Chapter Ten
DISCRETE UNIVARIATE DISTRIBUTIONS

1. INTRODUCTION.

1.1. Goals of this chapter.

We will provide the reader with some generators for the most popular families of discrete distributions, such as the geometric, binomial and Poisson distributions. These distributions are the fundamental building blocks in discrete probability. It is impossible to cover most distributions commonly used in practice. Indeed, there is a strong tendency to work more and more with so-called generalized distributions. These distributions are either defined constructively by combining more elementary distributions, or analytically by providing a multiparameter expression for the probability vector. In the latter case, random variate generation can be problematic since we cannot fall back on known distributions. Users are sometimes reluctant to design their own algorithms by mimicking the designs for similar distributions. We therefore include a short section with universal algorithms. These are in the spirit of chapter VII: the algorithms are very simple albeit not extremely fast, and very importantly, their expected time performance is known. Armed with the universal algorithms, the worked out examples of this chapter and the table methods of chapter VIII, the users should be able to handle most distributions to their satisfaction.

We assume throughout this chapter that the discrete random variables are all integer-valued.

1.2. Generating functions.

Let X be an integer-valued random variable with probability vector

$$p_i = P(X=i) \quad (i \text{ integer}) .$$

An important tool in the study of discrete distributions is the **moment generating function**

$$m(s) = E(e^{sX}) = \sum_i p_i e^{si} .$$

It is possible that $m(s)$ is not finite for some or all values $s > 0$. That of course is the main difference with the characteristic function of X. If $m(s)$ is finite in some open interval containing the origin, then the coefficient of $s^n/n!$ in the Taylor series expansion of $m(s)$ is the n-th moment of X.

A related tool is the **factorial moment generating function**, or simply generating function,

$$k(s) = E(s^X) = \sum_i p_i s^i ,$$

which is usually only employed for nonnegative random variables. Note that the series in the definition of $k(s)$ is convergent for $|s| \leq 1$ and that $m(s) = k(e^s)$. Note also that provided that the n-th factorial moment (i.e., $E(X(X-1) \cdots (X-n+1))$) of X is finite, we have

$$k^{(n)}(1) = E(X(X-1) \cdots (X-n+1)) .$$

In particular $E(X)=k'(1)$ and $Var(X)=k''(1)+k'(1)-k'^2(1)$. The generating function provides us often with the simplest method for computing moments.

It is clear that if $X_1, \ldots, X_n$ are independent random variables with moment generating functions $m_1, \ldots, m_n$, then $\sum X_i$ has moment generating function $\prod m_i$. The same property remains valid for the generating function.

Example 1.1. The binomial distribution.

A **Bernoulli** (p) **random variable** is a $\{0,1\}$-valued random variable taking the value 1 with probability p. Thus, it has generating function $1-p+ps$. A **binomial** (n,p) **random variable** is defined as the sum of n iid Bernoulli (p) random variables. Thus, it has generating function $(1-p+ps)^n$. ■

Example 1.2. The Poisson distribution.

Often it is easy to compute generating functions by explicitly computing the convergent infinite series $\sum s^i p_i$. This will be illustrated for the Poisson and geometric distributions. X is **Poisson** (λ) when $P(X=i)=\frac{\lambda^i}{i!}e^{-\lambda}$ ($i \geq 0$). By summing $s^i p_i$, we see that the generating function is $e^{-\lambda+\lambda s}$. X is **geometric** (p) when $P(X=i)=(1-p)^i p$ ($i \geq 0$). The corresponding generating function is $p/(1-(1-p)s)$. ■

If one is shown a generating function, then a careful analysis of its form can provide valuable clues as to how a random variable with such generating function can be obtained. For example, if the generating function is of the form

$$g(k(s))$$

where g, k are other generating functions, then it suffices to take $X_1 + \cdots + X_N$ where the X_i's are iid random variables with generating function k, and N is an independent random variable with generating function g. This follows from

$$g(k(s)) = \sum_{n=0}^{\infty} P(N=n) k^n(s) \quad \text{(definition of } g\text{)}$$

$$= \sum_{n=0}^{\infty} P(N=n) \sum_{i=0}^{\infty} P(X_1 + \cdots + X_n = i) s^i$$

$$= \sum_{i=0}^{\infty} \left\{ s^i \sum_{n=0}^{\infty} P(N=n) P(X_1 + \cdots + X_n = i) \right\}$$

$$= \sum_{i=0}^{\infty} s^i P(X_1 + \cdots + X_N = i) .$$

Example 1.3.

If $X_1, \ldots$ are Bernoulli (p) random variables and N is Poisson (λ), then $X_1 + \cdots + X_N$ has generating function

$$e^{-\lambda+\lambda(1-p+ps)} = e^{-\lambda p + \lambda p s} ,$$

i.e. the random sum is Poisson (λp) distributed (we already knew this - see chapter VI). ■

A **compound Poisson distribution** is a distribution with generating function of the form $e^{-\lambda+\lambda k(s)}$, where k is another generating function. By taking

$k(s)=s$, we see that the Poisson distribution itself is a compound Poisson distribution. Another example is given below.

Example 1.4. The negative binomial distribution.

We define the **negative binomial distribution** with parameters (n,p) ($n \geq 1$ is integer, $p \in (0,1)$) as the distribution of the sum of n iid geometric random variables. Thus, it has generating function

$$\left(\frac{p}{1-(1-p)s}\right)^n = e^{-\lambda+\lambda k(s)}$$

where $\lambda = n \log(\frac{1}{p})$ and

$$k(s) = \frac{\log(1-(1-p)s)}{\log(p)}$$

$$= -\frac{1}{\log(p)} \sum_{i=1}^{\infty} \frac{(1-p)^i}{i} s^i .$$

The function $k(s)$ is the generating function of the **logarithmic series distribution** with parameter $1-p$. Thus, we have just shown that the negative binomial distribution is a compound Poisson distribution, and that a negative binomial random variable can be generated by summing a Poisson (λ) number of iid logarithmic series random variables (Quenouille, 1949). ■

Another common operation is the mixture operation. Assume that given Y, X has generating function $k_Y(s)$ where Y is a parameter, and that Y itself has some (not necessarily discrete) distribution. Then the unconditional generating function of X is $E(k_Y(s))$. Let us illustrate this once more on the negative binomial distribution.

Example 1.5. The negative binomial distribution.

Let Y be gamma $(n, \frac{1-p}{p})$, and let k_Y be the Poisson (Y) generating function. Then

$$E(k_Y(s)) = \int_0^{\infty} \frac{y^n e^{-\frac{py}{1-p}}}{\Gamma(n)(\frac{1-p}{p})^n} e^{-y+ys} \, dy$$

$$= \left(\frac{p}{1-(1-p)s}\right)^n .$$

We have discovered yet another property of the negative binomial distribution with parameters (n, p), i.e. it can be generated as a Poisson (Y) random variable where Y in turn is a gamma $(n, \frac{1-p}{p})$ random variable. This property will be of great use to us for large values of n, because uniformly fast gamma and Poisson generators are in abundant supply. ■

1.3. Factorials.

The evaluation of the probabilities p_i frequently involves the computation of one or more factorials. Because our main worry is with the complexity of an algorithm, it is important to know just how we evaluate factorials. Should we evaluate them explicitly, i.e. should $n!$ be computed as $\prod_{i=1}^{n} i$, or should we use a good approximation for $n!$ or $\log(n!)$? In the former case, we are faced with time complexity proportional to n, and with accumulated round-off errors. In the latter case, the time complexity is $O(1)$, but the price can be steep. Stirling's series for example is a divergent asymptotic expansion. This means that for fixed n, taking more terms in the series is bad, because the partial sums in the series actually diverge. The only good news is that it is an asymptotic expansion: for a fixed number of terms in the series, the partial sum thus obtained is $\log(n!)+o(1)$ as $n \to \infty$. An algorithm based upon Stirling's series can only be used for n larger than some threshold n_0, which in turn depends upon the desired error margin.

Since our model does not allow inaccurate computations, we should either evaluate factorials as products, or use squeeze steps based upon Stirling's series to avoid the product most of the time, or avoid the product altogether by using a convergent series. We refer to sections X.3 and X.4 for worked out examples. At issue here is the tightness of the squeeze steps: the bounds should be so tight that the contribution of the evaluation of products in factorials to the total expected complexity is $O(1)$ or $o(1)$. It is therefore helpful to recall a few facts about approximations of factorials (Whittaker and Watson, 1927, chapter 12). We will state everything in terms of the gamma function since $n! = \Gamma(n+1)$.

Lemma 1.1. (Stirling's series, Whittaker and Watson, 1927.)

For $x > 0$, the value of $\log(\Gamma(x)) - (x - \frac{1}{2})\log(x) + x - \frac{1}{2}\log(2\pi)$ always lies between the n-th and $n+1$-st partial sums of the series

$$\sum_{i=1}^{\infty} \frac{(-1)^{i-1} B_i}{2i(2i-1)x^{2i-1}}$$

where B_i is the i-th Bernoulli number defined by

$$B_n = 4n \int_0^{\infty} \frac{t^{2n-1}}{e^{2\pi t} - 1}\, dt \; .$$

In particular, $B_1 = \frac{1}{6}, B_2 = \frac{1}{30}, B_3 = \frac{1}{42}, B_4 = \frac{1}{30}, B_5 = \frac{5}{66}, B_6 = \frac{691}{2730}, B_7 = \frac{7}{6}$.
We have as special cases the inequalities

$$(x + \frac{1}{2})\log(x+1) - (x+1) + \frac{1}{2}\log(2\pi) \le \log(\Gamma(x+1))$$

$$\le (x + \frac{1}{2})\log(x+1) - (x+1) + \frac{1}{2}\log(2\pi) + \frac{1}{12(x+1)} \; .$$

Stirling's series with the Whittaker-Watson lower and upper bounds of Lemma 1.1 is often sufficient in practice. As we have pointed out earlier, we will still have to evaluate the factorial explicitly no matter how many terms are considered in the series, and in fact, things could even get worse if more terms are considered. Luckily, there is a convergent series, attributed by Whittaker and Watson to Binet.

Lemma 1.2. (Binet's series for the log-gamma function.)

For $x > 0$,

$$\log(\Gamma(x)) = (x - \frac{1}{2})\log(x) - x + \frac{1}{2}\log(2\pi) + R(x) ,$$

where

$$R(x) = \frac{1}{2}\left\{ \frac{c_1}{(x+1)} + \frac{c_2}{2(x+1)(x+2)} + \frac{c_3}{3(x+1)(x+2)(x+3)} + \cdots \right\} ,$$

in which

$$c_n = \int_0^1 (u+1)(u+2)\cdots(u+n-1)(2u-1)u \; du .$$

In particular, $c_1 = \frac{1}{6}$, $c_2 = \frac{1}{3}$, $c_3 = \frac{59}{60}$, and $c_4 = \frac{227}{60}$. All terms in $R(x)$ are positive; thus, the value of $\log(\Gamma(x))$ is approached monotonically from below as we consider more terms in $R(x)$. If we consider the first n terms of $R(x)$, then the error is at most

$$C\frac{x+1}{x}\left(\frac{x+1}{x+n+1}\right)^x ,$$

where $C = \frac{5}{48}\sqrt{4\pi}e^{1/6}$. Another upper bound on the truncation error is provided by

$$C(1+a+\frac{1}{x+1})(\frac{a}{1+a}+\frac{1}{x+1})^{n+1} + C\frac{x+1}{x}(\frac{1}{1+a})^x .$$

where $a \in (0,1]$ is arbitrary (when x is large compared to n, then the value $\frac{n+1}{x}\log(\frac{x}{n+1})$ is suggested).

Proof of Lemma 1.2.

Binet's convergent series is given for example in Whittaker and Watson (1927, p. 253). We need only establish upper bounds for the tail sum in $R(x)$ beginning with the $n+1$-st term. The integrand in c_i is positive for $u > \frac{1}{2}$. Thus, the i-th term is at most

$$\frac{i!\int_{1/2}^{1}(2u-1)u \; du}{2i(x+1)\cdots(x+i)} = \frac{5(i-1)!}{48(1+x)\cdots(i+x)}$$

$$= \frac{5\Gamma(i)\Gamma(x+1)}{48\Gamma(i+x+1)}$$

$$\le \frac{5}{48}\sqrt{\frac{2\pi(x+i+1)}{i(x+1)}}\, e^{\frac{1}{12i}+\frac{1}{12(x+1)}}\left(\frac{i}{x+i+1}\right)^i\left(\frac{x+1}{x+i+1}\right)^{x+1}$$

(by Lemma 1.1)

$$\leq C\left(\frac{i}{x+i+1}\right)^i\left(\frac{x+1}{x+i+1}\right)^{x+1}$$

where $C=\frac{5}{48}\sqrt{4\pi}e^{1/6}$ (use the facts that $x>0, i\geq 1$). We obtain a first bound for the sum of all tail terms starting with $i=n+1$ as follows:

$$\sum_{i=n+1}^{\infty} C\left(\frac{i}{x+i+1}\right)^i\left(\frac{x+1}{x+i+1}\right)^{x+1} \leq \sum_{i=n+1}^{\infty} C\left(\frac{x+1}{x+i+1}\right)^{x+1}$$

$$\leq \int_n^{\infty} C\left(\frac{x+1}{x+t+1}\right)^{x+1} dt$$

$$= C\frac{x+1}{x}\left(\frac{x+1}{x+n+1}\right)^x .$$

Another bound is obtained by choosing a constant $a \in (0,1)$, and splitting the tail sum into a sum from $i=n+1$ to $i=m=\lceil a(x+1)\rceil$, and a right-infinite sum starting at $i=m+1$. The first sum does not exceed

$$\sum_{i=n+1}^{m} C\left(\frac{i}{x+i+1}\right)^i \leq \sum_{i=n+1}^{\infty} C\left(\frac{m}{x+m+1}\right)^i = C\frac{x+m+1}{x+1}\left(\frac{m}{x+m+1}\right)^{n+1}$$

$$\leq C\left(1+a+\frac{1}{x+1}\right)\left(\frac{a}{1+a}+\frac{1}{x+1}\right)^{n+1} .$$

Adding the two sums gives us the following upper bound for the remainder of the series starting with the $n+1$-st term:

$$C\left(1+a+\frac{1}{x+1}\right)\left(\frac{a}{1+a}+\frac{1}{x+1}\right)^{n+1} + C\frac{x+1}{x}\left(\frac{1}{1+a}\right)^x . \blacksquare$$

The error term given in Lemma 1.2 can be made to tend to 0 merely by keeping n fixed and letting x tend to ∞. Thus, Binet's series is also an asymptotic expansion, just as Stirling's series. It can be used to bypass the gamma function (or factorials) altogether if one needs to decide whether $\log(\Gamma(x))\leq t$ for some real number t. By taking n terms in Binet's series, we have an interval $[a_n, b_n]$ to which we know $\log(\Gamma(x))$ must belong. Since $b_n - a_n \rightarrow 0$ as $n \rightarrow \infty$, we know that when $t \neq \log(\Gamma(x))$, from a given n onwards, t will fall outside the interval, and the appropriate decision can be made. The convergence of the series is thus essential to insure that this method halts. In our applications, t is usually a uniform or exponential random variable, so that equality $t=\log(\Gamma(x))$ occurs with probability 0. The complexity analysis typically boils down to computing the expected number of terms needed in Binet's series for fixed x. A quantity useful in this respect is

$$\sum_{n=0}^{\infty} n(b_n - a_n) .$$

Based upon the error bounds of Lemma 1.2, it can be shown that this sum is $o(1)$ as $x \to \infty$, and that the sum is uniformly bounded over all $x \geq 1$ (see exercise 1.2). As we will see later, this implies that for many rejection algorithms, the expected time spent on the decision is uniformly bounded in x. Thus, it is almost as if we can compute the gamma function in constant time, just as the exponential and logarithmic functions. In fact, there is nothing that keeps us from adding the gamma function to our list of constant time functions, but unless explicitly mentioned, we will not do so. Another collection of inequalities useful in dealing with factorials via Stirling's series is given in Lemma 1.3:

Lemma 1.3. (Knopp, 1964, pp. 543,548)

For integer n, we have

$$\log(n!) = (n+\frac{1}{2})\log(n)-n+\log(\sqrt{2\pi})+\sum_{j=1}^{k}\frac{(-1)^{j-1}B_j n^{-(2j-1)}}{(2j-1)(2j)}+R_{k,n}$$

where $B_1, B_2, \cdots$ are the Bernoulli numbers and

$$|R_{k,n}| \leq \frac{4(2k-1)!}{2\pi(2\pi n)^{2k}}$$

is a residual factor.

