Week 7 – Exotic Options

MIT Sloan School of Management

Finance at MIT
Where ingenuity drives results

Outline

- Exotic options
  - Descriptions and uses
  - Pricing with Monte Carlo simulations and binomial trees

Exotic Options

- Nonstandard options, often constructed by tweaking ordinary options
- Exotic options solve specific business problems that ordinary options cannot
- Typically created and sold by investment banks and professional money managers, who in turn hedge the positions and earn a commission
- Goal is not to memorize or derive formulas (we have books & the web for that).
- The relevant questions are:
  1. What is the rationale for the use of an exotic option?
  2. Can the exotic option be approximated by a portfolio of ordinary options?
     - Such links can sometimes show us how to modify BSM to price exotics
  3. Is the exotic option cheap or expensive relative to a standard option that achieves a similar goal?
  4. What's the general approach for pricing them when there isn't a formula?

Non-standard American options

- Bermudan option
  - Can be exercised on certain pre-specified dates prior to expiration
  - Between American and European
  - Generally can be priced like American options on binomial tree
  - Strike price may change over the life of the option
    - E.g., employee stock options that are reset when they become far out-of-the-money
- Examples include some callable bonds and corporate warrants

Binary options

- Cash-or-nothing:
  - Call: pays 1 if $S_T > K$, zero otherwise: $\text{CashCall}(S, K, \sigma, r, T-t, \delta) = e^{-r(T-t)}N(d_2)$
  - Put: pays 1 if $S_T < K$, zero otherwise: $\text{CashPut}(S, K, \sigma, r, T-t, \delta) = e^{-r(T-t)}N(-d_2)$

- Asset-or-nothing:
  - Call: pays stock price if $S_T > K$, zero otherwise: $\text{AssetCall}(S, K, \sigma, r, T-t, \delta) = Se^{-\delta(T-t)}N(d_1)$
  - Put: pays stock price if $S_T < K$, zero otherwis $\text{AssetPut}(S, K, \sigma, r, T-t, \delta) = Se^{-\delta(T-t)}N(-d_1)$

- Look familiar? What is the value of a portfolio that is

  (1) long an asset-or-nothing call option with strike *K*, and

  (2) short *K* cash-or-nothing call options with strike K

Asian options

- The payoff on an Asian option is based on the average price over some period of time
  - Fundamentally different because they are path dependent
- Examples of when Asian options are useful
  - Profit depends on average price (e.g., of exchange rates, oil, electricity) over a period of time
  - There is concern that the price at a single point in time might be subject to manipulation
  - When price swings are frequent due to thin or illiquid markets

Example: convertible bonds have an embedded Asian option. Typically the exercise of the conversion option is based on the stock price over a 20-day period at the end of the bond's life

- ***Question:*** *What feature of an Asian call option tends to make it less valuable than an otherwise identical European call option?*

Basic types of Asian options

- Average can be based on geometric or arithmetic mean

- Suppose we record the stock price every h periods from $t = 0$ to $t = T$
- Arithmetic average:

$$A(T) = \frac{1}{N} \sum_{i=1}^{N} S_{ih}$$

- Geometric average:

$$G(T) = (S_h \times S_{2h} \times \cdots \times S_{Nh})^{1/N}$$

- Average used as the asset price or strike price: average price option and average strike option

	Arithmetic	Geometric
Average price call	$\max[0, A(T) - K]$	$\max[0, G(T) - K]$
Average price put	$\max[0, K - A(T)]$	$\max[0, K - G(T)]$
Average strike call	$\max[0, S_T - A(T)]$	$\max[0, S_T - G(T)]$
Average strike put	$\max[0, A(T) - S_T]$	$\max[0, G(T) - S_T]$

Example: Hedging currency exposure

The business: XYZ has monthly revenue of €100m, and cost in dollars $x =$ spot dollar price of a euro. In one year, the converted amount in dollars is

$$€100m \times \sum_{i=1}^{12} x_i e^{r(12-i)/12}$$

The problem: Ignoring interest, the amount of euro exposure that needs to be hedged is

$$\sum_{i=1}^{12} x_i = 12 \times \left(\frac{\sum_{i=1}^{12} x_i}{12} \right)$$

The solution: An arithmetic average price put option that puts a floor K, on the average exchange rate received

$$\max\left(0, K - \frac{1}{12}\sum_{i=1}^{12} x_i\right)$$

Example: Hedging currency exposure

- What are alternative strategies?
  - A basket of 12 options expiring in each of the 12 months
  - A currency swap

We saw that currency options can be valued using BSM formula with constant dividend yield, recognizing that currency earns risk-free interest rate

Example: Assume the current exchange rate is \$0.9/EUR, strike $K = 0.9$, $r_{\$} = 6\%$, $r_{€} = 3\%$, dollar/euro volatility $\sigma = 10\%$

• 12 European puts expiring in 1 year	0.2753
• A basket of 12 monthly options	0.2178
• 12 Geometric average puts	0.1796
• 12 Arithmetic average puts	0.1764
• Currency swap	?

Pricing Asian options

- Closed form solution for geometric case
  - Uses Black's Model and log-normal approximation for the average mean and variance
- Binomial tree (but path dependence is a problem)
- Monte Carlo simulation

