



23. Juni 2017

Nichtparametrische Verteilungen



Nichtparametrische Verteilungen

23. Juni 2017

Dichteschätzung:

- Gesucht: $p(\mathbf{x})$
- Es gilt immer: $P = \int_{\mathcal{R}} p(\mathbf{x}') d\mathbf{x}'$
- Daraus können wir einen Mittelwert von $p(\mathbf{x})$ in der Region \mathcal{R} (mit Volumen $V = \int_{\mathcal{R}} d\mathbf{x}'$) berechnen:

$$\frac{P}{V} = \frac{\int_{\mathcal{R}} p(\mathbf{x}') d\mathbf{x}'}{\int_{\mathcal{R}} d\mathbf{x}'}$$

Nichtparametrische Verteilungen

23. Juni 2017

- Bei n mit $p(\mathbf{x})$ verteilten Stichproben ist die Wahrscheinlichkeit, genau k davon in der Region \mathcal{R} zu finden:

$$P_k = \binom{n}{k} P^k (1 - P)^{n-k}$$

Binomialverteilung mit $n = 24$, $p = 0.166667$, $q = 0.833333$



natürlich mit Erwartungswert

$$\mathcal{E}[k] = nP$$



Nichtparametrische Verteilungen

23. Juni 2017

- Das ist bei großen n nicht schlecht:

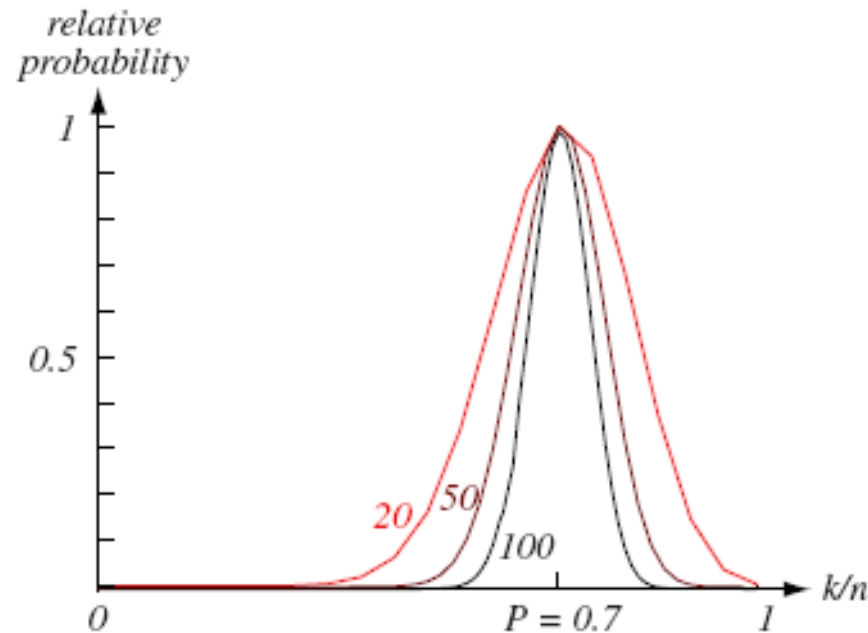


FIGURE 4.1. The relative probability an estimate given by Eq. 4 will yield a particular value for the probability density, here where the true probability was chosen to be 0.7. Each curve is labeled by the total number of patterns n sampled, and is scaled to give the same maximum (at the true probability). The form of each curve is binomial, as given by Eq. 2. For large n , such binomials peak strongly at the true probability. In the limit $n \rightarrow \infty$, the curve approaches a delta function, and we are guaranteed that our estimate will give the true probability. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.



Nichtparametrische Verteilungen

23. Juni 2017

- Schätzwert für $p(\mathbf{x})$ ist also:

$$p(\mathbf{x}) \approx \frac{k/n}{V}$$

- Wenn wir die Anzahl der Stichproben erhöhen, können wir bessere Schätzwerte $p_n(\mathbf{x})$ erhalten.

$$p_n(\mathbf{x}) = \frac{k_n/n}{V_n}$$

- Was müssen wir tun, um gute Schätzwerte zu erhalten?
 - Region kleiner: $V \rightarrow 0$
 - Mehr Stichproben: $k \rightarrow \infty$



Nichtparametrische Verteilungen

23. Juni 2017

Zwei grundsätzliche Möglichkeiten:

- Festlegen von V in Abhängigkeit von n , z.B.:

$$V_n \propto \frac{1}{\sqrt{n}} \quad (\text{Parzen-Fenster})$$

- Festlegen von k_n , z.B.:

$$k_n = \sqrt{n} \quad (k_n\text{-nearest-neighbor estimation})$$

Nichtparametrische Verteilungen

- Parzen-Fenster
- k_n -nearest-neighbor estimation

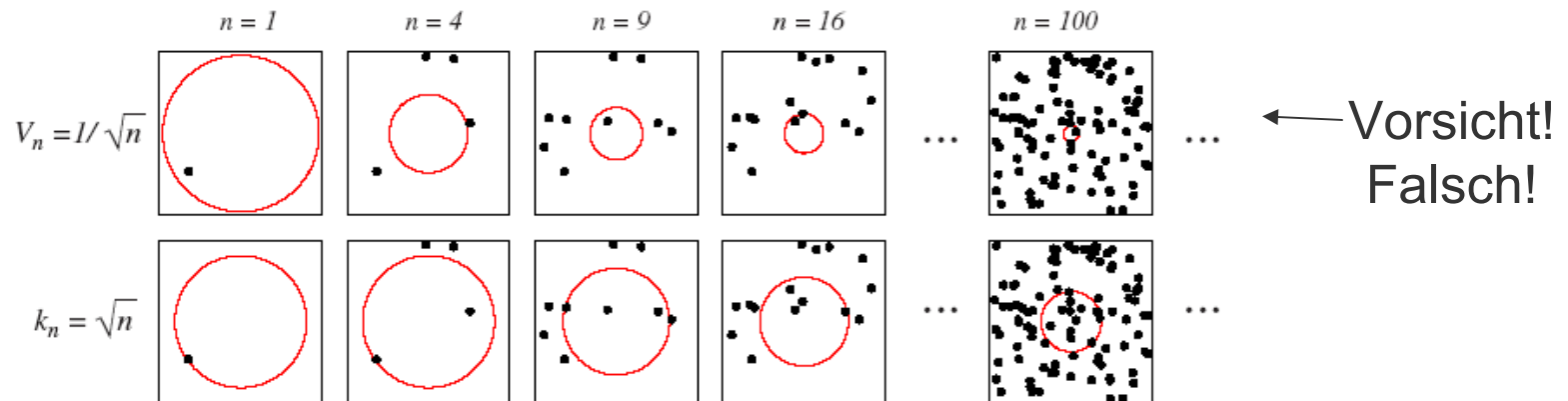


FIGURE 4.2. There are two leading methods for estimating the density at a point, here at the center of each square. The one shown in the top row is to start with a large volume centered on the test point and shrink it according to a function such as $V_n = 1/\sqrt{n}$. The other method, shown in the bottom row, is to decrease the volume in a data-dependent way, for instance letting the volume enclose some number $k_n = \sqrt{n}$ of sample points. The sequences in both cases represent random variables that generally converge and allow the true density at the test point to be calculated. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Parzen-Fenster

