

**DEPARTMENT OF
ACTUARIAL STUDIES
RESEARCH PAPER
SERIES**

**The multi-binomial model
and applications**

by

Tim Kyng

Research Paper No. 2005/03
July 2005

Division of Economic and Financial Studies
Macquarie University
Sydney NSW 2109 Australia

The Macquarie University *Actuarial Studies Research Papers* are written by members or affiliates of the Department of Actuarial Studies, Macquarie University. Although unrefereed, the papers are under the review and supervision of an editorial board.

Editorial Board:

Piet de Jong
Leonie Tickle

Copies of the research papers are available from the World Wide Web at:

http://www.actuary.mq.edu.au/research_papers/index.html

Views expressed in this paper are those of the author(s) and not necessarily those of the Department of Actuarial Studies.

THE MULTI-BINOMIAL MODEL AND APPLICATIONS

By Timothy Kyng, M Stats, M Ec, FIAA

Lecturer in Actuarial Studies, Macquarie University, Sydney

This paper develops a method for the valuation of multivariate contingent claims and is an extension of the well known binomial option pricing model. I refer to it as the **Multi-Binomial Model**.

Structure of the paper:

- 1) Introduction**
- 2) The problem the method is designed to address**
- 3) examples of real world financial problems / multivariate contingent claims**
- 4) analytic solutions and numerical solutions: the 1 dimensional case**
- 5) the multi-binomial method for valuation of contingent claims**
- 6) examples / numerical results**
- 7) conclusions**

1) **Introduction**

- The method presented is a numerical algorithm for the valuation of contingent claims involving multiple, correlated, normally distributed, or lognormally distributed, sources of risk.
- The method constructs a discrete approximation to a correlated multidimensional geometric Brownian motion process as the driving process which generates the amounts paid under the contingent claim being valued. It is a multidimensional “tree” approach to the valuation of contingent claims.
- Many such contingent claims are intractable when it comes to deriving or using an analytic approach.
- The methodology developed here can be applied to a wide range of option pricing and contingent claims valuation problems. In particular it is applicable to valuation of executive share option schemes and valuation of complex exotic derivatives involving more than one asset, such as executive share option schemes, “spread options” and “rainbow options”.
- This numerical method is easy to program and is efficient and practical, at least for low dimensional problems (up to about 5 assets). When the number of assets involved gets too big, it is better to use Monte Carlo Simulation.
- Various examples are included to illustrate the method.

2) The problem

We have some financial contract or instrument or cashflow with

- a payoff / cashflow made at time T , and
- the payoff $f = f(S_1(T), \dots, S_n(T))$ depends on the values of n different variables $(S_1(T), \dots, S_n(T))$.
- These variables could be the values of n different asset prices observed at time T .
- Assume these are either normally or lognormally distributed
- **What is the expected value** of the payoff / cashflow?
- **What is the present value** of this expectation?
- **can we express the pv or expected value as some function g** of the initial values $(S_1(0), \dots, S_n(0))$ and some other variable?
- **Is an analytic solution available?** If so is it easy to numerically compute it?
- **Numerical solutions (e.g. Monte Carlo Simulation)**

3) **Examples of real world financial problems include:**

- a. **exchange option** (option to exchange one risky asset for another) – option to exchange 1 oz of gold (G) for 100 oz of silver (S) .

$$Payoff = \max(S_T - G_T, 0) .$$

- b. **spread option** (option on the difference between the prices of 2 risky

assets) $Payoff = \max\left(\underbrace{S_T - G_T}_{\text{difference}} - K, 0\right)$ where K is some constant. Even

if both S and G are lognormal, then the difference will have some other distribution (not lognormal) so a BS type formula will not apply.

Alternatively instead of being an option contract S might be the assets and G the liabilities of an insurance company, so the payoff is the extent to which the surplus exceeds the level K.

- c. **3-asset rainbow option** (option on a portfolio of 3 assets R, S & V)

$$Payoff = \max(R_T + S_T + V_T - K, 0).$$

- d. **Option defined on asset S at 3 different times** (T_1, T_2, T_3) and with the

$$payoff \text{ made at time } T \text{ and is } V_T = \begin{cases} S_1 & \text{if } (S_1 > S_2 \text{ and } S_1 > S_3) \\ 0 & \text{if } (S_1 \leq S_2 \text{ or } S_1 \leq S_3) \end{cases}$$

where $S_i =$ value of S at time T_i

- e. **A 4-instrument payoff** (option on a portfolio of long and short positions in 4 assets R, E, A & L **or** the difference between a portfolio of 2 assets and 2 liabilities and some constant K.

$$Payoff = \max([R_T - E_T + A_T - L_T] - K, 0).$$

4) Analytic and Numerical Solutions: The univariate situation (n = 1)

Analytic solutions: For some types of payoff, there is an analytic formula.

- The payoff from a European call option contract over the stock S , with exercise price X , paid at time T is $payoff = \max(S - X, 0) = (S - X)^+$
- The payoff from a put option contract over the stock S , with exercise price X , paid at time T is $payoff = \max(X - S, 0) = (X - S)^+$

Standard call and put options over a single asset can be valued using the well known Black and Scholes formula. **This analytic formula applies only to options with European style exercise rights** (those that can be exercised only on the maturity date of the contract). This formula expresses the value of the option in terms of the cumulative density function of the standard normal distribution.

Black Scholes Formulae for European Options

$$c = [Se^{-yT} \times N(d_1)] - [Xe^{-rT} \times N(d_2)]$$
$$p = [Xe^{-rT} \times N(-d_2)] - [Se^{-yT} \times N(-d_1)]$$

where

- S = "spot price" (i.e. the current price) of the stock
- X = the exercise price of the option
- r = the risk free interest rate per annum
- y = dividend yield (dividend paid p.a. as a proportion of the stock price)
- T = term to maturity of the option
- σ = volatility of the stock
- c = value of a European call option
- p = value of a European put option

$$d_1 = \frac{1}{\sigma\sqrt{T}} \left[\log_e \left(\frac{S}{X} \right) + \left(r - y + \frac{1}{2}\sigma^2 \right) T \right] \text{ and } d_2 = d_1 - \sigma\sqrt{T}$$

$$N(x) = \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} dz = \Pr\{z < x \mid z \sim N(0,1)\}$$

From a statistical point of view,

1. the payoff / cashflow from an option is a random variable, and
2. the value of the option is the expected value of that cashflow, discounted to a present value.

