

我是8位的

I am 8 bits, what about you?

随笔 - 205, 文章 - 0, 评论 - 103, 阅读 - 101万

导航

博客园

首页

新随笔

联系

订阅

管理

<

2022年3月

>

日	一	二	三	四	五	六
27	28	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

公告

你的支持是我的动力

欢迎关注微信公众号“我是8位的”



昵称：我是8位的

园龄：4年7个月

粉丝：288

关注：5

+加关注

盖楼抽奖

#她的梦想在发光#

HWD科技女性故事有奖征集

分享最打动的科技女性故事

活动时间：2022年3月8日-3月18日

马上参与



搜索

找找看

谷歌搜索

常用链接

我的随笔

我的评论

我的参与

最新评论

我的标签

积分与排名

积分 - 457097

排名 - 1198

概率统计21——指数分布和无记忆性

指数分布（Exponential distribution）是一种连续型概率分布，可以用来表示独立随机事件发生的时间间隔的概率，比如婴儿出生的时间间隔、旅客进入机场的时间间隔、打进客服中心电话的时间间隔、系统出现bug的时间间隔等等。

指数分布的由来

指数分布与泊松分布存在着联系，它实际上可以由泊松分布推导而来。

泊松分布（概率统计15）中已经介绍过泊松分布，除了作为二项分布的近似外，当独立事件发生的频率固定时，泊松分布还可以刻画算单位时间内事件发生次数的概率分布。

假设某个公司有一个带伤上线的系统，每周平均的故障次数是2次，在下周不发生故障概率是多少？

每周平均的故障次数是2次，我们可以把“一周”看作单位时间，程序的故障率是 $\lambda=2$ ，单位时间内发生故障的次数 X 符合泊松分布 $X\sim Po(\lambda)$ 。在下周不发生故障的概率相当于发生了0个故障的概率：

$$P(X=r;\lambda)=e^{-\lambda}\frac{\lambda^r}{r!}$$
$$\Rightarrow P(X=0;\lambda=2)=e^{-2}\frac{2^0}{0!}=e^{-2}\approx 0.135$$

现在要求计算两周之内不发生故障的概率。我们用随机变量 $T>2$ 表示在2个单位时间内系统未发生故障的事件。在已知下周不发生故障的概率的情况下， $P(T>2)$ 计算起来很容易：

$$P(T>2)=e^{-2}\times e^{-2}=e^{-4}\quad ①$$

我们换一种思路。之前是把“一周”看作单位时间，单位时间内事件发生的频率是 $\lambda=2$ 。现在是变成了双倍的单位时间，故障发生的频率自然就变成 $2\lambda=4$ ，这样一来，两周之内不发生故障的概率是：

$$P(T>2;2\lambda)=P(X=0;2\lambda)=e^{-2\lambda}\frac{2\lambda^0}{0!}=e^{-2\lambda}=e^{-4}\quad ②$$

结果和①相等。

时间是连续的，如果计算 $T=1.5$ 周内不发生故障的概率，①就显得无能为力了。但是②却没有任何问题，只要把单位时间内事件的频率 λ 随着时间 T 进行放缩就可以了：

$$P(T>1.5;1.5\lambda)=P(X=0;1.5\lambda)=e^{-1.5\lambda}\frac{1.5\lambda^0}{0!}=e^{-1.5\lambda}$$

随笔分类 (211)

- ★★资源下载★★(1)
- Java并发编程(1)
- 程序员的数学(24)
- 单变量微积分(31)
- 多变量微积分(24)
- 概率(24)
- 机器学习(27)
- 软件设计(1)
- 数据分析(6)
- 数据结构与算法(27)
- 随笔(5)
- 线性代数(34)
- 项目管理(2)
- 转载(4)

随笔档案 (205)

- 2021年2月(1)
- 2020年3月(2)
- 2020年2月(6)
- 2020年1月(4)
- 2019年12月(7)
- 2019年11月(15)
- 2019年9月(3)
- 2019年8月(6)
- 2019年7月(1)
- 2019年6月(8)
- 2019年5月(3)
- 2019年4月(5)
- 2019年3月(7)
- 2019年2月(3)
- 2019年1月(7)
- 更多

