WIKIPEDIA

Rejection sampling

In numerical analysis and computational statistics, **rejection sampling** is a basic technique used to generate observations from a distribution. It is also commonly called the **acceptance-rejection method** or "accept-reject algorithm" and is a type of exact simulation method. The method works for any distribution in $\mathbb{R}^m$ with a density.

Rejection sampling is based on the observation that to sample a random variable in one dimension, one can perform a uniformly random sampling of the two-dimensional Cartesian graph, and keep the samples in the region under the graph of its density function.[1][2][3] Note that this property can be extended to N-dimension functions.

Contents

Description

To visualize the motivation behind rejection sampling, imagine graphing the density function of a random variable onto a large rectangular board and throwing darts at it. Assume that the darts are uniformly distributed around the board. Now remove all of the darts that are outside the area under the curve. The remaining darts will be distributed uniformly within the area under the curve, and the x-positions of these darts will be distributed according to the random variable's density. This is because there is the most room for the darts to land where the curve is highest and thus the probability density is greatest.

The visualization as just described is equivalent to a particular form of rejection sampling where the "proposal distribution" is uniform (hence its graph is a rectangle). The general form of rejection sampling assumes that the board is not necessarily rectangular but is shaped according to the density of some proposal distribution that we know how to sample from (for example, using inversion sampling), and which is at least as high at every point as the distribution we want to sample from, so that the former completely encloses the latter. (Otherwise, there would be parts of the curved area we want to sample from that could never be reached.)

Rejection sampling works as follows:

1. Sample a point on the x-axis from the proposal distribution.
2. Draw a vertical line at this x-position, up to the maximum y-value of the probability density function of the proposal distribution.
3. Sample uniformly along this line from 0 to the maximum of the probability density function. If the sampled value is greater than the value of the desired distribution at this vertical line, reject the x-value and return to step 1; else the x-value is a sample from the desired distribution.

This algorithm can be used to sample from the area under any curve, regardless of whether the function integrates to 1. In fact, scaling a function by a constant has no effect on the sampled x-positions. Thus, the algorithm can be used to sample from a distribution whose normalizing constant is unknown, which is common in computational statistics.

Theory

The rejection sampling method generates sampling values from a target distribution X with arbitrary probability density function $f(x)$ by using a proposal distribution Y with probability density $g(x)$. The idea is that one can generate a sample value from X by instead sampling from Y and accepting the sample from Y with probability $f(x)/(Mg(x))$, repeating the draws from Y until a value is accepted. M here is a constant, finite bound on the likelihood ratio $f(x)/g(x)$, satisfying $1 < M < \infty$ over the support of X; in other words, M must satisfy $f(x) \leq Mg(x)$ for all values of x. Note that this requires that the support of Y must include the support of X—in other words, $g(x) > 0$ whenever $f(x) > 0$.

The validation of this method is the envelope principle: when simulating the pair $(x, v = u \cdot Mg(x))$, one produces a uniform simulation over the subgraph of $Mg(x)$. Accepting only pairs such that $u < f(x)/(Mg(x))$ then produces pairs (x, v) uniformly distributed over the subgraph of $f(x)$ and thus, marginally, a simulation from $f(x)$.

This means that, with enough replicates, the algorithm generates a sample from the desired distribution $f(x)$. There are a number of extensions to this algorithm, such as the Metropolis algorithm.

This method relates to the general field of Monte Carlo techniques, including Markov chain Monte Carlo algorithms that also use a proxy distribution to achieve simulation from the target distribution $f(x)$. It forms the basis for algorithms such as the Metropolis algorithm.

The unconditional acceptance probability is the proportion of proposed samples which are accepted, which is

$$
\begin{aligned}
\mathbb{P}\left(U \le \frac{f(Y)}{Mg(Y)}\right) &= \mathbf{E}\,\mathbf{1}_{\left[U\le \frac{f(Y)}{Mg(Y)}\right]} \\
&= E\left[\mathbf{E}[\mathbf{1}_{\left[U\le \frac{f(Y)}{Mg(Y)}\right]}|Y]\right] && \text{(by t} \\
&= \mathbf{E}\left[\mathbb{P}\left(U\le \frac{f(Y)}{Mg(Y)}\middle|Y\right)\right] \\
&= E\left[\frac{f(Y)}{Mg(Y)}\right] && \text{(because } \Pr(U\le u)=u, \text{when } U \text{ is ur} \\
&= \int\limits_{y:g(y)>0} \frac{f(y)}{Mg(y)} g(y)\,dy \\
&= \frac{1}{M}\int\limits_{y:g(y)>0} f(y)\,dy \\
&= \frac{1}{M} && \text{(since support of } Y \text{ include}
\end{aligned}
$$

where $U \sim \mathrm{Unif}(0,1)$, and the value of y each time is generated under the density function $g(\cdot)$ of the proposal distribution Y.

