MSFE - Option Pricing 14,5825

Four use cases of Ito's lemma :

2.1 Stock model	$de^{\int_t^{t+2} r_s ds} S_t \;\; and \;\; d\ln S_t$
3.2 Derive BSPDE	df_t
3.5 Derive risk neutral pricing	$d(e^{-rt} f)$

Part 1. Probability / Expectation / Brownian

1.1 Probability triple, random variable and equivalence

Probability triple (Ω, F, P) is :

(1) Sample space Ω is a set of all possible outcomes. For example …

- set of all possible paths starting with *$100* is a sample space.

(2) Filtration F is a set of all events, while each event is a subset of outcomes. For example …

- set of all possible paths soaring above *$100* before T is an event
- set of all possible paths fluctuating within *$80-$90* is another event
- similarly, we can name numerous events, forming filtration F

(3) Probability measure P is the mapping $P:F\to[0,1]$, such that $P(\Omega) = 1$.

Random variables has value derived from outcome. Inversely a value-set for a random variable defines an event. Now lets consider the probability space of rolling two dices, there are *36* outcomes. Define random variable X as sum of two dices, while Y as absolute difference between the two dices. We can define event by plugging value-set for X and Y :

$$
\begin{array}{llll}
(4) & P(X=7) & = & \Pr((1,6),(2,5),(3,4),(4,3),(5,2),(6,1)) & = & 1/6 \\
 & P(X\ge 10) & = & \Pr((4,6),(5,5),(5,6),(6,4),(6,5),(6,6)) & = & 1/6 \\
 & P(Y=0) & = & \Pr((1,1),(2,2),(3,3),(4,4),(5,5),(6,6)) & = & 1/6
\end{array}
$$

Two measures are said to be **equivalent** if they agree to which events are possible or not (although different values) :

(5) *if* $\quad P(w)>0 \text{ iff } P'(w)>0 \text{ for all } w\in\Omega$

then $\quad$ *P and P' are said to be equivalent.*

1.2 Conditional probability

In general, we have conditional probability :

$$(1)\qquad \Pr(X\,|\,Y) = \Pr(X,Y)/\Pr(Y)$$

and recursion for conditional probability :

$$(2)\qquad \Pr(X) = \Pr(X\,|\,A)\Pr(A)+\Pr(X\,|\,B)\Pr(B)+\Pr(X\,|\,C)P(C)$$

1.3 Joint probability

Lets define the probability density function and the joint probability density function :

$$(1)\qquad p_X(x) = \partial_x \Pr(X<x) \qquad \textit{1st term in 1D Taylor series}$$

$$(2)\qquad p_{X,Y}(x,y) = \partial_x\partial_y \Pr(X<x,Y<y) \qquad \textit{4th term in 2D Taylor series}$$

$$\textit{1D}\qquad \Pr(X<x+\Delta x)-\Pr(X<x) = \underbrace{\partial_x \Pr(X<x)}_{p_X(x)}\Delta x+\frac{1}{2}\partial_{xx}\Pr(X<x)(\Delta x)^2+\ldots$$

$$\textit{2D}\qquad \begin{array}{l}\Pr(X<x+\Delta x,Y<y+\Delta y)\\ -\Pr(X<x,Y<y)\end{array} = \left[\begin{array}{l}\partial_x \Pr(X<x,Y<y)\Delta x+\partial_y \Pr(X<x,Y<y)\Delta y\\ +\frac{1}{2}[\partial_{xx}\Pr(X<x,Y<y)(\Delta x)^2+2\underbrace{\partial_{xy}\Pr(X<x,Y<y)}_{p_{XY}(x,y)}(\Delta x)(\Delta y)+\partial_{yy}\Pr(X<x,Y<y)(\Delta y)^2]+\ldots\end{array}\right.$$

1.4 Change of variable

Find new *pdf* with change of variables in *1D* and *2D*. We simply write $\Pr(X \in [x, x+\Delta x])$ as $\Pr(X = x)$.

$$
\begin{array}{llll}
(1) & \Pr(X = x) = & \Pr(Y = y) & \text{where } y = f(x) \\
 & p(x)dx = & p(y)dy & \text{where } dy = \partial_x f dx \\
 & = & p(y)\partial_x f dx & \\
 & p(y) = & p(x)/\partial_x f &
\end{array}
$$

$$
\begin{array}{llll}
(2) & \Pr(Y = y, X = x) = & \Pr(U = u, V = v) & \text{where } u = f(x,y) \text{ and } v = g(x,y) \\
 & p(x,y)dxdy = & p(u,v)dudv & \text{where } du = \partial_x f dx + \partial_y f dy \text{ and } dv = \partial_x g dx + \partial_y g dy \\
 & = & p(u,v) diag(J) dxdy & \text{where } dudv = (\partial_x f dx + \partial_y f dy)(\partial_x g dx + \partial_y g dy) = (\partial_x f \partial_y g + \partial_x g \partial_y f) dxdy \\
 & p(u,v) = & \dfrac{p(x,y)}{diag(J)} & \text{where } diag(J) = \partial_x f \partial_y g + \partial_x g \partial_y f
\end{array}
$$

1.5 Independent vs uncorrelated

Independence is defined as $p(x,y) = p(x)p(y)$, which implies

$$
\begin{aligned}
p(x|y) &= p(x) \\
E[X|Y] &= \int x p(x|y) dx \\
&= \int x p(x,y)/p(y) dx \\
&= \int x p(x) p(y)/p(y) dx \\
&= \int x p(x) dx \\
&= E[X] \\
E[XY] &= \iint xy p(x,y) dxdy \\
&= \iint xy p(x) p(y) dxdy \\
&= E[X]E[Y]
\end{aligned}
$$

$$
\begin{array}{lccc}
 & \underline{\textit{joint prob}} & & \underline{\textit{conditional prob}} \\
\text{independent} \rightarrow & (1)\ p(x,y) = p(x)p(y) & \rightarrow & (4)\ p(x|y) = p(x) \\
 & \downarrow & & \downarrow \\
\text{uncorrelated} \rightarrow & (2)\ E[XY] = E[X]E[Y] & \rightarrow & (5)\ E[X|Y] = E[X] \\
 & \downarrow & & \\
 & (3)\ Cov(XY) = 0 & &
\end{array}
$$

Uncorrelation is defined as $E[XY] = E[X]E[Y]$, which implies

$$
\begin{aligned}
Cov(XY) &= E[(X - E[X])(Y - E[Y])] \\
&= E[XY - E[X]Y - XE[Y] + E[X]E[Y])] \\
&= E[XY] - E[X]E[Y] \\
&= 0
\end{aligned}
$$

1.6 Expectation and conditional expectation

Intuitively it summarises information of a random variable X with the centre of mass.

$$
\begin{array}{llll}
(1) & E[f(X)] = & \sum_\Omega f(X(w))P(w) & \textit{for discrete X, in terms of w} \\
 & E[f(X)] = & \int_\Omega f(X(w))dP(w) & \textit{for continuous X, in terms of w}
\end{array}
$$

Conditional probability is defined on set, while conditional expectation is defined on function.

$$
\begin{array}{llll}
(2) & E[f(X)|A] = & \sum_\Omega f(X(w))P(w|A) & \textit{for discrete X, in terms of w} \\
 & E[f(X)|A] = & \int_\Omega f(X(w))dP(w|A) & \textit{for continuous X, in terms of w}
\end{array}
$$

Recursion of conditional expectation :

$$
(3) \quad E[X] = E[X|A]P(A) + E[X|B]P(B) + E[X|C]P(C)
$$

1.7 Expectation of sum

Sum of correlated/uncorrelated variables

$$
\begin{aligned}
(1)\quad E[ax+by+cz] &= aE[x]+bE[y]+cE[z] \\
(2)\quad V[ax+by+cz] &= E[(ax+by+cz)^2]-E^2[ax+by+cz] \\
&= \begin{bmatrix} a^2E[x^2]+b^2E[y^2]+c^2E[z^2]+2abE[xy]+2bcE[yz]+2caE[zx] \\ -a^2E^2[x]-b^2E^2[y]-c^2E^2[z]-2abE[x]E[y]-2bcE[y]E[z]-2caE[z]E[x] \end{bmatrix} \\
&= a^2V[x]+b^2V[y]+c^2V[z]+2abCov(x,y)+2bcCov(y,z)+2caCov(z,x)
\end{aligned}
$$

$$
\begin{array}{lllllll}
(3) & V[x+y] & = & 2V[x] & & & \text{uncorrelated } \sigma_x^2=\sigma_y^2 \\
 & V[2x] & = & 4V[x] & & & 100\%\text{ correlated } x=y \\
 & \varepsilon_1+\varepsilon_2 & = & \varepsilon(0,\sqrt{2}) & = & \sqrt{2}\varepsilon & \text{uncorrelated} \\
 & \varepsilon_1+\varepsilon_1 & = & \varepsilon(0,2) & = & 2\varepsilon & 100\%\text{ correlated}
\end{array}
$$

1.8 Moment of Gaussian and lognormal

Sum of Gaussian

$$
\begin{aligned}
(1)\quad w_1\varepsilon(\mu_1,\sigma_1)+w_2\varepsilon(\mu_2,\sigma_2) &= w_1\mu_1+w_1\sigma_1\varepsilon+w_2\mu_2+w_2\sigma_2\varepsilon \\
&= w_1\mu_1+w_2\mu_2+\varepsilon(0,w_1\sigma_1)+\varepsilon(0,w_2\sigma_2) \\
&= w_1\mu_1+w_2\mu_2+\varepsilon(0,\sqrt{(w_1\sigma_1)^2+(w_2\sigma_2)^2}) \\
&= \varepsilon(w_1\mu_1+w_2\mu_2,\sqrt{(w_1\sigma_1)^2+(w_2\sigma_2)^2})
\end{aligned}
$$

Moment of Gaussian

Given $x=\varepsilon$ and $y=\mu+\sigma x$

$$
\begin{array}{llll}
(2) & E[x] & = & 0 \\
 & E[x^2] & = & 1 \\
 & E[x^3] & = & 0 \\
 & E[x^4] & = & 3
\end{array}
$$

$$
\begin{array}{lllll}
E[y] & = & \mu+\sigma E[x] & = & \mu \\
E[y^2] & = & \mu^2+2\mu\sigma E[\varepsilon]+\sigma^2E[\varepsilon^2] & = & \mu^2+\sigma^2 \\
E[y^3] & = & \mu^3+3\mu^2\sigma E[\varepsilon]+3\mu\sigma^2E[\varepsilon^2]+\sigma^3E[\varepsilon^3] & = & \ldots \\
E[y^4] & = & \mu^4+4\mu^3\sigma E[\varepsilon]+6\mu^2\sigma^2E[\varepsilon^2]+6\mu\sigma^3E[\varepsilon^3]+\sigma^4E[\varepsilon^4] & = & \ldots
\end{array}
$$

$$
\begin{array}{lllllll}
\text{variance} & = & E[(x-E[x])^2] & = & E[x^2-2xE[x]+E^2[x]] & = & M_2-M_1^2 \\
\text{skewness} & = & E[(x-E[x])^3] & = & E[x^3-3x^2E[x]+3xE^2[x]-E^3[x]] & = & M_3-3M_2M_1+2M_1^3 \\
\text{kurtosis} & = & E[(x-E[x])^4] & = & E[x^4-4x^3E[x]+6x^2E^2[x]-4xE^3[x]+E^4[x]] & = & M_4-4M_3M_1+6M_2M_1^2-3M_1^4
\end{array}
$$

$$
\begin{aligned}
E[x] = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}xe^{-x^2/2}dx &= -\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}de^{-x^2/2} \\
&= -\frac{1}{\sqrt{2\pi}}\left[e^{-x^2/2}\right]_{-\infty}^{\infty} \\
&= 0
\end{aligned}
$$

$$
\begin{aligned}
E[x^2] &= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} x^2 e^{-x^2/2}dx = -\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} x\,de^{-x^2/2} \\
&= -\frac{1}{\sqrt{2\pi}}\left[xe^{-x^2/2}\right]_{-\infty}^{\infty} + \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} e^{-x^2/2}dx \\
&= -\frac{1}{\sqrt{2\pi}}\left[\lim_{x\to\infty}(x/e^{x^2/2}) - \lim_{x\to-\infty}(x/e^{x^2/2})\right]+1 \\
&= -\frac{1}{\sqrt{2\pi}}\left[\lim_{x\to\infty}(1/(xe^{x^2/2})) - \lim_{x\to-\infty}(1/(xe^{x^2/2}))\right]+1 \\
&= 1
\end{aligned}
$$

$$
\begin{aligned}
E[x^3] &= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} x^3 e^{-x^2/2}dx = -\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} x^2\,de^{-x^2/2} \\
&= -\frac{1}{\sqrt{2\pi}}\left[x^2e^{-x^2/2}\right]_{-\infty}^{\infty} + \frac{2}{\sqrt{2\pi}}\int_{-\infty}^{\infty} xe^{-x^2/2}dx \\
&= -\frac{1}{\sqrt{2\pi}}\left[\lim_{x\to\infty}(x^2/e^{x^2/2}) - \lim_{x\to-\infty}(x^2/e^{x^2/2})\right]+2E[x] \\
&= -\frac{1}{\sqrt{2\pi}}\left[\lim_{x\to\infty}(2x/(xe^{x^2/2})) - \lim_{x\to-\infty}(2x/(xe^{x^2/2}))\right]+0 \\
&= -\frac{1}{\sqrt{2\pi}}\left[\lim_{x\to\infty}(2/(e^{x^2/2})) - \lim_{x\to-\infty}(2/(e^{x^2/2}))\right] \\
&= 0
\end{aligned}
$$

$E[x^4] = \ldots$ *try doing it*

Moment of lognormal

Given $x = \mu + \sigma\varepsilon$

(3)
$$
\begin{aligned}
E[e^{kx}] &= \frac{1}{\sqrt{2\pi\sigma^2}}\int_{-\infty}^{\infty} e^{kx}e^{-(x-\mu)^2/(2\sigma^2)}dx \\
&= \frac{1}{\sqrt{2\pi\sigma^2}}\int_{-\infty}^{\infty} e^{-(x^2-2\mu x-2k\sigma^2x+\mu^2)/(2\sigma^2)}dx \\
&= \frac{1}{\sqrt{2\pi\sigma^2}}\int_{-\infty}^{\infty} e^{-[x-(\mu+k\sigma^2)]^2/(2\sigma^2)}e^{-[-(\mu+k\sigma^2)^2+\mu^2]/(2\sigma^2)}dx \\
&= \frac{1}{\sqrt{2\pi\sigma^2}}\int_{-\infty}^{\infty} e^{-[x-(\mu+k\sigma^2)]^2/(2\sigma^2)}e^{(2k\mu\sigma^2+k^2\sigma^4)/(2\sigma^2)}dx \\
&= \frac{e^{k\mu+k^2\sigma^2/2}}{\sqrt{2\pi\sigma^2}}\int_{-\infty}^{\infty} e^{-[x-(\mu+k\sigma^2)]^2/(2\sigma^2)}dx \\
&= e^{(k\mu)+(k\sigma)^2/2}
\end{aligned}
$$

1.9 Tower property

Two forms of Tower property and Law of total variance :

$$
\begin{aligned}
&(1) && E[E[X\,|\,Y]] &&= E[X] \\
&(2) && E[E[X\,|\,Y,Z]\,|\,Z] &&= E[X\,|\,Z] \\
&(3) && V(X) &&= E(V(X\,|\,Y)) - V(E(X\,|\,Y)) \qquad \textit{law of total variance}
\end{aligned}
$$

This is recursion formula for conditional expectation :

