Poisson processes

CADLAG functions

Consider a Poisson process that runs forever starting from time 0.

Let $N(t)$ denote the number of events that have occurred up to and including time t, so that
$N(t) = 0$ for $0 \leq t < T_1$,
$N(t) = 1$ for $T_1 \leq t < T_1+T_2$,
$N(t) = 2$ for $T_1+T_2 \leq t < T_1+T_2+T_3$,
...
where T_1, T_2, T_3, ... are i.i.d. r.v.'s, each exponentially distributed with parameter λ.

Here $N()$ is a random CADLAG function.
For all t, the expected value of $N(t)$ is λt, so (thinking now of averaging functions instead of just numbers), the expected "value" of the random function N is the linear function $f(t) = \lambda t$.

We saw last time that this function satisfies the differential equation $f'(t) = \lambda$.

A more interesting example is the function $[N(t)]^2$, taking its values in {0,1,4,9,...}; note that Exp[$[N(t)]^2$] is the expected value of the square of a Poisson random variable with parameter λt. Write $[N(t)]^2$ as $N^2(t)$. Let's view Exp[$N^2(t)$] as a function of t and find the differential equation it satisfies (continuing to use loose but justifiable reasoning). First let's condition on the event $N(t) = k$; then for small $\Delta t > 0$, it's nearly true that $N(t+\Delta t)$ is either $N(t)$ or $N(t)+1$, so that $N^2(t+\Delta t) - N^2(t)$ is either $k^2 - k^2 = 0$ or $(k+1)^2 - k^2 = 2k+1$, where the respective probabilities of these two cases are $\approx 1 - \lambda\,\Delta t$ and $\lambda\,\Delta t$. Hence the conditional expected value of $N^2(t+\Delta t) - N^2(t)$, given $N(t) = k$, is $(\lambda\,\Delta t)(2k+1) = (\lambda\,\Delta t)(2N(t)+1)$. Hence the unconditional expected value of $N^2(t+\Delta t) - N^2(t)$ is the expected value of $(\lambda\,\Delta t)(2N(t)+1)$, which is

$(\lambda\, \Delta t)(2\lambda t+1)$ (since $(\lambda\, \Delta t)(2N(t)+1)$ is linear in $N(t)$).

Dividing by Δt and taking the limit, we see that

$g(t) := \text{Exp}[N^2(t)]$ satisfies

$$g'(t) = \lim\, (g(t+\Delta t)-g(t))/\Delta t$$

$$= \lim\, (\text{Exp}[N^2(t+\Delta t)] - \text{Exp}[N^2(t)])/\Delta t$$

$$= \lim\, (\text{Exp}[N^2(t+\Delta t) - N^2(t)])/\Delta t$$

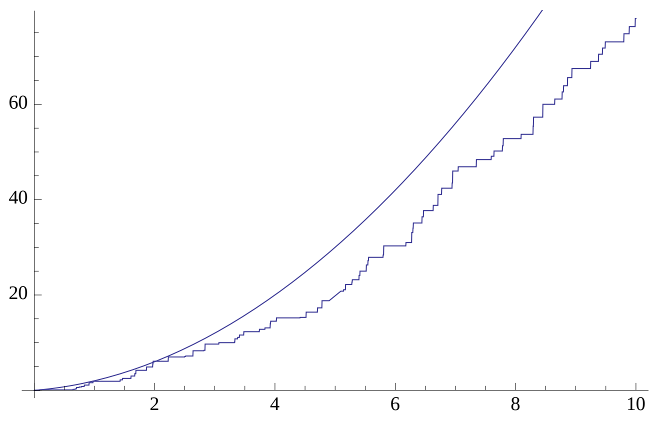
$$= \lambda(2\lambda t+1) = 2\lambda^2 t + \lambda.$$

Since $g(0) = 0$ (make sure you see why!), we get

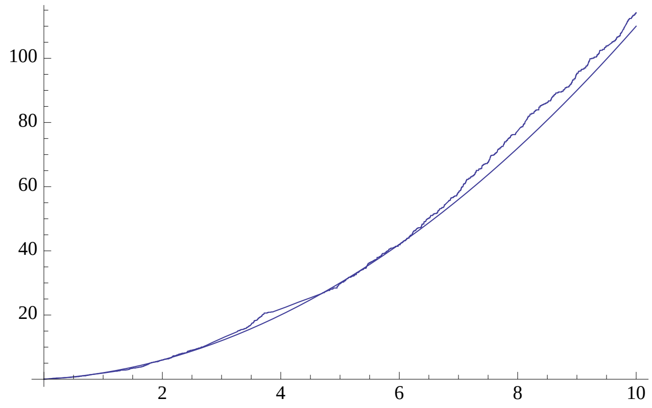
$$g(t) = \lambda^2 t^2 + \lambda t.$$

(Check this by simulation:

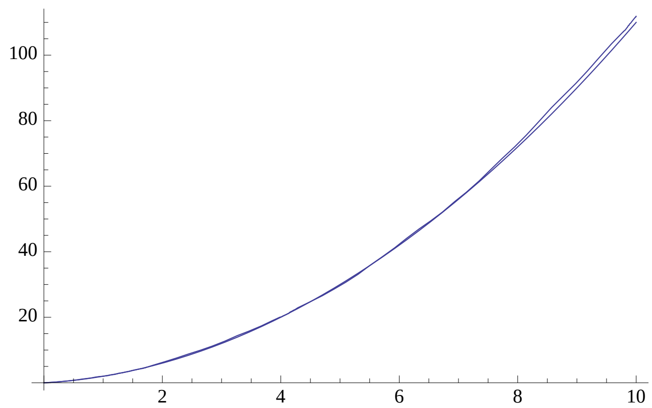
```
cadlag[] := Module[{L, Events}, L = Table[RandomReal[ExponentialDistribution[1]], {n, 100}];
  Events = Table[Sum[L[[k]], {k, 1, n}], {n, 1, 20}];
  Return[Sum[HeavisideTheta[x - Events[[n]]], {n, 1, 20}]]]

Show[Plot[Evaluate[Mean[Table[cadlag[]^2, {10}]]], {x, 0, 10}], Plot[x^2 + x, {x, 0, 10}]]
```



```
Show[Plot[Evaluate[Mean[Table[cadlag[]^2, {100}]]], {x, 0, 10}], Plot[x^2 + x, {x, 0, 10}]]
```



```
Show[Plot[Evaluate[Mean[Table[cadlag[]^2, {1000}]]], {x, 0, 10}], Plot[x^2 + x, {x, 0, 10}]]
```



Note that the average of the squares of n Poisson counting functions appears to be converging to the function $g(t) = \lambda^2 t^2 + \lambda t$, as predicted.)

In particular, the variance of a Poisson random variable with parameter λt is

$\text{Exp}[N(t)^2] - [\text{Exp}[N(t)]]^2 = g(t) - [f(t)]^2$

$= (\lambda^2 t^2 + \lambda t) - \lambda^2 t^2 = \lambda t$, which recovers the (perhaps familiar) result that the variance of a Poisson random variable is always equal to its mean.

We can also compute the variance of $N(t)$ by going back to the original Bernoulli trials picture. The number of Poisson events up to time t, for a Poisson process of rate λ, can be approximated by the number of successes in $n\,t$ independent random trials, where the probability of success on each trial is λ/n. Write this as $X_1+X_2+\ldots+X_{nt}$ where X_i is 1 if the ith trial is a success and 0 otherwise.

Since the trials are independent, the variance of the sum is the sum of the variances, or $n\,t$ times the variance of each X_i, or $(n\,t)(\lambda/n)(1-\lambda/n)$, which goes to λt as $n\to\infty$.

Memorylessness

Having computed the expected value of $N(t)\,N(t)$, we can ask, what about the expected value of $N(s)\,N(t)$ with $s < t$?

Since

$N(s)$ = the # of successes from time 0 to time s

and

$N(t) - N(s)$ = the # of successes from time s to time t, the random variables $N(s)$ and $N(t) - N(s)$ are <u>independent</u> of each other (think about Bernoulli trials: the number of successes in the first ns trials and the number of successes in the next $n(t\text{-}s)$ trials are independent), we have

$E[N(s)\,(N(t) - N(s))] = E[N(s)]\;E[N(t) - N(s)]$

$= (\lambda s)\,(\lambda(t\text{-}s)) = \lambda^2 s(t\text{-}s)$.

So $E[N(s)\,N(t)] = E[N(s)\,(N(t) - N(s)) + N(s)\,N(s)]$

$= E[N(s)\,(N(t) - N(s))] + E[N(s)\,N(s)]$

$= \lambda^2 s(t\text{-}s) + (\lambda^2 s^2 + \lambda s) = \lambda^2 st + \lambda s$

So $\mathrm{Cov}(N(s), N(t)) = E[N(s)\,N(t)] - E[N(s)]\;E[N(t)]$

$= \lambda^2 st + \lambda s - \lambda s\,\lambda t = \lambda s$.

(Check: as s goes to t, this recovers the formula Var(-
$N(t)) = \lambda t$.)

More generally, for any numbers $t_1 < t_2 < t_3$, ... , the random variables $N(t_1)$, $N(t_2)\text{-}N(t_1)$, $N(t_3)\text{-}N(t_2)$, ... are independent.

Calling these random variables "increments", we say that a Poisson counting process "**has independent increments**".

The independence of $N(s)$ and $N(t) - N(s)$ means that no matter how many, or how few, events occurred from time 0 to time s, the expected number of events that will occur between time s and time t is still $\lambda(t-s)$.
We say that the Poisson process is **memoryless**.

Suppose bus-arrivals on some route are governed by a Poisson process of intensity λ (a dubious assumption, since buses tend to cluster for well-understood reasons). Suppose λ has units of arrivals per hour. When you arrive at the bus stop, how long should you expect to have to wait? ...