23. Juni 2017

Ist die Region \mathcal{R} ein Hyperwürfel der Dimension d mit Kantenlänge h_n , so können wir eine Fensterfunktion $\varphi(\mathbf{u})$ definieren:

$$\varphi(\mathbf{u}) = \begin{cases} 1 & |u_j| \leq \frac{1}{2}; \quad j = 1, \dots, d \\ 0 & \text{sonst} \end{cases}$$

k_n kann dann geschrieben werden als:

$$k_n(\mathbf{x}) = \sum_{i=1}^n \varphi\left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n}\right)$$



Parzen-Fenster

Als Schätzwert für die Wahrscheinlichkeitsdichte erhalten wir damit:

$$p_n(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n \frac{1}{V_n} \varphi\left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n}\right)$$

Das entspricht dem Durchschnitt einer Summe von Wahrscheinlichkeitsdichtefunktionen.

Für $\varphi(\mathbf{x}/h_n) = \varphi(\mathbf{u})$ können wir nun auch andere Fensterfunktionen $\varphi'(\mathbf{u})$ einsetzen, nicht nur den Hyperwürfel. Bedingungen:

$$\varphi'(\mathbf{u}) \geq 0$$

$$\int \varphi'(\mathbf{u}) d\mathbf{u} = 1$$



Parzen-Fenster

23. Juni 2017

Die Wahrscheinlichkeitsdichtefunktionen, aus denen auf diese Weise $p_n(\mathbf{x})$ zusammengesetzt wird, ergeben sich damit zu:

$$\delta_n(\mathbf{x}) = \frac{1}{V_n} \varphi'(\mathbf{x}/h_n)$$

Beispiel: Gaußglocken

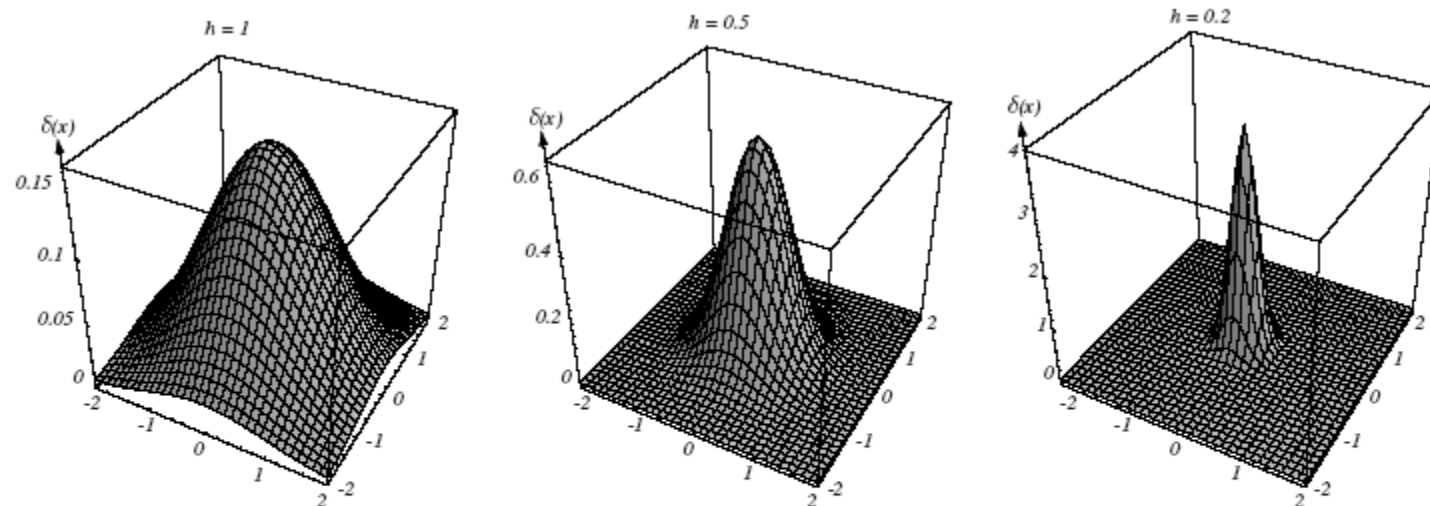


FIGURE 4.3. Examples of two-dimensional circularly symmetric normal Parzen windows for three different values of h . Note that because the $\delta(x)$ are normalized, different vertical scales must be used to show their structure. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.



23. Juni 2017

Beispiel: Dichteschätzung mit Gaußglocken als Parzen-Fenster

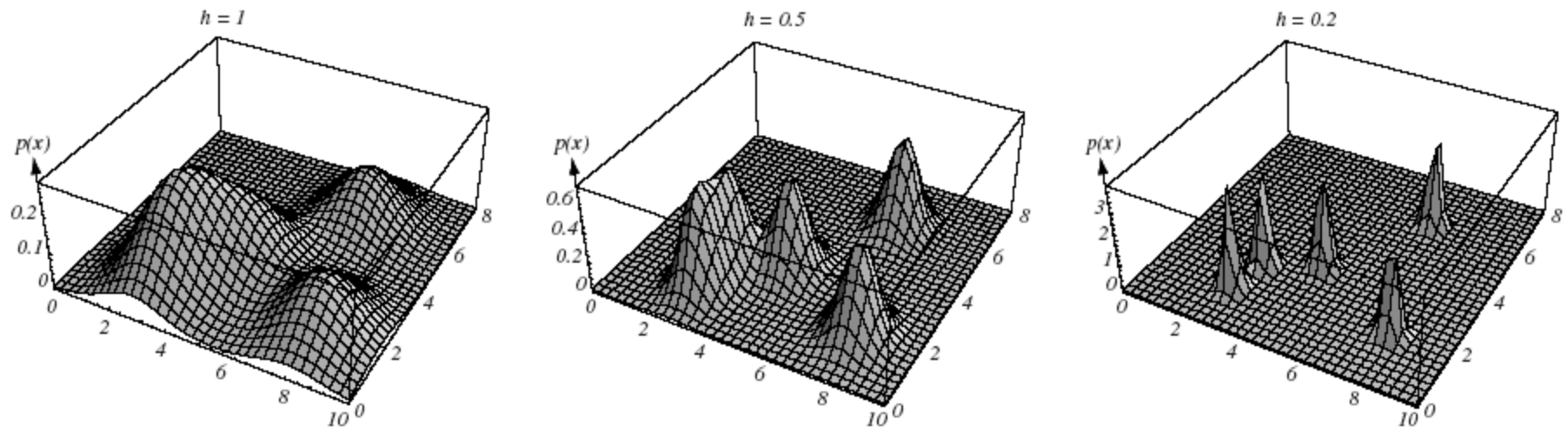


FIGURE 4.4. Three Parzen-window density estimates based on the same set of five samples, using the window functions in Fig. 4.3. As before, the vertical axes have been scaled to show the structure of each distribution. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.