阅读排行榜

- 1. 使用Apriori进行关联分析 (一) (29768)
- 2. 线性代数笔记12——列空间和零空间 (28772)
- 3. FP-growth算法发现频繁项集 (一)——构建FP树(24430)
- 4. 寻找“最好” (2) ——欧拉-拉格朗日方程(23099)
- 5. 多变量微积分笔记3——二元函数的极值(22772)

评论排行榜

- 1. 隐马尔可夫模型 (一) (8)
- 2. 线性代数笔记12——列空间和零空间 (7)
- 3. 线性代数笔记3——向量2 (点积) (7)
- 4. FP-growth算法发现频繁项集 (一)——构建FP树(5)
- 5. 寻找“最好” (2) ——欧拉-拉格朗日方程(4)

推荐排行榜

- 1. 寻找“最好” (2) ——欧拉-拉格朗日方程(7)
- 2. FP-growth算法发现频繁项集 (一)——构建FP树(7)
- 3. 线性代数笔记3——向量2 (点积) (6)
- 4. FP-growth算法发现频繁项集 (二)——发现频繁项集(5)
- 5. 隐马尔可夫模型 (一) (5)

最新评论

- 1. Re:线性代数笔记3——向量2 (点积)
如果点积小于0, 即夹角小于90°, 这个写错了吧。应该是夹角大于90°
--猫猫猫猫猫大人

将问题推广到任意时间间隔：

$$P(T > t; \lambda t) = P(X = 0; \lambda t) = e^{-\lambda t} \frac{(\lambda t)^0}{0!} = e^{-\lambda t}$$

反过来，故障在时间 t 之内发生的概率就是1-P(T>t)：

$$P(T \leq t; \lambda t) = 1 - P(T > t; \lambda t) = 1 - e^{-\lambda t}$$

现在把T换成X：

$$F(x; \lambda) = P(X \leq x; \lambda x) = 1 - e^{-\lambda x}$$

这里的F(x;λ)就是指数分布的分布函数，λ表示平均每单位时间内事件发生的次数，随机变量X表示时间间隔。

F(x;λ)对应的密度函数是：

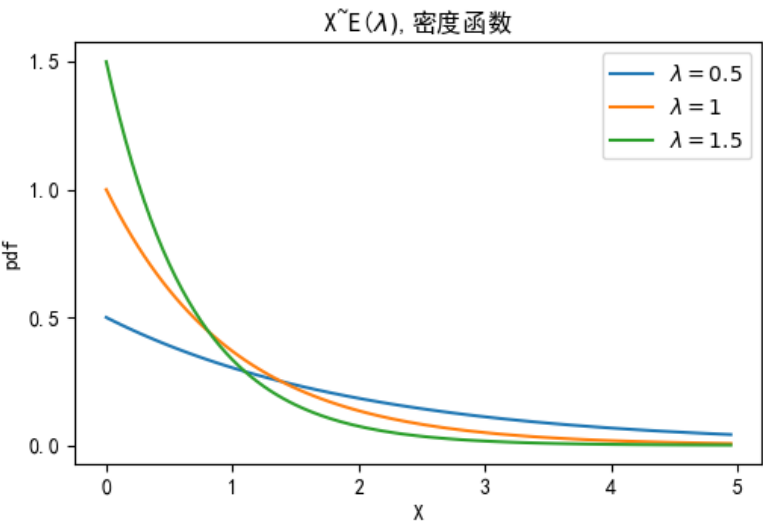
$$f(x; \lambda) = \begin{cases} \lambda e^{-\lambda x}, & x > 0 \\ 0, & \text{else} \end{cases}$$
$$\int_{-\infty}^0 0 \, dt + \int_0^x \lambda e^{-\lambda t} \, dt = \int_0^x e^{-\lambda t} \, d\lambda t = -e^{-\lambda t} \Big|_0^x = 1 - e^{-\lambda x}$$

