MEM6810 工程系统建模与仿真

案例 软件

第二讲: uniform$(0,1)$ 随机数

沈海辉

中美物流研究院
上海交通大学

shenhaihui.github.io/teaching/mem6810p
shenhaihui@sjtu.edu.cn

2025年春 (MEM非全日制)

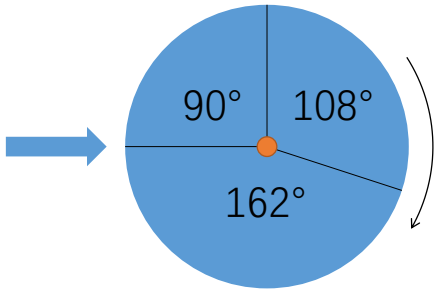
上海交通大學 SHANGHAI JIAO TONG UNIVERSITY
董浩云智能制造与服务管理研究院
CY TUNG Institute of Intelligent Manufacturing and Service Management
(中美物流研究院)
(Sino-US Global Logistics Institute)

目录

- 假设某个库房有一些运输车辆到达需要卸载，经统计发现卸载所需时长的概率分布表如下：

卸载时长/分	概率
10	~~2/6~~ 0.30
20	~~3/6~~ 0.45
30	~~1/6~~ 0.25

- 如何对到达车辆的卸载时长进行模拟呢？
  1. 掷骰子
  2. 转轮盘



- 如果我们知道如何从 0 到 1 之间“随机且均匀”地抽出若干数字，那么我们便可以模拟任何分布！

- 从 $(0,1)$ 区间上的连续均匀分布中抽取的独立随机样本的观测值, 被称为 $\mathrm{uniform}(0,1)$ 随机数 (random numbers), 有时也简称为随机数.
- 如果随机变量 $U \sim \mathrm{uniform}(0,1)$, 那么

$$\mathbb{E}[U] = 1/2,\ \mathrm{Var}(U) = 1/12.$$

- 使用 MATLAB 生成的 10 个 $\mathrm{uniform}(0,1)$ 随机数: 0.8147, 0.9058, 0.1270, 0.9134, 0.6324, 0.0975, 0.2785, 0.5469, 0.9575, 0.9649.
- $\mathrm{uniform}(0,1)$ 随机数的统计性质:
  - 均匀性: $(0,1)$ 区间上的每个值都有一样的可能性.
  - 独立性: 隐含前后的数相互之间无相关性.

