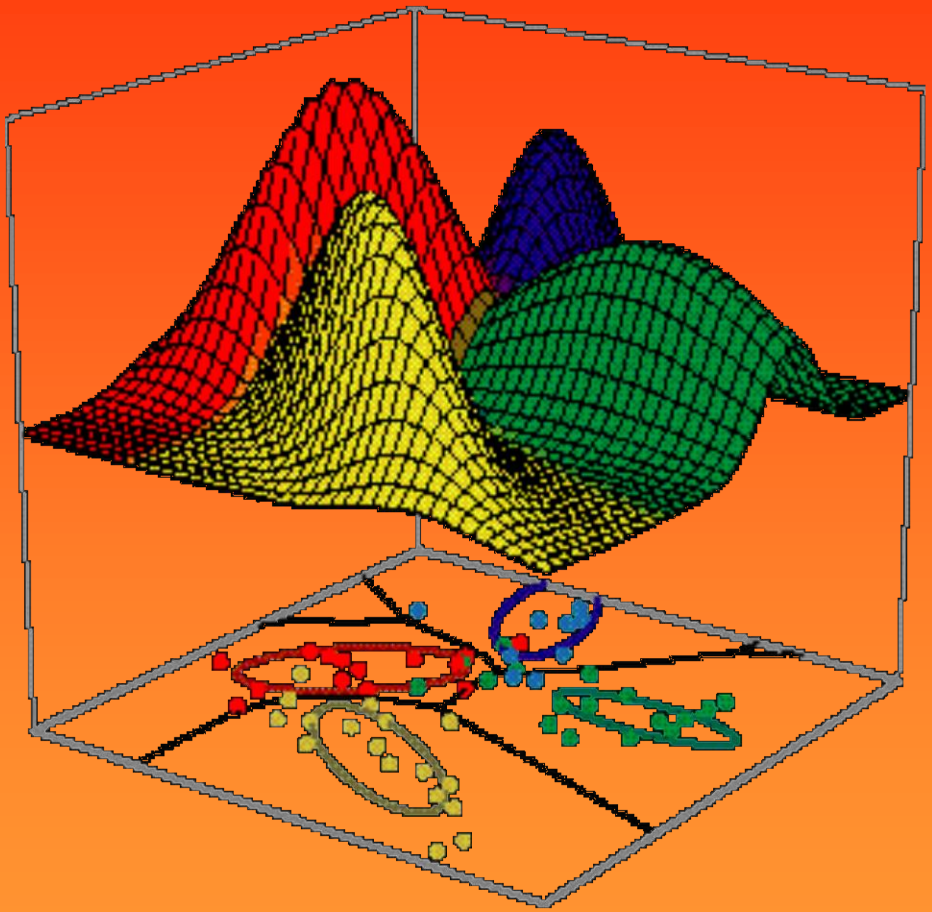

Pattern Classification

All materials in these slides were taken from *Pattern Classification (2nd ed) by R. O. Duda, P. E. Hart and D. G. Stork, John Wiley & Sons, 2000* with the permission of the authors and the publisher

Capitulo 4 (parte 3):
Clasificacion no parametrica
(Secciones 4.6-4.8)

k –Nearest Neighbor Estimation

and metrics

- La regla del vecino mas cercano
  - clasifica x asignado la etiqueta de la clase mejor representada, usando un esquema de votacion

