Capítulo 4

Esquemas de Clasificación.

Clasificador Polinomial

El clasificador lineal visto en el apartado anterior, es inadecuado para resolver problemas de clasificación de mayor complejidad que los ejemplos vistos, en particular casos tales como clasificación no lineal y clasificaciones donde la información esta contenida en vectores de dimensiones de dos dígitos.

Para comprender el clasificador no lineal a partir del concepto de función de decisión lineal, se introduce el concepto de funciones de decisión generalizada, lo cual significa aplicar funciones a cada una de las componentes del vector, tomándose estas como las componentes de un nuevo vector. Si al menos alguna de estas funciones es no lineal, se obtiene una función de clasificación no lineal.

Es decir, dado el vector $\mathbf{x} = (x_1, x_2, ..., x_n)'$ podemos transformarlo como

$\mathbf{x}^* = (f_1(x_1), f_2(x_2), ..., f_n(x_n))' = (x_1^*, x_2^*, ..., x_n^*)'$

Donde el superíndice (*) indica las nuevas componentes del vector luego de aplicar las funciones $f_i(x_i)$.

El caso mas simple se da cuando estas funciones son lineales, es decir, cuando las funciones de decisión se definen simplemente como $f_i(x_i) = x_i$, Bajo esta condición se obtiene la expresión ya vista anteriormente $d(\mathbf{x}) = \mathbf{w}_0 \mathbf{x} + w_{n+1}$.

El nivel siguiente en cuanto a complejidad se tiene con funciones cuadráticas. En el caso bidimensional transformamos los patrones según la expresión:

$\mathbf{x}^* = (x_1, x_2)^2 = (x_1^2, x_2^2, x_1x_2, x_1, x_2, 1) = (x_1^*, x_2^*, x_3^*, x_4^*, x_5^*, 1)$

Lo cual puede se explica a partir de la expansión del vector $x = (x_1, x_2, 1)$ y el cálculo de las combinaciones posibles de sus elementos domados de 2 en 2

Con esta transformación la función polinomial de segundo orden toma la forma;

$d(\mathbf{x}^*) = w_1 x_1^2 + w_2 x_2^2 + w_3 x_1 x_2 + w_4 x_1 + w_5 x_2 + w_6 1$

La que podría expresarse en forma lineal como:

$d(\mathbf{x}^*) = \mathbf{w}_0 \mathbf{x}^*$ con $\mathbf{w} = (w_1, w_2, w_3, w_4, w_5, w_6)$

De este modo, lo que se hace es expresar un clasificador no lineal mediante un sistema de ecuaciones lineales, a costa de expandir la dimensión de los vectores patrón.

El caso bidimensional puede generalizarse mediante una expresión que considere todas las

n n(n-1)/2 n ← Términos

combinaciones posibles de los componentes de **x** que forman términos de segundo orden o inferior, es decir:

$$d(x) = \sum_{j=1}^{n} w_{jj} x_j^2 + \sum_{j=1}^{n-1} \sum_{k=j+1}^{n} w_{jk} x_j x_k + \sum_{j=1}^{n} w_j x_j + w_{n+1}$$

En la expresión anterior, el primer término del segundo miembro consta de n términos, el segundo de n(n-1)/2 términos y el tercero de n términos. El total resulta (n + 1) (n + 2) / 2. Esto también nos da el número de parámetros del vector de pesos de **w**.

Comparando esta última expresión con la anterior dada para las funciones discriminante se ve que:

$f_i(\mathbf{x}) = x_p^{\,s} x_q^{\,t}$ con p, q = 1,2...n; y s, t = 0,1

La Ec. anterior sugiere el esquema general para formar funciones polinomiales de decisión de cualquier grado finito:

$f_i(\mathbf{x}) = x_{p1}^{\,s1} x_{p2}^{\,s2} \ldots x_{pr}^{\,sr}$ con $p_1\, p_2 \ldots p_r = 1,2,\ldots n;$ y $s_1\, s_2 \ldots s_r = 0, 1$

Dado que los términos de la última expresión contienen potencias *r* o inferiores es posible expresar las funciones en la siguiente forma recursiva:

$$d^r(x) = \left(\sum_{p1=1}^{n} \sum_{p2=p1}^{n} \cdots \sum_{pr=pr-1}^{n} w_{p1p2\ldots pr}\, x_{p1} x_{q2} \ldots x_{pr} \right) + d^{r-1}(x)$$

Donde r indica el orden de alinealidad y $d^0(x) = w_{n+1}$. Esto provee un método conveniente para computar funciones de decisión polinomiales completas de cualquier orden.

2.3.1. Expresión Matricial de la Aproximación Polinómica

En lugar de considerar solo una función de clasificación polinómica d(**x**), podemos necesitar, ante un problema dado, emplear un número mayor de estas. En este caso cada función de decisión pasa a ser una componente de un vector de salida (Vector de decisión **d,** que expresa el resultado de la clasificación). Cada una de estas funciones toma entonces la forma:

$$d_k(\mathbf{v}) = w_{0,k} + w_{1,k} x_1 + \ldots w_{N,k} x_N + w_{N+1,k} x_1^2 + w_{N+2,k} x_1 x_2 + = \mathrm{w}_k^T . \mathbf{x'(x)}$$

Definiendo dicho vector de decisión **d** que incluye *k* funciones de decisión d_k , se obtiene una forma compacta $\mathbf{d}(\mathbf{x}^*) = \mathbf{w'}\, \mathbf{x}^*(\mathbf{x})$, que es la expresión matricial del clasificador polinómico.

2.3.2. Grados de Libertad

El número de términos necesarios para definir una función de decisión polinomial crece muy rápidamente como función de r –grado del polinomio- y n –dimensión del patrón- . Este valor para una función de decisión esta dado por:

$C_r^{n+r} = \frac{(n+r)!}{r!\,n!}$ que es la combinación de n + r elementos tomados de r en r.[1]

y para r = 2 se calcula como: $\frac{(n+r)(n+r-1)}{2}$

Para separar ϖ clases se requerirán $C_r^{n+r} \times \omega = \frac{(n+r)!}{r!\,n!} \times \omega$ términos.

La tabla siguiente muestra los grados de libertad en función de los valores de n y r.

r n	1	2	3	4
1	2	3	4	5
2	3	6	10	15
3	4	10	20	35
4	5	15	35	56
5	6	21	56	126
6	7	28	84	210
7	8	36	120	330
8	9	45	165	495
9	10	55	220	715
10	11	66	286	1001

Para n = r = 10 se requieren 184.756 coeficientes.

