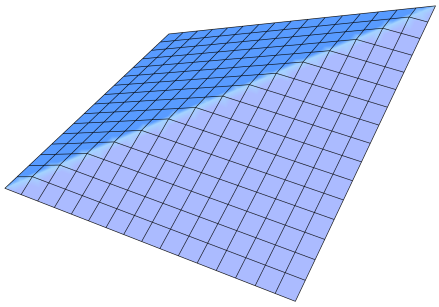

CS 476: Numeric Computation for Financial Modelling

Prof. Wei Xu • Winter 2015 • University of Waterloo
MWF 11:30AM–12:20PM in MC 2054

Transcribed, LaTeXed, and edited by Rob Zimmerman

Note: These are personal lecture notes, not official course materials. They may contain errors (which by default you should attribute to me, Rob Zimmerman, rather than the course instructor). The permanent home of these — and all of my other lecture notes — is https://rob-zimmerman.github.io/coursenotes.

Contents

Lecture 1 — Monday, January 05, 2015

1. No-Arbitrage and Binomial Pricing

Risk in Asset Prices

Short-term asset prices are unpredictable, and **risk** refers to an uncertain future outcome. The central issue in finance is that a future asset price can follow many different paths, as illustrated in Figure 1.

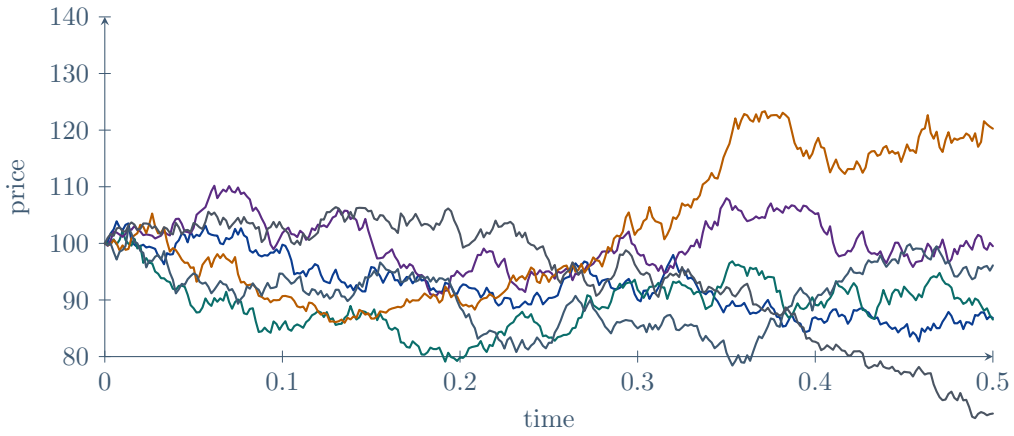

Figure 1

Example 1.1 Air Canada may be worried that the fuel price will rise too much one year from now. A fuel-producing company, on the other hand, may be worried that the fuel price will become too low in one year. Thus, in finance, risk can correspond either to the price going up or to the price going down.

Derivative Markets

Derivatives are tools for transferring, or trading, risk. They can be used to reduce risk, much like insurance, but they can also be used to increase risk exposure. The growth of derivative markets was fueled by deregulation, globalization of markets, and advances in mathematical and computational tools.

The derivative market has been described as the largest financial market in the world, with $531.2 trillion in notional value. It has also been characterized as "too complicated to explain and too important to ignore." (CBS *60 Minutes*) Derivatives on mortgage-backed securities played a central role in the 2008 global market meltdown.

Some milestones in the development of derivative markets include the growth of options in the late 1970s, mortgage-backed securities in the 1980s, warrants in the 1980s, index derivatives in the 1980s and 1990s, and credit derivatives in the late 1990s.

A **financial derivative** is a financial contract whose payoff or value is derived from one or more underlying market variables. The underlying may be a commodity, a stock, a market index, an interest rate, a bond, an exchange rate, or a similar traded quantity; the contract may depend on the terminal value, the path, or several underlying variables.

A **European call option** on an underlying asset gives its holder the right to buy the asset at a preset strike price K. This right may be exercised only at the expiry date T of the option. The option can be purchased today for a price C_0, called the premium. A **European put option** gives the right to sell. An American option may be exercised at any time before expiry.

The holder of the option enters a **long position**, while the writer enters a **short position**. There is an asymmetry between the two sides of the contract: the writer has an obligation, whereas the holder has a right but no obligation.

Example 1.2 Air Canada can buy a call option with one-year expiry and strike equal to today's fuel price. A fuel-producing company can buy put options to protect itself against

a fall in the underlying price. Thus, options can be used to hedge, meaning to reduce risk exposure, but they can also be used to increase risk exposure.

Options provide **leverage**. In particular, if $S_T > S_0$, then the percentage return on the option position,

$$\frac{C_T - C_0}{C_0},$$

can be much larger than the percentage return on the underlying asset,

$$\frac{S_T - S_0}{S_0}.$$

This immediately leads to a basic question:

Question 1.3 How much should an option contract cost today?

The historical search for an option pricing formula began with Louis Bachelier's 1900 thesis at the University of Paris, which introduced a random walk model and a call pricing formula. During the 1950s and 1960s, Paul Samuelson worked on call option pricing, introduced the lognormal distribution model, and rediscovered Bachelier's thesis. Ed Thorp used, though did not prove, the correct formula for pricing warrants in 1965. In the early 1970s, Fischer Black, Myron Scholes, and Robert Merton discovered the Black–Scholes–Merton formula. Scholes and Merton later received the Nobel Prize in 1997. Mathematical finance has flourished since 1970.

To **hedge** an option position, the writer can invest the proceeds from selling the option in an initial portfolio, alter the composition of this portfolio as needed over time, and aim to guarantee that in every state of nature the terminal value of the portfolio is sufficient to meet the terminal obligation.

A hedge is called **perfect** if the terminal value of the portfolio is exactly equal to the terminal obligation. In addition, the hedge policy must be **self-financing**, meaning that each revised portfolio is financed exactly by the proceeds from the previous position.

Derivatives can make it easy to take on very large amounts of risk, and this can lead to very large losses. Examples include Barings Bank, Orange County, and Long-Term Capital Management, as well as the 2008 failures of Lehman Brothers, Bear Stearns, and AIG.

There has also been substantial debate about the role of mathematical finance and computational finance in the financial crisis. Two representative examples are Steve Shreve's article "Don't Blame

the Quants" from October 8, 2008, and the *New York Times* article "Risk MisManagement: Were the measures used to evaluate Wall Street trades flawed?" from January 4, 2009.

Typical questions include how much to pay today for an option that allows the purchase of an Apple stock one year from now at today's price of \$107.71, how to hedge the future risk after selling such an option, and whether Value at Risk contributed to the financial market meltdown.

Several challenges arise in computational finance, including lack of sufficient information, sensitivity and robustness issues related to conditioning and stability, the dynamic and high-dimensional nature of financial problems, new financial innovations, and new technologies such as electronic trading.

Computational finance continues to attract mathematicians, computer scientists, physicists, and economists because the intellectual challenges intrinsic to financial markets remain substantial. Many problems remain unsolved, new ones continue to arise, and realistic models often become extremely complex and computationally intensive. As a result, computational mathematics plays an increasingly important role in financial markets.

Lecture 2 — Wednesday, January 07, 2015

No-Arbitrage Principles

Proposition 2.1 In a frictionless market with unrestricted borrowing and short selling, if two assets have the same future payoffs in every state, then they must sell at the same price today. Moreover, a positive-price asset cannot have a terminal return that is strictly greater than the risk-free return in every state.

Proof of the second statement. Suppose that

$$\frac{S_T - S_0}{S_0} > e^{rT} - 1.$$

Equivalently,

$$S_T > S_0 e^{rT}.$$

At time 0, borrow S_0 and use the borrowed money to buy the asset. This produces zero initial

cost. At time T, the position is worth

$$S_T - S_0 e^{rT} > 0,$$

which is a guaranteed profit. This is an arbitrage, so such an asset cannot exist in a no-arbitrage market. □

Let S_t denote the stock price at time t. A position of α shares has value αS_t. If $\alpha > 0$, the investor is **long** the stock, and if $\alpha < 0$, the investor is **short** the stock. A **short sale** means selling an asset today for its market price with the promise to buy it back in the future.

Proposition 2.2 (Put–call parity) Assume that the stock pays no dividends, the interest rate is deterministic, and the market has no arbitrage. Then, at any time t, a European call with value C_t and a European put with value P_t, both having the same strike K and expiry T, satisfy

$$C_t = P_t + S_t - Ke^{-\int_t^T r(s)\,\mathrm{d}s}.$$

One-Period Binomial Model

Consider a stock together with a European call of strike $K = 21$ in the one-period model in Figure 2. The stock moves from 20 to either 22 or 18, and the call payoff is therefore either 1 or 0. The motivating question is: if $p = 95\%$, what should C_0 be?

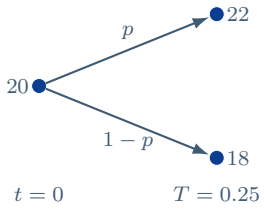

FIGURE 2

More generally, in the **one-period binomial model** shown in Figure 3, the stock price moves from S_t to

$$S_{t+1}^u = uS_t \qquad \text{or} \qquad S_{t+1}^d = dS_t,$$

where u is the up factor, d is the down factor, $0 < p < 1$, and $d < u$.

We also consider a risk-free asset that costs $e^{-r\Delta t}$ at time t and pays 1 at time $t + 1$. If the call

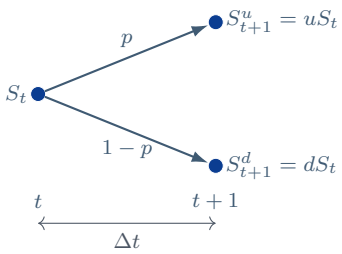

FIGURE 3

value at time $t+1$ is already known as a function of the stock price, for example

$$C^u_{t+1} = \text{payoff}(uS_t) \qquad \text{and} \qquad C^d_{t+1} = \text{payoff}(dS_t),$$

then the pricing problem is to determine C_t.

The corresponding risk-free and option value trees are shown in Figure 4.

FIGURE 4

The one-period binomial model assumes a frictionless market, allows short selling, treats assets as infinitely divisible, assumes there are no dividends, takes the interest rate $r \geq 0$ to be known, and imposes no arbitrage.

Proposition 2.3 In the one-period binomial model, there is no arbitrage if and only if

$$d < e^{r\Delta t} < u.$$

Consider numbers q^u and q^d such that

$$e^{-r\Delta t} \begin{bmatrix} 1 \\ uS_t \end{bmatrix} q^u + e^{-r\Delta t} \begin{bmatrix} 1 \\ dS_t \end{bmatrix} q^d = \begin{bmatrix} e^{-r\Delta t} \\ S_t \end{bmatrix}.$$

The two vector entries correspond to the bond and the stock, while the two weighted terms correspond to the up and down states at time $t+1$. Solving for the weight assigned to the up state gives

$$q^u = q^* = \frac{e^{r\Delta t} - d}{u - d} \qquad \text{and} \qquad q^d = 1 - q^*.$$

When the no-arbitrage condition holds, q^* lies in $(0,1)$, so it can be interpreted as a probability. This is called the **risk-neutral probability**.

With this notation,

$$S_t = e^{-r\Delta t}\big(q^* u S_t + (1-q^*) d S_t\big) = e^{-r\Delta t}\mathbb{E}^{\mathbb{Q}}\left[S_{t+1}\right],$$
$$\beta_t = e^{-r\Delta t}\mathbb{E}^{\mathbb{Q}}\left[\beta_{t+1}\right].$$

Proposition 2.4 Let C^u_{t+1} and C^d_{t+1} denote the option values in the up and down states at time $t+1$. Then there is a unique portfolio of stock and bond holdings that replicates the option payoff, and the no-arbitrage option price is

$$C_t = e^{-r\Delta t}\big(q^* C^u_{t+1} + (1-q^*) C^d_{t+1}\big) = e^{-r\Delta t}\mathbb{E}^{\mathbb{Q}}\left[C_{t+1}\right].$$

Proof. To replicate the option, choose bond and stock holdings η_t and ξ_t so that

$$\begin{bmatrix}1\\1\end{bmatrix}\eta_t + \begin{bmatrix}uS_t\\dS_t\end{bmatrix}\xi_t = \begin{bmatrix}C^u_{t+1}\\C^d_{t+1}\end{bmatrix}.$$

Equivalently, the portfolio $\Pi_t = \{\xi_t S_t, \eta_t \beta_t\}$ satisfies $\Pi_{t+1} = C_{t+1}$ in both states. By no arbitrage,

$$\begin{aligned}
C_t &= \xi_t S_t + \eta_t \beta_t \\
&= \xi_t e^{-r\Delta t}\big(q^* u S_t + (1-q^*) d S_t\big) + \eta_t e^{-r\Delta t}\big(q^* + (1-q^*)\big) \\
&= e^{-r\Delta t} q^* \big(\xi_t u S_t + \eta_t\big) + e^{-r\Delta t}(1-q^*)\big(\xi_t d S_t + \eta_t\big) \\
&= e^{-r\Delta t}\big(q^* C^u_{t+1} + (1-q^*) C^d_{t+1}\big).
\end{aligned}$$

Thus any European option can be priced by discounting its risk-neutral expected value one step backward. □

Solving the replication system gives the unique hedge ratios

$$\xi_t = \frac{C^u_{t+1} - C^d_{t+1}}{(u-d)S_t},$$
$$\eta_t = \frac{u C^d_{t+1} - d C^u_{t+1}}{u-d}.$$

Lecture 3 — Monday, January 12, 2015

A European call gives its holder the right to buy the underlying asset S at a preset strike price K at the expiry date T. A European put gives its holder the right to sell the underlying asset at the strike price K at time T. The holder enters a long position, while the writer enters a short position.

Calls and puts are financial derivatives because their values are determined by the value of the underlying asset at a future time. If the underlying price at expiry is S_T, then a call is exercised when $S_T > K$, since in that case the holder can buy the asset for K even though it is worth S_T in the market. If $S_T \leq K$, the call is not exercised because the asset can be purchased more cheaply in the market.

Thus, the payoff of a European call at expiry is

$$\text{payoff}_{\text{call}}(S_T) = \begin{cases} S_T - K, & \text{if } S_T > K, \\ 0, & \text{otherwise,} \end{cases}$$

or equivalently

$$\text{payoff}_{\text{call}}(S_T) = \max(S_T - K, 0).$$

The payoff of a European put at expiry is

$$\text{payoff}_{\text{put}}(S_T) = \begin{cases} 0, & \text{if } S_T > K, \\ K - S_T, & \text{otherwise.} \end{cases}$$

The call and put payoff profiles are plotted in Figure 5.

Many payoff patterns can be created by combining standard options. For example, holding a call and a put on the same underlying with the same strike and expiry produces a **straddle**. A **collar** can be formed by selling a call with a higher strike and using the proceeds to buy a put that offers protection if the stock price falls.

A common stock, or equity, represents an ownership share in a company. A stock may pay dividends, which are payments made to shareholders out of the company's profits. Note that if a stock pays dividends, the option holder does not receive those dividend payments.

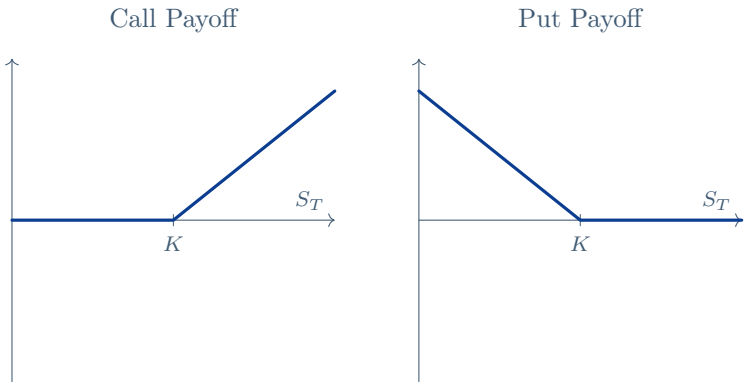

FIGURE 5

A **risk-free investment** can be made either by depositing money in the bank and earning interest or by buying riskless bonds such as treasury bills. Assume that a bond pays \$1 at maturity, that the continuously compounded interest rate $r(t)$ is known, and that the bond value at time t is B_t. Then

$$\frac{\mathrm{d}B_t}{B_t} = r(t)\,\mathrm{d}t.$$

At time $T > t$, the future value of the bond is

$$B_T = e^{\int_t^T r(s)\,\mathrm{d}s} B_t,$$

which is the **compounding** relation. Conversely, the present value of B_T dollars payable at time T is

$$B_t = e^{-\int_t^T r(s)\,\mathrm{d}s} B_T,$$

which is the **discounting** relation. In particular, if $B_T = 1$ and the interest rate is constant, then

$$B_t = e^{-\int_t^T r(s)\,\mathrm{d}s},$$

which is the **discount factor**.

An **arbitrage** is a free lunch, or getting something for nothing.

Example 3.1 From *Seinfeld* (Season 7, Episodes 21 and 22), Kramer's Coke can refund scheme is a simple toy example: in New York City the refund is 5 cents per can, whereas in Michigan it is 10 cents per can.

Two standard ways to create an arbitrage are as follows. We may construct an investment with zero cost at the present time that yields a sure positive profit in the future. Alternatively, we may

construct a portfolio of trades with negative value today whose future payoff is always nonnegative.

An asset is said to be **fairly priced** if there are no arbitrages at that price. The no-arbitrage assumption is justified by the presence of arbitrageurs, who seek simultaneous positions in different trades in order to generate guaranteed profits.

Lecture 4 — Friday, January 16, 2015

Consider again the one-period call option in Figure 6, with stock price $S_0 = 20$, price 22 in the up state, price 18 in the down state, strike $K = 21$, and expiry $T = 0.25$. The option payoff is therefore 1 in the up state and 0 in the down state.

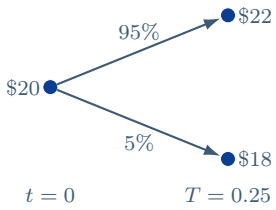

FIGURE 6

The up and down factors are

$$u = \frac{22}{20} = 1.1 \qquad \text{and} \qquad d = \frac{18}{20} = 0.9.$$

Example 4.1 If $r = 0$, then

$$q^* = \frac{e^{r\Delta t} - d}{u - d} = \frac{1 - 0.9}{1.1 - 0.9} = 0.5,$$

and hence

$$C_0 = e^{-r\Delta t}\big(q^* \cdot 1 + (1 - q^*) \cdot 0\big) = 0.5.$$

If instead $r = 12\%$, then

$$q^* = \frac{e^{0.12 \cdot 0.25} - 0.9}{1.1 - 0.9} \approx 0.6523,$$

so

$$C_0 = e^{-0.12 \cdot 0.25}(0.6523 \cdot 1 + 0) \approx 0.63.$$

To construct the replicating portfolio when $r = 0$, solve

$$\begin{bmatrix}1\\1\end{bmatrix}\eta_0 + \begin{bmatrix}22\\18\end{bmatrix}\xi_0 = \begin{bmatrix}1\\0\end{bmatrix}.$$

This gives

$$\xi_0 = \frac{C_1^u - C_1^d}{(u-d)S_0} = \frac{1}{4} \qquad \text{and} \qquad \eta_0 = -4.5.$$

Therefore the replicating portfolio at time 0 is

$$\Pi_0 = \{0.25\, S_0,\ -4.5\, \beta_0\}.$$

Example 4.2 Suppose the market price of the call is \$0.95 while $r = 0$. Then the call is overpriced relative to its no-arbitrage value of \$0.50. An arbitrageur can sell the call for \$0.95 and simultaneously form the replicating portfolio

$$\Pi_0 = \left\{\frac{1}{4}S_0,\ \text{borrow } 4.5\right\}.$$

The initial value of the combined position is

$$C_0 - \Pi_0 = 0.95 - 0.50 = 0.45,$$

so the sale proceeds exceed the cost of the replicating portfolio by \$0.45, which the arbitrageur receives at time 0. At expiry, the portfolio exactly matches the option payoff, so

$$\Pi_T - C_T = 0.$$

This is an arbitrage!

Risk-Neutral Valuation

In the one-period binomial model, no arbitrage is equivalent to the existence of a number

$$0 < q^* = \frac{e^{r\Delta t} - d}{u - d} < 1$$

such that

$$S_t = e^{-r\Delta t}\mathbb{E}_t^{\mathbb{Q}}\left[S_{t+1}\right],$$
$$\beta_t = e^{-r\Delta t}\mathbb{E}_t^{\mathbb{Q}}\left[\beta_{t+1}\right],$$

$$V_t = e^{-r\Delta t}\mathbb{E}_t^{\mathbb{Q}}\left[V_{t+1}\right],$$

where V_t is the derivative price and $\mathbb{E}_t^{\mathbb{Q}}[\cdot]$ denotes expectation under the risk-neutral probability $\mathbb{Q}$ conditional on the information at time t.

Theorem 4.3 (One-period fundamental theorem of asset pricing) There is no arbitrage if and only if there exists a risk-neutral probability under which the discounted prices of all traded assets are martingales.

Under risk-neutral pricing,

$$S_t = e^{-r\Delta t}\mathbb{E}_t^{\mathbb{Q}}\left[S_{t+1}\right]$$

implies

$$\mathbb{E}_t^{\mathbb{Q}}\left[\frac{S_{t+1}-S_t}{S_t}\right] = e^{r\Delta t}-1.$$

Similarly, whenever $C_t > 0$,

$$\mathbb{E}_t^{\mathbb{Q}}\left[\frac{C_{t+1}-C_t}{C_t}\right] = e^{r\Delta t}-1.$$

Thus, in the risk-neutral world, the expected return of both the stock and the option equals the risk-free return.

Note that the term "risk neutral" refers to the pricing measure, not to actual investor preferences. Under this measure, investors do not demand additional expected return for bearing risk.

Lecture 5 — Monday, January 19, 2015

Even though option prices are quoted in the market, it is still important to be able to compute fair values. A model-based price is needed for a new contract, pricing and hedging are tightly linked for the writer of an option, and risk measurement often requires the valuation of an option over future horizons such as one day or one week.

Recall that the one-period binomial model takes the form shown in Figure 7. with risk-neutral probability

$$q^* = \frac{e^{r\Delta t}-d}{u-d}, \quad 0 < q^* < 1,$$

and option value

$$V_t = e^{-r\Delta t}\mathbb{E}_t^{\mathbb{Q}}\left[V_{t+1}\right].$$

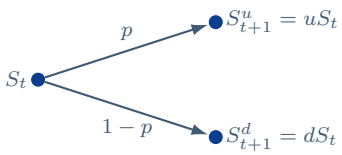

FIGURE 7

If we write $X_t = \log S_t$, then the same model becomes the log price tree in Figure 8. so the

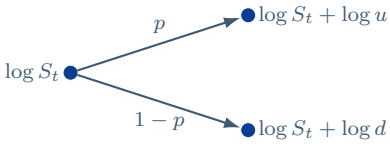

FIGURE 8

logarithmic return is

$$\log S_{t+1} - \log S_t = \log\left(\frac{S_{t+1}}{S_t}\right) = \log\left(1 + \frac{S_{t+1} - S_t}{S_t}\right) \approx \frac{S_{t+1} - S_t}{S_t}.$$

How do we determine the model (u, d)? Let α denote the expected logarithmic return per year, and let σ denote the standard deviation of the logarithmic return per year, which is the **volatility**. Over a time interval of length Δt, the expected logarithmic return is $\alpha\Delta t$ and the variance is $\sigma^2\Delta t$.

In the **Cox–Ross–Rubinstein model** from 1979, the binomial parameters are chosen as

$$\begin{aligned} u &= e^{\sigma\sqrt{\Delta t}}, \\ d &= \frac{1}{u} = e^{-\sigma\sqrt{\Delta t}}, \\ p &= \frac{1}{2} + \frac{1}{2}\frac{\alpha}{\sigma}\sqrt{\Delta t}. \end{aligned}$$

Here $\sigma > 0$, and Δt must be small enough that the displayed p lies in $[0, 1]$.

With this choice,

$$\mathbb{E}\left[\log \frac{S_{t+1}}{S_t}\right] = p \log u + (1-p) \log d = \alpha\Delta t,$$

and

$$\mathrm{Var}\left(\log\frac{S_{t+1}}{S_t}\right) = \mathbb{E}\left[\left(\log\frac{S_{t+1}}{S_t}\right)^2\right] - \left(\mathbb{E}\left[\log\frac{S_{t+1}}{S_t}\right]\right)^2 = \sigma^2\Delta t - (\alpha\Delta t)^2 \approx \sigma^2\Delta t.$$

Thus the parameters u and d depend on the volatility σ. The expected return α is irrelevant for determining today's option price. Other choices of (u, d, p) are also possible, provided that they match the desired mean and variance.

No-Arbitrage Binomial Lattice

To build a multi-period model, divide $[0, T]$ into N intervals,

$$0 < \Delta t < 2\Delta t < \cdots < N\Delta t = T,$$

and choose Δt sufficiently small so that

$$e^{-\sigma\sqrt{\Delta t}} < e^{r\Delta t} < e^{\sigma\sqrt{\Delta t}}.$$

This ensures that the no-arbitrage condition holds at each step.

The two-period recombining lattice is shown in Figure 9.

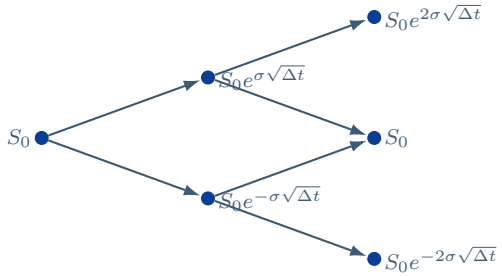

FIGURE 9

The pricing procedure has the following steps:

Step 0. Construct the stock price lattice. Let S_j^n denote the jth stock price at time step n. Then

$$S_j^n = S_0e^{(2j-n)\sigma\sqrt{\Delta t}}, \quad j = 0, 1, \ldots, n.$$

At expiry $T = t_N$, there are $N + 1$ possible stock prices.

Step 1. Set the option values V_j^N at expiry T according to the payoff(S_j^N). For a European call, for example, we set

$$V_j^N = \max(S_j^N - K, 0).$$

Step 2. Roll the values backward through the tree. At a typical node, the paired stock price and option value trees have the form shown in Figure 10. the backward induction formula is

FIGURE 10

$$V_j^n = e^{-r\Delta t}\left(q^* V_{j+1}^{n+1} + (1-q^*)V_j^{n+1}\right),$$

where

$$q^* = \frac{e^{r\Delta t} - e^{-\sigma\sqrt{\Delta t}}}{e^{\sigma\sqrt{\Delta t}} - e^{-\sigma\sqrt{\Delta t}}}.$$

This is repeated for $n = N-1, N-2, \ldots, 0$ and for each $j = 0, 1, \ldots, n$.

Remark 5.1

1. Binomial methods are widely used in practice because they are intuitive, and trinomial methods provide additional flexibility.
2. American options can also be handled naturally within a lattice.
3. Since $u = 1/d$ in the Cox–Ross–Rubinstein model, the tree is symmetric about the level $S = S_0$.

The lattice computes a full tree of option values. The work is $O(N^2)$ because backward induction updates

$$\sum_{n=0}^{N-1}(n+1) = \frac{N(N+1)}{2}$$

nodes, and each node update uses a constant number of floating-point operations. The terminal payoff initialization adds only $O(N)$ work, so the total remains $O(N^2)$.

The storage can be reduced to $O(N)$ when we only want the initial value V_0: we keep a single

working vector (or two vectors at adjacent times) of length at most $N+1$ and overwrite it level by level while marching backward in time.

This also raises the practical issue of vectorizing MATLAB code in order to reduce explicit loops; this can improve constants in runtime but does not change the asymptotic $O(N^2)$ work and $O(N)$ storage.

Convergence to Black–Scholes

As $N \to \infty$, the binomial lattice converges to a continuous limit. In logarithmic variables, the tree has the form shown in Figure 11.

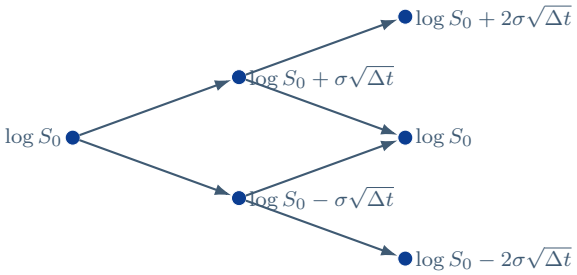

FIGURE 11

An equivalent description is

$$\log S_{t+1} = \log S_t + Z_t, \quad t = 0, 1, \ldots, N-1,$$

where

$$Z_t = \begin{cases} \sigma\sqrt{\Delta t}, & \text{with probability } q^*, \\ -\sigma\sqrt{\Delta t}, & \text{with probability } 1 - q^*. \end{cases}$$

The increments $\{Z_t\}$ are independent and identically distributed.

