

Máster en Ingeniería Industrial
2021-2022

Trabajo Fin de Máster

Detección y reconocimiento de imagen
con aplicación ferroviaria para la
validación de equipos industriales y
operación automática

Antonio Rodríguez Alhambra

Tutor/es

M. Isabel Herreros Cid

Miguel López Hernández

Lugar y fecha de presentación prevista

DETECCIÓN DEL PLAGIO

La Universidad utiliza el programa **Turnitin Feedback Studio** para comparar la originalidad del trabajo entregado por cada estudiante con millones de recursos electrónicos y detecta aquellas partes del texto copiadas y pegadas. Copiar o plagiar en un TFM es considerado una **Falta Grave**, y puede conllevar la expulsión definitiva de la Universidad.



[Incluir en el caso del interés en su publicación en el archivo abierto]

Esta obra se encuentra sujeta a la licencia Creative Commons **Reconocimiento - No Comercial - Sin Obra Derivada**

RESUMEN

Palabras clave:

DEDICATORIA

ÍNDICE GENERAL

1. ESTADO DEL ARTE	1
2. MARCO TEÓRICO	2
2.1. Extracción de características	2
2.1.1. Momentos de imagen.	3
2.1.2. Histograma de gradientes	5
2.1.3. Patrones locales binarios	7
2.1.4. Descriptores de Fourier	9
2.2. Selección de características	11
2.2.1. ANOVA	11
2.2.2. SFS	13
2.3. Métodos de evaluación	14
2.3.1. Curva ROC	14
2.3.2. Validación cruzada	16
2.4. Modelos de clasificación	18
2.4.1. K-vecinos más cercanos	18
2.5. SVM.	20
BIBLIOGRAFÍA	24

ÍNDICE DE FIGURAS

2.1	Esquema método LBP	7
2.2	Descriptores de Fourier	9
2.3	10
2.4	Distribuciones para un caso ejemplo de ANOVA	11
2.5	Sacado de wikipedia (EDITAR)	15
2.6	División de dataset en entrenamiento y validación	17
2.7	Validación cruzada de 4 particiones	17
2.8	Ejemplo clasificación Knn con datos bidimensionales (caso balanceado) .	19
2.9	Ejemplo de SVM	20
2.10	Recta óptima SVM	21
2.11	Hiperplanos adicionales SVM	22

ÍNDICE DE TABLAS

2.1	Histograma de gradientes, tabla ejemplo	5
2.2	Matriz de confusión	14
2.3	Ejemplo distancias clasificación knn	19

1. ESTADO DEL ARTE

2. MARCO TEÓRICO.

Nota: Este capítulo sirve como una breve introducción a todos los métodos, conceptos y modelos considerados para el desarrollo de este trabajo.

Todo proceso de clasificación de imágenes puede dividirse en los siguientes pasos:

- Tratado inicial de las imágenes
- Extracción de características
- Tratamiento de datos
- Entrenamiento de modelo de clasificación
- Evaluación modelo de clasificación

2.1. Extracción de características

En este paso el objetivo es obtener información cuantitativa (número) de las muestras a partir de diversos métodos.

Por ejemplo, de una clasificación de números escritos a mano se pueden obtener características como el número de trazos, anchura del contorno, color, etc. Es decir, se obtienen una serie de propiedades informativas de la muestra para, después, entrenar el modelo de clasificación.

Para este trabajo, se han utilizado los siguientes métodos:

- Descriptores de Fourier
- Momentos de imagen (Momentos Hu)
- Descriptores locales binarios (*Local Binary Patterns*)
- Histograma de gradientes
- *Método propio

2.1.1. Momentos de imagen.

Los momentos de imagen son un promedio de las intensidades de una imagen binaria. La convención habitual define un momento M para una imagen binaria B de la siguiente forma:

$$M_{ij} = \sum_x \sum_y x^i y^j B(x, y) \quad (2.1)$$

Utilizando la ecuación 2.1, se pueden obtener características la imagen como los centroides:

$$\bar{x} = \frac{M_{10}}{M_{00}} \quad (2.2)$$

$$\bar{y} = \frac{M_{01}}{M_{00}} \quad (2.3)$$

Sin embargo, aplicando una transformación a la ecuación 2.1, se pueden obtener momentos invariantes a la traslación:

$$\mu_{i,j} = \sum_x \sum_y (x - \bar{x})^i (y - \bar{y})^j I(x, y) \quad (2.4)$$

A la ecuación 2.4 se la conoce como **momentos centrales**.

Además, aplicando otra transformación se pueden obtener momentos invariantes al escalado también:

$$\eta_{i,j} = \frac{\mu_{i,j}}{\frac{i+j}{2} + 1} \quad (2.5)$$

La fórmula 2.5 se conoce como **momento centralizado**.

Momentos de Hu

Los momentos de Hu [1] son un conjunto de siete fórmulas obtenidas a partir de los momentos centralizados que permiten obtener siete momentos invariantes tanto a traslación, rotación, escalado y volteado (el séptimo momento es el invariante a volteado, cambiando de signo cuando la imagen es reflejada).

$$\begin{aligned}
M_1 &= \eta_{20} + \eta_{02} \\
M_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
M_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
M_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
M_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + 3\eta_{03})^2] \\
&\quad + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
M_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
&\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
M_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + 3\eta_{03})^2] \\
&\quad - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
\end{aligned} \tag{2.6}$$

A partir de 2.6, se pueden obtener siete características numéricas.

2.1.2. Histograma de gradientes

El histograma de gradientes [2] es una técnica de extracción de características muy utilizada en la detección de objetos.

A partir de una imagen binaria de dimensiones $n \times m$, se calculan los gradientes de intensidad, así como la magnitud y el ángulo:

$$\begin{aligned} G_x &= B(x+1, y) - B(x, y) \\ G_y &= B(x, y+1) - B(x, y) \end{aligned} \quad (2.7)$$

La ecuación 2.7 determina los gradientes de la imagen binaria B . Tanto G_x como G_y son dos imágenes binarias con la información de gradientes en ejes X e Y, respectivamente. A partir de la ecuación 2.7, se obtienen las magnitudes y ángulos:

$$Mag_{(x,y)}(\mu) = \sqrt{G_x^2 + G_y^2} \quad (2.8)$$

$$Ang_{(x,y)}(\theta) = |\tan^{-1}\left(\frac{G_y}{G_x}\right)| \quad (2.9)$$

Tanto la ecuación 2.8 como 2.9 representan imágenes binarias.

El siguiente paso consiste en dividir las imágenes de magnitud y ángulos en N cuadrículas, pudiendo ser $N = 1$ (una única cuadrícula).

Para cada cuadrícula se representa un histograma de 9 posiciones, con cada posición en el rango $[\theta, \theta + \delta\theta]$. Convencionalmente, se suele utilizar $\delta\theta = 20^\circ$. De esta forma se obtiene el siguiente histograma H_N :

Magnitud								
Ángulo	0	20	40	60	80	100	120	140

TABLA 2.1. HISTOGRAMA DE GRADIENTES, TABLA EJEMPLO

A cada intervalo angular y de magnitud, se le denomina θ_j y μ_j , respectivamente, para $j \in [0, 8]$.

De tal forma, si en la cuadrícula N_i existe un $\theta_{x,y}$ que pertenece al rango $[\theta, \theta + \delta\theta \cdot j]$, para $j \in [0, 8]$, se determina que $\mu_j = \mu_{x,y} + \mu_j$.

$$\mu_j = \sum \mu_{x,y} \iff \theta_{x,y} \in [\theta, \theta + \delta\theta \cdot j] \quad (2.10)$$

Existen casos en los que $\theta_{x,y} > 160^\circ$, entonces $\mu_{x,y}$ contribuye tanto a 0° como a 160° .

$$\begin{aligned} \frac{180 - \theta_{x,y}}{20} \cdot \mu_{x,y} &\implies [160, 180) \\ \left(1 - \frac{180 - \theta_{x,y}}{20}\right) \cdot \mu_{x,y} &\implies [0, 20) \end{aligned} \tag{2.11}$$

El último paso consiste en normalizar el histograma:

$$\mu_j = \frac{\mu_j}{\max(\mu_j)} \tag{2.12}$$

Tras la aplicación de 2.12, se tienen $9 \cdot N$ características, siendo N el número de cuadrículas.

2.1.3. Patrones locales binarios

También conocido por sus siglas del inglés **LBP** (del inglés, *Local Binary Patterns*), este método propuesto en la década de 1990 [3] permite extraer un histograma de 256 características de una muestra.

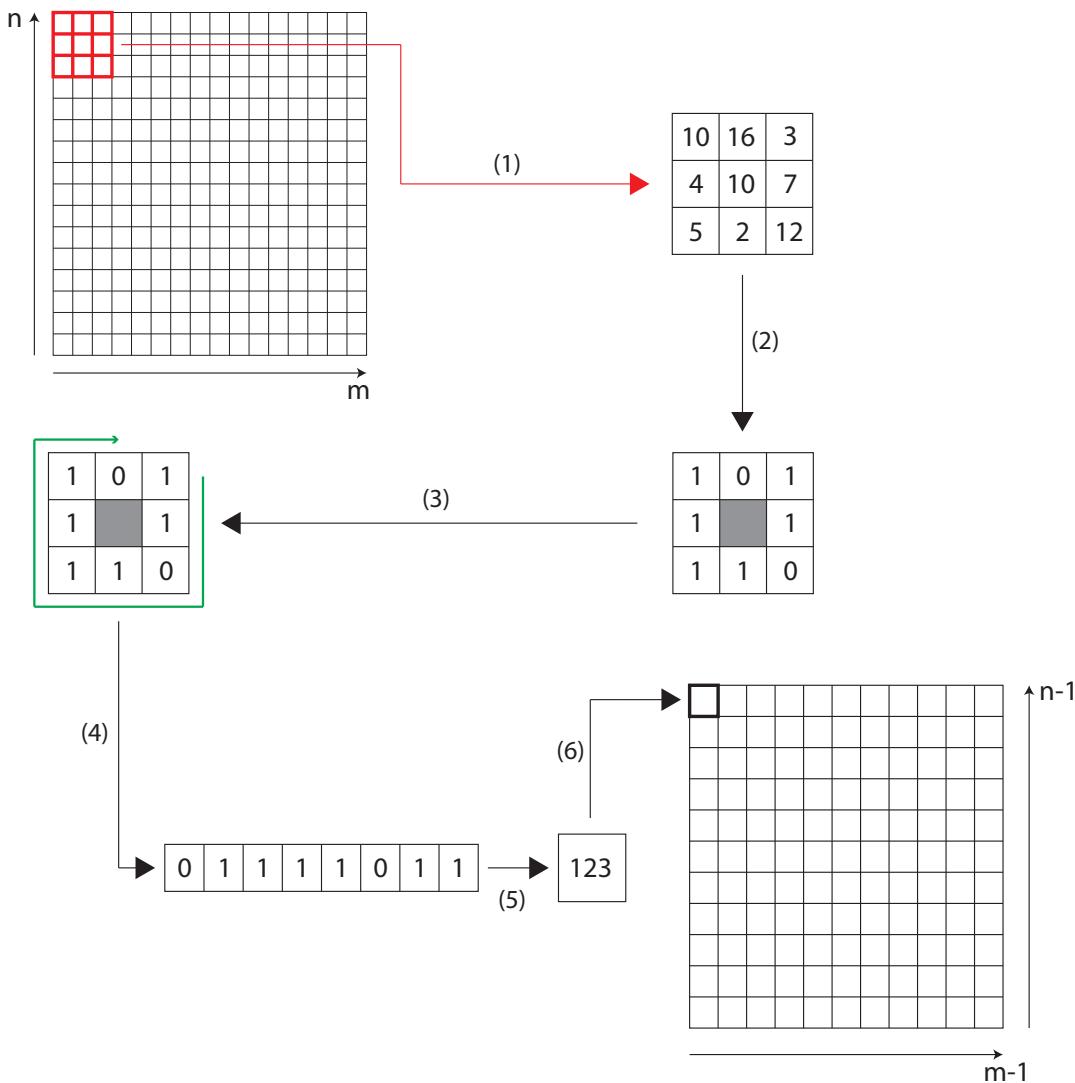


Fig. 2.1. Esquema método LBP

La figura 2.1 muestra un esquematizado resumen del proceso de extracción de características por este método.

La imagen original es tratada como una matriz de dimensiones $n(\text{filas}) \times m(\text{columnas})$. A partir de esta, se extraen submatrices de dimensiones 3×3 y se comienza el análisis de datos.

En la nueva submatriz, la posición central es comparada con las exteriores. Si, para una posición exterior, el valor central es menor o igual, se asigna a esta posición un valor 1. De ser mayor, se asigna un 0.

La nueva matriz binaria es tratada como un número binario de 8 dígitos, consistiendo el siguiente paso en obtener el equivalente en base diez. Este último valor es almacenado en una nueva matriz de dimensiones $(n - 1)x(m - 1)$.

Realizando este proceso para toda la imagen original se tiene un conjunto de datos a partir de los cuales obtener el histograma sobre el que se basa el método **LBP**.

Teóricamente, el número máximo de características extraíbles es 256 (ver 2.13), pero pueden agruparse por rangos obteniendo grupos de características. Esta última opción es dependiente de la aplicación y del problema a tratar.

$$[00000000, 11111111]_2 \rightarrow [0, 255]_{10} \quad (2.13)$$

2.1.4. Descriptores de Fourier

Los descriptores de Fourier son un método de extracción de características por el cual un objeto bidimensional, cuyos puntos tienen las coordenadas (x_k, y_k) , es mapeado a un dominio complejo de la forma (x_k, iy_k) . De tal forma, el objeto es tratado como una señal discreta compleja a la cual se aplica la transformada discreta de Fourier para encontrar sus armónicos (descriptores de Fourier).

La figura 2.2a representa una forma hexagonal rellena. Para obtener los descriptores de Fourier de este objeto, es necesario obtener el contorno exterior de la figura (véase 2.2b).

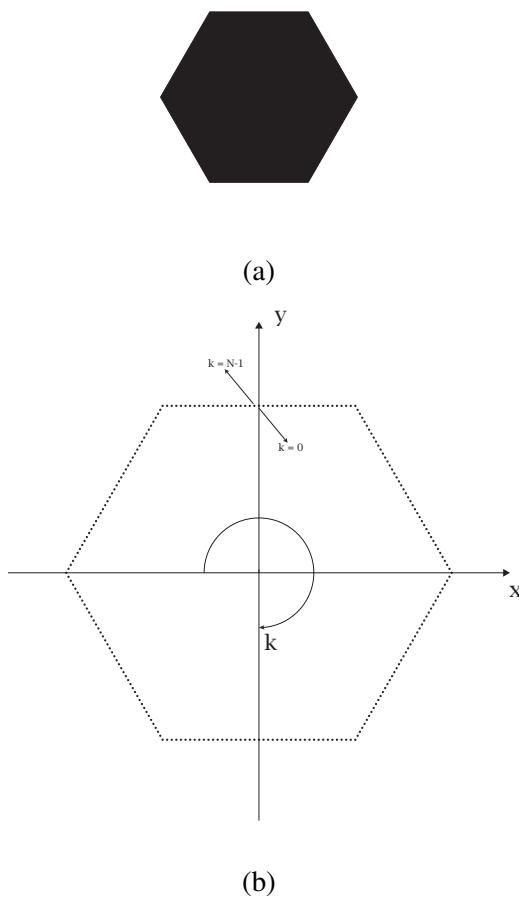


Fig. 2.2. Descriptores de Fourier

Este nuevo contorno está representado por N puntos, de tal forma que cada punto puede definirse de la forma (x_k, y_k) , con $k = 0, 1, 2, \dots, N - 1$ y el contorno, a su vez, como la señal $f_k = (x_k, y_k)$.

Si f_k es transformada al dominio complejo de la forma $f_k = x_k + iy_k$, es posible aplicar la transformada discreta de Fourier a f_k para obtener los N armónicos de la señal (descriptores de Fourier):

$$F_m = \sum_{k=0}^{N-1} f_k \cdot e^{-\frac{2\pi i}{N} mk}, \quad m = 0, 1, 2, \dots, N-1 \quad (2.14)$$

La ecuación 2.14 (transformada discreta de Fourier), proporciona la serie de valores $F_0, F_1, F_2, \dots, F_{N-1}$, a partir de los cuales se obtienen los descriptores de Fourier (valores absolutos) $|F_0, F_1, F_2, \dots, F_{N-1}|$.

En función de la resolución de la imagen original y su definición, el número de puntos del contorno N variará, obteniéndose una cantidad diferente para cada caso. Por ello, el número de descriptores de Fourier obtenibles de cada imagen no será el mismo. Es por esto que, generalmente, de los N descriptores obtenidos para cada imagen, se escogen solo $n \mid 0 \leq n \leq N$.

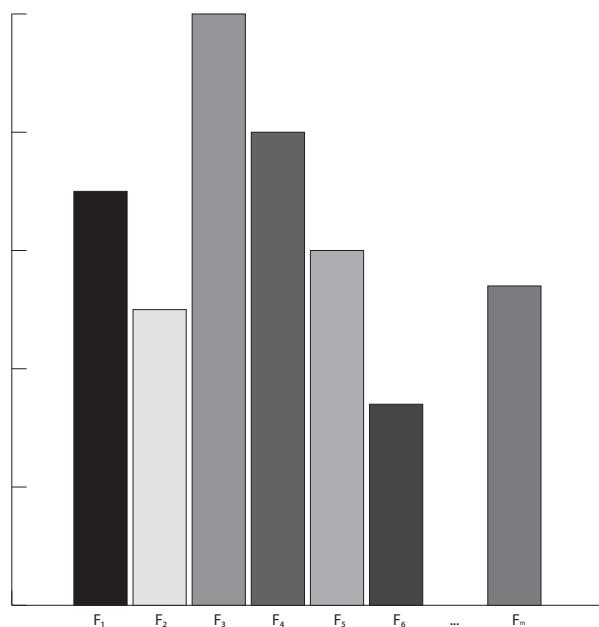


Fig. 2.3

2.2. Selección de características

Una vez se han aplicado los métodos propuestos en la sección 2.1, es necesario realizar un análisis del poder clasificatorio de las características extraídas.

Para este trabajo, se han utilizado los siguientes métodos:

- ANOVA
- SFS

2.2.1. ANOVA

Nota: ANOVA es un conjunto de métodos estadísticos, entre los cuales está el F-Test, que es el que se usa aquí, principalmente.

Nota: Casi toda la información está sacada de [4].

Conocido como Análisis de la Varianza (del inglés, *ANalysis Of VAriance*).

Dado un ejemplo como una distribución de datos de dos clases (véase figura 2.4), se disponen de dos características de clasificación: (x, y).

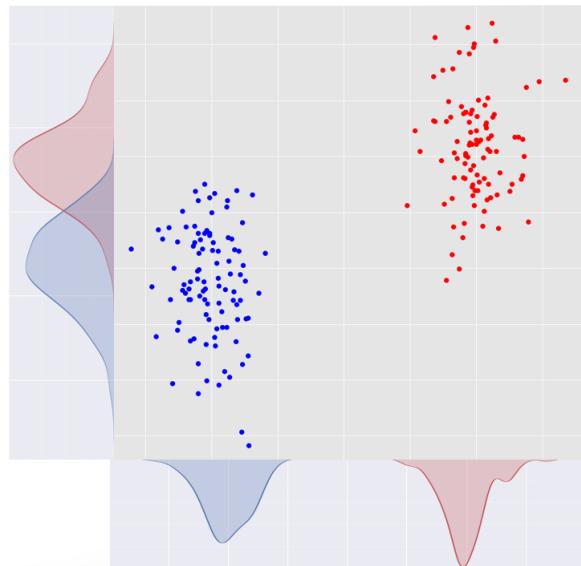


Fig. 2.4. Distribuciones para un caso ejemplo de ANOVA

La característica x es mucho mejor que la y a simple vista pues, comparando las distribuciones de ambas, para x se obtienen dos distribuciones claramente separadas por completo, mientras que para la característica y las distribuciones se solapan.

También se puede comprobar que la distribución para la característica x presenta una menor varianza (son más compactas) que con y .

Por tanto, para este ejemplo se puede definir la condición 2.15 como un parámetro de discriminación de características. Cuanto mayor sea para una característica, mejor será su poder de clasificación.

$$F = \frac{\text{Distancia entre clases}}{\text{Compacidad de clases}} \quad (2.15)$$

Matemáticamente, la fórmula 2.15 puede expresarse de la siguiente forma siguiendo el método ANOVA:

- El numerador (distancia entre clases) es definible con:

$$n_{azul}(\overline{x_{azul}} - \bar{x})^2 + n_{roja}(\overline{x_{roja}} - \bar{x})^2 \quad (2.16)$$

- El denominador (compacidad de clases), que no es sino la varianza de clases, puede expresarse como:

$$\left(\frac{1}{(n_{azul} - 1) + (n_{roja} - 1)} \right) \left(\sum_{i=1}^{n_{azul}} (x_i - \overline{x_{azul}})^2 + \sum_{i=1}^{n_{roja}} (x_i - \overline{x_{roja}})^2 \right) \quad (2.17)$$

De tal forma, la ecuación 2.15 queda como:

$$F = \frac{n_{azul}(\overline{x_{azul}} - \bar{x})^2 + n_{roja}(\overline{x_{roja}} - \bar{x})^2}{\left(\frac{1}{(n_{azul} - 1) + (n_{roja} - 1)} \right) \left(\sum_{i=1}^{n_{azul}} (x_i - \overline{x_{azul}})^2 + \sum_{i=1}^{n_{roja}} (x_i - \overline{x_{roja}})^2 \right)} \quad (2.18)$$

La ecuación 2.18 da un parámetro para una variable (en este caso x). En un problema con k características, se obtendrían k valores de F , es decir, una F para cada variable.

Utilizando las k características extraídas con los métodos propuestos en 2.1, ANOVA proporcionaría F_k valores, de entre los cuales se seleccionarían los N con mayor valor F . La ecuación 2.19 define el valor de F para un caso con $j = 1, 2, 3, \dots, N$ clases.

$$F = \frac{\sum_{j=1}^N n_j (\overline{x_j} - \bar{x})^2}{\left(\frac{1}{\sum_{j=1}^N (n_j - 1)} \right) \left(\sum_{j=1}^N \left(\sum_{i=1}^{n_j} (x_i - \overline{x_j})^2 \right) \right)} \quad (2.19)$$

Nota:

Este método solo da información sobre lo bien que UNA variable discrimina, pero no dice como de bien lo harían varias juntas.

2.2.2. SFS

Los algoritmos SFS (del inglés *Sequential Feature Selector*) son un conjunto de técnicas de selección de características (algoritmos voraces) usados para reducir un espacio inicial n-dimensional de características a otro k-dimensional donde $k \leq d$.

Si se tiene el siguiente conjunto n-dimensional $Y = \{y_1, y_2, y_3, \dots, y_n\}$ de características de entrada, utilizando un estimador determinado (SVM, Knn, SDG, etc.) y una métrica de rendimiento determinada, el algoritmo SFS busca obtener un conjunto de salida $X_k = \{x_j | j = 1, 2, \dots, d; x_j \in Y\}$, con $k = (0, 1, 2, \dots, n)$ tal que X_k esté formado por las k características que maximicen la métrica.

Existen dos formas de realizar este proceso: hacia adelante (*Forward*) y hacia detrás (*Backwards*).

- Sequential Forward Selection: se comienza con un conjunto vacío $X_0 = \emptyset/k = 0$. En cada iteración se aumenta el número de características hasta encontrar la combinación que de el mejor resultado (según la métrica a resolver).
- Sequential Backward Selection: se comienza con el conjunto inicial Y . En cada iteración se disminuye el número de características hasta encontrar la combinación que de el mejor resultado (según la métrica a resolver).

Es posible no llegar al mismo resultado empleando sentidos contrarios y tampoco obtener el mismo resultado, en varias iteraciones, utilizando el mismo método.

Nota:

En *Sklearn* la métrica para su modelo SFS es una evaluación de validación cruzada con un estimador definido en la llamada al método SFS.

2.3. Métodos de evaluación

Antes de entrar en la sección de modelos de clasificación, es necesario definir las métricas propuestas utilizadas para analizar los resultados de los clasificadores.

En una primera instancia, es necesario presentar la Matriz de confusión.

		Valor real	
		1	0
Valor predicho	1	True Positive (TP)	False Positive (FP)
	0	False Negative (FN)	True Negative (TN)

TABLA 2.2. MATRIZ DE CONFUSIÓN

La matriz de confusión es una herramienta muy utilizada para representar y ver de forma sencilla los resultados de una clasificación. En el caso de la tabla 2.2, se representa una matriz de confusión para un caso de clasificación binaria, sin embargo puede utilizarse para casos multclases.

2.3.1. Curva ROC

Lo habitual, además de deseable, a la hora de obtener los valores predichos de un clasificador, es hacerlo en forma de estimaciones probabilísticas en el rango [0, 1]. De tal forma que, cuando se obtenga un valor, se puede comprobar con qué nivel de confianza el clasificador asigna a qué clase la muestra a predecir.

Muchas librerías de algoritmos supervisados devuelven los resultados, por defecto, como valores enteros 0 o 1, aplicando un umbral de clasificación de 0,5. En el caso de obtener 0,5000001 para la clase positiva y 0,4999999 para la negativa, el clasificador asignaría automáticamente la muestra a la clase positiva, obviando el hecho de que realmente no existe un nivel de confianza suficiente para clasificar la muestra de tal forma.

Al obtenerse probabilidades y no valores discretos, para asignar clases, es necesario establecer un umbral de clasificación por el cual si $p(n) \geq \text{umbral} \rightarrow n = 1$. Por tanto, es conveniente encontrar un valor para el umbral de clasificación que haga que el clasificador funcione lo mejor posible.

Para cada valor de umbral se obtiene una matriz de confusión diferente, así como sus pertinentes métricas. Puede que exista un valor de umbral en el que el clasificador, para la base de datos dada, sea idóneo o puede que no existe ningún valor de umbral que haga que el clasificador discrimine correctamente. Sin embargo, es posible que, independientemente del umbral, el clasificador obtenga buenos resultados.

Para poder analizar estos casos, se recurre a la curva ROC (del inglés *Receiver operating characteristic*).

La curva ROC es un gráfico utilizado para determinar la habilidad discriminante de un clasificador binario según su umbral de clasificación es variado.

Este gráfico se basa en el espacio ROC, que es básicamente la representación de la tasa de verdaderos positivos (o exhaustividad) (2.20) frente a la tasa de falsos positivos (2.21).

Nota:

Exhaustividad: cantidad de valores reales positivos predichos como positivos.

$$\text{Exhaustividad (TPR)} = \frac{TP}{P} = \frac{TP}{TP + FN} \quad (2.20)$$

$$\text{Tasa de falsos positivos (FPR)} = \frac{FP}{N} = \frac{FP}{FP + TN} \quad (2.21)$$

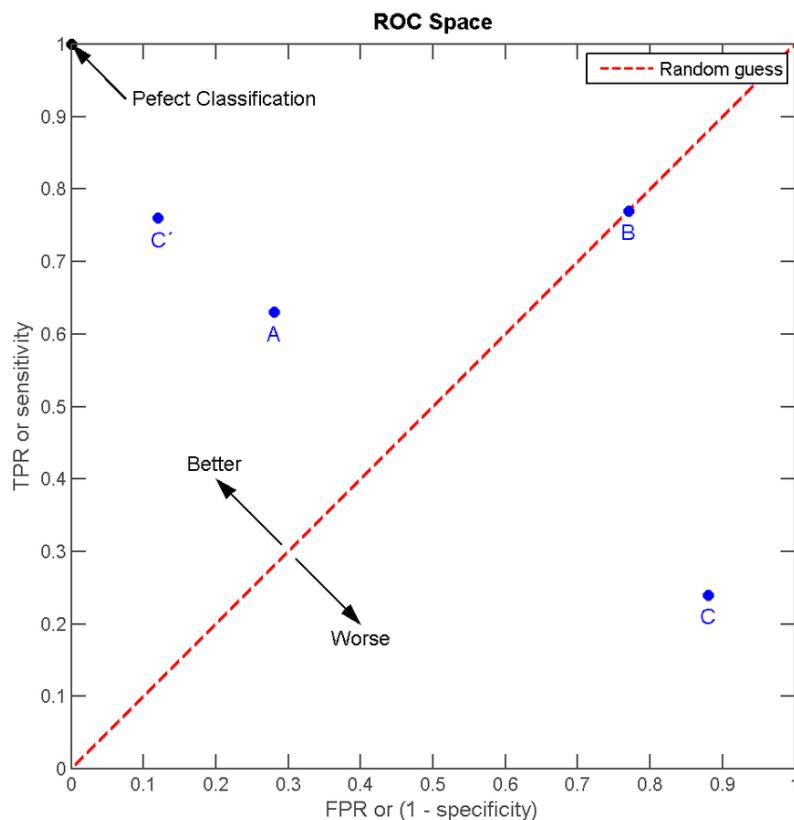


Fig. 2.5. Sacado de wikipedia (EDITAR)

La línea de puntos roja representa un clasificador puramente aleatorio que asigna un 0,5 de probabilidades de pertenecer a la clase positiva a cada muestra.

Para valores de umbral en el rango [0, 1], se obtienen los valores de 2.20 y 2.21, y se representa el punto correspondiente en el gráfico. De tal forma, se obtiene una curva escalonada que representa la calidad del clasificador.

El caso idóneo es aquel en el que la curva del clasificador es un escalón unidad, es decir, aquel clasificador que, para cualquier valor de umbral, se obtiene un 100 % de exhaustividad. Númericamente, este tipo de curva corresponde a una curva con una área bajo la curva de la unidad.

El peor caso es el clasificador cuya curva sea la inversa al escalón unidad pues, se obtiene un 0 % de exhaustividad para cualquier umbral, es decir, aquella curva con una área bajo la curva nula.

Por tanto, se define el parámetro **AUC**, del inglés *Area Under the Curve*, que determina la calidad discriminante del clasificador. Cuanto mayor sea el valor de AUC, mejor el clasificador.

Sin embargo, la curva ROC no es correcta para casos desbalanceados ya que no representa correctamente los resultados, siendo necesario sustituirla por una curva de precisión-exhaustividad [5].

Clasificación multiclas La curva ROC y sus derivadas sólo son utilizables para problemas binarios. Para casos con más de dos clases, se requiere una clasificación de Uno-Contra-Todos (considerar una clase como positiva y el resto negativa, y repetir este proceso para cada clase, obteniéndose tantos clasificadores como clases haya), obteniendo tantas curvas ROC como clases, así como un valor AUC para cada clase, que representaría la calidad del clasificador de discriminar esa clase respecto a las demás.

2.3.2. Validación cruzada

Todo proceso de clasificación necesita de dos conjuntos de datos para entrenamiento y validación de resultados.

Lo más simple es dividir el dataset de partida en dos particiones: entrenamiento y validación.

Sin embargo, cabe la posibilidad de que al dividir el dataset, los datos de la partición de entrenamiento sean diferentes a la de validación. Es decir, puede que los datos de entrenamiento no sean suficientemente generalistas (atípicos) y al validar el modelo con los datos de validación, se obtengan resultados muy pobres. Para evitar esto se recurre a la validación cruzada.

Del inglés *Cross-Validation*, la validación cruzada es un método de análisis de resultados en modelos de clasificación que garantiza la independencia de los resultados frente a la división de datos de entrenamiento y validación.

El dataset inicial es dividido en N particiones (además, es de buenas prácticas mezclar

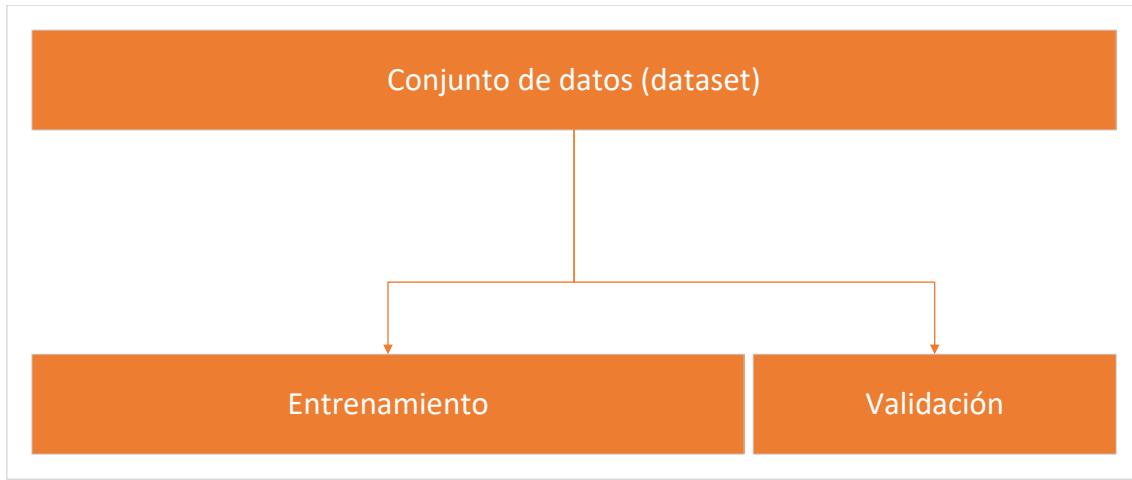


Fig. 2.6. División de dataset en entrenamiento y validación

los datos entre particiones) y cada partición es dividida en N partes. $N - 1$ partes de cada partición son empleadas para entrenar el modelo y la parte restante para validación. Este proceso se repite para todas las particiones, obteniendo N resultados.

La figura 2.7 representa una validación cruzada de cuatro particiones.

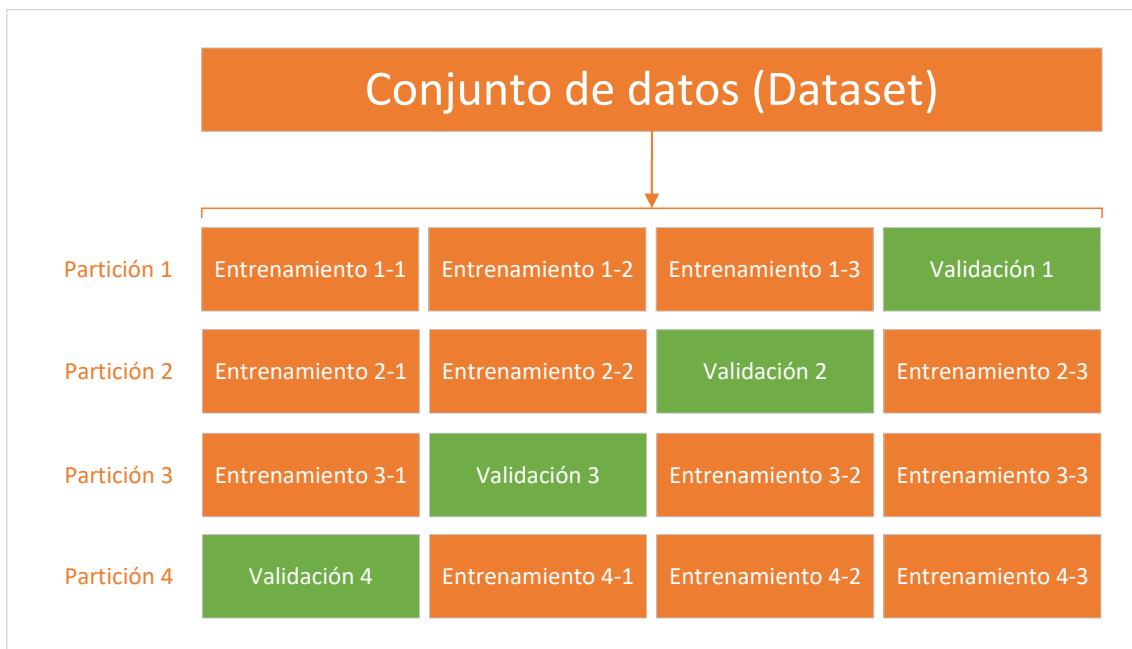


Fig. 2.7. Validación cruzada de 4 particiones

Con los N resultados se realizan análisis estadísticos (principalmente media y desviación típica) para obtener métricas independientes del punto de división de la base de datos.

2.4. Modelos de clasificación

Esta sección presenta, teóricamente, los modelos propuestos para la resolución del trabajo. Aunque existen muchos otros tipos, solo se tratarán los comentados a continuación.

Nota: Los siguientes modelos se caracterizan por ser de aprendizaje supervisado. Los datos de entrenamiento son presentados en pares, con características y etiquetas de identificación.

2.4.1. K-vecinos más cercanos

K-vecinos más cercanos, abreviado como Knn, (del inglés, *K-nearest neighbors*), es un modelo de clasificación supervisado basado en la idea de clasificar una muestra en función de los puntos más cercanos a él (vecinos más cercanos), con la muestra siendo asignada en función de la clase más repetida entre los k puntos menos distantes.

Knn soporta tanto clasificación multiclas como binaria.

Suponiendo que se tiene un conjunto de datos de la forma $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)$ siendo Y_n la etiqueta de la clase a la que pertenece la muestra X_n , con X_n siendo un conjunto d-dimensional de d características extraídas de la forma (x_1, x_2, \dots, x_d) y una muestra a clasificar M de la misma forma que X_n , el algoritmo calcula la distancia Minkowski (véase ec. 2.22) de la muestra M a todas las n muestras de X obteniéndose n distancias, de las cuales, la clase con más frecuencia entre las k distancias menores es asignada a la muestra M .

$$D(X_n, M) = \left(\sum_{i=1}^d |X_{ni} - M_i|^p \right)^{\frac{1}{p}}, \quad p \geq 1 \quad (2.22)$$

Es común que el parámetro p de 2.22 valga 1 (distancia Manhattan) o 2 (distancia Euclíadiana).

La figura 2.8 ilustra el procedimiento de un algoritmo knn en la clasificación de una muestra M respecto a tres clases balanceadas. A su vez, la tabla 2.3 muestra, ordenadas de menor a mayor, las distancias de la muestra a los puntos de cada clase.

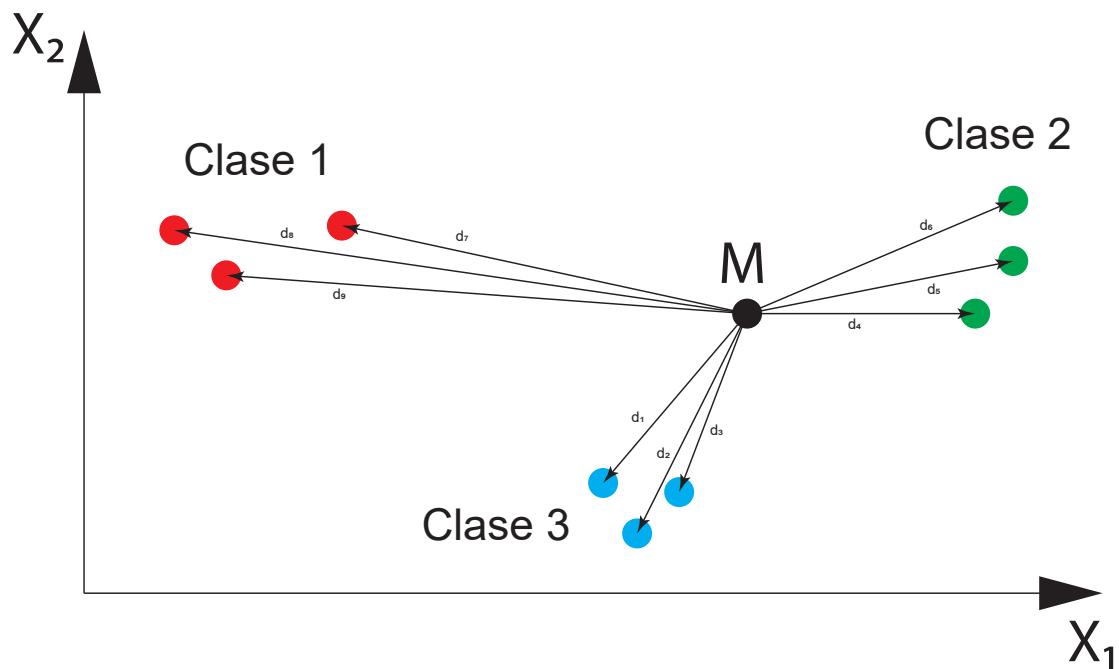


Fig. 2.8. Ejemplo clasificación Knn con datos bidimensionales (caso balanceado)

Distancia	Clase
d_1	3
d_2	3
d_4	2
d_3	3
d_5	2
d_6	2
d_7	1
d_8	1
d_9	1

TABLA 2.3. EJEMPLO DISTANCIAS CLASIFICACIÓN KNN

En función del parámetro k , la muestra M será asignada a la clase más repetida de entre las k menores distancias en la tabla 2.3. De ser $k = 4$, $M \rightarrow$ Clase 3 pues, de las 4 mínimas distancias, la clase 3 tiene la mayor frecuencia.

Centroide más cercano Existe una variación al método knn que se basa en calcular la distancia de la muestra M a los centroides de cada clase. Es decir, cada clase se representa como el centroide de todos sus puntos, a partir del cual se calcula la distancia con la muestra objetivo y se escoge la clase con la mínima distancia. La ventaja de este método es que no requiere ningún parámetro, sólo especificar el tipo de distancia a utilizar, sin

embargo, no es una buena opción cuando las clases son no-convexas así como si presentan varianzas muy diferentes.

Knn con datos desbalanceados El modelo knn no sufre con datos desbalanceados, sin embargo es posible implementar un peso por muestra invirtiendo su distancia, de tal forma que, a mayor distancia de la muestra, menor influencia en el resultado.

2.5. SVM

Del inglés, *Support Vector Machines*, SVM es un modelo de clasificación binaria supervisada que se basa en la aplicación de hiperplanos para separar las muestras de dos clases.

Suponiendo que se tiene un conjunto de datos de la forma $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)$ siendo Y_n la etiqueta de la clase a la que pertenece la muestra X_n de la forma $(-1, 1)$, con X_n siendo un conjunto d-dimensional de d características extraídas de la forma (x_1, x_2, \dots, x_d) y una muestra a clasificar S de la misma forma que X_n , el modelo SVM busca un hiperplano que separe los puntos X_n con $Y_n = -1$ de los puntos con $Y_n = 1$.

Para ilustrar, de forma sencilla, el funcionamiento de los modelos SVM, se suponen dos clases de datos, $(1, -1)$, en un espacio de características bidimensional.

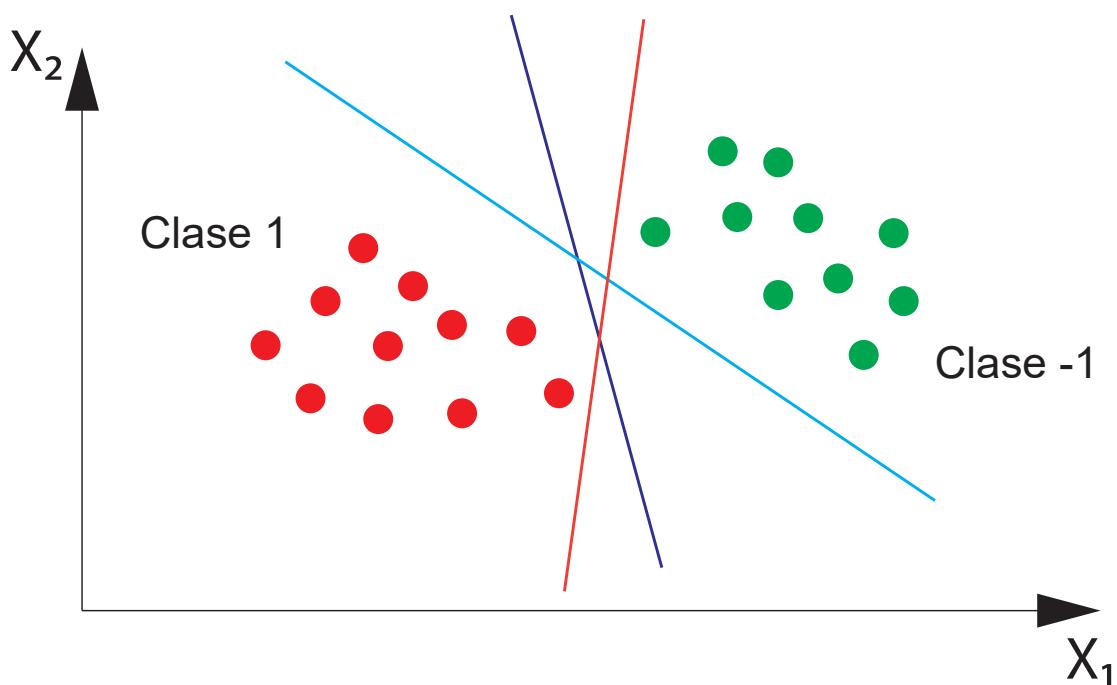


Fig. 2.9. Ejemplo de SVM

La figura 2.9 muestra posibles hiperplanos (en dos dimensiones, un hiperplano es una recta) para clasificar a las dos clases. Cualquiera de las tres rectas mostradas es correcta,

sin embargo, la mejor solución posible es aquella recta que maximice la distancia (máximo margen) a ambas clases (veáse figura 2.10).

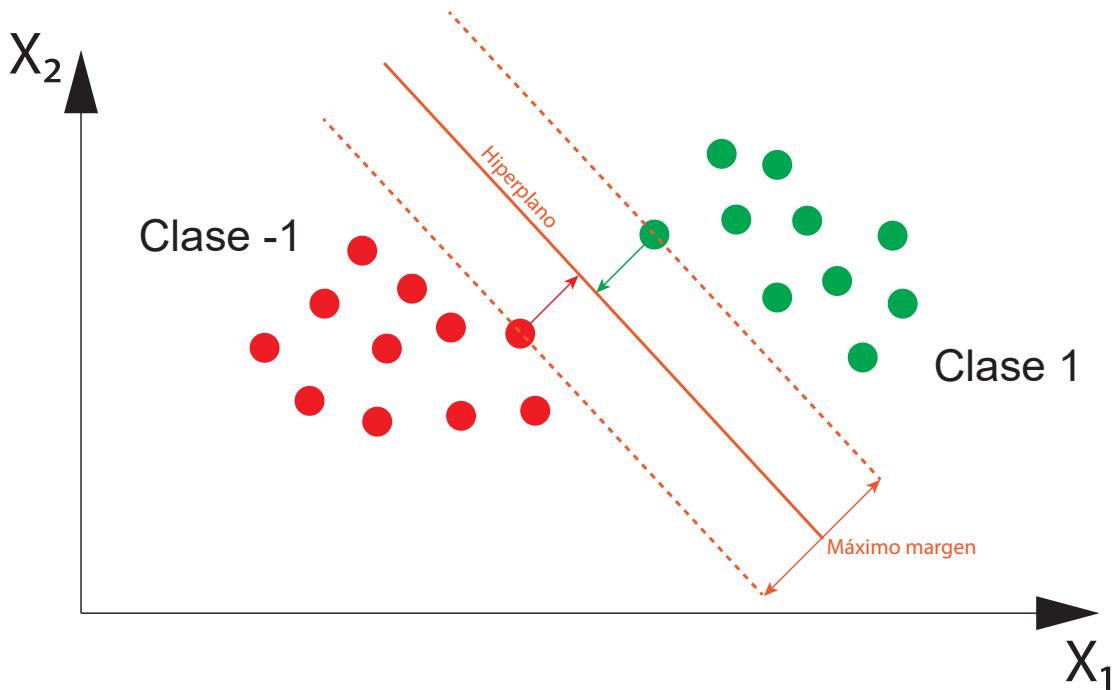


Fig. 2.10. Recta óptima SVM

Matemáticamente, un hiperplano es definible como $w^T x - b = 0$, siendo w el vector normal al hiperplano y $\frac{b}{\|w\|}$ el desplazamiento del hiperplano respecto al origen en la dirección de w .

Existen dos posibles casos según las características de los datos:

Datos linealmente separables

En este caso, es posible definir dos hiperplanos adicionales de la forma $w^T x - b = 1$ (hiperplano superior al principal) y $w^T x - b = -1$ (hiperplano inferior al principal).

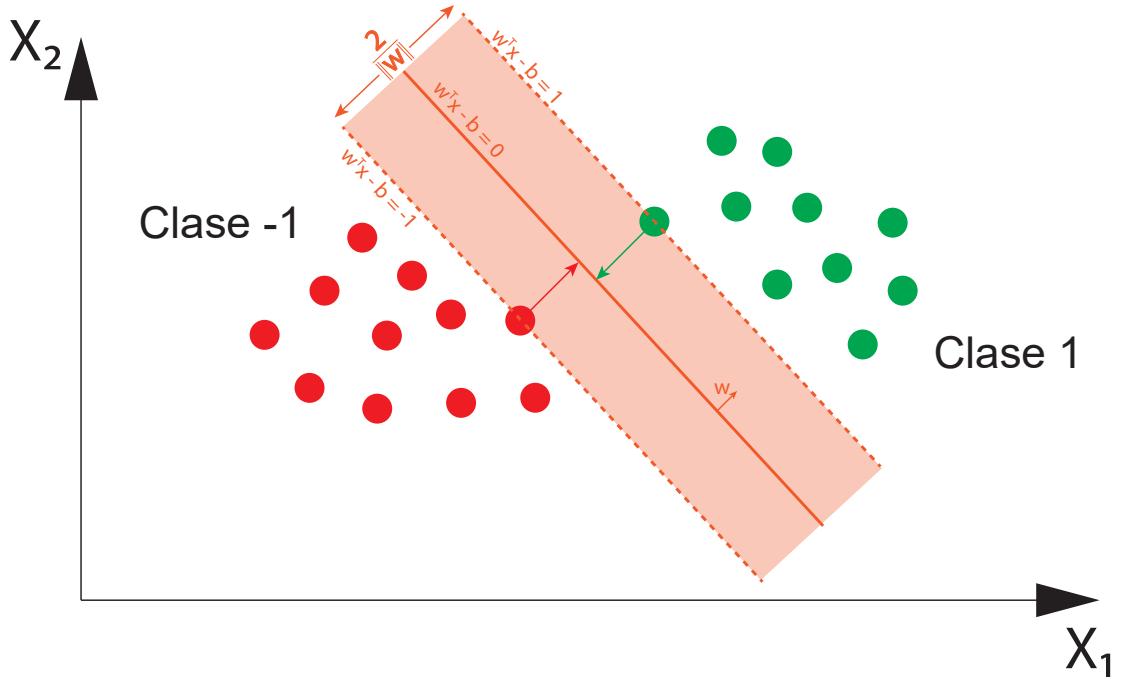


Fig. 2.11. Hiperplanos adicionales SVM

La distancia entre los dos hiperplanos adicionales es $\frac{2}{\|\mathbf{w}\|}$, por tanto, si se quiere maximizar el margen entre ambas rectas, se ha de minimizar $\|\mathbf{w}\|$. Además, dado que ningún punto de ambas clases debe estar entre los dos hiperplanos, se pueden añadir dos condiciones:

$$\mathbf{w}^T \mathbf{x}_i - b \geq 1, \text{ si } y_i = 1$$

$$\mathbf{w}^T \mathbf{x}_i - b \leq 1, \text{ si } y_i = -1$$

Reescribiendo ambas condiciones, se obtiene la expresión 2.23 y junto con la condición de minimizar $\|\mathbf{w}\|$ se define el problema de optimización 2.24:

$$y_i (\mathbf{w}^T \mathbf{x}_i - b) \geq 1, \forall 1 \leq i \leq n \quad (2.23)$$

$$\text{Minimizar } \|\mathbf{w}\| \text{ acorde a la condición } y_i (\mathbf{w}^T \mathbf{x}_i - b) \geq 1, \forall 1 \leq i \leq n \quad (2.24)$$

Datos no linealmente separables

Cuando los datos no son linealmente separables se utiliza la función de pérdida conocida como *Hinge loss* (no tiene traducción formal al castellano). Cuando los datos son inseparables linealmente, se busca minimizar el error (función de pérdida).

Nota:

Una función de pérdida analiza cuánto se ha desviado la predicción del modelo frente al valor real.

La función de pérdida es definida como:

$$h(y) = \max(0, 1 - t \cdot y) \quad (2.25)$$

La variable t representa el valor real a determinar, mientras que y es la predicción del clasificador.

Comparando con la ecuación 2.23, $t = w^T x_i - b = \pm 1$.

En el caso de que el clasificador prediga correctamente la clase, $y = t$, siendo $|y| = 1$ y, por tanto, la función de pérdida $h(y) = \max(0, 0) = 0$. Cuando la predicción no sea correcta, $y = -t$, la función de pérdida $h(y) = \max(0, 1 - (-1) \cdot (1)) = \max(0, 2) = 2$. Es decir, $h(y)$ será nula cuando la predicción sea correcta.

Esto significa que la función de pérdida $h(y)$ será nula cuando x_i se encuentre en el lado correcto del hiperplano y fuera del margen.

Para controlar cuánto error se permite mientras que la mayoría de puntos de cada clase se mantienen en su lado correcto del hiperplano y margen, se define un parámetro $C > 0$. De tal forma, la condición 2.24 se puede reescribir como:

$$\text{Minimizar } C\|w\| + \left[\frac{1}{n} \sum_{i=1}^n \max(0, 1 - y_i(w^T x_i - b)) \right] \quad (2.26)$$

Nota:

El desarrollo para llegar hasta 2.26 es mucho más extenso de lo presentado aquí (en verdad es la leche de largo xd) y no es el propósito de este trabajo.

BIBLIOGRAFÍA

- [1] M.-K. Hu, “Visual pattern recognition by moment invariants,” *IRE Transactions on Information Theory*, vol. 8, n.^o 2, pp. 179-187, 1962. doi: [10.1109/TIT.1962.1057692](https://doi.org/10.1109/TIT.1962.1057692).
- [2] R. K. McConnell, “Method of and apparatus for pattern recognition,” ene. de 1986. [En línea]. Disponible en: <https://www.osti.gov/biblio/6007283>.
- [3] D.-c. He y L. