Aprendizaje Adaptativo Mediante Vecinos Envolventes

J.S. Sánchez[1], F. Pla[1], F.J. Ferri[2]

[1]Departament d'Informàtica. Universitat Jaume I
E-12071 Castelló. SPAIN
[2]Institut de Robòtica. Universitat de València
E-46100 Burjassot (València). SPAIN
e-mail: {sanchez,pla}@uji.es, ferri@uv.es

Resumen

En este artículo se presentará un algoritmo de aprendizaje adaptativo para la regla de decisión del Vecino más Próximo (VP) o *Nearest Neighbour* (NN). A diferencia del conocido método LVQ, el esquema aquí propuesto seleccionará k vectores que se encuentren envolviendo a una muestra en sus proximidades. Este trabajo pondrá especial énfasis en tratar de mejorar los resultados obtenidos por los clásicos algoritmos LVQ al utilizar un escaso número de vectores. Finalmente, se proporcionará también un análisis empírico sobre la aplicación de diferentes técnicas de aprendizaje, demostrando la eficiencia del algoritmo propuesto.

1 Introducción

La regla VP [1] constituye uno de los clasificadores más ampliamente utilizados dentro del reconocimiento estadístico de patrones. En este caso, cada clase aparece como un conjunto de n prototipos previamente etiquetados o vectores (conjunto de entrenamiento), de modo que una nueva muestra de entrada resultará clasificada en función de la etiqueta de clase de su VP de entre todos los prototipos del conjunto de entrenamiento.

Además de otras ventajas comunes a la mayor parte de las aproximaciones no paramétricas, la regla VP y su extensión a k vecinos (o regla k-VP, en la cual los k vecinos más próximos "votan" por la etiqueta de la muestra de entrada) combinan su simplicidad conceptual y de implementación con el hecho de que su error asintótico (es decir, cuando $n \rightarrow \infty$) se encuentra convenientemente acotado en términos del error óptimo de Bayes cuando $k \rightarrow \infty$. Sin embargo, en la práctica, no siempre será posible alcanzar el resultado asintótico esperado debido al número relativamente pequeño de prototipos disponibles. Por otra parte, si se dispone de un conjunto de entrenamiento suficientemente grande, entonces el principal problema se refiere a la complejidad computacional que conlleva el importante número de distancias a calcular.

Con el objetivo de paliar estos inconvenientes de índole computacional, se han llevado a cabo numerosas investigaciones. De entre las diferentes propuestas, merece la pena mencionar aquellas que tratan de obtener un esquema de clasificación más eficiente mediante la reducción de la talla del conjunto de entrenamiento. Dentro de esta estrategia general, una primera posibilidad, denominada *edición-condensado*, consiste en seleccionar un subconjunto suficientemente pequeño de vectores que tienda a obtener aproximadamente los mismos resultados que la regla VP aplicada sobre la totalidad del conjunto de entrenamiento [2]. Otras técnicas tratan de generar un conjunto reducido de vectores que serán desplazados hacia posiciones óptimas para

garantizar una tasa de error baja. Las aproximaciones más representativas de este segundo grupo corresponden a los algoritmos de aprendizaje competitivo, generalmente referidos como LVQ [3].

La técnica que se propone en este artículo también consistirá en la generación de un reducido conjunto de prototipos, a semejanza del método LVQ. No obstante, mientras los algoritmos LVQ únicamente se centran en el primer VP, algunos trabajos recientes [4] han mostrado que el hecho de considerar un mayor número de vecinos durante el proceso de aprendizaje puede llevar a una significativa mejora en los resultados de la clasificación. De una manera similar, este artículo introduce algunas alternativas a los esquemas basados en el método LVQ a partir de la aplicación de un concepto de vecindad sensiblemente distinto, el cual ha dado ya resultados satisfactorios en diversos problemas de clasificación [5].

