# G3实验10 – 统计模拟

## 1简介

统计意义上的模拟意味着使用计算机生成的 *伪随机数* 模拟真实事件或数据集，或从给定分布生成观察结果或随机过程的实现。 这本身就是一个完整的研究领域。 在一个研讨会我们只能抓住这个主题的表面。 进一步阅读，三最后列出了参考书。

首先，我们看看如何生成具有均匀分布的随机变量，然后是如何生成将均匀随机变量转换为具有其他分布的随机变量。 最后你会的通过两个例子，使用模拟。

## 2生成一致随机数

与物理设备产生的随机数（例如滚动模具或绘图）形成对比国家彩票中的一个数字）计算机生成的随机数被正确调用*伪*随机因为它们是由确定性函数产生的。 这样的功能是称为随机数发生器（RNG），如果选择得当，则称为数字序列它产生的似乎是不可预测的，相互独立的。 大多数RNG都是旨在生成均匀分布在0和1之间的数字。有些工作比别人好。

一个简单但（可能）非常有效的随机数发生器是 *同余的发电机* 。 给定$a$，$b$，$m$和$y_0$，可以获得$N$个伪随机数的序列从递归

$$对于 n = 1，...，N, \quad y_n = ay_{n-1} + b \pmod{m} 。$$

换句话说，$y_n$是将$a_{n-1} + b$除以$m$时的余数。 通常$y_n$就是除以 $m$得到0到1之间的数字，（希望）可以看作是一个数字观察从统一$[0, 1]$的分布。 值$y_0$原样称为“种子”用作序列的起点。 这个想法是有效的，因为如果值 $a$， $b$和$m$是如果选择得好，连续余数 $y_1$，$y_2$，... 的序列几乎没有可辨别的结构。

**练习：** 一个简单的同余生成器

$$yn \equiv 5y_{n-1} + 1 \pmod{16}$$

如果我们开始与值 $y_0 = 9$，我们得到$Y_{1 \equiv 5 \times 9} + 1$（模16）= 14。在该R这是可以做到使用运算符%%如下：

> （5 * 9 + 1）%% 16
[1] 14

1

**第2页**

该序列中的下一个号码为 $y_{2 \equiv 5 \times 14}$ + 1（模16）= 7，等等。

创建以下功能，确保您了解所有命令，并随身携带
以下练习：

```
trivial.cg < – function（n = 1，y = 9）{
 out < – rep（NA，n）
 for（i in 1：n）{
  y < – （5 * y + 1）%% 16
  out [i] < – y / 16
 }
 出
}
```

1.调用trivial.cg返回一个种子为9的随机数。检查一下
正确的答案。

2.调用trivial.cg返回一个种子为14的随机数。

3.调用trivial.cg返回两个种子为9的随机数。你得到什么
第二个数字?

4.调用trivial.cg返回20个种子为9的随机数。

请注意，最后四个数字与前四个数字完全相同。 这种一致性
发电机的 *周期*为16 – 如果你考虑它，任何$m$ = 16的发电机
保证在最多16次迭代后重复自己。 大多数RNG产生序列
以这种方式重复，但好的发电机的周期远大于16（对于
例7 $\times 10^{12}$ ）。

### 2.1 R中的**不同RNG**

以获得 *N个统一[0，1]*中的R的随机数，使用该函数

> runif（n）

R中有几种类型的RNG，所有这些都比一致性更先进
发电机，但依靠类似的原则。

默认方法是“Mersenne–Twister”方法。 这期间为$2^{19937} - 1$，
这大约是$10^{6000}$ 。 如果你要求R计算这个数字，它会返回Inf！ 把它放进去
上下文：在现代计算机上，您可能能够生成2000万个随机均匀
每秒数字（尝试system.time（runif（2e7））以确切知道需要多长时间）。

现在一天有86400秒，一年有365天; 因此你可以生成约
6 。 每年$3 \times 10^{14}$个均匀随机数。 按照这个速度，它仍然需要更多
这个RNG需要重复$10^{5985}$年 - 此时太阳系将会很久
已不复存在，这可能是一个更大的问题！ 如果由于某种原因你想
指定另一个RNG，使用RNGkind函数和您要使用的RNG的名称。
例如

