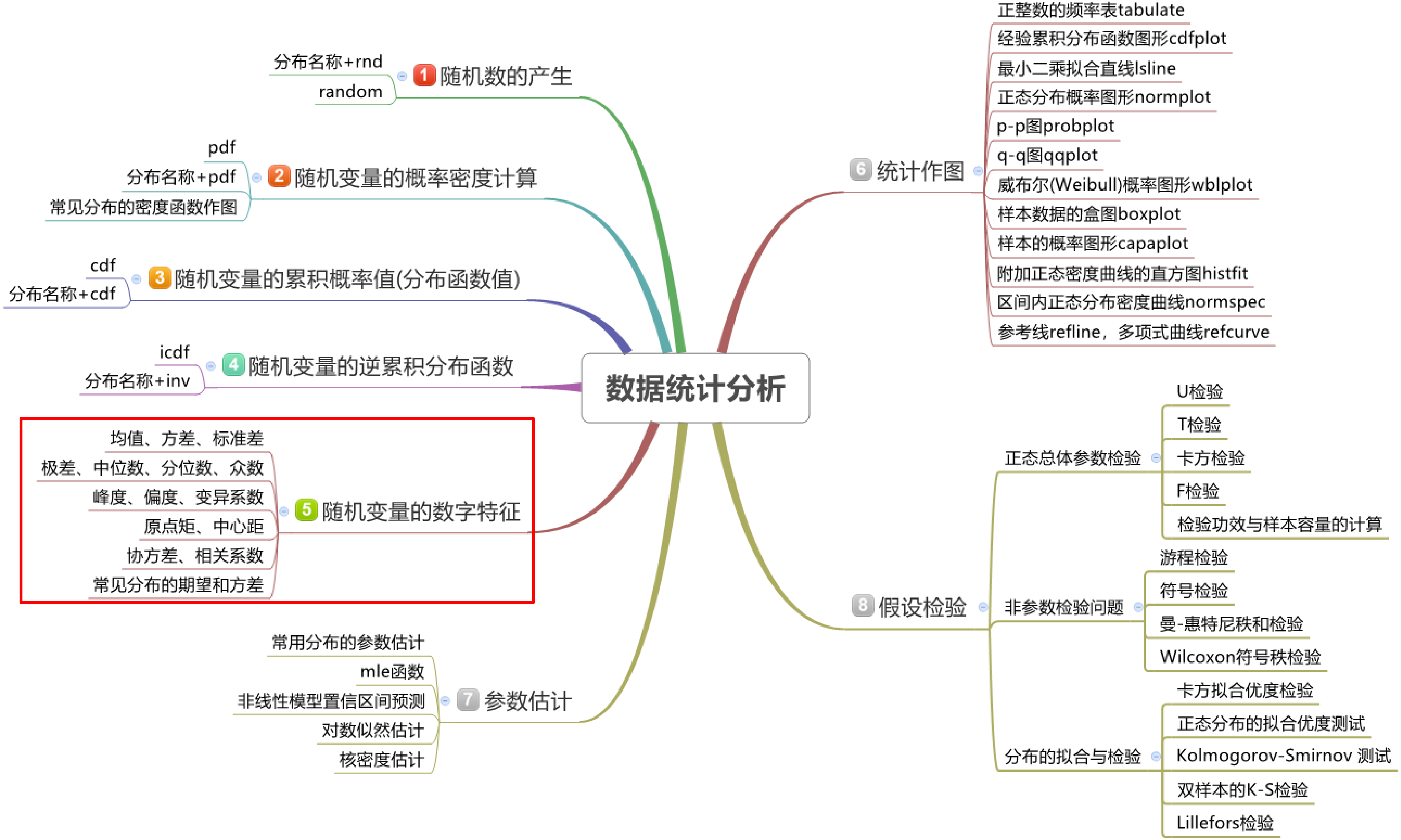
信陽師範學院
数学与统计学院
SCHOOL OF MATHEMATICS AND STATISTICS

第10章 数据统计分析

讲授人：牛言涛

日期：2020年4月8日

第10章 数据统计分析知识点思维导图



1. 均值

调用格式	解释
mean(X) , mean(A) mean(A,dim)	X为向量；A为矩阵，返回A中各列元素的平均值构成的向量；dim给出的维数内的平均值；算术平均值数学含义 $\bar{x}=\frac{1}{n}\sum_{i=1}^{n}x_i$，即样本均值。
nanmean(X),nanmean(A)	返回向量X或矩阵A中除NaN外元素的算术平均值（矩阵为各列）。
geomean(X),geomean(A)	几何平均数的数学含义是 $M=(\prod_{i=1}^{n}x_i)^{\frac{1}{n}}$，其中样本数据非负，主要用于对数正态分布。几何平均数仅适用于具有等比或近似等比关系的数据。
harmmean(A)	调和平均值的数学含义是 $M=\frac{n}{\sum_{i=1}^{n}\frac{1}{x_i}}$，其中：样本数据非0，主要用于严重偏斜分布。

3. 数据排序

信阳师范学院
数学与统计学院

B = sort(A) 按升序对 A 的元素进行排序。

- 如果 A 是向量，则 sort(A) 对向量元素进行排序。
- 如果 A 是矩阵，则 sort(A) 会将 A 的列视为向量并对每列进行排序。
- 如果 A 是多维数组，则 sort(A) 会沿大小不等于 1 的第一个数组维度计算，并将这些元素视为向量。

B = sort(A,dim) 返回 A 沿维度 dim 的排序元素。例如，如果 A 是一个矩阵，则 sort(A,2) 对每行中的元素进行排序。

B = sort(___,direction) 使用上述任何语法返回按 direction 指定的顺序显示的 A 的有序元素。'ascend' 表示升序（默认值），'descend' 表示降序。

B = sort(___,Name,Value) 指定用于排序的其他参数。例如，sort(A,'ComparisonMethod','abs') 按模对 A 的元素进行排序。

[B,I] = sort(___) 还会为上述任意语法返回一个索引向量的集合。I 的大小与 A 的大小相同，它描述了 A 的元素沿已排序的维度在 B 中的排列情况。例如，如果 A 是一个向量，则 B = A(I)。

```
>> V = [9 0 -7 5 3 8 -10 4 2];  %向量
>> sv = sort(V)  %向量排序，默认升序
>> B = sort(A,2)  %按升序对每一行排序
>> B = sort(A,2,'descend') %按升序对每一行降序排序
>> [B,I] = sort(A)  %返回I索引矩阵 A(I) == B
```