- the proof here is done in forward way
- the proof in quant.doc in backward way

proof

$$
\begin{aligned}
(1) \quad E[E[X\,|\,Y]] &= \int E[X\,|\,Y=y]p(Y=y)dy \\
&= \int\int xp(X=x\,|\,Y=y)dxp(Y=y)dy \\
&= \int\int x\frac{p(X=x,Y=y)}{p(Y=y)}p(Y=y)dxdy \\
&= \int x\int p(X=x,Y=y)dydx \\
&= \int xp(X=x)dx \\
&= E[X]
\end{aligned}
$$

$$
\begin{aligned}
(2) \quad E[E[X\,|\,Y,Z]\,|\,Z] &= \int E[X\,|\,Y=y,Z=z]p(Y=y\,|\,Z=z)dy \\
&= \int\int xp(X=x\,|\,Y=y,Z=z)dxp(Y=y\,|\,Z=z)dy \\
&= \int\int x\frac{p(X=x,Y=y,Z=z)}{p(Y=y,Z=z)}\frac{p(Y=y,Z=z)}{p(Z=z)}dxdy \\
&= \int x\int\frac{p(X=x,Y=y,Z=z)}{p(Z=z)}dydx \\
&= \int x\frac{p(X=x,Z=z)}{p(Z=z)}dx \\
&= E[X\,|\,Z]
\end{aligned}
$$

$$
\begin{aligned}
(3) \quad V(X\,|\,Y) &= E((X-E(X\,|\,Y))^2\,|\,Y) \\
&= E((X^2 - 2XE(X\,|\,Y) + E(X\,|\,Y)^2)\,|\,Y) \\
&= E(X^2\,|\,Y) - E(2XE(X\,|\,Y)\,|\,Y) + E(E(X\,|\,Y)^2)\,|\,Y) \\
&= E(X^2\,|\,Y) - 2E(X\,|\,Y)E(X\,|\,Y) + E(X\,|\,Y)^2 \\
&= E(X^2\,|\,Y) - E(X\,|\,Y)^2 \\
E(V(X\,|\,Y)) &= E(E(X^2\,|\,Y) - E(X\,|\,Y)^2) \\
&= E(E(X^2\,|\,Y)) - E(E(X\,|\,Y)^2) \\
&= E(X^2) - E(E(X\,|\,Y)^2) \\
V(E(X\,|\,Y)) &= E(E(X\,|\,Y)^2) - E(E(X\,|\,Y))^2 \\
&= E(E(X\,|\,Y)^2) - E(X)^2
\end{aligned}
$$

$$
\begin{aligned}
E(V(X\,|\,Y)) + V(E(X\,|\,Y)) &= E(X^2) - E(E(\cancel{X\,|\,Y)^2}) + E(E(\cancel{X\,|\,Y)^2) - E}(X)^2 \\
&= E(X^2) - E(X)^2 \\
&= V(X)
\end{aligned}
$$

1.10 Mean-variance estimation and central limit theorem

Suppose we are going to estimate the mean and variance of random variable X using its realization, we have :

(1)
$$\hat{\mu} = \sum x_n / N$$
$$\hat{\sigma}^2 = \sum_n (x_n - (\sum_m x_m / N))^2 / (N-1)$$

	data point	*mean est*	*variance est*
mean	$E[x_n] = \mu$	$E[\hat{\mu}] = \mu$	$E[\hat{\sigma}^2] = \sigma^2$
variance	$V[x_n] = \sigma^2$	$V[\hat{\mu}] = \sigma^2 / N$	
2nd moment	$E[x_n^2] = \mu^2 + \sigma^2$	$E[\hat{\mu}^2] = \mu^2 + \sigma^2 / N$	

Why N-1? Its unbiased estimation of variance, lets prove.

(2)
$$\begin{aligned}
E[\hat{\mu}] &= E[\sum x_n / N] \\
&= \sum E[x_n] / N \\
&= \mu \\
V[\hat{\mu}] &= V[\sum x_n / N] \\
&= \sum V[x_n] / N^2 \\
&= \sigma^2 / N \\
E[\hat{\sigma}^2] &= E[\sum_n (x_n - \hat{\mu})^2 / (N-1)] \\
&= E[\sum_n (x_n^2 - 2\hat{\mu}x_n + \hat{\mu}^2)] / (N-1) \\
&= E[\sum_n x_n^2 - 2\hat{\mu}(\sum_n x_n) + N\hat{\mu}^2] / (N-1) \\
&= E[\sum_n x_n^2 - 2N\hat{\mu}\hat{\mu} + N\hat{\mu}^2] / (N-1) \\
&= E[\sum_n x_n^2 - N\hat{\mu}^2] / (N-1) \\
&= (\sum_n E[x_n^2] - NE[\hat{\mu}^2]) / (N-1) \\
&= (N(\sigma^2 + \mu^2) - N(\sigma^2 / N + \mu^2)) / (N-1) \\
&= \sigma^2
\end{aligned}$$

as we dont know $E[\hat{\mu}x_n]$, so either convert $\hat{\mu} \to x_n$ or $x_n \to \hat{\mu}$

$$\boxed{\begin{aligned} E[x_n^2] &= \sigma^2 + \mu^2 \\ E[\hat{\mu}^2] &= \sigma^2 / N + \mu^2 \end{aligned}}$$

Central limit theorem tells us about the distribution of sum of *iid* :

(3)
$$\lim_{N\to\infty} \frac{\sum_n x_n / N - \mu}{\sigma / \sqrt{N}} \sim \varepsilon$$

σ is population stddev

formal description

i.e.
$$\lim_{N\to\infty} \frac{\hat{\mu} - E[\hat{\mu}]}{\sqrt{V[\hat{\mu}]}} \sim \varepsilon$$

easier to remember form

variance of estimation of mean

1.11 Brownian motion

(1) Wiener process (also known as Brownian motion) is a sequence of random variables z_t fulfilling :

- $z_0 = 0$
- $\Delta z_{t+\Delta t} \sim \varepsilon(0, \sqrt{\Delta t})$
- $Cov(\Delta z_{t+\Delta t}, \Delta z_t) = 0$

There are two ways to define the differential change in Brownian motion dz_t, so as to satisfy the above requirements.

(2)
$$\begin{array}{llll}
dz_t & = & \begin{bmatrix} +\sqrt{dt} & prob = 1/2 \\ -\sqrt{dt} & prob = 1/2 \end{bmatrix} & \textit{where } (dz_t)^2 = dt,\ dz_t dt = 0 \textit{ and } (dt)^2 = 0 \\
E[dz_t] & = & \frac{1}{2}\sqrt{dt} - \frac{1}{2}\sqrt{dt} & = 0 \\
E[(dz_t)^2] & = & \frac{1}{2}(\sqrt{dt})^2 + \frac{1}{2}(-\sqrt{dt})^2 & = dt \\
Var[dz_t] & = & E[(dz_t)^2] - E^2[dz_t] & = dt
\end{array}$$

(3)
$$\begin{array}{llllll}
dz_t & = & \sqrt{dt} \in & \textit{where } \in \sim N(0,1) & & \\
E[dz_t] & = & \sqrt{dt}\, E[\in] & = 0 & & \\
E[(dz_t)^2] & = & dt E[\varepsilon^2] & = dt(Var[\varepsilon] + E^2[\varepsilon]) & = dt(1+0^2) & = dt \\
Var[dz_t] & = & E[(dz_t)^2] - E^2[dz_t] & = dt - 0^2 & = dt &
\end{array}$$

1.12 Brownian motion path and sum

Lets derive z_t using central limit theorem.

(1)
$$\begin{array}{llll}
z_t & = & \int_{\tau=0}^{t} dz_\tau & \\
 & = & \lim_{N\to\infty} \underbrace{\sqrt{t/N}\varepsilon_1 + \sqrt{t/N}\varepsilon_2 + \sqrt{t/N}\varepsilon_3 + ... + \sqrt{t/N}\varepsilon_N}_{N} & \textit{where } \varepsilon_n \textit{ are independent identical standard normal} \\
 & = & \lim_{N\to\infty} \left(\frac{(\varepsilon + \varepsilon + \varepsilon + ... + \varepsilon)/N - 0}{1/\sqrt{N}} \right) \sqrt{t} & \\
 & = & \varepsilon\sqrt{t} & \textit{by central limit theorem} \\
\text{cov}(z_t, z_{t,T}) & = & 0 & \textit{uncorrelated (non overlapping)} \\
\text{cov}(z_t, z_T) & \neq & 0 & \textit{correlated (overlapping)}
\end{array}$$

Sum of two independent normals is : $\varepsilon(\mu_1, \sigma_1) + \varepsilon(\mu_2, \sigma_2) = \varepsilon(\mu_1 + \mu_2, \sqrt{(\sigma_1^2 + \sigma_2^2)})$. What happen if they are correlated?

(2)
$$\begin{array}{llll}
z_{t_1 - t_0} + z_{t_3 - t_2} & = & \varepsilon(0, \sqrt{t_1 - t_0}) + \varepsilon(0, \sqrt{t_3 - t_2}) & \textit{non overlapping} \\
 & = & \varepsilon(0, \sqrt{\sqrt{t_1 - t_0}^2 + \sqrt{t_3 - t_2}^2}) & \\
 & = & \varepsilon(0, \sqrt{t_1 - t_0 + t_3 - t_2}) &
\end{array}$$

(3)
$$\begin{array}{llll}
z_{t_2 - t_0} + z_{t_3 - t_1} & = & \varepsilon(0, \sqrt{t_1 - t_0}) + 2\varepsilon(0, \sqrt{t_2 - t_1}) + \varepsilon(0, \sqrt{t_3 - t_2}) & \textit{overlapping} \\
 & = & \varepsilon(0, \sqrt{\sqrt{t_1 - t_0}^2 + 4\sqrt{t_2 - t_1}^2 + \sqrt{t_3 - t_2}^2}) & \\
 & = & \varepsilon(0, \sqrt{t_1 - t_0 + 4(t_2 - t_1) + t_3 - t_2}) &
\end{array}$$

1.13 Brownian motion is martingale and driftless

Stochastic prcess X_t is said to be martingale (or equivalently, driftless) if :

$$
\begin{array}{lll}
X_t & = & E[X_T \mid I_t] & \textit{martingale in expectation form, or equivalently} \\
dX_t & = & \sigma_t dz_t & \textit{martingale in differential form, it is a driftless SDE}
\end{array}
$$

(1) Proof in forward direction :

$$
\begin{array}{lll}
E[X_T \mid I_t] & = & E[X_{T-\Delta t} + dX_{T-\Delta t} \mid I_t] \\
 & = & E[E[X_{T-\Delta t} + dX_{T-\Delta t} \mid I_{T-\Delta t}] \mid I_t] & \textit{tower property} \\
 & = & E[X_{T-\Delta t} + E[dX_{T-\Delta t} \mid I_{T-\Delta t}] \mid I_t] \\
 & = & E[X_{T-\Delta t} + E[\sigma_{T-\Delta t} dz_{T-\Delta t} \mid I_{T-\Delta t}] \mid I_t] \\
 & = & E[X_{T-\Delta t} \mid I_t] \\
 & = & \ldots \\
 & = & E[X_t \mid I_t] \\
 & = & X_t
\end{array}
$$

(2) Proof of backward direction :

$$
\begin{array}{lllll}
E(X_{t+dt} \mid I_t) & = & X_t \\
X_t + E(dX_t \mid I_t) & = & X_t & & \textit{where } X_{t+dt} = X_t + dX_t \\
E(\mu_t dt + \sigma_t dz_t \mid I_t) & = & 0 & & \textit{where } dX_t = \mu_t dt + \sigma_t dz_t \\
\mu_t dt + \underbrace{E(\sigma_t dz_t \mid I_t)}_{0} & = & 0 \quad \Rightarrow & \mu_t = 0 & \textit{hence it implies process with no drift}
\end{array}
$$

(3) Here is a summary :

	joint	*conditional*
probability	*independence*	*Markovian*
expectation	*uncorrelation*	*Martingale*

Markovian is defined with probability, while martingale is defined with expectation :

$$
\begin{array}{lll}
P(X_{n+1} \mid X_n, X_{n-1}, X_{n-2}, \ldots) & = & P(X_{n+1} \mid X_n) & \textbf{\textit{Markovian is memoryless.}} \\
E[X_{n+1} \mid X_n, X_{n-1}, X_{n-2}, \ldots] & = & X_n & \textbf{\textit{Martingale is driftless.}}
\end{array}
$$

Recall that independence is defined with probability, while uncorrelation is defined with expectation :

$$
\begin{array}{lll}
p(x,y) & = & p(x)p(y) & \textbf{\textit{independence}} \\
E[XY] & = & E[X]E[Y] & \textbf{\textit{uncorrelation}}
\end{array}
$$

1.14 Time integral and Ito's integral

(1) There are two types of stochastic integral : time integral and Ito's integral :

- time integral $\int_0^T f_t dt$ where f_t is stochastic
- Ito's integral $\int_0^T f_t dz_t$ where f_t is stochastic or deterministic

(2) Let's consider this time integral $\int_0^T z_t dt$:

$d(tz_t)$	=	$z_t dt + t dz_t$								
$z_t dt$	=	$d(tz_t) - t dz_t$								
$\int_0^T z_t dt$	=	$Tz_T - \int_0^T t dz_t$		*integration by parts*						
	=	$T\int_0^T dz_t - \int_0^T t dz_t$	=	$\int_0^T (T-t) dz_t$						
$E[\int_0^T z_t dt]$	=	$E\left[Tz_T - \int_0^T t dz_t\right]$	=	$0 - E\left[\int_0^T t dz_t\right]$	=	0				
$V[\int_0^T z_t dt]$	=	$V\left[\int_0^T (T-t) dz_t\right]$	=	$\int_0^T (T-t)^2 dt$	=	$-[(T-t)^3/3]_0^T$	=	$-(0-T^3/3)$	=	$T^3/3$
$d[\int_0^T z_t dt]$	=	$z_T dT$	=	*drifted, i.e. non-martingale*						

(3) Ito's integal is solved using these properties (please read *Chapter 5* for proof) :

property	*deterministic integrand*	*stochastic integrand*	
martingale	*yes*	*yes*	
$E[\int_0^T f_s dz_s \mid I_t]$	0	0	
$E[(\int_t^T f_s dz_s)^2 \mid I_t]$	$\int_t^T f_s^2 ds$	$\int_t^T E[f_s^2 \mid I_t] ds$	*also called Ito's isometry*
$E[(\int_t^T f_s dz_s)(\int_t^T g_s dz_s) \mid I_t]$	$\int_t^T f_s g_s ds$	$\int_t^T E[f_s g_s \mid I_t] ds$	*also called Ito's isometry*
distribution	$\varepsilon(0, \int_t^T f_s^2 ds)$	*unknown*	

Part 2. Stock model

2.1 Stock model 3SDE + 3ANS

Given *SDE* for stock S_t as of time *t* under *Q* measure :

$$dS_t = r_t S_t dt + \sigma_t S_t dz_t \qquad \textit{SDE(1)}$$

Lets derive *SDE* for *spot-price* X_t as of time *t* under *Q* measure, using Ito's lemma :

$$\begin{aligned}
X_t &= e^{\int_t^{t+2} r_s ds} S_t \\
dX_t &= (r_{t+2} - r_t) e^{\int_t^{t+2} r_s ds} S_t dt + e^{\int_t^{t+2} r_s ds} dS_t && \textit{by Leibniz Rule, see quant.doc} \\
&= (r_{t+2} - r_t) e^{\int_t^{t+2} r_s ds} S_t dt + e^{\int_t^{t+2} r_s ds} (r_t S_t dt + \sigma_t S_t dz_t) \\
&= (r_{t+2}) e^{\int_t^{t+2} r_s ds} S_t dt + \sigma_t e^{\int_t^{t+2} r_s ds} S_t dz_t \\
&= (r_{t+2}) X_t dt + \sigma_t X_t dz_t && \textit{SDE(2)}
\end{aligned}$$

Lets derive *SDE* for *T-forward* $F_t(\Gamma)$ as of time *t* under *Q* measure, using Ito's lemma :

$$\begin{aligned}
F_t(\Gamma) &= e^{\int_t^{\Gamma} r_s ds} S_t \\
dF_t(\Gamma) &= -r_t e^{\int_t^{\Gamma} r_s ds} S_t dt + e^{\int_t^{\Gamma} r_s ds} dS_t && \textit{by Leibniz Rule, see quant.doc} \\
&= -r_t e^{\int_t^{\Gamma} r_s ds} S_t dt + e^{\int_t^{\Gamma} r_s ds} (r_t S_t dt + \sigma_t S_t dz_t) \\
&= \sigma_t e^{\int_t^{\Gamma} r_s ds} S_t dz_t \\
&= \sigma_t F_t(\Gamma) dz_t && \textit{SDE(3)}
\end{aligned}$$

We have *3 SDEs* : today price, spot price and forward price. The main difference between X_t and $F_t(\Gamma)$ is that the former is always 2 days in the future, while the latter is a specific-fixed time point in the future.

