**第1章 概率论基本概念**

**🕮 1.1 随机事件&样本空间**

**🏵 1.1.1 引言**

随机试验满足3个条件：

|  |
| --- |
| 1) 可以在相同的条件下重复进行；  2) 每次试验可能结果不止一个，且能事先明确所有可能的结果；  3) 每次试验之前不能确定出现哪一个结果。 |

随机试验发生的结果称为随机事件，一般用A,B,C等大写字母表示。

**🏵 1.1.2 样本空间**

随机事件有两个事件较为特殊：在每次试验必然会发生的事件为必然事件(Ω)；必然不会发生的事件为不可能事件(Ø)。

在随机试验中可直接观察到的、最简单的不能再分解的结果称为基本事件。

由若干个基本事件组成的事件称为复合事件。

比如抛骰子结果为1的情况为基本事件；结果为偶数的情况为复合事件。

随机试验所有可能的结果组成的集合称为样本空间；样本空间的每一个元素称为样本点。样本点通常是基本事件，也可以是复合事件。

样本空间Ω = {样本点ω}

样本空间是必然事件；随机试验的任何一个随机事件都可以看出是样本空间的一个子集。

**例1**：将一枚硬币抛两次，正面为H，反面为T

样本空间Ω = {HH, HT, TH, TT}

**例2**：连续向一目标射击直至命中

令ω*i*表示前*i*-1次未能命中，第*i*次命中，*i*=1,2,3,...

样本空间Ω = {ω1, ω2, ω3,...}

**🏵 1.1.3 事件的关系与运算**

1) 包含关系：A⊂B

事件B包含事件A，事件A发生必然导致事件B发生。

特殊情况：如果A⊂B且B⊂A，则A=B，表示事件A与B相等。

2) 和事件(并)：A∪B，事件A与B至少有一个发生。

A1, A2, ..., An的和事件写为：

3) 积事件(交)：A∩B或AB，事件A与B同时发生。

类似，A1, A2, ..., An的积事件写为：

特殊情况：如果A∩B = Ø，表示事件A与B不会同时发生，称事件A与B是互不相容的或互斥的。

A1, A2, ..., An是随机试验的一组事件，样本空间为Ω。如果满足：

则A1, A2, ..., An是样本空间Ω的一个完备事件组。

4) 若A∩B=Ø且A∪B=Ω (A和B构成样本空间Ω的一个完备事件组)

称事件A与B互为逆事件或对立事件。每次试验，事件A、B必有一个发生，且仅有一个发生。

记为：

5) 差事件：A - B，事件A发生，B不发生。

**🏵 1.1.4 事件运算规律**

|  |
| --- |
| 1) 交换律：A∪B = B∪A；A∩B = B∩A  2) 结合律：(A∪B)∪C = A∪(B∪C)；(A∩B)∩C = A∩(B∩C)  3) 分配律：A∪(B∩C) = (A∪B)∩(A∪C)；A∩(B∪C) = (A∩B)∪(A∩C)  4) 德摩根律(对偶法则) |

**例1**：证明A - B = A - AB

**例2**：射击3次，令A*i* = {第*i*次命中了}, *i*=1,2,3；B*j* = {在3次中恰好命中*j*次}, *j*=0,1,2,3，用A*i*表示B0、B1、B3。

**🕮 1.2 概率的定义与性质**

**🏵 1.2.1 概率公理化定义**

对样本空间Ω每一个事件A，P(A)表示事件A的概率

|  |
| --- |
| 1) 0 ≤ P(A) ≤ 1 (非负性)  2) P(Ω) = 1 (规范性/正则性)  3) 可加性：若A1, A2, ...互不相容，则P(∪A*i*) = ΣP(A*i*) |

**🏵 1.2.2 概率的性质**

|  |
| --- |
| 1) P(Ø) = 0  2) 有限可加性，即上面的可加性也适用于有穷的情况：  P(A1∪A2∪...∪An) = P(A1) + P(A2) + ... + P(An)  3) 对任意事件A：  4) 若A⊂B，则P(A) ≤ P(B)且P(B - A) = P(B) - P(A)  5) 广义加法定理：对于任意两个事件A和B，P(A∪B) = P(A) + P(B) - P(AB) |

**🕮 1.3 古典概率模型**

**🏵 1.3.1 古典概型定义**

若随机试验E满足：1) 样本空间有限多个元素；2) 每个基本事件发生的可能性相同，这种试验称为古典概型(等可能概型)。

若事件A包含k (也称为有利场合数)个基本事件，n为Ω包含的基本事件总数，则P(A)=k/n

**例**：投掷两个骰子，设A={和不小于6}，B={点数相同}，求P(A)、、P(A∪B)、P(A∩B)。

|  |
| --- |
| 基本事件总数 |

古典概率计算要统计基本事件的总数，通常要用到排列组合。

**🏵 1.3.2 古典概型三个例子**

**例1**：盒子里有10个球，编号1,2,...,10，从中任取3个球。事件A：3个球中第二大的数字是4，求P(A)。

|  |
| --- |
| 总共取法：  3球最小数可以从1,2,3中选取：  中间数只能选4：  最大数可以从5,6,7,8,9,10中选取： |

**例2**：袋子中有a个黑球，b个白球，一个一个拿出来，第k (1 ≤ k ≤ a+b)个球是黑球的概率？

|  |
| --- |
| 1) 从排列角度考虑：  假设把球拿出来放到a+b个坑中，总共(a+b)!种放法；  第k个坑需要是黑球有a种取法，剩余a+b-1个坑随意放，有(a+b-1)!种；  有利场合数为a×(a+b-1)!  2) 从组合角度考虑：  从a+b个坑挑a个坑放黑球，剩余放白球：  第k个坑放黑球，再挑a-1个坑放黑球，剩余放白球，有利场合数： |

这是一个抽签的模型，抽签结果和先后顺序无关，保证公平性。

**例3**：将*n*个球随机放入*N* (*N* ≥ *n*)个坑中，求每个坑至多有一个球的概率(一个坑可以放多个球)。

|  |
| --- |
| 样本空间：每次都有*N*个坑选择，总共放*n*次，总共有*Nn*个放法。  有利场合数：*N*个坑选*n*个做全排列 |

类似问题，如假设一年365天，一个班*n* (*n* ≤ 365)个同学，求至少有两人生日相同的概率。

|  |
| --- |
| *def* cal(*n*, *N*):  assert 1 <= n <= N  p = 1  for i in range(n):  p \*= (N-i)/N  return 1-p  *def* main():  arr = [20, 23, 30, 40, 50, 64]  for i in arr:  print('{}\t{*:.4f*}'.format(i, cal(i, 365)))  if \_\_name\_\_ == '\_\_main\_\_':  main() |

结果：

|  |
| --- |
| 20 0.4114  23 0.5073  30 0.7063  40 0.8912  50 0.9704  64 0.9972 |

当一个班有64人时，至少有两人生日相同的概率已经很接近100%。

**🏵 1.3.3 几何概型**

古典概型的基本事件个数是有限的，对于无限的情况需要使用几何概型。

几何概型基本事件通常不可计数，只能通过一定的测度(如长度、面积、体积)的比值来表示。

**例**：(会面问题)两个人约定5点到6点在某地见面，先到者等待另一个人20min，如果过时即可离开，求两人见面的概率。

|  |
| --- |
| *x*, *y*表示两人到达时间，满足： 两人就能见面  其中*x*, *y*用相对时间表示都要满足：0 ≤ *x*, *y* ≤ 60 |

**🕮 1.4 条件概率**

**🏵 1.4.1 条件概率定义与性质**

**例**：一对夫妇有三个孩子。事件A：三个都是女孩的概率；事件B：已知有一个是女孩，剩余两个是女孩的概率。

|  |
| --- |
|  |

设两个事件A、B，P(B) > 0，事件B发生条件下事件A发生的条件概率记为：

试验的基本事件总数为n，A包含基本事件数为a，B为b (b>0)，AB为k。

如果B发生，新的基本事件总数为b，而A发生就是AB要发生：

**例1**：盒子里有编号为1到10的10个小球，从中有放回地取两次，两次号码分别为x和y。事件A：x=4；事件B：x+y=7。

求P(A|B)和P(B|A)。

|  |
| --- |
|  |

**例2**：证明

|  |
| --- |
|  |

条件概率本质上仍然是古典概型，也具有普通概率的性质。

|  |
| --- |
| 1) 0 ≤ P(A|B) ≤ 1  2) P(Ω|B) = 1 (Ω是必然事件)  3) 若A1, A2, ...互不相容，则： |

**🏵 1.4.2 乘法公式**

将条件概率公式改写得到：

称为乘法公式。其中P(B)或P(A)需要>0。

三个事件A、B、C，且P(AB)>0，则有：

**例**：对产品进行三种破坏性试验，产品没有通过第一种试验的概率是0.3；通过第一种没有通过第二种的概率是0.2；通过前两种没有通过第三种的概率是0.1。

事件A：没有通过这三种试验。求P(A)

|  |
| --- |
| 设A*i*为没有通过第*i*种试验 (*i*=1,2,3)  由题意可知： |

**🏵 1.4.3 全概率公式**

**例1**：有甲、乙两个箱子，甲中有编号1~15的红色卡片，乙中有编号1~10的白色卡片。求从甲、乙任选一个、再从中任选一张卡片，编号为偶数的概率。

|  |
| --- |
| 设事件A：抽到偶数；B1：抽到红色；B2：抽到白色  显然。B1、B2构成Ω的一个完备事件组。 |

对于复杂事件A，将样本空间Ω划分为多个不相容的事件B1, B2, ..., Bn (构成一个完备事件组，A发生的原因或途径)，则：

该公式就是全概率公式。

**例2**：一批产品50%来自甲厂、25%来自乙厂、25%来自丙厂；甲乙丙次品率分别为2%、2%、4%，求所有产品中任取一个是次品的概率。

|  |
| --- |
| 设A为抽到次品；B1为抽到产品来自甲厂；B2为抽到乙；B3为抽到丙 |

**🏵 1.4.4 贝叶斯公式**

根据条件概率和全概率公式可得：

该公式为贝叶斯公式。

B*i*是诸多原因或途径，P(B*i*)称为先验概率，反应各种原因或途径发生可能性大小，实际中通常是经验的总结或归纳。

P(B*i*|A)称为后验概率，反应在A试验做完后，对各个概率的重新认识。也就是从结果来探讨不同原因的可能性大小。

**例**：用某个方法诊断肺癌，事件A：被检验者患有肺癌，B：结果为阳性。已知

，求P(A|B)。

|  |
| --- |
| 题意：人群中患肺癌概率是0.005。患肺癌检出阳性、没肺癌检出阴性的概率都是0.95，说明准确率比较高。  由贝叶斯公式得：  结果检出阳性而确实患肺癌的概率只有8.7%。 |

**🕮 1.5 独立性**

**🏵 1.5.1 两个事件的独立性**

A，B是试验E的两个事件，一般而言B的发生对A有影响，即P(A|B)≠P(A)。如果满足P(A|B)=P(A)，即：

称事件A、B(相互)独立。

**注**：事件A、B相互独立和A、B互不相容不能同时成立。

推论：若A、B独立，则 也相互独立。

**例**：机器甲的次品率是0.05，乙的次品率是0.04.从甲乙各取一件，求：

1) 两件都是次品的概率p1；

2) 至少一件次品的概率p2；

3) 恰好一件次品的概率p3.

|  |
| --- |
| 设事件A：抽到甲的是次品；B：抽到乙的是次品。 |

**🏵 1.5.2 多个事件的独立性**

三个事件的ABC如果满足：

|  |
| --- |
| P(AB) = P(A)P(B)  P(BC) = P(B)P(C)  P(AC) = P(A)P(C)  P(ABC) = P(A)P(B)P(C) |

则事件ABC相互独立。

对*n* (*n*≥2)个事件A1, A2, ..., An，如果对任意2,3, ..., *n*个事件的积事件概率都等于各事件概率之积，则成这*n*个事件相互独立。

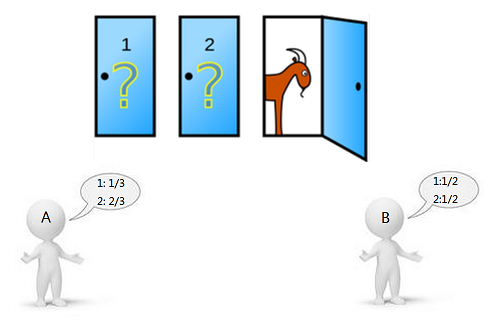
**例：**掷两个骰子，事件A：第1个骰子是奇数；B：第2个为奇数；C：两者之和为奇数。分析三个事件的独立性。

|  |
| --- |
| 说明A、B、C两两独立，并不是相互独立。 |

**🕮 1.6 蒙提霍尔三门问题**

游戏规则：

|  |
| --- |
| 1) 有三扇关闭的门，其中一扇后面有一辆汽车，另外两扇门后面各藏有一只山羊。选中后面有车的那扇门可赢得该汽车。  2) 当参赛者选定了一扇门，但未去开启它的时候，知道门后情形的节目主持人开启剩下两扇门的其中一扇，露出其中一只山羊。  3) 主持人其后会问参赛者要不要换另一扇仍然关上的门。 |