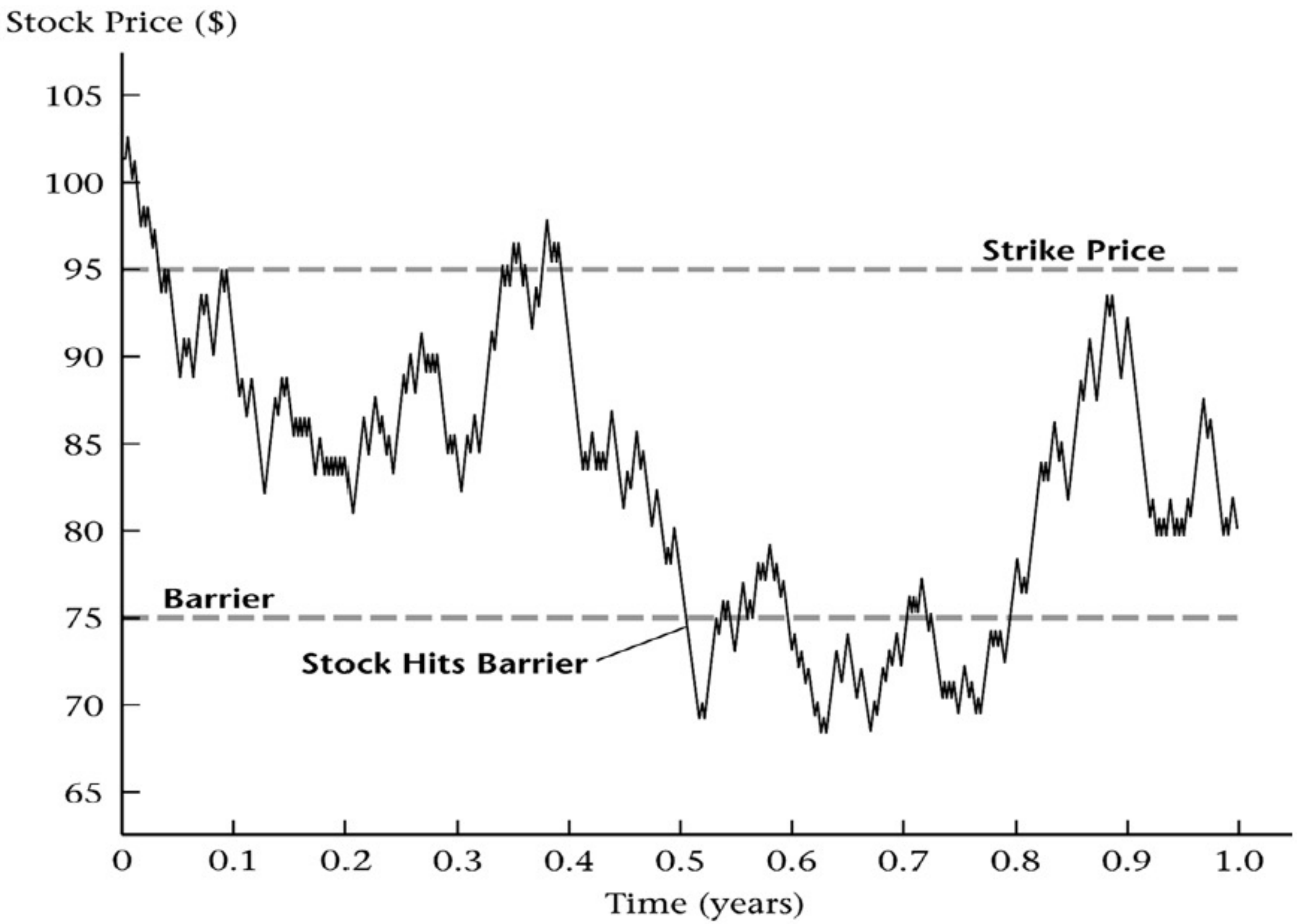
Barrier options

- Payoff depends on whether over its life the underlying price hits a certain barrier
  - Implies payoff is path dependent

Barrier puts and calls

- **Knock-out options**: *go out of existence* if the underlying price
  - Down-and-out: falls below a barrier
  - Up-and-out: rises above a barrier
- **Knock-in options**: *come into existence* if the underlying price
  - Down-and-in: falls below a barrier
  - Up-and-in: rises above a barrier
- **Rebate options**: make a fixed payment if the underlying price
  - Down rebates: falls below a barrier
  - Up rebates: rises above a barrier
- *Question: What is worth more, a barrier option or an otherwise identical option?*

Illustration: Down-and-in option



Pricing barrier options

- Parity relations:

$$c = c_{ui} + c_{uo}$$
$$c = c_{di} + c_{do}$$
$$p = p_{ui} + p_{uo}$$
$$p = p_{di} + p_{do}$$

- Can price on binomial tree (but complicated by path dependence) or use Monte Carlo simulation

Example:

Premiums of standard, down-and-in, and up-and-out currency put options with strikes K. The column headed "standard" contains prices of ordinary put options. Assumes $x_0 = 0.9$, $\sigma = 0.1$, $r_\$ = 0.06$, $r_€ = 0.03$, and $t = 0.5$.

Strike ($)	Standard ($)	Down-and-In Barrier ($)		Up-and-Out Barrier ($)		
		0.8000	**0.8500**	**0.9500**	**1.0000**	**1.0500**
$K = 0.8$	0.0007	0.0007	0.0007	0.0007	0.0007	0.0007
$K = 0.9$	0.0188	0.0066	0.0167	0.0174	0.0188	0.0188
$K = 1.0$	0.0870	0.0134	0.0501	0.0633	0.0847	0.0869

Lookback options

- Floating lookback call

$$S_T - S_{min}$$

  - allows buyer to buy stock at lowest observed price in some interval of time

- Floating lookback put

$$S_{max} - S_T$$

  - allows buyer to sell stock at highest observed price in some interval of time

Strike is what floats

- Fixed lookback call

$$\max(S_{max} - K, 0)$$

- Fixed lookback put

$$\max(K - S_{min}, 0)$$

- Analytical valuation for all types

- Relatively expensive
- Closed form solutions assume continuous looks and lognormal process
- Related to “shout” options

Exchange options

- Pays off only if the underlying asset outperforms some other asset (the benchmark asset)

$$\max(0, S_T - N_T)$$

The value of a European exchange call

$$C(S, N, \sigma_s, \sigma_n, r, T, \delta_s, \delta_n, \rho) = Se^{-\delta_s T}\mathcal{N}(d_1) - Ne^{-\delta_n T}\mathcal{N}(d_2)$$

where

$$d_1 = \frac{\ln\left(\frac{Se^{-\delta_s T}}{Ne^{-\delta_n T}}\right) + \frac{1}{2}\sigma^2 T}{\sigma\sqrt{T}}$$

$$d_2 = d_1 - \sigma\sqrt{t}$$

$$\sigma = \sqrt{\sigma_s^2 + \sigma_n^2 - 2\rho\sigma_s\sigma_n}$$

Compound options

- An option to buy or sell an option
  - Call on call
  - Put on call
  - Call on put
  - Put on put
- Often priced by backward induction on a binomial tree

Example: College Education

- A two-year program: one has the option to pay $\$10,000$ in year 1 to enroll, and the option to pay $\$10,000$ in year 2 to finish the degree
  - The option to continue in year 2 is a regular call with strike $K = 10,000$; exercise when $X > K$ ($X =$ value of degree)
  - The option to enter into the 2-year program is a compound call