2

**第3页**

> RNGkind（“Wichmann-Hill”）

可以使用找到所有可用RNG的名称

>帮助（RNGkind）

通常无需更改默认设置。 你也可以强制runif使用你的
拥有自己的RNG，但如果您认为自己有更好的RNG，那么您只想这样做
R中的那些

**2.2设定种子**

在第一个例子中，种子 $y_0$被选择为9.相同的数字序列是
每次使用相同的种子时获得。 类似地，由runif调用的RNG需要a
种子。 当然，每次调用时，runif都不会使用相同的种子，否则你会这样做
每次调用它时都会获得相同的随机数序列 - 通常这样做
不想要（除了有时，见下文）。

每次调用runif时，它都会访问工作空间中名为.Random.seed的对象。 它
使用它来计算所需的伪随机数，然后覆盖.Random.seed
准备下次你碰巧使用runif。

将默认RNG更改为另一种称为“Marsaglia-Multicarry”的类型：

> RNGkind（“Marsaglia-Multicarry”）

然后输入

> .Random.seed
[1] 401 247803058 64738443

注意使用正确的大写和小写字母。 你的种子将与之不同
一个在这里。 第一个数字401编码正在使用的RNG，最后两个整数是
种子。

Marsaglia-Multicarry方法的种子使用2个整数; 种子为默认
Mersenne-Twister方法使用625个整数。

使用保存当前种子

>种子< - .Random.seed

现在获得5次统一观察并再次查看随机种子。

> runif（5）
[1] 0.51476200 0.08528351 0.05742169 0.76459165 0.06405747
> .Random.seed
[1] 401 1852444774 328143383

注意种子已经改变了。 将种子重置为使用前的种子

3

**第4页**

> .Random.seed < – 种子
> runif（5）
[1] 0.51476200 0.08528351 0.05742169 0.76459165 0.06405747

命令runif返回与之前完全相同的数字序列，如同相同
使用种子。

你 **永远不** 应该组成一个新的.Random.seed，因为任意值都不需要 – 通常会产生良好的RNG。 在上面的例子中.Random.seed被简单地替换为早期种子，所以我们知道这是可以接受的。 但是，R中有一个允许的功能一个“选择”种子，保证所产生的RNG的良好行为，并且易于使用：

> set.seed（25）
> .Random.seed
[1] 401 539310548 –611118395
> runif（5）
[1] 0.09414048 0.07374299 0.79480559 0.99556102 0.29460221
> set.seed（25）
> .Random.seed
[1] 401 539310548 –611118395
> runif（5）
[1] 0.09414048 0.07374299 0.79480559 0.99556102 0.29460221

set.seed的参数是您选择的任何整数。

**种子的初始价值是多少？** .Random.seed因为你在你的工作区已经在G3研讨会上使用了runif（或rnorm，rexp等）。 如果你用新的R开始工作区，种子接受什么价值？ 仅创建对象.Random.seed
首次需要时 – 使用存储在计算机内部的整数启动它
那个时钟。

**我们为什么要设定种子？** 通常我们想要产生不同的序列
每次随机数，所以我们接受.Random.seed的当前值。 但一些–
确保生成*相同*的随机数集非常有用 。
假设您在MSc项目中运行模拟研究并写出结果。

后来你的主管说："这一切都很好，但你的标准错误是什么？模拟估计？"如果你不知道种子那么你需要做一个全新的模拟并因此更改所有结果，以便计算标准误差。如果你知道种子，你可以模拟相同的数据并计算标准误差，需要对您的写作进行最小的更改。

