

Chapter 2

Nonparametric Density Estimation

2.1 Introduction

This chapter describes the background material related to the nonparametric density estimation. Techniques such as histograms (together with its extension, known as ASH, see Sect. 2.3), Parzen windows and k -nearest neighbors are at the core of the applications of nonparametric density estimation. For that reason, we decided to include a chapter describing these for the sake of completeness and to allow less experienced readers develop their intuitions in terms of the nonparametric estimation.

Most of the material is presented taking into account only the univariate case; extending the results to cover more than one variable, however, is often a straightforward task.

The chapter is organized as follows: Sect. 2.2 presents a short overview of the fundamental concepts related to histograms. Section 2.3 is devoted to a description of a smart extension of certain well-known histograms aimed at avoiding some of their drawbacks. Section 2.4 presents basic concepts related to the nonparametric density estimation. Section 2.5 is devoted to the Parzen windows, while Sect. 2.6 to the k -nearest neighbors approach.

2.2 Density Estimation and Histograms

The well-known histogram is the simplest form of a nonparametric density estimation. The sample space is divided into disjoint categories, or *bins*, (note that the number of bins is expressed by a natural number) and the density is approximated by counting how many data points fall into each bin. Let B_l be the l -th bin and h be the width of the bins (all the bins have equal widths) and let $\#\{X_i \in B_l\}$ denote the total number of data points from X_1, X_2, \dots, X_n that fall into the corresponding bin B_l of width h . Then, the PDF estimator is

$$\hat{f}(x) = \frac{\#\{X_i \in B_l\}}{nh} = \frac{k}{nh}, \quad (2.1)$$

for every $x \in B_l$, with l being a natural number. The above uses a common convention, namely that $f(x)$ and $\hat{f}(x)$ represent the true (usually unknown) density and an estimator of the true density, respectively. Figure 2.1 shows a sample histogram generated for a toy univariate dataset of seven data points:

$$X_1 = 3.5, X_2 = 4.2, X_3 = 4.5, X_4 = 5.8, X_5 = 6.2, X_6 = 6.5, X_7 = 6.8. \quad (2.2)$$

In this example, $h = 1$ and the starting position of the first bin (also known as the *bin origin*) is $x_0 = 0$. Note also that bins B_1 and B_6 are empty.

As it can be easily seen, the histogram requires two parameters to be defined: the bin width h and the bin origin x_0 . While the histogram is a very simple form of the nonparametric density estimator, there are some serious drawbacks that are already noticeable. First, the final shape of the density estimate strongly depends on the starting position of the first bin. Second, the natural feature of the histogram