1.4. A universal rejection method.

Even when the probabilities p_i are explicitly given, it is often hard to come up with an efficient generator. Quantities such as the mode, the mean and the variance are known, but a useful dominating curve for use in a rejection algorithm is generally not known. The purpose of this section is to go through the mechanics of deriving one acceptable rejection algorithm, which will be useful for a huge class of distributions, the class of all unimodal distributions on the integers for which three quantities are known:

1. m, the location of the mode. If the mode is not unique, i.e. several adjacent integers are all modes, m is allowed to be any real number between the leftmost and rightmost modes.
2. M, an upper bound for the value of p_i at a mode i. If possible, M should be set equal to this value.
3. s^2, an upper bound for the second moment about m. Note that if the variance σ^2 and mean μ are known, then we can take $s^2=\sigma^2+(m-\mu)^2$.

The universal algorithm derived below is based upon the following inequalities:

Theorem 1.1.

For all unimodal distributions on the integers,

$$p_i \le \min(M, \frac{3s^2}{|i-m|^3}) \quad (i \text{ integer}) .$$

In addition, for all integer i and all $x \in [i-\frac{1}{2}, i+\frac{1}{2}]$,

$$p_i \le g(x) = \min(M, \frac{3s^2}{(|x-m|-\frac{1}{2})_+^3}) .$$

Furthermore,

$$\int g = M + 3(3s^2)^{\frac{1}{3}} M^{\frac{2}{3}} .$$

Proof of Theorem 1.1.

Note that for $i > m$,

$$s^2 = \sum_{j=-\infty}^{\infty} (j-m)^2 p_j \ge \sum_{i \ge j \ge m} (j-m)^2 p_i$$

$$\ge p_i \int_m^i (u-m)^2 \, du = p_i \frac{(i-m)^3}{3} .$$

This establishes the first inequality. The bounding argument for g uses a standard tool for making the transition from discrete probabilities to densities: we consider a histogram-shaped density on the real line with height p_i on $[i-\frac{1}{2}, i+\frac{1}{2})$. This density is bounded by $g(x)$ on the interval in question. Note the adjustment by a translation term of $\frac{1}{2}$ when compared with the first discrete bound. This adjustment is needed to insure that g dominates p_i over the entire interval.

Finally, the area under g is easy to compute. Define $\rho = (3s^2)^{1/3} M^{2/3}$, and observe that the M term in g is the minimum term on $[m-\frac{1}{2}-\frac{\rho}{M}, m+\frac{1}{2}+\frac{\rho}{M}]$. The area under this part is thus $M+2\rho$. Integrating the two tails of g gives the value ρ. ■

To understand our algorithm, it helps to go back to the proof of Theorem 1.1. We have turned the problem into a continuous one by replacing the probability vector p_i with a histogram-shaped density of height p_i on $[i-\frac{1}{2}, i+\frac{1}{2})$. Since this histogram is dominated by the function g given in the algorithm, it is clear how to proceed. Note that if Y is a random variable with the said histogram-shaped density, then round(Y) is discrete with probability vector p_i.

Universal rejection algorithm for unimodal distributions

[SET-UP]

Compute $\rho \leftarrow (3s^2)^{\frac{1}{3}} M^{\frac{2}{3}}$.

[GENERATOR]

REPEAT

Generate U, W uniformly on [0,1] and V uniformly on [-1,1].

IF $U < \frac{\rho}{3\rho+M}$

THEN

$Y \leftarrow m + (\frac{1}{2} + \frac{\rho}{M\sqrt{|V|}})\,\text{sign}(V)$

$X \leftarrow \text{round}(Y)$

$T \leftarrow WM\,|V|^{\frac{3}{2}}$

ELSE

$Y \leftarrow m + (\frac{1}{2} + \frac{\rho}{M})V$

$X \leftarrow \text{round}(Y)$

$T \leftarrow WM$

UNTIL $T \le p_X$

RETURN X

In the universal algorithm, no care was taken to reuse unused portions of uniform random variates. This is done mainly to show where independent uniform random variates are precisely needed. The expected number of iterations in the algorithm is precisely $M+3\rho$. Thus, the algorithm is uniformly fast over a class Q of unimodal distributions with uniformly bounded $(1+s)M$ if p_i can be evaluated in time independent of i and the distribution.

Example 1.6.

For the binomial distribution with parameters n, p, it is known (see section X.4) that the mean μ is np, and that the variance σ^2 is $np(1-p)$. Also, for fixed p, $M \sim 1/(\sqrt{2\pi}\sigma)$, and for all n, p, $M \le 2/(\sqrt{2\pi}\sigma)$. A mode is at $m = \lfloor (n+1)p \rfloor$. Since $|\mu - m| \le \min(1, np)$ (exercise 1.4), we can take $s^2 = \sigma^2 + \min(1, np)$. We can verify that

$$\rho^3 \le \frac{6}{\pi}(1 + \frac{\min(1,np)}{np(1-p)}) ,$$

and this is uniformly bounded over $n \geq 1, 0 \leq p \leq \frac{1}{2}$. This implies that we can generate binomial random variates uniformly fast provided that the binomial probabilities can be evaluated in constant time. In section X.4, we will see that even this is not necessary, as long as the factorials are taken care of appropriately. We should note that when p remains fixed and $n \rightarrow \infty$, $\rho \sim (3/(2\pi))^{1/3}$. The expected number of iterations $\sim 3\rho$, which is about 2.4. Even though this is far from optimal, we should recall that besides the unimodality, virtually no properties of the binomial distribution were used in deriving the bounds. ■

There are important sub-families of distributions for which the algorithm given here is uniformly fast. Consider for example all distributions that are sums of iid integer-valued random variables with maximal probability p and finite variance σ^2. Then the sum of n such random variables has variance $n\sigma^2$. Also, $M \leq \frac{1}{\sqrt{n(1-p)}}$ (Rogozin (1961); see Petrov (1975, p. 56)). Thus, if the n-sum is unimodal, Theorem 1.1 is applicable. The rejection constant is

$$3\rho + M \leq 3\left(\frac{3\sigma^2}{1-p}\right)^{1/3} + 1$$

uniformly over all n. Thus, we can handle unimodal sums of iid random variables in expected time bounded by a constant not depending upon n. This assumes that the probabilities can all be evaluated in constant time, an assumption which except in the simplest cases is difficult to support. Examples of such families are the binomial family for fixed p, and the Poisson family.

Let us close this section by noting that the rejection constant can be reduced in special cases, such as for monotone distributions, or symmetric unimodal distributions.

1.5. Exercises.

1. The discrete distributions considered in the text are all lattice distributions. In these distributions, the intervals between the atoms of the distribution are all integral multiples of one quantity, typically 1. Non-lattice distributions can be considerably more difficult to handle. For example, there are discrete distributions whose atoms form a dense set on the positive real line. One such distribution is defined by

$$P\left(X = \frac{i}{j}\right) = \frac{(e-1)^2}{(e^{i+j}-1)^2},$$

where i and j are relatively prime positive integers (Johnson and Kotz, 1969, p. 31). The atoms in this case are the rationals. Discuss how you could efficiently generate a random variate with this distribution.

2. Using Lemma 1.2, show that if ϵ_n is a bound on the error committed when using Binet's series for $\log(\Gamma(x))$ with $n \geq 0$ terms, then

$$\sup_{x \geq 1} \sum_{n=0}^{\infty} n \, \epsilon_n < \infty$$

and

$$\lim_{x \to \infty} \sum_{n=0}^{\infty} n \, \epsilon_n = 0 .$$

3. Assume that all p_i's are at most equal to M, and that the variance is at most equal to σ^2. Derive useful bounds for a universal rejection algorithm which are similar to those given in Theorem 1.1. Show that there exists no dominating curve for this class which has area smaller than a constant times $\sigma\sqrt{M}$, and show that your dominating curve is therefore close to optimal. Give the details of the rejection algorithm. When applied to the binomial distribution with parameters n, p varying in such a way that $np \to \infty$, show that the expected number of iterations grows as a constant times $(np)^{\frac{1}{4}}$ and conclude that for this class the universal algorithm is not uniformly fast.

4. Prove that for the binomial distribution with parameters n, p, the mean μ and the mode $m = \lfloor (n+1)p \rfloor$ differ by at most $\min(1, np)$.

5. Replace the inequalities of Theorem 1.1 by new ones when instead of s^2, we are given the r-th absolute moment about the mean ($r \geq 1$), and value of the mean. The unimodality is still understood, and values for m, M are as in the Theorem.

6. How can the rejection constant ($\int g$) in Theorem 1.1 be reduced for monotone distributions and symmetric unimodal distributions ?

7. **The discrete Student's t distribution.** Ord (1968) introduced a discrete distribution with parameters $m \geq 0$ (m is integer) and $a \in [0,1], b \neq 0$:

$$p_i = K \prod_{j=0}^{m} \frac{1}{(j+a+i)^2 + b^2} \qquad (-\infty < i < \infty) .$$

Here K is a normalization constant. This distribution on the integers has the remarkable property that all the odd moments are zero, yet it is only symmetric for $a=0, a=\frac{1}{2}$ and $a=1$. Develop a uniformly fast generator for the case $m=0$.

8. **Arfwedson's distribution.** Arfwedson (1951) introduced the distribution defined by

$$p_i = \binom{k}{i} \sum_{j=0}^{i} (-1)^j \binom{i}{j} \left(\frac{i-j}{k}\right)^n \qquad (i \geq 0) ,$$

where k, n are positive integers. See also Johnson and Kotz (1969, p. 251). Compute the mean and variance, and derive an inequality consisting of a flat center piece and two decreasing polynomial or exponential tails having the property that the sum of the upper bound expressions over all i is uniformly bounded over k, n.

9. Knopp (1964, p. 553) has shown that

$$\sum_{n=1}^{\infty} \frac{1}{c\,(4n^2\pi^2+t^2)} = 1 ,$$

where $c = \frac{1}{2t}\left(\frac{1}{e^t-1} - \frac{1}{t} + \frac{1}{2}\right)$ and $t>0$ is a parameter. Give a uniformly fast generator for the family of discrete probability vectors defined by this sum.

2. THE GEOMETRIC DISTRIBUTION.

2.1. Definition and genesis.

X is **geometrically distributed** with parameter $p \in (0,1)$ when

$$P(X=i) = p\,(1-p)^{i-1} \quad (i \geq 1) .$$

The geometric distribution is important in statistics and probability because it is the distribution of the waiting time until success in a sequence of Bernoulli trials. In other words, if $U_1, U_2, \ldots$ are iid uniform $[0,1]$ random variables, and X is the index of the first U_i for which $U_i \leq p$, then X is geometric with parameter p. This property can of course be used to generate X, but to do so has some serious drawbacks because the algorithm is not uniformly fast over all values of p: just consider that the number of uniform random variates needed is itself geometric (p), and the expected number of uniform random variates required is

$$E(X) = \frac{1}{p} .$$

For $p \geq \frac{1}{3}$, the method is probably difficult to beat in any programming environment.

2.2. Generators.

The experimental method described in the previous section is summarized below:

Experimental method for geometric random variates

```
X←0
REPEAT
    Generate a uniform [0,1] random variate U.
    X←X+1
UNTIL U≤p
RETURN X
```

This method requires on the average $\frac{1}{p}$ uniform random variates and $\frac{1}{p}$ comparisons and additions. The number of uniform random variates can be reduced to 1 if we use the inversion method (sequential version):

Inversion by sequential search for geometric random variates

```
Generate a uniform [0,1] random variate U.
X←1
Sum←p
Prod←p
WHILE U>Sum DO
    Prod←Prod(1-p)
    Sum←Sum+Prod
    X←X+1
RETURN X
```

Unfortunately, the expected number of additions is now $\frac{2}{p}-2$, the expected number of comparisons remains $\frac{1}{p}$, and the expected number of products is $\frac{1}{p}-1$. Inversion in constant time is possible by truncation of an exponential random variate. What we use here is the property that

$$F(i) = P(X \le i) = 1-\sum_{j>i} p(1-p)^{j-1} = 1-(1-p)^i .$$

Thus, if U is uniform [0,1] and E is exponential, it is clear that

$$\left\lceil \frac{\log(U)}{\log(1-p)} \right\rceil$$

and

$$\left\lceil \frac{-E}{\log(1-p)} \right\rceil$$

are both geometric (p).

If many geometric random variates are needed for one fixed value of p, extra speed can be found by eliminating the need for an exponential random variate and for truncation. This can be done by splitting the distribution into two parts, a tail carrying small probability, and a main body. For the main body, a fast table method is used. For the tail, we can use the memoryless property of the geometric distribution: given that $X > i$, $X-i$ is again geometric (p) distributed. This property follows directly from the genesis of the distribution.

2.3. Exercises.

1. The quantity $\log(1-p)$ is needed in the bounded time inversion method. For small values of p, there is an accuracy problem because $1-p$ is computed before the logarithm. One can create one's own new function by basing an approximation on the series

   $$-(p+\frac{1}{2}p^2+\frac{1}{3}p^3+\cdots) .$$

   Show that the following more quickly convergent series can also be used:

   $$\frac{2}{r}(1+\frac{1}{3}r^{-2}+\frac{1}{5}r^{-4}+\cdots) ,$$

   where $r=1-\frac{2}{p}$.
2. Compute the variance of a geometric (p) random variable.

3. THE POISSON DISTRIBUTION.

3.1. Basic properties.

X is said to be **Poisson** (λ) distributed when

$$P(X=i) = \frac{\lambda^i}{i!} e^{-\lambda} \quad (i \geq 0) .$$

$\lambda>0$ is the parameter of the distribution. We do not have to convince the readers that the Poisson distribution plays a key role in probability and statistics. It is thus rather important that a simple uniformly fast Poisson generator be available in any nontrivial statistical software package. Before we tackle the development of such generators, we will briefly review some properties of the Poisson distribution. The Poisson probabilities are unimodal with one mode or two adjacent modes. There is always a mode at $\lfloor \lambda \rfloor$. The tail probabilities drop off faster than the tail of the exponential density, but not as fast as the tail of the normal density. In the design of algorithms, it is also useful to know that as $\lambda \to \infty$, the random variable $(X-\lambda)/\sqrt{\lambda}$ tends to a normal random variable.

Lemma 3.1.

When X is Poisson (λ), then X has characteristic function

$$\phi(t) = E(e^{itX}) = e^{\lambda(e^{it}-1)} .$$

It has moment generating function $E(e^{tX})=\exp(\lambda(e^t-1))$, and factorial moment generating function $E(t^X)=e^{\lambda(t-1)}$. Thus,

$$E(X) = Var(X) = \lambda .$$

Also, if X,Y are independent Poisson (λ) and Poisson (μ) random variables, then $X+Y$ is Poisson ($\lambda+\mu$).

Proof of Lemma 3.1.

Note that

$$E(e^{itX}) = \sum_{j=0}^{\infty} e^{-\lambda} \frac{(\lambda e^{it})^j}{j!} = e^{-\lambda+\lambda e^{it}} .$$

The statements about the moment generating function and factorial moment generating function follow directly from this. Also, if the factorial moment generating function is called k, then $k'(1)=E(X)=\lambda$ and $k''(1)=E(X(X-1))=\lambda^2$. From this we deduce that $Var(X)=\lambda$. The statement about the sum of two independent Poisson random variables follows directly from the form of the characteristic function. ■

3.2. Overview of generators.

The generators proposed over the years can be classified into several groups:

1. Generators based upon the connection with homogeneous Poisson processes (Knuth, 1969). These generators are very simple, but run in expected time proportional to λ.
2. Inversion methods. Inversion by sequential search started at 0 runs in expected time proportional to λ (see below). If the sequential search is started at the mode, then the expected time is $O(\sqrt{\lambda})$ (Fishman, 1976). Inversion can always be sped up by storing tables of constants (Atkinson, 1979).
3. Generators based upon recursive properties of the distribution (Ahrens and Dieter, 1974). One such generator is known to take expected time proportional to $\log(\lambda)$.
4. Rejection methods. Rejection methods seem to lead to the simplest uniformly fast algorithms (Atkinson, 1979; Ahrens and Dieter, 1980; Devroye, 1981; Schmeiser and Kachitvichyanukul, 1981).
5. The acceptance-complement method with the normal distribution as starting distribution. See Ahrens and Dieter (1982). This approach leads to efficient uniformly fast algorithms, but the computer programs are rather long.