The payoff depends on the distribution of the underlying asset (a stock). Under the Black Scholes Model, the logarithm of the price relative has a normal distribution:

It is assumed that $\log_e \left(\frac{S_T}{S_0} \right)$ follows a Brownian Motion process with drift

and has a normal distribution with

1. $mean = \left(r - \frac{1}{2} \sigma^2 \right) T$

2. $variance = \sigma^2 \times T$

These parameters define the so called “risk neutral distribution” of the asset price at any time T. This distribution is the one that makes the expected return on the asset S equal to the risk free interest rate r.

Limitations of the BS formula:

- it is not applicable to options that have early exercise features (can be exercised before the maturity date) such as American options and Bermudan options.
- it is not applicable to many of the “exotic” options (those that are different from a standard call or put option).

The one dimensional binomial model:

One way to overcome some of the limitations of the BS model is to use the binomial model instead.

In particular it allows us to

- value **american options** and
- other **options with more complex payoffs** than the standard call and put options, in the case where the payoff depends on only one asset.

This is a discrete time, discrete state space approximation to the value of an option. There are several ways to implement it. The most common is called the “Cox Ross Rubinstein” (CRR) version. We look at this first. Another way to implement it is the “equal probability” version of the model.

The inputs to the model are $(S, X, r, y, \sigma, T, m)$

Model assumes that the asset price can change only at discrete points in time.

We break up the time interval $[0, T]$ into m equally spaced time intervals of length $\Delta T = \frac{1}{m} \times T$. Time goes in steps of length ΔT .

At each time step the asset price can either go up or it can go down.

The proportionate increase or decrease is constant:

Either $S_{t+1} = u \times S_t$ or $S_{t+1} = d \times S_t$

The u and d factors are constant at each time step.

- $u = \exp(\sigma\sqrt{\Delta T})$
- $d = \exp(-\sigma\sqrt{\Delta T})$
- $\ln(u) = -\ln(d) = \sigma\sqrt{\Delta T}$

The probability of an up step is $p = (\exp(r\Delta T) - d) / (u - d)$

For a European call option:

The value of the option is $E\{\max(S_T - X, 0)\} \times e^{(-r.T)}$ which is a discounted conditional expectation:

$$value = \underbrace{e^{(-r.m.\Delta T)}}_{\text{discount factor}} \left[\sum_{i=0}^m \underbrace{\max(S_0 u^k d^{m-k} - X, 0)}_{\text{payoff}} \times \underbrace{\binom{m}{k} p^k (1-p)^{(m-k)}}_{\text{probability}} \right]$$

- this formulation of the option price as a discounted expectation shows that we can value other more complicated payoffs – just replace the payoff function shown here with some other payoff function

It can be shown that as the number of time steps m gets bigger, the value of a call option given by this formula converges to the black scholes value

Looking at the model statistically:

- the stock price at time T (= time step m) after j up jumps in price is $S_T = S_0 u^j d^{m-j}$. This takes values in the set $\{S_0 u^k d^{m-k} : k = 0, 1, \dots, m\}$

- there are $(m+1)^1 = (m+1)$ different values in this set

- The probability distribution of the asset value at time step m is

$$\Pr(S = S_0 u^k d^{m-k}) = \binom{m}{k} p^k (1-p)^{(m-k)} \text{ for } k = 0, 1, \dots, m$$

- The probability distribution of the log price relative is

$$\Pr(\ln(S_T/S_0) = m \ln(d) + k \cdot \ln(u/d)) = \binom{m}{k} p^k (1-p)^{(m-k)}$$

- $k = \frac{(\ln(S_T/S_0) - m \ln(d))}{\ln(u/d)}$ has a binomial distribution. It represents the number of “up jumps” out of m “time steps”.

- The log of the price relative has a transformed binomial distribution, it is “log binomial” instead of “log normal”. The transformation is an “affine transformation” (a linear transformation plus a constant)

MOMENT MATCHING

- The crr version of the model fixes the up and down movements (u,d) and computes the probability p from the u and d . Parameters u, d and p are chosen in such a way that the probability distribution of the price relative has the right moments. In particular, the **expected return** on the asset is the risk free rate and the **standard deviation of the log of the return** is the volatility of the asset. This is the “risk neutral distribution”
- In order for the expected return on the asset to be the risk free rate, the rate of price growth must be equal to the difference between the risk free rate and the dividend yield on the asset. The total return is comprised of the dividend yield and the capital gain return.
- The equal probability version of the model fixes the probability p to be 0.50 and computes the factors u and d so as to achieve “moment matching” to produce the “risk neutral distribution” of the asset price.

In option pricing there are various analytic and numerical methods used.

These numerical methods involve

- **computing the expected value** of the payoff
- under the so called **risk neutral distribution** and then
- discounting this expectation **at the risk free rate.**

This is true of the bs model, the binomial model and various monte carlo simulation methods for valuation of options.

5) The Multi-Binomial Method for valuing Exotic Options

For some multi-asset multi period exotic options, it is possible to derive an analytic valuation formula, in a Black-Scholes framework. These often involve the cdf of the multivariate normal distribution instead of the univariate normal distribution.

The k dimensional multivariate normal cdf is the function

$$N_k(a_1, \dots, a_k; \rho_{1,2}, \dots, \rho_{k-1,k}) \\ = \Pr\{Z_1 < a_1, \dots, Z_k < a_k \mid \text{corr}(Z_i, Z_j) = \rho_{ij}, Z_i \sim N(0,1)\}$$

- It is often quite difficult to derive an analytic formula for such a multi asset option. Sometimes it is not possible to find one.
- Even if an analytic formula exists, it may not be particularly useful because it requires numerical computation of the multivariate normal cumulative distribution function.
- There exists a good numerical approximation for the 2 dimensional normal cdf (known as drezner's approximation).
- However for higher dimensions, there is no good numerical approximation and in practice monte carlo simulation is used to evaluate it.

Examples:

1) For the exchange option, an analytic formula exists. It is called the “margrabe formula” and is really the bs formula “in disguise”. This is possible because the ratio $\frac{S_T}{G_T}$ is lognormal if both the numerator and denominator are.

2) For the spread option ($Payoff = \max\left(\underbrace{S_T - G_T}_{\text{difference}} - K, 0\right)$ where K is some constant. Even if both S and G are lognormal, then the difference will have some other distribution (not lognormal) so a BS type formula will not apply.

