Spread Options 02-14-11

N. T. Gladd

Initialization: Be sure the files *NTGStylesheet2.nb* and *NTGUtilityFunctions.m* is are in the same directory as that from which this notebook was loaded. Then execute the cell immediately below by mousing left on the cell bar to the right of that cell and then typing “shift” + “enter”. Respond “Yes” in response to the query to evaluate initialization cells.

In[18]:=
```
SetDirectory[NotebookDirectory[]];
(* set directory where source files are located *)
SetOptions[EvaluationNotebook[], (* load the StyleSheet *)
 StyleDefinitions → Get["NTGStylesheet2.nb"]];
Get["NTGUtilityFunctions.m"]; (* Load utilities package *)
```

1 Introduction

Spread options are contingent claims on the difference in the prices of two or more underliers. Such contingencies arise in quite natural ways and have relevance in many financial contexts. For example, in 2010 I worked in the utility industry where a key concern is the spread between the price of power and the cost of the fuel used to generate that power— the so-called spark spread http://en.wikipedia.org/wiki/Spark_spread. Other examples related to the commodity world are the crack spread (oil refining) and the crush spread (soybean processing). In the financial world there is the TED spread (interest rates/ credit risk). Many equities derivatives involve contingencies on spreads. In short, relative differences are important in the financial world

Here, I will summarize some derivations and calculations I’ve performed that are relevant to spread options. The examples are simple in this first treatment. Notebooks on more details aspects of spread options will be posted in the future. In particular, the following valuation models will be discussed and derived

- Numerical double quadrature
- Exact special case - Margrabe exchange option
- Bachelier’s approximation
- Kirk’s approximation

To simplify the discussion, I will consider spreads between two stocks whose price processes follow geometric Brownian motion. That is the classical Black-Scholes framework and a good place to start. The payoff on a call spread option is

$$C(T) = \max(S_1(T) - S_2(T) - K, 0) \tag{1}$$

where S_1 and S_2 denote the two stock prices, K is the strike price for the option, and T is the expiry. I consider a European style option so exercise is possible only at T. The dynamics are

$$\frac{dS_i}{S_i} = r\, dt - \sigma_i\, dz_i,$$

$$S_i(T) = S_i(t) \exp\left[\left(r - \frac{\sigma_i^2}{2}\right)(T - t) + \epsilon_i\, \sigma_i \sqrt{T - t}\,\right] \tag{2}$$

$$\mathbb{E}[\epsilon_1\, \epsilon_2] = dt$$

where the ϵ_i are correlated normal random variables. In this model, the stocks are assumed to have constant volatilities and correlation. The inclusion of term structure for volatility and correlation is an important for practical applications but would complicate this initial exposition.

The option theoretic fair value is

$$C(t) = e^{-r\,(T-t)}\, \mathbb{E}_Q[\max(S_1(T) - S_2(T) - K, 0)]$$

$$= e^{-r\,(T-t)} \int_0^\infty dS_1 \int_0^\infty dS_2\, f_Q(S_1(T),\; S_2(T)) \max(S_1(T) - S_2(T) - K, 0) \tag{3}$$

where $f_Q(S_1, S_2)$ is the joint probability distribution for S_1 and S_2 under the risk neutral probability measure.

Calculations are simplified if the random variables ϵ_i are chosen as the independent variables. Also, I choose t = 0 to simplify matters.

$$\frac{C(0)}{e^{-r\,T}} = \int_{-\infty}^\infty d\epsilon_1 \int_{-\infty}^\infty d\epsilon_2\, f(\epsilon_1,\, \epsilon_2) \max(S_1(\epsilon_1) - S_2(\epsilon_2) - K, 0) \tag{4}$$

The random variable ϵ_1 and ϵ_2 are correlated, $\mathbb{E}[\eta_1\, \eta_2] = \rho$, but a Cholesky decomposition (see below) can be used to express the option price in terms of uncorrelated random variables η_1 and η_2

$$\frac{C(0)}{e^{-r\,T}} = \int_{-\infty}^\infty d\eta_1 \int_{-\infty}^\infty d\eta_2\, f(\eta_1)\, f(\eta_2) \max(S_1(\eta_1) - S_2(\eta_2) - K, 0) \tag{5}$$

where

$$S_1(T) = S_1(0) \exp\left[\left(r - \frac{\sigma_1^2}{2}\right) T + \eta_1\, \sigma_1 \sqrt{T}\,\right]$$

$$S_2(T) = S_2(0) \exp\left[\left(r - \frac{\sigma_2^2}{2}\right) T + \left(\eta_1\, \rho + \eta_2 \sqrt{1 - \rho^2}\right) \sigma_2 \sqrt{T}\,\right] \tag{6}$$

$$\mathbb{E}[\eta_1\, \eta_2] = 0$$

2 Valuation via double numerical quadrature

A straightforward way to value a spread option is to simply numerically evaluate the double integral in equation (5). The integrand has the form

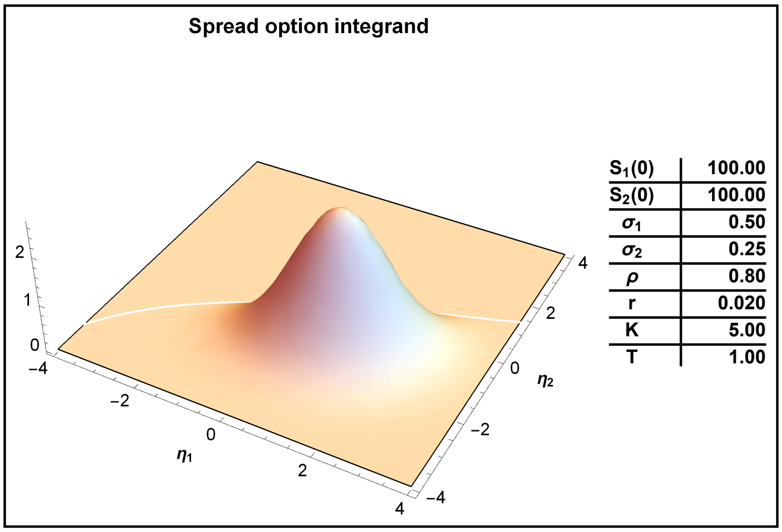


Figure 1: Integrand of spread option

and the double integral (5) can easily be performed using *Mathematica*'s NIntegrate. Here I show the fair value of a call spread C on the correlation parameter.

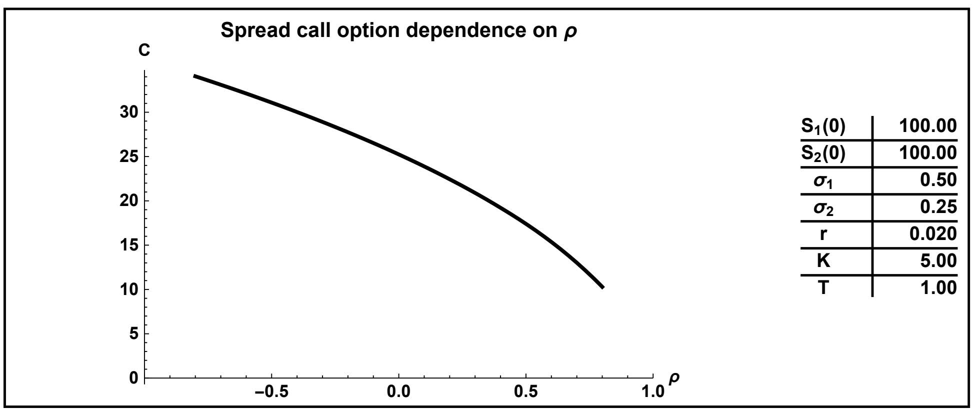


Figure 2: Dependence of call spread option on correlation.