... $1/\lambda$ hours.

Now suppose that when you arrive at the bus stop, someone tells you that a bus on that route just left 5 minutes ago. How long should you expect to have to wait? ...

... $1/\lambda$ hours.

Or, suppose someone tells you that no bus on that route has been there in the past hour. How long should you expect to have to wait? ...

... Still $1/\lambda$ hours!

This may seem counterintuitive, but that's mostly because buses in real life aren't governed by a Poisson process. If the result still seems surprising, consider Bernoulli trials: regardless of whether the last 10 tosses of a fair coin have come up heads, or the last 10 tosses have come up tails, or any outcome in between, the expected number of tosses required until you next toss heads is still 2.

Likewise: No matter what a Poisson process of rate λ did prior to time t, the expected time until the next event occurs is $1/\lambda$.

The *M*/*M*/1 queueing model

(adapted from section 8.3.1 of "Introduction to Probability Models" by Sheldon M. Ross (9th edition)

Suppose that customers arrive at a single-server service station in accordance with a Poisson process having rate λ (so that on average λ customers arrive per hour, and the average time from one customer-arrival to the next is $1/\lambda$). When a customer arrives, he/she joins a queue of customers awaiting service. Upon reaching the head of the queue, the customer is served in a random amount of time in accordance with an exponential distribution with parameter μ (so that the expected time to serve a customer who has reached the head of the queue, aka the expected service time, is $1/\mu$).

This is called an *M*/*M*/1 queue because the interarrival times are memoryless, the service times are memoryless, and there is just 1 server. We'll assume that there is no bound on the queue-length.

For $n = 0, 1, 2, \ldots$, let $P_n(t)$ be the probability that the queue is of length n at time t (with $t \geq 0$). (Thus, if we wanted to assume that the queue is empty at time 0, we would take initial conditions $P_0(t) = 1$ and $P_1(t) = P_2(t) = \ldots = 0$.) The state of the system (the current queue-length) is always a non-negative integer, and it always changes by +1 or -1: for $n \geq 0$, state n goes to state $n+1$ at rate λ, and for $n \geq 1$, state n goes to state $n-1$ at rate μ. We call λ and μ the transition rates associated with increase by 1 and decrease by 1, respectively.

(A nonnegative-integer-valued stochastic process whose jumps are always +1 or -1 is also called a birth-and-death process, if we think of n as being the size of a population instead of the size of a queue.)

The quantities $P_0(t), P_1(t), P_2(t), \ldots$ evolve over time in accordance with a system of differential equations:

$dP_0 / dt = \mu P_1 - \lambda P_0$

$dP_1 / dt = \lambda P_0 + \mu P_2 - \lambda P_1 - \mu P_1$

$dP_2 / dt = \lambda P_1 + \mu P_3 - \lambda P_2 - \mu P_2$

...

(It takes a little bit of work to get from the Poisson process model to the differential equations; I'll come back to this point if there's time.)

Thus we can say what the equilibrium looks like by setting $dP_n / dt = 0$ for all n, using each successive equation to simplify the next:

$\mu P_1 = \lambda P_0$

$\lambda P_0 + \mu P_2 = \lambda P_1 + \mu P_1 \Rightarrow \mu P_2 = \lambda P_1$

$\lambda P_1 + \mu P_3 = \lambda P_2 + \mu P_2 \Rightarrow \mu P_3 = \lambda P_2$

...

Hence $P_1 = (\lambda/\mu) P_0$, $P_2 = (\lambda/\mu) P_1$, $P_3 = (\lambda/\mu) P_2$, ... so

$$P_i = (\lambda/\mu)^i P_0 \text{ for all } i.$$

If $\lambda < \mu$, then letting

$$Z = \sum_{i=0}^{\infty} (\lambda/\mu)^i = 1 / (1 - \lambda/\mu) = \frac{\mu}{\mu-\lambda}$$

we get stationary distribution

$$P_i = (\lambda/\mu)^i / Z$$

for the continuous-time $M/M/1$ queue, exactly as in the case of a discrete-time queue where λ and μ are transition probabilities rather than transition rates.

(Caveat: continuous-time processes and their discrete-time analogues don't usually agree this precisely!)

This is an example of a continuous-time Markov chain, governed by transition <u>rates</u> instead of transition <u>probabilities</u>. In a future semester, I would probably sketch out the theory of finite-state continuous-time Markov chains, highlighting its resemblances to discrete-time Markov chain theory, as well as the differences. Some technical details differ, but for both the discrete-time-and continuous-time versions, linear algebra methods play a major role.

Splitting and thinning Poisson processes

In the preceding example, we had a Poisson process for arrivals of new customers at the queue and, for each customer, an exponential process that starts when that customer reaches the head of the line. (To be fanciful, imagine that each customer who reaches the head of the queue is handed a lump of some mildly radioactive stuff and a Geiger counter; when the Geiger counter clicks, the super-efficient but lazy clerk services him/her instantly and sends the customer home.)