设定种子的其他原因包括比较不同的统计程序或使用相同模拟数据的不同估计量; 并允许其他人复制你的结果。

4

**第5页**

### 2.3评估RNG的属性

有许多方法可以评估RNG的分布特性。 主要要求 – 产生的伪随机数表现得像独立观察从所要求的分布，在这种情况下均匀*[0, 1]*。 对均匀性的简单测试是构造为间隔的频率表在*[0, 1]*，并做配合的卡方优度测试。 可以进行许多其他测试，例如Kolmogorov-Smirnov拟合优度检验（在R中命令ks.test）。 选择将取决于最均匀的偏差相关。

离开独立也有许多测试。 一种方法是绘图连续的模拟数对，（ $U_1$，$U_2$ ），（ $U_3$，$U_4$ ），.... 对于差的同余RNG这些点落在格子上。

**示例：**创建一个非常类似于trivial.cg的新函数，名为test.cg，基于同余发生器 $y_{i+1} = 1229 * y_i + 1$（mod2048）。 然后获得1000伪随机数字使用

> ruout < – test.cg（1000）

并根据奇数观察绘制偶数观测值。

>情节（ruout [（1：500）* 2-1]，ruout [（1：500）* 2]，pch =“。”）

请注意，所有绘制的点都位于5行上。 这是一个点阵。 暗示如果我们知道观察 $U_i$与 *i*奇数的值，那么$U_{i+1}$的值将是五个价值观之一。 观察结果独立的假设现在看起来非常好脆弱。

现在将RNG更改回默认的Mersenne-Twister方法：

> RNGkind（“默认”）

并使用此RNG获得类似的图。 请注意，没有可辨别的晶格，表明这是一个更好的RNG。

这里给出的例子只是简单说明序列的方式
显得"非随机"。 在实践中，还有许多其他随机性测试是好的
RNG应该通过。 有一套标准的测试称为"Diehard电池"，即
通常用于检查新RNG的性能。 Mersenne-Twister发电机通过
所有这些; 很少有其他RNG可以做到。 维基百科上的文章，网址为https://en.wikipedia.org/wiki/Diehard_tests总结了测试。

### 3来自其他分布的随机数

通常，为了从其他分布模拟，一个或多个均匀随机数是
生成，然后转换为给出正确的分布属性。 改造

五

第6页

从均匀*[0，1]*至一个从均匀的随机值 *[A，B]*的分布是微不足道的并且是
离开作为锻炼。

以下示例说明了从com-获取随机值的简单方法
mon分布（虽然这些并不总是最有效的方法）。

#### 3.1伯努利

为了模拟具有参数*p*的伯努利分布的随机值 *X*，生成a
均匀随机数 *U*，并使用计算*X.*

$$X = \begin{cases} \text{如果 } U \leq p, & \text{则为1} \\ \text{如果 } U > p, & \text{则为0} \end{cases}$$

#### 3.2二项式

二项式（ *n，p* ）随机变量是*n*个独立伯努利（ *p* ）随机变量的总和。
使用上面的伯努利（ *p* ）方法生成 $X_1$，...，$X_n$，然后计算$Y = \sum_{i=1}^{n} X_i$
作为二项分布所需的值。

请注意，如果 *n*很大，则需要大量的均匀随机数
每个二项式值，这将是缓慢的。 在那种情况下，其他方法可能是优选的
作为下面的下一个。

#### 3.3任意离散随机变量

假设 *X*具有已知的概率质量函数*P（X = R）*=对于*r = 0，1，2*个$P_R$, ...。
从均匀生成 *U [0，1]，*然后计算X使用

□
□□□0如果用户 $U \leq P_0$ 的
1如果 $p_0 < U \leq P_0$ 的 $+ P_1$
2如果 $p_0 + P_1 < U \leq P_0$ 的 $+ P1 + P2_2$

$$X = \begin{cases} \square\square\square\square\square\square k \text{ 如果 } \sum_{i=0}^{k-1} p_i < U \leqslant \sum_{i=0}^{k} p_i \\ \ldots \end{cases}$$

### 3.4带参数 $\lambda$的 指数

从均匀生成 $U[0, 1]$，然后计算$X = -\frac{1}{\lambda}$记录（$U$）。 这是因为

$$P(X \leqslant x) = P\left(-\frac{1}{\lambda}\log(U) \leqslant x\right) = P(U \geqslant e^{-\lambda x}) = 1 - e^{-\lambda x}$$

