5.4 The local volatility model

Stochastic volatility models take the view that there is an extra Brownian motion that is responsible for volatility changes. This extra source of randomness creates a market that is incomplete, where options are not redundant securities. Practically, this means that in order to hedge a position one needs to hedge against volatility risk as well as market risk. Local volatility models take a completely different view. No extra source of randomness is introduced, and the markets remain complete. In order to account for the implied volatility skew there is a nonlinear (but deterministic) volatility structure

$$\mathrm{d}S_t = rS_t\mathrm{d}t + \sigma(t, S_t)S_t\mathrm{d}B_t$$

As vanilla options are expressed via the risk neutral expectation of the random variable S_T, local volatility models attempt to construct the function $\sigma(t, S)$ that is consistent with the implied risk neutral densities for different maturities. The methodology of local volatility models follows the one on implied risk neutral densities, originating in the pioneering work of Breeden and Litzenberger (1978).

These methods are inherently nonparametric, and rely on a large number of option contracts that span different strikes and maturities. In reality there is only a relatively small set of observed option prices that is traded, and for that reason some *interpolation* or *smoothing* techniques must be employed to artificially reconstruct the true pricing function or the volatility surface. Of course this implies that the results will be sensitive to the particular method that is used. Also, care has to be taken to ensure that the resulting prices are arbitrage free.

5.4.1 Interpolation methods

There are many interpolation methods that one can use on the implied volatility surface. As second order derivatives of the corresponding pricing function are required, it is paramount that the surface is sufficiently smooth. In fact, it is common practice to sacrifice the perfect fit in order to ensure smoothness, which suggests that we are actually implementing an implied volatility *smoother* rather than an *interpolator*. Within this obvious tradeoff we have to selecting the degree of fit versus smoothness, which is more of an art than a science.

One popular approach is to use a family of known functions, and reconstruct the volatility surface as a weighted sum of them. As an example we can use the *radial basis function* (RBF) interpolation, where we reconstruct an unknown function using the form

$$f(\boldsymbol{x}) = c_0 + \boldsymbol{c}'\boldsymbol{x} + \sum_{n=1}^{N} \lambda_n \phi\left(||\boldsymbol{x} - \boldsymbol{x}_n||\right)$$

Listing 5.7. `nadwat2.m`: Nadaraya-Watson smoother.

```
%nadwat2.m
function zi = nadwat2(x, y, z, w, xi, yi, hx, hy)
N = length(x);
if isempty(w)
    w = ones(N, 1);
end
zi1 = 0;
zi2 = 0;
for j = 1:N
    xe  = exp(-0.5/hx^2*(xi-x(j)).^2)/sqrt(2*pi)/hx;
    ye  = exp(-0.5/hy^2*(yi-y(j)).^2)/sqrt(2*pi)/hy;
    zi1 = zi1 + z(j)*w(j)*xe.*ye;
    zi2 = zi2 + w(j)*xe.*ye;
end
zi  = zi1./zi2;
```

The points that we observe are given at the nodes $\boldsymbol{x}_n$, for $n = 1, \ldots, N$. The radial function $\phi(x)$ will determine how the impact of the value at each node behaves. Common radial functions include the Gaussian $\phi(x) = \exp\left(-x^2/(2\sigma^2)\right)$ and the multiquadratic function $\phi(x) = \sqrt{1 + (x/\sigma)^2}$, among others.[22] The values of the parameters $c_0, \boldsymbol{c}$ and λ_n are determined using the observed value function at the nodes $\boldsymbol{x}_n$ and the required degree of smoothness. Figure 5.9(a) presents a set of observed implied volatilities together with the smoothed surface constructed using the RBF interpolation method. The implementation is given in listing 5.8.

The Nadaraya-Watson (NW) smoother is another popular choice. Here the approximating function takes the form

$$f(\boldsymbol{x}) = \frac{\sum_{n=1}^{N} w_n y_n \exp(-\boldsymbol{x}'\mathbf{H}\boldsymbol{x})}{\sum_{n=1}^{N} w_n \exp(-\boldsymbol{x}'\mathbf{H}\boldsymbol{x})}$$

where y_n is the observed value at the point $\boldsymbol{x}_n$, and the matrix $\mathbf{H} = \mathrm{diag}(h_1, \ldots, h_n)$ is user defined. This is implemented for the two-dimensional case in listing 5.7. Figure 5.9(b) gives the implied volatility surface smoothed using the Nadaraya-Watson method.

