Lecture 9: Random Walk Models

Steven Skiena

Department of Computer Science
State University of New York
Stony Brook, NY 11794–4400

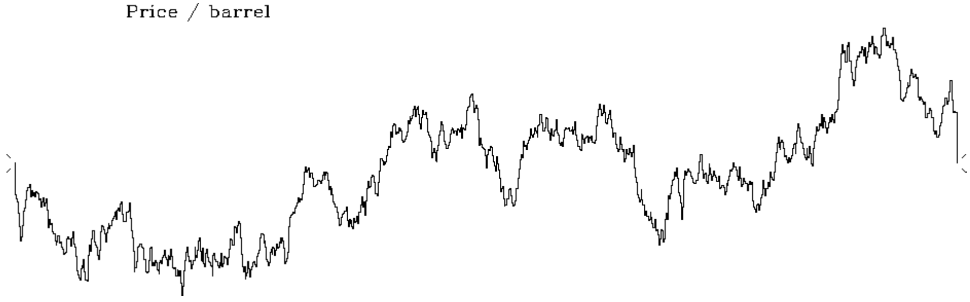
http://www.cs.sunysb.edu/∼skiena

Financial Time Series as Random Walks

J. P. Morgan’s famous stock market prediction was that “Prices will fluctuate.”

Bachelier’s *Theory of Speculation* in 1900 postulated that prices fluctuate randomly.

Indeed, simple random processes can generate time series which closely resemble real financial time series.



Why Random Price Changes?

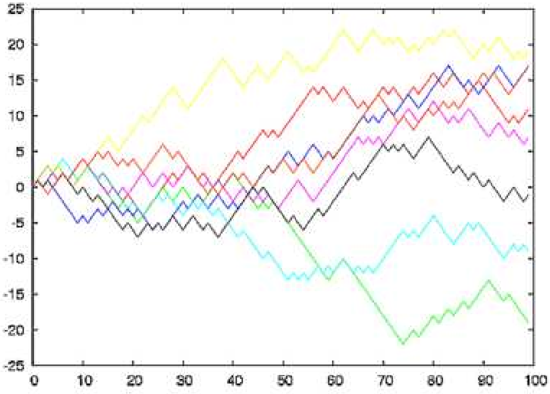
Random price movements makes sense in a world where

- Most price changes result from temporary imbalances between buyers and sellers,
- Stronger price shocks are inherently unpredictable, and
- The efficient market hypothesis, where the current price of a stock reflects all information about it.

If the next price movement was predictable, the market would have previously predicted it, leaving only random fluctuations.

Monte Carlo Methods

Monte Carlo methods use statistics gathered from random sampling to model and simulate a complicated distribution.



Generating random walks with the properties of financial time series and analyzing the resulting price distributions is a powerful technique in options pricing.

Models and Randomness

Models are useful when they have predictive power.

Simple models which capture large-scale phenomena are useful.

The model that "the earth is flat" is a very useful model.

If "the future" is the random selection of one sequence of events from a probability space, our focus should revolve around modeling the distribution more so than identifying the ultimate future path.

Random walk models give us a simple but powerful tool to model financial price distributions.

Binomial Random Walks

In a simple discrete random walk model, each step we move a distance of 1 either up or down, with the probability p of an upward move and $1-p$ of a downward move.

The path we take as a function of such moves defined a random walk, and is akin to flipping a biased coin.

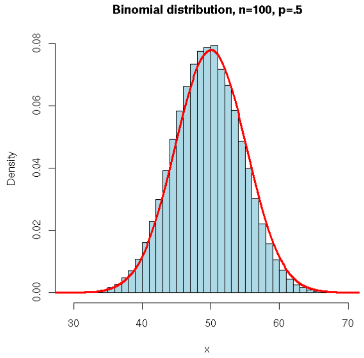
The probability that we get exactly h heads and $n-h$ tails with a coin that comes up heads with probability p is

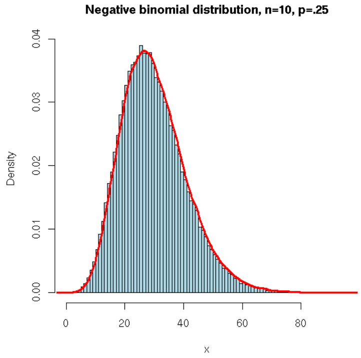
$$Pr(h,n,p) = \binom{n}{h} p^h (1-p)^{n-h}$$

Closed forms eliminate the need to simulate random walks for computing the underlying distribution, but typically exist only for measuring very simple properties of very simple walk models.

Mean and Variance

The expected number of heads on a binomial distribution is np, with variance $np(1-p)$.





For an unbiased coin ($p = 0.5$), the expected difference between heads and tails is 0, but the expected *absolute difference* between heads and tails is $O(\sqrt{n})$.

Additive Random Walks

A time series $\{V_t\}$ is a *additive* random walk if

$$V_t = V_{t-1} + a_t$$

where a_t is a random variable.
The binomial walk above is an example of an additive walk.
Selecting a_t from a *normal distribution* defines a continuous random walk model.
Price series that tend to increase with time can be modeled as a *random walk with drift*:

$$V_t = \mu + V_{t-1} + a_t$$

Multiplicative Random Walks

A time series $\{V_t\}$ is a *multiplicative* random walk if

$$V_t = a_t V_{t-1}$$

where a_t is a random variable.
The simplest discrete case is when $a_t = c$ with probability p and $a_t = 1/c$ with probability $(1-p)$.
Up-moves still cancel down-moves, but the changes are fixed *percentages* instead of fixed *values*, so they stay equally significant as V_t grows.