The two-step path probabilities are displayed in Figure 12.

Thus

$$\log S_N = \log S_0 + \sum_{t=0}^{N-1} Z_t.$$

As $N \to \infty$, the distribution of $\log S_T$ becomes normal by the central limit theorem, and the binomial call and put prices converge to the Black–Scholes formulas.

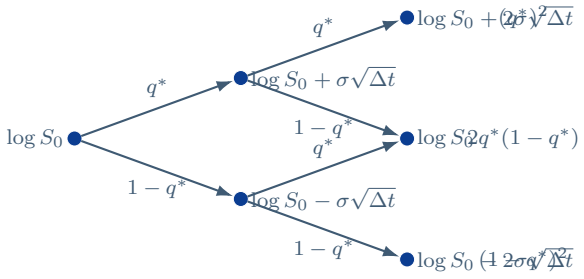

FIGURE 12

A short proof uses moment generating functions. Let $\Delta t = T/N$ and

$$Y_N := \sum_{t=0}^{N-1} Z_t.$$

For the Cox–Ross–Rubinstein probability,

$$q^* = \frac{e^{r\Delta t} - e^{-\sigma\sqrt{\Delta t}}}{e^{\sigma\sqrt{\Delta t}} - e^{-\sigma\sqrt{\Delta t}}} = \frac{1}{2} + \frac{r - \frac{1}{2}\sigma^2}{2\sigma}\sqrt{\Delta t} + O\big((\Delta t)^{3/2}\big).$$

Hence, for fixed $u \in \mathbb{R}$,

$$M_N(u) := \mathbb{E}\big[e^{uY_N}\big] = \left(q^* e^{u\sigma\sqrt{\Delta t}} + (1-q^*)e^{-u\sigma\sqrt{\Delta t}}\right)^N.$$

Using Taylor expansions of the exponential terms together with the expansion of q^*,

$$q^* e^{u\sigma\sqrt{\Delta t}} + (1-q^*)e^{-u\sigma\sqrt{\Delta t}} = 1 + \left(r - \frac{1}{2}\sigma^2\right) u\,\Delta t + \frac{1}{2}\sigma^2 u^2\,\Delta t + o(\Delta t).$$

Therefore,

$$\log M_N(u) = N \log\left(1 + \left(r - \frac{1}{2}\sigma^2\right) u\,\Delta t + \frac{1}{2}\sigma^2 u^2\,\Delta t + o(\Delta t)\right) \to \left(r - \frac{1}{2}\sigma^2\right) uT + \frac{1}{2}\sigma^2 u^2 T.$$

The limit is the moment generating function of

$$\mathcal{N}\left(\left(r - \frac{1}{2}\sigma^2\right) T,\, \sigma^2 T\right),$$

so

$$Y_N \Rightarrow \mathcal{N}\left(\left(r - \frac{1}{2}\sigma^2\right)T,\, \sigma^2 T\right),$$

$$\text{and} \qquad \log S_T = \log S_0 + Y_N \Rightarrow \mathcal{N}\left(\log S_0 + \left(r - \frac{1}{2}\sigma^2\right)T,\, \sigma^2 T\right).$$

This calculation establishes convergence of the terminal log-price distributions. Passing from that statement to convergence of option-price expectations requires an additional tail-control result, usually formulated in terms of uniform integrability. That result is beyond the scope of this course; weak convergence alone is not enough for the unbounded call payoff.

Proposition 5.2 Assume $S > 0$, $K > 0$, $\sigma > 0$, a constant interest rate r, and no dividends. For $0 \le t < T$, the **Black–Scholes value** of a European call is

$$C(S,t) = S\Phi(d_1) - Ke^{-r(T-t)}\Phi(d_2),$$

where

$$d_1 = \frac{\log(S/K) + \left(r + \frac{1}{2}\sigma^2\right)(T-t)}{\sigma\sqrt{T-t}},$$

$$d_2 = \frac{\log(S/K) + \left(r - \frac{1}{2}\sigma^2\right)(T-t)}{\sigma\sqrt{T-t}},$$

and

$$\Phi(d) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{d} e^{-y^2/2}\,\mathrm{d}y$$

is the standard normal cumulative distribution function. At $t = T$, the value is defined by the terminal payoff $C(S,T) = (S-K)^+$.

Recall that for random variables X and Y and constants α and β,

$$\mathbb{E}\left[\alpha X + \beta Y\right] = \alpha\mathbb{E}\left[X\right] + \beta\mathbb{E}\left[Y\right],$$

$$\mathrm{Var}(X) = \mathbb{E}\left[X^2\right] - \left(\mathbb{E}\left[X\right]\right)^2,$$

and

$$\mathrm{Var}(\alpha X + \beta Y) = \alpha^2\,\mathrm{Var}(X) + \beta^2\,\mathrm{Var}(Y) + 2\alpha\beta\,\mathrm{Cov}(X,Y).$$

If Z is a standard normal random variable, then

$$\mathbb{E}[Z^n] = \begin{cases} 0, & \text{if } n \text{ is odd}, \\ (n-1)!!, & \text{if } n \text{ is even}, \end{cases}$$

where $(n-1)!! = 1 \cdot 3 \cdot 5 \cdots (n-1)$ denotes the double factorial.

Notation 5.3 The notation ΔS denotes a change over a small time interval, while $\mathrm{d}S$ denotes a change over an infinitesimal time interval. The notation $f(\Delta t) = O(\Delta t)$ means that $|f(\Delta t)/\Delta t|$ remains bounded, and $f(\Delta t) = o(\Delta t)$ means that $f(\Delta t)/\Delta t \to 0$ as $\Delta t \to 0$. The symbols S_t and $S(t)$ are used interchangeably.

Finally, if f is a smooth deterministic function of t and S, then the ordinary differential rule is

$$\mathrm{d}f(t,S) = \frac{\partial f}{\partial S}\,\mathrm{d}S + \frac{\partial f}{\partial t}\,\mathrm{d}t.$$

Taylor's theorem gives

$$\Delta f = \frac{\partial f}{\partial S}\,\Delta S + \frac{\partial f}{\partial t}\,\Delta t + \frac{1}{2}\frac{\partial^2 f}{\partial S^2}^2 + \frac{1}{2}\frac{\partial^2 f}{\partial t^2}^2 + \frac{\partial^2 f}{\partial S \partial t^2}\Delta S \Delta t + \cdots.$$

Lecture 6 — Friday, January 23, 2015

2. Continuous-Time Models and Monte Carlo Pricing

Let V_0^{exact} denote the Black–Scholes value, and let $V_0^{\text{tree}}(\Delta t)$ denote the binomial lattice value with $\Delta t = T/N$. Under standard regularity conditions, binomial prices converge to V_0^{exact} and often have error of order $O(\Delta t)$. For vanilla payoffs with a kink, however, the leading error can oscillate with the placement of the strike in the terminal lattice, so a fixed expansion $V_0^{\text{tree}} = V_0^{\text{exact}} + \alpha \Delta t + o(\Delta t)$ need not hold along every sequence of meshes. This is one reason numerical convergence should be checked rather than inferred from a single grid.

Brownian Motion from Binomial Processes

Consider the special binomial process defined by $Z_0 = 0$ and

$$Z_{j+1} = Z_j + \Delta Z_j,$$

where

$$\Delta Z_j = \begin{cases} \sqrt{\Delta t}, & \text{with probability } 0.5, \\ -\sqrt{\Delta t}, & \text{with probability } 0.5. \end{cases}$$

For $i \neq j$, the increments ΔZ_i and ΔZ_j are independent. Moreover,

$$\mathbb{E}\left[\Delta Z_j\right] = 0 \qquad \text{and} \qquad \text{Var}(\Delta Z_j) = \Delta t.$$

If $\Delta t = t/m$ and

$$Z_t = Z_0 + \sum_{j=0}^{m-1} \Delta Z_j,$$

then Z_t is a discrete-time, discrete-state stochastic process with

$$\mathbb{E}\left[Z_t\right] = 0$$

and

$$\text{Var}(Z_t) = \text{Var}(Z_0) + \sum_{j=0}^{m-1} \text{Var}(\Delta Z_j) + \sum_{i \neq j} \text{Cov}(\Delta Z_i, \Delta Z_j) = t.$$

For each fixed t, the central limit theorem gives $Z_t \Rightarrow \mathcal{N}(0, t)$ as $m \to \infty$. More strongly, after a suitable continuous interpolation, the rescaled random walks converge in distribution as processes to standard Brownian motion.

Definition 6.1 A standard **Brownian motion** (or **Wiener process**) is a continuous-time process $Z(t)$ satisfying the following properties:

1. $Z(0) = 0$.
2. The sample paths $t \mapsto Z(t)$ are continuous almost surely.
3. The process has independent increments: increments over pairwise disjoint time intervals are independent.

4. For $0 \leq s < t$,
$$Z(t) - Z(s) \sim \mathcal{N}(0, t-s).$$

The notation $Z(t)$ and Z_t are used interchangeably. For each fixed interval of length Δt, the Brownian increment satisfies the distributional identity

$$\Delta Z_t \stackrel{\mathrm{d}}{=} \phi_t \sqrt{\Delta t} \text{ where } \phi_t \sim \mathcal{N}(0,1),$$

and increments on disjoint intervals use independent normal variables. The formal notation

$$\mathrm{d}Z_t = \phi_t \sqrt{\mathrm{d}t}$$

is only a mnemonic for this variance scaling; it does not define an ordinary random variable ϕ_t at each instant.

Remark 6.2 Since
$$\frac{\Delta Z_t}{\Delta t} = \frac{\phi_t}{\sqrt{\Delta t}},$$
the variance of the difference quotient diverges as $\Delta t \to 0$. This calculation suggests the process's roughness; in fact, Brownian paths are almost surely nowhere differentiable.

Next examine $(\Delta Z_t)^2$. Since $\Delta Z_t = \phi_t \sqrt{\Delta t}$,

$$\begin{aligned}
\mathbb{E}\left[(\Delta Z_t)^2\right] &= \mathbb{E}\left[\phi_t^2 \Delta t\right] = \Delta t, \\
\mathrm{Var}((\Delta Z_t)^2) &= \mathbb{E}\left[(\Delta Z_t)^4\right] - \left(\mathbb{E}\left[(\Delta Z_t)^2\right]\right)^2 \\
&= \mathbb{E}\left[\phi_t^4\right](\Delta t)^2 - (\Delta t)^2 \\
&= 3(\Delta t)^2 - (\Delta t)^2 \\
&= 2(\Delta t)^2.
\end{aligned}$$

This motivates the stochastic calculus rule

$$(\mathrm{d}Z_t)^2 = \mathrm{d}t.$$

This rule is shorthand for Brownian motion's quadratic variation: for partitions whose mesh tends to zero,

$$\sum_j \big(Z(t_{j+1}) - Z(t_j)\big)^2 \longrightarrow T$$

in probability.

Ito Integrals and Ito Processes

Definition 6.3 The **geometric Brownian motion** model is

$$\frac{\mathrm{d}S_t}{S_t} = \mu\,\mathrm{d}t + \sigma\,\mathrm{d}Z_t,$$

or equivalently

$$\mathrm{d}S_t = \mu S_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}Z_t.$$

More generally, an **Ito process** has the form

$$\mathrm{d}X_t = a(X_t, t)\,\mathrm{d}t + b(X_t, t)\,\mathrm{d}Z_t,$$

which represents

$$X_t = X_0 + \int_0^t a(X_\tau, \tau)\,\mathrm{d}\tau + \int_0^t b(X_\tau, \tau)\,\mathrm{d}Z_\tau.$$

In ordinary calculus, the Riemann integral is defined by

$$\int_0^t b(\tau)\,\mathrm{d}\tau = \lim_{N\to\infty} \sum_{j=0}^{N-1} b(t_j)(t_{j+1} - t_j).$$

Figure 13 depicts the left endpoint rectangle used in one term of this Riemann sum.

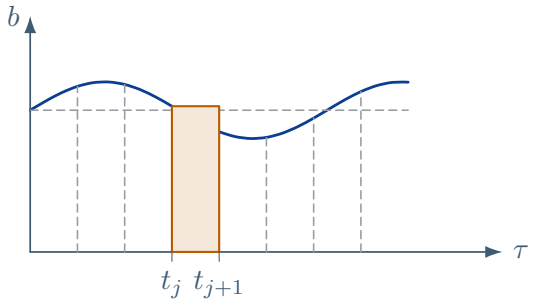

FIGURE 13

For a deterministic continuous integrand b, the **Ito integral** is defined by

$$\int_0^t b(\tau)\,\mathrm{d}Z_\tau = \lim_{N\to\infty} \sum_{j=0}^{N-1} b(t_j)\big(Z_{t_{j+1}} - Z_{t_j}\big),$$

where the limit is interpreted in mean square:

$$\lim_{N\to\infty} \mathbb{E}\left[\left(\int_0^t b(\tau)\,\mathrm{d}Z_\tau - \sum_{j=0}^{N-1} b(t_j)\big(Z_{t_{j+1}} - Z_{t_j}\big)\right)^2\right] = 0.$$

In the Ito integral, the integrand $b(\tau)$ is evaluated at the *left* endpoint t_j of each interval $[t_j, t_{j+1}]$. For a random integrand, the same construction requires it to be adapted (more precisely, progressively measurable) and square integrable.

Clearly,

$$\int_0^t \mathrm{d}Z_\tau = Z_t - Z_0.$$

If a and b are constants in

$$\mathrm{d}X_t = a\,\mathrm{d}t + b\,\mathrm{d}Z_t,$$

then

$$X_t = X_0 + at + bZ_t.$$

Ito's Lemma and the Black–Scholes PDE

Definition 6.4 The process S_t is a **geometric Brownian motion** if

$$\frac{\mathrm{d}S_t}{S_t} = \mu\,\mathrm{d}t + \sigma\,\mathrm{d}Z_t,$$

where Z_t is a standard Brownian motion. In the Black–Scholes model, the left-hand side is the instantaneous relative return, $\mu\,\mathrm{d}t$ is the deterministic **drift**, and $\sigma\,\mathrm{d}Z_t$ is the stochastic diffusion. The parameter σ is the **volatility**. Ito's lemma gives the corresponding logarithmic return,

$$\mathrm{d}\log S_t = \left(\mu - \frac{\sigma^2}{2}\right)\mathrm{d}t + \sigma\,\mathrm{d}Z_t.$$

Thus,

$$\mathbb{E}\left[\frac{\mathrm{d}S_t}{S_t}\right] = \mu\,\mathrm{d}t \qquad \text{and} \qquad \mathrm{Var}\left(\frac{\mathrm{d}S_t}{S_t}\right) = \sigma^2\,\mathrm{d}t.$$

The geometric Brownian motion can also be written as

$$\mathrm{d}S_t = a(S_t, t)S_t\,\mathrm{d}t + b(S_t, t)S_t\,\mathrm{d}Z_t,$$

where $a(S,t) = \mu$ and $b(S,t) = \sigma$ are constants. More generally, we can allow a and b to depend on S and t, which gives the generalized Black–Scholes model. The option value has the form $V(S_t, t)$. We ask how V evolves when S_t evolves according to the geometric Brownian motion.

For an ordinary smooth function $y = f(S,t)$ of deterministic variables,

$$\Delta y = \frac{\partial f}{\partial S}\Delta S + \frac{\partial f}{\partial t}\Delta t + \frac{1}{2}\frac{\partial^2 f}{\partial S^2}^2 + \frac{1}{2}\frac{\partial^2 f}{\partial t^2}^2 + \frac{\partial^2 f}{\partial S \partial t}(\Delta t) + \cdots,$$

so formally

$$\mathrm{d}y = \frac{\partial f}{\partial S}\,\mathrm{d}S + \frac{\partial f}{\partial t}\,\mathrm{d}t + o(\mathrm{d}t).$$

When S_t is itself an Ito process, the quadratic term $(\mathrm{d}S)^2$ contributes at order $\mathrm{d}t$ and must be retained.

Theorem 6.5 (Ito's lemma) Suppose that S_t satisfies

$$\mathrm{d}S_t = a(S_t,t)\,\mathrm{d}t + b(S_t,t)\,\mathrm{d}Z_t.$$

Let G have one continuous derivative in time and two continuous derivatives in its state variable on a domain containing (S_t,t), and define $Y_t = G(S_t,t)$. Then Y_t is an Ito process satisfying

$$\mathrm{d}Y_t = \left[\frac{\partial G}{\partial t} + a(S_t,t)\frac{\partial G}{\partial S} + \frac{1}{2}b(S_t,t)^2\frac{\partial^2 G}{\partial S^2}\right]\mathrm{d}t + b(S_t,t)\frac{\partial G}{\partial S}\,\mathrm{d}Z_t.$$

Proof sketch. Write $a = a(S_t,t)$ and $b = b(S_t,t)$ for brevity. Applying Taylor expansion along a refining partition and retaining the terms that survive in probability gives formally

$$\mathrm{d}G = \frac{\partial G}{\partial t}\,\mathrm{d}t + \frac{\partial G}{\partial S}\,\mathrm{d}S + \frac{1}{2}\frac{\partial^2 G}{\partial S^2}^2.$$

Since

$$(\mathrm{d}S)^2 = (a\,\mathrm{d}t + b\,\mathrm{d}Z_t)^2 = a^2(\mathrm{d}t)^2 + 2ab\,\mathrm{d}t\,\mathrm{d}Z_t + b^2(\mathrm{d}Z_t)^2 = b^2\,\mathrm{d}t,$$

after discarding higher-order terms and using $(\mathrm{d}Z_t)^2 = \mathrm{d}t$, it follows that

$$\mathrm{d}G = \frac{\partial G}{\partial t}\,\mathrm{d}t + \frac{\partial G}{\partial S}\,\mathrm{d}S + \frac{1}{2}\frac{\partial^2 G}{\partial S^2}b^2\,\mathrm{d}t.$$

Substituting $\mathrm{d}S = a\,\mathrm{d}t + b\,\mathrm{d}Z_t$ yields the stated formula. □

Example 6.6 Suppose that

$$\mathrm{d}S_t = \mu S_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}Z_t$$

and take $G(S,t) = \log S$. Then

$$\frac{\partial G}{\partial t} = 0, \qquad \frac{\partial G}{\partial S} = \frac{1}{S}, \qquad \text{and} \qquad \frac{\partial^2 G}{\partial S^2} = -\frac{1}{S^2}.$$

Ito's lemma gives

$$\mathrm{d}\log S_t = \frac{1}{S}\left(\mu S\,\mathrm{d}t + \sigma S\,\mathrm{d}Z_t\right) \quad -\frac{1}{2}\frac{1}{S^2}\sigma^2 S^2\,\mathrm{d}t = \left(\mu - \frac{1}{2}\sigma^2\right)\mathrm{d}t + \sigma\,\mathrm{d}Z_t.$$

Integrating,

$$\log S_t = \log S_0 + \left(\mu - \frac{1}{2}\sigma^2\right)t + \sigma Z_t,$$

so

$$S_t = S_0 e^{\left(\mu - \frac{1}{2}\sigma^2\right)t + \sigma Z_t}.$$

If $S_0 > 0$, the stock price is therefore always positive.

Assume the generalized Black–Scholes model

$$\frac{\mathrm{d}S_t}{S_t} = \mu(S_t,t)\,\mathrm{d}t + \sigma(S_t,t)\,\mathrm{d}Z_t.$$

Assume continuous trading, infinitely divisible assets, no dividends, short selling without transaction costs, no arbitrage, and a constant interest rate.

Let $V(S_t,t)$ denote the value of a European option at time t when the underlying price is S_t. Applying Ito's lemma with $a(S,t) = \mu S$ and $b(S,t) = \sigma S$ gives

$$\mathrm{d}V = \left(\mu S\frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2} + \frac{\partial V}{\partial t}\right)\mathrm{d}t + \sigma S\frac{\partial V}{\partial S}\,\mathrm{d}Z_t.$$

At time t, choose the stock holding α_t and hold it fixed over the next infinitesimal interval. The locally hedged position has value

$$\Pi_t = \{V_t,\ -\alpha_t S_t\}.$$

Its gain over that interval is

$$\Delta\Pi_t = \mathrm{d}V_t - \alpha_t\,\mathrm{d}S_t = \left(\mu S\frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2} + \frac{\partial V}{\partial t} - \mu\alpha_t S\right)\mathrm{d}t + \sigma S\left(\frac{\partial V}{\partial S} - \alpha_t\right)\mathrm{d}Z_t. \quad (6.1)$$

Choosing

$$\alpha_t = \frac{\partial V}{\partial S}$$

eliminates the random term, so the portfolio becomes locally riskless and satisfies

$$\Delta\Pi_t = \left(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2}\right) \mathrm{d}t.$$

No arbitrage then requires this riskless portfolio to earn the risk-free rate:

$$\Delta\Pi_t = r\Pi_t \,\mathrm{d}t = r\left(V - S\frac{\partial V}{\partial S}\right) \mathrm{d}t. \tag{6.2}$$

Combining (6.1) and (6.2), we see that V must satisfy

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} - rV + rS\frac{\partial V}{\partial S} = 0 \tag{6.3}$$

for $0 \le t < T$ and $0 < S < \infty$. (6.3) is the **Black–Scholes PDE**. The terminal condition is $V(S,T) = \text{payoff}(S)$, together with boundary conditions appropriate to the contract. For a European call, for example, $V(0,t) = 0$ and $V(S,t) \sim S - Ke^{-r(T-t)}$ as $S \to \infty$.

Any European option whose payoff is a function of the underlying asset price S_t satisfies this PDE. The Black–Scholes PDE is an expression of no arbitrage, and the expected return μ does not appear in it.

Risk-Neutral Valuation

Starting from

$$\frac{\mathrm{d}S_t}{S_t} = \mu\,\mathrm{d}t + \sigma\,\mathrm{d}Z_t,$$

the risk-neutral model replaces μ by r and replaces Z_t by another Brownian motion $Z_t^{\mathbb{Q}}$:

$$\frac{\mathrm{d}S_t}{S_t} = r\,\mathrm{d}t + \sigma\,\mathrm{d}Z_t^{\mathbb{Q}}.$$

Under these dynamics, the European option value is

$$V_t = e^{-r(T-t)}\mathbb{E}_t^{\mathbb{Q}}\left[\text{payoff}(S_T)\right],$$

which is called a **risk-neutral valuation**.

This connects to the binomial lattice with risk-neutral step distribution

$$\log S_{t+1} - \log S_t = \begin{cases} \sigma\sqrt{\Delta t}, & \text{with probability } q^*, \\ -\sigma\sqrt{\Delta t}, & \text{with probability } 1 - q^*. \end{cases}$$

Under this measure, it can be shown that

$$V_t = e^{-r(T-t)}\mathbb{E}_t^{\mathbb{Q}}\left[\text{payoff}(S_T)\right],$$

and as $\Delta t \to 0$ the lattice limit becomes

$$\mathrm{d}\log S_t = \left(r - \frac{1}{2}\sigma^2\right)\mathrm{d}t + \sigma\,\mathrm{d}Z_t^{\mathbb{Q}},$$

or equivalently

$$\frac{\mathrm{d}S_t}{S_t} = r\,\mathrm{d}t + \sigma\,\mathrm{d}Z_t^{\mathbb{Q}}.$$

Lecture 7 — Wednesday, January 28, 2015

Monte Carlo Pricing

For a European payoff at time T, the option value is

$$V_t = e^{-r(T-t)}\mathbb{E}_t^{\mathbb{Q}}\left[\text{payoff}\right],$$

where the expectation is taken under the risk-neutral dynamics

$$\frac{\mathrm{d}S_t}{S_t} = r\,\mathrm{d}t + \sigma\,\mathrm{d}Z_t^{\mathbb{Q}}.$$

Assume that we have the ability to generate independent standard normal samples (we will address this later). We want to simulate sample paths of the Black–Scholes model and use them for pricing and risk analysis.

To simulate one Black–Scholes path, choose a time step $\delta t = T/N$ and generate independent normal samples $\phi_t \sim \mathcal{N}(0,1)$. Then update the stock price by

$$S_{t+1} = S_t \exp\left(\left(r - \frac{1}{2}\sigma^2\right)\delta t + \sigma\sqrt{\delta t}\,\phi_t\right).$$

To price an option, generate many independent paths in this way. If the corresponding payoffs are $(\text{payoff})^j$, $j = 1, \ldots, M$, then using the law of large numbers,

$$V_0 = e^{-rT}\mathbb{E}^{\mathbb{Q}}\left[\text{payoff}\right] \approx \frac{e^{-rT}}{M}\sum_{j=1}^{M}(\text{payoff})^j.$$

If the payoff depends only on S_T, then $(\text{payoff})^j = \text{payoff}(S_T^j)$. The same framework also prices more complicated path-dependent options.

Path-Dependent (Exotic) Options

A **path-dependent option** (or **exotic option**) has a payoff or contractual feature that depends on the entire price path $\{S_t : 0 \le t \le T\}$ rather than only on S_T.

Example 7.1 A **barrier option** comes into existence or ceases to exist when a barrier is crossed. For an **up-and-out** barrier option, if the path S_t, for $0 \le t \le T$, ever crosses a barrier level S_u, then the option pays nothing; otherwise it pays the usual payoff. A representative path that does not breach the barrier appears in Figure 14.

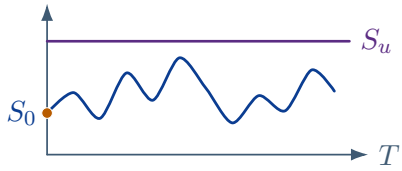

FIGURE 14

Example 7.2 An **Asian option** depends on an average price. For example,

$$\text{payoff}(A_T) = \max(A_T - K, 0) \text{ where } A_T = \frac{1}{T}\int_0^T S_t\,\mathrm{d}t.$$

If $t_n = t_{n-1} + \delta t$ and $N\delta t = T$, then we may simulate

$$S^n = S^{n-1}\exp\left(\left(r - \frac{1}{2}\sigma^2\right)\delta t + \sigma\sqrt{\delta t}\,\phi^n\right), \quad \phi^n \sim \mathcal{N}(0,1),$$

and approximate the average by

$$A^N = \frac{1}{N}\sum_{k=0}^{N-1} S^k \approx \frac{1}{T}\int_0^T S_t \, \mathrm{d}t.$$

The running average can be updated recursively as

$$A^{n+1} = \frac{n}{n+1}A^n + \frac{1}{n+1}S^n.$$

In MATLAB, we can implement this recursion with vectorized path updates as follows:

```
S = S0*ones(M,1);
A = zeros(M,1);

for n = 0:(N-1)
    A = (n/(n+1))*A + (1/(n+1))*S;
    S = S .* exp((r - 0.5*sigma^2)*dt + sigma*sqrt(dt)*randn(M,1));
end

payoff = max(A - K, 0);
V0 = mean(exp(-r*T) * payoff);
```

Note that the intermediate path values do not all need to be stored when pricing an Asian option; we only need to keep track of the current stock price and the current running average.

Errors in Monte Carlo Option Pricing

There are two main sources of error in Monte Carlo option pricing: sampling error and time discretization error. We first address sampling error.

For pricing a European option with payoff payoff(S_T) using $S_T = S_0 e^{(r-\frac{1}{2}\sigma^2)T+\sigma Z_T}$, let $S_T^1, \ldots, S_T^M$ be independent samples of S_T. The only source of error from the true option value is the sampling error in approximating $\mathbb{E}\left[\text{payoff}(S_T)\right]$ by the sample mean

$$V^M = \frac{1}{M}\sum_{j=1}^{M} \text{payoff}(S_T^j).$$

The sampling error is $O(M^{-1/2})$:

Proposition 7.3 Let

$$V^M = \frac{X^1 + \cdots + X^M}{M},$$

where $X^1, \ldots, X^M$ are independent and identically distributed copies of a random variable X with $\mathbb{E}[X^2] < \infty$, such as $X = \text{payoff}(S_T)$. Then V^M is an unbiased estimator of $\mathbb{E}[X]$, and its standard deviation is of order $M^{-1/2}$.

Proof. The estimator is unbiased because

$$\mathbb{E}\left[V^M\right] = \frac{1}{M}\sum_{j=1}^{M}\mathbb{E}\left[X^j\right] = \frac{1}{M}\sum_{j=1}^{M}\mathbb{E}[X] = \mathbb{E}[X].$$

Since the samples are independent,

$$\text{Var}(V^M) = \frac{1}{M}\text{Var}(X) = \frac{\sigma_X^2}{M},$$

where $\sigma_X^2 = \text{Var}(X)$. Therefore

$$\text{std}(V^M) = \frac{\sigma_X}{\sqrt{M}}.$$

□

Thus the sampling error is $O(M^{-1/2})$. Reducing the sampling error by a factor of 10 requires 100 times as many simulations! On the other hand, the convergence rate does not depend on the number of underlying risk factors, which is one reason Monte Carlo methods are attractive for high-dimensional problems such as basket options.