3. 数据扩展排序

按行方式排序，函数sortrows

- Y=sortrows(A) %A为矩阵，返回矩阵Y，Y按A的第1列由小到大，以行方式排序后生成矩阵。
- Y=sortrows(A, col)　%按指定列col由小到大进行排序
- [Y,I]=sortrows(A, col)　% Y为排序的结果，I表示Y中第col列元素在A中位置。说明　若X为复数，则通过|X|的大小排序。

```
>> A = floor(gallery('uniformdata',[6 7],0)*100);
>> A(1:4,1) = 95;  A(5:6,1) = 76;  A(2:4,2) = 7;  A(3,3) = 73
>> B = sortrows(A)   %类似于Excel的扩展区域排序
>> C = sortrows(A,2)   %按照第二列排序
>> D = sortrows(A,[1 7])  %首先按照第一列排序，若元素相同，则按照第7列排序
>> [E,index] = sortrows(A, -4)   %按照第4列降序排列
```

3. 数据扩展排序

LastName = {'Smith';'Johnson';'Williams';'Jones';'Brown'};

Age = [38;43;38;40;49];

Height = [71;69;64;67;64];

Weight = [176;163;131;133;119];

BloodPressure = [124 93; 109 77; 125 83; 117 75; 122 80];

tblA = table(Age,Height,Weight,BloodPressure,'RowNames',LastName)

tblB = sortrows(tblA) %The sortrows function sorts the rows in ascending order first by the variable Age, and then it sorts by the variable Height.

[tblB,index] = sortrows(tblA,'RowNames') %The sortrows function sorts the rows in ascending order by the row names.

%Sort the rows of the table in ascending order by Height, and then sort in descending order by Weight. Also, return an index vector, such that tblB = tblA(index,:).

[tblB,index] = sortrows(tblA,{'Height','Weight'},{'ascend','descend'})

4. 极差

Description

`y = range(X)` returns the difference between the maximum and minimum values of sample data in X.

- If X is a vector, then `range(X)` is the range of the values in X.
- If X is a matrix, then `range(X)` is a row vector containing the range of each column in X.
- If X is a multidimensional array, then `range` operates along the first nonsingleton dimension of X, treating the values as vectors. The size of this dimension becomes 1 while the sizes of all other dimensions remain the same. If X is an empty array with first dimension 0, then `range(X)` returns an empty array with the same size as X.

`y = range(X,'all')` returns the range of all elements in X.

`y = range(X,dim)` returns the range along the operating dimension `dim` of X. For example, if X is a matrix, then `range(X,2)` is a column vector containing the range value of each row.

`y = range(X,vecdim)` returns the range over the dimensions specified in the vector `vecdim`. For example, if X is a matrix, then `range(X,[1 2])` is the range of all elements in X because every element of a matrix is contained in the array slice defined by dimensions 1 and 2.

```
>> load examgrades
>> y = range(grades,1)   %按列
y =
    42    31    35    43    24
>> y = range(grades,'all') %2016b版本不支持
```

```
>> y = range(grades(:))  %所有值的极差
y = 46
>> r = max(grades(:)) - min(grades(:))
r = 46
>> y = range(grades,[1,2]) %2016b版本不支持
```

5. 中位数

将样本观测值从小到大依次排列，位于**中间**的那个观测值，称为**样本中位数**，它描述了样本观测数据的中间位置。median函数用来计算样本的中位数。

- median(M,dim)：dim为1，2。其中1表示按每列返回一个值，为该列从大到小排列的中间值，2表示按每行返回一个值，为该行从大到小排列的中间值。
- nanmedian(M)：忽略NaN计算中位数。
- **注意：如果行或列的个数为偶数，返回中间两个值的平均值。**

```
>> me = median(grades)   %load examgrades
me =
    75    75    75    74    75
>> me = median(grades(:))
me = 75
```

6. 众数

众数描述了样本数据中出现次数最多的数

- [M,F,C]=mode(X) %M记录矩阵每列最频繁的元素，F记录该元素出现的次数，C的每个元素是与M的对应元素相同频率的所有值的排序向量；
- mode(X,1)：计算每列的频率最大值的行向量。 当有多个值有相等的频率时，mode返回这个最小的值；mode(X,2)：计算每行。

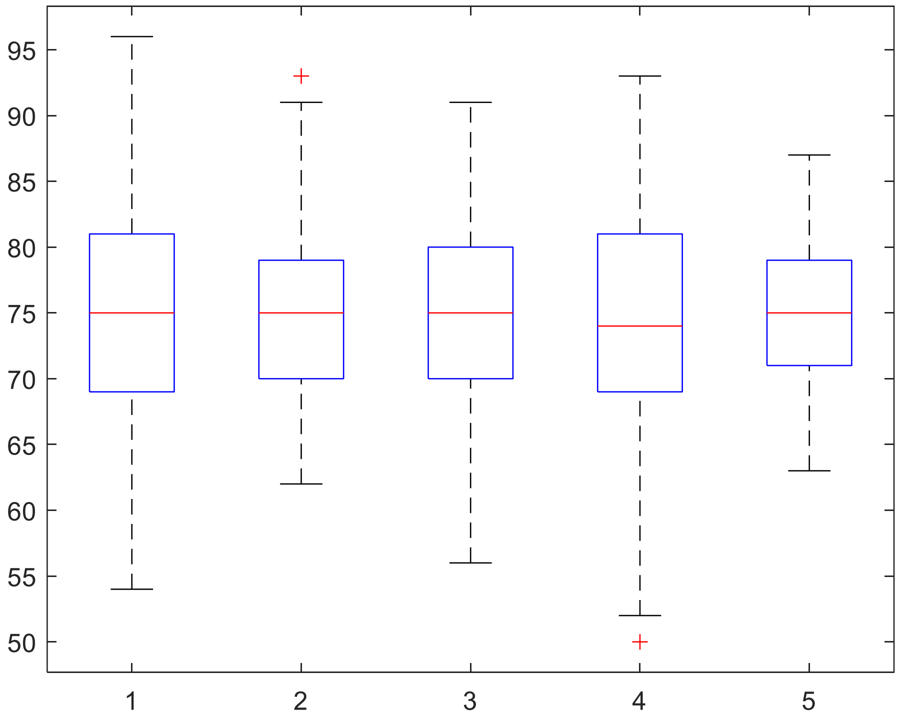
```
>> [M,F] = mode(grades)    %load examgrades
M =
    73    77    73    69    79
F =
    10    13     9    10    12
```