随机变量X符合参数为λ>0的指数分布，记作X~E(λ)。

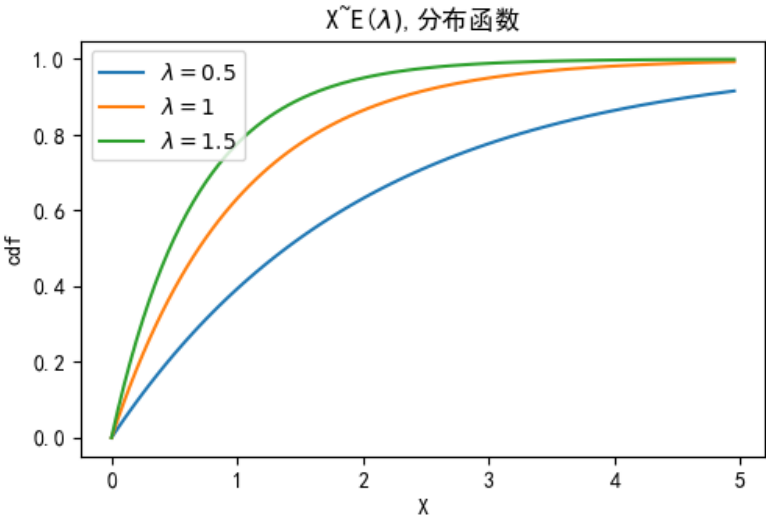
有些资料的写法是：

$$f(x; \theta) = \begin{cases} \frac{1}{\theta} e^{-\frac{x}{\theta}}, & x > 0 \\ 0, & \text{else} \end{cases}$$

仅仅是用θ代替了1/λ。



2. Re:线性代数笔记10——矩阵的LU分解写的很好，不过LU分解的前提是错的，LU分解只需要第三个条件，如果允许行置换就是下面写到的PLU，可以分解所有矩阵
--wiki3D
3. Re:单变量微积分笔记20——三角替换1 (sin和cos)很nice
--尹保棕
4. Re:线性代数笔记24——微分方程和exp(At)有些图片挂了呢
--ccchendada
5. Re:寻找“最好” (2) ——欧拉-拉格朗日方程提个issue，最速降线中 $v = \{2gh\}^{1/2}$ 与配图不一致，建议以起点为原点，向右伸出x轴，向下伸出y轴建立坐标系
--trustInU



无记忆性

指数分布的一个重要特征是无记忆性 (Memoryless Property, 又称遗失记忆性)。如果一个随机变量呈指数分布 $X \sim E(\lambda)$, 当 $s, t \geq 0$ 时:

$$P\{X > s + t | X > s\} = P\{X > t\}$$

先来看看这个等式为什么成立。

$$\begin{aligned} P\{X > s + t | X > s\} &= \frac{P\{(X > s + t) \cap (X > s)\}}{P\{X > s\}} \\ &= \frac{P\{X > s + t\}}{P\{X > s\}} \\ &= \frac{1 - P\{X \leq s + t\}}{1 - P\{X \leq s\}} \\ &= \frac{1 - (1 - e^{-\lambda(s+t)})}{1 - (1 - e^{-\lambda s})} \\ &= e^{-\lambda t} \end{aligned}$$