We are undoubtedly omitting a large fraction of the literature on Poisson random variate generation. The early papers on the subject often proposed some approximate method for generating Poisson random variates which was typically based upon the closeness of the Poisson distribution to the normal distribution for large values of λ. It is pointless to give an exhaustive historical survey. The algorithms that really matter are those that are either simple or fast or both. The definition of "fast" may or may not include the set-up time. Also, since our comparisons cannot be based upon actual implementations, it is important to distinguish between computational models. In particular, the availability of the factorial in constant time is a crucial factor.

3.3. Simple generators.

The connection between the Poisson distribution and exponential interarrival times in a homogeneous point process is the following.

Lemma 3.2.

If $E_1, E_2, \ldots$ are iid exponential random variables, and X is the smallest integer such that

$$\sum_{i=1}^{X+1} E_i > \lambda ,$$

then X is Poisson (λ).

Proof of Lemma 3.2.

Let f_k be the gamma (k) density. Then,

$$P(X \leq k) = P(\sum_{i=1}^{k+1} E_i > \lambda) = \int_{\lambda}^{\infty} f_{k+1}(y)\, dy .$$

Thus, by partial integration,

$$P(X = k) = P(X \leq k) - P(X \leq k-1)$$

$$= \int_{\lambda}^{\infty} (f_{k+1}(y) - f_k(y))\, dy$$

$$= \int_{\lambda}^{\infty} (y-k) \frac{y^{k-1}}{k!} e^{-y}\, dy$$

$$= \frac{1}{k!} \int_{\lambda}^{\infty} d(-y^k e^{-y})$$

$$= e^{-\lambda} \frac{\lambda^k}{k!} . \blacksquare$$

The algorithm based upon this property is:

Poisson generator based upon exponential inter-arrival times

```
X←0
Sum←0
WHILE True DO
    Generate an exponential random variate E.
    Sum←Sum+E
    IF Sum<λ
        THEN X←X+1
        ELSE RETURN X
```

Using the fact that a uniform random variable is distributed as e^{-E}, it is easy to see that Lemma 3.2 is equivalent to Lemma 3.3, and that the algorithm shown above is equivalent to the algorithm following Lemma 3.3:

Lemma 3.3.

Let $U_1, U_2, \ldots$ be iid uniform [0,1] random variables, and let X be the smallest integer such that

$$\prod_{i=1}^{X+1} U_i < e^{-\lambda} .$$

Then X is Poisson (λ).

Poisson generator based upon the multiplication of uniform random variates

```
X←0
Prod←1
WHILE True DO
    Generate a uniform [0,1] random variate U.
    Prod←Prod U
    IF Prod>e^{-λ} (the constant should be computed only once)
        THEN X←X+1
        ELSE RETURN X
```

3.4. Rejection methods.

To see how easy it is to improve over the algorithms of the previous section, it helps to get an idea of how the probabilities vary with λ. First of all, the peak at $\lfloor\lambda\rfloor$ varies as $1/\sqrt{\lambda}$:

Lemma 3.4.

The value of $P(X=\lfloor\lambda\rfloor)$ does not exceed

$$\frac{1}{\sqrt{2\pi\lfloor\lambda\rfloor}} ,$$

and $\sim 1/\sqrt{2\pi\lambda}$ as $\lambda\to\infty$.

Proof of Lemma 3.4.

We apply the inequality $i! \geq i^i e^{-i}\sqrt{2\pi i}$, valid for all integer $i \geq 1$. Thus,

$$e^{-\lambda}\frac{\lambda^{\lfloor\lambda\rfloor}}{\lambda!} \leq e^{-(\lambda-\lfloor\lambda\rfloor)}\left(\frac{\lambda}{\lfloor\lambda\rfloor}\right)^{\lfloor\lambda\rfloor}\frac{1}{\sqrt{2\pi\lfloor\lambda\rfloor}}$$

$$\leq \frac{1}{\sqrt{2\pi\lfloor\lambda\rfloor}} .$$

Furthermore, by Stirling's approximation, it is easy to establish the asymptotic result as well. ■

We also have the following inequality by monotonicity:

Lemma 3.5.

$$P(X=\lfloor\lambda\rfloor\pm i) \leq \frac{2(\sqrt{\lambda}+1)}{i(i+1)} \qquad (i>0).$$

Proof of Lemma 3.5.

We will argue for the positive side only. Writing p_i for $P(X=i)$, we have by unimodality,

$$\sqrt{\lambda}+1 \geq E(|X-\lambda|)+1$$

$$\geq E(|X-\lfloor\lambda\rfloor|) \geq \sum_{j\geq\lfloor\lambda\rfloor} |j-\lfloor\lambda\rfloor| p_j$$

$$\geq p_{i+\lfloor\lambda\rfloor}\sum_{j=0}^{i} j$$

The expected number of iterations is the same for both algorithms. However, an addition and an exponential random variate are replaced by a multiplication and a uniform random variate. This replacement usually works in favor of the multiplicative method. The expected complexity of both algorithms grows linearly with λ.

Another simple algorithm requiring only one uniform random variate is the inversion algorithm with sequential search. In view of the recurrence relation

$$\frac{P(X=i+1)}{P(X=i)}=\frac{\lambda}{i+1} \quad (i \geq 0) ,$$

this gives

Poisson generator based upon the inversion by sequential search

$X \leftarrow 0$

Sum$\leftarrow e^{-\lambda}$,Prod$\leftarrow e^{-\lambda}$

Generate a uniform [0,1] random variate U.

WHILE $U >$Sum DO

  $X \leftarrow X+1$

  Prod$\leftarrow \frac{\lambda}{X}$Prod

  Sum$\leftarrow$Sum+Prod

RETURN X

This algorithm too requires expected time proportional to λ as $\lambda \rightarrow \infty$. For large λ, round-off errors proliferate, which provides us with another reason for avoiding large values of λ. Speed-ups of the inversion algorithm are possible if sequential search is started near the mode. For example, we could compare U first with $b=P(X \leq \lfloor \lambda \rfloor)$, and then search sequentially upwards or downwards. If b is available in time $O(1)$, then the algorithm takes expected time $O(\sqrt{\lambda})$ because $E(\,|\,X-\lfloor \lambda \rfloor\,|\,)=O(\sqrt{\lambda})$. See Fishman (1976). If b has to be computed first, this method is hardly competitive. Atkinson (1979) describes various ways in which the inversion can be helped by the judicious use of tables. For small values of λ , there is no problem. He then custom builds fast table-based generators for all λ's that are powers of 2, starting with 2 and ending with 128. For a given value of λ, a sum of independent Poisson random variates is needed with parameters that are either powers of 2 or very small. The speed-up comes at a tremendous cost in terms of space and programming effort.

$$= \frac{i(i+1)}{2} p_{i+\lfloor \lambda \rfloor} . \blacksquare$$

If we take the minimum of the constant upper bound of Lemma 3.4 and the quadratically decreasing upper bound of Lemma 3.5, it is not difficult to see that the cross-over point is near $\lambda \pm c\sqrt{\lambda}$ where $c = (8\pi)^{1/4}$. The area under the bounding sequence of numbers is $O(1)$ as $\lambda \to \infty$. It is uniformly bounded over all values $\lambda \geq 1$. We do not imply that one should design a generator based upon this dominating curve. The point is that it is very easy to construct good bounding sequences. In fact, we already knew from Theorem 1.1 that the universal rejection algorithm of section 1.4 is uniformly fast. The dominating curves of Theorem 1.1 and Lemmas 3.4 and 3.5 are similar, both having a flat center part. Atkinson (1979) proposes a logistic majorizing curve, and Ahrens and Dieter (1980) propose a double exponential majorizing curve. Schmeiser and Kachitvichyanukul (1981) have a rejection method with a triangular hat and two exponential tails. We do not describe these methods here. Rather, we will describe an algorithm of Devroye (1981) which is based upon a normal-exponential dominating curve. This has the advantage that the rejection constant tends to 1 as $\lambda \to \infty$. In addition, we will illustrate how the factorial can be avoided most of the time by the judicious use of squeeze steps. Even if factorials are computed in linear time, the overall expected time per random variate remains uniformly bounded over λ. For large values of λ, we will return a truncated normal random variate with large probability.

Some inequalities are needed for the development of tight inequalities for the Poisson probabilities. These are collected in the next Lemma:

Lemma 3.6.

Assume that $u \geq 0$ and all the arguments of the logarithms are positive in the list of inequalities shown below. We have:

(i) $\log(1+u) \leq u$

(ii) $\log(1+u) \leq u - \frac{1}{2}u^2 + \frac{1}{3}u^3$

(iii) $\log(1+u) \geq u - \frac{1}{2}u^2$

(iv) $\log(1+u) \geq \frac{2u}{2+u}$

(v) $\log(1-u) \leq -\sum_{i=1}^{k} \frac{1}{i}u^i \quad (k \geq 1)$

(vi) $\log(1-u) \geq -\sum_{i=1}^{k-1} \frac{1}{i}u^i - \frac{u^k}{k(1-u)} \quad (k \geq 2)$

. Most of these inequalities are well-known. The other ones can be obtained without difficulty from Taylor's theorem (Whittaker and Watson, 1927, is a good source of information). We assume that $\lambda \geq 1$. Since we will use rejection algorithms, it can't harm to normalize the Poisson probabilities. Instead of the probabilities p_i, we will use the normalized log probabilities

$$q_j = \log(p_{\mu+j}) + \log(\mu!) - \mu\log(\lambda) + \lambda$$

where $\mu = \lfloor \lambda \rfloor$. This can conveniently be rewritten as follows:

$$q_j = j\log(\frac{\lambda}{\mu}) + j\log(\mu) - \log(\frac{(\mu+j)!}{\mu!})$$

$$= j\log(\frac{\lambda}{\mu}) + \begin{cases} -\log(\prod_{i=1}^{j}(1+\frac{i}{\mu})) & (j>0) \\ 0 & (j=0) \\ -\log(\prod_{i=0}^{-j-1}(1-\frac{i}{\mu})) & (j<0) \end{cases} .$$

Lemma 3.7.

Let us use the notation j_+ for $\max(j,0)$. Then, for all integer $j \geq -\mu$,

$$q_j \leq \frac{j_+}{\mu} - \frac{j(j+1)}{2\mu + j_+} .$$

Proof of Lemma 3.7.

Use (iv) and (v) of Lemma 3.6, together with the identity

$$\sum_{i=1}^{j} i = \frac{j(j+1)}{2} . \blacksquare$$

The inequality of Lemma 3.7 can be used as the starting point for the development of tight dominating curves. The last term on the right hand side in the upper bound is not in a familiar form. On the one hand, it suggests a normal bounding curve when j is small compared to μ. On the other hand, for large values of $|j|$, an exponential bounding curve seems more appropriate. Recall that the Poisson probabilities cannot be tucked under a normal curve because they drop off as $e^{-cj\log(j)}$ for some c as $j \to \infty$. In Lemma 3.8 we tuck the Poisson probabilities under a normal main body and an exponential right tail.

Lemma 3.8.

Assume that $\mu \geq 6$ and that δ is an integer satisfying

$$6 \leq \delta \leq \mu .$$

Then

$$q_j \leq -\frac{j(j+1)}{2\mu} \leq -\frac{j^2}{2\mu} \quad (j \leq 0)$$

$$q_0 \leq 0$$

$$q_1 \leq \frac{1}{\mu(2\mu+1)} \leq \frac{1}{78}$$

$$q_j \leq -\frac{(j-1)^2}{2\mu+\delta} + \frac{1}{2\mu+\delta} \quad (0 \leq j \leq \delta)$$

$$q_j \leq -\frac{\delta}{2\mu+\delta}\left(\frac{j}{2}+1\right) \quad (j \geq \delta) .$$

Proof of Lemma 3.8.

The first three inequalities follow without work from Lemma 3.7. For the fourth inequality, we observe that for $2 \le j \le \delta$,

$$q_j \le \frac{j+\frac{j}{2}}{\mu+\frac{j}{2}} - \frac{j(j+1)}{2(\mu+\frac{j}{2})} \quad (\text{since } j \le \delta \le \mu)$$

$$= \frac{2j-j^2}{2\mu+j}$$

$$\le \frac{2j-j^2}{2\mu+\delta} \quad (\text{since } 2 \le j \le \delta) .$$

The fourth inequality is also valid for $j=0$. For $j=1$, a quick check shows that $1/\mu(2\mu+1) \le 1/(2\mu+\delta)$ because $\delta \le \mu$. This leaves us with the fifth and last inequality. We note that $\delta \ge 6 \ge \frac{4\mu}{\mu-2}$. Thus,

$$q_j \le \frac{j}{\mu} - \frac{\delta}{2\mu+\delta}(j+1)$$

$$= -\frac{\delta}{2\mu+\delta} + j\left(\frac{1}{\mu} - \frac{\delta}{2\mu+\delta}\right)$$

$$\le -\frac{\delta}{2\mu+\delta}\left(1+\frac{j}{2}\right) . \blacksquare$$

Based on these inequalities, we can now give a first Poisson algorithm:

Rejection method for Poisson random variates

[SET-UP]

$\mu \leftarrow \lfloor \lambda \rfloor$

Choose δ integer such that $6 \le \delta \le \mu$.

$c_1 \leftarrow \sqrt{\pi\mu/2}$

$c_2 \leftarrow c_1 + \sqrt{\pi(\mu+\delta/2)/2}\, e^{\frac{1}{2\mu+\delta}}$

$c_3 \leftarrow c_2 + 1$

$c_4 \leftarrow c_3 + e^{\frac{1}{78}}$

$c \leftarrow c_4 + \frac{2}{\delta}(2\mu+\delta) e^{-\frac{\delta}{2\mu+\delta}(1+\frac{\delta}{2})}$

[NOTE]

The function q_j^* is defined as $q_j - j\log(\frac{\lambda}{\mu}) = j\log(\mu) - \log((\mu+j)!/\mu!)$.

[GENERATOR]

REPEAT

Generate a uniform $[0,c]$ random variate U and an exponential random variate E.
Accept $\leftarrow$ False.

CASE

$U \le c_1$:

Generate a normal random variate N.

$Y \leftarrow -|N|\sqrt{\mu}$

$X \leftarrow \lfloor Y \rfloor$

$W \leftarrow -\frac{N^2}{2} - E - X\log(\frac{\lambda}{\mu})$

IF $X \ge -\mu$ THEN $W \leftarrow \infty$

$c_1 < U \le c_2$:

Generate a normal random variate N.

$Y \leftarrow 1 + |N| \sqrt{\mu + \frac{\delta}{2}}$

$X \leftarrow \lceil Y \rceil$

$W \leftarrow \frac{-Y^2+2Y}{2\mu+\delta} - E - X\log(\frac{\lambda}{\mu})$

IF $X \le \delta$ THEN $W \leftarrow \infty$

$c_2 < U \le c_3$:

$X \leftarrow 0$

$W \leftarrow -E$

$c_3 < U \le c_4$:

$X \leftarrow 1$

$W \leftarrow -E - \log(\frac{\lambda}{\mu})$

$c_4 < U$:

Generate an exponential random variate V.

```
        Y ← δ + V (2/δ)(2μ+δ)
        X ← ⌈Y⌉
        W ← -(δ/(2μ+δ))(1+Y/2) - E - X log(λ/μ)
    Accept ← [W ≤ q*_X]
UNTIL Accept
RETURN X+μ
```

Observe the careful use of the floor and ceiling functions in the algorithm to insure that the continuous dominating curves exceed the Poisson staircase function at every point of the real line, not just the integers ! The monotonicity of the dominating curves is exploited of course. The function

$$q_x = x\log(\lambda) - \log\left(\frac{(\mu+x)!}{\mu!}\right)$$

is evaluated in every iteration at some point x. If the logarithm of the factorial is available at unit cost, then the algorithm can run in uniformly bounded time provided that δ is carefully picked. Thus, the first issue to be dealt with is that of the relationship between the expected number of iterations and δ.

