Jump Models

Craig Pirrong
Bauer College of Business
University of Houston

April 4, 2020

All of the analysis we have done so far assumes that the random movements in prices are driven by Brownian Motions. Brownian motions are continuous. But we know that in the real world, prices "jump" or "gap." That is, they exhibit discontinuities. If you didn't know that before the last month (this being written in April 2020), you sure as hell know it now!

So-called "jump models" have been developed to address this phenomenon. These models assume that prices move discontinuously ("jump") at random times. The time between jumps is described by a an exponential distribution, which has the density:

$$f(t) = \lambda e^{-\lambda t}$$

Here, λ is the "intensity" of the jump process, and measures the average number of jumps per unit time. For example, if $\lambda = 10$, on average there are 10 jumps per year. The average time between jumps is:

$$E(t) = \int_0^\infty tf(t)dt = \lambda \int_0^\infty te^{-\lambda t}dt = \frac{1}{\lambda}$$

where the last step can be shown using integration by parts and simplifying. Thus, if there are 10 jumps per year on average, on average there is 1/10th of a year between jumps.

Jump models specify a counting process N_t, which is the total number of jumps that have occurred by time t. Thus, $dN_t = 1$ or $dN_t = 0$, depending on whether a jump occurs at t.

Jump models also allow the magnitude of the jumps to be random. For example, jumps could be distributed lognormal, with mean zero and standard deviation ν.

Jump-diffusion models combine a jump process and a standard diffusion process to characterize the dynamics of a risky asset price:

$$d \ln S_t = (\mu - .5\sigma^2)dt + \sigma dW_t + dN_t \tilde{J}$$

where $\tilde{J}$ is the (random) size of the jump.

Note that in this expression μ is the drift under the equivalent measure that turns the discounted price process into a martingale. We *cannot* assume $\mu = r$ here because we cannot hedge the jump risk, and recall that there is a unique measure (with $\mu = r$) only if we can hedge all risks.

If we assume $\tilde{J}$ is lognormal, we can derive closed form expressions for the value of European options if the asset follows a jump process. If we want to value more complicated claims, or if we want to assume $\tilde{J}$ is not lognormal, we need to value contingent claims via simulation.

Just as we can create random normal variates in Matlab, we can create random exponential variates using the `random` command in Matlab. The first step is to create a probability density using the `makedist` command, such as `pdexp=makedist('Exponential',mu)` where `mu` is a previously defined average time between jumps.

Then, we can sample `jumpmax` jump intervals (i.e., times between jumps) by a command like `jumptimes=random(pdexp,jumpmax,1)`. We then determine the jump times by summing up the jump intervals. Choose `jumpmax` so that the time of the last jump almost always exceeds the expiration date of the opton you are interested in. Then once you have simulated the jump times, you can simulate the size of each jump by drawing from a distribution for $\tilde{J}$. This could be lognormal, but if you are simulating you are not limited to this distribution.

How you use the simulated jumps depends on the kind of option you are valuing. If you are valuing a European option, you can simulate the diffusion (non-jump) part of the stock price movement by drawing from the normal distribution as we have already done, and then simulate the jump contribution by adding up all of the jumps that occur at times before expiration, and adding together the jump and diffusion parts.

If you are valuing a path dependent option, like an Asian option, or an option with early exercise, like a Bermudan, you need to proceed as follows. Consider an Asian option with N_a averaging dates. Simulate the stock price based on the diffusion component for each averaging date. Call this value at averaging date i S_i^d. To figure out the simulated price at date i, S_i^s, add to S_i^d all the jumps that occur prior to date i. So note that you have to loop over all the payment dates.