When two stocks are highly correlated, the spread option is like an option on a single underlier. When the stocks are decorrelated, it is much less likely that their difference is less than the strike— the price of the option on the difference is correspondingly higher.

When practical consideration require that many spread options be valued, or when all of the relevant sensitivities (greeks) must be calculated, direct numerical double integration is too slow. One way to speed the calculation is to note that one of the integrations in equation (5) can be performed analytically. This procedural results in a single numerical integral that must be performed over a complicated integrand similar to the Black Scholes formula. While this method is useful, I will reserve its development for a later notebook. Instead, I review some of the classical approximate methods for spread options.

3 An exact solution — the Margrabe exchange option.

The product or ratio of two lognormally distributed random variable is lognormal—an observation that is key to the following exact solution for a special case of the spread option.

When the strike K is set to zero in equation (3), the option becomes that of exchanging underlier 2 for underlier 1. This is a classical exchange option first considered by William Margrabe. The payoff is

$$C_{\text{exchange}}(t) = e^{-r(T-t)}\,\mathbb{E}_Q[\max(S_1(T) - S_2(T), 0)] \\ = e^{-r(T-t)}\,\mathbb{E}_Q\left[S_2(T)\max\left(\frac{S_1(T)}{S_2(T)} - 1, 0\right)\right] \tag{7}$$

If S_2 is used as a numeraire (measure of value), and the expectation is valued under the T-forward measure.

$$\frac{C_{\text{exchange}}(t)}{S_2(t)} = e^{-r(T-t)}\,\mathbb{E}_T\left[\frac{S_2(T)\max\left(\frac{S_1(T)}{S_2(T)} - 1, 0\right)}{S_2(T)}\right] = e^{-r(T-t)}\,\mathbb{E}_T\left[\max\left(\frac{S_1(T)}{S_2(T)} - 1, 0\right)\right] \tag{8}$$

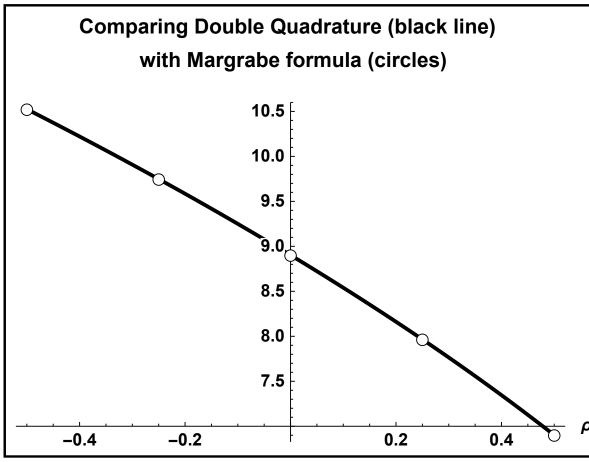
and the expectation is taken over the ratio of two lognormally distributed variables. The expectation in equation (8) is valued just as for a Black-Scholes vanilla call option to obtain

$$C_{\text{exchange}}(t) = e^{-r(T-t)}\left(S_2(t)\,\mathcal{N}(d) - S_1(t)\left(\mathcal{N}\left(d - \sigma\sqrt{T-t}\right)\right)\right) \tag{9}$$

where

$$d = \frac{\ln\left(\frac{S_1}{S_2}\right) + \frac{\sigma^2}{2}(T-t)}{\sigma\sqrt{T-t}} \\ \sigma = \sqrt{\sigma_1^2 - 2\,\sigma_1\,\sigma_2\,\rho + \sigma_2^2} \tag{10}$$

Below I compare the double quadrature call spread against the Margrabe closed form which obtains for K = 0. Also, I take the interest rate to be zero because the Margrabe option is to exchange one equity for another and discounting of fair value is not required.



T	1.00
$S_1(0)$	100.00
$S_2(0)$	100.00
σ_1	0.20
σ_2	0.10
ρ	0.00
r	0.000

Figure 3: Comparison of the Margrabe formula with double quadrature. As should be expected, the agreement is very close.

4 Bachelier approximation

Louis Bachelier was a remarkable mathematician and *waay* ahead of his time. While conferences are now held in his honor http://www.bfs2010.com/, his work was unappreciated during his lifetime. Bachelier proposed modeling stock prices with a stochastic Brownian process in 1900, some five years before Einstein applied Brownian motion to atomic and molecular motion.

Within the context of spread options, the Bachelier approximation consists of assuming the spread between stock prices, $\delta S = S_1 - S_2$ can be modeled by the Gaussian process. I sketch the derivation of the expression for a call spread option under the Bachelier approximation. Here, I follow the development in the working paper by Caramona and Durrleman 2009 but use *Mathematica* to perform some of the required calculations.

$$d\,\delta S = \mu\,\delta S\,dt + \sigma_B(t)\,dz_t \tag{11}$$

For this process we have that the first two moments of the distribution for δS are given by

$$\mathbb{E}[\delta S(t)] = \mathbb{E}_N = \delta S_0\, e^{\mu t} \tag{12}$$

$$\mathbb{var}(\delta S) \equiv V_N(t) = \int_0^t \sigma_B^2(u)\, e^{\mu(t-u)}\, dz_u \tag{13}$$

The spread option will be calculated by writing

$$\delta S\, t = \mathbb{E}_N(t) + \sqrt{V_N(t)}\;\epsilon \tag{14}$$

where ϵ is normally distributed.

The parameters of this normal model are obtain by equating these moments to the corresponding moments of the $S_1(t)$ - S_2(t) where S_1 and S_2 are lognormally distributed. Below I calculate

$$\mathbb{E}[S_1(t) - S_2(t)] \equiv \mathbb{E}_{\text{LN}} = (S_{10} - S_{20})\, e^{\mu t} \tag{15}$$

$$\mathbb{var}[S_1(t) - S_2(t)] \equiv V_{\text{LN}}(t) = e^{2\mu t}\left(-2\, S_{20}\, S_{10}\left(e^{\rho\,\sigma_1\,\sigma_2\, t} - 1\right) + S_{10}^2\left(e^{\sigma_1^2 t} - 1\right) + S_{20}^2\left(e^{\sigma_2^2 t} - 1\right)\right) \tag{16}$$

Thus, we approximate

$$\delta S\, t = \mathbb{E}_{\text{LN}}(t) + \sqrt{V_{\text{LN}}(t)}\;\epsilon \tag{17}$$

in which case

$$\begin{aligned} C(0) &= e^{-rT}\,\mathbb{E}_Q(\max(S(T) - K, 0)) \\ &= e^{-rT}\int_{-\infty}^{\infty} f(\epsilon)\max\left(\mathbb{E}_{\text{LN}}(T) + \sqrt{V_{\text{LN}}(T)}\;\epsilon - K, 0\right) d\epsilon \\ &= e^{-rT}\int_{\epsilon_*}^{\infty} f(\epsilon)\left(\mathbb{E}_{\text{LN}}(T) + \sqrt{V_{\text{LN}}(T)}\;\epsilon - K\right) d\epsilon \end{aligned} \tag{18}$$

where

$$\epsilon_* = \frac{K - \mathbb{E}_{\text{LN}}(T)}{\sqrt{V_{\text{LN}}(T)}} \tag{19}$$

Then

$$\begin{aligned} C(0) &= e^{-r\,T}\left((\mathbb{E}_{\text{LN}}(T) - K) \int_{\epsilon_*}^{\infty} f(\epsilon)\, d\epsilon + \sqrt{V_{\text{LN}}(T)} \int_{\epsilon_*}^{\infty} f(\epsilon)\,\epsilon\, d\epsilon \right) \\ &= e^{-r\,T}\left((\mathbb{E}_{\text{LN}}(T) - K)\, \mathcal{N}(-\epsilon_*) + \sqrt{V_{\text{LN}}(T)}\, f(\epsilon_*) \right) \end{aligned} \tag{20}$$

Below I compare the Bachelier approximation against the brute force double quadrature model.