Lemma 3.9.

If δ depends upon λ in such a way that

$$\delta = o(\mu) \ , \ \frac{\delta}{\sqrt{\mu}} \to \infty \ ,$$

then the expected number of iterations $E(N)$ tends to one as $\lambda \to \infty$. In particular, the expected number of iterations remains uniformly bounded over $\lambda \geq 6$.

Furthermore,

$$\inf_{\delta} E(N) = 1+(1+o(1))\sqrt{\frac{\log(\mu)}{32\mu}} \text{ as } \lambda\to\infty$$

where the infimum is reached if we choose

$$\delta \sim \sqrt{2\mu\log\left(\frac{128\mu}{\pi}\right)} \ .$$

Proof of Lemma 3.9.

In a preliminary computation, we have to evaluate

$$\sum_{j \geq -\mu} e^{q_j}$$

since this is the total weight of the normalized Poisson probabilities. It is easy to see that this gives

$$\sum_{j=0}^{\infty} p_j \, e^{\lambda} \mu! \lambda^{-\mu}$$

$$\sim e^{\lambda} \left(\frac{\mu}{e\,\lambda}\right)^{\mu} \sqrt{2\pi\mu}$$

$$\sim \sqrt{2\pi\mu}$$

where we used the fact that $\log(\lambda/\mu) = \log(1+(\lambda-\mu)/\mu) = (\lambda-\mu)/\mu + O(\mu^{-2})$. Thus, the expected number of iterations is the total area under the dominating curve (with the atoms at 0 and 1 having areas one and $e^{\frac{1}{78}}$ respectively) divided by $(1+o(1))\sqrt{2\pi\mu}$. The area under the dominating curve is, taking the five contributors from left to right,

$$\sqrt{\pi\mu/2} + 1 + e^{\frac{1}{78}} + \sqrt{\pi(\mu+\frac{\delta}{2})/2}\, e^{\frac{1}{2\mu+\delta}} + \frac{2(2\mu+\delta)}{\delta} e^{-\frac{\delta}{2\mu+\delta}(\frac{\delta}{2}+1)} .$$

If δ is not $o(\mu)$, this can not $\sim \sqrt{2\pi\mu}$. If $\delta \leq c\sqrt{\mu}$ for some constant c, then the last term is at least $\sim \frac{4}{c} e^{-c^2/4} \sqrt{\mu}$, while it should really be $o(\sqrt{\mu})$. Thus, the conditions imposed on δ are necessary for $E(N) \rightarrow 1$. That they are also sufficient can be seen as follows. The fifth term in the area under the dominating curves is $o(\sqrt{\mu})$, and so are the constant second and third terms. The fourth term $\sim \sqrt{\pi\mu/2}$, which establishes the result.

To minimize $E(N)-1$ in an asymptotically optimal fashion, we have to consider some sort of expansion of the area in terms of decreasing asymptotic importance. Using the Taylor series expansion for $\sqrt{1+u}$ for u near 0, we can write the first four terms as

$$\sqrt{\pi\mu/2} \left(1 + O(\mu^{-\frac{1}{2}}) + 1 + \frac{\delta}{4\mu} + O\left(\left(\frac{\delta}{\mu}\right)^2\right) \right) .$$

The main term in excess of $\sqrt{2\pi\mu}$ is

$$\sqrt{\pi\mu/2}\,\frac{\delta}{4\mu} .$$

We can also verify easily that the contribution from the exponential tail is

$$\frac{4\mu}{\delta}(1+o(1)) e^{-\frac{\delta^2}{2(2\mu+\delta)}} .$$

To obtain a first (but as we will see, good) guess for δ, we will minimize

$$\sqrt{\pi\mu/2}\frac{\delta}{4\mu}+\frac{4\mu}{\delta}e^{-\frac{\delta^2}{2(2\mu+\delta)}} .$$

This is equivalent to solving

$$(2+\frac{4\mu}{\delta^2})e^{-\frac{\delta^2}{4\mu}} = \sqrt{\frac{\pi}{32\mu}} .$$

If we ignore the $o(1)$ term $\frac{4\mu}{\delta^2}$, we can solve this explicitly and obtain

$$\delta = \sqrt{2\mu \log(\frac{128\mu}{\pi})} .$$

A plugback of this value in the original expression for the area under the dominating curve shows that it increases as

$$\sqrt{2\pi\mu}+(1+o(1))\frac{\sqrt{\pi}}{4}\sqrt{\log(\mu)} .$$

The constant terms are absorbed in $o(1)$; the exponential tail contribution is $O(1/\sqrt{\log(\mu)})$. If we replace δ by $\delta(1+\epsilon)$ where ϵ is allowed to vary with μ but is bounded from below by $c>0$, then the area is asymptotically larger because the $\sqrt{\log(\mu)}$ term should be multiplied by at least $1+c$. If we replace δ by $\delta(1-\epsilon)$, then the contribution from the exponential tail is at least $\Omega(\mu^{c/2}/\sqrt{\log(\mu)})$. This concludes the proof of the Lemma. ■

We have to insure that δ falls within the limits imposed on it when the dominating curves were derived. Thus, the following choice should prove failsafe in practice:

$$\delta = \max(6,\min(\mu,\sqrt{2\mu \log(\frac{128\mu}{\pi})})) .$$

We have now in detail dealt with the optimal design for our Poisson generator. If the log-factorial is available at unit cost, the rejection algorithm is uniformly fast, and asymptotically, the rejection constant tends to one. δ was picked to insure that the convergence to one takes place at the best possible rate. For the optimal δ, the algorithm basically returns a truncated normal random variate most of the time. The exponential tail becomes asymptotically negligible.

We may ask what would happen to our algorithm if we were to compute all products of successive integers explicitly ? Disregarding the horrible accuracy problems inherent in all repeated multiplications, we would also face a breakdown in our complexity. The computation of

$$q_X = X\log(\frac{\lambda}{\mu})+X\log(\mu)-\log(\frac{(X+\mu)!}{\mu}!)$$

can be done in time proportional to $1+|X|$. Now, X is with high probability normal with mean 0 and variance approximately equal to $\sqrt{\mu}$. Since q is computed only once with probability tending to one, it is clear that the expected time complexity now grows as $\sqrt{\mu}$. If we had perfect squeeze curves, i.e. squeeze curves in which the top and bottom bounds are equal, then we would get our uniform speed back. The same is true for very tight but imperfect squeeze curves. A class of such squeeze curves is presented below. Note that we are no longer concerned with the dominating curves. The squeeze curves given below are also not derived from the inequalities for Stirling's series or Binet's series for the log gamma function (see section 1). We could have used those, but it is instructive to show yet another method of deriving good bounds. See however exercise 3.9 for the application of Stirling's series in squeeze curves for Poisson probabilities.

Lemma 3.10.

Define

$$t_j = q_j - j\log(\frac{\lambda}{\mu}) + \frac{j(j+1)}{2\mu} .$$

Then for integer $j \geq 0$,

$$t_j \begin{cases} \geq \max\left(0, \frac{j(j+1)(2j+1)}{12\mu^2} - \frac{j^2(j+1)^2}{12\mu^3}\right) \\ \leq \frac{j(j+1)(2j+1)}{12\mu^2} \end{cases} .$$

Furthermore, for integer $-\mu \leq j \leq 0$, the converse is almost true:

$$t_j \begin{cases} \geq \frac{j(j+1)(2j+1)}{12\mu^2} - \frac{j^2(j+1)^2}{12\mu^2(\mu+j+1)} \\ \leq \min\left(0, \frac{j(j+1)(2j+1)}{12\mu^2}\right) \end{cases} .$$

Proof of Lemma 3.10.

The proof is based upon Lemma 3.6, the identities

$$\sum_{i=1}^{k} i = \frac{k(k+1)}{2} , \quad \sum_{i=1}^{k} i^2 = \frac{k(k+1)(2k+1)}{6} , \quad \sum_{i=1}^{k} i^3 = \frac{k^2(k+1)^2}{4} ,$$

and the fact that q_j can be rewritten as follows:

$$q_j - j\log(\frac{\lambda}{\mu}) = \begin{cases} -\log(\prod_{i=1}^{j}(1+\frac{i}{\mu})) & (j>0) \\ 0 & (j=0) \\ \log(\prod_{i=0}^{-j-1}(1-\frac{i}{\mu})) & (j<0) \end{cases} . \quad \blacksquare$$

The algorithm requires of course little modification. Only the line

Accept $\leftarrow [W \le q_X^*]$

needs replacing. The replacement looks like this:

$T \leftarrow \frac{X(X+1)}{2\mu}$
Accept $\leftarrow [W \le -T] \cap [X \ge 0]$
IF NOT Accept THEN
  $Q \leftarrow T(\frac{2X+1}{6\mu}-1)$
  $P \leftarrow Q - \frac{T^2}{3(\mu+(X+1)_-)}$
  Accept $\leftarrow [W \le Q]$
  IF NOT Accept AND $[W \le P]$ THEN Accept $\leftarrow [W \le q_X^*]$

It is interesting to go through the expected complexity proof in this one example because we are no longer counting iterations but multiplications.

Lemma 3.11.

The expected time taken by the modified Poisson generator is uniformly bounded over $\lambda \ge 6$ when δ is chosen as in Lemma 3.10, even when factorials are explicitly evaluated as products.

Proof of Lemma 3.11.

It suffices to establish the uniform boundedness of

$$E(|X| I_{[Q<W<P]})$$

where we use the notation of the algorithm. Note that this statement implicitly uses Wald's equation, and the fact that the expected number of iterations is uniformly bounded. The expression involving $|X|$ is arrived at by looking at the time needed to evaluate q_X^*. The expected value will be split into five parts according to the five components in the distribution of X. The atomic parts

$X=0, X=1$ are easy to take care of. The contribution from the normal portions can be bounded from above by a constant times

$$E(|X|(P-Q)) \leq E(|X| \frac{X^2(X+1)^2}{12\mu^2(\mu+(X+1)_-)}) .$$

Here we have used the fact that W consists of a sum of some random variable and an exponential random variable. When $X \geq 0$, the last upper bound is in turn not greater than a constant times $E(|X|^5)/\mu^3 = O(\mu^{-1/2})$. The case $X<0$ is taken care of similarly, provided that we first split off the case $X<-\frac{\mu}{2}$. The split-off part is bounded from above by

$$O(\mu^3)P(X<-\frac{\mu}{2}) \leq O(\mu^3)\frac{E(X^2)}{\mu^2} = O(1) .$$

For the exponential tail part, we need a uniform bound for

$$E(|X|^5\mu^{-3})(\log(\mu))^{-\frac{1}{2}}$$

where we have used a fact shown in the proof of Lemma 3.10, i.e. the probability that X is exponential decreases as a constant times $\log^{-1/2}(\mu)$. Verify next that given that X is from our exponential tail, $E(|X|^5)=O(\delta^5)$. Combining all of this shows that our expression in question is

$$O(\frac{\log^2(\mu)}{\sqrt{\mu}}) .$$

This concludes the proof of Lemma 3.11. ■

The computations of the previous Lemma reveal other interesting facets of the algorithm. For example, the expected time contribution of the evaluations of factorials is $O(\frac{\log^2(\mu)}{\sqrt{\mu}})$. In other words, it is asymptotically negligible. Even so, the main contribution to this $o(1)$ expected time comes from the exponential tail. This suggests that it is possible to obtain a new value for δ which would minimize the expected time spent on the evaluation of factorials, and that this value will differ from that obtained by minimizing the expected number of iterations.

3.5. Exercises.

1. Atkinson (1979) has developed a Poisson (λ) generator based upon rejection from the logistic density

$$f(x) = \frac{1}{b} e^{-\frac{x-a}{b}} \left(1+e^{-\frac{x-a}{b}} \right)^{-2} ,$$

where $a=\lambda$ and $b=\sqrt{3\lambda}/\pi$. A random variate with this density can be generated as $X \leftarrow a+b \log(\frac{1-U}{U})$ where U is uniform [0,1].

   A. Find the distribution of $\left\lfloor X+\frac{1}{2} \right\rfloor$.

   B. Prove that X has the same mean and variance as the Poisson distribution.

   C. Determine a rejection constant c for use with the distribution of part A.

   D. Prove that c is uniformly bounded over all values of λ.

2. **A recursive generator.** Let n be an integer somewhat smaller than λ, and let G be a gamma (n) random variable. Show that the random variable X defined below is Poisson (λ): if $G>\lambda$, X is binomial $(n-1,\lambda/G)$; if $G \leq \lambda$, then X is n plus a Poisson ($\lambda-G$) random variable. Then, taking $n = \lfloor 0.875\lambda \rfloor$, use this recursive property to develop a recursive Poisson generator. Note that one can leave the recursive loop either when at one point $G>\lambda$ or when λ falls below a fixed threshold (such as 10 or 15). By taking n a fixed fraction of λ, the value of λ falls at a geometric rate. Show that in view of this, the expected time complexity is $O(1+\log(\lambda))$ if a constant expected time gamma generator is used (Ahrens and Dieter, 1974).

3. Prove all the inequalities of Lemma 3.6.

4. Prove that for any λ and any $c>0$, $\lim_{j \to \infty} p_j / e^{-cj^2} = \infty$. Thus, the Poisson curve cannot be tucked under any normal curve.

5. **Poisson variates in batches.** Let $X_1, \ldots, X_n$ be a multinomial $(Y, p_1, \ldots, p_n)$ random vector (i.e., the probability of attaining the value $i_1, \ldots, i_n$ is 0 when $\sum i_j$ is not Y and is

$$\frac{Y!}{i_1! \cdots i_n!} p_1^{i_1} \cdots p_n^{i_n}$$

otherwise. Show that if Y is Poisson (λ), then $X_1, \ldots, X_n$ are independent Poisson random variables with parameters $\lambda p_1, \ldots, \lambda p_n$ respectively. (Moran, 1951; Patil and Seshadri, 1964; Bolshev, 1965; Tadikamalla, 1979).

6. Prove that as $\lambda \to \infty$, the distribution of $(X-\lambda)/\sqrt{\lambda}$ tends to the normal distribution by proving that the characteristic function tends to the characteristic function $e^{-t^2/2}$ of the normal distribution.

```
WHILE True DO
    Generate a Poisson (1) random variate X, and a uniform [0,1] random
    variate U.
    IF X ≤ n THEN
        k ← 1, j ← 0, s ← 1
        WHILE j ≤ n-X AND U ≤ s DO
            j ← j+1, k ← -jk, s ← s + 1/k
        IF j ≤ n-X AND U < s
            THEN RETURN X
            ELSE j ← j+1, k ← -jk, s ← s + 1/k
```

12. **The Borel-Tanner distribution.** A distribution important in queuing theory, with parameters $n \geq 1$ (n integer) and $\alpha \in (0,1)$ was discovered by Borel and Tanner (Tanner, 1951). The probabilities p_i are defined by

$$p_i = \frac{n}{(i-n)!} i^{i-n-1} \alpha^{i-n} e^{-\alpha i} \qquad (i \geq n) .$$

Show that the mean is $\frac{n}{1-\alpha}$ and that the variance is $\frac{n\alpha}{(1-\alpha)^3}$. The distribution has a very long positive tail. Develop a uniformly fast generator.

4. THE BINOMIAL DISTRIBUTION.

4.1. Properties.

X is **binomially distributed** with parameters $n \geq 1$ and $p \in [0,1]$ if

$$P(X=i) = \binom{n}{i} p^i (1-p)^{n-i} \qquad (0 \leq i \leq n) .$$

We will say that X is binomial (n,p).

7. Show that for the rejection method developed in the text, the expected time complexity is $O(\sqrt{\lambda})$ and $\Omega(\sqrt{\lambda})$ as $\lambda \to \infty$ when no squeeze steps are used and the factorial has to be evaluated explicitly.

