Matlab 与其他文件的交互

文件操作

1、fopen 函数
   a) fid=fopen(‘路径’,’r/w…’)
2、fclose 函数
   a) status=fclose(fid)

与 txt 文件交互

1、读取 txt 文件中的数据
   a) 函数 load 与 save
      用法：b=load(‘路径’)读取所有数据到 b 中
      save 用法：save 路径 b –方法
      说明：b 为存储的对象，方法为存储的方案。例如 ascii，按照 ascii 方式存入。
      注意：save 写入文档会先清空文档，再写入。
   b) 函数 textread
      i. 说明：若 txt 文档中存储了不同类型的数据，分类读取就需要使用 textread 函数了。使用时不需要 fopen 打开文件。
      ii. 用法：[A,B,…]=textread(‘filename’,’format’,N,’headerlines’,M)
      其中 filename 为文件名称，format 表示读出变量的字段格式，N 表示读次数，每一次读取一行。Headerlines 表示从第 M+1 行开始读入。

※与 excel 文件交互

1、xlsread 函数
   a) 用法：a=xlsread(‘路径’,sheet,’?:?’)
      说明：路径即为文件路径，sheet 为整数，表示第几个表格（sheet1），最后一个输入变量为表示读取数据的范围，例如 A5:B7。
2、xlswrite 函数
   a) 用法：xlswrite(‘路径’,a,sheet,’?:?’)
      说明：路径即为文件路径，sheet 为整数，表示写入的表格，例如 3 表示 sheet3；a 表示要写入的数据，最后一个输入变量表示写入数据的表格范围，例如 A1:C8

优化问题

优化问题是在多种约束下研究最佳目标方案的问题。Matlab 优化应用提供了求解优化问题的方法。包括线性规划、二次规划、非线性规划、多目标规划和最小二乘问题。

优化问题求解过程

1、确定目标方程的类型：线性，二次等
2、确定约束方程属于下面哪一类：无约束，边界，线性，平滑，离散
3、先择求解器。书 P220

线性规划的 matlab 求解

标准形式：min $c^T x$　s.t. Ax<=b
函数 linprog(c,A,b,Aeq,Beq,LB,UB,X0,OPTIONS)
其中 Aeq，beq 为对应等式（Ax=b）约束，若表达式中无等式，则写为[ ],[ ]
LB,UB 为 x 的上限下限，x0 为 x 的初始值。OPTIONS 为控制参数

约束优化

1、fmincon 函数（见非线性规划求解部分）
2、fminbnd 函数
   [x,fval]=fminbnd(fun,x1,x2,options)
   比较简单，fun 为函数，x 被限制在区间[x1,x2]上，options 为 optimiset 选项

非线性规划的 matlab 求解

非线性规划标准形式:
Minf(x)　s.t.
Aeq*x=beq Ax<=b　C(x)<=0 Ceq(x)=0
解释 A B 部分与线性规划同　C(x)表示：非线性向量函数，Ceq(x)同理
Fmincon(fun,x0,A,b,Aeq,beq,lb,ub,nonclone,options)
解释：fun 为定义的函数，min 后面的 fx　x0 为 x 的初始值 A B 同线性规划 lb ub 同线性规划　nonclone 为非线性向量的约束条件**函数。Options 为优化参数。**

例：

【例 2－7】 求下列非线性规划问题：

$$\min f(x) = x_1^2 + x_2^2 + 8$$

$$\text{s. t.} \begin{cases} x_1^2 - x_2 \geqslant 0 \\ -x_1 - x_2^2 + 2 = 0 \\ x_1, x_2 \geqslant 0 \end{cases}$$

① 编写 M 文件 fun1. m 和 fun2. m。

```
function f = fun1(x);
f = x(1)^2 + x(2)^2 + 8;

function [g,h] = fun2(x);
g = - x(1)^2 + x(2);
h = - x(1) - x(2)^2 + 2; % 等式约束
```

