This module is in two parts:

Part I  Financial data analysis, taught by Prof. M. Niranjan
Part II  Crypto-currencies and blockchain technology, taught by Dr Jie Zhang

Spring Semester 2017/2018

Financial Equilibrium
Caution: A peculiar and rather personal view

Generate products and services

In need of

- stability against fluctuations (e.g. demand, exchange rate)
- capital investment (e.g. to modernise, grow)

- Process wealth & capital
- Driven by gambling instinct and greed
The Setting

- Finance gets bad publicity; bankers and fund managers are sometimes disliked
- The system can fail badly
- When the system fails, large amounts of tax-payer money is used to bail them out. *I don’t like this!*
- Yet the system is useful
  - Investors interested in future returns
  - Greed?
  - Pay for retirement
  - Firms / Governments looking to raise capital for investment
  - Companies looking for stability; e.g. insure against exchange rate fluctuation
- What are the sources of computational problems?
  - Time - present value of money.
  - Uncertainty - of the future.

Overview of the Module

Topics in Part I: Financial Data Analysis

- Portfolio Optimization
- Derivatives Pricing

Keywords:

Mean-Variance optimization, Linear and quadratic programming, Multivariate Gaussian distribution, Constrained optimization, Value at risk and Conditional value at risk, Sharpe ratio, Present value, Stochastic differential equations, Ito’s Lemma, Black-Scholes model, Options pricing, Stochastic Simulations and Monte Carlo methods.
plus several academic papers.

### Financial Instruments (broad classes)

- **Bonds**
  - Debt instrument to raise capital; delivers periodic payment *(coupon)*; has a *face value* on *maturity*. No ownership associated.

- **Stocks**
  - Own a small *share* of a company; the ownership may be traded in the market; owning the share might earn *dividends*.

- **Derivatives**
  - Contracts written on the basis of a future value of a a stock, currency etc. Usually there is a time of *maturity* and a promised *payoff* in the contract. Variations in style of *exercising* the contract.
Wealth $W_0$ deposit in bank and get $W_1$ after one year

$W_1 = (1 + r) W_0$, \quad r \text{ interest rate}$

Compound interest over $n$ years: $W_n = (1 + r)^n W_0$

Define interest rate as $r$ per year; allow compounding at $m$ intervals within the year

$$W_1 = \left(1 + \frac{r}{m}\right)^m W_0$$

Continuous compounding $m \to \infty$

$$W_1 = \exp(r) W_0$$

Present value of your promise to give me cash $C$ in time $t$ is

$$\exp(-rt) C$$

Various Topics We Will Learn

Part I (Topic I): Portfolio Optimization

Portfolios:

- Notion of expected return and risk in investing - balancing it out
- Investing in a portfolio of assets, than in a single asset - “not all eggs in one basket”
- Optimization techniques we will learn and use
  - Linear programming
  - Quadratic programming
  - (Second order cone programming)
  - Inducing sparsity – $l_1$ or lasso regularization
  - Convex optimization using CVX toolbox
Various Topics We Will Learn (cont’d)
Part I (Topic II): Derivatives Pricing

Derivatives Pricing (contract in the future, in an uncertain world):
- Brownian motion, Geometric Brownian motion
- Stochastic differential equations

\[
\frac{dS}{S} = \mu \, dt + \sigma \, dZ
\]
\[
dZ = \phi \sqrt{dt}, \quad \phi \sim (0, 1)
\]

- Ito’s Lemma: Function of a Geometric Brownian Motion

\[
dG = \left( \mu S \frac{\partial G}{\partial S} + \frac{\sigma^2 S^2 \frac{\partial^2 G}{\partial S^2}}{2} + \frac{\partial G}{\partial t} \right) dt + \sigma S \frac{\partial G}{\partial S} \, dZ
\]

- Black-Scholes: options pricing under specific assumptions
- Monte Carlo / Stochastic simulations: general cases

Part I: Portfolio Optimization

- \( r_i(t) \) Return on asset \( i \) at time \( t \); i.e. invest at time \( t-1 \), what have you earned at time \( t \)?
- We think of this as a random variable, say Gaussian distributed with mean \( \mu_i \) and variance \( \sigma_i^2 \)
- The mean is what we expect (on average) to gain by investing
- We think of variance in return as risk
- When we look at more than one asset, we can think of how returns on them are correlated: \( \sigma_{ij} \)
- A portfolio (investment in \( N \) assets) with relative weights \( \pi_i \)
- Return on the portfolio: \( r_p = \sum_{i=1}^{N} \pi_i \, r_i = \pi^t \, r \)
- Returns on the portfolio have a multivariate Gaussian distribution

\[
\begin{bmatrix}
    r_1 \\
    r_2 \\
    \vdots \\
    r_N
\end{bmatrix}
= \begin{bmatrix}
    \mu_1 \\
    \mu_2 \\
    \vdots \\
    \mu_N
\end{bmatrix}
\begin{bmatrix}
    \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1N} \\
    \sigma_{12} & \sigma_{22} & \cdots & \sigma_{2N} \\
    \vdots & \vdots & \ddots & \vdots \\
    \sigma_{1N} & \sigma_{2N} & \cdots & \sigma_{NN}
\end{bmatrix}
\]
Gaussian Density

\[ p(x) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left\{ -\frac{(x - m)^2}{2\sigma^2} \right\} \]

\[ x = \text{linspace}(-5,5,50); \]
\[ m = 0; \]
\[ s = 1; \]
\[ y = \text{normpdf}(x,m,s); \]
\[ \text{figure(1), clf} \]
\[ \text{plot}(x,y,'LineWidth',3); \]
\[ \text{grid on} \]

Multivariate Gaussian Distribution

\[ p(x) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right\} \]

- Parameters: mean vector \( \mu \) covariance matrix \( \Sigma \)

- How do these shapes change with \( \mu \) and \( \Sigma \)?
### Portfolio Return

- Linear transform of multivariate Gaussian
  \[ x \sim \mathcal{N}(\mu, \Sigma); \quad y = Ax \quad \Rightarrow \quad y \sim \mathcal{N}(A\mu, A\Sigma A^T) \]

- Return on our portfolio is a linear transform of the vector of returns
  \[ r_P = \pi^T r \]

- We can immediately write down the distribution of the return on the portfolio
  \[ r_P \sim \mathcal{N}(\pi^T \mu, \pi^T \Sigma \pi) \]

- Mean return \( M = \pi^T \mu \) and the variance on it \( V = \pi^T \Sigma \pi \)
- When \( \pi \) changes, \( M \) and \( V \) change — how?

