Inference, Models and Simulation for Complex Systems — CSCI 7000-001
Lecture 6 — 13 September 2011
Prof. Aaron Clauset

1 Time series

In earlier lectures, we've relied on a crucial assumption: our observed data are drawn iid from our underlying generative process. This implies that the order of our observations $x_1, x_2, \ldots, x_n$ is unimportant and mathematically, it implies that any permutation of $\{x_n\}$ would have the same statistical properties as the original ordering.[1] For instance, when we draw data from a distributional model like the power-law distribution, e.g., using the transformation method from Problem Set 1, the data are iid. For a process to be iid in this way, it must satisfy the standard independence condition, i.e.,

$$\Pr(x_t, x_{t+\delta}) = \Pr(x_t)\ \Pr(x_{t+\delta}) \ , \tag{1}$$

for all choices of δ and t.

Most real-world process, however, do not generate completely independent events over time, and the order of events contains important information about the underlying generative process. Sometimes, the correlations are sufficiently small that they can be ignored, but for complex systems, this is often not the case and the sequential correlations must be accommodated.

A *time series* is an ordered set of events $\{x_t\}$ where t indexes the order of observations and increases monotonically. This sequence is often represented as a list of pairs ordered (t_i, x_i), for instance,

$$\{(t_1, x_1), (t_2, x_2), (t_3, x_3), (t_4, x_4), \ldots\} \tag{2}$$

Most standard time-series analysis techniques[2] assume a homogeneous temporal process, i.e., t increases a constant rate and the variable of interest x takes a value at each moment in time. Figure 1 shows two empirical examples of such time series, using data on mass violence. In each case, the variable of interest is defined and time proceeds monotonically. Unlike the case of iid random variables, a time series random variable exhibits some amount of sequential correlation. That is, a succeeding value x_t is highly correlated with the preceding value x_{t-1}.

Techniques like auto-correlation functions, cross-correlation functions, auto-regressive models, etc. all rely on the assumption of homogeneous time. Some complex systems, however, do not exhibit

[1] In the modern parlance of statistics, we could call these models *exchangeable*, meaning that any ordering is freely interchangeable with any other ordering. In other words, the future is just like the past.

[2] For instance, see J.D. Hamilton, *Time Series Analysis*. Princeton University Press (1994).

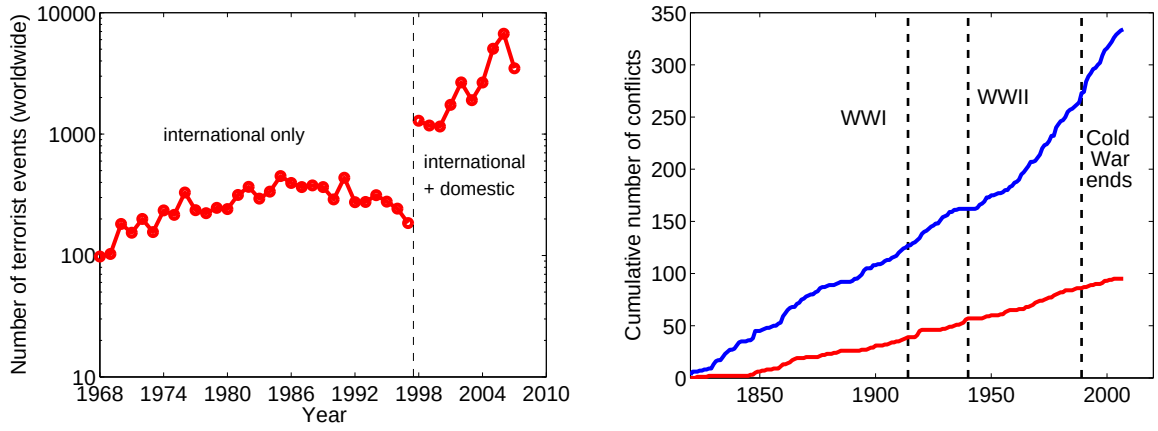

Figure 1: Examples of time series. (a) The number of terrorist attacks (deadly and non-deadly) recorded in the National Memorial Institute for the Prevention of Terrorism's Terrorism Knowledge Base, which collects events from media reports worldwide. (b) The cumulative number of wars (civil in blue; international in red) worldwide recorded by the Correlates of War project.