Dado que la cantidad de términos crece con el grado del polinomio, surge una restricción práctica para usar polinomios de tercer orden o superior.

---

1 Lo que también suele escribirse, $C_r^{n+r} = \binom{n+r}{r}$

2.3.3. Esquemas de Procesamiento Paralelo

Un esquema para un clasificador de segundo orden se muestra en la Fig. 2.15. (funciones de decisión cuadráticas y clasificador cuadrático).

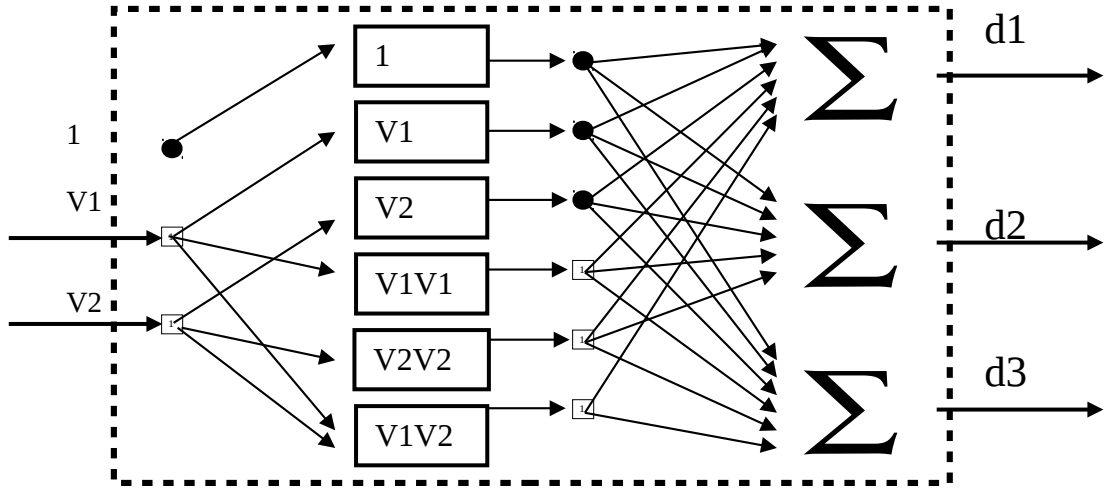


Fig. 2.15: Clasificador Polinomial completo

Este esquema tiene dos capas. La primera computa los productos entre las entradas y la segunda adiciona las salidas de la primera capa. Los elementos en la primera capa se llaman unidades PI y los de la segunda capa se llaman unidades Sigma. Por este motivo la construcción entera también es conocida como red Pi / Sigma.

La inclusión de la estructura alternativa mostrada en la Fig. 2.16 puede ser justificada a partir del hecho de que una

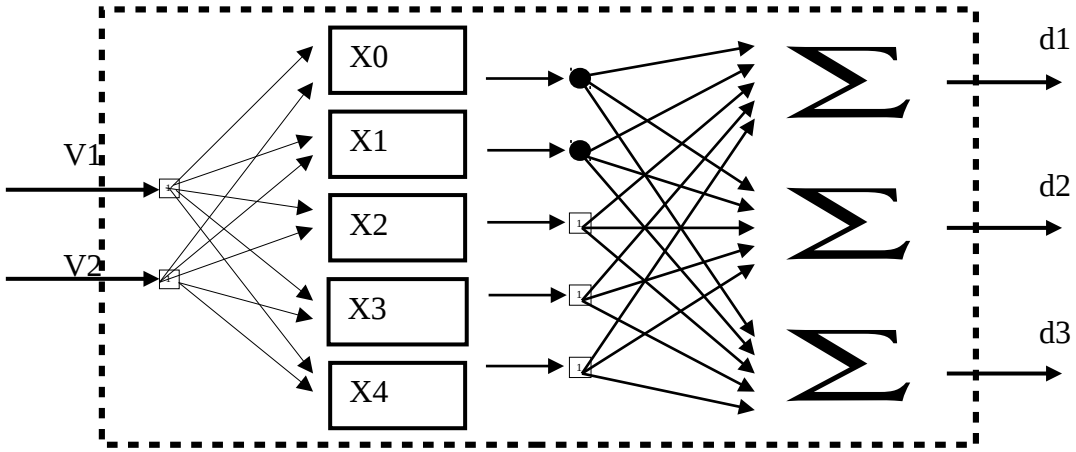


Fig. 2.16: Esquema alternativo de un Clasificador Polinomial

combinación lineal de polinomios es también un polinomio.

2.3.4. Esquema de Cómputo

En principio el cálculo de la matriz de coeficientes de un clasificador polinomial debería hacerse por el método general explicado para el clasificador lineal. Esto es puede hacerse así para un caso simple, sin embargo, el aumento las características de los problemas reales (Altamente no lineales, múltiples subclases y dimensión original de los vectores alta), hacen inviable un cálculo directo de la matriz de coeficientes, por lo que se aplica un procedimiento mas apropiado para calcular dicha matriz cuyas características se describen a continuación.

A los efectos de sistematizar este nuevo método de calculo de la matriz **w**, se introduce un nuevo vector z. Este vector z esta definido uniendo los componentes de x e y.

$$\mathbf{z} = (x_1, x_2, ..., x_n, y_1, y_2, ..., y_m)'$$

Luego se considera una nueva matriz de momentos del vector z:

$$\mathbf{M}_z = E\{ zz^T \} = E\left\{ \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \begin{pmatrix} \mathbf{x}^T & \mathbf{y}^T \end{pmatrix} \right\} = \begin{pmatrix} E\{ \mathbf{x}\mathbf{x}^T \} & E\{ \mathbf{x}\mathbf{y}^T \} \\ E\{ \mathbf{y}\mathbf{x}^T \} & E\{ \mathbf{y}\mathbf{y}^T \} \end{pmatrix}$$

A partir de aquí se puede demostrar que la matriz $\mathbf{M}_z$ puede considerarse formada por cuatro matrices menores, siendo estas la matriz identidad, la matriz de pesos optimizada **w,**
la matriz nula y una cuarta matriz que permite calcular el error residual del clasificador. Es decir que

$$\mathbf{M}_z = \begin{pmatrix} E\{ \mathbf{x}\mathbf{x}^T \} & E\{ \mathbf{x}\mathbf{y}^T \} \\ E\{ \mathbf{y}\mathbf{x}^T \} & E\{ \mathbf{y}\mathbf{y}^T \} \end{pmatrix} = \begin{pmatrix} I & \mathbf{w} \\ 0 & e \end{pmatrix}$$

De aquí surge que para resolver un clasificador polinomial, simplemente debemos construir la matriz $\mathbf{M}_z$ (**x**, **y**) y aplicar transformaciones por filas (Aplicando el método de Gauss-Jordan, en el cual los elementos de la diagonal principal son transformados a la unidad), hasta llegar a la expresión $\mathbf{M}_z$ (I, w, 0, e).