### Efficient Portfolio

- As we change \( \pi \) (i.e. invest in different proportions), \( M \) and \( V \) change
- Not all \( M \) and \( V \) are realizable.
- We can formulate constrained optimization problems
- For a given risk we tolerate, what is the highest return we can expect
  \[ \max_{\pi} \pi^T \mu \quad \text{subject to} \quad \pi^T \Sigma \pi = \sigma_0 \]
- If we hope for (expect) a given return, at what minimum risk can we achieve it?
  \[ \min_{\pi} \pi^T \Sigma \pi \quad \text{subject to} \quad \pi^T \mu = r_0 \]
- Other constraints
  possible: \[ \sum_{i=1}^{N} \pi_i = 1, \quad \pi_i \geq 0, \quad \alpha \leq \pi_i \leq \beta \]
Estimation

- We estimate $\mu$ and $\Sigma$ from historic data and apply optimization to allocate assets.
- We hope the past might be a good reflection of future!
- Estimation:

$$
\hat{\mu} = \frac{1}{T} \sum_{t=1}^{T} r(t)
$$

$$
\hat{\Sigma} = \frac{1}{T} \sum_{t=1}^{T} (r(t) - \hat{\mu}) (r(t) - \hat{\mu})^T
$$

Solving quadratic program in MATLAB

$$
\min_x x^T H x + f^T x \quad \text{such that} \quad \begin{cases} 
A x \leq 0 \\
A_{eq} x = b_{eq} \\
lb \leq x \leq ub.
\end{cases}
$$

$$
x = \text{quadprog}(H, f, A, b, Aeq, beq, lb, ub, x0)
$$

Do `doc quadprog` in MATLAB and read more.

For the portfolio optimization problem, we might have:

$$
pi = \text{quadprog}(\Sigma, [], [], [], \mu', rMax, 0, 1, [])
$$

Map the problem variables to the function in the tool.
Efficient Frontier

- Given $\mu$ and $\Sigma$
- What portfolio has the highest return, unconstrained by risk?

$$\max \pi^T \mu \quad \text{subject to} \sum_{i=1}^{N} \pi_i = 1, \text{ and } \pi_i \geq 0$$

Linear Programming:

$$\min f^T x \text{ such that } \begin{cases} Ax \leq b \\ A_{eq} x = b_{eq} \\ lb \leq x \leq ub \end{cases}$$

$$x = \text{linprog}(f, A, b, A_{eq}, b_{eq}, lb, ub)$$
$$w_1 = \text{linprog}(-\mu, [], [], \text{ones}(1,N), 1, 0, 0);$$
$$r_1 = w_1 \ast \mu;$$

Efficient Frontier (cont'd)

- What portfolio has lowest variance (unconstrained by expectation)?

$$\min \pi^T \Sigma \pi \quad \text{subject to} \sum_{i=1}^{N} \pi_i = 1$$

$$w_2 = \text{quadprog}(\mu, \text{zeros}(N,1), [], [], \text{ones}(1,N), 1, \text{zeros}(N,1), [], []);$$
$$r_2 = w_2^T \ast \mu;$$

- Portfolios on the efficient frontier will have returns in range $r_1$ to $r_2$
- Now, select a number of points (returns we need), and find the minimum variance portfolios that will offer these returns!

$$M = \text{linspace}(r_1, r_2, p)$$
$$\text{for } j=1:p$$
$$\text{ret} = M(j);$$
$$w = \text{quadprog}(...) ;$$
$$V(j) = w^T \ast \Sigma \ast w;$$
$$\text{end}$$
function [PRisk, PRoR, PWts] = NaiveMV(ERet, ECov, NPts)
ERet = ERet(:); % makes sure it is a column vector
NAssets = length(ERet); % get number of assets
% vector of lower bounds on weights
V0 = zeros(NAssets, 1);
% row vector of ones
V1 = ones(1, NAssets);
% set medium scale option
options = optimset('LargeScale', 'off');
% Find the maximum expected return
MaxReturnWeights = linprog(-ERet, [], [], V1, 1, V0);
MaxReturn = MaxReturnWeights' * ERet;
% Find the minimum variance return
MinVarWeights = quadprog(ECov, V0, [], [], V1, 1, V0, [], [], options);
MinVarReturn = MinVarWeights' * ERet;
MinVarStd = sqrt(MinVarWeights' * ECov * MinVarWeights);
% check if there is only one efficient portfolio
if MaxReturn > MinVarReturn
    RTarget = linspace(MinVarReturn, MaxReturn, NPts);
    NumFrontPoints = NPts;
else
    RTarget = MaxReturn;
    NumFrontPoints = 1;
end

% Store first portfolio
PRoR = zeros(NumFrontPoints, 1);
PRisk = zeros(NumFrontPoints, 1);
PWts = zeros(NumFrontPoints, NAssets);
PRoR(1) = MinVarReturn;
PRisk(1) = MinVarStd;
PWts(1,:) = MinVarWeights(:)';
% trace frontier by changing target return
VConstr = ERet';
A = [V1 ; VConstr ];
B = [1 ; 0];
for point = 2:NumFrontPoints
    B(2) = RTarget(point);
    Weights = quadprog(ECov, V0, [], [], A, B, 0, [], [], options);
    PRoR(point) = dot(Weights, ERet);
    PRisk(point) = sqrt(Weights' * ECov * Weights);
    PWts(point, :) = Weights(:)';
end
Summary: what have we achieved?

