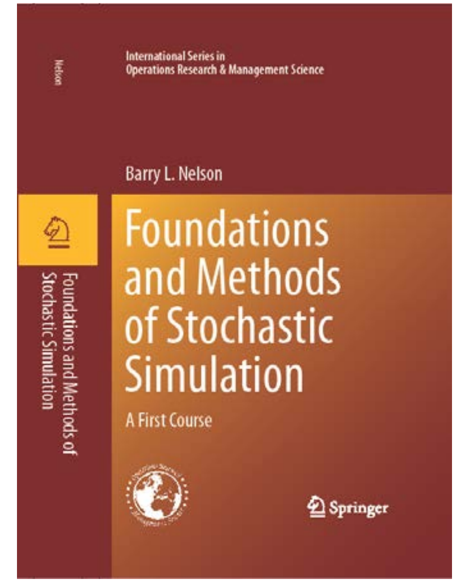


Chapter 6: Simulation Input

Northwestern University
July 2017

Simulation input

Input modeling: Selecting (and perhaps fitting) the probability models that represent the uncertainty.

- Inference approach
- Matching approach
- Special topic: Nonstationary arrival processes

Random-variate generation: Representing the inputs as transformations of i.i.d. $U(0,1)$ random variables.

Inversion, rejection and particular properties.

Random-number generation: Producing a good approximation to realizations of i.i.d. $U(0,1)$ random variates.

Combined generators for exceptionally long period.

An input modeling story

Recall the hospital kiosk which we modeled/simulated as a single-server queue. There are two input processes:

Patient arrival process: The hospital has records from which we can extract arrival counts over time intervals (why might they not have actual arrival *times*?).

Process physics suggests a Poisson arrival process.

Service time at the kiosk: The vendor has conducted a trial time study that provides data.

With no obvious process physics, we use distribution fitting software that combines MLEs with some measure of "best fit."

What can go wrong?

Poisson arrival process

The MLE for the arrival rate λ is $\widehat{\lambda} = N(T)/T$ assuming a *stationary* Poisson process. How could we detect nonstationarity?

Suppose we believe daily stationarity and do a goodness-of-fit test for being Poisson. Why might it reject?

1. The distribution of the data is substantially different from Poisson; e.g., includes scheduled arrivals.
2. The arrival rate might differ by day of the week, even if stationary Poisson each day.
3. We have so much data that the test has power against even meaningless deviations from Poisson.

Service process

"Best fit" usually means fit a large collection of distributions using MLEs and choose the one with the best summary statistic (e.g.,smallest test score or largest p-value).

Concerns?

- Are the data actually relevant? Similar to real patients? Will there be learning? Will there be different types of patients by time of day?
- There is no statistical meaning to selecting the distribution having the smallest test score from among an arbitrary collection. This is a heuristic.

View through the queue

Recall that for the $M/G/1$ queue

$$\mathrm{E}(Y) = \frac{\lambda(\sigma^2 + \tau^2)}{2(1 - \lambda\tau)}$$

- Notice that the service-time distribution does not matter *as long as it gets the mean and variance right*. Fortunately this is often true.

- Although the service-time distribution is not critical, the arrival process being Poisson is. So we want good process physics to force a distribution.

- The parameter estimates will almost certainly be wrong: $\widehat{\lambda} \neq \lambda$, $\widehat{\tau} \neq \tau$, and $\widehat{\sigma}^2 \neq \sigma^2$. In fact, we could have $\widehat{\lambda}\widehat{\tau} > 1$.

Univariate input models

$X_1, X_2, \ldots, X_m$ i.i.d real-world input data with a stable (unchanging) distribution F_X.

$F(x; \boldsymbol{\theta})$ is a parametric family with parameter vector $\boldsymbol{\theta}$ to model F_X; e.g., $\boldsymbol{\theta} = (\mu, \sigma^2)$.

"Fitting:" Use some estimator $\widehat{\boldsymbol{\theta}} = \widehat{\boldsymbol{\theta}}(X_1, X_2, \ldots, X_m)$.

$\widehat{X} \sim F(x; \widehat{\boldsymbol{\theta}})$, a simulated value from the fitted distribution (what we actually use in the simulation).

Inference vs. Matching

Inference: Treats input modeling as statistical inference about the true distribution F_X, and employs methods that have good statistical properties.

MLEs for $\boldsymbol{\theta}$ and testing H_0: $F(x; \widehat{\boldsymbol{\theta}}) = F_X$ arise here.

Matching: Get properties of $\widehat{X}$ to closely match those of $X_1, X_2, \ldots, X_m$; e.g.,

$$\mathrm{E}\left(\widehat{X}\,|X_1, X_2, \ldots, X_m\right) = \bar{X}.$$

Notice the expectation is with respect to $F(x; \widehat{\boldsymbol{\theta}})$.

Depending on the application, other properties will be important.

Inference approach

The inference paradigm is justified by having strong process physics that supports the choice of parametric family.

Many (perhaps most) probability distributions are derived from some basic process physics (e.g., summing, multiplication, minimum) leading to the distribution.

These are beautifully captured in a series of books by Johnson, Kotz and coauthors.

See also `www.math.wm.edu/~leemis/chart/UDR/UDR.html`

Example: Weibull vs. gamma

Both are used in reliability. Both can have increasing, decreasing or constant hazard: $h(t) = f_T(t)/(1 - F_T(t))$.

So how could we choose?

Look at the tails of the density functions with common scale parameter $\beta = 1$:

Weibull	gamma
$\alpha t^{\alpha-1} \mathrm{e}^{-t^{\alpha}}$	$\Gamma(\alpha)^{-1} t^{\alpha-1} \mathrm{e}^{-t}$

Gamma eventually has exponential tail (constant failure rate), but Weibull will not unless $\alpha = 1$ which makes it exponential *everywhere*.

Matching approach

The basic idea is to choose parameters for $F(x; \boldsymbol{\theta})$ to match sample properties of the real-world data $X_1, X_2, \ldots, X_m$.

The most common choice is to match central *moments*:

$\mu_X = \mathrm{E}(X)$	mean
$\sigma_X^2 = \mathrm{E}[(X - \mu_X)^2]$	variance
$\alpha_3 = \mathrm{E}[(X - \mu_X)^3]/\sigma_X^3$	skewness
$\alpha_4 = \mathrm{E}[(X - \mu_X)^4]/\sigma_X^4$	kurtosis

Symmetric distributions have $\alpha_3 = 0$.

The normal distribution has kurtosis $\alpha_4 = 3$.

$\alpha_4 - 3$ is called the excess kurtosis.

If $\alpha_3 < \infty$ and $\alpha_4 < \infty$ then $\alpha_4 > 1 + \alpha_3^2$.