- 均匀性

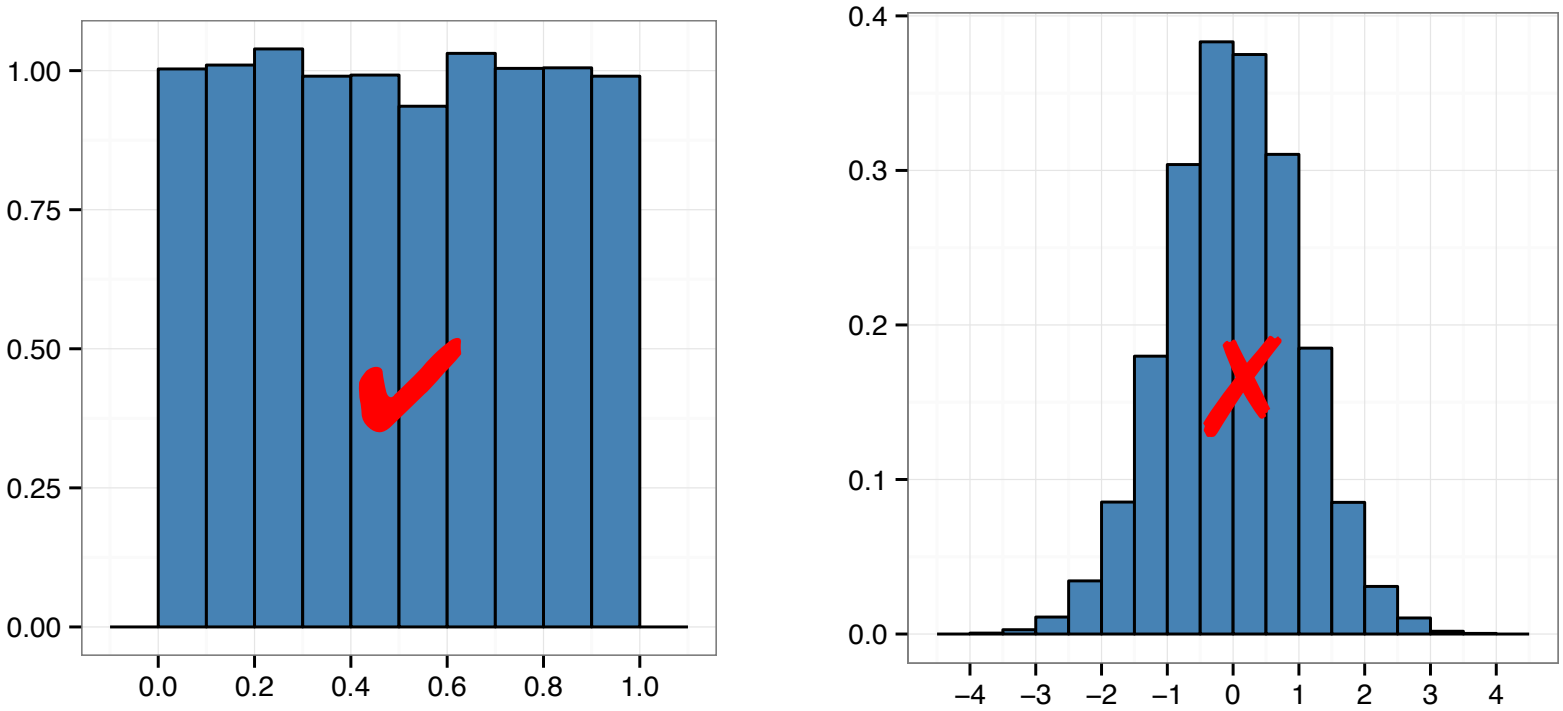


图: 经验概率密度函数 (即, 放缩后的频率直方图, 形状相同)

(*from* ZHANG Xiaowei)