Homework

- Three assets with the following properties:

  \[
  m = \begin{bmatrix} 0.1 \\ 0.2 \\ 0.15 \end{bmatrix}; \quad C = 100 \times \begin{bmatrix} 0.005 & -0.010 & 0.004 \\ -0.010 & 0.040 & -0.002 \\ 0.004 & -0.002 & 0.023 \end{bmatrix} \]

- Study the code of function NaiveMV and draw the efficient frontier.
- Use the function frontcon in MATLAB and draw the efficient frontier.
Estimation of Parameters

- Estimate parameters $\mu$ and $C$ from data within a window
- Optimize portfolio, invest and wait
- Periodically re-balance the portfolio (re-estimate parameters)
- Trade-off:
  - Need long window for accurate estimation
  - But relationships may not be stationary over long durations
  - Shrinkage in covariance estimates

Advances on the Mean-Variance Portfolio

- Do such portfolios make money?
  

- Including transaction costs into the optimization


- Forcing the portfolio to be sparse and stable


- Optimizing the execution of trade

  TBC
Portfolio Performance

- Sharpe Ratio: mean to standard deviation of portfolio return
  \[ S = \frac{m - r}{\sigma} \]
  
  \( r \) “risk free” interest rate

- Value at Risk (VAR): Value such that probability of loss exceeding this is 0.01.
  \[ V \text{ such that } P[-G > V] = 0.01 \]
  “if a portfolio of stocks has a one-day 5% VaR of 1 million, there is a 0.05 probability that the portfolio will fall in value by more than 1 million over a one day period”

\[ \text{ValueAtRisk} = \text{portvrisk}(\text{PortReturn, PortRisk, RiskThreshold, PortValue}) \]

- cVAR: Conditional Value at Risk (later)

Empirical Evaluation
DeMiguel, J. et al. (2009).

- Comparison of a number of portfolio optimization methods
  - \( \frac{1}{N} \) with re-balancing
  - Sample based mean-variance
  - Bayesian methods (of shrinking estimates)
  - Constraints
  - Combination of portfolios (model averaging / mixing)

- No method consistently beats the naive strategy!
Sparse Portfolios
Brodie et al. (2007) PNAS

- $N$ assets; $r_t$, return vector at time $t$
- Expected return and covariance:

\[
\begin{pmatrix}
  r_{1,t} \\
  r_{2,t} \\
  \vdots \\
  r_{N,t}
\end{pmatrix}
\]

\[E[r_t] = \mu \quad E[(r_t - \mu)(r_t - \mu)^T] = C\]

- Markowitz portfolio

\[
\begin{aligned}
\min_w & \quad w^T C w \\
\text{subject to} & \quad w^T \mu = \rho \quad \text{and} \quad 1_N^T w = 1
\end{aligned}
\]

- Short selling allowed; i.e. $w_j$ need not be positive
- Covariance: $C = E[r_t r_t^T - \mu \mu^T]$

---

Sparse Portfolios, Brodie et al. (2007) PNAS (cont’d)

- Mean and covariance estimated from data (expectations as sample averages):
- \( \hat{\mu} = \frac{1}{T} \sum_{t=1}^{T} r_t \)
- \( R \ T \times N \) matrix with rows as \( r_t^T \)
- Optimization problem rewritten as

\[
\begin{aligned}
\hat{w} = \min_w & \quad \frac{1}{T} \| \rho 1_T - R w \|_2^2 \\
\text{subject to} & \quad w^T \hat{\mu} = \rho, \ w^T 1_N = 1
\end{aligned}
\]

- Often there is strong correlation between returns
  - Assets in the same sector respond in similar ways
- Strong correlations make $R$ ill-conditioned \( \implies \) numerically unstable optimization
- Solution: regularization
- Brodie et al. suggest $l_1$ regularizer

\[
\begin{aligned}
\hat{w} = \min_w & \quad [ \| \rho 1_T - R w \|_2^2 + \tau \| w \|_1 ] \\
\text{subject to} & \quad w^T \hat{\mu} = \rho, \ w^T 1_N = 1
\end{aligned}
\]
Passive investor, wishing to get the same return as stock index (e.g. FTSE100)
Invest in all 100 stocks of the FTSE?
Transaction costs very high
Can we find a small subset of the 100 stocks (say 10), that will approximate the performance of the index?
Subset selection / cardinality constrained optimization

\[
\begin{align*}
\min_{w} \left[ \left\| y - Rw \right\|_2^2 \right] \\
\text{subject to } \left\| w \right\|_0 = w_0
\end{align*}
\]

0\text{th norm} \rightarrow \text{number of nonzero elements of } w \rightarrow \text{subset of assets}
The above is of combinatorial complexity
Suboptimal algorithm: greedy search

A convenient proxy to achieve sparsity is lasso (\(l_1\) constrained regression)

\[
\min_{w} \left[ \left\| y - Rw \right\|_2^2 + \tau \left\| w \right\|_1 \right]
\]

Several elements of \(w\) will be zero
Tune \(\tau\) to achieve different levels of sparsity
Can incorporate transaction costs into the optimization

\[
\min_{w} \left[ \left\| y - Rw \right\|_2^2 + \tau \sum_{i=1}^{N} s_i |w_i| \right]
\]

Transaction costs:
- Usually have fixed (overhead) part and transaction-dependent part
- Institutional investors fixed part negligible
- Small investors can assume fixed cost only
We are holding a portfolio \( w \)
- We want to make an adjustment \( \Delta_w \), new portfolio \( w + \Delta_w \)
- Transaction costs only on the adjustments

\[
\begin{align*}
\Delta_w &= \min_{\Delta_w} \left[ \|\rho \mathbf{1}_T - \mathbf{R}(w + \Delta_w)\|^2_2 + \tau \|\Delta_w\|_1 \right] \\
\text{subject to } \Delta_w^T \hat{\mu} &= 0 \text{ and } \Delta_w^T \mathbf{1}_N = 1
\end{align*}
\]

Homework:

Coursework 1 will involve confirming some claims in Brodie et al.’s paper. Please download the paper and start reading.
We will use the CVX toolbox within MATLAB to implement optimization
http://cvxr.com/
Download, uncompress, set MATLAB to the cvx directory and do cvx_setup
Take MATLAB back into your working directory