3) Options on portfolios such as **3-asset rainbow option** with $Payoff = \max(R_T + S_T + V_T - K, 0)$ and a **4-instrument payoff such as** $Payoff = \max([R_T - E_T + A_T - L_T] - K, 0)$. The sum or difference of lognormally distributed variables is not lognormal so a BS type formula won't apply

4) Option defined on asset S at 3 different times (T_1, T_2, T_3) and the payoff is made at time T and is

$$V_T = \begin{cases} S_1 & \text{if } (S_1 > S_2 \text{ and } S_1 > S_3) \\ 0 & \text{if } (S_1 \leq S_2 \text{ or } S_1 \leq S_3) \end{cases} \quad \text{where } S_i = \text{value of S at time } T_i$$

This one has an analytic formula

$$value = Se^{-r(T-T_1)} N_2\left(-\frac{\mu}{\sigma}\sqrt{T_2-T_1}, -\frac{\mu}{\sigma}\sqrt{T_3-T_1}, \sqrt{\frac{T_2-T_1}{T_3-T_1}}\right)$$

where $N_2(a, b, \rho) = \Pr\{Z_1 < a, Z_2 < b \mid \text{corr}(Z_1, Z_2) = \rho, Z_i \sim N(0, 1)\}$
 this involves the cdf of the 2 dimensional normal distribution

UTILITY OF MULTI -BINOMIAL MODEL

It is these types of problems which the multi-binomial model can value numerically. It is an alternative to monte carlo simulation when the dimension of the problem (the number of assets) is low (say less than 6). For higher dimensional problems, monte carlo simulation is more efficient.

This method is applicable to a wide range of option pricing and other financial problems, such as

1) Executive share option schemes with performance hurdles.

These are options where the payoff depends on

- the share price of the firm at the maturity date, and
- the absolute performance (rate of return) of the firm up to some time or set of times
- the relative performance of the firm (relative to other stocks, or to some stock market indexes) up to some time or set of times.
- other hurdles

2) spread options, rainbow options

3) asset liability studies where each of the assets and each of the liabilities are lognormally distributed

4) multi asset options with early exercise features (these are more difficult to value using monte carlo simulation)

The main result:

Assume that we have

- n assets, value at time t given by the vector: $(S_1(t), S_2(t), \dots, S_n(t))$

These are assumed to be lognormal.

- vector of returns $(\ln(S_1(t)/S_1(0)), \dots, \ln(S_n(t)/S_n(0)))$

- this returns vector follows a correlated n-dim Brownian motion with

$$\text{drift } \frac{dS_i(t)}{S_i(t)} = \mu_i \cdot dt + \sigma_i \cdot dW_i(t) : i = 1, \dots, n$$

- The mean vector of the returns is assumed to be

$$\underline{\mu} = \left(\left(r - q_1 - \frac{1}{2} \sigma_1^2 \right), \left(r - q_2 - \frac{1}{2} \sigma_2^2 \right), \dots, \left(r - q_n - \frac{1}{2} \sigma_n^2 \right) \right) T = \underline{\mu}_a \times T$$

- The covariance matrix of the returns is assumed to be

$$\underline{\Sigma} = \begin{pmatrix} \sigma_1^2 & \sigma_1 \sigma_2 \rho_{12} & \dots & \sigma_1 \sigma_n \rho_{1n} \\ \sigma_1 \sigma_2 \rho_{12} & \sigma_2^2 & \dots & \sigma_2 \sigma_n \rho_{2n} \\ \dots & \dots & \dots & \dots \\ \sigma_1 \sigma_n \rho_{1n} & \sigma_2 \sigma_n \rho_{2n} & \dots & \sigma_n^2 \end{pmatrix} T = \underline{\Sigma}_a \times T$$

Let $\underline{Y} \in R^n$ be a n-dimensional iid binomial random variable with parameters \mathbf{m} and probability $\mathbf{p} = 0.5$ so that

- the components Y_i of the vector Y are iid $Y_i \sim bin(m, 0.50)$

- mean vector $E(\underline{Y}) = \underline{\mu}_Y = \frac{m}{2} \times \underline{1} = \frac{m}{2} \begin{pmatrix} 1 \\ \dots \\ 1 \end{pmatrix} \in R^n$

- n×n covariance matrix $\underline{S}_Y = \frac{m}{4} \times \underline{I}_{n \times n}$

Then there exists a matrix $\underline{\underline{A}}_{(m)}$ and a vector $\underline{\underline{b}}_{(m)}$ such that the vector $\underline{\underline{X}}_{(m)} = \underline{\underline{A}}_{(m)} \underline{\underline{Y}} + \underline{\underline{b}}_{(m)}$ has

- dimension n
- mean vector $E(\underline{\underline{X}}) = \underline{\underline{\mu}} \times T$
- covariance matrix $\text{cov}(\underline{\underline{X}}) = T \times \underline{\underline{\Sigma}}$

The matrix A and the vector b are :

$$\underline{\underline{A}}_{(m)} = 2 \left(\sqrt{\frac{T}{m}} \right) \times \left(\sqrt{\underline{\underline{\Sigma}}} \right)$$

$$\underline{\underline{b}}_{(m)} = \underline{\underline{\mu}} - \underline{\underline{A}} \times \underline{\underline{1}}$$

Where

- $\left(\sqrt{\underline{\underline{\Sigma}}} \right)$ is a square root of the matrix $\underline{\underline{\Sigma}}$, defined to be a matrix L such that $\underline{\underline{L}} \times \underline{\underline{L}}' = \underline{\underline{\Sigma}}$. One type of square root is the cholesky square root.
- $\underline{\underline{1}} = (1, \dots, 1)$ is a vector of 1's

It can be shown that $\lim_{m \rightarrow \infty} \underline{\underline{X}}_{(m)}$ is multivariate normally distributed with the above mean vector and covariance matrix.

- The vector $\underline{\underline{X}}_{(m)}$ is an approximation to the vector of log price relatives.
- The vector $\underline{\underline{W}}_{(m)} = \exp(\underline{\underline{X}}_{(m)}) = (\exp(X_1), \dots, \exp(X_n))$ is an approximation to the vector of price relatives
- The vector $\underline{\underline{S}}_{(m)}(T) = (S_1(0) \exp(X_1), \dots, S_n(0) \exp(X_n))$ is an approximation to the vector of prices at time T. This is approximately lognormally distributed. We can write this more compactly as $\underline{\underline{S}}(T) = \underline{\underline{S}}(0) \cdot \exp(\underline{\underline{X}})$

Note that r is the risk free rate of interest and q_i is the dividend yield on asset i.

Algorithm for pricing European style option prices

For the n-dimensional, m step binomial model, we assume that:

- the parameters $r, m, T, \underline{\mu}_a, \underline{\Sigma}_a$ are known
- the initial values $(S_1(0), S_2(0), \dots, S_n(0))$ are known,
- the payoff (P) we are valuing happens at time T and is some function $P = f(S_1(T), S_2(T), \dots, S_n(T))$ of the n asset values at time T

Compute $\underline{A} = 2\sqrt{\frac{T}{m}} \times chol(\underline{\Sigma}_a)$

Compute $\underline{b} = \underline{\mu}_a T - \underline{A} \times \underline{1}$

Expectation = 0

For $i = 0$ to $(m+1)^n - 1$

 Compute n dimensional vector \underline{y}

 Compute $\underline{x} = \underline{A}\underline{y} + \underline{b}$

 Compute $\underline{w} = \exp(\underline{x})$ i.e.

$w_1 = \exp(x_1), w_2 = \exp(x_2), \dots, w_n = \exp(x_n)$

 Compute $(S_1(T), \dots, S_n(T)) = (W_1 S_1(0), \dots, W_n S_n(T))$

 Compute $Payoff = f(S_1(T), S_2(T), \dots, S_n(T))$

 Compute $\Pr\{(Y_1 \dots Y_n) = (y_1 \dots y_n)\} = \binom{m}{y_1} \times \dots \times \binom{m}{y_n} \left(\frac{1}{2}\right)^{nm}$

 expectation = expectation + Payoff × probability

Next i

price = expectation × exp(-rT)

This algorithm gives the price of the contingent claim as a discounted expected cashflow.

The vector b for arbitrage freeness:

The vector b defined above makes the distribution of x have the right mean vector and covariance matrix. However the expectation of the vector w (which represents the price relatives) will not be what it should be for “arbitrage freeness”. We require that $E(w_i) = e^{(r-q_i)T}$ so that the expected rate of return on asset i is the risk free rate.

Let \underline{e}_j be the n dimensional vector defined by $\underline{e}_{ji} = \delta_{ji} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$.

This vector has a 1 in position j, and 0 everywhere else.

Define the vector s by $\underline{s} = \underline{A}' \underline{e}_j$. The i-th component of this vector is the ith component of row j of the matrix A.

the choice $b_j = m(r - q_j)\Delta T - m \left[\sum_{i=1}^n \ln \left(\frac{e^{A_{ij}} + 1}{2} \right) \right]$ for the components of the vector b guarantees that $\ln(E(W_j)) = m(r - q_j)\Delta T$ and hence that $E(W_j) = e^{m(r-q_j)\Delta T}$ for all $j = 1 \dots n$

This variable W_j is the price relative for the jth asset in our model.

The numerical difference between the value of the vector b computed in this way and the value computed the other way is usually small and as m gets bigger the 2 converge.

Comments on the above result

1) This is a n -dimensional extension of the well known binomial option pricing method. It is an extension of the “equal probability” version of the binomial model, not of the Cox Ross Rubinstein version.

The CRR version fixes the size of the up and down movements (u and d) and then solves for the probability p of an up step. This version of the model is more difficult to generalise to more than one dimension. It can be generalised to 2 dimensions but it is difficult to go beyond 2.

The equal probability version of the model fixes the probability of an up step to be 0.50 and then solves for the up and down movements per time step. This is much easier to generalise to n dimensions.

We “solve” for either the probabilities or the up and down movements by doing “moment matching”.

2) The above result applies to European type options (those only exercisable at maturity and where the payoff depends on asset prices observed at maturity). However it can be extended to apply to options with early exercise payoffs and options where the payoff at time T depends on prices observed at some earlier time.

3) The b vector in the main result does not guarantee that the distribution of asset prices is “arbitrage free” but it is approximately arbitrage free and the approximation improves as m increases. The other formula for b does guarantee the model will match the moments of the arbitrage free distribution. This is a slightly more complicated expression but the numerical values it produces are very close to the values given by the first b -vector formula.

4) This uses an “affine transformation” of the n dimensional binomial distribution to model the vector of log price relatives. This is similar to what we do in the 1-dimensional binomial model.

6) Numerical Results / Testing the model

The algorithm was tested by applying it to various multi asset options where it is possible to perform a valuation analytically.

1) We shall price an “exchange option” to exchange 100 oz of silver in return for 1 oz of gold. The payoff at maturity is $payoff = \max(G_T - S_T, 0)$

The valuation assumptions are:

5 year term

m = 60 time steps

risk free rate = 10% p.a.

asset	div yield	volatility	drift rate	initial price
1 gold	0%	20%	4.00%	\$ 380.00
2 silver	0%	20%	4.00%	\$ 400.00

correlations

100%	70%
70%	100%

Results:

expected payoff	\$ 59.73
option value using binomial	\$ 44.25
option value using analytic formula	\$ 44.21

Ratio of analytic price to binomial price 0.9991

2) We shall price a spread option on the difference between the values of 100 oz of silver and 1 oz of gold, with an exercise price of $K=20$. The payoff at maturity is $payoff = \max(G_T - S_T - K, 0)$. This can't be valued analytically by the BS formula.

The valuation assumptions are as above.

Results: option value using binomial \$ 38.11

3) we shall price an option with payoff

$$\text{payoff} = \max(G_T - 380, 0) + \max(S_T - 400, 0)$$

This payoff can be valued analytically as the sum of 2 standard call options. Using the same valuation assumptions as above, the results are:

binomial option value	\$324.66
analytic value	\$324.60

4) we shall price an option with payoff

$$\text{payoff} = \max(G_T + S_T - 780, 0)$$

This is a 2 asset rainbow option. We value it using the same valuation assumptions as above, except that we vary the correlation assumption. This payoff can't be valued analytically by the bs formula.

When the correlation is very high (99%) the 2 asset portfolio behaves like a single asset so the option value (\$324.53) is close to that in ex 3. When we change the correlation to 0% the option value drops to \$311.92.

5) we price a 3 asset option with payoff

$$\text{payoff} = \max(S1_T - S1_0, 0) + \max(S2_T - S2_0, 0) + \max(S3_T - S3_0, 0)$$

This payoff is the sum of the payoffs on 3 standard call options, where the exercise price is equal to the initial asset price ("at the money calls"). This can be valued analytically, so it allows us to check the accuracy of the binomial method. The valuation assumptions were $r = 6\%$, $T = 0.25$,

asset (i)	div yld	volatility	correl matrix		
5	0.04	0.2	100.0%	90.0%	60.0%
3	0.01	0.4	90.0%	100.0%	80.0%
2	0.02	0.1	60.0%	80.0%	100.0%

The analytic value is 0.5156. The binomial model value (using 30 time steps) is \$0.5145. The ratio of these 2 values is 99.77%. If we change the payoff to the sum of 3 put option payoffs, then the analytic value is \$0.4340 while the binomial value is 0.4328. The ratio is 99.73%

7) Concluding remarks:

This paper has shown how to create a discrete approximation to the multivariate normal distribution and the multivariate lognormal distribution.