Where the action is

Suppose that the first four central moments of X, $(\mu_X, \sigma_X^2, \alpha_3, \alpha_4)$, exist. Let

$$X' = \mu + \sigma \left(\frac{X - \mu_X}{\sigma_X} \right).$$

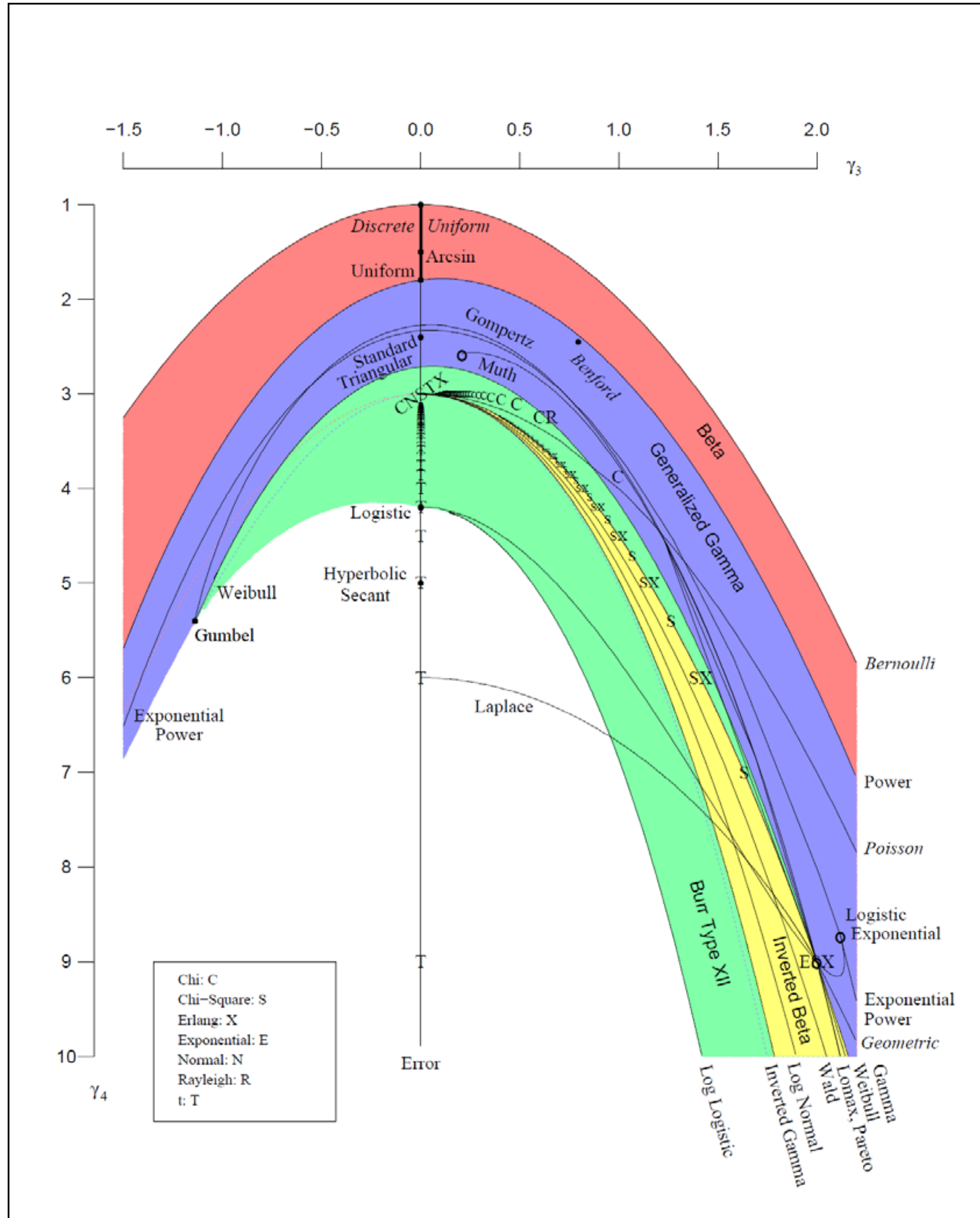
Then X' has mean μ and variance σ^2, but the *same* skewness and kurtosis as X.

So matching mean and variance is easy; the challenge is in matching α_3 and α_4.

There are distributions (e.g., Johnson, GLD) designed to cover all or nearly all of the (α_3^2, α_4) plane.

Skewness vs. Kurtosis for common univariate distributions

Vargo et al. 2010. *Journal of Quality Technology* **42** (3): 1-11.



http://www.math.wm.edu/~leemis/2010jqt.Plot4.pdf

Moment matching approach

1. Compute the sample standardized central moments of the input data:

$$
\begin{aligned}
\bar{X} &= \frac{1}{m}\sum_{i=1}^{m} X_i \\
\widehat{\sigma}^2 &= \frac{1}{m}\sum_{i=1}^{m}(X_i - \bar{X})^2 \\
\widehat{\alpha}_3 &= \frac{1}{m}\sum_{i=1}^{m}(X_i - \bar{X})^3/\widehat{\sigma}^3 \\
\widehat{\alpha}_4 &= \frac{1}{m}\sum_{i=1}^{m}(X_i - \bar{X})^4/\widehat{\sigma}^4.
\end{aligned}
$$

2. Express central moments of $F(\cdot;\boldsymbol{\theta})$ as functions of $\boldsymbol{\theta}$: $\mu(\boldsymbol{\theta}), \sigma^2(\boldsymbol{\theta}), \alpha_3(\boldsymbol{\theta})$ and $\alpha_4(\boldsymbol{\theta})$.

3. Match the moments by solving the system of equations for $\boldsymbol{\theta}$:

$$\begin{aligned} \mu(\boldsymbol{\theta}) &= \bar{X} \\ \sigma^2(\boldsymbol{\theta}) &= \widehat{\sigma}^2 \\ \alpha_3(\boldsymbol{\theta}) &= \widehat{\alpha}_3 \\ \alpha_4(\boldsymbol{\theta}) &= \widehat{\alpha}_4 \end{aligned}$$

Example: The $\text{gamma}(\alpha, \beta)$ distribution has $\mu(\alpha, \beta) = \alpha\beta$ and $\sigma^2(\alpha, \beta) = \alpha\beta^2$, so solve $\widehat{\alpha}\widehat{\beta} = \bar{X}$ and $\widehat{\alpha}\widehat{\beta}^2 = \widehat{\sigma}^2$.

FYI: $\alpha_3 = 2/\sqrt{\alpha}$ and $\alpha_4 = 3 + 6/\alpha$ for the gamma.

Other things to match

Let $\widehat{F}$ be the empirical cdf (ecdf) of the data

$$\widehat{F}(x) = \frac{1}{m}\sum_{i=1}^{m} I(X_i \le x)$$

and let $X_{(1)} \le X_{(2)} \le \cdots \le X_{(m)}$ be the order statistics (sorted data).

Then we could try to pick $\widehat{\boldsymbol{\theta}}$ so that

$$F(X_{(i)};\widehat{\boldsymbol{\theta}}) \approx \widehat{F}(X_{(i)}) = \frac{i}{m}$$

The book describes a least squares approach for the GLD.

Empirical distributions

Using the ecdf $\widehat{F}$ is easy (is this inverse cdf?):

1. Generate $U \sim U(0,1)$
2. Set $i = \lceil mU \rceil$
3. Return $\widehat{X} = X_{(i)}$

The ecdf is unbiased for F_X and matches sample properties; e.g., for $\widehat{X} \sim \widehat{F}$

$$\mathrm{E}\left(\widehat{X} \,\middle|\, X_1, \ldots, X_m\right) = \sum_{i=1}^{m} X_{(i)} \frac{1}{m} = \bar{X}.$$

