

CHEME 5660 Unit 2: Equity and Equity Pricing Models

Jeffrey D. Varner

Smith School of Chemical and Biomolecular Engineering
Cornell University, Ithaca NY 14853

1 Introduction

In this unit, we will introduce the concept of equity and equity pricing models.

2 Lattice models of equity prices

A lattice model discretizes the potential future states of the world into a finite number of options. For instance, a binomial lattice model has two future states: `up` and `down`, while a ternary lattice model has three: `up`, `down`, and `flat`. To make predictions, we must assign values and probabilities to each of these future states and then calculate the expected value and variance of future values. Thus, we do not precisely know quantities such as share price because we are projecting into the future. Instead, we have only a probabilistic model of the possible future values. We'll begin with the simplest possible lattice model, a binomial lattice (Fig. 1).

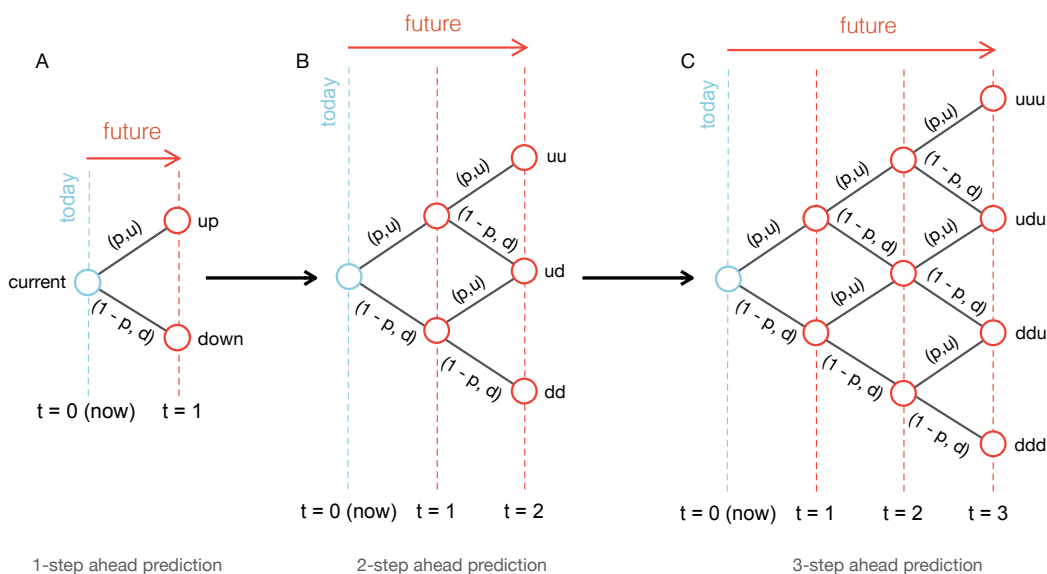


Fig. 1: Binomial lattice model schematic. At each node, the share price can either go up by u or down by d . The probability of going up is p , and the likelihood of going down is $1 - p$. **A:** Single time-step lookahead. **B:** Two time-step lookahead. **C:** Three time-step lookahead. At the tree l level, the potential share price can take on $l + 1$ values.

Let's start with a single time-step lookahead, with two possible future states (Fig. 1A). Let the initial share price at time 0 be S_0 and the share price at future time 1 be S_1 . During the transition from time $0 \rightarrow 1$, the world transitions from the current state to one of two possible future states: `up` or `down`. We move to the `up` state with probability p or the `down` state with probability $(1 - p)$.

Thus, at the time 1, the share price S_1 can take on one of two possible values: $S^u = u \cdot S_0$ if the world moves to the `up` state, or $S^d = d \cdot S_0$ if the world moves to the `down` state. As we move to the future, we can continue to build out the lattice model by adding additional time steps; for example, consider a two-step ahead prediction (Fig. 1B). At time 2, the share price can take on one of three possible values: $S^{uu} = u^2 \cdot S_0$ if the world moves to the `up-up` state, $S^{ud} = ud \cdot S_0$ if the world moves to the `up-down` state, or $S^{dd} = d^2 \cdot S_0$ if the world moves to the `down-down` state. We can continue to build out the lattice model by adding additional time steps; for example, consider a three-step ahead prediction (Fig. 1C).

2.1 Analytical solution

Let's consider a binomial lattice model with n time-steps. At each time step, the share price can either go `up` by a factor of u or `down` by a factor of d . Then, at time n , the share price can take on $n + 1$ possible values:

$$S_n = S_0 \times D_1 \times D_2 \times D_3 \times \cdots \times D_n \quad (1)$$

where D_i is a random variable that can take on one of two values: u or d , with probabilities p and $(1 - p)$ respectively. Thus, at each time-step, the world flips a coin and lands in either the `up` state with probability p or the `down` state with probability $(1 - p)$. For a single time step, we model this random process as a Bernoulli trial, where the probability of success is p and the likelihood of failure is $(1 - p)$. As the number of time steps increases, we have a series of Bernoulli trials, which is a binomial distribution (Defn: 1):

Definition 1: Binomial Share Price and Probability

Let S_0 denote the current share price at $t = 0$, u and d denote the `up` and `down` factors, and p denote the probability of going `up`. At time t , the binomial lattice model predicts the share price S_t is given by:

$$S_t = S_0 \cdot u^{t-k} \cdot d^k \quad \text{for } k = 0, 1, \dots, t$$

The probability that the share price takes on a particular value at time t is given by:

$$P(S_t = S_0 \cdot u^{t-k} \cdot d^k) = \binom{t}{k} \cdot (1 - p)^k \cdot p^{t-k} \quad \text{for } k = 0, 1, \dots, t$$

where $\binom{t}{k}$ denotes the binomial coefficient.

2.2 Models of u , d and p

The `up` and `down` factors u and d , and the probability p can be computed in various ways. For example, we can estimate them from historical data or propose models for their values. Let's start by looking at approaches to estimate these parameters from historical data. Then, we'll introduce an important model developed by Cox, Ross, and Rubinstein (CRR) to compute u , d , and p in the context of options pricing that we'll see later [1].

2.2.1 Real-world probability p Suppose we have a historical share price dataset from time $1, \dots, T$ for some ticker j denoted as $\mathcal{D}_j = \{S_1^{(j)}, S_2^{(j)}, \dots, S_T^{(j)}\}$, where $S_i^{(j)}$ denotes the share

price of ticker j at time i . We can use different values for the share price, e.g., the opening price, closing price, high price, low price, etc. When dealing with historical data, we will typically use the volume-weighted average price (VWAP) for the period, e.g., the VWAP for the day, week, month, etc. Over the time range of the dataset \mathcal{D}_j , we can calculate the number of up and down moves occurring between time $i - 1$ and i , and the magnitude of these moves. Then, the fraction of up moves is an estimate of the probability p , while some measure of the magnitude of the up and down moves, e.g., the average value are estimates of u and d respectively.