② 在 MATLAB 的命令窗口依次输入如下语句：

```
options = optimset;
[x,y] = fmincon('fun1',rand(2,1),[],[],[],[],zeros(2,1),[], ...
'fun2', options)
```

分析：fun1 为 fx　fun2 为约束条件 x1 x2 有两个参数 所以 x 为有两个元素的行向量

Fun2 为约束条件 同理 注意大于等于要加符号变为标准形式 函数 fun2 中 返回的第一个元素为不等约束，第二个为等于约束

其余没有的为空[]

整数规划

应用函数 intlinprog

[X,fval]=intlinprog(f,intcon，A,b,Aeq,beq,lb,ub)其中，f 为待求解函数，intcon 为整数决策变量所在的位置。A 为不等约束左矩阵，b 为不等约束列向量（右），Aeq，beq 为等式约束的左右。Lb 为下限，ub 为上限

例子：

```
>>f=[-9 -5 -6 -4];
>> A=[6 3 5 2;0 0 1 1;-1 0 1 0;0 -1 0 1];
>> b=[9 1 0 0]';
>> [x,fval]=intlinprog(f,[1,2,3,4],A,b,[],[],zeros(4,1),ones(4,1))
```

结果为：

```
x =
1.0000
1.0000
     0
     0
fval = -14.0000
```

无约束优化

1、 Fminsearch

a) 求解 fx 的最小值，fx 为非线性函数

函数[x,fval]=fminsearch(fun,x0,options)
其中 x 为解，fval 为求得的值，fun 为函数，x0 为起始搜索的位置
例如：求 f=(x+1)^2 的最小值
>> clear
>> f=@(x)x^2+2*x+1;
>> [x,fval]=fminsearch(f,0)
即可

2、Fminunc
   a) 该函数可以在定义域上计算单个函数的最小值，还可以计算导数和偏导数
      [x,fval,exitflag,output,grad,hessian]=fminunc(fun,x0,options)
      其中 exitflag 为退出条件，output 为优化信息数据，grad 为导数向量，hessian 为偏导矩阵。

      例子见书 P231

最小二乘问题

多项式分析

1、多项式及其函数
   a) 多项式系数矩阵[1，4，1]代表： y（x）=x^2+4x+1
2、多项式计算
   a) Polyval(p,x) p 为多项式向量，x 为自变量的值
3、多项式方程求根
   a) Roots（c） c 为系数向量
4、多项式四则运算
   a) 加法：通过补 0 的方法可以进行相加
   b) 乘法：conv（u，v）u，v 为两个多项式向量
   c) 除法：[q,r]=deconv（u，v）q 为返回的向量，r 为残余向量
5、求导：
   a) Polyder(p) 求导数
   b) Polyint(p，k) 求积分 k 为 C(常数)

插值与拟合

1、插值
   a) 定义：通过有限点来建立简单的连续的解析模型，并根据该模型得到未知点处的值。
      可分为：一维插值，多维插值和离散数据插值

2、 一维插值

a) Interp1 函数实现

Yi=Interp1(x, Y, xi, method)

其中 x 为自变量的取值向量，Y 为对应的函数值，Xi 为插值点，yi 为插值结果，method 为插值方法。书 P207 例如：三次样条插值：spline

例子：

x	1	2	3	4	5	6	7	8	9
y	Sin1	Sin2	Sin3	Sin4	Sin5	Sin6	Sin7	Sin8	Sin9

Matlab 代码：

```
X=1:9;                          %x 坐标
Y=sin(x);                       %对应的函数值 y
Xi=1:.25:9                      %构造插值点
Yi=interp1(x,y,xi,'spline');    %进行插值，插值所得的 y（函数）点
Plot(x,y,'o',xi,yi)             %画插值的图
```

3、 多维插值

a) Interp2 函数 二维数据的插值

i. 用法：zi=interp2(x,y,z,xi,yi,method)