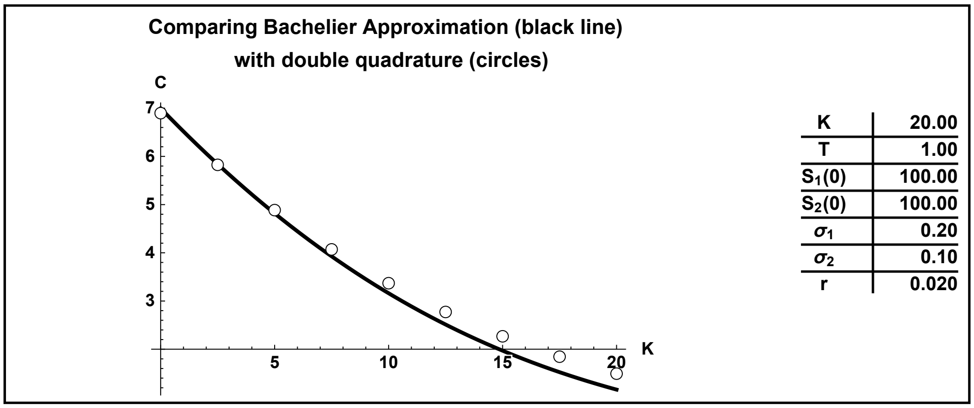


Figure 3: Comparison of the Bachelier approximation with double quadrature.

This result indicates that the Bachelier approximation is breaking down as the strike is increased. This can be explained as follows. Simulation of the distribution $S_1(T)$ - S_2(T) shows that approximating it with a moment matched normal distribution is reasonable for the center of the distribution, but not accurate in the tails. For larger values of the option strike, the expected fair value of the spread payoff is ever more dependent on the tail of the spread distribution. Thus the discrepancy between the Bachelier approximation and the brute force numerical calculation becomes more noticeable.

5 Kirk's approximation

Another widely used approximation for spread options follows from writing

$$\begin{aligned} \frac{C(0)}{e^{-r\,T}} &= \mathbb{E}_Q[\max(S_1(T) - S_2(T) - K, 0)] \\ &= \mathbb{E}_Q\left[(S_2(T) + K) \max\left(\frac{S_1(T)}{S_2(T) + K} - 1, 0\right)\right] \\ &= \mathbb{E}_Q\left[Y(T) \max\left(Z(\text{T}) - 1, 0\right)\right] \end{aligned} \tag{21}$$

Then, Y(T) = $S_2(T) + K$ is chosen as a numeraire and the expectation is taken in the T-forward measure

$$\frac{C(0)}{e^{-r\,T}\, Y(t)} = \mathbb{E}_T\left[\frac{Y(T) \max\left(Z(\text{T}) - 1, 0\right)}{Y(T)}\right] = \mathbb{E}_T\left[\max\left(Z(\text{T}) - 1, 0\right)\right] \tag{22}$$

The Kirk's approximation (E. Kirk, *Correlation in the energy markets, in managing energy price risk*, Risk Publications 1995) consists of approximating Z(T) = $\frac{S_1(T)}{S_2(T)+K}$ with a lognormal distribution, which is only exactly true for K = 0. This is equivalent to choosing the Z dynamics to be

$$\frac{dZ}{Z} = dt\, \mu_Z + dz_t\, \sigma_Z \tag{23}$$

Below, I show that

$$\mu_Z = r - \hat{r} \tag{24}$$

$$\sigma_Z = \sqrt{\sigma_1^2 - 2\,\rho\,\sigma_1\,\hat{\sigma}_2 + \hat{\sigma}_2^2} \tag{25}$$

with

$$\begin{aligned} \hat{r} &= r\,\frac{S_2}{Y} = r\,\frac{S_2}{S_2 + K} \\ \hat{\sigma}_2 &= \sigma_2\,\frac{S_2}{Y} = \sigma_2\,\frac{S_2}{S_2 + K} \end{aligned} \tag{26}$$

Then, by analogy with Black-Scholes

$$\frac{C(0)}{e^{-r\,T}\,Y(0)} = Z(0)\,e^{\mu_Z\,T}\,\mathcal{N}\left(-\left(d_Z - \sigma_Z\sqrt{T}\right) - \mathcal{N}(-d_Z)\right) \tag{27}$$

where

$$d_Z = \left(\log(1/Z(0)) - \left(\mu_Z - \frac{\sigma_Z^2}{2}\right)T\right)\Big/\left(\sigma_Z\sqrt{T}\right) \tag{28}$$

and, so

$$C(0) = S_1(0)\,e^{(\mu-r)\,T}\,\mathcal{N}\left(-\left(d_Z - \sigma_Z\sqrt{T}\right) - (S_2(0) + K)\,e^{-r\,T}\,\mathcal{N}(-d_Z)\right) \tag{29}$$

Detailed calculations

Figure 1,2 and discussion

Recasting the integral (4) into (5). The joint distribution function for ϵ_1 and ϵ_2 is

In[20]:=
```
Module[{μ, σ, temp},
 μ = {0, 0};
 σ = {{1, ρ}, {ρ, 1}};
 temp = PDF[MultinormalDistribution[μ, σ], {ϵ1, ϵ2}] // Simplify;
 temp // PowerExpand]
```

Out[20]=

$$\frac{e^{\frac{\epsilon1^2 + \epsilon2^2 - 2\,\epsilon1\,\epsilon2\,\rho}{2\,(-1+\rho^2)}}}{2\,\pi\,\sqrt{1-\rho^2}}$$

It is convenient to re-express ϵ_1 and ϵ_2 in terms of uncorrelated random variables. This is accomplished with

In[21]:=
```
A = Simplify[CholeskyDecomposition[{{1, ρ}, {ρ, 1}}], {ρ ∈ Reals}]
```

Out[21]=
$$\left\{\{1, \rho\}, \left\{0, \sqrt{1-\rho^2}\right\}\right\}$$

The correlated ϵ_1 and ϵ_2 are then expressed in terms of the uncorrelated η_1 and η_2 by

In[22]:=
$$\{\epsilon_1, \epsilon_2\} == A^{\mathsf{T}}.\{\eta_1, \eta_2\}$$

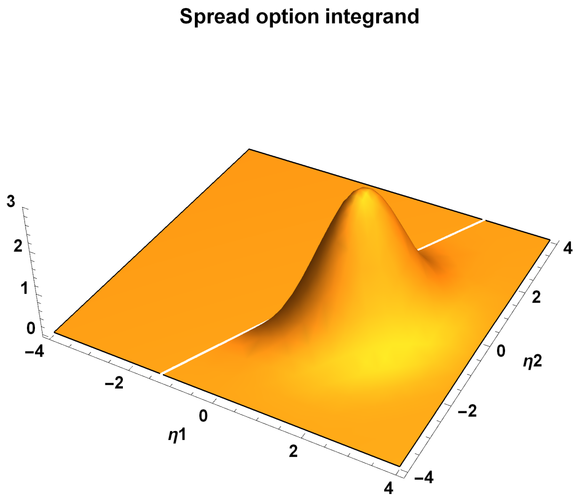
Out[22]=
$$\{\epsilon_1, \epsilon_2\} == \left\{\eta_1, \rho\,\eta_1 + \sqrt{1-\rho^2}\;\eta_2\right\}$$

I visualize the integrand of equation (5)

In[23]:=

```
Module[{S10 = 100.00, S20 = 100, K = 5., r = 0.02, σ1 = 0.5,
  σ2 = 0.25, T = 1., ρ = 0.8, ρb, ηMax = 4, lab, params, g1, g2},
 lab = Style["Spread option integrand", Bold];
 ρb = Sqrt[1 - ρ^2];
 params = {{"S₁(0)", S10, {5, 2}}, {"S₂(0)", S20, {5, 2}},
   {"σ₁", σ1, {5, 2}}, {"σ₂", σ2, {5, 2}}, {"ρ", ρ, {5, 2}},
   {"r", r, {5, 3}}, {"K", K, {5, 2}}, {"T", T, {5, 2}}};
 g1 = Plot3D[Max[(S10 Exp[(r - σ1^2/2) T + σ1 Sqrt[T] η1] -
       S20 Exp[(r - σ2^2/2) T + σ2 Sqrt[T] (ρ η1 + ρb η2)] - K), 0]
    f[η1] f[η2], {η1, -ηMax, ηMax}, {η2, -ηMax, ηMax},
   AxesLabel → {Style["η₁", Bold], Style["η₂", Bold]},
   PlotLabel → Style[lab, Bold],
   Mesh → False, Boxed → False,
    PlotRange → All,
   ImageSize → 300];
 g2 = ParameterTable[params];
 Grid[{{g1, g2}}, Frame → True]]
```

Out[23]=



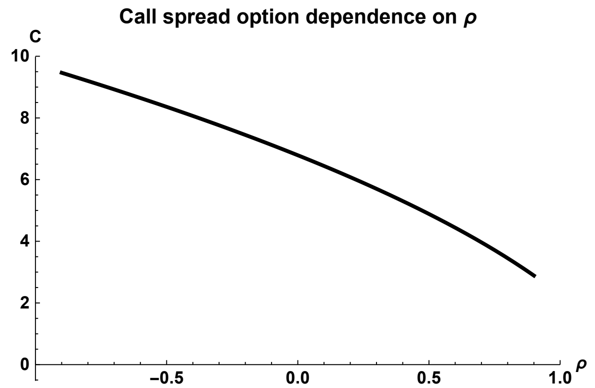
$S_1(0)$	100.00
$S_2(0)$	100.00
σ_1	0.50
σ_2	0.25
ρ	0.80
r	0.020
K	5.00
T	1.00

The integral (5) may be easily evaluated.

In[26]:=

```
Off[NIntegrate::slwcon];
Module[{S10 = 100.00, S20 = 100, K = 5., r = 0.02, σ1 = 0.2, σ2 = 0.1,
  T = 1., ρ = 0.8, ηMax = 4, lab, params, results, fit, g1, g2},
 lab = Style["Spread option integrand", Bold];
 params = {{"K", K, {5, 2}}, {"T", T, {5, 2}}, {"S₁(0)", S10, {5, 2}},
   {"S₂(0)", S20, {5, 2}}, {"σ₁", σ1, {5, 2}}, {"σ₂", σ2, {5, 2}}, {"r", r, {5, 3}}};
 results =
  ({#, CallSpreadDoubleQuadrature[K, T, S10, S20, σ1, σ2, #, r] }) & /@
   Range[-0.9, 0.9, 0.1];
 fit = Interpolation[results];
 g1 = Plot[fit[ρ], {ρ, -0.9, 0.9},
   PlotRange → {{-1, 1}, Automatic}, AxesOrigin → {-1, 0},
   PlotStyle → BLACK,
   PlotLabel → Stl["Call spread option dependence on ρ"],
   AxesLabel → {Stl["ρ"], Stl["C"]}, ImageSize → {400, 200}];
 g2 = ParameterTable[params];
 Grid[{{g1, g2}}, Frame → True] ]
```

Out[26]=



K	5.00
T	1.00
$S_1(0)$	100.00
$S_2(0)$	100.00
σ_1	0.20
σ_2	0.10
r	0.020

In[24]:=

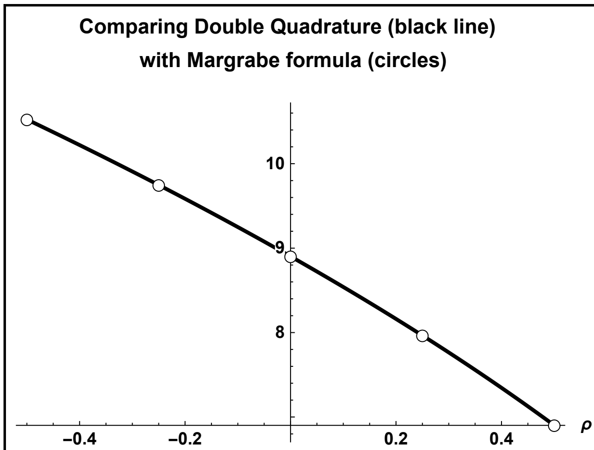
```
Clear[CallSpreadDoubleQuadrature];
CallSpreadDoubleQuadrature[K_, T_, S10_, S20_, σ1_, σ2_, ρ_, r_] :=
 Module[{ηMax = 4, ρb},
  Exp[-r T] NIntegrate[Max[(S10 Exp[(r - σ1^2/2) T + σ1 Sqrt[T] η1] -
       S20 Exp[(r - σ2^2/2) T + σ2 Sqrt[T] (ρ η1 + Sqrt[1 - ρ^2] η2)] - K), 0]
     f[η1] f[η2], {η1, -ηMax, ηMax}, {η2, -ηMax, ηMax},
    Method → {Automatic, "SymbolicProcessing" → 0}]]
```

Margrabe

In[35]:=

```
Off[NIntegrate::slwcon];
Module[{S10 = 100.00, S20 = 100, K = 0, r = 0.0, σ1 = 0.2, σ2 = 0.1,
  T = 1., ρ = 0., ηMax = 5, lab, params, results, fit, Marg, g1, g2},
 lab = Style["Spread option integrand", Bold];
 params = { {"T", T, {5, 2}}, {"S₁(0)", S10, {5, 2}}, {"S₂(0)", S20, {5, 2}},
   {"σ₁", σ1, {5, 2}}, {"σ₂", σ2, {5, 2}}, {"ρ", ρ, {5, 2}}, {"r", r, {5, 3}}};
 results = ({#, CallSpreadDoubleQuadrature[0, T, S10, S20, σ1, σ2, #, r]}) & /@
   Range[-0.5, 0.5, 0.25];
 g1 = Plot[MargrabeExchange[T, S10, S20, σ1, σ2, ρ, r], {ρ, -0.5, 0.5},
   Epilog → {OpenCircle /@ results}, AxesLabel → {Stl["ρ"], ""},
   PlotLabel → Stl["Comparing Double Quadrature
       (black line) \n with Margrabe formula (circles)"],
   ImageSize → 300];
 g2 = ParameterTable[params];
 Grid[{{g1, g2}}, Frame → True]]
```

Out[35]=



T	1.00
$S_1(0)$	100.00
$S_2(0)$	100.00
σ_1	0.20
σ_2	0.10
ρ	0.00
r	0.000

In[27]:=

```
Clear[MargrabeExchange];
MargrabeExchange[T_, S10_, S20_, σ1_, σ2_, ρ_, r_] :=
 Module[{d, σ},
  σ = Sqrt[σ1^2 - 2 σ1 σ2 ρ + σ2^2];
  d = (Log[S10/S20] + σ^2/2 T)/(σ Sqrt[T]);
  Exp[-r T] (S20 𝒩[d] - S10 𝒩[d - σ Sqrt[T]])]
```

Bachelier

I illustrate how *Mathematica* can be used to directly calculate the moments of the spread of two lognormal variables.

In[36]:=
```
w[1] = S10 Exp[(r - σ1^2/2) t + σ1 √t ϵ1] - S20 Exp[(r - σ2^2/2) t + σ2 √t (ρ ϵ1 + ρ̄ ϵ2)]
```

Out[36]=

$$e^{\sqrt{t}\,\epsilon_1\,\sigma_1+t\left(r-\frac{\sigma_1^2}{2}\right)} S_{10} - e^{\sqrt{t}\,(\rho\,\epsilon_1+\bar{\rho}\,\epsilon_2)\,\sigma_2+t\left(r-\frac{\sigma_2^2}{2}\right)} S_{20}$$

Calculation of 𝔼[S]

In[37]:=
```
w[2] = Integrate[w[1] f[ϵ1] f[ϵ2], {ϵ1, -∞, ∞}, {ϵ2, -∞, ∞}]
```

Out[37]=

$$e^{r\,t} S_{10} - e^{\frac{1}{2} t \left(2 r + \left(-1+\rho^2+\bar{\rho}^2\right) \sigma_2^2\right)} S_{20}$$

In[38]:=
```
w[3] = w[2] /. {α1 → 1, α2 → 1, ρ̄ → √(1 - ρ^2)} // Simplify
```

Out[38]=

$$e^{r\,t} \left(S_{10} - S_{20}\right)$$

Calculation of 𝕧ar[S]

In[39]:=
```
w[4] = Integrate[w[1]^2 f[ϵ1] f[ϵ2], {ϵ1, -∞, ∞}, {ϵ2, -∞, ∞}] - w[3]^2
```

Out[39]=

$$e^{t\left(2 r+\sigma_1^2\right)} S_{10}^2 - e^{2 r t} \left(S_{10}-S_{20}\right)^2 - 2\, e^{\frac{1}{2} t \left(4 r+2 \rho \sigma_1 \sigma_2+\left(-1+\rho^2+\bar{\rho}^2\right) \sigma_2^2\right)} S_{10} S_{20} + e^{t\left(2 r+\left(-1+2 \rho^2+2 \bar{\rho}^2\right) \sigma_2^2\right)} S_{20}^2$$

In[40]:=
```
w[5] = w[4] /. {α1 → 1, α2 → 1, ρ̄ → √(1 - ρ^2)} // Simplify
```

Out[40]=

$$e^{2 r t} \left(\left(-1 + e^{t \sigma_1^2}\right) S_{10}^2 - 2 \left(-1 + e^{t \rho \sigma_1 \sigma_2}\right) S_{10} S_{20} + \left(-1 + e^{t \sigma_2^2}\right) S_{20}^2\right)$$

In[41]:=
```
w[6] = w[5] /. {S10 → S10, S20 → S20, σ1 → σ1, σ2 → σ2, t → T}
```

Out[41]=

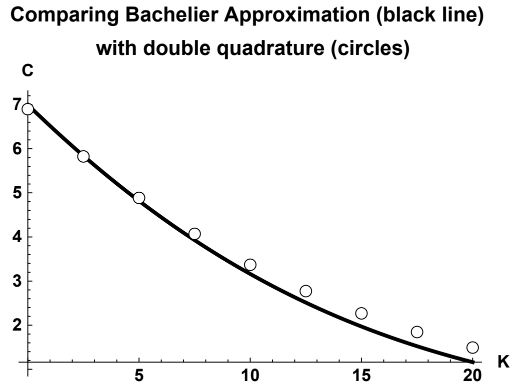
$$e^{2 r T} \left(\left(-1 + e^{T\,\sigma 1^2}\right) S10^2 - 2 \left(-1 + e^{T \rho\,\sigma 1\,\sigma 2}\right) S10\,S20 + \left(-1 + e^{T\,\sigma 2^2}\right) S20^2\right)$$

In[42]:=
```
Clear[CallSpreadBachelier];
CallSpreadBachelier[K_, T_, S10_, S20_, σ1_, σ2_, ρ_, r_] :=
 Module[{ΔS0, V, ϵΔS},
  ΔS0 = S20 - S10;
  V = e^(2 r T) ((-1 + e^(T σ1^2)) S10^2 - 2 (-1 + e^(T ρ σ1 σ2)) S10 S20 + (-1 + e^(T σ2^2)) S20^2);
  ϵΔS = (K - ΔS0 Exp[r T]) / √V;
  Exp[-r T] ((ΔS0 Exp[r T] - K) 𝒩[-ϵΔS] + √V f[ϵΔS])]
```

In[44]:=

```
Off[NIntegrate::slwcon];
Module[{S10 = 100.00, S20 = 100, K = 20, r = 0.02, σ1 = 0.2, σ2 = 0.1,
  T = 1., ρ = 0.5, ηMax = 4, lab, params, results, fit, g1, g2},
 params = {{"K", K, {5, 2}}, {"T", T, {5, 2}}, {"S₁(0)", S10, {5, 2}},
   {"S₂(0)", S20, {5, 2}}, {"σ₁", σ1, {5, 2}}, {"σ₂", σ2, {5, 2}}, {"r", r, {5, 3}}};
 results =
  ({#, CallSpreadDoubleQuadrature[#, T, S10, S20, σ1, σ2, ρ, r] }) & /@
   Range[0, 20, 2.5];
 g1 = Plot[CallSpreadBachelier[K, T, S10, S20, σ1, σ2, ρ, r],
   {K, 0, 20},
   Epilog → {OpenCircle /@ results},
   PlotStyle → BLACK,
   PlotLabel → Stl["Comparing Bachelier Approximation
       (black line) \n with double quadrature (circles)"],
   AxesLabel → {Stl["K"], Stl["C"]}, ImageSize → {400, 200}];
 g2 = ParameterTable[params];
 Grid[{{g1, g2}}, Frame → True] ]
```

Out[44]=



K	20.00
T	1.00
$S_1(0)$	100.00
$S_2(0)$	100.00
σ_1	0.20
σ_2	0.10
r	0.020

Derivation of σ_Z for Kirk's approximation

Here, I use *Mathematica* to carry out the calculations described in Analytic Approximations for Spread Options, C. Alexander and A. Venkatramanam (on web).

Although what follows is not a difficult hand calculation, it is useful to use *Mathematica* to carry it out. Some reasons are

- You only think you understand it unless YOU can program it (Chaitin)
- It illustrates techniques that will be useful in other contexts.
- It is an exercise in deliberative practice of symbolic manipulation skills

The idea is to use the dynamics of S_1 and S_2 to derive the dynamics of Y and finally Z.

The details are easier to follow if a decorated notation is used.

In[45]:=
```
<< Notation`
```

In[46]:=
```
Symbolize[ S₁ ]; Symbolize[ S₂ ]; Symbolize[ S₁₀ ]; Symbolize[ S₂₀ ];
Symbolize[ Y₀ ];
Symbolize[ r̂ ];
Symbolize[ σ̂₂ ];
Symbolize[ ρ̄ ];
```

I define the stock price dynamics

In[48]:= $w["Seqns"] = \left\{\frac{dS_1}{S_1} == r\,dt + \sigma_1\,dz_1,\ \frac{dS_2}{S_2} == r\,dt + \sigma_2\,dz_2\right\}$

Out[48]= $\left\{\frac{dS_1}{S_1} == dt\,r + dz_1\,\sigma_1,\ \frac{dS_2}{S_2} == dt\,r + dz_2\,\sigma_2\right\}$

and also for Y = $S_2 + K$.

In[49]:= $w["Yeqn"] = \{Y[t] == S_2[t] + K,\ dY[t] == dS_2[t] + K\,Exp[-r\,(T - t)]\,r\,dt\}$

Out[49]= $\{Y[t] == K + S_2[t],\ dY[t] == dt\,e^{-r\,(-t+T)}\,K\,r + dS_2[t]\}$

We have Z = $\frac{S_1}{Y}$. I make an Ito expansion to obtain a form for the dynamics of Z. The dependencies of the dependent variable must be made explicit so that *Mathematica* can carry out the expansion.

In[50]:= $w[1] = Z[S_1, Y] == Series[Z[S_1, Y], \{S_1, S_{10}, 2\}, \{Y, Y_0, 2\}]\ //\ Normal$

Out[50]= $Z[S_1, Y] == Z[S_{10}, Y_0] + (Y - Y_0)\,Z^{(0,1)}[S_{10}, Y_0] + \frac{1}{2}\,(Y - Y_0)^2\,Z^{(0,2)}[S_{10}, Y_0] + (S_1 - S_{10})\left(Z^{(1,0)}[S_{10}, Y_0] + (Y - Y_0)\,Z^{(1,1)}[S_{10}, Y_0] + \frac{1}{2}\,(Y - Y_0)^2\,Z^{(1,2)}[S_{10}, Y_0]\right) + (S_1 - S_{10})^2\left(\frac{1}{2}\,Z^{(2,0)}[S_{10}, Y_0] + \frac{1}{2}\,(Y - Y_0)\,Z^{(2,1)}[S_{10}, Y_0] + \frac{1}{4}\,(Y - Y_0)^2\,Z^{(2,2)}[S_{10}, Y_0]\right)$

The differentials are introduced. Here is ϵ << 1 is an expansion parameter

In[51]:= $w[2] = w[1]\ /.\ \{Z[S_1, Y] \to dZ + Z[S_{10}, Y_0],\ S_1 \to \epsilon\,dS_1 + S_{10},\ Y \to \epsilon\,dY + Y_0\}\ //\ ExpandAll$

Out[51]= $dZ + Z[S_{10}, Y_0] == Z[S_{10}, Y_0] + dY\,\epsilon\,Z^{(0,1)}[S_{10}, Y_0] + \frac{1}{2}\,dY^2\,\epsilon^2\,Z^{(0,2)}[S_{10}, Y_0] + \epsilon\,dS_1\,Z^{(1,0)}[S_{10}, Y_0] + dY\,\epsilon^2\,dS_1\,Z^{(1,1)}[S_{10}, Y_0] + \frac{1}{2}\,dY^2\,\epsilon^3\,dS_1\,Z^{(1,2)}[S_{10}, Y_0] + \frac{1}{2}\,\epsilon^2\,dS_1^2\,Z^{(2,0)}[S_{10}, Y_0] + \frac{1}{2}\,dY\,\epsilon^3\,dS_1^2\,Z^{(2,1)}[S_{10}, Y_0] + \frac{1}{4}\,dY^2\,\epsilon^4\,dS_1^2\,Z^{(2,2)}[S_{10}, Y_0]$

Rules can be used to truncate the expansion

In[52]:=
```
w[3] = w[2] /. ϵ^n_/; n>2 → 0 /. ϵ → 1
```

Out[52]=

$$dZ + Z[S_{10}, Y_0] == Z[S_{10}, Y_0] + dY\, Z^{(0,1)}[S_{10}, Y_0] + \frac{1}{2} dY^2 Z^{(0,2)}[S_{10}, Y_0] + dS_1 Z^{(1,0)}[S_{10}, Y_0] + dY\, dS_1 Z^{(1,1)}[S_{10}, Y_0] + \frac{1}{2} dS_1^2 Z^{(2,0)}[S_{10}, Y_0]$$

Solve this for dZ and redefine the expansion point variables to make the notation more standard

In[53]:=
```
w[4] = Solve[w[3], dZ][[1, 1]] /. Rule → Equal /. {Y_0 → Y, S_10 → S_1}
```

Out[53]=

$$dZ == \frac{1}{2} \left(2\, dY\, Z^{(0,1)}[S_1, Y] + dY^2 Z^{(0,2)}[S_1, Y] + 2\, dS_1 Z^{(1,0)}[S_1, Y] + 2\, dY\, dS_1 Z^{(1,1)}[S_1, Y] + dS_1^2 Z^{(2,0)}[S_1, Y]\right)$$

Introduce the explicit dependencies of Z = $Z(S_1, \text{Y}) = \frac{S_1}{Y}$

In[54]:=
```
w["Zeqn"] = w[4] /. Z → ((#1 / #2) &) // Expand
```

Out[54]=

$$dZ == \frac{dY^2 S_1}{Y^3} - \frac{dY\, S_1}{Y^2} - \frac{dY\, dS_1}{Y^2} + \frac{dS_1}{Y}$$

Recall the Y eqn and remove the [t] dependence that is no longer needed

In[55]:=
```
w2[1] = w["Yeqn"][[2]] /. a_[t] → a // ExpandAll
```

Out[55]=

$$dY == dt\, e^{r t - r T} K r + dS_2$$

This is rewritten

In[56]:=
```
w2[2] = (# / Y) & /@ w2[1] /. a_[t] → a // ExpandAll
```

Out[56]=

$$\frac{dY}{Y} == \frac{dt\, e^{r t - r T} K r}{Y} + \frac{dS_2}{Y}$$

Introduce the dependence on S_2

In[57]:=
```
w2[3] = w2[2] /. Solve[w["Seqns"][[2]], dS_2][[1, 1]] // ExpandAll
```

Out[57]=

$$\frac{dY}{Y} == \frac{dt\, e^{r t - r T} K r}{Y} + \frac{dt\, r\, S_2}{Y} + \frac{S_2\, dz_2\, \sigma_2}{Y}$$

Notice that K << S_2, the first term on the rhs is small with respect to the second. Operationally, this can be accomplished by just setting K to zero. Note that the lowest order K dependence is still embedded in the definition of Y.

In[58]:=
```
w2[4] = w2[3] /. K → 0
```

Out[58]=

$$\frac{dY}{Y} == \frac{dt\, r\, S_2}{Y} + \frac{S_2\, dz_2\, \sigma_2}{Y}$$

Following Alexander and Venkatramanam, I introduce some variables of convenience.

In[59]:=
```
{def[r̂] = r̂ == r S2/Y, def[σ̂2] = σ̂2 == σ2 S2/Y, def[ρ̄] = ρ̄ == Sqrt[1 - ρ^2]}
```

Out[59]=

$$\left\{\hat{r} == \frac{r\, S_2}{Y},\ \hat{\sigma}_2 == \frac{S_2\, \sigma_2}{Y},\ \bar{\rho} == \sqrt{1-\rho^2}\right\}$$

In[60]:=
```
w["newYeqn"] = w2[4] /. {Solve[def[r̂], r][[1, 1]], Solve[def[σ̂2], σ2][[1, 1]]}
```

Out[60]=

$$\frac{dY}{Y} == dt\, \hat{r} + \hat{\sigma}_2\, dz_2$$

This approximate equation for the Y dynamics is now substituted into the equation for Z

In[61]:=
```
w["Zeqn"]
```

Out[61]=

$$dZ == \frac{dY^2\, S_1}{Y^3} - \frac{dY\, S_1}{Y^2} - \frac{dY\, dS_1}{Y^2} + \frac{dS_1}{Y}$$

In[62]:=
```
w3[1] = w["Zeqn"] /. {Solve[w["newYeqn"], dY][[1, 1]],
    Solve[w["Seqns"][[1]], dS1][[1, 1]]} // ExpandAll
```

Out[62]=

$$dZ == \frac{dt\, r\, S_1}{Y} - \frac{dt\, \hat{r}\, S_1}{Y} - \frac{dt^2\, r\, \hat{r}\, S_1}{Y} + \frac{dt^2\, \hat{r}^2\, S_1}{Y} - \frac{S_1\, \hat{\sigma}_2\, dz_2}{Y} - \frac{dt\, r\, S_1\, \hat{\sigma}_2\, dz_2}{Y} + \frac{2\, dt\, \hat{r}\, S_1\, \hat{\sigma}_2\, dz_2}{Y} + \frac{S_1\, \hat{\sigma}_2^2\, dz_2^2}{Y} + \frac{S_1\, dz_1\, \sigma_1}{Y} - \frac{dt\, \hat{r}\, S_1\, dz_1\, \sigma_1}{Y} - \frac{S_1\, \hat{\sigma}_2\, dz_1\, dz_2\, \sigma_1}{Y}$$

This must be truncated by using the Ito orderings

In[63]:=
```
w3[2] = w3[1] /. {dz1 dz2 → ρ dt, dz2^2 → dt, dt dzi_ → 0, dt^2 → 0}
```

Out[63]=

$$dZ == \frac{dt\, r\, S_1}{Y} - \frac{dt\, \hat{r}\, S_1}{Y} + \frac{dt\, S_1\, \hat{\sigma}_2^2}{Y} - \frac{S_1\, \hat{\sigma}_2\, dz_2}{Y} - \frac{dt\, S_1\, \rho\, \hat{\sigma}_2\, \sigma_1}{Y} + \frac{S_1\, dz_1\, \sigma_1}{Y}$$

or

In[64]:=
```
w3[3] = (# / Z) & /@ w3[2] /. S1 → Z Y // Expand
```

Out[64]=

$$\frac{dZ}{Z} == dt\, r - dt\, \hat{r} + dt\, \hat{\sigma}_2^2 - \hat{\sigma}_2\, dz_2 - dt\, \rho\, \hat{\sigma}_2\, \sigma_1 + dz_1\, \sigma_1$$

or

In[65]:=
```
w3[4] = w3[3][[1]] == Collect[w3[3][[2]], {dt, dz1, dz2}]
```
Out[65]=
$$\frac{dZ}{Z} = -\hat{\sigma}_2\, dz_2 + dz_1\, \sigma_1 + dt\left(r - \hat{r} + \hat{\sigma}_2^2 - \rho\, \hat{\sigma}_2\, \sigma_1\right)$$

The next step is to introduce the explicit dependency on ρ

In[66]:=
```
w3[5] = w3[4] /. dz1 → ρ dz2 + ρ̄ dz3
```
Out[66]=
$$\frac{dZ}{Z} = -\hat{\sigma}_2\, dz_2 + \left(\rho\, dz_2 + \overline{\rho}\, dz_3\right)\sigma_1 + dt\left(r - \hat{r} + \hat{\sigma}_2^2 - \rho\, \hat{\sigma}_2\, \sigma_1\right)$$

In order that the process for Z be a martingale, we must use Girsanov

In[67]:=
```
w3[6] = w3[5] /. dz2 → dz4 + σ̂2 dt // Expand
```
Out[67]=
$$\frac{dZ}{Z} = dt\, r - dt\, \hat{r} - \hat{\sigma}_2\, dz_4 + \overline{\rho}\, dz_3\, \sigma_1 + \rho\, dz_4\, \sigma_1$$

or

In[68]:=
```
w3[7] = w3[6][[1]] == Collect[w3[6][[2]], {dt, dz3, dz4}]
```
Out[68]=
$$\frac{dZ}{Z} = dt\left(r - \hat{r}\right) + \overline{\rho}\, dz_3\, \sigma_1 + dz_4\left(-\hat{\sigma}_2 + \rho\, \sigma_1\right)$$

Here, dz_3 and dz_4 are normally distributed and independent. An equivalent process can be defined

In[69]:=
```
w3["finalZeqn"] = w3[7][[1]] == μZ dt + σZ dz5
```
Out[69]=
$$\frac{dZ}{Z} = dt\, \mu_Z + dz_5\, \sigma_Z$$

To calculate μ_Z and σ_Z, we can use some new tools from *Mathematica* version 8.

To obtain μ_Z we take the expectation of the the rhs of the dZ equation for the case that the Gaussian processes dz_3 and dz_4 are binormally distributed with zero correlation

In[70]:=
```
w4[1] = μZ dt ==
  Expectation[w3[7][[2]], {dz3, dz4} ≈ BinormalDistribution[{0, 0}, {1, 1}, 0]]
```
Out[70]=
$$dt\, \mu_Z = dt\left(r - \hat{r}\right)$$

or

In[71]:=
```
w["μZ"] = Solve[w4[1], μZ][[1, 1]] /. Rule → Equal
```
Out[71]=
$$\mu_Z = r - \hat{r}$$

Similarly, for the variance

In[72]:=
```
w4[3] =
 σZ^2 == Expectation[(w3[7][[2]] - μZ dt)^2, {dz3, dz4} ≈ BinormalDistribution[{0, 0},
       {1, 1}, 0]] /. (w["μZ"] /. Equal → Rule) // Expand
```

Out[72]= $\sigma_Z^2 = \hat{\sigma}_2^2 - 2\rho\,\hat{\sigma}_2\,\sigma_1 + \rho^2\,\sigma_1^2 + \bar{\rho}^2\,\sigma_1^2$

or

In[73]:=
```
w4[4] = w4[3] /. Solve[def[ρ̄], ρ̄][[1, 1]] // Expand
```

Out[73]= $\sigma_Z^2 = \hat{\sigma}_2^2 - 2\rho\,\hat{\sigma}_2\,\sigma_1 + \sigma_1^2$

Thus we have

In[74]:=
```
w["σZ"] = Solve[w4[4], σZ][[2, 1]] /. Rule → Equal
```

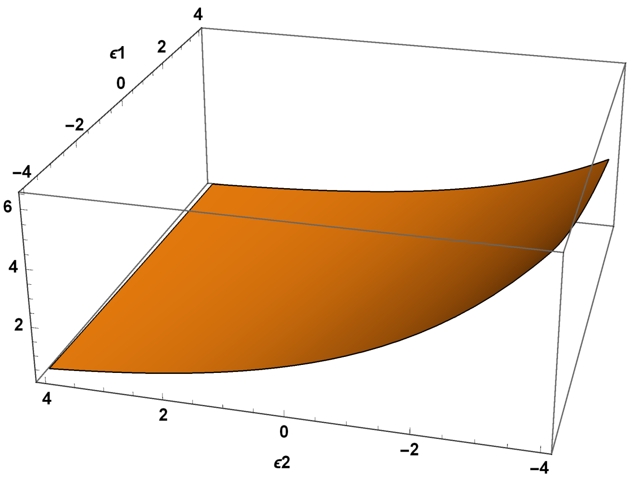
Out[74]= $\sigma_Z = \sqrt{\hat{\sigma}_2^2 - 2\rho\,\hat{\sigma}_2\,\sigma_1 + \sigma_1^2}$