7. 分位数

- 分位数就是先把一列数按从小到大排序，如果一共有n个数，四分之一分位数就是第n*0.25个数，四分之三分位数就是第n*0.75个数，以此类推。p分位数就是第n*p个数。如果n*p不是整数则往最接近的较大的整数上归。样本的0.5分位数就是样本的中位数。
- quantile和prctile，均可用来计算样本的分位数，一个用小数表示分位数，一个用百分数表示分位数。
  - y = quantile(x,.50); % the median of x
  - y = quantile(x,[.25 .50 .75]); % the quartiles of x
  - y = quantile(x,3); % another way to get quartiles
  - y = quantile(x,[.025 .25 .50 .75 .975]); % a useful summary of x

7. 分位数

```
>> load examgrades
>> y1 = quantile(grades,[0.025,0.25,0.5,0.75,0.975]) %等价于prctile(score,[2.5,25,50,75,97.5])
y1 =
   57.5000   63.0000   59.5000   59.5000   64.5000
   69.0000   70.0000   70.0000   69.0000   71.0000
   75.0000   75.0000   75.0000   74.0000   75.0000
   81.0000   79.0000   80.0000   81.0000   79.0000
   91.0000   89.0000   89.0000   91.5000   85.5000
>> y2 = quantile(grades,3)  %求四分位数，%25，%50，75%
y2 =
    69    70    70    69    71
    75    75    75    74    75
    81    79    80    81    79
>> boxplot(grades)  %盒图
```



9. 原点矩与中心距

- 原点矩：cen_k = sum(X.^k)/length(X) $A_k = \frac{1}{n}\sum_{i=1}^{n} X_i^k, k = 1, 2, \cdots$
- 中心矩 moment，任意阶的中心矩 $B_k = \frac{1}{n}\sum_{i=1}^{n}\left(X_i - \bar{X}\right)^k, k = 1, 2, \cdots$
  - M = moment(X,order)：order为阶，函数本身除以X的长度

```
>> load examgrades
>> score = grades(:,1);
>> cen1 = sum(score.^1)/length(score)  %一阶矩，均值
cen1 =
   75.0083
>> cen2 = sum(score.^2)/length(score)  %二阶原点矩
cen2 =
   5.7017e+03
```

```
>> m2 = moment(score,2)  %二阶中心矩
m2 =
   75.4083
>> mm2 = sum((score-mean(score)).^2)/length(score)
mm2 =
   75.4083
>> var(score,1)   %简单方差
```

10. 数学期望

信阳师范学院
数学与统计学院

1. 数组的平均值，可求样本均值：Y=mean(X)

- 功能：当X为向量时，输出一个平均数；当X为矩阵时，输出为行向量，对应于矩阵每列的平均值；因此计算矩阵所有数的平均值，应用嵌套：mean(mean(X))或m=mean(X(:))
- 与此类似的有：求和(sum),最大(max),最小(min)等

2. 离散型随机变量的期望：EX=sum(X.*P)

- 功能：计算随机值向量X与对应概率向量P的乘积之和

3. 连续型随机变量的期望：EX=int(x*fx,x,a,b)

- 功能：用积分计算期望。$E(X)=\int_{-\infty}^{+\infty} xf(x)\mathrm{d}x$

10. 数学期望

例1: 按规定，某车站每天8:00~9:00,9:00~10:00都恰有一辆客车到站，但到站的时刻是随机的，且两者到站的时间相互独立。其规律如下表。一旅客8:20到车站，求他候车时间的数学期望。

到站时刻	8:10 9:10	8:30 9:30	8:50 9:50
概率	1/6	3/6	2/6

解：设旅客的候车时间为X（以分计），X的分布律为

X	10	30	50	70	90
p_k	3/6	2/6	$1/6\times 1/6$	$1/6\times 3/6$	$1/6\times 2/6$

```
>> X = [10:20:90];
>> pk = [3/6,2/6,1/36,3/36,2/36];
>> EX = sum(X.*pk)    %离散型随机变量期望
EX =  27.2222
```

10. 数学期望

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

例2：设随机变量(X,Y)的概率密度

$$f(x,y)=\begin{cases}\dfrac{3}{2x^3y^2}, & \dfrac{1}{x}<y<x, x>1\\ 0, & other\end{cases}$$

求数学期望 $E(Y)$, $E\left(\dfrac{1}{XY}\right)$.

```
>> syms x y
>> fxy = 3/2/x^3/y^2;
>> Ey = int(int(y*fxy,y,1/x,x),x,1,inf)
Ey =
3/4
>> Exy = int(int(1/x/y*fxy,y,1/x,x),x,1,inf)
Exy =
3/5
```

例3：设二维随机变量(X,Y)的密度函数为

$$f(x,y)=\begin{cases}ax(1+4y^3), & 0<x<1, \quad 0<y<1,\\ 0, & other.\end{cases}$$

求a，$E(X)$，$E(Y)$，和$E(XY)$.

```
>> syms a x y
>> fxy = a*x*(1+4*y^3);
>> sa = solve(int(int(fxy,x,0,1),y,0,1) == 1,a);
>> fxy = subs(fxy,a,sa);
>> EX = int(int(x*fxy,x,0,1),y,0,1)
EX = 2/3
>> EY = int(int(y*fxy,x,0,1),y,0,1)
EY = 13/20
>> EXY = int(int(x*y*fxy,x,0,1),y,0,1)
EXY = 13/30
```