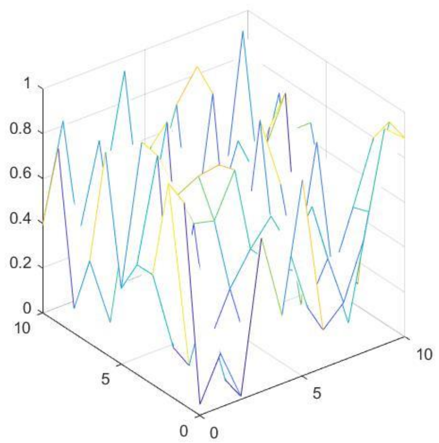
解释：x,y,z 为采样数据点坐标，xi,yi,zi 为等待插值点的坐标和返回值。Method 为插值方法。

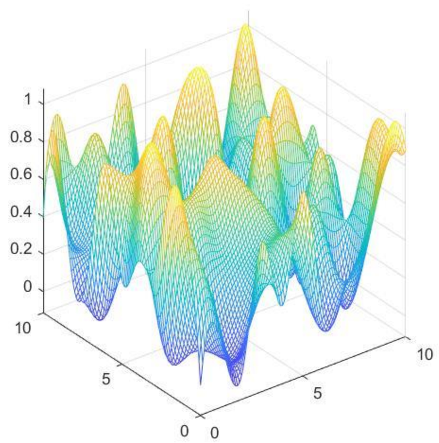
注意※： x y z 要满足网格向量。

例：

```
z=rand(11,11);                                          %生成随机数 z
[x,y]=meshgrid(0:10);                                   %构造网格向量 x, y
[xi,yi]=meshgrid(0:.1:10);                              %构造插值点
zi=interp2(x,y,z,xi,yi,'spline');                       %插值
subplot(121);mesh(x,y,z);subplot(122);mesh(xi,yi,zi) %画图
```

插值结果如下图：





b) Griddedinterpolant 函数进行二维数据的插值（见书 P208）

c) 三维插值：

   i. Interp3 函数

      Vi=Interp3(x,y,z,v,xi,yi,zi,method)

      例子见书 P210

4、离散数据插值

a) Griddata 函数

   Vq=griddata（x,y,v,,xq,yq）

   其中 x，y 为坐标，v 为对应采样值，xq，yq 为插值点坐标。Vq 为插值结果。Method 为插值方法。

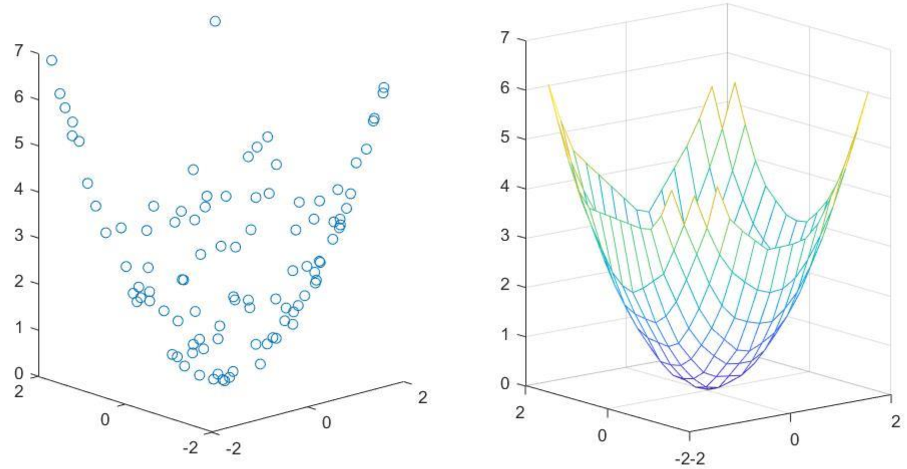
b) Triscatteredinterp 函数

   F=Triscatteredinterp(x,y,v,method) ，与上类似，输入变量没有了插值点坐标。

   例子：

```
x=rand(100,1)*4-2;
y=rand(100,1)*4-2;                      %产生随机坐标点
z=x.^2+y.^2;                            %对应坐标点的 z 值
f=TriScatteredInterp(x,y,z);            %进行插值，f 返回的是一个二元函数，注意大小写
[qx,qy]=meshgrid(-2:.25:2,-2:.25:2)     %构造网格点
qz=f(qx,qy);
subplot(122);mesh(qx,qy,qz);subplot(121);plot3(x,y,z,'o')   %绘图
```