This result is useful for option pricing applications and other financial applications involving correlated normal or lognormal variables. Because the number of calculations required is proportional to $(m + 1)^n$ (where m = number of time steps and n = dimension), and this grows very quickly, the method is practical only for n up to about 4. It is an alternative to monte carlo simulation for low dimensional problems.

The method covered here for option pricing can be extended to cover American style multi asset options and other options with early exercise features.

The results shown also have applications to statistics and mathematics, as a way to numerically compute expected values of functions of a multivariate normal or multivariate lognormal random variable. In particular for $n > 2$, expectations and probabilities from these distributions have to be computed numerically anyway. This is an alternative way to do it.

The results shown here also have academic applications, for generating discrete multivariate random variables with a specified mean vector and specified covariance matrix. This could be useful for creating examples for students to examine and work on.

It has also shown how to compute expectations of functions of random variables with these distributions.

Speed of computation is an issue with this method. However for many applications speed is not the only consideration. For an option dealer doing a deal on the phone, instant computation may be required but for many other situations it is not such a big issue.

Appendix 1: Detailed Numerical Example:

The best way to understand the above result and algorithm is by working through a numerical example. We shall do a 4 step valuation of a 3 asset “rainbow option” with a term to maturity of 3 months. Assume we have an “at the money put option” over the value of a portfolio of 3 assets. The valuation details are:

$$T = 0.25, \quad m = 4, \quad \frac{T}{m} = \frac{1}{16} = 0.0625 \quad \sqrt{\frac{T}{m}} = 0.25$$

$$r = 6\%$$

asset	div yield	volatility	drift rate	initial price
1	4%	20%	0.00%	\$5.00
2	1%	40%	-3.00%	\$3.00
3	2%	10%	3.50%	\$2.00
				\$10.00

$$\underline{\mu}_a = \begin{pmatrix} +0.0000 \\ -0.0300 \\ +0.0350 \end{pmatrix}$$

Correlation matrix

100%	90%	60%
90%	100%	80%
60%	80%	100%

Covariance matrix (annualised) $\underline{\Sigma}_a$

0.040	0.072	0.012
0.072	0.160	0.032
0.012	0.032	0.010

Covariance matrix $\underline{\Sigma} = T \cdot \underline{\Sigma}_a$

0.0100	0.0180	0.0030
0.0180	0.0400	0.0080
0.0030	0.0080	0.0025

Cholesky square root of annual cov matrix $chol(\underline{\underline{\Sigma}}_a)$

0.200000	0.000000	0.000000
0.360000	0.174356	0.000000
0.060000	0.059648	0.053311

Computing the A matrix

$$\underline{\underline{A}} = 2\sqrt{\frac{T}{m}} \times chol(\underline{\underline{\Sigma}}_a)$$

0.100000	0.000000	0.000000
0.180000	0.087178	0.000000
0.030000	0.029824	0.026656

Computing the b vector:

We can compute the \underline{b} vector so that the distribution of the vector \underline{X} the vector of log price relatives, has the correct mean and covariance structure. This gives us

$$\underline{b} = \underline{\underline{\mu}}_a T - \underline{\underline{A}} \times \underline{1}$$

$$\underline{b} = \begin{pmatrix} +0.0000 \\ -0.0300 \\ +0.0350 \end{pmatrix} \times 0.25 - \sqrt{4 \times 0.25} \times \begin{pmatrix} 0.200000 & 0.000000 & 0.000000 \\ 0.360000 & 0.174356 & 0.000000 \\ 0.060000 & 0.059648 & 0.053311 \end{pmatrix} \times \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\underline{b} = \begin{pmatrix} -0.200000 \\ -0.541856 \\ -0.164209 \end{pmatrix}$$

Alternatively we can compute the \underline{b} vector so that the distribution of the vector \underline{X} has the right covariance structure but the distribution of the vector $\underline{W} = \exp(\underline{X})$ has the correct mean vector (for the arbitrage free joint distribution of the n asset prices)

$$b_j = m(r - q_j)\Delta T - m \left[\sum_{i=1}^n \ln \left(\frac{e^{s_i} + 1}{2} \right) \right] \quad \text{where } \underline{s} = \underline{\underline{A}}' \underline{e}_j$$

For $j = 1$ we have $\underline{e}_j = (1 \ 0 \ 0)$ and $\underline{s} = \underline{\underline{A}}' \underline{e}_j = (0.1 \ 0 \ 0)$ and

$$\begin{aligned} b_1 &= (r - q_1)\Delta T - m \times \left[\sum_{i=1}^n \ln \left(\frac{e^{s_i} + 1}{2} \right) \right] \\ &= (0.06 - 0.04) \times 0.25 - 4 \times (0.0512495 + 0 + 0) = -0.199998 \end{aligned}$$

For $j = 2$ we have $\underline{e}_j = (0 \ 1 \ 0)$ and $\underline{s} = \underline{\underline{A}}' \underline{e}_j = (0.18 \ 0.087178 \ 0)$ and

$$\begin{aligned} b_2 &= (r - q_2)\Delta T - m \times \left[\sum_{i=1}^n \ln \left(\frac{e^{s_i} + 1}{2} \right) \right] \\ &= (0.06 - 0.01) \times 0.25 - 4 \times (0.094045 + 0.044539 + 0) = -0.541833 \end{aligned}$$

For $j = 3$ we have $\underline{e}_j = (0 \ 0 \ 1)$ and

$\underline{s} = \underline{\underline{A}}' \underline{e}_j = (0.03 \ 0.029824 \ 0.026656)$ and

$$\begin{aligned} b_3 &= (r - q_3)\Delta T - m \times \left[\sum_{i=1}^n \ln \left(\frac{e^{s_i} + 1}{2} \right) \right] \\ &= (0.06 - 0.02) \times 0.25 - 4 \times (0.015112 + 0.015023 + 0.013417) = -0.164209 \end{aligned}$$

Note that the numerical values in the \underline{b} vector computed this way

$$\underline{b} = \begin{pmatrix} -0.199998 \\ -0.541833 \\ -0.164209 \end{pmatrix}$$

are very close to the numerical values in the \underline{b} vector computed in the other

$$\text{way. } \underline{b} = \begin{pmatrix} -0.200000 \\ -0.541856 \\ -0.164209 \end{pmatrix}$$

Note also that the 3 \underline{s} -vectors are the 3 rows of the matrix $\underline{\underline{A}}$