Suppose we assume the share price of ticker j is continuously compounded with an instantaneous discount (interest) rate of $r_{i,i-1}^{(j)} \equiv \mu_{i,i-1}^{(j)} \cdot \Delta t$, i.e., we split the return into a growth rate $\mu_{i,i-1}^{(j)}$ and a time step size Δt . Then, the share price at time i is an expression of the form

$$S_i^{(j)} = \exp(\mu_{i,i-1} \cdot \Delta t) \cdot S_{i-1}^{(j)} \quad (2)$$

where $\mu_{i,i-1}^{(j)}$ denotes the *growth rate* (units: 1/time) for ticker j , and Δt (units: time) is the time step size during the time period $(i - 1) \rightarrow i$. Solving for the growth rate (and dropping the ticker j superscript for simplicity) gives:

$$\mu_{i,i-1} = \left(\frac{1}{\Delta t} \right) \cdot \ln \left(\frac{S_i}{S_{i-1}} \right) \quad (3)$$

We often use daily price data; thus, a single trading day is a natural time frame between S_{i-1} and S_i . However, subsequently, it will be easier to use an annualized value for the μ parameter; thus, we let $\Delta t = 1/252$, i.e., the fraction of an average trading year that occurs in a single trading day; thus, our base time will be years. We compute the growth rate for each trading day i in a collection of datasets \mathcal{D} using Algorithm 4. Then we can estimate the up and down factors u and d from

Algorithm 1 Logarithmic excess growth rate

Require: Collection of price datasets \mathcal{D} , where $\mathcal{D}_j \in \mathcal{D}$. All datasets have the same length N .

Require: list of stocks \mathcal{L} , where $\dim \mathcal{L} = M$.

Require: time step size Δt between t and $t - 1$ (units: years),

Require: risk-free rate r_f (units: inverse years).

```

1:  $M \leftarrow \text{length}(\mathcal{D}_1)$                                 ▷ Number of trading days in the dataset  $\mathcal{D}_j$ 
2:  $N \leftarrow \text{length}(\mathcal{L})$                                 ▷ Number of stocks in the dataset  $\mathcal{D}$ 
3:  $\mu \leftarrow \text{Array}(M - 1, N)$                           ▷ Initialize empty array of growth rates

4: for  $i \in \mathcal{L}$  do
5:    $\mathcal{D}_i \leftarrow \mathcal{D}[i]$                                 ▷ Get dataset for stock  $i$ 
6:   for  $t = 2 \rightarrow N$  do
7:      $S_1 \leftarrow \text{VWAP}(\mathcal{D}_i[t - 1])$     ▷ Get volume weighted average price for stock  $i$  at time  $t - 1$ 
8:      $S_2 \leftarrow \text{VWAP}(\mathcal{D}_i[t])$         ▷ Get volume weighted average price for stock  $i$  at time  $t$ 
9:      $\mu[t - 1, i] \leftarrow \left( \frac{1}{\Delta t} \right) \cdot \ln \left( \frac{S_2}{S_1} \right) - r_f$     ▷ Set  $r_f = 0$  for regular growth rate
10:  end for
11: end for
```

the growth rate array generated using something like Algorithm 2. While this strategy is simple,

Algorithm 2 Estimating u , d and p from the μ -array**Require:** Growth-rate μ -array from Algorithm 4.**Require:** Effective risk-free rate \bar{r} (units: inverse years).**Require:** time step size Δt between t and $t - 1$ (units: years) of the price data**Require:** mean function, find function, length function, and push function.

```

1: procedure REAL WORLD PROBABILITY( $\mu$ ,  $\bar{r}$ ,  $\Delta t$ )
2:
3:   Initialize:
4:    $N \leftarrow \text{length}(\mu)$                                 ▷ Number of growth rates
5:    $\text{up} \leftarrow \text{Array}()$                                 ▷ Initialize empty array of up factors
6:    $\text{down} \leftarrow \text{Array}()$                              ▷ Initialize empty array of down factors

7:   Compute  $u$ :
8:    $i_+ \leftarrow \text{find}(\mu > 0)$                             ▷ Find the indices of all positive growth rates
9:   for  $i \in i_+$  do
10:     $\text{push}(\text{up}, \exp(\mu[i] \cdot \Delta t))$                 ▷ Push the positive return  $\mu \cdot \Delta t$  onto up-array
11:  end for
12:   $u \leftarrow \text{mean}(\text{up})$                                 ▷ mean is our estimate of the up factor  $u$ 

13:  Compute  $d$ :
14:   $i_- \leftarrow \text{find}(\mu < 0)$                             ▷ Find the indices of all negative growth rates
15:  for  $i \in i_-$  do
16:     $\text{push}(\text{down}, \exp(\mu[i] \cdot \Delta t))$                 ▷ Push the negative return  $\mu \cdot \Delta t$  onto down-array
17:  end for
18:   $d \leftarrow \text{mean}(\text{down})$                                 ▷ mean is our estimate of the down factor  $d$ 

19:   $N_+ \leftarrow \text{length}(i_+)$                                 ▷ Number of positive growth rates
20:   $p \leftarrow N_+/N$                                         ▷ Estimate of the probability  $p$ 

21:  return  $u$ ,  $d$  and  $p$ 

22: end procedure

```

it may not be robust. For example, if the dataset \mathcal{D}_j is short, i.e., only a few trading days, then the number of up and down moves will be small, and the estimates of u , d and p will be poor. If a precise estimate of u , d and p is required, the number of trading days in the dataset \mathcal{D}_j should be large. Furthermore, the estimates of u , d and p are not robust to outliers in the dataset \mathcal{D}_j . Thus, we may want to consider other models for computing u , d and p .

2.2.2 Risk-neutral probability q Another approach to computing the parameters in the lattice is to use the risk-neutral probability q . Let's consider a single-step binomial lattice model (Fig. 1A). The expected value of the share price at time 1 for a single step is given by:

$$\mathbb{E}_{\mathbb{Q}}(S_1|S_0) = q \cdot S^u + (1 - q) \cdot S^d$$

where $\mathbb{E}_{\mathbb{Q}}(\dots)$ denotes the expectation operator written with respect to the risk-neutral probability measure \mathbb{Q} , the term q denotes the risk-neutral probability of the up state, and S^u and S^d are the share prices in the up and down states respectively. The hypothetical probability q is used to price derivatives (as we shall see later), however, we could also think of it as a tool to compute a *fair* price for a share of stock. Suppose we rewrite Eqn. (2) as:

$$\mathcal{D}_{1,0}(\bar{r}) \cdot S_0 = \mathbb{E}_{\mathbb{Q}}(S_1|S_0) \quad (4)$$

where $\mathcal{D}_{1,0}(\bar{r})$ is the continuous discount factor between period $0 \rightarrow 1$, and \bar{r} is the effective (constant) risk-free rate. Thus, unlike the previous case, where the share price S_i was discounted by the return $\mu_{i,i-1} \cdot \Delta t$ (which could vary in time), here we discount the share price S_i by an effective (constant) risk-free rate \bar{r} . The expectation operator $\mathbb{E}_{\mathbb{Q}}(\dots)$ can be matched with the expansion:

$$\mathcal{D}_{1,0}(\bar{r}) \cdot S_0 = q \cdot S^u + (1 - q) \cdot S^d \quad (5)$$