Interpolated ecdf

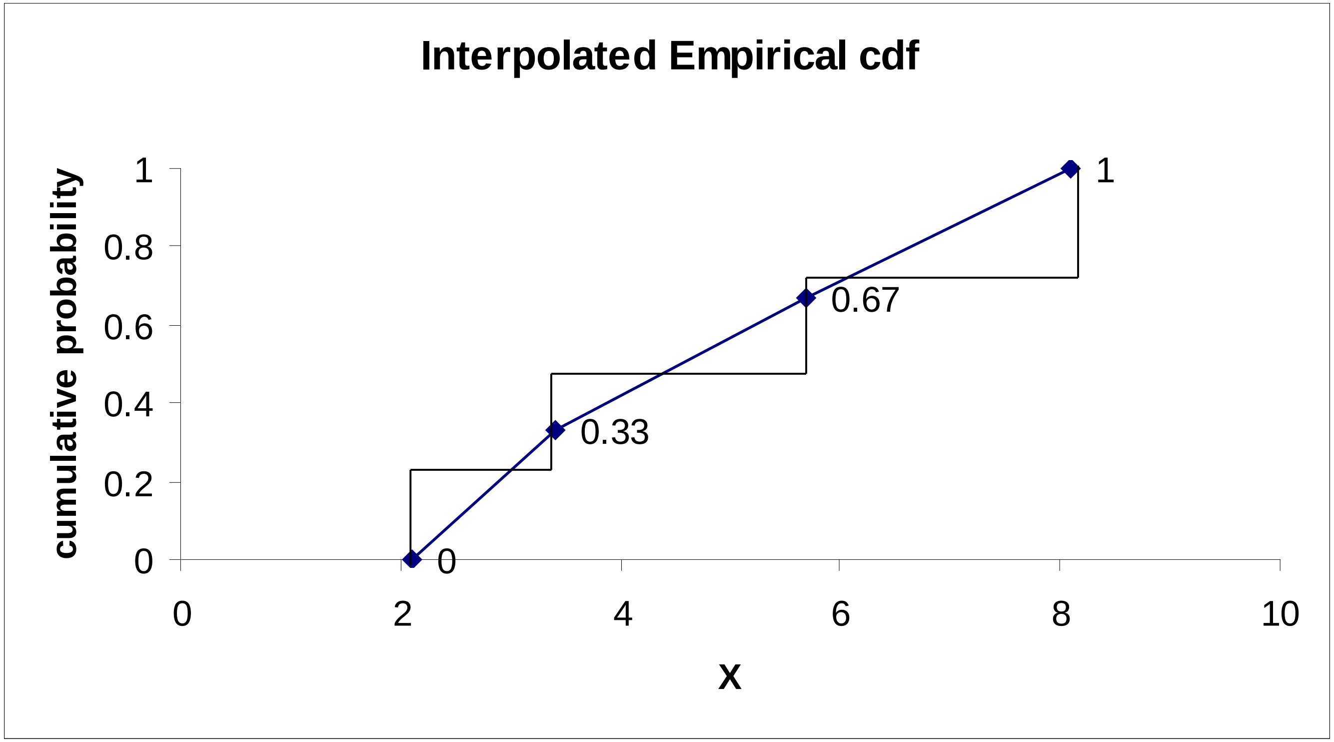
We can fill in the gaps with linear interpolation:

$$\tilde{F}(x) = \begin{cases} 0, & x < X_{(1)} \\ \dfrac{i-1}{m-1} + \dfrac{x - X_{(i)}}{(m-1)(X_{(i+1)} - X_{(i)})}, & X_{(i)} \leq x < X_{(i+1)} \\ 1, & x \geq X_{(m)}. \end{cases}$$

Inversion:

1. Generate $U \sim U(0,1)$
2. Set $i = \lceil (m-1)U \rceil$
3. Return $\tilde{X} = X_{(i)} + (m-1)(X_{(i+1)} - X_{(i)}) \left(U - \dfrac{i-1}{m-1} \right)$

Empirical cdf and interpolated empirical cdf for data points X = {5.7, 2.1, 3.4, 8.1}



Notice that the interpolated ecdf fills in the gaps between observed values, but the domain is still limited to the smallest and largest observed values (no tails).

Properties of the interpolated ecdf

$\tilde{F}$ is no longer unbiased, nor does it exactly match sample properties. But it is consistent. For any fixed x, let

$$\bar{F}(x) = \frac{1}{m-1}\sum_{i=1}^{m} I(X_i \le x) = \frac{m}{m-1}\widehat{F}(x).$$

Then $\bar{F}(x) \xrightarrow{a.s.} F_X(x)$ because $m/(m-1) \to 1$. Also for $X_{(i)} \le x < X_{(i+1)}$ we have

$$0 \le \frac{x - X_{(i)}}{(m-1)(X_{(i+1)} - X_{(i)})} < \frac{1}{m-1}.$$

Therefore, $\bar{F}(x) - \dfrac{1}{m-1} \le \tilde{F}(x) < \bar{F}(x)$.

Since both the lower and upper bounds on $\tilde{F}(x)$ converge a.s. to $F_X(x)$, so does $\tilde{F}(x)$.

Input modeling without data

1. Use the process physics to suggest a good choice, then translate subjective information people can give you into parameters.

   Example: If given the 30th and 80th percentiles $x_{0.3}$ and $x_{0.8}$ of a normal distribution, then solve

$$\begin{aligned} \mu + z_{0.3}\sigma &= x_{0.3} \\ \mu + z_{0.8}\sigma &= x_{0.8} \end{aligned}$$

2. Use a distribution with intuitive parameters; e.g., the triangular distribution uses minimum, maximum and most likely values.

Nonstationary arrival processes

Arrival processes are fundamental to many OR simulations. In addition to scheduled arrivals (how would you simulate?), the most common model is random *interarrival* times.

Our background in queueing theory tends to make us think about *stationary renewal arrival processes*, meaning i.i.d. interarrival times.

When the arrival rate changes dramatically over time, approximating it as stationary (e.g., average rate, max rate) may miss a critical aspect of system behavior.

Renewal arrivals

Interarrival times: $\tilde{A}_1, \tilde{A}_2, \ldots$ i.i.d. nonnegative with distribution G (finite mean, variance and has density)

Arrival times: $\tilde{S}_n = \begin{cases} 0, & n = 0 \\ \sum_{i=1}^{n} \tilde{A}_i = \tilde{S}_{n-1} + \tilde{A}_n, & n = 1, 2, \ldots \end{cases}$

Arrival-counting process: $\tilde{N}(t) = \max\{n \geq 0 : \tilde{S}_n \leq t\}$

Arrival rate: $\lim_{t\to\infty} \dfrac{\mathrm{E}(\tilde{N}(t))}{t} = \tilde{\lambda} = \dfrac{1}{\mathrm{E}(\tilde{A}_i)}$

Equilibrium renewal: If $\tilde{A}_1 \sim G_e(t) = \tilde{\lambda} \int_0^t (1 - G(s))\, ds$

then $\dfrac{\mathrm{E}(\tilde{N}(t))}{t} = \tilde{\lambda}$ or $\mathrm{E}(\tilde{N}(t)) = \tilde{\lambda} t$, $\forall t$.

Nonstationary arrivals

To generalize to a time varying rate, let $N(t)$ be an arrival-counting process (possibly nonstationary) and

$$\Lambda(t) = \mathrm{E}(N(t))$$

the expected number of arrivals by time t (the "integrated rate function").