2 Los algoritmos LVQ

El objetivo del método LVQ se centra en definir regiones de clase óptimas en el espacio de representación de las muestras: inicialmente, se asigna un primer subconjunto de vectores a cada clase del problema y, posteriormente, estos son situados en el espacio de representación de tal manera que la regla VP minimice la probabilidad del error de clasificación medio esperado. De este modo, el correspondiente proceso de aprendizaje consiste en acercar progresivamente algunos de los vectores próximos a una muestra de entrenamiento hacia ella o alejarlos de ella en función del resultado de la regla VP.

2.1 El esquema LVQ1

Supongamos que se ha dispuesto un determinado número inicial de vectores en el espacio de representación. Sea $x(t)$ una muestra de entrada y supongamos que $m_i(t)$ representa secuencias de su VP en el tiempo. En pocas palabras, el proceso de aprendizaje correspondiente al algoritmo LVQ básico consiste en actualizar la posición de m_i. Si la etiqueta de clase del vector m_i coincide con la de la muestra de entrenamiento x, entonces dicho vector deberá ser desplazado hacia x. En caso contrario, aquel vector deberá ser alejado de la muestra. Las modificaciones sobre el vector m_i serán llevadas a cabo atendiendo a la siguiente regla [3]:

$$m_i(t+1) = m_i(t) \pm \alpha(t)[x(t) - m_i(t)] \tag{1}$$

donde $0 < \alpha(t) < 1$ denota la correspondiente velocidad de aprendizaje.

2.2 El esquema LVQ2

En este caso, dos vectores, m_i y m_j, que hacen referencia a los VPs de una muestra x, serán actualizados simultáneamente. Uno de ellos debe pertenecer a la clase correcta y el otro a la errónea. Además, x deberá encontrarse dentro de una zona de valores o "ventana", que estará definida alrededor del semiplano de m_i y m_j. Los reajustes en el algoritmo LVQ2 pueden expresarse del modo siguiente [3]:

$$\begin{aligned} m_i(t+1) &= m_i(t) + \alpha(t)[x(t) - m_i(t)] \\ m_j(t+1) &= m_j(t) - \alpha(t)[x(t) - m_j(t)] \end{aligned} \tag{2}$$

2.3 El esquema LVQ3

El algoritmo LVQ2 será ahora mejorado con el fin de incorporar correcciones que aseguren que los m_i continuan aproximándose a la distribución de clases incluso en el caso de que el proceso de aprendizaje llegue a ser excesivamente largo. Las modificaciones en el algoritmo LVQ3 serán realizadas según las mismas condiciones que en el esquema LVQ2. Además, si x, m_i y m_j pertenecen a la misma clase, entonces se definirá la siguiente regla de aprendizaje [3]:

$$m_k(t+1) = m_k(t) + \epsilon\alpha(t)[x(t) - m_k(t)] \tag{3}$$

para $k \in \{i, j\}$. El valor óptimo de ϵ, el cual denota la velocidad de aprendizaje relativa, depende del tamaño de la ventana [3].

2.4 El esquema LVQ1 optimizado(OLVQ1)

El algoritmo LVQ1 básico puede extenderse de manera que se asigne una velocidad de aprendizaje distinta $\alpha_i(t)$ a cada m_i. Así, el proceso de aprendizaje podrá definirse como sigue [3]:

$$m_i(t+1) = m_i(t) \pm \alpha_i(t)[x(t) - m_i(t)] \tag{4}$$

Además, los valores "óptimos" de $\alpha_i(t)$ para una rápida convergencia de la Ecuación (4) se calculan en [3] mediante la recursion

$$\alpha_i(t) = \frac{\alpha_i(t-1)}{1 \pm \alpha_i(t-1)} \tag{5}$$

3 Las reglas k-VP con aprendizaje

En [4], Laaksonen sugiere un conjunto de reglas de aprendizaje adaptativo a partir de un mayor refinamiento del algoritmo LVQ1 básico. La idea general sobre la que se basan estas aproximaciones (denominadas k-VP con aprendizaje o *Learning k-NN*) consiste en utilizar un cierto número k de vectores más cercanos a la muestra de entrenamiento para obtener, presumiblemente, un incremento en la correspondiente efectividad.

En concreto, en [4] se proponen tres reglas de aprendizaje con ligeras diferencias. Así, en el primer esquema, se desplazarán todos los k vectores más próximos a la muestra de entrada. Por el contrario, la segunda aproximación consiste en mover sólo el k-ésimo y el $k+1$-ésimo vectores más próximos si el intercambio del orden entre ellos hiciera variar la clasificación de la muestra de entrenamiento de incorrecta a correcta. Finalmente, mediante la tercera regla de aprendizaje, se actualizará la posición de todos los $k+1$ vectores más próximos. Cabe añadir que las modificaciones sobre los vectores serán llevadas a cabo, en todos los casos, según la regla de aprendizaje de la Ecuación (1).

A partir del análisis empírico realizado en [4], parece que el algoritmo LVQ1 estándar funciona generalmente mejor que cualquiera de las reglas k-VP con aprendizaje. Sin embargo, estas extensiones a k vecinos obtienen tasas de error menores que el esquema LVQ1 básico cuando se dispone de un elevado número de vectores y, especialmente, en el caso de la primera regla propuesta. De hecho, estos resultados están en consonancia con el correspondiente análisis teórico sobre el caso asintótico para la regla de decisión k-VP [2], en el sentido de que dicha regla únicamente puede alcanzar los resultados de clasificación óptimos cuando existe un número suficientemente grande de prototipos.

4 Una alternativa a los algoritmos LVQ

El objetivo final de las reglas k-VP con aprendizaje será mejorar los resultados de los algoritmos LVQ estándar. Estos procedimientos alternativos recogen la idea de que al considerar un mayor número de vectores próximos a la muestra se podrá tender a una tasa de error menor que en el caso de utilizar un único vector próximo, de forma similar a como la regla k-VP es capaz de superar generalmente al clasificador VP simple (1-VP).

A pesar de las expectativas iniciales, el análisis experimental que aparece en [4] demuestra que el incremento de la efectividad sólo se consigue si el tamaño del conjunto de entrenamiento es suficientemente grande. En caso contrario, el resultado de clasificación obtenido por el algoritmo LVQ1 básico es mejor que el correspondiente a las reglas k-VP con aprendizaje. Esto se podría explicar parcialmente por el hecho de que el comportamiento del clasificador que se utiliza en el proceso de aprendizaje (es decir, la regla k-VP) no llega a ser tan bueno como cabría esperar en el caso de un tamaño de muestras finito.

Aplicando una estrategia similar a la de las reglas k-VP con aprendizaje, en esta sección vamos a proponer una nueva extensión al método LVQ. Esta aproximación estará basada en la aplicación de un concepto de vecindad alternativo, denominado Vecindad de Centroide más Próximo o *Nearest Centroid Neighbourhood* [6], el cual se ha utilizado ya con éxito en diferentes problemas de clasificación sobre conjuntos de talla pequeña [5].

4.1 Vecindad de centroide más próximo

El concepto de vecindad de centroide más próximo hace referencia básicamente a la idea de que la vecindad de un punto se encuentra sujeta a dos restricciones complementarias. En primer lugar, por el *criterio de distancia*, los k vecinos de una muestra p deberán estar tan próximos a ella como sea posible. En segundo lugar, mediante el *criterio de simetría*, su centroide también deberá encontrarse tan próximo a p como sea posible. Obsérvese que el tradicional concepto de vecindad más próxima utilizado por la reglas k-VP y los algoritmos LVQ sólo tiene en cuenta la primera de aquellas propiedades, de modo que puede ocurrir que los vectores más próximos no estén distribuidos alrededor de la muestra p.

Sea p un punto del que debemos encontrar sus k vecinos de centroide más próximo (VCP) o *nearest centroid neighbours* (NCN) en un cierto conjunto de entrenamiento, $X = \{m_1, \ldots, m_n\}$. Estos k vecinos pueden obtenerse mediante un procedimiento iterativo [6] de la siguiente manera:

1. El primer VCP a p se corresponde con su VP, q_1.
2. El i-ésimo vecino, q_i, $i \geq 2$, será tal que el centroide entre él y todos los vecinos previamente seleccionados, $q_1, \ldots, q_i$ resulte el más cercano a p.