- 独立性

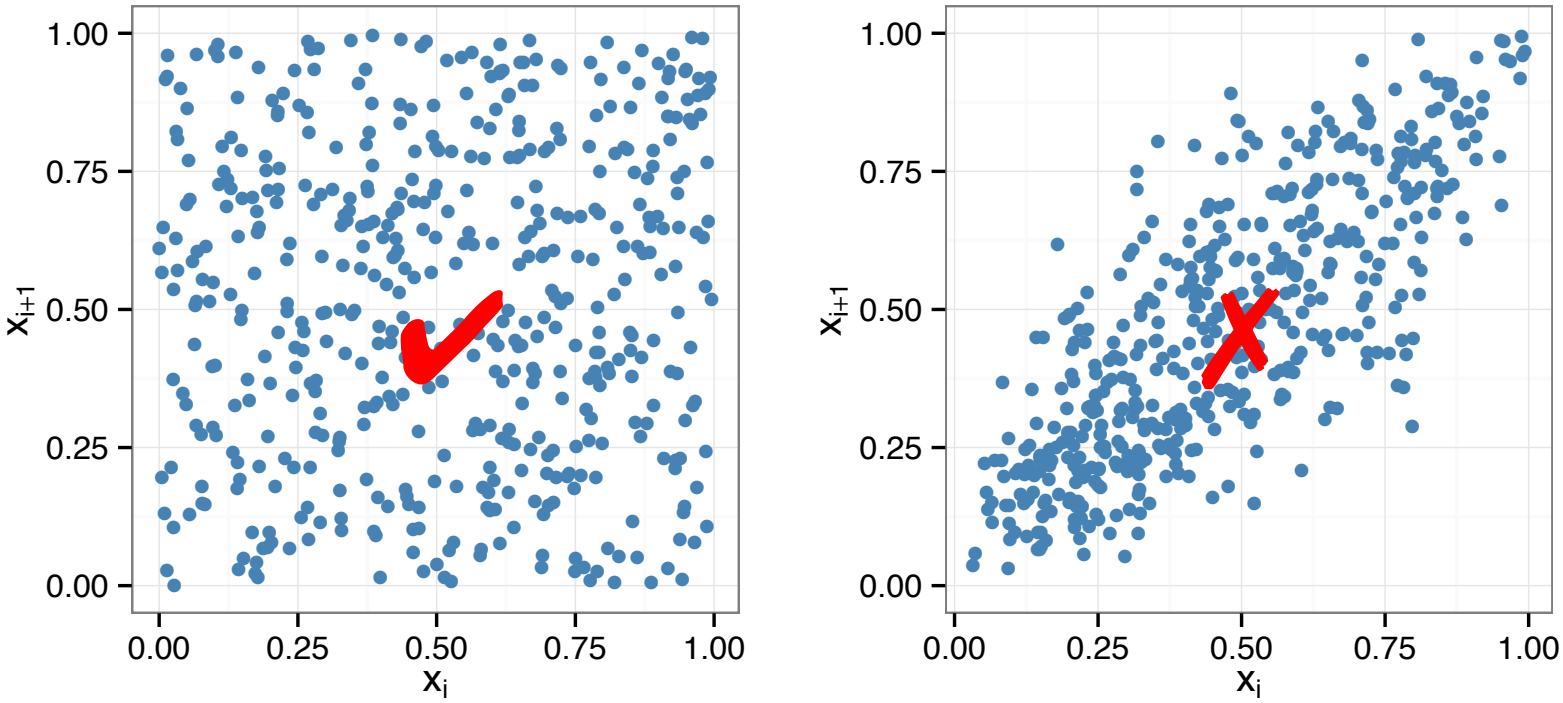


图: 散点图 (*from* ZHANG Xiaowei)

伪随机数

- 计算机无法产生真正的随机性! 它只能产生一些**伪随机**数 (pseudo-random numbers).
- "伪" 意味着不是真正的随机.
  - 随机数是通过某种算法来生成的, 这就消除了随机性.
  - 生成的随机数序列可以被复现.
- 目标: 生成 $(0, 1)$ 范围内的一系列数字, 使他们可以显示出和 $\mathrm{uniform}(0, 1)$ 随机数一样的性质.
  - 统计性质是最重要的.
  - 是否是真随机是次要的.

伪随机数

- 优秀的随机数发生器 (random number generator, RNG) 所需的性质:
  1. 通过统计性检验.
  2. 坚实的理论基础.
  3. 快.
  4. 足够长的周期.
  5. 可移植性好.
  6. 可复现.

- 随机数发生器的一些技术:
  - 线性同余发生器 (Linear Congruential Generator, LCG)
  - 组合线性同余发生器 (Combined LCG)
  - 多重递归发生器 (Multiple Recursive Generator, MRG)

线性同余发生器

- 线性同余发生器是一种简单的早期的随机数发生器.

1. 通过下述递归式产生一系列 0 到 $m-1$ 之间的整数 $x_0, x_1, x_2, \ldots$:

$$x_{i+1} = (ax_i + c) \bmod m, \quad i = 0, 1, 2, \ldots.$$

   - mod 表示取模操作; 初始值 x_0 称为**种子** (seed), a 称为**乘子** (multiplier), c 称为**增量** (increment), m 称为**模数** (modulus).

2. 将 x_i 变换到 0 和 1 之间的数值 u_i:

$$u_i = \frac{x_i}{m}, \quad i = 0, 1, 2, \ldots.$$

   - u_i 的可能取值: $\{0, \frac{1}{m}, \ldots, \frac{m-1}{m}\}$. (可能不完全覆盖!)

- a, c, m, 和 x_0 的选取对统计性质和周期长度有极大的影响.

线性同余发生器

- 例子: 使用 LCG, 并取 $x_0 = 27$, $a = 17$, $c = 43$, 及 $m = 100$.

$$
\begin{aligned}
x_0 &= 27 \\
x_1 &= (17 \times 27 + 43) \bmod 100 = 502 \bmod 100 = 2 \\
u_1 &= 2/100 = 0.02 \\
x_2 &= (17 \times 2 + 43) \bmod 100 = 77 \bmod 100 = 77 \\
u_2 &= 77/100 = 0.77 \\
x_3 &= (17 \times 77 + 43) \bmod 100 = 1352 \bmod 100 = 52 \\
u_3 &= 52/100 = 0.52 \\
x_4 &= (17 \times 52 + 43) \bmod 100 = 927 \bmod 100 = 27 \\
u_4 &= 27/100 = 0.27
\end{aligned}
$$

  周期长度只有 4!

- 访问 `https://xiaoweiz.shinyapps.io/randNumGen` 尝试不同的参数值.

线性同余发生器

- LCG 的一个实际使用 (Lewis et al. 1969): $a = 7^5$, $c = 0$, $m = 2^{31} - 1 = 2{,}147{,}483{,}647$ (一个质数).
  - 它采用 $u_i = \frac{x_i}{m+1}$.
  - 它可通过许多标准的统计性检验.
  - 周期长度 $\approx 2^{31} - 2 \approx 2 \times 10^9$ (超过 20 亿).
- 注: 通过令模数 m 为 2 的幂 (或者接近), 取模运算可以更加高效, 因为大多数计算机是采用二进制来表示数字的.
- 随机计算机算力的增长, 简单的 LCG 如今已经无法胜任了; 实际中我们使用更加复杂的随机数发生器.

更复杂的随机数发生器*

- Combined LCG: 将 J (≥ 2) 个 LCG 组合起来 (其中 $c = 0$).
- 对于 32 位计算机, L'Ecuyer (1988) 提出将 $J = 2$ 个 LCG 组合, 其中 $a_1 = 40{,}014$, $m_1 = 2{,}147{,}483{,}563$, $a_2 = 40{,}692$, 及 $m_2 = 2{,}147{,}483{,}399$.
  1. 从 $[1, m_1 - 1]$ 中为第一个发生器选择种子 $x_{1,0}$, 从 $[1, m_2 - 1]$ 中为第二个发生器选择种子 $x_{2,0}$. 令 $j = 0$.
  2. 计算
  $$x_{1,j+1} = a_1 x_{1,j} \bmod m_1,$$
  $$x_{2,j+1} = a_2 x_{2,j} \bmod m_2.$$
  3. 令 $x_{j+1} = (x_{1,j+1} - x_{2,j+1}) \bmod (m_1 - 1)$.
  (注: mod 使用 floored division, 即, $y \bmod m = y - m\lfloor \frac{y}{m} \rfloor$.)
  4. 返回
  $$u_{j+1} = \begin{cases} \frac{x_{j+1}}{m_1}, & \text{当 } x_{j+1} > 0, \\ \frac{m_1 - 1}{m_1}, & \text{当 } x_{j+1} = 0. \end{cases}$$
  5. 令 $j = j + 1$ 并跳转至第 2 步.

它的周期长度为 $(m_1 - 1)(m_2 - 1)/2 \approx 2 \times 10^{18}$.

更复杂的随机数发生器*

- Multiple Recursive Generator (MRG): 通过使用更高阶的递归来拓展 LCG:

$$x_i = (a_1 x_{i-1} + a_2 x_{i-2} + \cdots + a_k x_{i-K}) \bmod m.$$

- 一个被广泛采用的特例为 MRG32k3a† (L'Ecuyer 1999), 它属于 *combined MRG*, 其中 $J = 2$ 及 $K = 3$.
  - 它的周期长度为 $\approx 3 \times 10^{57}$, 这是一个极大的数.
  - 假设你每秒可以生成 10 亿 (10^9) 个伪随机数, 那么穷尽 MRG32k3a 的周期所需的时间比当前宇宙的年龄还要长!
- 在仿真软件及编程语言中广泛使用的那些知名的随机数发生器, 其统计性质都接受过广泛的检验并被证明有效.
- 当你手中的随机数发生器并不知名或者没有任何记录, 你需要格外小心!
  - 即便是在一些大众商业软件 (如, Excel, Visual Basic) 中用了多年的发生器, 都曾被发现存在一些缺陷 (L'Ecuyer 2001).