Solve *SDE(1)*

$$\begin{aligned}
d\ln S_t &= \frac{1}{S_t}(r_t S_t dt + \sigma_t S_t dz_t) - \frac{1}{2S_t^{\,2}}(\sigma_t^2 S_t^2 dt) && \textit{since } \frac{\partial \ln S_t}{\partial S_t} dS_t + \frac{1}{2!}\frac{\partial^2 \ln S_t}{\partial S_t^{\,2}}(dS_t)^2 + \ldots \\
&= (r_t - \sigma_t^2/2)dt + \sigma_t dz_t \\
\ln S_T - \ln S_t &= \int_t^T (r_s - \sigma_s^2/2)ds + \int_t^T \sigma_s dz_s \\
S_T &= fwd \times e^{-v/2 + \sqrt{v}\varepsilon} && \textit{where } fwd = S_t e^{\int_t^T r_s ds},\ v = \int_t^T \sigma_s^2 ds \textit{ and } \varepsilon \textit{ is standard normal}
\end{aligned}$$

Solve *SDE(2)*

$$\begin{aligned}
d\ln X_t &= \frac{1}{X_t}(r_{t+2} X_t dt + \sigma_t X_t dz_t) - \frac{1}{2X_t^{\,2}}(\sigma_t^2 X_t^2 dt) && \textit{since } \frac{\partial \ln X_t}{\partial X_t} dX_t + \frac{1}{2!}\frac{\partial^2 \ln X_t}{\partial X_t^{\,2}}(dX_t)^2 + \ldots \\
&= (r_{t+2} - \sigma_t^2/2)dt + \sigma_t dz_t \\
\ln X_T - \ln X_t &= \int_t^T (r_{s+2} - \sigma_s^2/2)ds + \int_t^T \sigma_s dz_s \\
X_T &= fwd \times e^{-v/2 + \sqrt{v}\varepsilon} && \textit{where } fwd = X_t e^{\int_t^T r_{s+2} ds},\ v = \int_t^T \sigma_s^2 ds \textit{ and } \varepsilon \textit{ is standard normal}
\end{aligned}$$

Solve *SDE(3)*

$$\begin{aligned}
d\ln F_t(\Gamma) &= \frac{1}{F_t(\Gamma)}\sigma_t F_t(\Gamma) dz_t - \frac{1}{2F_t(\Gamma)^2}(\sigma_t^2 F_t(\Gamma)^2 dt) && \textit{since } \frac{\partial \ln F_t(\Gamma)}{\partial F_t(\Gamma)} dF_t(\Gamma) + \frac{1}{2!}\frac{\partial^2 \ln F_t(\Gamma)}{\partial F_t(\Gamma)^2}(dF_t(\Gamma))^2 + \ldots \\
&= (-\sigma_t^2/2)dt + \sigma_t dz_t \\
\ln F_T(\Gamma) - \ln F_t(\Gamma) &= \int_t^T (-\sigma_s^2/2)ds + \int_t^T \sigma_s dz_s \\
F_T(\Gamma) &= fwd \times e^{-v/2 + \sqrt{v}\varepsilon} && \textit{where } fwd = F_t(\Gamma),\ v = \int_t^T \sigma_s^2 ds \textit{ and } \varepsilon \textit{ is standard normal}
\end{aligned}$$

very similar but slightly different in fwd-value

2.2 Risk preference

Risk adverse is the tendency to avoid risk, hence risk premium is requested as a compensation for risk exposure. Risk seeking is the opposite, like different kinds of gambling for entertainment. Risk neutral is indifference in risk preference. Normally, human is risk adverse, hence the drift term in stock's *SDE* under physical measure does include risk premium (market price of risk). Market price of risk is stock-dependent, subjective and determined by the market (λ is different for different stocks, independent of σ).

$$\begin{aligned} dS_t &= \mu S_t dt + \sigma S_t dz_t \\ &= rS_t dt + \sigma S_t dz_t + \underbrace{\lambda \sigma S_t dt}_{premium} \qquad \textit{where } \lambda = (\mu - r)/\sigma \textit{ is market price of risk} \end{aligned}$$

2.3 Expectation and variance of stock price

$$E[s_T^k] = s_t^k e^{k(\mu-\sigma^2/2)(T-t)+k^2\sigma^2(T-t)/2} \quad or \quad E[s_T^k] = (F_t(T))^k e^{k(-v/2)+(k\sqrt{v})^2/2}$$

- $$E[s_T] = s_t e^{\mu(T-t)} \qquad E[s_T] = F_t(T) e^{(-v/2)+\sqrt{v}^2/2} = F_t(T)$$
- $$\begin{aligned} E[s_T^2] &= s_t^2 e^{2(\mu-\sigma^2/2)(T-t)+2^2\sigma^2(T-t)/2} & E[s_T^2] &= (F_t(T))^2 e^{2(-v/2)+(2\sqrt{v})^2/2} \\ &= s_t^2 e^{2\mu(T-t)-\sigma^2(T-t)+2\sigma^2(T-t)} & &= (F_t(T))^2 e^{-v+2v} \\ &= s_t^2 e^{(2\mu+\sigma^2)(T-t)} & &= (F_t(T))^2 e^{v} \end{aligned}$$
- $$\begin{aligned} V[s_T] &= E[s_T^2] - E[s_T]^2 & V[s_T] &= E[s_T^2] - E[s_T]^2 \\ &= s_t^2 e^{(2\mu+\sigma^2)(T-t)} - s_t^2 e^{2\mu(T-t)} & &= (F_t(T))^2 e^{v} - (F_t(T))^2 \\ &= s_t^2 e^{2\mu(T-t)}(e^{\sigma^2(T-t)} - 1) & &= (F_t(T))^2 (e^{v} - 1) \end{aligned}$$

2.4 Arithmetic return and geometric return

Arithmetic return is defined as :

$$\begin{aligned} \frac{s_{t+dt} - s_t}{s_t} &= ds_t / s_t \\ &= \mu dt + \sigma dZ_t \\ &= \varepsilon(\mu dt, \sigma\sqrt{dt}) \qquad \textit{arithmetic return is normal} \end{aligned}$$

Geometric return is defined as :

$$\begin{aligned} \ln\frac{s_{t+dt}}{s_t} &= \ln s_{t+dt} - \ln s_t \\ &= d\ln s_t \\ &= (\mu - \sigma^2/2)dt + \sigma dz_t \\ &= \varepsilon((\mu - \sigma^2/2)dt, \sigma\sqrt{dt}) \qquad \textit{geometric return is normal} \end{aligned}$$

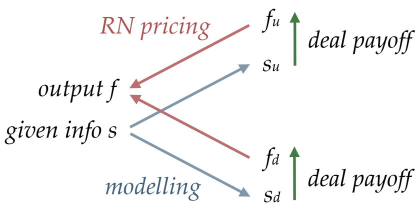
2.5 Multiplication and division of stock price

$$\begin{aligned} s_T s_U &= s_0 e^{(\mu-\sigma^2/2)T+\sigma z_T} \times s_0 e^{(\mu-\sigma^2/2)U+\sigma z_U} \qquad \textit{when } T < U \\ &= s_0^2 e^{(\mu-\sigma^2/2)(T+U)+\sigma(z_T+z_U)} \\ &= s_0^2 e^{(\mu-\sigma^2/2)(T+U)+\sigma(2z_T+z_{U-T})} \\ &= s_0^2 e^{(\mu-\sigma^2/2)(T+U)+\sigma\varepsilon(0,stdd=\sqrt{4T+U-T}=\sqrt{3T+U})} \\ \frac{s_T}{s_U} &= \frac{s_0 e^{(\mu-\sigma^2/2)T+\sigma z_T}}{s_0 e^{(\mu-\sigma^2/2)U+\sigma z_U}} \qquad \textit{when } T < U \\ &= e^{(\mu-\sigma^2/2)(T-U)+\sigma(z_T-z_U)} \\ &= e^{(\mu-\sigma^2/2)(T-U)+\sigma\varepsilon(0,stdd=\sqrt{U-T})} \end{aligned}$$

Option pricing

3.1 Discrete option pricing `3 = 1DAG + 2methods`

Two step model illustrated as a *DAG* :



Method 1 : Option pricing by (1) hedging and (2) no arbitrage

- hedging $\quad -f_u + \Delta s_u = -f_d + \Delta s_d \quad \Rightarrow \quad \Delta = \dfrac{f_u - f_d}{s_u - s_d}$
- no arbitrage $\quad -f_u + \Delta s_u = (-f + \Delta s)e^{rt}$

$$
\begin{aligned}
\Rightarrow \quad f &= (f_u - \Delta s_u)e^{-rt} + \Delta s && \textit{keeps the discount factor} \\
&= (f_u - \Delta s_u + \Delta s e^{rt})e^{-rt} && \textit{keeps the delta, factor it out} \\
&= (f_u - \Delta(s_u - s'))e^{-rt} && \textit{where } s' = se^{rt} \textit{ is forward price} \\
&= (f_u - \frac{f_u - f_d}{s_u - s_d}(s_u - s'))e^{-rt} && \textit{group } f_u \textit{ and } f_d \textit{ terms} \\
&= (f_u - \frac{s_u - s'}{s_u - s_d}(f_u - f_d))e^{-rt} \\
&= (f_u - p_d(f_u - f_d))e^{-rt} && \textit{define } p_d = \frac{s_u - s'}{s_u - s_d} = \frac{s_u - se^{rt}}{s_u - s_d} \\
&= ((1 - p_d)f_u + p_d f_d)e^{-rt} \\
&= (p_u f_u + p_d f_d)e^{-rt} && \textit{where } p_u = 1 - p_d
\end{aligned}
$$

Method 2 : Option pricing by replication using cash and delta shares

$$
c + \Delta s_u = f \quad \begin{cases} ce^{rt} + \Delta s_u = f_u \\ ce^{rt} + \Delta s_d = f_d \end{cases}
$$

$$
\begin{bmatrix} e^{rt} & s_u \\ e^{rt} & s_d \end{bmatrix} \begin{bmatrix} c \\ \Delta \end{bmatrix} = \begin{bmatrix} f_u \\ f_d \end{bmatrix}
$$

$$
\begin{bmatrix} c \\ \Delta \end{bmatrix} = \begin{bmatrix} \dfrac{f_u s_d - f_d s_u}{(s_d - s_u)e^{rt}} \\ \dfrac{(f_d - f_u)e^{rt}}{(s_d - s_u)e^{rt}} \end{bmatrix} \qquad \textit{applying Cramers rule}
$$

$$
= \begin{bmatrix} \dfrac{f_u s_d - f_d s_u}{s_d - s_u} e^{-rt} \\ \dfrac{f_u - f_d}{s_u - s_d} \end{bmatrix}
$$

$$
\begin{aligned}
f &= c + \Delta s_u \\
&= \frac{f_u s_d - f_d s_u}{s_d - s_u} e^{-rt} + \frac{f_u - f_d}{s_u - s_d} s_u \\
&= \ldots && \textit{group } f_u \textit{ and } f_d \textit{ terms, we will get the same result}
\end{aligned}
$$

3.2 Derive BSPDE `3 = 1DAG + 2methods`

Given a function $f(s_t,t)$, its differential is (we sometimes denote $f(s_t,t)$ as f_t for simplicity) denoted as a *DAG* :

$$
\begin{tikzcd}
df_t \arrow[r] \arrow[dr] & dt & \\
& dS_t \arrow[r] \arrow[dr] & dt \\
& & dz_t
\end{tikzcd}
$$

$$
\begin{aligned}
df_t &= \partial_t f_t dt + \partial_{s_t} f_t ds_t + \frac{1}{2}\left(\partial_t\partial_t f_t \cancel{(dt)^2} + 2\partial_t\partial_{s_t} f_t \cancel{dtds_t} + \partial_{s_t}\partial_{s_t} f_t (ds_t)^2\right) + \cancel{\ldots} \\
&= \partial_t f_t dt + \partial_{s_t} f_t ds_t + \frac{1}{2}\partial_{s_t}\partial_{s_t} f_t (ds_t)^2 \\
&= \partial_t f_t dt + \partial_{s_t} f_t (rs_t dt + \sigma s_t dz_t) + \frac{1}{2}\partial_{s_t}\partial_{s_t} f_t (\sigma^2 s_t^2 dt) \\
&= \left(\partial_t f_t + rs_t\partial_{s_t} f_t + \frac{1}{2}\sigma^2 s_t^2 \partial_{s_t}\partial_{s_t} f_t\right)dt + (\sigma s_t \partial_{s_t} f_t)dz_t
\end{aligned}
$$

These two parts are the same, one appears in Itos expansion, while one appears in BSPDE.

Method 1 : Option pricing by (1) hedging and (2) no arbitrage

$$
\begin{aligned}
-f_t + \Delta s_t &= \text{portfolio of selling option and hedging with underling} \\
-df_t + \Delta ds_t &= -\left(\partial_t f_t + \overset{once}{\boxed{rs_t\partial_{s_t} f_t}} + \frac{1}{2}\sigma^2 s_t^2 \partial_{s_t}\partial_{s_t} f_t\right)dt - (\sigma s_t \partial_{s_t} f_t)dz_t + \overset{twice}{\boxed{\Delta(rs_t dt}} + \sigma s_t dz_t) \\
&= \left(-\partial_t f_t + rs_t(\Delta - \partial_{s_t} f_t) - \frac{1}{2}\sigma^2 s_t^2 \partial_{s_t}\partial_{s_t} f_t\right)dt + (\sigma s_t(\Delta - \partial_{s_t} f_t))dz_t \\
&= \left(-\partial_t f_t + rs_t(\cancel{\partial_{s_t} f_t - \partial_{s_t} f_t}) - \frac{1}{2}\sigma^2 s_t^2 \partial_{s_t}\partial_{s_t} f_t\right)dt + (\sigma S_t(\cancel{\partial_{s_t} f_t - \partial_{s_t} f_t}))dz_t \\
&= \left(-\partial_t f_t - \frac{1}{2}\sigma^2 s_t^2 \partial_{s_t}\partial_{s_t} f_t\right)dt \\
&= (-f_t + \overset{thrice}{\boxed{s_t\partial_{s_t} f)rdt}} \\
(f_t - s_t\partial_{s_t} f)r &= \partial_t f_t + \frac{1}{2}\sigma^2 s_t^2 \partial_{s_t}\partial_{s_t} f_t \\
rf_t &= \partial_t f_t + rs_t\partial_{s_t} f + \frac{1}{2}\sigma^2 s_t^2 \partial_{s_t}\partial_{s_t} f_t \\
&= theta + rs_t delta + \frac{1}{2}\sigma^2 s_t^2 gamma
\end{aligned}
$$

term $rs_t\partial_{s_t} f_t$ *appears 3 times, 2 cancelled*

(1) hedging : remove z_t *by setting* $\Delta = \partial_{s_t} f_t$ *:*

(2) no arbitrage

known as BSPDE

Method 2 : Option pricing by ***Kolmogorov equation***. In order to illustrate this method, we consider 2 factor model :

$$
\begin{aligned}
dx_t &= \alpha_{1t}dt + \beta_{1t}dz_{1t} && \textit{factor 1} \\
dy_t &= \alpha_{2t}dt + \beta_{2t}dz_{2t} && \textit{factor 2} \\
dz_{1t}dz_{2t} &= \rho dt && \textit{two factors are correlated}
\end{aligned}
$$

Alphas are drift terms while betas are diffusion terms. The answer can be read directly :

$$
\begin{aligned}
rf_t &= \partial_t f + \alpha_{1t}(\partial_x f) + \alpha_{2t}(\partial_y f) + \frac{1}{2}\beta_{1t}^2(\partial_{xx} f) + \rho\beta_{1t}\beta_{2t}(\partial_{xy} f) + \frac{1}{2}\beta_{2t}^2(\partial_{yy} f) \\
&= theta + \alpha_{1t} delta + \alpha_{2t} vega + \frac{1}{2}\beta_{1t}^2 gamma + \rho\beta_{1t}\beta_{2t} vanna + \frac{1}{2}\beta_{2t}^2 vomma
\end{aligned}
$$

3.3 Solve BSPDE analytically (by heat equation)

skipped

We will transform BSPDE into one dimensional heat equation by transformation from (s_t, t) space into (x_τ, τ) space :

$$
\begin{array}{lll}
x_\tau & = & \ln s_t \\
\tau & = & \sigma^2(T-t)/2
\end{array}
$$

i.e. x_τ ~ normal

i.e. remaining variance

and consider the solution with form :

$$f(s_t,t) \;=\; g(x_\tau,\tau) \;=\; e^{\alpha x_\tau+\beta\tau}h(x_\tau,\tau)$$

With this transformation, we have a diagonal Jacobian matrix :

$$\begin{bmatrix} \partial_s x & \partial_t x \\ \partial_s \tau & \partial_t \tau \end{bmatrix} = \begin{bmatrix} 1/s & 0 \\ 0 & -\sigma^2/2 \end{bmatrix}$$

$$
\Rightarrow\quad
\begin{array}{lllllll}
\partial_t h & = & \partial_x h\partial_t x+\partial_\tau h\partial_t\tau & = & (-\sigma^2/2)\partial_\tau h \\
\partial_s h & = & \partial_x h\partial_s x+\partial_\tau h\partial_s\tau & = & \partial_x h/s \\
\partial_s\partial_s h & = & \partial_s((\partial_x h)/s) & = & \partial_x\partial_x h\partial_s x/s-\partial_x h/s^2 & = & \partial_x\partial_x h/s^2-\partial_x h/s^2
\end{array}
$$