$$P\{X > t\} = 1 - P\{X \leq t\} = 1 - (1 - e^{-\lambda t}) = e^{-\lambda t}$$

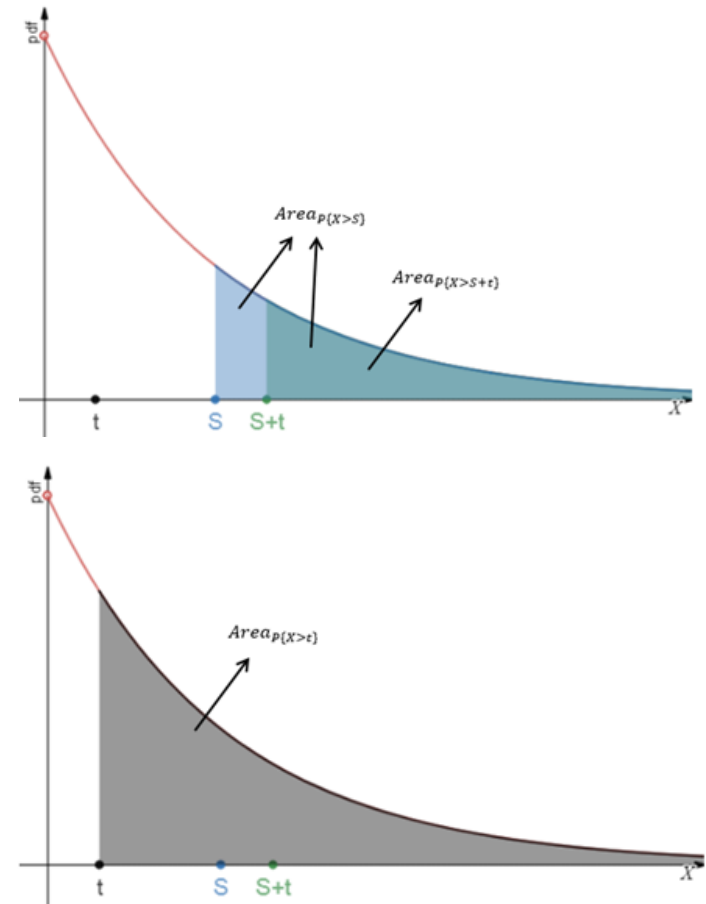
因此二者相等。

在前面的推导中可以看到:

$$\begin{aligned} P\{X > s + t | X > s\} &= \frac{P\{X > s + t\}}{P\{X > s\}} = P\{X > t\} = \frac{P\{X > t\}}{1} \\ &\Rightarrow \frac{P\{X > s + t\}}{P\{X > s\}} = \frac{P\{X > t\}}{1} \end{aligned}$$

累积概率对应面积, 这个结论告诉我们, 在指数分布下, $P\{X > s + t\}$ 和 $P\{X > s\}$ 对应的面积的比值等于 $P\{X > t\}$ 对应的面积和总体面积的比值:

$$\frac{A_{P\{X > s + t\}}}{A_{P\{X > s\}}} = \frac{A_{P\{X > t\}}}{A_{whole}}$$



理解无记忆性

我们已经知道指数分布可以用来表示独立随机事件发生的时间间隔的概率分布，在精密元件的可靠性研究中，指数分布通常用于描述对元件发生缺陷数测量结果。但是无记忆性又指出，元件在经过 s 时间的工作之后，它的寿命分布与原来还未工作时的寿命分布相同。这就好比有一个已经用了10年的灯泡和一个刚买的灯泡，现在我告诉你它俩在未来一年里报废的概率相同，你信吗？

一个著名的问题是“赌徒心理”。假设在赌桌上只能押大或小，某个赌徒已经连续押了10次小，但都输了，于是赌徒认为，下次再出现小的可能性非常低，于是他想把兜里的钱全部押大，试图翻本。



不妨帮助赌徒分析一下翻本的概率。

假设这是一个公平的赌场，大和小的概率都是0.5，用随机变量X表示第一次出现“大”时所参与的赌局数，X符合几何分布：

$$P(X = r) = pq^{r-1}$$

$$\Rightarrow P(X = 11) = 0.5 \times 0.5^{10} = 0.5^{11}$$

其中p表示出现“大”的概率，q表示出现“小”的概率。直到第11次才出现“大”的概率是 0.5^{11} ，这是个非常小的数值，可见这个赌徒确实运气不佳。

接着分析，用 $X > 10$ 表示连续10次以上没出现“大”的事件，以此为前提，下一次（第11次）出现大的概率是：

$$P(X = 11 | X > 10) = \frac{P(X = 11 \cap X > 10)}{P(X > 10)} = \frac{P(X = 11)}{P(X > 10)} = \frac{pq^{11-1}}{q^{10}} = p = 0.5$$

$$P(X = 1) = 0.5$$

在连续押小输掉10次的条件下，下一次出现“大”，和第1次就出现“大”的概率相同。对于赌徒来说，每一局都是全新的，下一次赢钱的概率和之前的输赢没有任何关系，这就是所谓的无记忆性。