问题是：换另一扇门会否增加参赛者赢得汽车的机率？

该问题也称为蒙特霍尔悖论。因为很具迷惑性，违反大多人的直觉。

如果严格按照上述的条件，即主持人清楚地知道，自己打开的那扇门后是羊，那么换门赢得汽车的几率是2/3；不换门赢得汽车的几率是1/3。

|  |
| --- |
| from random import randint  *def* monty\_hall(*num*, *is\_change*):  car = randint(1, 3)  # 选中但是改变, 或者没选中且不改变, 则没有得到汽车  if (car == num and is\_change) or (car != num and not is\_change):  return False  return True  *def* test(*flag*):  # flag: 0表示不改变,1表示改变,其他表示随机  n = 100000 # 运行次数  win = 0  for i in range(n):  num = randint(1, 3)  is\_change = flag if flag in [0, 1] else randint(0, 1)  win += 1 if monty\_hall(num, is\_change) else 0  return win/n  *def* main():  dct = {0: '不改变', 1: '改变', 2: '随机'}  for k, v in dct.items():  print('{}: {*:.4f*}'.format(v, test(k)))  if \_\_name\_\_ == '\_\_main\_\_':  main() |

结果：

|  |
| --- |
| 不改变: 0.3341  改变: 0.6654  随机: 0.5034 |

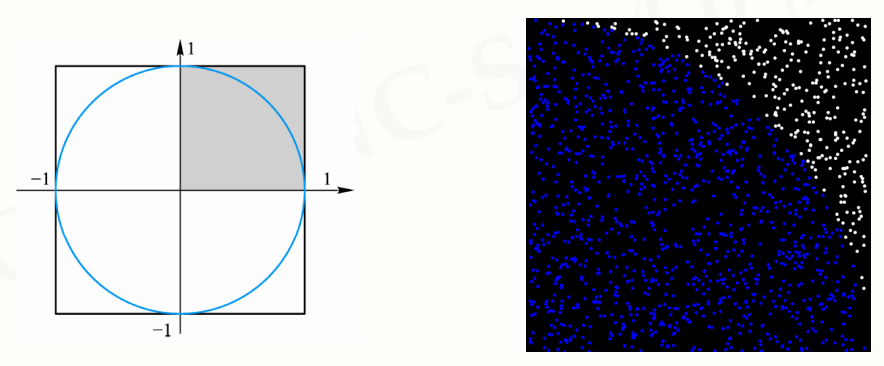
使用简单的Python程序可以帮助理解较为迷惑性的条件概率问题。

**🕮 1.7 蒙特卡罗方法初步**

也称统计模拟方法，使用(伪)随机数解决很多计算问题。

比如计算圆周率，在边长为1的正方形区域随机撒点*n*个，距离圆心≤1的点个数为*m*。

撒点个数越多，得到的结果越精确。



|  |
| --- |
| from random import random  from time import perf\_counter  *def* cal\_pi(): # 蒙特卡罗方法  darts = 1000\*1000 # 随机撒点个数  hits = 0 # 记录落在扇形中的点数  start = perf\_counter()  for i in range(darts):  # 随机撒点  x, y = random(), random()  distance = (x\*\*2+y\*\*2)\*\*0.5 # 该点到圆心距离  if distance <= 1:  hits += 1  pi = hits/darts\*4  end = perf\_counter()  print('运行时间: {*:.2f*}s'.format(end-start))  return pi  if \_\_name\_\_ == '\_\_main\_\_':  print(cal\_pi()) |

结果：

|  |
| --- |
| 运行时间: 1.04s  3.143004 |

**🕮 1.8 随机测试初步**

某个应用需要一个计算多项式的程序，两个攻城狮实现了两个程序F(*x*)和G(*x*)，问如何验证F(*x*)=G(*x*)?

假设它们最高阶(*x*的最高次)为*d*，随机算法首先从[1, 100*d*]均匀随机选择一个整数*t*进行测试。

如果F(*t*)=G(*t*)，判断两个程序相同，但F(*x*)≠G(*x*)，此时产生误判。

也就是*t*是F(*x*)-G(*x*)=0的根时，产生误判。

因为F(*x*)-G(*x*)最高次不大于*d*，F(*x*)-G(*x*)=0最多*d*个根。

误判概率为：

如何降低误判概率？

|  |
| --- |
| ① 增加数据范围，如从[1, 1000*d*]随机选择一个整数*t*进行测试，此时误判概率为1/1000；  ② 增加测试强度，每次从[1, 100*d*]随机选择一个整数*t*，重复*k*次随机测试：  1)当F(*t*)=G(*t*)，选择新的*t*进行下一轮测试；  2)当F(*t*) ≠G(*t*)，结束测试。 |

因为每次测试都是独立事件，则*k*次测试的误判为1/100*k*(呈指数级降低)。

抽样方法分为有放回抽样和无放回抽样。之前讨论的是有放回抽样。

使用无放回抽样的误判概率：

随机测试使用无放回抽样的误判概率比有放回抽样低，但有放回抽样的实现更简单和实用。

当*k*>*d*次随机测试，能确保误判概率为0；随机测试的总复杂度为O(*d*2)(每一轮为O(*d*))。但如果*d*特别大，这样做不合算。

**第2章 随机变量及其分布**

**🕮 2.1 随机变量**

随机变量取值随机会而定，是试验结果的函数。其反面是确定性变量。

随机试验样本空间为Ω = {ω}，X=X(ω)是定义在样本空间Ω上的实值单值函数，称X=X(ω)为随机变量。

示例：

① 掷一枚硬币：

② 检测m件产品，次品数X = 0, 1, 2, ..., m

③ 人的身高：。近似处理方法，离散问题当作连续问题考虑。

④ 射击击中位置：以靶心为圆点，靶半径为r：X=(x, y), x≤r, y≤r

**🕮 2.2 离散型随机变量**

如果随机变量的可能取到的值有限多个或可列无限多个，称为离散型随机变量。

离散型随机变量X，其所有可能取值为*ai*，X取到*ai*的概率：

P(X = *ai*) = *pi*, *i* = 1, 2, 3, ...

如果满足：

1)

2)

则称离散型随机变量X的分布列或分布律。使用表格的形式表示更加直观：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| X | *a1* | *a2* | ... | *an* | ... |
| *pi* | *p*1 | *p*2 | ... | *p*n | ... |

**🕮 2.3 常用三种离散型随机分布**

重复独立试验：每次试验相互独立且相应概率保持不变。

贝努里(Bernoulli)试验：只有两个结果的重复独立试验。

如，则

试验重复独立进行*n*次，称为n重贝努里试验。

① 两点分布

只进行1次贝努里试验：

|  |  |  |
| --- | --- | --- |
| X | 0 | 1 |
| *pi* | 1 - *p* | *p* |

称X服从以*p*为参数的两点分布或(0-1)分布。

常见例子为抛硬币、性别统计、检查产品质量是否合格等。

② 二项分布

进行*n*重贝努里试验，事件A出现*k*次的概率。也就是*k*次试验A事件发生，

*n - k*次试验A没有发生。

比如前*k*次试验A发生，后*n - k*次A没有发生的概率为：

*n*次试验选择*k*次表示A发生，总共取法是种。

因为它们两两互不相容，所以*n*次试验A发生*k*次的概率为

记，有：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| X | 0 | 1 | ... | k | ... | n |
| *pk* |  |  | ... |  | ... |  |

是二项式的展开式的一项。

所以：

满足离散型随机分布要求。

由于是二项展开式的形式，故称为二项分布。

称随机变量X服从参数为*n*, p的二项分布，记为X~*b*(*n*, *p*)；

每一项记为：

使b(k; n, p)取最大值的项b(m; n, p)称为中心项；m为最可能成功的次数。

**例1**：大批元件有10%是次品，抽取20个，都是合格的概率。

|  |
| --- |
| 不是放回抽样，但是因为总数很大，抽取的数目相对而言很少，所以可以近似认为是有放回抽样。相当于做了n=20重贝努里试验。 |

**例2**：一次抽卡抽到UR的概率0.01，500次单抽，最可能抽到多少张UR，并求相应的概率。

|  |
| --- |
| 最可能抽到5张UR，概率为： |

③ 泊松(Poisson)分布

随机变量X所有可能取值为0,1,2,...(非负整数)，各取值概率为：

其中λ>0是常数，称X服从参数为λ的泊松分布，记为。

根据泰勒级数：

得：

泊松分布可以作为二项分布的近似(逼近)。

泊松定理：λ>0是常数，*n*是任意正整数，设*npn*=λ，则对于任一固定非负整数*k*，有：

也就是当n很大、p很小时，以n, p为参数的二项分布可近似于以λ=np的泊松分布。

**例**：某个产品次品率为0.1%，求1000个产品中至少有2个是次品的概率。

|  |
| --- |
| 使用二项分布计算  使用泊松分布近似，λ=1 |

当n≥20、p≤0.05时，使用泊松分布作为二项分布的近似，效果颇佳。

**🕮 2.4 分赌本问题**

|  |
| --- |
| 1654年，一个职业赌徒向法国数学家帕斯卡提出了一个令他苦恼已久的分赌本问题：  甲乙两赌徒赌技相同，各出赌注100法郎，每局无平局。约定谁先赢3局则得到全部200法郎的赌本。  当甲赢了2局，乙赢了1局时，因故中止赌博，问这200法郎怎么分才算公平？  赌徒的不同理解：  1) 谁都没赢，各得100法郎；2) 甲得2/3，乙得1/3。 |

帕斯卡与法国数学家费马就此问题展开了信件♂讨论。

假设比赛继续，如果乙要赢，需要赢下第4、5局，概率为1/4；而甲要赢，只需赢第4局或输了第4局而赢下第5局，概率为1/2+1/4=3/4。

所以甲赢的概率3/4，分得150法郎；乙赢的概率1/4，分得50法郎。

**🕮 2.5 分布函数**

对于非离散型(连续型)随机变量：

|  |
| --- |
| 1) 其可能取值不能一一列举，因而不能使用分布律描述。  2) 连续型随机变量在任一点处的概率都是0。  3) 更感兴趣的是随机变量在某个区间的概率，如取值落在区间(*x*1,*x*2]的概率P{*x*1<X≤*x*2}。 |

设X是一个随机变量，*x*是任意实数，

函数F(*x*)称为X的分布函数。

对于任意实数x1,x2(x1<x2)，有：

分布函数完整地描述了随机变量的统计规律性。通过分布函数，可以使用数学分析(微积分)的方法研究随机变量。

分布函数F(*x*)的性质：

|  |
| --- |
| 1) F(*x*)单调不减；  2) ，且  3) F(*x*+0) = F(*x*)，即F(*x*)是右连续的。 |

**例1**：将一枚硬币连续抛两次，X表示正面出现的次数，求X的分布函数F(*x*)及P{0<X≤1}和P{1≤X≤2}。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| X的分布律为：   |  |  |  |  | | --- | --- | --- | --- | | X | 0 | 1 | 2 | | p |  |  |  |   分布函数为：  F(*x*)是一个阶梯型的曲线：    在0,1,2处有跳跃点。 |

**例2**：某脉冲信号在时间区间(0, T)内随机出现，出现时刻记为t。事件{t1 < t ≤ t2}的概率为：

求X(t)=t的分布函数。

|  |
| --- |
| F(t)处处连续 |

**🕮 2.6 连续型随机变量及其概率密度函数**

设随机变量X的分布函数为F(*x*)，如果存在非负函数*f* (*x*)，对任意实数*x*有：

则称X为连续型随机变量，*f* (*x*)为X的概率密度函数。

概率密度函数*f* (*x*)性质：

|  |
| --- |
| 1) *f* (*x*) ≥ 0  2)  3) 对于任意实数*x*1, *x*2(*x*1 ≤ *x*2)：  4) 若*f* (*x*)在点*x*处连续，则F'(*x*) = *f* (*x*)  5) 对于连续型随机变量X，取任意实数值a的概率都是0  也就是若事件A是不可能事件，则P(A)=0；反之若P(A)=0，并不一定表示A是不可能事件。 |

约定：随机变量的概率分布，离散型→分布律；连续型→概率密度函数。

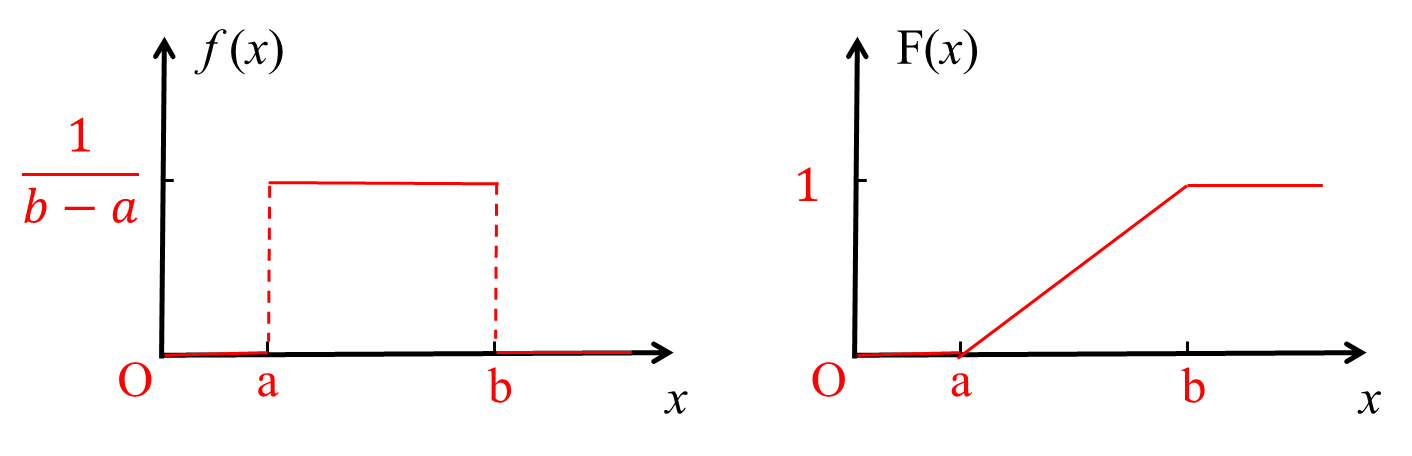
**🕮 2.7 常用的三种连续型随机分布**

① 均匀分布

若连续型随机变量X的概率密度满足：

则称X在区间(*a*, *b*)上服从均匀分布，记为X~U(*a*, *b*)