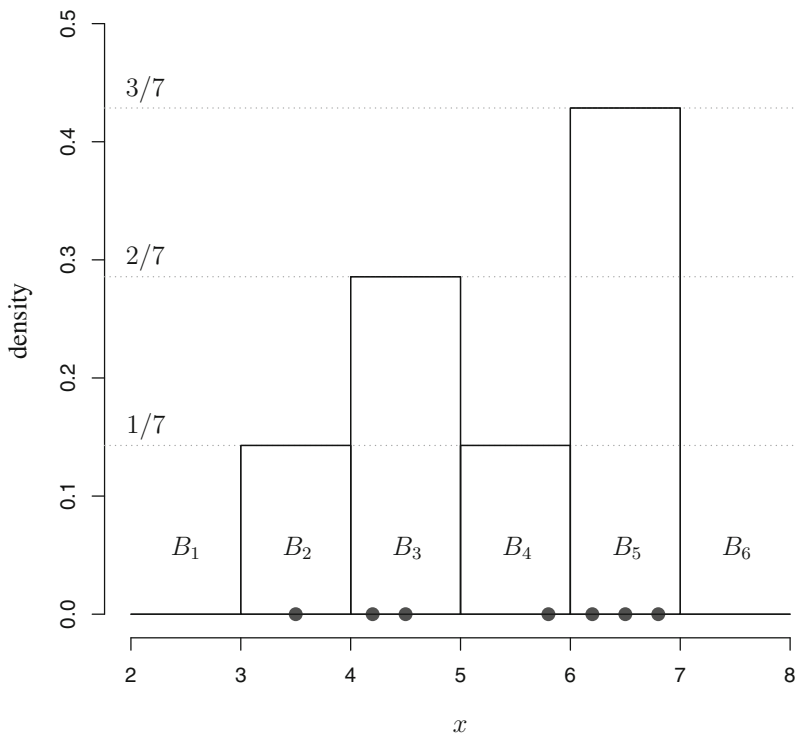


Fig. 2.1 A sample histogram for a toy univariate dataset of seven data points

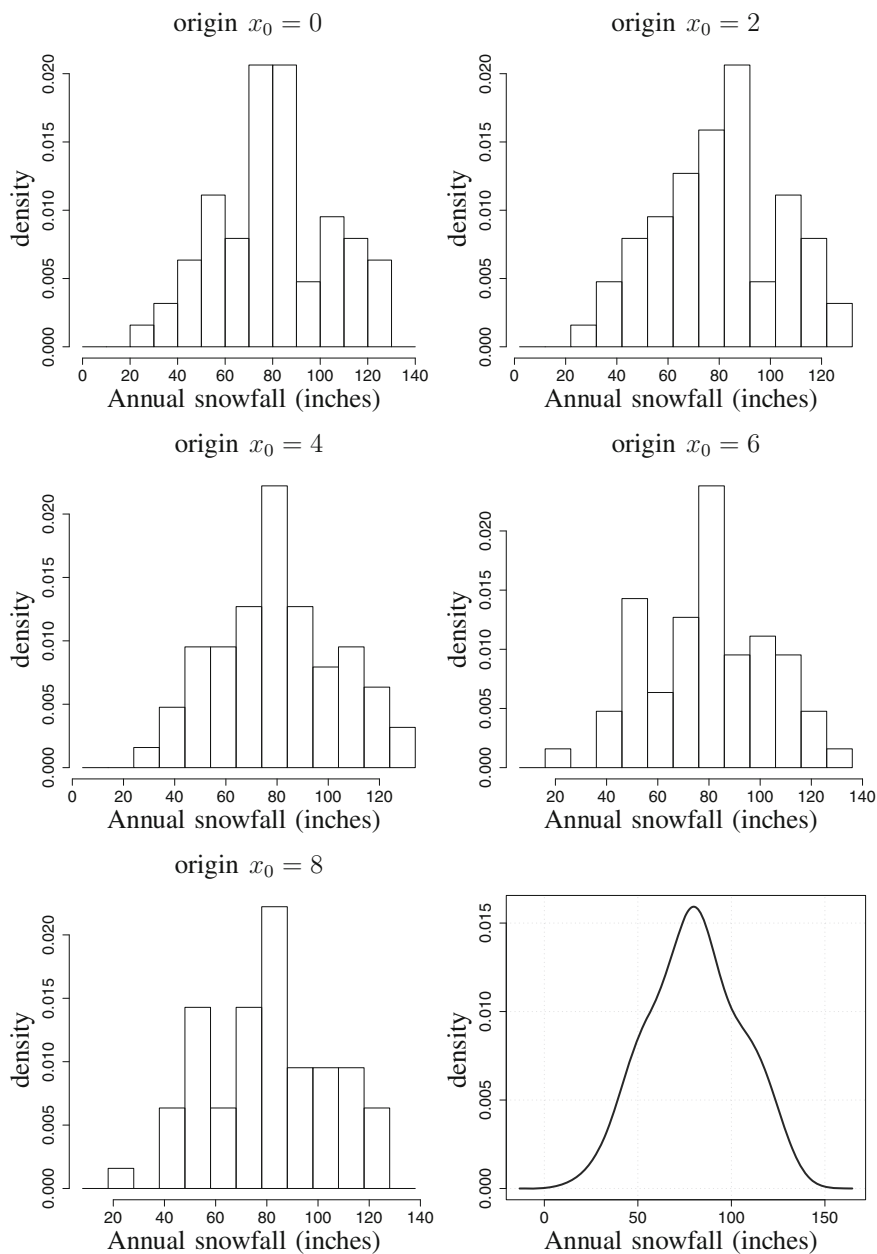


Fig. 2.2 Five histograms with different bin origins and constant bin width. The dataset is annual snowfall accumulations in Buffalo, NY from 1910 to 1973 (in inches)

is the presence of discontinuities of density. These are not, however, related to the underlying density and, instead, are only an artifact of the chosen bin locations. Third drawback is the so-called curse of dimensionality (see Sect. 3.9), which constitutes a much more serious problem, since the number of bins grows exponentially with the number of dimensions. In higher dimensions one would require a very large number of examples or else most of the bins would be empty (incidentally, the curse of dimensionality phenomena is a common problem for all the nonparametric techniques for density estimation). All these drawbacks make the histogram unsuitable for most practical applications except for rapid visualization of results in one or two dimensions (less often in three dimensions).

Figure 2.2 illustrates the phenomenon of the strong dependence of the histogram shape on the starting position of the first bin (here the *buffalo* dataset from the *gss* R package was used). To showcase this, the bin width remains constant ($h = 10$), while different origins are used ($x_0 = 0, 2, 4, 6, 8$). It is not obvious which histogram should be considered as the best one. All five histograms suggest a mode around the value of 80, but in some cases the existence of two or even three modes is not excluded. In the lower right corner the ‘true’ density is depicted, generated with a smooth nonparametric kernel density estimator. This shows that, in fact, only one mode is present in the input data.

2.3 Smoothing Histograms

As demonstrated in the previous section, the bin origin (sometimes called a *nuisance parameter*) has a significant influence on the final histogram shape. A smart extension of the classical histogram was presented in [156, 159], referred to as the *averaged shifted histogram* (ASH). ASH enjoys several advantages compared with the classical histogram: better visual interpretation, better approximation, and nearly the same computational efficiency as classical histograms. ASH provides a bridge between the classical histogram and advanced kernel-based methods presented in Chap. 3.

ASH algorithm avoids the pitfall of choosing an arbitrary value for the bin origin x_0 . It is a nonparametric density estimator that averages several classical histograms with different origins. A collection of m classical histograms, each with the bin width h , but with slightly different (or shifted) origin is constructed. Then, the average of these histograms is calculated and the ASH estimate at x is then defined as

$$\hat{f}_{ASH}(x) = \frac{1}{m} \sum_{i=1}^m \hat{f}_i(x). \quad (2.3)$$

Figure 2.3 shows the example ASH density estimates of a sample dataset generated from a mixture of three Gaussians given by

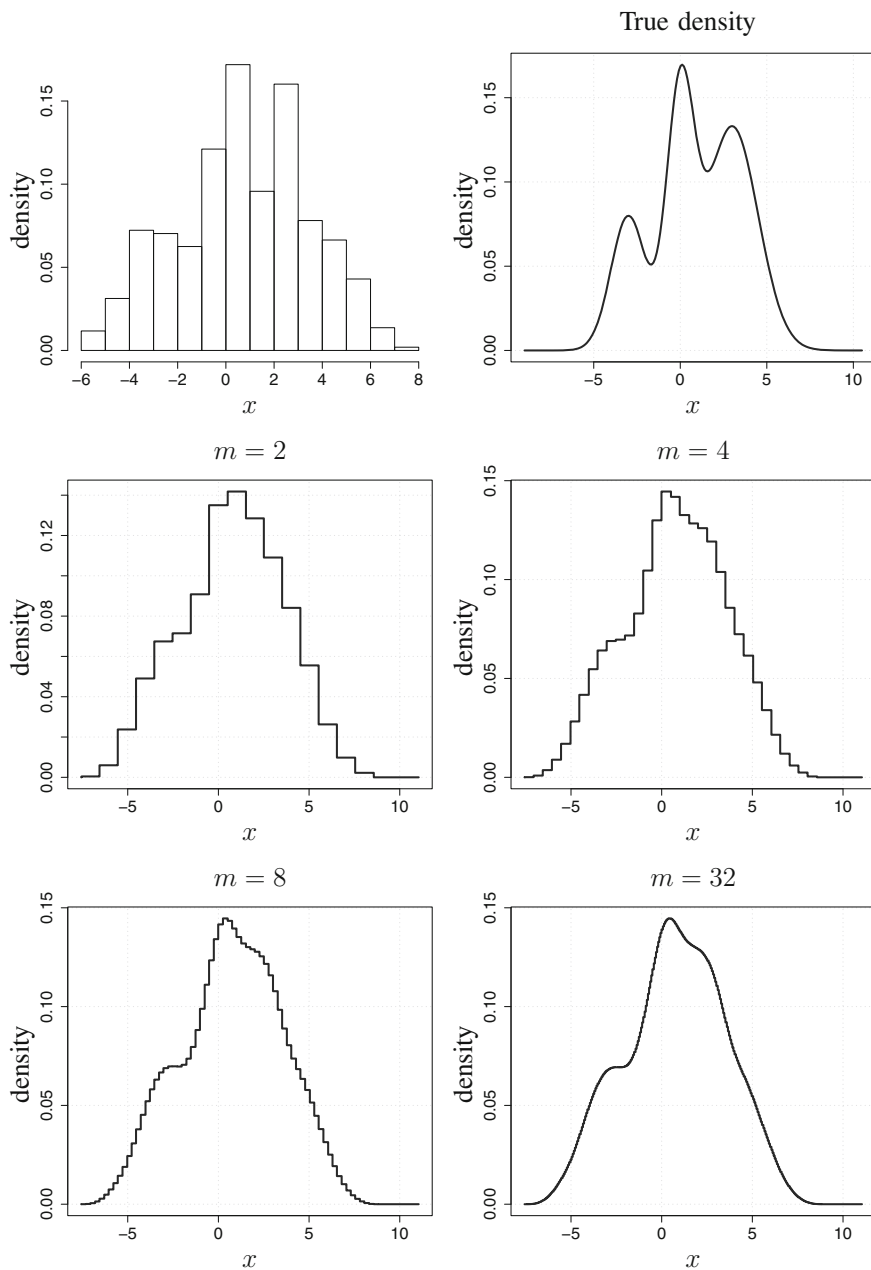


Fig. 2.3 ASH density estimates of a trimodal mixture of Gaussians with $m = \{2, 4, 8, 32\}$

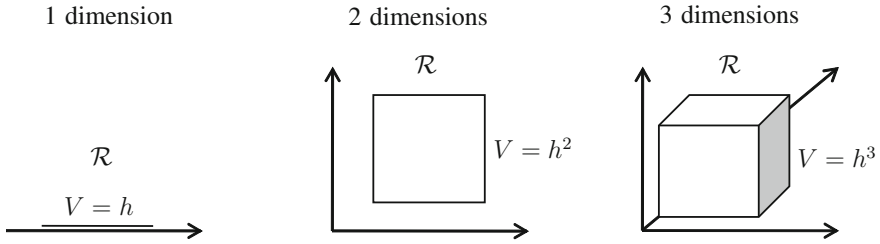


Fig. 2.4 Hypercubes in one, two and three dimensions

$$f(x) = \frac{2}{10}\mathcal{N}(x; -3, 1) + \frac{3}{10}\mathcal{N}(x; 0, 0.8) + \frac{5}{10}\mathcal{N}(x; 3, 1.5). \quad (2.4)$$

It can be easily observed that already for m at the level of several dozen, the resulting density is smooth and accurate enough with the trimodal nature of the data being evident.

2.4 A General Formulation of Nonparametric Density Estimation

This section aims at developing intuitions related to the nonparametric density estimation. Assume that n samples X_1, X_2, \dots, X_n are drawn independently and are identically distributed (*iid*) from a (usually unknown) density function p . The goal is to estimate this density at an arbitrarily chosen point x based on these n samples.

Now, let \mathcal{R} be a region around x . A region \mathcal{R} is considered to be a d -dimensional hypercube with side length h and volume $V = h^d$, as depicted in Fig. 2.4.

The probability that a training sample will fall in a region \mathcal{R} is

$$P = \Pr[x \in \mathcal{R}] = \int_{\mathcal{R}} p(x)dx. \quad (2.5)$$

It is well known that the probability that k of these n samples fall in \mathcal{R} is given by the binomial distribution

$$\Pr(k; n, p) = \Pr(X = k) = \binom{n}{k} P^k (1 - P)^{n-k}. \quad (2.6)$$

It is also known, given the properties of the binomial distribution, that the mean and the variance of the ratio k/n are described by the following equations

$$\begin{aligned} E\left(\frac{k}{n}\right) &= P, \\ \text{Var}\left(\frac{k}{n}\right) &= \frac{P(1-P)}{n}. \end{aligned} \quad (2.7)$$

Now, it should be clear that the variance approaches zero as $n \rightarrow \infty$. So, it can be expected that the mean fraction of points that fall within \mathcal{R} , that is

$$P \cong \frac{k}{n}, \quad (2.8)$$

would be a good estimate of the probability P . On the other hand, if it is assumed that \mathcal{R} is so small that the density $p(x)$ does not vary too much within it, then the integral in (2.5) can be approximated by

$$P = \int_{\mathcal{R}} p(x) dx \cong p(x)V, \quad (2.9)$$

where V is the volume of the region \mathcal{R} (for example the volume V being the width of the gray area in Fig. 1.1, is obviously too wide to satisfy the conditions of (2.9)). Finally, by merging (2.8) and (2.9) we obtain that

$$p(x) \cong \frac{k}{nV}. \quad (2.10)$$

Equation (2.10) converges in probability to the true density according to the following

$$f(x) = \lim_{n \rightarrow \infty} p(x) = \lim_{n \rightarrow \infty} \frac{k}{nV}. \quad (2.11)$$

The following conditions are required for convergence

$$\begin{aligned} \lim_{n \rightarrow \infty} V_n &= 0, \\ \lim_{n \rightarrow \infty} k_n &= \infty, \\ \lim_{n \rightarrow \infty} \frac{k_n}{n} &= 0. \end{aligned} \quad (2.12)$$

The estimate becomes more accurate as the number of samples n increases and the volume V shrinks. In practical applications, the value of n is always fixed (the input datasets are unchangeable). So, to improve the estimate of $f(x)$ one could set V to be sufficiently small. In that case however, the region \mathcal{R} can become so small that it would enclose no data. This means that in practice, a kind of compromise in terms of finding a proper value of V is needed. It should be large enough to include a sufficient number of samples within \mathcal{R} and, at the same time, be small enough to satisfy the

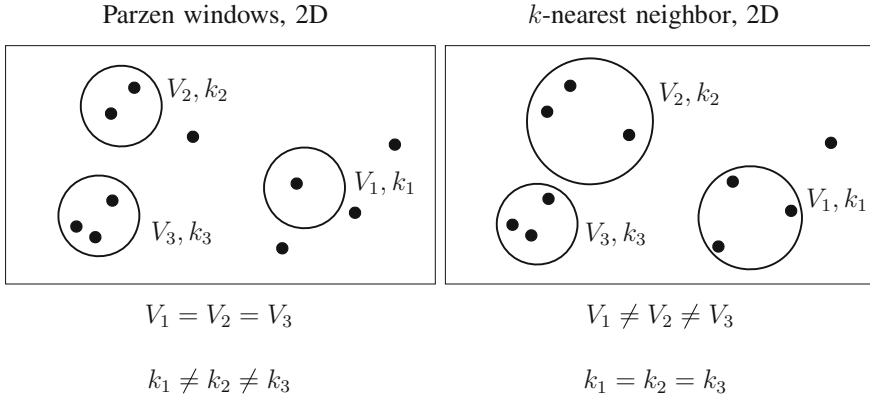


Fig. 2.5 A visualization of the idea of the Parzen windows and the k -nearest neighbors techniques

assumption of (2.9). This is a typical *trade-off dilemma* encountered when dealing with different knowledge domains.

Equation (2.10) can be regarded as a starting point for every nonparametric density estimation analysis. It is sometimes referred to as the *naive density estimator*. Obviously, if $V = h$, the Eq. (2.10) is the same as the one in the definition of the histogram (2.1).

In practical situations related to the use of density estimators two basic approaches can be distinguished:

- *Parzen windows* [130]: choosing a fixed value of the volume V and determining the corresponding k directly from the data,
- *k-nearest neighbors* (in [131] it is reported that the first mention of the k -nearest neighbor technique was given in [62]): choosing a fixed value of k and determining the corresponding volume V directly from the data.

The general idea of the above is visualized in Fig. 2.5. It can be proved that both approaches converge in probability to the true density $f(x)$ as $n \rightarrow \infty$, assuming that volume V shrinks with n , and k grows with n , appropriately. The parameter k is usually called the *smoothing parameter* or the *bandwidth*. In general, the estimation of its optimal value is not trivial.

2.5 Parzen Windows

This section provides a brief introduction to the first approach (Parzen windows) for constructing nonparametric density estimators. Suppose that the region \mathcal{R} is a hypercube of side length h and it encloses the k samples. Then, it should be obvious that its volume V is given by $V = h^d$, where d is the dimensionality of the problem.

To estimate the unknown density $f(x)$ at a point x , simply center the region \mathcal{R} at this point x and count the number of samples k in \mathcal{R} and then substitute this value into (2.10). The above can be expressed analytically by defining a *kernel function* (or a *window function*)

$$K(u) = \begin{cases} 1 & |u_i| \leq 1/2 \quad \forall i = 1, \dots, d \\ 0 & \text{otherwise,} \end{cases} \quad (2.13)$$

and can be visualized (in one and two dimensions) as shown in Fig. 2.6. This kernel corresponds to a unit hypercube centered at the origin and is known as the *Parzen window*.

Now, to count the total number of points k from the input dataset X_1, X_2, \dots, X_n that are inside the hypercube with side length h centered at x the following expression can be used

$$k = \sum_{i=1}^n K\left(\frac{x - X_i}{h}\right). \quad (2.14)$$

Taking into account (2.10), the desired analytical expression for the estimate of the true density $f(x)$ can be formulated as follows

$$\hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^n K\left(\frac{x - X_i}{h}\right). \quad (2.15)$$

It is evident that the Parzen windows method is very similar to the histogram with the only exception that the bin origins are determined by the input data itself. Equation (2.13) can obviously be rewritten as

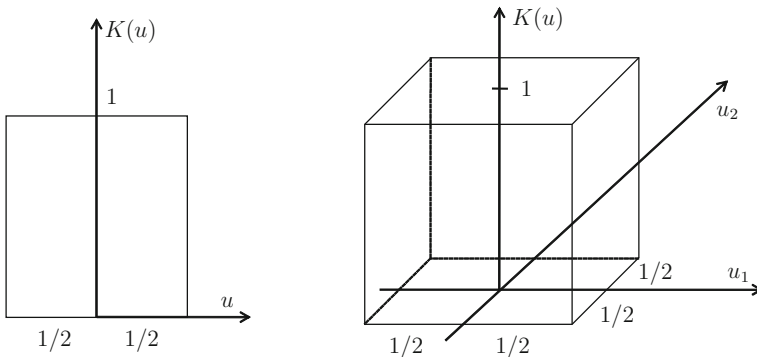


Fig. 2.6 Window functions in one and two dimensions

$$K\left(\frac{x - X_i}{h}\right) = \begin{cases} 1 & |x - X_i| \leq h/2 \\ 0 & \text{otherwise.} \end{cases} \quad \forall i = 1, \dots, d \quad (2.16)$$

Figure 2.7 demonstrates the construction of the Parzen windows estimator. It is a simple observation that the Parzen windows density estimator can be considered as a sum of boxes (for general multivariate case—hyperboxes) centered at the observations. Figure 2.8 shows several Parzen windows estimates of a mixture of two Gaussians given by

$$f(x) = \frac{1}{2}\mathcal{N}(4, 1) + \frac{1}{2}\mathcal{N}(7, 0.5). \quad (2.17)$$

The resulted densities are very jagged, with many discontinuities. This example uses the *normal scale* selector introduced in [168] described by the following equation

$$h_{NS} = \left(\frac{4}{n(d+2)}\right)^{1/(d+4)} \sigma, \quad (2.18)$$

where σ is the standard deviation of the input data.

Coming to the end of this section, we should also point out that the Parzen windows methodology has a few serious disadvantages. First, the density estimates have discontinuities and are not smooth enough. Second, all the data points are weighted

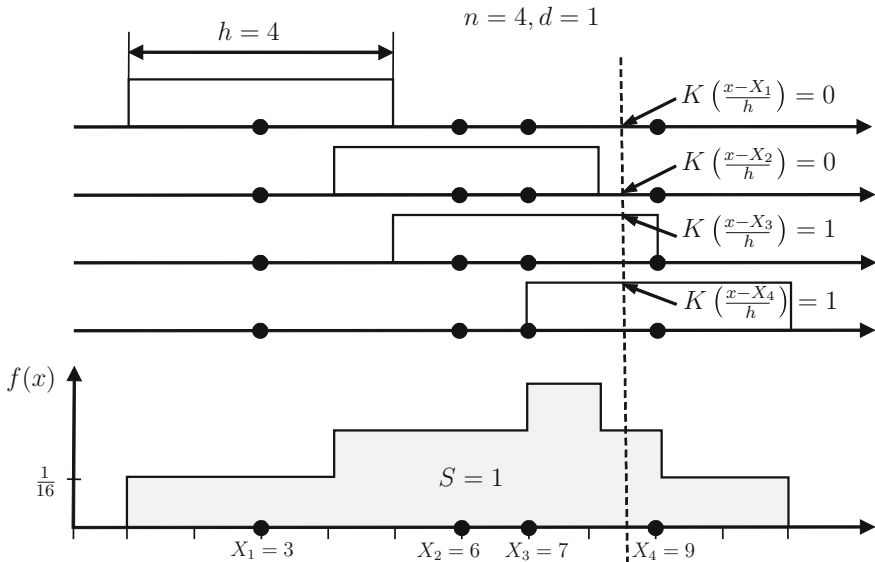


Fig. 2.7 Construction of the Parzen windows estimator for $n = 4$ as a sum of boxes centered at the observations

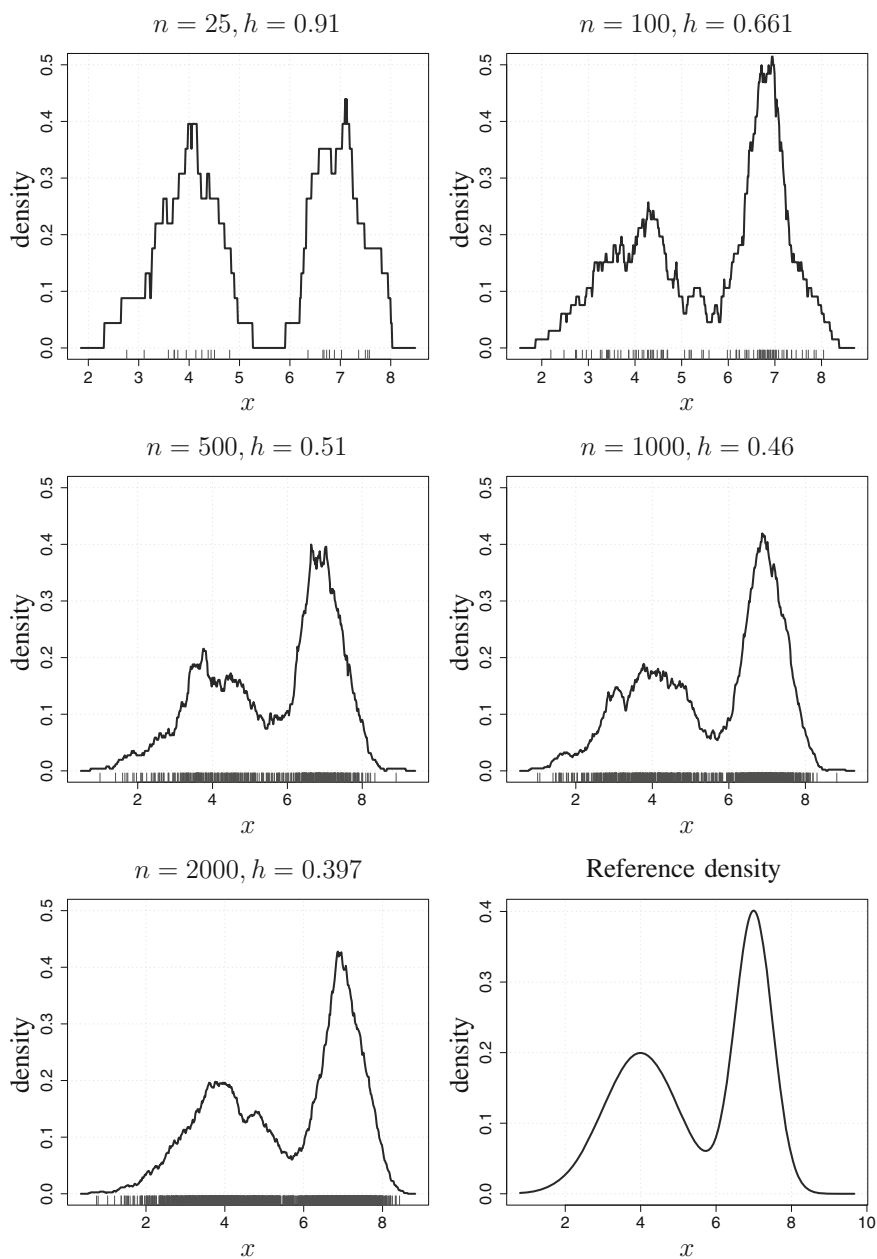


Fig. 2.8 Parzen window estimates of the mixture of two Gaussians. The lower right plot shows the true density curve

equally, regardless of their distance to the estimated point x (kernel function is ‘flat’). Third, one might need a larger number of samples in order to obtain accurate estimates of the searched density.

2.6 k -nearest Neighbors

This section briefly describes the second approach (k -nearest neighbors) for constructing nonparametric density estimators. In the k -nearest neighbors (KNN) method, a fixed value of k is chosen and the corresponding volume V is determined directly from the data. The density estimate now is

$$p(x) = \frac{k}{nV} = \frac{k}{nc_d R_k^d(x)}, \quad (2.19)$$

where $R_k^d(x)$ is the distance between the estimation point x and its k -th closest neighbor and c_d is the volume of the unit sphere in d dimensions given by

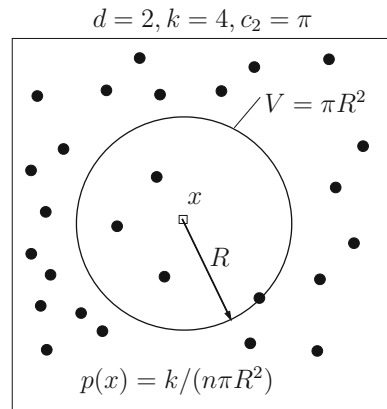
$$c_d = \frac{\pi^{d/2}}{(d/2)!} = \frac{\pi^{d/2}}{\Gamma(d/2 + 1)}. \quad (2.20)$$

This is depicted in Fig. 2.9 (for $d = 2$ and $k = 4$).

In general, the estimates generated by KNN are not considered to be adequate. The main drawbacks are as follows:

- these estimates are very prone to local noises,
- these estimates are far from zero (have very heavy tails), even for large regions where no data samples are present,

Fig. 2.9 The distance between the estimation point x and its k -th closest neighbors



- these estimates have many discontinuities (look very spiky, are not differentiable),
- the condition (1.3) does not hold (in fact no ‘legal’ densities are obtained).

The last point can be demonstrated using a toy univariate example where only one data point is present, i.e. $n = 1$. k is estimated using a popular rule-of-thumb saying that $k = \sqrt{n}$ (and rounded to the nearest integer). Here one obtains (see (2.19))

$$p(x) = \frac{k}{nV} = \frac{1}{2|x - X_i|}. \quad (2.21)$$

In this example, it is obvious that

$$\int_{-\infty}^{+\infty} \frac{1}{2|x - X_i|} = \infty \neq 1. \quad (2.22)$$

Figure 2.10 shows a very simple toy example where four datasets contain only 1, 3, 4 and 5 points respectively (marked as small black-filled circles). It goes without saying that such dataset sizes are of no practical importance in terms of any plausible nonparametric density estimation. Nevertheless, this simple example succeeds in demonstrating the above-mentioned drawbacks of the KNN estimator. The values of the integral (1.3) are 7.51, 1.27, 1.33 and 1.56 respectively (the integral limits were narrowed down to the values roughly the same as the x -label limits shown in the plots, so the value of the integral in the upper left plot is smaller than ∞).

Figure 2.11 shows several KNN estimates of a mixture of two Gaussians given by (2.17). The resulting densities (which are in fact not true densities, see the considerations above) are very jagged, with too heavy tails and many discontinuities. The values of the integral (1.3) are 0.93, 1.33, 1.23, 1.24 and 1.20 for sample datasets with $n = \{16, 32, 64, 256, 512\}$, respectively (the integral limits are narrowed down to the values roughly the same as the x -label limits shown in the plots). This example uses a more sophisticated k selector, introduced in [50], given by the following equation

$$k_{opt} = v_0 \left[\frac{4}{d+2} \right]^{d/(d+4)} n^{4/(d+4)}, \quad v_0 = \pi^{d/2} \Gamma((d+2)/d), \quad (2.23)$$

where v_0 is the hyper-volume of the unit d -ball. Figure 2.12 shows several KNN estimates of a mixture of two Gaussians given by (2.17) for two different n values and four different k values. Note that for $n = 100$ the optimal k is 37 and for $n = 250$ the optimal k is 78 (as calculated using (2.23)).

A question naturally suggests itself at this point whether one should give up the KNN approach completely. The answer is ‘no’. Typically, instead of approximating unknown densities $f(x)$, the KNN method is often used for the classification task and it is a simple approximation of the (optimal) Bayes classifier [44].

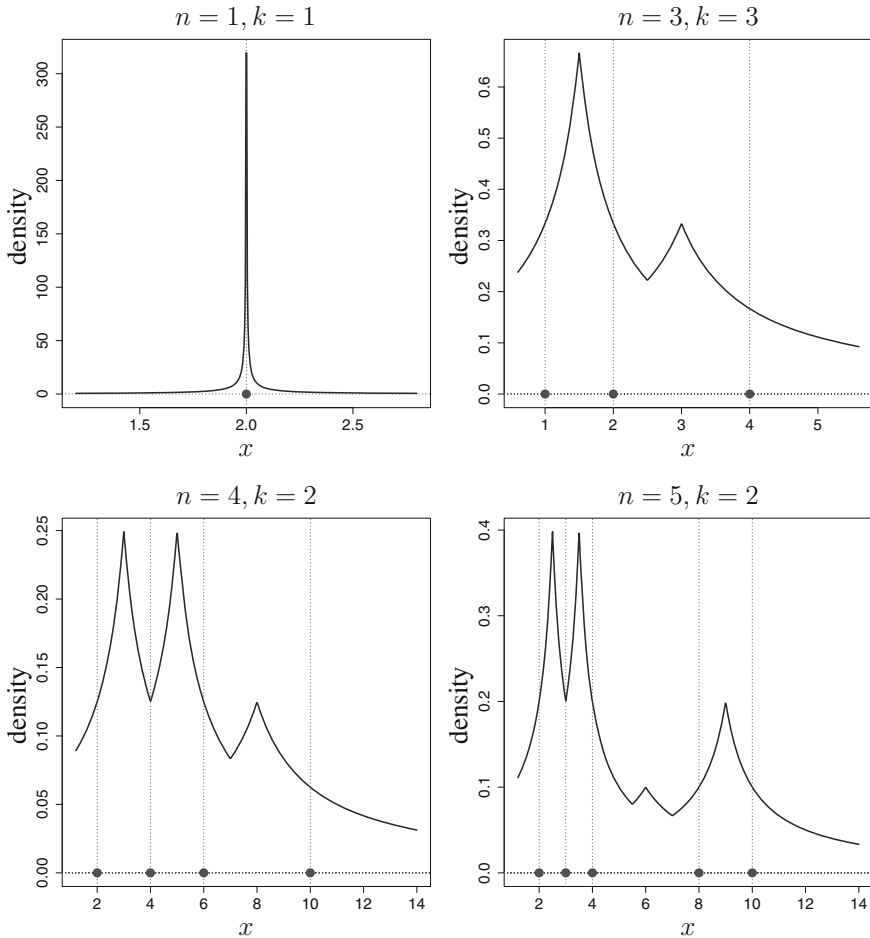


Fig. 2.10 Four toy datasets (of size 1, 3, 4 and 5) used to demonstrate the main idea behind KNN estimators

The idea here is to classify a new sample using a form of *majority voting*. Figure 2.13 is an example of that approach. The training samples, marked by filled circles, belong to the first class and the samples marked by open circles belong to the second class. Here, a 5-NN classifier ($k = 5$ nearest neighbors) is used: that is a circle is added to enclose the five nearest neighbors of the new sample marked by the star sign. The task is to decide to which class this new instance should be assigned. The 5-NN classifier would classify the star to the first class (filled circles), since the majority of the objects (three out of five) located inside the circled region are filled circles.

To put the above in more general terms, assume that one has a dataset with n examples, with n_i of them coming from the class c_i so that $\sum_i n_i = n$. The goal is

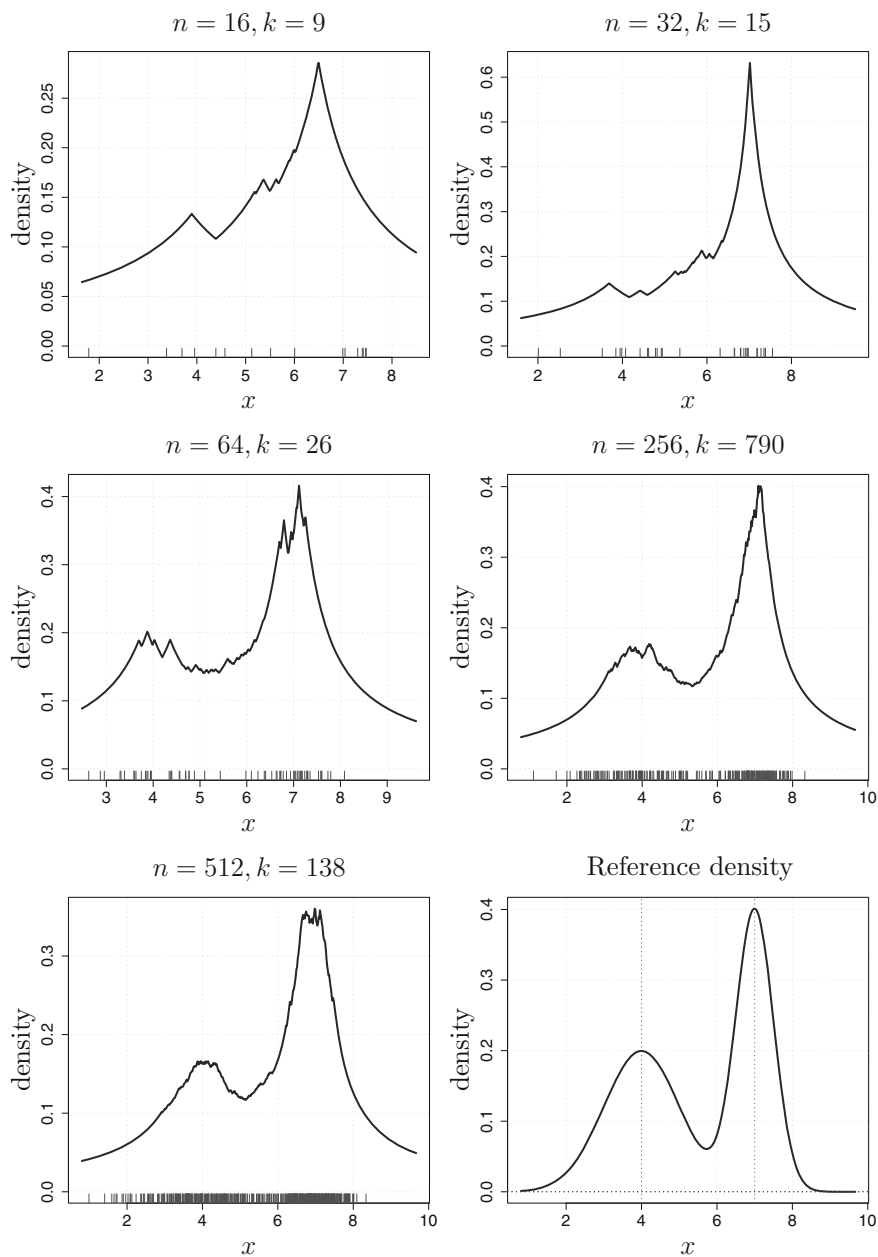


Fig. 2.11 KNN estimates of a mixture of two Gaussians. The lower right plot shows the true density curve

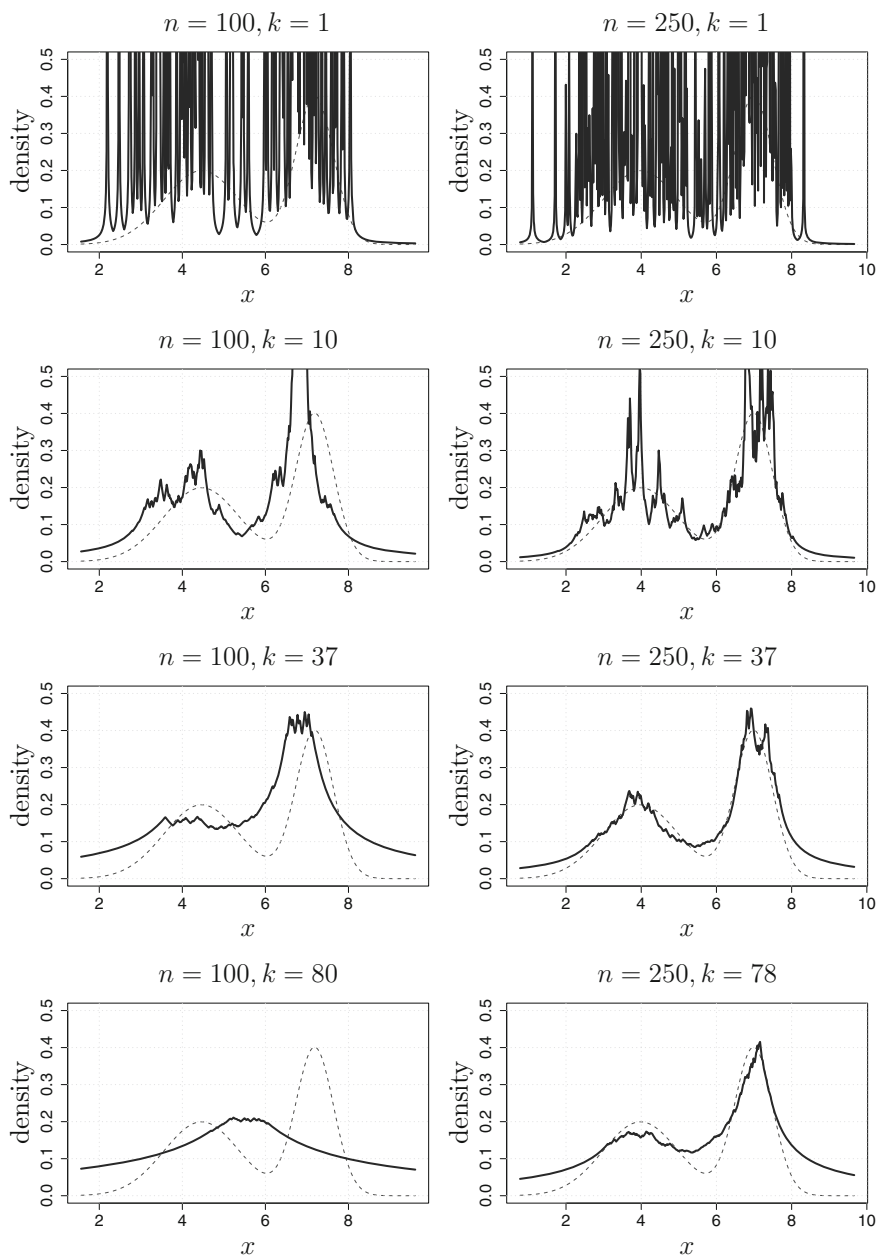
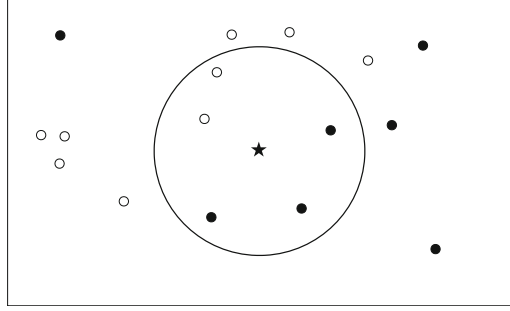


