Inference, Models and Simulation for Complex Systems CSCI 7000-001
Lecture 1 25 August 2011
Prof. Aaron Clauset

1 The Poisson process

1.1 Introduction

Suppose we have a stochastic system in which events of interest occur independently with small and constant probability q (thus, the events are iid).

This kind of process is called a Poisson process (or a homogeneous Lévy process, or a type of memoryless Markov process).[1] It is also called "pure-birth" process, and is the simplest of the family of models called "birth-death" processes. The name Poisson comes from the French mathematician Siméon Denis Poisson (1781–1840).

Examples might include

- the number of hikers passing some particular trailhead in the foothills above Boulder,
- a "death" event, e.g., of a computer program, an organism or a social group,
- the arrival of an email to your inbox.

Of course, these are probably not well modeled as Poisson processes: hikers tend to appear at certain times of day; computer processes can have complicated internal structures which deviate from the iid assumptions; and emails are generated by people and people tend to synchronize and coordinate their behavior in complicated ways.

The good thing about such a simple model, however, is that we can calculate (and simulate) many properties of it, and it's a good starting place to try to understand which assumptions to relax in order to get more realistic behavior. We'll start with two:

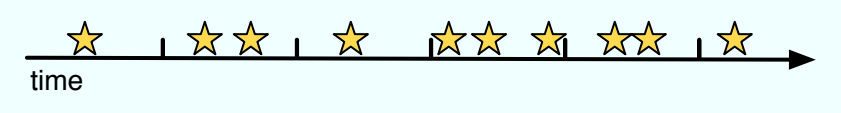
1. the distribution of "lifetimes" or delays between individual events, and
2. the distribution of the number of events observed within a given time window.

There are several ways to do these calculations; we'll use the *master equation* approach to study the continuous-time case. Other approaches, including the discrete case (see Section 4), yield equivalent results. At the end of the lecture notes, we'll also see that Poisson processes are simple to simulate, which is a nice way to explore their properties.

[1] To see why it's a Markov process, consider a two-state system, in which one state A corresponds to "no event" and the other B to "event." The transition probabilities are $\Pr(A \to B) = q$, $\Pr(A \to A) = 1-q$ and $\Pr(B \to A) = 1$.

1.2 The continuous case

To begin:
(1) Let λ be the *arrival rate* of events (events per unit time).
(2) Let $P_x(t)$ denote the probability of observing exactly x events during a time interval t.
(3) Let $P_x(t + \Delta t)$ denote the probability of observing x events in the time interval $t + \Delta t$.
(4) Thus, by assumption, $q = P_1(\Delta t) = \lambda \Delta t$ and $1 - q = P_0(\Delta t) = 1 - \lambda \Delta t$.



For general $x > 0$ and sufficiently small Δt, this can written mathematically as

$$P_x(t + \Delta t) = P_x(t)P_0(\Delta t) + P_{x-1}(t)P_1(\Delta t) \tag{1}$$
$$= P_x(t)(1 - \lambda \Delta t) + P_{x-1}(t)\lambda \Delta t \ . \tag{2}$$

In words, either we observe x events over t and no events over the Δt *or* we observe $x - 1$ events over t and exactly one event over the Δt.

With a little algebra, this can be turned into a difference equation

$$\frac{P_x(t + \Delta t) - P_x(t)}{\Delta t} = \lambda P_{x-1}(t) - \lambda P_x(t) \ ,$$

and letting $\Delta t \to 0$ turns it into a differential equation

$$\frac{\mathrm{d}P_x(t)}{\mathrm{d}t} = \lambda P_{x-1}(t) - \lambda P_x(t) \ . \tag{3}$$

When $x = 0$, i.e., there are no events over $t + \Delta t$, the first term of Eq. (3) can be dropped:

$$\frac{\mathrm{d}P_0(t)}{\mathrm{d}t} = -\lambda P_0(t) \ .$$

This is an ordinary differential equation (ODE) and admits a solution $P_0(t) = C\mathrm{e}^{-\lambda t}$, where it can be shown that $C = 1$ because $P_0(0) = 1$.