Parzen-Fenster

Die Schätzung $p_n(\mathbf{x})$ hängt von der jeweiligen Stichprobe ab. Der Erwartungswert $\bar{p}_n(\mathbf{x})$ jedoch kann berechnet werden, da die Stichproben ja aus einer Verteilung $p(\mathbf{x})$ stammen:

$$\begin{aligned}\bar{p}_n(\mathbf{x}) &= \mathcal{E} [p_n(\mathbf{x})] \\ &= \frac{1}{n} \sum_{i=1}^n \mathcal{E} \left[\frac{1}{V_n} \varphi \left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n} \right) \right] \\ &= \int \frac{1}{V_n} \varphi \left(\frac{\mathbf{x} - \mathbf{v}}{h_n} \right) p(\mathbf{v}) d\mathbf{v} \\ &= \int \delta_n(\mathbf{x} - \mathbf{v}) p(\mathbf{v}) d\mathbf{v}\end{aligned}$$



Parzen-Fenster

Als Ergebnis erhalten wir, daß $\bar{p}_n(\mathbf{x})$ eine Faltung von $p(\mathbf{x})$ mit $\delta_n(\mathbf{x})$ darstellt.

Dabei geht $\delta_n(\mathbf{x})$ für $n \rightarrow \infty$ gegen die Diracsche Deltafunktion.

Für breitere $\delta_n(\mathbf{x})$ (die normalerweise zu kleineren n gehören) erhalten wir eine geglättete, tiefpaßgefilterte Verteilung.

Ein beliebig gutes $\bar{p}_n(\mathbf{x})$ kann man theoretisch unabhängig von n erhalten, wenn man nur $\delta_n(\mathbf{x})$ genügend schmal macht. Allerdings bekommt man die korrekte Verteilung nur als Erwartungswert, d.h. man müßte viele Versuche mit je n Stichproben machen.

Deshalb interessiert uns die Varianz der Schätzung



Parzen-Fenster

Für die Varianz der $p_n(\mathbf{x})$ erhalten wir:

$$\begin{aligned}
 \sigma_n^2(\mathbf{x}) &= \sum_{i=1}^n \mathcal{E} \left[\left(\frac{1}{nV_n} \varphi \left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n} \right) - \frac{1}{n} \bar{p}_n(\mathbf{x}) \right)^2 \right] \\
 &= n \mathcal{E} \left[\frac{1}{n^2 V_n^2} \varphi^2 \left(\frac{\mathbf{x} - \mathbf{x}_i}{h_n} \right) \right] - \frac{1}{n} \bar{p}_n^2(\mathbf{x}) \\
 &= \frac{1}{nV_n} \int \frac{1}{V_n} \varphi^2 \left(\frac{\mathbf{x} - \mathbf{v}}{h_n} \right) p(\mathbf{v}) d\mathbf{v} - \frac{1}{n} \bar{p}_n^2(\mathbf{x}) \\
 &\leq \frac{\sup_{\mathbf{u}} (\varphi(\mathbf{u})) \bar{p}_n(\mathbf{x})}{nV_n}
 \end{aligned}$$

Parzen-Fenster

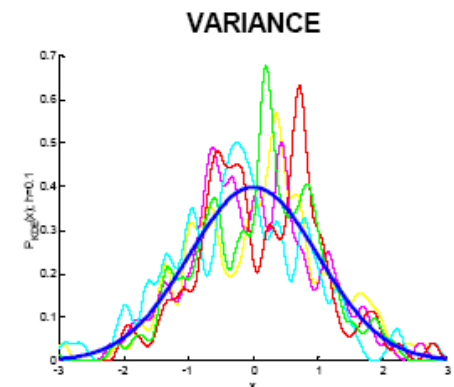
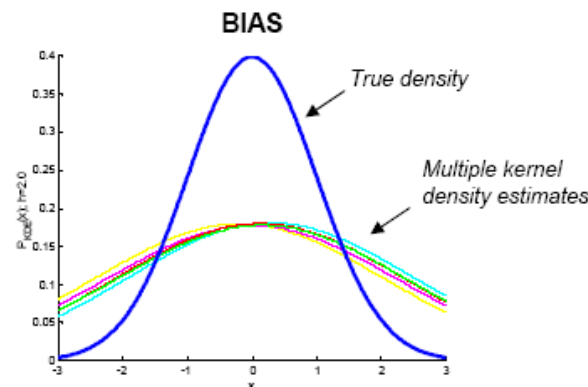
Benötigt: Kompromiss:

- Einerseits, wenn Fenster groß:
Fehler durch Glättung
- Andererseits, wenn Fenster klein:
Fehler durch Streuung

Wichtig: $V_n \rightarrow 0$ und $nV_n \rightarrow \infty$

Z.B.: $V_n = \frac{V_1}{\sqrt{n}}$

oder $V_n = \frac{V_1}{\ln n}$





Parzen-Fenster

Eindimensionales Beispiel (auch für Bilder auf den folgenden Seiten):

$$\varphi(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}}$$

Damit wird:

$$p_n(x) = \frac{1}{n} \sum_{i=1}^n \frac{1}{h_n} \varphi\left(\frac{x - x_i}{h_n}\right)$$