The share prices in the up and down states are the product of the up factor u (or a down factor d) and the initial share price, i.e., $S^u = u \cdot S_0$ and $S^d = d \cdot S_0$. Substituting these price values into Eqn. (5) and solving for q gives:

$$q = \frac{\mathcal{D}_{1,0}(\bar{r}) - d}{u - d} \quad (6)$$

Thus, we can compute the risk-neutral probability q from the up and down factors u and d and the effective risk-free rate \bar{r} . The risk neutral probability q , and the up and down factors u and d can be estimated using Algorithm 3.

2.3 Cox-Ross-Rubinstein (CRR) model

The Cox, Ross, and Rubinstein (CRR) lattice model assumes a risk-neutral probability q to describe the probability of an up move. However, unlike a purely data-driven risk-neutral probability where we estimate values for u and d from data, the CRR model proposes a model for u and d that is consistent with the variance of the return of the underlying asset. The up factor u is constructed so that the CRR model matches the variance of the log return $\text{Var}(S_j/S_{j-1}) = \sigma^2 \Delta t$:

$$\text{Var}(S_j/S_{j-1}) = \mathbb{E}[(S_j/S_{j-1})^2] - \mathbb{E}[S_j/S_{j-1}]^2 = \sigma^2 \Delta t$$

which after substitution of the price and probability expressions from Defn. 1 gives:

$$q(u - 1)^2 + (1 - q)(d - 1)^2 + [p(u - 1) + (1 - q)(d - 1)]^2 = \sigma^2 \Delta t \quad (7)$$

where σ is the volatility (standard deviation) of the return, and Δt is the time step. Equation (7) is a quadratic equation in u , which can be solved for u (assuming Δt^2 terms and above are ignored):

$$u = \exp(\sigma \cdot \sqrt{\Delta t})$$

In the CRR approach, the up and down factors are related by $ud = 1$, i.e., the lattice is symmetric.

To use the CRR model, we must first specify the volatility σ of the return of the underlying asset, e.g., shares of stock. We can estimate this from data by first computing the return (or the

Algorithm 3 Risk-neutral estimate of u , d and q **Require:** Growth-rate μ -array from Algorithm 4.**Require:** Effective risk-free rate \bar{r} (units: inverse years).**Require:** time step size Δt between t and $t - 1$ (units: years) of the price data**Require:** mean function, find function, length function, and push function.

```

1: procedure RISK NEUTRAL PROBABILITY( $\mu$ ,  $\bar{r}$ ,  $\Delta t$ )
2:
3:   Initialize:
4:    $N \leftarrow \text{length}(\mu)$                                 ▷ Number of growth rates
5:    $\text{up} \leftarrow \text{Array}()$                                 ▷ Initialize empty array of up factors
6:    $\text{down} \leftarrow \text{Array}()$                               ▷ Initialize empty array of down factors

7:   Compute  $u$ :
8:    $i_+ \leftarrow \text{find}(\mu > 0)$                             ▷ Find the indices of all positive growth rates
9:   for  $i \in i_+$  do
10:     $\text{push}(\text{up}, \exp(\mu[i] \cdot \Delta t))$                     ▷ Push the positive return  $\mu \cdot \Delta t$  onto up-array
11:  end for
12:   $u \leftarrow \text{mean}(\text{up})$                                 ▷ mean is our estimate of the up factor  $u$ 

13:  Compute  $d$ :
14:   $i_- \leftarrow \text{find}(\mu < 0)$                             ▷ Find the indices of all negative growth rates
15:  for  $i \in i_-$  do
16:     $\text{push}(\text{down}, \exp(\mu[i] \cdot \Delta t))$                 ▷ Push the negative return  $\mu \cdot \Delta t$  onto down-array
17:  end for
18:   $d \leftarrow \text{mean}(\text{down})$                                 ▷ mean is our estimate of the down factor  $d$ 

19:   $q \leftarrow \frac{\exp(\bar{r} \cdot \Delta t) - d}{u - d}$                 ▷ Compute the risk-neutral probability  $q$ 

20:  return  $u$ ,  $d$  and  $q$ 

21: end procedure

```

growth) and the estimation of the volatility of the return, which is a backward-looking approach, or we can use the implied volatility from the options market, which is a forward-looking approach (the market's estimate of future price movements).

2.4 Binomial lattice trade strategy

Suppose we purchase n_o share of ticker XYZ at time $t = 0$ for S_o USD/share. Then, at some later date $t = T$, we sell all n_o shares for S_T USD/share. The net present value (NPV) of this trade has two cashflow events, the initial purchase of n_o shares for S_o USD/share, and the sale of n_o shares for S_T USD/share at some later date $t = T$:

$$\text{NPV}(\bar{r}, T) = -n_o \cdot S_o + n_o \cdot S_T \cdot \mathcal{D}_{T,0}^{-1}(\bar{r}) \quad (8)$$

where $\mathcal{D}_{T,0}^{-1}(\bar{r})$ is the inverse of the continuous discount factor for period $0 \rightarrow T$, assuming an effective discount rate \bar{r} . Factoring the initial number of shares n_o from the NPV expression gives:

$$\text{NPV}(\bar{r}, T) = n_o \cdot (S_T \cdot \mathcal{D}_{T,0}^{-1}(\bar{r}) - S_o) \quad (9)$$

The term in the parenthesis is the net change in the share price, i.e., the change in the share price S_T discounted by the effective discount rate \bar{r} . However, if we hold the shares for only a short period, e.g., on order days or months, then $\mathcal{D}_{T,0}^{-1}(\bar{r}) \approx 1$, which simplifies the NPV expression to:

$$\text{NPV}(\bar{r}, T) \simeq n_o \cdot (S_T - S_o) \quad (10)$$

Dividing by the initial investment $n_o \cdot S_o$ gives the fraction return on investment, or just the fractional return:

$$\frac{\text{NPV}(\bar{r}, T)}{n_o \cdot S_o} \simeq \frac{S_T - S_o}{S_o} = \frac{S_T}{S_o} - 1 \quad (11)$$

In a binomial world, the share price S_T is a random variable which takes on $T + 1$ possible values:

$$S_T \in \left\{ S_o \cdot u^{T-k} \cdot d^k \right\}_{k=0}^T \quad (12)$$

where S_o is the initial share price, u and d are the up and down factors, and p is the probability of going up. Substituting the S_T expression into Eqn. (11) gives:

$$\frac{\text{NPV}(\bar{r}, T)}{n_o \cdot S_o} \in \left\{ u^{T-k} \cdot d^k - 1 \right\}_{k=0}^T \quad (13)$$

Equation (13) describes the distribution of the fractional return on investment predicted by the binomial lattice model. Because we also have the probability of each future share price, we can compute a number of interesting properties of the distribution of the fractional returns, e.g., the mean, variance, etc. Furthermore, we can compute the probability that the fractional return is greater than some desired threshold, e.g., 10%, by reconstructing the cumulative distribution function (CDF) of the fractional return. We'll illustrate this idea with the project for this module.