What if each customer gets a radioactive lump when joining the queue, rather than when reaching its head, and that the clerk's policy is to instantly service a customer who reaches the head of the queue the next time the customer's Geiger counter clicks? Would this lead to shorter wait-times, or longer wait times? ...

... There'd be no difference, because Poisson processes are memoryless!

What if the business has just one radioactive lump and just one Geiger counter, kept behind the desk, and when a customer reaches the head of the queue, that customer gets serviced the very next time that the Geiger counter clicks? Would this lead to shorter wait-times, or longer wait times? ...

... Again, there'd be no difference, because Poisson processes are memoryless.

So one way to simulate the $M/M/1$ queue is to simulate two independent Poisson processes, one for arrivals and one for departures, which we think of as "Poisson timers" that "go off" at unpredictable times (just like Geiger counters); when the first Poisson timer goes off, we add a person to the end of the queue, and when the second Poisson timer goes off, we remove the person at the head of the queue (unless there's no one there, in which case nothing happens).

"But wouldn't it make more sense to stop the second timer when the queue is empty, and re-start it again after someone actually arrives?" ...

... You could, but it wouldn't affect the probability distribution governing how the queue evolves, because ...

... Poisson processes have no memory!

In fact, to simulate the $M/M/1$ queue, we can get by with just a single Poisson process handling both the arrivals and the departures. Take a Poisson process with rate $\lambda+\mu$, and each time it "goes off", toss a coin with bias $\lambda:\mu$ (that is, the coin comes up heads with probability $\frac{\lambda}{\lambda+\mu}$ and tails with probability $\frac{\mu}{\lambda+\mu}$); if the coin comes up heads, add a person to the queue, and if it comes up tails, remove a person from the queue.

It's fairly intuitive that if you take a Poisson process with rate $\lambda+\mu$ and accept only $\frac{\lambda}{\lambda+\mu}$ of the events, the resulting thinned process will be a Poisson process with rate $(\lambda+\mu)\left(\frac{\lambda}{\lambda+\mu}\right) = \lambda$.

(Think about Bernoulli trials: if you generate Bernoulli trials where each trial has probability of success equal to p, and each time you get a success you toss a coin and "accept" the success with probability q, the resulting sequence of "accepted successes" is a Bernoulli process with parameter pq; if you make the Bernoulli trials happen faster and faster, you see that a randomly thinned Poisson process is again a Poisson process.)

But what's counter-intuitive is that in the continuous time context, when you split a Poisson process into two thinner Poisson processes, the two processes are independent!

(This is false for Bernoulli trials: E.g., consider Bernoulli trials with parameter $p = 1/10$, and every time there's a success, use a coin flip to decide whether it's a "red success" or a "blue success". The sequence of red successes is a sequence of Bernoulli trials with parameter $p' = 1/20$, with all trials independent of each other; and the sequence of blue successes is a sequence of Bernoulli trials with parameter p' = 1/20, with all trials independent of each other; but the blue sequence is NOT independent of the red sequence, because if it were, then 1 out of 400 trials would have to give a success that's BOTH red and blue, and our scheme won't permit that.)

The preceding analysis might make us doubt the claim about splitting Poisson processes, but examined more closely, it shows us the "escape-hatch", namely, that for two independent Poisson processes with rate λ (unlike two independent Bernoulli processes with probability p), the chance of success occurring simultaneously in both processes must be **zero**.

In fact, if you repeat the construction of Poisson processes as a limit of Bernoulli processes, you'll get a proof of the claim about splitting Poisson processes.

Going in the other direction, if you have a Poisson process of rate λ and an independent Poisson process of rate μ, and you take the set of all event-times for both processes and lump them together, the result is a Poisson process of rate $\lambda + \mu$.

Non-homogeneous Poisson processes

Recall that a Poisson process is a counting process that associates some (random) finite non-negative integer with every interval I; the Poisson process of intensity λ is characterized by two properties (plus a few technical hypotheses I won't worry about here):

(1) if $I = [t_1, t_2]$, the expected number of events in the time-interval I is $\lambda(t_2 - t_1)$; and

(2) for any two disjoint time-intervals I_1 and I_2, the number of events occurring in I_1 is independent of the number of events occurring in I_2 (we saw this in the special case of the intervals from 0 to s and from s to t). This is called the independent increments property.

(If this looks unfamiliar, recall that last time we represented a Poisson process as a function $N(t)$ that signifies, for each $t \geq 0$, the number of events that occurred in $[0, t]$. So the number of events that occurred in $[t_1, t_2]$ is just $N(t_2) - N(t_1)$.)

So for a Poisson process, the expected number of events from time t to time $t+\Delta t$ is exactly $\lambda \Delta t$. That is, the expected number of events from time t to time $t+\Delta t$, divided by Δt, equals λ.

More generally, we can have a counting process with independent increments such that the expected number of events from time t to time $t+\Delta t$, divided by Δt, converges to some function $\lambda(t)$, rather than some constant λ, as Δt goes to 0.