指数分布与几何分布类似，只不过把离散随机变量变成了连续型。

实际上生活中有很多无记忆性的例子，比如车牌摇号。对于参与摇号的人来说，在每一次摇号中摇中的概率都是相等的。有人说不对啊，明明告诉我摇中的概率是1%，100次里面怎么还不能中一回吗？所谓的1%摇中率，是在大数定律下才起作用，而现实生活中我们面对的往往是“小数”。可以用程序模拟一下：

```
1 import numpy as np
2
3 np.random.seed(41)
4 for i in range(1, 11, 1):
5     m = 100 * i # 试验次数
6     nums = np.random.randint(1, 101, 200) # 从1~100中随机选择m个数字
7     p_50_size = nums[nums == 50].size # m个数字中出现50的次数
8     print('一共参与摇号{}次, 摇中{}次, 概率{}'.format(m, p_50_size, p_50_size/m))
```

结果显示：

一共参与100次摇号，摇中0次，概率0.0

一共参与200次摇号，摇中0次，概率0.0

一共参与300次摇号，摇中3次，概率0.01

一共参与400次摇号，摇中3次，概率0.0075

一共参与500次摇号，摇中2次，概率0.004

一共参与600次摇号，摇中3次，概率0.005

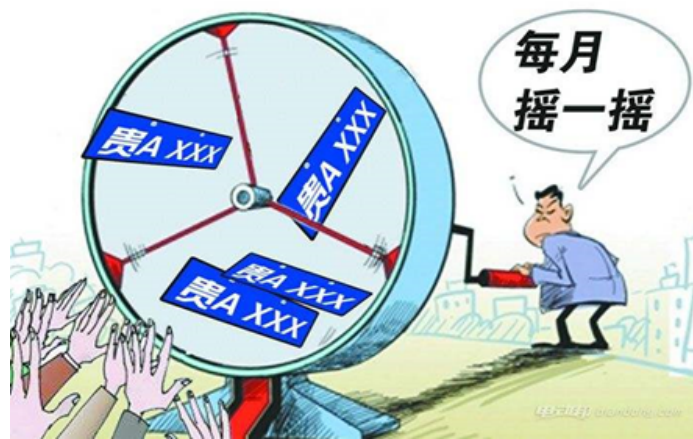
一共参与700次摇号，摇中4次，概率0.005714285714285714

一共参与800次摇号，摇中3次，概率0.00375

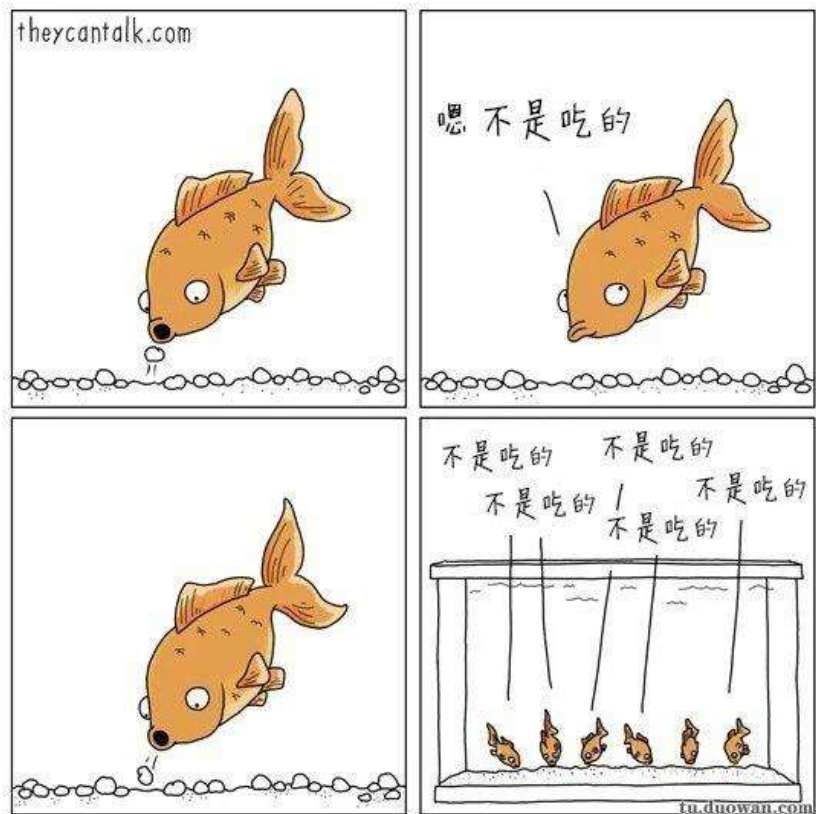
一共参与900次摇号，摇中3次，概率0.0033333333333333335

一共参与1000次摇号，摇中5次，概率0.005

每月摇号一次，一个人穷尽一生也满足不了“大数”，摇号还真是件没谱的事。



因为无记忆性的关系，有人戏称服从指数分布的随机变量就像传说中金鱼只有7秒钟记忆一样，是“永远年轻的”。也正是这一点限制了指数分布的应用，因为指数分布忽略了损耗。但是，指数分布仍然可以近似地作为高可靠性的复杂部件、机器或系统的失效分布模型，特别是在部件或机器的整机试验中得到广泛的应用。



至于灯泡的寿命是否符合指数分布，其实我也不知道。也许现在制作工艺提升了，灯泡已经是具有高可靠性的产品，灯泡明天是否还能正常工作，

完全取决于是否遭到了外力的破坏，日常损耗所起到的作用可以忽略不计，这种情况下，灯泡的使用寿命才有可能符合指数分布。

关于保修期的问题

冰箱平均10年出现一次大的故障，求：

- (1) 冰箱使用15年后还没有出现大故障的比例。
- (2) 如果厂家想提供大故障免费维修的质保，试确定保修1~5年内，需要维修的冰箱的占比。

冰箱平均10年出现大的故障，可见故障率不高，可以认为故障次数服从泊松分布，单位时间是1年， $\lambda=0.1$ 。

(1) 根据指数分布：

$$P(X > x) = 1 - P(X \leq x; \lambda x) = e^{-\lambda x}$$
