

Load Useful Package

```
In[1]:= << Documents/MacUseful.m
```

Text

■ When You Cannot Hedge Continuously: The Corrections to Black-Scholes

The insight behind the Black-Scholes formula for options valuation is the recognition that, if you know the future volatility of a stock, you can replicate an option payoff exactly by a continuous rebalancing of a portfolio consisting of the underlying stock and a risk-free bond. If no arbitrage is possible, then the value of the option should be the cost of the replication strategy.

It is widely recognized that the Black-Scholes model is imperfect in many respects. It makes the idealized assumption that future stock evolution is lognormal with a known volatility, it ignores transaction costs and market impact, and it assumes that trading can be carried out continuously. Nevertheless the model is rational, and therefore amenable to rational modification of its assumptions and results; the Black-Scholes theory can be used to investigate its own shortcomings. This is an active area of research for the Quantitative Strategies Group at Goldman Sachs.

In this note we examine the effect of dropping only one of the key Black-Scholes assumptions, that of the possibility of continuous hedging. We examine the error that arises when you replicate a single option throughout its lifetime according to the Black-Scholes replication strategy, with the constraint that only a discrete number of rebalancing trades at regular intervals are allowed. To study the replication error, we carry out Monte Carlo simulations of the Black-Scholes hedging strategy for a single European-style option and analyze the resulting uncertainty (that is, the error) in the replication value. We will show that these errors follow a simple rule-of-thumb that is related to statistical uncertainty that arises in estimating volatility from a discrete number of observations: the typical error in the replication value is proportional to the vega of the option (its volatility sensitivity, sometimes also called kappa) multiplied by the uncertainty in its observed volatility.

■ Simulating The Hedging Scenario and Calculating the Replication Error

Suppose at time $t = 0$ the option hedger sells a European put with strike K and expiration $t = T$, and receives the Black-Scholes value as the options premium. We assume that the world operates within the Black-Scholes framework: the underlying stock price evolves lognormally with a fixed known volatility σ that stays constant throughout time. For simplicity, we assume the stock pays no dividends.

The hedger now follows a Black-Scholes hedging strategy, rehedging at discrete, evenly spaced time intervals as the underlying stock changes. At expiration, the hedger delivers the option payoff to the option holder, and unwinds the hedge. We are interested in understanding the final profit or loss of this strategy:

Value of Final P&L = Black-Scholes hedge at T - Final option payoff

If the hedger had followed the exact Black-Scholes replication strategy, rehedging continuously as the underlying stock evolved towards its final value at expiration, then, no matter what path the stock took, the final P&L would be exactly zero. This is the essence of the statement that the Black-Scholes formula provides the "fair" value of the option. When the replication strategy deviates from the exact Black-Scholes method, the final P&L may deviate from zero. This deviation is called the replication error.

When the hedger rebalances at discrete rather than continuous intervals, the hedge is imperfect and the replication is inexact. Sometimes this imperfection favors the P&L, sometimes it penalizes it. The more often hedging occurs, the smaller the replication error.

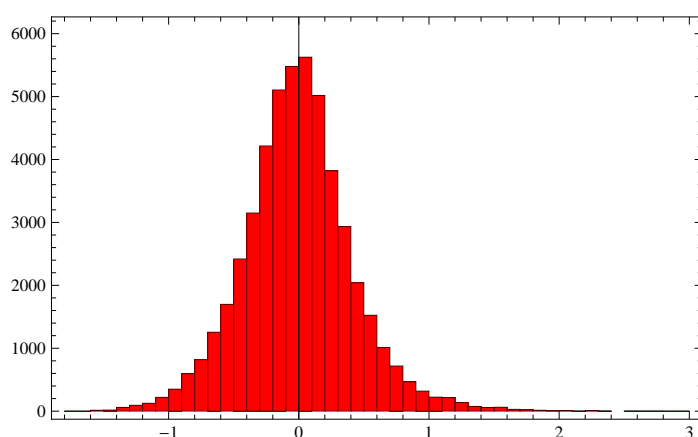
To examine the range of possibilities, we have carried out Monte Carlo simulations in which we evaluate the outcome of the discrete hedging strategy over 50,000 different, randomly generated scenarios of future stock price evolution, assuming that the future stock price grows at a continuously compounded rate equal to the discount rate $r = 5\%$, and with a volatility $\sigma = 20\%$. In each scenario we carry out N rehedging trades spaced evenly in time over the life of the option, using the Black-Scholes hedge ratio (also calculated with $r = 5\%$, and $\sigma = 20\%$).

Exhibit 1 shows histograms for the final P&L for a one-month at-the-money put hedged, at discrete times, to expiration. The initial stock price is taken to be $S_0 = 100$, with the strike K also at a level of 100 and an option time to expiration of $T = 1/12$ years. The fair Black-Scholes value of the option is $C_0 = 2.512$. In Exhibit 1a, the option is hedged, approximately once per business day; in Exhibit 1b the option is hedged 84 times, or four times as frequently. Table 1 summarizes the mean and standard deviation of the final P&L from the 50,000 scenario.

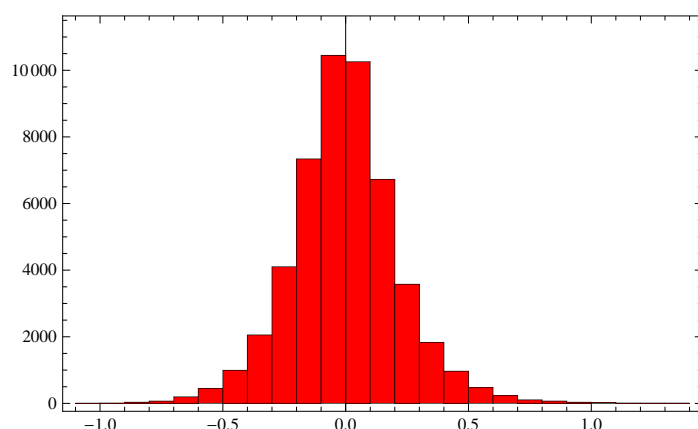
We can draw several conclusions from these results. First, within the statistical error of the simulations, the average final profit/loss is zero. Hedging discretely does not bias the outcome in either direction when all other parameters (volatility, rates, dividends) are known correctly. Second, hedging more frequently reduces the standard deviation of the P&L. The standard deviation, as a fraction of the option premium is 16.3% for $N = 21$, and 8.1% for $N = 84$; hedging four times as frequently roughly halves the standard deviation of the replication error. Third, the final distribution of replication error resembles a normal distribution, which means characterizing risk in terms of a standard deviation of the distribution makes good sense. Even though the standard deviation is 16% of premium for daily reheding, there is a normally distributed probability of much larger hedging errors, both positive and negative.

■ **Exhibit 1: Histograms for the final profit/loss of the hedging strategy for (a) 21 and (b) 84 rebalancing trades.**

(a)



(b)



■ **Table 1: Statistical summary of the simulated profit/loss. The fair Black-Scholes value of the option is 2.512**

Number of trades	Mean P&L	Standard Dev. of P&L	StDev of P&L as a % of option premium
21	-0.006	0.42	16.7
84	-0.003	0.22	8.7

■ **Understanding the Results Intuitively**

The simulation plots of Exhibit 1 are a compilation of numerical results. We have derived a simple

analytical rule that provides good quantitative agreement with these results and lends itself to a simple intuitive interpretation; the standard deviation of the final P&L is given, to a good approximation, by:

$$\sigma_{P\&L} = \sqrt{\frac{\pi}{4}} (\kappa) \frac{\sigma}{\sqrt{N}} \quad (1)$$

Here κ is the options vega -- the standard Black- Scholes sensitivity of the option price with respect to volatility, evaluated at the initial spot and trading date:

$$\kappa = \frac{dC}{d\sigma} (t = 0, S = S_0) = S_0 \sqrt{T} \frac{\text{Exp}[-d_1^2]}{\sqrt{2\pi}} \quad (2)$$

where d_1 is given the formula:

$$d_1 = \frac{\text{Log}[S_0 / K] + (r + \sigma^2 / 2) T}{\sigma \sqrt{T}} \quad (3)$$

as is used in the Black-Scholes formula. The number of times the hedger can rebalance is given by N . If you rebalance four times as frequently, you halve the typical size of the hedging error. The numerical prefactor, $\sqrt{\pi/4}$ is slightly less than one: 0.886; for a rough estimate it can be taken to be one.

Using a standard Black-Scholes calculator, it easy to verify that the initial vega of the option we have analyzed is $\kappa = 0.115/\text{vol point}$. Once you know the relevant vega, it is easy to apply equation 1; for $N = 21$, we would estimate $\sigma_{P\&L} = 0.443$, while $\sigma_{P\&L} = 0.222$. These estimates are quite close to the simulation results of Table 1.

There is an even simpler way of using this rule for the case of an option that is struck close to the initial spot price. In this case, since the option price itself is approximately proportional to the volatility, the relation expressing the standard deviation of replication error as a multiple of initial option price becomes:

$$\frac{\sigma_{P\&L}}{C_0} \approx \sqrt{\frac{\pi}{4}} \frac{1}{\sqrt{N}} \quad (4)$$

Observe that the right-hand side of this equation only depends on the number of times you hedge. This simpler approximation remains reasonably accurate: $\sigma_{P\&L} / C_0$ is 19.3% for $N = 21$, 9.7% for $N = 84$.

Equation (1) for the replication error is actually a reasonable approximation of a more complicated formula, which we will not derive here. Although the full formula provides better agreement than Equation (1) for options that are far out-of-the- money, have a longer term to expiration, or are priced under higher interest rates, their qualitative behavior is quite similar. In any case, for a wide range of pricing parameters of practical interest ($T < 1$ year, $r < 10\%$, strikes close enough to spot, say within 0.25 put-delta), the estimates are quite close.

There is an intuitive way of understanding Equations (1) and (2). When you hedge intermittently rather than continuously along some path to expiration, you are sampling the underlying price discretely, and therefore obtaining only an approximate measure of the true volatility of the underlying. Therefore, the estimated volatility itself will have an error. The histograms of Exhibit 2 show the range of volatilities estimated from each of the 50,000 paths used in the simulation analysis.

The volatility measured over all paths is, on average, close to 20%. But there is considerable variation in the volatility measured from discretely sampled prices --- even, as is the case here, when the underlying process really is lognormal with known constant volatility, and no jumps or other deviations are allowed. The discrete sampling itself introduces this error. This error is analogous to the statistical fluctuations that are seen when flipping a fair coin N times -- the measured frequency of heads will on average be $1/2$, but there will be variations around the average roughly of size $1/\sqrt{N}$. The same \sqrt{N} shows up here: the standard deviation of the measured volatility based on N samples is $\sigma / \sqrt{2N}$.

Therefore, in an approximate way, the volatility you may actually experience in hedging the option is:

$$\sigma_{\text{sampled}} \approx \sigma \pm \frac{\sigma}{\sqrt{2N}} \quad (5)$$

Consequently, the option price you may capture by replication is an option price corresponding to a volatility somewhere in this range, so that:

$$C[\sigma_{\text{sampled}}] - C[\sigma] \approx \kappa(\sigma_{\text{sampled}} - \sigma) \quad (6)$$

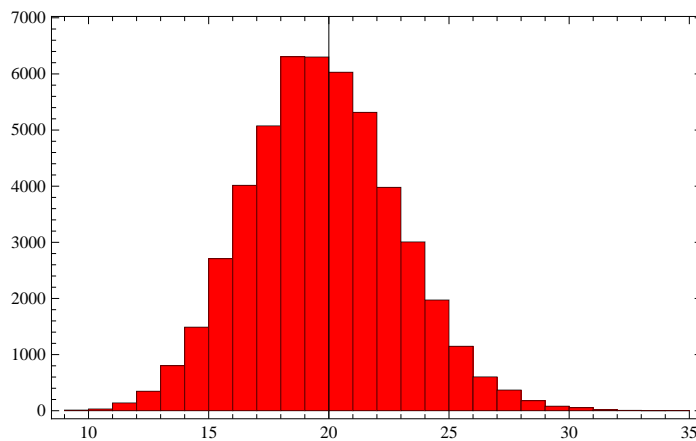
where $C[\sigma]$ is the fair Black-Scholes option price for volatility σ . We have used the option vega, κ , to approximately account for the option's sensitivity to this error in volatility. As a result, we should expect an average profit/loss of zero and a standard deviation in profit/loss that is proportional to the standard deviation of measured volatility:

$$\sigma_{P\&L} \approx \kappa \frac{\sigma}{\sqrt{2N}} \quad (7)$$

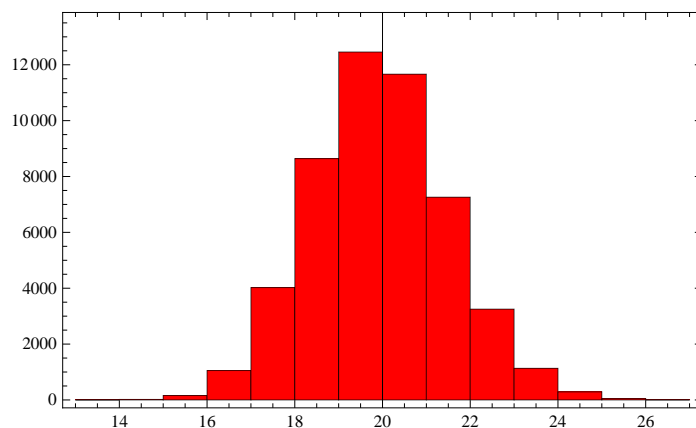
Equation (7), except for the slightly different numerical coefficient, agrees with the more rigorous result obtained in Equation (1), and provides an easily understandable heuristic explanation for the behavior of the replication error displayed in Exhibit 1.

■ **Exhibit 2: Histograms showing the volatilities estimated from (a) 21 and (b) 84 simulated returns.**

(a)



(b)



■ **Trading Volatility and its Risks**

In this note we have focused on only one of the many risks of hedging options. Intermittent hedging alone can lead to large replication errors and consequent P&L fluctuations, even when all other option market parameters are known. The only way to reduce this particular error is to re hedge more frequently, or to run a more closely matched book whose gamma is close to zero, thereby avoiding the need to re hedge. In practice, however, the advantage gained by increased rehedging must be weighed against the losses due to transaction costs and potentially adverse market impact.

Currently, global index volatilities and their skews are extremely high, perhaps making the idea of selling volatility attractive (See the Global Risk and Option Strategy Perspective section in this Quar-

terly). If implementing this volatility view requires delta-hedging, then our analysis shows that the premium collected by selling options must be at least large enough to leave a cushion for the replication error induced by discrete hedging. This error is not small; expressed in volatility terms, a one-standard-deviation estimate of this error is approximately σ / \sqrt{N} , where N is the number of reheding trades.

Including other sources of hedging risk, such as the possibility of jumps, changes in the future levels of volatility, and transaction costs, will only increase the need for an even larger cushion.

Investors interested in trading volatility should be aware that there are other strategies that either allow one to take a volatility position outright, such as volatility swaps, or that contain short volatility exposure link to index exposure -- spreads across options of with different strikes or terms to expirations. Volatility swaps are forward contracts on realized volatility. Much like a standard forward contract, the holder of a long position receives at expiration the difference between the actual volatility of the underlying over the period and a strike that is set at inception. In this way, the investor can implement a view on volatility while leaving the actual mechanics of capture to the option dealer. More information on volatility trading can be found in the references listed at the end of the note, or from your Goldman Sachs equity derivatives salesperson.

■ References

1. Investing in Volatility, E. Derman, Michael Kamal, Iraj Kani, John McClure, Cyrus Pirasteh, Joseph Zou, Quantitative Strategies Research Notes (Goldman Sachs & Co.), October 1996.
2. Valuing Derivatives with Payoff based on Future Realized Volatility, Emanuel Derman, Michael Kamal, Iraj Kani, Joseph Zou, Global Derivatives Quarterly Review, Equity Derivatives Research (Goldman Sachs & Co., July 1996)

Formulas

? **bspremium**

bspremium[ϕ ,S,K,r,q, σ ,t] returns the B&S premium for a call ($\phi=1$) or a put($\phi=-1$)

? **bsdelta**

bsdelta[ϕ ,S,K,r,q, σ , τ] returns the B&S delta for a call ($\phi=1$) or a put($\phi=-1$)

? **bsgamma**

bsgamma[S,K,r,q, σ ,t] returns the B&S gamma for a call or a put

? **MCPaths**

MCPaths[paths,steps,S0,drift,vol,t] returns paths/2 pairs of antithetic MC paths

```
In[2]:= {bspremium[1, 100, 100, 0.05, 0, 0.20, 1 / 12],
        bspremium[-1, 100, 100, 0.05, 0, 0.20, 1 / 12]}
```

```
Out[2]= {2.51207, 2.09627}
```

2.512 is the price of the call, not of the put ...

Paths

```
In[3]:= Timing[paths = MCPaths[50 000, 84, 100, 0.05, 0.20, 1 / 12];]
```

```
Out[3]= {108.634, Null}
```

```
In[4]:= Dimensions[paths]
```

```
Out[4]= {50 000, 85}
```

```
In[5]:= filter8421 = Table[If[Mod[j - 1, 4] == 0, True, False], {j, 1, 85}];
```

? Pick

Pick[list, sel] picks out those elements of list for which the corresponding element of sel is True.

Pick[list, sel, patt] picks out those elements of list for which the corresponding element of sel matches patt. >>

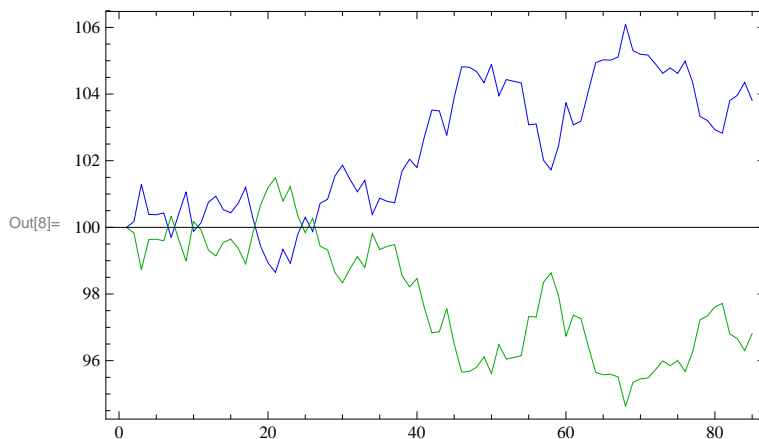
```
In[6]:= paths[[1]]
```

```
Out[6]= {100, 100.175, 101.295, 100.388, 100.379, 100.434, 99.697, 100.396, 101.068, 99.8813,
 100.13, 100.753, 100.935, 100.533, 100.437, 100.72, 101.203, 100.246, 99.4415, 98.9356,
 98.6497, 99.3412, 98.9211, 99.7916, 100.303, 99.8789, 100.722, 100.844, 101.546,
 101.868, 101.442, 101.068, 101.409, 100.386, 100.875, 100.781, 100.738, 101.693,
 102.044, 101.796, 102.721, 103.519, 103.496, 102.759, 103.901, 104.818, 104.808,
 104.677, 104.341, 104.895, 103.952, 104.435, 104.383, 104.333, 103.077, 103.102,
 102.01, 101.726, 102.435, 103.743, 103.076, 103.186, 104.079, 104.942, 105.027,
 105.017, 105.11, 106.099, 105.302, 105.194, 105.17, 104.909, 104.622, 104.785, 104.62,
 104.987, 104.36, 103.329, 103.205, 102.928, 102.825, 103.806, 103.96, 104.351, 103.81}
```

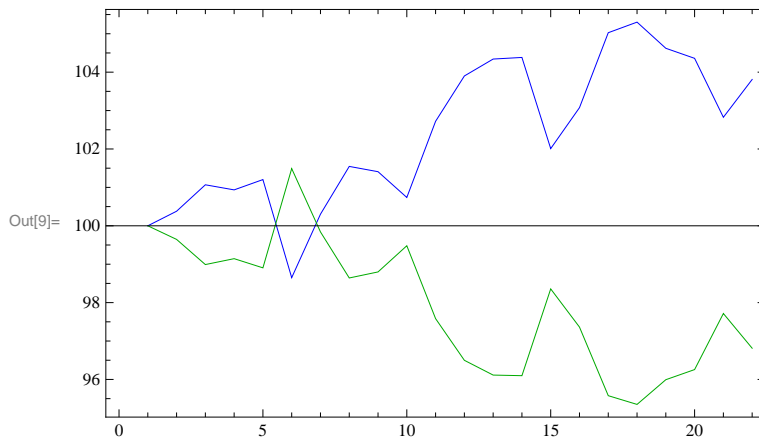
```
In[7]:= Pick[paths[[1]], filter8421]
```

```
Out[7]= {100, 100.379, 101.068, 100.935, 101.203, 98.6497, 100.303,
 101.546, 101.409, 100.738, 102.721, 103.901, 104.341, 104.383,
 102.01, 103.076, 105.027, 105.302, 104.622, 104.36, 102.825, 103.81}
```

```
In[8]:= ListLinePlot[{paths[[1]], paths[[2]]}, PlotRange -> All,
  Frame -> True, AxesOrigin -> {0, 100}, PlotStyle -> {Blue, Darker[Green]}]
```



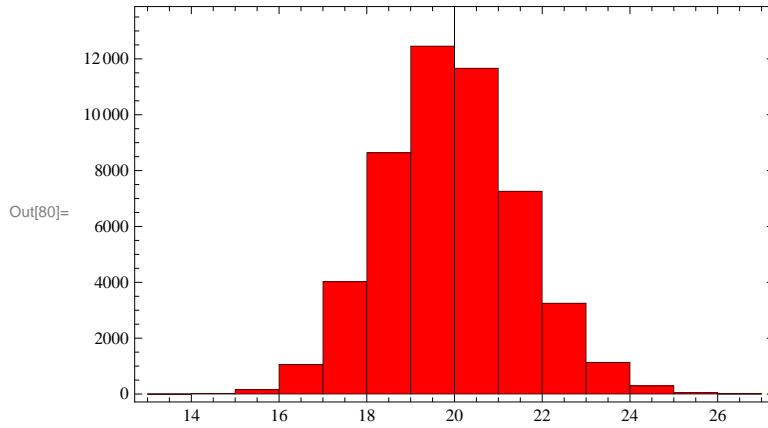
```
In[9]:= ListLinePlot[{Pick[paths[[1]], filter8421], Pick[paths[[2]], filter8421]}, PlotRange -> All,
  Frame -> True, AxesOrigin -> {0, 100}, PlotStyle -> {Blue, Darker[Green]}]
```



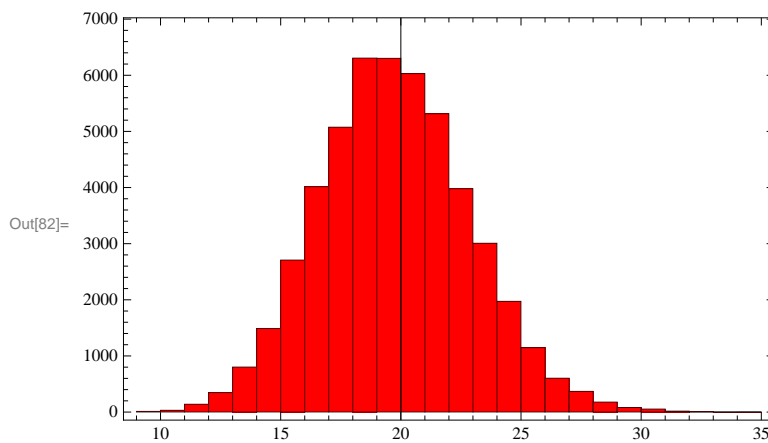
```
In[78]:= realizedvols84 =
  100 Sqrt[252 × 4] Map[StandardDeviation[(Rest[#] / Most[#] - 1)] &, paths];
```

```
In[81]:= realizedvols21 = 100 Sqrt[252] Map[
  StandardDeviation[(Rest[#] / Most[#] - 1)] &, Map[Pick[#, filter8421] &, paths]];
```

```
In[80]:= Histogram[realizedvols84, {1}, PlotRange → All,
  Frame → True, AxesOrigin → {20, 0}, ChartStyle → Red]
```



```
In[82]:= Histogram[realizedvols21, {1}, PlotRange → All,
  Frame → True, AxesOrigin → {20, 0}, ChartStyle → Red]
```



Hedging and P&L

```
In[11]:= tarray84 = Range[21, 0, -1 / 4] / 252;
  tarray21 = Range[21, 0, -1] / 252;
```

? MapThread

MapThread[f, {{a₁, a₂, E}, {b₁, b₂, E}, E} gives {f[a₁, b₁, E], f[a₂, b₂, E], E}.

MapThread[f, {expr₁, expr₂, E}, n] applies f to the parts of the expr_i at level n. >>

```
In[13]:= premiavol[series_, times_, ϕ_, K_, r_, q_, σ_] :=
  MapThread[bspremium[ϕ, #1, K, r, q, σ, #2] &, {series, times}];
deltasvol[series_, times_, ϕ_, K_, r_, q_, σ_] :=
  MapThread[bsdelta[ϕ, #1, K, r, q, σ, #2] &, {series, times}];
gammaevol[series_, times_, ϕ_, K_, r_, q_, σ_] :=
  MapThread[bsgamma[ϕ, #1, K, r, q, σ, #2] &, {series, times}];
```

```

In[16]:= Pick[paths[[1]], filter8421]

Out[16]= {100, 100.379, 101.068, 100.935, 101.203, 98.6497, 100.303,
  101.546, 101.409, 100.738, 102.721, 103.901, 104.341, 104.383,
  102.01, 103.076, 105.027, 105.302, 104.622, 104.36, 102.825, 103.81}

In[17]:= premiavol[Pick[paths[[1]], filter8421], tarray21, 1, 100, 0.05, 0, 0.20]

Out[17]= {2.51207, 2.65619, 2.99512, 2.84594, 2.94103, 1.5193, 2.26115,
  2.94226, 2.77543, 2.27641, 3.54866, 4.43038, 4.74897, 4.72639,
  2.68119, 3.44565, 5.16983, 5.39929, 4.69564, 4.40436, 2.85106, 3.80992}

In[18]:= Exp[0.05 tarray21]

Out[18]= {1.00418, 1.00398, 1.00378, 1.00358, 1.00338, 1.00318, 1.00298,
  1.00278, 1.00258, 1.00238, 1.00218, 1.00199, 1.00179, 1.00159,
  1.00139, 1.00119, 1.00099, 1.00079, 1.0006, 1.0004, 1.0002, 1.}

In[24]:= ? Accumulate

```

Accumulate[list] gives a list of the successive accumulated totals of elements in list. >>

```

In[53]:= plseries[series_, times_,  $\phi$ _, K_, r_, q_,  $\sigma$ _] :=
Module[{dtR, varspotR, premiaR, deltasR, assetR, cfwpremiaR,
  cfwdividR, cfwintR, cfwallR, cashaccR, portfolioR},
  dtR = Most[times] - Rest[times];
  varspotR = Rest[series] - Most[series];
  premiaR = premiavol[series, times,  $\phi$ , K, r, q,  $\sigma$ ];
  deltasR = deltasvol[series, times,  $\phi$ , K, r, q,  $\sigma$ ];
  assetR = -Append[Most[deltasR], 0] series;
  cfwpremiaR = Prepend[ConstantArray[0, Length[series] - 1], -1] premiaR;
  cfwdividR = (Append[Most[deltasR], 0] - Prepend[Most[deltasR], 0]) series;
  cfwintR = Prepend[Most[deltasR series] (Exp[q dtR] - 1), 0];
  cfwallR = cfwpremiaR + cfwdividR + cfwintR;
  cashaccR = Accumulate[cfwallR];
  portfolioR = premiaR + assetR + cashaccR;
  (* {series, dtR, varspotR, premiaR, deltasR, assetR, cfwpremiaR,
    cfwdividR, Accumulate[Most[cfwpremiaR+cfwdividR+cfwintR]],
    cfwintR, cfwallR, cashaccR, portfolioR} *)
  portfolioR
];

```

```

In[23]:= netplseries[series_, times_,  $\phi$ _, K_, r_, q_,  $\sigma$ _] :=
  Last[plseries[series, times,  $\phi$ , K, r, q,  $\sigma$ ]];

```

```
In[83]:= ? FindRoot
```

FindRoot[f, {x, x₀}] searches for a numerical root of f, starting from the point x = x₀.
 FindRoot[lhs == rhs, {x, x₀}] searches for a numerical solution to the equation lhs == rhs.
 FindRoot[{f₁, f₂, \dot{E} }, {{x, x₀}, {y, y₀}, \dot{E} }] searches for a simultaneous numerical root of all the f_i.
 FindRoot[{eqn₁, eqn₂, \dot{E} }, {{x, x₀}, {y, y₀}, \dot{E} }] searches for a numerical solution to the simultaneous equations eqn_i. >>

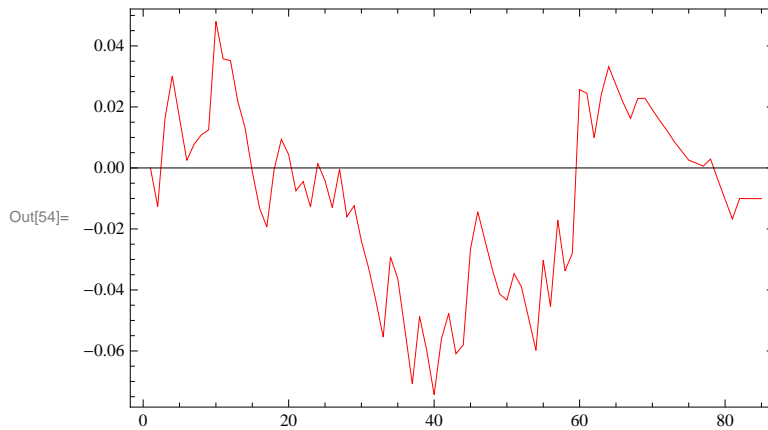
```

In[25]:= breakevenvol[series_, times_,  $\phi$ _, K_, r_, q_,  $\sigma$ 0_] :=
  FindRoot[netplseries[series, times,  $\phi$ , K, r, q,  $\sigma$ ], { $\sigma$ ,  $\sigma$ 0}][[1, 2]];

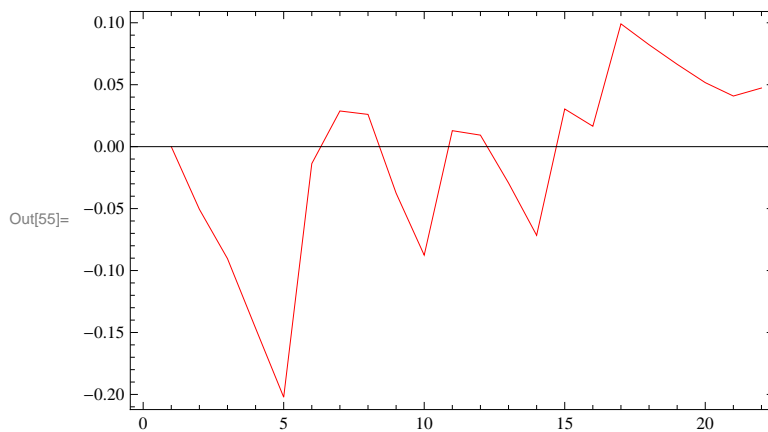
```



```
In[54]:= ListLinePlot[plseries[paths[[1]], tarray84, 1, 100, 0.05, 0, 0.20],
  PlotRange -> All, Frame -> True, AxesOrigin -> {0, 0}, PlotStyle -> {Red}]
```



```
In[55]:= ListLinePlot[plseries[Pick[paths[[1]], filter8421], tarray21, 1, 100, 0.05, 0, 0.20],
  PlotRange -> All, Frame -> True, AxesOrigin -> {0, 0}, PlotStyle -> {Red}]
```



```
In[62]:= Timing[netpls84 = Map[netplseries[#, tarray84, 1, 100, 0.05, 0, 0.20] &, paths];]
```