this kind of structure and are instead event driven. That is, at each moment of time, there can be any of $k \in \{0, 1, \ldots, \text{inf}\}$ events, with $k = 0$ predominating. In this case, different time-series analysis techniques are required. For instance, suppose we increased the temporal resolution for either of our mass violence time series so that $\Delta t \approx 1$ hour. Time still proceeds monotonically, but now the time series variable is dominated by zeros: most hours in most days do not see the appearance of any new terrorist attacks or new wars.[3]

A general temporal generative process is one in which $x_{t+\mathrm{d}t} = f(\{x_t\}, \theta)$. That is, the value of the next observation is an arbitrary function of the entire history of the system $\{x_t\}$ and possibly some set of parameters θ. This is not a very useful statement, however, either for inference or for modeling. More typically, time-series analysis assumes some amount of conditional independence, i.e., that Eq. (1) holds for all values of $\delta > t_c$ where t_c represents the correlation length or *memory* of the system. If t_c is very large, or if the "weight" of distant events is small but non-zero, then a system is said to have a *long memory*.

[3]One trick to convert such event-driven time series into the nicer homogeneous forms is to, as I did with the wars data, convert to a cumulative variable. The slope of such a time series can be interpreted as the average *production rate* of new events. When the process is largely stationary, as for international wars (red line), this works quite well; non-stationarities, as in the change in production rate for civil conflicts (blue line) around WWII, complicate things somewhat.

1.1 Comments on stationarity, ergodicity and equilibrium

What makes many complex systems interesting, and thus difficult to study using conventional techniques, is a property called *non-stationarity*, which basically means that the future evolution of a system is not exactly the same as its past history. This idea is well summed up by the aphorism, sometimes attributed to Niels Bohr (1885–1962): "Prediction is extremely difficult. Especially about the future. "

What's surprising is that many social, biological, physical and technological complex systems do not exhibit arbitrarily complicated temporal dependences, and many even seem to exhibit some amount of stationarity or at least conditional independence. Identifying where this structure lies often requires genuine insight and understanding, but there are tools that can facilitate that process.

Most time-series analysis techniques assume stationary underlying processes because it makes everything simpler. (Just as assuming iid random variables makes analyzing empirical data simpler.) Many further assume *ergodicity*, which means that every possible state of the system is accessible (even with very small probability) from every other state. This property implies the existence of a single stationary distribution over the system's states, and it is often convenient to assume that observed data describe that equilibrium distribution rather than some arbitrary transient distribution. These assumptions allow us to employ tools like Markov chains, which can often be analyzed mathematically.

Of course, non-stationary, non-ergodic, and/or non-equilibrium systems don't follow the nice rules we often assume, and conventional techniques may fail catastrophically when applied in these contexts. That is, like for model testing and model selection, these techniques derive their power from their mathematical assumptions, but if these assumptions are not met, some of these tools fail in bad ways.

2 Random walks

Random walks are a special kind of temporal process and can be used to model the dynamics of many complex systems. There are also deep connections between random walks, diffusion, spectral methods, synchronization, etc. and many stochastic processes or techniques can be expressed in terms of random walk dynamics. For instance, the PageRank search algorithm is a kind of model of "random surfers" on the Web graph. PageRank converts the random surfers into an ergodic system where their stationary distribution over webpages can be derived numerically. The density of walkers on a given node is then taken to be proportional to the importance of that page.

In general, random walks exhibit what's called *serial correlation*, conditional independence for fairly small values of t_c, and a simple stochastic historical dependence $f(.)$. For instance, a simple

additive random walk has the form

$$x_t = x_{t-1} + \lambda \ , \tag{3}$$

where λ is a random variable drawn iid from a distribution $\Pr(\lambda)$, called the noise or fluctuation distribution. Conventionally, fluctuations are normally (Gaussian) distribution with mean μ and variance σ^2, that is, $\Pr(\lambda) \sim N(\mu, \sigma^2)$, as this makes mathematical analysis considerably simpler. But, any probability distribution, for instance, a Laplace (exponential tails) or double-Pareto distribution (power-law tails), also works. If $\Pr(\lambda)$ is symmetric and the average fluctuation is zero $\langle\lambda\rangle = 0$, the random walk is said to be *unbiased* and $\Pr(x_t > x_{t-1}) = \Pr(x_t < x_{t-1})$.