Using the central limit theorem,

$$\sqrt{M}\left(V^M - \mathbb{E}[X]\right) \xrightarrow{\text{d}} \mathcal{N}(0, \sigma_X^2).$$

Consequently, a 95% confidence interval is approximately

$$V^M - \frac{1.96\sigma_X}{\sqrt{M}} \le \mathbb{E}[X] \le V^M + \frac{1.96\sigma_X}{\sqrt{M}}.$$

The unknown variance can be estimated from the samples by

$$\sigma_X^2 \approx \frac{1}{M-1}\sum_{j=1}^{M}(X^j - V^M)^2.$$

We next address time discretization error. For a European option in Black–Scholes, there is no time discretization error if S_T is sampled from its exact lognormal distribution. For an Asian option, however, the continuous average is approximated by a discrete sum. Under the regularity assumptions behind the usual weak-error analysis, this introduces bias of order $O(\delta t)$. Since the root-mean-square sampling error is $O(M^{-1/2})$, the total root-mean-square error behaves like

$$O\left(\max\left(\delta t, \frac{1}{\sqrt{M}}\right)\right).$$

It is therefore wasteful to drive the sampling error far below the time discretization error. To make the two errors decrease at the same rate, choose

$$M = O\left(\frac{1}{\delta t^2}\right).$$

Then the total error is $O(\delta t)$.

If the total computational work is of order $O(MN) = O(M/\delta t)$, then with this balancing choice the computational complexity becomes

$$O\left(\frac{1}{\delta t^3}\right),$$

so

$$\delta t = O\left(\frac{1}{(\text{complexity})^{1/3}}\right).$$

Accordingly, the error is

$$O(\delta t) = O\left(\frac{1}{(\text{complexity})^{1/3}}\right).$$

Thus, to reduce the error by a factor of 10, we must increase the work by a factor of 10^3.

Remark 7.4 Monte Carlo methods are easy to implement and reliable, and they handle multiple assets and European path-dependent options naturally. Their dimension-independent sampling rate makes them especially attractive when many random factors are present. Their disadvantages are that they do not handle American options easily and can be inefficient for

low-dimensional problems.

Hedging and Dynamic Trading

Black–Scholes theory suggests that option risk can be eliminated by continuous trading: to hedge a short position in an option V, we hold $\partial V/\partial S$ units of the underlying asset at each instant. The quantity

$$\frac{\partial V}{\partial S} = \lim_{\delta S \to 0} \frac{V(S+\delta S, t) - V(S,t)}{\delta S}$$

is called the **option delta**, and it measures the sensitivity of the option value to the underlying price.

In practice, continuous trading is impossible (even if you are Jane Street), and transaction costs are present. This leads to dynamic hedging questions such as how much risk remains under daily, weekly, or monthly rebalancing, and what profit or loss is generated by a given trading strategy.

To hedge a short position in an option $V(S,t)$, fix a rebalancing time and consider the portfolio

$$\Pi = \{-V,\ \alpha S,\ B\},$$

where $\alpha = \partial V/\partial S$ is the number of shares chosen at that instant and B is a bank account used to finance the trades. Between rebalancing times, α and B are held fixed. At the instant the hedge is set, its value is

$$\Pi(S,t) = -V(S,t) + \alpha S + B(t).$$

Since

$$\frac{\partial \Pi}{\partial S} = -\frac{\partial V}{\partial S} + \alpha = 0,$$

the portfolio is instantaneously insensitive to changes in the underlying price.

Now, assume that the trading times are

$$0 = t_0 < t_1 < \cdots < t_N = T = N\delta t,$$

and that $\alpha(t_k)$ units of the asset are held over the interval $(t_k, t_{k+1}]$, as shown in Figure 15.

For delta hedging a short position in V, start at $t_0 = 0$ with

$$\Pi(0) = \{-V,\ \alpha(0)S(0),\ B(0)\},$$

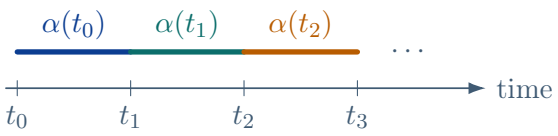

FIGURE 15

and choose

$$B(0) = V(S(0), 0) - \alpha(t_0)S(0),$$

so that $\Pi(0) = 0$. At time t_k, the portfolio value immediately before rebalancing is

$$-V(S(t_k), t_k) + \alpha(t_{k-1})S(t_k) + B(t_{k-1})e^{r\delta t}.$$

Immediately after rebalancing, the self-financing condition gives

$$-V(S(t_k), t_k) + \alpha(t_k)S(t_k) + B(t_k),$$

so

$$B(t_k) = B(t_{k-1})e^{r\delta t} + \big(\alpha(t_{k-1}) - \alpha(t_k)\big)S(t_k).$$

If $\alpha(t_k) > \alpha(t_{k-1})$, then $\alpha(t_k) - \alpha(t_{k-1})$ additional shares must be bought; if $\alpha(t_k) < \alpha(t_{k-1})$, then $\alpha(t_{k-1}) - \alpha(t_k)$ shares must be sold. At the final time $t = t_N = T$, the portfolio value is

$$\Pi(T) = -V(S(T), T) + \alpha(t_{N-1})S(T) + B(t_{N-1})e^{r\delta t}.$$

This terminal value is random. If it is positive, there is a profit; if it is negative, there is a loss. It is also the hedging error. The discounted relative profit and loss (or **relative hedge error**) is

$$\text{P\&L} = \frac{e^{-rT}\Pi(T)}{V(S_0, 0)}.$$

If $\Pi(T) \equiv 0$, the hedge is perfect. Studying the distribution of $\Pi(T)$ provides information about the quality of the strategy.

The same self-financing recursion applies to path-dependent options. Given $\alpha(t_0), \ldots, \alpha(t_{N-1})$, the self-financing process for hedging a short position in V is

$$B(0) = V(S(0), 0) - \alpha(t_0)S(0)$$

so

$$\Pi(0) = -V(S(0), 0) + \alpha(t_0)S(0) + B(0) = 0,$$

and for $k = 1, \ldots, N-1$,

$$B(t_k) = B(t_{k-1})e^{r\delta t} + \big(\alpha(t_{k-1}) - \alpha(t_k)\big)S(t_k),$$

with

$$\Pi(T) = -V(S_T, T) + \alpha(t_{N-1})S(T) + B(t_{N-1})e^{r\delta t}.$$

For delta hedging we set

$$\alpha(t_k) = \frac{\partial V(S_{t_k}, t_k)}{\partial S}.$$

If an analytic pricing formula is available, this derivative can be computed directly. In a binomial lattice, we can approximate delta from the replicating portfolio equations

$$\begin{cases} \alpha_t S^u_{t+1} + B_t = V^u_{t+1}, \\ \alpha_t S^d_{t+1} + B_t = V^d_{t+1}, \end{cases}$$

which gives

$$\alpha_t = \frac{V^u_{t+1} - V^d_{t+1}}{uS_t - dS_t} \approx \frac{\partial V}{\partial S}.$$

Using Monte Carlo, we may generally approximate delta by a **centered finite difference**,

$$\frac{\partial V}{\partial S} \approx \frac{V(S+h, t) - V(S-h, t)}{2h},$$

whose deterministic truncation error is $O(h^2)$ when V is sufficiently smooth. In simulation, we should use common random numbers for the two valuations so that sampling noise does not swamp their difference.

An option's **gamma** is defined by

$$\frac{\partial^2 V}{\partial S^2},$$

and measures second-order sensitivity to the underlying asset. To form a **delta–gamma neutral** portfolio, introduce another derivative instrument $I(S, t)$ with nonzero gamma and choose a portfolio

$$\Pi = \{-V,\ \alpha S,\ \beta I,\ B\}$$

so that

$$\frac{\partial \Pi}{\partial S} = 0 \qquad \text{and} \qquad \frac{\partial^2 \Pi}{\partial S^2} = 0.$$

This yields

$$\beta = \frac{\partial^2 V / \partial S^2}{\partial^2 I / \partial S^2} \qquad \text{and} \qquad \alpha = \frac{\partial V}{\partial S} - \beta \frac{\partial I}{\partial S}.$$

Which instrument I should we choose? A natural choice is a vanilla option with the same underlying and maturity as V but a different strike price. This is called **gamma hedging**.

Under the Black–Scholes model, the volatility σ is the key parameter that is not directly observable, and in practice it is not constant. The sensitivity

$$\frac{\partial V}{\partial \sigma}$$

is called the option's **vega**. To build a **delta–vega neutral** portfolio, take

$$P(t) = -V + \alpha^h S + \beta I + B(t)$$

and impose

$$\frac{\partial P}{\partial S} = -\frac{\partial V}{\partial S} + \alpha^h + \beta \frac{\partial I}{\partial S} = 0,$$
$$\frac{\partial P}{\partial \sigma} = -\frac{\partial V}{\partial \sigma} + \beta \frac{\partial I}{\partial \sigma} = 0.$$

This gives the setup for choosing the hedge ratios.

Lecture 8 — Monday, February 02, 2015

3. Simulation and Risk Measurement

Pseudorandom Number Generation

A **pseudorandom number generator** is a deterministic algorithm that produces a sequence intended to behave like independent and identically distributed random samples. Once its seed is fixed, a software generator produces a deterministic and reproducible sequence that mimics randomness.

Uniform random variables on $[0, 1]$ provide the starting point. If $u_1, u_2, u_3, \ldots$ is such a sequence, then independence means that later values do not depend on earlier ones, while identical distribution means that, over repeated runs, the histogram of each u_j should look the same. Once good uniform samples are available, we can transform them into samples from other distributions.

Definition 8.1 A **linear congruential generator** (**LCG**) starts from an integer seed x_0 and updates by

$$x_i = ax_{i-1} + b \pmod{m} \qquad \text{and} \qquad u_i = \frac{x_i}{m},$$

where a is the multiplier, b is the offset, and m is the modulus. Here "(mod m)" means that we keep the remainder after division by m.

Example 8.2 The historically notorious **RANDU** generator used on early IBM machines took

$$a = 2^{16} + 3 = 65539, \qquad m = 2^{31}, \qquad \text{and} \qquad b = 0.$$

It was later shown to have severe correlation defects. A better alternative is the minimal standard generator

$$a = 7^5 = 16807, \qquad m = 2^{31} - 1, \qquad \text{and} \qquad b = 0,$$

whose repetition period is the maximal possible value $2^{31} - 2$. The modulus $2^{31} - 1$ was proved to be a **Mersenne prime** by Leonhard Euler in 1772.

The Inversion Method

Theorem 8.3 Let $U \sim \text{Unif}[0, 1]$, and let F be a continuous strictly increasing cumulative distribution function. Then the random variable

$$F^{-1}(U)$$

has distribution function F.

Proof. For any real number x, strict monotonicity of F gives

$$\mathbb{P}\big(F^{-1}(U) \le x\big) = \mathbb{P}\big(U \le F(x)\big).$$

Since U is uniform on $[0, 1]$, we have $\mathbb{P}(U \le u) = u$ for $0 \le u \le 1$. Therefore

$$\mathbb{P}\big(F^{-1}(U) \le x\big) = F(x),$$

which is exactly the statement that $F^{-1}(U)$ has cumulative distribution function F. □

Example 8.4 Consider the exponential density

$$p(x) = \frac{1}{2}e^{-x/2}, \quad x \geq 0.$$

Its cumulative distribution function is

$$F(x) = \int_0^x \frac{1}{2}e^{-x'/2}\,\mathrm{d}x' = 1 - e^{-x/2}.$$

Solving $u = F(x)$ for x gives

$$F^{-1}(u) = -2\log(1-u).$$

Hence, if the U_i are independent Unif$[0,1]$ samples, then the variables $-2\log(1-U_i)$ are independent and exponentially distributed.

The density of the $\mathcal{N}(\mu, \sigma^2)$ distribution is

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma}e^{-(x-\mu)^2/(2\sigma^2)},$$

and its cumulative distribution function is

$$F(x) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{x-\mu}{\sqrt{2}\sigma}\right)\right),$$

where $\operatorname{erf}(z) = \frac{2}{\sqrt{\pi}}\int_0^z e^{-t^2}\,\mathrm{d}t$ is the error function. In principle, the inversion method can be used for normal random variables, but the normal cumulative distribution function does not have a convenient closed-form inverse. MATLAB's `randn` provides standard normal pseudorandom samples generated from MATLAB's current random-number stream.

Pseudorandom numbers and quasirandom numbers differ in an important way. A sequence of **quasirandom numbers** sacrifices independence in order to fill the sampling space more evenly, which is the basis for **quasi-Monte Carlo methods**. In a path simulation with M time increments, the vector of increments must be viewed as a point in M-dimensional space, so a **low-discrepancy sequence** such as the **Halton sequence** assigns a different prime base to each coordinate direction. Figure 16 compares the resulting coverage with pseudorandom sampling.

What if we want to generate correlated samples? If X is a random variable with mean μ and positive variance σ^2, then the standardized variable

$$Z = \frac{X-\mu}{\sigma}$$

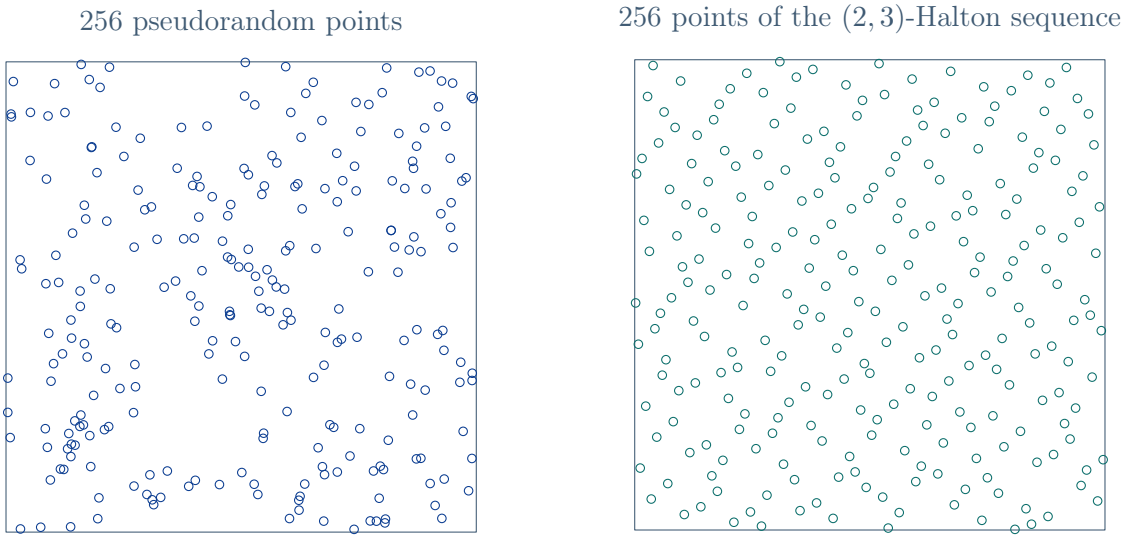

FIGURE 16

has mean 0 and variance 1. If X_1 and X_2 are independent with means μ_1 and μ_2 and positive variances σ_1^2 and σ_2^2, then the standardized variables

$$Z_1 = \frac{X_1 - \mu_1}{\sigma_1} \qquad \text{and} \qquad Z_2 = \frac{X_2 - \mu_2}{\sigma_2}$$

are independent standardized random variables. To generate correlated samples, we can take linear combinations of independent samples. For example, if X_1 and X_2 are independent standard normal variables, then

$$Y_1 = \rho X_1 + \sqrt{1 - \rho^2}\, X_2, \qquad -1 \le \rho \le 1,$$

is a standard normal variable with correlation ρ with X_1. More generally, let $\mathbf{C}$ be a symmetric positive-semidefinite correlation matrix and choose $\mathbf{U}$ so that

$$\mathbf{U}^\top \mathbf{U} = \mathbf{C}.$$

Such a factor always exists, although it may be singular. If the rows of $\mathbf{R}$ are independent standardized random vectors with covariance $\mathbf{I}$, then samples with covariance $\mathbf{C}$ are formed by

$$\mathbf{R}_c = \mathbf{R}\mathbf{U}.$$

Cholesky decomposition when $\mathbf{C}$ is positive definite, or eigenvalue decomposition in the semidefinite case, provides a standard way to construct $\mathbf{U}$.

Variance Reduction: Antithetic Sampling

Variance reduction aims to improve accuracy without increasing the number of simulated paths proportionally. There are many methods available, including control variates, importance sampling, and stratified sampling. One simple and often effective technique is the **antithetic method**.

In this method, two payoffs are simulated using the same underlying random input but with opposite signs, and then the two results are averaged. This can reduce variance because the two payoffs tend to be negatively correlated, especially when the payoff is reasonably monotone in the driving random input. For example, if the payoff depends only on S_T, then we can simulate S_T using ϕ_j and $-\phi_j$ for the same j, and then average the two payoffs to get a better estimate of the option value.

More generally, let $\phi_1, \ldots, \phi_M$ be independent standard normal samples. For each ϕ_j, simulate one payoff using ϕ_j and a second payoff using $-\phi_j$, and then average the two results. If the payoff depends only on S_T, this gives the estimator

$$V_0 \approx \frac{e^{-rT}}{2M} \sum_{j=1}^{M} \left[\text{payoff}\left(S_0 e^{(r-\frac{1}{2}\sigma^2)T+\sigma\sqrt{T}\,\phi_j} \right) + \text{payoff}\left(S_0 e^{(r-\frac{1}{2}\sigma^2)T-\sigma\sqrt{T}\,\phi_j} \right) \right].$$

The intuition is that when a simulated shock lies above its mean, the paired shock with opposite sign lies below it. Averaging the two corresponding payoffs can therefore reduce variance, especially when the payoff is reasonably monotone in the driving random input.

Lecture 9 — Monday, February 09, 2015

The historical record is not lacking in examples of financial disasters. In 1987, the Dow Jones Industrial Average dropped by more than 20% in a single day. In 1994, Orange County, California, filed for bankruptcy after losing $1.6 billion on risky derivatives. In 1995, Barings Bank, the oldest merchant bank in the United Kingdom, collapsed after a rogue trader lost £700 million on unauthorized derivatives trades. In 1998, Long-Term Capital Management nearly failed after losing $4.6 billion in less than four months on highly leveraged positions in derivatives. In 2008, the global financial crisis was triggered by the collapse of the subprime mortgage market, which was fueled by complex derivatives such as collateralized debt obligations (CDOs) and credit default swaps (CDSs).

In 2020, the COVID-19 pandemic caused unprecedented market volatility and economic disruption. On February 20, 2020, stock markets across the world crashed: the S&P 500 index experienced a rapid decline of over 30% in just a few weeks, and many companies faced liquidity crises, plunging the global economy into a recession.

These events highlight the importance of risk management in finance.

Market risk is the risk that the value of a financial position changes because underlying market variables, such as stock prices, interest rates, exchange rates, or commodity prices, move. **Credit risk** is the risk that promised payments are not received because a borrower or counterparty defaults. **Operational risk** is the risk arising from failed or inadequate processes, people, or systems.

For a portfolio containing derivatives, the return distribution is typically neither normal nor even symmetric about its mean. If $V(S,t)$ denotes the price of a call option and the stock satisfies

$$\frac{\mathrm{d}S_t}{S_t} = \mu\,\mathrm{d}t + \sigma\,\mathrm{d}W_t,$$

then Ito's lemma gives

$$\mathrm{d}V_t = \left(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + \mu S \frac{\partial V}{\partial S}\right)\mathrm{d}t + \sigma S \frac{\partial V}{\partial S}\,\mathrm{d}W_t.$$

This nonlinear dependence on the underlying asset price already shows why standard deviation alone is not an adequate summary of portfolio risk when derivative positions are present.

Risk management should focus on unexpected, abnormal, or extreme outcomes rather than merely average behavior. Alan Greenspan said it himself in 1995: "Improving the characterization of the distribution of extreme values is of paramount importance." You wouldn't ignore Alan Greenspan, would you?

In his 2007 book *The Black Swan*, certified moron Nassim Nicholas Taleb coined the term **Black Swan** to describe rare and unpredictable events with severe consequences. The term "Black Swan" comes from the 17th century European assumption that all swans were white; a black swan was thus a symbol for something that was thought to be impossible. In the 18th century, the discovery of black swans in Australia metamorphosed the term to imply that a perceived impossibility may actually come to pass.

In the wake of the 1987 crash, Dennis Weatherstone, then chairman of J.P. Morgan, required that a one-page report be produced at 4:15 p.m. each day summarizing the firm's risk exposure, which

became known as the "4:15 report". The methodology used in the report to estimate potential large losses has become known as **value-at-risk** (**VaR**). The 4:15 report was a pioneering effort to quantify and manage market risk, and it played a significant role in the development of modern risk management practices in the financial industry. The RiskMetrics framework, which was developed by J.P. Morgan in the 1990s, is a widely used approach to calculating VaR and managing market risk.

In the 21st century, financial risk management has been guided by the **Basel Accords**. The Basel Committee on Banking Supervision, which is part of the Bank for International Settlements, has issued a series of regulatory frameworks known as Basel I, Basel II, and Basel III, which set out minimum capital requirements for banks to cover various types of risk, including market risk, credit risk, and operational risk. The Basel Accord I (1996) amendment prescribes a standard model, but allows banks to use an internal model if they meet certain criteria. The Basel Accord II (2001) requires banks to calculate a **minimum capital charge** (**regulatory capital**) based on VaR. The Basel Accord III (2010) was a response to the 2008 financial crisis and introduced stressed VaR, stricter capital requirements, leverage ratios, and liquidity standards. Almost all financial firms now use VaR to manage risk.

Your First Day on Job as a Risk Manager

On our first assignment, a trading desk asks us for a single number that summarizes one-week risk for a portfolio containing three stocks A, B, and C, together with eight options:

- three out-of-the-money puts on stock A with strikes $0.7S_0$, $0.6S_0$, and $0.5S_0$, and expiries of 3, 6, and 9 months;
- three out-of-the-money calls on stock B with strikes $1.2S_0$, $1.3S_0$, and $1.4S_0$, and expiries of 3, 6, and 9 months;
- one Asian option on stock C with 3-month expiry and one barrier option on stock C with 6-month expiry.

We are given correlation estimates

$$\rho_{AB} = 0.20, \qquad \rho_{AC} = 0.30, \qquad \text{and} \qquad \rho_{BC} = 0.10,$$

as well as historical return samples and current option market prices.

The one-week portfolio value may be written schematically as

$$\Pi_{\bar{t}} = S_A(\bar{t}) + S_B(\bar{t}) + S_C(\bar{t}) + V_1^A(S_A(\bar{t}), \bar{t}) + \cdots + V_3^C(S_C(\bar{t}), \bar{t})\,.$$

The key computational issues are dependence among underlyings and nonlinear dependence of derivative values on those underlyings.

A practical implementation plan has three stages:

1. Estimate a stochastic model for the joint *actual* distribution of $(S_A(\bar{t}), S_B(\bar{t}), S_C(\bar{t}))$, including drift and correlation effects.
2. Calibrate option pricing models for instruments on A, B, and C, and revalue each derivative at time $\bar{t}$ under simulated market scenarios, using PDE or Monte Carlo methods as appropriate.
3. From the induced distribution of portfolio values (or losses) at $\bar{t}$, extract a tail risk summary statistic, such as VaR or CVaR.

Risk Measures: VaR and CVaR

Classical mean–variance portfolio theory, as in **Markowitz's model**, measures risk by the covariance structure of returns. For stock-only portfolios this leads to optimization problems of the form

$$\min \boldsymbol{x}^\top \mathbf{Q}\boldsymbol{x}$$

subject to a target expected return $\boldsymbol{\mu}^\top \boldsymbol{x} = r_0$ and the budget constraint $\boldsymbol{x}^\top \mathbf{1} = 1$, where $\mathbf{1}$ is the vector of all ones. Here x_i is the portfolio weight of asset i, $\boldsymbol{\mu}$ is the vector of expected returns, and $\mathbf{Q}$ is the covariance matrix.

For derivative portfolios it is more natural to work directly with future portfolio value or loss.

Definition 9.1 If Π_t denotes the portfolio value at time t, then the **loss** over the horizon $[0, t]$ is

$$L = \Pi_0 - \Pi_t.$$

Equivalently, we may work with profit and loss, in which case $L = -\text{P\&L}$.

The absolute worst possible loss is often not a useful risk measure, since under many stochastic models the loss is unbounded above. This motivates the use of percentile-based tail measures instead.

Definition 9.2 For a confidence level $\beta \in (0,1)$, the **value-at-risk** at level β, denoted VaR_β, is the loss threshold satisfying

$$\mathbb{P}\big(L \geq \mathrm{VaR}_\beta\big) = 1 - \beta,$$

assuming the loss distribution is continuous. Thus VaR_β is the β-quantile of the loss distribution.

The main weakness of VaR is that it says where the dangerous tail begins, but it says nothing about how severe the losses are once we have entered that tail. CVaR remedies this by averaging the tail losses.

Definition 9.3 When the loss distribution is continuous, the **conditional value-at-risk** (or **expected shortfall**) at level β, denoted CVaR_β, is the expected loss conditional on being in the tail beyond the corresponding value-at-risk:

$$\mathrm{CVaR}_\beta = \mathbb{E}\left[L \mid L \geq \mathrm{VaR}_\beta\right].$$

The VaR threshold and the tail averaged by CVaR are shown in Figure 17.

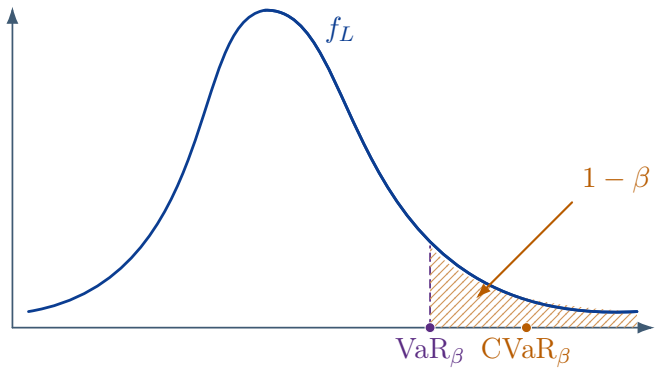

FIGURE 17

Example 9.4 Suppose independent simulated losses are ordered as

$$L_1 < L_2 < \cdots < L_M,$$

and let $\beta = 95\%$. If $M = 1000$, then

$$\mathrm{VaR}_{95\%} = L_{950}$$

and

$$\mathrm{CVaR}_{95\%} = \frac{1}{50}\big(L_{951} + \cdots + L_{1000}\big).$$

Thus VaR is read off as an empirical quantile, while CVaR is the empirical mean of the losses beyond that quantile.

Lecture 10 — Wednesday, February 11, 2015

Portfolio Loss for Derivatives

Consider a general portfolio consisting of holdings $x_1, \ldots, x_n$ in instruments $V_1, \ldots, V_n$. The instruments may depend on a single underlying asset. More generally, they may depend on a vector of risk factors

$$\boldsymbol{S} = [S_1, \ldots, S_k].$$

When multiple risk factors are present, correlations among them must be incorporated into any realistic risk calculation.

Let $\boldsymbol{x}$ denote the portfolio holdings, and let $L(\boldsymbol{x}, \boldsymbol{S})$ be the loss of the portfolio when the risk factors move to the state $\boldsymbol{S}$. For example, over a fixed horizon $\bar{t}$ we may write

$$L(\boldsymbol{x}, \boldsymbol{S}) = \sum_{j=1}^{n} \big(V_j(S_0, 0) - V_j(S_{\bar{t}}, \bar{t})\big) x_j.$$

If $p(\boldsymbol{S})$ denotes the density of the risk factor vector, then the cumulative distribution function of the loss is

$$\Psi(\boldsymbol{x}, \alpha) = \int_{L(\boldsymbol{x}, \boldsymbol{S}) \le \alpha} p(\boldsymbol{S})\, \mathrm{d}\boldsymbol{S}.$$

For a confidence level $\beta \in (0, 1)$, the value-at-risk of portfolio $\boldsymbol{x}$ is

$$\alpha_\beta(\boldsymbol{x}) = \min\{\alpha \in \mathbb{R} : \Psi(\boldsymbol{x}, \alpha) \ge \beta\}.$$

If the loss distribution is continuous, every solution of

$$\Psi(\boldsymbol{x}, \alpha) = \beta$$

is a β-quantile; uniqueness additionally requires that the distribution function be strictly increasing at that level. The minimum convention above selects the lower quantile when there is a flat interval. The corresponding conditional value-at-risk is the expected loss in the tail beyond the VaR threshold.