Example of using CVX

T = 150; N = 50;
R = randn(T, N);
rho = 0.02;
tau = 1;
mu = rand(N,1);

cvx_begin quiet
variable w(N)
    minimize( norm(rho*ones(T,1)-R*w) + tau*norm(w,1) )
    subject to
        w'*ones(N,1) == 1;
        w'*mu == rho;
        w > 0;
    cvx_end

figure(1), clf, bar(w); grid on

Note: Data random - probably won’t work all the time
Portfolio weights: $w = [w_1 \ w_2 \ \ldots \ w_n]^T$

Returns: $a$; $E[a] = \bar{a}$; $E[(a - \bar{a})(a - \bar{a})^T] = \Sigma$

We consider an adjustment to the portfolio of value $x$

New portfolio: $w + x$; Wealth: $a^T (w + x)$

Portfolio return and variance:

$$E[W] = \bar{a}^T (w + x)$$
$$E[(W - E[W])^2] = (w + x)^T \Sigma (w + x)$$

Transaction cost: $\phi(x)$

Budget Constraint:

$$1^T x + \phi(x) \leq 0$$

Possible Optimizations:

$$\maximize_{x} \ \bar{a}^T (w + x)$$
subject to
$$1^T x + \phi(x) \leq 0$$
$$w + x \in S$$

$S$ some feasible set (with other constraints)

$$\minimize_{x} \ \phi(x)$$
subject to
$$\bar{a}^T (w + x) \geq r_{\min}$$
$$w + x \in S$$
Lobo et al. (2007) (cont’d)

Modeling Transaction Costs

Costs are separable (usual assumption):

\[ \phi(x) = \sum_{i=1}^{n} \phi_i(x_i) \]

\[ \phi_i(x_i) = \begin{cases} \alpha_i^+ x_i, & x_i \geq 0 \\ -\alpha_i^- x_i, & x_i \leq 0 \end{cases} \]

- \( \alpha_i^+ \) and \( \alpha_i^- \) cost rates for buying and selling asset \( i \)
- Convex cost function
- \( \phi_i = \alpha_i^+ x_i^+ + \alpha_i^- x_i^- \) with \( x_i^+ \geq 0 \) and \( x_i^- \geq 0 \)

Fixed plus linear transaction costs:

\[ \phi_i(x_i) = \begin{cases} 0, & x_i = 0 \\ \beta_i^+ + \alpha_i^+ x_i, & x_i \geq 0 \\ \beta_i^- - \alpha_i^- x_i, & x_i \leq 0 \end{cases} \]

- This is non-convex

Diversification Constraints

- Limit the amount of investment in any asset

\[ w_i + x_i \leq p_i, \quad i = 1, 2, \ldots, n \]

- Limit the fraction of total wealth held in each asset

\[ w_i + x_i \leq 1^T (w + x) \]

- Limit exposure in any small group of assets (say in a sector)

\[ \sum_{i=1}^{r} (w_i + x_i)_i \leq 1^T (w + x) \]

( Tricky, but can show this is convex – see Eqn (11) in paper)
Lobo et al. (2007) (cont’d)

- Constraints on short-selling
  ... on individual asset

  \[ w_i + x_i \geq -s_i, \quad i = 1, \ldots, n \]

  ...or as bound on the total short position

  \[ \sum_{i=1}^{n} (w_i + x_i)_- \leq S \]

- Collateralization:

  \[ \sum_{i=1}^{n} (w_i + x_i)_- \leq \gamma \sum_{i=1}^{n} (w_i + x_i)_+ \]

  *What I have borrowed to sell is smaller than a fraction of what I own*

Lobo et al. (2007) (cont’d)

Variance:

\[ (\mathbf{w} + \mathbf{x})^T \mathbf{\Sigma} (\mathbf{w} + \mathbf{x}) \leq \sigma_{\text{max}} \]

Can also be written as:

\[ || \mathbf{\Sigma}^{1/2} (\mathbf{w} + \mathbf{x}) || \leq \sigma_{\text{max}} \]

- This is *Second Order Cone* constraint.
Lobo et al. (2007) (cont’d)

Shortfall Risk Constraint:

\[ P \left( W \geq W^{\text{low}} \right) \geq \eta \]

\[ W = a^T (w + x) \sim \mathcal{N}(\mu, \sigma^2) \]

\[ P \left( \frac{W - \mu}{\sigma} \leq \frac{W^{\text{low}} - \mu}{\sigma} \right) \leq 1 - \eta \]

But \((W - \mu)/\sigma \sim \mathcal{N}(0, 1)\)

Hence

\[ P \left( \frac{W - \mu}{\sigma} \leq \frac{W^{\text{low}} - \mu}{\sigma} \right) = \Phi \left( \left( \frac{W^{\text{low}} - \mu}{\sigma} \right) \right) \]

\[ \Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} \exp \left\{ -\frac{t^2}{2} \right\} dt \]

\[ \frac{W^{\text{low}} - \mu}{\sigma} \leq \Phi^{-1}(1 - \eta) \]

\[ \Phi^{-1}(1 - \eta) = -\Phi^{-1}(\eta) \]

\[ \mu - W^{\text{low}} \geq \Phi^{-1}(\eta) \sigma \]

Using \(\mu = \bar{a}^T (w + x)\) and \(\sigma^2 = (w + x)^T \Sigma (w + x)\)

\[ \Phi^{-1}(\eta) \left\| \Sigma^{1/2} (w + x) \right\| \leq \bar{a}^T (w + x) - W^{\text{low}} \]
Lobo *et al.* (2007) (cont’d)
Shortfall Risk on M-V Space
Lobo et al. (2007) (cont’d)

Question in Assignment 1

maximize \( \mathbf{a}^T (\mathbf{w} + \mathbf{x}^+ - \mathbf{x}^-) \)

subject to \( 1^T (\mathbf{x}^+ - \mathbf{x}^-) + \sum_{i=1}^{n} (\alpha_i^+ x_i^+ + \alpha_i^- x_i^-) \leq 0 \)

\( x_i^+ \geq 0, x_i^- \geq 0, i = 1, 2, ..., n \)

\( w_i + x_i^+ - x_i^- \geq s_i, i = 1, 2, ..., n \)