Then define the *arrival rate* to be

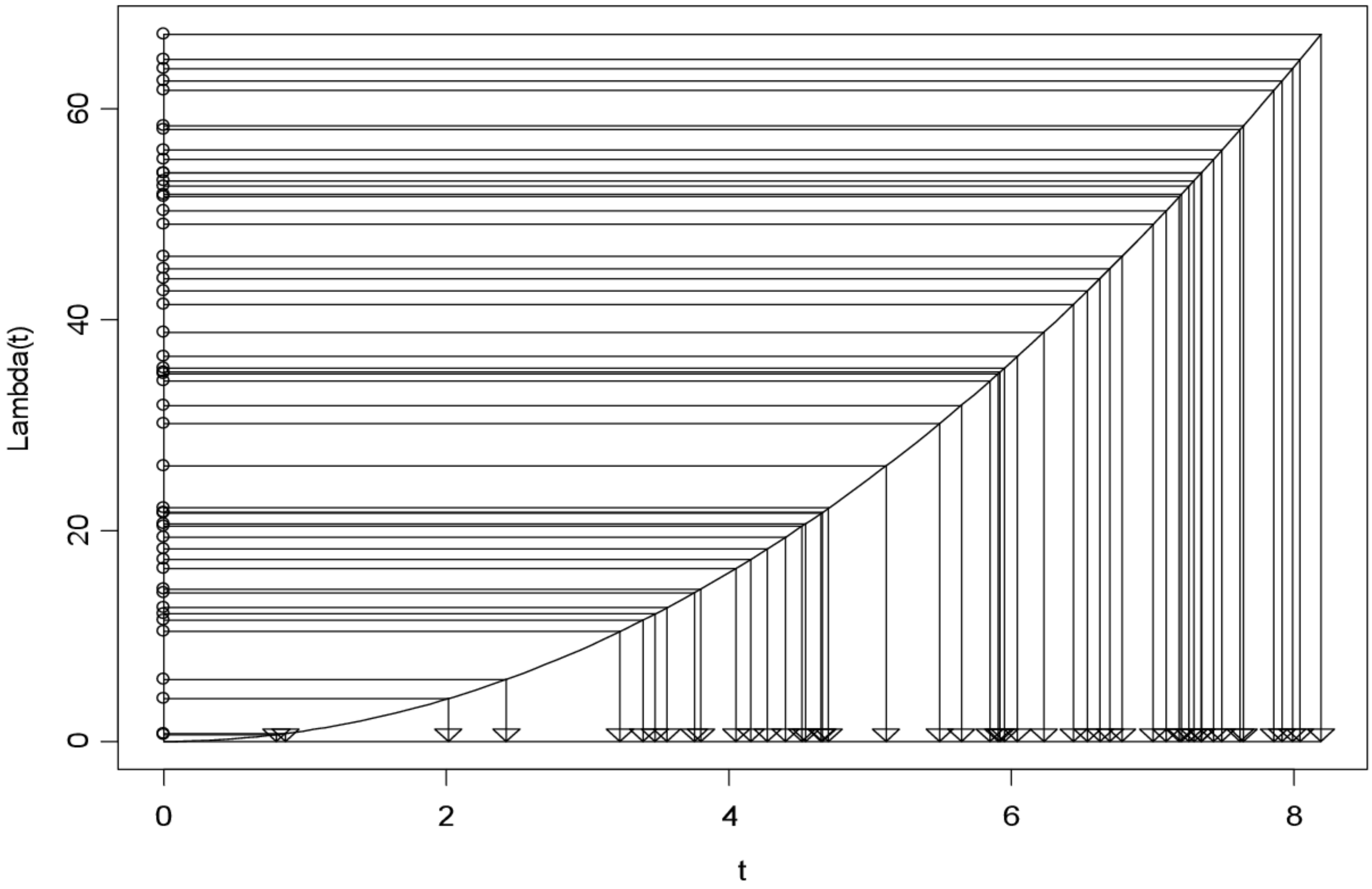
$$\lambda(t) = \frac{d}{dt}\Lambda(t)$$

To get arrival rate $\lambda(t)$, we transform equilibrium renewal arrivals with rate $\tilde{\lambda}$.

Inverting $\Lambda(t)$

1. Set index $n = 1$ and $\tilde{S}_0 = 0$
2. Generate $\tilde{A}_n$ with rate $\tilde{\lambda} = 1$
3. Let
   (a) $\tilde{S}_n = \tilde{S}_{n-1} + \tilde{A}_n$
   (b) $S_n = \Lambda^{-1}(\tilde{S}_n)$
   (c) $A_n = S_n - S_{n-1}$
4. $n = n + 1$
5. Go to Step 2

$$\lambda(t) = 2t \rightarrow \Lambda(t) = t^2 \rightarrow \Lambda^{-1}(s) = \sqrt{s}$$



Proof

Remember that $E(\tilde{N}(s)) = 1 \cdot s$

$$\begin{aligned} \mathrm{E}(N(t)) &= \mathrm{E}\left[\mathrm{E}\left(N(t) \middle| \tilde{N}(\Lambda(t))\right)\right] \\ &= \mathrm{E}\left[\tilde{N}(\Lambda(t))\right] \\ &= 1 \cdot \Lambda(t) = \Lambda(t) \end{aligned}$$

This method is ideal when $\Lambda(t)$ is easily invertible.

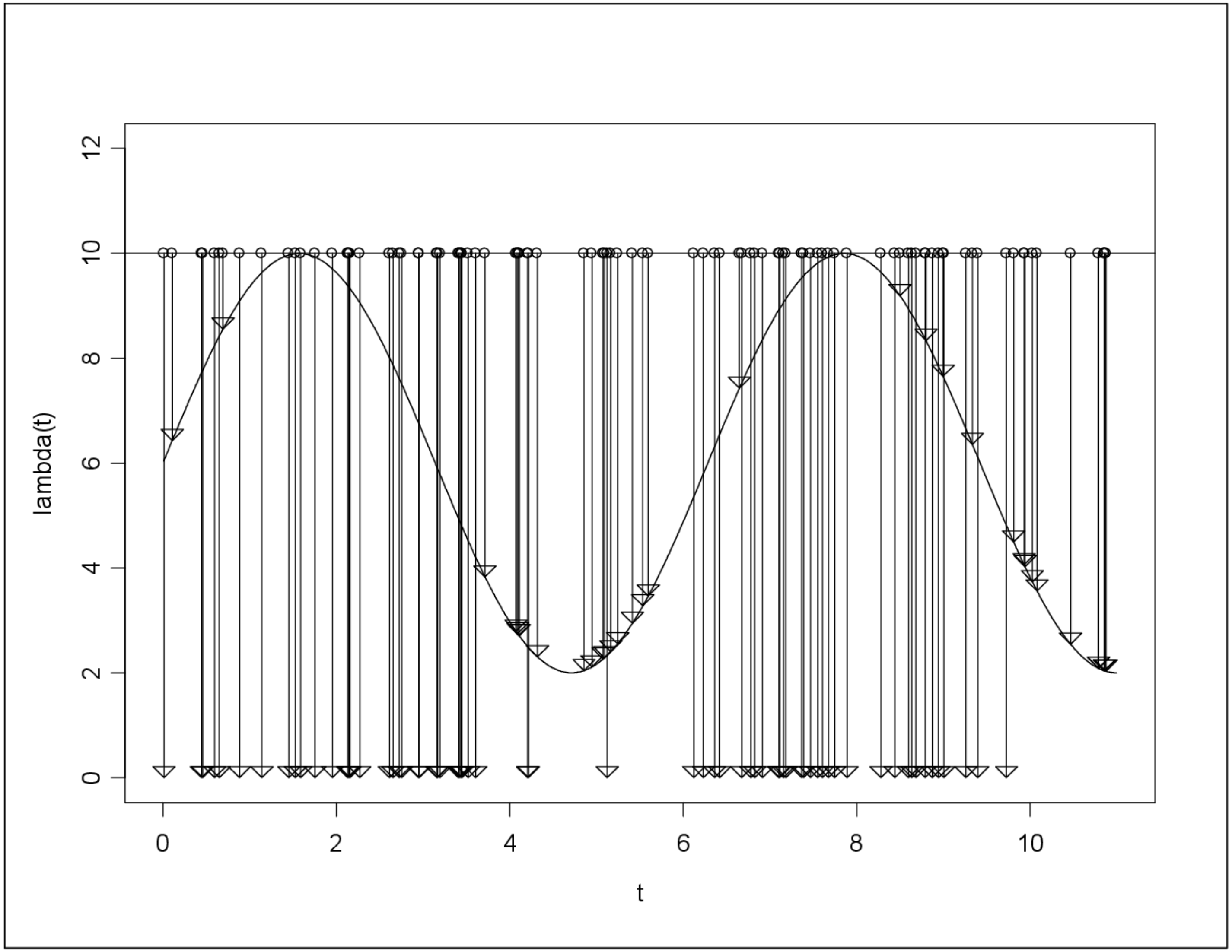
Any equilibrium renewal arrival process with rate 1 achieves the desired $\Lambda(t)$, but the specific choice affects other aspects of the arrivals (e.g., variability).