Wir wählen:

$$h_n = \frac{h_1}{\sqrt{n}}$$

Parzen-Fenster

23. Juni 2017

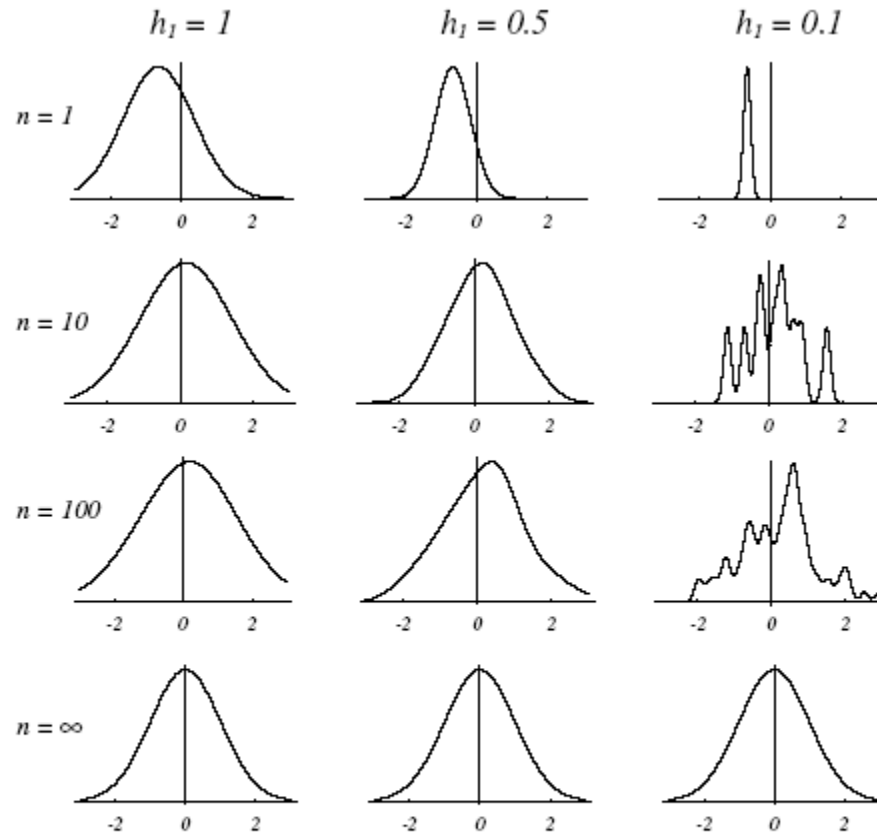


FIGURE 4.5. Parzen-window estimates of a univariate normal density using different window widths and numbers of samples. The vertical axes have been scaled to best show the structure in each graph. Note particularly that the $n = \infty$ estimates are the same (and match the true density function), regardless of window width. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.



Parzen-Fenster

23. Juni 2017

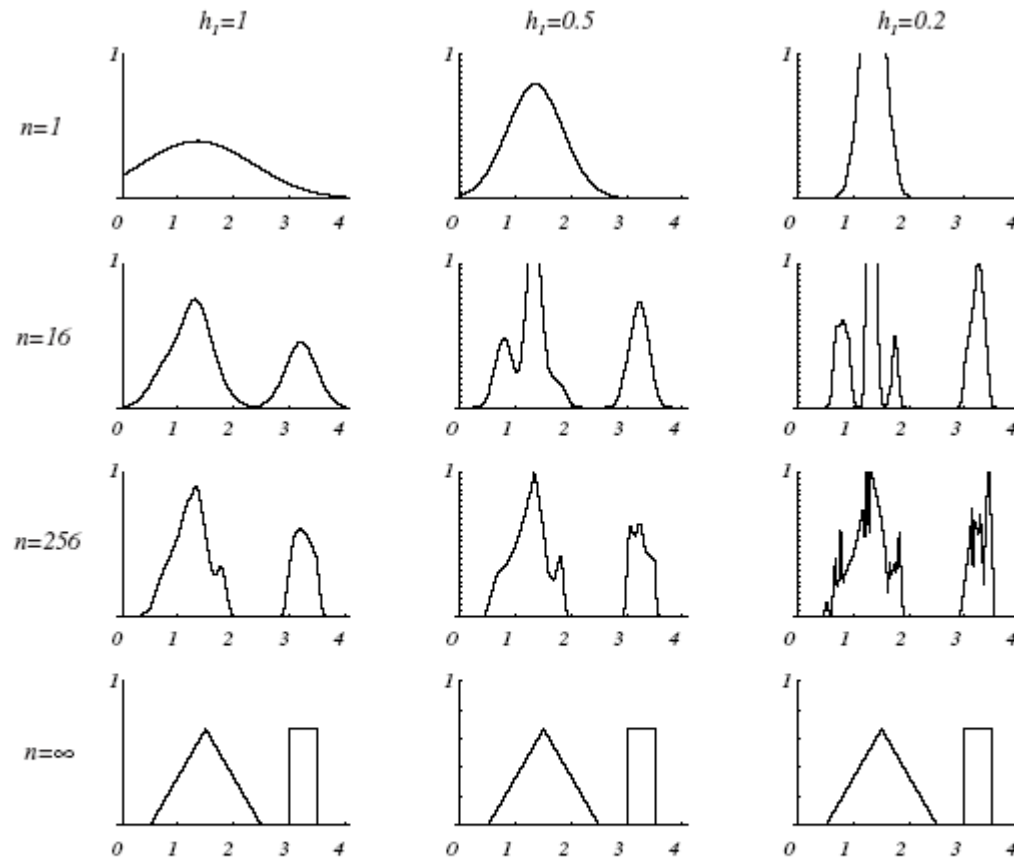


FIGURE 4.7. Parzen-window estimates of a bimodal distribution using different window widths and numbers of samples. Note particularly that the $n = \infty$ estimates are the same (and match the true distribution), regardless of window width. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