The number of samples required from Y to obtain an accepted value thus follows a geometric distribution with probability $1/M$, which has mean M. Intuitively, M is the expected number of the iterations that are needed, as a measure of the computational complexity of the algorithm.

Rewrite the above equation,

$$
M = \frac{1}{\mathbb{P}\left(U \le \frac{f(Y)}{Mg(Y)}\right)}
$$

Note that $1 \le M < \infty$, due to the above formula, where $\mathbb{P}\left(U \le \frac{f(Y)}{Mg(Y)}\right)$ is a probability which can only take values in the interval $[0,1]$. When M is chosen closer to one, the unconditional acceptance probability is higher the less that ratio varies, since M is the upper bound for the likelihood ratio $f(x)/g(x)$. In practice, a value of M closer to 1 is preferred as it implies fewer rejected samples, on average, and thus fewer iterations of the algorithm. In this sense, one prefers to have M as small as possible (while still satisfying $f(x) \le Mg(x)$, which suggests that $g(x)$ should generally resemble $f(x)$ in some way. Note, however, that M cannot be equal to 1: such would imply that $f(x) = g(x)$, i.e. that the target and proposal distributions are actually the same distribution.

Rejection sampling is most often used in cases where the form of $f(x)$ makes sampling difficult. A single iteration of the rejection algorithm requires sampling from the proposal distribution, drawing from a uniform distribution, and evaluating the $f(x)/(Mg(x))$ expression. Rejection sampling is thus more efficient than some other method whenever M times the cost of these operations—which is the expected cost of obtaining a sample with rejection sampling—is lower than the cost of obtaining a sample using the other method.

Algorithm

The algorithm (used by John von Neumann and dating back to Buffon and his needle) to obtain a sample from distribution X with density f using samples from distribution Y with density g is as follows:

- Obtain a sample y from distribution Y and a sample u from $\mathrm{Unif}(0,1)$ (the uniform distribution over the unit interval).
- Check whether or not $u < f(y)/Mg(y)$.
  - If this holds, accept y as a sample drawn from f;
  - if not, reject the value of y and return to the sampling step.

The algorithm will take an average of M iterations to obtain a sample.

Advantages over sampling using naive methods

Rejection sampling can be far more efficient compared with the naive methods in some situations. For example, given a problem as sampling $X \sim F(\cdot)$ conditionally on X given the set A, i.e., $X|X \in A$, sometimes X can be easily simulated, using the naive methods (e.g. by inverse transform sampling):

- Sample $X \sim F(\cdot)$ independently, and leave those satisfying $\{n \geq 1 : X_n \in A\}$
- Output: $\{X_1, X_2, \ldots, X_N : X_i \in A, i = 1, \ldots, N\}$

The problem is this sampling can be difficult and inefficient, if $\mathbb{P}(X \in A) \approx 0$. The expected number of iterations would be $\dfrac{1}{\mathbb{P}(X \in A)}$, which could be close to infinity. Moreover, even when you apply the Rejection sampling method, it is always hard to optimize the bound M for the likelihood ratio. More often than not, M is large and the rejection rate is high, the algorithm can be very inefficient. The Natural Exponential Family (if it exists), also known as exponential tilting, provides a class of proposal distributions that can lower the computation complexity, the value of M and speed up the computations (see examples: working with Natural Exponential Families).