Thinning

1. Set indices $n = 1$ and $k = 1$ and $\tilde{S}_0 = 0$
2. Generate $\tilde{A}_n$ with rate $\tilde{\lambda} = \max_t \lambda(t)$, and let $\tilde{S}_n = \tilde{S}_{n-1} + \tilde{A}_n$
3. Generate $U \sim U(0, 1)$
4. If $U \leq \lambda(\tilde{S}_n)/\tilde{\lambda}$ then
   (a) $S_k = \tilde{S}_n$
   (b) $A_k = S_k - S_{k-1}$
   (c) $k = k + 1$

   Endif
5. $n = n + 1$
6. Go to Step 2

$\lambda(t) = 6 + 4\sin(t)$



Thinning with exponential base

1. Set indices $n = 1$ and $k = 1$ and $\tilde{S}_0 = 0$
2. Generate $V \sim U(0,1)$, set $\tilde{A}_n = -\ln(1 - V) \cdot 10$, and let $\tilde{S}_n = \tilde{S}_{n-1} + \tilde{A}_n$
3. Generate $U \sim U(0,1)$
4. If $U \leq (6 + 4\sin(\tilde{S}_n))/10$ then
   (a) $S_k = \tilde{S}_n$
   (b) $A_k = S_k - S_{k-1}$
   (c) $k = k + 1$

   Endif
5. $n = n + 1$
6. Go to Step 2

Exponential interarrival time

When $G(t) = 1 - e^{-\tilde{\lambda} t}$ then...

1. We call it a nonstationary Poisson process.
2. $G_e(t) = G(t)$ (memoryless property).
3. Inversion and thinning are probabilistically equivalent.
4. $\dfrac{\mathrm{Var}\,(N(t))}{\mathrm{E}\,(N(t))} = 1$ for all $t \geq 0$.

   This contrasts with general inversion for which

$$\frac{\mathrm{Var}\,(N(t))}{\mathrm{E}\,(N(t))} \approx \sigma_A^2 \text{ for large } t$$

Fitting $\lambda(t)$ or $\Lambda(t)$

A lot of interesting work has been done on fitting parametric arrival rate models; e.g.,

$$\lambda(t) = \exp\left\{\sum_{i=0}^{m} \alpha_i t^i + \beta \sin(\omega t + \phi)\right\}$$

If we stay nonparametric, then typically...

- Fit $\Lambda(t)$ is we have actual arrival times.
- Fit a piecewise-constant $\lambda(t)$ if we have only counts over intervals.

Fitting $\Lambda(t)$

Data

$$\{T_{ij}; i = 1, 2, \ldots, C_j(T)\},\ j = 1, 2, \ldots, k$$

where $C_j(t)$ is the cumulative number of arrivals by time t on the jth realization, $0 \leq t \leq T$.

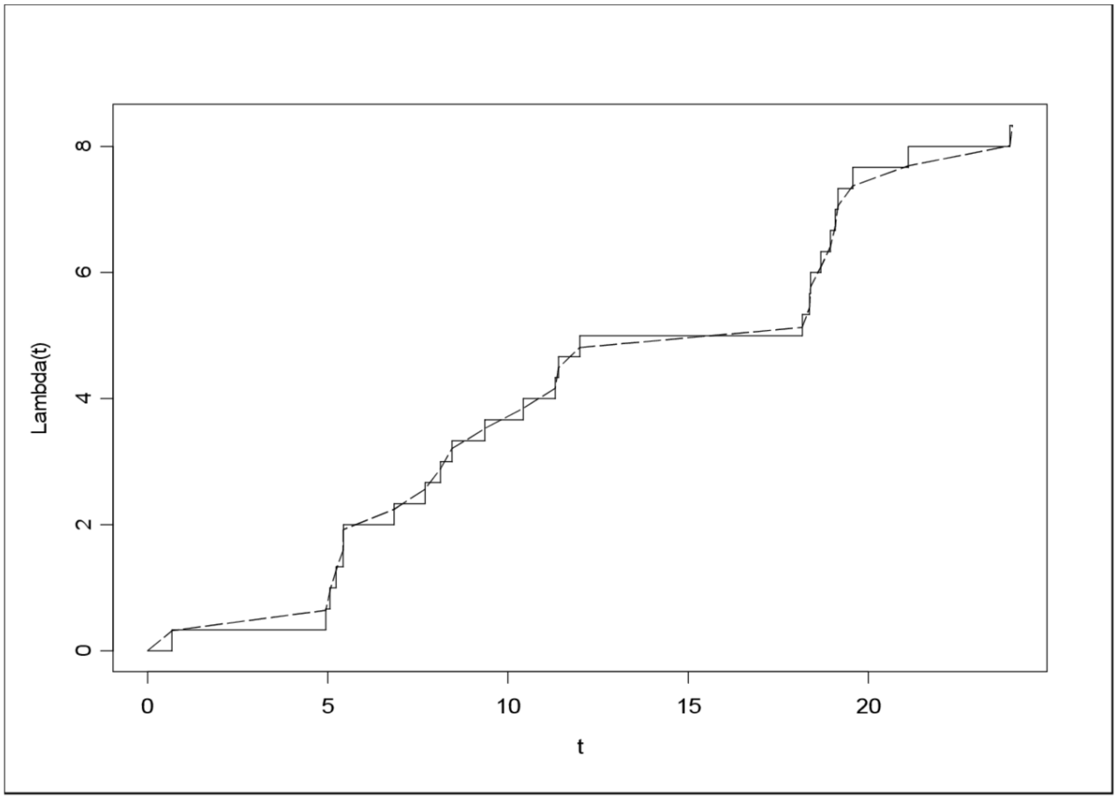
Example: Arrivals of emergency calls to 911 for a $T = 24$ hour period over $k = 3$ Mondays.

Merge these into a single data set of $C = \sum_{j=1}^{k} C_j(T)$ arrivals and sort

$$0 = T_{(0)} < T_{(1)} \leq T_{(2)} \leq \cdots \leq T_{(C)} < T_{(C+1)} = T$$

Linearly interpolated $\Lambda(t)$

$$\widehat{\Lambda}(t) = \left(\frac{C}{C+1}\right)\left\{\frac{i}{k} + \frac{1}{k}\left(\frac{t - T_{(i)}}{T_{(i+1)} - T_{(i)}}\right)\right\} \text{ when } T_{(i)} < t \le T_{(i+1)}$$



Emergency call data for $k = 3$ Mondays over $T = 24$ hours.

Jumps $(C/(C+1))(1/k)$ at each arrival time.

$\widehat{\Lambda}(t) \xrightarrow{a.s.} \Lambda(t)$ as $k \to \infty$.

Is easily simulated using inversion.

Algorithm for inversion of $\Lambda(t)$

1. Set $n = 1$ and $S_0 = 0$
2. Generate $\tilde{S}_1 \sim G_e$
3. While $\tilde{S}_n \leq C/k$ do
   (a) $m = \left\lfloor \left(\frac{C+1}{C}\right) k\tilde{S}_n \right\rfloor$
   (b) $S_n = T_{(m)} + \left(T_{(m+1)} - T_{(m)}\right)\left(\left(\frac{C+1}{C}\right) k\tilde{S}_n - m\right)$
   (c) $A_n = S_n - S_{n-1}$
   (d) $n = n + 1$
   (e) Generate $\tilde{A}_n \sim G$
   (f) $\tilde{S}_n = \tilde{S}_{n-1} + \tilde{A}_n$

   Loop

Fitting piecewise constant *λ(t)*

Now assume arrival rate is constant over intervals of length $\delta > 0$ for which we have arrival counts.