所以，X的分布函数为：



② 指数分布

若连续型随机变量X的概率密度满足：

或

其中λ>0 (λθ=1)为常数，则称X服从参数λ的指数分布。

当*x*<0时，

当*x*≥0时，

所以，X的分布函数为：

指数分布的一个有趣的性质：对任意*s*, *t* > 0，有：

该性质称为无记忆性。比如X指某个产品的寿命，如果使用了s小时，在此条件下再使用t小时的概率和从头算能使用t小时的概率相等，也就是产品对已经使用过s小时没有记忆。所以，指数分布也称寿命分布。

③ **正态分布**

若连续型随机变量X的概率密度满足：

其中μ, σ (σ>0)为常数，则称X服从参数为μ, σ的正态分布或高斯(Gauss)分布，记为X~N(μ, σ2)。

使用广义二重积分和极坐标变换成累次积分可得：

**正态分布性质**：

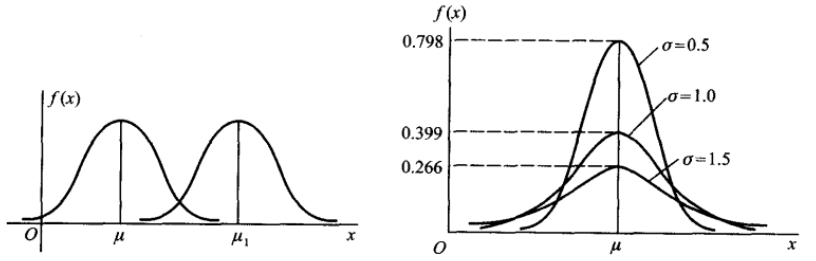
1) *f* (*x*)曲线关于*x*=μ对称。

2) 当*x*=μ时，*f* (*x*)取得最大值：

*x*离μ越远，*f* (*x*)越小。在处曲线有拐点；曲线以*x*轴为渐近线。

μ决定*f* (*x*)的位置，而不影响形状，μ称为位置参数。

σ越小，图形变得越尖，X落在μ附近的概率越大。



3) 当μ=0, σ=1时，随机变量X服从标准正态分布，X~N(0,1)。其概率密度和分布函数用φ(*x*)、Φ(*x*)表示。

人们已经编制了Φ(*x*)的函数表，可供查找。

4) 对于一般X~N(μ, σ2)，可以做线性变换转化为标准正态分布。

证明：

|  |
| --- |
|  |

X的分布函数F(*x*)写为：

5) 3σ法则

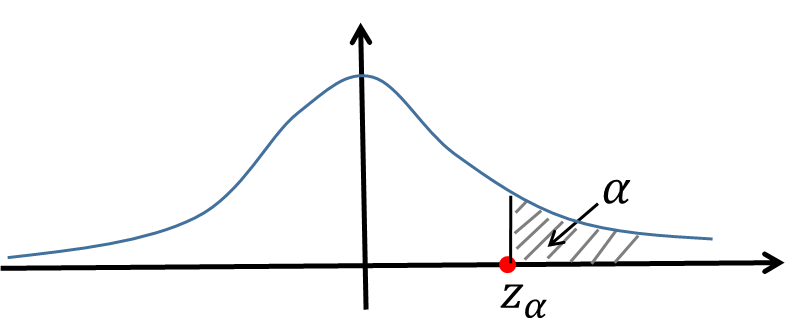
其值非常接近1，所以认为取值落在内几乎是肯定的。

**例**：将温度调节器放在有某种液体的容器内，调节器定在d℃，液体温度X ~ N(d, 0.52)。(1)若d=90℃，求X小于89℃的概率。(2)若要求液体的温度至少为80℃的概率不低于0.99，d至少为多少？

|  |
| --- |
| 1) 线性变换转为标准正态分布：  查表的  所以  2) 由题意得  查表得 |

设X~N(0,1)，若满足：

则称点为标准正态分布的上α分位点。



由对称性可知：

生活中如男性成年人的身高、测量零件的误差、海洋波浪的高度等都服从正态分布。正态随机变量在概率论与数理统计理论和实际应用中都有着重要的作用。

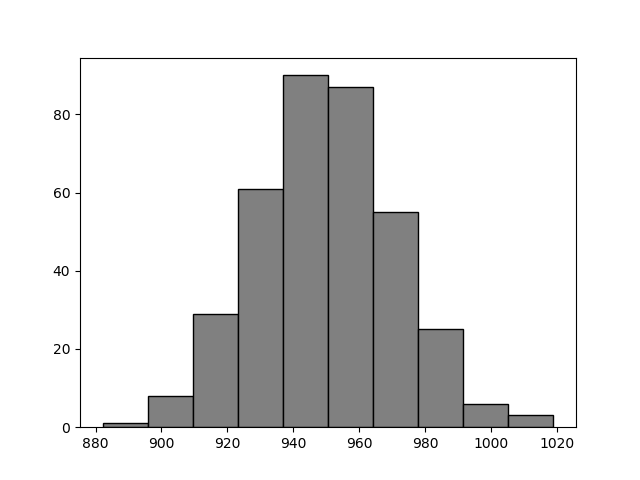
**🕮 2.8 庞加莱买面包问题**

|  |
| --- |
| 一个叫庞加莱的帅哥每天买1kg的面包，都回家称量并做记录；他发现一年平均质量是0.95kg；于是他认为面包店缺斤少两，投诉了该面包店。  面包店的策略是：叮嘱店员，每次都给庞加莱最大的。  一年后，庞加莱又来投诉面包店还是缺斤少两，欺骗百姓，只不过每次都给自己大的面包。庞加莱是如何知道的？ |

|  |
| --- |
| import numpy  import scipy.stats as sta  import matplotlib.pyplot as plt  import random  *def* buy\_bread(*f*, *pic\_name*):  X = sta.norm(*loc*=950, *scale*=20) # 平均值950,标准差20的正态随机变量X  lst = []  for i in range(365):  x = X.rvs(*size*=100) # 面包店一天生产100个面包  lst.append(f(x)) # 根据函数f挑选一个  print(*f*'平均值: {numpy.mean(lst)}') # 一年平均值  print(*f*'偏度: {sta.skew(lst)}') # 偏度  plt.hist(lst, *color*='grey', *edgecolor*='black') # 直方图  plt.savefig(*f*'{pic\_name}.png', *format*='png') # 存储为图片  *def* main():  # 两个一起执行第2张图会有问题?  buy\_bread(*f*=random.choice, *pic\_name*='first\_year')  # buy\_bread(f=max, pic\_name='second\_year')  if \_\_name\_\_ == '\_\_main\_\_':  main() |

执行第1年结果：

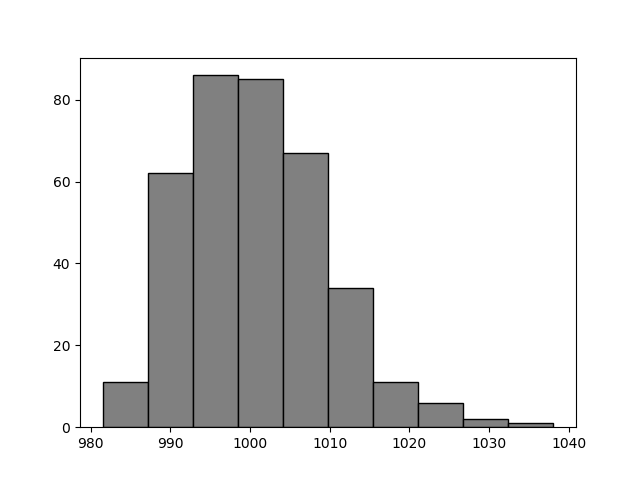
|  |
| --- |
| 平均值: 949.5629805482257  偏度: 0.07426643059780352 |



偏度越小，数据分布越对称。

执行第2年结果：

|  |
| --- |
| 平均值: 1000.7355805031303  偏度: 0.6751619291306432 |



虽然平均买到1000，但是根据是否对称(数据正偏)，判断每次拿到都是大的。

**🕮 2.9 随机变量的函数的分布**

已知随机变量X，Y=*f* (X)，求Y的分布。

比如测量一个圆的半径为*r*，但是更关心圆的面积

**例1**：已知离散型随机变量X的分布律：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | -2 | 0 | 1 | 2 |
| p | 0.1 | 0.2 | 0.3 | 0.4 |

求的分布律。

|  |  |  |  |
| --- | --- | --- | --- |
| Y | 1 | 2 | 5 |
| p | 0.2 | 0.3 | 0.5 |

**例2**：已知连续型随机变量X的概率密度：

求的概率密度。

|  |
| --- |
| 1) 求Y的分布函数G(y)  2) 求Y的概率密度g(y) |

设X的概率密度是*f* (*x*)，*a* < *x* < *b*，*y* = *g*(*x*)在(*a*, *b*)严格单调连续，且存在唯一的反函数*x* = *h*(*y*), α < *y* < β，且*h*'(*y*)连续，则Y = *g*(X)也是连续型随机变量，其概率密度是：

**例**：设)，求Y = tanX的概率密度。

|  |
| --- |
| X服从均匀分布，其概率密度为：  反函数  这是柯西(Cauchy)分布 |

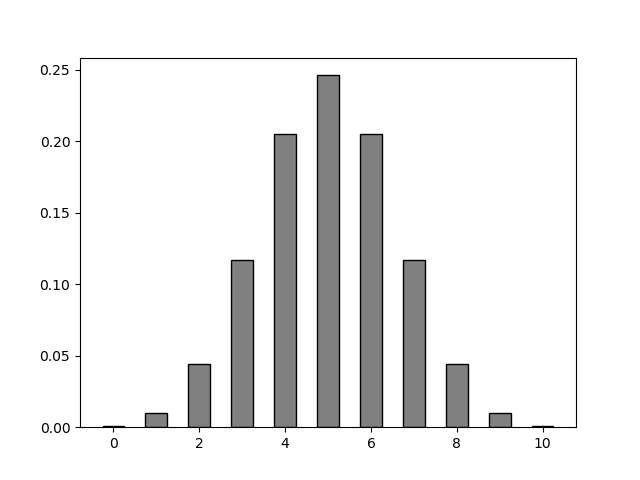
**🕮 2.10 概率分布的Python实现**

① 二项分布X~*b*(*n*, *p*)

|  |
| --- |
| import numpy as np  import matplotlib.pyplot as plt  from scipy.stats import binom  *def* b(*n*, *p*): # 二项分布  rv = binom(n, p)  x = np.arange(n+1)  y = rv.pmf(x)  print(y)  plt.bar(x, y, *width*=0.5, *color*='grey', *edgecolor*='black')  plt.savefig('binom.png', *format*='png')  if \_\_name\_\_ == '\_\_main\_\_':  b(10, 0.5) |

结果：

|  |
| --- |
| [0.00097656 0.00976563 0.04394531 0.1171875 0.20507813 0.24609375  0.20507813 0.1171875 0.04394531 0.00976563 0.00097656] |



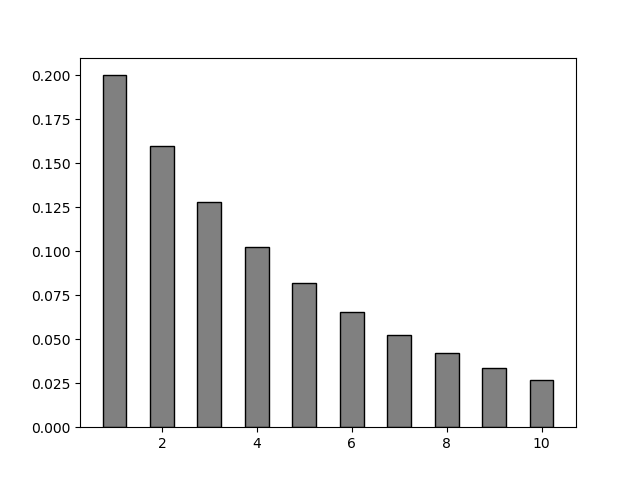
② 几何分布

在n次贝努里试验中，试验k次才第一次成功的机率。也就是：前k-1次都失败但第k次成功的概率。

|  |
| --- |
| from scipy.stats import geom  *def* g(*p*): # 几何分布  N = 10 # 只画前N次  rv = geom(p)  x = np.arange(1, N+1)  y = rv.pmf(x)  print(y)  plt.bar(x, y, *width*=0.5, *color*='grey', *edgecolor*='black')  plt.savefig('geom.png', *format*='png')  if \_\_name\_\_ == '\_\_main\_\_':  g(0.2) |

结果：

|  |
| --- |
| [0.2 0.16 0.128 0.1024 0.08192 0.065536  0.0524288 0.04194304 0.03355443 0.02684355] |