2.5 Summary

In this module we introduced the binomial lattice model for equity share price. The binomial lattice model is a discrete-time model, where the future share price can take on one of two possible values at each time step: up or down. In the up state, the share price increases by a factor of u , while in the down state, the share price decreases by a factor of d . The binomial lattice model is a probabilistic model, where the probability of going up is p and the probability of going down is $(1 - p)$. The up and down factors u and d , and the probability p can be computed in various ways, but are assumed to be constant over time. For example, we can compute the up and down factors u and d from historical data, or we can propose a model for u and d . Finally, we developed a simple trade strategy which relied on the binomial lattice model to compute a future price distribution, where the net present value was used to value the trade strategy.

3 Single Asset Geometric Brownian Motion

Geometric Brownian motion (GBM) is a continuous-time stochastic model in which the random variable $S(t)$, e.g., the share price of a firm. Geometric Brownian motion was popularized as a financial model by Samuelson in the 1950s and 1960s [2], but is arguably most commonly associated with the Black–Scholes options pricing model, which we will describe later [3]. Let's start with the single asset case (in the absence of dividends) and then consider the multiple asset case in the next module.

Geometric Brownian motion (GBM) assumes that the share price $S(t)$ of a firm can be modeled as a deterministic drift term (which is proportional to the share price) that is corrupted by a Wiener noise process, also proportional to the share price:

$$\frac{dS}{S} = \mu dt + \sigma dW \quad (14)$$

The constant μ denotes a drift parameter, i.e., the growth rate of the share price return, $\sigma > 0$ is a volatility parameter, i.e., the dispersion of the return, dt denotes an infinitesimal time step, and dW is the output of a Wiener noise process. A Wiener Process (also often referred to as a standard Brownian motion) is a real-valued continuous-time stochastic process named after Norbert Wiener for the study of one-dimensional Brownian motion (Defn. 2):

Definition 2: Wiener Process

A Wiener process is a continuous one-dimensional stochastic process $\{W(t), 0 \leq t \leq T\}$ with the following properties:

- $W(0) = 0$ with probability 1
- The increments $\{W(t_1) - W(t_0), \dots, W(t_k) - W(t_{k-1})\}$ are independent for any k and $0 \leq t_0 < t_1 < \dots < t_k \leq T$
- The increment $W(t) - W(s) \sim N(0, t - s)$ for any $0 \leq s < t \leq T$, where $N(0, t - s)$ denotes a normally distributed random variable with mean 0 and variance $t - s$.

Equation 14 is a continuous-time analog of the discrete-time binomial lattice model we developed previously. It has several exciting properties; for example, it has an analytical solution and analytical expressions for the expectation and variance of the share price. In this module, we will develop analytical solutions to Eqn. 14, and tools to estimate the parameters μ and σ from historical data. We'll then use these tools to simulate the share price of firms and analyze properties of the return distributions in the example and project for this module.

3.1 Analytical solution

Using Ito's lemma, we can formulate an analytical solution to the GBM equation for a single asset. K. Ito's Lemma, developed in 1951, is an analog of the Taylor series for stochastic systems. Let the random variable $X(t)$ be governed by the general stochastic differential equation:

$$dX = a(X(t), t) dt + b(X(t), t) dW(t)$$

where $dW(t)$ is a one-dimensional Wiener process and a and b are functions of $X(t)$ and t . Let $Y(t) = \phi(t, X(t))$ be twice differentiable with respect to $X(t)$, and singly differentiable with respect to t . Then, $Y(t)$ is governed by the equation:

$$dY = \left(\frac{\partial Y}{\partial t} + a \frac{\partial Y}{\partial X} + \frac{b^2}{2} \frac{\partial^2 Y}{\partial X^2} \right) dt + b \left(\frac{\partial Y}{\partial X} \right) dW(t)$$

Let $Y = \ln(S)$, $a = \mu \cdot S$, and $b = \sigma \cdot S$. Then, Y is governed by the stochastic differential equation (using Ito's Lemma):

$$d \ln(S) = \left(\mu - \frac{\sigma^2}{2} \right) dt + \sigma \cdot dW(t)$$

We integrate both sides of the equation to obtain from t_o to t :

$$\int_{t_o}^t d \ln(S) = \int_{t_o}^t \left(\mu - \frac{\sigma^2}{2} \right) dt + \int_{t_o}^t \sigma \cdot dW(t)$$

which gives:

$$\ln \left(\frac{S_t}{S_o} \right) = \left(\mu - \frac{\sigma^2}{2} \right) (t - t_o) + \sigma \cdot \sqrt{t - t_o} \cdot Z(0, 1)$$

where the noise term makes use of the definition of the integral of a Wiener process, and the properties of Normal random variables. Finally, we exponentiate both sides of the equation to obtain the analytical solution to the GBM model:

$$S(t) = S_o \exp \left[\left(\mu - \frac{\sigma^2}{2} \right) (t - t_o) + (\sigma \sqrt{t - t_o}) \cdot Z_t(0, 1) \right] \quad (15)$$

where S_o denotes the share price at t_o , and $Z_t(0, 1)$ denotes a standard normal random variable at time t . The expectation and variance of the GBM model is given by:

$$\begin{aligned} \mathbb{E}(S_t) &= S_o \cdot \exp(\mu \cdot \Delta t) \\ \text{Var}(S_t) &= S_o^2 e^{2\mu \cdot \Delta t} \left[e^{\sigma^2 \Delta t} - 1 \right] \end{aligned}$$

where $\Delta t = t - t_o$.

3.2 Estimating the drift parameter μ

Let's assume that we have a time series of share price values $S(t_1), S(t_2), \dots, S(t_k)$ and we want to estimate the deterministic growth of the share price, i.e., the drift parameter μ . There are several ways to do this, but we will use a deterministic linear model of the natural log of the share price values. To estimate the deterministic component of the share price, we first set the volatility parameter $\sigma = 0$ in Eqn. 15. Then, at some future time t , the share price (after some algebra) is given by:

$$\ln S_i = \ln S_o + \mu \cdot (t_i - t_o) \quad (16)$$

where $\ln S_*$ denotes the natural log of the share price at time t_* . Equation 16 is a linear model of the form $y = mx + b$, where $y = \ln S_i$, $x = t_i - t_o$, $m = \mu$, and $b = \ln S_o$. Thus, we can estimate

the growth parameter μ by fitting a linear model to the log of the share price values by solving an overdetermined system of linear equations.