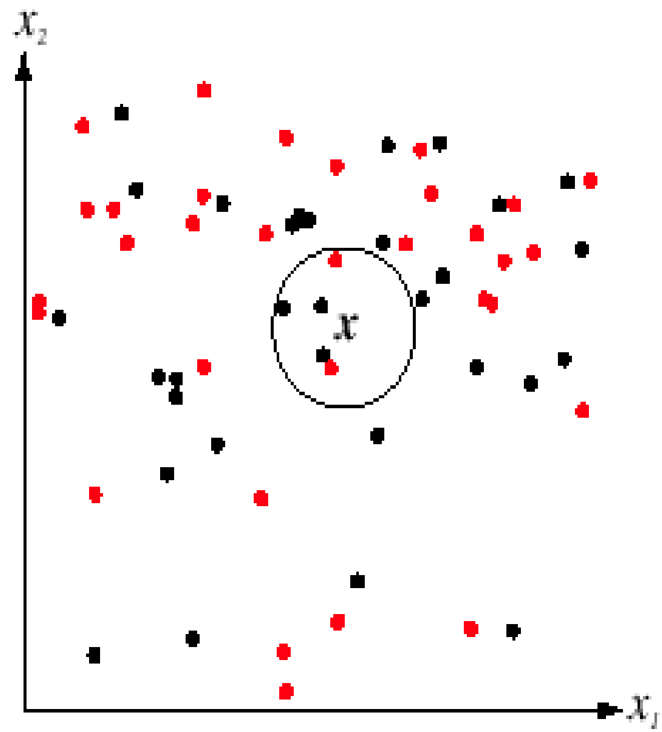

FIGURE 4.15. The *k*-nearest-neighbor query starts at the test point **x** and grows a spherical region until it encloses *k* training samples, and it labels the test point by a majority vote of these samples. In this $k = 5$ case, the test point **x** would be labeled the category of the black points. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Metricas

- Este tipo de clasificador depende de una funcion de distancia o métrica.
- Una métrica es una funcion de dos vectores que cumple las siguientes propiedades
  - nonegatividad: $D(a,b) \geq 0$
  - reflexion: $D(a,b)=0$ sii $a=b$
  - simetria: $D(a,b)=D(b,a)$
  - desigualdad triangular: $D(a,b)+D(b,c) \geq D(a,c)$

Escalar cambia la clasificacion

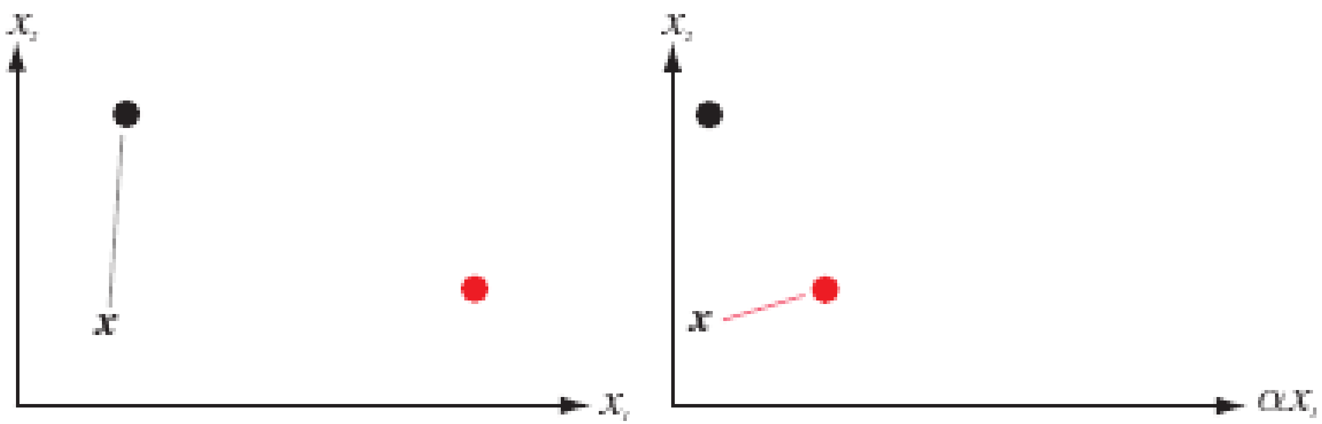

FIGURE 4.18. Scaling the coordinates of a feature space can change the distance relationships computed by the Euclidean metric. Here we see how such scaling can change the behavior of a nearest-neighbor classifer. Consider the test point **x** and its nearest neighbor. In the original space (left), the black prototype is closest. In the figure at the right, the x_1 axis has been rescaled by a factor 1/3; now the nearest prototype is the red one. If there is a large disparity in the ranges of the full data in each dimension, a common procedure is to rescale all the data to equalize such ranges, and this is equivalent to changing the metric in the original space. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Metrica de Minkowski