23. Juni 2017

Parzen-Fenster

2D-Dichteschätzung

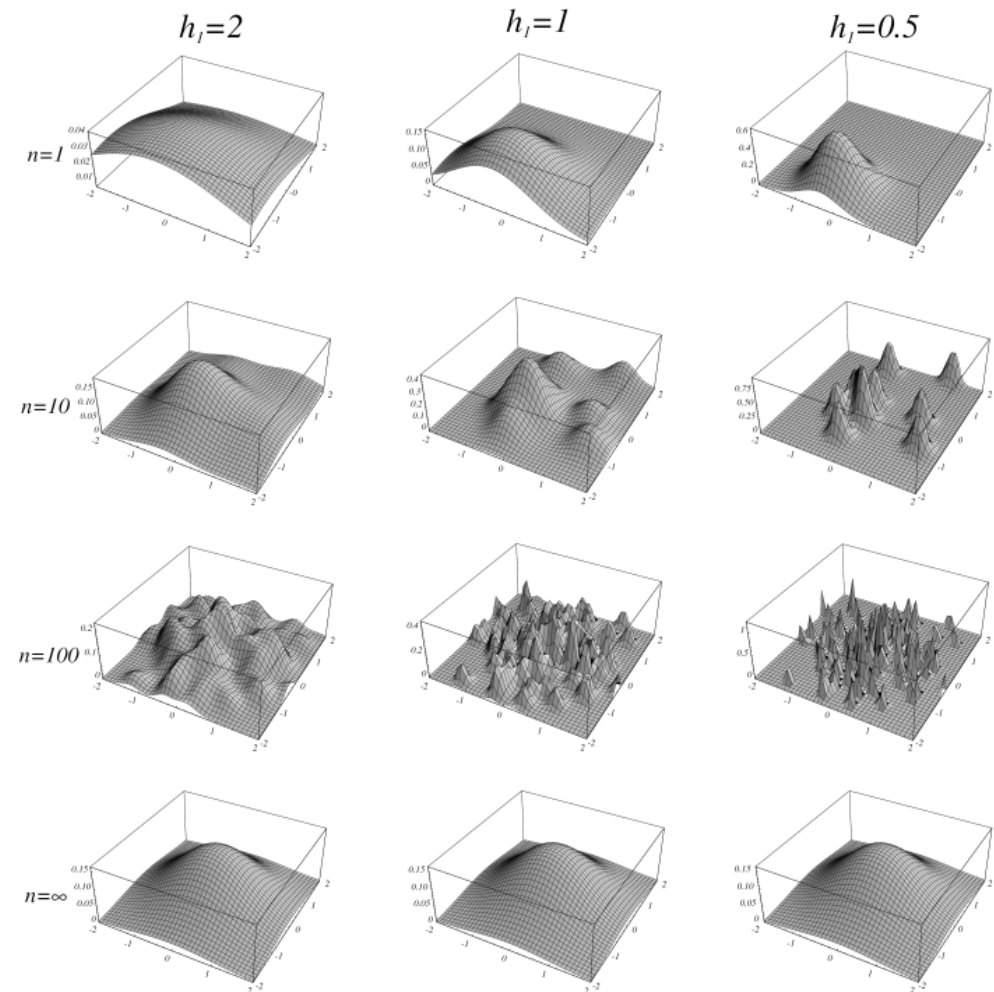


FIGURE 4.6. Parzen-window estimates of a bivariate normal density using different window widths and numbers of samples. The vertical axes have been scaled to best show the structure in each graph. Note particularly that the $n = \infty$ estimates are the same (and match the true distribution), regardless of window width. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Parzen-Fenster

Klassifizierung mit Parzen-Fenster

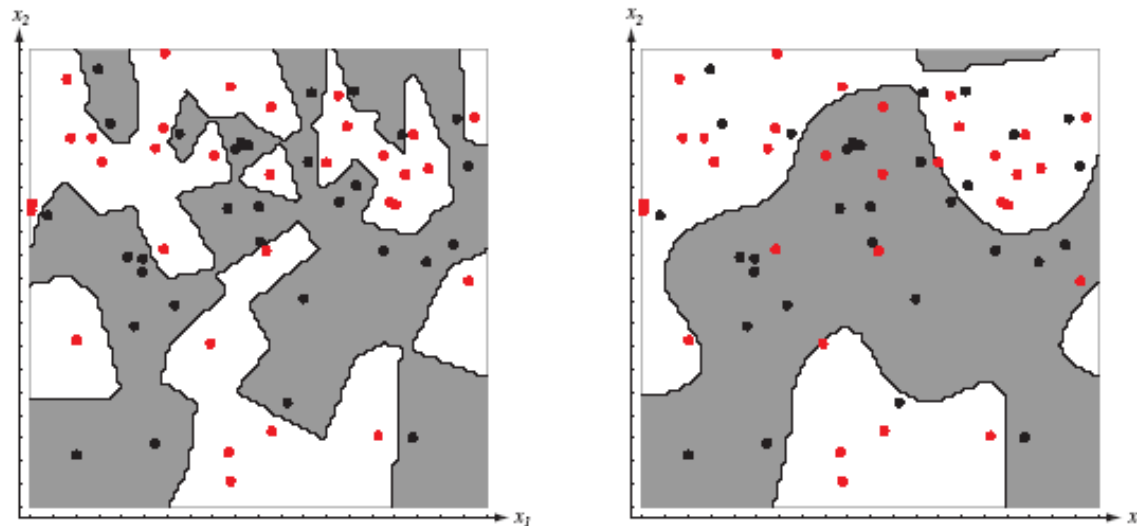


FIGURE 4.8. The decision boundaries in a two-dimensional Parzen-window dichotomizer depend on the window width h . At the left a small h leads to boundaries that are more complicated than for large h on same data set, shown at the right. Apparently, for these data a small h would be appropriate for the upper region, while a large h would be appropriate for the lower region; no single window width is ideal overall. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.



Parzen-Fenster

Klassifizierung mit Parzen-Fenster

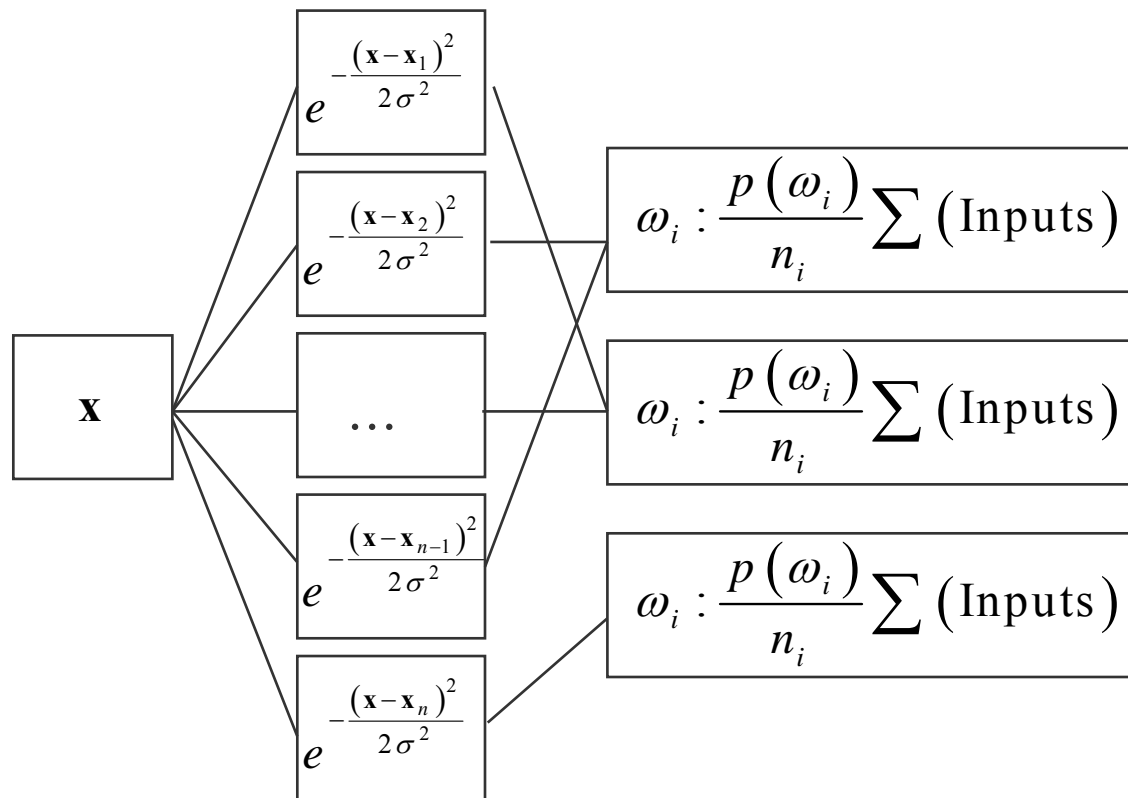
- Vorteile:
 - Allgemeinheit (gleiches Verfahren f. verschiedene Verteilungen)
 - Konvergenz bei beliebig komplizierten Dichtefunktionen
- Nachteile
 - Hohe Zahl von Stichproben; VIEL höher, als wenn Form bekannt.
 - Keine Datenreduktion
 - „Curse of Dimensionality“