Out[62]= {740.29, Null}

```
In[63]:= Mean[netpls84]
```

Out[63]= -0.00313834

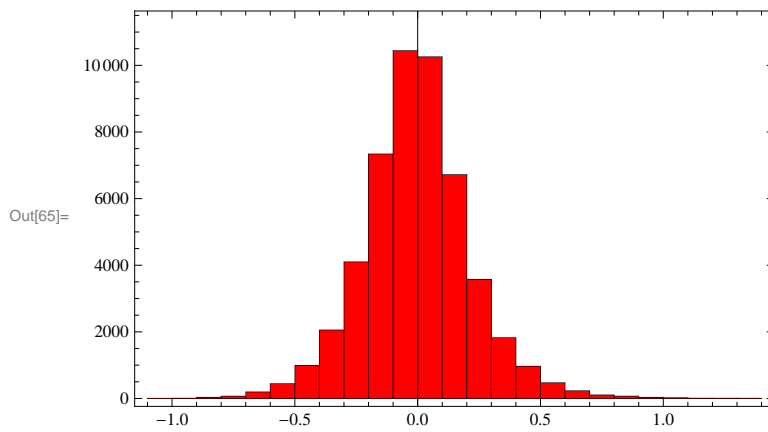
```
In[56]:= Timing[netpls21 = Map[netplseries[#, tarray21, 1, 100, 0.05, 0, 0.20] &,
  Map[Pick[#, filter8421] &, paths]];
```

Out[56]= {179.227, Null}

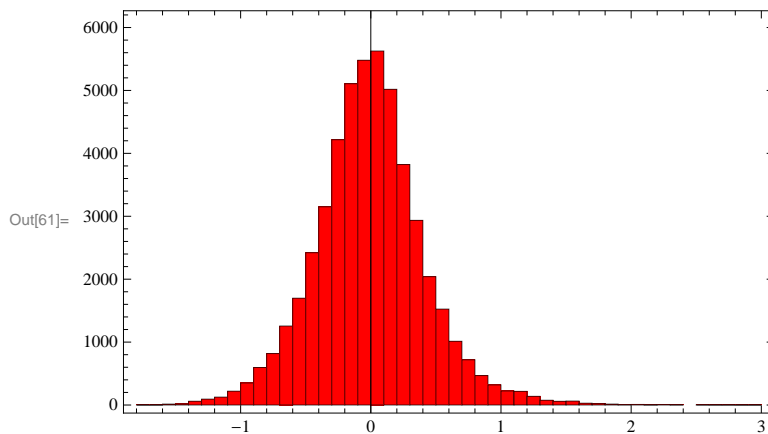
```
In[57]:= Mean[netpls21]
```

Out[57]= -0.0061724

```
In[65]:= Histogram[netpls84, {0.1}, PlotRange → All,
  Frame → True, AxesOrigin → {0, 0}, ChartStyle → Red]
```



```
In[61]:= Histogram[netpls21, {0.1}, PlotRange → All,
  Frame → True, AxesOrigin → {0, 0}, ChartStyle → Red]
```



```
In[68]:= tbl1 = {"Number of trades", "Mean P&L",
  "Standard Dev. of P&L", "StDev of P&L as a % of option premium"},
  {21, Round[Mean[netpls21], 0.001], Round[StandardDeviation[netpls21], 0.01],
    100 Round[StandardDeviation[netpls21] / bspremium[1, 100, 100, 0.05, 0, 0.20, 1 / 12],
      0.001]}, {84, Round[Mean[netpls84], 0.001],
    Round[StandardDeviation[netpls84], 0.01], 100 Round[StandardDeviation[netpls84] /
      bspremium[1, 100, 100, 0.05, 0, 0.20, 1 / 12], 0.001]}};
```

```
In[69]:= Grid[tbl1, Frame → All, Alignment → Center]
```

Out[69]=

Number of trades	Mean P&L	Standard Dev. of P&L	StDev of P&L as a % of option premium
21	-0.006	0.42	16.7
84	-0.003	0.22	8.7