Gap options

- A gap call options pays $S - K_1$ when $S > K_2$

The value of a gap call

$$C(S, K_1, K_2, \sigma, r, T, \delta) = Se^{-\delta T}\mathcal{N}(d_1) - K_1 e^{-rT}\mathcal{N}(d_2)$$

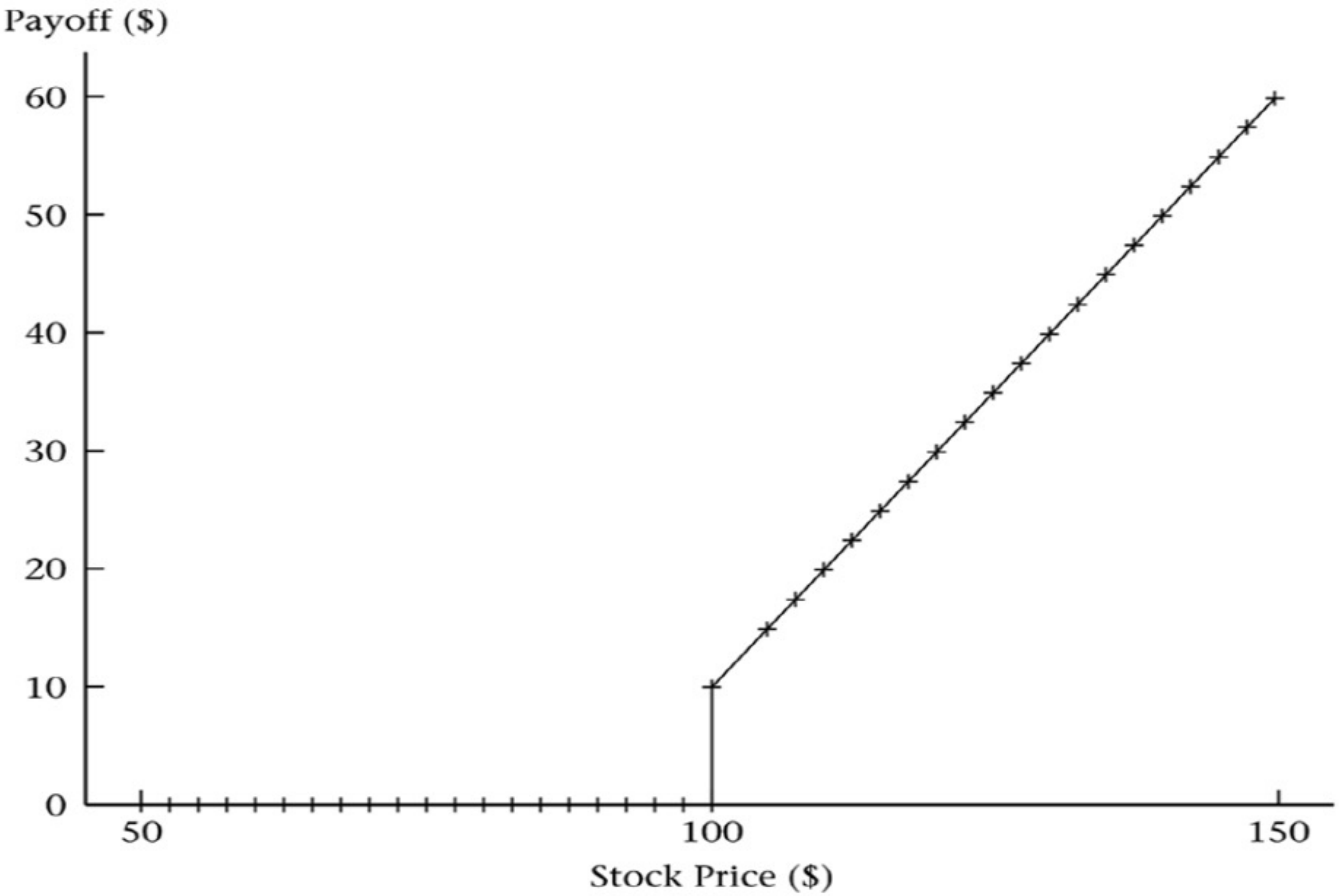
where

$$d_1 = \frac{\ln(S/K_2) + (r - \delta + \frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}}$$

$$d_2 = d_1 - \sigma\sqrt{t}$$

(d_1 and d_2 are the same as in Black-Scholes)

Illustration: Gap call



Pays S – K1 when S > K2

K1 = 90
K2 = 100

Q: Does this option cost more or less than without the gap?

Quantos

- A **quanto** is a contract that allows an investor in one currency to hold an asset denominated in another currency without exchange rate risk
- Example: Nikkei put warrants traded on the American Stock Exchange
- Payoff and premium are in dollars, but directly scaled by the yen price of the Nikkei index relative to a yen strike price

Quantos are attractive because they shield the purchaser from exchange rate fluctuations. If a US investor were to invest directly in the Japanese stocks that comprise the Nikkei, he would be exposed to both fluctuations in the Nikkei index and fluctuations in the USD/JPY exchange rate.

Essentially, a quanto has an embedded currency forward with a variable notional amount. It is that variable notional amount that give quantos their name**—"quanto" is short for "quantity adjusting option."**

Pricing exotic options

Pricing exotics

- Multiple approaches to pricing
  - Modified Black-Scholes-Merton
  - Binomial trees
  - Monte Carlo simulation

Risk neutral trees

- Recall the one-step tree (to begin with)
- Assume $S_0 = 100$; $K = 100$, $T = 1$, $r = 2\%$, $\sigma = 30\%$
- Given that $u = e^{\sigma\sqrt{T}} = 1.34986$, the price of *any* derivative security with payoff $V(S_1)$ can be computed as

$$V_0 = \mathbf{E}^* \left[e^{-rT} V(S_1)\right] = e^{-rT} \left[q^* V(S_{1,u}) + (1 - q^*) V(S_{1,d})\right]$$

- For instance, a call option has price given by

$i = 0$

$S_0 = 100.000$
$q_0^* = 0.4587$
$c_0 = e^{-rT} \times q_0^* \times c_{1,u} = 15.731$