The idea of the proof is to express $\partial_t f$, $\partial_s f$ and $\partial_s\partial_s f$ in terms of $\partial_\tau h$, $\partial_x h$ and $\partial_x\partial_x h$, plug them into BSPDE, convert it into a new PDE in space (x,τ), then reduce it to homogeneous heat equation by specific α and β values, finally solve BSPDE using standard solutions for heat equation.

$$
\begin{array}{lllll}
\partial_t f & = & \partial_t(e^{\alpha x+\beta\tau}h) & = & (\beta e^{\alpha x+\beta\tau}\partial_t\tau)h+(e^{\alpha x+\beta\tau})\partial_t h \\
 & & & = & e^{\alpha x+\beta\tau}(-\sigma^2/2)\beta h+e^{\alpha x+\beta\tau}(-\sigma^2/2)\partial_\tau h \\
 & & & = & e^{\alpha x+\beta\tau}(-\sigma^2/2)(\beta h+\partial_\tau h) \\
\partial_s f & = & \partial_s(e^{\alpha x+\beta\tau}h) & = & (\alpha e^{\alpha x+\beta\tau}\partial_s x)h+(e^{\alpha x+\beta\tau})\partial_s h \\
 & & & = & (e^{\alpha x+\beta\tau}/s)\alpha h+(e^{\alpha x+\beta\tau}/s)\partial_x h \\
 & & & = & (e^{\alpha x+\beta\tau}/s)(\alpha h+\partial_x h) \\
\partial_s\partial_s f & = & \partial_s\partial_s(e^{\alpha x+\beta\tau}h) & = & \partial_s((e^{\alpha x+\beta\tau}/s)(\alpha h+\partial_x h)) \\
 & & & = & \partial_s(e^{\alpha x+\beta\tau}/s)(\alpha h+\partial_x h)+(e^{\alpha x+\beta\tau}/s)\partial_s(\alpha h+\partial_x h) \\
 & & & = & (\alpha e^{\alpha x+\beta\tau}\partial_s x/s-e^{\alpha x+\beta\tau}/s^2)(\alpha h+\partial_x h)+(e^{\alpha x+\beta\tau}/s)(\alpha\partial_x h\partial_s x+\partial_x\partial_x h\partial_s x) \\
 & & & = & (\alpha e^{\alpha x+\beta\tau}/s^2-e^{\alpha x+\beta\tau}/s^2)(\alpha h+\partial_x h)+(e^{\alpha x+\beta\tau}/s)(\alpha\partial_x h/s+\partial_x\partial_x h/s) \\
 & & & = & e^{\alpha x+\beta\tau}/s^2((\alpha-1)(\alpha h+\partial_x h)+(\alpha\partial_x h+\partial_x\partial_x h)) \\
 & & & = & e^{\alpha x+\beta\tau}/s^2((\alpha-1)\alpha h+(\alpha-1)\partial_x h+\alpha\partial_x h+\partial_x\partial_x h)
\end{array}
$$

Replace all derivatives and remove all $e^{\alpha x+\beta\tau}$ terms from both sides, we have :

$$
\begin{array}{lll}
rf & = & \partial_t f+rs\partial_s f+(\sigma^2/2)s^2\partial_s\partial_s f \\
rh & = & (-\sigma^2/2)(\beta h+\partial_\tau h)+rs(1/s)(\alpha h+\partial_x h)+(\sigma^2/2)s^2(1/s^2)((\alpha-1)\alpha h+(\alpha-1)\partial_x h+\alpha\partial_x h+\partial_x\partial_x h) \\
 & = & (-\sigma^2/2)(\beta h+\partial_\tau h)+r(\alpha h+\partial_x h)+(\sigma^2/2)((\alpha-1)\alpha h+(\alpha-1)\partial_x h+\alpha\partial_x h+\partial_x\partial_x h) \\
0 & = & (-\sigma^2/2)\beta h+(-\sigma^2/2)\partial_\tau h+r\alpha h+r\partial_x h+(\sigma^2/2)(\alpha-1)\alpha h+(\sigma^2/2)(\alpha-1)\partial_x h+(\sigma^2/2)\alpha\partial_x h+(\sigma^2/2)\partial_x\partial_x h-rh \\
 & = & (-r+r\alpha+(\sigma^2/2)(\alpha-1)\alpha-(\sigma^2/2)\beta)h+(-\sigma^2/2)\partial_\tau h+(r+(\sigma^2/2)(\alpha-1)+(\sigma^2/2)\alpha)\partial_x h+(\sigma^2/2)\partial_x\partial_x h
\end{array}
$$

The above PDE can be converted into heat equation if we remove $\partial_\tau h$ and $\partial_x h$ term by setting :

$$0 = r + (\sigma^2/2)(\alpha - 1) + (\sigma^2/2)\alpha \quad \text{and} \quad 0 = -r + r\alpha + (\sigma^2/2)(\alpha-1)\alpha - (\sigma^2/2)\beta$$

$$\begin{array}{rcl} \Rightarrow \quad 0 &=& r - \sigma^2/2 + \sigma^2\alpha \\ \alpha &=& 1/2 - r/\sigma^2 \end{array}$$

$$\begin{array}{rcl} \Rightarrow \quad (\sigma^2/2)\beta &=& -r + r\alpha + (\sigma^2/2)(\alpha-1)\alpha \\ &=& (\alpha-1)(r + (\sigma^2/2)\alpha) \\ &=& (-1/2 - r/\sigma^2)(r + (\sigma^2/2)(1/2 - r/\sigma^2)) \\ &=& (-1/2 - r/\sigma^2)(r + \sigma^2/4 - r/2) \\ &=& (-1/2 - r/\sigma^2)(r/2 + \sigma^2/4) \\ &=& -(1/2 + r/\sigma^2)(r/\sigma^2 + 1/2)(\sigma^2/2) \\ &=& -(1/2 + r/\sigma^2)^2(\sigma^2/2) \\ \beta &=& -(1/2 + r/\sigma^2)^2 \end{array}$$

Thus the PDE becomes :

$$\partial_\tau h = \partial_x \partial_x h$$

and the boundary condition becomes :

$$\begin{array}{rcccl} f(s,T) &=& e^{\alpha x + \beta\tau} h(x,\tau)\Big|_{x=\ln s, \tau=\sigma^2(T-T)/2=0} &=& e^{\alpha x} h(\ln s, 0) \\ \Rightarrow \quad h(x,0) &=& e^{-\alpha x} f(e^x, T) &=& e^{-\alpha x}(e^x - k)^+ \end{array}$$

which can be considered as heat source in one dimensional space x at time $\tau = 0$. Solution for homogeneous heat equation $\partial_\tau h = \partial_x \partial_x h$ with boundary condition $h(x,0)$ is (see remark) :

$$\begin{array}{rcl} h(x,\tau) &=& h(x,\tau=0) * \psi(x,\tau) \\ &=& \int_{-\infty}^{+\infty} h(y,0)\psi(x-y,\tau)dy \\ &=& \frac{1}{\sqrt{2\pi(2\tau)}} \int_{-\infty}^{+\infty} e^{-\alpha y}(e^y - k)^+ e^{-(x-y)^2/(2(2\tau))} dy \\ &=& \frac{1}{\sqrt{2\pi(2\tau)}} \int_{\ln k}^{+\infty} e^{-\alpha y} e^y e^{-(x-y)^2/(2(2\tau))} dy - \frac{k}{\sqrt{2\pi(2\tau)}} \int_{\ln k}^{+\infty} e^{-\alpha y} e^{-(x-y)^2/(2(2\tau))} dy \\ &=& \frac{1}{\sqrt{2\pi(2\tau)}} \int_{\ln k}^{+\infty} e^{-(y^2 - 2xy - 4\tau(1-\alpha)y + x^2)/(2(2\tau))} dy - \frac{k}{\sqrt{2\pi(2\tau)}} \int_{\ln k}^{+\infty} e^{-(y^2 - 2xy + 4\tau\alpha y + x^2)/(2(2\tau))} dy \\ &=& \frac{1}{\sqrt{2\pi(2\tau)}} \int_{\ln k}^{+\infty} e^{-(y^2 - 2(x+2\tau(1-\alpha))y + x^2)/(2(2\tau))} dy - \frac{k}{\sqrt{2\pi(2\tau)}} \int_{\ln k}^{+\infty} e^{-(y^2 - 2(x-2\tau\alpha)y + x^2)/(2(2\tau))} dy \\ &=& \left[\begin{array}{l} \frac{1}{\sqrt{2\pi(2\tau)}} \int_{\ln k}^{+\infty} e^{-(x^2 - (x+2\tau(1-\alpha))^2)/(2(2\tau))} e^{-(y-(x+2\tau(1-\alpha)))^2/(2(2\tau))} dy - \\ \frac{k}{\sqrt{2\pi(2\tau)}} \int_{\ln k}^{+\infty} e^{-(x^2 - (x-2\tau\alpha)^2)/(2(2\tau))} e^{-(y-(x-2\tau\alpha))^2/(2(2\tau))} dy \end{array} \right. \\ &=& u_1 N(v_1) - u_2 N(v_2) \end{array}$$

where

$$\begin{array}{rclcrcl} u_1 &=& e^{-(x^2 - (x+2\tau(1-\alpha))^2)/(2(2\tau))} & \text{and} & u_2 &=& k e^{-(x^2 - (x-2\tau\alpha)^2)/(2(2\tau))} \\ d_1 &=& -\frac{\ln k - (x + 2\tau(1-\alpha))}{\sqrt{2\tau}} & \text{and} & d_2 &=& -\frac{\ln k - (x - 2\tau\alpha)}{\sqrt{2\tau}} \end{array}$$

$$\begin{array}{rcl} \Rightarrow \quad f(s,t) &=& e^{\alpha x + \beta\tau} h(x,\tau)\Big|_{x=\ln s_t, \tau=\sigma^2(T-t)/2} \\ &=& (e^{\alpha x + \beta\tau} u_1 N(d_1) - e^{\alpha x + \beta\tau} u_2 N(d_2))\Big|_{x=\ln s_t, \tau=\sigma^2(T-t)/2} \\ &=& (w_1 N(d_1) - w_2 N(d_2))\Big|_{x=\ln s_t, \tau=\sigma^2(T-t)/2} \end{array}$$

substitute $x = \ln s$, $\tau = \sigma^2(T-t)/2$ and $\alpha = 1/2 - r/\sigma^2$ into d_1 and d_2, we have :

$$\begin{aligned} d_1 &= -\frac{\ln k-(x+2\tau(1-\alpha))}{\sqrt{2\tau}} \\ &= -\frac{\ln k-(\ln s+2(\sigma^2(T-t)/2)(1/2+r/\sigma^2))}{\sqrt{2\sigma^2(T-t)/2}} \\ &= \frac{\ln s-\ln k+\sigma^2(T-t)(1/2+r/\sigma^2)}{\sqrt{\sigma^2(T-t)}} \\ &= \frac{\ln s/k+(r+\sigma^2/2)(T-t)}{\sigma\sqrt{T-t}} \end{aligned}$$

$$\begin{aligned} d_2 &= -\frac{\ln k-(x-2\tau\alpha)}{\sqrt{2\tau}} \\ &= -\frac{\ln k-(\ln s-2(\sigma^2(T-t)/2)(1/2-r/\sigma^2))}{\sqrt{2\sigma^2(T-t)/2}} \\ &= \frac{\ln s-\ln k-\sigma^2(T-t)(1/2-r/\sigma^2)}{\sqrt{\sigma^2(T-t)}} \\ &= \frac{\ln s/k+(r-\sigma^2/2)(T-t)}{\sigma\sqrt{T-t}} \end{aligned}$$

Consider term w_1 and w_2 :

$$\begin{aligned} w_1 &= e^{\alpha x+\beta\tau}u_1 \\ &= e^{\alpha x+\beta\tau}e^{-(x^2-(x+2\tau(1-\alpha))^2)/(2(2\tau))} \\ &= e^{\alpha x+\beta\tau}e^{(4\tau(1-\alpha)x+4\tau^2(1-\alpha)^2)/(4\tau)} \\ &= e^{\alpha x+\beta\tau}e^{((1-\alpha)x+\tau(1-\alpha)^2)} \\ &= e^{x+\tau((1-\alpha)^2+\beta)} \end{aligned}$$

$$\begin{aligned} w_2 &= e^{\alpha x+\beta\tau}u_2 \\ &= ke^{\alpha x+\beta\tau}e^{-(x^2-(x-2\tau\alpha)^2)/(2(2\tau))} \\ &= ke^{\alpha x+\beta\tau}e^{(-4\tau\alpha x+4\tau^2\alpha^2))/(4\tau)} \\ &= ke^{\alpha x+\beta\tau}e^{(-\alpha x+\tau\alpha^2)} \\ &= ke^{\tau(\alpha^2+\beta)} \end{aligned}$$

substitute $x = \ln s$, $\tau = \sigma^2(T-t)/2$ and $\alpha = 1/2 - r/\sigma^2$, $\beta = -(1/2+r/\sigma^2)^2$ into w_1 and w_2, we have :

$$\begin{aligned} w_1 &= e^{\ln s+(\sigma^2(T-t)/2)\times((1/2+r/\sigma^2)^2-(1/2+r/\sigma^2)^2)} \\ &= e^{\ln s} \\ &= s \end{aligned}$$

$$\begin{aligned} w_2 &= ke^{\tau(\alpha^2+\beta)} \\ &= ke^{(\sigma^2(T-t)/2)\times((1/2-r/\sigma^2)^2-(1/2+r/\sigma^2)^2)} \\ &= ke^{(\sigma^2(T-t)/2)\times(-2r/\sigma^2)} \\ &= ke^{-r(T-t)} \end{aligned}$$

thus $\quad f(s,t) = sN(d_1)-ke^{-r(T-t)}N(d_2)$

Remark – Differentiation of integral

$$\begin{aligned} \partial_x \int f(x,y)dy &= \lim_{\Delta x\to 0}\left(\int f(x+\Delta x,y)dy-\int f(x,y)dy\right)/\Delta x \\ &= \lim_{\Delta x\to 0}\left(\int (f(x,y)+\partial_x f(x,y)\Delta x+\partial_x\partial_x f(x,y)(\Delta x)^2/2+...)dy-\int f(x,y)dy\right)/\Delta x \\ &= \lim_{\Delta x\to 0}\left(\int (\partial_x f(x,y)\Delta x+\partial_x\partial_x f(x,y)(\Delta x)^2/2+...)dy/\Delta x\right. \\ &= \lim_{\Delta x\to 0}\left(\int (\partial_x f(x,y)+\partial_x\partial_x f(x,y)(\Delta x)/2+...)dy\right. \\ &= \int \partial_x f(x,y)dy \end{aligned}$$

Remark – Standard solution to heat equation

The solution for homogeneous heat equation $\partial_\tau h = c\partial_x\partial_x h$ with boundary condition $h(x,0)$ is the convolution between boundary condition with fundalmental solution $\psi(x,\tau)$, which is a zero mean Gaussian function with spread equals to $(2c\tau)^{1/2}$. It can be regarded as a Gaussian smoothing of the heat source through time τ.

$$h(x,\tau) = h(x,\tau=0)*\psi(x,\tau) = \int_{-\infty}^{+\infty}h(y,0)\psi(x-y,\tau)dy$$

where $\quad \psi(x,\tau) = \dfrac{1}{\sqrt{2\pi(2c\tau)}}e^{-x^2/(2(2c\tau))}$

recall $\quad (f*g)(x) = \int_{-\infty}^{+\infty}f(y)g(x-y)dy$

Let prove the homogeneous heat equation solution by substitution.

$$\Rightarrow \partial_\tau h(x,\tau) = \partial_\tau \int_{-\infty}^{+\infty} h(y,0)\psi(x-y,\tau)dy = \int_{-\infty}^{+\infty} h(y,0)\partial_\tau \psi(x-y,\tau)dy \qquad \text{equation 3a}$$

$$\partial_x h(x,\tau) = \partial_x \int_{-\infty}^{+\infty} h(y,0)\psi(x-y,\tau)dy = \int_{-\infty}^{+\infty} h(y,0)\partial_x \psi(x-y,\tau)dy \qquad \text{equation 3b}$$

$$\partial_x\partial_x h(x,\tau) = \partial_x\partial_x \int_{-\infty}^{+\infty} h(y,0)\psi(x-y,\tau)dy = \int_{-\infty}^{+\infty} h(y,0)\partial_x\partial_x \psi(x-y,\tau)dy \qquad \text{equation 3c}$$

$$\begin{aligned}
\Rightarrow \partial_\tau \psi(x,\tau) &= -\frac{1}{2}\frac{1}{2\pi(2c\tau)\sqrt{2\pi(2c\tau)}}(2\pi(2c))e^{-x^2/(2(2c\tau))} + \frac{1}{\sqrt{2\pi(2c\tau)}}\frac{x^2}{2(2c\tau^2)}e^{-x^2/(2(2c\tau))} \\
&= -\frac{1}{2\tau}\frac{1}{\sqrt{2\pi(2c\tau)}}e^{-x^2/(2(2c\tau))} + \frac{1}{\sqrt{2\pi(2c\tau)}}\frac{x^2}{2(2c\tau^2)}e^{-x^2/(2(2c\tau))} \\
&= \left(-\frac{1}{2\tau} + \frac{x^2}{2(2c\tau^2)}\right)\psi(x,\tau) \\
\partial_x \psi(x,\tau) &= \frac{1}{\sqrt{2\pi(2c\tau)}}\frac{-2x}{2(2c\tau)}e^{-x^2/(2(2c\tau))} \\
&= \frac{-x}{2c\tau}\psi(x,\tau) \\
\partial_x\partial_x \psi(x,\tau) &= \partial_x\left(\frac{-x}{2c\tau}\psi(x,\tau)\right) \\
&= \frac{-1}{2c\tau}\psi(x,\tau) + \frac{-x}{2c\tau}\partial_x \psi(x,\tau) \\
&= \frac{-1}{2c\tau}\psi(x,\tau) + \frac{-x}{2c\tau}\frac{-x}{2c\tau}\psi(x,\tau) \\
&= \frac{\partial_\tau \psi(x,\tau)}{c}
\end{aligned}$$

$$\begin{aligned}
\Rightarrow \partial_\tau \psi(x,\tau) &= c\partial_x\partial_x \psi(x,\tau) \\
\partial_\tau h(x,\tau) &= c\partial_x\partial_x h(x,\tau)
\end{aligned}$$

skipped

3.4 Solve BSPDE numerically (by finite difference)

(1) Assume $f(n,m) = f(t,s_t)$, where $(t=n\Delta t,\ T=N\Delta t)$ and $(s_t=m\Delta s,\ max(s_t)=M\Delta s)$, then the current option price is $f(0,s_0/\Delta s)$.