\( \Phi^{-1}(\eta_j) \| \Sigma^{1/2} (\mathbf{w} + \mathbf{x}^+ - \mathbf{x}^-) \| \leq \mathbf{a}^T (\mathbf{w} + \mathbf{x}^+ - \mathbf{x}^-) - W_{\text{low}}, j = 1, 2 \)

Puzzle

\[
\begin{align*}
m1 &= [0.15 \ 0.2 \ 0.08 \ 0.1]', \\
C1 &= [ 0.2 \hspace{1em} 0.05 \hspace{1em} -0.01 \hspace{1em} 0.0 \\
&\quad 0.05 \hspace{1em} 0.30 \hspace{1em} 0.015 \hspace{1em} 0.0 \\
&\quad -0.01 \hspace{1em} 0.015 \hspace{1em} 0.10 \hspace{1em} 0.0 \\
&\quad 0.0 \hspace{1em} 0.0 \hspace{1em} 0.0 \hspace{1em} 0.0 ]; \\
\end{align*}
\]

\[
\begin{align*}
m2 &= [0.15 \ 0.2 \ 0.08]', \\
C2 &= [ 0.2 \hspace{1em} 0.05 \hspace{1em} -0.01 \\
&\quad 0.05 \hspace{1em} 0.30 \hspace{1em} 0.015 \\
&\quad -0.01 \hspace{1em} 0.015 \hspace{1em} 0.10 ]; \\
\end{align*}
\]

\[
\begin{align*}
[V1, M1, PWts1] &= \text{NaiveMV}(m1, C1, 25); \\
[V2, M2, PWts2] &= \text{NaiveMV}(m2, C2, 25); \\
\end{align*}
\]

figure(2), clf, 
plot(V1, M1, ‘b’, V2, M2, ‘r’, 'LineWidth', 3), 
title('Mean Variance Portfolio', 'FontSize', 22) 
xlabel('Portfolio Risk', 'FontSize',18) 
ylabel('Portfolio Return', 'FontSize', 18);
Derivatives Pricing

- Efficiency, no-arbitrage and fair price

Example:

- Price today $S(0)$
- $A$ and $B$ enter into a future contract to sell/buy at price $F$ at time $T$
- $A$ borrows $S(0)$ from the bank, buys the asset and waits till $T$
- At time $T$, $A$ owes the bank $S(0) \exp(rT)$ and has the asset to sell to $B$
- $F = S(0) \exp(rT)$, else arbitrage opportunity
Options

- Call: right to buy at price $K$ at time $T$

- Put: right to sell at price $K$ at time $T$

- Exercise of contract
  - European style: only at time $T$
  - American style: any time in $0 \rightarrow T$

Example: Put-Call Parity

- Portfolio $P_1$: European Call + cash $K \exp(-rT)$
- Portfolio $P_2$: European Put + one share of underlying stock
- Values at time $t = 0$
  $$
  P_1 = C + K \exp(-rT) \\
  P_2 = P + S(0)
  $$
- Value of portfolios at time $t = T$
  $$
  S(T) > K \\
  P_1 = [S(T) - K] + K = S(T) \\
  P_2 = 0 + S(T) = S(T)
  $$
  $$
  S(T) < K \\
  P_1 = 0 + K = K \\
  P_2 = [K - S(T)] + S(T) = K
  $$
- Both portfolios having the same value at time $t = T$ should also have the same value at $t = 0$.
  $$
  C + K \exp(-rT) = P + S(0)
  $$
• Geometric Brownian motion for stock price
\[ dS(t) = \mu S(t)dt + \sigma S(t)dW(t) \]
\[ \frac{dS(t)}{S(t)} = \mu dt + \sigma dW(t) \]

• Stochastic differential equation for the log of the process
\[ F(S, t) = \log S(t) \]

• Ito’s lemma tells us about increments \( dF \)
• Terms needed to apply Ito’s lemma

\[
\begin{align*}
\frac{\partial F}{\partial t} &= 0 \\
\frac{\partial F}{\partial S} &= 1 \\
\frac{\partial^2 F}{\partial S^2} &= -\frac{1}{S^2}
\end{align*}
\]

\[
dF = \left( \frac{\partial F}{\partial t} + \mu S \frac{\partial F}{\partial S} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 F}{\partial S^2} \right) dt + \sigma S \frac{\partial F}{\partial S} dW
\]
\[= \left( \mu - \frac{\sigma^2}{2} \right) dt + \sigma dW
\]

\[\log S(t) = \log S(0) + \left( \mu - \frac{\sigma^2}{2} \right) t + \sigma dW(t)\]

• \( dW(t) = \epsilon \sqrt{t} \) where \( \epsilon \sim \mathcal{N}(0, 1) \)

\[\log S(t) \sim \mathcal{N} \left[ \log S(0) + \left( \mu - \frac{\sigma^2}{2} \right) t, \ \sigma^2 t \right] \]

• Log of asset price has a normal distribution
• Also

\[S(t) = S(0) \exp \left( (\mu - \sigma^2/2)t + \sigma \sqrt{t} \epsilon \right)\]
Black-Scholes Model

- Model
  \[ dS(t) = \mu S(t)dt + \sigma S(t)dW(t) \]

- Change in option price
  \[ df = \frac{\partial f}{\partial t}dt + \frac{\partial f}{\partial S}dS + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2}dt \]

- At maturity
  \[ f(S(T), T) = \max\{S(T) - K, 0\} \]

- Consider a portfolio
  - Own \( \Delta \) stocks (long)
  - One call option sold
  \[ \Pi = \Delta S - f(S, t) \]

\[
\begin{align*}
d\Pi &= \Delta dS - df \\
&= \left( \Delta - \frac{\partial f}{\partial S} \right) dS - \left( \frac{\partial f}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} \right) dt \\
\text{Term in } dS \text{ (stochastic) can be eliminated by choosing } \Delta \\
\Delta &= \frac{\partial f}{\partial S} \\
\text{With this choice of } \Delta \text{ (balance between short and long), the portfolio is riskless.} \\
d\Pi &= r \Pi dt \\
\text{Eliminating } d\Pi
\end{align*}
\]
\[
\frac{\partial f}{\partial t} + rS \frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} - rf = 0
\]
Partial differential equation

\[ \frac{\partial f}{\partial t} + rS \frac{\partial f}{\partial S} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} - rf = 0 \]