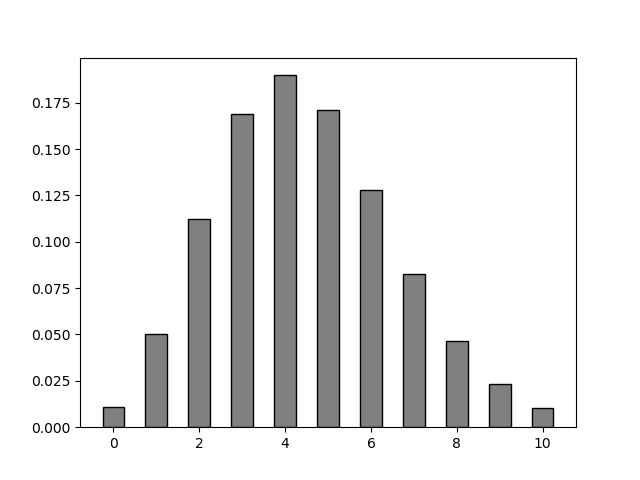


③ 泊松分布

|  |
| --- |
| from scipy.stats import poisson  *def* pi(*lmd*): # 泊松分布  N = 10 # 只画前N次  rv = poisson(lmd)  x = np.arange(N+1)  y = rv.pmf(x)  print(y)  plt.bar(x, y, *width*=0.5, *color*='grey', *edgecolor*='black')  plt.savefig('poisson.png', *format*='png')  if \_\_name\_\_ == '\_\_main\_\_':  pi(4.5) |

结果：

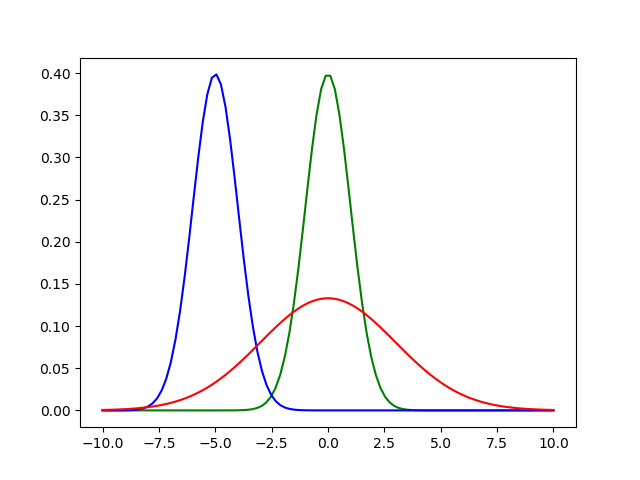
|  |
| --- |
| [0.011109 0.04999048 0.11247859 0.16871788 0.18980762 0.17082686  0.12812014 0.08236295 0.04632916 0.02316458 0.01042406] |



④ 正态分布X~N(μ, σ2)

|  |
| --- |
| from scipy.stats import norm  *def* n(): # 正态分布  x = np.linspace(-10, 10, 100)  #loc相当于μ; scale相当于σ  rv1 = norm(*loc*=0, *scale*=1)  rv2 = norm(*loc*=-5, *scale*=1)  rv3 = norm(*loc*=0, *scale*=3)  # 3个正态分布的叠加图  plt.plot(x, rv1.pdf(x), *color*='green')  plt.plot(x, rv2.pdf(x), *color*='blue')  plt.plot(x, rv3.pdf(x), *color*='red')  plt.savefig('norm.png', *format*='png') |

结果：

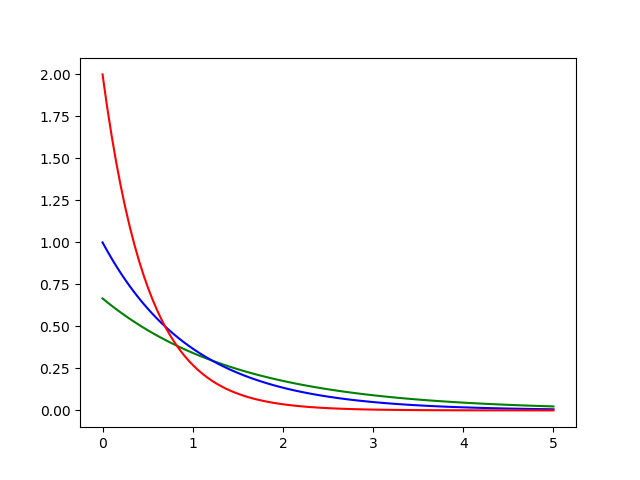


⑤ 指数分布

或

|  |
| --- |
| from scipy.stats import expon  *def* e(): # 指数分布  x = np.linspace(0, 5, 100)  # scale相当于1/λ或θ  rv1 = expon(*scale*=1.5)  rv2 = expon(*scale*=1)  rv3 = expon(*scale*=0.5)  # 3个指数分布的叠加图  plt.plot(x, rv1.pdf(x), *color*='green')  plt.plot(x, rv2.pdf(x), *color*='blue')  plt.plot(x, rv3.pdf(x), *color*='red')  plt.savefig('expon.png', *format*='png') |

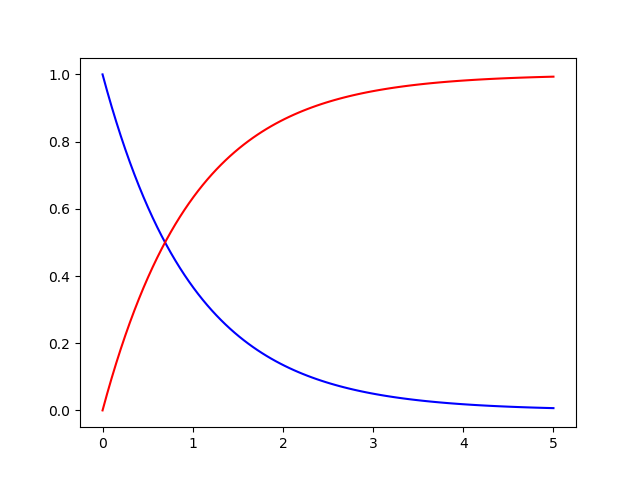
结果：



指数分布的分布函数：

|  |
| --- |
| *def* compare\_pdf\_cdf():  x = np.linspace(0, 5, 100)  rv = expon(*scale*=1)  # pdf为概率密度,cdf为累计概率,就是分布函数  plt.plot(x, rv.pdf(x), *color*='blue')  plt.plot(x, rv.cdf(x), *color*='red')  plt.savefig('指数分布概率密度和分布函数.png', *format*='png') |

结果：



**第3章 二维随机变量及其分布**

**🕮 3.1 二维随机变量**

**🏵 3.1.1 二维随机变量的分布函数**

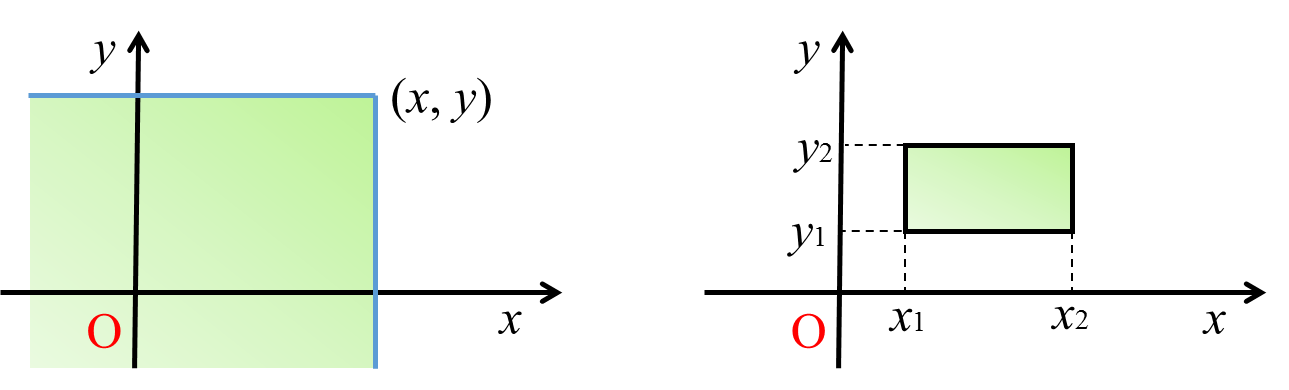
设随机事件E的样本空间是S={*e*}，X=X(*e*)和Y=Y(*e*)是定义在S上的随机变量，向量(X, Y)叫做二维随机向量或二维随机变量。

对于二维随机变量(X, Y)的分布函数：

称为随机变量X和Y的联合分布函数。

如果将(X, Y)看成平面上的随机点的坐标，分布函数F(*x*, *y*)的值就是随机点落在如图以点(*x*, *y*)为顶点是左下方无穷矩形区域内的概率。

由此可得随机点(X, Y)落在矩形区域的概率是：



联合分布函数F(*x*, *y*)的性质：

|  |
| --- |
| 1) F(*x*, *y*)是变量*x*和*y*的不减函数。如对固定的*y*当时  2) ；；  3) 右连续性：  4)  因为上式左边就是的值，由概率的非负性即得。 |

**🏵 3.1.2 二维离散型和连续型随机变量**

① 二维离散型

若(X, Y)取值有限多对或可列无限多对，则(X, Y)是离散型随机变量。

所有可能取值记为：

满足：

*p*ij称为二维离散型随机变量(X, Y)的分布律，或随机变量X和Y的联合分布律。

② 二维连续型

设(X, Y)的分布函数F(*x*, *y*)，如果存在非负函数*f* (*x*, *y*)使对任意*x*,*y*满足：

则(X, Y)是连续型二维随机变量，函数*f* (*x*, *y*)称为二维随机变量(X, Y)的概率密度，或随机变量X和Y的联合概率密度。

概率密度*f* (*x*, *y*)的性质：

|  |
| --- |
| 1)  2)  3) 点(X, Y)落在平面区域D的概率：  4) 若在点连续，则有： |

二维随机变量的情况，也适用于*n* (*n* > 2)维随机变量。

**🕮 3.2 边缘分布**

二维随机变量(X, Y)作为整体，具有分布函数，而X和Y各自也有分布函数，记为和，称为二维随机变量(X, Y)关于X和Y的边缘分布函数。

同理：

① 离散型

X的分布律：

记为并称其为关于X的边缘分布律。

同理Y的边缘分布律：

② 连续型

设连续型随机变量(X, Y)的概率密度是，关于X的边缘分布函数：

X是连续型随机变量，其概率密度是分布函数求导所得：

同理，Y的概率密度：

分别称和是(X, Y)关于X和Y的边缘概率密度。

**例**：袋中有2个编号为1的球，3个编号为0的球，有/无放回一个一个取出。

X为第1次取出的编号；Y为第2次取出的编号。求X和Y的联合分布律和各自的边缘分布律。

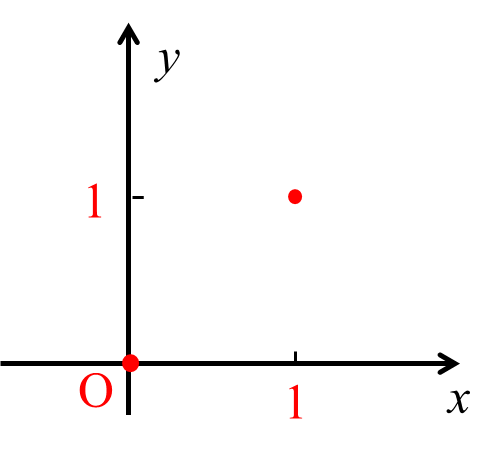
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1) 有放回   |  |  |  |  | | --- | --- | --- | --- | | Y X | 0 | 1 | P{Y=*yj*} | | 0 | 0.6×0.6 | 0.4×0.6 | 0.6 | | 1 | 0.6×0.4 | 0.4×0.4 | 0.4 | | P{X=*xi*} | 0.6 | 0.4 | 1 |   2) 无放回   |  |  |  |  | | --- | --- | --- | --- | | Y X | 0 | 1 | P{Y=*yj*} | | 0 | 0.6×0.5 | 0.4×0.75 | 0.6 | | 1 | 0.6×0.5 | 0.4×0.25 | 0.4 | | P{X=*xi*} | 0.6 | 0.4 | 1 |   常常将边缘分布律写在联合分布律表格的边缘上，这就是边缘分布律的由来。  可以看出两种情况对应边缘分布是一样的，但是联合分布不同。 |

所以，已知X和Y的联合分布可以确定边缘分布；但是已知X和Y的边缘分布，不能确定联合分布。

**🕮 3.3 常见的三种二维分布**

① 二维两点分布

|  |  |  |  |
| --- | --- | --- | --- |
| Y X | 0 | 1 |  |
| 0 | 1-p | 0 | 1-p |
| 1 | 0 | p | p |
|  | 1-p | p | 1 |