A continuación se desarrolla para el lector interesado el pasaje de $\mathbf{M}_z$ (**x**, **y**) a $\mathbf{M}_z$ (I, w, 0, e).

Introduciendo una matriz T_z definida como:

$$\mathbf{T}_z = \begin{pmatrix} \left[E\{ \mathbf{x}\mathbf{x}^T \} \right]^{-1} & 0 \\ -E\{ \mathbf{y}\mathbf{x}^T \} . \left[E\{ \mathbf{y}\mathbf{y}^T \} \right]^{-1} & \mathbf{I} \end{pmatrix}$$

En la definición de $\mathbf{T}_z$ interviene la inversa de la matriz de momentos $E\{ x\ x^T \}$, esto equivale precisamente a suponer a dicha matriz singular lo que en la práctica en general no ocurre pero se acepta provisoriamente aquí a los efectos de hallar la solución del problema.

Recordamos que una matriz cuadrada **w** es *no-singular* si existe otra matriz $\mathbf{w}^{-1}$ tal que $\mathbf{w}\,\mathbf{w}^{-1} = \mathbf{w}^{-1}\,\mathbf{w} = \mathbf{I}$, donde $\mathbf{w}^{-1}$ se denomina *inversa* de **w**. Una matriz que no tiene inversa se llama *singular*.

Multiplicando por $\mathbf{T}_z$ por $\mathbf{M}_z$:

$$T_z \,.\, M_z =$$

$$\begin{matrix} I & \left[E\left\{ x\ x^T \right\}\right]^{-1} E\left\{ x\ y^T \right\} \\ 0 & E\left\{ y\ y^T \right\} - E\left\{ y\ x^T \right\} . \left[E\left\{ x\ x^T \right\}\right]^{-1} . E\left\{ x\ y^T \right\} \end{matrix}$$

¿
righ
¿
¿
¿
(¿)¿
¿¿

$$= \begin{pmatrix} \mathbf{I} & \mathbf{w} \\ 0 & \mathrm{E}\left\{ \Delta\mathbf{d}\ \Delta\mathbf{d}^{\mathrm{T}} \right\} \end{pmatrix} \begin{matrix} \\ \div \\ \div \end{matrix}$$

El error mínimo puede obtenerse de:

$$Traza\left[\mathrm{E}\left\{ \Delta\mathbf{d} . \Delta\mathbf{d}^{\mathrm{T}} \right\} \right] = Traza\left[\mathrm{E}\left\{ \mathbf{y}\,\mathbf{y}^{\mathrm{T}} \right\} - \mathbf{w}^{\mathbf{T}}\, \mathrm{E}\left\{ \mathbf{x}\,\mathbf{y}^{\mathrm{T}} \right\} \right]$$

$$= \mathrm{E}\left\{ \left|\mathbf{y}^{\mathrm{T}}\right|^2 \right\} - Traza\left[\mathbf{w}^{\mathbf{T}}\, \mathrm{E}\left\{ \mathbf{x}\,\mathbf{y}^{\mathrm{T}} \right\} \right]$$

$$= S^2_{\min}$$

Considerando que hallar la solución mediante el producto de $T_z \,.\, M_z$ requiere el cómputo adicional de T_z , se observa que no es deseable introducir una matriz adicional T_z , y que dado que un producto matricial del lado izquierdo de una matriz puede realizarse o corresponde a una sucesión de transformaciones elementales de filas o columnas, pueden aplicarse estas transformaciones directamente a la matriz M_z . Dicho de otro modo se debe llevar M_z a la forma descrita previamente mediante transformaciones elementales de filas o columnas.

Para obtener la mayor precisión deben emplearse estrategias de pivoteo adecuadas.

2.3.5. Ejemplo de Cálculo de un Clasificador Polinómico

Resolveremos un clasificador polinómico de primer orden (lineal) para el problema ya visto y ya resuelto para el clasificador lineal.

Sean $\mathbf{x} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ $\mathbf{x}\,\mathbf{x}' = \begin{bmatrix} 2 & 1 & 2 \\ 1 & 2 & 2 \\ 2 & 2 & 4 \end{bmatrix}$ $\mathrm{E}(\mathbf{x}\,\mathbf{x}') = \begin{bmatrix} .5 & .25 & .5 \\ .25 & .5 & .5 \\ .5 & .5 & 1 \end{bmatrix}$

$\mathbf{y} = \begin{bmatrix} 0\ 0\ 1\ 1 \\ 1\ 1\ 0\ 0 \end{bmatrix}$ $\mathbf{y}\,\mathbf{y}' = \begin{bmatrix} 2\ 0 \\ 0\ 2 \end{bmatrix}$ $\mathrm{E}(\mathbf{y}\,\mathbf{y}') = \begin{bmatrix} .5\ 0 \\ 0\ .5 \end{bmatrix}$

$\mathbf{x}\,\mathbf{y}' = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 2 \\ 1 & 1 \\ 2 & 2 \end{bmatrix}$ $\mathrm{E}(\mathbf{x}\,\mathbf{y}') = \begin{bmatrix} 0 & .5 \\ .25 & .25 \\ .5 & .5 \end{bmatrix}$

La matriz $\mathbf{M}_z$ resulta: $\mathrm{Mz} = \begin{bmatrix} .5 & .25 & .5 & 0 & .5 \\ .25 & .5 & .5 & .25 & .25 \\ .5 & .5 & 1 & .5 & .5 \\ 0 & .25 & .5 & .5 & 0 \\ .5 & .25 & .5 & 0 & .5 \end{bmatrix}$

Diagonalizando obtenemos: $\mathrm{Mz} = \begin{bmatrix} 1 & 0 & 0 & -1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

La solución esta dada por: $\mathbf{w} = \begin{bmatrix} -1 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}$

Probando el vector x_1 obtenemos: $x_1\mathbf{w} = \begin{bmatrix} 0\,0\,1 \end{bmatrix} \begin{bmatrix} -1 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 10 \end{bmatrix}$

Repitiendo la operación para el resto de los vectores x_2 x_3 y x_4 del conjunto verificamos el funcionamiento correcto del clasificador.

2.3.5. Estrategias de Pivoteo

Al derivar el esquema de cálculo visto, se supuso que la matriz de momentos E{**xx'**} era no-singular y por lo tanto poseía inversa.