   如图所示



5、曲线拟合

a) 概念：用解析表达式逼近离散数据的一种方法。曲线拟合可以在数据间支持定量关系式。

b) Polyfit 函数

   i. P=polyfit(x,y,n)

      其中，x y n 为输入的 x 值，y 值和拟合多项式的阶数，p 为得到的多项式系数

      例子见数 P213

6、 回归

a) regress 函数

用法：[B,BINT,R,RINT,STATS] = regress(Y,X,ALPHA)

说明：B 为回归系数，是一个向量，即 Y=B0+X1*B1+X2*B2+…

BINT 为回归系数的区间估计

R 为残差

RINT 为置信区间

STATS 为用于检验回归模型的统计量，有四个数值。

ALPHA 为显著性水平，默认为 0.05

Y,X 为待回归数据。

（注意：在进行 regress 函数回归时，要在数据 X 前面加上一列为 1 的向量，即代表回归后得到的常数项（B0））

7、 插值、拟合、回归、逼近的区别

①回归一般指**线性回归**，是求最小二乘解的过程。在求回归前，已经假设所有型值点同时满足某一曲线方程，计算只要求出该方程的系数

②多项式**插值**：用一个多项式来近似代替数据列表函数，并要求多项式通过列表函数中给定的数据点。（插值曲线要经过型值点。）

③多项式**逼近**：为复杂函数寻找近似替代多项式函数，其误差在某种度量意义下最小。（逼近只要求曲线接近型值点，符合型值点趋势。）

④多项式**拟合**：在插值问题中考虑给定数据点的误差，只要求在用多项式近似代替列表函数时，其误差在某种度量意义下最小。

常微分方程

书 P217

1、 常微分方程数值求解的常用调用格式：

[T,Y]=solver(odefun,tspan,y0)

其中：solver 可使用 ode23、ode45 等代替，odefun 为常微分方程右项的函数句柄，即满足了 y'=f(t,y)的格式。Tspan 为积分间隔的向量。Y0 为初始条件。

例子：

求解：微分方程组：$\frac{dy_1}{dt}=y_2$ ，$\frac{dy_2}{dt}=(1-y_1^2)\times y_2-y_1$ 数值解 [0,20]初始值：[y1=2;y2=0]