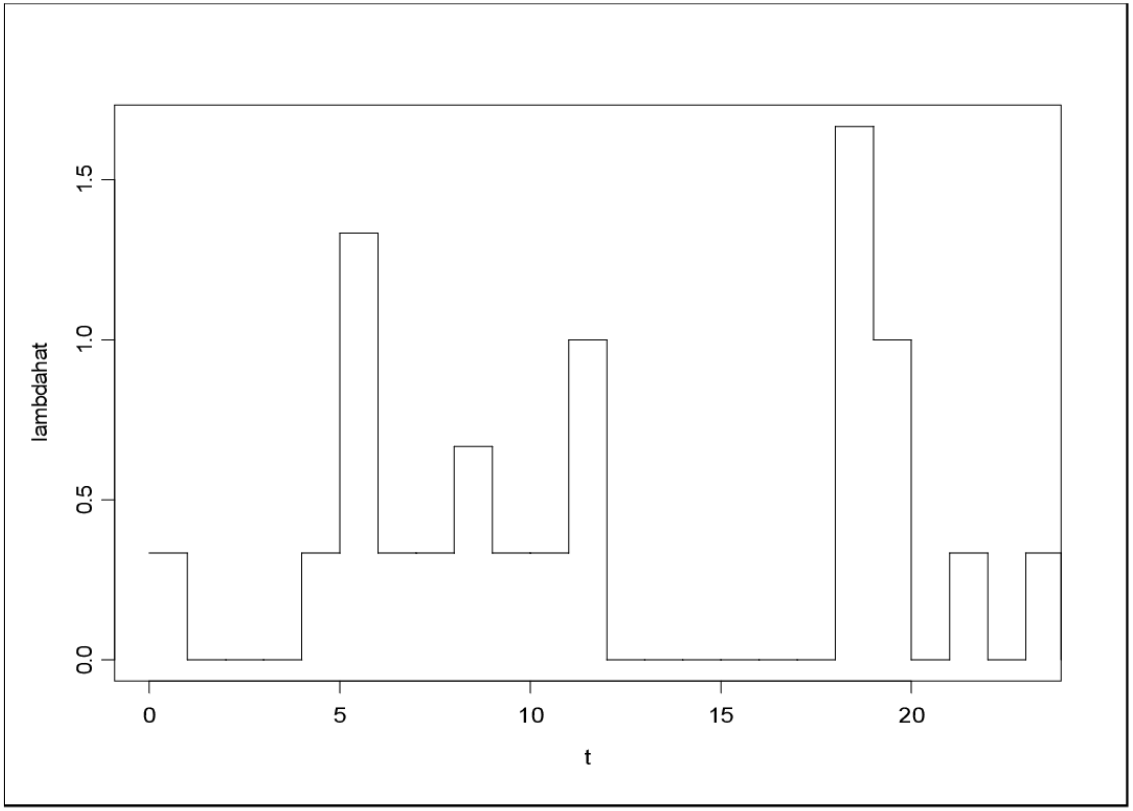
The arrival rate for an interval is just the average number of arrivals from the k realizations that fell in that interval per unit time:

$$\widehat{\lambda}(t) = \frac{1}{k\delta} \sum_{j=1}^{k} \left[\underbrace{C_j\left(\ell(t+\delta)\right) - C_j\left(\ell(t)\right)} \right]$$

where $\ell(t) = \lfloor t/\delta \rfloor \delta$.

Count of the number of arrivals on realization j for $i\delta < t \leq (i+1)\delta$ when $i = \lfloor t/\delta \rfloor$.

Piecewise constant $\lambda(t)$



Emergency call data for $k = 3$ Mondays with $\delta = 1$ hour.

Sparse arrivals can lead to $\widehat{\lambda}(t) = 0$ if δ too small.

δ too large can miss critical behavior.

Sensible when only have counts over fixed intervals; then use the natural δ.

Generating random variates

We know **inversion** is general. Even without a closed-form inverse, we can treat it as a root-finding problem:

$$\text{solve for } X\text{: } U = F_X(X) = \int_{-\infty}^{X} f_X(x)\,dx$$

Or use approximate inverses, such as

$$F_Z^{-1}(U) \approx \frac{U^{0.1349} + (1-U)^{0.1349}}{0.1975}$$

the GLD approximation to the standard normal inverse cdf which has about 1 decimal place accuracy.

Plus there are tricks to make inversion faster for discrete distributions.

Rejection

Applies when we can express

$$\Pr\{X \leq x\} = \Pr\{V \leq x | \mathcal{A}\}$$

where V is easy to generate and $\mathcal{A}$ is some event.

1. Generate V
2. If $\mathcal{A}$ occurs, then return $X = V$
   Otherwise, reject V and go to Step 1

Example: Generate $\{1, 2, 3, 4, 5\}$ equally likely by rolling a die but rejecting 6's.

An approach for $f_X(x)$ a density

Suppose $m(x) \geq f_X(x)$ for all x ("majorizes"). Then a density with the same shape is

$$g(x) = \frac{m(x)}{\int_{-\infty}^{\infty} m(y)\, dy} = \frac{m(x)}{c}$$

Rejection:

1. Generate $V \sim g$ (needs to be easy)
2. Generate $U \sim U(0,1)$
3. If $U \leq f_X(V)/m(V)$ then return $X = V$
   Otherwise go to Step 1

Intuition

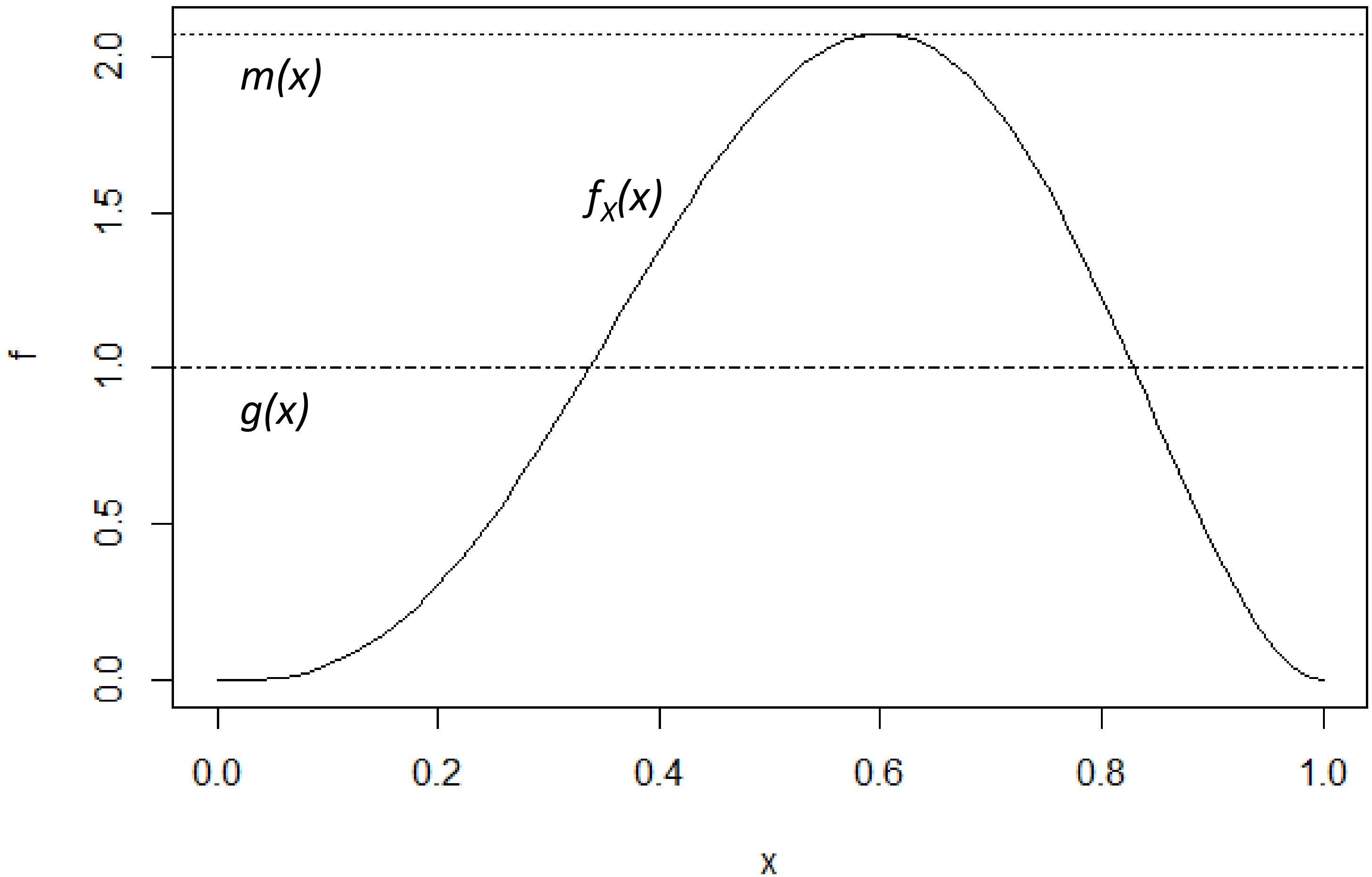
Notice that if $m(x) = f_X(x)$, then $g(x) = f_X(x)$ and we accept every time, as we should.