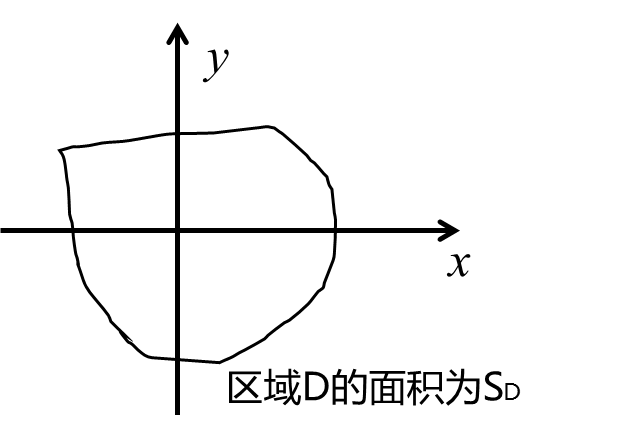
(X, Y)的联合分布函数F(*x*, *y*)：

1) 当x<0或y<0时，F(*x*, *y*) = 0；

2) 当0≤x<1,y≥0或0≤y<1,x≥0时，F(*x*, *y*) = 1-p；

3) 当x≥1,y≥1时，F(*x*, *y*) = 1

② 二维均匀分布



(X, Y)的联合概率密度*f* (*x*, *y*)：

**例**：在[-1,2]任取两数X和Y，求X+Y>1且XY<1的概率。

|  |
| --- |
| 根据几何概型： |

③ 二维正态分布

概率密度为：

**🕮 3.4 条件分布**

① (X, Y)是二维离散型随机变量

分布律：

，由条件概率公式得：

称为在Y=*yj*条件下随机变量X的条件分布律。

② (X, Y)是二维连续型随机变量

对于固定的*y*，若*f* Y(*y*)>0：

称为在Y=*y*条件下X的条件概率密度。

此条件下的条件分布函数为：

**例**：设(X, Y)的联合概率密度函数为：

求P{X>1|Y=y}。

|  |
| --- |
| 在Y=y>0时，X的条件概率密度是：  所求条件概率是条件概率密度的积分： |

**🕮 3.5 独立性**

设二维随机变量(X, Y)的分布函数是F(*x*, *y*)，边缘分布函数分别为FX(*x*)和FY(*y*)，对于所有*x*, *y*有：

即：

则称随机变量X和Y的相互独立的。

1) 当是离散型时有：

即：

2) 当是连续型时，概率密度有：

**例**：设X,Y独立同分布，X的概率密度是：

求P{X+Y≤1}

|  |
| --- |
| 因为独立，所以X,Y的联合概率密度为：  根据二重积分的累次积分方法： |

独立性的性质：

1) X与Y独立，则f(X)和g(Y)独立，f, g是任意连续函数。

2) 常数c与任意随机变量独立。

3) 若联合概率密度*f* (*x*, *y*)可分离成：

且g(x)的非0区域与y无关，h(y)的非0区域与x无关，则X, Y独立。

**🕮 3.6 两个随机变量的函数的分布**

**例1**：设X, Y独立，且服从泊松分布：

求X+Y的分布律。

|  |
| --- |
| 即： |

**例2**：最大值最小值分布

设X, Y独立，分布函数分别为FX(*x*)和FY(*y*)，求Z1=max(X, Y)，Z2=min(X, Y)的分布函数。

|  |
| --- |
|  |

**例3**：设(X, Y)的联合概率密度*f* (*x*, *y*)，求Z=X+Y的分布。

|  |
| --- |
| Z的分布函数为：  固定*z*与*y*，令*x* = *u* - *y*，则：  所以，Z的概率密度函数为：  也称为卷积公式。 |

同样方法也可以求出：如Z=X-Y、X·Y、X/Y等的分布。

若X, Y独立，且服从正态分布：

则。

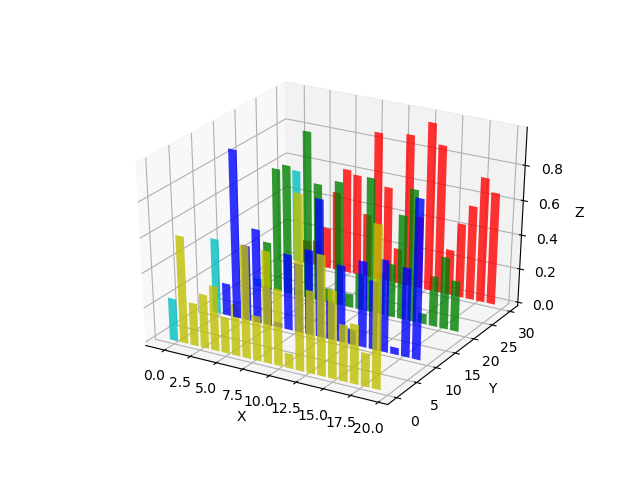
正态分布具有可加性：有限个相互独立的正态随机变量的线性组合仍然服从正态分布。

**🕮 3.7 二维随机变量的Python实现**

matplotlib官网mplot3d的[示例](https://matplotlib.org/examples/mplot3d/bars3d_demo.html)：

|  |
| --- |
| from mpl\_toolkits.mplot3d import Axes3D  import matplotlib.pyplot as plt  import numpy as np  *def* bars3d\_demo():  """  ========================================  Create 2D bar graphs in different planes  ========================================  Demonstrates making a 3D plot which has 2D bar graphs projected onto  planes y=0, y=1, etc.  """  fig = plt.figure()  ax = fig.add\_subplot(111, *projection*='3d')  for c, z in zip(['r', 'g', 'b', 'y'], [30, 20, 10, 0]):  xs = np.arange(20)  ys = np.random.rand(20)  # You can provide either a single color or an array. To demonstrate this,  # the first bar of each set will be colored cyan.  cs = [c] \* len(xs)  cs[0] = 'c'  ax.bar(xs, ys, *zs*=z, *zdir*='y', *color*=cs, *alpha*=0.8)  ax.set\_xlabel('X')  ax.set\_ylabel('Y')  ax.set\_zlabel('Z')  plt.savefig('bars3d\_demo.png', *format*='png')  if \_\_name\_\_ == '\_\_main\_\_':  bars3d\_demo() |

结果：



**例**：对一人群吸烟情况X和健康情况Y调查：

X=1不吸烟；X=2吸烟一般；X=3吸烟严重

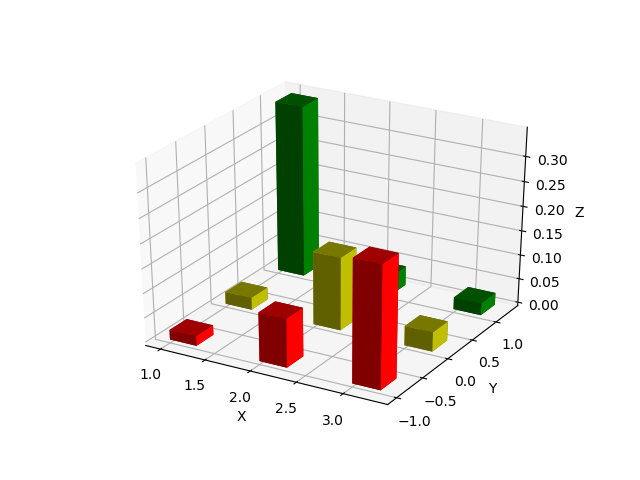
Y=-1不健康；Y=0一般；Y=1健康

(X,Y)的联合分布律为：

|  |  |  |  |
| --- | --- | --- | --- |
| Y X | 1 | 2 | 3 |
| -1 | 0.02 | 0.1 | 0.25 |
| 0 | 0.025 | 0.15 | 0.04 |
| 1 | 0.35 | 0.04 | 0.025 |

|  |
| --- |
| from mpl\_toolkits.mplot3d import Axes3D  import matplotlib.pyplot as plt  import numpy as np  *def* smoke\_demo():  # 3D柱状图  fig = plt.figure()  ax = fig.add\_subplot(111, *projection*='3d')  dx, dy = 0.3, 0.3 # 柱状底面是边长0.3的正方形  # z高度对应联合分布律的概率  dz = [0.02, 0.025, 0.35, 0.1, 0.15, 0.04, 0.25, 0.04, 0.025]  zpos = 0  i = 0  for xpos in range(1, 4):  for c, ypos in zip(['r', 'y', 'g'], [-1, 0, 1]):  ax.bar3d(xpos, ypos, zpos, dx, dy, dz[i], *color*=c)  i += 1  ax.set\_xlabel('X')  ax.set\_ylabel('Y')  ax.set\_zlabel('Z')  plt.savefig('smoke\_demo.png', *format*='png')  if \_\_name\_\_ == '\_\_main\_\_':  smoke\_demo() |

结果：



二维正态分布：bivariate\_normal()函数

// The bivariate\_normal function was deprecated in Matplotlib 2.2 and will be removed in 3.1

**第4章 随机变量的数字特征**

**🕮 4.1 数学期望**

**🏵 4.1.1 数学期望的定义**

① 离散型

设离散型随机变量X的分布律是：

定义E(X)为X的数学期望(简称期望，也称均值)：

其中级数需要**绝对收敛**。

|  |
| --- |
| 设有级数：  1) 若收敛，则称绝对收敛，绝对收敛级数一定收敛；  2) 若收敛，而发散，则称条件收敛。  黎曼定理：绝对收敛的级数可以任意交换次序。 |

因为随机试验的结果具有任意次序性，所以任意次序的数学期望应是一样的。对于可列无限多的级数必须收敛。

② 连续型

设连续型随机变量X的概率密度是*f* (*x*)，数学期望E(X)为：

其中广义积分需要**绝对收敛**，也就是需要收敛。

**🏵 4.1.2 常见三种离散型随机变量的期望**

① 两点分布

② 二项分布：X~B(n, p)

③ 泊松分布：

**🏵 4.1.3 常见三种连续型随机变量的期望**

① 均匀分布：X~U(*a*, *b*)

② 指数分布：X~EXP(λ)

③ 正态分布：X~N(μ, σ2)

令，则

根据广义二重积分和极坐标变换成累次积分，可得：

**🏵 4.1.4 随机变量的函数的期望**

设X分布已知，Y=g(X)，g是连续函数，求E(Y)。

1) 离散型

2) 连续型

也需要级数和广义积分绝对收敛。

数学期望的性质：

1) E(c) = c，c是常数；

2) E(cX) = cE(X)，c是常数；

**3)** 设有两个随机变量X和Y，则E(X±Y) = E(X) ± E(Y)，可以推广到有限个；

4) 若X、Y独立，则E(X·Y)=E(X)·E(Y)，可以推广到有限个(独立)；

**🕮 4.2 方差**

**🏵 4.2.1 方差的定义及其性质**

研究随机变量与均值的偏离程度，可使用E{|X-E(X)|}衡量。

但因为使用绝对值，计算不方便，通常取平方：E{(X-E(X))2}。

设随机变量X，若E{(X-E(X))2}存在，则称其为X的方差。

记为D(X)或Var(X)：

引入，称为标准差或均方差。

1) 离散型

2) 连续型

方差的性质：

1) D(c) = 0，c是常数；

2)

{}里面的E(X)是常数，可以提出来：

这是计算方差**最常用**的公式。

3) *a*,*b*是常数，有：

4) 两个随机变量X、Y，X+Y的方差：

当X与Y相互独立时，

**🏵 4.2.2 常见三种离散型随机变量的方差**

① 两点分布

② 二项分布：X~B(n, p)

将二项分布看成n个独立的两点分布X1,X2,X3...：

③ 泊松分布：

也就是泊松分布的期望和方差都是λ。

**🏵 4.2.3 常见三种连续型随机变量的方差**

设X的期望E(X)=μ，方差

即Y的数学期望为0，方差为1，Y称为X的标准化变量。

① 均匀分布：X~U(*a*, *b*)

② 指数分布：X~EXP(λ)

③ 正态分布：X~N(μ, σ2)

对于标准正态分布X~N(0,1)：

对于一般的正态分布Y，令：

所以正态分布的两个参数：代表数学期望，代表标准差，代表方差。

**🕮 4.3 协方差&相关系数**

二维随机变量(X, Y)，除了研究X和Y的数学期望和方差外，还需讨论描述X与Y之间相互关系的数字特征。

**🏵 4.3.1 协方差**

随机变量X与Y的协方差记为Cov(X, Y)：

P34：X±Y的方差可改写为：

协方差的性质：

|  |
| --- |
| 1) Cov(X, Y) = Cov(Y, X)  2) Cov(X, X) = D(X)  3) Cov(X, Y) = E(XY) - E(X)E(Y) // 计算协方差主要公式  4) 如果X、Y独立，则Cov(X, Y) = 0；反之不一定。  5) Cov(X1+X2, Y) = Cov(X1, Y) + Cov(X2, Y)  6) Cov(*a*X, *b*Y) = *ab*Cov(X, Y)，其中*a*、*b*是常数 |

**例**：(X, Y)的联合概率密度函数为：

求Cov(X, Y)。

|  |
| --- |
|  |

**🏵 4.3.2 相关系数**

随机变量X和Y的相关系数：

ρXY没有单位(量纲)，消除了单位的影响。

也就是ρXY是标准尺度下的协方差。

相关系数的性质：

|  |
| --- |
| 1) 若X、Y独立，则ρXY = 0  2)  若表示存在，使得 // 完全正相关  若表示存在，使得 // 完全负相关  若表示X与Y不相关  叫线性相关系数，反应X与Y直接的线性依赖程度。 |

**例1**：

|  |
| --- |
| ，说明X与Y不相关，但它们之间有非线性关系。 |

**例2**：抛硬币*n*次，正面出现次数X，反面出现次数Y。求。

|  |
| --- |
| ，表示完全的负相关。 |

**例3**：(X, Y)服从平面上一单位圆内的均匀分布，求及X与Y的独立性。

|  |
| --- |
| 同理：  ，代表X与Y不相关  因为不独立 |

对于二维正态分布

也就是二维正态分布的参数就是X和Y的相关系数。因此二维正态分布完全可由X、Y各自的数学期望、方差和它们的相关系数确定。

**🕮 4.4 矩&协方差矩阵**

**🏵 4.4.1 协方差矩阵**

设*n*维随机变量，任意两个分量的协方差：

都存在，则称矩阵：

是*n*维随机变量的协方差矩阵。

协方差矩阵的性质：

|  |
| --- |
| 1)  2) |

**🏵 4.4.2 矩**

随机变量X，若： (即存在)，则称：

是X的k阶原点矩，简称k阶矩。

若E(X)存在，且 (即存在)，则称：

是X的k阶中心矩。

数学期望E(X)是X的一阶原点矩；方差D(X)是X的二阶中心矩。

所以，矩是更具一般意义的数字特征。

**🕮 4.5 数字特征的Python实现**

① 二项分布

|  |
| --- |
| from scipy.stats import binom  rv = binom(10, 0.2) # 二项分布  print(*f*'数学期望: {rv.mean()}')  print(*f*'方差: {rv.var()}')  for i in range(1, 5):  print(*f*'{i}阶原点矩: {rv.moment(i)}')  print(rv.stats(*moments*='mvsk')) # 获取统计数据 |