Sample Calculation at time step m=4

Consider time step $m=4$: the set of vectors / nodes at time step 4 is of size $125 = (4+1)^3$

- The 117th y vector is $\underline{y} = (1, 3, 4)$
- The affine transformation $\underline{x} = \underline{A}\underline{y} + \underline{b}$ maps this vector to $\underline{x} = (-0.099998, -0.100299, 0.061885)$
- The exponentiation of \underline{x} is $\underline{w} = (0.904839 \quad 0.904567 \quad 1.063841)$
- The vector of asset prices at time step $m=4$ is thus $\underline{S} = (5 \times 0.904839 \quad 3 \times 0.904567 \quad 2 \times 1.063841)$
- The value of the portfolio is $\$9.3656 = 5 \times 0.904839 + 3 \times 0.904567 + 2 \times 1.063841$
- The put option payoff is $\max(10.0000 - 9.3656, 0) = 0.6344$
- The probability of this payoff is $= \binom{4}{1} \times \binom{4}{3} \times \binom{4}{4} \left(\frac{1}{2}\right)^{12} = 0.00390625$

The Option Price for a European Option

The expected payoff at maturity (time step 4) is \$0.4214. This is the sum of the products of payoff times probability across all 125 possible “nodes” in the “tree” at time step 4, i.e. the expectation at time step 0 of the payoff at time step 4.

The value of the option using 4 time steps is \$0.4151. This is the expected payoff at time step 4, discounted back to the valuation date (time step 0).

If we increase the number of time steps to 20 the option price is \$0.4139.

If we increase the number of time steps to 30 the option price is \$0.4134.

Appendix 2: Proofs

It is well known that

$$1. Y \sim \text{bin}(m, p) \Rightarrow \Pr(Y = y) = \frac{m!}{y!(m-y)!} p^y (1-p)^{m-y} \text{ for } y = 0, 1, \dots, m$$

$$2. M_Y(t) = E(e^{yt}) = [pe^t + (1-p)]^m$$

3. if Y_1, \dots, Y_n are iid $\text{bin}(m, p)$ then

$$\Pr(Y_1 = y_1, \dots, Y_n = y_n) = \prod_{i=1}^n \left(\frac{m!}{y_i!(m-y_i)!} p^{y_i} (1-p)^{m-y_i} \right)$$

$$M_{Y_1, \dots, Y_n}(t_1, \dots, t_n) = E(e^{Y't}) = \prod_{i=1}^n [pe^{t_i} + (1-p)]^m$$

$$\therefore \ln(M_{Y_1, \dots, Y_n}(t_1, \dots, t_n)) = m \left[\sum_{i=1}^n \ln(pe^{t_i} + (1-p)) \right]$$

$$\therefore \ln(M_{Y_1, \dots, Y_n}(t_1, \dots, t_n)) = m \left[\sum_{i=1}^n \ln\left(\frac{e^{t_i} + 1}{2}\right) \right] \text{ when } p = 0.5$$

4. Result from multivariate statistics (about moment generating functions and linear transformations): $\underline{x} = \underline{A}\underline{y} + \underline{b} \Rightarrow M_{\underline{x}}(\underline{t}) = \exp(\underline{t}'\underline{b})M_{\underline{y}}(\underline{A}'\underline{t})$

5 Define the vector \underline{e}_j by $e_j(i) = 1, i = j$ and $e_j(i) = 0, i \neq j$. Then it follows that $E(e^{x_j}) = M_{\underline{x}}(\underline{e}_j) = \exp(\underline{e}_j'\underline{b})M_{\underline{y}}(\underline{A}'\underline{e}_j)$. We also have $\underline{e}_j'\underline{b} = b_j$ and $\underline{A}'\underline{e}_j = \text{jth column of the transpose of A} = \text{jth row of A}$.

6. Let $\underline{Y} \in R^n$ be a random variable with mean vector $E(\underline{Y}) = \underline{\mu}_Y \in R^n$ and $n \times n$ covariance matrix $\underline{S}_Y = \text{cov}(\underline{Y})$

Let \underline{A} be some constant $n \times n$ matrix and let $\underline{b} \in R^n$ be some constant vector

Then the vector random variable $\underline{X} = \underline{A}\underline{Y} + \underline{b}$ has

- dimension n
- mean vector $E(\underline{X}) = \underline{A} \times \underline{\mu} + \underline{b}$
- covariance matrix $\text{cov}(\underline{X}) = \underline{A} \times \underline{S}_Y \times \underline{A}^T$

7. Let \underline{S} be a positive definite, symmetric $n \times n$ covariance matrix. Then there exists a unique $n \times n$ lower triangular matrix \underline{L} such that $\underline{L} \times \underline{L}^T = \underline{S}$. Also the matrix \underline{L} is invertible. This matrix \underline{L} is known as the cholesky square root of the matrix $\underline{S} = \text{cov}(\underline{Y})$. Being lower triangular means that $L_{ij} = 0$ for $i > j$ so that the entries above the main diagonal are all zero.

The notation $\underline{L} = \text{chol}(\underline{S})$ means \underline{L} is the cholesky square root of \underline{S}

Lemma

$$\underline{L} = \text{chol}(\underline{S}) \Rightarrow \sqrt{T} \times \underline{L} = \text{chol}(T \times \underline{S}) \text{ for a constant } T \in R$$

8. if Y_1, \dots, Y_n are iid $\text{bin}(m, p)$ then we say that the vector $\underline{Y} = (Y_1, \dots, Y_n)$ has the binomial $\text{bin}(n, m, p)$ distribution . The mean vector and covariance matrix of this distribution are $\underline{\mu} = \frac{m}{2} \times \underline{1}$ and $\underline{\Sigma}_{\underline{Y}} = \frac{m}{4} \times \underline{I}$

9. Affine Transformations and Moment Matching

Let $\underline{\Sigma}$ be a specified $n \times n$ positive definite symmetric covariance matrix

Let $\underline{\mu} \in R^n$ be some specified constant vector

Let $\underline{Y} \in R^n$ be a random variable with mean vector $E(\underline{Y}) = \underline{\mu}_Y \in R^n$ and $n \times n$ covariance matrix $\underline{S}_Y = \text{cov}(\underline{Y})$

Then there exists

a $n \times n$ matrix \underline{A} and

a vector $\underline{b} \in R^n$

Such that the affine transformation of the vector $\underline{Y} \in R^n$ defined by $\underline{X} = \underline{A}\underline{Y} + \underline{b}$ has