Fig. 2.12 KNN estimates of a mixture of two Gaussians. Different k and n are used. For $n = 100$ the optimal k is 37 and for $n = 250$ the optimal k is 78. The true density curve is plotted with the dashed line

Fig. 2.13 Two-dimensional toy example of a KNN ($k = 5$) classifier



to classify an unknown sample x . Recall that the unconditional density (now labeled $p(x)$) is estimated by $p(x) = k/(nV)$. Place a hyper-sphere with volume V around x enclosing k samples. Let k_i samples out of k be labeled c_i , then we have that

$$p(x|c_i) \cong \frac{k_i}{n_i V}, \quad (2.24)$$

and the prior probabilities (also known as class priors) are approximated by

$$p(c_i) \cong \frac{n_i}{n}. \quad (2.25)$$

Now, putting everything together, and using Bayes' theorem one obtains the posterior probability of the class membership using

$$p(c_i|x) = \frac{p(x|c_i)p(c_i)}{p(x)} = \frac{\frac{k_i}{n_i V} \frac{n_i}{n}}{\frac{k}{nV}} = \frac{k_i}{k}. \quad (2.26)$$

The estimate of the posterior probability is provided simply by the fraction of samples that belong to the class c_i . This is a very simple and intuitive estimator. In other words, given an unlabeled example x , find its k closest neighbors among the example labeled points. Then, assign x to the most frequent class among these neighbors (as demonstrated in Fig. 2.13). The KNN classifier only requires the following:

- an integer k (determined using for example (2.23)),
- a set of labeled examples (as training data, used in the verification process),
- a metric to measure the distance between x and potential clusters. Usually, the classical Euclidean distance is used.

Finally, in Fig. 2.14 a simple example based on an artificial dataset is presented (a Gaussian mixture model).

The training samples belong to two classes, denoted by open and filled circles, respectively. Based on this training dataset, four KNN classifiers have been built (for

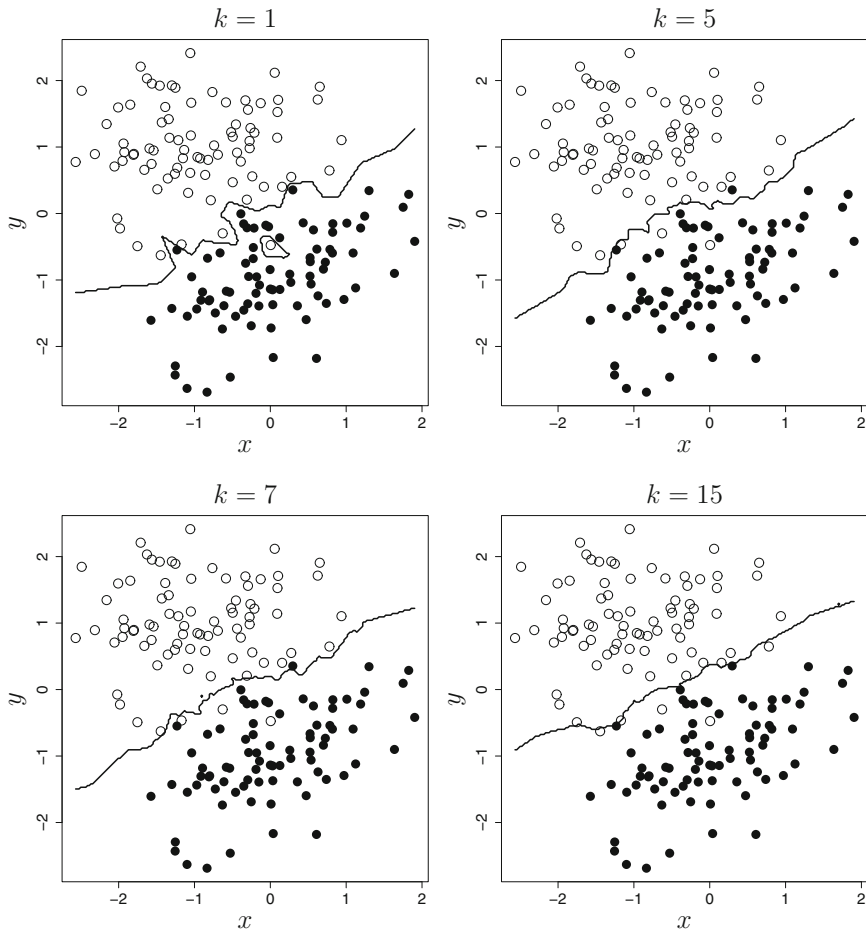


Fig. 2.14 KNN classifiers on a two-class Gaussian mixture data

$k = \{1, 5, 7, 15\}$) with decision boundaries represented by solid lines. It is easily seen how k affects the boundaries. The problem of selecting the optimal k is a critical one in terms of obtaining the final results. A small value of k means that noise can have a higher influence on the result (risk of overfitting). On the other hand, a large value of k makes it computationally expensive and the KNN classifier may misclassify the test sample because its list of nearest neighbors can include many samples from other classes (risk of oversimplifying). Moreover, more numerous classes can have a dominant impact on assigning x to these classes. A popular rule of thumb in choosing the value of k is to set $k = \sqrt{n}$, where n is the number of samples in the training dataset. Another popular method is based on certain cross-validation techniques. This approach however, is often unsuccessful as reported in [65].

Nonparametric Kernel Density Estimation and Its
Computational Aspects

Gramacki, A.

2018, XXIX, 176 p. 70 illus., Hardcover

ISBN: 978-3-319-71687-9