Of course the smoothed or interpolated volatility surface can be mapped to call and put prices using the Black-Scholes formula. There is also a number of restrictions that one needs to take into account when constructing the volatility surface. In particular, it is important to verify that the resulting

[22] The parameter σ is user defined. In Matlab the RBF interpolation is implemented in the package of Alex Chirokov that can be download at `http://www.mathworks.com/matlabcentral/`.

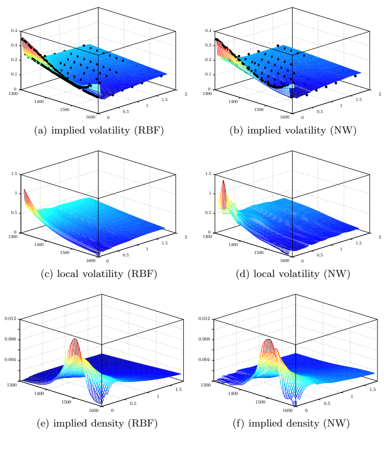
Listing 5.8. `imp_vol.m`: Implied volatility surface smoothing.

```
%imp_vol.m
data = xlsread('SPX_24_04_07_a.xls');
CP = data(:,1); T = data(:,2)/365; K = data(:,3);
P = data(:,4); S0 = data(:,5); r = data(:,6)/100;
[IV, IVi] = bs_iv(S0, K, r, T, CP, P);
[Tu, Iu] = unique(T);
% create output grids
dK = 5; dT = 0.02;
Ko = (1300:dK:1600)';
To = (min(0.90*T):dT:max(1.10*T))';
NKo = length(Ko); NTo = length(To);
[Kgo, Tgo] = meshgrid(Ko, To);
Kvo = reshape(Kgo, [NKo*NTo, 1]); % vectorize
Tvo = reshape(Tgo, [NKo*NTo, 1]);
Tmo = log(Tvo);                    % transform
Kmo = log(S0(1)./Kvo)./sqrt(Tvo);
% prepare actual data
Tm = log(T);                       % transform
Km = log(S0(1)./K)./sqrt(T);

% Nadaraya-Watson smoother
% IVvo = nadwat2(Km, Tm, IV, [], Kmo, Tmo, 0.05, 0.10);
% Radial Basis Function smoother
coef = rbfcreate([Km'; Tm'], IV',...
    'RBFFunction', 'multiquadric', 'RBFSmooth', 0.25);
IVvo = rbfinterp([Kmo'; Tmo'], coef)';

% interpolate risk free rate for output
rvo = interp1(Tu, r(Iu), Tvo, 'linear', 'extrap');
% compute prices and reshape to matrices
Pvo = bs_greeks(S0(1), Kvo, rvo, IVvo, Tvo, 1);
IVo = reshape(IVvo,[NTo, NKo]);
Po  = reshape(Pvo, [NTo, NKo]);
ro  = reshape(rvo, [NTo, NKo]);
```

prices do not permit arbitrage opportunities. As shown in Carr and Madan (2005) it is straightforward to rule out static arbitrage by checking the prices of simple vertical spreads, butterflies and calendar spreads. More precisely, having constructed a grid of call prices for different strikes $0 = K_0, K_1, K_2, \ldots$ and maturities $0 = T_0, T_1, T_2, \ldots$, with $C_{i,j} = f_{BS}\big(t, S; K_i, T_j, r, \hat{\sigma}(K_i, T_j)\big)$, we need to construct the following quantities

1. Vertical spreads $VS_{i,j} = \frac{C_{i-1,j} - C_{i,j}}{K_i - K_{i-1}}$. There should be $0 \leqslant VS_{i,j} \leqslant 1$ for all $i, j = 0, 1, \ldots$

Fig. 5.9. Implied volatilities smoothed with the radial basis function (RBF, left) and the Nadaraya-Watson (NW, right) methods. The corresponding local volatility surfaces and the implied probability density functions for different horizons are also presented.



(a) implied volatility (RBF) (b) implied volatility (NW)

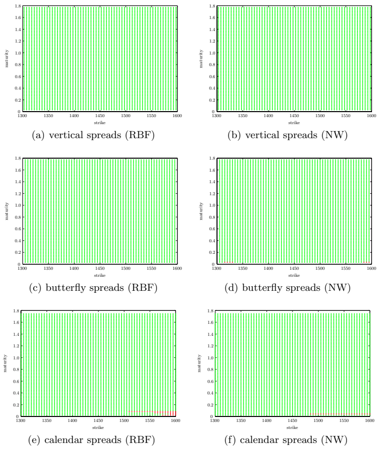
(c) local volatility (RBF) (d) local volatility (NW)

(e) implied density (RBF) (f) implied density (NW)

2. Butterfly spreads $BS_{i,j} = C_{i-1,j} - \frac{K_{i+1} - K_{i-1}}{K_{i+1} - K_i} C_{i,j} + \frac{K_i - K_{i-1}}{K_{i+1} - K_i} C_{i+1,j}$. There should be $BS_{i,j} \geqslant 0$ for all $i, j = 0, 1, \ldots$
3. Calendar spreads $CS_{i,j} = C_{i,j+1} - C_{i,j}$. There should be $CS_{i,j} \geqslant 0$ for all $i, j = 0, 1, \ldots$

In figure 5.10 we construct these tests for the resulting volatility surfaces based on the two smoothing methods, implemented in listing 5.9. With green dots we denote the points where no arbitrage opportunities exist, while red

Fig. 5.10. Static arbitrage tests for the smoothed implied volatility functions of figure 5.9. Vertical, butterfly and calendar spreads are constructed and their prices are examined. Green dotes represent spreads that have admissible prices, while red dots indicate spreads that offer arbitrage opportunities as they are violating the corresponding bounds.



(a) vertical spreads (RBF) (b) vertical spreads (NW)

(c) butterfly spreads (RBF) (d) butterfly spreads (NW)

(e) calendar spreads (RBF) (f) calendar spreads (NW)

dots represent arbitrage opportunities. Both RBF and NW methods yield prices that pass the vertical spread tests. The NW smoother produces a very small number of very short away-from-the-money prices that allow the setup of butterfly spreads with negative value. Both methods fail the calendar spread test for far-out-of-the-money calls with very short maturities. Nevertheless,

Listing 5.9. `test_vol.m`: Tests for static arbitrage.

```
%test_vol.m
imp_vol; % load data and smooth volatility surface
% vertical spreads
VS = (Po(:,1:end-1)-Po(:,2:end))/dK;
VS = (VS>=0)&(VS<=1);
% butterfly spreads
BS = Po(:,3:end)-2*Po(:,2:end-1)+Po(:,1:end-2);
BS = (BS>=0);
% calendar spreads
CS = Po(2:end,:)-Po(1:end-1,:);
CS = (CS>=0);
```

the bid-ask spreads in these areas are wide enough to ensure that these opportunities are not actually exploitable. Overall the results are very good, but if needed one can incorporate these tests within the fitting procedures, and thus find smoothed volatility surfaces that by construction pass all three arbitrage tests.

Another important feature of the implied volatility is that it should behave in a linear fashion for extreme log-strikes (Lee, 2004a; Gatheral, 2004). This indicates that it makes sense to extrapolate the implied volatility linearly to extend outside the region of observed prices.

Apart from these nonparametric methods one can set up parametric curves to fit the implied volatility skew. A parametric form might be less accurate, but it can offer more a robust fit where the resulting prices are by construction free of arbitrage. Gatheral (2004) proposes an implied variance function for each maturity horizon, coined the *stochastic volatility inspired* (SVI) parameterization, of the form

$$v(k;\alpha,\beta,\sigma,\rho,\mu) = \alpha + \beta\left(\rho(k-\mu) + \sqrt{(k-\mu)^2+\sigma^2}\right)$$

where $k = \log(K/F)$. This form always remains positive and ensures that it grows in a linear fashion for extreme log-strikes. In particular Gatheral (2004) shows that α controls for the variance level, β controls the angle between the two asymptotes, σ controls the smoothness around the turning point, ρ controls the orientation of the skew, and μ shifts the skew across the moneyness level.

5.4.2 Implied densities

Based on the implied volatility function $\hat{\sigma}(T,K)$ the empirical pricing function is easily determined via the Black-Scholes formula

Listing 5.10. `loc_vol.m`: Construction of implied densities and the local volatility surface.

```
%loc_vol.m
imp_vol; % load data and smooth volatility surface
% 1st deriv wrt T
D1T = (Po(3:end,:)-Po(1:end-2,:))/(2*dT);
% 1st deriv wrt K
D1K = (Po(:,3:end)-Po(:,1:end-2))/(2*dK);
% 2nd deriv wrt K
D2K = (Po(:,3:end)-2*Po(:,2:end-1)+Po(:,1:end-2))/dK^2;
% implied probability density function
F   = exp(-ro(:,2:end-1).*Tgo(:,2:end-1)).*D2K;
% local volatility function
LV = Po(2:end-1,2:end-1)-...
        Kgo(2:end-1,2:end-1).*D1K(2:end-1,:);
LV = D1T(:,2:end-1) + ro(2:end-1,2:end-1).*LV;
LV = LV./(0.5*Kgo(2:end-1,2:end-1).^2.*D2K(2:end-1,:));
```