† MRG32k3a 或其适配是 MATLAB, R, SAS, Arena 等软件所使用的随机数发生器中的一种.

用Excel产生随机数

- 在 Excel 中, 可以直接使用函数

  `RAND()`

  生成 $\mathrm{uniform}(0,1)$ 随机数.

- 若要生成 $\mathrm{uniform}(a,b)$ 随机数, 其中 $a<b$, 可使用

  `a+(b-a)*RAND()`

- 若要生成 $[a,b]$ 上的离散均匀分布的随机数 (包含端点), 其中 $a<b$, 可使用

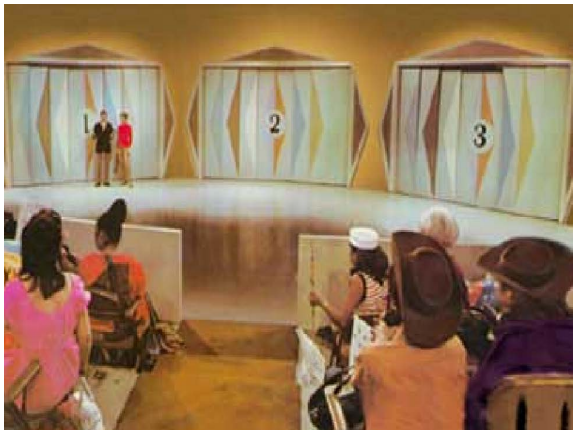
  `RANDBETWEEN(a, b)`

  或者

  `FLOOR(a+(b+1-a)*RAND())`

  `FLOOR.MATH(a+(b+1-a)*RAND())`

- Monty Hall Problem, 又称三门问题、山羊汽车问题
  - 出自美国电视游戏节目 *Let's Make a Deal*, 并以它的主持人 Monty Hall 命名.



决胜21点 (2008)

导演: 罗伯特·路克蒂克
编剧: Peter Steinfeld / 阿兰·里布
主演: 吉姆·斯特吉斯 / 凯文·史派西 / 凯特·波茨沃斯 / 艾伦·余 / 莉萨·拉皮拉 / 更多...
类型: 剧情 / 犯罪
官方网站: www.sonypictures.com/movies/21/
制片国家/地区: 美国
语言: 英语
上映日期: 2008-03-28(美国)
片长: 123分钟
又名: 玩转21点 / 斗智21点 / 攻陷拉斯维加斯 / 21 - The Movie / 21: Blackjack
IMDb链接: tt0478087

豆瓣评分
6.9 60016人评价
5星 8.8%
4星 35.3%
3星 49.3%
2星 6.0%
1星 0.6%

好于 29% 剧情片
好于 48% 犯罪片

想看 看过 评价: ☆☆☆☆☆

写短评 写影评 分享到 推荐

决胜21点的剧情简介 · · · · · ·

Ben Campbell（吉姆·斯特加斯 Jim Sturgess 饰）有着惊人的才华，身为麻省理工高材生的他学业无懈可击，他亦毫无意外地赢得了哈佛医学院的录取通知书。然而30万的高昂学费和生活费令他的大学梦摇摇欲坠。在争取奖学金的面试中，教授对他说胜出者必须要有过人的经历而不是像本这种一张白纸的学生。

Ben在一服装店打工，挣取每小时8美元的薪酬。同时和两个好友准备竞赛2.09以期获得认同和奖金。数学课上本的天才头脑被教授Mickey Rosa（凯文·史派西 Kevin Spacey 饰）发现，Mickey 希望本加入自己的21算法团队，专门去赌场依靠算牌赢得大钱。Ben并不同意，但Ben一直暗恋的女孩Jill Taylor（#凯特·波茨沃斯 Kate Bosworth 饰）也出面诱惑时，Ben开始动摇。

Ben开始了严密的训练，出师的成功让Ben尝到了金钱、虚荣、欲望的权力。同时他和旧友开始疏远，渐渐迷失在赌场的漩涡里。 ©豆瓣

- 最简单的分析:
  - 如果“不换”, 一旦选好结果便确定了,

$$\mathbb{P}(\text{选中车}) = 1/3.$$

  - 如果“换”, 一旦选好结果也确定了 (一开始选中车, 最后会选中羊; 一开始选中羊, 最后会选中车). 因此,

$$\mathbb{P}(\text{最后选中车}) = \mathbb{P}(\text{一开始选中羊}) = 2/3.$$

- 不信? 让我们来做一下仿真实验
  - 访问 http://www.rossmanchance.com/applets/MontyHall/Monty04.html 试一下!
  - 用 Excel 来实现.