$i = 1$

$S_{1,u} = 134.986$
$c_{1,u} = 34.986$

$S_{1,d} = 74.082$
$c_{1,d} = 0$

Monte Carlo simulations on risk-neutral trees

- An alternative way of computing the expected future payoff is to *simulate* up and down movements using a computer
- For instance, in Excel the function *RAND*() simulates a uniform between [0, 1]
  - Thus, $RAND() > q^*$ (for q^* between $[0, 1]$) occurs with probability $(1 - q^*)$, and vice versa

1. We can simulate $RAND()$ many times, say N:
   - Whenever the realization $RAND() > q^*$ we say that we went *down* the tree;
   - Whenever $RAND() < q^*$, we say we went *up* the tree
2. The stock price at time $T = 1$ will then be $S_{1,u}$ or $S_{1,d}$, depending on the outcome of Step 1. Let S_1^i denote the realization of S_1 in simulation run i
3. Compute the payoff of the security at time $T = 1$ for each simulation run, e.g. $V(S_1^i) = max(S_1^i - K, 0)$
4. The value of the security is the average of the many realizations

$$\widehat{V}_0 = \text{average of } \left[e^{-rT}V(S_1^1), e^{-rT}V(S_1^2), \cdots, e^{-rT}V(S_1^N)\right] = \frac{1}{N}\sum_{i=1}^{N} e^{-rT}V(S_1^i)$$

Monte Carlo simulations on risk-neutral trees

MIT MANAGEMENT SLOAN SCHOOL

- For instance, given $q^* = 0.4587$, we obtain the following table

RAND()	Move on Tree	Price at T	Payoff	discounted
0.457335	up	134.986	34.986	34.293
0.393937	up	134.986	34.986	34.293
0.090053	up	134.986	34.986	34.293
0.878148	down	74.082	0	0
0.658659	down	74.082	0	0
0.759579	down	74.082	0	0
0.798027	down	74.082	0	0
0.061689	up	134.986	34.986	34.293
0.969222	down	74.082	0	0
0.392675	up	134.986	34.986	34.293
			Average	17.147
			st. error	5.715

- With only $N = 10$ simulation, it is no surprise that the value of the security $\widehat{V}_0 = 17.147$ is rather different from the value from the tree ($V_0 = 15.731$)
- As N increases, the value gets more and more precise

Monte Carlo simulations on risk-neutral trees

MIT MANAGEMENT SLOAN SCHOOL

- How many simulations?
  - The number of simulations N should be large enough to obtain a small "standard error" for our estimate of the option price
  - This is computed as the standard deviation of the discounted payoffs from the simulations, divided by $\sqrt{N}$:

$$\text{standard error} = \frac{\text{Standard Deviation of } \{e^{-rT}V(S_1^1), e^{-rT}V(S_1^2), \cdots, e^{-rT}V(S_1^N)\}}{\sqrt{\mathbf{N}}}$$

(This formula is the standard deviation of the mean of N independent draws.)

- In the previous example, the standard error was $s.e. = 5.715$
  - This implies that with 95% probability, the true value of the security is between $[\widehat{V}_0 - 2 \times s.e., \widehat{V}_0 + 2 \times s.e.] = [5.715, 28.577]$
  - Given the number of simulations $N = 10$, we are 95% confident that the true value is between 5.715 and 28.577, rather imprecise!
- Increasing the number of simulations to $N = 1000$, we obtain $\widehat{V}_0 = 15.725$ with $s.e. = 0.54$.
  - The confidence interval is [14.644, 16.806], much tighter than before

Multi-step trees

- A 10-step tree is as follows:

BINOMIAL TREE MODEL

Stock Assumption		Option Assumption			Tree		Risk Neutral Prob	
mu	0.1	T	1		n	10	q*	0.486836
sigma	0.3	K	100		h	0.1		
r	0.02	Call or Put	1	(=1 for call	u	1.099514	Price Binomial	12.530
q	0				d	0.909493	Delta Binomial	0.584
S0	100				p	0.52919		

time ==>	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
i==>	0	1	2	3	4	5	6	7	8	9	10
0	100.000	109.951	120.893	132.924	146.151	160.696	176.687	194.270	213.603	234.859	258.231
1		90.949	100.000	109.951	120.893	132.924	146.151	160.696	176.687	194.270	213.603
2			82.718	90.949	100.000	109.951	120.893	132.924	146.151	160.696	176.687
3				75.231	82.718	90.949	100.000	109.951	120.893	132.924	146.151
4					68.422	75.231	82.718	90.949	100.000	109.951	120.893
5						62.229	68.422	75.231	82.718	90.949	100.000
6							56.597	62.229	68.422	75.231	82.718
7								51.475	56.597	62.229	68.422
8									46.816	51.475	56.597
9										42.579	46.816
10											38.725

time ==>	0.000	0.100	0.200	0.300	0.400	0.500	0.600	0.700	0.800	0.900	1.000
i==>	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0
0	12.530	18.247	25.881	35.666	47.656	61.691	77.484	94.868	114.002	135.059	158.231
1		7.156	11.075	16.699	24.431	34.528	46.948	61.294	77.086	94.470	113.603
2			3.465	5.782	9.429	14.947	22.879	33.522	46.551	60.895	76.687
3				1.280	2.345	4.231	7.481	12.871	21.292	33.123	46.151
4					0.275	0.566	1.164	2.396	4.932	10.151	20.893
5						0.000	0.000	0.000	0.000	0.000	0.000
6							0.000	0.000	0.000	0.000	0.000
7								0.000	0.000	0.000	0.000
8									0.000	0.000	0.000
9										0.000	0.000
10											0.000

Monte Carlo simulations on multi-step trees

Option Prices By Simulations (on the Tree)

Simulated Put Price	Simulated Call Price	
11.030	12.637	Price
0.480	0.675	St. Error

