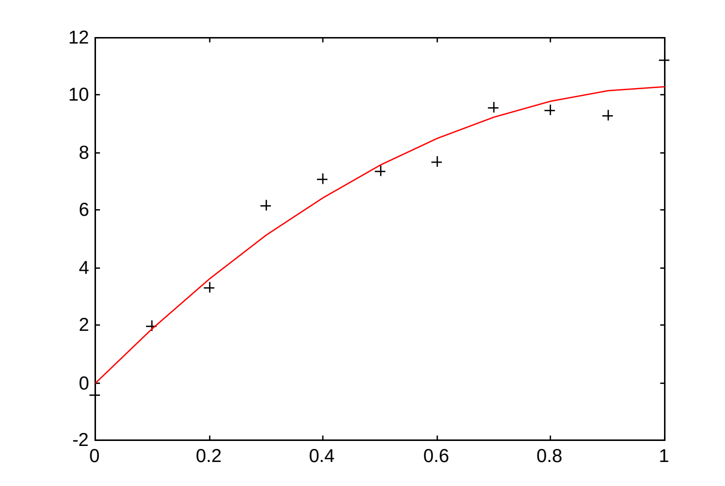
1）输入以下命令：

```
x=0:0.1:1;
y=[-0.447 1.978 3.28 6.16 7.08 7.34 7.66 9.56 9.48 9.30 11.2];
R=[(x.^2)' x' ones(11,1)];
A=R\y'
```

MATLAB(zxec1)

2）计算结果： A = -9.8108 20.1293 -0.0317

$$f(x)=-9.8108x^2+20.1293x-0.0317$$



解法2．用多项式拟合的命令

1）输入以下命令：

```
x=0:0.1:1;
y=[-0.447 1.978 3.28 6.16 7.08 7.34 7.66 9.56 9.48 9.30 11.2];
A=polyfit(x,y,2)
z=polyval(A,x);
plot(x,y,'k+',x,z,'r')     %作出数据点和拟合曲线的图形
```

MATLAB(zxec2)

2）计算结果： A = -9.8108 20.1293 -0.0317

$$f(x) = -9.8108x^2 + 20.1293x - 0.0317$$

2.用MATLAB作非线性最小二乘拟合

Matlab的提供了两个求非线性最小二乘拟合的函数：lsqcurvefit和lsqnonlin。两个命令都要先建立M-文件fun.m，在其中定义函数f(x)，但两者定义f(x)的方式是不同的,可参考例题.

1. lsqcurvefit

已知数据点：　xdata=（$xdata_1$，$xdata_2$，…，$xdata_n$），

ydata=（$ydata_1$，$ydata_2$，…，$ydata_n$）

lsqcurvefit用以求含参量x（向量）的向量值函数

F(x,xdata)=（F（x，$xdata_1$），…，F（x，$xdata_n$））^{T}

中的参变量x(向量),使得

$$\sum_{i=1}^{n}\left(F(x, xdata_i) - ydata_i\right)^2 \text{最小}$$

输入格式为:

(1) x = lsqcurvefit (‘fun’,x0,xdata,ydata) ;

(2) x =lsqcurvefit (‘fun’,x0,xdata,ydata,options);

(3) x = lsqcurvefit (‘fun’,x0,xdata,ydata,options,’grad’);

(4) [x, options] = lsqcurvefit (‘fun’,x0,xdata,ydata,…);

(5) [x, options,funval] = lsqcurvefit (‘fun’,x0,xdata,ydata,…);

(6) [x, options,funval, Jacob] = lsqcurvefit (‘fun’,x0,xdata,ydata,…);

说明： x = lsqcurvefit (‘fun’,x0,xdata,ydata,options);

fun是一个事先建立的定义函数F(x,xdata) 的M-文件, 自变量为x和xdata

迭代初值

已知数据点

选项见无约束优化

2. lsqnonlin

已知数据点：xdata=（$xdata_1$，$xdata_2$，…，$xdata_n$）

ydata=（$ydata_1$，$ydata_2$，…，$ydata_n$）

lsqnonlin用以求含参量x（向量）的向量值函数

$f(x)=(f_1(x),f_2(x),\ldots,f_n(x))^T$ 中的参量x，使得

$$f^T(x)f(x)=f_1(x)^2+f_2(x)^2+\cdots+f_n(x)^2$$

最小。

其中 f_i（x）=f（x，$xdata_i$，$ydata_i$）

$=F(x,xdata_i)-ydata_i$

输入格式为:

1） x=lsqnonlin（‘fun’，x0）；

2） x= lsqnonlin （‘fun’，x0，options）；

3） x= lsqnonlin （‘fun’，x0，options，‘grad’）；

4） [x，options]= lsqnonlin （‘fun’，x0，…）；

5） [x，options，funval]= lsqnonlin （‘fun’，x0，…）；

说明：**x= lsqnonlin （‘fun’，x0，options）；**

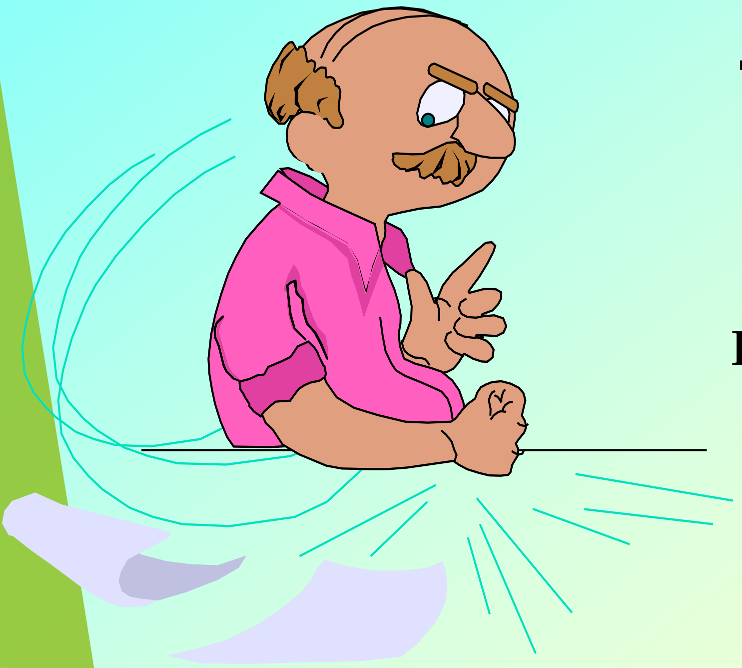
fun是一个事先建立的定义函数*f(x)*的M-文件，自变量为x

迭代初值

选项见无约束优化

例2 用下面一组数据拟合 $c(t)=a+be^{0.0.2kt}$ 中的参数a，b，k

t_j	100	200	300	400	500	600	700	800	900	1000
$c_j \times 10^3$	4.54	4.99	5.35	5.65	5.90	6.10	6.26	6.39	6.50	6.59



该问题即解最优化问题：

$$\min F(a,b,k)=\sum_{j=1}^{10}[a+be^{-0.02kt_j}-c_j]^2$$

解法1. 用命令lsqcurvefit

F(x, tdata)= $(a+be^{-0.02kt_1},\cdots,a+be^{-0.02kt_{10}})^T$, x=(a, b, k)

1）编写M-文件 curvefun1.m

```
function f=curvefun1(x,tdata)
f=x(1)+x(2)*exp(-0.02*x(3)*tdata)
                    %其中 x(1)=a; x(2)=b; x(3)=k;
```

2) 输入命令

```
tdata=100:100:1000
cdata=1e-03*[4.54,4.99,5.35,5.65,5.90,6.10,6.26,6.39,
6.50,6.59];
x0=[0.2,0.05,0.05];
x=lsqcurvefit ('curvefun1',x0,tdata,cdata)
f= curvefun1(x,tdata)
```


MATLAB(fzxec1)

3）运算结果为：

f =0.0043 0.0051 0.0056 0.0059 0.0061

0.0062 0.0062 0.0063 0.0063 0.0063

x = 0.0063 -0.0034 0.2542

4）结论： **a=0.0063, b=-0.0034, k=0.2542**

解法 2　用命令lsqnonlin

f(x)=F(x,tdata,ctada)= $(a+be^{-0.02kt_1}-c_1,\cdots,a+be^{-0.02kt_{10}}-c_1)^T$

x=（a，b，k）

函数curvefun2的自变量是x，cdata和tdata是已知参数，故应将cdata tdata的值写在curvefun2.m中

1）编写M-文件 curvefun2.m

```
function f=curvefun2(x)
tdata=100:100:1000;
cdata=1e-03*[4.54,4.99,5.35,5.65,5.90,
             6.10,6.26,6.39,6.50,6.59];
f=x(1)+x(2)*exp(-0.02*x(3)*tdata)- cdata
```