- 假设班上有 60 名同学, 那么至少有两个同学生日为同一天 (月日) 的概率为多少? (一年按 365 天计.) 99.41%
- 分析计算
  - 先计算全班生日不同的概率:

$$\mathbb{P}(\text{全班不同}) = \frac{365 \times 364 \times \cdots \times 306}{365^{60}}.$$

  - 于是,

$$\begin{aligned}\mathbb{P}(\text{至少有两人相同}) &= 1 - \mathbb{P}(\text{全班不同}) \\ &= 1 - \frac{365 \times 364 \times \cdots \times 306}{365^{60}} \\ &= 1 - 0.0059 = 0.9941.\end{aligned}$$

- 使用 Excel 进行仿真.

- 未婚妻问题 (Fiancee Problem), 又称公主选驸马问题、秘书问题 (Secretary Problem)
  - 最早由美国数学家 Merrill M. Flood 在 1949 提出.
- 基本问题描述:
  - 要从 N 个人中挑选出一位; N 是一个已知数, 比如, $N = 10$.
  - 候选者以随机 (谁先谁后概率均等) 的顺序到来.
  - 我们看到候选者之后, 会为TA打一个分数 (不会出现同分):
    - 这个分数只与候选者的特质有关, 与出现顺序无关;
    - 可理解为候选者的客观的优秀 (匹配) 程度.
  - 看到一位候选者之后, 我们有两种选择:
    - 选择接受, 则挑选环节结束;
    - 选择拒绝, 则继续看下一位, 并且之后不能再反悔重新选TA.
  - 如果前 $N-1$ 位都没接受, 则必须接受第 N 位.
  - **问题**: 采用何种策略, 可以以最大的概率选择到真正最优秀 (最匹配) 的人?

分析计算*

- 已知最优的策略具有如下结构: 拒绝前 k 人, 从第 $k+1$ 位起, 一旦TA的分数超过一开始的 k 人, 就接受TA; 否则继续.
  - 可通过**动态规划**的方法来得出严格的证明.
- 在最优策略的结构下, 如何确定最优的 k (记为 k^*)?
  - 以 $\mathbb{P}(k)$ 表示选中最优秀 (最匹配) 者的概率.
  - 先推导出 $\mathbb{P}(k)$ 关于 k 的表达式.
  - 再求解使 $\mathbb{P}(k)$ 最大的 k, 即 k^*.
- 特殊情形
  - 若 $N=2$, 任何策略下, 选对的概率都为 $1/2$, 问题退化; 故以下只考虑 $N \geq 3$ 的情形.
  - $k=0$, 对应情况为, 一定接受第一位, 此时 $\mathbb{P}(0)=1/N$.
  - $k=N-1$, 对应情况为, 一定接受第 N 位, 此时 $\mathbb{P}(N-1)=1/N$.

分析计算 (续)*

对于 $1 \le k \le N-1$,

$$\begin{aligned}
\mathbb{P}(k) &= \sum_{i=k+1}^{N} \mathbb{P}(\text{选中第 } i \text{ 个} \cap \text{第 } i \text{ 个为最优}) \\
&= \sum_{i=k+1}^{N} \mathbb{P}(\text{选中第 } i \text{ 个} | \text{第 } i \text{ 个为最优}) \mathbb{P}(\text{第 } i \text{ 个为最优}) \\
&= \frac{1}{N} \sum_{i=k+1}^{N} \mathbb{P}(\text{选中第 } i \text{ 个} | \text{第 } i \text{ 个为最优}) \\
&= \frac{1}{N} \sum_{i=k+1}^{N} \mathbb{P}(\text{前 } i-1 \text{ 人中的最优者在前 } k \text{ 人中} | \text{第 } i \text{ 个为最优}) \\
&= \frac{1}{N} \sum_{i=k+1}^{N} \frac{k}{i-1} = \frac{k}{N}\left(\frac{1}{k} + \frac{1}{k+1} + \cdots + \frac{1}{N-1}\right).
\end{aligned}$$