Sin embargo en general no puede suponerse tal cosa debido a que es dable suponer que existan dependencias lineales entre las componentes de **x**. Esto implica que no exista una sino múltiples soluciones para la matriz de coeficientes **w**.

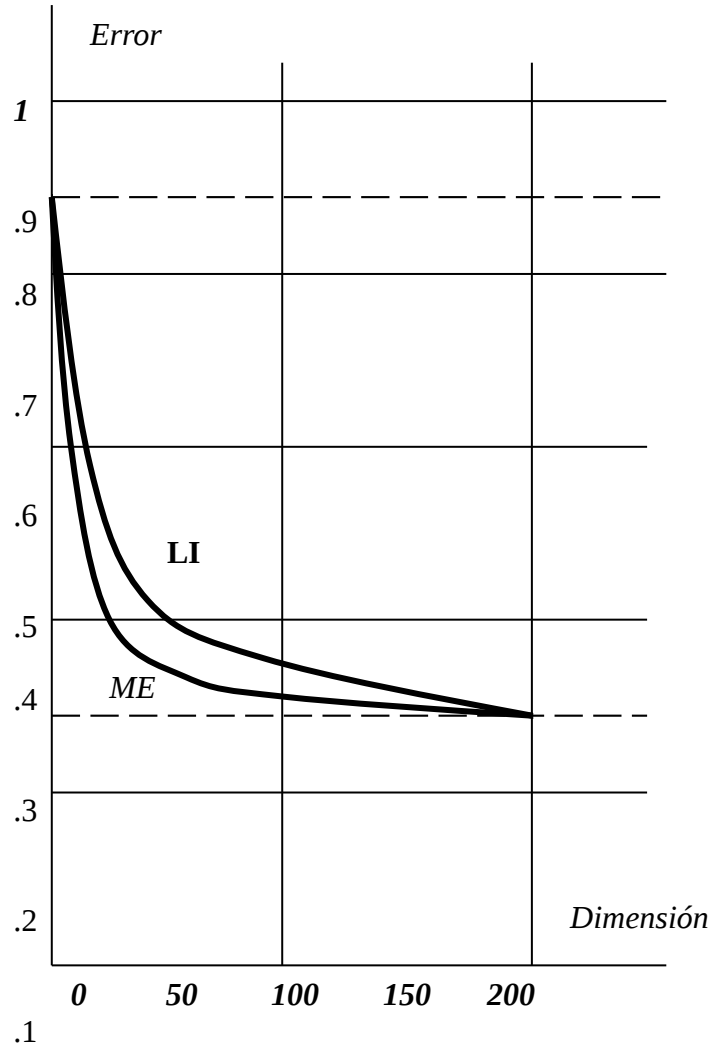
A pesar de esto, se puede alcanzar la solución óptima aplicando no solo el algoritmo de Gauss Jordan, sino también empleando las estrategias de pivoteo apropiadas.

Cuando se describió el cómputo de la matriz **w**, se considera al vector **z** como un único vector, sin diferenciar sus componentes **x** e **y**.

En vez de esto se considera a **z** como compuesto por tres tipos de vectores.

1) Aquellos que corresponden a columnas que no serán tomadas para transformar a otras. Estos son las componentes de **y** = $\left(y_1\, y_2 \cdots y_{M-1} \right)$.
2) Aquellas que ya han sido transformadas.
3) Aquellos (potenciales) vectores candidatos a usar en transformaciones.

Del campo del análisis numérico sabemos que el error residual al resolver numéricamente un sistema de ecuaciones, depende del orden en que se seleccionen los elementos pivote que se utilizan en las transformaciones.



Sabemos también que el orden adecuado surge de seleccionar la adecuada estrategia de pivoteo.

El pivoteo máximo de columna implica seleccionar en cada paso el mayor valor en la diagonal principal. Este valor corresponde a la contribución al error final de esa columna.

Esto permite minimizar el error final.

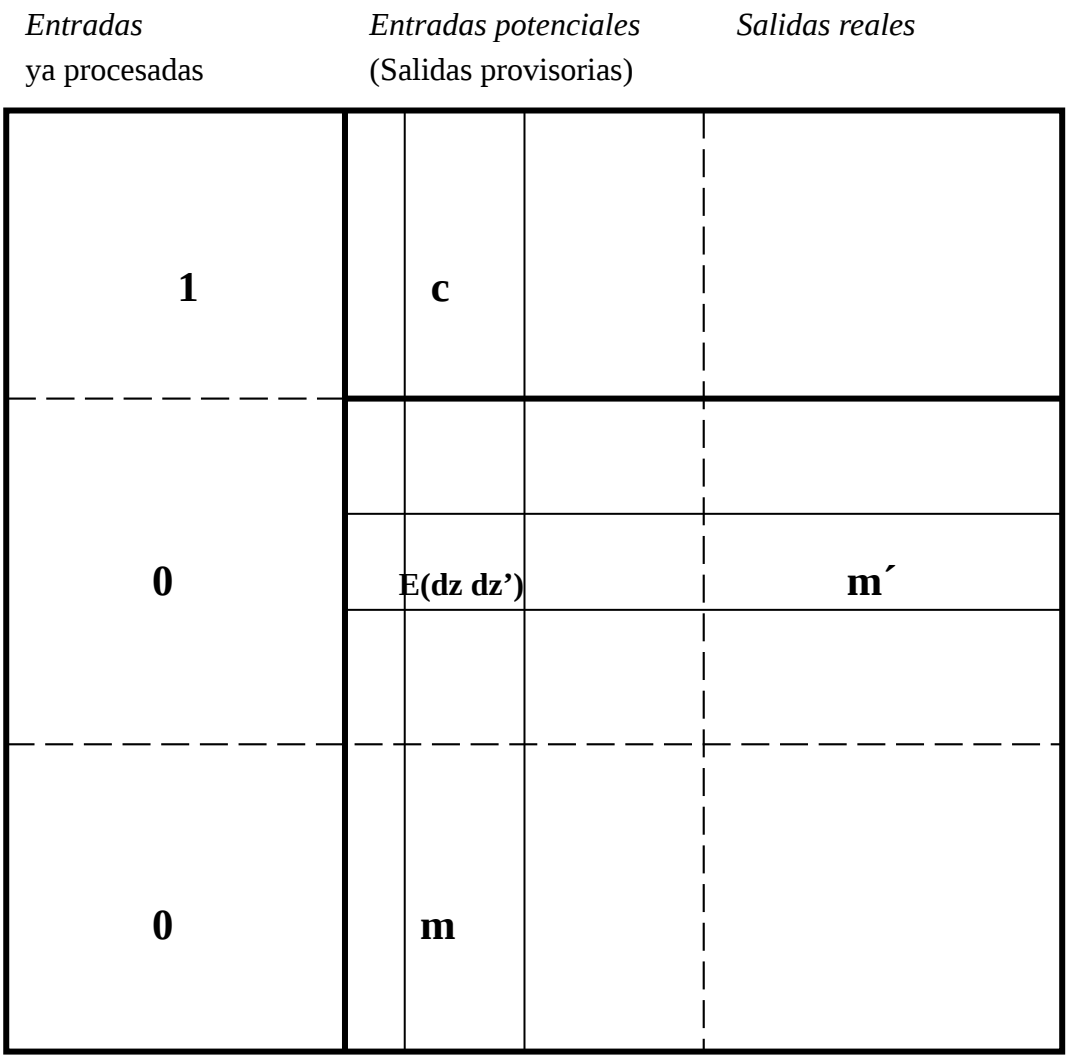
También sabemos por análisis numérico que este valor mide la dependencia lineal de esta columna en el sistema, de manera que este método equivale también a eliminar primeramente aquellas columnas que son mas linealmente dependientes. Esta estrategia de pivoteo es llamada entonces LI.