This is called a <u>nonhomogeneous Poisson process</u>.

In the case where we have an upper bound Λ on $\lambda(t)$ valid for all t, we can use the thinning trick (also called the sampling trick) discussed earlier: generate an ordinary Poisson process of rate Λ, and if the timer goes off at time t, accept the event with probability $\lambda(t)/\Lambda$ and reject it otherwise. (If $\lambda(t)$ is some constant less than Λ, this is ordinary Poisson thinning.) Note the resemblance to acceptance/rejection sampling.

Application: Recall that the expected number of Poisson events in a time-interval *I* is proportional to the length of *I*. But don't think of *I* as a time-interval anymore; think of it as an interval in a 1-dimensional space, and think of the Poisson process as defining a way of throwing "darts" at the line and seeing how many of them land in *I*. Analogously, define a <u>2-dimensional Poisson process</u> with intensity λ as a random variable whose "values" are sets of points in the plane, such that:

(1) for any subset S of the plane with area A, the expected number of darts landing in S is λA; and

(2) for any two disjoint subsets S_1 and S_2 of the plane, the number of darts landing in S_1 is independent of the number of darts landing in S_2 (compare this with the comparable statement in 1 dimension about the intervals I_1 and I_2); and

... (some technical hypotheses I won't include here).

For a nice applet showing the 2-dimensional Poisson process on a rectangle, see
http://www.math.uah.edu/stat/applets/Poisson2DExperiment.xhtml

How do we simulate a 2-dimensional Poisson process on a rectangle?

You can use 2-dimensional versions of the methods we used in 1 dimension last time. E.g., use a Poisson random variable (not to be confused with a Poisson process!) to decide how many points the Poisson process will assign to the whole rectangle, and then choose that many points uniformly and independent from the rectangle, where choosing a point (x,y) uniformly in the rectangle $[a,b]\times[c,d]$ means choosing x uniformly in $[a,b]$ and choosing y (independently) uniformly in $[c,d]$.

How do we simulate a 2-dimensional Poisson process on the disk of radius R?

Answer #1: Simulate a 2-dimensional Poisson process on the square of side-length $2R$, and throw out all the points that lie outside the concentric disk of radius R (a version of acceptance/rejection sampling).

Answer #2: Construct the points radially from the center. Note that the annulus from radius r to radius $r+\Delta r$ has area $\approx 2\pi r\ \Delta r$, so the annuli further out are more likely to contain points than the ones further in, even if they have the same thickness Δr. Sending $\Delta r \to 0$, we find that if we replace the random points in the disk by their distances r from the center, and order these distances by size, the result is a nonhomogeneous Poisson process with rate $\lambda(r) = Cr$ for some suitable constant C. (Note that distance r plays the role of time t here.) To simulate this sequence of distances, simulate a Poisson process of rate CR and apply non-homogeneous thinning, accepting a proposed r with probability r/R. (This ceases to be feasible when $r > R$, since then r/R is not a probability, but this is okay, since we're only interested in points inside the disk of radius R; once our proposed distances from the center exceed R, we can stop generating proposed distances.) Then take the resulting sequence of random distances $r_1, r_2, \ldots$ and choose a random point uniformly on the circle of radius r_1, a random point uniformly on the circle of radius r_2, etc.; the result will be a finite set of points in the disk governed by the 2-dimensional Poisson distribution of rate λ (so that, in particular, for any subset S of the disk of area A, the expected number of points in the randomly-chosen subset that will lie in S equals λA).

Application to a variant Polya urn model

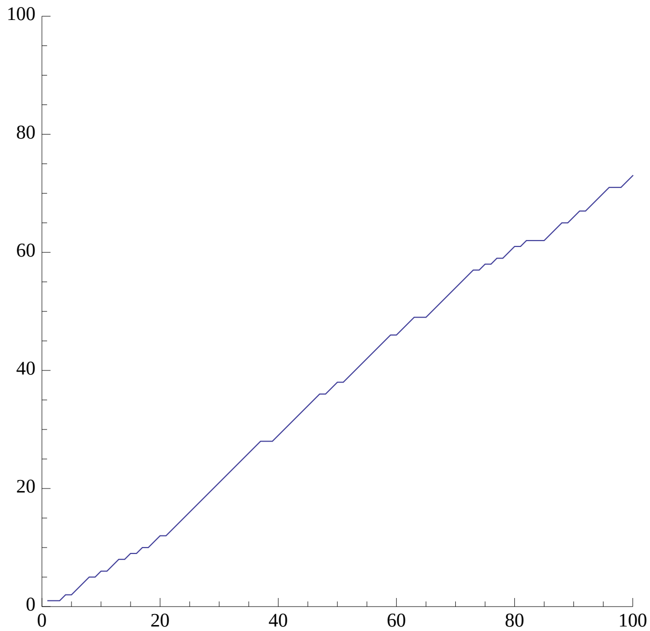
Recall the Polya urn model: starting with an urn containing at least one white ball and at least one black ball, we repeatedly draw a ball from the urn and replace it by two balls of that same color, increasing the number of balls in the urn by 1. Thus, if the urn currently contains a white balls and b black balls, the operation adds a white ball with probability $a/(a+b)$ and adds a black ball with probability $b/(a+b)$.

```
Polya[n_] := Module[{X, k, a, b}, X = Table[1, {n}]; k = 2; While[k < n, a = X[[k]];
   b = k - X[[k]]; X[[k + 1]] = X[[k]] + RandomInteger[BernoulliDistribution[a / (a + b)]];
   (* Print["X is now ",X]; *) k++]; Return[X]]
```