$$P(X > 15) = e^{-1.5} \approx 0.223$$

冰箱使用15年后还没有出现大故障的比例约等于22.3%。

(2)

```
1 from scipy import stats
2
3 lam = 0.1
4 for i in range(1, 6):
5     print(stats.expon.cdf(i, scale=1/lam))
```

保修年数 x	保修期内维修的冰箱占比 $P(X \leq x) = 1 - e^{-\lambda x}$
1	0.095
2	0.181
3	0.259
4	0.330
5	0.393

上表告诉了我们冰箱的保修期一般都是2年以内的原因，厂家为了把上门修理的次数控制在20%以内，一般选择保修2年。

期望和方差

对于 $X \sim E(\lambda)$ 的指数分布来说，它的期望是 $1/\lambda$ ，方差是 $1/\lambda^2$ 。

先来看期望：

$$E[X] = \int_0^\infty x f(x; \lambda) dx = \int_0^\infty x \lambda e^{-\lambda x} dx$$
$$\text{let } u = \lambda x, \quad \text{then } du = \lambda dx, x = \frac{u}{\lambda}$$
$$\int_0^\infty x \lambda e^{-\lambda x} dx = \int_0^\infty \frac{u}{\lambda} e^{-u} du = \frac{1}{\lambda} \int_0^\infty u e^{-u} du$$

根据分部积分：

$$\int uv' dx = uv - \int u'v dx$$

$$\text{let } u = u, v' = e^{-u}, \quad \text{then } u' = 1, v = -e^{-u}$$

$$\frac{1}{\lambda} \int_0^{\infty} u e^{-u} du = \frac{1}{\lambda} (-u e^{-u} - e^{-u}) \Big|_0^{\infty} = \lim_{u \rightarrow \infty} \frac{-(u+1)}{\lambda e^u} + \frac{1}{\lambda} = \frac{1}{\lambda} \quad (3)$$

再来看方差：

$$\text{Var}(X) = E[X^2] - E^2[X] = E[X^2] - \frac{1}{\lambda^2}$$

$$E[X^2] = \int_0^{\infty} x^2 \lambda e^{-\lambda x} dx$$

$$\text{let } u = \lambda x, \quad \text{then } du = \lambda dx, x = \frac{u}{\lambda}$$

$$\int_0^{\infty} x^2 e^{-u} du = \frac{1}{\lambda^2} \int_0^{\infty} u^2 e^{-u} du$$