Importantly, $P_0(t)$ is the distribution of waiting times between events, because it's the distribution of times during which no events occur. When an event represents the "death" of an object, this is the distribution of object lifetimes and is our first result.

To get our second result, take Eq. (3), set $x = 1$ and substitute in our expression for $P_0(t)$.

$$\frac{\mathrm{d}P_1(t)}{\mathrm{d}t} = \lambda \mathrm{e}^{-\lambda t} - \lambda P_1(t) \ .$$

This gives a differential equation for $P_1(t)$. Solving this (another ODE) yields a solution $P_1(t) = \lambda t \mathrm{e}^{-\lambda t}$ (with boundary condition $P_1(0) = 0$). For general x, it yields

$$P_x(t) = \frac{(\lambda t)^x}{x!} \mathrm{e}^{-\lambda t} \ . \tag{4}$$

Finally, to get the probability of observing exactly x events per unit time, take $t = 1$ in Eq. (4). This yields

$$P_x = \frac{\lambda^x}{x!} \mathrm{e}^{-\lambda} \ , \tag{5}$$

which is the Poisson distribution, our second result.

1.3 The discrete case

We won't solve the entire discrete case, but we'll point out a few important connections. First, for a finite number of trials n where the probability of an event is q, the distribution of the number of events x is given by the binomial distribution

$$\Pr(X = x) = \binom{n}{x} q^x (1-q)^{n-x} \tag{6}$$

When q is very small, the binomial distribution is approximately equal to the Poisson distribution.

$$\Pr(X = x) = \frac{(qn)^x}{x!} \mathrm{e}^{-(qn)} \ , \tag{7}$$

where $\lambda = qn$. (Showing this yourself is a useful mathematical exercise. Hint: remember from calculus that $\lim_{n\to\infty}(1 - \lambda/n)^n = \mathrm{e}^{-\lambda}$.)

The distribution of lifetimes follows the discrete analog of the exponential distribution, which is called the geometric distribution

$$\begin{aligned} \Pr(X = x) &= 1 - (1-q)^x \\ &\approx \mathrm{e}^{-qx} \end{aligned} \tag{8}$$

2 The exponential distribution and maximum likelihood

Suppose now that we observe some empirical data on object lifetimes, i.e., we observe the waiting times for a series of rare events. If we assume that the data were generated by a Poisson-type process, how can we infer the underlying parameter λ directly from the observed data?

To do this, we'll introduce a technique called *maximum likelihood*, which was popularized by R. A. Fisher in the early 1900s, but actually first used by notables like Gauss and Laplace in the 18th and 19th centuries.

Recall that the (continuous) exponential distribution for the interval $[x_{\min}, +\infty)$ has the form

$$\Pr(x) = \lambda \mathrm{e}^{-\lambda(x - x_{\min})} \ . \tag{9}$$

(Note that when $x_{\min} \to 0$, we recover the classic exponential distribution $\Pr(x) = \lambda \mathrm{e}^{-\lambda x}$.) Now, let $\{x_i\} = \{x_1, x_2, \ldots, x_n\}$ denote our observed lifetime data. The likelihood of these data under the exponential model is defined as

$$\begin{aligned} \mathcal{L}(\vec{\theta} \,|\, \{x_i\}) &= \prod_{i=1}^{n} \Pr(x_i \,|\, \vec{\theta}) \\ \mathcal{L}(\lambda \,|\, \{x_i\}) &= \prod_{i=1}^{n} \lambda \mathrm{e}^{-\lambda(x_i - x_{\min})} \ , \end{aligned}$$

where we substitute the particular model parameter λ for the generalized parameter $\vec{\theta}$ once we substitute the particular probability distribution for the model we're studying. (NB: This step is entirely general and only requires assuming that your data are iid.)