Let \mathbf{A} denote a $S \times 2$ matrix, where each row corresponds to a time value $t > 0$. The first column of \mathbf{A} is all 1's while the second column holds the $(t_k - t_o)$ values. Further, let \mathbf{Y} denote the natural log of the share price values (in the same temporal order as the \mathbf{A} matrix). Then, the y-intercept and slope (drift parameter) can be estimated by solving the overdetermined system of equations:

$$\mathbf{A}\theta + \epsilon = \mathbf{Y}$$

where θ denotes the vector of unknown parameters (the y-intercept and slope), and ϵ denotes an error model, e.g., a normal distribution with mean zero and variance σ_ϵ^2 . This system can be solved as:

$$\theta = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{Y} - (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \epsilon$$

where \mathbf{A}^T denotes the transpose of the matrix \mathbf{A} , and $(\mathbf{A}^T \mathbf{A})^{-1}$ denotes the inverse of the square matrix product $\mathbf{A}^T \mathbf{A}$. Finally, we can estimate the error term ϵ by calculating the residuals:

$$\epsilon = \mathbf{Y} - \mathbf{A}\theta$$

and then fitting a normal distribution to the residuals, using some technique such as maximum likelihood estimation, to compute the uncertainty in the estimate of the drift parameter $\hat{\mu}$ (where the $\hat{\star}$ denotes an estimate of the parameter).

3.3 Estimating the volatility parameter σ

Generally, methods to estimate the volatility parameter can be classified into two categories - historical volatility estimates based on past return data and future volatility estimates based on the Implied Volatility (IV) of options contracts. Here, we estimate the volatility σ from historical data, i.e., from the past returns of the share price. A return refers to the increase or decrease in the price of an asset, e.g., shares of a stock, over a specific period, e.g., minutes, days, weeks, or even years. Assume the share price of a stock at time t is given by $S(t)$. Then, if we assume continuous compounding, the price of the stock at time $t + \Delta t$ is given by:

$$S(t + \Delta t) = S(t) \cdot \exp(\mu \cdot \Delta t) \quad (17)$$

where μ denotes the instantaneous growth rate (units: inverse years) of the share price over the time horizon $t \rightarrow t + \Delta t$. The continuously compounded share price gives rise to the logarithmic return model (Defn. 3):

Definition 3: Logarithmic Return

Let $S_{i,t-1}$ denote the continuously compounding share price of ticker i at time $t - 1$. Then, at time t , the share price for ticker i , denoted as $S_{i,t}$, is given by:

$$S_{i,t} = S_{i,t-1} \cdot \exp\left(\mu_{t,t-1}^{(i)} \cdot \Delta t\right) \quad (18)$$

where $\mu_{t,t-1}^{(i)}$ denotes the instantaneous growth rate (units: inverse years) of ticker i over time horizon $(t-1) \rightarrow t$, and Δt is the time interval between $t-1$ and t (units: years). The logarithmic return on asset i over time horizon $(t-1) \rightarrow t$, denoted by $r_{t,t-1}^{(i)}$, is defined as:

$$r_{t,t-1}^{(i)} \equiv \ln \left(\frac{S_{i,t}}{S_{i,t-1}} \right) = \mu_{t,t-1}^{(i)} \cdot \Delta t \quad (19)$$

Given Defn. 3, we compute the logarithmic return distribution using Algorithm 4. For each stock i , we compute the logarithmic return $\mu_{t,t-1}^{(i)}$ for each time interval $(t-1) \rightarrow t$, and then compute the mean and variance of the logarithmic return distribution. Once we estimated the logarithmic

Algorithm 4 Logarithmic Excess Growth Rate

Require: data set $\mathcal{D}_i = \{S_{i,t}\}_{t=1}^N \in \mathcal{D}$ where $S_{i,t}$ denotes the price of stock i at time t , all stocks have the same time horizon $N \gg 2$, and \mathcal{D} denotes the data set of all stocks.

Require: The time interval Δt between t and $t-1$ (units: years), and a list of stocks $\mathcal{L} = \{i\}_{i=1}^M$ where $M = \dim \mathcal{L}$.

Require: The risk-free rate r_f (units: inverse years).

```

1: procedure LOG GROWTH RATE( $\mathcal{D}, \mathcal{L}, \Delta t, r_f$ )
2:    $N \leftarrow \text{length}(\mathcal{D})$                                  $\triangleright$  Number of trading days for each stock  $i \in \mathcal{L}$ 
3:   for  $i \in \mathcal{L}$  do
4:      $\mathcal{D}_i \leftarrow \mathcal{D}[i]$                                  $\triangleright$  Select the data for stock  $i$  from the dataset collection  $\mathcal{D}$ 
5:     for  $t = 2 \rightarrow N$  do
6:        $S_{i,t-1} \leftarrow \mathcal{D}_i[t-1]$                          $\triangleright$  Select the price of stock  $i$  at time  $t-1$ 
7:        $S_{i,t} \leftarrow \mathcal{D}_i[t]$                              $\triangleright$  Select the price of stock  $i$  at time  $t$ 
8:        $\mu_{t,t-1}^{(i)} \leftarrow (1/\Delta t) \cdot \ln(S_{i,t}/S_{i,t-1}) - r_f$   $\triangleright$  Set  $r_f = 0$  for regular growth rate
9:     end for
10:  end for
11:  return  $\mu^{(1)}, \dots, \mu^{(\dim \mathcal{L})}$                      $\triangleright$  Return the logarithmic growth rate array for each stock  $i \in \mathcal{L}$ 
12: end procedure
```

mic growth distribution for each stock, we can estimate the volatility parameter σ by computing the standard deviation of the return distribution. Typically, we use the daily data, but we want the annualized standard deviation of the logarithmic return distribution, thus we multiply the daily standard deviation by $\sqrt{252}$, where 252 is the average number of trading days in a year, i.e, $\sigma \leftarrow \sqrt{252} \cdot \sigma$.

3.4 Stylized facts

Geometric Brownian motion is a widely used as a pricing model. However, whether geometric Brownian motion replicates many of the statistical properties of actual pricing and return data is unclear. These properties, referred to as *stylized facts* have been observed for decades, dating back to early work of Mandelbrot [4, 5] and later by Cont [6] and more recently by Ratliff-Crain et al. [7] who reviewed the original stylized facts proposed by Cont using newer data.