结果：

|  |
| --- |
| 数学期望: 2.0  方差: 1.6  1阶原点矩: 2.0  2阶原点矩: 5.6  3阶原点矩: 18.560000000000002  4阶原点矩: 69.82400000000007  (array(2.), array(1.6), array(0.47434165), array(0.025)) |

② 泊松分布

|  |
| --- |
| from scipy.stats import poisson  rv = poisson(*mu*=5) # 泊松分布λ=5  print(*f*'数学期望: {rv.mean()}')  print(*f*'方差: {rv.var()}')  for i in range(1, 5):  print(*f*'{i}阶原点矩: {rv.moment(i)}')  print(rv.stats(*moments*='mvsk')) # 获取统计数据 |

结果：

|  |
| --- |
| 数学期望: 5.0  方差: 5.0  1阶原点矩: 5  2阶原点矩: 30  3阶原点矩: 205.0  4阶原点矩: 1555.0  (array(5.), array(5.), array(0.4472136), array(0.2)) |

scipy.stats内置[离散型](https://docs.scipy.org/doc/scipy/reference/stats.html#discrete-distributions)、[连续型](https://docs.scipy.org/doc/scipy/reference/stats.html#continuous-distributions)概率分布。只要更改分布，基本步骤类似。

③ 均匀分布

|  |
| --- |
| from scipy.stats import uniform  rv = uniform(*loc*=2, *scale*=6) # 均匀分布a=2,b=2+6=8  # ... |

④ 指数分布

|  |
| --- |
| from scipy.stats import expon  import numpy as np  from scipy import integrate  rv = expon(*scale*=2) # 指数分布θ=1/λ=2  print(*f*'数学期望: {rv.mean()}')  print(*f*'方差: {rv.var()}')  for i in range(1, 5):  print(*f*'{i}阶原点矩: {rv.moment(i)}')  print(rv.stats(*moments*='mvsk')) # 获取统计数据  # -----------------------------------------------  # 使用积分求指数分布的数学期望和方差  F1 = *lambda* *x*: x/2\*np.exp(-x/2)  F2 = *lambda* *x*: x\*\*2/2\*np.exp(-x/2)  # 积分结果是(积分,偏差)的元组  E1 = integrate.quad(F1, 0, np.inf) # 0到正无穷积分  E2 = integrate.quad(F2, 0, np.inf)  print(*f*'数学期望=1阶原点矩={E1[0]}')  print(*f*'方差=2阶原点矩-1阶原点矩的平方={E2[0]-E1[0]\*\*2}') |

结果：

|  |
| --- |
| 数学期望: 2.0  方差: 4.0  1阶原点矩: 2.0  2阶原点矩: 8.0  3阶原点矩: 48.0  4阶原点矩: 384.0  (array(2.), array(4.), array(2.), array(6.))  数学期望=1阶原点矩=1.9999999999999998  方差=2阶原点矩-1阶原点矩的平方=4.000000000000001 |

**第5章 大数定律和中心极限定理**

**🕮 5.1 概率统计常用的两种收敛性**

① 依概率收敛

设是随机变量序列，X是一随机变量或常数，对于任意

则称随机变量序列依概率收敛于X，记为：

② 依分布收敛

设是随机变量序列，是的分布函数；

X是一随机变量，F(X)是X的分布函数。

若在F(X)的连续点*x*处都有：

则称随机变量序列依分布收敛于X，记为：

这表明以X的分布为极限分布。

而微积分的收敛性：依距离收敛

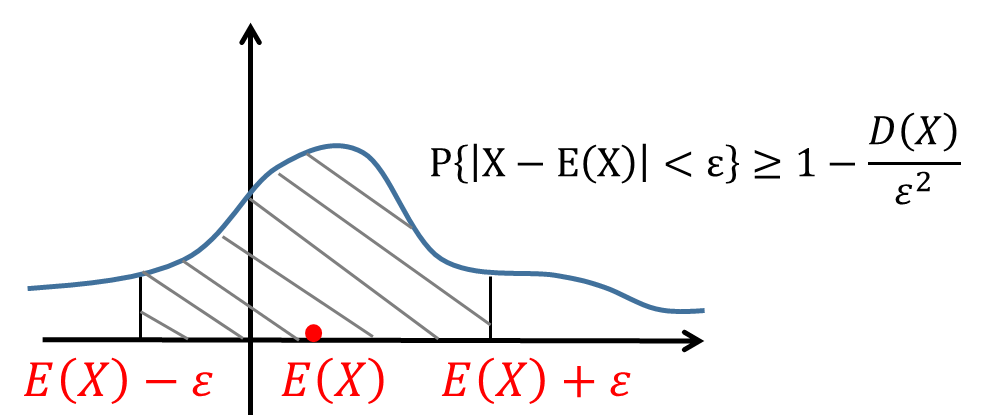
**🕮 5.2 切比雪夫不等式**

设随机变量X的E(X)和D(X)都存在，对于任意，有：

以连续型随机变量为例证明：

|  |
| --- |
|  |

切比雪夫不等式用于未知估计分布概率，如在以E(X)为中心的区间发生的概率。但这个估计是比较粗糙的。



**例1**：X的分布未知，但。试估计：

和

|  |
| --- |
|  |

**例2**：设是独立同分布的随机变量序列，有相同的数学期望和方差。

证明：

|  |
| --- |
| 代入切比雪夫不等式：  当时， |

此例的称为服从弱大数定律。

**🕮 5.3 大数定律**

随机现象的统计规律：在相同条件下进行大量重复试验才能显现出来。

如抛一枚硬币，虽然不能准确预测每次的结果，但随着次数增加，正面出现的概率趋于0.5。

设随机变量序列都存在，若对任意，有：

即：

则称服从大数定律。

这说明了算术平均值和频率(试验次数)的稳定性。

① 切比雪夫大数定律

是相互独立的随机变量序列，都存在，且**一致有界**，则服从大数定律。

一致有界：

② 独立同分布大数定律

是独立同分布的随机变量序列，，则服从大数定律。

② 贝努里大数定律

在*n*重贝努里试验中，A事件出现*m*次，，则对任意

也就是*n*充分大时，频率与概率的偏差小于，这就是频率的稳定性。实际应用中，当试验次数很大时，常使用事件的频率代替事件的概率。

如果事件A的概率很小，则其发生的频率也很小。实际生活中，常常忽略概率很小的事件发生的可能性。

小概率事件原理：概率很小的事件在一次试验中几乎不会发生，认为它(在一次试验中)是不可能事件。

**🕮 5.4 中心极限定理**

设是相互独立的随机变量序列，是前*n*项和的标准化变量：

记的分布函数是。若：

即：依分布收敛于标准正态变量

则称服从中心极限定理。

现实中许多变量可以表示为诸多相互独立的随机变量的和，其中每个随机变量对总和的影响非常小，则这个总和近似服从正态分布。

① 独立同分布的中心极限定理

是独立同分布的随机变量序列，，则：

② 棣莫弗-拉普拉斯中心极限定理

服从二项分布，，则：

**例**：一批数量很大的产品，次品率为0.02，从中抽取10000件，求次品数不超过221件的概率。

|  |
| --- |
| 设10000件次品数是Y，则 |

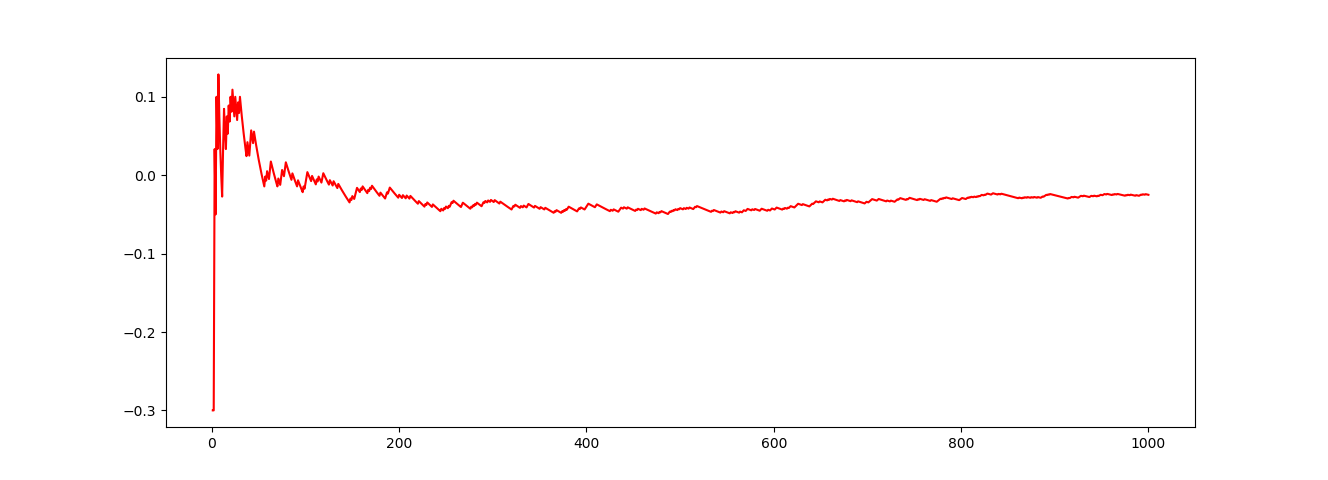
中心极限定理奠定了**正态分布**的至高无上的地位。

**🕮 5.5 大数定律和中心极限定理的Python实现**

① 贝努里大数定律

|  |
| --- |
| import numpy as np  import matplotlib.pyplot as plt  from scipy.stats import bernoulli  n = 1000  x = np.arange(1, n+1)  p = 0.3  # 1000次贝努里随机抽样,p=0.3  r = bernoulli.rvs(p, *size*=n)  y = []  m = 0  for i in range(n):  if r[i] == 1:  m += 1  # y表示1000次的频率-概率,当x足够大,y趋于0  y.append(m/(i+1)-p)  plt.plot(x, y, *color*='red')  plt.show() |

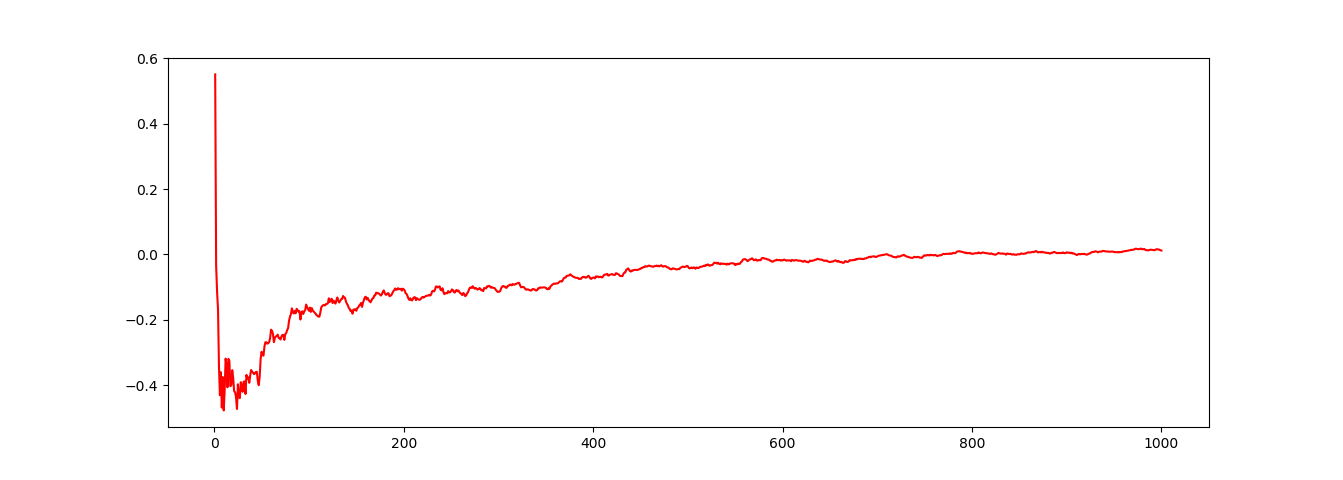
结果：



② 一般的大数定律

|  |
| --- |
| import numpy as np  import matplotlib.pyplot as plt  from scipy.stats import binom, poisson, norm  n = 1000  x = np.arange(1, n+1)  # 3个分布期望都是6  r1 = binom.rvs(10, 0.6, *size*=n) # 二项分布抽样n=10,p=0.6  r2 = poisson.rvs(*mu*=6, *size*=n) # 泊松分布抽样λ=6  r3 = norm.rvs(*loc*=6, *size*=n) # 正态分布抽样μ=6  y = []  rsum = 0  for i in range(n):  rsum += r1[i]+r2[i]+r3[i]  # y表示1000次的频率-概率,当x足够大,y趋于0  y.append(rsum/(i+1)/3-6)  plt.plot(x, y, *color*='red')  plt.show() |