$$z = \left(x_0 x_1 x_2 .... x_{M-1}\ y_1 y_2 .... y_{M-1} \right)^{\mathrm{T}}$$

Existe una segunda estrategia de pivoteo que analiza el error que producen el conjunto de los componentes de una columna buscando también minimizar el error. Esta estrategia de pivoteo es llamada entonces ME (Mínimo error).

Estas estrategias implican al mismo tiempo establecer un orden de prioridad para las coordenadas del vector **x** en cuanto a la influencia de cada una en el reconocimiento de los patrones que deben confrontar.

Comparando ambas estrategias se obtienen curvas semejantes a las de la figura.



Para comprender mejor lo expuesto consideremos la figura simbólica siguiente donde se representa la matriz de momentos de **z,** $\mathbf{M}_z$.

En esta figura se representa un estado intermedio durante la resolución de la matriz por el método numérico de Gauss-Jordan.

La mejor aproximación al vector z en un estado intermedio de la transformación de la matriz esta dada por $\hat{z} = \mathbf{c'z}\hat{}$. Para esta aproximación resulta un error de aproximación $\hat{\mathbf{z}} = \mathbf{z} - \mathbf{z}\hat{}$ con la varianza $E\{(\mathbf{z}-\mathbf{z}\hat{})^2\}$.

La subcolumna de coeficientes ***c*** *se indica en el diagrama. El valor del error corresponde a los elementos de la diagonal de acuerdo a la fórmula vista.*

Si ese error es nulo la ecuación correspondiente es linealmente independiente, si no ese valor del error puede ser tomado como una medida de cuán linealmente dependiente es la ecuación. Seleccionar las columnas a transformar según los mayores valores de la diagonal implica una estrategia de pivoteo que denominamos LI.

El error correspondiente a esa componente z deviene también de los coeficientes de **m** y **m**' (Que son los mismos), este aporte al error puede medirse por el valor $\mathbf{mm'} = \mathbf{m}^2$**,** normalizados con respecto a $E\{(\mathbf{z}-\mathbf{z}\hat{})^2\}$. De este modo tenemos una segunda estrategia de pivoteo que llamamos de minimización del error ME.

2.3.6. Optimización recursiva de Clasificadores Polinómicos

La ecuación que permite calcular la matriz de pesos puede escribirse como:

$$\mathbf{w}_I = \left(\sum_{i=1}^{I} \mathbf{x_i} \mathbf{x_i}^{\mathrm{T}} \right)^{-1} \cdot \left(\sum_{i=1}^{I} \mathbf{x_i} \mathbf{y_i}^{\mathrm{T}} \right)$$

Ambos términos de la última ecuación pueden ser expresados recursivamente.

$$\sum_{i=1}^{I} \mathbf{x_i} \mathbf{x_i}^{\mathrm{T}} = (1-\alpha) \sum_{i=1}^{I-1} \mathbf{x_i} \mathbf{x_i}^{\mathrm{T}} + \alpha \mathbf{x_i} \mathbf{x_i}^{\mathrm{T}}$$

$$\sum_{i=1}^{I} \mathbf{x_i} \mathbf{y_i}^{\mathrm{T}} = (1-\alpha) \sum_{i=1}^{I-1} \mathbf{x_i} \mathbf{y_i}^{\mathrm{T}} + \alpha \mathbf{x_i} \mathbf{y_i}^{\mathrm{T}}$$

A partir de esto puede obtenerse una expresión para el cálculo recursivo de **w** (a través de un desarrollo). Esta expresión, conocida como la *Ley de aprendizaje "Rápida y Sucia"*, toma la forma:

$$\mathbf{w}_I = \mathbf{w}_{I-1} + \alpha \, . \, \mathbf{x_I} \, . \left(\mathbf{y_I} - \mathbf{w}_{\mathbf{I-1}}^{\mathbf{T}} \, \mathbf{x_I} \right)^{\mathrm{T}}$$

Si consideramos el esquema visto para la red PI-SIGMA y el empleo de funciones escalón a la salida para fijar los valores de **y**, podremos, de acuerdo a los capítulos siguientes, ver que existe una completa analogía con las llamadas redes neurales.

2.4. Clasificador De Margen Óptimo

Los resultados obtenidos usando diferentes clasificadores, en particular los aquellos que han sido descriptos en este capítulo pueden ser mejorados aplicando un método conocido como Máquina de Vectores de Soporte (MVS) -) o Clasificador de Margen Óptimo. Los clasificadores como el Clasificador Polinomial, el Perceptrón Multicapa (se explica mas adelante como red neuronal) y las redes denominadas Función de Base Radial se basan en minimizan el error cuadrático. Esta idea es la base sobre la que se estructura la teoría de muchos de los clasificadores y redes neuronales mas empleados. Este error se obtiene de calcular el error cuadrático entre los resultados "pronosticados" por el sistema y los reales es decir es proporcional a:

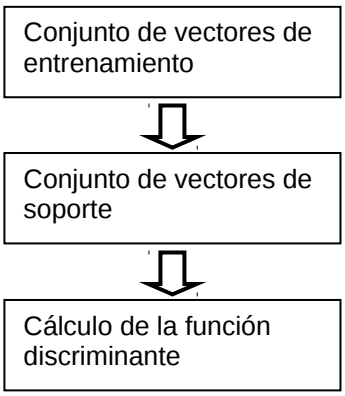


Figura 2.20: El método MVS permite reducir la cantidad de vectores de entrenamiento, y con este conjunto reducido calcular los coeficientes del clasificador

$$\sum_l | \, d_l \, (x) \, - y_l \, | \qquad (1)$$

El criterio de los MVS, también permite separar un conjunto dado de vectores en clases diferentes, como el resto de los clasificadores, pero al mismo tiempo logran una mejora adicional en la determinación del factor de reconocimiento del clasificador que podríamos intentar explicar como que se toma en cuenta la periferia o envolvente de los clusters formados por los patrones de entrenamiento, lo que permite que en cada punto la función discriminante equidiste de los clusters mas próximos.