Computing and understanding the stylized facts of financial data is important for several reasons. However, here we'll focus on only two, namely, the shape of the return distribution and the autocorrelation of the returns. The shape of the return distribution is important because it is used

to compute the probability of extreme events. For example, the probability of a 10% drop in the share price of a firm in a single day. Typically, we assume that the return distribution is normal, but this is not always the case. Instead, returns are often leptokurtic, i.e., they have fat tails, for example following a Laplace distribution. The autocorrelation of the returns is important because it is used to compute the probability of a run of consecutive up or down days. If the returns are uncorrelated in time, then consecutive up or down days are independent events, i.e., the return (up or down) on day t is independent of the return on day $t - 1$. Thus, the market is behaving like a random walk. However, if the returns are correlated in time, then consecutive up or down days are not independent events, i.e., past returns influence future returns. This is a very active area of research, and we will not be able to cover all of the stylized facts here. However, the lack of autocorrelation goes to the heart of the efficient market hypothesis, which is at the core of modern finance theory, and one of the foundation principles of quantitative finance over the last 50 years.

In the example and project for this module, we will compute the stylized facts of the share price of firms in the S&P 500 index. In particular, we will compute the shape of the return distribution and the autocorrelation of the returns. We'll show that the returns are leptokurtic, for the majority of the firms in the S&P 500 index, and that the returns are uncorrelated in time.

3.5 Summary

In this module we introduced the single asset geometric Brownian motion model, and developed analytical solutions to the model. This model is a continuous-time analog of the discrete-time binomial lattice model we developed previously. The GBM model requires two parameters, the drift parameter μ and the volatility parameter σ . We developed methods to estimate these parameters from historical data, and then used these parameters to simulate the share price of firms in the example and project for this module. Finally, we discussed the stylized facts of financial data, and particularly the shape of the return distribution and the autocorrelation of the returns. The shape of the return distribution is important because it is used to compute the probability of extreme events, while the autocorrelation of the returns is important because it is used to compute the correlation in time of the returns.

4 Multiple Asset Geometric Brownian Motion Models

Geometric Brownian motion (GBM) is a continuous-time stochastic model in which the random variable $S(t)$, e.g., the share price of a firm is described by a stochastic differential equation. Geometric Brownian motion was popularized as a financial model by Samuelson in the 1950s and 1960s [2], but is arguably most commonly associated with the Black–Scholes options pricing model, which we will describe later [3]. Previously, we considered the single asset GBM model, where a firm's share price was modeled as a deterministic drift term corrupted by a Wiener noise process. We showed that a firm's share price follows a log-normal distribution and derived the analytical solution for a firm's share price at a future point in time. However, in practice, investors often hold portfolios of assets, and a firm's share price is correlated with other firms' share price. Thus, let's consider the GBM for multiple simultaneous assets, where the share price of a firm is modeled as a deterministic drift term that is corrupted by a Wiener noise process (same as

before), but now we consider how the noise is correlated with other firms in a portfolio.

Consider an asset portfolio \mathcal{P} , e.g., a collection of equities with a logarithmic return covariance matrix Σ and drift vector μ . The multi-dimensional geometric Brownian motion model describing the share price $S_i(t)$ for asset $i \in \mathcal{P}$ at time t is given by:

$$\frac{dS_i(t)}{S_i(t)} = \mu_i dt + \sum_{j=1}^{\mathcal{P}} a_{ij} \cdot dW_j(t) \quad \text{for } i = 1, 2, \dots, \mathcal{P}$$

where $a_{ij} \in \mathbf{A}$ and $\mathbf{A}\mathbf{A}^\top = \Sigma$ are noise coefficients describing the connection between firms i and firms j in the portfolio \mathcal{P} , and μ_i denotes the drift parameter for asset i in the portfolio \mathcal{P} . The multi-dimensional GBM model has the analytical solution:

$$S_i(t_k) = S_i(t_{k-1}) \cdot \exp \left[\left(\mu_i - \frac{\sigma_i^2}{2} \right) \Delta t + \sqrt{\Delta t} \cdot \sum_{j \in \mathcal{P}} a_{ij} \cdot Z_j(0, 1) \right] \quad i \in \mathcal{P}$$

where $S_i(t_{k-1})$ is the share price for firm $i \in \mathcal{P}$ at time t_{k-1} , the term $\Delta t = t_k - t_{k-1}$ denotes the time difference (step-size) for each time step (fixed), and $Z_j(0, 1)$ is a standard normal random variable for firm $j \in \mathcal{P}$. For more information on the derivation of the multi-dimensional GBM model, see Glasserman [8].

4.1 Drift and Covariance Estimation

The drift vector μ and covariance matrix Σ are critical parameters in the multi-dimensional GBM model. We can estimate the drift vector μ , which will be a $\dim \mathcal{P} \times 1$ vector, from historical data by computing the logarithmic excess growth rate of the assets in the portfolio \mathcal{P} , as we have shown previously; we repeat the process for each asset in the portfolio \mathcal{P} . On the other hand, the covariance matrix, a $\dim \mathcal{P} \times \dim \mathcal{P}$ symmetric matrix, describes the relationship between the logarithmic return series of firms i and j . The (i, j) element of the covariance matrix $\sigma_{ij} \in \Sigma$ is given by:

$$\sigma_{ij} = \text{COV}(r_i, r_j) = \sigma_i \sigma_j \rho_{ij} \quad \text{for } i, j \in \mathcal{P}$$

where σ_* denote the standard deviation of the logarithmic return of asset $*$, i.e., the volatility parameter, and ρ_{ij} denotes the correlation between the returns of asset i and j in the portfolio \mathcal{P} .

The noise coefficients a_{ij} modify the noise terms in the multi-dimensional GBM model, and describe the connection between firms i and firms j in the portfolio \mathcal{P} . The noise coefficients are given by the Cholesky decomposition of the covariance matrix Σ :

$$\Sigma = \mathbf{A}\mathbf{A}^\top$$

where \mathbf{A} is a lower triangular matrix, and \mathbf{A}^\top denotes the complex conjugate transpose of \mathbf{A} . The diagonal elements of the Cholesky factor \mathbf{A} are the standard deviations of the logarithmic return series of the assets in the portfolio \mathcal{P} , i.e., the volatility of firm i . On the other hand, the off-diagonal elements of the Cholesky factor \mathbf{A} can be used to compute σ_{ij} , i.e., the covariance between the logarithmic return series of the assets in the portfolio \mathcal{P} . The Cholesky decomposition of the covariance matrix Σ is unique and can be computed using the Cholesky decomposition

algorithms provided by many common numerical linear algebra libraries.

4.2 Multiasset Portfolio Simulations

We simulate the multi-dimensional GBM model using the analytical solution for the share price of each firm in the portfolio \mathcal{P} , given the initial share price $S(t_0)$ vector, the drift vector μ , the covariance matrix Σ , along with user-defined time points t_0, t_1, \dots, t_N , and the number of samples to simulate N_{samples} .