$$P(T,K) = f_{BS}\big(t,S_0;T,K,r,\hat{\sigma}(T,K)\big)$$

It has been recognized, since Breeden and Litzenberger (1978), that the empirical pricing function can reveal information on the risk neutral probability density that is *implied* by the market. In particular, if $\mathbb{Q}_t(S)$ is this risk neutral probability measure of the underlying asset with horizon t, then the call price can be written as the expectation

$$P(T,K) = \exp(-rT)\int_K^\infty (S-K)\mathrm{d}\mathbb{Q}_T(S)$$

If we differentiate twice with respect to the strike price, using the Leibniz rule

$$\frac{\partial}{\partial t}\int_{\alpha(t)}^{\beta(t)} g(x,t)\mathrm{d}x = g\big(\beta(t),t\big)\frac{\mathrm{d}\beta(t)}{\mathrm{d}t} - g\big(\alpha(t),t\big)\frac{\mathrm{d}\alpha(t)}{\mathrm{d}t} + \int_{\alpha(t)}^{\beta(t)}\left[\frac{\partial}{\partial t}g(x,t)\right]\mathrm{d}x$$

we obtain the Breeden and Litzenberger (1978) expression for the implied probability density function

$$\mathrm{d}\mathbb{Q}_T(S) = \exp(rT)\left.\frac{\partial^2 P(T,K)}{\partial K^2}\right|_{K=S} \tag{5.1}$$

It is easy to compute this derivative numerically, and therefore approximate the implied density using central differences. In particular

$$\mathrm{d}\mathbb{Q}_T(S) \approx \exp(rT)\frac{P(T,S-\triangle K) - 2P(T,S) + P(T,S+\triangle K)}{(\triangle K)^2}$$

One can recognize that the above expression is the price of $1/(\triangle K)^2$ units of a very tight butterfly spread around S, like the one used in the static arbitrage tests above. The relation between the butterfly spread and the risk neutral probability density is well known amongst practitioners, and can be used to isolate the exposure to specific ranges of the underlying. We carry out this approximation in listing 5.10, and the resulting densities are presented in figures 5.9(e,f).

5.4.3 Local volatilities

A natural question that follows is whether or not a process exists that is consistent with the sequence of implied risk neutral densities. After all, Kolmogorov's extension theorem 1.3.2 postulates that given a collection of transition densities such a process might exist. Dupire (1994) recognized that one might be able to find a *diffusion* which is consistent with the observed option prices, constructing the so called *local volatility model*, where the return volatility is a deterministic function of time and the underlying asset. In a series of papers Derman and Kani (1994), Derman et al. (1996), and Derman et al. (1996) outline the use of the local volatility function for pricing and hedging, while Barle and Cakici (1998) present a method of constructing an implied trinomial tree that is consistent with observed option prices.

The dynamics of the underlying asset (under the risk neutral measure) are given by

$$\mathrm{d}S_t = rS_t\mathrm{d}t + \sigma(t,S_t)S_t\mathrm{d}B_t \tag{5.2}$$

The popularity of the local volatility approach stems from the fact that the the steps taken in the derivation of the Black-Scholes PDE can be replicated, since the *local volatility function* $\sigma(t,S)$ is deterministic. In particular, the markets remain complete as there is only one source of uncertainty that can be hedged out using the underlying asset and the risk free bank account.

The pricing function for any derivative under the local volatility dynamics will therefore satisfy a PDE that resembles the Black-Scholes one

$$\frac{\partial}{\partial t}f(t,S) + rS\frac{\partial}{\partial S}f(t,S) + \frac{1}{2}\sigma^2(t,S)S^2\frac{\partial^2}{\partial S^2}f(t,S) = rf(t,S)$$

Of course, having a functional form for the volatility will mean that closed form expressions are unattainable even for plain vanilla contracts. Nevertheless, it is straightforward to modify the finite difference methods that we outlined in chapter 4 (for example the θ-method in listing 3.3) to account for the local volatility structure.

Dupire (1993) notes that if the diffusion (5.2) is consistent with the risk neutral densities (5.1), then the risk neutral densities must satisfy the forward Kolmogorov equation (see section 1.6.1). In particular, if we denote the transition density with with $f^{\mathbb{Q}}(t,K;T,S) = \mathbb{Q}(S_T \in \mathrm{d}S|S_t = K)$, then the forward Kolmogorov equation will take the form