Este procedimiento dará lugar a un tipo de vecindad en el que se tiene en cuenta la distribución espacial de los vecinos, debido al criterio de centroide que se está utilizando. Además, se garantiza la proximidad de los k VCPs a la muestra por la naturaleza incremental del algoritmo por la que se obtienen a partir del primer VP. Mediante esta definición de vecindad, es posible establecer una regla de clasificación denominada k-VCP o k-NCN [5]:

1. Buscar los k VCPs a p, $X^p = \{m_1^p, \ldots, m_k^p\}$, donde $k \leq n$.
2. Asignar p a la clase con una mayoría de votos entre sus k VCPs en el conjunto X^p (resolver los empates de forma aleatoria).

4.2 Regla k-VCP con aprendizaje

Al igual que cualquiera de los algoritmos LVQ, la aproximación que aquí se propone también utiliza un número fijo de vectores, cuyas posiciones iniciales en el espacio de representación serán posteriormente modificadas con el fin de minimizar el error de clasificación. Cuando se selecciona una muestra de entrenamiento, algunos de sus k vectores más próximos serán desplazados hacia ella y otros serán alejados de ella en función de sus etiquetas de clase. Sin embargo, estos procedimientos difieren esencialmente de las reglas k-VP con aprendizaje en el tipo de vecindad considerada por cada uno de ellos. Así, la alternativa que se sugiere en este trabajo (a partir de ahora denominada k-VCP con aprendizaje o Learning k-NCN) hace uso de la vecindad de centroide más próximo que acabamos de describir, mientras que los esquemas k-VP con aprendizaje están simplemente basados en el tradicional concepto de vecindad por proximidad.

El objetivo básico de todas las reglas VCP se centra en inspeccionar una región suficientemente pequeña *alrededor* de la muestra de entrenamiento. Esto podrá estar garantizado porque la vecindad de centroide más próximo tiene en cuenta tanto la proximidad de los k VCPs como su distribución geométrica con respecto a la muestra de entrenamiento mientras que, en general, la vecindad clásica no la envuelve completamente debido a que sólo está difinida en términos de mínima distancia (Euclídea).

La nueva regla de aprendizaje consistirá en mover todos aquellos vecinos que se encuentren envolviendo a una muestra de entrada. Si la clase de un vector coincide con la de la muestra, dicho vector se aproximará a ella. En caso contrario, desplazaremos el prototipo seleccionado en la dirección opuesta a la muestra. Las correspondientes modificaciones sobre los vectores se realizarán según la Ecuación (1). Para $k = 1$, esta regla será equivalente al esquema LVQ1. Sin embargo, cuando $k > 1$, entonces los VCPs realmente tienden a envolver a la muestra.

5 Experimentos y resultados

En esta sección, se presenta un análisis experimental comparativo entre la regla k-VP, el método LVQ, la regla k-VP con aprendizaje y el esquema k-VCP con aprendizaje que acabamos de introducir. En el caso de las reglas k-VP simple y con aprendizaje y del clasificador k-VCP con aprendizaje, se han probado diferentes valores típicos del parámetro k (variando de 3 a 11), incluyendo en las posteriores figuras sólo el mejor resultado de cada uno de aquellos esquemas.

Se ha utilizado el método de partición promediado sobre cinco particiones aleatorias (50% para entrenamiento del clasificador y 50% para test) sobre la base de datos original para obtener estimaciones de las correspondientes tasas de error. Además, cabe decir que la ejecución de aquellos algoritmos se ha repetido diez veces sobre cada partición. Debe también mencionarse que el número relativo de prototipos por clase dentro de cada partición ha sido seleccionado de tal manera que se mantengan las probabilidades a priori de las clases en el conjunto inicial.