8. Give a detailed rejection algorithm based upon the constant upper bound of Lemma 3.4 and the quadratically decreasing tails of Lemma 3.5.

9. Assume that factorials are avoided by using the zero-term and one-term Stirling approximations (Lemma 1.1) as lower and upper bounds in squeeze steps (the difference between the zero-term and one-term approximations of $\log(\Gamma(n))$ is the term $1/(12n)$). Show that this suffices for the following rejection algorithms to be uniformly fast:

   A. The universal algorithm of section 1.

   B. The algorithm based upon Lemmas 3.4 and 3.5 (and developed in Exercise 8).

   C. The normal-exponential rejection algorithm developed in the text.

10. Repeat exercise 9, but assume now that factorials are avoided altogether by evaluating an increasing number of terms in Binet's convergent series for the log gamma function (Lemma 1.2) until an acceptance or rejection decision can be made. Read first the text following Lemma 1.2.

11. **The matching distribution.** Suppose that n cars are parked in front of Hanna's rubber skin suit shop, and that each of Hanna's satisfied customers leaves in a randomly picked car. The number N of persons who leave in their own car has the matching distribution with parameter n:

$$P(N=i) = \frac{1}{i!} \sum_{j=0}^{n-i} \frac{(-1)^j}{j!} \quad (0 \leq i \leq n) .$$

   A. Show this by invoking the inclusion exclusion principle.

   B. Show that $\lim_{n \to \infty} P(N=i) = \frac{1}{e\ i!}$, i.e. that the Poisson (1) distribution is the limit (Barton, 1958).

   C. Show that $P(N=i) \leq \frac{1}{i!}$, i.e. rejection from the Poisson (1) distribution can be used with rejection constant e not depending upon n.

   D. Show that the algorithm given below is valid, and that its expected complexity is uniformly bounded in n.

Lemma 4.1. (Genesis.)

Let X be the number of successes in a sequence of n Bernoulli trials with success probability p, i.e.

$$X = \sum_{i=1}^{n} I_{[U_i < p]} ,$$

where $U_1, \ldots, U_n$ are iid uniform [0,1] random variables. Then X is binomial (n, p).

Lemma 4.2.

The binomial distribution with parameters n, p has generating function $(1-p+ps)^n$. The mean is np, and the variance is $np(1-p)$.

Proof of Lemma 4.2.

The factorial moment generating function of X (or simply generating function) is

$$k(s) = E(s^X) = \prod_{i=1}^{n} E(s^{I_{[U_i < p]}}) ,$$

where we used the Lemma 4.1 and its notation. Each factor in the product is obviously equal to $1-p+ps$. This concludes the proof of the first statement. Next, $E(X) = k'(1) = np$, and $E(X(X-1)) = k''(1) = n(n-1)p^2$. Hence, $Var(X) = E(X^2)-E^2(X) = E(X(X-1))+E(X)-E^2(X) = np(1-p)$. ■

From Lemma 4.1, we can conclude without further work:

Lemma 4.3.

If $X_1, \ldots, X_k$ are independent binomial $(n_1, p), \ldots, (n_k, p)$ random variables, then $\sum_{i=1}^{k} X_i$ is binomial $(\sum_{i=1}^{k} n_i, p)$.

Lemma 4.4.(First waiting time property.)

Let $G_1, G_2, \ldots$ be iid geometric (p) random variables, and let X be the smallest integer such that

$$\sum_{i=1}^{X+1} G_i > n \ .$$

Then X is binomial (n, p).

Proof of Lemma 4.4.

G_1 is the number of Bernoulli trials up to and including the first success. Thus, by the independence of the G_i's, $G_1 + \cdots + G_{X+1}$ is the number of Bernoulli trials up to and including the $X+1$-st success. This number is greater than n if and only if among the first n Bernoulli trials there are at most X successes. Thus,

$$P(X \le k) = P\left(\sum_{i=1}^{k+1} G_i > n\right) = \sum_{j=0}^{k} \binom{n}{j} p^j (1-p)^{n-j} \quad \text{(integer } k \text{)} \ . \blacksquare$$

Lemma 4.5. (Second waiting time property.)

Let $E_1, E_2, \ldots$ be iid exponential random variables, and let X be the smallest integer such that

$$\sum_{i=1}^{X+1} \frac{E_i}{n-i+1} > -\log(1-p) \ .$$

Then X is binomial (n, p).

Proof of Lemma 4.5.

Let $E_{(1)} < E_{(2)} < \cdots < E_{(n)}$ be the order statistics of an exponential distribution. Clearly, the number of $E_{(i)}$'s smaller than $-\log(1-p)$ is binomially distributed with parameters n and $P(E_1 < -\log(1-p)) = 1 - e^{\log(1-p)} = p$. Thus, if X is the smallest integer such that $E_{(X+1)} \ge -\log(1-p)$, then X is binomial (n, p). Lemma 4.5 now follows from the fact (section V.2) that $(E_{(1)}, \ldots, E_{(n)})$ is distributed as

$$\left(\frac{E_1}{n}, \frac{E_1}{n} + \frac{E_2}{n-1}, \ldots, \frac{E_1}{n} + \frac{E_2}{n-1} + \cdots + \frac{E_n}{1}\right) . \blacksquare$$

4.2. Overview of generators.

The binomial generators can be partitioned into a number of classes:

A. The simple generators. These generators are based upon the direct application of one of the lemmas of the previous section. Typically, the expected complexity grows as n or as np, the computer programs are very short, and no additional workspace is required.

B. Uniformly fast generators based upon the rejection method (Fishman (1979), Ahrens and Dieter (1980), Kachitvichyanukul (1982), Devroye and Naderisamani (1980)). We will not bother with older algorithms which are not uniformly fast. Fishman's method is based upon rejection from the Poisson distribution, and is explored in exercise 4.1. The universal rejection algorithm derived from Theorem 1.1 is also uniformly fast, but since it was not specifically designed for the binomial distribution, it is not competitive with tailor-made rejection algorithms. To save space, only the algorithm of Devroye and Naderisamani (1980) will be developed in detail. Although this algorithm may not be the fastest on all computers, it has two desirable properties: the dominating curve is asymptotically tight because it exploits convergence to the normal distribution, and it does not require a subprogram for computing the log factorial in constant time.

C. Table methods. The finite number of values make the binomial distribution a good candidate for the table methods. To obtain uniformly fast speed, the table size has to grow in proportion to n, and a set-up time proportional to n is needed. It is generally accepted that the marginal execution times of the alias or alias-urn methods are difficult to beat. See sections III.3 and III.4 for details.

D. Generators based upon recursion (Relles (1972), Ahrens and Dieter (1974)). The problem of generating a binomial (n,p) random variate is usually reduced in constant time to that of generating another binomial random variate with much smaller value for n. This leads to $O(\log(n))$ or $O(\log\log(n))$ expected time algorithms. In view of the superior performance of the generators in classes B and C, the principle of recursion will be described very briefly, and most details can be found in the exercises.

4.3. Simple generators.

Lemma 4.1 leads to the

Coin flip method

$X \leftarrow 0$

FOR $i:=1$ TO n DO

Generate a random bit B (B is 1 with probability p, and can be obtained by generating a uniform [0,1] random variate U and setting $B = I_{[U \leq p]}$).

$X \leftarrow X + B$

RETURN X

This simple method requires time proportional to n. One can use n uniform random variates, but it is often preferable to generate just one uniform random variate and recycle the unused portion. This can be done by noting that a random bit and an independent uniform random variate can be obtained as $(I_{[U<p]}, \min(\frac{U}{p}, \frac{1-U}{1-p}))$. The coin flip method with recycling of uniform random variates can be rewritten as follows:

[NOTE: We assume that $p \leq 1/2$.]

$X \leftarrow 0$

Generate a uniform [0,1] random variate U.

FOR $i:=1$ TO n DO

$B \leftarrow I_{[U > 1-p]}$

$U \leftarrow \frac{U-(1-p)B}{pB+(1-p)(1-B)}$ (reuse the uniform random variate)

$X \leftarrow X + B$

RETURN X

For the important case $p = \frac{1}{2}$, it suffices to generate a random uniformly distributed computer word of n bits, and to count the number of ones in the word. In machine language, this can be implemented very efficiently by the standard bit operations.

Inversion by sequential search takes as we know expected time proportional to $E(X)+1 = np+1$. We can avoid tables of probabilities because of the recurrence relation

$$p_{i+1} = p_i \frac{(n-i)p}{(i+1)(1-p)} \quad (0 \leq i < n),$$

where $p_i = P(X=i)$. The algorithm will not be given here. It suffices to mention that for large n, the repeated use of the recurrence relation could also lead to accuracy problems. These problems can be avoided if one of the two waiting time algorithms (based upon Lemmas 4.4 and 4.5) is used:

First waiting time algorithm

$X \leftarrow -1$

Sum $\leftarrow 0$

REPEAT

  Generate a geometric (p) random variate G.

  Sum $\leftarrow$ Sum $+G$

  $X \leftarrow X+1$

UNTIL Sum $> n$

RETURN X

Second waiting time method

[SET-UP]

$q \leftarrow -\log(1-p)$

[GENERATOR]

$X \leftarrow 0$

Sum $\leftarrow 0$

REPEAT

  Generate an exponential random variate E.

  Sum $\leftarrow$ Sum $+\frac{E}{n-X}$ (Note: Sum is allowed to be ∞.)

  $X \leftarrow X+1$

UNTIL Sum $> q$

RETURN $X \leftarrow X-1$

Both waiting time methods have expected time complexities that grow as $np+1$.

4.4. The rejection method.

To develop good dominating curves, it helps to recall that by the central limit theorem, the binomial distribution tends to the normal distribution as $n \to \infty$ and p remains fixed. When p varies with n in such a way that $np \to c$, a positive constant, then the binomial distribution tends to the Poisson (c) distribution, which in turn is very close to the normal distribution for large values of c. It seems thus reasonable to consider the normal density as our dominating curve. Unfortunately, the binomial probabilities do not decrease quickly enough for one single normal density to be useful as a dominating curve. We cover the binomial tails with exponential curves and make use of Lemma 3.6. To keep things simple, we assume:

1. $\lambda = np$ is a nonzero integer.
2. $p \leq \frac{1}{2}$.

So as not to confuse p with $p_i = P(X=i)$, we use the notation

$$b_i = \binom{n}{i} p^i (1-p)^{n-i} \qquad (0 \leq i \leq n) .$$

The second assumption is not restrictive because a binomial (n,p) random variable is distributed as n minus a binomial ($n,1-p$) random variable. The first assumption is not limiting in any sense because of the following property.

Lemma 4.6.

If Y is a binomial (n,p') random variable with $p' \leq p$, and if conditional on Y, Z is a binomial $(n-Y, \frac{p-p'}{1-p'})$ random variable, then $X \leftarrow Y+Z$ is binomial (n,p).

Proof of Lemma 4.6.

The lemma is based upon the decomposition

$$X = \sum_{i=1}^{n} I_{[U_i \leq p]} = \sum_{i=1}^{n} I_{[U_i \leq p']} + \sum_{i=1}^{n} I_{[p' < U_i \leq p]} = Y+Z ,$$

where $U_1, \ldots, U_n$ are iid uniform [0,1] random variables. ■

To recapitulate, we offer the following generator for general values of n,p, but $0 < p \leq \frac{1}{2}$:

Splitting algorithm for binomial random variates

[NOTE: t is a fixed threshold, typically about 7. For $np \le t$, one of the waiting time algorithms is recommended. Assume thus that $np > t$.]

$p' \leftarrow \frac{1}{n} \lfloor np \rfloor$

Generate a binomial (n, p') random variate Y by the rejection method in uniformly bounded expected time.

Generate a binomial $(n-Y, \frac{p-p'}{1-p'})$ random variate Z by one of the waiting time methods.

RETURN $X \leftarrow Y+Z$

The expected time taken by this generator when $np > t$ is bounded from above by $c_1 + c_2 n \frac{p-p'}{1-p'} \le c_1 + 2c_2$ for some universal constants c_1, c_2. Thus, it can't harm to impose assumption 1.

Lemma 4.7.

For integer $0 \le i \le n(1-p)$ and integer $\lambda = np \ge 1$, we have

$$\log\left(\frac{b_{\lambda+i}}{b_\lambda}\right) \le -\frac{i(i-1)}{2n(1-p)} - \frac{i(i+1)}{2np+i}$$

and

$$\log\left(\frac{b_{\lambda+i}}{b_\lambda}\right) + \frac{i^2 + ((1-p)-p)i}{2np(1-p)} \begin{cases} \le s \\ \ge s-t \end{cases}$$

where

$$s = \frac{i(i+1)(2i+1)}{12n^2p^2} - \frac{(i-1)i(2i-1)}{12n^2(1-p)^2}$$

and

$$t = \frac{i^2(i-1)^2}{12n^2(1-p)^2(n(1-p)-i+1)} + \frac{i^2(i+1)^2}{12n^3p^3} .$$

For all integer $0 \le i \le np$, $\log\left(\frac{b_{\lambda-i}}{b_\lambda}\right)$ satisfies the same inequalities provided that p is replaced throughout by $1-p$ in the various expressions.

Proof of Lemma 4.7.

For $i=0$, the statements are obviously true because equality is reached. Assume thus that $0<i\leq n(1-p)$. We have

$$\frac{b_{\lambda+i}}{b_\lambda}=\frac{\binom{n}{\lambda+i}p^{\lambda+i}(1-p)^{n-\lambda-i}}{\binom{n}{\lambda}p^{\lambda}(1-p)^{n-\lambda}}=\left(\frac{p}{1-p}\right)^i\frac{\binom{n}{\lambda+i}}{\binom{n}{\lambda}}$$

$$=\left(\frac{p}{1-p}\right)^i\frac{(n-\lambda)!\lambda!}{(n-\lambda-i)!(\lambda+i)!}$$

$$=\frac{\prod_{j=0}^{i-1}\left(1-\frac{j}{n(1-p)}\right)}{\prod_{j=0}^{i}\left(1+\frac{j}{np}\right)} .$$

Thus,

$$\log\left(\frac{b_{\lambda+i}}{b_\lambda}\right)=\sum_{j=0}^{i-1}\log\left(1-\frac{j}{n(1-p)}\right)-\sum_{j=0}^{i}\log\left(1+\frac{j}{np}\right)$$

$$\leq-\sum_{j=0}^{i-1}\frac{j}{n(1-p)}-\sum_{j=0}^{i}\frac{2j}{2np+j}$$

$$\leq-\frac{i(i-1)}{2n(1-p)}-\frac{i(i+1)}{2np+i} .$$

Here we used Lemma 3.6. This proves the first statement of the lemma. Again by Lemma 3.6, we see that

$$\log\left(\frac{b_{\lambda+i}}{b_\lambda}\right)\leq\sum_{j=0}^{i-1}\left(-\frac{j}{n(1-p)}-\frac{j^2}{2n^2(1-p)^2}\right)+\sum_{j=0}^{i}\left(-\frac{j}{np}+\frac{j^2}{2n^2p^2}\right)$$

$$=-\frac{i^2+((1-p)-p)i}{2np(1-p)}+s .$$

Furthermore,

$$\log\left(\frac{b_{\lambda+i}}{b_\lambda}\right)\geq\sum_{j=0}^{i-1}\left(-\frac{j}{n(1-p)}-\frac{j^2}{2n^2(1-p)^2}-\frac{j^3}{3n^3(1-p)^3\left(1-\frac{i-1}{n(1-p)}\right)}\right)$$

$$+\sum_{j=0}^{i}\left(-\frac{j}{np}+\frac{j^2}{2n^2p^2}-\frac{j^3}{3n^3p^3}\right)$$

$$=-\frac{i^2+((1-p)-p)i}{2np(1-p)}+s-t .$$

This concludes the proof of the first part of Lemma 4.7. For integer $0<i\leq np$,

we have

$$\frac{b_{\lambda-i}}{b_\lambda} = \left(\frac{p}{1-p}\right)^{-i} \frac{\binom{n}{\lambda-i}}{\binom{n}{\lambda}}$$

$$= \frac{\prod_{j=0}^{i-1}\left(1-\frac{j}{np}\right)}{\prod_{j=0}^{i}\left(1+\frac{j}{n(1-p)}\right)} .$$

This is formally the same as an expression used as starting point above, provided that p is replaced throughout by $1-p$. ■

Lemma 4.7 is used in the construction of a useful function $g(x)$ with the property that for all $x \in [i, i+1)$, and all allowable i ($-np \le i \le n(1-p)$),

$$g(x) \ge \log\left(\frac{b_{\lambda+i}}{b_\lambda}\right) .$$

The algorithm is of the form:

REPEAT

Generate a random variate Y with density proportional to e^g.