The density g redistributes the probability, making some x's more likely than f_X does, and others less likely.

Since $g \propto m$, where g makes x's more likely than f_X, then $m(x)$ will be big, $f_X(x)/m(x)$ will be small and we tend to reject.

Similarly where g makes x's less likely than f_X, then $m(x)$ will be small, $f_X(x)/m(x)$ will be big and we tend to accept.

Beta majorized by uniform



Beta distribution example

The beta density $f_X(x) = x^{\alpha_1 - 1}(1-x)^{\alpha_2 - 1}/\mathrm{B}(\alpha_1, \alpha_2)$ has its mode at $x^\star = (\alpha_1 - 1)/(\alpha_1 + \alpha_2 - 2)$ when $\alpha_1 > 1$, $\alpha_2 > 1$. Use $m(x) = f_X(x^\star)$.

0. Compute $f^\star = f_X\left(\frac{\alpha_1 - 1}{\alpha_1 + \alpha_2 - 2}\right)$
1. Generate $V \sim g = U(0,1)$
2. Generate $U \sim U(0,1)$
3. If $U \leq f_X(V)/m(V) = [V^{\alpha_1 - 1}(1-V)^{\alpha_2 - 1}/\mathrm{B}(\alpha_1, \alpha_2)]/f^\star$ then return $X = V$

   Otherwise go to Step 1

Key proof steps

Need to show that

$$\Pr\{V \le x|\mathcal{A}\} = \frac{\Pr\{V \le x, U \le f_X(V)/m(V)\}}{\Pr\{U \le f_X(V)/m(V)\}} = F_X(x).$$

$$\begin{aligned}
&\Pr\{V \le x, U \le f_X(V)/m(V)\} \\
&\quad = \int_{-\infty}^{\infty} \Pr\{y \le x, U \le f_X(y)/m(y)|V = y\}g(y)\,dy \\
&\quad = \int_{-\infty}^{x} \Pr\{U \le f_X(y)/m(y)\}g(y)\,dy \\
&\quad = \frac{1}{c}\int_{-\infty}^{x} f_X(y)\,dy = \frac{1}{c}F_X(x)
\end{aligned}$$

Rejection notes

- Number of trials to get one X is geometric with expected value c (can you see why?).
- The beta example had $c = 2.0736$ which is very bad. Good rejection algorithms have c very close to 1. These algorithms tend to majorize f_X in a piecewise manner.
- We can develop a corresponding rejection algorithm for discrete distributions.

Particular properties

Exploit relationships among distributions:

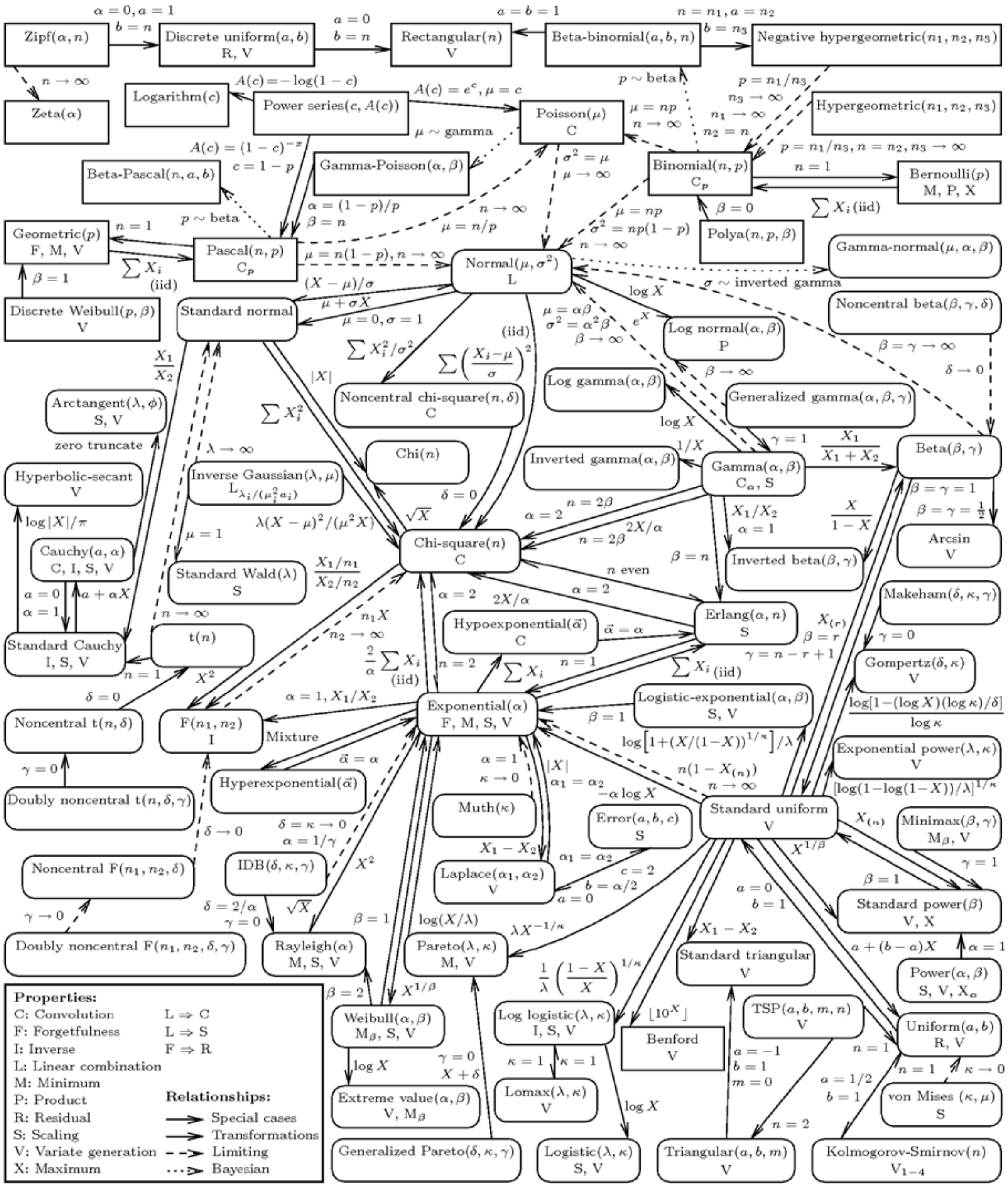
Normal (μ, σ^2)**:** $X = \mu + \sigma \cdot \mathrm{N}(0,1)$

Lognormal: $Y = \exp(X)$

Beta: If $X_i \sim \mathrm{gamma}(\alpha_i, \beta)$ independent, then
$X = X_1/(X_1 + X_2)$ is $\mathrm{beta}(\alpha_1, \alpha_2)$.