Definition 4: Distribution of Share Price

The share price of each firm in the portfolio \mathcal{P} at time t follows a log-normal distribution:

$$S(t) \sim \text{LogNormal}(\mu_t, \Sigma_t)$$

where μ_t denotes the mean log-transformed share price vector, and Σ_t denotes the covariance matrix of the log-transformed share prices at time t . A log-normal distribution is a right-skewed distribution. It is often used to model the share price because of its positive support and its ability to model the exponential growth of the share price.

We simulate the multi-dimensional GBM model using Algorithm 5, and then fit a distribution to the simulated share price of each firm in the portfolio \mathcal{P} (Defn. 4).

4.3 Computing the Net Present Value (NPV) of a portfolio \mathcal{P}

To compute the Net Present Value (NPV) of a portfolio \mathcal{P} , we can use Algorithm 5 to simulate the share price of each firm in \mathcal{P} over a time-horizon, e.g., from now to until next year. Given the share price of each firm over that horizon, we can compute what combinations of shares of each firm in the portfolio \mathcal{P} are the best in some way. For example, we can compute which share combinations lead to the largest projected NPV of the portfolio \mathcal{P} at time t by computing the two cash flow events that occur, first the purchase of the shares of each firm in the portfolio \mathcal{P} at time t_0 , and then the liquidation of the portfolio \mathcal{P} at time T , where the future share price value is discounted back to the present value using an appropriate discount rate, such as the average risk-free rate over the period \bar{r}_f :

$$\text{NPV}(T, \bar{r}_f) = \sum_{i \in \mathcal{P}} n_i \cdot \left(\mathcal{D}_{T,0}^{-1}(\bar{r}_f) \cdot S_i(T) - S_i(0) \right) \quad (20)$$

where n_i denotes the number of shares of firm $i \in \mathcal{P}$, $S_i(T)$ denotes the share price of firm $i \in \mathcal{P}$ at time T , $S_i(0)$ denotes the share price of firm $i \in \mathcal{P}$ at time 0, and $\mathcal{D}_{T,0}(\bar{r}_f)$ denotes the multiperiod discount factor between time $0 \rightarrow T$ with the discount rate equal to the risk-free rate \bar{r}_f .

To calculate the net present value at $t = T$, we need the share price of each firm in the portfolio \mathcal{P} at time T and the number of shares of each firm in the portfolio \mathcal{P} at time t_0 , where we assume the investor does not buy or sell any shares of any firm in the portfolio \mathcal{P} over the period $t_0 \rightarrow T$. The number of shares of each firm in the portfolio \mathcal{P} at time t is given by the fraction of the total wealth invested in each firm in the portfolio \mathcal{P} . Let's imagine a scenario where an investor has a

Algorithm 5 Multiasset Geometric Brownian Motion

Require: The initial share price vector $\mathbf{S}(t_0) = \{S_i(t_0)\}_{i=1}^{\dim \mathcal{P}}$ for each firm $i \in \mathcal{P}$.

Require: The drift vector $\mu = \{\mu_i\}_{i=1}^{\dim \mathcal{P}}$ for each firm $i \in \mathcal{P}$.

Require: The covariance matrix Σ for the portfolio \mathcal{P} .

Require: The time step Δt (units: years), the initial time t_o , the final time t_f , and the number of samples N_s .

Require: A Cholesky decomposition function `cholesky(Σ)` that returns the Cholesky factor \mathbf{A} .

```

1: procedure MULTIASSET GEOMETRIC BROWNIAN MOTION

2:    $N_a \leftarrow \text{length}(\mu)$                                 ▷ Number of assets in the portfolio  $\mathcal{P}$ 
3:    $N_t \leftarrow (t_f - t_o)/\Delta t$                         ▷ Number of time steps
4:    $T \leftarrow \text{arange}(t_o, t_f, \Delta t)$                 ▷ Time points
5:    $X \leftarrow \text{zeros}(N_s, N_t, N_a + 1)$                 ▷ Pre-allocate the share price array
6:    $\mathcal{Z} \leftarrow \mathcal{N}(0, 1)$                             ▷ Instantiate a standard normal distribution
7:    $\mathbf{A} \leftarrow \text{cholesky}(\Sigma)$                     ▷ Cholesky decomposition of the covariance matrix  $\Sigma$ 

8:   for  $i \in 1$  to  $N_s$  do                                ▷ Loop over the number of samples
9:     for  $j \in \text{eachindex}(T)$  do                          ▷ Loop over the time points
10:       $X_{i,j,1} \leftarrow T_j$                             ▷ Set the time points in first column
11:    end for

12:    for  $j \in 1$  to  $N_a$  do                                ▷ Loop over the number of assets in the portfolio  $\mathcal{P}$ 
13:       $X_{i,1,j+1} \leftarrow S_{o,j}$                         ▷ Set the initial share price for asset  $j$ 
14:    end for

15:    for  $j \in 2$  to  $N_t$  do                                ▷ Loop over the number of time steps
16:      for  $k \in 1$  to  $N_a$  do                                ▷ Loop over the number of assets in the portfolio  $\mathcal{P}$ 

17:        noise  $\leftarrow 0.0$ 
18:        for  $l \in 1$  to  $N_a$  do                            ▷ Loop over the number of assets in the portfolio  $\mathcal{P}$ 
19:          noise  $\leftarrow \text{noise} + \mathbf{A}_{k,l} \cdot \text{rand}(\mathcal{Z})$   ▷ Compute the noise term for asset  $k$ 
20:        end for

21:         $X_{i,j,k+1} \leftarrow X_{i,j-1,k+1} \cdot \exp \left[ \left( \mu_k - a_{kk}^2/2 \right) \cdot \Delta t + \sqrt{\Delta t} \cdot \text{noise} \right]$ 
22:      end for
23:    end for
24:  end for
25: end procedure

```

total wealth of W_{total} at time t , and the fraction of the total wealth invested in each firm $i \in \mathcal{P}$ is given by $0 \leq \omega_i \leq 1$:

$$\omega_i(t) = \frac{W_i(t)}{W_{\text{total}}} \quad \text{for } i \in \mathcal{P}$$

where $W_i(t)$ denotes the wealth invested in firm $i \in \mathcal{P}$ at time t . However, we know the total wealth

of portfolio \mathcal{P} , is just the liquidation value of the portfolio:

$$W(t) = \sum_{i \in \mathcal{P}} n_i(t) \cdot S_i(t)$$

where $n_i(t)$ denotes the number of shares of firm $i \in \mathcal{P}$ at time t , and $S_i(t)$ denotes the share price of firm $i \in \mathcal{P}$ at time t . We can rewrite the number of shares of each firm in the portfolio \mathcal{P} at time t as:

$$n_i(t) = \frac{\omega_i(t) \cdot W_{\text{total}}}{S_i(t)} \quad \text{for } i \in \mathcal{P}$$

Further, we can rewrite the fraction of the total wealth invested in each firm $i \in \mathcal{P}$ as:

$$\omega_i(t) = \frac{n_i(t) \cdot S_i(t)}{\sum_{k \in \mathcal{P}} n_k(t) \cdot S_k(t)} \quad \text{for } i \in \mathcal{P} \quad (21)$$

The fractions $\omega_i(t)$, i.e., the portfolio weights at time t , have a few interesting properties (Concept 1).