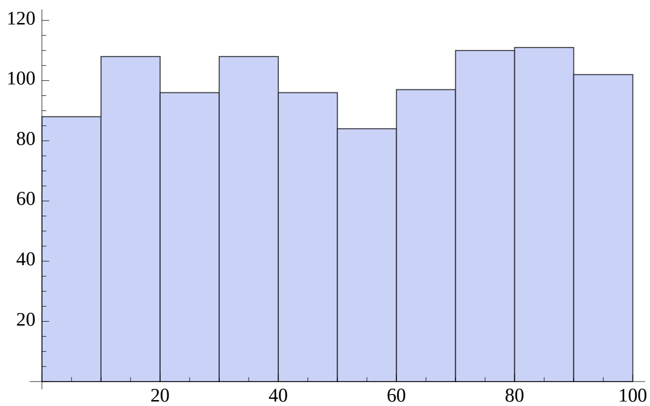
```
Polya[16]
```

{1, 1, 2, 2, 3, 4, 4, 4, 4, 4, 5, 5, 6, 7, 8, 8}

```
ListPlot[Polya[100], PlotRange → {{0, 100}, {0, 100}}, AspectRatio → Automatic, Joined → True]
```




```
Histogram[Table[Polya[100][[100]], {1000}], 10]
```



If we let the random variable X_n be the number of white balls in the urn when the total number of balls in the urn is n, then one can show that X_n/n converges almost surely to a (random) real number W that is uniformly distributed in [0,1].

Here's a variant procedure: Starting with an urn containing at least one white ball and at least one black ball, repeat the following operation: Draw a ball, put it back, and draw a ball again (so that the second draw is independent of the first draw, and in particular it's possible that we drew the same ball both times). If we drew the same color ball on both draws, add two balls of that color (the one we just removed plus a new one); otherwise just put the ball back, restoring the urn to its previous condition.

Thus, if the urn currently contains a white balls and b black balls, the operation adds a white ball with probability $[a/(a+b)]^2$, adds a black ball with probability $[b/(a+b)]^2$, and does nothing with probability $1 - [a/(a+b)]^2 - [b/(a+b)]^2 = 2ab/(a+b)^2$.

In the third case, we can keep trying again, until eventually we succeed in drawing two balls of the same color and we get to increase the number of balls in the urn by 1. When this happens, the operation adds a white ball with probability

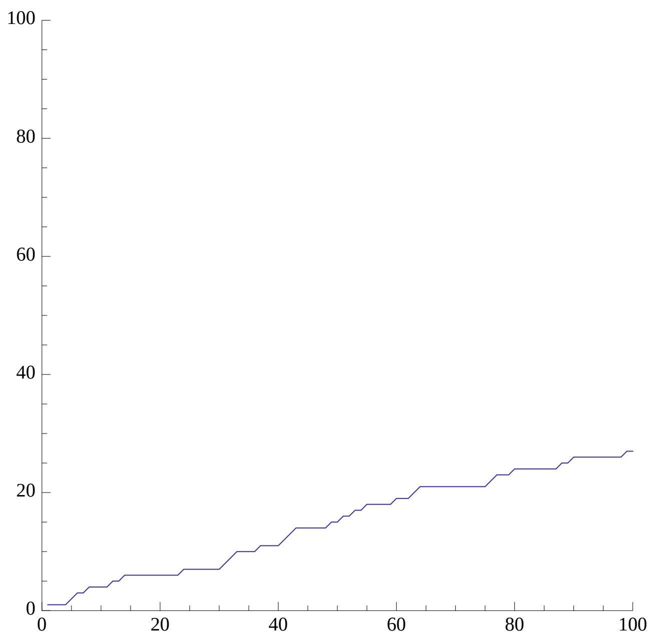
$[a/(a+b)]^2 / ([a/(a+b)]^2 + [b/(a+b)]^2) = a^2/(a^2+b^2)$