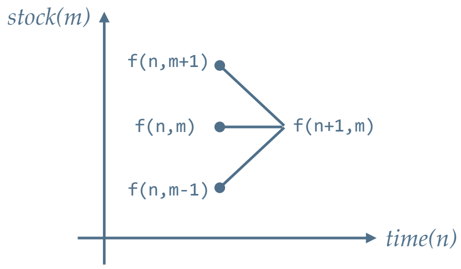
$$
\begin{aligned}
f &= f(n,m) \\
\partial_t f &= (f(n+1,m)-f(n,m))/\Delta t \\
\partial_s f &= (f(n,m+1)-f(n,m-1))/(2\Delta s) \\
\partial_s\partial_s f &= ((f(n,m+1)-f(n,m))/\Delta s-(f(n,m)-f(n,m-1))/\Delta s)/\Delta s \\
&= (f(n,m+1)-2f(n,m)+f(n,m-1))/(\Delta s)^2
\end{aligned}
$$

(2) For ***forward propagation***, we find $f(n+1,m)$ in terms of $f(n,m-1)$, $f(n,m)$ and $f(n,m+1)$.

$$
\begin{aligned}
rf &= \partial_t f + rs\partial_s f + (\sigma^2 s^2/2)\partial_s\partial_s f \\
rf(n,m) &= (f(n+1,m)-f(n,m))/\Delta t + rm\Delta s(f(n,m+1)-f(n,m-1))/(2\Delta s) + (\sigma^2(m\Delta s)^2/2)(f(n,m+1)-2f(n,m)+f(n,m-1))/(\Delta s)^2 \\
r\Delta t f(n,m) &= f(n+1,m)-f(n,m)+(rm\Delta t/2)(f(n,m+1)-f(n,m-1))+(\sigma^2 m^2\Delta t/2)(f(n,m+1)-2f(n,m)+f(n,m-1)) \\
f(n+1,m) &= A_m f(n,m-1)+B_m f(n,m)+C_m f(n,m+1)
\end{aligned}
$$

$$
\begin{bmatrix} A_m \\ B_m \\ C_m \end{bmatrix} = \begin{bmatrix} rm\Delta t/2-\sigma^2 m^2\Delta t/2 \\ 1+r\Delta t+\sigma^2 m^2\Delta t \\ -rm\Delta t/2-\sigma^2 m^2\Delta t/2 \end{bmatrix}
$$

$$
\begin{bmatrix} A_0 \\ B_0 \\ C_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1+r\Delta t \\ 0 \end{bmatrix}
$$



Please check how to handle the case when $m = M$ better, *(my suggestion)* should we add a constant c for $m = M$ case? For example, if M is very large, and if f is a plain vanilla call option, then we can set asymptotic value for boundary case $f(n,M+1)$.

$$
\begin{bmatrix} f(n+1,0) \\ f(n+1,1) \\ f(n+1,2) \\ \dots \\ f(n+1,M-1) \\ f(n+1,M) \end{bmatrix} = \begin{bmatrix} B_0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ A_1 & B_1 & C_1 & 0 & \dots & 0 & 0 & 0 \\ 0 & A_2 & B_2 & C_2 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & A_{M-1} & B_{M-1} & C_{M-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & A_M & B_M \end{bmatrix} \begin{bmatrix} f(n,0) \\ f(n,1) \\ f(n,2) \\ \dots \\ f(n,M-1) \\ f(n,M) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dots \\ 0 \\ C_M\, payoff(M\Delta s) \end{bmatrix} \qquad \text{i.e. } F_{n+1} = GF_n + H
$$

(3) For ***backward propagation***, we simply inverse the above matrix equation : $F_n = G^{-1}(F_{n+1}-H)$.

3.5 Derive risk neutral pricing

According to *FTAP*, we can construct martingale by fulfilling two requirements :

(1) numeraire deflatted security price is martingale only when ...

(2) under risk neutral measure implied by that numeraire

For example, using *USD* cash as numeraire, its *RN measure* is *OIS* risk free rate :

$$
\begin{aligned}
(3) \qquad d(e^{-rt}f) &= (-re^{-rt}f + e^{-rt}\partial_t f)dt + (e^{-rt}\partial_s f)ds + 1/2(e^{-rt}\partial_s\partial_s f)(ds)^2 \\
&= (-re^{-rt}f + e^{-rt}\partial_t f)dt + (e^{-rt}\partial_s f)(rsdt + \sigma sdz_t) + 1/2(e^{-rt}\partial_s\partial_s f)(rsdt + \sigma sdz_t)^2 \\
&= (-re^{-rt}f + e^{-rt}\partial_t f)dt + (e^{-rt}\partial_s f)(rsdt + \sigma sdz_t) + 1/2(e^{-rt}\partial_s\partial_s f)(\sigma^2 s^2 dt) \\
&= (-re^{-rt}f + e^{-rt}\partial_t f + e^{-rt}rs\partial_s f + e^{-rt}1/2\sigma^2 s^2\partial_s\partial_s f)dt + (e^{-rt}\sigma s\partial_s f)dz_t \\
&= e^{-rt}\cancel{(-rf + \partial_t f + rs\partial_s f + 1/2\sigma^2 s^2\partial_s\partial_s f)}dt + e^{-rt}(\sigma s\partial_s f)dz_t \qquad \text{By BSPDE} \\
&= \underbrace{e^{-rt}(\sigma s\partial_s f)}_{k}dz_t \\
&= kdz_t
\end{aligned}
$$

$$
\begin{aligned}
\Rightarrow \qquad e^{-rt}f(s_t,t) &= \hat{E}[e^{-rT}f(s_T,T)\,|\,s_t] \\
\Rightarrow \qquad f(s_t,t) &= e^{-r(T-t)}\hat{E}[f(s_T,T)\,|\,s_t]
\end{aligned}
$$

3.6 Solve risk neutral pricing analytically

$$
\begin{aligned}
f(s_t,t) &= e^{-r(T-t)}\hat{E}[(s_T-k)^+\,|\,s_t] \\
&= e^{-r(T-t)}\hat{E}[(Fe^{-v/2+\sqrt{v}\varepsilon}-k)^+] && \textit{where } s_T = Fe^{-v/2+\sqrt{v}\varepsilon} \\
&= \frac{e^{-r(T-t)}}{\sqrt{2\pi}}\int_{-\infty}^{\infty}(Fe^{-v/2+\sqrt{v}x}-k)^+e^{-x^2/2}dx && \textit{where } F = s_t e^{r(T-t)} \textit{ and } v = \sigma^2(T-t) \\
&= \frac{e^{-r(T-t)}}{\sqrt{2\pi}}\int_{-d_2}^{\infty}(Fe^{-v/2+\sqrt{v}x}-k)e^{-x^2/2}dx && \textit{non zero for } x > (\ln(k/F)+v/2)/\sqrt{v} = -d_2 \\
&= \frac{e^{-r(T-t)}}{\sqrt{2\pi}}\left[F\int_{-d_2}^{\infty}e^{-x^2/2+\sqrt{v}x-v/2}dx - k\int_{-d_2}^{\infty}e^{-x^2/2}dx\right] \\
&= \frac{e^{-r(T-t)}}{\sqrt{2\pi}}\left[F\int_{-d_2}^{\infty}e^{-(x-\sqrt{v})^2/2}dx - k\int_{-d_2}^{\infty}e^{-x^2/2}dx\right] \\
&= \frac{e^{-r(T-t)}}{\sqrt{2\pi}}\left[F\int_{-d_2-\sqrt{v}}^{\infty}e^{-y^2/2}dy - k\int_{-d_2}^{\infty}e^{-x^2/2}dx\right] && \textit{where } y = x-\sqrt{v} \\
&= \frac{e^{-r(T-t)}}{\sqrt{2\pi}}\left[F\int_{-d_1}^{\infty}e^{-y^2/2}dy - k\int_{-d_2}^{\infty}e^{-x^2/2}dx\right] && \textit{where } d_1 = d_2+\sqrt{v} \\
&= \frac{e^{-r(T-t)}}{\sqrt{2\pi}}\left[F\int_{-\infty}^{d_1}e^{-y^2/2}dy - k\int_{-\infty}^{d_2}e^{-x^2/2}dx\right] && \textit{integration } (-d,\infty) = \textit{integration } (-\infty, d) \textit{ for } \varepsilon \\
&= e^{-r(T-t)}[FN(d_1)-kN(d_2)]
\end{aligned}
$$

This holds true for stocks offering dividend yield g, just need to modify :

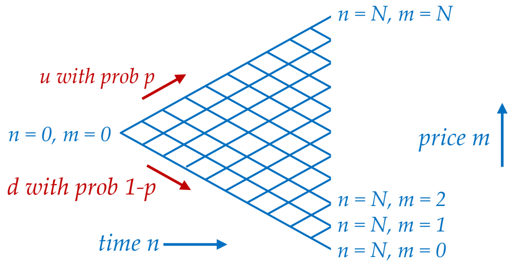
- forward becomes $F = s_t e^{(r-g)(T-t)}$
- variance unchanged
- discount factor unchanged

3.7 Solve risk neutral pricing numerically (by tree)

3.7a Tree topology

Binomial tree is characterised by 3 parameters upward scale u, downward scale d and upward probability p. If all these parameters are independent of time, then the tree is a *recombining tree* :

- there is one root s_t
- there are $N+1$ leaves at the maturity side, where $T-t = N\Delta t$
- there are ${}_NC_m$ different paths joining s_t and $s_T = s_t u^m d^{N-m}$



3.7b Calibration with CRR model

Calibration is done by matching 1st moment, 2nd moment of stock prices, together with constraint $u = 1/d$. We have three equations solving for three unknowns.

$$
\begin{array}{llllll}
\Rightarrow & \hat{E}[s_{t+\Delta t} \mid I_t] & = & s_t e^{(r-\sigma^2/2)\Delta t+\sigma^2\Delta t/2} & = & p(us_t)+(1-p)(ds_t) & \textit{1st moment} \\
& \hat{E}[s_{t+\Delta t}^2 \mid I_t] & = & s_t^2 e^{2(r-\sigma^2/2)\Delta t+2^2\sigma^2\Delta t/2} & = & p(us_t)^2+(1-p)(ds_t)^2 & \textit{2nd moment}
\end{array}
$$

$$
\begin{array}{llll}
\Rightarrow & s_t e^{r\Delta t} & = & p(us_t)+(1-p)(ds_t) \\
& s_t^2 e^{2r\Delta t+\sigma^2\Delta t} & = & p(us_t)^2+(1-p)(ds_t)^2
\end{array}
$$

$$
\begin{array}{llll}
\Rightarrow & e^{r\Delta t} & = & pu+(1-p)d \\
& e^{2r\Delta t+\sigma^2\Delta t} & = & pu^2+(1-p)d^2
\end{array}
$$

$$
\begin{array}{llllll}
\Rightarrow & F & = & p(u-d)+d & \Rightarrow \; p = (F-d)/(u-d) & \textit{substitute into another equation} \\
& F^2 e^{\sigma^2\Delta t} & = & p(u^2-d^2)+d^2 & & \textit{substitute } F = \$1\times e^{r\Delta t} \textit{ for simplicity}
\end{array}
$$

$$
\begin{array}{lllll}
\Rightarrow & F^2 e^{\sigma^2\Delta t} & = & (F-d)/(u-d)\times(u^2-d^2)+d^2 & \textit{simplify } u^2\textit{-}d^2 \\
& & = & (F-d)(u+d)+d^2 & \\
& & = & F(u+d)-ud & \\
& & = & F(u+1/u)-1 &
\end{array}
$$

$$
\begin{array}{llll}
\Rightarrow & 0 & = & Fu^2-(F^2 e^{\sigma^2\Delta t}+1)u+F \\
& & = & u^2-(Fe^{\sigma^2\Delta t}+F^{-1})u+1
\end{array}
$$

$$
\begin{array}{lllll}
\Rightarrow & u & = & \left((Fe^{\sigma^2\Delta t}+F^{-1})\pm\sqrt{(Fe^{\sigma^2\Delta t}+F^{-1})^2-4}\right)/2 & \textit{recall that } F = \$1\times e^{r\Delta t} \\
& d & = & 1/u & \\
& p & = & (F-d)/(u-d) &
\end{array}
$$

There is no solution if determinant is negative : $|Fe^{\sigma^2\Delta t}+F^{-1}|<2$.