Generate an exponential random variate E.

$X \leftarrow \lfloor Y \rfloor$ (this is truncation to the left, even for negative values of Y)

UNTIL $[-np \le X \le n(1-p)]$ AND $[g(Y) \le \log(\frac{b_{\lambda+X}}{b_\lambda})+E]$

RETURN $X \leftarrow \lambda+X$

The normal-exponential dominating curve e^g suggested earlier is defined in Lemma 4.8:

Lemma 4.8.

Let $\delta_1 \geq 1$, $\delta_2 \geq 1$ be given integers. Define furthermore

$$\sigma_1 = \sqrt{np(1-p)}(1+\frac{\delta_1}{4np}) ,$$

$$\sigma_2 = \sqrt{np(1-p)}(1+\frac{\delta_2}{4n(1-p)}) ,$$

$$c = \frac{2\delta_1}{np} .$$

Then the function g can be chosen as follows:

$$g(x) = \begin{cases} c - \frac{x^2}{2\sigma_1^2} & (0 \leq x < \delta_1) \\ \frac{\delta_1}{n(1-p)} - \frac{\delta_1 x}{2\sigma_1^2} & (\delta_1 < x) \\ -\frac{x^2}{2\sigma_2^2} & (-\delta_2 < x < 0) \\ -\frac{\delta_2 x}{2\sigma_2^2} & (x \leq -\delta_2) \end{cases} .$$

Proof of Lemma 4.8.

For $i=0$ we need to show that $c \geq 1/(2\sigma_1^2)$. This follows from

$$2c\,\sigma_1^2 = \frac{4\delta_1}{np} np(1-p)(1+\frac{\delta_1}{4np})^2 \geq 4\delta_1(1-p) \geq 2\delta_1 \geq 2 .$$

When $0 < i < \delta_1$, we have

$$-\frac{i(i-1)}{2n(1-p)} \leq -\frac{(x-1)(x-2)}{2n(1-p)} ,$$

$$-\frac{i(i+1)}{2np+i} \leq -\frac{x(x-1)}{2np+\delta_1} .$$

By Lemma 4.7,

$$\log(\frac{b_{np+i}}{b_{np}}) \leq -\frac{(x-1)(x-2)}{2n(1-p)} - \frac{x(x-1)}{2np+\delta_1}$$

$$= -(\frac{1}{2n(1-p)} + \frac{1}{2np+\delta_1})x^2 + (\frac{3}{2n(1-p)} + \frac{1}{2np+\delta_1})x - \frac{1}{n(1-p)}$$

$$\leq -(\frac{1}{2n(1-p)} + \frac{1}{2np+\delta_1})x^2 + \frac{2\delta_1}{np}$$

$$\leq -\frac{x^2}{2\sigma_1^2} + \frac{2\delta_1}{np} .$$

The last step follows by application of the inequality $\sqrt{1+u} < 1+\frac{u}{2}$, valid for $u > 0$, in the following chain of inequalities:

$$\frac{1}{2n(1-p)}+\frac{1}{2np+\delta_1} = \frac{1+\frac{\delta_1}{2n}}{2np(1-p)(1+\frac{\delta_1}{2np})}$$

$$\geq \frac{1}{2np(1-p)(1+\frac{\delta_1}{2np})}$$

$$\geq \frac{1}{(\sqrt{2np(1-p)}(1+\frac{\delta_1}{4np}))^2} = \frac{1}{2\sigma_1^2} .$$

When $i \geq \delta_1$, we have

$$-\frac{i(i-1)}{2n(1-p)} \leq -\frac{\delta_1(x-2)}{2n(1-p)} ; -\frac{i(i+1)}{2np+i} \leq -\frac{\delta_1 x}{2np+\delta_1} .$$

By Lemma 4.7,

$$\log(\frac{b_{np+i}}{b_{np}}) \leq -\frac{\delta_1(x-2)}{2n(1-p)} - \frac{\delta_1 x}{2np+\delta_1}$$

$$= -(\frac{1}{2np}+\frac{1}{2np+\delta_1})\delta_1 x+\frac{\delta_1}{n(1-p)}$$

$$\leq -\frac{\delta_1 x}{2\sigma_1^2}+\frac{\delta_1}{n(1-p)} .$$

When $0 > i \geq -\delta_2$, we have

$$\log(\frac{b_{np+i}}{b_{np}}) \leq -\frac{i(i+1)}{2np} - \frac{i(i-1)}{2n(1-p)-i}$$

$$= -(\frac{1}{2np}+\frac{1}{2n(1-p)+\delta_2})i^2-\frac{i}{2np}+\frac{i}{2n(1-p)+\delta_2}$$

$$\leq -(\frac{1}{2np}+\frac{1}{2n(1-p)+\delta_2})x^2$$

$$\leq -\frac{x^2}{2\sigma_2^2} .$$

Finally, when $i < -\delta_2$, we see that

$$-\frac{i(i+1)}{2np} \leq \frac{\delta_2 x}{2np} ;$$

$$-\frac{i(i-1)}{2n(1-p)-i} \leq \frac{\delta_2(i-1)}{2n(1-p)+\delta_2} \leq \frac{\delta_2(x-1)}{2n(1-p)+\delta_2} .$$

Therefore,

$$\log\left(\frac{b_{np+i}}{b_{np}}\right) \le -\frac{\delta_2 x}{2np} + \frac{\delta_2(x-1)}{2n(1-p)+\delta_2}$$

$$= \left(\frac{1}{2np} + \frac{1}{2n(1-p)+\delta_2}\right)\delta_2 x - \frac{\delta_2}{2n(1-p)+\delta_2}$$

$$\le \frac{\delta_2 x}{2\sigma_2^2} . \blacksquare$$

The dominating curve e^g suggested by Lemma 4.8 consists of four pieces, one piece per interval. The integrals of e^g over these intervals are needed by the generator. These are easy to compute for the exponential tails, but not for the normal center intervals. Not much will be lost if we replace the two normal pieces by halfnormals on the positive and negative real line respectively, and reject when the normal random variates fall outside $[-\delta_2,\delta_1]$. This at least allows us to work with the integrals of halfnormal curves. We will call the areas under the different components of e^g a_i $(1 \le i \le 4)$. Thus,

$$a_1 = \int_0^\infty e^{c-\frac{x^2}{2\sigma_1^2}}\, dx = \frac{1}{2} e^c \sigma_1 \sqrt{2\pi} ,$$

$$a_2 = \frac{1}{2}\sigma_2\sqrt{2\pi} ,$$

$$a_3 = \int_{\delta_1}^\infty e^{\frac{\delta_1}{n(1-p)} - \frac{\delta_1 x}{2\sigma_1^2}}\, dx = e^{\frac{\delta_1}{n(1-p)}} \frac{2\sigma_1^2}{\delta_1} e^{-\frac{\delta_1^2}{2\sigma_1^2}} ,$$

$$a_4 = \frac{2\sigma_2^2}{\delta_2} e^{-\frac{\delta_2^2}{2\sigma_2^2}} .$$

We can now summarize the algorithm:

A rejection algorithm for binomial random variates

[SET-UP]

$\sigma_1 \leftarrow \sqrt{np(1-p)}(1+\delta_1/(4np)), \sigma_2 \leftarrow \sqrt{np(1-p)}(1+\delta_2/(4n(1-p))),\ c \leftarrow 2\delta_1/(np)$

$a_1 \leftarrow \frac{1}{2} e^c \sigma_1 \sqrt{2\pi},\ a_2 \leftarrow \frac{1}{2}\sigma_2\sqrt{2\pi}$

$a_3 \leftarrow e^{\frac{\delta_1}{n(1-p)}} \frac{2\sigma_1^2}{\delta_1} e^{-\frac{\delta_1^2}{2\sigma_1^2}}$

$a_4 \leftarrow \frac{2\sigma_2^2}{\delta_2} e^{-\frac{\delta_2^2}{2\sigma_2^2}}$

$s \leftarrow a_1+a_2+a_3+a_4$

[GENERATOR]

REPEAT

  Generate a uniform $[0,s]$ random variate U.

  CASE

    $U \leq a_1$:

      Generate a normal random variate N; $Y \leftarrow \sigma_1 |N|$

      Reject $\leftarrow [Y \geq \delta_1]$

      IF NOT Reject THEN $X \leftarrow \lfloor Y \rfloor, V \leftarrow -E - \frac{N^2}{2} + c$ where E is an exponential random variate.

    $a_1 < U \leq a_1+a_2$:

      Generate a normal random variate N; $Y \leftarrow \sigma_2 |N|$

      Reject $\leftarrow [Y \geq \delta_2]$

      IF NOT Reject THEN $X \leftarrow \lfloor -Y \rfloor, V \leftarrow -E - \frac{N^2}{2}$ where E is an exponential random variate.

    $a_1+a_2 < U \leq a_1+a_2+a_3$:

      Generate two iid exponential random variates E_1, E_2.

      $Y \leftarrow \delta_1 + 2\sigma_1^2 E_1/\delta_1$

      $X \leftarrow \lfloor Y \rfloor, V \leftarrow -E_2 - \delta_1 Y/(2\sigma_1^2) + \delta_1/(n(1-p$

      Reject $\leftarrow$ False

    $a_1+a_2+a_3 < U$:

      Generate two iid exponential random variates E_1, E_2.

      $Y \leftarrow \delta_2 + 2\sigma_2^2 E_1/\delta_2$

      $X \leftarrow \lfloor -Y \rfloor, V \leftarrow -E_2 - \delta_2 Y/(2\sigma_2^2)$

      Reject $\leftarrow$ False

  Reject $\leftarrow$ Reject OR $[X < -np]$ OR $[X > n(1-p)]$

  Reject $\leftarrow$ Reject OR $[V > \log(b_{np+X}/b_{np})]$

UNTIL NOT Reject

RETURN X

We need only choose δ_1,δ_2 so that the expected number of iterations is approximately minimal. This is done in Lemma 4.9.

Lemma 4.9.

Assume that $p \le \frac{1}{2}$ and that as $\lambda=np \to \infty$, we have uniformly in p, $\delta_1=o(\lambda), \delta_2=o(n)$, $\delta_1/\sqrt{\lambda}\to\infty$, $\delta_2/\sqrt{np} \to \infty$. Then the expected number of iterations is uniformly bounded over $n \ge 1, 0 \le p \le \frac{1}{2}$, and tends to 1 uniformly in p as $\lambda\to\infty$.

The conditions on δ_1,δ_2 are satisfied for the following (nearly optimal) choices:

$$\delta_1 = \left\lfloor \max\left(1, \sqrt{np(1-p)\log\left(\frac{128np}{81\pi(1-p)}\right)}\right) \right\rfloor ;$$

$$\delta_2 = \left\lfloor \max\left(1, \sqrt{np(1-p)\log\left(\frac{128n(1-p)}{\pi p}\right)}\right) \right\rfloor .$$

Proof of Lemma 4.9.

We first observe that under the stated conditions on δ_1,δ_2, we have

$$\sigma_1 = \sqrt{np(1-p)}(1+o(1)),\ \sigma_2 = \sqrt{np(1-p)}(1+o(1)),$$

$$c = o(1),$$

$$a_1 = \sqrt{\frac{\pi np(1-p)}{2}}(1+o(1)),\ a_2 = \sqrt{\frac{\pi np(1-p)}{2}}(1+o(1)),$$

$$a_3 = \frac{2np(1-p)}{\delta_1}(1+o(1))e^{-\frac{\delta_1^2(1+o(1))}{2np(1-p)}},$$

$$a_4 = \frac{2np(1-p)}{\delta_2}(1+o(1))e^{-\frac{\delta_2^2(1+o(1))}{2np(1-p)}},$$

$$a_1+a_3 \sim a_1,\ a_2+a_4 \sim a_2,$$

$$a_1+a_2+a_3+a_4 \sim \sqrt{2\pi np(1-p)} .$$

The expected number of iterations in the algorithm is $(a_1+a_2+a_3+a_4)b_{np} \sim \sqrt{2\pi np(1-p)}/\sqrt{2\pi np(1-p)} = 1$. All $o(.)$ and $\sim$ symbols inherit the uniformity with respect to p, as long as $\lambda\to\infty$. The uniform boundedness of the expected number of iterations follows from this.

The particular choices for δ_1,δ_2 are easily seen to satisfy the convergence conditions. That they are nearly optimal (with respect to the minimization of the expected number of iterations) is now shown. The minimization of a_1+a_3 would provide us with a good value for δ_1. In the asymptotic expansions for a_1,a_3, it is now necessary to consider the first two terms, not just the main term. In particular, we have

$$a_1 = \sqrt{\frac{\pi np(1-p)}{2}} e^c (1+\frac{\delta_1}{4np}) = \sqrt{\frac{\pi np(1-p)}{2}} (1+\frac{(9+o(1))\delta_1}{4np}) ,$$

$$a_3 = \frac{2np(1-p)}{\delta_1} e^{-\frac{(1+o(1))\delta_1^2}{2np(1-p)}} \approx \frac{2np(1-p)}{\delta_1} e^{-\frac{\delta_1^2}{2np(1-p)}} .$$

Setting the derivative of the sum of the two right-hand-side expressions equal to zero gives the equation

$$\frac{\delta_1^2}{np(1-p)} e^{\frac{\delta_1^2}{2np(1-p)}} = (1+\frac{\delta_1^2}{np(1-p)})\sqrt{np(1-p)} \frac{8\sqrt{2}}{9(1-p)\sqrt{\pi}} .$$

Disregarding the term "1" with respect to $\frac{\delta_1^2}{np(1-p)}$ and solving with respect to δ_1 gives

$$\delta_1 = \sqrt{np(1-p)\log(\frac{128np}{81\pi(1-p)})} .$$

A suitable expression for δ_2 can be obtained by a similar argument. Indeed,

$$a_2+a_4 = \sqrt{\frac{\pi np(1-p)}{2}}(1+\frac{\delta_2}{4n(1-p)})$$
$$+ (1+o(1))\frac{2np(1-p)}{\delta_2} e^{-\frac{(1+o(1))\delta_2^2}{2np(1-p)}} .$$

Disregard the $o(1)$ term, and set the derivative of the resulting expression with respect to δ_2 equal to zero. This gives the equation

$$\frac{e^{\frac{\delta_2^2}{2np(1-p)}}}{4n(1-p)} = 2(np(1-p)+\delta_2^2)\sqrt{\frac{2}{\pi np(1-p)}} \sim \sqrt{\frac{8}{\pi np(1-p)}}\delta_2^2 .$$

If $\sim$ is replaced by equality, then the solution with respect to δ_2 is

$$\delta_2 = \sqrt{np(1-p)\log(\frac{128n(1-p)}{\pi p})} . \blacksquare$$

Lemma 4.9 is crucial for us. For large values of np, the rejection constant is nearly 1. Also, since δ_1 and δ_2 are large compared to the standard deviation $\sqrt{np(1-p)}$ of the distribution, the exponential tails float to infinity as $np \to \infty$. In other words, we exit most of the time with a properly scaled normal random variate. At this point we leave the algorithm. The interested readers can find more information in the exercises. For example, the evaluation of b_{np+i}/b_{np}

takes time proportional to $1+|i|$. This implies that the expected complexity grows as $\sqrt{np(1-p)}$ when $np \to \infty$. It can be shown that the expected complexity is uniformly bounded if we do one of the following:

A. Use squeeze steps suggested in Lemma 4.7, and evaluate b_{np+i}/b_{np} explicitly when the squeeze steps fail.

B. Use squeeze steps based upon Stirling's series (Lemma 1.1), and evaluate b_{np+i}/b_{np} explicitly when the squeeze steps fail.