Our goal now is to find the value of λ, denoted $\hat{\lambda}$, that *maximizes* this expression. Equivalently, we can find the value that maximizes the logarithm of the expression. (This works because the log is a monotonic function, and thus doesn't move the location of the maximum.) Thus,

$$\begin{aligned} \ln \mathcal{L}(\lambda \,|\, \{x_i\}) &= \ln \prod_{i=1}^{n} \lambda \mathrm{e}^{-\lambda(x_i - x_{\min})} \\ &= \sum_{i=1}^{n} \ln\Big(\lambda \mathrm{e}^{-\lambda(x_i - x_{\min})}\Big) \\ &= \sum_{i=1}^{n} \ln \lambda + \ln \mathrm{e}^{-\lambda(x_i - x_{\min})} \end{aligned}$$

$$
\begin{aligned}
&= \sum_{i=1}^{n} \ln\lambda - \lambda(x_i - x_{\min}) \\
&= n\ln\lambda - \lambda\sum_{i=1}^{n}(x_i - x_{\min}) \\
\ln\mathcal{L}(\lambda \,|\, \{x_i\}) &= n\left(\ln\lambda + \lambda x_{\min}\right) - \lambda\sum_{i=1}^{n} x_i \ . \qquad (10)
\end{aligned}
$$

Eq. (10) is the *log-likelihood function* for the exponential distribution and is useful for a wide variety of tasks. It appears in Bayesian statistics, frequentist statistics, machine learning methods, etc., and tells us just about everything we might like to know about how well the model $\Pr(x_i)$ fits the data. When it can be written down and analyzed exactly, as in this case, we can calculate many useful things directly from the log-likelihood function. When its form is too complicated to work with analytically, we can still often use numerical methods like Markov chain Monte Carlo (MCMC) algorithms to calculate what we want.

Fortunately, the exponential distribution is simple, and we may calculate analytically the value $\hat{\lambda}$ that maximizes the likelihood of our observed data. Recall from calculus that we can do this by taking derivatives. When the log-likelihood function is not simple, taking derivatives may not be possible, and we may need to use numerical methods to find the location of the maximum (see the Nelder-Mead method, also called the "simplex" method, among many other techniques).[2]

$$
\begin{aligned}
0 &= \frac{\partial}{\partial\lambda}\ln\mathcal{L}(\{x_i\} \,|\, \lambda) \\
0 &= \frac{\partial}{\partial\lambda}\left(n\ln\lambda - \lambda\sum_{i=1}^{n}(x_i - x_{\min})\right) \\
0 &= \frac{n}{\hat{\lambda}} - \sum_{i=1}^{n}(x_i - x_{\min}) \\
\hat{\lambda} &= 1 \Bigg/ \frac{1}{n}\sum_{i=1}^{n}(x_i - x_{\min}) = \frac{1}{\langle x_i - x_{\min}\rangle} \ . \qquad (11)
\end{aligned}
$$

Eq. (11) is called the *maximum likelihood estimator* (MLE) for the exponential distribution.

[2] It will be useful to use the Nelder-Mead or some other numerical maximizer in the first problem set, when you're working with maximizing non-trivial log-likelihood functions. In Matlab, look up the function `fminsearch` and recall that maximizing a function $f(x)$ is equivalent to minimizing the function $g(x) = -f(x)$. A less elegant but often sufficient approach is to use a "grid search," in which you define a vector of candidate values at which you'll evaluate $f(x)$, and then let $\hat{x}$ be the one that yields the maximum over that grid of points. The finer the grid, the longer the computation time, but the more accurate the estimate of the maximum's location.