Proposition 10.1 For a confidence level β and an integrable continuously distributed loss, the conditional value-at-risk may be written as

$$\phi_\beta(\boldsymbol{x}) = \min_{\alpha \in \mathbb{R}} \left[\alpha + \frac{1}{1-\beta} \mathbb{E}\left[\left((L(\boldsymbol{x}, \boldsymbol{S}) - \alpha)^+ \right) \right] \right],$$

where

$$(L(\boldsymbol{x}, \boldsymbol{S}) - \alpha)^+ = \max(L(\boldsymbol{x}, \boldsymbol{S}) - \alpha, 0).$$

Proof. Fix $\boldsymbol{x}$ and write

$$L = L(\boldsymbol{x}, \boldsymbol{S}) \qquad \text{and} \qquad F(\alpha) := \alpha + \frac{1}{1-\beta} \mathbb{E}\big[(L-\alpha)^+\big].$$

The map $u \mapsto u^+$ is convex, so $\alpha \mapsto (L-\alpha)^+$ is convex for each realization of L, and therefore F is convex. Since the distribution of L is continuous, let α_β denote the VaR level, so that

$$\mathbb{P}(L \le \alpha_\beta) = \beta \qquad \text{and} \qquad \mathbb{P}(L \ge \alpha_\beta) = 1 - \beta.$$

For differentiable points of F,

$$F'(\alpha) = 1 + \frac{1}{1-\beta} \mathbb{E}\left[\frac{\partial}{\partial \alpha} (L-\alpha)^+ \right] = 1 - \frac{1}{1-\beta} \mathbb{P}(L > \alpha).$$

Hence $F'(\alpha_\beta) = 0$. Since F is convex, this first-order condition identifies a global minimizer, so

$$\min_{\alpha \in \mathbb{R}} F(\alpha) = F(\alpha_\beta).$$

Now evaluate at α_β:

$$\begin{aligned} F(\alpha_\beta) &= \alpha_\beta + \frac{1}{1-\beta} \mathbb{E}\big[(L-\alpha_\beta)^+\big] \\ &= \alpha_\beta + \frac{1}{1-\beta} \mathbb{E}\Big[(L-\alpha_\beta) \mathbb{1}_{\{L \ge \alpha_\beta\}}\Big] \end{aligned}$$

$$
\begin{aligned}
&= \alpha_\beta + \frac{1}{1-\beta}\left(\mathbb{E}[L\mathbb{1}_{\{L\geq\alpha_\beta\}}] - \alpha_\beta \mathbb{P}(L \geq \alpha_\beta)\right) \\
&= \frac{1}{1-\beta}\,\mathbb{E}[L\mathbb{1}_{\{L\geq\alpha_\beta\}}] \\
&= \mathbb{E}[L \mid L \geq \alpha_\beta] \\
&= \mathrm{CVaR}_\beta(\boldsymbol{x}).
\end{aligned}
$$

Therefore

$$
\phi_\beta(\boldsymbol{x}) = \min_{\alpha\in\mathbb{R}} \left[\alpha + \frac{1}{1-\beta}\,\mathbb{E}\big[(L(\boldsymbol{x},\boldsymbol{S})-\alpha)^+\big]\right].
$$

□

When the portfolio includes derivatives, the loss function $L(\boldsymbol{x}, \boldsymbol{S})$ is typically nonlinear and does not admit a simple analytic formula. This is why Monte Carlo methods become the default computational tool for VaR and CVaR in realistic books.

To estimate VaR or CVaR over a horizon such as $\bar{t} = 10$ days, we may simulate the underlying risk factors under an actual price dynamics model, including correlation effects, then revalue the derivatives in each scenario and record the corresponding portfolio losses.

Example 10.2 Suppose

$$
L_1 < L_2 < \cdots < L_M
$$

are sorted independent loss samples, and choose an index $i_\beta < M$ such that

$$
\frac{i_\beta}{M} \geq \beta > \frac{i_\beta - 1}{M}.
$$

Then we approximate

$$
\mathrm{VaR}_\beta \approx L_{i_\beta}
$$

and

$$
\mathrm{CVaR}_\beta \approx \frac{1}{1-\beta}\left[\left(\frac{i_\beta}{M} - \beta\right) L_{i_\beta} + \frac{1}{M}\sum_{i=i_\beta+1}^{M} L_i\right].
$$

This is the empirical formula of Rockafellar and Uryasev (2002).

Warning For large portfolios this procedure is computationally expensive because each simulated market scenario may require repricing many nonlinear instruments. Linear ap-

proximations are often used in practice to reduce cost, but those approximations must be treated cautiously.

The Black–Scholes model

$$\frac{\mathrm{d}S_t}{S_t} = \mu\,\mathrm{d}t + \sigma\,\mathrm{d}W_t$$

implies normally distributed log returns. Under that assumption, extremely large daily moves should be fantastically rare:

Range	Fraction out of range	Approx. frequency for daily event
$[\mu - 1\sigma, \mu + 1\sigma]$	1 in 3	Roughly twice a trading week
$[\mu - 2\sigma, \mu + 2\sigma]$	1 in 22	Roughly once a trading month
$[\mu - 3\sigma, \mu + 3\sigma]$	1 in 370	Roughly every 1.5 trading years
$[\mu - 4\sigma, \mu + 4\sigma]$	1 in 15,787	Roughly every 63 trading years
$[\mu - 5\sigma, \mu + 5\sigma]$	1 in 1,744,278	Roughly every 6,900 trading years
$[\mu - 6\sigma, \mu + 6\sigma]$	1 in 506,797,000	Roughly every 2 million trading years

The last column uses approximately 252 trading days per year.

However, the historical record contains many daily moves that are far into the tails of a fitted normal model. Under particular historical-volatility calibrations, for example, the 1987 crash has been reported as a 22.6σ event, while the 2008 financial crisis included several moves exceeding 9σ. The precise multiples depend on the volatility estimate, but their frequency makes the central point robust: a normal model substantially understates the tails of asset returns.

Definition 10.3 Let $V^{\mathrm{BS}}(\sigma; S_0, 0)$ denote the Black–Scholes price of a European option as a function of volatility σ, with all other parameters fixed. If the market price V^{mkt} of that option is observed, then an **implied volatility** is a value $\hat{\sigma}$ satisfying

$$V^{\mathrm{BS}}(\hat{\sigma}; S_0, 0) = V^{\mathrm{mkt}}.$$

Implied volatility is not directly observable, but it packages the market price into a volatility number that is easier to compare across strikes and maturities. Traders therefore often quote options by implied volatility rather than by raw price.

For a European call, the root-finding problem can be written as

$$f(\sigma) = S_0\Phi(d_1) - Ke^{-rT}\Phi(d_2) - V^{\mathrm{mkt}} = 0,$$

where

$$d_1 = \frac{\log(S_0/K) + \left(r + \sigma^2/2\right) T}{\sigma\sqrt{T}},$$
$$d_2 = \frac{\log(S_0/K) + \left(r - \sigma^2/2\right) T}{\sigma\sqrt{T}},$$

and Φ is the standard normal cumulative distribution function.

Recall that the vega of an option is the sensitivity of the option's price to volatility. In the Black–Scholes model,

$$\text{vega} = \frac{\partial V^{\text{BS}}}{\partial \sigma}.$$

For both calls and puts under Black–Scholes,

$$\text{vega} = S_0 \varphi(d_1)\sqrt{T},$$

where φ is the standard normal density. In particular, vega is strictly positive.

The implied volatility tells a trader how expensive the option is relative to the Black–Scholes benchmark, with r, q, K and T taken into account. Traders often quote implied volatilities rather than raw prices because the former are more comparable across strikes and maturities. The market option price can be used to determine information about the underlying, such as volatility.

Proposition 10.4 (Manaster and Koehler (1982)) For a European call, the Black–Scholes price is strictly increasing in $\sigma \in (0, \infty)$. Moreover,

$$\lim_{\sigma \to 0^+} V^{\text{BS}}(\sigma; S_0, 0) = \max(0, S_0 - Ke^{-rT}),$$
$$\lim_{\sigma \to \infty} V^{\text{BS}}(\sigma; S_0, 0) = S_0.$$

Therefore a positive implied volatility exists if and only if the market price satisfies

$$\max(0, S_0 - Ke^{-rT}) < V^{\text{mkt}} < S_0.$$

Since implied volatility is defined as the root of a scalar nonlinear equation, Newton's method is a natural choice:

$$\sigma_{k+1} = \sigma_k - \frac{f(\sigma_k)}{f'(\sigma_k)}.$$

Here $f'(\sigma_k)$ is the vega, so each Newton step requires repeated evaluation of the Black–Scholes formula and its volatility derivative.

In general, an iterative method generates a sequence of approximations $\{x_k\}_k$ which converges to a solution x^* of a nonlinear equation $f(x) = 0$. The method is said to have **local convergence** if there exists a neighborhood around x^* such that for any starting point in that neighborhood, the sequence converges to x^*. The method has **global convergence** if it converges to x^* for any starting point in the domain of interest. The method is **quadratically convergent** if the error at step $k + 1$ is approximately the square of the error at step k; that is,

$$|x_{k+1} - x^*| = O(|x_k - x^*|^2).$$

For the implied volatility problem, Manaster and Koehler (1982) suggest using Newton's method with the starting point

$$\sigma_0 = \sqrt{\frac{2}{T}\left|\log\left(\frac{S_0}{K}\right) + rT\right|}.$$

Under the conditions given in their analysis, and provided $S_0 \neq Ke^{-rT}$ so that this starting value is positive, the iterates converge monotonically and quadratically to the unique implied volatility. In the at-the-money-forward case $S_0 = Ke^{-rT}$, the displayed expression is zero and a separate positive starting value is required.

Running the Newton update requires evaluating both $V^{\mathrm{BS}}(\sigma)$ and $\mathrm{vega}(\sigma)$ repeatedly. MATLAB's `blsimpv` computes implied volatility using the Jäckel 2016 rational method by default and also offers a search method.

The Volatility Smile

The Black–Scholes model assumes a constant volatility. If that were correct, then all options on the same underlying would produce the same implied volatility regardless of strike and maturity. In practice, empirical evidence shows that the graph of implied volatility against strike or maturity is not flat, but rather curved. This dependence of implied volatility on contract characteristics is called the **volatility smile**. For equity options, the smile often tilts downward as strike increases. This asymmetric pattern is often called a **volatility sneer**.

Index options provide a standard empirical example: the market-settled option is written on an index level, but the extracted implied volatilities still vary significantly across strikes.

Example 10.5 For S&P 500 index options, observed implied volatilities are higher for lower strikes and lower for higher strikes, which is consistent with the idea that equity markets assign extra weight to severe downside moves. Consider, for example, a call with expiry T

and strike price K. With a contract multiplier of 100, its cash payoff is $100\max(I_T - K, 0)$, where

$$I_t = \sum_{i=1}^{500} w^{(i)} S_t^{(i)}$$

represents the S&P 500 index level at time t, with $w^{(i)}$ denoting the weight of stock i in the index and $S_t^{(i)}$ denoting the price of stock i at time t. If the option is out-of-the-money, it expires worthless. The implied volatility extracted from market prices of such options shows the downward-sloping pattern in Figure 18, indicating that investors are willing to pay more for protection against large drops in the index than for potential gains.

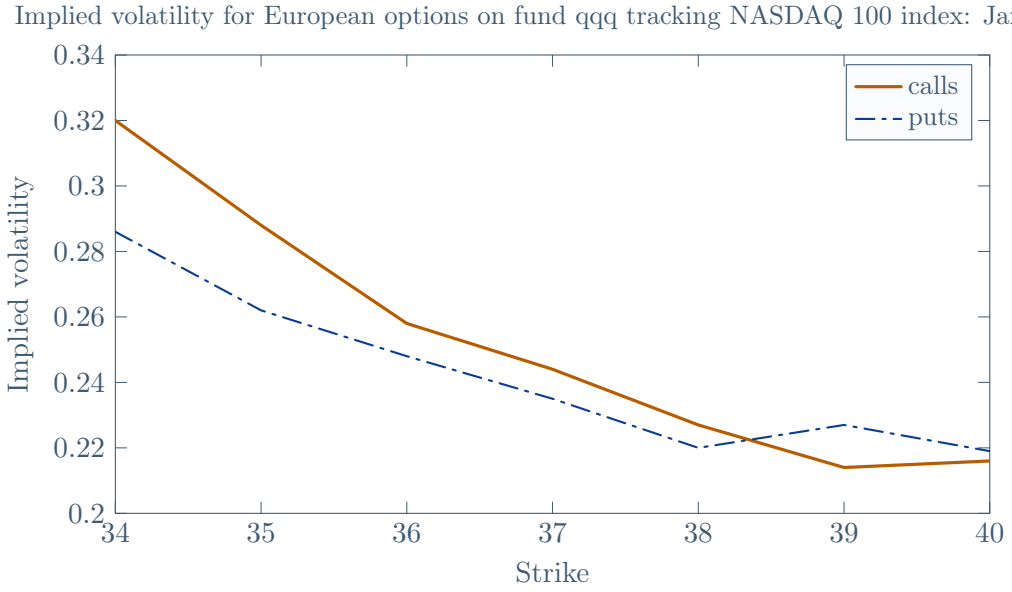

FIGURE 18

Lecture 11 — Friday, February 13, 2015

For S&P 500 index options, observed implied volatilities are higher for lower strikes and lower for higher strikes, which is consistent with the idea that equity markets assign extra weight to severe downside moves.

Several explanations for the volatility smile are commonly given. First, empirical return distribu-

tions tend to have fatter left tails than the normal model predicts. Second, volatility itself appears to vary randomly over time. Third, market prices sometimes jump discontinuously. Modeling the volatility smile remains an active research area precisely because several mechanisms can generate similar option price patterns.

In the **local volatility framework** (**LVF**) of Dupire (1994), the asset price satisfies

$$\frac{\mathrm{d}S_t}{S_t} = \mu\,\mathrm{d}t + \sigma(S_t, t)\,\mathrm{d}W_t,$$

where the volatility $\sigma(S, t)$ is a deterministic function of state and time.

Because the volatility function is deterministic, we can still eliminate option risk through replication, and no-arbitrage arguments lead to a generalized Black–Scholes PDE. In this sense the local volatility framework is the simplest extension of Black–Scholes that can fit a smile while allowing option pricing through replication.

The **stochastic volatility model** of Heston (1993) introduces a separate variance process:

$$\begin{aligned}\frac{\mathrm{d}S}{S} &= r\,\mathrm{d}t + \sqrt{v}\,\mathrm{d}Z^{(1)},\\ \mathrm{d}v &= -\lambda(v - \bar{v})\,\mathrm{d}t + \eta\sqrt{v}\,\mathrm{d}Z^{(2)},\end{aligned}$$

with

$$\mathbb{E}\left[\mathrm{d}Z^{(1)}\,\mathrm{d}Z^{(2)}\right] = \rho\,\mathrm{d}t.$$

where $\lambda > 0$, $\bar{v} > 0$, $\eta > 0$, and $\rho \in [-1, 1]$. Here λ controls the speed of reversion of the variance toward the long-run level $\bar{v}$. The parameter η controls the volatility of volatility, and ρ controls the correlation between the asset price and its variance. The square-root variance process remains nonnegative; when $v_0 > 0$, the stronger Feller condition $2\lambda\bar{v} \geq \eta^2$ keeps zero inaccessible. The Heston model can produce a wide variety of implied volatility patterns, including skew and smile effects, by adjusting these parameters.

However, stochastic volatility markets are incomplete. The added volatility risk cannot in general be hedged away using only the underlying and the money market account, and parameter estimation becomes substantially harder.

The **jump diffusion model** of Merton (1976) augments Black–Scholes by allowing sudden jumps:

$$\frac{\mathrm{d}S_t}{S_t} = (r - \kappa\lambda)\,\mathrm{d}t + \sigma\,\mathrm{d}W_t + (J - 1)\,\mathrm{d}\pi_t.$$

Here π_t is a Poisson counting process with jump intensity λ (for example, $\lambda = 0.1$ implies one jump every ten years, on average) so that

$$\mathrm{d}\pi = \begin{cases} 1, & \text{with probability } \lambda\,\mathrm{d}t, \\ 0, & \text{with probability } 1 - \lambda\,\mathrm{d}t, \end{cases}$$

each jump uses an independent copy of a positive multiplier J, independent of the Brownian motion and Poisson process, with $\log J$ normally distributed with constant mean μ_J and constant variance σ_J^2, and

$$\kappa = \mathbb{E}\left[J - 1\right].$$

The compensating drift term $-\kappa\lambda$ ensures the risk-neutral drift remains correct on average. European calls and puts can still be priced analytically in this model (see, e.g., Merton (1976) or Lewis (2002)), but the market remains incomplete because jump risk cannot be hedged away using finitely many standard traded instruments.

Question 11.1 More complex hybrid models also arise, such as jump-plus-stochastic-volatility and jump-plus-local-volatility models. This leads naturally to model risk questions: if two models both calibrate well to market prices, which should be used, and how much do pricing and hedging results depend on that choice?

4. SDE and PDE Numerical Methods

SDE Simulation: Euler–Maruyama and Milstein

For the Black–Scholes model we can write the exact solution explicitly:

$$S_t = S_0 \exp\left(\left(r - \frac{1}{2}\sigma^2\right)t + \sigma Z_t\right).$$

Thus each Brownian path immediately determines one asset price path. More complicated models, such as the local volatility framework model

$$\frac{\mathrm{d}S_t}{S_t} = r\,\mathrm{d}t + \sigma(S_t, t)\,\mathrm{d}Z_t,$$

do not admit such a closed-form solution, so we must approximate sample paths numerically. How can we generate approximate sample paths of S_t in this case?

Recall that the general Ito process has the form

$$\mathrm{d}X_t = a(X_t, t)\,\mathrm{d}t + b(X_t, t)\,\mathrm{d}Z_t.$$

Equivalently,

$$X_t = X_0 + \int_0^t a(X_\tau, \tau)\,\mathrm{d}\tau + \int_0^t b(X_\tau, \tau)\,\mathrm{d}Z_\tau.$$

To implement this in practice, we can use the **Euler–Maruyama method**, which is a straightforward extension of Euler's method for ordinary differential equations to the stochastic setting.

Divide the integration interval into steps of size δt, and write $\Delta Z_t = Z_{t+\delta t} - Z_t$. Approximating the integrands by their left endpoint values on each subinterval gives

$$\int_t^{t+\delta t} a(X_\tau, \tau)\,\mathrm{d}\tau \approx a(X_t, t)\delta t \qquad \text{and} \qquad \int_t^{t+\delta t} b(X_\tau, \tau)\,\mathrm{d}Z_\tau \approx b(X_t, t)\Delta Z_t.$$

Then

$$\tilde{X}_{t+\delta t} = \tilde{X}_t + a(\tilde{X}_t, t)\delta t + b(\tilde{X}_t, t)\Delta Z_t.$$

Since $\Delta Z_t \sim \mathcal{N}(0, \delta t)$, we may write

$$\Delta Z_t = \sqrt{\delta t}\,\phi_t, \quad \phi_t \sim \mathcal{N}(0, 1),$$

and hence

$$\tilde{X}_{t+\delta t} = \tilde{X}_t + a(\tilde{X}_t, t)\delta t + b(\tilde{X}_t, t)\sqrt{\delta t}\,\phi_t.$$

To implement this in practice, we choose grid points $t_j = j\delta t$ with $\delta t = T/N$, start from $\tilde{X}_0 = X_0$, and for $j = 0, 1, \ldots, N-1$ iterate

$$\tilde{X}_{j+1} = \tilde{X}_j + a(\tilde{X}_j, t_j)\delta t + b(\tilde{X}_j, t_j)\sqrt{\delta t}\,\phi_j, \quad \phi_j \sim \mathcal{N}(0, 1).$$

Applying this update componentwise over many independent normal draws produces many independent approximate sample paths simultaneously.

The Milstein Method is a higher-order numerical scheme. Using Ito's lemma on $b(X_t, t)$ but ignoring terms of order higher than $O(\sqrt{\delta t})$ leads to

$$b(X_{t+\delta t}, t) \approx b(X_t, t) + \frac{\partial b}{\partial X}\Delta Z_t.$$

After some further stochastic integration (see Seydel (2006) for details), we obtain the Milstein

update formula:

$$\tilde{X}_{t+\delta t} = \tilde{X}_t + a(\tilde{X}_t, t)\delta t + b(\tilde{X}_t, t)\Delta Z_t + \frac{1}{2} b(\tilde{X}_t, t)\frac{\partial b}{\partial X}\big((\Delta Z_t)^2 - \delta t\big).$$

Lecture 12 — Monday, February 23, 2015

For a scalar SDE

$$\mathrm{d}X_t = a(X_t, t)\,\mathrm{d}t + b(X_t, t)\,\mathrm{d}Z_t,$$

the Euler–Maruyama method replaces the drift and diffusion integrands on each time interval $[t, t+\delta t]$ by their left endpoint values. This yields the update

$$\begin{aligned}\tilde{X}_{t+\delta t} &= \tilde{X}_t + a(\tilde{X}_t, t)\delta t + b(\tilde{X}_t, t)\Delta Z_t,\\ \Delta Z_t &= \sqrt{\delta t}\,\phi_t, \quad \phi_t \sim \mathcal{N}(0,1).\end{aligned}$$

The Milstein method adds the correction

$$\frac{1}{2} b(\tilde{X}_t, t)\frac{\partial b}{\partial X}\big((\Delta Z_t)^2 - \delta t\big),$$

and therefore reduces to Euler–Maruyama when the diffusion coefficient b does not depend on the state variable X.

The multidimensional setting is more delicate. For a coupled system such as

$$\begin{aligned}\mathrm{d}X_t &= a(X_t, Y_t, t)\,\mathrm{d}t + b(X_t, Y_t, t)\,\mathrm{d}Z_t^{(1)},\\ \mathrm{d}Y_t &= e(X_t, Y_t, t)\,\mathrm{d}t + f(X_t, Y_t, t)\,\mathrm{d}Z_t^{(2)},\end{aligned}$$

the Euler–Maruyama method generalizes directly, but higher-order Milstein-type formulas become substantially more complicated.

Definition 12.1 A numerical scheme for an SDE is said to have **strong order** γ if, for sufficiently small δt and a fixed terminal time T,

$$\mathbb{E}\left[|X_T - \tilde{X}_T|\right] \leq C(\delta t)^\gamma$$

for some constant C independent of δt. The scheme is said to have **strong convergence** if

$\gamma > 0$.

Definition 12.2 A numerical scheme is said to have **weak order** β if, for sufficiently small δt and every sufficiently smooth test function g of polynomial growth,

$$\left|\mathbb{E}\left[g(X_T)\right] - \mathbb{E}\left[g(\tilde{X}_T)\right]\right| \leq C(\delta t)^\beta$$

for a constant C that may depend on g and T but is independent of δt. The scheme is said to have **weak convergence** if $\beta > 0$.

Proposition 12.3 Under suitable assumptions on the coefficients of the SDE, the Euler–Maruyama method has strong order $\gamma = 1/2$ and weak order $\beta = 1$, while the Milstein method has both strong and weak order equal to 1.

For Monte Carlo pricing this matters because the weak error controls the bias in option values, whereas strong convergence is more closely tied to pathwise approximation and therefore to simulation of functionals that depend sensitively on the whole sample path.

If the Euler–Maruyama discretization scheme is used inside Monte Carlo pricing, then the weak time discretization error is $O(\delta t)$, while the sampling error remains $O(M^{-1/2})$. Hence the combined error behaves like

$$O\left(\max\left(\delta t, \frac{1}{\sqrt{M}}\right)\right).$$

Balancing the two terms again suggests choosing

$$M = O(\delta t^{-2}),$$

which yields an overall error of order $O(\delta t)$.

From SDEs to PDEs

Deterministic pricing equations arise naturally from the same stochastic model. If the underlying follows

$$\mathrm{d}S_t = (\mu - q)S_t\,\mathrm{d}t + \sigma(S_t, t)S_t\,\mathrm{d}Z_t,$$

then any European option value $V(S, t)$ satisfies the PDE

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r - q)S\frac{\partial V}{\partial S} - rV = 0, \quad 0 < S < \infty, \quad 0 \leq t < T.$$

A PDE by itself does not determine a unique solution. We must supplement it with terminal or initial data and with suitable boundary conditions. For an option pricing problem, the terminal condition is typically the payoff at expiry.

When σ, r, and q are constant, using the transformations

$$S = Ke^x, \qquad t = T - \frac{2\tau}{\sigma^2}, \qquad \hat{\delta} = \frac{2r}{\sigma^2}, \qquad \text{and} \qquad \delta = \frac{2(r-q)}{\sigma^2}$$

and

$$V(S,t) = V\left(Ke^x, T - \frac{2\tau}{\sigma^2}\right) =: v(x,\tau)$$

and

$$v(x,\tau) = Ke^{-(\delta-1)x/2 - ((\delta-1)^2/4 + \hat{\delta})\tau} u(x,\tau),$$

the Black–Scholes PDE can be transformed into the heat equation

$$\frac{\partial u}{\partial \tau} = \frac{\partial^2 u}{\partial x^2}.$$

Solutions to the heat equation become instantly smooth for positive time, even if the initial profile is not smooth.

Definition 12.4 For a second-order PDE of the form

$$Au_{xx} + Bu_{xy} + Cu_{yy} + F(u_x, u_y, u, x, y) = 0,$$

the equation is called **parabolic** if $B^2 - 4AC = 0$, **hyperbolic** if $B^2 - 4AC > 0$, and **elliptic** if $B^2 - 4AC < 0$.

The Black–Scholes equation is parabolic. The heat equation is also parabolic. Parabolic equations are often used to model diffusion processes, and they have well-developed theories for existence, uniqueness, and regularity of solutions; thus, the Black–Scholes PDE is well-posed and has a unique solution under suitable conditions.

However, local volatility produces a variable-coefficient linear PDE, American exercise produces a complementarity problem, and jump diffusion produces an integro-differential equation. These extensions are more complicated to analyze and solve, so we usually obtain approximate solutions by numerical discretization.

Three major computational approaches are relevant here. **Lattice methods** are simple and explicit, but become highly complex in certain settings. Monte Carlo methods are natural for

high-dimensional problems, but they are inefficient in low dimension and handle American exercise poorly.

Numerical PDE methods are often the method of choice in low dimension because they handle complex exercise features well and produce values and hedging parameters for a whole range of asset prices simultaneously, which is useful for simulating hedging strategies. A general solver can price many contracts after we change the terminal and boundary conditions.

To prepare for finite-difference discretization, switch to time-to-expiry

$$\tau = T - t,$$

so the Black–Scholes PDE becomes

$$\frac{\partial V}{\partial \tau} = \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV.$$

The numerical problem is then to solve this PDE on a truncated computational domain $[0, S_{\max}] \times [0, T]$ (which approximates $[0, \infty) \times [0, T]$) with initial condition $V(S, 0) = \text{payoff}(S)$ and suitable boundary conditions at $S = 0$ and $S = S_{\max}$. The truncation error caused by the artificial upper boundary can be made small by moving $S_{\max}$ far enough from the asset-price region of interest and imposing an asymptotically appropriate boundary condition.

We discretize the continuous region by a finite set of grids $\{(S_i, \tau^n)\}$ for $n = 0, 1, \ldots, N$ and $i = 0, 1, \ldots, M$. We then discretize the PDE itself at each grid point to obtain a system of algebraic equations that relates values at n with values at $n + 1$. If the scheme is consistent and stable and the continuous boundary-value problem is well posed, then the discrete solution converges to the continuous solution as the mesh is refined.

Remark 12.5 Finite difference methods replace derivatives by finite differences. In contrast, **finite element methods** approximate solutions by polynomial interpolants. A standard finite difference analysis requires smooth payoffs, whereas finite element methods can handle discontinuous payoffs more easily. Finite element analysis shows that finite difference equations are better than a simple Taylor expansion would suggest.

Finite volume methods are another alternative, which are based on integrating the PDE over small control volumes and applying the divergence theorem. They are particularly useful in engineering problems, but they are less common in financial applications compared to finite difference and finite element methods.

In many cases, finite element methods, finite difference methods, and finite volume methods lead to essentially the same algebraic equations. The difference is a philosophical issue in this case: all roads lead to Rome. Engineers prefer finite volume methods, mathematicians prefer finite element methods, and portfolio managers prefer finite difference methods. The choice of method is less important than the choice of discretization and the analysis of stability and convergence.

Lecture 13 — Wednesday, February 25, 2015

Finite-Difference Methods

The finite-difference approach can now be developed in detail. Working with time-to-expiry $\tau = T - t$, the Black–Scholes PDE is written as

$$\frac{\partial V}{\partial \tau} = \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV.$$

The numerical domain is truncated to $0 \le S \le S_{\max}$ and $0 \le \tau \le T$, and we introduce grid points

$$\{S_0, S_1, \ldots, S_m\} \qquad \text{and} \qquad \{\tau^0, \tau^1, \ldots, \tau^N\}.$$

The discrete approximation to $V(S_i, \tau^n)$ is denoted by V_i^n.