But avoid approximations and use exact methods.

Example: $Z = \sum_{i=1}^{12} U_i - 6$ for standard normal.



An interactive version of this chart can be found at www.math.wm.edu/~leemis/chart/UDR/UDR.html

Generating pseudorandom numbers

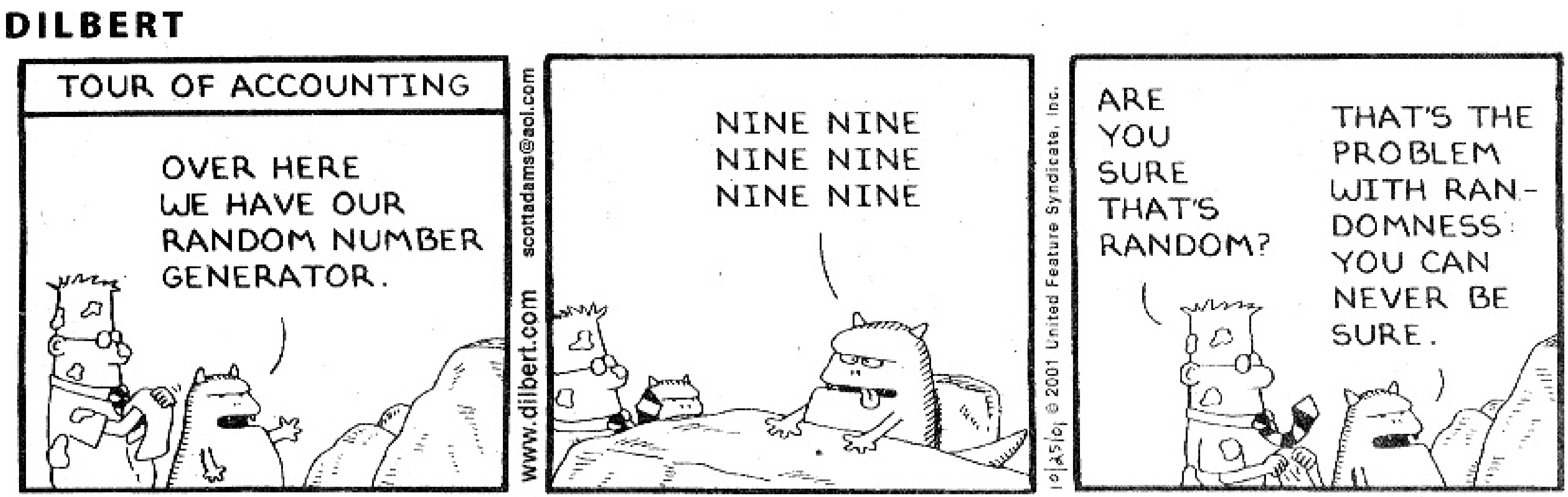
The theory of simulation is based on having a source of i.i.d. $U(0, 1)$ random variates. In reality we use a deterministic, algorithm-generated list that repeats

$$u_1, u_2, u_3, \ldots, u_i, u_{i+1}, \ldots, u_{P-1}, u_P, u_1, u_2, u_3, \ldots$$

Producing good pseudorandom number generators (RNG) is *hard*.

On the interval $(0, 1)$ the values $u_1, u_2, u_3, \ldots$ can appear i.i.d. uniform, but in $(0, 1)^d$ the points $(u_{id+1}, u_{id+2}, \ldots, u_{id+d})$ may appear very non-random.

Modern generators have periods longer than we can test.


DILBERT
TOUR OF ACCOUNTING
OVER HERE WE HAVE OUR RANDOM NUMBER GENERATOR.
scottadams@aol.com
www.dilbert.com
NINE NINE NINE NINE NINE NINE
© 2001 United Feature Syndicate, Inc.
ARE YOU SURE THAT'S RANDOM?
THAT'S THE PROBLEM WITH RAN-DOMNESS: YOU CAN NEVER BE SURE.

Building block: MCG

Multiplicative congruential generator (MCG):

$$
\begin{aligned}
z_i &= a z_{i-1} \mod m \\
u_i &= \frac{z_i}{m}
\end{aligned}
$$

In VBASim

$$
\begin{aligned}
z_i &= 630{,}360{,}016 z_{i-1} \mod 2^{31} - 1 \\
u_i &= \frac{z_i}{2^{31} - 1}
\end{aligned}
$$

Maximal period is $P = m - 1 \approx 2$ billion generating $\{0, 1/m, 2/m, \ldots, (m-1)/m\}$. Why?

We don't want 0 or 1. Why?

Extending the period

Multiple recursive generators (MRG): Max $P = m^K - 1$

$$z_i = (a_1 z_{i-1} + a_2 z_{i-2} + \cdots + a_K z_{i-K}) \mod m$$

$$u_i = \begin{cases} \dfrac{z_i}{m+1}, & z_i > 0 \\ \dfrac{m}{m+1}, & \text{otherwise.} \end{cases}$$

Combined generators: Max $P = (P_1 P_2 \cdots P_J)/2^{J-1}$

$$z_i = (\delta_1 \, z_{i,1} + \delta_2 \, z_{i,2} + \cdots + \delta_J \, z_{i,J}) \mod m_1$$

$$u_i = \begin{cases} \dfrac{z_i}{m_1+1}, & z_i > 0 \\ \dfrac{m_1}{m_1+1}, & \text{otherwise} \end{cases}$$

Each $z_{i,j}$ a MRG.

L'Ecuyer's MRG32k3a

Uses $J = 2$ MRGs of order $K = 3$.

$$
\begin{aligned}
z_{i,1} &= (1{,}403{,}580\, z_{i-2,1} - 810{,}728\, z_{i-3,1}) \quad \bmod\ 2^{32} - 209 \\
z_{i,2} &= (527{,}612\, z_{i-1,2} - 1{,}370{,}589\, z_{i-3,2}) \quad \bmod\ 2^{32} - 22{,}853 \\
z_i &= (z_{i,1} - z_{i,2}) \quad \bmod\ 2^{32} - 209.
\end{aligned}
$$

Has $P \approx 3 \times 10^{57}$. If you could generate 2 billion U's per second, it would take longer than the age of the universe to exhaust this generator!

A great deal of theory and computation was required to prove good properties.

Proper use of RNGs

To get RNGs started we need a *seed*. A single seed z_0 is needed for a MCG; several seeds are needed for a combined MRG.

RNGs provide multiple *streams* (pointers to seeds) spaced far apart in the sequence. Why?
  VBASim: `Expon(Mean, Stream)`

Provided we make enough replications or long enough runs, no seed/stream is better than any other in a good RNG.

Streams exist to help us make *comparisons*. More later.

Question: Why don't we randomly sample the starting seed?