分析计算 (续)*

对于 $2 \le k \le N-1$, 有

$$\mathbb{P}(k) = \frac{k}{N}\left(\frac{1}{k} + \frac{1}{k+1} + \cdots + \frac{1}{N-1}\right),$$

$$\begin{aligned}\mathbb{P}(k-1) &= \frac{k-1}{N}\left(\frac{1}{k-1} + \frac{1}{k} + \frac{1}{k+1} + \cdots + \frac{1}{N-1}\right) \\ &= \frac{1}{N} + \frac{k-1}{N}\left(\frac{1}{k} + \frac{1}{k+1} + \cdots + \frac{1}{N-1}\right).\end{aligned}$$

因此,

$$\begin{aligned}\mathbb{P}(k) - \mathbb{P}(k-1) &= \frac{1}{N}\left(\frac{1}{k} + \frac{1}{k+1} + \cdots + \frac{1}{N-1}\right) - \frac{1}{N} \\ &= \frac{1}{N}\left(\frac{1}{k} + \frac{1}{k+1} + \cdots + \frac{1}{N-1} - 1\right).\end{aligned}$$

注意到, 该等式在 $k = 1$ 时, 也成立.

分析计算 (续)*

进一步注意到以下几点:

- $\mathbb{P}(k)-\mathbb{P}(k-1)$ 随着 k 增大而减小;
- $k=1$ 时, $\mathbb{P}(1)-\mathbb{P}(0)=\frac{1}{N}\left(1+\frac{1}{k+1}+\cdots+\frac{1}{N-1}-1\right)>0$;
- $k=N-1$ 时, $\mathbb{P}(N-1)-\mathbb{P}(N-2)=\frac{1}{N}\left(\frac{1}{N-1}-1\right)<0$.

因此, 必定存在一个 k, 使得 $\mathbb{P}(k)$ 取到最大值, 该值即为所求 k^*. 且 k^* 必定满足, $\mathbb{P}(k^*)-\mathbb{P}(k^*-1)\geq 0$, $\mathbb{P}(k^*+1)-\mathbb{P}(k^*)<0$. 换言之, k^* 为满足条件

$$\mathbb{P}(k)-\mathbb{P}(k-1)\geq 0, \text{ 即,}$$
$$\frac{1}{k}+\frac{1}{k+1}+\cdots+\frac{1}{N-1}\geq 1,$$

的最大的 k.

- 结论: 最优的策略为, 拒绝前 k^* 人, 从第 k^*+1 位起, 一旦TA的分数超过一开始的 k^* 人, 就接受TA; 否则继续. 其中 k^* 为满足 $\frac{1}{k}+\frac{1}{k+1}+\cdots+\frac{1}{N-1}\geq 1$ 的最大的 k, 且在该策略下, 选中最优的概率为 $\mathbb{P}(k^*)=\frac{k^*}{N}\left(\frac{1}{k^*}+\frac{1}{k^*+1}+\cdots+\frac{1}{N-1}\right)$.
- 若 $N=10$, 则 $k^*=3$, $\mathbb{P}(k^*)=0.3987$;
  若 $N=50$, 则 $k^*=18$, $\mathbb{P}(k^*)=0.3743$;
  若 $N=100$, 则 $k^*=37$, $\mathbb{P}(k^*)=0.3710$.
- 通过进一步的分析, 可以证明, 当 $N\to\infty$ 时,

$$\frac{k^*}{N}\to\frac{1}{e},\quad \mathbb{P}(k^*)\to\frac{1}{e},$$

  其中 $e:=\lim\limits_{n\to\infty}\left(1+\frac{1}{n}\right)^n\approx 2.7183$ 为自然常数, $\frac{1}{e}=0.3679$.
- 使用 Excel 进行仿真.