Un tema de especial interés para este tipo de clasificadores se refiere al tamaño del conjunto de vectores, es decir, al número de prototipos inicialmente ubicados en el espacio de representación. Por tanto, este análisis comparativo se centrará fundamentalmente en la influencia de la talla de dicho conjunto sobre la tasa de error resultante al utilizar cada uno de los algoritmos. El número de iteraciones en el proceso de aprendizaje será 50 veces el número total de vectores disponibles, mientras que la velocidad de aprendizaje será constante ($\alpha(t) = \pm 0.2$) para todos los experimentos. El esquema LVQ se aplicará de acuerdo con el método tradicionalmente utilizado

[7], es decir, el aprendizaje comenzará con el OLVQ1, el cual converge rápidamente, y luego se utilizarán los algoritmos báicos.

5.1 Una base de datos sintéticos

Los experimentos que acabamos de describir han sido llevados a cabo sobre algunas bases de datos sintéticos y reales [8]. El primer problema que aquí se presenta trata de distinguir entre una distribución normal unimodal y una distribución gaussiana multimodal (resultado de la suma de tres distribuciones normales distintas), con un elevado grado de solapamiento entre dichas clases. El conjunto original consta de un toal de 5.000 prototipos (50% de cada clase).

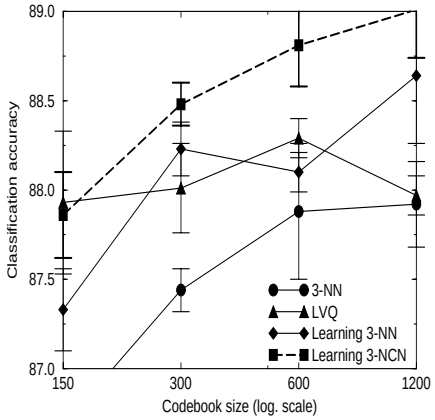

Figura 1: *Tasa de aciertos respecto a la talla del conjunto.*

Como se puede ver en la Figura 1, el algoritmo aquí propuesto se comporta claramente mejor que cualquier otro esquema. Al considerar 150 vectores en el espacio de representación, el método LVQ y el nuevo esquema de aprendizaje obtienen unos resultados muy similares. Sin embargo, a partir de 300 prototipos, las diferencias entre la tasa de aciertos de la regla k-VCP con aprendizaje y la del resto de aproximaciones llegan a ser estadísticamente significativas.

5.2 Un problema real

Este conjunto de datos reales trata de determinar la presencia o ausencia de diabetes entre hembras mayores de 21 años en una comunidad de indios, de acuerdo a ocho atributos cuantitativos. La base de datos original contiene un total de 768 prototipos: 500 de la clase 0 y 268 de la clase 1 (el valor 1 debe aquí interpretarse como "test de diabetes positivo").

Los resultados representados en la Figura 2 son bastante similares a los obtenidos para los datos sintéticos. La mejor opción en la mayor parte de los casos corresponde al algoritmo propuesto en este trabajo. Además, obsérvese que la tasa de aciertos de la regla k-VCP con aprendizaje sobre el conjunto con 100 muestras es incluso superior a la de cualquier otro algoritmo aplicado sobre el conjunto de 300 vectores.

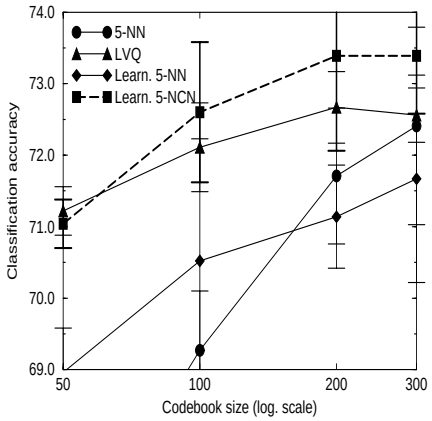

Figura 2: *Resultado de clasificación comparado con el número de vectores.*

La Figura 3 permite comparar la efectividad de las reglas k-VP simple y con aprendizaje respecto a la del esquema k-VCP con aprendizaje, utilizando diferentes valores del parámetro k sobre el conjunto de 100 vectores. El hecho más destacado se refiere a que el algoritmo aquí propuesto presenta un comportamiento considerablemente mejor que las otras dos alternativas. En particular, las diferencias más significativas entre las tres aproximaciones se obtienen con $k = 5$: concretamente, mientras que la regla k-VCP con aprendizaje alcanza un 72'60% de aciertos, los resultados de los clasificadores k-VP estándar y con aprendizaje sólo llegan al 69'27% y 70'52%, respectivamente.