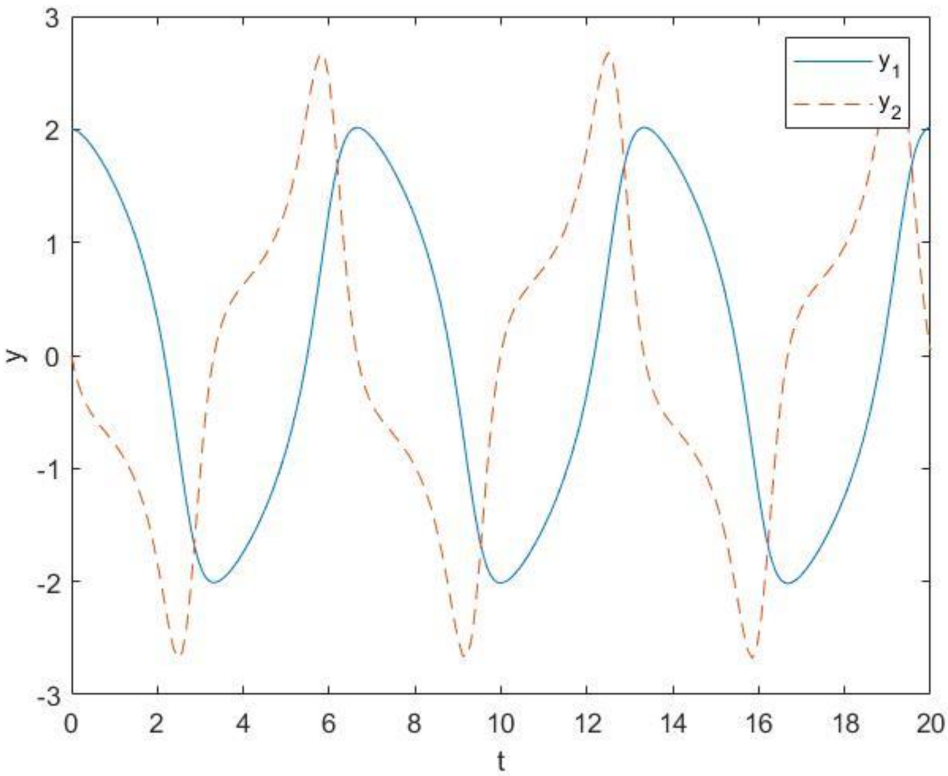
Matlab 代码：

```
[t,y]=ode45(@fx,[0 20],[2;0]);
plot(t,y(:,1),'-',t,y(:,2),'--');
xlabel('t');ylabel('y');legend('y_1','y_2')
```

可以得到图像:

（注：xlabel，ylabel 标注 x 轴，y 轴；

legend 在坐标区上添加图例）



图

1、 图的定义与术语
   a) 顶点：数据元素
   b) 有向图：边是有方向的，弧
   c) 无向图：边无方向
   d) 完全图：任意两元素都相邻
   e) 稀疏图：有很少边
   f) 稠密图：与稀疏图相反，有很多边
   g) 顶点的度：与一个顶点相连的边数
   h) 连通图：任意两个顶点都是连通的
2、 图的存储结构:
   a) 数组表示法：邻接矩阵
   b) 邻接表:
3、 最短路径问题
   a) 迪杰斯特拉(Dijkstra)算法:

I. 初始时，S 只包含起点 s；U 包含除 s 外的其他顶点，且 U 中顶点的距离为"起点 s 到该顶点的距离"[例如，U 中顶点 v 的距离为(s,v)的长度，然后 s 和 v 不相邻，则 v 的距离为∞]。

II. 从 U 中选出"距离最短的顶点 k"，并将顶点 k 加入到 S 中；同时，从 U 中移除顶点 k。

III. 更新 U 中各个顶点到起点 s 的距离。之所以更新 U 中顶点的距离，是由于上一步中确定了 k 是求出最短路径的顶点，从而可以利用 k 来更新其它顶点的距离；例如，(s,v)的距离可能大于(s,k)+(k,v)的距离。

IV.重复步骤(2)和(3)，直到遍历完所有顶点。

遗传算法

1、 Matlab 工具箱函数

a) ga 函数遗传算法

格式：[x,fval,reason]=ga(@funfitness,nvars,options)

其中，x 为经过遗传进化后自变量最佳染色体的返回值，fval 为对应的最佳染色体对应的适应度；reason 为算法停止的原因；@funfitness 为适应度句柄函数；nvars 为目标函数自变量的个数；options 为算法设置的属性。

b) 函数 gaoptimset

函数 gaoptimset 的语法格式为

```
options = gaoptimset('PropertyName1','PropertyValue1',
'PropertyName 2','PropertyValue 2','PropertyName 3','PropertyValue 3'……)
```