Maturity (years)	1.000
Maturity (steps)	10.000

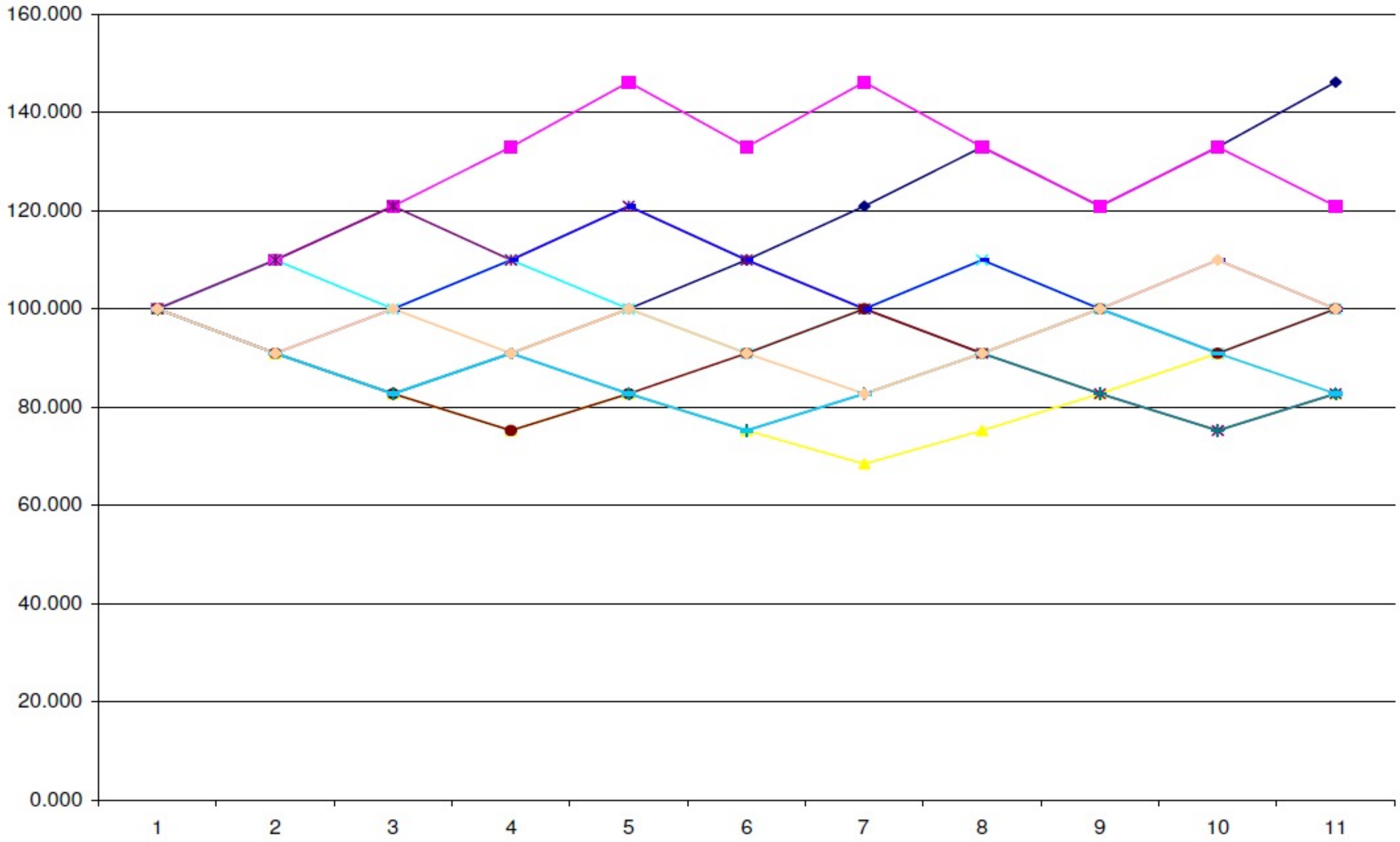
Call Price Binomial Tree	12.530
Call Price Black and Schole	12.822

SIMULATION OF RISK NEUTRAL PRICE PROCESS

		Time											
Discounted	Discounted		0.000	0.100	0.200	0.300	0.400	0.500	0.600	0.700	0.800	0.900	1.000
Put Payoff	Call Payoff	Simulation	0.000	1.000	2.000	3.000	4.000	5.000	6.000	7.000	8.000	9.000	10.000
0.000	45.238	1.000	100.000	90.949	82.718	90.949	100.000	109.951	120.893	132.924	120.893	132.924	146.151
0.000	20.479	2.000	100.000	109.951	120.893	132.924	146.151	132.924	146.151	132.924	120.893	132.924	120.893
16.940	0.000	3.000	100.000	90.949	82.718	75.231	82.718	75.231	68.422	75.231	82.718	90.949	82.718
0.000	0.000	4.000	100.000	109.951	100.000	109.951	100.000	90.949	100.000	109.951	100.000	90.949	100.000
16.940	0.000	5.000	100.000	109.951	120.893	109.951	120.893	109.951	100.000	90.949	82.718	75.231	82.718
0.000	0.000	6.000	100.000	90.949	82.718	75.231	82.718	90.949	100.000	90.949	100.000	90.949	100.000
16.940	0.000	7.000	100.000	90.949	82.718	90.949	82.718	75.231	82.718	90.949	82.718	75.231	82.718
0.000	0.000	8.000	100.000	90.949	100.000	109.951	120.893	109.951	100.000	109.951	100.000	109.951	100.000
16.940	0.000	9.000	100.000	90.949	82.718	90.949	82.718	75.231	82.718	90.949	100.000	90.949	82.718
0.000	0.000	10.000	100.000	90.949	100.000	90.949	100.000	90.949	82.718	90.949	100.000	109.951	100.000

Monte Carlo simulations on multi-step trees

- A few simulation paths (they look like a tree, with missing branches)



Why Monte Carlo simulations?

- Why do we need Monte Carlo simulations when we have the tree itself?
  - Monte Carlo Simulations may be useful to price derivative securities with path dependent payoff
  - That is, the value at maturity depends on the path taken by the stock during the life of the security
- For instance, recall that Asian options have a payoff given by

$$\text{Asian Call Option} = \max\left(\{\text{Average of } S_t \text{ from 0 to T}\} - K, 0\right)$$
$$\text{Asian Put Option} = \max\left(K - \{\text{Average of } S_t \text{ from 0 to T}\}, 0\right)$$