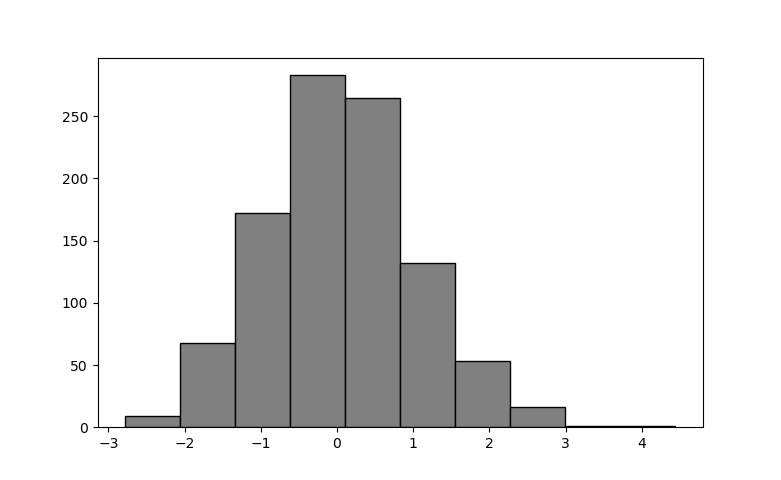
结果：



③ 独立同分布的中心极限定理：

|  |
| --- |
| import numpy as np  import matplotlib.pyplot as plt  from scipy.stats import expon  n = 100  y = []  for i in range(1000): # 1000次重复  # 指数分布: λ=1,期望μ=1,标准差σ=1  r = expon.rvs(*scale*=1, *size*=n) # n次抽样  rsum = np.sum(r) # n次抽样求和  # 标准化变换  z = (rsum-n)/np.sqrt(n)  y.append(z)  plt.hist(y, *color*='grey', *edgecolor*='black')  plt.show() |

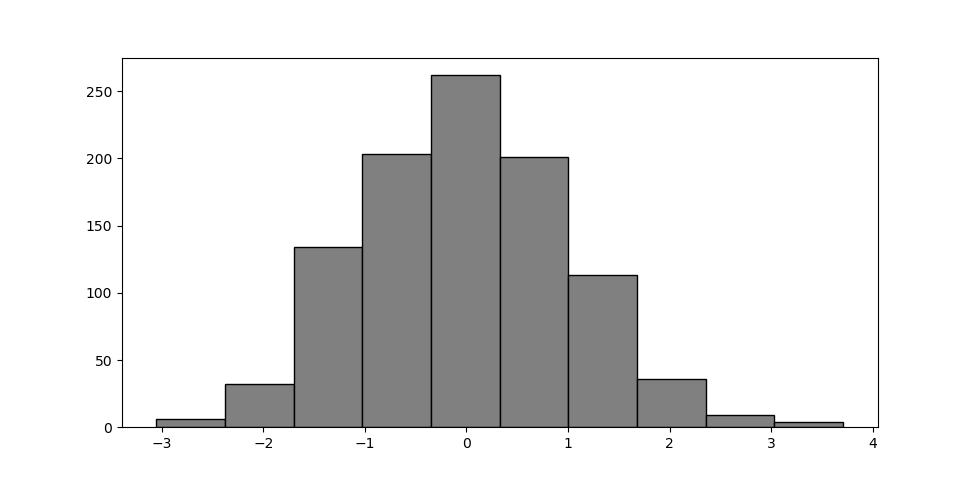
结果：



④ 棣莫弗-拉普拉斯中心极限定理：

|  |
| --- |
| import numpy as np  import matplotlib.pyplot as plt  from scipy.stats import binom  # 二项分布B(n.p)  n, p = 100, 0.3  y = []  for i in range(1000): # 1000次重复  r = binom.rvs(n, p)  rsum = np.sum(r) # n次抽样求和  # 标准化变换  z = (rsum-n\*p)/np.sqrt(n\*p\*(1-p))  y.append(z)  plt.hist(y, *color*='grey', *edgecolor*='black')  plt.show() |

结果：



**第6章 抽样分布理论**

**🕮 6.1 随机样本&统计量**

**🏵 6.1.1 数理统计引言**

数理统计是数学的一个分支，主要研究如何收集和使用随机性数据。

|  |
| --- |
| 1) 有效地收集数据：普查、抽样调查、安排试验  2) 有效地使用数据：分析和提取数据中的信息，对所研究问题作出统计推断(建立一个统计模型，给定某些准则)  3) 归纳性质：一般数学是演绎式推理；而统计是归纳推理。  由于归纳推理依赖与随机性数据，结果也具有不确定性。  统计就是提供归纳推理和计算不确定性程度的方法。 |

**🏵 6.1.2 数理统计基本概念**

总体：由所研究问题的有关个体组成

按包含个体数目，分为：有限总体和无限总体

样本：从总体中抽取的一部分个体，不同抽样方法得到不同的样本

定义：

统计问题研究对象的全体称为总体，总体可以用一个随机变量及其概率分布描述。

样本的两重性：抽样前是随机变量；抽样后是具体的数。

简单随机样本：

1) 代表性：总体每个个体同等机会被抽入样本，即每个个体与总体分布相同

2) 独立性：样本每个个体取值不影响其他个体，即每个个体是相互独立的

此处的样本一般指简单随机样本，故简单随机样本在此处简称样本。

设总体为X，分布函数为是从总体中抽取的样本，则的联合分布为：

若X的概率密度是，则的联合概率密度为：

统计模型：指该问题所得样本的分布，如正态分布对应于正态模型

统计推断：

1) 样本的分布已知，对其所含未知参数的推断——参数统计推断。常见有参数估计、假设检验等。

2) 样本的分布未知——非参数统计推断。

**🏵 6.1.3 统计量的概念**

数理统计任务的要通过样本去推断总体，需要对样本的数据进行分析整理，构造出合适的量以解决总体的相关问题。

统计量是由样本所确定(计算)的量，是样本的函数，完全由样本决定。

**注意**：

1) 统计量只和样本有关，不能含有总体的未知参数

2) 由于样本具有两重性，因此统计量也具有两重性：既可以看成一个数，也可以看成随机变量。因此统计量有概率分布，是这统计推断的依据。

3) 统计量应最好的集中了样本的(与总体关联度最高)信息以解决总体相应问题。

**🏵 6.1.4 常见的统计量**

设X是总体，是样本：

① 样本均值：

② 样本方差：

前者也称为修正的方差，与"无偏性"有关。

③ 矩 (k阶原点矩和k阶中心矩)

④ 将按大小排列为：

则称为次序统计量。为极小值；为极大值。

**🕮 6.2 三大抽样分布**

**🏵 6.2.1 卡方分布 (分布)**

设是来自总体的样本，称统计量：

服从自由度为*n*的分布，记为。自由度是包含独立变量的个数。

分布的性质：

|  |
| --- |
| 1) 概率密度函数比较复杂，图形如下：  C:\Users\hikari\Desktop\QQ浏览器截屏未命名.png  图形只在第一象限，自由度*n*越大，密度曲线越对称。  2) 若，有  3) 可加性：若，且相互独立，则 |

**🏵 6.2.2 t分布&分位数的定义**

设，且相互独立，则称随机变量：

服从自由度为*n*的t分布，记为。

t分布最早由戈塞(Gosset)发表，其笔名为Student，故t分布又称学生分布。

t分布的特点：

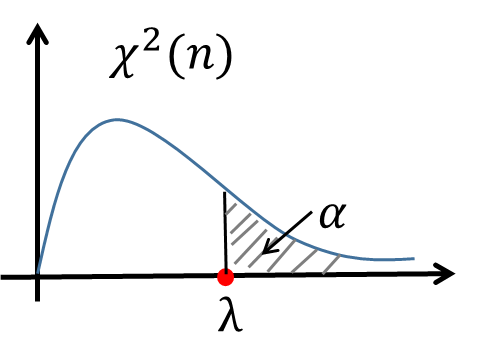
|  |
| --- |
| 1) t分布概率密度函数的特点：类似于  2) 若时，；时，  3)时，其极限分布为 |

分位点或分位数

一些分布的概率密度函数非常复杂，直接用其计算概率很麻烦。

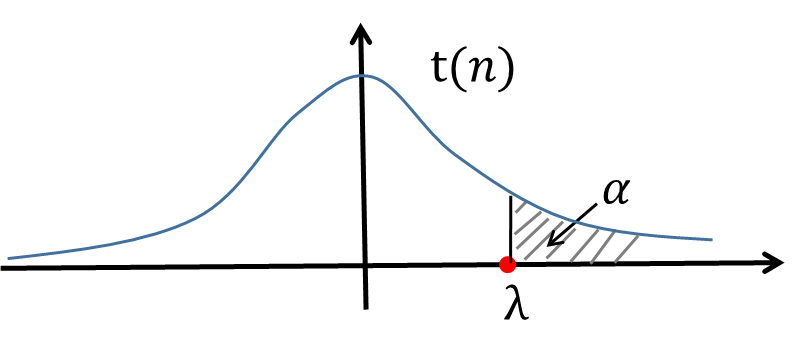
数学家将常见的值编成各种表(如正态分布表、卡方分布表等)，利用分位点的概念，通过查表就可得到结果。

如分布：



，记：为分布的上分位点。

如t分布：



，记：为分布的上分位点。

**🏵 6.2.3 F分布**

设，且相互独立，称随机变量：

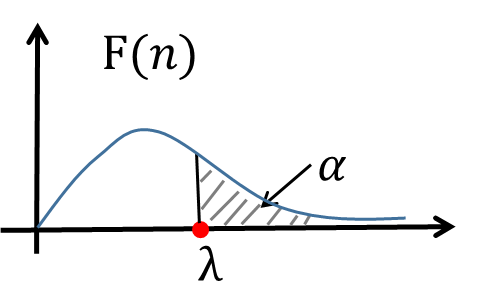
服从自由度为的F分布，记为。

F分布的性质：

1) 若，则：

2) 若，则：

3) F分布的分位点：



，记：为分布的上分位点。

F分布的上分位点的性质：

**🕮 6.3 抽样分布定理**

正态总体样本均值和样本方差的分布。

设是来自正态总体的样本，样本均值和样本方差为：

|  |
| --- |
|  |

如：测量某个元件长度时，多次测量取平均值能减少误差，就是基于这个原理(方差原来的1/*n*)。

|  |
| --- |
| 因为，所以：  // 不会了... |

**☆☆☆**

(4) 由(1)(2)构造出：

|  |
| --- |
| 根据t分布定义得到： |

(5) 是来自正态总体的样本；是来自正态总体的样本，两个样本相互独立。样本均值和方差为：。

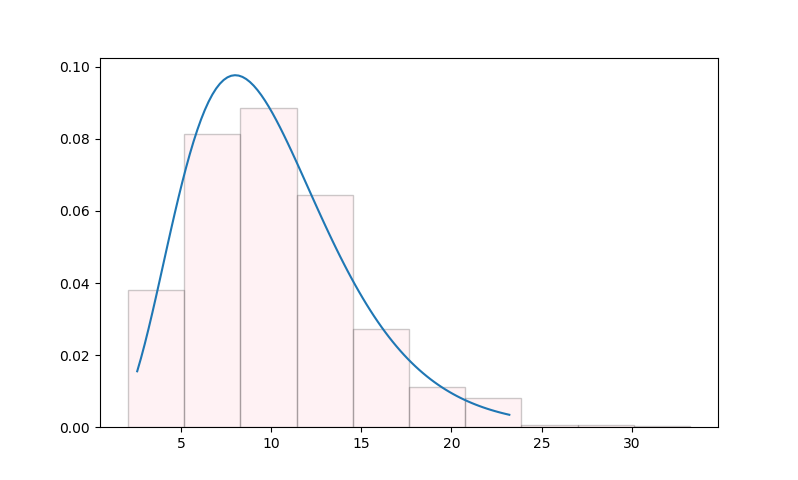
|  |
| --- |
| ：  其中 |

**🕮 6.4 抽样分布的Python实现**

① 卡方分布

|  |
| --- |
| import matplotlib.pyplot as plt  import numpy as np  from scipy.stats import chi2, norm  fig, ax = plt.subplots(1, 1)  df = 10 # 自由度  x = np.linspace(chi2.ppf(0.01, df), chi2.ppf(0.99, df), 100)  # 画出scipy内置的卡方分布曲线,自由度df=10  ax.plot(x, chi2.pdf(x, df))  # 根据卡方分布定义,模拟1000个样本近似,每次样本抽10个正态分布随机变量  N, n = 1000, 10  y = []  for i in range(N):  chi2\_ret = 0  r = norm.rvs(*size*=n)  for j in r:  chi2\_ret += j\*\*2  y.append(chi2\_ret)  ax.hist(y, *density*=True, *color*='pink', *alpha*=0.2, *edgecolor*='black')  plt.show() |