In the above formulation, time t proceeds in discrete steps; if time is a continuous variable, then we have what's called a Wiener process, which is itself a special kind of Lévy process. These models are often used in physics and finance modeling, and a great deal is known about them mathematically. Here, we'll stick with the discrete version, which is simpler to simulate. Many of the intuitions, however, carry over to the continuous time versions.

2.1 Hypothesis tests

Simple random walk dynamics like these are sometimes used as a null model in a hypothesis test framework. For instance, Can the sequence of variables $\{x_t\}$ be explained by a trivial unbiased random walk with unknown variance, e.g., by letting $\Pr(\lambda) \sim N(0, \sigma^2)$? Note that the structure in Eq. (3) is different from a traditional iid process because it's the the differences in sequential observations that are iid, i.e., $x_t - x_{t-1} = \Delta x \sim \Pr(\lambda)$, not the observations themselves.

Thus, we can test whether some time series data could have been plausibly generated by a simple random walk by tabulating the empirical distribution of iid events, that is, the fluctuations $\Pr(\Delta x)$, and then asking whether they could plausibly have been drawn iid from $N(0, \hat{\sigma}^2)$ (where we estimate σ from our data). This is exactly the kind of test that underlies what is done in statistical genomics to identify genes under strong selection: neutral or non-selective "drift" of a set of DNA sequences is modeled as an unbiased random walk on the n-dimensional hypercube (for a gene of n base pairs), where fluctuations in the sequence are created by neutral mutations. To carry out the test, we would first need an estimate the probability of a mutation, and then we could compare the distribution of differences within our empirical sequences to the expected distribution derived from our random walk model.

2.2 Examples of random walk dynamics

Simple and sometimes even moderately less-simple random walks can be studied mathematically. Complicated random walks almost always have to be simulated. In general, however, there are a number of intuitions that it's important to have about the dynamics of random walks under different kinds of conditions or constraints. Let's go through a few.

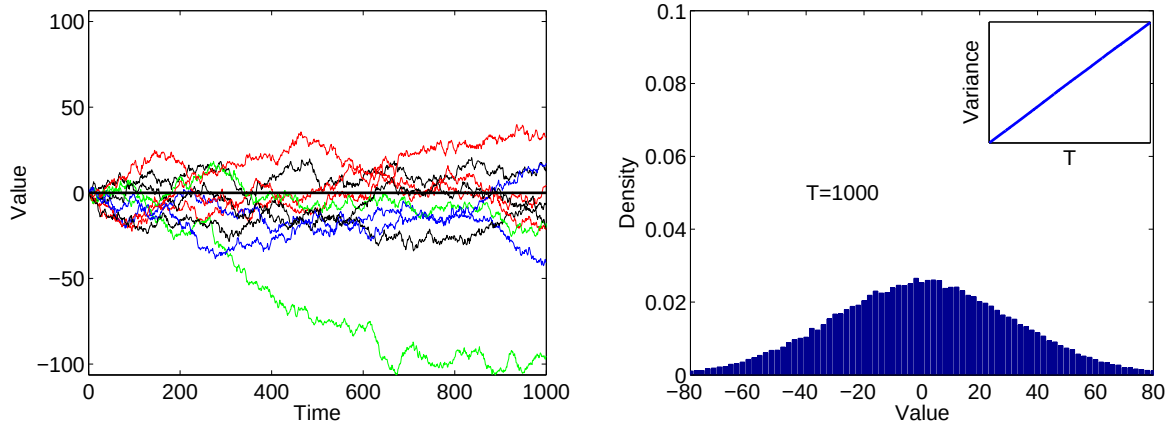

Figure 2: (a) The trajectories of 10 simple random walkers. (b) The distribution of $n = 10^5$ simple random walkers, after $T = 1000$ time steps. The inset shows the variance of the distribution over time, illustrating the linear increase in the variance with time.

2.2.1 Unbiased random walks on unbounded intervals

As a first example, consider a simple unbiased random walk where $\Pr(\lambda) \sim N(0, 1)$ and the walkers wander around on an unbounded 1-dimensional interval. In this case, almost everything about the system's dynamics and trajectories can be, and has been, calculated analytically. It's also dirt simple to simulate. Figure 1a shows the trajectories of 10 simulated random walkers, and Fig. 1b shows the distribution of 100,000 walkers. Because this process has no constraints, there's nothing to prevent the walkers wandering off to infinity and thus they do; the time-evolution of the distribution is a normal with mean zero and variance that increases linearly with time t. That is, as time progresses, the distribution gets flatter and flatter.

2.2.2 Random walks on semi-bounded intervals

If we impose a very simple constraint on the random walk so that walkers cannot cross below some lower limit $x_{\min}$, we get rather different dynamics.

In the case of an unbiased walk, there is still no stationary distribution, but the diffusion out to infinite happens much slower as the lower limit causes probability density to pile up there for a long transient period (see Fig. 2a). Introducing any bias in opposite direction of $x_{\min}$ speeds up the diffusion away to infinity. In contrast, a sufficiently large bias[4] in the direction of $x_{\min}$, i.e.,

[4]The precise size of a bias required to produce a stationary distribution depends on the variance σ^2 in the diffusion

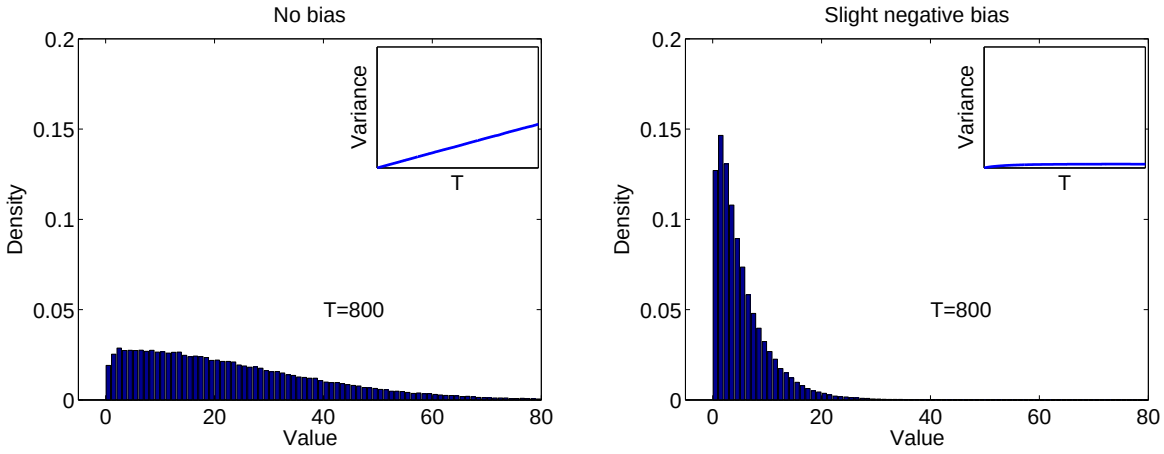

Figure 3: The distribution of $n = 10^5$ random walkers on the interval $[x_{\min}, \infty)$ with (a) no bias and (b) a slightly negative bias, after $T = 800$ time steps. For (a) the variance still grows linearly with time, implying no steady-state distribution, but much more slowly than in the unbounded case (Fig. 1b); for (b), however, the variance stabilizes, indicating that a stationary distribution of the walkers exists.

a bias that makes walkers on average move toward the boundary, causes the system to exhibit a stationary distribution of walkers pressed up against the boundary (see Fig. 2b).

2.2.3 Random walks on bounded intervals

In the previous section, we imposed a lower-boundary. If we impose both a lower and an upper boundary, that is, restrict the random walk to some bounded interval $[x_{\min}, x_{\max}]$, a stationary distribution is fairly easy to produce since there's no chance for a walker to diffuse away to infinity. In fact, it takes effort to produce a non-stationary distribution in this case. Figure 3 shows simulation results for the case of no bias and a very slight positive bias.

In both cases, the distributions are reminiscent of the water level in a glass when (a) the glass is flat on a table and (b) the glass is slightly tilted. And, this is exactly analogous: gravity acts as a bias on the diffusion of water molecules in the liquid. When the glass it flat, gravity is orthogonal to the surface and thus exerts no left-right bias on the motion of the molecules; however, when the glass it tilted, gravity is no longer orthogonal to bottom of the glass, and this effectively induces a bias in the direction of motion relative to the walls of the glass.

kernel $\Pr(\lambda)$. Too small a bias, and the variance dominates.

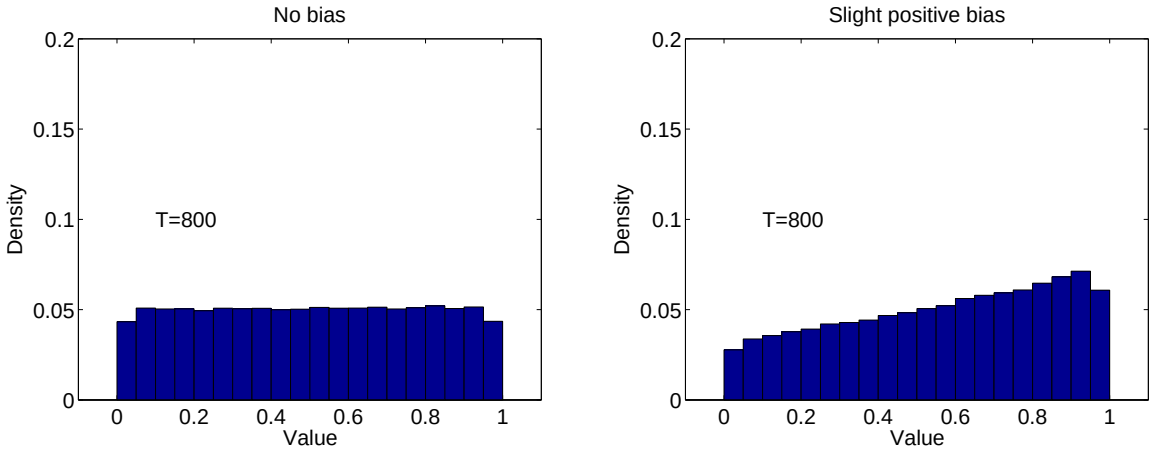

Figure 4: The distribution of $n = 10^5$ random walkers on the interval $[x_{\min}, x_{\max}]$ with (a) no bias and (b) a slightly positive bias, after $T = 800$ time steps. In both cases, a steady-state distribution exists; in the case of no bias, it's a uniform distribution over the interval. The slight dip in density at the boundaries are caused by the fact that the diffusion kernel is truncated so that, e.g., near $x_{\min}$, $\Pr(\lambda < x_{\min}/x_t) = 0$, since steps that would cross the boundary are disallowed. This means that walkers near the boundaries, on average, move away from the boundary rather than stay there.

2.2.4 Random walks with size-dependent fluctuations

Boundary conditions are not the only way to generate stationary distributions under random-walk dynamics. So far, we've only considered diffusion kernels $\Pr(\lambda)$ that are completely independent of the random walk itself. These represent the simplest of random walks. If we let the form of the diffusion kernel itself depend on the location of the random walk, we can introduce more complex behavior.

Figure 3a shows an example of such size-dependent fluctuations, plotting $\Pr(\lambda \,|\, x)$ as a function of x. In this case, I've chosen the diffusion kernel $\Pr(\lambda \,|\, x)$ to be a normal distribution with unit variance and a mean μ that varies with x according to a 3rd-order polynomial function (a cubic): $N(\mu = \sum_{i=0}^{3} a_i x^i, 1)$. Other functions work just as well, but this particular choice creates some interesting dynamics.

These dynamics are mainly generated by the three locations on the x-axis where the cubic crosses $\lambda = 0$. The outer two behave like attractors, meaning that walkers in their vicinity that take small steps away from the attractor's location tend to move back toward the attractor. For example, if some x_t is smaller than the left-most attractor, then because the average size of a step in that

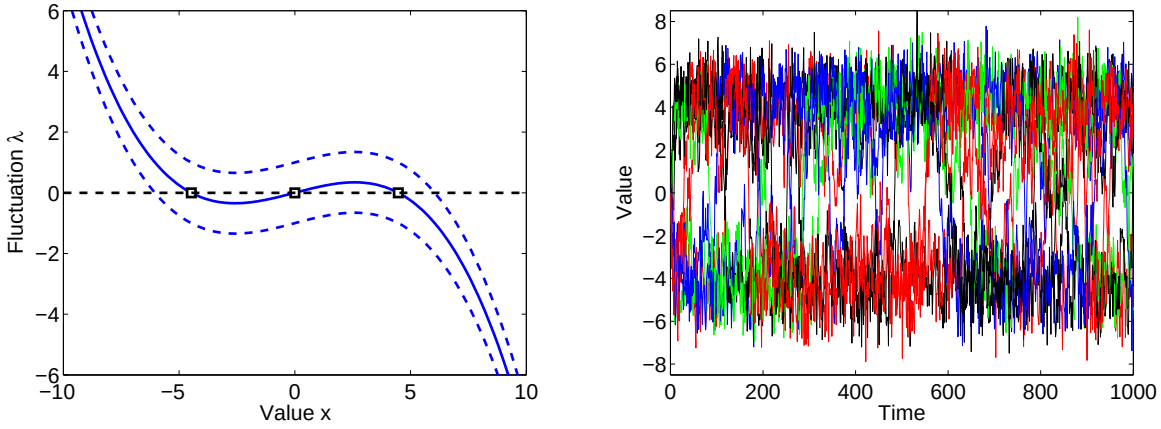

Figure 5: (a) A diffusion kernel that exhibits dependent fluctuations $\Pr(\lambda \,|\, x)$ that has a symmetric cubic structure. (b) The trajectories of 10 random walkers with the size-dependent diffusion kernel shown in (a), which resemble the dynamics of a bistable system.

regime is positive, x_{t+1} will tend to be closer to the attractor. In contrast, the inner one behaves like a repeller: small perturbations away from it tend to be amplified, e.g., a walker with x_t slightly to the left of the repeller point will, on average, take a step further to the left. In general, attractor points in a system exhibiting size-dependent fluctuations are those where the mean step size crosses $\lambda = 0$ from positive to negative; repellers are those that cross in the opposite direction.

Another interesting property of this system is that it exhibits a steady-state distribution without the need for hard boundaries like those we saw above. This happens because the cubic function's structure tends to cause very small values of x to increase, and large values of x to decrease. The net effect is that walkers are prevented from diffusing out to infinity. Fig. 4 shows the equilibrium distribution.

A last interesting feature of this dynamical system is that although it's defined as a continuous random walk, its dynamics resemble a discrete stochastic system switching between two states, which are defined by the two attractors points. The repeller point does not completely split the system into two distinct populations because with small probability, a walker can cross from one attractor to the other (and back). We see this kind of migrational behavior in Fig. 3b, which shows the trajectories of 10 random walkers. If we decreased the variance in the size-dependent fluctuations, this would reduce the transit probability between the two states; similarly, we can force a random walker to spend more time in one or the other state by making the cubic function more asymmetric.

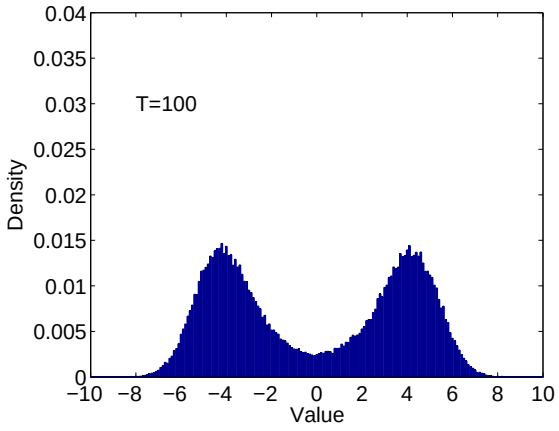

Figure 6: The steady-state distribution for the size-dependent fluctuations shown in Fig. 3.

In general, size-dependent fluctuations in a random walk can produce rich dynamics and they seem to be a pretty good model of many complex systems. The precise structure of the fluctuations, that is, the way the diffusion kernel changes its shape as a function of x, can be quite complicated. For instance, in addition to the mean showing size-dependence, so too can the variance depend on the size; larger variances at some location than others tends makes those locations behave like unstable points, while lower variances tend to concentrate random walkers there. A good analogy is hot spots an cold spots: hot spots lead to more vigorous local mixing due to increased thermal vibration, while cold spots behave like localized "traps" for diffusing particles due to decreased thermal vibration.

2.2.5 Multiplicative random walks

All of the random walks we considered above are *additive* random walks, because changes in a walker's location are added to the current location. A *multiplicative* random walk is exactly what it sounds like and the dynamics follow

$$x_t = \lambda\, x_{t-1} \ , \tag{4}$$

where λ is drawn iid from some $\Pr(\lambda)$. That is, changes in value are proportional rather than additive. Of course, if we take the logarithm of both sides of Eq. (4), we see that a multiplicative random walk is simply an additive random walk on a logarithmic scale: $\ln x_t = \ln x_{t-1} + \ln \lambda$. Unsurprisingly, many of the results for additive random walks hold exactly for multiplicative random walks.

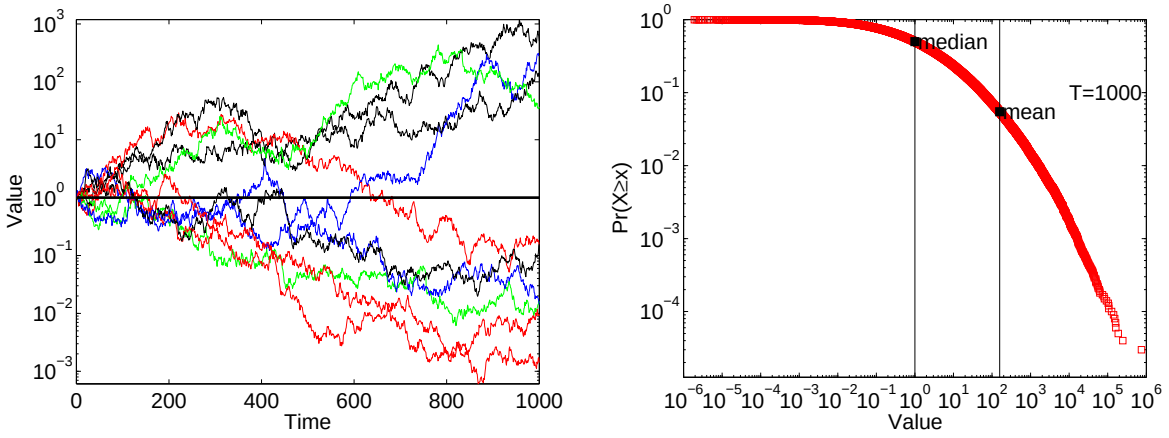

Figure 7: (a) The trajectories of 10 simple multiplicative random walkers on the interval $(-\infty, \infty)$, all of which initially started at $x_0 = 1$. (b) The distribution of $n = 10^5$ simple multiplicative random walkers, after $T = 1000$ time steps, with the mean and the median values marked.

There are, however, a few important things that are different. In particular, the behavior of the mean for a multiplicative random walk is different than the mean for an additive random walk because $\ln\langle x_t\rangle \neq \langle \ln x_t\rangle$. Figure 5 illustrates this for a simple multiplicative random walk on the interval $(-\infty, \infty)$. This is fundamentally the same process as shown in Fig. 1, which showed an additive random walk with the same conditions. Recall that in this case, when all the random walkers start at $x = 0$, the mean and the median coincided for all time t at $x = 0$. Figure 5b shows the same simulation but for a multiplicative random walk. Here, the median stays put, exactly in the middle of the distribution at $\ln x = 0$, while the mean increases without bound. It does this because the average is becoming more and more dominated by the extreme upper (right) tail of the distribution. In the additive case, very positive values were balanced, on average, by large negative values so that the average is representative of the middle of the distribution. Here, large positive values ($x > 10^0$) would be paired with small positive values ($x < 10^0$), which skews the average high.

3 Matlab code

```
% ---- unbiased random walk on an unbounded interval ---- Figure 1b
N  = 10^5;       % number of random walkers to simulate
T  = 1000;       % number of steps of walk to simulate
x  = zeros(1,N); % data structure for x_{t}; set initial condition
v  = zeros(T,1); % data structure for var(x_{t})
for i=2:T
    steps  = randn(1,N);    % steps for all N walkers
    x      = x+steps;       % take the walk
    v(i)   = var(x);        % compute the variance

    if mod(i,10)==0         % make a pretty picture every 10 time steps
        r = (-82:2:82);
        h = hist(x,r);
        figure(2); clf;
        bar(gca,r,h./N); set(gca,'YLim',[0 0.1],'XLim',[-80 80]);
        z = text(-40,0.05,strcat('T=',num2str(i))); set(z,'FontSize',16);
        set(gca,'Fontsize',16,'XTick',(-80:20:80));
        xlabel('Value','FontSize',16); ylabel('Density','FontSize',16);
        z = axes('Position',[0.63 0.63 0.27 0.27]);
        plot(z,(1:i),v(1:i),'b-','LineWidth',2);
        set(z,'YLim',[0 T],'XLim',[0 T],'FontSize',14,'XTick',[],'YTick',[]);
        xlabel('T','FontSize',16); ylabel('Variance','FontSize',16);
        drawnow;
    end;
end;
% ------------------------------------------------------
```

```
% ---- biased random walk on a semi-bounded interval ---- Figure 2b
N  = 10^5;       % number of random walkers to simulate
T  = 800;        % number of steps of walk to simulate
x  = zeros(1,N); % data structure for x_{t}; set initial condition
v  = zeros(T,1); % data structure for var(x_{t})
a  = -0.1;       % slight negative bias
for i=2:T
    steps  = a+randn(1,N);
    y      = x+steps;
    g      = y<0;
    while any(g)          % rejection sampling for step sizes (slow)
        steps(g) = a+randn(1,sum(g));
        y = x+steps;
        g = y<0;
    end;
    x    = x+steps;       % take the walk
    v(i) = var(x);        % compute the variance

    if mod(i,10)==0       % make a pretty picture every 10 time steps
        r = (0:1:85);
        h = histc(x,r);
        figure(5); clf;
        bar(0.5+r,h./N); set(gca,'YLim',[0 0.2],'XLim',[-5 80]);
        z = text(40,0.05,strcat('T=',num2str(i))); set(z,'FontSize',16);
        set(gca,'Fontsize',16);
        xlabel('Value','FontSize',16); ylabel('Density','FontSize',16);
        title('Slight negative bias','FontSize',16);
        z = axes('Position',[0.63 0.63 0.27 0.27]);
        plot(z,(1:i),v(1:i),'b-','LineWidth',2);
        set(z,'YLim',[0 T],'XLim',[0 T],'FontSize',14,'XTick',[],'YTick',[]);
        xlabel('T','FontSize',16); ylabel('Variance','FontSize',16);
        drawnow;
    end;
end;
% -------------------------------------------------------
```

```
% ---- size-dependent fluctuations on a unbounded interval ---- Figure 4
N   = 10^5;       % number of random walkers to simulate
T   = 100;        % number of steps of walk to simulate
x   = zeros(1,N); % data structure for x_{t}; set initial condition
a  =  0;          % size dependence
b  =  0.20;       % linear    term
c  =  0;          % quadratic term
d  = -0.01;       % cubic     term
for i=2:T
    steps  = (a+b.*x+c.*x.^2+d.*x.^3)+randn(1,N);   % size-dependent step sizes
    x      = x+steps;                                % take those steps

    r = (-10:0.1:10);                                % make a pretty picture
    h = hist(x,r);
    figure(10); clf;
    bar(gca,r,h./N); set(gca,'YLim',[0 0.04],'XLim',[-10 10]);
    z = text(-8,0.03,strcat('T=',num2str(i))); set(z,'FontSize',16);
    set(gca,'Fontsize',16,'XTick',(-10:2:10));
    xlabel('Value','FontSize',16); ylabel('Density','FontSize',16);
    drawnow;
end;
% ---------------------------------------------------------
```

```
% ---- unbiased multiplicative random walk on unbounded interval ---- Figure 5b
N  = 10^5;       % number of random walkers to simulate
T  = 1000;       % number of steps of walk to simulate
x  = ones(1,N);  % data structure for x_{t}; set initial condition
s  = 0.1;
for i=2:T
    steps  = exp(s.*randn(1,N));  % steps for all N walkers
    x      = x.*steps;            % take the multiplicative walk

    if mod(i,20)==0               % make a pretty picture every 20 time steps
        a = sum(x>=mean(x))./N;
        figure(5);
        loglog(sort(x),(N:-1:1)./N,'rs'); hold on;
        loglog(mean(x),a,'ks','LineWidth',2,'MarkerFaceColor',[0 0 0]);
        loglog(median(x),0.5,'ks','LineWidth',2,'MarkerFaceColor',[0 0 0]);
        loglog(median(x).*[1 1],10.^[-5 0],'k-');
        loglog(mean(x).*[1 1],10.^[-5 0],'k-'); hold off;
        set(gca,'YLim',[10^-4.9 1],'XLim',10.^[-6 6],'XTick',10.^(-6:6));
        z = text(10^4,0.1,strcat('T=',num2str(i))); set(z,'FontSize',16);
        z = text(1.2*mean(x),a,'mean'); set(z,'FontSize',16);
        z = text(1.2*median(x),0.5,'median'); set(z,'FontSize',16);
        set(gca,'Fontsize',16,'XScale','log');
        xlabel('Value','FontSize',16); ylabel('Pr(X\geqx)','FontSize',16);
        drawnow;
    end;
end;
% -------------------------------------------------------
```