3.7c Calibration with 1/2 probability model

Calibration is done by matching 1st moment, 2nd moment of stock prices, together with constraint $p = 1/2$. We have three equations solving for three unknowns.

$$\Rightarrow \quad \hat{E}[s_{t+\Delta t} \mid I_t] = s_t e^{(r-\sigma^2/2)\Delta t+\sigma^2\Delta t/2} = (us_t + ds_t)/2 \qquad \textit{1st moment}$$

$$\hat{E}[s_{t+\Delta t}^2 \mid I_t] = s_t^2 e^{2(r-\sigma^2/2)\Delta t+2^2\sigma^2\Delta t/2} = ((us_t)^2 + (ds_t)^2)/2 \qquad \textit{2nd moment}$$

$$\Rightarrow \quad s_t e^{r\Delta t} = (us_t + ds_t)/2$$

$$s_t^2 e^{2r\Delta t+\sigma^2\Delta t} = ((us_t)^2 + (ds_t)^2)/2$$

$$\Rightarrow \quad e^{r\Delta t} = (u+d)/2$$

$$e^{2r\Delta t+\sigma^2\Delta t} = (u^2+d^2)/2$$

$$\Rightarrow \quad F = (u+d)/2 \quad \Rightarrow \quad u = 2F-d \qquad \textit{substitute into another equation}$$

$$F^2 e^{\sigma^2\Delta t} = (u^2+d^2)/2 \qquad \textit{substitute } F = \$1\times e^{r\Delta t} \textit{ for simplicity}$$

$$\begin{aligned}
\Rightarrow \quad F^2 e^{\sigma^2\Delta t} &= ((2F-d)^2+d^2)/2 \\
&= 2F^2 - 2Fd + d^2 \\
0 &= d^2 - 2Fd + F^2(2-e^{\sigma^2\Delta t}) \\
d &= \left(2F \pm \sqrt{4F^2 - 4F^2(2-e^{\sigma^2\Delta t})}\right)/2 \\
&= F \pm \sqrt{F^2 - F^2(2-e^{\sigma^2\Delta t})} \\
&= F\left(1 \pm \sqrt{-1+e^{\sigma^2\Delta t}}\right)
\end{aligned}$$

3.7d Pricing with backward propagation

Suppose matrix `f(n,m)` denotes option price when underlying price $s_{t+n\Delta t} = s_t u^m d^{n-m}$, then :

$$f(N,m) = \textit{payoff}(s_{t+N\Delta t} = s_t u^m d^{N-m})$$

$$f(n,m) = e^{-r\Delta t}(pf(n+1,m+1)+(1-p)f(n+1,m)) \qquad \textit{for } n \in [0, N-1]$$

```
double DF = exp(-r*dt);
for(m=0; m<=N; ++m) // looping in final payoff
{
    double ST = st* (u^m) * (d^(N-m));
    f(N,m) = payoff(ST);
}
for(n=N-1; n>=0; --n) // looping in time domain
{
    for(m=0; m<=n; ++m) // looping in price domain
    {
        f(n,m) = DF * (p*f(n+1,m+1) +(1-p)*f(n+1,m));
    }
}
```

3.7e Pricing with discounted expectation formula

We can also do that with single formula (it does not work for path dependent option) :

$$f_t = e^{-rN\Delta t}\left[\sum_{m=0}^{N}\left(C_m^N p^m (1-p)^{N-m} \textit{payoff}(s_t u^m d^{N-m})\right)\right] \qquad \textit{where } T-t = N\Delta t$$

3.8 Solve risk neutral pricing numerically (by simulation)

3.8a Naive version

Monte Carlo is a numerical method for finding $E_Q(f(X))$ given samples $x_1, x_2, x_3, \ldots x_N$ of risk factor X drawn under measure Q. Most of the time, we prefer to use $X=\varepsilon(0,1)$ as the risk factor. Firstly we define f, which is ***NOT*** payoff ***NOR*** PV, it is discounted payoff.

This is discounted payoff.

(1) $\boxed{f(X)} = DF \times payoff(F\exp(-v/2+\sqrt{v}X))$ — *discounted exotic payoff depending on X*

(2) $f_{truePV} = E_Q[f(X)]$ — *population mean by risk neutral pricing*

(3) $f_{estPV} = \dfrac{\sum_{n=1}^{N} f(x_n)}{N}$ — *sample mean by Monte Carlo simulation*

These are true PV and est PV.

From central limit theorem :

$$\lim_{N\to\infty} \frac{\sum_{n=1}^{N} f(x_n)/N - E_Q[f(X)]}{\sqrt{V_Q[f(X)]/N}} \sim \varepsilon(0,1)$$

$$\frac{f_{estPV} - f_{truePV}}{\sqrt{V_Q[f(X)]/N}} \sim \varepsilon(0,1)$$

$$f_{estPV} \sim \varepsilon(f_{truePV}, \sqrt{V_Q[f(X)]/N})$$

hence it is unbiased estimation

3.8b Control variate

Given analytic solution of a vanilla product, which does not depend on particular sample x_n :

(1') $g(X) = DF \times (F\exp(-v/2+\sqrt{v}X)-K)^+$ — *discounted vanilla payoff depending on X*

(2') $g_{truePV} = DF \times (FN(d_1) - KN(d_2))$

(3') $f_{estPV} = \dfrac{\sum_{n=1}^{N}(f(x_n)-g(x_n))}{N} + g_{truePV}$ — *sample mean by Monte Carlo simulation*

From central limit theorem :

$$\lim_{N\to\infty} \frac{\sum_{n=1}^{N}(f(x_n)-g(x_n))/N - E_Q[f(X)-g(X)]}{\sqrt{V_Q[f(X)-g(X)]/N}} \sim \varepsilon(0,1)$$

$$\frac{\sum_{n=1}^{N}(f(x_n)-g(x_n))/N - f_{truePV} + g_{truePV}}{\sqrt{V_Q[f(X)-g(X)]/N}} \sim \varepsilon(0,1)$$

$$\sum_{n=1}^{N}(f(x_n)-g(x_n))/N + g_{truePV} \sim \varepsilon(f_{truePV}, \sqrt{V_Q[f(X)-g(X)]/N})$$

$$f_{estPV} \sim \varepsilon(f_{truePV}, \sqrt{V_Q[f(X)-g(X)]/N})$$

With control variate, estimation variance can be reduced if …

$$V_Q[f(X)-g(X)] < V_Q[f(X)]$$

$$V_Q[f(X)]+V_Q[g(X)]-2Cov_Q[f(X)g(X)] < V_Q[f(X)]$$

$$V_Q[g(X)] < 2Cov_Q[f(X)g(X)]$$

$$Cov_Q[f(X)g(X)] > \frac{1}{2}V_Q[g(X)]$$

i.e. f(X) and g(X) are positively correlated

3.8c Antithetic variate

Due to symmetric property of standard normal, both x and $-x$ are valid samples, thus they are used in the calculation.

$$f_{estPV} = \frac{\sum_{n=1}^{N}(f(x_n)+f(-x_n))/2}{N} \tag{3"}$$

From central limit theorem :

$$\lim_{N\to\infty} \frac{\sum_{n=1}^{N}(f(x_n)+f(-x_n))/(2N)-E_Q[(f(X)+f(-X))/2]}{\sqrt{V_Q[(f(X)+f(-X))/2]/N}} \sim \varepsilon(0,1) \qquad note\ \ f_{truePV}=E_Q[f(X)]=E_Q[f(-X)]$$

$$\frac{f_{estPV}-(f_{truePV}+f_{truePV})/2}{\sqrt{V_Q[(f(X)+f(-X))/2]/N}} \sim \varepsilon(0,1)$$

$$f_{estPV} \sim \varepsilon\left(f_{truePV},\sqrt{\frac{V_Q[(f(X)+f(-X))/2]}{N}}\right)$$

With control variate, estimation variance can be reduced if …

$$\begin{aligned}
V_Q[(f(X)+f(-X))/2] &< V_Q[f(X)] \\
\frac{V_Q[f(X)]+V_Q[f(-X)]+2Cov_Q[f(X)f(-X)]}{4} &< V_Q[f(X)] \\
\frac{V_Q[f(X)]+Cov_Q[f(X)f(-X)]}{2} &< V_Q[f(X)] \\
Cov_Q[f(X)f(-X)] &< V_Q[f(X)]
\end{aligned}$$

Remark : Recursive mean and variance

Recursive mean anad variance are useful for implementation of Monte Carlo simulation :

$$mean = \frac{\sum_n y_n}{N} \qquad where\ y_n = f(x_n)$$

$$\begin{aligned}
variance &= \frac{1}{N-1}\sum_n\left[y_n-\frac{\sum_m y_m}{N}\right]^2 \\
&= \frac{1}{N-1}\sum_n\left[y_n^2-\frac{2}{N}y_n(\sum_m y_m)+\frac{1}{N}\frac{1}{N}(\sum_m y_m)^2\right] \\
&= \frac{1}{N-1}\left[\sum_n y_n^2-\frac{2}{N}(\sum_n y_n)(\sum_m y_m)+\frac{1}{N}(\sum_m y_m)^2\right] \\
&= \frac{1}{N-1}\left[\sum_n y_n^2-\frac{1}{N}(\sum_n y_n)^2\right]
\end{aligned}$$

Part 4. Discussion

4.1 Call put parity

Call put parity can be derived by considering these two portfolios at maturity :

- one call option c_T and cash amount k
- one put option p_T and a share of underlying s_T.

	$c_T + k$	$=$	$p_T + s_T$	*same payoff for both up/down states*
(1)	$c_t + ke^{-r(T-t)}$	$=$	$p_t + s_t$	*same price according to law of one price*

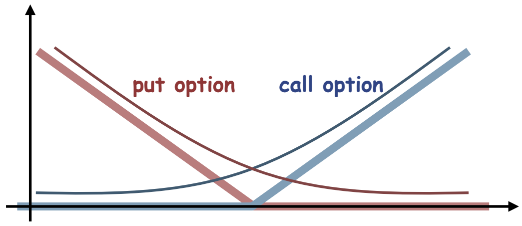
Lets take derivative with respect to s_t on both sides of call put parity, we have :

$$\frac{\partial c_t}{\partial s_t} + \frac{\partial ke^{-r(T-t)}}{\partial s_t} = \frac{\partial p_t}{\partial s_t} + \frac{\partial s_t}{\partial s_t}$$

(2) $$\frac{\partial c_t}{\partial s_t} = \frac{\partial p_t}{\partial s_t} + 1 \qquad \textit{note}:\ 0 \le \frac{\partial c_t}{\partial s_t} \le +1 \ \textit{and}\ -1 \le \frac{\partial p_t}{\partial s_t} \le 0$$

Lets take a look at the hockey sticks.

(3)



4.2 No early exercise for American call

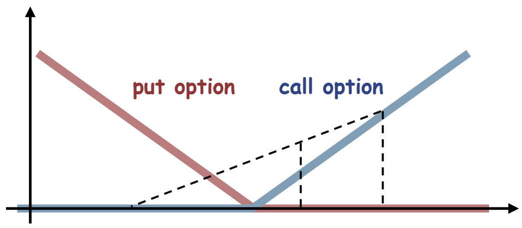
Why don't we exercise American call option early (but we do exercise American put option early)? Due to two reasons :

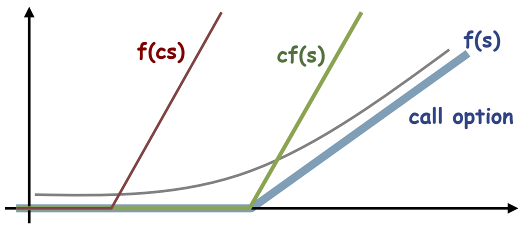
(1) convex payoff function f (**not** the option price)

$f(w_1 s_1 + w_2 s_2) < w_1 f(s_1) + w_2 f(s_2)$

(2) scaling in underlying vs scaling in derivative

$f(cs) \ge cf(s) \ge f(s)$ where $c \ge 1$





value of American call option if we don't exercise early

$=$	$e^{-r\Delta t}(pf(us_t) + (1-p)f(ds_t))$	
$\ge$	$e^{-r\Delta t} f(pus_t + (1-p)ds_t)$	*using property 1*
$=$	$e^{-r\Delta t} f(s_t e^{r\Delta t})$	*using risk neutral*
$\ge$	$e^{-r\Delta t} e^{r\Delta t} f(s_t)$	*using property 2, with* $c = e^{r\Delta t} \ge 1$
$=$	$f(s_t)$	
$=$	*value of American call option if we exercise now,* (2) *is not true for American put option*	

(3) Another intuitive explanation is that :

- if we early exercise American call, we give up time value $\sigma^2(T\text{-}t)$ while buying stock at unfavorable price K vs $Ke^{-r(T-t)}$
- if we early exercise American put, we give up time value $\sigma^2(T\text{-}t)$ for the sake of selling stock at favorable price K vs $Ke^{-r(T-t)}$

Part 5. Numerical methods

5.1 Portfolio theory

Given a portfolio with N risky assets, each has a random rate of return $r_n \ \forall n \in (1,N)$, find the optimal asset combination to minimise portfolio risk while fulfilling client's requested rate of return. Portfolio return is *a linear combo of correlated random variables* :

$$
\begin{aligned}
p &= \textstyle\sum_{n=1}^{N} w_n r_n \\
E[p] &= \textstyle\sum_{n=1}^{N} w_n E[r_n] &&= \mu^T w \\
V[p] &= \textstyle\sum_{n=1}^{N}\sum_{m=1}^{N} w_n w_m Cov(r_n, r_m) &&= w^T C w
\end{aligned}
$$

where

$$
w = \begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_N \end{bmatrix} \qquad l = \begin{bmatrix} 1 \\ 1 \\ \dots \\ 1 \end{bmatrix}
$$

$$
\mu = \begin{bmatrix} E[r_1] \\ E[r_2] \\ \dots \\ E[r_N] \end{bmatrix} \qquad C = \begin{bmatrix} Var[r_1] & Cov(r_1,r_2) & \dots & Cov(r_1,r_N) \\ Cov(r_2,r_1) & Var[r_2] & \dots & Cov(r_2,r_N) \\ \dots & \dots & \dots & \dots \\ Cov(r_N,r_1) & Cov(r_N,r_2) & \dots & Var[r_N] \end{bmatrix}
$$

Minimise potfolio risk $w^T C w$ such that $\mu^T w = z$ and $l^T w = 1$ (short selling is allowed in this case). We set up Lagrangian :

$$
\begin{aligned}
L &= w^T C w - \lambda_1(\mu^T w - z) - \lambda_2(l^T w - 1) \\
\frac{\partial L}{\partial w} &= Cw - \lambda_1 \mu - \lambda_2 u &&= 0 && (1a) \\
\frac{\partial L}{\partial \lambda_1} &= -(\mu^T w - z) &&= 0 && (1b) \\
\frac{\partial L}{\partial \lambda_2} &= -(u^T w - 1) &&= 0 && (1c)
\end{aligned}
$$

$$
\Rightarrow \quad \begin{bmatrix} 2C_{1,1} & 2C_{1,2} & 2C_{1,3} & \dots & 2C_{1,N} & -\mu_1 & -1 \\ 2C_{2,1} & 2C_{2,2} & 2C_{2,3} & \dots & 2C_{2,N} & -\mu_2 & -1 \\ 2C_{3,1} & 2C_{3,2} & 2C_{3,3} & \dots & 2C_{3,N} & -\mu_3 & -1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 2C_{N,1} & 2C_{N,2} & 2C_{N,3} & \dots & 2C_{N,N} & -\mu_N & -1 \\ \mu_1 & \mu_2 & \mu_3 & \dots & \mu_N & 0 & 0 \\ 1 & 1 & 1 & \dots & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \dots \\ w_N \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dots \\ 0 \\ z \\ 1 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} 2C & \mu & u \\ \mu^T & 0 & 0 \\ u^T & 0 & 0 \end{bmatrix} \begin{bmatrix} w \\ \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} 0 \\ z \\ 1 \end{bmatrix}
$$

This can be solved by standard algorithm for $AX = B$.

Extension by Robert Merton

skipped

Please read "An analytic derivation of the efficient portfolio frontier, Robert Merton".

In this paper, Merton has analytically :

- solved optimal portfolio combination w_{opt}
- found efficient frontier in the expected return *(as y axis)* vs volatility of return *(as x axis)* space.

The efficient frontier denotes pareto efficiency :

- maximum expected return given a volatility of return, or
- minimum volatility of return given an expected return (Lagrange duality).

Merton considered two cases :

- when all N assets are risky and short selling is allowed, then efficiency frontier is a horizontal parabola
- when one of the assets is risk free, then efficiency frontier is the tangent to the parabola and passes through risk free point (i.e. y = risk free rate and x = 0), this tangent is called ***capital market line*** *(CML)*.