C. Make all decisions involving factorials based upon sequentially evaluating more and more terms in Binet's convergent series for factorials (Lemma 1.2).

D. Assume that the log gamma function is a unit cost function.

4.5. Recursive methods.

The recursive methods are all based upon the connection between the binomial and beta distributions given in Lemma 4.6. This is best visualized by considering the order statistics $U_{(1)} < \cdots < U_{(n)}$ of iid uniform [0,1] random variables, and noting that the number of $U_{(i)}$'s in $[0,p]$ is binomial (n,p). Let us call this quantity X. Furthermore, $U_{(i)}$ itself is beta $(i,n+1-i)$ distributed. Because $U_{(i)}$ is approximately $\frac{i}{n+1}$, we can begin with generating a beta $(i,n+1-i)$ random variate Y with $i = \lfloor (n+1)p \rfloor$. Y should be close to p. In any case, we have gone a long way toward solving our problem. Indeed, if $Y \le p$, we note that X is equal to i plus the number of $U_{(j)}$'s in the interval $(Y,p]$, which we know is binomial $(n-i, \frac{p-Y}{1-Y})$ distributed. By symmetry, if $Y > p$, X is equal to i minus a binomial $(i-1, \frac{Y-p}{Y})$ random variate. Thus, the following recursive program can be used:

Recursive binomial generator

[NOTE: n and p will be destroyed by the algorithm.]

$X \leftarrow 0, S \leftarrow +1$ (S is a sign)

REPEAT

IF $np < t$ (t is a design constant)

THEN

Generate a binomial (n, p) random variate B by a simple method such as the waiting time method.

RETURN $X \leftarrow X + SB$

ELSE

Generate a beta $(i, n+1-i)$ random variate Y with $i = \lfloor (n+1)p \rfloor$.

$X \leftarrow X + Si$

IF $Y \leq p$

THEN $n \leftarrow n-i, p \leftarrow \frac{p-Y}{1-Y}$

ELSE $S \leftarrow -S, n \leftarrow i-1, p \leftarrow \frac{Y-p}{Y}$

UNTIL False

In this simple algorithm, we use a uniformly fast beta generator. The simple binomial generator alluded to should be such that its expected time is $O(np)$. Note however that it is not crucial: the algorithm works fine even if we set $t=0$ and thus bypass the simple binomial generator. The algorithm halts when $n=0$, which happens with probability one.

Let us give an informal outline of the proof of the claim that the expected time taken by the algorithm is bounded by a constant times $\log(\log(n))$. By the properties of the beta distribution, $Y-p$ is of the order of $\sqrt{\frac{i(n-i)}{n^3}}$, i.e. it is approximately $\sqrt{p(1-p)/n}$. Since Y itself is close to p, we see that the new values for (n,p) are either about $(n(1-p), \sqrt{p/((1-p)n)})$ or about $(np, \sqrt{(1-p)/(pn)})$. The new product np is thus of the order of magnitude of $\sqrt{np(1-p)}$. We see that np gets replaced at worst by about $\sqrt{np}$ in one iteration. In k iterations, we have about

$$(np)^{2^{-k}} .$$

Since we stop when this reaches t, our constant, the number of iterations should be of the order of magnitude of

$$\log(\frac{\log(np)}{\log(t)}) .$$

This argument can be formalized, and the mathematically inclined reader is urged to do so (exercise 4.7). Since the loglog function increases very slowly, the recursive method can be competitive depending upon the beta generator. It was precisely the latter point, poor speed of the pre-1975 beta generators, which prompted Relles (1972) and Ahrens and Dieter (1974) to propose slightly different recursive generators in which i is not chosen as $\lfloor (n+1)p \rfloor$, but rather as $(n+1)/2$ when n is odd. This implies that all beta random variates needed are symmetric beta random variates, which can be generated quite efficiently. Because n gets halved at every iteration, their algorithm runs in $O(\log(n))$ time.

4.6. Symmetric binomial random variates.

The purpose of this section is to point out that in the case $p = \frac{1}{2}$ a single normal dominating curve suffices in the rejection algorithm, and to present and analyze the following simple rejection algorithm:

Rejection method for symmetric binomial random variates

[NOTE: This generator returns a binomial $(2n, \frac{1}{2})$ random variate.]

[SET-UP]

$s \leftarrow 1/\sqrt{2(n^{-1}-(2n^2)^{-1})}, \sigma \leftarrow s+\frac{1}{4}, \ c \leftarrow 2/(1+8s)$

[GENERATOR]

REPEAT

Generate a normal random variate N and an exponential random variate E.

$Y \leftarrow \sigma N, \ X \leftarrow \text{round}(Y)$

$T \leftarrow -E+c-\frac{1}{2}N^2+\frac{1}{n}X^2$

Reject $\leftarrow [\,|X| > n\,]$

IF NOT Reject THEN

Accept $\leftarrow \left[T < -\frac{X^4}{6n^3(1-(\frac{|X|-1}{n})^2)}\right]$

IF NOT Accept THEN

Reject $\leftarrow \left[T > \frac{X^2}{2n^2}\right]$

IF NOT Reject THEN

Accept $\leftarrow \left[T > \log(\frac{b_{n+X}}{b_n})+\frac{X^2}{n}\right]$

UNTIL NOT Reject AND Accept

RETURN $X \leftarrow n+X$

The algorithm has one quick acceptance step and one quick rejection step designed to reduce the probability of having to evaluate the final acceptance step which involves computing the logarithms of two binomial probabilities. The validity of the algorithm follows from the following Lemma.

Lemma 4.10.

Let $b_0, \ldots, b_{2n}$ be the probabilities of a binomial $(2n,p)$ distribution. Then, for any $\sigma > s$,

$$\log\left(\frac{b_{n+i}}{b_n}\right) \leq c - \frac{(|i|+\frac{1}{2})^2}{2\sigma^2} \quad (\text{integer } i, \ |i| \leq n),$$

where $c = 1/(8(\sigma^2 - s^2))$. Also, for all $n > i > 0$,

$$-\frac{i^4}{6n^3(1-(\frac{i-1}{n})^2)} \leq \log\left(\frac{b_{n+i}}{b_n}\right) + \frac{i^2}{n} \leq \frac{i^2}{2n^2}.$$

Proof of Lemma 4.10.

We will use repeatedly the following fact: for $1 > x > 0$,

$$-2x - \frac{2x^3}{3(1-x^2)} < \log\left(\frac{1-x}{1+x}\right) < -2x - \frac{2x^3}{3},$$

$$-\frac{1}{2}x^2 < \log(1+x) - x < 0.$$

The first inequality follows from the fact that $\log(\frac{1-x}{1+x})$ has series expansion $-2(x + \frac{1}{3}x^3 + \frac{1}{5}x^5 + \cdots)$. Thus, for $n > i > 0$,

$$\log\left(\frac{b_{n+i}}{b_n}\right) = \log\left(\frac{n!n!}{(n+i)!(n-i)!}\right) = \log\left(\prod_{j=1}^{i-1} \frac{1-\frac{j}{n}}{1+\frac{j}{n}} \frac{1}{1+\frac{i}{n}}\right)$$

$$= \sum_{j=1}^{i-1} \left(\log\left(\frac{1-\frac{j}{n}}{1+\frac{j}{n}}\right) + \frac{2j}{n}\right) - \left(\log(1+\frac{i}{n}) - \frac{i}{n}\right) - \frac{i^2}{n}$$

$$= c_i + d_i - \frac{i^2}{n}.$$

We have

$$0 + \frac{1}{2}\left(\frac{i}{n}\right)^2 \geq c_i + d_i$$

$$\geq -\sum_{j=1}^{i-1} \frac{2}{3}\left(\frac{j}{n}\right)^3 \left(1-\left(\frac{j}{n}\right)^2\right)^{-1} + 0$$

$$\geq -\frac{2}{3}\left(1-\left(\frac{i-1}{n}\right)^2\right)^{-1} \sum_{j=1}^{i-1} \left(\frac{j}{n}\right)^3$$

$$\geq -\frac{2}{3}(1-(\frac{i-1}{n})^2)^{-1}\frac{i^4}{4n^3} .$$

Thus,

$$\log(\frac{b_{n+i}}{b_n}) \leq -\frac{i^2}{n}+\frac{i^2}{2n^2} = -\frac{i^2}{2s^2} \leq c-\frac{(|i|+\frac{1}{2})^2}{2\sigma^2} \quad (|i| \leq n),$$

where

$$c = \sup_{u>0} \frac{(u+\frac{1}{2})^2}{2\sigma^2}-\frac{u^2}{2s^2} .$$

Assuming that $\sigma > s$, this supremum is reached for

$$u = \frac{s^2}{2(\sigma^2-s^2)} , c = \frac{1}{8(\sigma^2-s^2)} . \blacksquare$$

The dominating curve suggested by Lemma 4.11 is a centered normal density with variance σ^2. The best value for σ is that for which the area $\sqrt{2\pi}\sigma e^c$ is minimal. Setting the derivative with respect to σ of the logarithm of this expression equal to 0 gives the equation

$$\sigma^2-\frac{1}{2}\sigma-s^2 = 0.$$

The solution is $\sigma = \frac{1}{4}+s\sqrt{1+1/(16s^2)} = \frac{1}{4}+s+o(1)$. It is for this reason that the value $\sigma=s+\frac{1}{4}$ was taken in the algorithm. The corresponding value for c is $2/(1+8s)$.

The expected number of iterations is $b_n \sqrt{2\pi}\sigma e^c \sim \frac{1}{\sqrt{\pi n}}\sqrt{2\pi}\sqrt{\frac{n}{2}} = 1$ as $n \to \infty$. Assuming that b_{n+i}/b_n takes time $1+|i|$ when evaluated explicitly, it is clear that without the squeeze steps, we would have obtained an expected time which would grow as $\sqrt{n}$ (because the i is distributed as σ times a normal random variate). The efficiency of the squeeze steps is highlighted in the following Lemma.

Lemma 4.11.

The algorithm shown above is uniformly fast in n when the quick acceptance step is used. If in addition a quick rejection step is used, then the expected time due to the explicit evaluation of b_{n+i}/b_n is $O(1/\sqrt{n})$.

Proof of Lemma 4.11.

Let $p(x)$ be the probability that the inequality in the quick acceptance step is not satisfied for fixed $X=x$. We have $P(|X| \geq 1+n\sqrt{5/6}) = O(r^{-n})$ for some $r>1$. For $|x| \leq 1+n\sqrt{5/6}$, we have in view of $|Y^2-x^2| \leq (|x|+\frac{1}{2})/2$,

$$p(x) \leq P(-E+c-\frac{(x^2-\frac{1}{4}-\frac{|x|}{2})}{2\sigma^2}+\frac{x^2}{n} > -\frac{x^4}{n^3})$$

$$\leq P(E < c+\frac{1}{8\sigma^2}+\frac{|x|}{4\sigma^2}+x^2(\frac{1}{n}-\frac{1}{2\sigma^2})+\frac{x^4}{n^3})$$

$$\leq 2c+\frac{|x|}{4\sigma^2}+x^2(\frac{1}{n}-\frac{1}{2\sigma^2})+\frac{x^4}{n^3}$$

$$= O(n^{-\frac{1}{2}})+|x|\,O(n^{-1})+x^2O(n^{-\frac{3}{2}})+x^4O(n^{-3}) .$$

Thus, the probability that a couple (X,E) does not satisfy the quick acceptance condition is $E(p(X))$. Since $E(|X|)=O(\sigma)=O(\sqrt{n})$, $E(X^2)=O(n)$ and $E(X^4)=O(n^2)$, we conclude that $E(p(X))=O(1/\sqrt{n})$. If every time we rejected, we were to start afresh with a new couple (X,E), the expected number of such couples needed before halting would be $1+O(1/\sqrt{n})$. Using this, it is also clear that in the algorithm without quick rejection step, the expected time is bounded by a constant times $1+E(|X|\,p(X))$. But

$$E(|X|\,p(X)) \leq E(|X|\,I_{[|X|>1+n\sqrt{5/6}]})+E(|X|)O(n^{-\frac{1}{2}})$$

$$+E(X^2)O(n^{-1})+E(|X|^3)O(n^{-\frac{3}{2}})+E(|X|^5)O(n^{-3})$$

$$= O(1) .$$

This concludes the proof of the first statement of the Lemma. If a quick rejection step is added, and $q(x)$ is the probability that for $X=x$, both the quick acceptance and rejection steps are failed, then, arguing as before, we see that for $|x| \leq 1+n\sqrt{5/6}$,

$$q(x) \leq \frac{x^4}{n^3}+\frac{x^2}{n^2} .$$

Thus, the probability that both inequalities are violated is

$$E(q(X)) \leq \frac{E(X^4)}{n^3}+\frac{E(X^2)}{n^2}+P(|X| \geq 1+n\sqrt{5/6}) = O(\frac{1}{n}).$$

The expected time spent on explicitly evaluating factorials is bounded by a constant times $1+E(|X|\,q(X))=O(1/\sqrt{n})$. ■

4.7. The negative binomial distribution.

In section X.1, we introduced the negative binomial distribution with parameters (n,p), where $n \geq 1$ is an integer and $p \in (0,1)$ is a real number as the distribution of the sum of n iid geometric random variables. It has generating function

$$\left(\frac{p}{1-(1-p)s}\right)^n .$$

Using the binomial theorem, and equating the coefficients of s^i with the probabilities p_i for all i shows that the probabilities are

$$P(X=i) = p_i = \binom{-n}{i} p^n (-1+p)^i = \binom{n+i-1}{i} p^n (1-p)^i \quad (i \geq 0) .$$

When $n=1$, we obtain the geometric (p) distribution. For $n=1$, X is distributed as the number of failures in a sequence of independent experiments, each having success probability p, before the n-th success is encountered. From the properties of the geometric distribution, we see that the negative binomial distribution has mean $\frac{n(1-p)}{p}$ and variance $\frac{n(1-p)}{p^2}$.

Generation by summing n iid geometric p random variates yields at best an algorithm taking expected time proportional to n. The situation is even worse if we employ Example 1.4, in which we showed that it suffices to sum N iid logarithmic series $(1-p)$ random variates where N itself is Poisson (λ) and $\lambda = n\log(\frac{1}{p})$. Here, at best, the expected time grows as $E(N) = n\log(\frac{1}{p})$.

The property that one can use to construct a uniformly fast generator is obtained in Example 1.5: a negative binomial random variate can be generated as a Poisson (Y) random variate where Y in turn is a gamma $(n,\frac{1-p}{p})$ random variate. The same can be achieved by designing a uniformly fast rejection algorithm from scratch.

4.8. Exercises.

1. **Binomial random variates from Poisson random variates.** This exercise is motivated by an idea first proposed by Fishman (1979), namely to generate binomial random variates by rejection from Poisson random variates. Let b_i be the probability that a binomial (n,p) random variable takes the value i, and let p_i be the probability that a Poisson $((n+1)p)$ random variable takes the value i.

   A. Prove the crucial inequality $\sup_i b_i/p_i \leq e^{1/(12(n+1))}/\sqrt{1-p}$, valid for all n and p. Since we can without loss of generality assume that $p \leq \frac{1}{2}$, this implies that we have a uniformly fast binomial generator if we have a uniformly fast Poisson generator, and if we can handle the evaluation of b_i / p_i in uniformly bounded time. To prove the inequality, start with inequalities for the factorial given in Lemma 1.1, write i as $(n+1)p+x$, note that $x \leq (n+1)(1-p)$, and use the inequality $1+u \geq e^{u/(1+u)}$, valid for all $u > -1$.

B. Give the details of the rejection algorithm, in which factorials are squeezed by using the zero-term and one-term bounds of Lemma 1.1, and are explicitly evaluated as products when the squeezing fails.

C. Prove that the algorithm given in B is uniformly fast over all $n \geq 1, p \leq 1/2$ if Poisson random variates are generated in uniformly bounded expected time (not worst case time).