这是指数（$\lambda$）分布的分布函数 。 上面的最后一步
如下因为如果 $U$～统一$[0, 1]$则$P(U \geqslant U) = 1 - U$对于所有$u \in [0, 1]$。





### 3.5标准正态分布

#### 中央限制法

如果 $U_1, U_2, \ldots, U_n$的独立统一$[0,1]$的随机变量那么他们有平均1/2
和方差1/12; 因此分配

$$\left(\sum_{i=1}^{\tilde{n}} U_i - \frac{\tilde{n}}{2}\right) \Big/ \sqrt{\frac{\tilde{n}}{12}}$$

倾向于正常（0,1）为 $n \to \infty$ 。 $n$的一个方便的值是12.然后我们得到

$$Z = \sum_{i=1}^{12} U_i - 6 \sim N(0, 1) \text{ 大约}$$

注意，这种分布的尾部不是很准确，特别是 $Z$不能超过
± 6.这对于大规模模拟或尾部行为很重要是一个问题
（如极值模拟）。

#### Box-Müller方法

先前的方法每个需要十二个（或通常 $n$个 ）均匀随机数
生成正常随机值。 Box-Müller方法是一种精确的变换方法
使用指数和均匀来生成两个独立的标准正常值。

设 $R$~指数（$\frac{1}{2}$）和 $W$～统一$[0, 2\pi]$，并限定$X$和$Y$通过

$$X = \sqrt{R}\cos W \text{和} Y = \sqrt{R}\sin W$$

然后可以证明 $X$和$Y$是独立的并且具有标准的正态分布。
由于我们知道如何生成制服和指数，我们可以使用这些来计算

标准正态随机变量。

### 3.6反演方法

假设 *X*是具有已知累积分布函数的连续随机变量
*F*（*x*）。 如果*F*（*x*）严格增加则反函数*F.* －1（*u*）是独一无二的。 如果分析
*F*的 表达式 －1（*u*）可以找到，然后 使用反演 从 *F* 模拟 很容易
方法。

从均匀生成 *U [0，1]*，然后设置*X* = *F* －1（*U*）。 这是因为：

$$P\ (X \leqslant x) = P.\ \left(F^{-1}(U) \leqslant X\right) = P.\ \left(\hat{u} \leqslant F(x)\right) 的 = \int_0^{F(x) 的} 1\, du = F\ (x)$$

按要求。 该方法是方法3.3节的连续模拟
分配。 此外，上述用于指数分布的方法是特殊的
这种情况。





f（x）

U × M × g（Y） M × g（x）

ÿ X

图1：从密度 *f*（阴影区域） 采样的拒绝方法的图示 。 一个
随机变量 *Y*是从密度*g*（与上面的曲线成比例）和该值绘制的
被保留的概率 *F（Y）/镁 （Y）*，即当*U*≤*F（Y）/镁 （Y）*其中，*U*～*U（0,1）*。 在
这个例子，*U*> *f*（*Y*）/ *Mg*（*Y*）因此候选值被拒绝。 候选人的价值观是
在 *g*相对于*f*大的区域中最常拒绝 ，因为*g*生成太多
这些地区的价值观。

### 3.7拒绝方法

假设我们想要使用pdf *f*从分布中模拟 *X*，并且我们知道如何
使用pdf *g*和与*f*相同的支持来模拟随机变量 *Y.*
如果对于所有*x* 存在常数 *M*使得*f*（*x*）≤*Mg*（*x*），那么我们可以使用拒绝
方法。 算法如下：

1.从$G$和$U$从统一生成 $Y[0,1]$。

2.如果 $U<f(Y)/Mg(Y)$，则设定$X=Y$.