and adds a black ball with probability

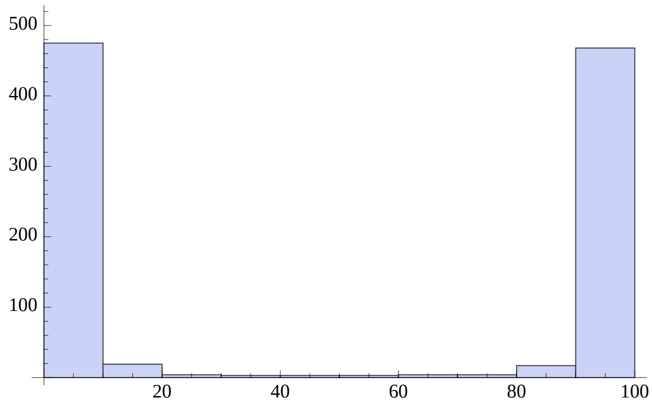
$[b/(a+b)]^2 / ([a/(a+b)]^2 + [b/(a+b)]^2) = b^2/(a^2+b^2)$.

```
VarPolya[n_] :=
 Module[{X, k, a, b}, X = Table[1, {n}]; k = 2; While[k < n, a = X[[k]]; b = k - X[[k]]; X[[k + 1]] =
    X[[k]] + RandomInteger[BernoulliDistribution[(a^2) / (a^2 + b^2)]]; k++]; Return[X]]
```

```
ListPlot[VarPolya[100], PlotRange → {{0, 100}, {0, 100}},
 AspectRatio → Automatic, Joined → True]
```



```
Histogram[Table[VarPolya[100][[100]], {1000}], 10]
```



If we let the random variable X_n be the number of white balls in the urn when the total number of balls in the urn is n, then it can be shown that X_n / n converges almost surely to 0 or 1 (each with probability 1/2).

Indeed, it can be shown that with probability 1, either the values X_n stay bounded as $n \to \infty$ or the values $n - X_n$ stay bounded as $n \to \infty$.

A cute way to prove this is to recast the discrete-time variant Polya urn process as a sequence of snapshots of a continuous time process where each color is governed by its own Poisson arrival process, with both colors evolving independently.

Over continuous time, associate with each color an arrival process where the waiting time from ball m to ball $m+1$ is exponentially distributed with parameter m^2 (mean m^{-2}). Let $a_i(t)$ (resp. $b_i(t)$) denote the number of white (resp. black) balls at time t. At any time t, the next arrival is white with probability proportional to $(a_i(t)^2)/(a_i(t)^2+b_i(t)^2)$, so if we look only at moments when a ball arrives, this discrete-time process is an instance of the variant Polya process introduced above. The expected time until the urn acquires N new white balls is

$$\sum_{m=1}^{N} m^{-2}.$$

Indeed, with probability 1 there will come a time when the urn contains infinitely many white balls, and the time T_W at which this first occurs has expected value

$$\sum_{m=1}^{\infty} m^{-2} = \pi^2/6 < \infty.$$

Likewise, with probability 1 there will come a time when the urn contains infinitely many black balls, and the time T_B at which this first occurs has expected value $E(T_B) = E(T_W) = \pi^2/6$.
The event $T_B = T_W$ has probability 0, since T_B and T_W are independent and since each of them (being a sum of infinitely many independent exponentially-distributed random variables) is a continuous random variable of positive variance; so with probability 1 there will come a moment when the urn contains infinitely many balls of one color and only finitely many of the other.
In the discrete-time ball-arrival model, this means that, after a last arrival of one color, every new ball is of the other color.

More generally, one can take a variant Polya model in which the probability that the next ball is white (resp. black) is $a^\gamma/(a^\gamma+b^\gamma)$ (resp. $b^\gamma/(a^\gamma+b^\gamma)$). The standard case is $\gamma = 1$ and the variant we looked at above is $\gamma = 2$. For any $\gamma > 1$, the series $\sum_{m=1}^{\infty} m^{-\gamma}$ converges, and the same reasoning as above can be used to show that with probability 1, either the number of white balls is bounded and every subsequent ball is black or vice versa.

Brownian motion

(very loosely adapted from *Introduction to Probability Models* by Sheldon Ross, section 10.1)

Definition of Wiener process

One-dimensional Brownian motion is like one-dimensional random walk, except that the step-sizes and the time-scale on which the steps occur both go to 0 (although, as we'll see, it's important that they go to 0 at different rates).

Suppose at each Δt time-step we go either Δx to the left or Δx to the right, each with probability $\frac{1}{2}$, with successive steps being independent. Let $X(t)$ be the position of the walker at time t (so in particular $X(0) = 0$). The random variable $X(t)$ is a sum of $t/\Delta t$ steps, each with mean 0 and variance $(\Delta x)^2$, and so has mean 0 and variance $(t/\Delta t)(\Delta x)^2$.

If we let $\Delta x = \sqrt{\Delta t}$, then $(t/\Delta t)(\Delta x)^2 = (t/\Delta t)\,\Delta t = t$.

If we send Δt to 0 and apply the Central Limit Theorem, the following properties of the limiting behavior of $X(t)$ seem reasonable:

(1) For all $t \geq 0$, $X(t)$ is normal (aka Gaussian) with mean 0 and variance t.
(2) The process $\{X(t),\ t \geq 0\}$ has independent increments, in the sense that for all $t_1 < t_2 < \ldots < t_n$, the increments $X(t_n)-X(t_{n-1})$, $X(t_{n-1})-X(t_{n-2})$, ..., $X(t_2)-X(t_1)$, $X(t_1)$ are independent. In fact, each increment $X(t)-X(s)$ (with $s < t$) is normal with mean 0 and variance $t-s$.