La esencia de los SVM, es entonces, que a la condición clásica de la minimización del error se le agrega "algo mas".

Recordando que el criterio de minimización del error complicaba la posibilidad de aplicar una solución numérica rigurosa a la solución, debido a la dimensionalidad del problema, podemos agregar que el método MVS busca adicionalmente a la mejora del factor de reconocimiento, reducir la cantidad de patrones a emplear en el cálculo del clasificador de modo de permitir una solución analítica.

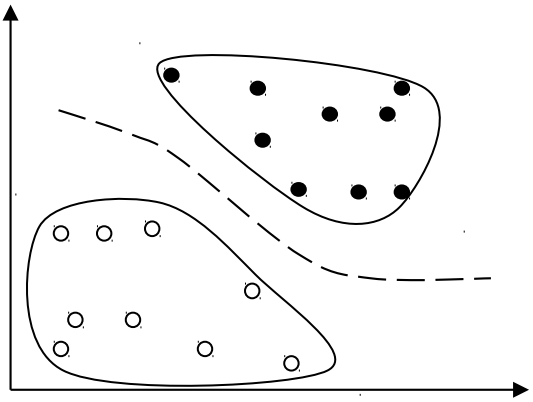


Figura 2.21: Calcular el perímetro de cada cluster del conjunto de entrenamiento puede permitir la construcción de mejores clasificadores.

La idea fundamental de los MVS es muy simple, considerar solo aquellos vectores del conjunto de entrenamiento que estén muy próximos a la zona de transición entre una clase y otra. El problema radicó en su momento en encontrar un modelo matemático que permita saber cuales son esos vectores, y a partir de la individualización de estos, calcular el hiperplano discriminante (o función no lineal para el caso de clasificación no lineal).

Para comprender la idea básica subyacente en este método, podemos considerar patrones en el plano correspondientes a dos clases linealmente separables agrupadas en dos clusters. Podríamos querer "envolver" a cada cluster con una función (Fig. 2.9), y en base a esta referencia calcular la función discriminante de modo que "equidiste en cada punto" de los patrones conocidos que forman el conjunto de entrenamiento. En los esquemas de clasificación clásicos la función discriminante puede tomar cualquier posición entre dos clusters y siempre es considerada una solución correcta

(Fig. 2). En cambio en el método de vectores de soporte la solución es única, y la mejor posible dados los vectores conocidos (de entrenamiento)

En el apéndice C se explica como se determina la expresión que permite calcular la distancia de un punto a un hiperplano, lo que es requerido para comprender la teoría de los SVM. Esta distancia viene dada por la siguiente fórmula:

$$D_x = \frac{\left| \boldsymbol{w}^T \boldsymbol{x}_l + b \right|}{\left\| \boldsymbol{w} \right\|} \qquad (2)$$

Para explicar el método, considérense dos clases linealmente separables, en este caso el hiperplano lineal esta dado por:

$$f_{\text{lin}}(x) = w^T x + b = 0 \qquad (3)$$

El principio de SVM optimiza la matriz de coeficientes w^T definiendo dos restricciones. Primero, el hiperplano discrimina las clases por minimización del error de clasificación.

$$\begin{aligned} w^T x_l + b &> 0 \quad \forall \quad l \in \textit{clase} + \\ w^T x_l + b &< 0 \quad \forall \quad l \in \textit{clase} - \end{aligned} \qquad (4)$$

Esta primera restricción corresponde a la minimización del error, que como se ha dicho es la forma en que en general se basan los otros clasificadores.

Para comprender la segunda restricción se deben considerar las Fig. 1 y 2. La Fig. 1 muestra dos clases separables. Esta figura muestra además tres líneas punteadas, las cuales representan tres hiperplanos diferentes caracterizados por una distancia mínima a uno de los conjuntos dados. Cualquiera de estas líneas se podría obtener a partir de la condición de minimización del error.

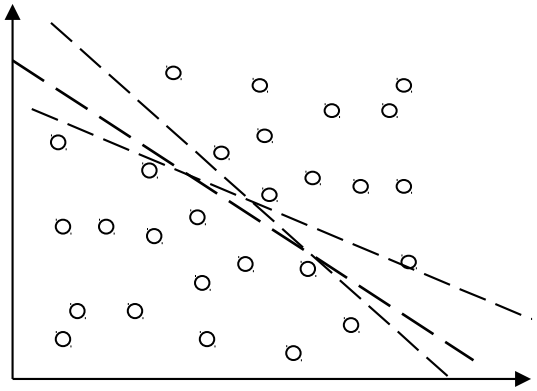


Figura 2.22: Soluciones clásicas para el caso linealmente separable. La solución no es única.

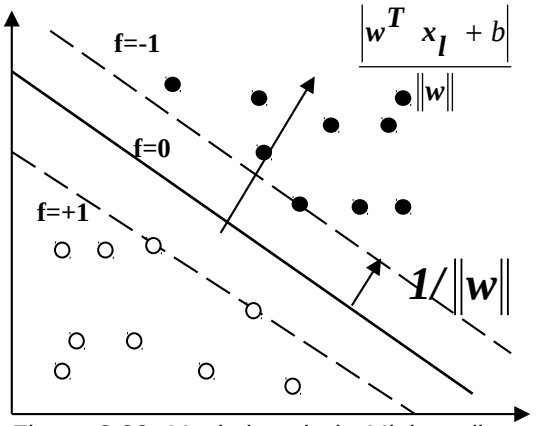


Figura 2.23: Maximizando la Minima distancia entre el hiperplano discriminante y los vectores mas próximos se logra una solución única y óptima.