3.否则拒绝 $Y$（和$U$）并返回步骤1。

图1提供了对拒绝方法工作原理的一些了解。 假设对 $f$和$g$的 支持是R.以下数学练习可以帮助你更详细地了解方法：

1.如果对于所有$x$， 如果 $f(x)\leqslant Mg(x)$，则必须得到$M\geqslant 1$ 。（ **提示**：记住$f$和 $g$是密度）。

2.注意到

$$P(U<f(Y)/Mg(Y))P=\int_{y\in R}(U<f(y)/Mg(y))\,g(y)\,dy$$

通过总概率定律，表明接受生成的概率算法的步骤2中的$Y$是$1/M$.

8

第9页

3.现在注意随机变量 $X$ 的分布函数来自拒绝算法是

$$F(x)=P(X\leqslant x)=P(Y\leqslant x|U<f(Y)/Mg(Y))=\frac{P([Y\leqslant x]\cap[U<f(Y)/Mg(Y)])}{P(U<f(Y)/Mg(Y))},$$

其中 $Y$的密度为$g$。 设$E$表示事件$[Y\leqslant x]\cap[U<f(Y)/Mg(Y)]$。 通过写 $P(E)=\int_{y\in R}P(E|Y=y)\,g(y)\,dy$（或者，如果你能找到一种更简单的方法）表明 $X$具有密度$f$，因此拒绝算法起作用。

实际上，如果大量的 $Y$在被接受之前被拒绝，那么算法会很慢；由于接受概率为$1/M$，我们可以看出理想情况下$M$应该小到1可能的（如果 $g=f$，则通常$M=1$，这是最小的可能值，在这种情况下所有的 $Y$都被接受！）。 这意味着我们应该尝试选择一个尽可能接近的$g$在某种意义上可能是 $f$。

**示例**：假设我们希望模拟$X\sim\beta(\alpha,\beta)$， $\alpha\geqslant 1$且$\beta\geqslant 1$这种分布的密度是

$$F(X)=X^{1\alpha-}(1-x)\text{的}^{\beta-1}\Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha+\beta)\text{对于}x\in[0,1]。$$

让 $Y\sim$统一$[0,1]$，从而使$G(X)=1$对于$x\in[0,1]$。

当$x=(\alpha-1)/(\alpha+\beta-2)$时， 你可以检查 $f(x)$是否最大化，因此 $f(x)$的最大值是

$$M = \frac{(\alpha-1)^{\alpha-1}(\beta-1)^{\beta-1}\Gamma(\alpha)\Gamma(\beta)}{(\alpha+\beta-2)^{(\alpha+\beta-2)}\Gamma(\alpha+\beta)}$$

使得 $F(X) \leqslant M = Mg$的 $(X)$为0和1之间$x$若要使用排斥的方法，模拟
$Y$均从统一$[0,1]$和也从$\ddot{u}$统一$[0,1]$。 如果

$$U < \frac{Y^{\alpha-1}(1-Y)^{\beta-1}(\alpha+\beta-2)^{(\alpha+\beta-2)}}{(\alpha-1)^{\alpha-1}(\beta-1)^{\beta-1}} \quad \text{即} \quad \left(\frac{f(Y)}{镁(Y)}\right)$$

那么我们接受 $X = Y$作为β分布的值。 否则我们丢弃两个$Y$.
和 $U$重复这个过程，直到上述不等式成立，当我们接受$X = Y$。

**问题：**为什么在这个例子中需要$\alpha \geqslant 1$， $\beta \geqslant 1$的限制？

### 3.8练习

使用runif（）和上述信息来模拟每个观察结果的100个样本
以下分发。 使用图表，表格和摘要统计信息来验证您的
样本似乎来自正确的分布。

1.统一$[-3,10]$。

9

**第10页**

2.伯努利（0.6）。

3.二项式（15,0.6）。

4.指数（$\frac{1}{2}$）。

5.正常（2,5）使用中心极限和Box-Müller方法。

6.具有 $F(x) = (1+e^{-x})^{-1}$， $-\infty < x < \infty$ 使用反演方法。

7.使用拒绝方法的Beta（2,3.5）。

**注意：**这些是练习，以帮助您了解一些生成算法
来自不同分布的随机数。 在实践中，我们将使用标准R.
功能。 例如，要模拟Beta分布中的值，请使用函数rbeta。
它应该比我们可能编写的任何内容更加可靠和快速 - 并且速度更快
如果函数被多次调用很重要，这很可能是随机的
数字生成器。