2）输入命令:

```
x0=[0.2,0.05,0.05];
x=lsqnonlin('curvefun2',x0)
f= curvefun2(x)
```

MATLAB(fzxec2)

3）运算结果为

f =1.0e-003 *(0.2322 -0.1243 -0.2495 -0.2413 -0.1668 -0.0724 0.0241 0.1159 0.2030 0.2792

x =0.0063 -0.0034 0.2542

4）**结论**：**即拟合得**a=0.0063 b=-0.0034 k=0.2542

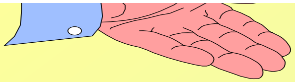
可以看出,两个命令的计算结果是相同的.

三、MATLAB解应用问题实例

1.电阻问题

2.给药方案问题

1.电阻问题



例.由数据

温度t(^{0}C)	20.5	32.7	51.0	73.0	95.7
电阻R(Ω)	765	826	873	942	1032

拟合$R=a_1t+a_2$

方法1.用命令 `polyfit(x,y,m)`

MATLAB(dianzu1)

得到 $a_1=3.3940$, $a_2=702.4918$

方法2.直接用 $a = R \backslash y$

MATLAB(dianzu2)

结果相同。

2.给药方案问题

——拟合问题实例2

一种新药用于临床之前，必须设计给药方案.

药物进入机体后血液输送到全身，在这个过程中不断地被吸收、分布、代谢，最终排出体外，药物在血液中的浓度，即单位体积血液中的药物含量，称为血药浓度。

一室模型：将整个机体看作一个房室，称中心室，室内血药浓度是均匀的。快速静脉注射后，浓度立即上升；然后迅速下降。当浓度太低时，达不到预期的治疗效果；当浓度太高，又可能导致药物中毒或副作用太强。临床上，每种药物有一个最小有效浓度c_1和一个最大有效浓度c_2。设计给药方案时，要使血药浓度 保持在$c_1 \sim c_2$之间。本题设$c_1=10$，$c_2=25(ug/ml)$.

要设计给药方案,必须知道给药后血药浓度随时间变化的规律。从实验和理论两方面着手:

在实验方面,对某人用快速静脉注射方式一次注入该药物300mg后,在一定时刻t(小时)采集血药,测得血药浓度c(ug/ml)如下表:

t (h)	0.25	0.5	1	1.5	2	3	4	6	8
c (μg/ml)	19.21	18.15	15.36	14.10	12.89	9.32	7.45	5.24	3.01

给药方案

问题

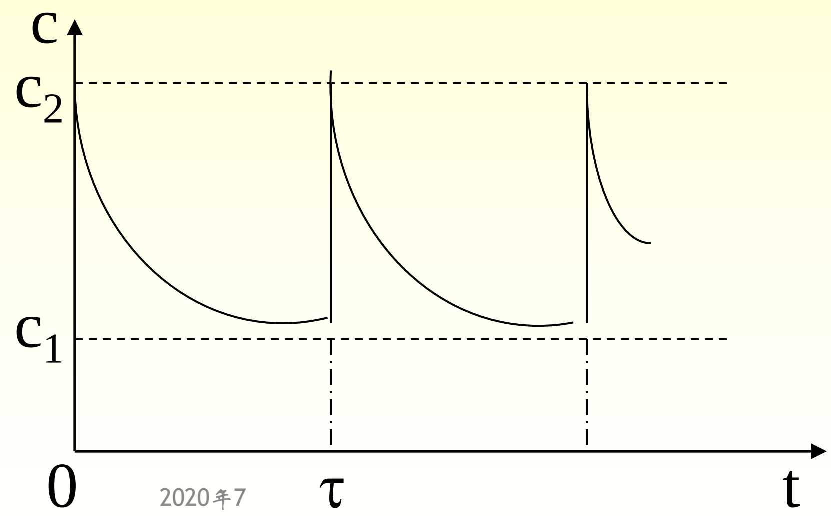
1. 在快速静脉注射的给药方式下，研究血药浓度（单位体积血液中的药物含量）的变化规律。