- Una clase general de métricas para patrones d-dimensionales es la metrica de Minkowski

$$L_p(a,b) = \left(\sum_{i=1}^{d} | a_i - b_i |^p \right)^{1/p}$$

tambien llamada norma Lp.

- La norma L_1 es llamada distancia city block o Manhattan, y mide camino mas corto entre dos puntos a y b usando solo segmentos paralelos a los ejes.
- La norma L_2 es la norma Euclidea
- La norma L_∞ entre a y b corresponde al supremo de la diferencia entre a y b, coordenada a coordenada.

Metrica de Minkowski

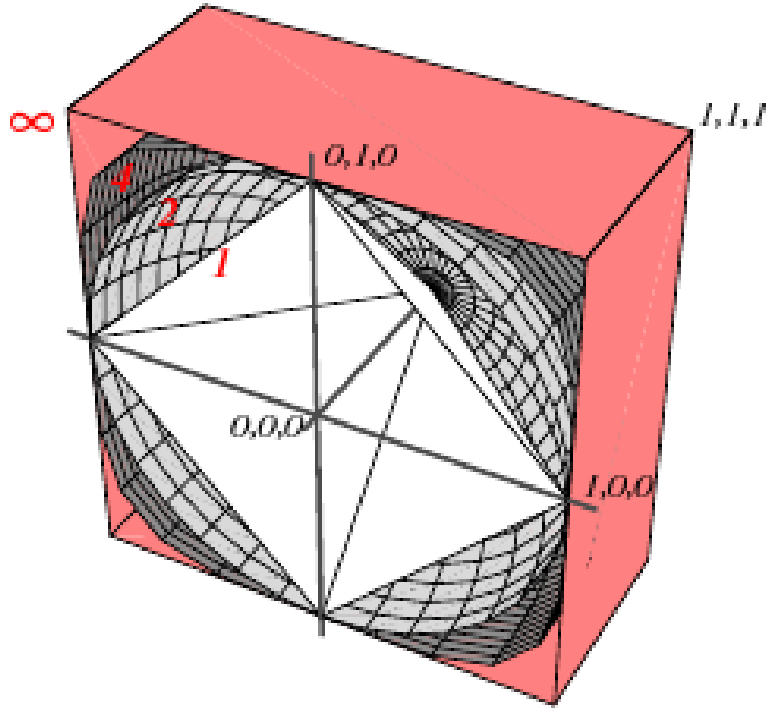

FIGURE 4.19. Each colored surface consists of points a distance 1.0 from the origin, measured using different values for *k* in the Minkowski metric (*k* is printed in red). Thus the white surfaces correspond to the L_1 norm (Manhattan distance), the light gray sphere corresponds to the L_2 norm (Euclidean distance), the dark gray ones correspond to the L_4 norm, and the pink box corresponds to the L_∞ norm. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Métrica Tanimoto

- La distancia entre dos conjuntos S_1 y S_2 se define como

$$D(S_1, S_2) = \frac{n_1 + n_2 - 2n_{12}}{n_1 + n_2 - n_{12}}$$

donde n1 y n2 son el numero de elementos de S_1 y S_2 respectivamente, y n_{12} es el numero de elementos de la interseccion.