### 4模拟的用途

模拟随机变量的想法最初可能看起来很奇怪，但它有
许多统计用途，例如：

- 测试或研究其他R函数的行为，例如> hist（rnorm（100,3,2））;

- 为了评估统计方法，我们可以模拟已知分布的数据并查看方法如何执行;
- 设计实验并随机进行临床试验;
- 研究可用表示的概率模型或随机过程条件分布，但难以分析解决;
- 实现数值方法，如蒙特卡洛积分函数（见以下）或模拟退火以优化功能;
- 通过使用引导程序或其他重新采样方法获得可靠的估计值;
- 使用MCMC方法从贝叶斯后验分布中进行采样，如下所示 STATG / M004。0031

**4.1蒙特卡洛整合**

你已经满足了梯形法则和辛普森计算积分的规则，
在实验室6中。另一种方法使用模拟（通常称为蒙特卡罗）方法
*估计*积分（好像它是一个未知参数）。



**第11页**

假设所需的积分可以表示为

$$\Theta = \int_{一个}^{b} \phi(x) f(x) dx = E[\phi(X)],$$

其中 $f$是概率密度函数。 模拟随机样本$X_1, X_2, \ldots, X_n$
来自 $f$。 设$Y_1 = \phi(X_1), \ldots, Y_n = \phi(X_n)$。 那么积分的估计是

$$\Theta = \frac{1}{\tilde{n}} \sum_{i=1}^{\tilde{n}} \phi(X_i) = \bar{Y}。$$

也就是说，我们的估算器是 $Y_i$的样本均值 。 如果我们生成另一个随机样本
我们会得到一个不同的估计。 我们可能估计的分布差异
得到（即估计量的方差）是

$$\mathrm{var}(\Theta) = \frac{1}{n^2} \sum_{i=1}^{\tilde{n}} \mathrm{var}(\phi(X_i)) = \frac{\mathrm{var}(Y)}{\tilde{n}},$$

其中 $Y = \phi(X)$。 通常我们会选择$n$足够大，以便我们估算Θ
足够精确，即，使其标准误差足够小。 估计的
标准错误是

se（Θ） 是的 ≈ $\tilde{n}$ 其中 $S_Y^2 = \frac{1}{n-1}\sum_{i=1}^{\tilde{n}} (\phi(X_i) - \Theta)^2$。

**例如：**假设我们希望集成日志（日志 *(X)*）*2*和10之间。即*X,*
计算

$$\Theta = \int_2^{10} \log(\log(x))\, dx。$$

我们可以通过输入 $\phi(x) = 8\log(\log(x))$ 和 $f(x) = \frac{1}{8}$ 来表达上述形式 对于 *2≤X≤10，*使*X*～统一*[2,10]*。

因此，我们产生 $X_1, X_2 \ldots, X_n$的统一*[2,10]*的分布，并估计积分如$\Theta = \frac{8}{\tilde{n}}\sum_{i=1} \log(\log(X_i))$。 在R，Θ及其近似标准误差（例如，对于$n = 1000$）可以使用获得

```
> n < – 1000; ruout < – runif（n，min = 2，max = 10）
> phi.x < – 8 * log（log（ruout））
> mean（phi.x）
[1] 3.984976
> sd（phi.x）/ sqrt（n）
[1] 0.07529205
```

11

**第12页**

**注释：**上面的积分可以很容易地用其他方法计算，比如梯形规则。 但蒙特卡洛集成可以很难应用高维和更复杂的积分，其他方法变得难以处理。例如，如果 *f*是*k*维上的函数，则很容易生成*k*维均匀的随机样本（取 *k*个单变量制服的矢量 ），因而是积分可以用与上面完全相同的方式估算。 另一方面，方法就像如果 *k*很大，则梯形规则迅速变得不可行 （如果你评估*f*，例如，25点每个维度，你最终需要25 *k*函数评估来计算你的积分！）。