Examples: working with natural exponential families

Given a random variable $X \sim F(\cdot)$, $F(x) = \mathbb{P}(X \leq x)$ is the target distribution. Assume for the simplicity, the density function can be explicitly written as $f(x)$. Choose the proposal as

$$\begin{aligned} F_\theta(x) &= \mathbb{E}\left[\exp(\theta X - \psi(\theta))\mathbb{I}(X \leq x)\right] \\ &= \int_{-\infty}^{x} e^{\theta y - \psi(\theta)} f(y) dy \\ g_\theta(x) &= F'_\theta(x) = e^{\theta x - \psi(\theta)} f(x) \end{aligned}$$

where $\psi(\theta) = \log(\mathbb{E}\exp(\theta X))$ and $\Theta = \{\theta : \psi(\theta) < \infty\}$. Clearly, $\{F_\theta(\cdot)\}_{\theta \in \Theta}$, is from a natural exponential family. Moreover, the likelihood ratio is

$$Z(x) = \frac{f(x)}{g_\theta(x)} = \frac{f(x)}{e^{\theta x - \psi(\theta)} f(x)} = e^{-\theta x + \psi(\theta)}$$

Note that $\psi(\theta) < \infty$ implies that it is indeed a log moment-generation function, that is, $\psi(\theta) = \log \mathbb{E}\exp(tX)|_{t=\theta} = \log M_X(t)|_{t=\theta}$. And it is easy to derive the log moment-generation function of the proposal and therefore the proposal's moments.

$$\begin{aligned}\psi_\theta(\eta) &= \log(\mathbb{E}_\theta \exp(\eta X)) = \psi(\theta+\eta) - \psi(\theta) < \infty \\ \mathbb{E}_\theta(X) &= \frac{\partial \psi_\theta(\eta)}{\partial \eta}\bigg|_{\eta=0} \\ \mathrm{Var}_\theta(X) &= \frac{\partial^2 \psi_\theta(\eta)}{\partial^2 \eta}\bigg|_{\eta=0}\end{aligned}$$

As a simple example, suppose under $F(\cdot)$, $X \sim \mathrm{N}(\mu, \sigma^2)$, with $\psi(\theta) = \theta\mu + \frac{\sigma^2\theta^2}{2}$. The goal is to sample $X|X \in [b, \infty]$, $b > \mu$. The analysis goes as follows.

- Choose the form of the proposal distribution $F_\theta(\cdot)$, with log moment-generating function as $\psi_\theta(\eta) = \psi(\theta+\eta) - \psi(\eta) = \eta(\mu+\theta\sigma^2) + \frac{\sigma^2\eta^2}{2}$, which further implies it is a normal distribution $\mathrm{N}(\mu+\theta\sigma^2, \sigma^2)$.
- Decide the well chosen θ^* for the proposal distribution. In this setup, the intuitive way to choose θ^* is to set $\mathbb{E}_\theta(X) = \mu + \theta\sigma^2 = b$, that is $\theta^* = \dfrac{b-\mu}{\sigma^2}$
- Explicitly write out the target, the proposal and the likelihood ratio

$$\begin{aligned}f_{X|X\geq b}(x) &= \frac{f(x)\mathbb{I}(x \geq b)}{\mathbb{P}(x \geq b)} \\ g_{\theta^*}(x) &= f(x)\exp(\theta^* x - \psi(\theta^*)) \\ Z(x) &= \frac{f_{X|X\geq b}(x)}{g_{\theta^*}(x)} = \frac{\exp(-\theta^* x + \psi(\theta^*))\mathbb{I}(x \geq b)}{\mathbb{P}(x \geq b)}\end{aligned}$$

- Derive the bound M for the likelihood ratio $z(x)$, which is a decreasing function for $x \in [b, \infty]$, therefore

$$M = Z(b) = \frac{\exp(-\theta^* b + \psi(\theta^*))}{\mathbb{P}(X \geq b)} = \frac{\exp\left(-\frac{(b-\mu)^2}{2\sigma^2}\right)}{\mathbb{P}(X \geq b)} = \frac{\exp\left(-\frac{(b-\mu)^2}{2\sigma^2}\right)}{\mathbb{P}\left(\mathrm{N}(0,1) \geq \frac{b-\mu}{\sigma}\right)}$$

- Rejection sampling criterion: for $U \sim \mathrm{Unif}(0,1)$, if

$$U \leq \frac{Z(x)}{M} = e^{-\theta^*(x-b)}\mathbb{I}(x \geq b)$$

  holds, accept the value of X; if not, continue sampling new $X \sim_{i.i.d.} \mathrm{N}(\mu+\theta^*\sigma^2, \sigma^2)$ and new $U \sim \mathrm{Unif}(0,1)$ until acceptance.

For the above example, as the measurement of the efficiency, the expected number of the iterations the NEF-Based Rejection sampling method is of order b, that is $M(b) = O(b)$, while under the naive method, the expected number of the iterations is $\frac{1}{\mathbb{P}(X \geq b)} = O(b \cdot e^{\frac{(b-\mu)^2}{2\sigma^2}})$, which is far more inefficient.