The local spatial mesh widths are

$$\Delta S_{i+1/2} = S_{i+1} - S_i \qquad \text{and} \qquad \Delta S_{i-1/2} = S_i - S_{i-1}.$$

For time stepping, we use

$$\Delta\tau = \tau^{n+1} - \tau^n.$$

We assume that f is smooth, and that $x_{i-1} < x_i < x_{i+1}$ are three consecutive grid points. We write $f_i := f(x_i)$ for the function values at those points, and assume that f_{i-1}, f_i, f_{i+1} are all known. Let

$$\Delta x_{i+1/2} = x_{i+1} - x_i \qquad \text{and} \qquad \Delta x_{i-1/2} = x_i - x_{i-1},$$

and let f_x denote the derivative of f with respect to x. We thus want to approximate $(f_x)_i$ and $(f_{xx})_i$ by finite differences.

A Taylor expansion gives

$$f_{i+1} = f(x_i + \Delta x_{i+1/2}) = f_i + \Delta x_{i+1/2} f_x(x_i) + \frac{(\Delta x_{i+1/2})^2}{2} f_{xx}(x_i) + O((\Delta x_{i+1/2})^3),$$

which leads to the **forward difference**

$$(f_x)_i = \frac{f_{i+1} - f_i}{\Delta x_{i+1/2}} + O(\Delta x_{i+1/2})$$

and the **backward difference**

$$(f_x)_i = \frac{f_i - f_{i-1}}{\Delta x_{i-1/2}} + O(\Delta x_{i-1/2}).$$

If the grid is uniform, so that $\Delta x_{i+1/2} = \Delta x_{i-1/2} = \Delta x$, then

$$f_{i+1} = f_i + \Delta x (f_x)_i + \frac{\Delta x^2}{2} (f_{xx})_i + O(\Delta x^3) \tag{13.1}$$

and

$$f_{i-1} = f_i - \Delta x (f_x)_i + \frac{\Delta x^2}{2} (f_{xx})_i + O(\Delta x^3). \tag{13.2}$$

Combining (13.1) and (13.2) gives the **second-order central difference**

$$(f_x)_i = \frac{f_{i+1} - f_{i-1}}{2\Delta x} + O(\Delta x^2).$$

Thus the central approximation is second-order accurate in space.

More generally, on a nonuniform grid with $\Delta x_{i+1/2} \neq \Delta x_{i-1/2}$, we can still combine the Taylor expansions to obtain a central approximation:

$$(f_x)_i = \frac{f_{i+1} - f_{i-1}}{\Delta x_{i+1/2} + \Delta x_{i-1/2}} - \frac{(f_{xx})_i}{2} (\Delta x_{i+1/2} - \Delta x_{i-1/2}) + O\left(\frac{\Delta x_{i+1/2}^3 + \Delta x_{i-1/2}^3}{\Delta x_{i+1/2} + \Delta x_{i-1/2}} \right),$$

where

$$(f_{xx})_i = \frac{\frac{f_{i+1} - f_i}{\Delta x_{i+1/2}} + \frac{f_{i-1} - f_i}{\Delta x_{i-1/2}}}{\frac{1}{2}(\Delta x_{i+1/2} + \Delta x_{i-1/2})} + O\left(\left| \Delta x_{i+1/2} - \Delta x_{i-1/2} \right| \right) + O\left(\frac{\Delta x_{i+1/2}^3 + \Delta x_{i-1/2}^3}{\Delta x_{i+1/2} + \Delta x_{i-1/2}} \right).$$

On a uniform grid with $\Delta x_{i+1/2} = \Delta x_{i-1/2} = \Delta x$, this reduces to

$$f''(x_i) = \frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2} + O(\Delta x^2).$$

Explicit Discretization of the Black–Scholes PDE

For reference, we rewrite the Black–Scholes equation in time-to-expiry form:

$$\frac{\partial V}{\partial \tau} = \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV.$$

This is a **drift–diffusion equation** with a **decay term**. The diffusion term is $\frac{\sigma^2}{2} S^2 V_{SS}$, the drift term is rSV_S, and the decay term is $-rV$.

Let $\{S_0, S_1, \ldots, S_m\}$ be spatial nodes with $0 \le S_i \le S_{\max}$, and let $\{\tau^0, \tau^1, \ldots, \tau^N\}$ be time levels with $0 \le \tau^n \le T$. We write

$$V(S_i, \tau^n) = V_i^n.$$

Note that here, S_0 denotes the first spatial node of the grid. When needed, we denote the initial asset price at time $t = 0$ by S^0 to avoid ambiguity. The local mesh widths are

$$\Delta S_{i+1/2} = S_{i+1} - S_i \qquad \text{and} \qquad \Delta S_{i-1/2} = S_i - S_{i-1}.$$

At an interior node, we evaluate all right-hand side terms at time level n, which leads to an explicit scheme:

$$\left(\frac{\partial V}{\partial \tau}\right)_i^n = \left(\frac{\sigma^2}{2} S^2 V_{SS}\right)_i^n + (rSV_S)_i^n - (rV)_i^n . \tag{13.3}$$

The time derivative is approximated by

$$\left(\frac{\partial V}{\partial \tau}\right)_i^n \approx \frac{V_i^{n+1} - V_i^n}{\Delta \tau}, \quad \Delta \tau = \tau^{n+1} - \tau^n. \tag{13.4}$$

For the diffusion term we use the second derivative on a nonuniform grid:

$$\left(\frac{\sigma^2}{2} S^2 V_{SS}\right)_i^n \approx \frac{\sigma^2 S_i^2}{2} \left(\frac{\dfrac{V_{i+1}^n - V_i^n}{\Delta S_{i+1/2}} - \dfrac{V_i^n - V_{i-1}^n}{\Delta S_{i-1/2}}}{\dfrac{\Delta S_{i+1/2} + \Delta S_{i-1/2}}{2}} \right). \tag{13.5}$$

The discounting term is simply

$$(rV)_i^n = rV_i^n. \tag{13.6}$$

For the drift term, there are three standard choices. Central weighting gives

$$(rSV_S)_i^n \approx rS_i \frac{V_{i+1}^n - V_{i-1}^n}{\Delta S_{i+1/2} + \Delta S_{i-1/2}}. \tag{13.7}$$

Forward differencing gives

$$(rSV_S)_i^n \approx rS_i \frac{V_{i+1}^n - V_i^n}{\Delta S_{i+1/2}}, \tag{13.8}$$

and backward differencing gives

$$(rSV_S)_i^n \approx rS_i \frac{V_i^n - V_{i-1}^n}{\Delta S_{i-1/2}}. \tag{13.9}$$

If the grid is uniform, central differencing is second-order accurate in ΔS, while forward and backward differencing are first-order accurate. Hence we prefer central differencing whenever its qualitative properties are acceptable.

Substituting (13.4), (13.5), and (13.6) together with one of (13.7), (13.8), and (13.9) yields

$$V_i^{n+1} = V_i^n\big(1 - (\alpha_i + \beta_i + r)\Delta\tau\big) + \alpha_i \Delta\tau\, V_{i-1}^n + \beta_i \Delta\tau\, V_{i+1}^n. \tag{13.10}$$

The formula is always the same, and only the definitions of α_i and β_i change.

For central weighting,

$$\begin{aligned}
\alpha_i^{\text{central}} &= \frac{\sigma^2 S_i^2}{(S_i - S_{i-1})(S_{i+1} - S_{i-1})} - \frac{rS_i}{S_{i+1} - S_{i-1}}, \\
\beta_i^{\text{central}} &= \frac{\sigma^2 S_i^2}{(S_{i+1} - S_i)(S_{i+1} - S_{i-1})} + \frac{rS_i}{S_{i+1} - S_{i-1}}.
\end{aligned}$$

For forward differencing,

$$\begin{aligned}
\alpha_i^{\text{forward}} &= \frac{\sigma^2 S_i^2}{(S_i - S_{i-1})(S_{i+1} - S_{i-1})}, \\
\beta_i^{\text{forward}} &= \frac{\sigma^2 S_i^2}{(S_{i+1} - S_i)(S_{i+1} - S_{i-1})} + \frac{rS_i}{S_{i+1} - S_i}.
\end{aligned}$$

For backward differencing,

$$\alpha_i^{\text{backward}} = \frac{\sigma^2 S_i^2}{(S_i - S_{i-1})(S_{i+1} - S_{i-1})} - \frac{rS_i}{S_i - S_{i-1}},$$
$$\beta_i^{\text{backward}} = \frac{\sigma^2 S_i^2}{(S_{i+1} - S_i)(S_{i+1} - S_{i-1})}.$$

At the upper boundary $S = S_m$, the boundary condition is

$$V_m^n = \begin{cases} S_m - Ke^{-r\tau^n}, & \text{for a call,} \\ 0, & \text{for a put.} \end{cases}$$

At the lower boundary $S = 0$, the Black–Scholes equation reduces to

$$V_\tau = -rV,$$

so the explicit time step gives

$$V_0^{n+1} = V_0^n(1 - r\Delta\tau).$$

The explicit method can be summarized as follows.

Algorithm 1: Explicit finite-difference method for Black–Scholes

for $i = 0, 1, \ldots, m$ **do**
  $V_i^0 \leftarrow \text{Payoff}(S_i)$
for $n = 0, 1, \ldots, N-1$ **do**
  $V_0^{n+1} \leftarrow V_0^n(1 - r\Delta\tau)$
  $V_m^{n+1} \leftarrow S_m - Ke^{-r\tau^{n+1}}$ for a call, or 0 for a put
  for $i = 1, 2, \ldots, m-1$ **do**
    $V_i^{n+1} \leftarrow V_i^n\big(1 - (\alpha_i + \beta_i + r)\Delta\tau\big) + \alpha_i\Delta\tau\, V_{i-1}^n + \beta_i\Delta\tau\, V_{i+1}^n$

Lecture 14 — Friday, February 27, 2015

Positive Coefficients and Upstream Weighting

We begin with the supershare pure drift test case $\sigma = 0$. If forward differencing is used for the drift term, the explicit update becomes

$$V_i^{n+1} = V_i^n\big(1 - (1+i)r\Delta\tau\big) + ir\Delta\tau\, V_{i+1}^n. \tag{14.1}$$

With the same discrete initial condition as before,

$$V_i^0 = \begin{cases} 0, & i \neq p, \\ \varepsilon, & i = p, \end{cases} \tag{14.2}$$

the first time step gives

$$V_p^1 = \varepsilon\big(1 - (1+p)r\Delta\tau\big), \qquad V_{p-1}^1 = \varepsilon(p-1)r\Delta\tau, \qquad \text{and} \qquad V_{p+1}^1 = 0.$$

Hence, if

$$(1+i)r\Delta\tau \leq 1 \quad \text{for all relevant } i,$$

then all coefficients in (14.1) are nonnegative, and the update preserves nonnegativity at the next time level.

By contrast, central differencing in this pure drift case gives

$$V_i^{n+1} = V_i^n(1 - r\Delta\tau) - \frac{ir\Delta\tau}{2}V_{i-1}^n + \frac{ir\Delta\tau}{2}V_{i+1}^n,$$

which contains a negative coefficient at V_{i-1}^n. This illustrates why a formally lower-order discretization may be preferable when it preserves important qualitative properties.

Definition 14.1 For the explicit update

$$V_i^{n+1} = V_i^n\big(1 - (\alpha_i + \beta_i + r)\Delta\tau\big) + \alpha_i\Delta\tau\, V_{i-1}^n + \beta_i\Delta\tau\, V_{i+1}^n,$$

we call the scheme a **positive coefficient discretization** if all coefficients multiplying values at time level n are nonnegative.

For this explicit form, a necessary condition is

$$\alpha_i \geq 0 \qquad \text{and} \qquad \beta_i \geq 0.$$

These inequalities are required later for stability and for suppressing spurious oscillations.

Recall the drift-dependent coefficient formulas. Forward differencing gives

$$\alpha_i^{\text{forward}} = \frac{\sigma^2 S_i^2}{(S_i - S_{i-1})(S_{i+1} - S_{i-1})},$$
$$\beta_i^{\text{forward}} = \frac{\sigma^2 S_i^2}{(S_{i+1} - S_i)(S_{i+1} - S_{i-1})} + \frac{rS_i}{S_{i+1} - S_i},$$

and backward differencing gives

$$\alpha_i^{\text{backward}} = \frac{\sigma^2 S_i^2}{(S_i - S_{i-1})(S_{i+1} - S_{i-1})} - \frac{rS_i}{S_i - S_{i-1}},$$
$$\beta_i^{\text{backward}} = \frac{\sigma^2 S_i^2}{(S_{i+1} - S_i)(S_{i+1} - S_{i-1})}.$$

In the Black–Scholes case with $r > 0$, forward differencing yields nonnegative coefficients at each node.

Definition 14.2 The **upstream weighting** rule is

$$(\alpha_i^{\text{up}}, \beta_i^{\text{up}}) = \begin{cases} (\alpha_i^{\text{forward}}, \beta_i^{\text{forward}}), & \text{if } \alpha_i^{\text{forward}} \geq 0 \text{ and } \beta_i^{\text{forward}} \geq 0, \\ (\alpha_i^{\text{backward}}, \beta_i^{\text{backward}}), & \text{otherwise.} \end{cases}$$

This guarantees $\alpha_i^{\text{up}} \geq 0$ and $\beta_i^{\text{up}} \geq 0$.

Remark 14.3 The terminology *upstream* or *upwind* comes from advection modeling in engineering. Suppose we are simulating the air quality near a pig farm. We can only smell the pigs if we are downwind of a farm. Upstream weighting ensures that we can only smell discrete pigs if we are discretely downwind of a farm. In other words, information should be taken from the numerically upwind direction.

Since central weighting is more accurate, the practical choice is nodewise: use central coefficients whenever they satisfy positivity, and otherwise use upstream coefficients.

For the simple Black–Scholes model with $r > 0$, this upstream choice is usually the forward choice.

Algorithm 2: Preprocessing for central-versus-upstream weighting

for $i = 1, 2, \ldots, m-1$ **do**
  if $\alpha_i^{\text{central}} \geq 0$ *and* $\beta_i^{\text{central}} \geq 0$ **then**
    $\alpha_i \leftarrow \alpha_i^{\text{central}}$
    $\beta_i \leftarrow \beta_i^{\text{central}}$
  else
    $\alpha_i \leftarrow \alpha_i^{\text{up}}$
    $\beta_i \leftarrow \beta_i^{\text{up}}$

Note: After preprocessing, either central or upstream weighting may appear at a node, but in every case the selected coefficients satisfy

$$\alpha_i \geq 0 \qquad \text{and} \qquad \beta_i \geq 0.$$

The method therefore uses central weighting as much as possible while enforcing the positivity requirement.

To determine when upstream weighting is needed, consider

$$\alpha_i^{\text{central}} = \frac{\sigma^2 S_i^2}{(S_i - S_{i-1})(S_{i+1} - S_{i-1})} - \frac{rS_i}{S_{i+1} - S_{i-1}}.$$

For Black–Scholes, this is nonnegative exactly when

$$\sigma^2 \geq \frac{r(S_i - S_{i-1})}{S_i}.$$

On a uniform mesh with $S_i = i\Delta S$, this reduces to

$$\sigma^2 \geq \frac{r}{i}, \quad i \geq 1.$$

For example, if $r = 0.05$ and $\sigma = 0.4$, the condition is satisfied for all $i > 0$. If $r = 0.10$ and $\sigma = 0.2$, the condition holds for $i \geq 3$. Thus upstream weighting is typically needed only near $S = 0$ for standard Black–Scholes parameters, though it is often required more broadly in other models such as mean-reverting interest-rate problems.

The PDE is solved numerically on a finite domain $[0, S_{\max}]$. A practical way to choose $S_{\max}$ is

based on the exact solution of geometric Brownian motion:

$$\mathrm{d}S = rS\,\mathrm{d}t + \sigma S\,\mathrm{d}Z \qquad \text{and} \qquad S(T) = S_0 \exp\left[\left(r - \frac{\sigma^2}{2}\right)T + \sigma\phi\sqrt{T}\right], \quad \phi \sim \mathcal{N}(0,1).$$

If a three-standard-deviation event is treated as an extreme tail event, then a useful truncation is

$$S_{\max} \approx S_0 \exp\Big(rT + 3\sigma\sqrt{T}\Big).$$

In practice, for $\sigma < 0.4$ and $T < 1$ year, taking $S_{\max} \approx 10K$ is conservative. On nonuniform grids, where spacing can be larger near $S_{\max}$, such a large boundary is usually inexpensive.

Explicit methods face a restrictive stability condition. The allowable time step behaves like

$$\Delta\tau \asymp \frac{(\Delta S)^2}{\sigma^2 S^2},$$

which becomes especially severe when local mesh refinement is required, for example near barriers. The global time truncation order is first order in $\Delta\tau$, and the update is algebraically equivalent to a lattice method up to $O(\Delta\tau)$ terms.

Implicit and Crank–Nicolson Formulations

Introduce the differential operator

$$\mathcal{L}V := \frac{\sigma^2}{2}S^2 V_{SS} + rSV_S - rV,$$

so the Black–Scholes equation is

$$V_\tau = \mathcal{L}V. \tag{14.3}$$

Let $(\mathcal{L}V)_i^n$ denote a spatial finite-difference approximation of $(\mathcal{L}V)$ at (S_i, τ^n). A **fully implicit method** is

$$\frac{V_i^{n+1} - V_i^n}{\Delta\tau} = (\mathcal{L}V)_i^{n+1},$$

so all right-hand side terms are evaluated at the new time level. However, a fully implicit method has a global time truncation error of order $O(\Delta\tau)$. Instead, averaging the right-hand side between levels n and $n+1$ gives the **Crank–Nicolson method**:

$$\frac{V_i^{n+1} - V_i^n}{\Delta\tau} = \frac{1}{2}\Big((\mathcal{L}V)_i^{n+1} + (\mathcal{L}V)_i^n\Big).$$

On a uniform grid, Crank–Nicolson is second order in both space and time for a sufficiently smooth solution. More generally, on a sufficiently regular nonuniform mesh, if central differences are used in space and

$$(\Delta S)_{\max} := \max_i (S_{i+1} - S_i),$$

then finite-element analysis yields global discretization error of order

$$O\big((\Delta S)_{\max}^2, (\Delta\tau)^2\big)$$

even on a nonuniform spatial mesh.

Lecture 15 — Wednesday, March 04, 2015

After substituting the finite-difference formulas for diffusion and drift, both methods can be written in implementation form. The fully implicit method is

$$V_i^{n+1} = V_i^n + \Delta\tau\Big[\alpha_i\big(V_{i-1}^{n+1} - V_i^{n+1}\big) + \beta_i\big(V_{i+1}^{n+1} - V_i^{n+1}\big) - rV_i^{n+1}\Big], \tag{15.1}$$

and Crank–Nicolson is

$$\begin{aligned} V_i^{n+1} = V_i^n &+ \frac{\Delta\tau}{2}\Big[\alpha_i\big(V_{i-1}^{n+1} - V_i^{n+1}\big) + \beta_i\big(V_{i+1}^{n+1} - V_i^{n+1}\big) - rV_i^{n+1}\Big] \\ &+ \frac{\Delta\tau}{2}\Big[\alpha_i\big(V_{i-1}^{n} - V_i^{n}\big) + \beta_i\big(V_{i+1}^{n} - V_i^{n}\big) - rV_i^{n}\Big]. \end{aligned} \tag{15.2}$$

As before, α_i and β_i are obtained from the selected central or upstream stencil, after preprocessing has enforced nonnegativity.

Matrix Form, Local Volatility, and Tridiagonal Solvers

Define the vectors

$$\boldsymbol{V}^n := [V_0^n, V_1^n, \ldots, V_m^n]^\top \qquad \text{and} \qquad \boldsymbol{V}^{n+1} := [V_0^{n+1}, V_1^{n+1}, \ldots, V_m^{n+1}]^\top.$$

Let $\mathbf{M}$ be the tridiagonal matrix whose interior rows satisfy

$$[\mathbf{M}\boldsymbol{V}]_i = -\Delta\tau\,\alpha_i V_{i-1} + \Delta\tau(\alpha_i + \beta_i + r)V_i - \Delta\tau\,\beta_i V_{i+1}, \quad i = 1, \ldots, m-1. \tag{15.3}$$

At $S_0 = 0$, using the implicit boundary discretization of $V_\tau = -rV$,

$$\frac{V_0^{n+1} - V_0^n}{\Delta\tau} = -rV_0^{n+1},$$

which is equivalent to

$$(1 + r\Delta\tau)V_0^{n+1} = V_0^n.$$

At $S_m = S_{\max}$, let the prescribed Dirichlet value at the new time be g_m^{n+1}; for the call boundary used above,

$$g_m^{n+1} = S_m - Ke^{-r\tau^{n+1}}.$$

The last matrix row is therefore the identity row with right-hand side g_m^{n+1}, rather than the generally false condition $V_m^{n+1} = V_m^n$. If $\boldsymbol{b}^{n+1}$ equals $\boldsymbol{V}^n$ in the interior and contains the prescribed boundary right-hand sides, fully implicit stepping is

$$(I + \mathbf{M})\boldsymbol{V}^{n+1} = \boldsymbol{b}^{n+1}. \tag{15.4}$$

For Crank–Nicolson, define $\widehat{\mathbf{M}}$ by

$$[\widehat{\mathbf{M}}\boldsymbol{V}]_i = -\frac{\Delta\tau}{2}\alpha_i V_{i-1} + \frac{\Delta\tau}{2}(\alpha_i + \beta_i + r)V_i - \frac{\Delta\tau}{2}\beta_i V_{i+1}, \quad i = 1, \ldots, m-1, \tag{15.5}$$

and use the corresponding half-step boundary rows. For interior rows, Crank–Nicolson becomes

$$(\mathbf{I} + \widehat{\mathbf{M}})\boldsymbol{V}^{n+1} = (\mathbf{I} - \widehat{\mathbf{M}})\boldsymbol{V}^n. \tag{15.6}$$

Known boundary contributions are moved to the right-hand side; if boundary nodes are retained in the full vector, their rows are replaced by the boundary equations, including $V_m^{n+1} = g_m^{n+1}$. The compact homogeneous matrix formulas used below are the corresponding error equations, for which the boundary data are zero.

For a local-volatility function, coefficients depend on time level n through σ_i^n. The central formulas become

$$\alpha_i^{n,\text{central}} = \frac{(\sigma_i^n)^2 S_i^2}{(S_i - S_{i-1})(S_{i+1} - S_{i-1})} - \frac{rS_i}{S_{i+1} - S_{i-1}},$$
$$\beta_i^{n,\text{central}} = \frac{(\sigma_i^n)^2 S_i^2}{(S_{i+1} - S_i)(S_{i+1} - S_{i-1})} + \frac{rS_i}{S_{i+1} - S_{i-1}}.$$

Forward formulas are

$$\alpha_i^{n,\text{forward}} = \frac{(\sigma_i^n)^2 S_i^2}{(S_i - S_{i-1})(S_{i+1} - S_{i-1})},$$

$$\beta_i^{n,\text{forward}} = \frac{(\sigma_i^n)^2 S_i^2}{(S_{i+1} - S_i)(S_{i+1} - S_{i-1})} + \frac{rS_i}{S_{i+1} - S_i},$$

and backward formulas are

$$\alpha_i^{n,\text{backward}} = \frac{(\sigma_i^n)^2 S_i^2}{(S_i - S_{i-1})(S_{i+1} - S_{i-1})} - \frac{rS_i}{S_i - S_{i-1}},$$
$$\beta_i^{n,\text{backward}} = \frac{(\sigma_i^n)^2 S_i^2}{(S_{i+1} - S_i)(S_{i+1} - S_{i-1})}.$$

The algebraic update formulas retain the same form; only the coefficient definitions change.

Define $\mathbf{M}^n$ by replacing (α_i, β_i) with (α_i^n, β_i^n) in (15.3). On the interior, with known boundary terms moved to the right-hand side as above, the fully implicit method is

$$(I + \mathbf{M}^{n+1})\boldsymbol{V}^{n+1} = \boldsymbol{b}_{\text{imp}}^{n+1}. \tag{15.7}$$

Similarly, with half-step matrix $\widehat{\mathbf{M}}^n$, Crank–Nicolson is

$$(\mathbf{I} + \widehat{\mathbf{M}}^{n+1})\boldsymbol{V}^{n+1} = \boldsymbol{b}_{\text{CN}}^{n+1}, \tag{15.8}$$

where the interior entries of $\boldsymbol{b}_{\text{CN}}^{n+1}$ come from $(\mathbf{I} - \widehat{\mathbf{M}}^n)\boldsymbol{V}^n$ and both right-hand-side vectors include the known boundary contributions.

At each implicit or Crank–Nicolson step we solve

$$\mathbf{A}\boldsymbol{x} = \boldsymbol{b},$$

where $\mathbf{A}$ is tridiagonal. When the selected coefficients are nonnegative, $r \geq 0$, and the boundary rows are imposed consistently, the fully implicit matrix is diagonally dominant and is an M-matrix. In this setting, pivoting is not required in practice, and we compute

$$\mathbf{A} = \mathbf{L}\mathbf{U},$$

with lower triangular $\mathbf{L}$ and upper triangular $\mathbf{U}$, both banded with one off-diagonal. The solution is then obtained by forward substitution in $\mathbf{L}\boldsymbol{y} = \boldsymbol{b}$ and backward substitution in $\mathbf{U}\boldsymbol{x} = \boldsymbol{y}$.

Remark 15.1 For tridiagonal systems this procedure is linear in the number of grid nodes. In MATLAB implementations, `spdiags`, `lu`, and \ exploit sparsity efficiently; the estimate is roughly $8(m + 1)$ arithmetic operations for the core tridiagonal solve.

Convergence, Consistency, and Stability

Three distinct error mechanisms appear in practice: floating point roundoff, local discretization error from derivative approximation, and propagation of existing error from one time level to the next.

Definition 15.2 A finite-difference method is **convergent** if

$$\lim_{\Delta S, \Delta\tau \to 0} \left(V_i^n - V(S_i, \tau^n)\right) = 0$$

for all grid indices in the computational domain.

For truncation error analysis of Black–Scholes,

$$V_\tau - \mathcal{L}V = 0,$$

and let ϕ be a smooth test function with bounded derivatives.

Definition 15.3 The **truncation error** of a discrete Black–Scholes operator is

$$\text{T.E.} = \Big[(\phi_\tau)_i^n - (\mathcal{L}\phi)_i^n\Big]_{\text{discrete}} - \Big[(\phi_\tau - \mathcal{L}\phi)(S_i, \tau^n)\Big].$$

Let

$$\Delta S := \max_i (S_{i+1} - S_i) \qquad \text{and} \qquad \Delta\tau := \max_n (\tau^{n+1} - \tau^n).$$

Definition 15.4 A discretization is **consistent** if

$$\text{T.E.} \to 0 \qquad \text{as} \qquad \Delta S, \Delta\tau \to 0.$$

Example 15.5 Consider

$$U_\tau = U_{SS} \tag{15.9}$$

with discretization

$$\frac{U_i^{n+1} - U_i^n}{\Delta\tau} = \frac{U_{i+1}^n - 2U_i^n + U_{i-1}^n}{(\Delta S)^2}. \tag{15.10}$$

For smooth ϕ,

$$\frac{\phi_i^{n+1} - \phi_i^n}{\Delta\tau} = \phi_\tau(S_i, \tau^n) + O(\Delta\tau),$$

and

$$\frac{\phi_{i+1}^n - 2\phi_i^n + \phi_{i-1}^n}{(\Delta S)^2} = \phi_{SS}(S_i, \tau^n) + O((\Delta S)^2).$$

Hence the residual of (15.10) applied to ϕ is

$$O(\Delta\tau) + O((\Delta S)^2),$$

so the method is consistent.

A necessary condition for convergence is consistency. For smooth solutions on regular meshes, the expected convergence rates are $O(\Delta\tau, (\Delta S)^2)$ for explicit and fully implicit schemes, and $O((\Delta\tau)^2, (\Delta S)^2)$ for Crank–Nicolson. Nonsmooth payoffs can reduce these rates unless the initial steps are treated carefully.

Definition 15.6 A finite-difference method is **stable** if its discrete evolution operators remain uniformly bounded, in an appropriate norm, as the grid is refined over a fixed time interval. For a scalar Fourier mode, this reduces to requiring that the mode's amplification factor have magnitude at most one.

Theorem 15.7 (Lax equivalence theorem) For a consistent finite-difference approximation of a well-posed linear initial-value problem, stability is equivalent to convergence.

Explicit Stability Restriction and Practical Cost

For the heat equation, explicit time stepping is stable only when

$$0 < \frac{\Delta\tau}{(\Delta x)^2} \leq \frac{1}{2}.$$

The stability bound is sharp in the standard analysis and leads to a severe time step limitation when the spatial mesh is refined.

Example 15.8 A put with $K = 10$, $r = 5\%$, $\sigma = 20\%$, and $T = \frac{1}{2}$ illustrates this effect numerically: a ratio near $\frac{\Delta\tau}{(\Delta x)^2} = 0.5$ remains accurate, whereas slightly violating the bound produces unstable and nonphysical values. This is a primary reason to prefer implicit or Crank–Nicolson methods in practical pricing grids.