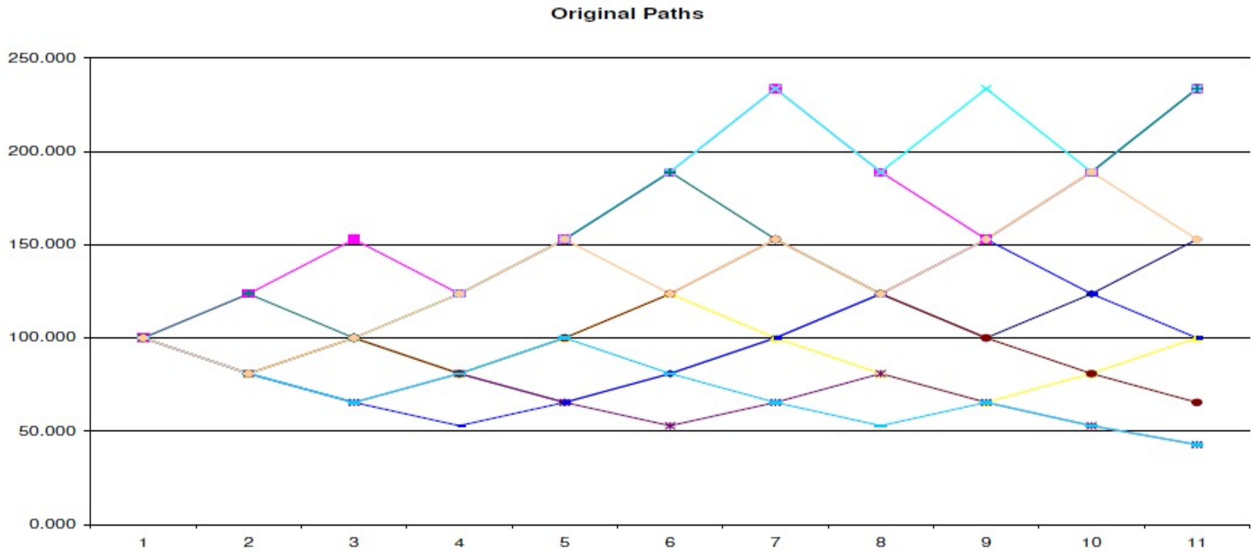
- These options are very hard to price on a tree without simulations
- Consider a two-step tree…

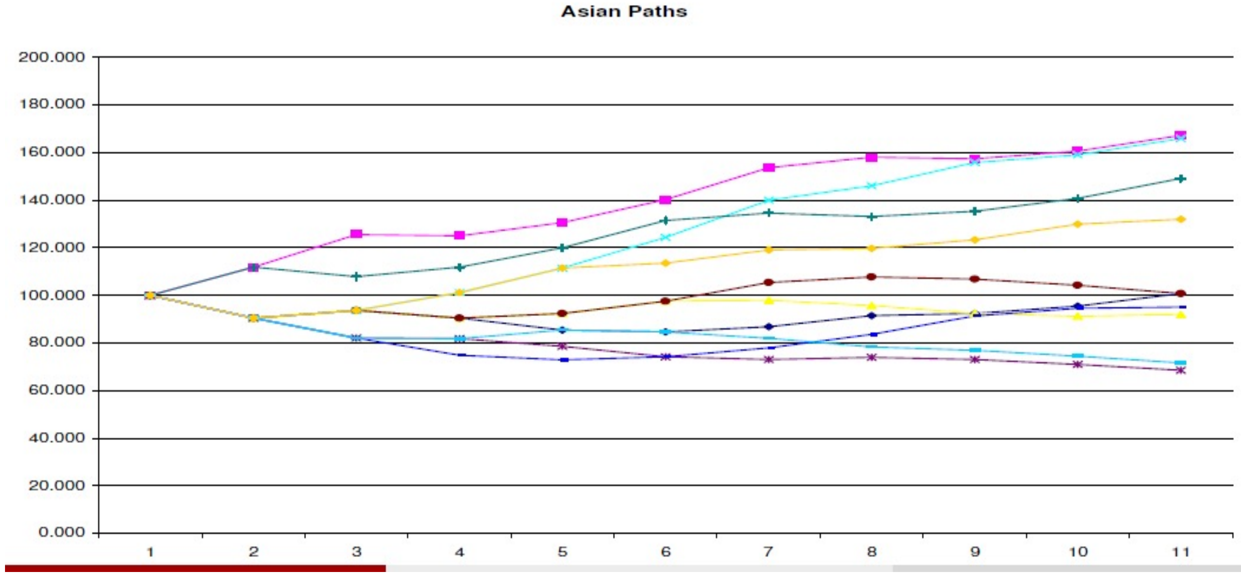
Path-dependent option

$i = 0$	$i = 1$	$i = 2$		$(r.n.prob)$
		$S_{2,uu} = 152.847$ $\overline{S}_{2,uu} = 125.4925$	$\Longrightarrow$ $c_{2,uu} = 25.4925$	$(Q^*_{2,uu} = 22.15\%)$
	$S_{1,u} = 123.631$			
		$S_{2,ud} = 100$ $\overline{S}_{2,ud} = 107.877$	$\Longrightarrow$ $c_{2,ud} = 7.877$	$(Q^*_{2,ud} = 24.91\%)$
$S_0 = 100$				
		$S_{2,du} = 100$ $\overline{S}_{2,du} = 93.628$	$\Longrightarrow$ $c_{2,du} = 0$	$(Q^*_{2,du} = 24.91\%)$
	$S_{1,d} = 80.886$			
		$S_{2,dd} = 65.425$ $\overline{S}_{2,dd} = 82.1036$	$\Longrightarrow$ $c_{2,dd} = 0$	$(Q^*_{2,dd} = 28.01\%)$

- Even if $S_{2,ud} = S_{2,du} = 100$, the payoff when the final price is 100 depends on the path of S, namely, whether $S_1 = S_{1,u} = 123.631$ or $S_1 = S_{1,d} = 80.886$
- In this case, we can compute the value of the security on the tree $V_0 = \sum_{j=1}^{4} Q_j^* V_{2,j} = 7.45$
- 1000 Monte Carlo simuations yield $\widehat{V}_0 = 7.560$ with $s.e. = 0.316$

Why Monte Carlo simulations?





When the number of steps gets large, path dependent options become much more difficult to price without Monte Carlo

This shows the stock price paths (top) and averages stock price paths (bottom) over 10 Monte Carlo runs.

- While the original stock price paths look like the recombining tree we started with, the averages look like paths on a non-recombining tree
- Non-recombining trees are much harder to evaluate numerically for a large number of time steps

Why Monte Carlo simulations?