- dimension n
- mean vector $E(\underline{X}) = \underline{\mu}$
- covariance matrix $\text{cov}(\underline{X}) = \underline{\Sigma}$

The matrix \underline{A} is defined by

$$\underline{A} = \underline{L}_1 \times \underline{L}_2^{-1} \quad \text{where } \underline{L}_1 = \text{chol}(\underline{\Sigma}) \quad \text{and} \quad \underline{L}_2 = \text{chol}(\underline{S}_Y)$$

The vector \underline{b} is defined by $\underline{b} = \underline{\mu} - \underline{A}\underline{\mu}_Y$

10. Affine Transformation for equal probability binomial model

Let $\underline{Y} \in R^n$ be a n-dimensional binomial random variable with parameters \mathbf{m} and equally likely probabilities so that

- mean vector $E(\underline{Y}) = \underline{\mu}_Y = \frac{m}{2} \times \underline{1} = \frac{m}{2} \begin{pmatrix} 1 \\ \dots \\ 1 \end{pmatrix} \in R^n$
- $n \times n$ covariance matrix $S_{\underline{Y}} = \frac{m}{4} \times I_{n \times n}$
- the components Y_i of the vector \underline{Y} are iid $Y_i \sim \text{bin}(m, 0.50)$
- $\Pr\{(Y_1 \dots Y_n) = (y_1 \dots y_n)\} = \binom{m}{y_1} \times \dots \times \binom{m}{y_n} \left(\frac{1}{2}\right)^{nm}$

Then

$$\underline{L}_2 = \text{chol}(S_{\underline{Y}}) = \frac{\sqrt{m}}{2} \times I_{n \times n}$$

$$\underline{L}_2^{-1} = \frac{2}{\sqrt{m}} \times I_{n \times n}$$

$$\underline{A} = \underline{L}_1 \underline{L}_2^{-1} = \frac{2}{\sqrt{m}} \times \underline{L}_1 = \frac{2}{\sqrt{m}} \times \text{chol}(\underline{\Sigma})$$

$$\underline{b} = \underline{\mu} - \underline{A} \underline{\mu}_Y = \underline{\mu} - \sqrt{m} \times \text{chol}(\underline{\Sigma}) \times \underline{1} = \underline{\mu} - \sqrt{m} \underline{L}_1 \times \underline{1}$$

The i th component of this vector is $b_i = \mu_i - \sqrt{m} \times \left[\sum_{j=1}^n L_1(i, j) \right]$

Example: 2 dimensional case

$$\underline{\underline{\Sigma}} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix} \quad \underline{\underline{L}}_1 = chol(\underline{\underline{\Sigma}}) = \begin{pmatrix} \sigma_1 & 0 \\ \rho\sigma_2 & \sigma_2\sqrt{1-\rho^2} \end{pmatrix}$$

$$\underline{\underline{L}}_1^{-1} = \begin{pmatrix} \frac{1}{\sigma_1} & 0 \\ -\frac{\rho}{\sigma_1\sqrt{1-\rho^2}} & \frac{1}{\sigma_2\sqrt{1-\rho^2}} \end{pmatrix}$$

$$\underline{\underline{S}} = \frac{m}{4} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\underline{\underline{L}}_2 = chol(\underline{\underline{S}}) = \frac{\sqrt{m}}{2} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \underline{\underline{L}}_2^{-1} = \frac{2}{\sqrt{m}} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\underline{\underline{A}} = \underline{\underline{L}}_1 \times \underline{\underline{L}}_2^{-1} = \frac{2}{\sqrt{m}} \begin{pmatrix} \sigma_1 & 0 \\ \rho\sigma_2 & \sigma_2\sqrt{1-\rho^2} \end{pmatrix}$$

$$\underline{\underline{\mu}} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \quad \underline{\underline{\mu}}_Y = \frac{m}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$\underline{\underline{b}} = \underline{\underline{\mu}} - \underline{\underline{A}}\underline{\underline{\mu}}_Y = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} - \frac{2}{\sqrt{m}} \times \begin{pmatrix} \sigma_1 & 0 \\ \rho\sigma_2 & \sigma_2\sqrt{1-\rho^2} \end{pmatrix} \times \frac{m}{2} \times \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$= \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} - \begin{pmatrix} \sqrt{m}\sigma_1 \\ \sqrt{m}\sigma_2(\rho + \sqrt{1-\rho^2}) \end{pmatrix}$$

$$\underline{\underline{b}} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} - \sqrt{m} \begin{pmatrix} \sigma_1 \\ \sigma_2(\rho + \sqrt{1-\rho^2}) \end{pmatrix}$$

Example: 3 dimensional case

$$\underline{\underline{\Sigma}} = \begin{pmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 & \rho_{13}\sigma_1\sigma_3 \\ \rho_{12}\sigma_1\sigma_2 & \sigma_2^2 & \rho_{23}\sigma_2\sigma_3 \\ \rho_{13}\sigma_1\sigma_3 & \rho_{23}\sigma_2\sigma_3 & \sigma_3^2 \end{pmatrix}$$

$$\underline{\underline{L}}_1 = chol(\underline{\underline{\Sigma}})$$

$$= \begin{pmatrix} \sigma_1 & 0 & 0 \\ \sigma_2\rho_{12} & \sigma_2\sqrt{1-\rho_{12}^2} & 0 \\ \sigma_3\rho_{13} & \sigma_3\frac{(\rho_{23}-\rho_{12}\rho_{13})}{\sqrt{1-\rho_{12}^2}} & \sigma_3\sqrt{(1-\rho_{13}^2)-\frac{(\rho_{23}-\rho_{12}\rho_{13})^2}{(1-\rho_{12}^2)}} \end{pmatrix}$$

$$\underline{\underline{S}} = \frac{m}{4} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\underline{\underline{L}}_2 = chol(\underline{\underline{S}}) = \frac{\sqrt{m}}{2} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\underline{\underline{L}}_2^{-1} = \frac{2}{\sqrt{m}} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\underline{\underline{A}} = \underline{\underline{L}}_1 \times \underline{\underline{L}}_2^{-1} =$$

$$\frac{2}{\sqrt{m}} \begin{pmatrix} \sigma_1 & 0 & 0 \\ \sigma_2 \rho_{12} & \sigma_2 \sqrt{1 - \rho_{12}^2} & 0 \\ \sigma_3 \rho_{13} & \sigma_3 \frac{(\rho_{23} - \rho_{12} \rho_{13})}{\sqrt{1 - \rho_{12}^2}} & \sigma_3 \sqrt{(1 - \rho_{13}^2) - \frac{(\rho_{23} - \rho_{12} \rho_{13})^2}{(1 - \rho_{12}^2)}} \end{pmatrix}$$

$$\underline{\underline{\mu}} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix}$$

$$\underline{\underline{\mu}}_Y = \frac{m}{2} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\underline{\underline{b}} = \underline{\underline{\mu}} - \underline{\underline{A}} \underline{\underline{\mu}}_Y = \underline{\underline{\mu}} - \sqrt{m} \underline{\underline{L}}_1 \underline{\underline{1}}$$

where $\underline{\underline{1}}$ is vector of 1's

The adjustment vector $\sqrt{m} \underline{\underline{L}}_1 \underline{\underline{1}}$ is subtracted from the vector $\underline{\underline{\mu}} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix}$