2.1 Nice properties of maximum likelihood

The principle of *maximum likelihood* is a particular approach to fitting models to data, which says that for a parametric model[3] the best way to choose the parameters $\vec{\theta}$ is to choose the ones that maximize the probability that the model generates precisely the data observed. That is, we want to calculate the probility $\Pr(\theta \,|\, \{x_i\})$ of a particular value of θ given the observed data $\{x_i\}$, which is related to $\Pr(\{x_i\} \,|\, \theta)$ via Bayes' law:

$$\Pr(\theta \,|\, \{x_i\}) = \Pr(\{x_i\} \,|\, \theta) \frac{\Pr(\theta)}{\Pr(\{x_i\})}$$

The probability of the data $\Pr(\{x_i\})$ is fixed because the data we have do not vary with the calculation. And, in the absence of other information, we conventionally assume that all values of θ are equally likely, and thus the prior probability $\Pr(\theta)$ is uniform, i.e., a constant independent of θ. This implies $\Pr(\theta \,|\, \{x_i\}) \propto \Pr(\{x_i\} \,|\, \theta)$. Because we typically work with the logarithm of the likelihood function, these two distributions are equal to within an additive constant. This implies that the location of the maximum of one coincides with the location of the maximum of the other, and maximizing the log-likelihood will yield the correct result.

Parameter estimates derived using the maximum likelihood principle and can be shown to have many nice properties. One of the most important is that of *asymptotic consistency*, in which as $n \to \infty$, $\hat{\theta} \to \theta$ almost surely. In other words, if the model we are fitting is the true generative process for our observed data, then as we accumulate more and more of that data, our sample estimates of the parameters converge on the true values. We revisit this property in the problem set.

Likelihood function can also be used to derive an estimate of the uncertainty or standard error in our parameter estimate, so that when we report our parameter estimate using real data, we say $\hat{\theta} \pm \hat{\sigma}$. It can be shown that the variance in the maximum likelihood estimate $\hat{\sigma}^2 = 1/I(\theta)$ where $\partial^2 \mathcal{L}(\hat{\theta})/\partial\theta^2 \to I(\theta)$, and $I(\theta)$ is the *Fisher Information* at θ. (The Fisher Information basically captures the width of the curvature of the likelihood function at the maximum; the more narrow the function, the more certain our estimate.) For the exponential distribution, it's not hard to show that $\hat{\sigma} = \hat{\lambda}/\sqrt{n}$. (Doesn't this look familiar? Recall Lecture 0.)

[3] Models are "parametric" if they have free parameters that need to be estimated, which we typically denote as $\vec{\theta}$. Non-parametric models are an important class of models in modern statistics that (kind of) have no free parameters. Perhaps the best known example of a non-parametric model is a "spline". We will not cover non-parametric models directly in the class, but an excellent modern introduction to them is *All of Nonparametric Statistics* by Larry Wasserman.

3 Simulations

A Poisson process is easy to simulate numerically, especially in the discrete case. Here's some Matlab code that does this and generates the results shown in Figure 1.

```
n = 10^3;  q = 5/n;  lambda = q*n;
r=(1:20)';

x = zeros(length(r),1);     % analytic Poisson distribution
x(1) = exp(-lambda)*lambda; % constructed via tail-recursion
for i=2:length(r)           %
    x(i) = x(i-1)*lambda/i; %
end;

M = rand(n,n)<q;            % n trials, each with n coin tosses
y = sum(M);                 % compute counts of events per trial
h = hist(y,(1:20))./n;      % convert counts into a histogram

figure(1);
g=bar((1:20),h); hold on;
plot(r,x,'ro','MarkerFaceColor',[1 0 0],'MarkerSize',8);  hold off;
set(g,'BarWidth',1.0,'FaceColor','none','LineWidth',2);
set(gca,'FontSize',16,'XLim',[1/2 17],'XTick',(1:2:20),'YLim',[0 0.22]);
ylabel('Proportion','FontSize',16);
xlabel('Number','FontSize',16);
k=legend('\lambda=5, n=1000','Expected'); set(k,'FontSize',16);

z = zeros(n,1);             % tabulate time-to-first event
for i=1:n                   % for each trial
    if sum(M(:,i))>0, z(i) = find(M(:,i)==1,1,'first'); end;
end;
z(z==0) = [];               % clear out instances where nothing happened

figure(2);
semilogy(sort(z),(length(z):-1:1)./length(z),'k-','LineWidth',2); hold on;
semilogy(sort(z),exp(-q*sort(z)),'r--','LineWidth',2); hold off;
set(gca,'FontSize',16);
xlabel('Waiting time, t','FontSize',16);
ylabel('Pr(T\geqt)','FontSize',16);
k=legend('\lambda=5, n=1000','Expected'); set(k,'FontSize',16);
```