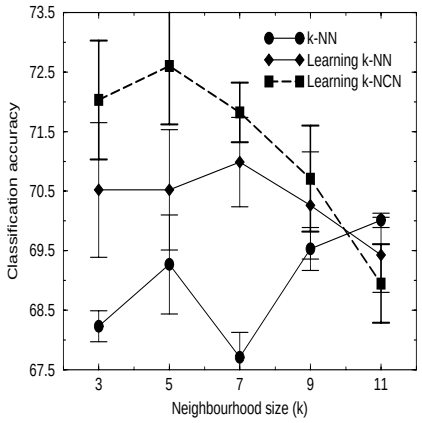

Figura 3: *Tasa de aciertos de los clasificadores en función de la talla de vecindad.*

6 Conclusiones y trabajo futuro

En este trabajo se ha presentado un algoritmo de aprendizaje adaptativo denominado k-VCP con aprendizaje. Este método hace uso del concepto de vecindad de centroide más próximo, en el que se calculan los k vecinos de una muestra no sólo en términos de proximidad sino también teniendo en cuenta su distribución geométrica alrededor de dicha muestra. El principal propósito de la aplicación de este concepto sobre problemas de aprendizaje consiste en obtener un determinado tipo de información que nos permita estimar con mayor precisión la posición de los vectores y, muy especialmente, en el caso de conjuntos pequeños. Los experimentos que se han llevado a cabo muestran un mejor comportamiento de esta aproximación que el de otros esquemas clásicos.

A parte de asociar una velocidad de aprendizaje distinta $\alpha_i(t)$ a cada m_i (tal como lo hace el algoritmo OLVQ1), son también posibles otras extensiones al esquema de aprendizaje aquí propuesto. En particular, a semejanza de la regla k-VP [2], este modelo también permite una opción de rechazo para descartar aquellas muestras cuyos k VCPs no pertenezcan mayoritariamente a una misma clase.

Agradecimientos

Este trabajo ha estado parcialmente financiado por los proyectos P1B98-03 de la Fundació Caixa Castelló - Bancaixa y GV98-14-134 de la Generalitat Valenciana.

Referencias

[1] R. Duda and P.E. Hart, Pattern Classification and Scene Analysis, John Wiley & Sons: New York, 1973.

[2] P.A. Devijver and J. Kittler, Pattern Recognition: A Statistical Approach, Prentice Hall: Englewood Cliffs, NJ, 1982.

[3] T. Kohonen, Self-Organizing Maps, Springer-Verlag: Berlin, Germany, 1995.

[4] J. Laaksonen and E. Oja, "Classification with Learning k-Nearest Neighbors", in Proc. Int. Conf. Neural Networks, pp. 1480-1483, 1996.

[5] J.S. Sánchez, F. Pla and F.J. Ferri, "On the use of neighbourhood-based non-parametric classifiers", Pattern Recognition Letters, Vol. 18, pp. 1179-1186, 1997.

[6] B.B. Chaudhuri, "A new definition of neighbourhood of a point in multi-dimensional space", Pattern Recognition Letters, Vol. 17, pp. 11-17, 1996.

[7] T. Kohonen, J. Kangas, J. Laaksonen and K. Torkkola, "LVQ_PAK: The Learning Vector Quantization Program Package", Laboratory of Computer and Information Science, Helsinki University of Technology, Finland, 1992.

[8] P.M. Murphy, UCI Repository of machine learning databases, Department of Information and Computer Science, University of California, Irvine, CA, 1995.