In general, exponential tilting, a parametric class of proposal distribution, solves the optimization problems conveniently, with its useful properties that directly characterize the distribution of the proposal. For this type of problem, to simulate X conditionally on $X \in A$, among the class of simple distributions, the trick is to use NEFs, which helps to gain some control over the complexity and considerably speed up the computation. Indeed, there are deep mathematical reasons for using NEFs.

Drawbacks

Rejection sampling can lead to a lot of unwanted samples being taken if the function being sampled is highly concentrated in a certain region, for example a function that has a spike at some location. For many distributions, this problem can be solved using an adaptive extension (see adaptive rejection sampling), or with an appropriate change of variables with the method of the ratio of uniforms. In addition, as the dimensions of the problem get larger, the ratio of the embedded volume to the "corners" of the embedding volume tends towards zero, thus a lot of rejections can take place before a useful sample is generated, thus making the algorithm inefficient and impractical. See curse of dimensionality. In high dimensions, it is necessary to use a different approach, typically a Markov chain Monte Carlo method such as Metropolis sampling or Gibbs sampling. (However, Gibbs sampling, which breaks down a multi-dimensional sampling problem into a series of low-dimensional samples, may use rejection sampling as one of its steps.)

Adaptive rejection sampling

For many distributions, finding a proposal distribution that includes the given distribution without a lot of wasted space is difficult. An extension of rejection sampling that can be used to overcome this difficulty and efficiently sample from a wide variety of distributions (provided that they have log-concave density functions, which is in fact the case for most of the common distributions—even those whose *density* functions are not concave themselves!) is known as **adaptive rejection sampling (ARS)**.

There are three basic ideas to this technique as ultimately introduced by Gilks in 1992:[4]

1. If it helps, define your envelope distribution in log space (e.g. log-probability or log-density) instead. That is, work with $h(x) = \log g(x)$ instead of $g(x)$ directly.
   - Often, distributions that have algebraically messy density functions have reasonably simpler log density functions (i.e. when $f(x)$ is messy, $\log f(x)$ may be easier to work with or, at least, closer to piecewise linear).
2. Instead of a single uniform envelope density function, use a piecewise linear density function as your envelope instead.
   - Each time you have to reject a sample, you can use the value of $f(x)$ that you evaluated, to improve the piecewise approximation $h(x)$. This therefore reduces the chance that your next attempt will be rejected. Asymptotically, the probability of needing to reject your sample should converge to zero, and in practice, often very rapidly.
   - As proposed, any time we choose a point that is rejected, we tighten the envelope with another line segment that is tangent to the curve at the point with the same x-coordinate as the chosen point.
   - A piecewise linear model of the proposal log distribution results in a set of piecewise exponential distributions (i.e. segments of one or more exponential distributions, attached end to end). Exponential distributions are well behaved and well understood. The logarithm of an exponential distribution is a straight line, and hence this method essentially involves enclosing the logarithm of the density in a series of line segments. This is the source of the log-concave restriction: if a distribution is log-concave, then its

logarithm is concave (shaped like an upside-down U), meaning that a line segment tangent to the curve will always pass over the curve.
   - If not working in log space, a piecewise linear density function can also be sampled via triangle distributions [5]

3. We can take even further advantage of the (log) concavity requirement, to potentially avoid the cost of evaluating $f(x)$ when your sample *is* accepted.
   - Just like we can construct a piecewise linear upper bound (the "envelope" function) using the values of $h(x)$ that we had to evaluate in the current chain of rejections, we can also construct a piecewise linear lower bound (the "squeezing" function) using these values as well.
   - Before evaluating (the potentially expensive) $f(x)$ to see if your sample will be accepted, we may *already know* if it will be accepted by comparing against the (ideally cheaper) $g_l(x)$ (or $h_l(x)$ in this case) squeezing function that have available.
   - This squeezing step is optional, even when suggested by Gilks. At best it saves you from only one extra evaluation of your (messy and/or expensive) target density. However, presumably for particularly expensive density functions (and assuming the rapid convergence of the rejection rate toward zero) this can make a sizable difference in ultimate runtime.

The method essentially involves successively determining an envelope of straight-line segments that approximates the logarithm better and better while still remaining above the curve, starting with a fixed number of segments (possibly just a single tangent line). Sampling from a truncated exponential random variable is straightforward. Just take the log of a uniform random variable (with appropriate interval and corresponding truncation).