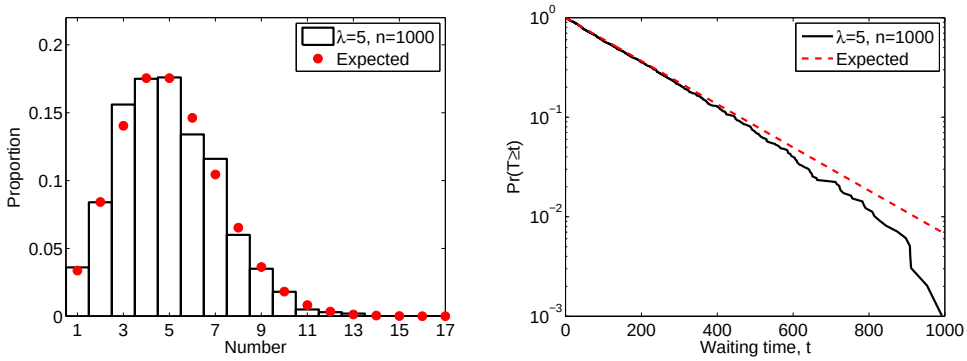
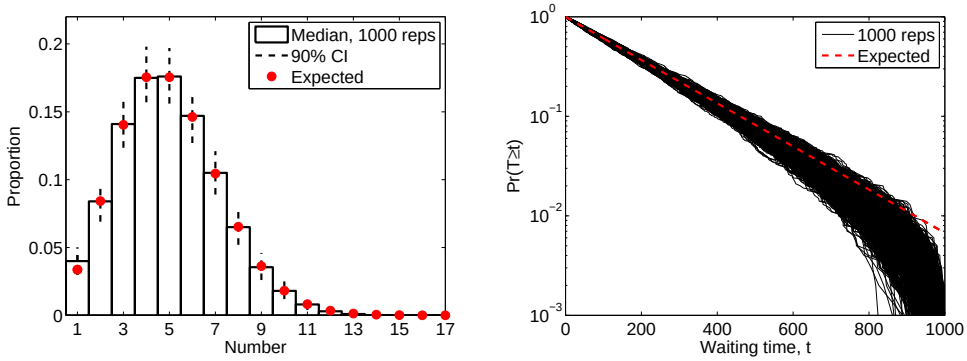


Figure 1: (A) The distribution for $n = 1000$ trials of a Poisson process with $\lambda = 5$, along with the expected counts for such a process, from Eq. (5). (B) The waiting-time distribution for the delay until the first event, for the same trials, along with the expected distribution.

If we apply our MLE to these data, we find $\hat{\lambda} = 0.0051$, which is very close to the true value of $q = 0.0050$. (NB: I'm abusing my notation a little here, by mixing λ and q.)

Notice that the observed counts (Fig. 1a) tend to deviate a little from the expected counts. Since the counts are themselves random variables, this is entirely reasonable. But, how much deviation should we expect to observe when we observe data drawn from a Poisson process?

Repeating the simulation m times, we can estimate a distribution for each count and put error bars on the expected values. Figure 3a shows the results for the counts, and Fig. 3b shows the variation in the distribution of waiting times. Note that this distribution bends downward close to $t = 1000$. This is because of a finite-size effect imposed by flipping only 1000 coins for each trial.