Some numerics

In[75]:=
```
Clear[ZT];
ZT[ϵ1_, ϵ2_, S10_, S20_, σ1_, σ2_, ρ_, r_, K_, T_] :=
 (S10 Exp[(r - σ1^2/2) T + σ1 Sqrt[T] ϵ1]) /
  (S20 Exp[(r - σ2^2/2) T + σ2 Sqrt[T] (ρ ϵ1 + Sqrt[1 - ρ^2] ϵ2)] + K)
```

In[77]:=
```
Module[{S10 = 100, S20 = 100, σ1 = 0.1,
  σ2 = 0.4, ρ = 0.5, r = 0, K = 0, T = 1, ϵMax = 4},
 Plot3D[ZT[ϵ1, ϵ2, S10, S20, σ1, σ2, ρ, r, K, T], {ϵ1, -ϵMax, ϵMax},
  {ϵ2, -ϵMax, ϵMax}, PlotRange → All, PlotPoints → 30, Mesh → False,
  AxesLabel → {Stl["ϵ1"], Stl["ϵ2"]}]]
```

Out[77]=



In[78]:=
```
Module[{S10 = 100, S20 = 100, σ1 = 0.1,
  σ2 = 0.4, ρ = 0.5, r = 0, K = 0, T = 1, ϵMax = 4},
 NIntegrate[ZT[ϵ1, ϵ2, S10, S20, σ1, σ2, ρ, r, K, T] f[ϵ1] f[ϵ2],
  {ϵ1, -ϵMax, ϵMax}, {ϵ2, -ϵMax, ϵMax}]]
```

Out[78]= 1.15004

In[79]:=

$$\text{ww}[1] = \mathbb{E}Z == \text{Expectation}\left[Z0\,\text{Exp}\left[\left(\mu - \frac{\sigma^2}{2}\right) T + \epsilon\,\sigma\,\sqrt{T}\right],\ \epsilon \approx \text{NormalDistribution}[0, 1]\right];$$

```
ww[2] = Solve[ww[1], μ][[1, 1]] // Quiet
```

Out[80]=

$$\mu \to \text{ConditionalExpression}\left[\frac{2\,i\,\pi\,C[1] + \text{Log}\left[\frac{\mathbb{E}Z}{Z0}\right]}{T},\ C[1] \in \text{Integers}\right]$$

In[88]:=
```
Module[{S10 = 100, S20 = 100, σ1 = 0.1, σ2 = 0.4,
  ρ = 0.5, r = 0.04, K = 5, T = 1, μNum, μAnal, results},
 μNum = μNumerical[S10, S20, σ1, σ2, ρ, r, K, T] ;
 μAnal = μAnalytical[S10, S20, σ1, σ2, ρ, r, K, T];
 results = Table[{K, μAnalytical[S10, S20, σ1, σ2, ρ, r, K, T],
    μNumerical[S10, S20, σ1, σ2, ρ, r, K, T]}, {K, 0, 10, 2.5}];
 Grid[results, Frame → All]]
```

Out[88]=

0.	0.	0.14
2.5	0.00097561	0.132952
5.	0.00190476	0.126584
7.5	0.0027907	0.120802
10.	0.00363636	0.115532

In[81]:=
```
Clear[μAnalytical];
μAnalytical[S10_, S20_, σ1_, σ2_, ρ_, r_, K_, T_] :=
 Module[{rhat, Y0 = S20 + K},
  rhat = r S20/Y0;
  r - rhat]
```

In[85]:=
```
Clear[μNumerical];
μNumerical[S10_, S20_, σ1_, σ2_, ρ_, r_, K_, T_] :=
 Module[{𝔼Z, Y0 = S20 + K, Z0},
  Z0 = S20/Y0;
  𝔼Z = NExpectation[ZT[ϵ1, ϵ2, S10, S20, σ1, σ2, ρ, r, K, T],
     {ϵ1, ϵ2} ≈ BinormalDistribution[{0, 0}, {1, 1}, 0]];
  1/T Log[𝔼Z/Z0]]
```

In[87]:=
```
Module[{S10 = 100, S20 = 100, σ1 = 0.1,
  σ2 = 0.4, ρ = 0.5, r = 0, K = 0, T = 1, ϵMax = 4},
 NExpectation[ZT[ϵ1, ϵ2, S10, S20, σ1, σ2, ρ, r, K, T],
  {ϵ1, ϵ2} ≈ BinormalDistribution[{0, 0}, {1, 1}, 0]]]
```

Out[87]=
```
1.15027
```

Functions

In[89]:=
```
Clear[f];
f[ϵ_] := 1/Sqrt[2 π] Exp[-ϵ^2/2]
```

In[102]:=
```
Clear[𝒩];
𝒩::usage = "Cumulative standard normal distribution function";
𝒩[z_?NumberQ] := (1/2) (1 + Erf[z/Sqrt[2]]) // N;
```

In[91]:=
```
Clear[ParameterTable];
ParameterTable[pList_] :=
 Module[{i, nList = {}, name, value, description, format, NF, SN},
  SN[x_] := Style[x, 10, Bold, FontFamily → "Helvetica"];
  NF[x_, f_] := If[f[[2]] == 0, Round[x], NumberForm[x, f]];
  For[i = 1, i ≤ Length[pList], i++,
   {name, value, format} = pList[[i]];
   AppendTo[nList,
    {Item[SN[name]], Item[SN[NF[value, format]], Alignment → Right]}]];
  Grid[nList,
   Dividers → {{2 → True}, Center}, Spacings → {{1, {2}}, 1/3}]]
```

In[93]:=
```
Clear[Stl];
Stl[x_] := Style[x, Bold, FontFamily → "Helvetica"]
```

In[95]:=
```
BLACK = Directive[Thick, Black];
BLUE = Directive[Thick, Blue];
RED = Directive[Thick, Red];
GREEN = Directive[Thick, Green];
```

In[99]:=
```
SetOptions[Plot, PlotStyle → BLACK, LabelStyle → Directive[Bold, "Helvetica"]];
SetOptions[Plot3D, LabelStyle → Directive[Bold, "Helvetica"]];
SetOptions[Graphics, LabelStyle → Directive[Bold, "Helvetica"]];
```

References to earlier notebooks

Spread Option model for Gateway 07 - 01 - 10. nb
Checking FEA Kirk Model 06-09-10.nb
Spread Option model for Power Plant Hedge 05-17-10.nb
SpreadOption Expected 05-3-10.nb
Spread Option - Bills Hueristic 05-11-10.nb
Put Spread 04-07-10.nb
Energy Derivatives and Spread Options 06-08-09.nb
PowerGen Spread Option 01-21-10.nb
PGE Demos 06-03-09.nb
Debugging Single Quadrature 06-03-09.nb
Spread Options ala Alexander 05-18-09.nb
Spread Option Approximations R3 05-13-09.nb
Bachelier Spread Option R3 (for Joe Isaac) 05-07-09.nb