一些方法（例如，梯形方法）不能在无限限制之间进行整合。 您蒙特卡罗积分可以有无限的限制，提供随机范围生成的数字具有相同的限制（请参阅下面的练习）。 还有几个聪明可用于减少Θ方差的技巧。 有关详细信息，请参阅本书参考书目。

**练习：**使用蒙特卡洛积分计算积分

$$\Theta = \int_0^{\infty} x^2 \bar{e}^{X} DX。$$

您需要考虑从中采样的分布 $f$ （提示：它必须具有支持[0 ， ∞ ））。
生成大小为1000的样本并获得Θ的估计值。 这个值如何比较
理论答案？ 重复以获得几个不同的Θ估计值。 结果如何
如你所料可靠吗？ 找出估算的近似标准误差。

## 4.2仿真建模

对于复杂概率模型，获取分析可能非常困难或不可能
表达所需的分布。 例如，为长时间的交通流量建模
由于问题的复杂性，高速公路的路段难以分析。
在这种情况下可以使用模拟建模来预测交通流量的分布，
并了解它对模型假设和交通网络变化的敏感程度
（例如，由于道路施工而关闭一条车道）。

我们将考虑一个更简单的模型。 建筑物的空调系统包括
冷却装置和空气泵，八个主要空气管道布置在网络中，如图所示
2，网络末端的安全插座。 每个管道都有一个空气过滤器，没有
维护，将在 *指数分布的*时间后被阻止 。 平均时间
每个管道中的过滤器堵塞：管道A，B和C为1年，管道D和管道为2年
E，管道F，G和H为3年。假设每个过滤器堵塞的时间是
相互独立。

空气不必流过每个管道。 但是，如果空气无法流动
（从任何路线）从泵到安全出口，然后空调机组中断
需要昂贵的维修费用。 此时，所有空气过滤器都会同时更换。
任务是模拟空调机组发生故障的时间分布
由于空气过滤器堵塞。





一个 d F

泵 乙 G 出口

C Ë H

空气仅沿箭头方向流动

图2：空调系统中的气流图。

**练习：** 首先写下从泵到安全的五条路径
出口，例如 *{A, F}* 。 现在假设我们有 $T_A$ ， ...， $T_H$ ，每个空气过滤器的时间
受阻。 记下 $X_1$路径*{A, F}*在 $T_A$和 *T*方面被阻塞的时间
$T_F$. 为所有5条路径执行此操作。 记下 $Y$ ，即整个系统被阻止的时间
$X_1$ ， ...， $X_5$的 术语 。

现在编写一个模拟 $T_1, \ldots, T_m$ $m$次的R函数（以$m$为函数）
参数），并返回 $m$ 值为 $Y$ 的向量。。运行你的函数来获得一个向量 1000个模拟故障时间。 绘制输出的直方图，找到平均值和模拟故障时间的方差。 空调机组应该多久使用一次服务以确保连续服务之间只有1%的可能性出现故障?

调整您的功能以打印出最终未阻塞路径的次数是 *{B，G}* 。

上面的分布可以使用分析方法找到，因为所有的分布 – tions是指数级的。 然而，这种方法可以很容易地适用于任何组合分布，假设可以从它们生成伪随机数。 适应你的如果堵塞时间有Weibull dis–，则再次运行以模拟故障时间具有形状参数2和比例参数的分配：管道A，B和C为1，管道为2管道F，G和H的D和E和3

### 相关书籍


- JF Monahan， *统计数值方法* 。 剑桥，2001年。
- BD Ripley， *随机模拟* 。 Wiley，1987。
- JR汤普森， *模拟：建模者的方法* 。 Wiley，1999。


### 最后

现在你已经完成了课程的最后一个研讨会：你已经学到了很多R命令，并希望发现如何找到你 *没有的* 其他命令学到了。 课程Moodle页面包含Tom Short的“R参考卡”链接，这是常用命令的“一目了然”总结。 你可能会觉得这很有用在将来。

13