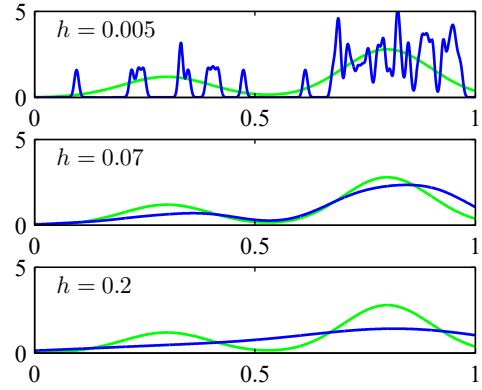


**Figure 3.14** Illustration of the kernel density model (3.184) applied to the same data set used to demonstrate the histogram approach in Figure 3.13. We see that $h$ acts as a smoothing parameter and that if it is set too small (top panel), the result is a very noisy density model, whereas if it is set too large (bottom panel), then the bimodal nature of the underlying distribution from which the data is generated (shown by the green curve) is washed out. The best density model is obtained for some intermediate value of $h$ (middle panel).

density model if we choose a smoother kernel function, and a common choice is the Gaussian, which gives rise to the following kernel density model:

$$p(\mathbf{x}) = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{(2\pi h^2)^{D/2}}\exp\left\{-\frac{\|\mathbf{x}-\mathbf{x}_n\|^2}{2h^2}\right\} \tag{3.184}$$

where $h$ represents the standard deviation of the Gaussian components. Thus, our density model is obtained by placing a Gaussian over each data point, adding up the contributions over the whole data set, and then dividing by $N$ so that the density is correctly normalized. In Figure 3.14, we apply the model (3.184) to the data set used earlier to demonstrate the histogram technique. We see that, as expected, the parameter $h$ plays the role of a smoothing parameter, and there is a trade-off between sensitivity to noise at small $h$ and over-smoothing at large $h$. Again, the optimization of $h$ is a problem in model complexity, analogous to the choice of bin width in histogram density estimation or the degree of the polynomial used in curve fitting.

We can choose any other kernel function $k(\mathbf{u})$ in (3.183) subject to the conditions

$$k(\mathbf{u}) \geqslant 0, \tag{3.185}$$

$$\int k(\mathbf{u})\,\mathrm{d}\mathbf{u} = 1, \tag{3.186}$$

which ensure that the resulting probability distribution is non-negative everywhere and integrates to one. The class of density model given by (3.183) is called a kernel density estimator or *Parzen* estimator. It has a great merit that there is no computation involved in the 'training' phase because this simply requires the training set to be stored. However, this is also one of its great weaknesses because the computational cost of evaluating the density grows linearly with the size of the data set.