2. **Bounds for the mode of the binomial distribution.** Consider a binomial (n,p) distribution in which np is integer. Then the mode m is at np, and

$$\binom{n}{m} p^m (1-p)^{n-m} \leq \frac{e^{\frac{1}{12(n+1)}+\frac{1}{n^2 p(1-p)+n+1}}}{\sqrt{2\pi np(1-p)}} \leq \frac{2}{\sqrt{2\pi np(1-p)}} .$$

Prove this inequality by using the Stirling-Whittaker-Watson inequality of Lemma 1.1, and the inequalities $e^{u/(1+u)} \leq 1+u \leq e^u$, valid for $u \geq 0$ (Devroye and Naderisamani, 1980).

3. Add the squeeze steps suggested in the text to the normal-exponential algorithm, and prove that with this addition the expected complexity of the algorithm is uniformly bounded over all $n \geq 1$, $0 < p \leq \frac{1}{2}$, np integer (Devroye and Naderisamani, 1980).

4. A continuation of the previous exercise. Show that for fixed $p \leq \frac{1}{2}$, the expected time spent on the explicit evaluation of b_{np+i} / b_{np} is $O(1/\sqrt{np(1-p)})$ as $n \to \infty$. (This implies that the squeeze steps of Lemma 4.7 are very powerful indeed.)

5. Repeat exercise 3 but use squeeze steps based upon bounds for the log gamma function given in Lemma 1.1.

6. **The hypergeometric distribution.** Suppose an urn contains N balls, of which M are white and $N-M$ are black. If a sample of n balls is drawn at random without replacement from the urn, then the number (X) of white balls drawn is hypergeometrically distributed with parameters n, M, N. We have

$$P(X=i) = \frac{\binom{M}{i}\binom{N-M}{n-i}}{\binom{N}{n}} \quad (\max(0,n-N+M) \leq i \leq \min(n,M)) .$$

Note that the same distribution is obtained when n and M are interchanged. Note also that if we had sampled with replacement, we would have obtained the binomial $(n, \frac{M}{N})$ distribution.

A. Show that if a hypergeometric random variate is generated by rejection from the binomial $(n, \frac{M}{N})$ distribution, then we can take $(1-\frac{n}{N})^{-n}$ as rejection constant. Note that this tends to 1 as $n^2/N \to 0$.

B. Using the facts that the mean is $n\frac{M}{N}$, that the variance σ^2 is $\frac{N-n}{N-1}n\frac{M}{N}(1-\frac{M}{N})$, and that the distribution is unimodal with a mode at $\left\lfloor (n+1)\frac{M+1}{N+2} \right\rfloor$, give the details for the universal rejection algorithm of section X.1. Comment on the expected time complexity, i.e. on the maximal value for $(\sigma B)^{2/3}$ where B is an upper bound for the value of the distribution at the mode.

C. Find a function $g(x)$ consisting of a constant center piece and two exponential tails, having the properties that the area under the function is uniformly bounded, and that the function has the property that for every i and all $x \in [i-\frac{1}{2}, i+\frac{1}{2})$, $g(x) \geq P(X=i)$. Give the corresponding rejection algorithm (hint: recall the universal rejection algorithm of section X.1) (Kachitvichyanukul, 1982; Kachitvichyanukul and Schmeiser, 1985).

7. Prove that for all constant $t>0$, there exists a constant C only depending upon t such that the expected time needed by the recursive binomial algorithm given in the text is not larger than $C\log(\log(n+10))$ for all n and p. The term "10" is added to make sure that the loglog function is always strictly positive. Show also that for a fixed $p \in (0,1)$ and a fixed $t>0$, the expected time of the algorithm grows as a constant times $c\log(\log(n))$ as $n \to \infty$, where c depends upon p and t only. If time is equated with the number of beta random variates needed before halting, determine c.

5. THE LOGARITHMIC SERIES DISTRIBUTION.

5.1. Introduction.

A random variable X has the **logarithmic series distribution** with parameter $p \in (0,1)$ if

$$P(X=i) = p_i = \frac{a}{i}p^i \quad (i=1,2,\ldots),$$

where $a = -1/\log(1-p)$ is a normalization constant. In the tail, the probabilities decrease exponentially. Its generating function is

$$a \sum_{i=1}^{\infty} \frac{1}{i} p^i s^i = \frac{\log(1-ps)}{\log(1-p)} .$$

From this, one can easily find the mean $ap/(1-p)$ and second moment $ap/(1-p)^2$.

5.2. Generators.

The material in this section is based upon the fundamental work of Kemp (1981) on logarithmic series distributions. The problems with the logarithmic series distribution are best highlighted by noting that the obvious inversion and rejection methods are not uniformly fast.

If we were to use sequential search in the inversion method, using the recurrence relation

$$p_i = (1-\frac{1}{i})pp_{i-1} \quad (i \geq 2) ,$$

the inversion method could be implemented as follows:

Inversion by sequential search

[SET-UP]

Sum $\leftarrow -p/\log(1-p)$

[GENERATOR]

Generate a uniform [0,1] random variate U.

$X \leftarrow 1$

WHILE $U >$ Sum DO

  $U \leftarrow U -$ Sum

  $X \leftarrow X+1$

  Sum $\leftarrow$ Sum $\frac{p(X-1)}{X}$

RETURN X

The expected number of comparisons required is equal to the mean of the distribution, $ap/(1-p)$, and this quantity increases monotonically from 1 ($p \downarrow 0$) to ∞ ($p \uparrow \infty$). For $p < 0.95$, it is difficult to beat this simple algorithm in terms of expected time. Interestingly, if rejection from the geometric distribution $(1-p)p^i$ $(i \geq 1)$ is used, the expected number of geometric random variates required is again equal to the same mean. But because the geometric random variates themselves are rather costly, the sequential search method is to be preferred at this stage.

We can obtain a one-line generator based upon the following distributional property:

Theorem 5.1. (Kendall (1948), Kemp (1981))
Let U,V be iid uniform [0,1] random variables. Then

$$X \leftarrow \left\lfloor 1+\frac{\log(V)}{\log(1-(1-p)^U)} \right\rfloor$$

has the logarithmic series distribution with parameter p.

Proof of Theorem 5.1.

The logarithmic series distribution is the distribution of a geometric $(1-Y)$ random variate X (i.e. $P(X=i \mid Y)=Y(1-Y)^{i-1}$ $(i \geq 1)$), provided that Y has distribution function

$$F(y) = \int_0^y \frac{1}{(z-1)\log(1-p)} dz = \frac{\log(1-y)}{\log(1-p)} \quad (0 \leq y \leq p) .$$

This can be seen from the integral

$$\int_0^p \frac{s(1-y)}{(1-ys)(y-1)\log(1-p)} dy = \frac{\log(1-ps)}{\log(1-p)}$$

and from the fact that the generating function of a geometric $(1-Y)$ random variate is $\frac{s(1-Y)}{(1-Ys)}$. A random variable Y with distribution function F can be obtained by the inversion method as $Y \leftarrow 1-(1-p)^U$ where U is a uniform [0,1] random variable. ∎

Kemp (1981) has suggested two clever tricks for accelerating the algorithm suggested by Theorem 5.1. First, when $V > p$, the value $X \leftarrow 1$ is delivered because

$$V > p \geq 1-(1-p)^U .$$

For small p, the savings thus obtained are enormous. We summarize:

Kemp's generator with acceleration

[SET-UP]

$r \leftarrow \log(1-p)$

[GENERATOR]

$X \leftarrow 1$

Generate a uniform [0,1] random variate V.

IF $V \geq p$

    THEN RETURN X

    ELSE

        Generate a uniform [0,1] random variate U.

        RETURN $X \leftarrow \left\lfloor 1+\frac{\log(V)}{\log(1-e^{rU})} \right\rfloor$

Kemp's second trick involves taking care of the values 1 and 2 separately. He notes that $X=1$ if and only if $V \geq 1-e^{rU}$, and that $X \in \{1,2\}$ if and only if $V \geq (1-e^{rU})^2$ where r is as in the algorithm shown above. The algorithm incorporating this is given below.

Kemp's second accelerated generator

[SET-UP]

$r \leftarrow \log(1-p)$

[GENERATOR]

$X \leftarrow 1$

Generate a uniform [0,1] random variate V.

IF $V \geq p$

    THEN RETURN X

    ELSE

        Generate a uniform [0,1] random variate U.

        $q \leftarrow 1-e^{rU}$

        CASE

            $V \leq q^2$: RETURN $X \leftarrow \left\lfloor 1+\frac{\log(V)}{\log(q)} \right\rfloor$

            $q^2 < V \leq q$: RETURN $X \leftarrow 1$

            $V > q$: RETURN $X \leftarrow 2$

5.3. Exercises.

1. The following logarithmic series generator is based upon rejection from the geometric distribution:

Logarithmic series generator based upon rejection

REPEAT

Generate a uniform [0,1] random variate U and an exponential random variate E.

$X \leftarrow \left\lceil -\frac{E}{\log(p)} \right\rceil$

UNTIL $UX < 1$

RETURN X

Show that the expected number of exponential random variates needed is equal to the mean of the logarithmic series distribution, i.e. $-p/((1-p)\log(1-p))$. Show furthermore that this number increases monotonically to ∞ as $p \uparrow 1$.

2. **The generalized logarithmic series distribution.** Patel (1981) has proposed the following generalization of the logarithmic series distribution with parameter p:

$$p_i = \frac{p^i (1-p)^{bi-i} \Gamma(bi)}{-i \log(1-p) \Gamma(i) \Gamma(bi-i+1)} \quad (i \geq 1) .$$

Here $b \geq 1$ is a new parameter satisfying the inequality

$$0 < pb \left(\frac{b-bp}{b-1}\right)^{b-1} < 1 .$$

Suggest one or more efficient generators for this two-parameter family.

3. Consider the following discrete distribution:

$$p_i = \frac{1}{ci} \quad (1 \leq i \leq k) ,$$

where the integer k can be considered as a parameter, and c is a normalization constant. Show that the following bounded workspace algorithm generates random variates with this distribution:

REPEAT

  Generate iid uniform [0,1] random variates U,V.

  $Y \leftarrow (k+1)^U$

  $X \leftarrow \lfloor Y \rfloor$

UNTIL $2VX < Y$

RETURN X

Analyze the expected number of iterations as a function of k. Suggest at least one effective improvement.

6. THE ZIPF DISTRIBUTION.

6.1. A simple generator.

In linguistics and social sciences, the **Zipf distribution** is frequently used to model certain quantities. This distribution has one parameter $a > 1$, and is defined by the probabilities

$$p_i = \frac{1}{\zeta(a) i^a} \quad (i \geq 1)$$

where

$$\zeta(a) = \sum_{i=1}^{\infty} \frac{1}{i^a}$$

is the Riemann zeta function. Simple expressions for the zeta function are known in special cases. For example, when a is integer, then

$$\zeta(2a) = \frac{2^{2a-1}\pi^{2a}}{(2a)!} B_a$$

where B_a is the a-th Bernoulli number (Titchmarsh, 1951, p. 20). Thus, for $a=2,4,6$ we obtain the probability vectors $\{6/(\pi i)^2\}$,$\{90/(\pi i)^4\}$ and $\{945/(\pi i)^6\}$ respectively.

To generate a random Zipf variate in uniformly bounded expected time, we propose the rejection method. Consider for example the distribution of the random variable $Y \leftarrow \left\lfloor U^{-1/(a-1)} \right\rfloor$ where U is uniformly distributed on [0,1]:

$$P(Y=i) = \frac{1}{(i+1)^{a-1}}((1+\frac{1}{i})^{a-1}-1) \quad (i \geq 1) .$$

This distribution is a good candidate because the probabilities vary as $(a-1)i^{-a}$ as $i \to \infty$. For the sake of simplicity, let us define $q_i = P(Y=i)$. First, we note that the rejection constant c is

$$c = \sup_{i \geq 1} \frac{p_i}{q_i} = \frac{p_1}{q_1} = \frac{2^{a-1}}{\zeta(a)(2^{a-1}-1)} .$$

Hence, the following rejection algorithm can be used:

A Zipf generator based upon rejection

[SET-UP]

$b \leftarrow 2^{a-1}$

[GENERATOR]

REPEAT

Generate iid uniform [0,1] random variates U, V.

$X \leftarrow \left\lfloor U^{-\frac{1}{a-1}} \right\rfloor$

$T \leftarrow (1+\frac{1}{X})^{a-1}$

UNTIL $VX\frac{T-1}{b-1} \leq \frac{T}{b}$

RETURN X

Lemma 6.1.

The rejection constant c in the rejection algorithm shown above satisfies the following properties:

A. $\sup_{a \geq 2} c \leq \frac{12}{\pi^2}$.

B. $\sup_{1<a \leq 2} c \leq \frac{2}{\log(2)}$.

C. $\lim_{a \to \infty} c = 1$.

D. $\lim_{a \downarrow 1} c = \frac{1}{\log(2)}$.

Proof of Lemma 6.1.

Part A follows from

$$c \le \frac{2^{a-1}}{2^{a-1}-1}\frac{6}{\pi^2} \le \frac{12}{\pi^2} .$$

Part B follows from

$$c \le \frac{2^{a-1}}{(2^{a-1}-1)\int_1^\infty x^{-a}\,dx} = \frac{(a-1)2^{a-1}}{2^{a-1}-1}$$

$$\le \frac{(a-1)2^{a-1}}{(a-1)\log(2)} = \frac{2^{a-1}}{\log(2)} \le \frac{2}{\log(2)} .$$

Part C follows by observing that $\zeta(a)\to 1$ as $a \uparrow \infty$. Finally, part D uses the fact that $\zeta(a) \sim \frac{1}{a-1}$ as $a \downarrow 1$ (in fact, $\zeta(a)-\frac{1}{a-1}\to\gamma$, Euler's constant (Whittaker and Watson, 1927, p. 271). ■

6.2. The Planck distribution.

The Planck distribution is a two-parameter distribution with density

$$f(x) = \frac{b^{a+1}}{\Gamma(a+1)\zeta(a+1)}\frac{x^a}{e^{bx}-1} \quad (x>0) .$$

Here $a>0$ is a shape parameter and $b>0$ is a scale parameter (Johnson and Kotz, 1970). The density f can be written as a mixture:

$$f(x) = \sum_{i=1}^{\infty} \frac{1}{i^{a+1}\zeta(a+1)} \frac{x^a e^{-ibx}(ib)^{a+1}}{\Gamma(a+1)} .$$

In view of this, the following algorithm can be used to generate a random variate with the Planck distribution.

Planck random variate generator

Generate a gamma $(a+1)$ random variate G.

Generate a Zipf $(a+1)$ random variate Z.

RETURN $X \leftarrow \frac{G}{bZ}$.

6.3. The Yule distribution.

Simon (1954,1960) has suggested the Yule distribution as a better approximation of word frequencies than the Zipf distribution. He defined the discrete distribution by the probabilities

$$p_i = c(a)\int_0^1 (1-u)^{i-1}u^{a-1}du \qquad (i \geq 1),$$

where $c(a)$ is a normalization constant and $a > 1$ is a parameter. Using the fact that this is a mixture of the geometric distribution with parameter $e^{-Y/(a-1)}$ where Y is exponentially distributed, we conclude that a random variate X with the Yule distribution can be generated as

$$X \leftarrow \left\lceil -\frac{E}{\log(1-e^{-\frac{E^*}{a-1}})} \right\rceil ,$$

where E, E^* are iid exponential random variates.

6.4. Exercises.

1. **The digamma and trigamma distributions.** Sibuya (1979) introduced two distributions, termed the digamma and trigamma distributions. The digamma distribution has two parameters, a, c satisfying $c > 0, a > -1$, $a+c > 0$. It is defined by

$$p_i = \frac{1}{\psi(a+c)-\psi(c)} \frac{a(a+1)\cdots(a+i-1)}{i(a+c)(a+c+1)\cdots(a+c+i-1)} \qquad (i \geq 1).$$

Here ψ is the derivative of the log gamma function, i.e. $\psi = \Gamma'/\Gamma$. When we let $a \downarrow 0$, the trigamma distribution with parameter $c > 0$ is obtained:

$$p_i = \frac{1}{\psi'(c)} \frac{(i-1)!}{ic(c+1)\cdots(c+i-1)} \qquad (i \geq 1).$$

For $c = 1$ this is a zeta distribution. Discuss random variate generation for this family of distributions, and provide a uniformly fast rejection algorithm.