N risky assets and short selling is allowed

From *equation 1*, we have 3 equations in 3 unknowns *(w, λ_1 and λ_2)* :

$$\begin{aligned} Cw &= \lambda_1\mu + \lambda_2 u && && (2a) \\ \mu^T w &= z && (\text{note}: \mu^T w = w^T\mu \text{ is a scalar}) && (2b) \\ u^T w &= 1 && (\text{note}: u^T w = w^T u \text{ is a scalar}) && (2c) \end{aligned}$$

Our trick is to express w in equation 2a in terms the two Lagrange multipliers, substitute the result into equation 2b and 2c, which then results in two equations in two unknowns, solve the system for the two Lagrange multipliers, which are put into equation 3a to obtain the optimal weights.

$$\begin{aligned} w &= C^{-1}(\lambda_1\mu + \lambda_2 u) && \text{put this into equation 2b and 2c} && (3a) \\ z &= \mu^T C^{-1}(\lambda_1\mu + \lambda_2 u) = \lambda_1(\mu^T C^{-1}\mu) + \lambda_2(\mu^T C^{-1}u) && = \lambda_1 a + \lambda_2 b && (3b) \\ 1 &= u^T C^{-1}(\lambda_1\mu + \lambda_2 u) = \lambda_1(u^T C^{-1}\mu) + \lambda_2(u^T C^{-1}u) && = \lambda_1 b + \lambda_2 c && (3c) \end{aligned}$$

where

$$\begin{aligned} a &= \mu^T C^{-1}\mu = a^T > 0 \quad (\text{scalar}) && \text{since } C \text{ and } C^{-1} \text{ are positive definite} \\ b &= \mu^T C^{-1}u = b^T \quad (\text{scalar}) && \\ c &= u^T C^{-1}u = c^T > 0 \quad (\text{scalar}) && \text{since } C \text{ and } C^{-1} \text{ are positive definite} \end{aligned}$$

Covariance matrix C is symmetric and positive definite, so is its inverse. Solving equation 3b and 3c, we have :

$$\begin{aligned} \lambda_1 &= (cz-b)/d \\ \lambda_2 &= (a-bz)/d \qquad \text{where } d = ac - b^2 \\ \Rightarrow \quad w^* &= C^{-1}(\lambda_1\mu + \lambda_2 u) = C^{-1}((cz-b)\mu + (a-bz)u)/d \end{aligned}$$

After getting the optimal weights, we can calculate the portfolio variance :

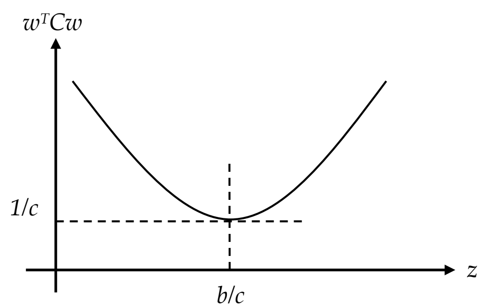
$$\begin{aligned} w^T Cw &= w^T C(C^{-1}(\lambda_1\mu + \lambda_2 u)) \\ &= \lambda_1 w^T\mu + \lambda_2 w^T u \\ &= \lambda_1 z + \lambda_2 && \text{applying 2b and 2c} \\ &= ((cz-b)z + (a-bz))/d \\ &= (cz^2 - 2bz + a)/d \end{aligned}$$

Recall that a, b and c are scalars determined by covariance matrix of asset returns. Please note that d is positive as :

$$\begin{aligned} (b\mu - au)^T C^{-1}(b\mu - au) &> 0 && \text{since } C^{-1} \text{ is positive definite} \\ b^2(\mu^T C^{-1}\mu) - 2ab(\mu^T C^{-1}u) + a^2(u^T C^{-1}u) &> 0 \\ b^2(a) - 2ab(b) + a^2(c) &> 0 \\ a(ac-b^2) &> 0 \\ ad &> 0 && \Rightarrow \quad a, d > 0 \end{aligned}$$

Therefore the portfolio variance is quadratic in terms of client's requested portfolio return z, to be precise, in z vs portfolio variance space, the efficient frontier is a parabola with horizontal axis of symmetry. Besides, it is strictly convex as the quadratic term c/d is positive. If requested expected return can be modified, the minimum possible portfolio variance is :

$$\begin{aligned} 0 &= \partial_z(w^T Cw) \\ 0 &= \partial_z(cz^2 - 2bz + a)/d \\ 0 &= 2cz - 2b \\ z &= b/c \\ \min(w^T Cw) &= (c(b/c)^2 - 2bb/c + a)/d \\ &= (a - b^2/c)/d \\ &= (ac - b^2)/(dc) \\ &= 1/c \end{aligned}$$



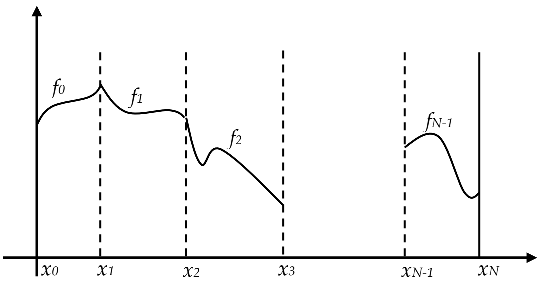
In practice, we plot the frontier in z vs portfolio deviation space. In this space, the frontier becomes hyperbola.

skipped

5.2 Spline interpolation

Given *N+1* points $(x_n, y_n)\ \forall n \in [0,N]$, we perform interpolation using spline, which is a *N* segment-piecewise cubic curve. The spline should pass through the *N+1* points, we also introduce the *1*st order and the *2*nd order continuity constraints in between n^{th} segment and $n+1^{th}$ segment. Each cubic curve has *4* parameters, there are *4N* parameters in the spline, thus we need *4N* equations to solve for all parameters. The spline is defined as :

$$\begin{aligned} y &= f_n(x) && \exists n \in [0, N-1] \text{ such that } x_n \le x \le x_{n+1} \\ &= a_n x^3 + b_n x^2 + c_n x + d_n \end{aligned}$$



The parameters are solved by *2N* zero order constraints :

$$\begin{aligned} y_n &= a_n x_n^3 + b_n x_n^2 + c_n x_n + d_n && \forall n \in [0, N-1] \\ y_{n+1} &= a_n x_{n+1}^3 + b_n x_{n+1}^2 + c_n x_{n+1} + d_n && \forall n \in [0, N-1] \end{aligned}$$

and *N+1 1*st order constraints and *N+1 2*nd order constraints :

$$\begin{aligned} 3a_n x_n^2 + 2b_n x_n + c_n &= 3a_{n-1} x_n^2 + 2b_{n-1} x_n + c_{n-1} && \forall n \in [1, N-1] && \text{continuity in 1}^{st}\text{ order derivative} \\ 6a_n x_n + 2b_n &= 6a_{n-1} x_n + 2b_{n-1} && \forall n \in [0, N-1] && \text{continuity in 2}^{nd}\text{ order derivative} \end{aligned}$$

and finally 2 more 2nd order constraints at both ends of the spline :

$$\begin{aligned} 6a_0 x_0 + 2b_0 &= 0 \\ 6a_{N-1} x_N + 2b_{N-1} &= 0 \end{aligned}$$

Number of equations is *2N+2(N-1)+2 = 4N*. We have in matrix form :

$$\begin{bmatrix}
x_0^3 & x_0^2 & x_0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & \dots & \dots & \dots & \dots \\
x_1^3 & x_1^2 & x_1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & \dots & \dots & \dots & \dots \\
0 & 0 & 0 & 0 & x_1^3 & x_1^2 & x_1 & 1 & 0 & 0 & 0 & \dots & \dots & \dots & \dots & \dots \\
0 & 0 & 0 & 0 & x_2^3 & x_2^2 & x_2 & 1 & 0 & 0 & 0 & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & x_{N-1}^3 & x_{N-1}^2 & x_{N-1} & 1 \\
\dots & \dots & \dots & \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & x_N^3 & x_N^2 & x_N & 1 \\
3x_1^2 & 2x_1 & 1 & 0 & -3x_1^2 & -2x_1 & -1 & 0 & 0 & 0 & 0 & \dots & \dots & \dots & \dots & \dots \\
6x_1 & 2 & 0 & 0 & -6x_1 & -2 & 0 & 0 & 0 & 0 & 0 & \dots & \dots & \dots & \dots & \dots \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
6x_0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 6x_N & 2
\end{bmatrix}
\times
\begin{bmatrix} a_0 \\ b_0 \\ c_0 \\ d_0 \\ a_1 \\ b_1 \\ c_1 \\ d_1 \\ \dots \\ a_{N-1} \\ b_{N-1} \\ c_{N-1} \\ d_{N-1} \end{bmatrix}
=
\begin{bmatrix} y_0 \\ y_1 \\ y_1 \\ y_2 \\ \dots \\ y_{N-1} \\ y_N \\ 0 \\ 0 \\ \dots \\ 0 \\ 0 \end{bmatrix}$$

Suppose we have `spline_interpolation(std::vector<double>& xs, std::vector<double>& ys, double x)`, which solves all spline coefficients given vector of `xs` and `ys` and return the `y` value corresponding to a `x` value. We will use it in yield curve bootstrapping next section.

5.3 Newton Raphson method

Firstly, lets review Taylor series for two cases :

1 equation N unknowns

$$\begin{aligned} f &= [f_1] \\ \Delta x &= [\Delta x_1, \Delta x_2, \Delta x_3, \dots, \Delta x_N]^T \\ \rightarrow \quad f(x+\Delta x) &= f(x) + J\Delta x + 1/2\,(\Delta x)^T H \Delta x + \dots \\ \textit{where} \quad J &= [\partial_{x_1} f, \partial_{x_2} f, \dots, \partial_{x_N} f] \\ H &= \begin{bmatrix} \partial_{x_1}\partial_{x_1} f & \partial_{x_1}\partial_{x_2} f & \dots & \partial_{x_1}\partial_{x_N} f \\ \partial_{x_2}\partial_{x_1} f & \partial_{x_2}\partial_{x_2} f & \dots & \partial_{x_2}\partial_{x_N} f \\ \dots & \dots & \dots & \dots \\ \partial_{x_N}\partial_{x_1} f & \partial_{x_N}\partial_{x_2} f & \dots & \partial_{x_N}\partial_{x_N} f \end{bmatrix} \end{aligned}$$

N equations N unknowns

$$\begin{aligned} f &= [f_1, f_2, f_3, \dots, f_N]^T \\ \Delta x &= [\Delta x_1, \Delta x_2, \Delta x_3, \dots, \Delta x_N]^T \\ f(x+\Delta x) &= f(x) + J\Delta x + \dots \\ J &= \begin{bmatrix} \partial_{x_1} f_1 & \partial_{x_2} f_1 & \dots & \partial_{x_N} f_1 \\ \partial_{x_1} f_2 & \partial_{x_2} f_2 & \dots & \partial_{x_N} f_2 \\ \dots & \dots & \dots & \dots \\ \partial_{x_1} f_N & \partial_{x_N} f_N & \dots & \partial_{x_N} f_N \end{bmatrix} \end{aligned}$$

5.3a Newton Raphson

Newton Raphson is an iterative method for solving N unknowns in N nonlinear equations using *1st* term in Taylor series.

$$
\begin{aligned}
f(x)+J\Delta x &= 0 \\
\Delta x &= -J^{-1}f(x) \\
\textit{where} \quad J_{n,m} &= \frac{f_n(x_1,x_2,...x_m+\Delta,...)-f_n(x_1,x_2,...x_m,...)}{\Delta}
\end{aligned}
$$

5.3b Application to yield curve bootstrapping

Given prices of N bonds with different maturities / payment schedules / coupon rates, solve for the yield curve. Suppose :

$$
\begin{aligned}
b_n &= c_n e^{-r(t_{n,1})t_{n,1}} + c_n e^{-r(t_{n,2})t_{n,2}} + ... + c_n e^{-r(t_{n,K_n})t_{n,K_n}} && \textit{bond price of nth bond} \\
\textit{schedule} &= [t_{n,1}, t_{n,2}, ..., t_{n,K_n}] && \textit{where final payment equals to maturity } t_{n,K_n} = T_n \\
\textit{zero rate curve} &= r(t)
\end{aligned}
$$

Given N bonds, we model yield curve by cubic spline with $N+1$ knots at bond maturities $0, T_1, T_2, T_3, \dots T_N$, having zero rate $0, r_1, r_2, r_3, \dots r_N$ respectively, i.e. there are N segments in the spline. Coupon payment time may not coincide with the spline knots, i.e. $t_{n,m} \notin \{0, T_1, T_2, T_3, \dots T_N\}$, thus we need to interpolate $r(t_{n,m})$ from the knots. Let's define x = $[r_1, r_2, r_3, \dots r_N]$ be a vector of N unknown zero rates, we can now set up a system of N nonlinear equations to solve for the N unknowns : $f(x) = 0$.

$$
x = [r_1, r_2, ..., r_N]^T
$$

$$
\begin{bmatrix} f_1(x) \\ f_2(x) \\ ... \\ f_N(x) \end{bmatrix} = \begin{bmatrix} c_1 e^{-r(t_{1.1}|x)t_{1.1}} + c_1 e^{-r(t_{1.2}|x)t_{1.2}} + ... + c_1 e^{-r(t_{1.K_1}|x)t_{1.K_1}} - b_1 \\ c_2 e^{-r(t_{2.1}|x)t_{2.1}} + c_2 e^{-r(t_{2.2}|x)t_{2.2}} + ... + c_2 e^{-r(t_{2.K_2}|x)t_{2.K_2}} - b_2 \\ ... \\ c_N e^{-r(t_{N.1}|x)t_{N.1}} + c_N e^{-r(t_{N.2}|x)t_{N.2}} + ... + c_N e^{-r(t_{N.K_N}|x)t_{N.K_N}} - b_N \end{bmatrix}
$$

5.3c Implementation of yield curve bootstrapping

We define main classes `yield_curve`, `bond_helpers` and one main function `newton_raphson` :

```
struct yield_curve
{
    double get_df(double t) const { return exp(- spline_interpolation(ts, zrs, t) * t); } // DF = exp(-zr(t)*t)
    std::vector<double> ts;  // x-axis : date time on knots
    std::vector<double> zrs; // y-axis : zero rate on knots
};

class bond_helpers // housekeeper of N bonds
{
    bond_helpers(const std::vector<bond>& bonds) : bonds(bonds) // suppose bonds are sorted in maturity
    {
        for(const auto& bond:bonds) curve.ts.push_back(bond.maturity);
    }

    std::vector<double> operator()(const std::vector<double>& zero_rates) const // given N zero rates, return N errors
    {
        curve.zrs = zero_rates;
        std::vector<double> errors;
        for(const auto& bond:bonds)
        {
            double npv = 0;
            for(const auto& x:bond.schedule) npv += x.coupon * curve.get_df(x.time);
            errors.push_back(pow(bond.market_price-npv, 2));
        }
        return errors;
    }

    const std::vector<bond>& bonds;
    yield_curve curve;
};

template<typename FCT = bond_helpers>
std::vector<double> newton_raphson(const FCT& f, const std::vector<double>& init_x) // find x such that f(x) = [0,0,...,0]
{
    std::vector<double> x = init_x;
    while(!converged)
    {
        matrix J = estimate_Jacobian(f, x, delta_x); // double-for-loop to build J matrix inside
        x -= inverse(J) * f(x);
    }
    return x;
}
```

5.4 Time varying variance by GARCH(1,1)

5.4a Define model

Recall the underlying price model :

$$(ds_t)/s_t = \varepsilon(\mu dt, \sigma\sqrt{dt}) \quad \textit{for constant variance}$$
$$(ds_t)/s_t = \varepsilon(\mu dt, \sigma_t\sqrt{dt}) \quad \textit{for time varying variance}$$

Now we model the underlying price with time dependent variance using GARCH(1,1), with ***6 parameters*** $[\alpha,\beta,\gamma,\mu,\sigma_1,\sigma_{LR}]$.

$$(s_{t+1}-s_t)/s_t = \varepsilon(\mu,\sigma_t) \quad \textit{with } \sigma_{t+1}^2 = \alpha(r_t-\mu)^2+\beta\sigma_t^2+\gamma\sigma_{LR}^2 \textit{ where } \alpha+\beta+\gamma=1 \textit{ and } \alpha,\beta,\gamma>0$$

5.4b ML estimation

Given time series $s_0, s_1, s_2, \ldots, s_T$, we obtain return time series $r_1, r_2, r_3, \ldots, r_T$, where $r_t = (s_{t+1}-s_t)/s_t$. Firstly we can estimate two para :

$$\mu = \textstyle\sum_{t=1}^T r_t / T \quad \textit{which is unbiased estimation, check by taking expection on both sides}$$
$$\sigma_1 = \sqrt{\textstyle\sum_{t=1}^1 (r_t-\mu)^2/1} = r_1-\mu$$

Secondly, we apply constraint on the third para $\gamma = 1-\alpha-\beta$. Thirdly, we write down the likelihood :

$$L(\alpha,\beta,\sigma_{LR}) = \frac{e^{-(r_1-\mu)^2/(2\sigma_1^2)}}{\sqrt{2\pi\sigma_1^2}}\frac{e^{-(r_2-\mu)^2/(2\sigma_2^2)}}{\sqrt{2\pi\sigma_2^2}}\cdots\frac{e^{-(r_T-\mu)^2/(2\sigma_T^2)}}{\sqrt{2\pi\sigma_T^2}}$$

where :

$$\sigma_2^2 = \alpha(r_2-\mu)^2+\beta\sigma_1^2+(1-\alpha-\beta)\sigma_{LR}^2$$
$$\sigma_3^2 = \alpha(r_3-\mu)^2+\beta\sigma_2^2+(1-\alpha-\beta)\sigma_{LR}^2$$
$$\ldots = \ldots$$

$$\begin{aligned}\arg\max \ln L(\alpha,\beta,\sigma_{LR}) &= \arg\max[-\textstyle\sum_{t=1}^T (r_t-\mu)^2/(2\sigma_t^2) - \sum_{t=1}^T \frac{1}{2}\ln(2\pi\sigma_t^2)] \quad \textit{take log on both sides}\\ &= \arg\max[-\textstyle\sum_{t=1}^T ((r_t-\mu)/\sigma_t)^2 - \sum_{t=1}^T (\ln(2\pi)+\ln\sigma_t^2)]\\ &= \arg\min[\textstyle\sum_{t=1}^T ((r_t-\mu)/\sigma_t)^2 + \sum_{t=1}^T \ln\sigma_t^2]\end{aligned}$$

This nonlinear optimization can be solved by gradient descent or EXCEL's solver.