Binomial and trinomial lattices are algebraically equivalent to explicit finite-difference schemes, so they inherit a corresponding effective upper bound on the time step for stable behavior.

Lecture 16 — Friday, March 06, 2015

Convergence and Consistency

We set

$$\Delta S = \max_i (S_{i+1} - S_i) \qquad \text{and} \qquad \Delta\tau = \max_n (\tau^{n+1} - \tau^n).$$

Definition 16.1 A discretization of the Black–Scholes equation is consistent if its truncation error tends to zero as $\Delta S, \Delta\tau \to 0$.

Example 16.2 Consider

$$U_\tau = U_{SS}, \tag{16.1}$$

and the finite difference equation

$$\frac{U_i^{n+1} - U_i^n}{\Delta\tau} = \frac{U_{i+1}^n - 2U_i^n + U_{i-1}^n}{(\Delta S)^2}. \tag{16.2}$$

For a smooth test function $\phi(S, \tau)$,

$$\frac{\phi_i^{n+1} - \phi_i^n}{\Delta\tau} = (\phi_\tau)_i^n + O(\Delta\tau),$$

and

$$\frac{\phi_{i+1}^n - 2\phi_i^n + \phi_{i-1}^n}{(\Delta S)^2} = (\phi_{SS})_i^n + O((\Delta S)^2).$$

Hence

$$\frac{\phi_i^{n+1}-\phi_i^n}{\Delta\tau}-\frac{\phi_{i+1}^n-2\phi_i^n+\phi_{i-1}^n}{(\Delta S)^2}=O(\Delta\tau)+O((\Delta S)^2)=\left[(\phi_\tau-\phi_{SS})\right]_i^n+O(\Delta\tau)+O((\Delta S)^2),$$

so (16.2) is consistent for (16.1).

A necessary condition for convergence is that truncation error tends to zero as $\Delta\tau, \Delta S \to 0$. When the solution is smooth and the method is stable, the corresponding rates are $O(\Delta\tau, (\Delta S)^2)$ for explicit and fully implicit schemes, and $O((\Delta\tau)^2, (\Delta S)^2)$ for Crank–Nicolson. A kink or discontinuity in the payoff can degrade these rates.

For this well-posed linear problem, the Lax equivalence theorem makes stability the remaining criterion for a consistent scheme to converge.

Stability of the Explicit Method

For Black–Scholes, the explicit method is

$$V_i^{n+1}=V_i^n\big(1-(\alpha_i+\beta_i+r)\Delta\tau\big)+\alpha_i\Delta\tau\,V_{i-1}^n+\beta_i\Delta\tau\,V_{i+1}^n. \tag{16.3}$$

A stability estimate is applied to the difference between two numerical solutions with the same prescribed boundary data. Denote this perturbation again by V_i^n; it has homogeneous boundary data, so at $S=S_{\max}$,

$$V_m^{n+1}=0.$$

Assume central or upstream weighting has been selected so that

$$\alpha_i\geq 0 \qquad \text{and} \qquad \beta_i\geq 0.$$

From (16.3),

$$|V_i^{n+1}|\leq\big|V_i^n\big(1-(\alpha_i+\beta_i+r)\Delta\tau\big)\big|+\alpha_i\Delta\tau|V_{i-1}^n|+\beta_i\Delta\tau|V_{i+1}^n|. \tag{16.4}$$

Now assume

$$1-(\alpha_i+\beta_i+r)\Delta\tau\geq 0 \quad \text{for all } i.$$

Then

$$|V_i^{n+1}|\leq|V_i^n|\big(1-(\alpha_i+\beta_i+r)\Delta\tau\big)+\alpha_i\Delta\tau|V_{i-1}^n|+\beta_i\Delta\tau|V_{i+1}^n|.$$

Let

$$\|\boldsymbol{V}^n\|_\infty := \max_i |V_i^n|.$$

Then for any i,

$$|V_i^{n+1}| \le \|\boldsymbol{V}^n\|_\infty (1 - r\Delta\tau) \le \|\boldsymbol{V}^n\|_\infty. \tag{16.5}$$

Thus

$$\|\boldsymbol{V}^{n+1}\|_\infty \le \|\boldsymbol{V}^n\|_\infty. \tag{16.6}$$

Thus perturbations cannot grow in the maximum norm over the fixed time interval, which proves stability under the stated step restriction.

The stability condition can be rewritten as

$$\Delta\tau \le \frac{1}{\alpha_i + \beta_i + r} \quad \text{for all } i, \tag{16.7}$$

or equivalently

$$\Delta\tau \le \min_i \frac{1}{\alpha_i + \beta_i + r}. \tag{16.8}$$

This coefficient condition is sufficient for maximum-norm stability and monotonicity. It is not asserted to be necessary for every possible norm or finite boundary treatment.

For uniform spacing $\Delta S_{i+1/2} = \Delta S_{i-1/2} = \Delta S$ and central differencing,

$$\Delta\tau \le \frac{1}{\dfrac{\sigma^2 S_i^2}{(\Delta S)^2} + r} \sim \frac{(\Delta S)^2}{\sigma^2 S_i^2}, \quad \Delta S \to 0.$$

This can be very restrictive when local mesh refinement is used near barriers or near the strike of a digital option, and it is also costly for long maturity options and bond problems.

The fully implicit method is

$$V_i^{n+1} = V_i^n + \Delta\tau\Big[\alpha_i\big(V_{i-1}^{n+1} - V_i^{n+1}\big) + \beta_i\big(V_{i+1}^{n+1} - V_i^{n+1}\big) - rV_i^{n+1}\Big]. \tag{16.9}$$

Equivalently,

$$V_i^{n+1}\big[1 + \Delta\tau(r + \alpha_i + \beta_i)\big] = V_i^n + \Delta\tau\alpha_i V_{i-1}^{n+1} + \Delta\tau\beta_i V_{i+1}^{n+1}.$$

Since $\alpha_i, \beta_i, r \ge 0$,

$$|V_i^{n+1}|\big[1 + \Delta\tau(r + \alpha_i + \beta_i)\big] \le |V_i^n| + \Delta\tau\alpha_i|V_{i-1}^{n+1}| + \Delta\tau\beta_i|V_{i+1}^{n+1}|.$$

Therefore

$$|V_i^{n+1}|\big[1+\Delta\tau(r+\alpha_i+\beta_i)\big] \le \|\boldsymbol{V}^n\|_\infty + \Delta\tau(\alpha_i+\beta_i)\|\boldsymbol{V}^{n+1}\|_\infty.$$

Choose i^* such that $|V_{i^*}^{n+1}| = \|\boldsymbol{V}^{n+1}\|_\infty$. Then

$$\|\boldsymbol{V}^{n+1}\|_\infty\big[1+\Delta\tau(r+\alpha_{i^*}+\beta_{i^*})\big] \le \|\boldsymbol{V}^n\|_\infty + \Delta\tau(\alpha_{i^*}+\beta_{i^*})\|\boldsymbol{V}^{n+1}\|_\infty,$$

which gives

$$\|\boldsymbol{V}^{n+1}\|_\infty \le \frac{1}{1+r\Delta\tau}\|\boldsymbol{V}^n\|_\infty \le \|\boldsymbol{V}^n\|_\infty. \tag{16.10}$$

Hence the fully implicit method is unconditionally stable.

For Crank–Nicolson, define vector states

$$\boldsymbol{V}^n := [V_0^n, V_1^n, \ldots, V_m^n]^\top \qquad \text{and} \qquad \boldsymbol{V}^{n+1} := [V_0^{n+1}, V_1^{n+1}, \ldots, V_m^{n+1}]^\top,$$

and define the tridiagonal matrix $\widehat{\mathbf{M}}$ by

$$[\widehat{\mathbf{M}}\boldsymbol{V}^n]_i = -\frac{\Delta\tau}{2}\alpha_i V_{i-1}^n + \frac{\Delta\tau}{2}(\alpha_i+\beta_i+r)V_i^n - \frac{\Delta\tau}{2}\beta_i V_{i+1}^n, \quad i \ne 0, m.$$

At $S=0$, the first row uses the same form with $\alpha_i = \beta_i = 0$. For the homogeneous perturbation equation at $S_{\max}$, the last row is identically zero; prescribed nonhomogeneous boundary values belong in the right-hand side of the original pricing system.

The Crank–Nicolson system is

$$[\mathbf{I}+\widehat{\mathbf{M}}]\boldsymbol{V}^{n+1} = [\mathbf{I}-\widehat{\mathbf{M}}]\boldsymbol{V}^n. \tag{16.11}$$

If we set $\mathbf{M} = 2\widehat{\mathbf{M}}$, the fully implicit method is

$$[\mathbf{I}+\mathbf{M}]\boldsymbol{V}^{n+1} = \boldsymbol{V}^n.$$

Both methods are included in

$$[\mathbf{I}+\theta\mathbf{M}]\boldsymbol{V}^{n+1} = [\mathbf{I}-(1-\theta)\mathbf{M}]\boldsymbol{V}^n, \tag{16.12}$$

with $\theta = 1$ for the fully implicit method, $\theta = 1/2$ for Crank–Nicolson, and $\theta = 0$ for the explicit method.

Assume $\widehat{\mathbf{M}} \in \mathbb{R}^{(m+1)\times(m+1)}$ has a complete set of linearly independent eigenvectors $\boldsymbol{X}_k$ with eigenvalues λ_k:

$$\widehat{\mathbf{M}}\boldsymbol{X}_k = \lambda_k \boldsymbol{X}_k.$$

From (16.11),

$$\boldsymbol{V}^{n+1} = [\mathbf{I} + \widehat{\mathbf{M}}]^{-1}[\mathbf{I} - \widehat{\mathbf{M}}]\boldsymbol{V}^n,$$

so

$$\boldsymbol{V}^{n+1} = \left([\mathbf{I} + \widehat{\mathbf{M}}]^{-1}[\mathbf{I} - \widehat{\mathbf{M}}]\right)^{n+1}\boldsymbol{V}^0.$$

Write

$$\boldsymbol{V}^0 = \sum_k C_k \boldsymbol{X}_k.$$

Then

$$\boldsymbol{V}^{n+1} = \sum_k C_k \left(\frac{1-\lambda_k}{1+\lambda_k}\right)^{n+1} \boldsymbol{X}_k. \tag{16.13}$$

For boundedness as $n \to \infty$, it is necessary that

$$\left|\frac{1-\lambda_k}{1+\lambda_k}\right| \leq 1 \quad \text{for all } k. \tag{16.14}$$

This holds whenever

$$\mathrm{Re}(\lambda_k) \geq 0 \quad \text{for all } k.$$

Theorem 16.3 (Gershgorin circle theorem) Let $\mathbf{A} = [A_{ij}]_{i,j=1}^n$ be a complex matrix. Every eigenvalue λ of $\mathbf{A}$ lies in at least one disk

$$|\lambda - A_{ii}| \leq \sum_{j \neq i} |A_{ij}|.$$

Proof. Let λ be an eigenvalue of $\mathbf{A}$ with eigenvector $\boldsymbol{x} \neq \mathbf{0}$. Choose i such that $|x_i| = \max_j |x_j|$. From

$$\mathbf{A}\boldsymbol{x} = \lambda \boldsymbol{x}$$

we have

$$\lambda x_i = \sum_{j=1}^n A_{ij}x_j \qquad \text{and} \qquad (\lambda - A_{ii})x_i = \sum_{j \neq i} A_{ij}x_j.$$

Taking absolute values gives

$$|\lambda - A_{ii}|\,|x_i| \leq \sum_{j \neq i} |A_{ij}|\,|x_j| \leq |x_i| \sum_{j \neq i} |A_{ij}|.$$

Since $x_i \neq 0$, dividing by $|x_i|$ yields

$$|\lambda - A_{ii}| \leq \sum_{j \neq i} |A_{ij}|.$$

Thus λ lies in one of the Gershgorin disks. □

Now note that $\widehat{\mathbf{M}}$ has nonnegative diagonal entries and is row diagonally dominant:

$$\widehat{M}_{ii} \geq 0 \qquad \text{and} \qquad |\widehat{M}_{ii}| \geq \sum_{j \neq i} |\widehat{M}_{ij}|.$$

Applying the Gershgorin circle theorem to $\widehat{\mathbf{M}}$ implies that each eigenvalue lies in a disk centered on the nonnegative real axis with radius not exceeding the center. Hence

$$\mathrm{Re}(\lambda_k) \geq 0 \quad \text{for all } k.$$

So Crank–Nicolson satisfies the necessary spectral condition for unconditional stability. A full proof of sufficient conditions requires additional argument.

Lecture 17 — Monday, March 09, 2015

The fully implicit method is unconditionally stable and does not require a time step tied to mesh spacing. It can be written as

$$V_i^{n+1} = \frac{V_i^n + \Delta\tau\alpha_i V_{i-1}^{n+1} + \Delta\tau\beta_i V_{i+1}^{n+1}}{1 + \Delta\tau(r + \alpha_i + \beta_i)}.$$

If $\alpha_i, \beta_i \geq 0$, then the coefficients on the right hand side are nonnegative and their sum is less than one.

Proposition 17.1 For an explicit scheme, nonnegative update weights make the one-step map componentwise monotone and prevent a new extremum caused by a negative stencil weight. For a fully implicit scheme, the analogous conclusion follows when its system matrix is an M-matrix, because the inverse of that matrix is componentwise nonnegative.

The explicit method

$$V_i^{n+1} = V_i^n\big(1 - (\alpha_i + \beta_i + r)\Delta\tau\big) + \alpha_i\Delta\tau V_{i-1}^n + \beta_i\Delta\tau V_{i+1}^n$$

is also a positive coefficient discretization when

$$1 - (\alpha_i + \beta_i + r)\Delta\tau \geq 0,$$

which is exactly its stability condition.

Crank–Nicolson can be written as

$$\begin{aligned} V_i^{n+1}\left(1 + \frac{\Delta\tau}{2}(\alpha_i + \beta_i + r)\right) = {} & \frac{\alpha_i\Delta\tau}{2}\left(V_{i-1}^{n+1} + V_{i-1}^{n}\right) \\ & + \frac{\beta_i\Delta\tau}{2}\left(V_{i+1}^{n+1} + V_{i+1}^{n}\right) \\ & + V_i^n\left(1 - \frac{\Delta\tau}{2}(\alpha_i + \beta_i + r)\right). \end{aligned}$$

This method is not a positive coefficient discretization unless

$$\Delta\tau \leq \frac{2}{\alpha_i + \beta_i + r} \quad \text{for all } i.$$

This threshold is twice the maximum stable explicit time step.

For the standard dissipative Black–Scholes discretization with consistent homogeneous error boundary conditions, Crank–Nicolson is unconditionally stable in the usual spectral sense, but loss of positivity can matter in barrier and digital contracts. Oscillations may be small in option value but can be magnified in delta V_S and gamma V_{SS}.

Warning If Crank–Nicolson uses a step size above the positive coefficient threshold, oscillations may appear even though stability is preserved.

Positivity, Rannacher Smoothing, and Numerical Behavior

Rannacher smoothing replaces the first Crank–Nicolson step after each rough initial state or nonsmooth event by two half-sized fully implicit steps, then resumes Crank–Nicolson.

For discontinuous payoffs, we should also project or average the payoff onto the spatial grid before time stepping. On a locally uniform grid near the discontinuity, one simple choice is to assign the average of the left and right limits at the discontinuity.

For a standard put or call payoff, the payoff is continuous but not differentiable at $S = K$. If there is a node at the strike, the payoff is already represented by piecewise linear basis functions,

so additional smoothing is not required.

Remarkably, using two half-sized fully implicit startup steps recovers second-order convergence in many nonsmooth payoff cases. The method is not guaranteed to be oscillation free, and second-order convergence does not by itself imply no oscillations. In practice, however, this method works very well.

If discontinuities occur very frequently, such as daily discrete barrier monitoring, there may be too few time steps between events to smooth the solution sufficiently. In that case a fully implicit method may still be preferable.

For standard payoffs

$$V = \max(K - S, 0) \quad \text{for a put} \qquad \text{and} \qquad V = \max(S - K, 0) \quad \text{for a call,}$$

placing a grid node at $S = K$ is still not enough to recover clean second order behavior for standard Crank–Nicolson when the time step exceeds the explicit threshold. Subtle oscillations can remain in delta and gamma near the strike.

Consider a European put with

$$T = 0.25, \qquad K = 100, \qquad r = 0.10, \qquad S = 100, \qquad \text{and} \qquad \sigma = 0.8,$$

whose exact value is 14.45191.

For standard Crank–Nicolson:

Nodes	Timesteps	Value	Change	Ratio
68	25	14.50470		
135	50	14.41032	0.09438	
269	100	14.43238	0.02215	4.3
537	200	14.44246	0.01008	2.2
1073	400	14.44726	0.00480	2.1

For Crank–Nicolson with Rannacher smoothing:

Nodes	Timesteps	Value	Change	Ratio
68	25	14.41872		
135	50	14.44357	0.02485	
269	100	14.44982	0.00625	4.0
537	200	14.45138	0.00156	4.0
1073	400	14.45177	0.00039	4.0

These data show that standard Crank–Nicolson can converge slowly in this regime, while Rannacher startup recovers the expected second-order pattern. It is a robust default after a nonsmooth payoff or discrete event.

Timestep Selection

Constant time steps are often used, but adaptive time stepping is beneficial when rapid and slow temporal regimes coexist. Small steps are needed after sharp events, while larger steps can be efficient after initial transients decay.

A simple, but effective timestep selector, which uses only current timestep information to predict the next step, is based on the maximum relative change in the solution from one step to the next. Given current step $\Delta\tau^{n+1}$, choose the next step by

$$\Delta\tau^{n+2} = \frac{\texttt{dnorm}}{\texttt{MaxRelChange}}\,\Delta\tau^{n+1},$$

where `dnorm` is a user selected target relative change and

$$\texttt{MaxRelChange} = \max_i \frac{\left|V(S_i,\tau^n+\Delta\tau^{n+1}) - V(S_i,\tau^n)\right|}{\max\left(D, \left|V(S_i,\tau^n+\Delta\tau^{n+1})\right|, \left|V(S_i,\tau^n)\right|\right)}.$$

The scale factor D prevents very small steps when the solution itself is very small. For dollar units, $D = 1$ is a practical choice.

If `dnorm` $= 0.05$, the selector targets a maximum relative change near 0.05 per step. This is a practical heuristic rather than a truncation-error estimate: decreasing `dnorm` generally forces smaller steps, but by itself does not prove convergence. If the solution changes rapidly the selector reduces the step, and if the solution changes slowly it increases the step.

Near discrete barrier dates, local one step indicators are unreliable, so timestep selection should be disabled near those points.

What have we learned? For the standard Black–Scholes equation, upstream first order differencing is rarely needed to enforce positivity. Positive coefficient schemes preserve nonnegative option values for finite meshes and timesteps. Fully implicit stepping is positive but first order in time. Crank–Nicolson can lose positivity when the step exceeds twice the explicit threshold. For nonsmooth payoffs, Rannacher startup with sensible timestep selection gives robust accuracy in practice.

Lecture 18 — Wednesday, March 11, 2015

5. American Options and Dynamic Programming

American and Bermuda Contracts

An **American option** gives the holder the right to buy or sell the underlying asset at strike K at any time $t \leq T$. A **Bermuda option** allows exercise only on a discrete set of dates

$$0 \leq t_1 < t_2 < \cdots < t_k \leq T.$$

In computation, a Bermuda contract with many exercise dates is often used as an approximation to an American contract.

Most stock options are American, while index options are typically European. The key modeling questions are the **optimal exercise strategy**, the **fair value**, and the numerical procedure used to compute that value.

The no-arbitrage price reflects both the holder's optimal exercise policy and the writer's cost of hedging the resulting liability. Hence the American option value is the value of the contract when the holder exercises optimally.

A European option is a special Bermuda option with maturity T as the only exercise date. At maturity, the optimal rule is immediate: exercise if and only if the option is in the money.

Sequential Decisions and Bellman Optimality

American options are **weakly path dependent**: at time t, whether the contract still exists depends on whether exercise has occurred earlier along the price path.

Let $V^A(S,t)$ denote the American option value at time t, conditional on no prior exercise. Let C_t^A and P_t^A denote American call and put values, and let C_t^E and P_t^E denote the corresponding European values.

Proposition 18.1 For fixed underlying, strike, and maturity,

$$C_t^A \geq C_t^E \qquad \text{and} \qquad P_t^A \geq P_t^E.$$

Also,

$$V^A(S,t) \geq \text{payoff}(S).$$

Assume the underlying pays no dividend. The European call satisfies

$$C^E(S,t) \geq \max\left(S - Ke^{-r(T-t)}, 0\right),$$

which implies the call continuation value remains above immediate exercise payoff. By contrast, a European put can lie below immediate payoff for some states.

Proposition 18.2 When there is no dividend payment and $r \geq 0$, early exercise is never optimal for an American call, but it may be optimal for an American put.

For a Bermuda option with times $t_0, t_1, \ldots, t_N$, the holder makes a sequence of exercise decisions

$$\text{ex}_0, \text{ex}_1, \ldots, \text{ex}_N,$$

where each decision indicates exercise or continuation at that date.

Definition 18.3 Given decision sequence $\{\text{ex}_j\}_{j=0}^N$, the **stopping time** is

$$\tau = \min\{t_j : \text{ex}_j = 1\},$$

with the convention $\tau = T$ if no earlier exercise occurs.

The stopping time is random because the decision depends on the realized state. To find the optimal exercise policy, current and future decisions must be optimized together, so American

pricing is a stochastic programming problem and no closed form valuation formula is generally available.

The **Bellman principle of optimality** says that in a sequential decision problem, once a state is reached, all remaining decisions must be optimal for that state. This principle supports backward time iteration and yields a **dynamic programming** method for American options.

Binomial Valuation with Early Exercise

Consider one step in a binomial tree from t to $t+1$ with up and down states. Let V_{t+1}^u and V_{t+1}^d be option values at the next time level, conditional on no exercise before $t+1$.

At time t, define the **replicating portfolio**

$$\Pi_t = \{\xi_t S_t, \eta_t e^{-r\delta t}\},$$

where (ξ_t, η_t) is chosen to replicate the one step continuation payoff. Its no arbitrage value is

$$\Pi_t = e^{-r\delta t}\left(q^* V_{t+1}^u + (1-q^*)V_{t+1}^d\right), \quad q^* = \frac{e^{r\delta t} - d}{u - d}.$$

Definition 18.4 The **continuation value** at time t is the value of the contract conditional on not exercising at t. In the one step binomial setting, this value equals Π_t.

For a European option, no arbitrage implies $V_t = \Pi_t$. If $V_t < \Pi_t$, then buying the option and shorting the replicating portfolio is an arbitrage. If $V_t > \Pi_t$, then shorting the option and buying the replicating portfolio is an arbitrage.

For an American option, no arbitrage implies only

$$V_t \geq \Pi_t,$$

because the short option position can be forced into early assignment.

If $\Pi_t <$ payoff(S_t), immediate exercise is optimal and $V_t =$ payoff(S_t). If $\Pi_t \geq$ payoff(S_t), no arbitrage implies $V_t = \Pi_t$.

Proposition 18.5 At each node in the binomial model,

$$V_t = \max\left(\text{payoff}(S_t), \Pi_t\right) = \max\left(\text{payoff}(S_t), e^{-r\delta t}\left(q^* V_{t+1}^u + (1-q^*)V_{t+1}^d\right)\right).$$

The **optimal exercise decision** is

$$\text{continue if } \Pi_t > \text{payoff}(S_t) \qquad \text{and} \qquad \text{exercise otherwise.}$$

Definition 18.6 The **optimal exercise time** is

$$\tau^* = \min\{t_j : V_{t_j} = \text{payoff}(S_{t_j})\}.$$

American options are significantly harder to price by Monte Carlo than European options. Path simulation moves forward in time, but the exercise decision at time t requires a continuation value from that state onward. Estimating this continuation value requires additional conditional path information at each candidate exercise state, which creates a nested backward optimization structure inside forward simulation.

Lecture 19 — Friday, March 13, 2015

Dividend Yield and the Black–Scholes Equation

Assume the underlying pays a constant continuous **dividend yield** q. Over an interval $\mathrm{d}t$, the dividend paid per share is $qS_t\,\mathrm{d}t$. A consistent stock model is

$$\frac{\mathrm{d}S_t + qS_t\,\mathrm{d}t}{S_t} = \mu\,\mathrm{d}t + \sigma\,\mathrm{d}X_t,$$

or equivalently

$$\mathrm{d}S_t = (\mu - q)S_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}X_t.$$

In the hedge portfolio derivation, dividend payments modify portfolio growth. If

$$\Pi_t = \{V_t, -\xi S_t\},$$

then

$$\mathrm{d}\Pi_t = \mathrm{d}V_t - \xi\,\mathrm{d}S_t - qS_t\xi\,\mathrm{d}t.$$

Hence the Black–Scholes PDE with dividend yield is

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r-q)S\frac{\partial V}{\partial S} - rV = 0.$$

Early Exercise for American Calls with Dividend Yield

Consider an American call when the underlying pays continuous dividend yield. The European call has asymptotic behavior

$$C(S,t) \approx Se^{-q(T-t)} - Ke^{-r(T-t)} \quad \text{as } S \to +\infty.$$

For sufficiently large S, it is possible that

$$Se^{-q(T-t)} - Ke^{-r(T-t)} < S - K,$$

so the European call value can lie below payoff. Therefore, for sufficiently large S, immediate exercise can be optimal for an American call.

When $q > 0$ and the standard one-boundary structure holds, there is a critical asset level S_t^* above which call exercise is optimal. The curve $\{S_t^*\}_{t=0}^T$ is the **early exercise boundary**. For $q = 0$ and $r \geq 0$, the boundary is effectively at infinity before expiry because early exercise is never optimal.

Definition 19.1 The **exercise region** and **continuation region** are

$$\mathcal{E} = \{(S,t) : V(S,t) = \text{payoff}(S)\} \qquad \text{and} \qquad \mathcal{C} = \{(S,t) : V(S,t) > \text{payoff}(S)\}.$$

In $\mathcal{E}$ it is optimal to exercise, and in $\mathcal{C}$ it is optimal to continue. Determining $\{S_t^*\}$ is part of the valuation problem.

Hedging Inequalities for American Options

Assume

$$\frac{\mathrm{d}S_t}{S_t} = \mu\,\mathrm{d}t + \sigma\,\mathrm{d}W_t,$$

and $V(S,t)$ is twice differentiable in S. Consider the hedge portfolio

$$\Pi_t = \left\{ V_t, -\frac{\partial V}{\partial S} S_t \right\}.$$

Ito's lemma yields

$$\mathrm{d}\Pi_t = \left(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} \right) \mathrm{d}t.$$

For a European option, no arbitrage with long and short trading implies $\mathrm{d}\Pi_t = r\Pi_t \,\mathrm{d}t$. For an American option, the short option argument can fail because exercise can occur at time t.

Set a cash account $B_t = -\Pi_t$ and define the self financing strategy

$$\overline{\Pi}_t = \left\{ V_t, -\frac{\partial V}{\partial S} S_t, B_t \right\}.$$

Because $\overline{\Pi}_t = 0$, if $\mathrm{d}\Pi_t > r\Pi_t \,\mathrm{d}t$ then $\mathrm{d}\overline{\Pi}_t > 0$, which is an arbitrage. Therefore

$$\mathrm{d}\Pi_t \leq r\Pi_t \,\mathrm{d}t,$$

equivalently

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV \leq 0.$$

If $V_t > \text{payoff}(S_t)$, assume for contradiction that $\mathrm{d}\Pi_t < r\Pi_t \,\mathrm{d}t$. Now take $B_t = \Pi_t$ and

$$\overline{\Pi}_t = \left\{ -V_t, \frac{\partial V}{\partial S} S_t, B_t \right\}.$$

Then $\overline{\Pi}_t = 0$. If exercise occurs at t, unwinding gives

$$V_t - \text{payoff}(S_t) > 0.$$

If exercise does not occur at t, the assumed strict inequality gives $\mathrm{d}\overline{\Pi}_t > 0$. Either way, there is arbitrage. Hence when $V_t > \text{payoff}(S_t)$ we must have equality in the hedge growth relation.

Financially, $V(S,t) > \text{payoff}(S)$ means $(S,t) \in \mathcal{C}$, so the usual no arbitrage equality applies in the continuation region.

Proposition 19.2 In the continuation region,

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r-q)S\frac{\partial V}{\partial S} - rV = 0.$$

Partial Differential Complementarity Formulation

No arbitrage implies

$$V(S,t) \geq \text{payoff}(S),$$

and

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r-q)S\frac{\partial V}{\partial S} - rV \leq 0.$$

In addition,

$$(V(S,t) - \text{payoff}(S)) \left(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r-q)S\frac{\partial V}{\partial S} - rV \right) = 0.$$

Together with terminal condition $V(S,T) = \text{payoff}(S)$, this gives a **partial differential complementarity problem**.