Concept 1: Properties of allocation ω

The fraction of the total wealth invested in each firm $i \in \mathcal{P}$ at time t , in the absence of borrowing shares, sums to one for all time points t :

$$\sum_{i \in \mathcal{P}} \omega_i(t) = 1 \quad \forall t \geq t_0$$

Further, because the share price of each firm in the portfolio \mathcal{P} is positive, the fraction of the total wealth invested in each firm $i \in \mathcal{P}$ at time t is non-negative $\omega_i(t) \geq 0$ for all $t \geq t_0$. Finally, ω_i , even with constant shares, is a function of time as the share price of each firm in the portfolio \mathcal{P} changes over time.

The investor can choose the fraction of the total wealth ω_i invested in each firm $i \in \mathcal{P}$ by some approach, e.g., by formulating the choice as an optimization problem or by sampling from a probability distribution such as a Dirchlet distribution $\omega_i \sim \text{Dirchlet}(\alpha_1, \alpha_2, \dots, \alpha_{\dim \mathcal{P}})$, where α_i denotes the concentration parameter for firm $i \in \mathcal{P}$. Then, when combined with the multi-dimensional GBM model, the investor can simulate the share price of each firm in the portfolio \mathcal{P} over a time period and compute the NPV of the portfolio. This process can be repeated many times, and the expected NPV of the portfolio \mathcal{P} at time T can be computed by averaging the NPV over all the samples (Algorithm 6).

4.4 Summary

In this module, we have considered the multi-dimensional GBM model, which describes the share price of firms in a portfolio \mathcal{P} . We demonstrated how to estimate the drift vector μ , the covariance matrix Σ , and the noise coefficients a_{ij} , and provided an algorithm to simulate the multi-dimensional GBM model. Finally, we introduced the concept of the Net Present Value (NPV) of

Algorithm 6 Monte-Carlo Portfolio Net Present Value

Require: The initial share price vector $\mathbf{S}(t_0) = \{S_i(t_0)\}_{i=1}^{\dim \mathcal{P}}$ for each firm $i \in \mathcal{P}$.

Require: Share price distribution $S(T) \sim \text{LogNormal}(\mu_T, \Sigma_T)$

Require: An allocation distribution function $\omega \sim \text{Dir}(\alpha_1, \alpha_2, \dots, \alpha_{\dim \mathcal{P}})$, where we assume equal concentration parameters $\alpha_i = 1/\dim \mathcal{P}$ firm $i \in \mathcal{P}$.

Require: The average risk-free rate \bar{r}_f , the number of samples N_s and the total wealth W_{total} .

```

1: procedure PORTFOLIO SIMULATION

2:   NPV  $\leftarrow$  zeros( $N_s, \dim \mathcal{P} + 1$ ) ▷ Pre-allocate the NPV array
3:   for  $i \in 1$  to  $N_s$  do ▷ Loop over the number of samples
4:      $\mathbf{S} \leftarrow S(T)$  ▷ Sample the share price of each firm in the portfolio  $\mathcal{P}$  at time  $T$ 
5:      $\omega \leftarrow \text{Dir}(\alpha)$  ▷ Sample allocation  $\omega$  at  $t = t_0$ 

6:      $n \leftarrow$  zeros( $\dim \mathcal{P}$ ) ▷ Pre-allocate the number of shares array
7:     for  $j \in 1$  to  $\dim \mathcal{P}$  do ▷ Loop over the number of assets in the portfolio  $\mathcal{P}$ 
8:        $n_j \leftarrow \omega_j \cdot W_{\text{total}} \cdot (S_j(t_0)^{-1})$  ▷ Compute the number of shares of firm  $j$  at time  $t = t_0$ 
9:     end for

10:    NPV $_{i,1} \leftarrow \sum_{k \in \mathcal{P}} n_k \cdot (\mathcal{D}_{T,0}^{-1}(\bar{r}_f) \cdot S_k(T) - S_k(0))$  ▷ NPV of  $\mathcal{P}$  at  $T$ , store column 1
11:    for  $j \in 1$  to  $\dim \mathcal{P}$  do ▷ Loop over the number of assets in the portfolio  $\mathcal{P}$ 
12:      NPV $_{i,j+1} \leftarrow \omega_j$  ▷ Store the portfolio weights in columns 2,  $\dots$ ,  $\dim \mathcal{P} + 1$ 
13:    end for
14:  end for

15: end procedure

```

a portfolio \mathcal{P} . We presented an algorithm to compute the NPV of a portfolio \mathcal{P} at time T , where we sampled the fraction of the total wealth ω_i invested in each firm $i \in \mathcal{P}$ from a Dirichlet distribution and the share price at $t = T$ from a LogNormal distribution. Later, we'll develop tools to compute optimal values of these fractions, i.e., the portfolio weights, but for now, we considered the case where the portfolio weights are chosen by the investor by some approach (perhaps even randomly).

References

1. Cox JC, Ross SA, Rubinstein M. Option pricing: A simplified approach. *Journal of Financial Economics*. 1979;7(3):229–263. doi:[https://doi.org/10.1016/0304-405X\(79\)90015-1](https://doi.org/10.1016/0304-405X(79)90015-1).
2. Merton RC. Paul Samuelson And Financial Economics. *The American Economist*. 2006;50(2):9–31.
3. Black F, Scholes M. The Pricing of Options and Corporate Liabilities. *Journal of Political Economy*. 1973;81(3):637–654.
4. Mandelbrot B. The Variation of Certain Speculative Prices. *The Journal of Business*. 1963;36(4):394–419.

5. Mandelbrot B. The Variation of Some Other Speculative Prices. The Journal of Business. 1967;40(4):393–413.
6. Cont R. Empirical properties of asset returns: stylized facts and statistical issues. Quantitative Finance. 2001;1(2):223–236.
7. Ratliff-Crain E, Oort CMV, Bagrow J, Koehler MTK, Tivnan BF. Revisiting Stylized Facts for Modern Stock Markets; 2023.
8. Glasserman P. Monte Carlo methods in financial engineering. vol. 53. New York: Springer; 2004. Available from: <http://www.loc.gov/catdir/enhancements/fy0813/2003050499-d.html>.