It turns out that there is exactly one continuous-time stochastic process satisfying (1) and (2); it is called the (unit-variance) Wiener process, aka Brownian motion.

The technical construction involves a (big!) probability space Ω whose elements are function f from $[0, \infty)$ to $\mathbb{R}$ and a probability measure ("Wiener measure") on Ω such that, for each fixed $0 \leq s < t$, if we pick f from Ω in accordance with Wiener measure, the derived random variable $f(t) - f(s)$ is normal with mean 0 and variance $t - s$.

It can be shown that, with probability 1, such a random f is continuous EVERYWHERE and differentiable NOWHERE.

Gaussians are "universal", in a sense made precise by the Central Limit Theorem: if you add lots of i.i.d. random variables with finite mean and variance, the distribution of the sum looks more and more Gaussian, even if the individual summands didn't have this property. Likewise, Brownian motion is universal: if you look at all the partial sums of an infinite sequence of i.i.d. random variables with finite mean and variance and rescale time and space appropriately, you get Brownian motion in the limit.

Constructing Brownian paths

If we're only interested in graphing $f(t)$ for t in $\mathbb{Z}$, we can let $f(1) = X_1$, $f(2) = X_1+X_2$, $f(3) = X_1+X_2+X_3$, etc., where $X_1, X_2, X_3, \ldots$ are normal with mean 0 and variance 1 (recall that a Brownian process has independent, normal increments). But what if we want to know $f(t)$ for values of t not in $\mathbb{Z}$?

E.g., if we have taken $f(1) = B$, how should we pick $f(\frac{1}{2})$?

We can answer this with facts about conditional expectation of Gaussians.

Recall that, before we conditioned on the value of $f(1)$, the definition of Brownian motion told us that $f(\frac{1}{2})$-$f(0)$ and $f(1)$-$f(\frac{1}{2})$ are independent Gaussians of mean 0 and variance $\frac{1}{2}$. Let Y_1 and Y_2 denote these independent Gaussians, so that $f(0) = 0$, $f(\frac{1}{2}) = Y_1$, and $f(1) = Y_1+Y_2$. So in conditioning on $f(1)=B$ we are conditioning on $Y_1+Y_2=B$, and in trying to simulate $f(\frac{1}{2})$ we need to know "If Y_1 and Y_2 are independent Gaussians of mean 0 and variance $\frac{1}{2}$, and B is some real number, what is the conditional distribution of Y_1, given that $Y_1+Y_2=B$?"

It can be shown that the conditional distribution of Y_1 is Gaussian with mean $B/2$ (that makes intuitive sense, by symmetry) and with variance $\frac{1}{2}$. So we can pick $f(\frac{1}{2})$ to be $f(0)=0$ plus a random increment governed by a Gaussian distribution with mean $f(1)/2$ and variance $\frac{1}{2}$.

More generally, suppose we have already specified the values of $f(.)$ at points $\ldots < s < u < \ldots$, and we want to specify the value of $f(.)$ at some point t in (s,u) (it could be the midpoint $(s+u)/2$ or it could be something else). Say we have chosen $f(s)=A$ and $f(u)=B$. Then we should pick $f(t)$ to be a Gaussian with mean

$$\frac{u-t}{u-s}A + \frac{t-s}{u-s}B$$

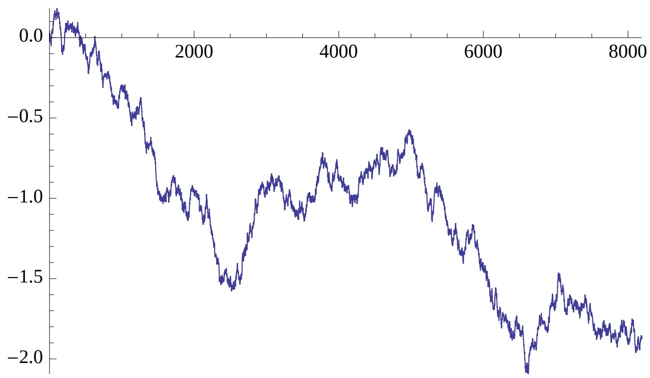
and variance

$$\frac{(u-t)(t-s)}{u-s}$$

Brownian motion on an interval (s,u), conditioned upon the endpoint values $f(s)=A$ and $f(u)=B$, is called <u>Brownian bridge</u>. Constructing a Brownian motion by repeatedly subdividing intervals via Brownian bridges is called the <u>Levy construction</u> of Brownian motion.

If $u = s + 1/n$ and t is the midpoint of $[s,u]$, then we should pick $f(t)$ to be a Gaussian with mean $(f(s)+f(u))/2$ and variance $1/(4n)$.

When we try it on the interval [0,1], we get functions that look like this:



More on this next time!