Boundary condition
- European Call: \( f(S, T) = \max\{S - K, 0\} \)
- European Put: \( f(S, T) = \max\{K - S, 0\} \)

Black-Scholes

\[ C = S_0 \mathcal{N}(d_1) - K \exp(-rT) \mathcal{N}(d_2) \]

\[ d_1 = \frac{\log(S_0/K) + (r + \sigma^2/2)T}{\sigma \sqrt{T}} \]

\[ d_2 = \frac{\log(S_0/K) + (r - \sigma^2/2)T}{\sigma \sqrt{T}} \]

\[ = d_1 - \sigma \sqrt{T} \]

\[ \mathcal{N}(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp(-y^2/2)dy \]

Put-Call parity

\[ P = K \exp(-rT) \mathcal{N}(-d_2) - S_0 \mathcal{N}(-d_1) \]
Options Pricing on a Binomial Model

- Construct a portfolio:
  - A riskless bond, initial price $B_0 = 1$ and future value $B_1 = \exp(r\delta t)$
  - Underlying asset, initial value $S_0$
  - Number of stocks $\Delta$, number of bonds $\Psi$

- Initial value of this portfolio

\[
\Pi_0 = \Delta S_0 + \Psi
\]

- Future value depends on price movement up or down

\[
\begin{align*}
\Pi_u &= \Delta S_0u + \Psi \exp(r\delta t) \\
\Pi_d &= \Delta S_0d + \Psi \exp(r\delta t)
\end{align*}
\]

- We can solve for a portfolio that will replicate option payoff

\[
\Delta S_0u + \Psi \exp(r\delta t) = f_u \\
\Delta S_0d + \Psi \exp(r\delta t) = f_d
\]

... and solve for $\Delta$ and $\Psi$

- ... solving

\[
\begin{align*}
\Delta &= \frac{f_u - f_d}{S_0(u - d)} \\
\Psi &= \exp(-r\delta t) \frac{uf_d - df_u}{u - d}
\end{align*}
\]

- No arbitrage $\implies$ initial value of this portfolio should be $f_0$

\[
\begin{align*}
f_0 &= \Delta S_0 + \Psi \\
&= \frac{f_u - f_d}{(u - d)} + \exp(-r\delta t) \frac{uf_d - df_u}{u - d} \\
&= \exp(-r\delta t) \left\{ \frac{\pi_u}{u - d} f_u + \frac{\pi_d}{u - d} f_d \right\}
\end{align*}
\]

- Defining probabilities

\[
\pi_u = \frac{\exp(r\delta t) - d}{u - d} \quad \text{and} \quad \pi_d = \frac{u - \exp(r\delta t)}{u - d}
\]

option price interpreted as discounted expected value

\[
f_0 = \exp(-r\delta t) \left( \pi_u f_u + \pi_d f_d \right)
\]
Binomial Lattice

Calibrating a Binomial Lattice

- When are these equivalent?

\[ dS = rS dt + \sigma S dW \]

- Log normal distribution

\[ \log(S_{t+\delta t}) \sim \mathcal{N} \left( (r - \sigma^2/2), \sigma^2 \delta t \right) \]

- Mean and variance of log normal distribution

(log of the variable is normal, what is mean and variance of the variable?)

\[
\begin{align*}
E \left[ S_{t+\delta t} \right] &= \exp(r \delta t) \\
\text{Var} \left[ S_{t+\delta t} \right] &= \exp(2r \delta t) \left( \exp(\sigma^2 \delta t) - 1 \right)
\end{align*}
\]
Calibrating binomial lattice (cont’d)

- Mean for the lattice
  \[ E[S_{t+\delta t}] = p u S_t + (1 - p) d S_t \]

- Equating the means...
  \[ p u S_t + (1 - p) d S_t = \exp(r \delta t) S_t \]
  \[ p = \frac{\exp(r \delta t) - d}{u - d} \]

- Variance on the lattice
  \[ \text{Var}[S_{t+\delta t}] = E[S_{t+\delta t}^2] - E^2[S_{t+\delta t}] \]
  \[ = S_t^2 (p u^2 + (1 - p) d^2) - S_t^2 \exp(2r \delta t) \]
  ...
  which from the dynamical model is...
  \[ \text{Var}[S_{t+\delta t}] = S_t^2 \exp(2r \delta t) (\exp(\sigma^2 \delta t) - 1) \]

(cont’d)

- Equating the two variances
  \[ S_t^2 \exp(2r \delta t) (\exp(\sigma^2 \delta t) - 1) = S_t^2 (p u^2 + (1 - p) d^2) - S_t^2 \exp(2r \delta t) \]
  Which reduces to
  \[ \exp(2r \delta t + \sigma^2 \delta t) = p u^2 + (1 - p) d^2 \]
  Substitute for \( p \) and simplify
  \[ \exp(2r \delta t + \sigma^2 \delta t) = (u + d) \exp(r \delta t) - 1 \]
  ...
  and because \( u = 1/d \),
  \[ u^2 \exp(r \delta t) - u (1 + \exp(2r \delta t + \sigma^2 \delta t)) + \exp(r \delta t) = 0 \]
  ...
  a quadratic equation in \( u \).
\[ u = \frac{(1 + \exp(2r\delta t + \sigma^2\delta t)) + \sqrt{(1 + \exp(2r\delta t + \sigma^2\delta t)^2 - 4\exp(2r\delta t)^2}}{2\exp(r\delta t)} \]

Taylor series expansion of \(\exp(x)\)

\[(1 + \exp(2r\delta t + \sigma^2\delta t))^2 - 4\exp(2r\delta t) \approx (2 + (2r + \sigma^2)\delta t)^2 - 4(1 + 2r\delta t) \approx 4\sigma^2\delta t\]

\[
\begin{align*}
u & \approx \frac{2 + (2r + \sigma^2)\delta t + 2\sigma\sqrt{\delta t}}{2\exp(r\delta t)} \\
& \approx \left(1 + r\delta t + \frac{\sigma^2}{2}\delta t + \sigma\sqrt{\delta t}\right)(1 - r\delta t) \\
& \approx 1 + r\delta t + \frac{\sigma^2}{2}\delta t + \sigma\sqrt{\delta t} - r\delta t \\
& = 1 + \sigma\sqrt{\delta t} + \frac{\sigma^2}{2}\delta t
\end{align*}
\]

Calibrating the Binomial Lattice (cont’d)

\[
\begin{align*}
u &= \exp(\sigma\sqrt{\delta t}) \\
d &= \exp(-\sigma\sqrt{\delta t}) \\
p &= \frac{\exp(r\delta t) - d}{u - d}
\end{align*}
\]

\[dS = rS\,dt + \sigma S\,dW\]
Example

- European call option; $S_0 = K = 50; \ r = 0.1; \ \sigma = 0.4$; maturity in five months.