- Se usa cuando las caracteristicas (elementos de los subconjuntos) o bien son iguales o bien distintas, y no hay nocion de grado de similaridad

Invariantes

- k-vecinos mas cercanos precisan métricas invariantes, las cuales no son universales
- cada problema tiene transformaciones asociadas, a las cuales la medida debe ser invariante
- transformaciones comunes son:
  - rotaciones
  - traslaciones
  - distorsiones
  - escalado

Distancia euclidea

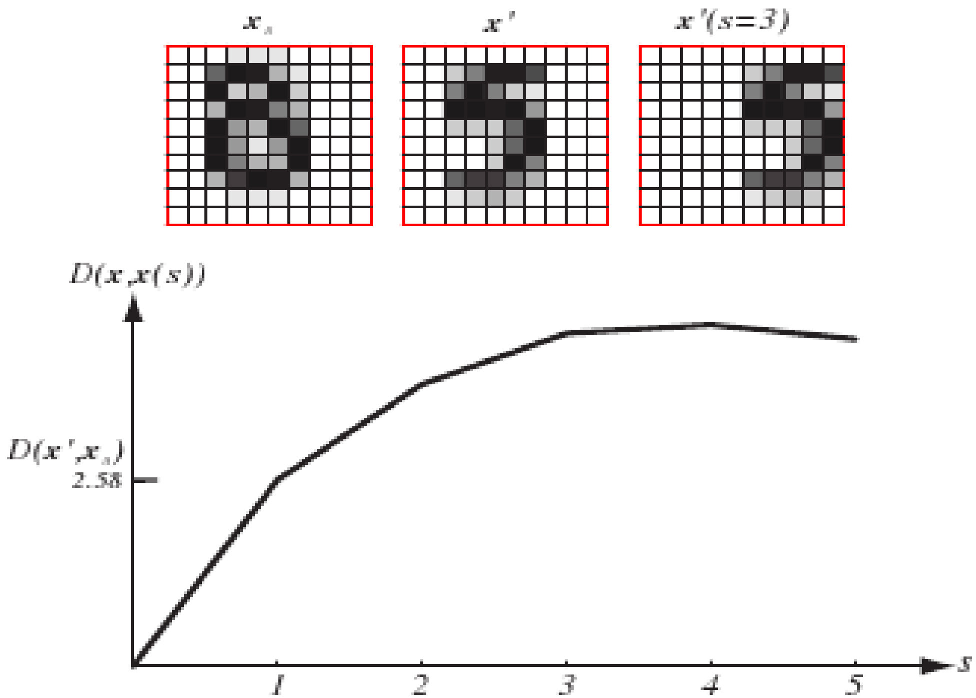

FIGURE 4.20. The uncritical use of Euclidean metric cannot address the problem of translation invariance. Pattern $\mathbf{x}'$ represents a handwritten 5, and $\mathbf{x}'(s = 3)$ represents the same shape but shifted three pixels to the right. The Euclidean distance $D(\mathbf{x}', \mathbf{x}'(s = 3))$ is much larger than $D(\mathbf{x}', \mathbf{x}_8)$, where $\mathbf{x}_8$ represents the handwritten 8. Nearest-neighbor classification based on the Euclidean distance in this way leads to very large errors. Instead, we seek a distance measure that would be insensitive to such translations, or indeed other known invariances, such as scale or rotation. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

soluciones

- se pueden procesar los datos para ponerlos en pose
- esto es muy costoso
- en imágenes, se realiza cuando hay que fusionar datos para determinar cambios o generar tres dimensiones
- para clasificacion, es indispensable trabajar con una distancia invariante

Clasificador basado en distancia tangente

- Se supone que se conocen r transformaciones para las cuales se tiene que obtener invariancia
- Para cada vector de entrenamiento x, se calcula $F_i(x)$ la transformacion de x (para cada transformacion diferente)
- Se construye el vector tangente TV de cada transformacion

$$TV_i=F_i(x)-x$$

- El subespacio de los r vectores tangentes que pasan por x representa una aproximacion lineal de la combinacion de las transformaciones.