4 Alternative derivation of Poisson distribution

Consider a process in which we flip a biased coin, where the probability that the coin comes up 1 is q (an event occurs) and the probability of 0 is $(1-q)$ (an event does not occur). From the binomial theorem, we know that the distribution of the number of events (the number of 1s) in a long sequence of coin flips follows the binomial distribution

$$\Pr(X = x) = \binom{n}{x} q^x (1-q)^{n-x} \ , \tag{12}$$

where x is the number of events and n is the number of trials (and technically $0 \leq x \leq n$).
Recall that, by assumption, q is small. In this limit, we can simplify the binomial distribution in the following way.

To begin, rewrite $q = \lambda/n$ where λ is the expected number of events in n trials and recall from combinatorics that $\binom{n}{x} = \frac{n!}{(n-x)!\,x!}$:

$$\lim_{n\to\infty} \Pr(X = x) = \lim_{n\to\infty} \binom{n}{x} q^x (1-q)^{n-x} \tag{13}$$

$$= \lim_{n\to\infty} \frac{n!}{(n-x)!\,x!} \left(\frac{\lambda}{n}\right)^x \left(1 - \frac{\lambda}{n}\right)^{n-x} \tag{14}$$

$$= \lim_{n\to\infty} \frac{n!}{(n-x)!\,x!} \left(\frac{\lambda}{n}\right)^x \left(1 - \frac{\lambda}{n}\right)^{n} \left(1 - \frac{\lambda}{n}\right)^{-x} \ . \tag{15}$$

This form is convenient because we can use a basic equality from calculus

$$\lim_{n\to\infty} \left(1 - \frac{\lambda}{n}\right)^n = \mathrm{e}^{-\lambda} \ . \tag{16}$$

which allows us to simplify the second-to-last term in Eq. (15):

$$\lim_{n\to\infty} \Pr(X = x) = \lim_{n\to\infty} \frac{n!}{(n-x)!\,x!} \left(\frac{\lambda}{n}\right)^x \mathrm{e}^{-\lambda} \left(1 - \frac{\lambda}{n}\right)^{-x} \ . \tag{17}$$

Notice also that the last term is going to 1 because x is some constant, while $n \to \infty$. Thus, we can drop the last term, which yields

$$\lim_{n\to\infty} \Pr(X = x) = \lim_{n\to\infty} \frac{n!}{(n-x)!\,x!} \left(\frac{\lambda}{n}\right)^x \mathrm{e}^{-\lambda} \tag{18}$$

$$= \lim_{n\to\infty} \left(\frac{n!}{(n-x)!\,n^x}\right) \frac{\lambda^x}{x!} \mathrm{e}^{-\lambda} \tag{19}$$

Note that the left-hand term $\to 1$. This can be seen by observing that

$$\lim_{n\to\infty} \frac{n!}{(n-x)!\, n^x} = \lim_{n\to\infty} \frac{n}{n}\frac{(n-1)}{n}\frac{(n-2)}{n}\cdots\frac{(n-x+1)}{n} \tag{20}$$

$$= 1 \cdot 1 \cdot 1 \cdots 1 \tag{21}$$

for constant $x \geq 1$, where we apply the limit to each term individually (this is allowed because there are a finite number of terms). Thus, we have our main result, the Poisson distribution:

$$\Pr(X = x) = \frac{\lambda^x}{x!}\mathrm{e}^{-\lambda} \ . \tag{22}$$

An easy way to derive the distribution of waiting times between events for a Poisson process is to recall that λ is the expected number of events per *unit* time. Thus, if we rescale $\lambda \to \lambda t$, we have the number of events over some time span t. Setting $x = 0$ lets us consider waiting at least t time units see the first event. This yields

$$\begin{aligned}\Pr(X = 0, T > t) &= \frac{(\lambda t)^0}{0!}\mathrm{e}^{-\lambda t} \\ &= \frac{(\lambda t)^0}{0!}\mathrm{e}^{-\lambda t} \\ &= \mathrm{e}^{-\lambda t} \ .\end{aligned}$$

To get the distribution for waiting exactly t time units, we now simply differentiate with respect to time the expression $1 - \Pr(X = 0, T > t)$, which yields the exponential distribution $P(T = t) = \lambda\mathrm{e}^{-\lambda t}$.