11. 随机样本—方差与标准差

- 求样本方差，函数var，调用格式如下：
  - D=var(X) % $s^2=\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{X})^2$ ，若X为向量，则返回向量的样本方差。
  - D=var(A) %A为矩阵，则D为A的列向量的样本方差构成的行向量。
  - D=var(X, 1) %返回向量（矩阵）X的简单方差（即置前因子为1/n的方差）。

$$s^2=\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{X})^2$$

  - D=var(X, w) %返回向量（矩阵）X的以w为权重的方差。

11. 随机样本—方差与标准差

- 求标准差，函数std
  - 格式std(X)　%返回向量（矩阵）X的样本标准差（置前因子为1/(n-1)）即：

$$std = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} x_i - \bar{X}}$$

  - std(X,1)　%返回向量（矩阵）X的标准差（置前因子为1/n）
  - std(X, 0)　%与std (X)相同
  - std(X, flag, dim)　%返回向量（矩阵）中维数为dim的标准差值，其中flag=0时，置前因子为1/(n-1)；否则置前因子为1/n 。

11. 随机样本—方差与标准差

例4：从一批均值250g的袋装糖果中随机抽取10袋，测得其质量（单位: g）为：249，247, 245, 246, 238, 246.5, 244, 241, 250, 247.

求样本均值、样本方差、样本标准差、样本二阶原点矩，并检验均值.

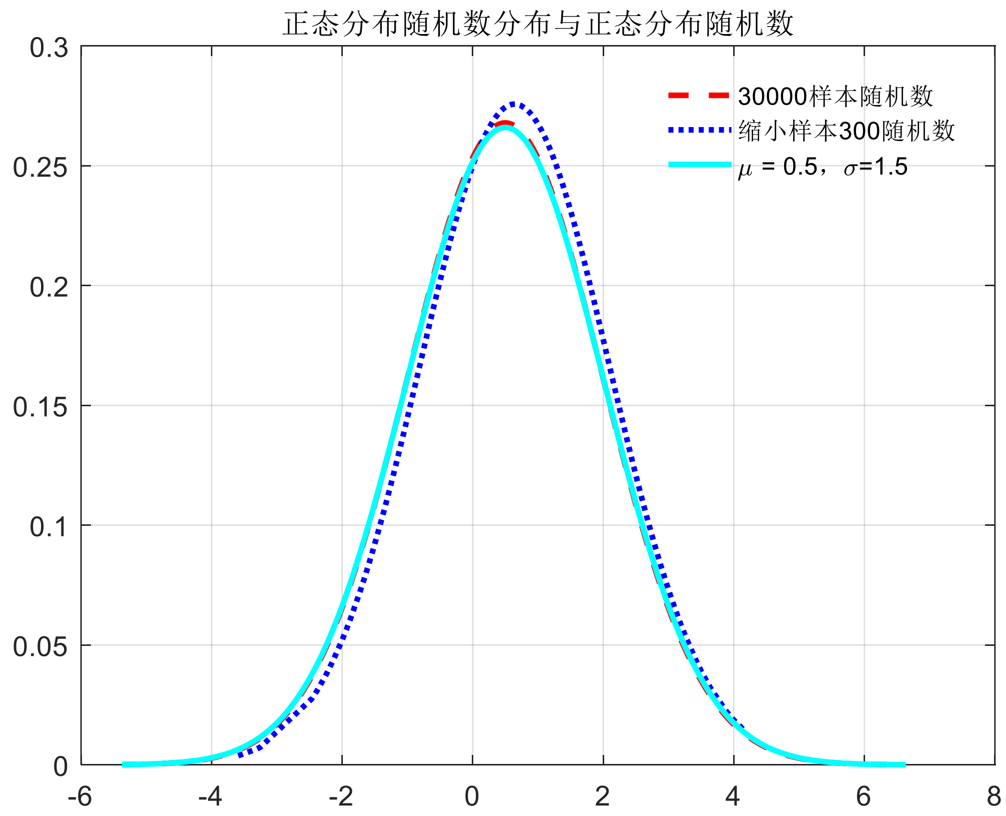
```
>> G = [249,  247, 245, 246, 238, 246.5, 244, 241, 250, 247];
>> Gm = mean(G)    %样本均值
Gm = 245.3500
>> Gv = var(G)   %样本方差
Gv = 13.0028
>> Gs = std(G)   %样本标准差
Gs = 3.6059
>> GA2 = mean(G.^2)   %样本二阶原点矩
GA2 = 6.0208e+04
```

```
>> [h,p,ci,stat] = ttest(G,250,0.05,0)  %t检验
h = 1  %拒绝原假设
p = 0.0028  %从概率值也可看出拒绝原假设
ci = 242.7705  247.9295  %置信区间
stat =
  包含以下字段的 struct:
    tstat: -4.0779
       df: 9
       sd: 3.6059
```

11. 随机样本—方差与标准差

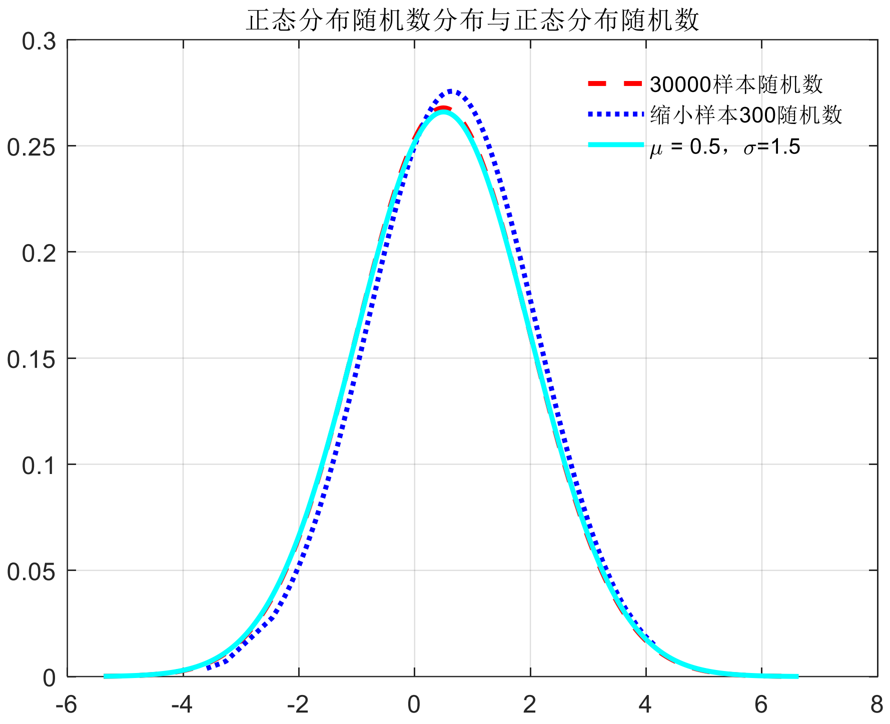
例5：生成一组 30000 个正态分布随机数，使其均值为0.5，标准差为1.5，分析数据实际的均值、方差和标准差，如果减小随机变量个数，会有什么结果?

```
>> R = normrnd(0.5,1.5,30000,1);
>> total =  [mean(R),var(R,1),std(R,1)]
total =
    0.5062    2.2276    1.4925
>> Rm = normrnd(0.5,1.5,300,1);
>> totalm =  [mean(Rm),var(Rm,1),std(Rm,1)]
totalm =
    0.4123    2.3991    1.5489
```