函数 gaoptimset 实现的功能为，设置遗传算法的参数和句柄函数，表 5－4 所列为函数 gaoptimset 常用的 11 种属性。

表 5－4 函数 gaoptimset 属性

属性名	默认值	实现功能
PopInitRange	[0,1]	初始种群生成区间
PopulationSize	20	种群规模
CrossoverFraction	0.8	交配概率
MigrationFraction	0.2	变异概率
Generations	100	超过进化代数时算法停止
TimeLimit	Inf	超过运算时间限制时算法停止
FitnessLimit	－Inf	最佳个体等于或小于适应度阈值时算法停止
StallGenLimit	50	超过连续代数不进化则算法停止
StallTimeLimit	20	超过连续时间不进化则算法停止
InitialPopulation	[ ]	初始化种群
PlotFcns	[ ]	绘图函数，可供选择的有@gaplotbestf,@gaplotbestindiv 等

概率统计

统计量操作

1、产生随机数

见书 P244

其中包括产生 均匀分布连续随机数，产生指数分布随机数，产生卡方分布随机数等等

①均匀分布随机数

Unifrnd 函数产生[A,B]区间上的均匀分布随机数。

R=unfirnd(A,B,[n,m,…])

其中，AB 代表范围，n、m 等代表返回的矩阵维数

②二项分布随机数 binornd 函数 和 nbinrnd 函数 用法简单见书 P246

③正态分布随机数 randn 函数 P246

2、抽样

a) 自助抽样法

i. 函数 bootstrp

[bootstat,bootsam]=bootstrap(nboot,bootfun,d1,…)

其中 nboot 为抽样数据，bootfun 为采用的计算函数，d1 为 bootfun 的输入数据，返回值 bootstat 为向量、bootsam 为下标矩阵

b) 褶刀抽样法（略）

数据统计分析

1、特征统计量

a) 平均值与中值

使用 mean 函数求均值，使用 median 函数求中位数。使用 nanmedian 求忽略 nan 的中位数。用 geomean 求几何平均数， harmmean 求调和平均数。用法简单例子见 P249

b) 数据比较

数据比较是指由数据比较引发的各种数据操作，包括普通排序，按行排序和求解值域大小等操作。Sort，sortrows（以第一列为基准行行变换）、range 。用法简单

例如：A=randn(5);

Y1=sort(A);Y2=sortrows(A);Y3=range(A)

c) 方差与标准差

i. 函数 var 求解方差 var（A）

ii. 函数 std 求解标准差 std(A)

d) 协方差与相关系数

i. 函数 cov 计算协方差

ii. 函数 corrcoef 计算相关系数

对于一般的矩阵 X，执行 A=corrcoef(X)后，A 中每个值的所在行 a 和列 b，反应的是原矩阵 X 中相应的第 a 列向量和第 b 列向量的相似程度（即相关系数）。

易得，A 为一个对称矩阵。(aij=aji)

2、 ※统计图表

a) 频次表

i. 函数 tabulate 获得元素出现频次的频次表。

TABLE =tabulate(x)