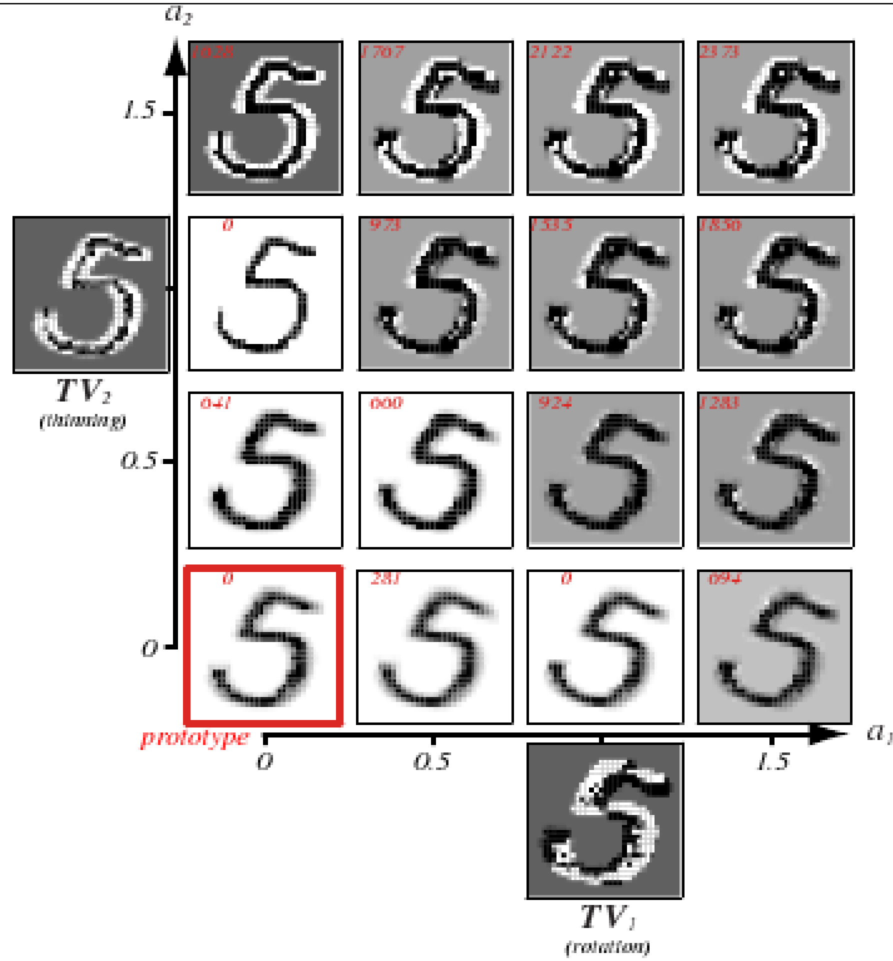

FIGURE 4.21. The pixel image of the handwritten 5 prototype at the lower left was subjected to two transformations, rotation, and line thinning, to obtain the tangent vectors $\mathbf{TV}_1$ and $\mathbf{TV}_2$; images corresponding to these tangent vectors are shown outside the axes. Each of the 16 images within the axes represents the prototype plus linear combination of the two tangent vectors with coefficients a_1 and a_2. The small red number in each image is the Euclidean distance between the tangent approximation and the image generated by the unapproximated transformations. Of course, this Euclidean distance is 0 for the prototype and for the cases $a_1 = 1, a_2 = 0$ and $a_1 = 0, a_2 = 1$. (The patterns generated with $a_1 + a_2 > 1$ have a gray background because of automatic grayscale conversion of images with negative pixel values.) From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Cómputo distancia

- Para calcular la distancia tangente de un prototipo x a un nuevo vector y, se usa la matriz T de todos los r vectores tangentes a x

$$D(x, y) = \min_{a} \left\| (x + Ta) - y \right\|$$

- Esto es, la distancia euclidea entre el vector y al espacio tangente de x.
- Se calculan los k-vecinos mas cercanos (con esta distancia) y se vota en el entorno apra dar la etiqueta.