Logarithmic Returns

Observe that

$$V_t = a_t V_{t-1} \longrightarrow \ln V_t = \ln a_t + \ln V_{t-1}$$

Additive random walks make sense to model a financial time series only when modeling changes in the log price or returns, since otherwise the impact of $\{a_t\}$ will diminish with time.

Caveats of Random Walk Models

Random walk models of prices does not make sense if you believe in technical analysis, where it assume the price trajectory to this point offers insight into the future.

One downside of conventional random walk models is that they predict returns as being normally or log-normally distributed.

Such distributions tend to underestimate the frequency of extreme events.

Still, random walks can be very useful in modeling financial risks and returns.

Applications of Random Walks

Estimating the probability distribution for the price of a stock at a given future time t is critical to pricing certain options.

This probability distribution can be modeled as the distribution of positions after $f(t)$ steps of a random walk.

Simulations of random walks enable one to compute the probabilities of more complicated events...

Suppose you want to know the probability that a stock will hit a given price at *some* point between now and time t, or what is the expected high price reached over this interval.

Bankruptcy Prediction

Suppose you want to know the probability a company will go bankrupt at *some* point between now and time t.

We can define a company as bankrupt, say, when its capitalization falls to less than its debt minus its assets.

Random walk models can simulate the fluctuations in stock price and debt-levels. The probability of bankruptcy is estimated by the fraction of simulated runs ending in doom.

Generating Random Numbers

Simulating random walks require a source of random numbers, but truly random numbers cannot be produced by a deterministic computer.

The accuracy of a simulation depends on generating truly *pseudo-random* numbers. Would a random walk alternating up and down look like a price series?

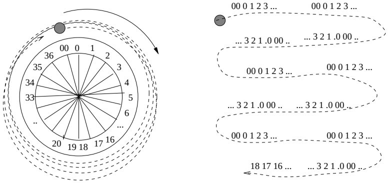
Statistical tests can measure the validity of a random number generator, but library functions *should* be good until you exceed the *period* where they start to repeat.

Linear Congruential Generators

Linear congruential generators are a reliable source of random numbers, where

$$r_n = (ar_{n-1} + c) \bmod m$$

for appropriate constants a, c, and m.



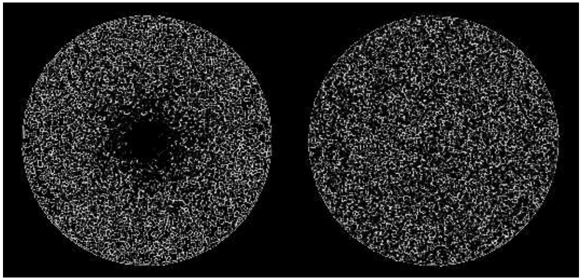
They generate random numbers by the same principle as a roulette wheel!

Non-Uniform Random Number Generation

Complications exist when generating numbers from a given, non-uniform distribution using a uniform generator.

How can you select random points in a disk?

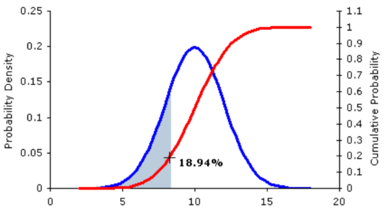
Selecting the radius and angle either uniformly or normally gives the wrong distribution!

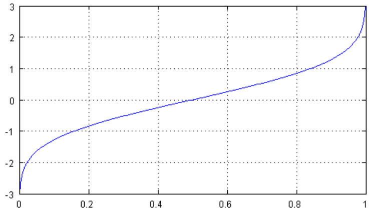


The accept-reject method generates uniform points over a volume, and discards all points outside the target region.

Generating Normal Variables

Random numbers can be generated for any distribution D by mapping uniform random numbers to the corresponding value on the inverse cummulative distribution function of D.





But performing the interpolation incurs some cost in time and accuracy…

The Box-Muller Method

A clever, efficient method to constuct random normal numbers relies on a polar transformation.
Let x_1 and x_2 be independent, uniformly generated (0,1] random variables.
Then

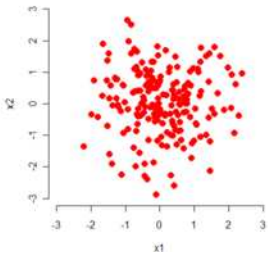
$$y_1 = \sqrt{-2\ln x_1}\cos(2\pi x_2), \text{ and}$$

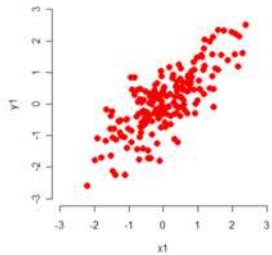
$$y_2 = \sqrt{-2\ln x_1}\sin(2\pi x_2)$$

are both independent and normally distributed!

Generating Corrolated Random Sequences

Suppose we are generating two random time series, representing the price of Google and Yahoo stock.
These sequences are not independent but are correlated with a coefficient of say $\rho = 0.8$
Let X_1 and X_2 be two uncorrelated random sequences.
Define $Y_1 = \rho X_1 + \sqrt{1-\rho^2} X_2$.
Then sequences X_1 and Y_1 have a correlation of ρ.





Cholesky factorization can be used to generate d random sequences with a specified pairwise correlation matrix.