2. 给定药物的最小有效浓度和最大治疗浓度，设计给药方案：每次注射剂量多大；间隔时间多长。

分析

- 实验：对血药浓度数据作拟合，符合负指数变化规律

- 理论：用一室模型研究血药浓度变化规律



模型假设

1. 机体看作一个房室，室内血药浓度均匀——一室模型

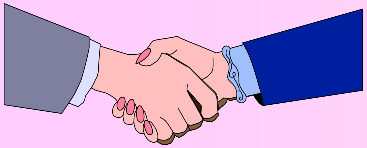
2.药物排除速率与血药浓度成正比，比例系数 k(>0)

3.血液容积v, t=0注射剂量d, 血药浓度立即为d/v.

模型建立

$$\left.\begin{aligned}&\text{由假设2得：}\frac{\mathrm{dc}}{\mathrm{dt}}=-\mathrm{kc}\\&\text{由假设3得：}\mathrm{c}(0)=\mathrm{d}/v\end{aligned}\right\}\Rightarrow c(t)=\frac{d}{v}e^{-kt}$$

在此，d=300mg，t及c（t）在某些点处的值见前表，需经拟合求出参数*k*、*v*



用线性最小二乘拟合c(t)

$$c(t)=\frac{d}{v}e^{-kt}\Rightarrow \ln c=\ln(d/v)-kt$$

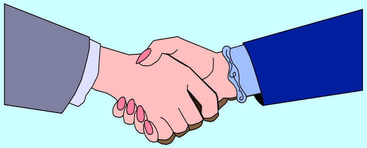
$$\left.\begin{array}{l} c(t)=\frac{d}{v}e^{-kt}\Rightarrow \ln c=\ln(d/v)-kt \\ y=\ln c,\ a_1=-k,\ a_2=\ln(d/v) \end{array}\right\}\Rightarrow \begin{array}{c} y=a_1t+a_2 \\ k=-a_1, v=d/e^{a_2} \end{array}$$

程序：

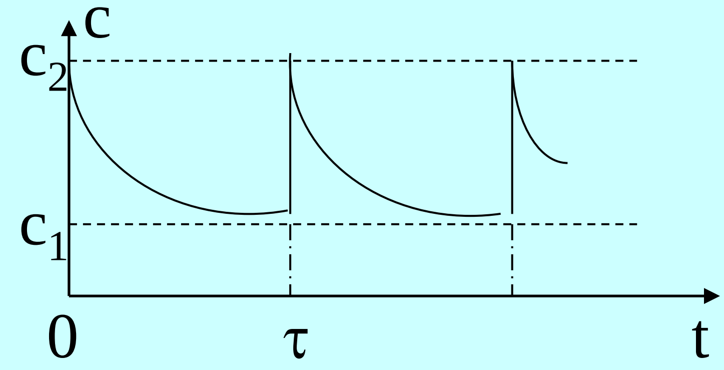
```
d=300;
t=[0.25 0.5 1 1.5 2 3 4 6 8];
c=[19.21 18.15 15.36 14.10 12.89 9.32 7.45 5.24 3.01];
y=log(c);
a=polyfit(t,y,1)
k=-a(1)
v=d/exp(a(2))
```

MATLAB(lihe1)

计算结果： $k=0.2347(1/h),\ v=15.02(l)$



给药方案 设计



- 设每次注射剂量D, 间隔时间τ
- 血药浓度c(t) 应$c_1 \le c(t) \le c_2$
- 初次剂量D_0 应加大

给药方案记为：$\{D_0, D, \tau\}$

1. $D_0 = vc_2,\ D = v(c_2 - c_1)$

2. $c_1 = c_2 e^{-k\tau} \Rightarrow \tau = \dfrac{1}{k}\ln\dfrac{c_2}{c_1}$

c_1=10,c_2=25

k=0.2347

v=15.02

计算结果：$D_0 = 375.5, D = 225.3, \tau = 3.9$

给药方案：$D_0 = 375(mg),\ D = 225(mg),\ \tau = 4(h)$

故可制定给药方案：

$$D_0 = 375(mg),\ D = 225(mg),\ \tau = 4(h)$$

即：

首次注射375mg，

其余每次注射225mg，

注射的间隔时间为4小时。