# StockProblem

November 12, 2018

1 Binomial Model for Asset Pricing

The binomial model for asset pricing can be used to price financial derivatives, such as options. While not as widely used as the Black-Scholes model, it is simpler and more intuitive. Instead of pricing derivatives, in this exercise, we will simply use the binomial model to model stock price movements and find the stock's price distribution.

1.0.1 One period Binomial Model

Let us first go over a one-period binomial model. In the binomial model, we construct a tree (technically a recombining tree) in which the nodes represent possible stock prices at various times. Over a single period, we assume that there are only two possibilites for the stock price at the end of a period: an "up" price and a "down" price. We find these prices using an "up" and "down" multiplier respectively. If S is the price of the stock at the beginning of the period and u, d denote the up, down multipliers respectively, then the two possible prices at the end of the period are $u \times S = uS$ and $d \times S = dS$. A one period binomial tree would look like:

$$
\begin{array}{c}
 & & uS \\
 & \nearrow & \\
S & & \\
 & \searrow & \\
 & & dS
\end{array}
$$
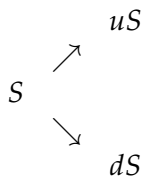
1.0.2 Multi-period Binomial Model

For more periods, we continue in the same fashion. For example, if we are at the node uS then at the end of the second period, there are two possibilites, we either go up to u^2S or down to duS. Notice, that if we are at the node dS and we go up, then we are at the node $udS = duS$ (we are simply doing the multiplication $u \times d \times S$). This is where the word recombining comes from. This makes it so the trees we build are not too "bushy". At the end of the first period, we only have 2 nodes, at the end of the second period, we have 3 nodes, at the end of the third period, we have 4 nodes, and so on. Here is an example of a two period binomial tree.

$$u^2S$$

$$uS \nearrow$$
$$uS$$
$$\nearrow \qquad \searrow$$
$$S \qquad\qquad udS = duS$$
$$\searrow \qquad \nearrow$$
$$dS$$
$$\searrow$$
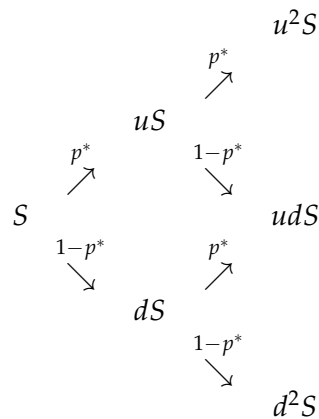$$d^2S$$
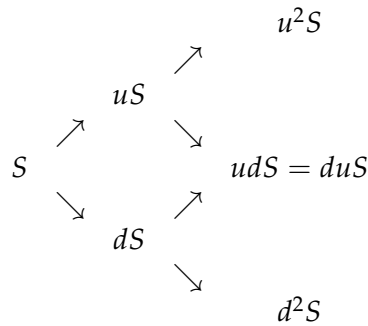
1.0.3 Modeling future stock prices

Now that we know how to make binomial stock price trees, we can use them to model future stock prices. When our trees have enough periods, say 10,000, a single path through the tree is a possible path the stock price could take in the future. How can we choose a path? Instead of building up the whole tree, we can just take it one step at a time; given a current price S we will either choose to go up to uS or down to dS and whatever we choose will be our starting point for the next iteration. We then repeat the same process from this point.

How do we determine if we should choose uS or dS? We can use a probability p^*, known as the *risk-neutral probability*. We assume the stock price goes up with probability p^* and down with probability $(1 - p^*)$.

Problem 1 Let's plot a path of possible stock prices as described above. Here are the notations and formulas we need: - S is the initial stock price - T is the length of the binomial tree in years - n is the number of periods - $h = \frac{T}{n}$ is the length of a single period - σ is the volatility of the stock - r is the annual interest rate - $u = e^{rh + \sigma\sqrt{h}}$ - $d = e^{rh - \sigma\sqrt{h}}$ - $p^* = \frac{e^{rh} - d}{u - d}$

    a) Assume $S = 100$, $T = 1$, $n = 10,000$, $\sigma = 0.3$, and $r = 0.08$. Try to plot a single stock price path. The function `rand()` generates a random number between 0 and 1. You can use the `plot` function from the `Plots` library.

    b) Create a function `createPath(S::Float64, r::Float64, sigma::Float64, T::Float64, n::Int64)` which creates a stock price path given initial parameters. Use `createPath` to create 10 different stock paths and plot them all on the same plot. You can use the function `plot!` to modify an existing plot.

    c) (Optional) Parallelize your path generation! See the parallelism documentation for details on how to do this. Try using `Threads.@threads`, `pmap`, and `@parallel for`.

Problem 2 Now, lets try and figure out what distribution the stock prices follow when using the binomial model. In this case, we are only worried about the final prices in our model. If 150 is one of the final prices in our model, the question we are trying to answer is: what is the probability that the final stock price will be 150? Notice, that the binomial tree is easily adjusted to accomodate probabilities.

$$u^2S$$

$$
\begin{array}{c}
\\
p^* \nearrow \\
uS \\
p^* \nearrow \qquad 1-p^* \searrow \\
S \qquad\qquad udS \\
1-p^* \searrow \qquad p^* \nearrow \\
dS \\
1-p^* \searrow \\
d^2S
\end{array}
$$

We can find the probability of any node in the tree, simply by following the paths to that node and multiplying the probabilities along the way. For example, the probability that the final price is u^2S is p^{*2}, and the probability that the final price is udS is $2p^*(1-p^*)$.

a) Draw a 3 period binomial tree and find the probability for all four final nodes (u^3S, u^2dS, ud^2S, d^3S). Don't forget to consider *all* paths to the node!

Do you notice anything about the probabilities for the final nodes?
Click to expand The probabilities follow the binomial formula, the probabilities in the third period from top to bottom are $p^{*3}, 3p^{*2}(1-p^*), 3p^*(1-p^*)^2, (1-p^*)^3$, the probabilities for a fourth period would be $p^{*4}, 4p^{*3}(1-p^*), 6p^{*2}(1-p^*)^2, 4p^*(1-p^*)^3, (1-p^*)^4$, and so on.

b) Assume $S = 100$, $T = 1$, $\sigma = 0.3$, and $r = 0.08$. Using a scatterplot (`scatter` from the `Plots` library), plot the distribution of the final prices for $n = 10, 50, 100$. The function `binomial(n, k)` may come in handy. If you get an `InexactError` try converting to `BigInts`. Does this distribution look familiar?