Definition 19.3 Let $\mathcal{L}$ denote the Black–Scholes operator with dividend yield,

$$\mathcal{L}V = \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r-q)S\frac{\partial V}{\partial S} - rV.$$

The American valuation problem is the **complementarity system**

$$\frac{\partial V}{\partial t} + \mathcal{L}V \leq 0, \qquad V - \text{payoff} \geq 0, \qquad \text{and} \qquad (V - \text{payoff}) \left(\frac{\partial V}{\partial t} + \mathcal{L}V \right) = 0,$$

for $0 \leq S < \infty$, $0 < t < T$, with $V(S,T) = \text{payoff}(S)$.

The early exercise boundary is determined implicitly by the solution. When the usual structure with one exercise boundary holds, a put has an exercise region of the form $\{S : S \leq S_t^*\}$, while a dividend-paying call has an exercise region of the form $\{S : S \geq S_t^*\}$.

Using time to maturity $\tau = T - t$ and payoff function $V^*(S)$, the same problem is written as a **linear complementarity problem**

$$V_\tau - \frac{\sigma^2}{2} S^2 V_{SS} - (r-q)SV_S + rV \geq 0,$$

$$V - V^* \geq 0,$$
$$(V - V^*)\left(V_\tau - \frac{\sigma^2}{2}S^2 V_{SS} - (r-q)SV_S + rV\right) = 0.$$

A common numerical strategy, including lattice methods, enforces the exercise constraint explicitly at each step. For example, with a fully implicit step, first compute a provisional unconstrained value $\widehat{V}_i^{n+1}$ at τ^{n+1}. Then impose

$$V_i^{n+1} \geq \text{payoff}(S_i)$$

before advancing to the next time level.

Lecture 20 — Monday, March 16, 2015

Suppose we want to price an American option with payoff $V^*(S,\tau)$, where $\tau = T - t$ is time to expiry. The continuous problem is a partial differential **linear complementarity problem**:

$$V_\tau - \frac{1}{2}\sigma^2 S^2 V_{SS} - rSV_S + rV \geq 0 \qquad \text{and} \qquad V - V^* \geq 0,$$
$$\left(V - V^*\right)\left(V_\tau - \frac{1}{2}\sigma^2 S^2 V_{SS} - rSV_S + rV\right) = 0.$$

After space and time discretization, collect values at one time level into vectors

$$\boldsymbol{V}^{n+1} = \begin{bmatrix} V_0^{n+1} \\ V_1^{n+1} \\ \vdots \\ V_m^{n+1} \end{bmatrix} \qquad \text{and} \qquad \boldsymbol{V}^{n} = \begin{bmatrix} V_0^{n} \\ V_1^{n} \\ \vdots \\ V_m^{n} \end{bmatrix},$$

and define the tridiagonal matrix $\mathbf{M}$ by

$$[\mathbf{M}\boldsymbol{V}^n]_i = -\Delta\tau\,\alpha_i V_{i-1}^n + \Delta\tau(\alpha_i + \beta_i + r)V_i^n - \Delta\tau\,\beta_i V_{i+1}^n.$$

Eliminate prescribed boundary nodes, or equivalently move their known contributions into right-hand-side vectors $\boldsymbol{b}_{\text{exp}}^n$ and $\boldsymbol{b}_{\text{imp}}^n$. For European options,

$$\boldsymbol{V}^{n+1} = \boldsymbol{b}_{\text{exp}}^n \quad \text{(explicit)},$$
$$(\mathbf{I} + \mathbf{M})\boldsymbol{V}^{n+1} = \boldsymbol{b}_{\text{imp}}^n \quad \text{(fully implicit)}.$$

With homogeneous boundary data, these vectors reduce to $(\mathbf{I}-\mathbf{M})\boldsymbol{V}^n$ and $\boldsymbol{V}^n$, respectively.

For American options, the explicit form is

$$\boldsymbol{V}^{n+1}-\boldsymbol{b}^n_{\text{exp}}\geq 0 \qquad \text{and} \qquad \boldsymbol{V}^{n+1}-\boldsymbol{V}^*\geq 0,$$

$$(\boldsymbol{V}^{n+1}-\boldsymbol{V}^*)\cdot *\left(\boldsymbol{V}^{n+1}-\boldsymbol{b}^n_{\text{exp}}\right)=\mathbf{0}.$$

The fully implicit form is

$$(\mathbf{I}+\mathbf{M})\boldsymbol{V}^{n+1}-\boldsymbol{b}^n_{\text{imp}}\geq 0 \qquad \text{and} \qquad \boldsymbol{V}^{n+1}-\boldsymbol{V}^*\geq 0,$$

$$(\boldsymbol{V}^{n+1}-\boldsymbol{V}^*)\cdot *\left((\mathbf{I}+\mathbf{M})\boldsymbol{V}^{n+1}-\boldsymbol{b}^n_{\text{imp}}\right)=\mathbf{0}.$$

Definition 20.1 In finite dimensions, the American step is the **linear complementarity problem**

$$\mathbf{A}\boldsymbol{V}^{n+1}-\boldsymbol{b}^n\geq 0 \qquad \text{and} \qquad \boldsymbol{V}^{n+1}-\boldsymbol{V}^*\geq 0,$$

$$(\boldsymbol{V}^{n+1}-\boldsymbol{V}^*)\cdot *(\mathbf{A}\boldsymbol{V}^{n+1}-\boldsymbol{b}^n)=\mathbf{0}.$$

The complexity depends strongly on $\mathbf{A}$. A general LCP can be difficult, but the M-matrix structure inherited from a monotone PDE discretization supports specialized and reliable solvers.

For an explicit evaluation of the American constraint, in the fully implicit scheme, first solve

$$(\mathbf{I}+\mathbf{M})\widehat{\boldsymbol{V}}^{n+1}=\boldsymbol{b}^n_{\text{imp}},$$

then project componentwise,

$$V_i^{n+1}=\max\left(V_i^*,\widehat{V}_i^{n+1}\right), \quad i=0,\ldots,m.$$

This is easy to implement, but the resulting state is inconsistent with the PDE solve at the same step. Delta becomes nonsmooth across the exercise boundary, and Crank–Nicolson loses second-order behavior.

Example 20.2 For an American put with $S=100$, $K=100$, $\sigma=0.2$, $T=0.25$, $r=0.10$:

Nodes	$\Delta\tau$	Explicit Constraint Value
100	0.0040	3.05060
200	0.0020	3.06037
400	0.0010	3.06519
800	0.0005	3.06761

If the constraint is handled implicitly, the LCP is solved directly. Common approaches include projected SOR, UL-type methods for constant-volatility standard options, Newton-type optimization methods, and **penalty methods**. Penalty methods are attractive because they generalize to multi-factor settings and preserve sparse linear system structure.

Penalty Methods

Definition 20.3 The complementarity constraint can be approximated by the **penalized system**

$$\mathbf{A}\boldsymbol{V}^{n+1} - \boldsymbol{b}^n - \text{Large}\,\max(\boldsymbol{V}^* - \boldsymbol{V}^{n+1}, \mathbf{0}) = \mathbf{0},$$

where the maximum is componentwise and `Large`> 0.

Proposition 20.4 If $\boldsymbol{V}^{n+1}$ satisfies the penalized system, then

$$\mathbf{A}\boldsymbol{V}^{n+1} - \boldsymbol{b}^n \geq \mathbf{0}.$$

Moreover, if $V_i^{n+1} \geq V_i^*$ then

$$\left(\mathbf{A}\boldsymbol{V}^{n+1} - \boldsymbol{b}^n\right)_i = 0,$$

and if $V_i^{n+1} < V_i^*$ then

$$V_i^* - V_i^{n+1} = \frac{1}{\text{Large}}\left(\mathbf{A}\boldsymbol{V}^{n+1} - \boldsymbol{b}^n\right)_i.$$

Proof. Rearranging the penalized equation gives

$$\mathbf{A}\boldsymbol{V}^{n+1} - \boldsymbol{b}^n = \text{Large}\,\max(\boldsymbol{V}^* - \boldsymbol{V}^{n+1}, \mathbf{0}) \geq \mathbf{0},$$

and the two componentwise cases follow immediately. □

Define the diagonal matrix

$$\overline{\mathbf{P}}(\boldsymbol{V}^{n+1})_{ii} = \begin{cases} \text{Large}, & V_i^{n+1} < V_i^*, \\ 0, & \text{otherwise.} \end{cases}$$

Then the nonlinear system can be written as

$$\left(\mathbf{A} + \overline{\mathbf{P}}(\boldsymbol{V}^{n+1})\right) \boldsymbol{V}^{n+1} = \boldsymbol{b}^n + \overline{\mathbf{P}}(\boldsymbol{V}^{n+1}) \boldsymbol{V}^*.$$

For Crank–Nicolson, let $\widehat{\mathbf{M}}$ be defined by

$$[\widehat{\mathbf{M}} \boldsymbol{V}^n]_i = -\frac{\Delta\tau}{2}\alpha_i V_{i-1}^n + \frac{\Delta\tau}{2}(\alpha_i + \beta_i + r) V_i^n - \frac{\Delta\tau}{2}\beta_i V_{i+1}^n.$$

Starting from $(\boldsymbol{V}^{n+1})^{(0)} = \boldsymbol{V}^n$, iterate

$$\left(\mathbf{I} + \widehat{\mathbf{M}} + \overline{\mathbf{P}}((\boldsymbol{V}^{n+1})^{(k)})\right) (\boldsymbol{V}^{n+1})^{(k+1)} = \left(\mathbf{I} - \widehat{\mathbf{M}}\right) \boldsymbol{V}^n + \overline{\mathbf{P}}((\boldsymbol{V}^{n+1})^{(k)}) \boldsymbol{V}^*,$$

until

$$\max_i \frac{\left|(V_i^{n+1})^{(k+1)} - (V_i^{n+1})^{(k)}\right|}{\max\left(1, \left|(V_i^{n+1})^{(k+1)}\right|\right)} < \texttt{tol}.$$

Note that the system matrix must usually be refactorized at each nonlinear iteration.

If an iterate falls below payoff at node i, then $\overline{\mathbf{P}}_{ii}$ switches to `Large`, which raises the next iterate at that node. If an iterate stays above payoff, no penalty is applied there.

Proposition 20.5 If the residual $\left(\mathbf{A}\boldsymbol{V}^{n+1} - \boldsymbol{b}^n\right)_i$ remains bounded under a fixed scaling, then the violation of the payoff constraint in the exercise region is $O(1/\text{Large})$. After nondimensionalizing the system so that the relevant residuals are order one, a practical starting choice is Large $\approx 1/\texttt{tol}$.

Taking `Large`$\to \infty$ improves constraint enforcement, but too large a value can cause roundoff issues once it approaches the reciprocal of machine precision.

Example 20.6 Average nonlinear iterations per step for the same American put benchmark:

Nodes	$\Delta\tau$	Iterations/step ($\sigma = 0.2$)	Iterations/step ($\sigma = 0.8$)
100	0.0040	2.3	2.2
200	0.0020	2.3	2.2
400	0.0010	2.2	2.3
800	0.0005	2.2	2.3

When $\Delta S = O(\Delta\tau)$, the iteration count is roughly constant as the mesh is refined and not strongly sensitive to volatility.

Example 20.7 Direct comparison of implicit and explicit constraint handling:

Nodes	$\Delta\tau$	Implicit Constraint	Explicit Constraint
100	0.0040	3.06704	3.05060
200	0.0020	3.06910	3.06037
400	0.0010	3.06975	3.06519
800	0.0005	3.06997	3.06761

Implicit enforcement is more accurate and better aligned with Crank–Nicolson than explicit post projection.

Lecture 21 — Wednesday, March 18, 2015

Return to the penalized formulation

$$\mathbf{A}\boldsymbol{V}^{n+1} - \boldsymbol{b}^n - \text{Large max}(\boldsymbol{V}^* - \boldsymbol{V}^{n+1}, \mathbf{0}) = \mathbf{0}.$$

Let

$$\overline{\mathbf{P}}(\boldsymbol{V}^{n+1})_{ii} = \begin{cases} \text{Large}, & V_i^{n+1} < V_i^*, \\ 0, & \text{otherwise.} \end{cases}$$

Then

$$\left(\mathbf{A} + \overline{\mathbf{P}}(\boldsymbol{V}^{n+1})\right) \boldsymbol{V}^{n+1} = \boldsymbol{b}^n + \overline{\mathbf{P}}(\boldsymbol{V}^{n+1})\boldsymbol{V}^*.$$

For Crank–Nicolson, with $(\boldsymbol{V}^{n+1})^{(0)} = \boldsymbol{V}^n$, the iteration is

$$\left(\mathbf{I} + \widehat{\mathbf{M}} + \overline{\mathbf{P}}((\boldsymbol{V}^{n+1})^{(k)})\right)(\boldsymbol{V}^{n+1})^{(k+1)} = \left(\mathbf{I} - \widehat{\mathbf{M}}\right)\boldsymbol{V}^n + \overline{\mathbf{P}}((\boldsymbol{V}^{n+1})^{(k)})\boldsymbol{V}^*.$$

Terminate when the maximum relative componentwise change is below `tol`. Note that matrix refactorization is usually required at each nonlinear iteration.

The iteration logic is straightforward. If $(V_i^{n+1})^{(k)} > V_i^*$, then $\overline{\mathbf{P}}_{ii} = 0$. If $(V_i^{n+1})^{(k)} < V_i^*$, then $\overline{\mathbf{P}}_{ii} =$ Large and the next iterate is pushed upward. At convergence, either $V_i^{n+1} > V_i^*$ with zero penalty, or $V_i^{n+1} = V_i^* - \varepsilon_i$ with $\varepsilon_i = O(1/\text{Large})$.

In the exercise region, accuracy improves as `Large` increases, but roundoff can become problematic if `Large`$> 1/\epsilon_{\text{mach}}$. Under a fixed scaling and bounded residuals, the constraint violation is $O(1/\text{Large})$; after scaling residuals to order one, Large $\approx 1/$`tol` is a practical starting choice.

Example 21.1 Nonlinear iterations per step for the benchmark American put, Crank–Nicolson with two startup implicit steps:

Nodes	$\Delta\tau$	$\sigma = 0.2$	$\sigma = 0.8$
100	0.0040	2.3	2.2
200	0.0020	2.3	2.2
400	0.0010	2.2	2.3
800	0.0005	2.2	2.3

Iterations per timestep are roughly constant and are not strongly sensitive to volatility.

Consider convergence in detail with Rannacher smoothing, $T = 0.25$, $r = 0.10$, $K = 100$, $S = 100$, and work measured in multiplies and divides.

Example 21.2 $\sigma = 0.2$ with constant timesteps:

Nodes	Steps	Iters	Value	Change	Ratio	Work
55	25	25	3.04600			8.3×10^3
109	50	50	3.06049	0.01449		3.3×10^4
217	100	100	3.06598	0.00549	2.6	1.3×10^5
433	200	200	3.06822	0.00224	2.5	5.2×10^5
865	400	400	3.06922	0.00100	2.2	2.1×10^6
55	25	58	3.05607			1.7×10^4
109	50	117	3.06555	0.00948		6.7×10^4
217	100	234	3.06854	0.00299	3.2	2.7×10^5
433	200	471	3.06953	0.00099	3.0	1.1×10^6
865	400	937	3.06988	0.00035	2.8	4.3×10^6

First block: explicit constraint. Second block: implicit constraint.

Example 21.3 $\sigma = 0.8$ with constant timesteps:

Nodes	Steps	Iters	Value	Change	Ratio	Work
68	25	25	14.61682			1.0×10^4
135	50	50	14.65685	0.04002		4.1×10^4
269	100	100	14.67045	0.01360	2.9	1.6×10^5
537	200	200	14.67542	0.00497	2.7	6.5×10^5
1073	400	400	14.67738	0.00196	2.5	2.6×10^6
68	25	56	14.62708			2.1×10^4
135	50	112	14.66219	0.03511		8.5×10^4
269	100	226	14.67324	0.01105	3.2	3.5×10^5
537	200	461	14.67686	0.00362	3.1	1.5×10^6
1073	400	940	14.67813	0.00127	2.9	6.0×10^6

First block: explicit constraint. Second block: implicit constraint.

These results show better accuracy for implicit treatment, but with constant timesteps convergence is still not fully quadratic.

Near $\tau = T - t = 0$, both the value and the early exercise boundary vary rapidly. If timesteps

satisfy

$$\max_i \left|V_i^{n+1} - V_i^n\right| \approx d,$$

then $O(d^2)$ error can be achieved. This is exactly the behavior encouraged by the **variable timestep selector**: small steps when changes are rapid, larger steps when changes are slow.

Quadratic convergence under refinement is observed when the number of spatial nodes is doubled, the target change level d is halved, and the initial timestep τ^0 is reduced by a factor of 4.

Example 21.4 Same benchmark with timestep selector, $\sigma = 0.2$:

Nodes	Steps	Iters	Value	Change	Ratio	Work
55	18	18	3.04499			9.9×10^3
109	33	33	3.05825	0.01326		3.6×10^4
217	63	63	3.06425	0.00600	2.2	1.4×10^5
433	122	122	3.06717	0.00292	2.1	5.3×10^5
865	239	239	3.06863	0.00146	2.0	2.1×10^6
55	18	45	3.06403			1.5×10^4
109	33	85	3.06867	0.00464		5.7×10^4
217	63	164	3.06975	0.00108	4.3	2.2×10^5
433	122	322	3.07002	0.00027	4.0	8.5×10^5
865	239	636	3.07008	0.00006	4.5	3.2×10^6

First block: explicit constraint. Second block: implicit constraint.

With timestep selection, the implicit method regains quadratic convergence. The same quadratic behavior is observed for $\sigma = 0.8$.

Plots of delta and gamma for both Crank–Nicolson and fully implicit schemes show strongest sensitivity near the free boundary and clearly expose irregularity introduced by explicit post projection.

Explicit constraint handling makes delta nonsmooth at the exercise boundary and converges only at $O(\Delta\tau)$, so it wastes the formal second-order behavior of Crank–Nicolson. Implicit handling is more accurate, but constant timesteps are not enough to reveal quadratic convergence, which is why timestep selection is needed in practice. PSOR-type methods have poor complexity and poor multi-factor scalability, whereas penalty methods converge in one iteration when no constraint is active, keep iterations per timestep stable under refinement, and extend naturally to multi-factor

models.

A recurring matrix property is the **M-matrix** property. For fully implicit stepping,

$$(\mathbf{I} + \mathbf{M})\boldsymbol{V}^{n+1} = \boldsymbol{V}^{n}.$$

Definition 21.5 A matrix $\mathbf{Q}$ is a **nonsingular M-matrix** if it can be written as

$$\mathbf{Q} = s\mathbf{I} - \mathbf{B},$$

where $\mathbf{B} \geq \mathbf{0}$ componentwise and $s > \rho(\mathbf{B})$. Equivalently, $\mathbf{Q}$ has nonpositive off-diagonal entries and $\mathbf{Q}^{-1} \geq \mathbf{0}$ componentwise. A useful sufficient condition in finite-difference applications is that $\mathbf{Q}$ be irreducible and weakly diagonally dominant by rows, with positive diagonal entries, nonpositive off-diagonal entries, and at least one strictly dominant row.

Theorem 21.6 If $\mathbf{Q}$ is an M-matrix, then

$$\mathbf{Q}^{-1} \geq \mathbf{0}$$

componentwise, and all diagonal entries of $\mathbf{Q}^{-1}$ are strictly positive.

This inverse positivity property is central in finite-difference finance because it supports monotonicity, positivity of option values, and stable handling of inequality constraints.

Lecture 22 — Monday, March 23, 2015

Start from the fully implicit system

$$(\mathbf{I} + \mathbf{M})\boldsymbol{V}^{n+1} = \boldsymbol{V}^{n},$$

with

$$[\mathbf{M}\boldsymbol{V}^{n}]_i = -\Delta\tau\,\alpha_i V_{i-1}^{n} + \Delta\tau(\alpha_i + \beta_i + r)V_i^{n} - \Delta\tau\,\beta_i V_{i+1}^{n}.$$

If differencing is chosen so that $\alpha_i, \beta_i \geq 0$ and the boundary condition at $S_{\max}$ is imposed consistently, then $\mathbf{I} + \mathbf{M}$ is an M-matrix.

Theorem 22.1 If fully implicit timestepping is used and $\mathbf{I}+\mathbf{M}$ is an M-matrix, then

$$\boldsymbol{V}^n \geq \boldsymbol{W}^n \implies \boldsymbol{V}^{n+1} \geq \boldsymbol{W}^{n+1}$$

componentwise.

Proof. The two solutions satisfy

$$(\mathbf{I}+\mathbf{M})\boldsymbol{W}^{n+1} = \boldsymbol{W}^n \qquad \text{and} \qquad (\mathbf{I}+\mathbf{M})\boldsymbol{V}^{n+1} = \boldsymbol{V}^n.$$

Subtracting gives

$$(\mathbf{I}+\mathbf{M})(\boldsymbol{V}^{n+1} - \boldsymbol{W}^{n+1}) = \boldsymbol{V}^n - \boldsymbol{W}^n.$$

Because $\mathbf{I}+\mathbf{M}$ is an M-matrix, its inverse is componentwise nonnegative, so

$$\boldsymbol{V}^{n+1} - \boldsymbol{W}^{n+1} = (\mathbf{I}+\mathbf{M})^{-1}(\boldsymbol{V}^n - \boldsymbol{W}^n) \geq \mathbf{0}.$$

□

This discrete arbitrage inequality says that a portfolio ordering at one step is not reversed by the scheme at the next step.

After selecting a model class, finite-difference and Monte Carlo methods can be used to price options, hedge options, and report risk metrics such as VaR and CVaR. The key modeling task is choosing a class consistent with observed market prices. Implied volatility data for S&P 500 options provide a typical smile pattern that constant-volatility Black–Scholes cannot reproduce, so richer structures such as local-volatility, stochastic-volatility, and jump-diffusion models must be calibrated.

Calibration Problem

Let observed market prices be

$$V_0^{\text{mkt}}(K_j, T_j), \quad j = 1, \ldots, m,$$

and let Ω denote the admissible model class. For $H \in \Omega$, write model prices as $V_0(K, T; H)$. Calibration can be posed as a zero residual system

$$V_0(K_j, T_j; H) - V_0^{\text{mkt}}(K_j, T_j) = 0, \quad j = 1, \ldots, m,$$

or as nonlinear least squares

$$\min_{H\in\Omega}\sum_{j=1}^{m}\Big(V_0(K_j,T_j;H)-V_0^{\mathrm{mkt}}(K_j,T_j)\Big)^2,$$

possibly with additional constraints on H. In practical calibration workflows, the least-squares form is usually preferred.

For the Merton jump diffusion model, one parameter vector is

$$H=\{\sigma,\lambda_J,\mu_J,\sigma_J\}.$$

For a smooth positive parametric local volatility, one simple example is

$$\sigma(S,t;\boldsymbol{x})=\exp\left(x_1+x_2S+x_3S^2\right),\quad \boldsymbol{x}\in\mathbb{R}^3,\ 3\le m.$$

Because $V_0(K_j,T_j;H)$ depends nonlinearly on parameters, the least-squares problem is nontrivial, and the zero residual and least-squares formulations are not equivalent when the model class cannot fit all quotes exactly.

A calibrated model is **marked to market** when parameter values are chosen so that model prices are as consistent as possible with current liquid market quotes. That consistency is essential when the same model is used to price less liquid instruments.

Practical calibration raises several technical questions: the number of local minimizers may be large, method choice can materially affect outcomes, approximate minimizers must be assessed for quality rather than accepted blindly, and large residuals can come from optimizer failure or model misspecification, not only from one source.

For local volatility model calibration, define the residual vector

$$\boldsymbol{F}(\boldsymbol{x})=\begin{bmatrix}V_0(K_1,T_1;\sigma(S,t;\boldsymbol{x}))-V_0^{\mathrm{mkt}}(K_1,T_1)\\ \vdots\\ V_0(K_m,T_m;\sigma(S,t;\boldsymbol{x}))-V_0^{\mathrm{mkt}}(K_m,T_m)\end{bmatrix}\in\mathbb{R}^m,$$

and objective

$$f(\boldsymbol{x})=\frac{1}{2}\|\boldsymbol{F}(\boldsymbol{x})\|_2^2\qquad\text{and}\qquad\min_{\boldsymbol{x}\in\mathbb{R}^n}f(\boldsymbol{x}).$$

The calibration problem is thus a **nonlinear least squares problem**, which is a special kind of

unconstrained nonlinear optimization problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x})$$

where f is continuously differentiable and has the least-squares structure $f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{F}(\boldsymbol{x})\|_2^2$ for some $\boldsymbol{F} : \mathbb{R}^n \to \mathbb{R}^m$.

Definition 22.2 A point $\boldsymbol{x}^*$ is a **local minimizer** if there exists $\delta > 0$ such that

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}) \quad \text{for all } \|\boldsymbol{x} - \boldsymbol{x}^*\| \leq \delta,$$

and it is a **global minimizer** if

$$f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}) \quad \text{for all } \boldsymbol{x}.$$

Proposition 22.3

1. (**First-order necessary condition**) If f is continuously differentiable and $\boldsymbol{x}^*$ is a local minimizer, then $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$.
2. (**Second-order necessary condition**) If f is twice continuously differentiable and $\boldsymbol{x}^*$ is a local minimizer, then $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$ and $\nabla^2 f(\boldsymbol{x}^*)$ is positive semidefinite.
3. (**Second-order sufficient condition**) If f is twice continuously differentiable and $\nabla f(\boldsymbol{x}^*) = \boldsymbol{0}$ and $\nabla^2 f(\boldsymbol{x}^*)$ is positive definite, then $\boldsymbol{x}^*$ is a local minimizer.

Let

$$\mathbf{J}(\boldsymbol{x}) = \begin{bmatrix} \dfrac{\partial F_1}{\partial x_1} & \cdots & \dfrac{\partial F_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial F_m}{\partial x_1} & \cdots & \dfrac{\partial F_m}{\partial x_n} \end{bmatrix} \in \mathbb{R}^{m\times n}$$

be the Jacobian of $\boldsymbol{F}(\boldsymbol{x})$. Then the gradient and Hessian of f can be expressed in terms of $\boldsymbol{F}$ and $\mathbf{J}$ as

$$\nabla f(\boldsymbol{x}) = \mathbf{J}(\boldsymbol{x})^T \boldsymbol{F}(\boldsymbol{x}),$$

and

$$\nabla^2 f(\boldsymbol{x}) = \mathbf{J}(\boldsymbol{x})^T \mathbf{J}(\boldsymbol{x}) + \sum_{i=1}^{m} \nabla^2 F_i(\boldsymbol{x})\, F_i(\boldsymbol{x}).$$

Computational difficulty is substantial: each model value often requires a PDE solve, gradients require differentiating each model price with respect to each parameter, second derivatives are usually hard to compute accurately, and multiple local minimizers are common. Large calibration errors therefore do not automatically reveal whether the issue is algorithmic or structural in the model family.

Lecture 23 — Wednesday, March 25, 2015

Newton and Gauss–Newton Steps

Continue with the nonlinear least-squares calibration objective

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) \qquad \text{and} \qquad f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{F}(\boldsymbol{x})\|_2^2,$$

where

$$\boldsymbol{F}(\boldsymbol{x}) = \begin{bmatrix} V_0(K_1, T_1; \sigma(S,t;\boldsymbol{x})) - V_0^{\text{mkt}}(K_1,T_1) \\ \vdots \\ V_0(K_m, T_m; \sigma(S,t;\boldsymbol{x})) - V_0^{\text{mkt}}(K_m,T_m) \end{bmatrix} \in \mathbb{R}^m.$$

Each model value is a complicated nonlinear function of parameters, so iterative optimization is required.

The gradient is

$$\nabla f(\boldsymbol{x}) = \mathbf{J}(\boldsymbol{x})^T \boldsymbol{F}(\boldsymbol{x}),$$

where the full $m \times n$ Jacobian matrix is

$$\mathbf{J}(\boldsymbol{x}) = \begin{bmatrix} \dfrac{\partial F_1}{\partial x_1} & \dfrac{\partial F_1}{\partial x_2} & \cdots & \dfrac{\partial F_1}{\partial x_n} \\ \dfrac{\partial F_2}{\partial x_1} & \dfrac{\partial F_2}{\partial x_2} & \cdots & \dfrac{\partial F_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial F_m}{\partial x_1} & \dfrac{\partial F_m}{\partial x_2} & \cdots & \dfrac{\partial F_m}{\partial x_n} \end{bmatrix}.$$

The Hessian is

$$\nabla^2 f(\boldsymbol{x}) = \mathbf{J}(\boldsymbol{x})^T \mathbf{J}(\boldsymbol{x}) + \sum_{i=1}^{m} \nabla^2 F_i(\boldsymbol{x}) F_i(\boldsymbol{x}).$$

Computational difficulty arises immediately because each model value may require a PDE solve, gradient computation requires sensitivities of each model value with respect to each parameter, and second derivatives are often expensive and inaccurate.

Let $\boldsymbol{x}_c$ be the current iterate. With

$$\mathbf{S}(\boldsymbol{x}) = \sum_{i=1}^{m} \nabla^2 F_i(\boldsymbol{x}) F_i(\boldsymbol{x}),$$

a Newton step is

$$\boldsymbol{x}^+ = \boldsymbol{x}_c - \left(\mathbf{J}(\boldsymbol{x}_c)^T \mathbf{J}(\boldsymbol{x}_c) + \mathbf{S}(\boldsymbol{x}_c)\right)^{-1} \mathbf{J}(\boldsymbol{x}_c)^T \boldsymbol{F}(\boldsymbol{x}_c).$$

A Gauss–Newton step uses the affine model

$$\boldsymbol{M}_c(\boldsymbol{x}) = \boldsymbol{F}(\boldsymbol{x}_c) + \mathbf{J}(\boldsymbol{x}_c)(\boldsymbol{x} - \boldsymbol{x}_c)$$

and solves

$$\min_{\boldsymbol{x}} \frac{1}{2} \|\boldsymbol{F}(\boldsymbol{x}_c) + \mathbf{J}(\boldsymbol{x}_c)(\boldsymbol{x} - \boldsymbol{x}_c)\|_2^2,$$

which yields

$$\boldsymbol{x}^+ = \boldsymbol{x}_c - \left(\mathbf{J}(\boldsymbol{x}_c)^T \mathbf{J}(\boldsymbol{x}_c)\right)^{-1} \mathbf{J}(\boldsymbol{x}_c)^T \boldsymbol{F}(\boldsymbol{x}_c).$$

This approximation omits $\mathbf{S}(\boldsymbol{x}_c)$ from the Hessian model.

Proposition 23.1 Suppose $\boldsymbol{F}$ is twice continuously differentiable near a solution $\boldsymbol{x}^*$, $\boldsymbol{F}(\boldsymbol{x}^*) = \mathbf{0}$, $\mathbf{J}(\boldsymbol{x}^*)$ has full column rank, and $\mathbf{J}$ is locally Lipschitz continuous. Then Gauss–Newton is locally quadratically convergent.

Gauss–Newton requires only residuals and Jacobians and is efficient on mildly nonlinear problems with small residuals, but it can converge slowly when residuals are large, can fail on very nonlinear problems, is not well defined if $\mathbf{J}(\boldsymbol{x}_c)$ is rank deficient, and does not guarantee global convergence.

The **Levenberg–Marquardt method** computes $\boldsymbol{x}^+$ from a trust-region subproblem

$$\min_{\boldsymbol{x}^+} \left\|\boldsymbol{F}(\boldsymbol{x}_c) + \mathbf{J}(\boldsymbol{x}_c)(\boldsymbol{x}^+ - \boldsymbol{x}_c)\right\|_2^2$$

subject to

$$\|\boldsymbol{x}^+ - \boldsymbol{x}_c\|_2 \le \Delta_c,$$

and, whenever the displayed inverse exists, the solution has form

$$\boldsymbol{x}^{+} = \boldsymbol{x}_c - \left(\mathbf{J}(\boldsymbol{x}_c)^T\mathbf{J}(\boldsymbol{x}_c) + \mu_c\mathbf{I}\right)^{-1}\mathbf{J}(\boldsymbol{x}_c)^T\boldsymbol{F}(\boldsymbol{x}_c)$$

for a suitable damping parameter $\mu_c \geq 0$ chosen to satisfy the trust-region constraint. Local convergence properties are similar to Gauss–Newton, and practical robustness is often better, although performance can still degrade in strongly nonlinear regions with large residuals.

When exact Jacobians are unavailable, finite differences are used. For scalar $g(z)$,

$$g'(z) = \frac{g(z+\delta z) - g(z)}{\delta z} + O(\delta z).$$

If $g(z)$ is already computed, one additional evaluation at $z + \delta z$ gives the approximation. The increment must balance truncation and roundoff error. If machine precision has t binary digits, then $\texttt{eps} = 2^{-t}$ and unit roundoff is $\frac{1}{2}\texttt{eps}$.

For

$$v(\alpha) = \mathbf{A}(\alpha)^{-1}\boldsymbol{b}, \qquad \mathbf{A}(\alpha) = \begin{bmatrix} \alpha & 0.9 \\ 0.9 & \alpha \end{bmatrix}, \qquad \text{and} \qquad \boldsymbol{b} = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$

the finite-difference approximation of $v_1'(\alpha)$ at $\alpha = 1$ behaves as follows:

δz	Approximation
1.0×10^{-1}	$-2.631578947368418 \times 10^{-1}$
1.0×10^{-2}	$-2.755580049599526 \times 10^{-1}$
1.0×10^{-3}	$-2.768625930948954 \times 10^{-1}$
1.0×10^{-4}	$-2.769937316215998 \times 10^{-1}$
1.0×10^{-5}	$-2.770068522339741 \times 10^{-1}$
1.0×10^{-6}	$-2.770081646286116 \times 10^{-1}$
1.0×10^{-7}	$-2.770082929703932 \times 10^{-1}$
1.0×10^{-8}	$-2.770083051828465 \times 10^{-1}$
1.0×10^{-9}	$-2.770081941605440 \times 10^{-1}$
1.0×10^{-10}	$-2.770084162051489 \times 10^{-1}$
1.0×10^{-11}	$-2.769673379532378 \times 10^{-1}$
1.0×10^{-12}	$-2.772226892489016 \times 10^{-1}$
1.0×10^{-13}	$-2.764455331316640 \times 10^{-1}$
1.0×10^{-14}	$-2.331468351712829 \times 10^{-1}$
1.0×10^{-15}	$1.110223024625156 \times 10^{-1}$
1.0×10^{-16}	0

For a unit-scaled variable, a standard forward-difference choice is

$$\delta z = \sqrt{\texttt{eps}}.$$

More generally, the perturbation should be scaled to the magnitude and units of z, for example $\delta z = \sqrt{\texttt{eps}}\max(1, |z|)$.

Assuming $\boldsymbol{F}(\boldsymbol{x}_c)$ is known, Jacobian entries can be approximated by

$$\frac{\partial F_j}{\partial x_i} \approx \frac{F_j(\boldsymbol{x}_c + \delta \boldsymbol{e}_i) - F_j(\boldsymbol{x}_c)}{\delta},$$

where $\boldsymbol{e}_i$ is the ith coordinate vector. Computing the full Jacobian therefore requires parameter-by-parameter perturbations.

Multiple local minimizers are common in calibration, and a good starting point can materially improve the chance of reaching a useful minimizer. Gauss–Newton and Levenberg–Marquardt use Jacobian information and only part of the Hessian structure, so progress may still be slow in highly nonlinear regions with large residuals. Alternative methods use exact or approximate Hessian information, and automatic differentiation can be used, but the computational cost increases.

MATLAB `lsqnonlin` solves nonlinear least-squares problems in forms such as

```
x = lsqnonlin(fun,x0,lb,ub,options),
```

```
[x,resnorm,residual,exitflag] = lsqnonlin(...)
```

and

```
[x,resnorm,residual,exitflag,output] = lsqnonlin(...).
```

Typical option settings are

```
options = optimoptions(@lsqnonlin, ...
    'Algorithm','levenberg-marquardt', ...
    'SpecifyObjectiveGradient',true, ...
    'Display','iter');
```

and a function interface may be written as

```
function [F,J] = myfun(x)
```

```
    F = ...
    if nargout > 1
        J = ...
    end
end
```

If model assumptions are inadequate, residuals can remain large even when the solver terminates normally. Deciding whether a model class is rich enough to fit market prices, selecting an appropriate local-volatility parametrization, and controlling overfitting remain fundamental unresolved issues in practical calibration.

Lecture 24 — Friday, March 27, 2015

Portfolio Selection and CVaR Optimization

Portfolio selection determines optimal allocations across assets. Let R_i denote the random return of asset i, for $i = 1, \ldots, n$. The core statistical quantities are expected returns $\mu_i = \mathbb{E}[R_i]$, variances $\sigma_i^2 = \mathrm{Var}(R_i)$, and correlations $-1 \le \rho_{ij} \le 1$.

Given samples $\{R_{i,t} : t = 1, \ldots, T\}$, the usual empirical estimates are

$$\bar{\mu}_i = \frac{1}{T} \sum_{t=1}^{T} R_{i,t},$$

$$\bar{\sigma}_i^2 = \frac{1}{T-1} \sum_{t=1}^{T} \left(R_{i,t} - \bar{\mu}_i\right)^2,$$

$$\bar{\rho}_{i,j} = \frac{\sum_{t=1}^{T} (R_{i,t} - \bar{\mu}_i)(R_{j,t} - \bar{\mu}_j)}{\sqrt{\sum_{t=1}^{T} (R_{i,t} - \bar{\mu}_i)^2} \sqrt{\sum_{t=1}^{T} (R_{j,t} - \bar{\mu}_j)^2}}.$$

Recall that the covariance matrix of $R_1, \ldots, R_n$ is

$$\mathbf{Q}_{ij} = \rho_{ij} \sigma_i \sigma_j.$$

Equivalently,

$$\mathbf{Q} = \begin{bmatrix} \sigma_1^2 & \sigma_1\sigma_2\rho_{1,2} & \cdots & \sigma_1\sigma_n\rho_{1,n} \\ \sigma_2\sigma_1\rho_{2,1} & \sigma_2^2 & \cdots & \sigma_2\sigma_n\rho_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_n\sigma_1\rho_{n,1} & \sigma_n\sigma_2\rho_{n,2} & \cdots & \sigma_n^2 \end{bmatrix}.$$

For proportional holdings $\boldsymbol{x} \in \mathbb{R}^n$, the portfolio return is $\sum_{i=1}^n x_i R_i$. Its mean and variance are

$$\mathbb{E}\left[\sum_{i=1}^n x_i R_i\right] = \sum_{i=1}^n x_i \mu_i \qquad \text{and} \qquad \operatorname{Var}\left(\sum_{i=1}^n x_i R_i\right) = \boldsymbol{x}^T \mathbf{Q} \boldsymbol{x}.$$

The matrix $\mathbf{Q}$ is positive semidefinite.

Harry Markowitz's portfolio theory characterizes portfolios by mean return and risk measured by standard deviation. In the (standard deviation, mean) plane, each feasible portfolio is represented by a point.

Definition 24.1 Given a target expected return μ_0, the mean–variance optimization problem is

$$\min_{\boldsymbol{x} \in \mathbb{R}^n} \boldsymbol{x}^T \mathbf{Q} \boldsymbol{x}$$

subject to

$$\mathbf{1}^T \boldsymbol{x} = 1, \qquad \boldsymbol{\mu}^T \boldsymbol{x} \geq \mu_0, \qquad \text{and} \qquad \boldsymbol{x} \geq \mathbf{0},$$

where $\mathbf{1}^T = [1, 1, \ldots, 1]$.

The portfolio with minimum risk for a fixed expected return is called **efficient**. If the feasible set is nonempty and $\mathbf{Q}$ is positive definite, this is a strictly convex quadratic program with a unique solution. The set of efficient portfolios forms the **efficient frontier** in mean-standard-deviation coordinates.

When returns are nonnormal, especially for hedge fund and derivative-heavy portfolios with option-like behavior, standard deviation is less suitable for tail risk. In that setting, VaR and CVaR are used instead.

Let $L(\boldsymbol{x}, S)$ denote portfolio loss under scenario S, and let $p(S)$ be the joint density of risk factors. Define the cumulative distribution

$$\Psi(\boldsymbol{x}, \alpha) = \int_{L(\boldsymbol{x},S) \leq \alpha} p(S)\, \mathrm{d}S.$$

For continuous loss distributions and confidence level β, the VaR $\alpha_\beta(\boldsymbol{x})$ satisfies

$$\Psi(\boldsymbol{x}, \alpha_\beta(\boldsymbol{x})) = \beta.$$

For general distributions,

$$\alpha_\beta(\boldsymbol{x}) = \inf\{\alpha : \Psi(\boldsymbol{x}, \alpha) \geq \beta\}.$$

Define

$$(L(\boldsymbol{x}, S) - \alpha)^+ = \max\left(L(\boldsymbol{x}, S) - \alpha, 0\right).$$

For a confidence level β and an integrable loss, the generalized expected shortfall is represented by

$$\phi_\beta(\boldsymbol{x}) = \min_\alpha \left[\alpha + (1-\beta)^{-1}\mathbb{E}\left[(L(\boldsymbol{x}, S) - \alpha)^+\right]\right].$$

Formally, let

$$F_\beta(\boldsymbol{x}, \alpha) = \alpha + (1-\beta)^{-1}\mathbb{E}\left[(L(\boldsymbol{x}, S) - \alpha)^+\right].$$

Then

$$\mathrm{CVaR}_\beta(\boldsymbol{x}) = \min_\alpha F_\beta(\boldsymbol{x}, \alpha),$$

and the lower β-quantile $\mathrm{VaR}_\beta(\boldsymbol{x})$ is a minimizer of $F_\beta(\boldsymbol{x}, \alpha)$ in α. More generally, when the distribution has an interval of β-quantiles, every point in that interval is a minimizer. When the loss distribution is continuous, this agrees with the conditional tail mean defined earlier; the optimization formula also handles atoms at the quantile correctly.

The optimization

$$\min_{\boldsymbol{x}} \phi_\beta(\boldsymbol{x}) = \min_{\boldsymbol{x}, \alpha} F_\beta(\boldsymbol{x}, \alpha)$$

is convex whenever $L(\boldsymbol{x}, S)$ is convex in $\boldsymbol{x}$ for every scenario, so every local minimizer is then global. For asset portfolios with return vector $\boldsymbol{R} \in \mathbb{R}^n$, the loss is linear: $L(\boldsymbol{x}, S) = \boldsymbol{L}^T\boldsymbol{x}$ with $\boldsymbol{L} = -\boldsymbol{R}$.

Assume independent loss samples $\{\boldsymbol{L}_i\}_{i=1}^M$. A sample approximation is

$$\min_{\boldsymbol{x}, \alpha} \left[\alpha + \frac{1}{M(1-\beta)} \sum_{i=1}^{M} \left(\boldsymbol{L}_i^T\boldsymbol{x} - \alpha\right)^+\right].$$

Introduce auxiliary variables $y_i = \left(\boldsymbol{L}_i^T \boldsymbol{x} - \alpha\right)^+$ and solve

$$\min_{\boldsymbol{x},\boldsymbol{y},\alpha} \left[\alpha + \frac{1}{M(1-\beta)} \sum_{i=1}^{M} y_i \right]$$

subject to

$$y_i \geq 0 \qquad \text{and} \qquad y_i \geq \boldsymbol{L}_i^T \boldsymbol{x} - \alpha, \quad i = 1, \ldots, M.$$

This linear program has $M + n + 1$ variables and $2M$ constraints.

Additional portfolio constraints are often imposed:

$$0 \leq x_i \leq 1, \quad i = 1, \ldots, n,$$

$$\sum_{i=1}^{n} x_i \leq 1.$$

A common formulation instead maximizes expected return subject to a CVaR bound,

$$\max_{x_1,\ldots,x_n} \mathbb{E}\left[\sum_{i=1}^{n} r_i x_i \right]$$

with the above constraints and

$$\mathrm{CVaR}_\beta\left(-\boldsymbol{r}^T \boldsymbol{x}\right) \leq \rho.$$

With M independent samples of asset returns, the sample problem becomes

$$\max_{\boldsymbol{x},\boldsymbol{y},\alpha} \sum_{i=1}^{n} \mathbb{E}[r_i] x_i$$

subject to

$$\boldsymbol{0} \leq \boldsymbol{x} \leq \boldsymbol{1} \qquad \text{and} \qquad \sum_{i=1}^{n} x_i \leq 1,$$

$$\alpha + \frac{1}{M(1-\beta)} \sum_{j=1}^{M} y_j \leq \rho,$$

$$y_j \geq -\boldsymbol{x}^T (\boldsymbol{r})_j - \alpha \qquad \text{and} \qquad y_j \geq 0, \quad j = 1, \ldots, M,$$

where $(\boldsymbol{r})_j$ is the jth return sample and $\mathbb{E}[r_i]$ is estimated from the samples.

Reliable LP software such as MOSEK, CPLEX, and MATLAB's `linprog` can solve these models.

In canonical form,

$$\min_{\boldsymbol{z}} \boldsymbol{f}^T \boldsymbol{z}$$

subject to

$$\mathbf{A}\boldsymbol{z} \leq \boldsymbol{b}, \qquad \mathbf{A}_{\text{eq}}\boldsymbol{z} = \boldsymbol{b}_{\text{eq}}, \qquad \text{and} \qquad \boldsymbol{l} \leq \boldsymbol{z} \leq \boldsymbol{u}.$$

A typical MATLAB setting is `options = optimset('Display','iter','MaxIter',200)`.

Lecture 25 — Monday, March 30, 2015

Consider a portfolio of derivatives whose instrument values are collected in

$$\boldsymbol{V}(S,t) = \begin{bmatrix} V_1(S,t) & \cdots & V_n(S,t) \end{bmatrix}^T,$$

where S denotes the underlying risk factors. A stock can be represented as a component of this vector, for example by writing $V_j(S,t) = S_A$ for stock A.

Fix a horizon $\bar{t}$, for example $\bar{t} = 1/12$, and define

$$\delta\boldsymbol{V} = \boldsymbol{V}(S^{\bar{t}}, \bar{t}) - \boldsymbol{V}(S^0, 0).$$

For portfolio weights $\boldsymbol{x} \in \mathbb{R}^n$, one loss model is

$$L(\boldsymbol{x}, S) = -\boldsymbol{x}^T(\delta\boldsymbol{V}).$$

Assume M independent samples of price change losses are available. Define the sampled criterion

$$F_M(\boldsymbol{x}, \alpha) = \alpha + \frac{1}{M(1-\beta)} \sum_{i=1}^{M} \left(-(\delta\boldsymbol{V})_i^T \boldsymbol{x} - \alpha\right)^+$$

and minimize it over the feasible $(\boldsymbol{x}, \alpha)$. For each fixed $(\boldsymbol{x}, \alpha)$ with integrable positive-part loss, the law of large numbers gives

$$F_M(\boldsymbol{x}, \alpha) \longrightarrow F(\boldsymbol{x}, \alpha) := \alpha + (1-\beta)^{-1}\mathbb{E}\left[\left(-\delta\boldsymbol{V}^T\boldsymbol{x} - \alpha\right)^+\right]$$

almost surely as $M \to \infty$. Convergence of the optimized values or optimizers requires additional compactness and uniform-convergence conditions. Introducing auxiliary variables

$$y_i = \left(-(\delta\boldsymbol{V})_i^T \boldsymbol{x} - \alpha\right)^+,$$

gives the linear program

$$\min_{\boldsymbol{x},\boldsymbol{y},\alpha} \left[\alpha + \frac{1}{M(1-\beta)} \sum_{i=1}^{M} y_i \right]$$

subject to

$$y_i \geq 0 \qquad \text{and} \qquad y_i \geq -(\delta \boldsymbol{V})_i^T \boldsymbol{x} - \alpha, \quad i = 1, \ldots, M.$$

The sampled model has $M + n + 1$ variables and $2M$ inequality constraints.

Two common linear constraints are a budget condition and a target return condition. For example, to encode a \$1 initial investment and a target return $\bar{r}$, write

$$\boldsymbol{x}^T \boldsymbol{V}(S^0, 0) = 1,$$

$$\delta \bar{\boldsymbol{V}}^T \boldsymbol{x} = \bar{r} \qquad \text{and} \qquad \delta \bar{\boldsymbol{V}} = \frac{1}{M} \sum_{i=1}^{M} (\delta \boldsymbol{V})_i.$$

The inequality system can be written as

$$\mathbf{A}\boldsymbol{z} \leq \boldsymbol{b},$$

If the constraints $y_i \geq 0$ are supplied as variable lower bounds, $\mathbf{A}$ has size $M \times (M + n + 1)$ and contains one dense $M \times (n + 1)$ block plus one diagonal block for the auxiliary variables. If all bounds are written as inequality rows, the full system has $2M$ rows. For the coupling matrix, the sparsity ratio is

$$1 - \frac{\text{nnz}(\mathbf{A})}{\text{prod}(\text{size}(\mathbf{A}))} = 1 - \frac{M(n+1) + M}{M(M+n+1)},$$

which is close to one when $M \gg n$, where $\text{nnz}(\mathbf{A})$ is the number of nonzero entries and $\text{prod}(\text{size}(\mathbf{A}))$ is the total number of entries. Hence the LP is very sparse, and exploiting this structure can materially improve solution times.

In MATLAB, sparse matrices for this LP can be assembled directly from triplets or preallocated and filled column by column:

```
A = sparse(i,j,s,m,n);
A = sparse(m,n);
A = spalloc(m,n,nnz);
A(:,j) = ...;
```

Path-Dependent Options

Asian options are strongly path dependent because the payoff depends on an average of underlying prices over time. This averaging typically reduces payoff volatility, so Asian options are often cheaper than vanilla options and can be easier to hedge in applications such as gradually received foreign currency cash flows; for example, a company that expects to receive some foreign fund spread evenly over the next year may want the average exchange rate to stay above some level, which is a natural use case for an Asian put option.

If A denotes the average price, two common call payoffs are

$$\max(S_T - A_T, 0) \quad \text{(floating strike)},$$

and

$$\max(A_T - K, 0) \quad \text{(fixed strike)}.$$

Arithmetic averaging can be continuous or discrete:

$$A(t) = \frac{1}{t}\int_0^t S(\tau)\,\mathrm{d}\tau \qquad \text{and} \qquad A_i = \frac{1}{i}\sum_{k=1}^{i} S(t_k^s).$$

Writing

$$I(t) = \int_0^t S(\tau)\,\mathrm{d}\tau,$$

the arithmetic average is $A(t) = I(t)/t$. For geometric averaging, with

$$I(t) = \int_0^t \log(S(\tau))\,\mathrm{d}\tau,$$

the average is $A(t) = \exp(I(t)/t)$.

A continuously sampled arithmetic-average-strike call has payoff

$$\max(S_T - A_T, 0) \qquad \text{and} \qquad A_T = \frac{1}{T}\int_0^T S(\tau)\,\mathrm{d}\tau.$$

In practice, averaging dates are discrete. Let

$$0 = t_1^s < t_2^s < \cdots < t_\ell^s \le T.$$

The running average is piecewise constant between sampling dates:

$$A(t) \equiv A(t_{i-1}^s), \quad t \in [t_{i-1}^s, t_i^s).$$

It evolves by

$$A(t_1^s) = S_0,$$

$$A(t_i^s) = \frac{1}{i}\sum_{k=1}^{i} S(t_k^s) = \frac{i-1}{i}A(t_{i-1}^s) + \frac{1}{i}S(t_i^s), \quad i > 1.$$

Hence the option value is $V(S, A, t)$ and the initial price is $V(S_0, S_0, 0)$. On each interval (t_{i-1}^s, t_i^s), the average is fixed and satisfies $\mathrm{d}A = 0$, so only S evolves randomly. Therefore, for each fixed A, the value solves the one-dimensional Black–Scholes equation

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r-q)S\frac{\partial V}{\partial S} - rV = 0, \quad t \in (t_{i-1}^s, t_i^s).$$

At a sampling date, the average jumps by

$$A(t_i^s) = \frac{i-1}{i}A(t_{i-1}^s) + \frac{1}{i}S(t_i^s).$$

No-arbitrage continuity of realized option values implies the **jump condition**

$$V\big(S, A(t_{i-1}^s), t_i^{s-}\big) = V\left(S, \frac{i-1}{i}A(t_{i-1}^s) + \frac{1}{i}S, t_i^s\right),$$

which is analogous to the case of discrete dividends.

To price the option, use the grid

$$\{(S_i, A_k, t_j)\}_{i=1,\ldots,N+1;\, k=1,\ldots,L+1;\, j=1,\ldots,M+1} \approx [0, S_{\max}] \times [0, A_{\max}] \times [0, T],$$

with each sampling date aligned to a grid time for simplicity, and define

$$V_j^{i,k} \approx V(S_i, A_k, t_j).$$

Along the S direction, the upper boundary uses

$$\frac{\partial^2 V}{\partial S^2} = 0 \quad \text{as } S \to \infty,$$

which discretizes to

$$V_{j-1}^{N-1,k} - 2V_{j-1}^{N,k} + V_{j-1}^{N+1,k} = 0,$$

or equivalently

$$V_{j-1}^{N+1,k} = 2V_{j-1}^{N,k} - V_{j-1}^{N-1,k}.$$

The lower boundary at $S = 0$ depends on contract type.

Backward valuation alternates between PDE solves and jump updates. The procedure can be written explicitly as follows.

Algorithm 3: Pricing Asian options with discrete updates

Compute $V_{M+1}^{i,k}$ for each $i = 1, \ldots, N+1$ and $k = 1, \ldots, L+1$
for $j = M+1, M, \ldots, 2$ **do**
  if $t_j \in \{t_1^s, \ldots, t_\ell^s\}$ **then**
    Update $\{V_j^{i,k}\}_{i=1,\ldots,N+1;\, k=1,\ldots,L+1}$ to enforce the jump condition
  for $k = 1, \ldots, L+1$ **do**
    Compute $\{V_{j-1}^{i,k}\}_{i=1,\ldots,N+1}$ from $\{V_j^{i,k}\}_{i=1,\ldots,N+1}$ using the finite-difference equation or LCP

Algorithm 4: Jump condition update at a sampling date

Assume $t_j = t_{k_0}^s$ with $1 < k_0 \le \ell$
for $k = 1, \ldots, L+1$ **do**
  for $i = 1, \ldots, N+1$ **do**
    Enforce $V(S_i, A_k, t_j^-) \leftarrow V\left(S_i, \dfrac{k_0 - 1}{k_0} A_k + \dfrac{1}{k_0} S_i, t_j\right)$
Apply interpolation or extrapolation in A when the updated average is off grid

The two-dimensional valuation problem is therefore handled through a sequence of one-dimensional PDE solves in S.

Lecture 26 — Wednesday, April 01, 2015

Barrier Options via PDEs

A barrier contract includes a preset price level that triggers a contract feature when reached. If the barrier is above the initial spot, it is an **up barrier** with $S_u > S_0$. If the barrier is below the initial spot, it is a **down barrier** with $S_d < S_0$.

In a **knock-out barrier contract**, the standard terminal payoff is paid only if the barrier is never reached; once reached, the contract is knocked out. In a **knock-in barrier contract**, the standard terminal payoff is paid only if the barrier is reached; the contract is knocked in at that event.

For a single barrier, an up-and-out European call pays $\max(S_T - K, 0)$ when S_t never reaches S_u on $0 \le t \le T$, and is worthless otherwise. An up-and-in European put pays $\max(K - S_T, 0)$ only when S_t reaches S_u during $0 \le t \le T$.

Barrier options are usually cheaper than corresponding vanilla contracts, and barrier features can be combined with many payoff structures. Common variants include rebates after knock-out, double barriers with both S_u and S_d that trigger when either is reached, repeated-hitting barriers that trigger after both are reached, and rainbow barriers where a second underlying affects either payoff or trigger conditions.

For pricing, consider an up-and-out call. The value is highly sensitive to volatility because volatility controls the probability of barrier hits. Assume

$$\frac{\mathrm{d}S_t}{S_t} = (\mu - q)\,\mathrm{d}t + \sigma(S,t)\,\mathrm{d}W_t.$$

Define

$$J_t = \max_{0 \le \tau \le t} S_\tau.$$

If $J_t \ge S_u$, the up-and-out option value is zero. If $J_t < S_u$, denote the value by $V^{\text{out}}(S_t, t)$. Then

$$W^{\text{out}}(S_t, J_t, t) = \begin{cases} 0, & J_t \ge S_u, \\ V^{\text{out}}(S_t, t), & J_t < S_u. \end{cases}$$

In particular, the initial value is $V^{\text{out}}(S_0, 0)$.

Before the barrier has been reached, no-arbitrage hedging implies

$$\frac{\partial V^{\text{out}}}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V^{\text{out}}}{\partial S^2} + (r-q)S\frac{\partial V^{\text{out}}}{\partial S} - rV^{\text{out}} = 0, \quad 0 < S < S_u, \quad 0 \le t < T.$$

Equivalently,

$$\frac{\partial V^{\text{out}}}{\partial t} + \mathcal{L}_{\text{BS}}[V^{\text{out}}] = 0,$$

with

$$\mathcal{L}_{\text{BS}}[V] = \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r-q)S\frac{\partial V}{\partial S} - rV.$$

The associated barrier PDE problem requires boundary and terminal conditions for $V^{\text{out}}(S_u, t)$, $V^{\text{out}}(0, t)$, and $V^{\text{out}}(S, T)$. When $K < S_u$, a jump occurs at maturity at $S = S_u$, so nonuniform spatial discretization near the barrier is typically needed to resolve the discontinuity accurately.

Appendix: Lecture and Tutorial Index