La Fig. 2 muestra las mismas clases pero la línea que representa el hiperplano corresponde a la maximización de la mínima distancia entre las clases, es decir se verifica una restricción adicionla. Esta segunda condición, esta dada por la expresión siguiente:

$$\max_{\mathbf{w,b}} \left(\min_{\forall l} \frac{|w^T x_l + b|}{\|w\|} \right) \qquad (5)$$

Para encontrar los vectores mas próximos a la función discriminante, que son los vectores de soporte, se supone normalizados los valores de las distancias de estos vectores a la función discriminante de modo que esa distancia es la unidad. Bajo esta suposición el denominado margen (la distancia entre vectores de soporte) vale

$$2/\|w\| \qquad (6)$$

En otras palabras, la optimización consiste en encontrar un hiperplano con el máximo margen de separación. El hiperplano que tenga mayor margen es el mejor calificador. El margen de separación se define como la distancia mínima de los puntos de entrenamiento (vectores soportes) a la superficie de separación. A partir de estos puntos se calcula entonces el hiperplano de separación. Los vectores soportes tocan el límite del margen.
para maximizar este margen se debe minimizar $\|w\|$

$$min\ F(w) = \frac{1}{2}\ w^T\ w\ =\ \frac{1}{2}\ \|w\|^2 \qquad (7)$$

para encontrar el par de hiperplanos que dan el máximo margen debemos minimizar $\|w\|^2$ sujeto a la restricción de que la distancia de los vectores a la función discriminante vale la unidad, lo que se expresa introduciendo un valor y_l = + / - 1 para ambas clases, quedando como expresión de la restricción (Ver explicación mas detallada en apéndice C):

$$y_l\left(w^T\ x_l\ +\ b\right) \geq\ 1\ \ \forall\ \ l\ =\ 1,\ldots,L \qquad (8)$$

2.4.1. La formulación Lagrangiana

Para resolver esta tarea de optimización se pasa a una formulación Lagrangiana del problema, la cual podríamos describir con s4encillez como “un método de optimización sujeta a restricciones”.

Esto se hace por dos razones:

- Primero las ecuaciones (19) se pueden asociar a variables denominadas operadores de Lagrange, simplificando la solución.
- Segundo los patrones de entrenamiento solo aparecerán como productos punto entre vectores. Esta es una propiedad crucial que permite luego extender este principio a casos no lineales.

De acuerdo a lo expresado se introducen los denominados “operadores de Lagrange generalizados con factores no-negativos”.

$$\alpha_i, i\ =\ 1,\ldots,l,$$

uno para cada una de las restricciones vistas. (Es decir que hay un alfa por cada vector del conjunto de entrenamiento).
Con estos operadores y algunas transformaciones se forma la función denominada Lagrangiano que se muestra a continuación.
(Las ecuaciones de restricción son multiplicadas por multiplicadores de Lagrange *positivos* –ya que la restricción es positiva- y sustraídos de la función objetivo, tal como es denominada en el método de Lagrange la función a optimizar)

$$L_p \equiv \frac{1}{2}\|w\|^2 - \sum \alpha_i\left(y_l\left(w^T\ x_l\ +\ b\right) - 1\right) = 0$$

$$L_p \equiv \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{l} y_l\ \boldsymbol{w}^{\mathrm{T}}\ \boldsymbol{x}_{\mathrm{l}} - b\ \sum_{i=1}^{l} \qquad (9)$$

Una vez obtenido el Lagrangiano L_p , debe minimizarse con respecto a w y *b*, anulándose simultáneamente la derivada de L_p con respecto a α_i , sujeto a la restricción $\alpha_i \geq 0$.

$$\frac{\partial}{\partial \omega_v} L_p = \omega_v - \sum_{i=1}^{l} \alpha_i y_l x_i \quad (10)$$

$$\frac{\partial}{\partial b} L_p = - \sum_{i=1}^{l} \alpha_i y_l = 0 \quad (11)$$

$$\frac{\partial}{\partial \alpha_i} L_p = -y_l \left(w^T x_l + b \right) + 1 = 0 \quad (12)$$

Ahora bien, el problema en cuestión es un problema cuadrático *convexo*, (debido a que la función objetivo es convexa y los puntos que satisfacen las restricciones forman también un conjunto convexo). En definitiva esto lleva a traves de un análisis mas detallado (Ver apéndice) a las ecuaciones

$$w = \sum_{i=1}^{l} \alpha_i y_l x_l \quad (13)$$

$$L_d = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \alpha_i \alpha_j y_i y_j x_i^T x_j \quad (14)$$

En las que el método de los vectores de soporte requiere de maximizar L_d con respecto a los α_i sujeto a la restricción vista anteriormente y positividad de α_i .

Hay un operador de Lagrange por cada punto de entrenamiento. Solo los vectores cuyo $\alpha_i \geq 0$ son considerados para calcular la función discriminante. Estos son los llamados vectores de soporte. Estos son precisamente los patrones que pueden interpretarse como que “estan ubicados en la periferia del cluster de una clase”.

2.4.2. La solución del problema cuadrático

Los operadores de Lagrange ($\alpha_1 \alpha_2$ …….α_i) se obtienen de la ecuación de Lagrange:

$$L_D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j z_i z_j x . x^T \quad , \; 0 \leq \alpha_i \leq \beta \qquad \sum_{i=1}^{n} \alpha_i z_i = 0 \quad \text{para todo i}$$

El método se puede resolver minimizando la formulación dual L_d por el método de optimización del problema cuadrático. Los algoritmos de optimización cuadrática identifican los puntos x_i que hacen los multiplicadores de Lagrange (α_i)distintos de cero.

$$\min_x \frac{1}{2} * x' H x + f' x \qquad \text{sujeto a: } Ax <= b \quad (15)$$

Los elementos de H están definidos como $h_{ij} = y_i y_j x_i^T x_j$. Esta matriz H llamada Hessiana, debe calcularse previamente a la optimización cuadrática.

$$H = \sum_{i=1}^{l} y_i y_j x_i^T x_j \quad (16)$$

Es decir que de la ecuación (27) se obtiene un vector de valores α_i uno por cada vector de entrenamiento.

A los efectos del cálculo de los SVM, se debe:

1) Obtener H

2) Aplicar la Ec. (28) la cual esta codificada en librerías y programas de aplicaciónes matemáticas. De esta ecuación se obtienen los operadores alfa, uno para cada vector del conjunto de entrenamiento.
3) Encontrar los vectores de soporte (solo aquellos para los que alfa es positivo).
4) Calcular el hiperplano empleando solo los vectores de soporte.