23. Juni 2017

Probabilistic Neural Networks (PNNs)

Maschine für Parzen-Klassifizierung (mit Gauss-Fenstern):



Probabilistic Neural Networks (PNNs)

23. Juni 2017

Alternative: Probabilistic Neural Networks

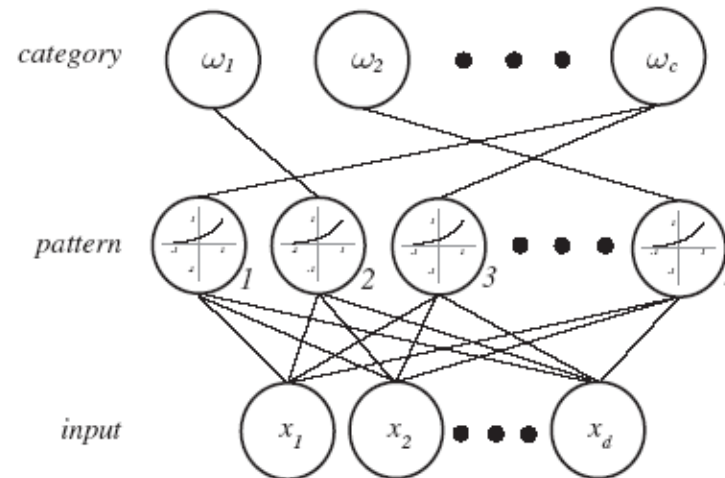


FIGURE 4.9. A probabilistic neural network (PNN) consists of d input units, n pattern units, and c category units. Each pattern unit forms the inner product of its weight vector and the normalized pattern vector \mathbf{x} to form $z = \mathbf{w}^t \mathbf{x}$, and then it emits $\exp[(z - 1)/\sigma^2]$. Each category unit sums such contributions from the pattern unit connected to it. This ensures that the activity in each of the category units represents the Parzen-window density estimate using a circularly symmetric Gaussian window of covariance $\sigma^2 \mathbf{I}$, where \mathbf{I} is the $d \times d$ identity matrix. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

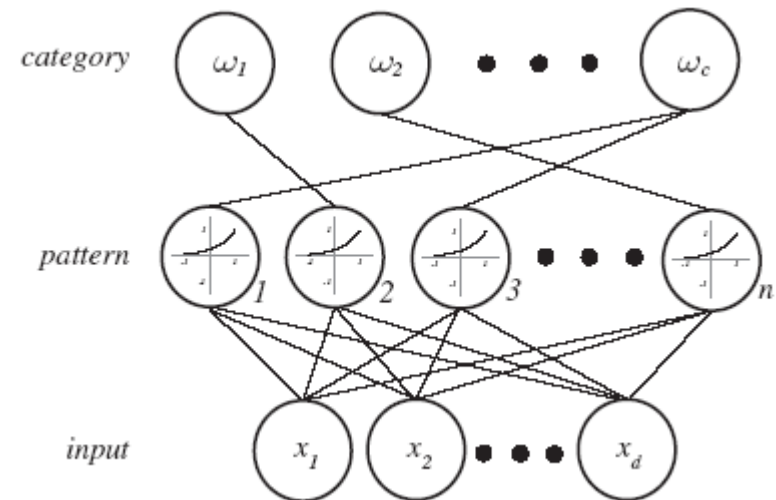


Probabilistic Neural Networks (PNNs)

23. Juni 2017

PNN:

- Eingabevektoren werden immer normalisiert: $\mathbf{x}_k^T \mathbf{x} = 1$
- Trainingsvektoren werden als Gewichte \mathbf{w}_k in den Links (input – pattern) gespeichert. Jeder Trainingsvektor entspricht einem pattern.
- Jede pattern-Einheit berechnet das Skalarprodukt $net_k = \mathbf{w}_k^T \mathbf{x}$ und anschließend davon die Funktion $e^{(net_k - 1)/\sigma^2}$





Probabilistic Neural Networks (PNNs)

23. Juni 2017

PNN:

- Eingabevektoren werden immer normalisiert: $\mathbf{x}_k^T \mathbf{x} = 1$
- Das bedeutet Projektion auf Einheitskreis

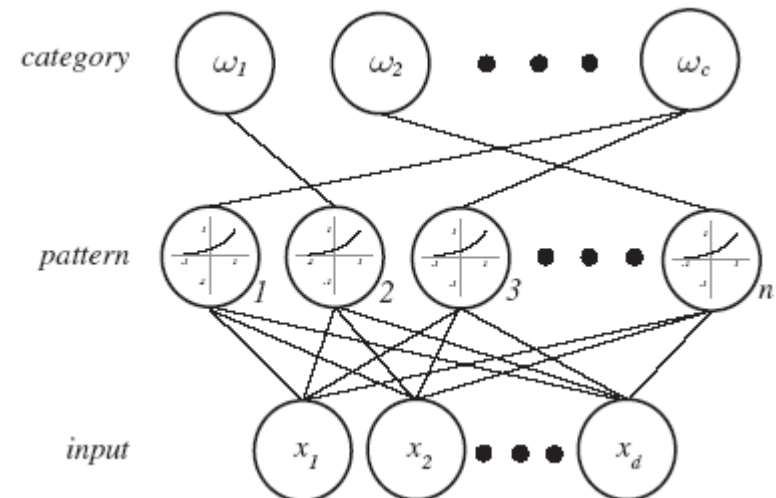


Probabilistic Neural Networks (PNNs)

23. Juni 2017

PNN:

- Das Ergebnis wird in den category-Einheiten – getrennt nach Kategorien – aufsummiert.
- Die Kategorie mit der größten Summe wird gewählt





Probabilistic Neural Networks (PNNs)

23. Juni 2017

PNN:

Begründung f. Funktion in pattern-Einheit
(activation function /
transfer function):