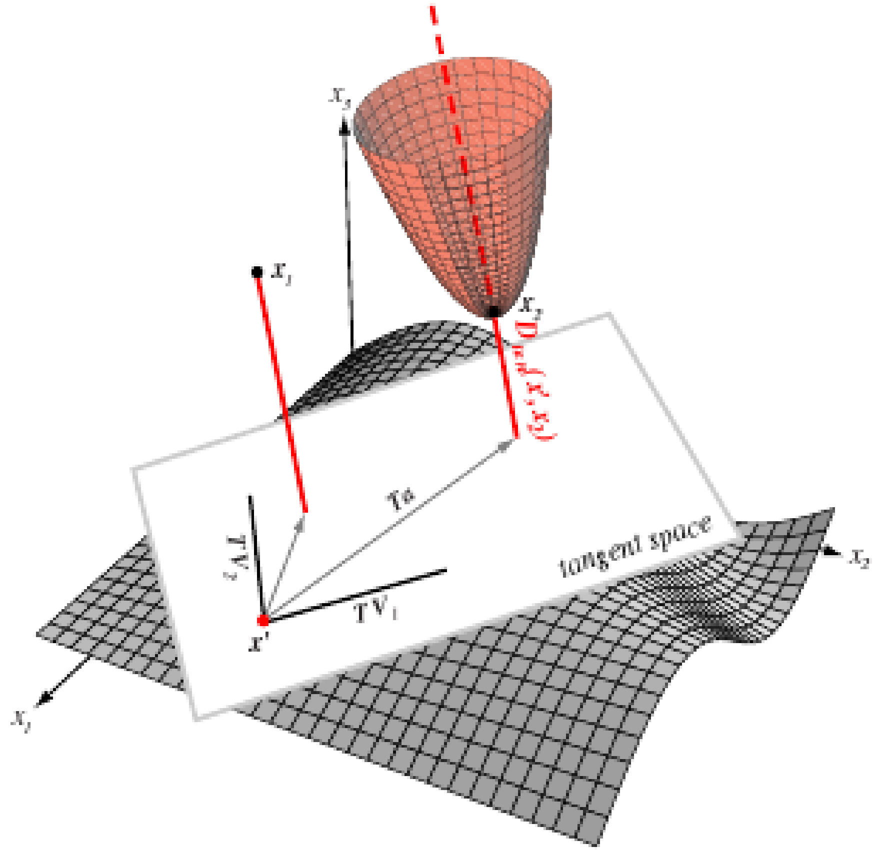

FIGURE 4.22. A stored prototype $\mathbf{x}'$, if transformed by combinations of two basic transformations, would fall somewhere on a complicated curved surface in the full d-dimensional space (gray). The tangent space at $\mathbf{x}'$ is an r-dimensional Euclidean space, spanned by the tangent vectors (here $\mathbf{TV}_1$ and $\mathbf{TV}_2$). The tangent distance $D_{tan}(\mathbf{x}', \mathbf{x})$ is the smallest Euclidean distance from $\mathbf{x}$ to the tangent space of $\mathbf{x}'$, shown in the solid red lines for two points, $\mathbf{x}_1$ and $\mathbf{x}_2$. Thus although the Euclidean distance from $\mathbf{x}'$ to $\mathbf{x}_1$ is less than that to $\mathbf{x}_2$, for the tangent distance the situation is reversed. The Euclidean distance from $\mathbf{x}_2$ to the tangent space of $\mathbf{x}'$ is a quadratic function of the parameter vector $\mathbf{a}$, as shown by the pink paraboloid. Thus simple gradient descent methods can find the optimal vector $\mathbf{a}$ and hence the tangent distance $D_{tan}(\mathbf{x}', \mathbf{x}_2)$. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Clasificacion fuzzy