2.4.3. Cálculo de los coeficientes

Considerando un conjunto de expresiones matemáticas que surgen de los conceptos previos (Conocidas como condiciones de Karush-Kuhn-Tucker), y que pueden consultarse en el apéndice C, los coeficientes del hiperplano pueden obtenerse de:

$$b_{opt} = 1/y_l - \boldsymbol{w}_{opt}^{\mathsf{T}} \boldsymbol{x}_i \qquad para \;\; i \; con \; \alpha_l > 0, \tag{17}$$

y $$w_{opt} = \sum_{i=1}^{l} \alpha_i y_l x_i \; para \;\; i \;\; con \;\; \alpha_l > 0 \tag{18}$$

En otras palabras, la matriz de coeficientes w_{opt} puede obtenerse como una suma de algunos (vectores de soporte) de los vectores de aprendizaje.

2.4.4. Extensión al caso lineal con clases no-separables

Para poder extender lo anterior al caso de clases no separables se relaja la restricción de que la distancia a los vectores de soporte sea exactamente +/-1, para esto se introducen las variables de relajación $\varepsilon_i, \; i = 1,\ldots,l$ en las restricciones, quedando:

$$w \; x_i + b \geq +1 - \varepsilon_i \; para \;\; y_i = +1$$

$$w \; x_i + b \leq -1 + \varepsilon_i \; para \;\; y_i = -1 \varepsilon_i \geq 0 \forall \; i$$

Para considerar estos factores de relajación se adiciona un costo extra C a la función objetivo. *C* es un parámetro elegido libremente. A un mayor valor de *C* corresponde una penalización mayor a los errores. Con esto resulta que debe maximizarse:

$$L_d \equiv \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \alpha_i \alpha_j y_i \; y_j \; x_i^T x_j$$

Sujeto a $0 \leq \alpha_i \leq C$

$$\sum_{i=1} \alpha_i y_i = 0$$

Y la solución nuevamente viene dada por:

$$w_{opt} = \sum_{i=1}^{l} \alpha_i y_l x_i \; para \;\; i \;\; con \;\; \alpha_i > 0$$

La única diferencia con el caso de clases separables es que ahora existe para α_i un límite superior C.

2.4.5. Extensión a clasificadores no lineales

Para aplicar este principio a los clasificadores no lineales se aplica la transformación no lineal $\Phi(x)$ al vector x :

$$f_{nonlin}(x) = w_{opt}^{T} \Phi(x) + b_{opt} \qquad (19)$$

Reemplazando en la ec. (32) el valor de w dado por la ec. 31,

$$f_{nonlin}(x) = \sum_{\alpha_l > 0} \alpha_l y_l \Phi(x_{sv})^T \Phi(x) + b_{opt} \qquad (20)$$

e introduciendo una *función kernel K*:

$$K(\boldsymbol{x},\boldsymbol{x}_{sv}) = \Phi(x_{sv})^T \Phi(x), \qquad (21)$$

la ecuación (33) puede ser escrita como:

$$f_{nonlin}(x) = \sum_{\alpha_l > 0} \alpha_l y_l K(\boldsymbol{x},\boldsymbol{x}_{sv}) + b_{opt} \qquad (22)$$

En el caso de los Clasificadores Polinómicos, la función kernel que representa la expresión polinómica es:

$$K_{PC}(\boldsymbol{x},\boldsymbol{x}_{sv}) = (x\, x_{sv} + 1)^d \qquad (23)$$

donde d representa el grado del polinomio de un clasificador polinómico dado.
Estas adaptaciones a casos no lineales implican simplemente introducir la alinealidad en la matriz H a través de la mencionada función *K*, es decir para el entrenamiento:

$$H = \sum_{i=1}^{l} y_i y_j K(x_i x_j) \qquad (27)$$

2.4.6. Extensión a mas de dos clases

El principio de vectores de apoyo puede aplicarse para el separar dos clases diferentes. En dos caso de la ocurrencia de más de clases, existen dos maneras (ya discutidas en el inicio del capítulo) para aplicar MVS. El primero se llama *Uno contra todos*. Se ocupa de la división de un clases problema de clasificación de K En en K problemas de clasificación. las cada problema k, una clase es clase. separada de las otras. El resto de clases se toma como una segunda El segundo método se llama caso, *Pairwise (por trozos)*. En este en sólo dos clases son clasificadas

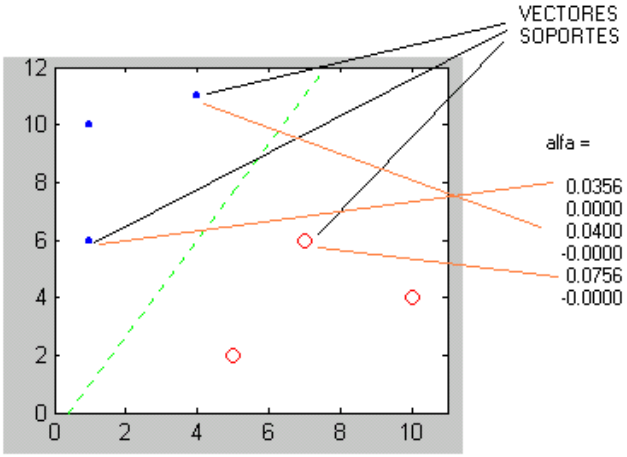


Figura 2.24: Valores de alfa no nulos corresponden a los vectores de soporte del ejemplo

pasos sucesivos.

De experiencias realizadas es posible concluir que la performance del clasificador puede mejorarse al usar el método Pairwise. La diferencia en el valor del factor de clasificación no es trivial. En ciertos casos se ha encontrado que para un factor de clasificación de aproximadamente 99.0 cuando se aplica el método uno contra todos es posible obtener para el mismo factor un valor de aproximadamente 99.9 en caso de usar el método Pairwise.

2.4.7. Ejemplo de aplicación

Dadas dos clases : Clase1= (1, 6) ;(1, 10) ;(4, 11) y Clase2= (5 ,2) ;(7 ,6) ;(10, 4)

el vector **x** resulta: $\begin{bmatrix} 1 & 6 \\ 1 & 10 \\ 4 & 11 \\ 5 & 2 \\ 7 & 6 \\ 10 & 4 \end{bmatrix}$ y el vector **y** es: $\begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ -1 \end{bmatrix}$

La matriz H = (x * x').*(z* z') resulta

37	61	70	-17	-43	-34
61	101	114	-25	-67	-50
70	114	137	-42	-94	-84
-17	-25	-42	29	47	58
-43	-67	-94	47	85	94
-34	-50	-84	58	94	116

¿
righ
¿
¿
¿
[¿][¿][¿][¿][¿]¿
¿
¿

Hallamos α a partir de la ecuación $L_D(\alpha)=\sum_{i=1}^{n}\alpha_i+\frac{1}{2}\alpha^T H\alpha$

(Por ejemplo utilizando el comando QUADPROG(H, f, A, a, B, b) del programa Matlab)