This adjustment is $\sqrt{m} \underline{\underline{L}}_1 \underline{\underline{1}} = \begin{pmatrix} \sigma_1 \\ \sigma_2 \alpha \\ \sigma_3 \beta \end{pmatrix}$

where $\alpha = \rho_{12} + \sqrt{1 - \rho_{12}^2}$

$$\beta = \rho_{13} + \sqrt{\frac{(\rho_{23} - \rho_{12} \rho_{13})^2}{(1 - \rho_{12}^2)}} + \sqrt{(1 - \rho_{13}^2) - \frac{(\rho_{23} - \rho_{12} \rho_{13})^2}{(1 - \rho_{12}^2)}}$$

Note that the sum of the squares of the components of α & β is 1.0

11. Algorithm for pricing European style option prices

For the n-dimensional, m step binomial model, we assume that:

- the parameters $r, m, T, \underline{\mu}_a, \underline{\Sigma}_a$ are known
- the initial values $(S_1(0), S_2(0), \dots, S_n(0))$ are known,
- the payoff (P) we are valuing happens at time T and is some function $P = f(S_1(T), S_2(T), \dots, S_n(T))$ of the n asset values at time T

Compute $\underline{A} = 2\sqrt{\frac{T}{m}} \times chol(\underline{\Sigma}_a)$

Compute $\underline{b} = \underline{\mu}_a T - \underline{A} \times \underline{1}$

Expectation = 0

For $i = 0$ to $(m+1)^n - 1$

 Compute n dimensional vector \underline{y}

 Compute $\underline{x} = \underline{A}\underline{y} + \underline{b}$

 Compute $\underline{w} = \exp(\underline{x})$ i.e.

$w_1 = \exp(x_1), w_2 = \exp(x_2), \dots, w_n = \exp(x_n)$

 Compute $(S_1(T), \dots, S_n(T)) = (W_1 S_1(0), \dots, W_n S_n(T))$

 Compute $Payoff = f(S_1(T), S_2(T), \dots, S_n(T))$

 Compute $\Pr\{(Y_1 \dots Y_n) = (y_1 \dots y_n)\} = \binom{m}{y_1} \times \dots \times \binom{m}{y_n} \left(\frac{1}{2}\right)^{nm}$

 expectation = expectation + Payoff × probability

Next i

price = expectation × exp(-rT)

This algorithm gives the price of the contingent claim. The term “expectation” here is the risk neutral expected payoff

12. The vector random variable $\underline{X} = \underline{\underline{A}}\underline{Y} + \underline{b}$ is a discrete model for the vector of logarithms of price relatives of the n assets.

As $m \rightarrow \infty$ the distribution of $\underline{X} = \underline{\underline{A}}\underline{Y} + \underline{b}$ converges to a multivariate normal distribution with mean vector

$$\underline{\underline{\mu}} = \left(\left(r - q_1 - \frac{1}{2}\sigma_1^2 \right), \left(r - q_2 - \frac{1}{2}\sigma_2^2 \right), \dots, \left(r - q_n - \frac{1}{2}\sigma_n^2 \right) \right) T = \underline{\underline{\mu}}_a \times T$$

$$\text{covariance matrix } \underline{\underline{\Sigma}} = \begin{pmatrix} \sigma_1^2 & \sigma_1\sigma_2\rho_{12} & \dots & \sigma_1\sigma_n\rho_{1n} \\ \sigma_1\sigma_2\rho_{12} & \sigma_2^2 & \dots & \sigma_2\sigma_n\rho_{2n} \\ \dots & \dots & \dots & \dots \\ \sigma_1\sigma_n\rho_{1n} & \sigma_2\sigma_n\rho_{2n} & \dots & \sigma_n^2 \end{pmatrix} T = \underline{\underline{\Sigma}}_a \times T$$

This is true because of the central limit theorem. Each component of Y converges to a normal distribution, so Y converges to a multivariate normal. X being an affine transformation of Y also converges to a multivariate normal.

It follows that asymptotically ($m \rightarrow \infty$), the vector $\underline{w} = \exp(\underline{x})$ i.e. $w_i = \exp(x_i), i = 1 \dots n$ converges to a multivariate lognormal distribution and that $E(w_i) = E(\exp(x_i)) = \exp((r - q_i)T)$

This is the “arbitrage free joint distribution” of the n assets.

However, being a discrete random variable, for m finite we have

$E(w_i) = E(\exp(x_i)) \approx \exp((r - q_i)T)$ but the approximation improves in accuracy as m gets bigger.

We can use a different b vector here, which will guarantee that

$E(w_i) = E(\exp(x_i)) = \exp((r - q_i)T)$ for all values of m, so that the model is “arbitrage free” for all values of m. The formula for the b vector is more complicated however.

The vector b for arbitrage freeness:

Let \underline{e}_j be the n dimensional vector defined by $\underline{e}_{ji} = \delta_{ji} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$

Define the vector s by $\underline{s} = \underline{A}' \underline{e}_j$

By result 5 above we have $E(e^{X_j}) = e^{b_j} M_y(\underline{s})$

by result 4 we have $\ln(E(e^{X_j})) = b_j + m \left[\sum_{i=1}^n \ln \left(\frac{e^{s_i} + 1}{2} \right) \right]$

It follows that the choice $b_j = m(r - q_j)\Delta T - m \left[\sum_{i=1}^n \ln \left(\frac{e^{s_i} + 1}{2} \right) \right]$ for the

components of the vector b

guarantees that $\ln(E(W_j)) = m(r - q_j)$

and hence that $E(W_j) = e^{m(r - q_j)}$ for all $j = 1 \dots n$

This variable W_j is the price relative for the jth asset in our model.

Compute \underline{b} from $b_j = (r - q_j)T - m \left[\sum_{i=1}^n \ln \left(\frac{e^{s_i} + 1}{2} \right) \right]$ for all $j = 1 \dots n$