继续利用分部积分：

$$\text{let } U = u^2, V' = e^{-u}, \quad \text{then } U' = 2u, V = -e^{-u}$$

$$\begin{aligned} \frac{1}{\lambda^2} \int_0^{\infty} u^2 e^{-u} du &= \frac{1}{\lambda^2} \left(UV \Big|_0^{\infty} - \int_0^{\infty} U' V du \right) \\ &= \underbrace{\frac{1}{\lambda^2} (-u^2 e^{-u}) \Big|_0^{\infty}}_{(4)} - \underbrace{\frac{1}{\lambda^2} \int_0^{\infty} 2u(-e^{-u}) du}_{(5)} \end{aligned}$$

将 $u=\lambda x$ 代入④：

$$\frac{1}{\lambda^2} (-u^2 e^{-u}) \Big|_0^{\infty} = \frac{1}{\lambda^2} (-\lambda^2 x^2 e^{-\lambda x}) \Big|_0^{\infty} = -x^2 e^{-\lambda x} \Big|_0^{\infty} = \lim_{x \rightarrow \infty} \frac{-x^2}{e^{\lambda x}} = 0$$

将在 $E[X]$ 中求得的③代入⑤中：

$$\frac{1}{\lambda^2} \int_0^{\infty} 2u(-e^{-u}) du = \frac{2}{\lambda^2}$$

最终：

$$E[X^2] = (4) + (5) = \frac{2}{\lambda^2}$$

$$\text{Var}(X) = E[X^2] - E^2[X] = \frac{2}{\lambda^2} - \frac{1}{\lambda^2} = \frac{1}{\lambda^2}$$

出处：微信公众号“我是8位的”

本文以学习、研究和分享为主，如需转载，请联系本人，标明作者和出处，非商业用途！

扫描二维码关注作者公众号“我是8位的”



随笔

分类: 概率

标签: 公式来源, 无记忆性, 指数分布

好文要顶

关注我

收藏该文

我是8位的

关注 - 5

粉丝 - 288

+加关注

0

推荐

0

反对

« 上一篇: 概率统计20——估计量的评选标准

» 下一篇: 概率统计22——假设检验理论 (1)

posted on 2020-02-27 22:56

我是8位的

阅读(7816)

评论(0)

编辑

收藏

举报

[刷新评论](#) [刷新页面](#) [返回顶部](#)

登录后才能查看或发表评论, 立即 [登录](#) 或者 [逛逛](#) [博客园首页](#)

- 【推荐】华为 HWD 2022 故事征集，分享最打动你的科技女性故事
- 【推荐】华为开发者专区，与开发者一起构建万物互联的智能世界

广告

yo zodcs.com

永中DCS，文档在线预览处理专家

10年行业经验，受到众多政府客户、央企单位、邮箱客户、OA办公系统、招聘网站、教育客户青睐

了解详情 >

- 编辑推荐:
- 革命性创新，动画杀手铜 @scroll-timeline
 - 戏说领域驱动设计（十二）—— 服务
 - ASP.NET Core 6框架揭秘实例演示[16]：内存缓存与分布式缓存的使用
 - .Net Core 中无处不在的 Async/Await 是如何提升性能的？
 - 分布式系统改造方案 —— 老旧系统改造篇

#她的梦想在发光#

HWD科技女性故事有奖征集

活动时间: 2022年3月8日-3月18日

- 最新新闻:
- 乔布斯的创业搭档：他缺乏工程师才能，不得不锻炼营销能力来弥补
 - 美国大厂码农薪资曝光：年薪18万美元，够养家，不够买海景房
 - 两张照片就能转视频！Google提出FLIM帧插值模型

- Android 再推 “杀手级” 功能，可回收 60% 存储空间
- 溺在理财暴雷潮的投资人：本金63万，月兑25元不够卖菜
- » 更多新闻...

历史上的今天：

2018-02-27 多变量微积分笔记6——拉格朗日乘数法

Powered by:
博客园

Copyright © 2022 我是8位的
Powered by .NET 6 on Kubernetes