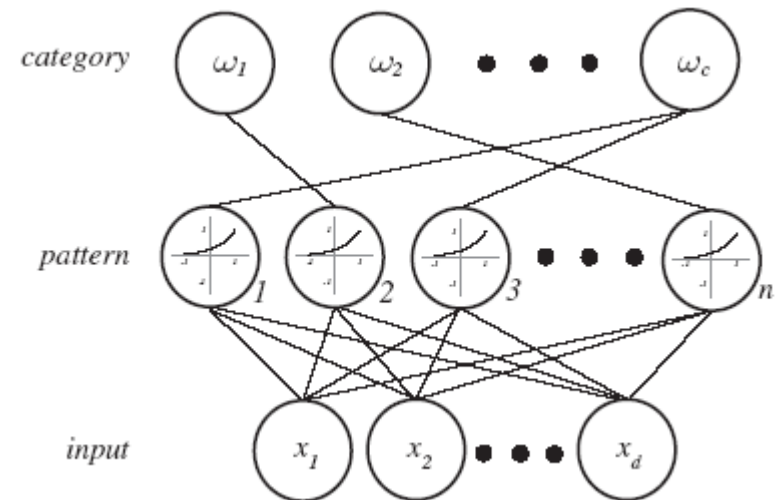
Wir wollen Gaußfunktion:

$$\begin{aligned}
 \varphi\left(\frac{\mathbf{x} - \mathbf{w}_k}{h_n}\right) &= e^{-(\mathbf{x} - \mathbf{w}_k)^T (\mathbf{x} - \mathbf{w}_k) / 2\sigma^2} \\
 &= e^{-\left(\mathbf{x}^T \mathbf{x} + \mathbf{w}_k^T \mathbf{w}_k - 2\mathbf{x}^T \mathbf{w}_k\right) / 2\sigma^2}
 \end{aligned}$$

Mit $\mathbf{x}^T \mathbf{x} = \mathbf{w}^T \mathbf{w} = 1$

Können wir das aber schreiben als:

$$e^{(net_k - 1) / \sigma^2}$$



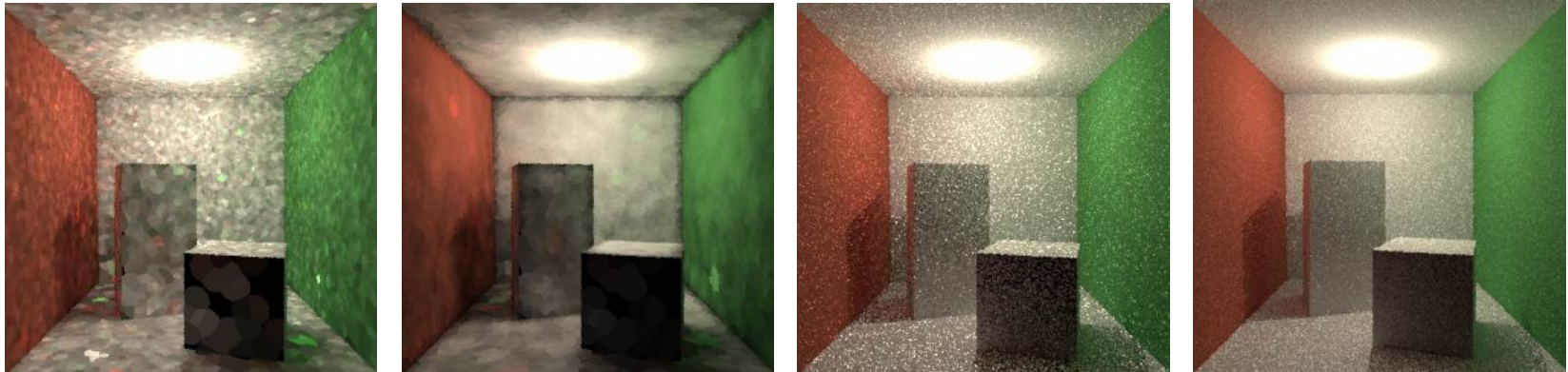
K_n -Nearest-Neighbor Estimation

23. Juni 2017

Problem mit Parzen-Fenster:

- Wahl von h_n bzw. V_n

Ähnliches Problem: Photonmapping, Schätzen der Photonendichte



Bilder aus: Johann Won: CS348b Photon Map Assignment Gallery www.stanford.edu/~jhwon/cs348b/gallery.html



K_n -Nearest-Neighbor Estimation

23. Juni 2017

Problem mit Parzen-Fenster:

- Wahl von h_n bzw. V_n

Lösungsmöglichkeit:

- V_n soll so groß sein, daß es eine festgelegte Zahl k_n von Trainingsdaten einschließt.
- Durch garantierte Zahl von Samples in Zelle kann Streuung in Schach gehalten werden.



K_n -Nearest-Neighbor Estimation

K_n -Nearest-Neighbor Estimation:

- Fenster wird so lange vergrößert, bis k_n Trainingsdaten enthalten sind.

- Dann Berechnung von $p_n(\mathbf{x})$ als:
$$p_n(\mathbf{x}) = \frac{k_n/n}{V_n}$$

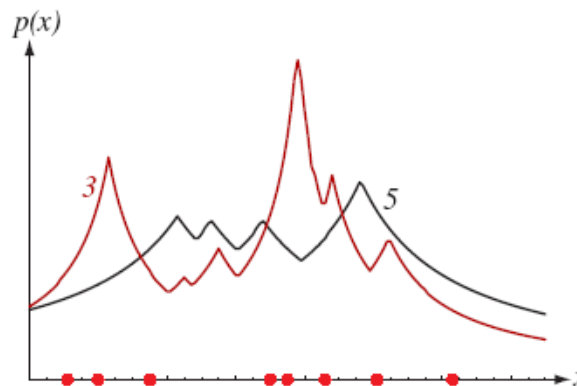


FIGURE 4.10. Eight points in one dimension and the k -nearest-neighbor density estimates, for $k = 3$ and 5 . Note especially that the discontinuities in the slopes in the estimates generally lie away from the positions of the prototype points. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.