>> call = blsprice(50, 50, 0.1, 5/12, 0.4)  
call =  
       6.1165

We can now build the lattice

\[ \delta t = \frac{1}{12} \quad 0.0833 \]
\[ u = \exp(\sigma \sqrt{t}) \quad 1.1224 \]
\[ d = \frac{1}{u} \quad 0.8909 \]
\[ p = \frac{(\exp(r\delta t) - d)}{(u - d)} \quad 0.5073 \]
function [price, lattice] = LatticeEurCall(S0,K,r,T,sigma,N)

deltaT = T/N;
u=exp(sigma * sqrt(deltaT));
d=1/u;
p=(exp(r*deltaT) - d)/(u-d);
lattice = zeros(N+1,N+1);

for i=0:N
    lattice(i+1,N+1)=max(0 , S0*(u^i)*(d^(N-i)) - K);
end

for j=N-1:-1:0
    for i=0:j
        lattice(i+1,j+1) = exp(-r*deltaT) * ...
        (p * lattice(i+2,j+2) + (1-p) * lattice(i+1,j+2));
    end
end

price = lattice(1,1);

function price = AmPutLattice(S0,K,r,T,sigma,N)
deltaT = T/N;
u=exp(sigma * sqrt(deltaT));
d=1/u;
p=(exp(r*deltaT) - d)/(u-d);
discount = exp(-r*deltaT);
p_u = discount*p;
p_d = discount*(1-p);
SVals = zeros(2*N+1,1);
SVals(N+1) = S0;

[...]
function price = AmPutLattice(S0,K,r,T,sigma,N)

[...]

for i=1:N
    SVals(N+1+i) = u*SVals(N+i);
    SVals(N+1-i) = d*SVals(N+2-i);
end
PVals = zeros(2*N+1,1);
for i=1:2:2*N+1
    PVals(i) = max(K-SVals(i),0);
end
[...]

Decisions at every point during backtracking

\[ f_{i,j} = \max\{K - S_{i,j}, \exp(-r\delta t)(p f_{i+1,j+1} + (1-p) f_{i,j+1})\} \]
We will look at inference as expectations...

\[ E[g(X)] = \int_{-\infty}^{+\infty} g(x)f_X(x) \, dx \]

Consider the integral

\[ I = \int_{0}^{1} g(x) \, dx \]

Think of this as computing the expected value

( of a function of a uniform random variable):

\[ E[g(U)], \quad \text{where } U \sim (0, 1) \]

We approximate the integral by

\[ \hat{I}_m = \frac{1}{m} \sum_{i=1}^{m} g(U_i) \]

Where will we use this?

European call option

\[ f = \exp(-rT) E[f_T] \]

\( f_T \) is payoff at maturity \( T \); fair price is discounted expected payoff

\[ f_T = \max\left\{ 0, \, S(0) \exp((r - \sigma^2/2)T + \sigma \sqrt{T} \epsilon) - K \right\} \]

```matlab
% BlsMC1.m
function Price = BlsMC1(S0,K,r,T,sigma,NRepl)
    nuT = (r - 0.5*sigma^2)*T;
    siT = sigma * sqrt(T);
    DiscPayoff = exp(-r*T)*max(0, S0*exp(nuT+siT*randn(NRepl,1))-K);
    Price = mean(DiscPayoff);
```

> S0=50; K=60; r=0.05; T=1; sigma=0.2;
> randn('state', 0);
> BlsMC1(S0, K, r, T, sigma, 1000)
> ans =

1.2562
Is this a good approach?

- Different answers on different runs
  ```matlab
  S0=50; K=60; r=0.05; T=1; sigma=0.2;
  randn('state', 0);
  BlsMC1(S0, K, r, T, sigma, 1000)
  ans =
  1.2562
  BlsMC1(S0, K, r, T, sigma, 1000)
  ans =
  1.8783
  BlsMC1(S0, K, r, T, sigma, 1000)
  ans =
  1.7864
  ```

- What if we had large number of samples?
  ```matlab
  BlsMC1(S0, K, r, T, sigma, 1000000)
  ans =
  1.6295
  BlsMC1(S0, K, r, T, sigma, 1000000)
  ans =
  1.6164
  BlsMC1(S0, K, r, T, sigma, 1000000)
  ans =
  1.6141
  ```

Sampling: Inverse Transform

- Sample $X$ from $f(x)$; Cumulative distribution $F(x)$
  - Draw $U \sim U(0.1)$
  - Return $X = F^{-1}(U)$

  $$P\{X \leq x\} = P\{F^{-1}(U) \leq x\} = P\{U \leq F(x)\} = F(x)$$

- Example: Exponential distribution $X \sim \exp(\mu)$
  - Cumulative
    $$F(x) = 1 - \exp(-\mu x)$$
  - Inverse
    $$x = -\frac{1}{\mu} \log(1 - U)$$
  - Distributions of $U$ and $(1 - U)$ are the same
    Hence return: $-\log(U)/\mu$
Sampling: Acceptance-Rejection Method

- Probability density function: \( f(x) \)
- Consider a known function \( t(x) \), such that
  \[ t(x) \geq f(x), \quad \forall x \in \mathcal{I} \]

- \( \mathcal{I} \) is the support for \( f \) (region in which it is defined)
- \( t(x) \) is a probability density of normalized
  \[ r(x) = \frac{t(x)}{c} \quad c = \int_{\mathcal{I}} t(x) \, dx \]