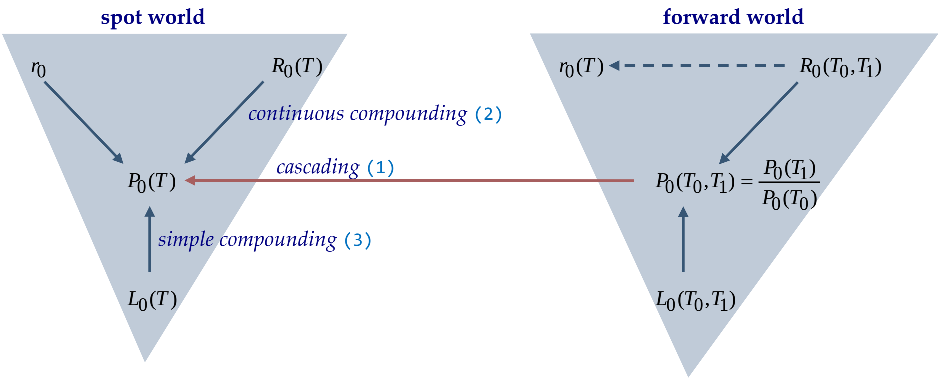
parameter	*solution*
α	*gradient descent*
β	*gradient descent*
γ	*constraint*
μ	*averaging*
σ_1	*averaging*
σ_{LR}	*gradient descent*

5.5 How to bootstrap?

5.5a Interest rate DAG

Given market data as of today *0* and swap trading day *t = 0+2days*, we have *DAG* :

- cascading is done in *DF*, see formula (1) below
- interpolation is done in zero rate, using continuous compounding, see formula (2) below
- mktdata quote is done in *LIBOR*, using simple compounding, see formula (3) below



Bootstrapping means construction of yield curve from the market data of liquid instruments including :

- deposit rate
- forward rate agreement FRA (mathematically, it is equivalent to deposit rate)
- interest rate futures IRF
- interest rate swap IRS

Daycounter

- Daycounter is a function that converts *[start date, end date]* to *yearfraction*.
- Daycounter of curve δ_{curve} is different from daycounter of market quoted instruments.

5.5b The 8 fore formula

```
(1)  DF formula
(2)  ZR formula
(3)  LIBOR formula for deposit
(4)  LIBOR formula for swap fixed leg
(5)  LIBOR formula for swap fixed leg, solution without interpolation
(6)  LIBOR formula for swap fixed leg, solution with interpolation
(7)  LIBOR formula for swap fixed/floating leg, for OIS discounting
(8)  LIBOR formula for swap LIBOR prediction
```

Each instrument measures interest accrual of a period. Cascading is needed for interest accrual of a long period. Cascading is done in discount factor domain. One line proof :

$$
\text{(1)} \qquad
\begin{aligned}
DF_0(T_0,T_N) &= P_0(T_0,T_N) \\
&= \frac{P_0(T_N)}{P_0(T_0)} \\
&= \frac{P_0(T_1)}{P_0(T_0)}\times\frac{P_0(T_2)}{P_0(T_1)}\times\frac{P_0(T_3)}{P_0(T_2)}\times\ldots\times\frac{P_0(T_N)}{P_0(T_{N-1})} \\
&= DF_0(T_0,T_1)\times DF_0(T_1,T_2)\times\ldots\times DF_0(T_{N-1},T_N)
\end{aligned}
$$

Conversion between discount factor and zero rate is done by :

$$
\text{(2)} \qquad P_0(T) = e^{-R_0(T)\delta_{curve}(0,T)}
$$

Conversion between discount factor and *LIBOR* depends on specific instrument, please refer to next page.

Both deposit rate and forward rate agreement adopt simple compounding. Each quote gives discount factor of one period.

$$(3)\quad DF_0(0,1) = \frac{1}{1+r_{deposit}(ON)\delta_{deposit}(0,1)} \qquad \textit{overnight refers to period from day0 to day1}$$

$$DF_0(1,2) = \frac{1}{1+r_{deposit}(TN)\delta_{deposit}(1,2)} \qquad \textit{tom-next refers to period from day1 to day2}$$

$$DF_0(2,2+1M) = \frac{1}{1+r_{deposit}(1M)\delta_{deposit}(2,2+1M)} \qquad \textit{for deposit rate with tenors 1W 2W 1M 1Y etc}$$

$$DF_0(2,2+1Y) = \frac{1}{1+r_{fra}(1Y)\delta_{fra}(2,2+1Y)}$$

Interest rate swap *(payer)* can be considered as buying a stream of forward *LIBOR* with swap rate. As the floating leg must be at par on *trading day t = 0+2days (since it is LIBOR coupon discounted by LIBOR rate)*, we have to consider the fixed leg only. *S* is swap rate.

$$(4)\quad 1 = S_0(\Gamma_0,\Gamma_M)[\textstyle\sum_{m=1}^{M}\delta_{irs}(\Gamma_{m-1},\Gamma_m)P_0(t,\Gamma_m)]+P_0(t,\Gamma_M) \qquad \textit{PV of fixed leg is at par at } t = 0+2days$$

$$\text{or}\quad P_0(2) = S_0(\Gamma_0,\Gamma_M)[\textstyle\sum_{m=1}^{M}\delta_{irs}(\Gamma_{m-1},\Gamma_m)P_0(\Gamma_m)]+P_0(\Gamma_M) \qquad \textit{Normally, we have } t=T_0=\Gamma_0.$$

If all discount factors except the last point $P_0(\Gamma_M)$ in the bootstrapping curve are known, then we have to solve :

$$P_0(2) = S_0(\Gamma_0,\Gamma_M)[\textstyle\sum_{m=1}^{M-1}\delta_{irs}(\Gamma_{m-1},\Gamma_m)P_0(\Gamma_m)]+[1+S_0(0,\Gamma_M)\delta_{irs}(\Gamma_{M-1},\Gamma_M)]P_0(\Gamma_M)$$

$$(5)\quad P_0(\Gamma_M) = \frac{P_0(2)-S_0(\Gamma_0,\Gamma_M)[\sum_{m=1}^{M-1}\delta_{irs}(\Gamma_{m-1},\Gamma_m)P_0(\Gamma_m)]}{1+S_0(0,\Gamma_M)\delta_{irs}(\Gamma_{M-1},\Gamma_M)}$$

If discount factors beyond $P_0(\Gamma_K)$ are unknown, interpolation kicks in. We try to interpolate in zero rate domain for simplicity.

$$P_0(2) = \left[\begin{array}{l} S_0(0,\Gamma_M)[\sum_{k=1}^{K}\delta_{irs}(\Gamma_{k-1},\Gamma_k)P_0(\Gamma_k)]+ \\ S_0(0,\Gamma_M)[\sum_{m=K+1}^{M-1}\delta_{irs}(\Gamma_{m-1},\Gamma_m)P_0(\Gamma_m)]+ \\ [1+S_0(0,\Gamma_M)\delta_{irs}(\Gamma_{M-1},\Gamma_M)]P_0(\Gamma_M) \end{array}\right. \qquad \textit{where } w_m = \frac{\delta_{irs}(0,\Gamma_m)-\delta_{irs}(0,\Gamma_K)}{\delta_{irs}(0,\Gamma_M)-\delta_{irs}(0,\Gamma_K)} \textit{ and } w'_m = 1-w_m$$

$$= \left[\begin{array}{l} S_0(0,\Gamma_M)[\sum_{k=1}^{K}\delta_{irs}(\Gamma_{k-1},\Gamma_k)P_0(\Gamma_k)]+ \\ S_0(0,\Gamma_M)[\sum_{m=K+1}^{M-1}\delta_{irs}(\Gamma_{m-1},\Gamma_m)(w'_m P_0(\Gamma_K)+w_m P_0(\Gamma_M))]+ \\ [1+S_0(0,\Gamma_M)\delta_{irs}(\Gamma_{M-1},\Gamma_M)]P_0(\Gamma_M) \end{array}\right.$$

$$(6)\quad P_0(\Gamma_M) = \frac{P_0(2)-S_0(0,\Gamma_M)[\sum_{k=1}^{K}\delta_{irs}(\Gamma_{k-1},\Gamma_k)P_0(\Gamma_k)]-S_0(0,\Gamma_M)[\sum_{m=K+1}^{M-1}\delta_{irs}(\Gamma_{m-1},\Gamma_m)w'_m P_0(\Gamma_K)]}{1+S_0(0,\Gamma_M)\delta_{irs}(\Gamma_{M-1},\Gamma_M)+S_0(0,\Gamma_M)\sum_{m=K+1}^{M-1}\delta_{irs}(\Gamma_{m-1},\Gamma_m)w_m}$$

When we perform OIS discounting, floating leg is not necessary at par, we need to consider both legs (no notional exchange) :

$$(7)\quad S_0(\Gamma_0,\Gamma_M)[\textstyle\sum_{m=1}^{M}\delta_{irs}(\Gamma_{m-1},\Gamma_m)P_{OIS,0}(\Gamma_m)] = \sum_{n=1}^{N}L_0(T_{n-1},T_n)\delta_{irs}(T_{n-1},T_n)P_{OIS,0}(T_n)$$

where *LIBOR* is predicted by interpolation in *LIBOR* discount curve :

$$P_{LIBOR,0}(T_{n-1},T_n) = \frac{P_{LIBOR,0}(T_n)}{P_{LIBOR,0}(T_{n-1})}$$

$$(8)\quad L_0(T_{n-1},T_n) = \left(\frac{1}{P_{LIBOR,0}(T_{n-1},T_n)}-1\right)\frac{1}{\delta(T_{n-1},T_n)} \qquad \textit{inverse of simple compounding}$$

$$= \left(\frac{P_{LIBOR,0}(T_{n-1})}{P_{LIBOR,0}(T_n)}-1\right)\frac{1}{\delta(T_{n-1},T_n)}$$

5.5c Excel implementation for LIBOR discounting

Define global variables :

- `today`
- `spotday = today+2`

Define these columns for ***LIBOR-table*** :

A	instrument type	`A[n] = input`	
B	instrument tenor	`B[n] = input`	
C	market quote	`C[n] = input`	
D	instrument start date	`D[n] = 0/1/2`	
E	instrument end date	`E[n] = D[n]+C[n]`	
F	daycount from start date to end date	`F[n] = YEARFRAC(D[n],E[n],1)`	`1=act/act 2=act360 3=act365 4=30/360`
G	daycount from `today` to end date	`G[n] = YEARFRAC(today,E[n],1)`	`1=act/act 2=act360 3=act365 4=30/360`
H	discount factor from end date to start date	`H[n] = 1/(1+C[n]*F[n])`	using equation `(3)`
I	discount factor from end date to `today`	`I[n] = I[2]*H[n]`	using equation `(1)`
J	zero rate	`J[n] = -ln(I[n])/G[n]`	using equation `(2)`

We construct one ***IRS-table*** for solving one `J[n]` entry (corresponding to one *IRS* quote) in ***LIBOR-table***. Define global variable :

- `swaprate` for market quote
- `DF_02` as discount factor from `spotday` to `today`
- one fixed leg payment occupies one row
- floating leg payments are omitted as they are not involved in `(4,5,6)`

Define these columns for ***IRS-table*** :

A	fixed payment start date	`A[n] = 2 + n * period`	
B	fixed payment end date	`B[n] = A[n+1]`	
C	daycount from start date to end date	`C[n] = YEARFRAC(A[n],B[n],1)`	`1=act/act 2=act360 3=act365 4=30/360`
D	daycount from `today` to end date	`D[n] = YEARFRAC(today,B[n],1)`	`1=act/act 2=act360 3=act365 4=30/360`
E	cashflow on end date	`E[n] = notional*swaprate*C[n]`	
F	zero rate from `today` to end date	`F[n] = EInterpolation('LIBOR'!$E$0:$E$25, 'LIBOR'!$J$0:$J$25, B[n])`	
G	discount factor from end date to `today`	`G[n] = exp(-D[n]*F[n])`	*Interpolation of zero rate using main-table, suppose there are 26 market data in main-table.*
H	present value on `spotday` (not on `today`)	`H[n] = E[n]*G[n]/DF_02`	

```
H[N] = notional * G[N]/DF_02
I[1] = sum(H0:H[N])
```

Finally invoke *Excel* solver to find `'LIBOR'!J[n]` so that `'IRS'!I[1]` equals to `notional`, making use of `(4,5,6)`. *IRS* is at par means :

- *floating leg PV = fixed leg PV* on `spotday` and
- *floating leg PV = fixed leg PV* on `today` (i.e. everyday)
- *fixed leg PV = $1* on `spotday` but not on `today`

5.5d Excel implementation for OIS discounting

For *OIS* discounting, we need to build two curves in sequence :

- ***OIS discounting curve*** governs $P_{OIS,0}(T_0,T_1)$, bootstrapped by identical algorithm as shown in previous part
- ***LIBOR prediction curve*** governs $P_{LIBOR,0}(T_0,T_1)$, bootstrapped by modified algorithm as shown below

Modified algorithm

In the modified bootstrapping algorithm :

- there is no change in deposit rate and *FRA* part, since :
- *deposit* and *FRA* do not involve prediction of cashflow
- *deposit* and *FRA* are quoted directly in spot *LIBOR* or forward *LIBOR*
- there are some change for *IRS*, using equation `(7,8)` instead of `(4,5,6)`

For each *IRS* market quote, we bootstrap one $R_{LIBOR0}(0,T)$ entry with two *IRS-tables* :

- ***fixed-table*** involving *OIS* curve only
- ***float-table*** involving both curves

Define these columns for ***fixed-table*** (nearly identical to *IRS-table*) :

A	fixed payment start date Γ_{m-1}	`A[n] = 2 + m*fixed_period`
B	fixed payment end date Γ_m	`B[n] = A[n+1]`
C	daycount from start date to end date	`C[n] = YEARFRAC(A[n],B[n],1)`
D	daycount from `today` to end date	`D[n] = YEARFRAC(today,B[n],1)`
E	cashflow on end date	`E[n] = notional*swaprate*C[n]`
F	*OIS* zero rate from `today` to end date	`F[n] = EInterpolation('OIS'!$E$0:$E$25, 'OIS'!$J$0:$J$25, B[n])`
G	*OIS* discount factor from end date to `today`	`G[n] = exp(-D[n]*F[n])`
H	present value on `spotday` (not on `today`)	`H[n] = E[n]*G[n]/DF_02`

These are zero rates from OIS curve.

Define these columns for ***float-table*** (extra columns are added for calculation of $L_{LIBOR0}(T_{n-1},T_n)$) :

A	fixed payment start date T_{n-1}	`A[n] = 2 + n*float_period`
B	fixed payment end date T_n	`B[n] = A[n+1]`
C	daycount from start date to end date	`C[n] = YEARFRAC(A[n],B[n],1)`
Z	daycount from `today` to end date	`Z[n] = YEARFRAC(today,A[n],1)`
D	daycount from `today` to end date	`D[n] = YEARFRAC(today,B[n],1)`
P	*LIBOR* zero rate from `today` to start date	`P[n] = EInterpolation('LIBOR'!$E$0:$E$25, 'LIBOR'!$J$0:$J$25, A[n])`
Q	*LIBOR* zero rate from `today` to end date	`Q[n] = EInterpolation('LIBOR'!$E$0:$E$25, 'LIBOR'!$J$0:$J$25, B[n])`
R	*LIBOR* discount factor from `today` to start date	`R[n] = exp(-P[n]*Z[n])`
S	*LIBOR* discount factor from `today` to end date	`S[n] = exp(-Q[n]*D[n])`
T	*LIBOR* forward rate from start date to end date	`T[n] = (R[n]/S[n]-1)/C[n]`
E	cashflow on end date	`E[n] = notional*T[n]*C[n]`
F	OIS zero rate from `today` to end date	`F[n] = EInterpolation('OIS'!$E$0:$E$25, 'OIS'!$J$0:$J$25, B[n])`
G	discount factor from end date to `today`	`G[n] = exp(-D[n]*F[n])`
H	present value on `spotday` (not on `today`)	`H[n] = E[n]*G[n]/DF_02`

These are zero rates from LIBOR curve.

cancel each other

Finally, we invoke *Excel* solver to find `'LIBOR'!J[n]` so that `sum('fixed'!H0:H[N])` equals to `sum('float'!H0:H[N])`.