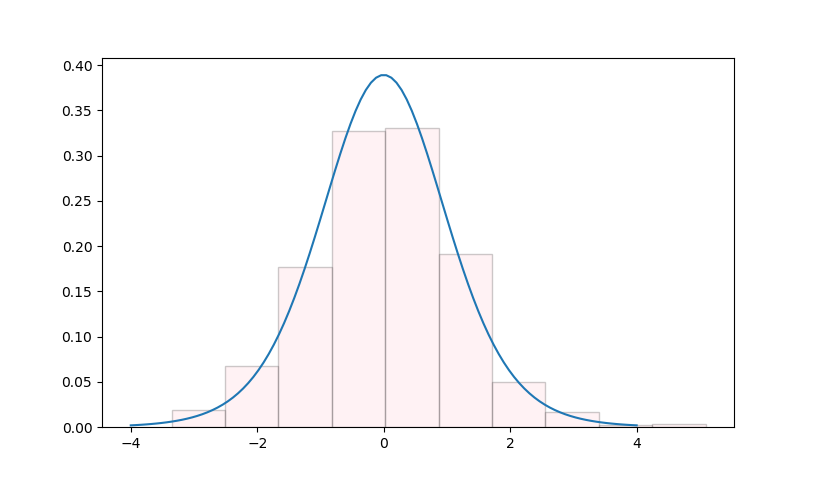
结果：



② t分布

|  |
| --- |
| import matplotlib.pyplot as plt  import numpy as np  from scipy.stats import chi2, norm, t  *def* t\_dis():  fig, ax = plt.subplots(1, 1)  df = 10 # 自由度  x = np.linspace(-4, 4, 100)  # scipy内置的t分布曲线,自由度df=10  ax.plot(x, t.pdf(x, df))  N = 1000  y = []  # 根据定义构造t分布  for i in range(N):  rx = norm.rvs()  ry = chi2.rvs(df)  ret = rx/np.sqrt(ry/df)  y.append(ret)  # 画出模拟的直方图和标准曲线的叠加图  ax.hist(y, *density*=True, *color*='pink', *alpha*=0.2, *edgecolor*='black')  plt.show() |

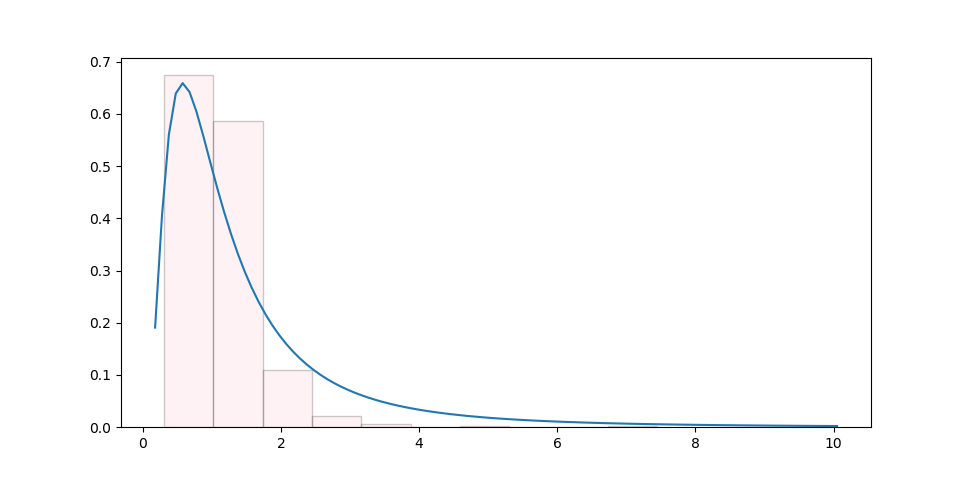
结果：



③ F分布

|  |
| --- |
| *def* F\_dis():  fig, ax = plt.subplots(1, 1)  dfm, dfn = 10, 5 # 自由度  x = np.linspace(f.ppf(0.01, dfm, dfn), f.ppf(0.99, dfm, dfn), 100)  # scipy内置的F分布曲线~F(10,5)  ax.plot(x, f.pdf(x, dfm, dfn))  N = 1000  y = []  # 根据定义构造F分布  for i in range(N):  rx = chi2.rvs(dfm)  ry = chi2.rvs(dfn)  ret = np.sqrt((rx/dfm)/(ry/dfn))  y.append(ret)  # 画出模拟的直方图和标准曲线的叠加图  ax.hist(y, *density*=True, *color*='pink', *alpha*=0.2, *edgecolor*='black')  plt.show() |

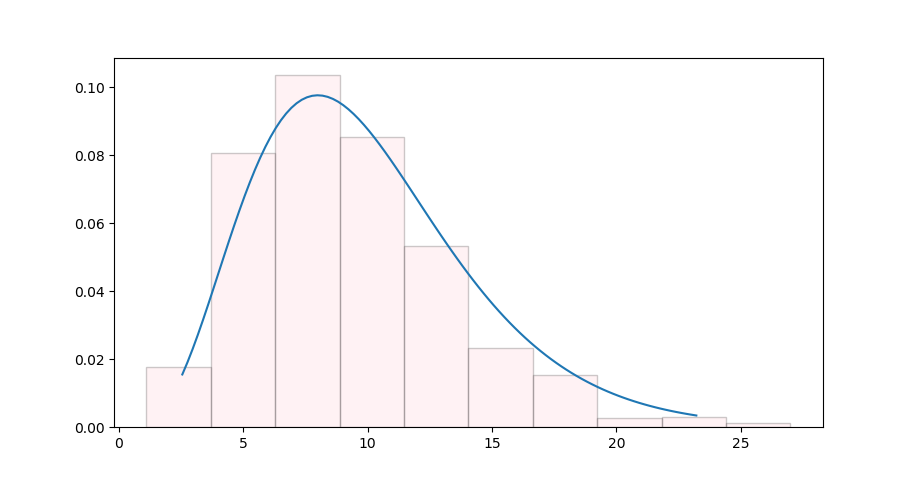
结果：



④ 样本方差抽样分布

|  |
| --- |
| *def* S2():  fig, ax = plt.subplots(1, 1)  df = 10  x = np.linspace(chi2.ppf(0.01, df), chi2.ppf(0.99, df), 100)  ax.plot(x, chi2.pdf(x, df))  N = 1000  y = []  for i in range(N):  # 每次样本抽取n(即df+1)个正态分布(μ=5,σ=2)随机变量  sigma = 2  n = df+1  r = norm.rvs(*loc*=5, *scale*=sigma, *size*=n)  # np.var(r)为求样本的方差  ret = (n-1)\*np.var(r)/(sigma\*\*2)  y.append(ret)  ax.hist(y, *density*=True, *color*='pink', *alpha*=0.2, *edgecolor*='black')  plt.show() |

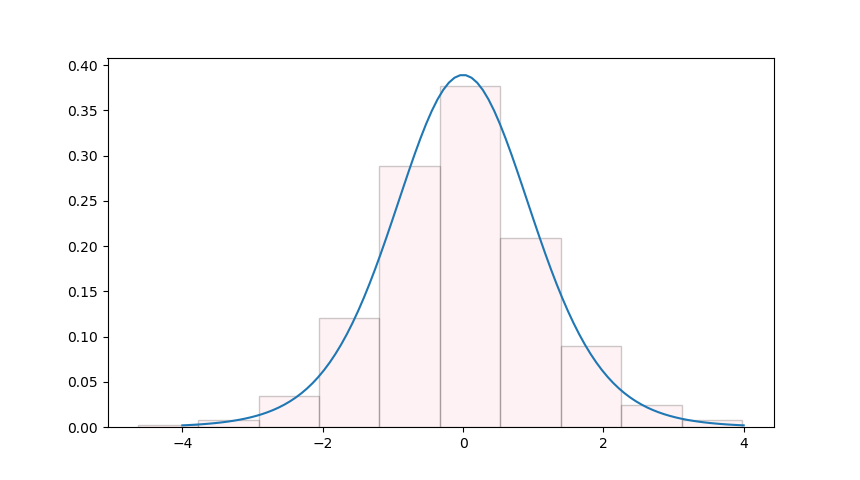
结果：



⑤ 抽样分布-t

|  |
| --- |
| *def* t1():  fig, ax = plt.subplots(1, 1)  df = 10  x = np.linspace(-4, 4, 100)  # scipy内置的t分布曲线,自由度df=10  ax.plot(x, t.pdf(x, df))  N = 1000  y = []  for i in range(N):  # μ=5, σ=2的正态分布,每次样本抽取n次  miu, sigma, n = 5, 2, df+1  r = norm.rvs(*loc*=miu, *scale*=sigma, *size*=n)  ret = (np.mean(r)-miu)/np.sqrt(np.var(r)/n)  y.append(ret)  # 画出模拟的直方图和标准曲线的叠加图  ax.hist(y, *density*=True, *color*='pink', *alpha*=0.2, *edgecolor*='black')  plt.show() |

结果：



**第7章 描述统计**

**🕮 7.1 变量定义**

为根据不同度量水平将变量分为：

|  |
| --- |
| 1) 定类变量(Nominal Variable)，又称名义变量，通常用于代表不同的分类。  如1代表男0代表女；C、Java、Python定义为0、1、2；  定类变量数学关系只有等于(=)和不等于(≠)。  2) 定序变量(Ordinal Variable)：采用数字表示顺序。每个分类有差别，还有等级之分。如优等品、合格品、次品，分别记为2、1、0。  定序变量数学关系有=、≠、<、>等。注意：不一定是等距的。  3) 定距变量(Interval Variable)，也称间隔变量，描述事物类别或次序间距。  定距变量中，0是强行规定的，不代表完全没有的意思，如温度0℃不代表没有温度，只是代表水结冰的特定温度。  4) 定比变量(Ratio Variable)：在定距变量基础上，扩展可作为比率的基数而成。  需要统一的单位，如m、cm、kg、s等。身高体重都是定比变量，和定距变量的一个根本区别是定比变量的0代表完全没有。 |

变量类型主要基于相应的变量数值可做的有意义数学运算区分的。

4种变量的计量层次由低级到高级、由粗略到精确递进。高层次变量有低层次变量全部特性，反之没有。

|  |  |  |  |
| --- | --- | --- | --- |
| **变量类型** | **特点** | **关系和运算** | **举例** |
| 定类变量 | 无顺序分类 |  | 性别 |
| 定序变量 | 有顺序分类 |  | 严重级别 |
| 定距变量 | 等间隔但没有绝对零点 |  | 温度、标准成绩 |
| 定比变量 | 等间隔且有绝对零点 |  | 身高、年龄、数量 |

**🕮 7.2 统计图表**

使用统计图表展示数据更加直观，令人信服。

**🏵 7.2.1 分类型数据统计图表**

分类型数据来自于定类或定序变量，主要用于事物的分类描述，计算每一类别的频数、频率、累积频率等。

① 频数分布表

将每个类别及其频数以表格形式表现。频数是每个类别数据出现的次数。

如：黄金周过节方式的网络调查：

|  |  |  |  |
| --- | --- | --- | --- |
| **过节方式** | **频数** | **频率** | **百分比** |
| 宅在家 | 7853 | 0.45 | 45% |
| 探亲访友 | 6632 | 0.38 | 38% |
| 旅游 | 873 | 0.05 | 5% |
| 加班 | 873 | 0.05 | 5% |
| 其他 | 1221 | 0.07 | 7% |
| 合计 | 17452 | 1 | 100% |

② 柱状图

等宽的垂直图条按类别分组显示，柱状高度描述统计量的值。用于各组之间的比较。水平显示的称为条状图。

③ 饼图

主要描述频数、频率、百分比之间相对关系，研究结构性问题。

饼图具有较好的视觉效果，但表现的信息相对较少。

④ 折线图

用直线将数据点连接起来，显示数据的变化趋势。适用于显示相等时间间隔下数据的趋势。

⑤ 帕累托图

图中包含柱状图和折线图。柱状图按发生的频率降序排列，而折线图显示了累积频率。帕累托图来源于帕累托定律(即28定律)：80%的结果取决于20%的原因。

柱状图上方的折线是帕累托曲线。

**🏵 7.2.2 数值型数据统计图表**

数值型数据包括定距和定比数据。通常要进行数据分组。分组再计算各组的频数，就形成了频数分布表。

组数k，可以根据Sturges提出的经验公式：

确定组距的简单方法就是均分，即(max-min)/k。

① 直方图

矩形的宽度和高度表示频数分布。组距一般相等，也可不等。

直方图和柱状图类似，但直方图的横坐标是连续的数据，是刻度轴；而柱状图是分类轴。

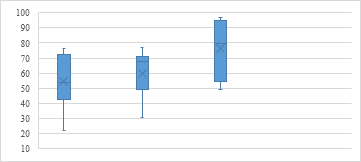
柱状图使用高度表示各类的频数，而直方图使用面积表示。

② 茎叶图

③ 盒状图

又称箱形图，用来表示数据集是位置和变异信息。

最上方的横线表示最大值；最下方的横表示最小值；方体中的三个横线从上到下为第三个分位数(Q3)、中位数(Q2)，第一个分位数(Q1)。



**🏵 7.2.3 多变量数据统计图表**

考虑来自两个(或多个)变量(X, Y)样本数据，

① 散点图

完全线性相关、线性相关、非线性相关、不相关。

② 雷达图

也称蜘蛛图，是显示多个变量的常用统计图形。

雷达图用于显示和比较不同批次的数据和不同变量的数值总和，或显示比较不同数据间的相似性。

③ 列联表

观测数据按两个或多个属性(定类或定序)分类时列出的频数表。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **性别/等级** | **不及格** | **及格** | **中** | **良** | **优** | **合计** |
| **男生** | 4 | 13 | 8 | 4 | 3 | 32 |
| **女生** | 1 | 5 | 3 | 5 | 4 | 18 |
| **合计** | 5 | 18 | 11 | 9 | 7 | 50 |

**🕮 7.3 常用数据汇总**

**🏵 7.3.1 数据汇总**

**🏵 7.3.2 分散趋势度量**

**🏵 7.3.3 形态度量**

**🕮 7.4 统计图表的Python实现**