其中 x 为带统计向量，若为数值向量，则返回含有三列数据的表格。第一列为 x 的取值， 第二列为频次， 第三列为百分比。

b) 概率分布函数图

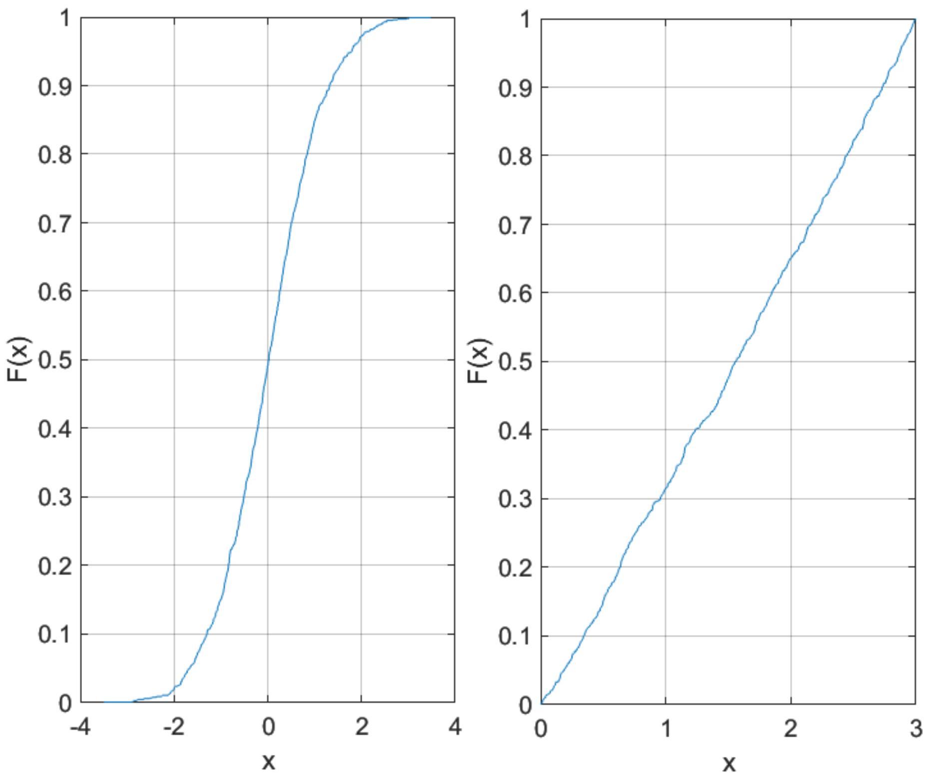
i. 函数 cdfplot 可以绘制累积分布函数的图形。

[h,stats]=cdfplot(X)

其中， X 为向量， h 表示曲线的句柄， stats 表示样本部分特征

```
例子：y1=randn(1000,1);
subplot(121);
cdfplot(y1);title('');
y2=3*rand(1000,1);
subplot(122);
cdfplot(y2);title('');
```

图为：

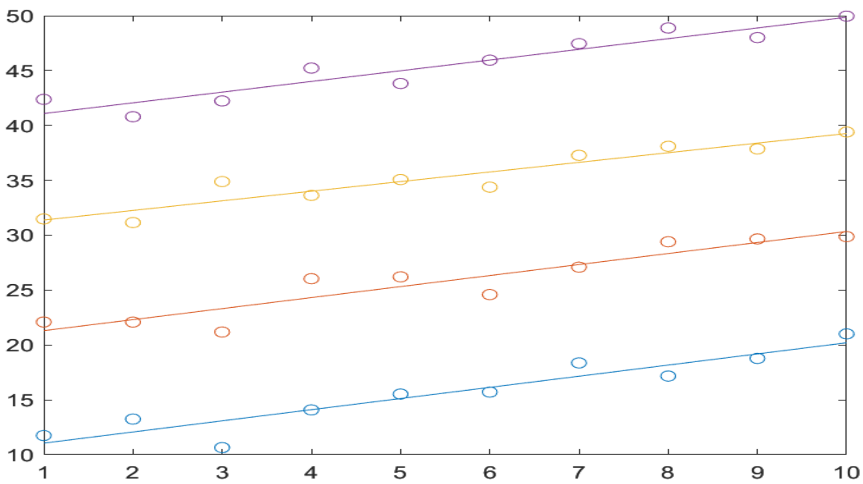


c) 最小二乘拟合直线

i. 函数 h=lsline h 为拟合曲线的句柄 用法见书 P253

例子:

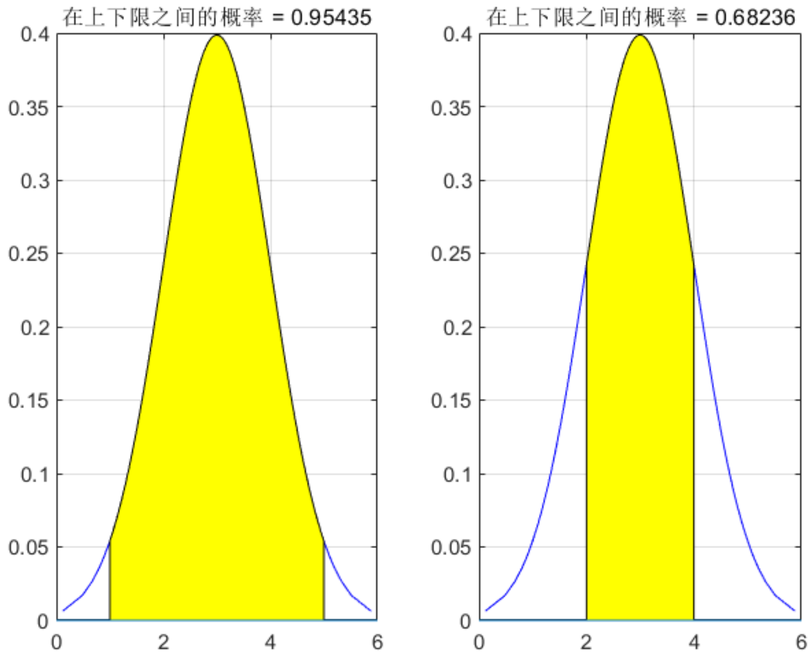
```
x=1:10;
for i=1:4
    y(i,:)=x+randn(1,10)+10*i;
    scatter(x,y(i,:),25,"blue",'*');
end
plot(x,y,'o');
h=lsline;
for j=1:4
    h(1)
end
```



d) 正态分布概率分布图
   i. 函数 normplot
e) 盒图
   i. 函数 boxplot 书 P254
f) 参考线
   i. 函数 refline 绘制参考直线
      refline(m,b)或 refline(coeffs);
      其中 m 为斜率，b 为截距。coeffs 为前面两个参数构成的向量
   ii. 函数 refcurve 绘制参考曲线
      refcurve(p) 其中 p 为多项式系数向量
g) 样本概率图
   i. 函数 capaplot 可以绘制样本的概率图
      [p,h]=capaplot(data,specs)
      其中 data 为所给样本数据，specs 为指定范围，p 表示在指定范围内的概率。
      该函数返回来自于估计分布的随机变量落在指定范围内的概率。
      （注意：这是已知一组数据画出的概率图，再求某一给定范围出现的概率）
      例子:

```
data=normrnd(3,1,1000000,1);
subplot(121);p1=capaplot(data,[1 5])
grid on;
subplot(122);p2=capaplot(data,[2 4])
```

```
grid on;
```



h) 正态拟合直方图

i. 函数 histfit 可以绘制含有正态拟合曲线的直方图。

histfit(data,nbins,dist)

其中，data 为向量，nbins 指定 bar 的个数，dist 为分布类型。函数返回直方图和正态曲线。

例子见书 P257

概率分布与计算

1、概率密度计算

a) 通用概率密度计算函数

i. pdf 函数用于计算概率密度。

用法：y=pdf(name.X,A,B,C)

该函数返回在 X 处，参数为 A,B,C 时数据的概率密度值。对于不同的分布，参数个数不同。Name 为分布函数名。

例子见书 P258

ii. 函数 ksdensity 函数求取一般函数/数据的概率密度函数

用法：f=ksdensity(x,xi)

其中，f 为返回的概率密度函数，x 为待统计向量，xi 为计算概率密度的点

b) 专用概率密度计算函数

i. 概述：计算特定的函数的概率密度。见书 P259

2、概率分布计算

a) 通用概率分布计算函数

i. 函数 cdf

Y=cdf('name',X,A,B,C)

该函数返回在 X 处，参数为 ABC 的分布的累计概率值。Name 为分布函数名。

b) 专用概率分布计算函数

   i. 概述：计算特定函数的概率分布 见书 P261