Algorithm

1. Generate \( Y \sim r \)
2. Generate \( U \sim U(0, 1) \)
3. If \( U \leq \frac{f(Y)}{t(Y)} \) return \( X = Y \)
   Else go to 1

Homework
Page 235, Brandimarte

- \( f(x) = 30(x^2 - 2x^3 + x^4), \quad x \in [0, 1] \)
- Algorithm
  1. Draw \( U_1 \) and \( U_2 \)
  2. If \( U_2 \leq 16(U_1^2 - 2U_1^3 + U_1^4) \)
     accept \( X = U_1 \)
     Else
     go to 1
- Exercise:
  - Draw the graph of \( f(x) \)
  - Simulate 1000 samples using above algorithm
  - Draw a histogram to the same scale as \( f(x) \) – do they match? Is it better with 100000 samples?
  - On average, how many trials were needed through the accept-reject loop for each sample?
Variance Reduction

- Independent samples \( X_i \)
- Sample mean (estimates mean \( \mu = E[X_i] \) from \( n \) samples)
  \[
  \overline{X}(n) = \frac{1}{n} \sum_{i=1}^{n} X_i
  \]

- Sample variance
  \[
  S^2(n) = \frac{1}{(n-1)} \sum_{i=1}^{n} [X_i - \overline{X}(n)]^2
  \]

- Error of the estimator
  \[
  E \left[ (\overline{X}(n) - \mu)^2 \right] = \text{Var} \left[ \overline{X}(n) \right] \\
  = \text{Var} \left[ \frac{1}{n} \sum_{i=1}^{n} X_i \right] \\
  = \frac{1}{n^2} \times n \times \sigma^2 = \frac{\sigma^2}{n}
  \]

- Two points:
  - More samples \( n \) reduces the variance in estimation
  - Variance reduction schemes can control \( \sigma^2 \)

Variance reduction: Antithetic Sampling

- Pair of sequences
  \[
  \left\{ \begin{array}{cccc}
  X_1^{(1)} & X_1^{(2)} & \ldots & X_1^n \\
  X_2^{(1)} & X_2^{(2)} & \ldots & X_2^n
  \end{array} \right\}
  \]

- Columns (horizontally) are independent
- \( X_1^{(i)} \) and \( X_2^{(i)} \) are dependent.
- Sample is a function of each pair: \( X^{(i)} = (X_1^{(i)} + X_2^{(i)}) / 2 \)
- Variance
  \[
  \text{Var} \left[ \overline{X}(n) \right] = \frac{1}{n} \text{Var} \left[ X^{(i)} \right] \\
  = \frac{1}{4n} \left\{ \text{Var}(X_1^{(i)}) + \text{Var}(X_2^{(i)}) + 2 \text{Cov}(X_1^{(i)}, X_2^{(i)}) \right\} \\
  = \frac{1}{2n} \text{Var}(X) (1 + \rho)
  \]

- Uniform random number \( \{U_k\} \) and \( \{1 - U_k\} \) as sequences.
function [Price, CI] = BlsMC2(S0,K,r,T,sigma,NRepl)
nuT = (r - 0.5*sigma^2)*T;
siT = sigma * sqrt(T);
DiscPayoff = exp(-r*T)*max(0, S0*exp(nuT+siT*randn(NRepl,1))-K);
[Price, VarPrice, CI] = normfit(DiscPayoff);

function [Price, CI] = BlsMCAV(S0,K,r,T,sigma,NPairs)
nuT = (r - 0.5*sigma^2)*T;
siT = sigma * sqrt(T);
Veps = randn(NPairs,1);
Payoff1 = max( 0 , S0*exp(nuT+siT*Veps) - K);
Payoff2 = max( 0 , S0*exp(nuT+siT*(-Veps)) - K);
DiscPayoff = exp(-r*T) * 0.5 * (Payoff1+Payoff2);
[Price, VarPrice, CI] = normfit(DiscPayoff);

Mahesan Niranjan
COMP6212

Homework
Test the two functions: BlsMC and BlsMCAV
(Brandimarte, p248)
> randn('state', 0)
> [Price, CI] = BlsMC2(50,50,0.05,1,0.4,200000)
Price= 9.0843
CI =
9.0154
9.1532
> (CI(2)-CI(1))/Price
ans =
0.0152
> randn('state', 0)
> [Price, CI] = BlsMCAV(50,50,0.05,1,0.4,200000)
Price= 9.0553
CI =
8.9987
9.1118
> (CI(2)-CI(1))/Price
ans =
0.0125
We have seen three tools for pricing options
- Closed form Black-Scholes
- Binomial lattice
- Monte Carlo

How well can the relationship between asset price and option price be approximated?


\[
x = \left[ \frac{S}{X} \ (T - t) \right]^T
\]

\[
c = \sum_{j=1}^{J} \lambda_j \phi_j(x) + w^T x + w_0
\]

*Figure 4. Simulated call option prices normalized by strike price and plotted versus*
\[
\frac{C}{X} = -0.06 \sqrt{\frac{S}{X - 1.35}} \cdot \frac{59.79}{T - t - 0.45} - 0.03 \frac{S}{X - 1.35} \cdot \frac{10.24}{T - t - 0.45} + 2.55 \\
- 0.03 \sqrt{\frac{S}{X - 1.18}} \cdot \frac{59.79}{T - t - 0.24} - 0.03 \frac{S}{X - 1.18} \cdot \frac{10.24}{T - t - 0.24} + 1.97 \\
+ 0.03 \sqrt{\frac{S}{X - 0.98}} \cdot \frac{59.79}{T - t + 0.20} - 0.03 \frac{S}{X - 0.98} \cdot \frac{10.24}{T - t + 0.20} + 0.00 \\
+ 0.10 \sqrt{\frac{S}{X - 1.05}} \cdot \frac{59.79}{T - t + 0.10} - 0.03 \frac{S}{X - 1.05} \cdot \frac{10.24}{T - t + 0.10} + 1.62 \\
+ 0.14 S/X - 0.24(T - t) - 0.01. \]