Unfortunately, ARS can only be applied from sampling from log-concave target densities. For this reason, several extensions of ARS have been proposed in literature for tackling non-log-concave target distributions.[6][7][8] Furthermore, different combinations of ARS and the Metropolis-Hastings method have been designed in order to obtain a universal sampler that builds a self-tuning proposal densities (i.e., a proposal automatically constructed and adapted to the target). This class of methods are often called as **Adaptive Rejection Metropolis Sampling (ARMS) algorithms**.[9][10] The resulting adaptive techniques can be always applied but the generated samples are correlated in this case (although the correlation vanishes quickly to zero as the number of iterations grows).

See also

- Inverse transform sampling
- Ratio of uniforms
- Pseudo-random number sampling
- Ziggurat algorithm

References

1. Casella, George; Robert, Christian P.; Wells, Martin T. (2004). *Generalized Accept-Reject sampling schemes*. Institute of Mathematical Statistics. pp. 342–347. doi:10.1214/lnms/1196285403 (https://doi.org/10.1214%2Flnms%2F1196285403). ISBN 9780940600614.
2. Neal, Radford M. (2003). "Slice Sampling" (https://doi.org/10.1214%2Faos%2F1056562461). *Annals of Statistics*. **31** (3): 705–767. doi:10.1214/aos/1056562461 (https://doi.org/10.1214%2Faos%2F1056562461).

MR 1994729 (https://www.ams.org/mathscinet-getitem?mr=1994729). Zbl 1051.65007 (https://zbmath.org/?format=complete&q=an:1051.65007).

3. Bishop, Christopher (2006). "11.4: Slice sampling". *Pattern Recognition and Machine Learning*. Springer. ISBN 978-0-387-31073-2.
4. Adaptive Rejection Sampling for Gibbs Sampling. https://stat.duke.edu/~cnk/Links/tangent.method.pdf
5. D.B. Thomas and W. Luk, Non-uniform random number generation through piecewise linear approximations, 2006. http://www.doc.ic.ac.uk/~wl/papers/iee07dt.pdf
6. Hörmann, Wolfgang (1995-06-01). "A Rejection Technique for Sampling from T-concave Distributions". *ACM Trans. Math. Softw*. **21** (2): 182–193. CiteSeerX 10.1.1.56.6055 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.56.6055). doi:10.1145/203082.203089 (https://doi.org/10.1145%2F203082.203089). ISSN 0098-3500 (https://www.worldcat.org/issn/0098-3500).
7. Evans, M.; Swartz, T. (1998-12-01). "Random Variable Generation Using Concavity Properties of Transformed Densities". *Journal of Computational and Graphical Statistics*. **7** (4): 514–528. CiteSeerX 10.1.1.53.9001 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.53.9001). doi:10.2307/1390680 (https://doi.org/10.2307%2F1390680). JSTOR 1390680 (https://www.jstor.org/stable/1390680).
8. Görür, Dilan; Teh, Yee Whye (2011-01-01). "Concave-Convex Adaptive Rejection Sampling". *Journal of Computational and Graphical Statistics*. **20** (3): 670–691. doi:10.1198/jcgs.2011.09058 (https://doi.org/10.1198%2Fjcgs.2011.09058). ISSN 1061-8600 (https://www.worldcat.org/issn/1061-8600).
9. Gilks, W. R.; Best, N. G.; Tan, K. K. C. (1995-01-01). "Adaptive Rejection Metropolis Sampling within Gibbs Sampling". *Journal of the Royal Statistical Society. Series C (Applied Statistics)*. **44** (4): 455–472. doi:10.2307/2986138 (https://doi.org/10.2307%2F2986138). JSTOR 2986138 (https://www.jstor.org/stable/2986138).
10. Meyer, Renate; Cai, Bo; Perron, François (2008-03-15). "Adaptive rejection Metropolis sampling using Lagrange interpolation polynomials of degree 2". *Computational Statistics & Data Analysis*. **52** (7): 3408–3423. doi:10.1016/j.csda.2008.01.005 (https://doi.org/10.1016%2Fj.csda.2008.01.005).

- Robert, C.P. and Casella, G. "Monte Carlo Statistical Methods" (second edition). New York: Springer-Verlag, 2004.
- J. von Neumann, "Various techniques used in connection with random digits. Monte Carlo methods", Nat. Bureau Standards, 12 (1951), pp. 36–38.