11. 随机样本—方差与标准差

```
>> p = normpdf(sort(R),total(1),total(3));
>> pm = normpdf(sort(Rm),totalm(1),totalm(3));
>> plot(sort(R),p,'r--','LineWidth',2)
>> hold on
>> plot(sort(Rm),pm,'b:','LineWidth',2)
>> Rd = linspace(min(R),max(R),1000);
>> pd = normpdf(Rd,0.5,1.5);
>> plot(Rd,pd,'c-','LineWidth',2)
>> legend('30000样本随机数','缩小样本300随机数','\mu = 0.5，\sigma=1.5')
>> grid on
>> legend('boxoff')
>> title('正态分布随机数分布与正态分布随机数')
```



11. 随机样本—方差与标准差

- 一般随机变量的方差$DX=E(X^2)-(EX)^2$ 功能：用积分或级数编程计算。

例6：设随机变量X的概率密度：

$$f(x)=\begin{cases}\dfrac{2}{\pi}\cos 2x, & |x|\le\dfrac{\pi}{2}\\ 0 & , \quad otherwise\end{cases}$$

求随机变量X的期望和方差。

```
>> syms x
>> fx=2/pi*(cos(x))^2;
>> EX=int(x*fx,x,-pi/2,pi/2)  %期望
EX =
   0
```

```
>> E2X=int(x^2*fx,x,-pi/2,pi/2)
E2X =
(1911387046407553*pi*(pi^2 - 6))/72057594037927936
>> DX=E2X-EX^2   %求方差
DX =
(1911387046407553*pi*(pi^2 - 6))/72057594037927936
>> dx = vpa(DX,8)
dx =
0.32246703
```

11. 随机样本—方差与标准差

- 忽略NaN的标准差，函数 nanstd

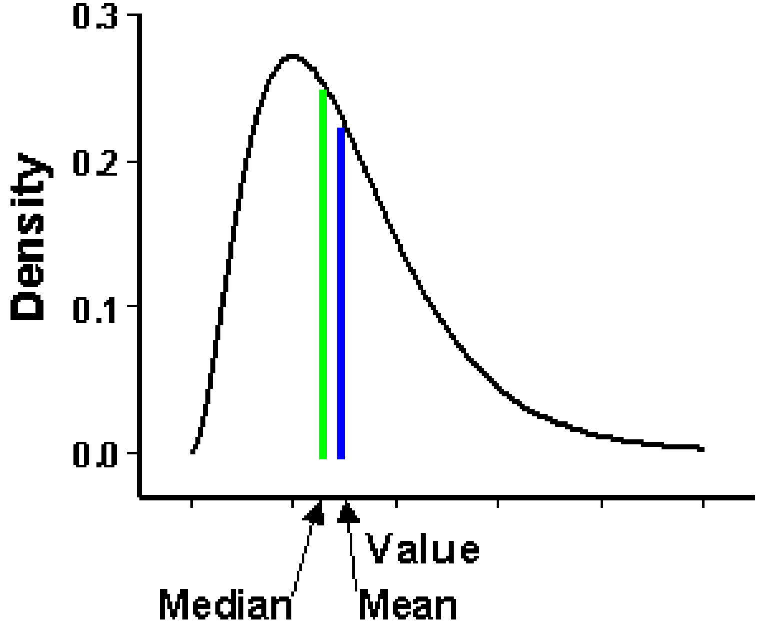
格式：y = nanstd(X)　%若X为含有元素NaN的向量，则返回除NaN外的元素的标准差，若X为含元素NaN的矩阵，则返回各列除NaN外的标准差构成的向量。

```
>> M=magic(3);   %产生3阶魔方阵
>> M([1 6 8])=[NaN NaN NaN]   %替换3阶魔方阵中第1、6、8个元素为NaN
>> y=nanstd(M)   %求忽略NaN的各列向量的标准差
y =
   0.7071    2.8284    2.8284
>> X=[1 5];    %忽略NaN的第2列元素，只有1和5元素
>> y2=std(X)   %验证第2列忽略NaN元素的标准差
y2 =
     2.8284
```

12. 偏度系数

- 峰度（Kurtosis）与偏度（Skewness）就是测量数据正态分布特性的两个指标。
- 偏度系数度量对称性。0说明是最完美的对称性，正态分布的偏态就是0。右偏态为正，表明平均值大于中位数。反之为左偏态，为负。

$$S=\frac{E(x-EX)^3}{[Var(X)]^{\frac{3}{2}}}\qquad S=\frac{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^3}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2\right)^{\frac{3}{2}}}$$



说明：偏斜度样本数据关于均值不对称的一个测度，如果偏斜度为负，说明均值左边的数据比均值右边的数据更散；如果偏斜度为正，说明均值右边的数据比均值左边的数据更散，因而正态分布的偏斜度为0；

12. 偏度系数

- 样本的偏斜度，函数skewness
  - y = skewness(X,flag) %X为向量，返回X的元素的偏斜度；X为矩阵，返回X各列元素的偏斜度构成的行向量；flag=0表示偏斜纠正，flag=1（默认）表示偏斜不纠正。

Description

`y = skewness(X)` returns the sample skewness of `X`.

- If `X` is a vector, then `skewness(X)` returns a scalar value that is the skewness of the elements in `X`.
- If `X` is a matrix, then `skewness(X)` returns a row vector containing the sample skewness of each column in `X`.
- If `X` is a multidimensional array, then `skewness(X)` operates along the first nonsingleton dimension of `X`.

`y = skewness(X,flag)` specifies whether to correct for bias (`flag = 0`) or not (`flag = 1`, the default). When `X` represents a sample from a population, the skewness of `X` is biased, meaning it tends to differ from the population skewness by a systematic amount based on the sample size. You can set `flag` to `0` to correct for this systematic bias.

`y = skewness(X,flag,'all')` returns the skewness of all elements of `X`.

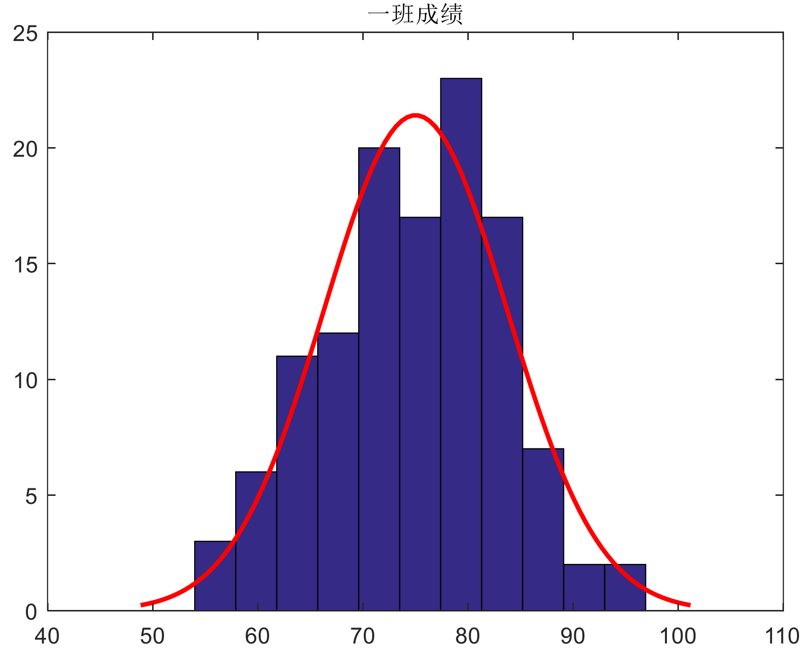
`y = skewness(X,flag,dim)` returns the skewness along the operating dimension `dim` of `X`.

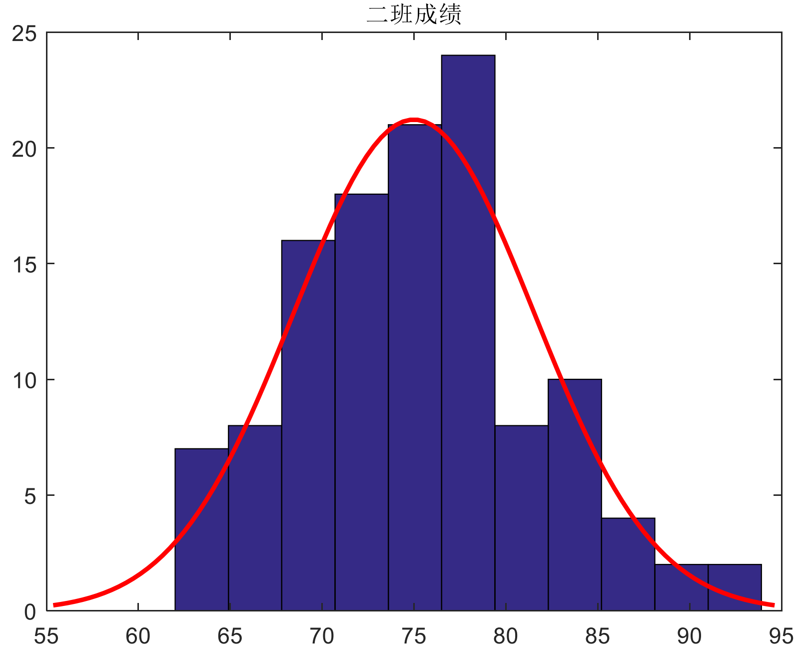
`y = skewness(X,flag,vecdim)` returns the skewness over the dimensions specified in the vector `vecdim`. For example, if `X` is a 2-by-3-by-4 array, then `skewness(X,1,[1 2])` returns a 1-by-1-by-4 array. Each element of the output array is the biased skewness of the elements on the corresponding page of `X`.

12. 偏度系数

```
>> load examgrades
>> gs = skewness(grades)
gs =
   -0.1983    0.3275   -0.1266   -0.0175    0.0413
>> histfit(grades(:,1))
>> figure
>> histfit(grades(:,2))
```

说明：偏斜度样本数据关于均值不对称的一个测度，如果偏斜度为负，说明均值左边的数据比均值右边的数据更散；如果偏斜度为正，说明均值右边的数据比均值左边的数据更散，因而正态分布的偏斜度为0；

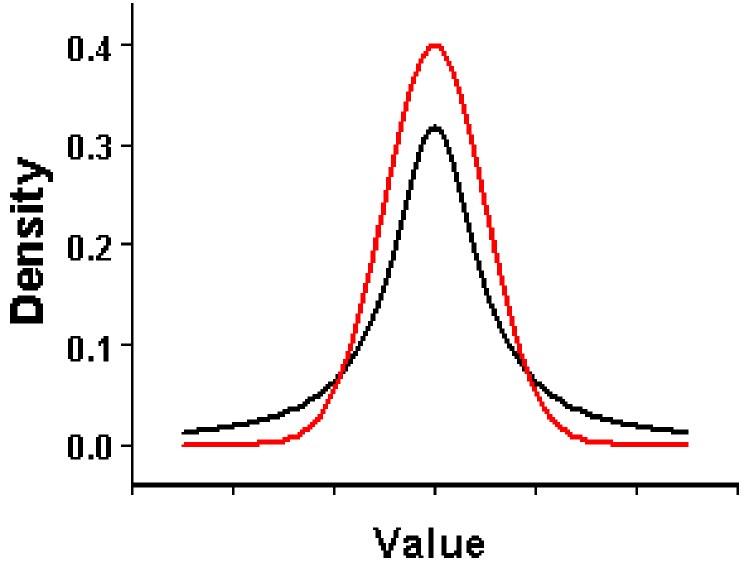




13. 峰度

- 峰度：用来衡量数据尾部分散性，它反映了分布曲线的陡缓程度。正态分布峰度为3，可用来检验分布的正态性，峰度>3，则厚尾，峰值附近陡峭；峰度<3，则细尾，峰值附近平缓。在金融时间序列分析中，通常要研究数据是否为尖峰、细腰、厚尾等特性。

$$K=\frac{E(X-EX)^4}{[Var(X)]^2}-3 \qquad K=\frac{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^4}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2\right)^2}$$




```
>> load examgrades
>> scorekur = kurtosis(grades)  %矩阵120*5成绩数据
scorekur =
    2.5522    2.7979    2.6116    2.9209    2.5646
>> scorekur = kurtosis(grades(:))  %所有元素的峰度
scorekur =
    3.0121
```

13. 峰度

- **峰度：** 用来衡量数据尾部分散性，它反映了分布曲线的陡缓程度。函数kurtosis

`k = kurtosis(X)` returns the sample kurtosis of `X`.

- If `X` is a vector, then `kurtosis(X)` returns a scalar value that is the kurtosis of the elements in `X`.
- If `X` is a matrix, then `kurtosis(X)` returns a row vector that contains the sample kurtosis of each column in `X`.
- If `X` is a multidimensional array, then `kurtosis(X)` operates along the first nonsingleton dimension of `X`.

`k = kurtosis(X,flag)` specifies whether to correct for bias (`flag` is `0`) or not (`flag` is `1`, the default). When `X` represents a sample from a population, the kurtosis of `X` is biased, meaning it tends to differ from the population kurtosis by a systematic amount based on the sample size. You can set `flag` to `0` to correct for this systematic bias.

`k = kurtosis(X,flag,'all')` returns the kurtosis of all elements of `X`.

`k = kurtosis(X,flag,dim)` returns the kurtosis along the operating dimension `dim` of `X`.

`k = kurtosis(X,flag,vecdim)` returns the kurtosis over the dimensions specified in the vector `vecdim`. For example, if `X` is a 2-by-3-by-4 array, then `kurtosis(X,1,[1 2])` returns a 1-by-1-by-4 array. Each element of the output array is the biased kurtosis of the elements on the corresponding page of `X`.

14. 变异系数

- 在概率论和统计学中，变异系数又称“离散系数”（coefficient of variation），是概率分布离散程度的一个归一化量度。
- 变异系数用于刻画数据的变化大小，不同指标的变异系数常用来计算客观性权重。计算变异系数的公式为：

$$CV = \frac{s}{|\bar{x}|} \times 100\% = \frac{std(X)}{abs(mean(X))} \times 100\%$$

- 变异系数越小，变异(偏离)程度越小，风险也就越小；反之，变异系数越大，变异(偏离)程度越大，风险也就越大。

14. 变异系数

```
>> load examgrades
>> sc_m = mean(grades)   %五个班级成绩均值
sc_m =
   75.0083   74.9917   74.9917   75.0333   74.9917
>> sc_std = std(grades)   %五个班级成绩标准差
sc_std =
    8.7202    6.5420    7.4309    8.6013    5.2588
>> cv = [sc_std./sc_m].*100   %五个班级成绩变异系数
cv =
   11.6256    8.7237    9.9090   11.4633    7.0126
```

注意，变异系数的大小，同时受平均数和标准差两个统计量的影响，因而在利用变异系数表示资料的变异程度时，最好将平均数和标准差也列出。

15. 协方差与相关系数

- 协方差cov(X,Y)=E(X-EX)(Y-EY)，函数cov

$$C=\begin{pmatrix} D(x) & \mathrm{cov}(x,y) \\ \mathrm{cov}(y,x) & D(y) \end{pmatrix}$$

C = cov(A) 返回协方差。

- 如果 A 是由观测值组成的向量，则 C 为标量值方差。
- 如果 A 是其列表示随机变量或行表示观测值的矩阵，则 C 为对应的列方差沿着对角线排列的协方差矩阵。
- C 按观测值数量 -1 实现归一化。如果仅有一个观测值，应按 1 进行归一化。
- 如果 A 是标量，则 cov(A) 返回 0。如果 A 是空数组，则 cov(A) 返回 NaN。

C = cov(A,B) 返回两个随机变量 A 和 B 之间的协方差。

- 如果 A 和 B 是长度相同的观测值向量，则 cov(A,B) 为 2×2 协方差矩阵。
- 如果 A 和 B 是观测值矩阵，则 cov(A,B) 将 A 和 B 视为向量，并等价于 cov(A(:),B(:))。A 和 B 的大小必须相同。
- 如果 A 和 B 为标量，则 cov(A,B) 返回零的 2×2 块。如果 A 和 B 为空数组，则 cov(A,B) 返回 NaN 的 2×2 块。

C = cov(___ ,w) 为之前的任何语法指定归一化权重。如果 w = 0（默认值），则 C 按观测值数量 -1 实现归一化。w = 1 时，按观测值数量对它实现归一化。

C = cov(___ ,nanflag) 指定一个条件，用于在之前的任何语法的计算中忽略 NaN 值。例如，cov(A,'omitrows') 将忽略 A 的具有一个或多个 NaN 元素的所有行。

协方差代表了两个变量之间的是否同时偏离均值。正值，一个变量变大另一个变量也变大；负值，一个变量变大另一个变量变小，取0说明两个变量没有相关关系。

15. 协方差与相关系数

数学与统计学院

- 相关系数ρ=cov(X,Y)/sqrt(DX*DY)，函数 corrcoef

`R = corrcoef(A)` 返回 A 的相关系数的矩阵，其中 A 的列表示随机变量，行表示观测值。

`R = corrcoef(A,B)` 返回两个随机变量 A 和 B 之间的系数。

`[R,P] = corrcoef(___)` 返回相关系数的矩阵和 p 值矩阵，用于测试观测到的现象之间没有关系的假设（原假设）。此语法可与上述语法中的任何参数结合使用。如果 P 的非对角线元素小于显著性水平（默认值为 0.05），则 R 中的相应相关性被视为显著。如果 R 包含复数元素，则此语法无效。

`[R,P,RL,RU] = corrcoef(___)` 包括矩阵，这些矩阵包含每个系数的 95% 置信区间的下界和上界。如果 R 包含复数元素，则此语法无效。

`___ = corrcoef(___,Name,Value)` 在上述语法的基础上，通过一个或多个 Name,Value 对组参数指定其他选项以返回任意输出参数。例如，`corrcoef(A,'Alpha',0.1)` 指定 90% 置信区间，`corrcoef(A,'Rows','complete')` 省略 A 的包含一个或多个 NaN 值的所有行。

不仅表示线性相关的方向，还表示线性相关的程度，取值[-1,1]。通常情况下，当相关系数的绝对值大于2/sqrt(N)，N为样本点的数量时，我们认为线性关系是存在的。

15. 协方差与相关系数


```
>> gcov = cov(grades)  %计算协方差
gcov =
   76.0419   31.6219   26.5967   29.4115   25.1093
   31.6219   42.7982   23.4537   24.2608   21.0167
   26.5967   23.4537   55.2184   38.7902   27.6386
   29.4115   24.2608   38.7902   73.9821   29.8406
   25.1093   21.0167   27.6386   29.8406   27.6554
>> [R,P,RL,RU] = corrcoef(grades)  %计算相关系数
R =
    1.0000    0.5543    0.4104    0.3921    0.5475
    0.5543    1.0000    0.4825    0.4312    0.6109
    0.4104    0.4825    1.0000    0.6069    0.7073
    0.3921    0.4312    0.6069    1.0000    0.6597
    0.5475    0.6109    0.7073    0.6597    1.0000
```

```
P =
    1.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    1.0000    0.0000    0.0000    0.0000
    0.0000    0.0000    1.0000    0.0000    0.0000
    0.0000    0.0000    0.0000    1.0000    0.0000
    0.0000    0.0000    0.0000    0.0000    1.0000
RL =
    1.0000    0.4164    0.2496    0.2290    0.4084
    0.4164    1.0000    0.3319    0.2730    0.4847
    0.2496    0.3319    1.0000    0.4799    0.6047
    0.2290    0.2730    0.4799    1.0000    0.5449
    0.4084    0.4847    0.6047    0.5449    1.0000
RU =
    1.0000    0.6673    0.5493    0.5338    0.6618
    0.6673    1.0000    0.6090    0.5666    0.7122
    0.5493    0.6090    1.0000    0.7090    0.7868
    0.5338    0.5666    0.7090    1.0000    0.7502
    0.6618    0.7122    0.7868    0.7502    1.0000
```

默认情况下，矩阵 RL 和 RU 根据 95% 置信区间分别给出每个相关系数的下界和上界。可以通过指定 Alpha 的值来更改置信水平，该值定义百分比置信度 100*(1-Alpha)%。例如，使用 Alpha 值 0.01 来计算 99% 置信区间，这将反映在 RL 和 RU 边界中。99% 置信度的 RL 和 RU 中的系数边界定义的区间大于 95% 置信度所定义的区间，因为置信度越高，需要的可能相关性值范围越大。

16. 常见分布的期望和方差

函数名	调用形式	注 释
unifstat	[M,V]=unifstat (a, b)	均匀分布(连续)的期望和方差，M为期望，V为方差
unidstat	[M,V]=unidstat (n)	均匀分布（离散）的期望和方差
expstat	[M,V]=expstat (p, Lambda)	指数分布的期望和方差
normstat	[M,V]=normstat(mu,sigma)	正态分布的期望和方差
chi2stat	[M,V]=chi2stat (x, n)	卡方分布的期望和方差
tstat	[M,V]=tstat (n)	t分布的期望和方差
fstat	[M,V]=fstat (n_1, n_2)	F分布的期望和方差
gamstat	[M,V]=gamstat (a, b)	γ分布的期望和方差
betastat	[M,V]=betastat (a, b)	β分布的期望和方差
lognstat	[M,V]=lognstat (mu, sigma)	对数正态分布的期望和方差

16. 常见分布的期望和方差

函数名	调用形式	注 释
nbinstat	[M,V]=nbinstat (R, P)	负二项式分布的期望和方差
ncfstat	[M,V]=ncfstat (n_1, n_2, delta)	非中心F分布的期望和方差
nctstat	[M,V]=nctstat (n, delta)	非中心t分布的期望和方差
ncx2stat	[M,V]=ncx2stat (n, delta)	非中心卡方分布的期望和方差
raylstat	[M,V]=raylstat (b)	瑞利分布的期望和方差
wblstat	[M,V]=wblstat (a, b)	韦伯分布的期望和方差
binostat	[M,V]=binostat (n,p)	二项分布的期望和方差
geostat	[M,V]=geostat (p)	几何分布的期望和方差
hygestat	[M,V]=hygestat (M,K,N)	超几何分布的期望和方差
poisstat	[M,V]=poisstat (Lambda)	泊松分布的期望和方差

16. 常见分布的期望和方差


```
>>a = 1:6; b = 2.*a;
>>[M,V] = unifstat(a,b)  %均匀分布的期望和方差
M =
    1.5000    3.0000    4.5000    6.0000    7.5000    9.0000
V =
    0.0833    0.3333    0.7500    1.3333    2.0833    3.0000
>> mu = randi([20,30],1,5)
>> sigma = randi([2,5],1,5)
>> [M,V] = normstat(mu,sigma)  %正态分布的期望和方差
M =
    20    24    24    28    28
V =
     4     9     9    16    16
```

```
>>n = logspace(1,5,5)
>>[M,V] = binostat(n,1./n)   %二项分布的期望和方差
M =
     1     1     1     1     1
V =
    0.9000    0.9900    0.9990    0.9999    1.0000
>>[m,v] = binostat(n,1/2)
m =
           5          50         500        5000       50000
v =
   1.0e+04 *
    0.0003    0.0025    0.0250    0.2500    2.5000
```


01
IDEA
02
LEARNING
03
CREATIVE
INSPIRATION
BUSINESS
EDUCATION
PHOTO BOOK

感谢聆听