- Recall some of the other types of popular path-dependent exotic options:
  - **Barrier Options:**
    - The option expires if stock hits an upper (**up and out**) or a lower (**down and out**) barrier
    - The option comes into existence if the stock hits and upper (**up and in**) or a lower (**down and in**) barrier
  - **Lookback Options**: The final payoff depends on the maximum or minimum value achieved by the stock before maturity
  - **Asian Strike Options**: The strike price is equal to the average stock price

Monte Carlo simulations without trees

- There is no reason to limit MC simulations to trees
- The main requirement to be able to price by MC simulations is to satisfy conditions for risk neutral pricing to be valid
  - That is, dynamic replication can be achieved
- These no arbitrage conditions are naturally satisfied on the trees we have constructed
- However, once we decide we can use risk neutral pricing, we can simulate out of any distribution
  - For example, MC can generate prices based on the lognormal model, as in BSM
  - MC can be used to incorporate time-varying volatility, for instance by using the Heston Model.

Monte Carlo simulations under log-normality

- With the lognormal model, one way to simulate stock price is to use the following algorithm:

  For a given h,

$$\mathbf{S_{t+h} = S_t \times e^{(r-\frac{\sigma^2}{2})h+\sigma\sqrt{h}\epsilon_t}}$$

  where

$$\epsilon_t \sim \mathbf{N(0,1)}$$

In Excel, you can draw a standard normal shock using NORMINV(RAND(),0,1)

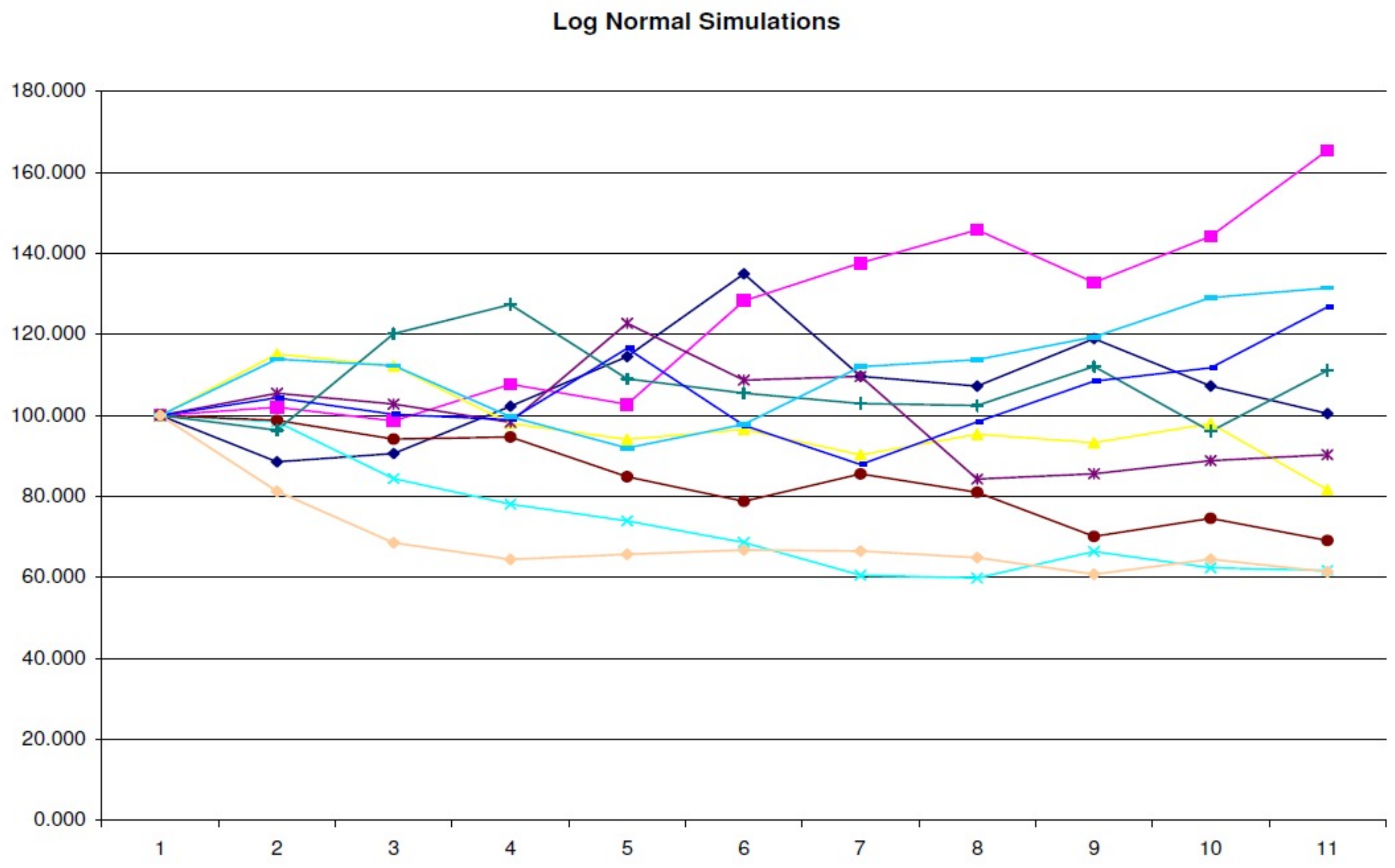
- The rules of the log-normal distribution imply

$$E^*\left(\frac{S_{t+h}}{S_t}\right) = e^{(r-\frac{\sigma^2}{2})h+\left\{E^*[\sigma\sqrt{h}\times\epsilon_t]+1/2Var[\sigma\sqrt{h}\times\epsilon_t]\right\}} = e^{rh}$$

- Moreover, σ^2 converges to the (annualized) variance of log returns $Var[\log(S_{t+h}/S_t)]$

Monte Carlo simulations under log-normality

This figure shows the outcomes for 10-period paths, for 10 Monte Carlo simulations:



Monte Carlo simulations with multiple factors

- Consider an option that pays the maximum between the return on Google and Apple stock from 0 to T
- That is, if S_0 and N_0 are the current prices of the two stocks, the payoff at maturity of the option is

$$\text{Payoff at } T = \max\left(\frac{S_T}{S_0}, \frac{N_T}{N_0}\right)$$

- How much would one pay for this security?
  - The risk neutral processes of S_t and N_t are the same as before

$$S_{t+h} = S_t \times e^{(r-\frac{\sigma_S^2}{2})h+\sigma_S\sqrt{h}\epsilon_{1,t}}$$

$$N_{t+h} = N_t \times e^{(r-\frac{\sigma_N^2}{2})h+\sigma_N\sqrt{h}\epsilon_{2,t}}$$

- Since returns on Google and Apple are correlated, we need a methodology to simulate correlated shocks $\epsilon_{1,t}$, $\epsilon_{2,t}$
  - Let $\widehat{\epsilon}_t$ be a standard normal, uncorrelated with $\epsilon_{1,t}$. Then, define

$$\epsilon_{2,t} = \rho\epsilon_{1,t} + \sqrt{1-\rho^2}\widehat{\epsilon}_t$$

  - **Claim**: $\epsilon_{2,t}$ has zero mean, variance 1, and correlation ρ with $\epsilon_{1,t}$ (verify!)

Monte Carlo simulations with multiple factors

MIT MANAGEMENT SLOAN SCHOOL

- For each simulation run i, compute the discounted payoff

$$V^i = e^{-rT} \max\left(\frac{S_T^i}{S_0}, \frac{N_T^i}{N_0}\right)$$

- The price of the security is then

$$\widehat{V}_0 = \frac{1}{n}\sum_{i=1}^{n} V^i$$

  - Assuming $\sigma_S = \sigma_N = .3$, $r = .02$ and $\rho = .7$, then $\widehat{V}_0 = 1.134$

Monte Carlo simulations with multiple factors

- As a second example, consider an option with the payoff

$$\text{Payoff at } T = \max\left(\frac{S_T}{S_0} - \frac{N_T}{N_0}, 0\right)$$

  - That is, it pays only when the first stock (say Google) does better than the second (say Apple)
  - The same simulations show that the fair value of this option is $\widehat{V}_0 = 0.1$

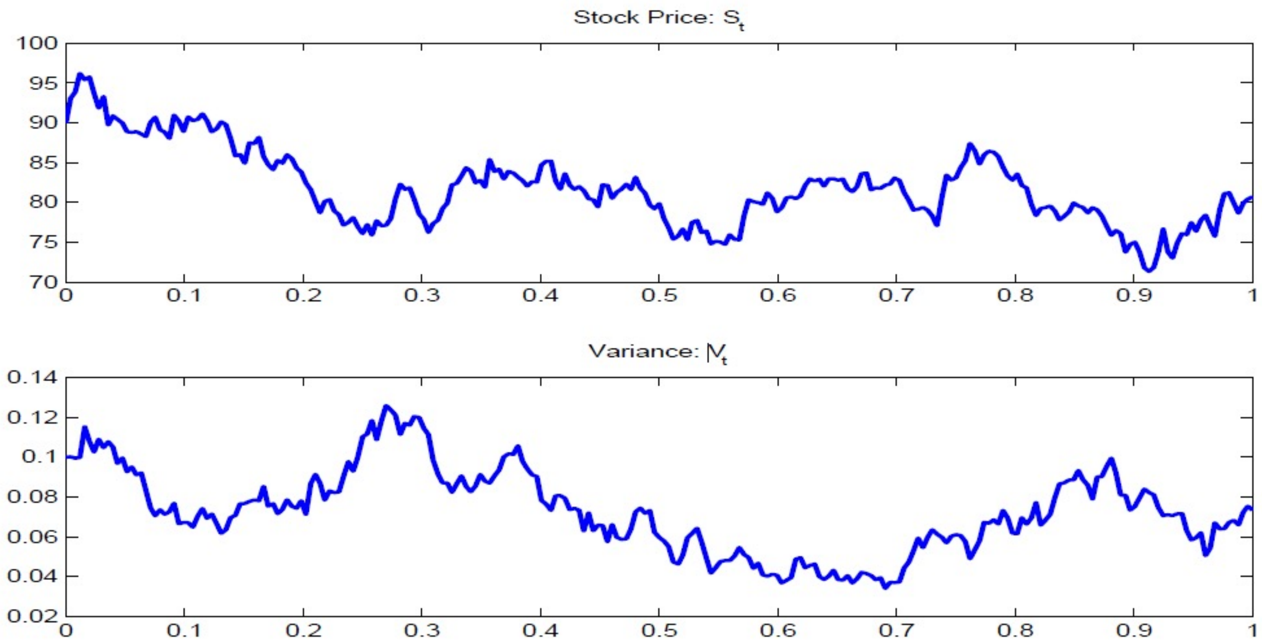
Monte Carlo simulation with multiple factors
Stochastic volatility and the Heston model

- Heston model for stochastic volatility is an example with non-traded factor
- Assume that under the risk-neutral probability, for some small h,

$$S_{t+h} = S_t + S_t \left(rh + \sqrt{V_t h} \epsilon_{s,t} \right) \tag{1}$$

$$V_{t+h} = \kappa h \overline{V} + (1 - \kappa h) V_t + \xi \sqrt{V_t h} \epsilon_{v,t} \tag{2}$$

where κ and ξ are constants, and $\epsilon_{s,t}$ and $\epsilon_{v,t}$ have correlation ρ



Takeaways on exotics

- We've seen that to understand exotic options some key questions are:
  - What purpose(s) does the exotic option serve?
  - Can the exotic option be approximated by a portfolio of ordinary options?
  - What are the key determinants of the value of an exotic option? Intuition?

Takeaways on pricing exotics numerically

- Tools for pricing include modified BSM, binomial trees, and Monte Carlo
- Binomial trees
    - Generally need to use binomial trees for American-style options where a decision has to be made about when to exercise
    - Most useful when working backwards and seeing ordered outcomes is essential
- Monte Carlo simulation
    - One of the main tools used by practitioners to price complex securities under fairly general assumptions about the underlying stochastic processes
        - Just three steps:

    (1) Simulate many paths of underlying stochastic variables under the risk neutral probabilities

    (2) For each path, compute the discounted simulated payoff of the derivative security

    (3) Estimate the derivative price as the average of discounted payoffs across paths

Takeaways on numerical pricing of exotics

- MCS are especially useful to value *path dependent* securities, or securities that depend on the value of multiple underlying variables
  - Barrier options, Asian options, Lookback options
  - Options on maximum, options on relative returns across securities

- MCS are also very useful to value securities under general processes for underlying stochastic variable, such as
  - Stochastic volatility
  - Stochastic interest rates
  - Jumps

- The ever increasing gains in the computer speed make MCS methodology increasingly attractive