K_n -Nearest-Neighbor Estimation

23. Juni 2017

K_n -Nearest-Neighbor Estimation

Wahl von k_n :

Bedingungen für $n \rightarrow \infty$:

- $k_n \rightarrow \infty$
- $\frac{k_n}{n} \rightarrow 0$

Beispiel: $k_n = \sqrt{n}$
damit wird: $V_n \simeq \frac{1}{\sqrt{n} p(\mathbf{x})}$

K_n -Nearest-Neighbor Estimation

23. Juni 2017

Beispiel:

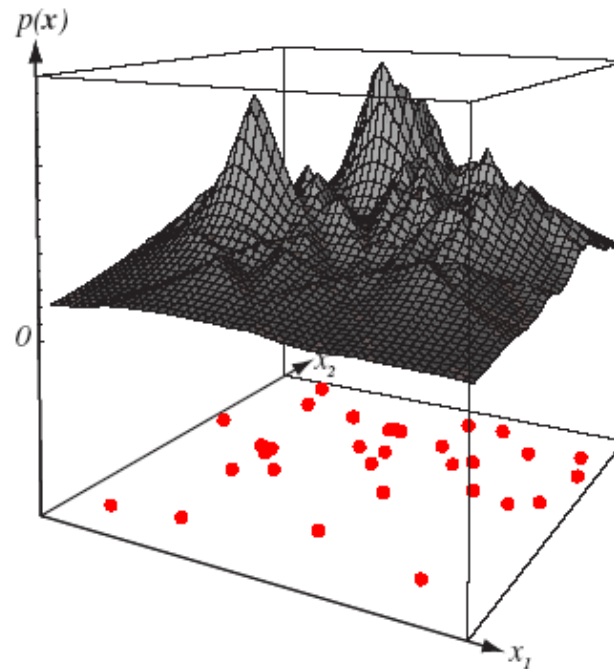


FIGURE 4.11. The k -nearest-neighbor estimate of a two-dimensional density for $k = 5$. Notice how such a finite n estimate can be quite “jagged,” and notice that discontinuities in the slopes generally occur along lines away from the positions of the points themselves. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.



23. Juni 2017

K_n -Nearest-Neighbor Estimation

Eindimensionales Beispiel:

$$p_1(x) = \frac{1}{2|x - x_1|}$$

Integral über

$$p_n(x)$$

leider nicht

beschränkt.

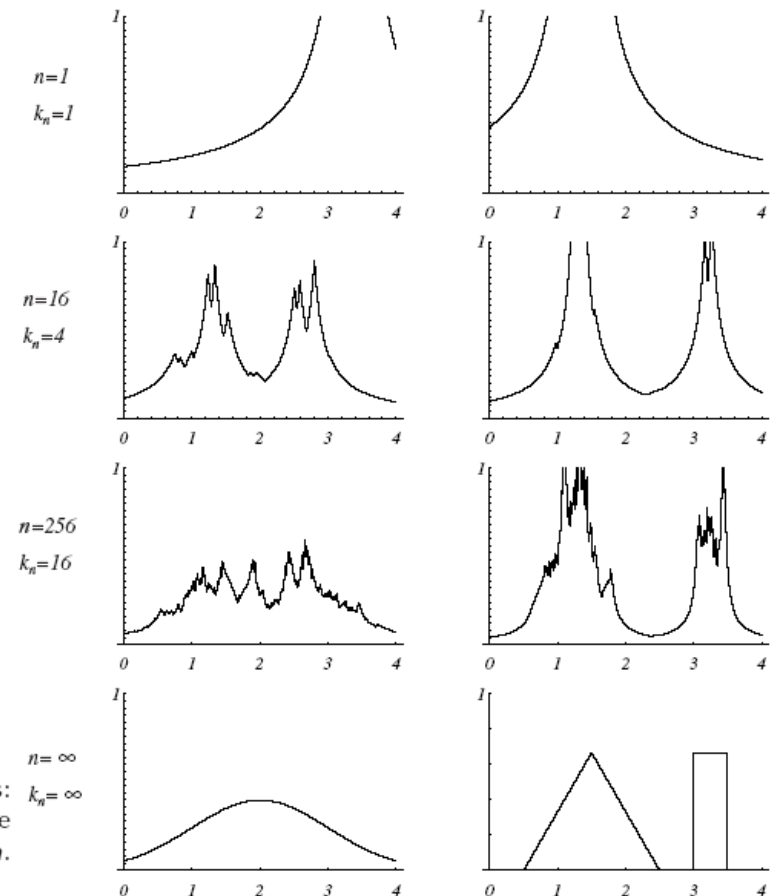


FIGURE 4.12. Several k -nearest-neighbor estimates of two unidimensional densities: a Gaussian and a bimodal distribution. Notice how the finite n estimates can be quite “spiky.” From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.



K_n -Nearest-Neighbor Estimation

23. Juni 2017

Klassifizierung:

Schätzen der a-posteriori Wahrscheinlichkeiten $P_n(\omega_i | \mathbf{x})$ für Klasse ω_i .

Mit der Dichteschätzung $p_n(\mathbf{x}, \omega_i) = \frac{k_i/n}{V}$

erhalten wir:
$$P_n(\omega_i | \mathbf{x}) = \frac{p_n(\mathbf{x}, \omega_i)}{\sum_j p_n(\mathbf{x}, \omega_j)} = \frac{k_i}{k}$$

Nearest-Neighbor Estimation

Nearest-Neighbor-Klassifizierung

Wir werden zeigen, dass der eine nächste Nachbar genügt, um gut zu klassifizieren

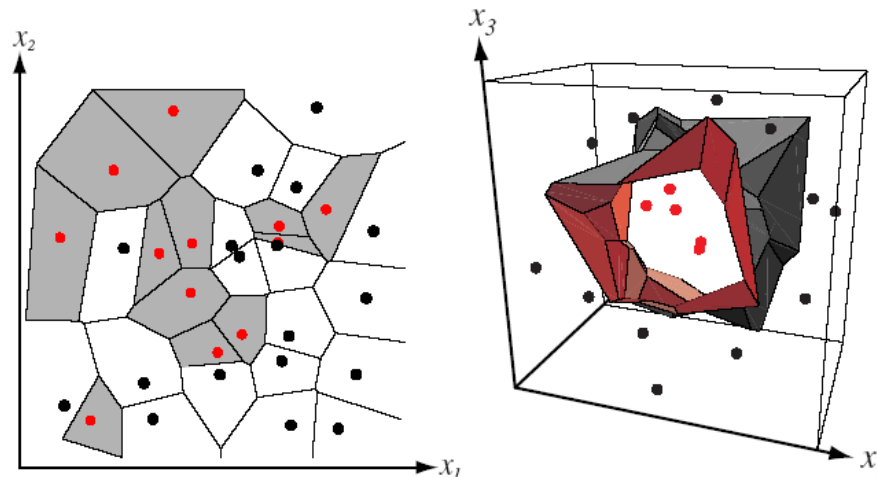


FIGURE 4.13. In two dimensions, the nearest-neighbor algorithm leads to a partitioning of the input space into Voronoi cells, each labeled by the category of the training point it contains. In three dimensions, the cells are three-dimensional, and the decision boundary resembles the surface of a crystal. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.



Parzen-Fenster

23. Juni 2017

Literatur:

Parzen, Emanuel. On Estimation of a Probability Density Function and Mode. Ann. Math. Statist. 33 (1962), no. 3, 1065-1076.

<http://projecteuclid.org/euclid.aoms/1177704472>