- Hay una gran discusión acerca de que si la logica difusa es otra forma de ver probabilidades, o es una nueva forma de caracterizar la informacion
- Los clasificadores basados en logica difusa no se adaptan bien a grandes dimensiones, y son muy pobres cuando hay falta de normalizacion.
- No usan datos de entrenamiento, por lo cual, cuando fallan, es costumbre adaptarlos a sistemas llamados Neuro-fuzzy, que son neural networks con patrones caracterizados por logica difusa. Anfis es uno de sus ejemplos mas importantes

Reduced Coulumb energy network

- Metodo de Parzen fija una ventana para todo el espacio de caracteristicas.
- Sin embargo, en algunas regiones un ancho de banda mas chico seria mas eficiente mientras que en otras regiones uno mas grande seria apropiado.
- K-nearest neighbors reduce ese problema considerando regiones ajustadas a la densidad de puntos.
- Un punto de vista intermedio seria ajustar la ventana durante el entrenamiento de acuerdo al punto mas cercano de una categoria diferente.

Reduced Coulumb energy network

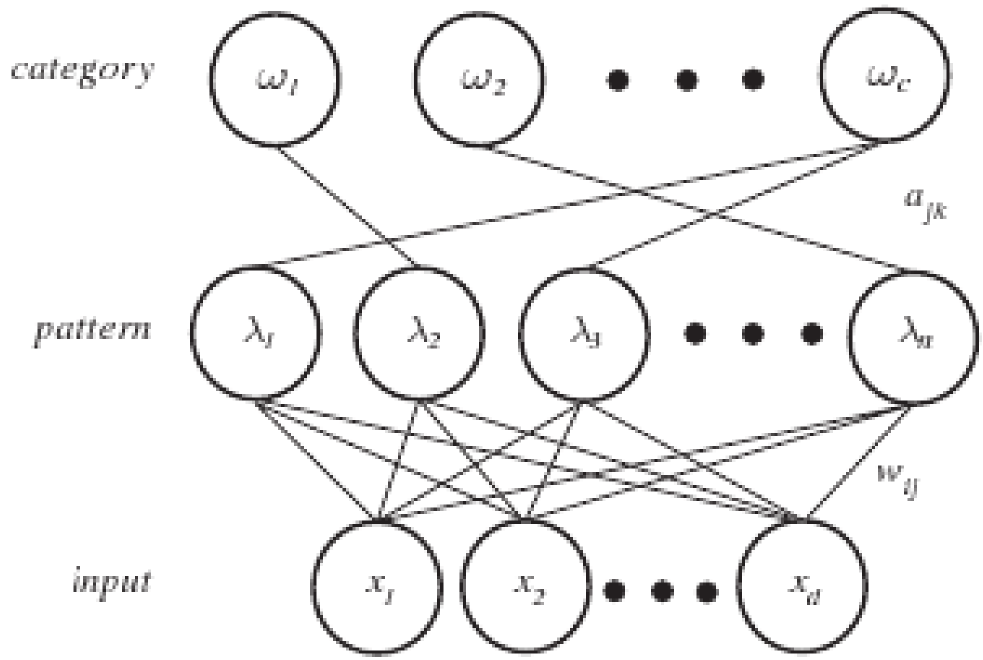

FIGURE 4.25. An RCE network is topologically equivalent to the PNN of Fig. 4.9. During training, normalized weights are adjusted to have the same values as the normalized pattern presented, just as in a PNN. In this way, distances can be calculated by an inner product. Pattern units in an RCE network also have a modifiable threshold corresponding to a "radius" λ. During training, each threshold is adjusted so that its radius is as large as possible without containing training patterns from a different category. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

algoritmos

Algorithm 4 (RCE training)

begin initialize $j = 0, n = \#\text{patterns}, \epsilon = \text{small param}, \lambda_m = \text{max radius}$

  do $j \leftarrow j + 1$

    train weight: $w_{jk} \leftarrow x_k$

    find nearest pt not in ω_i: $\hat{\mathbf{x}} \leftarrow \arg\min_{\mathbf{x} \notin \omega_i} D(\mathbf{x}, \mathbf{x}')$

    set radius: $\lambda_j \leftarrow Min[D(\hat{\mathbf{x}}, \mathbf{x}') - \epsilon, \lambda_m]$

    if $\mathbf{x} \in \omega_i$ **then** $a_{ic} \leftarrow 1$

    until $j = n$

end

begin initialize $j = 0, k = 0, \mathbf{x} = \text{test pattern}, \mathcal{D}_t = \{\}$

  do $j \leftarrow j + 1$

    if $D(\mathbf{x}, \mathbf{x}'_j) < \lambda_j$ **then** $\mathcal{D}_t \leftarrow \mathcal{D}_t \cup \mathbf{x}'_j$

    until $j = n$

    if cat of all $\mathbf{x}'_j \in \mathcal{D}_t$ is the same **then return** label of all $\mathbf{x}_k \in \mathcal{D}_t$

      else return "ambiguous" label

  end

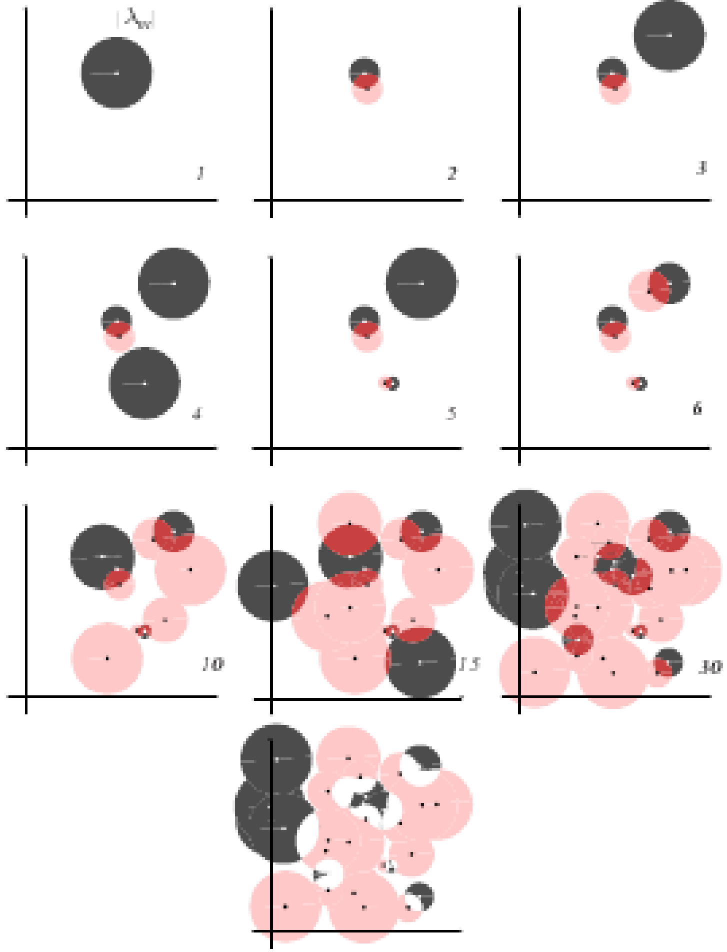

FIGURE 4.26. During training of an RCE network, each pattern has a parameter—equivalent to a radius in the *d*-dimensional space—that is adjusted to be as large as possible without enclosing any points from a different category (up to a maximum λ_m). As new patterns are presented, each such radius is decreased so that no sphere encloses a pattern of a different category. In this way, each sphere can enclose only patterns having the same category label. In this figure, the regions corresponding to one category are pink, and the other category are gray. Ambiguous regions (those enclosed by spheres of both categories) are shown in dark red. The number of points is shown in each component figure. The figure at the bottom shows the final decision regions, colored by category. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.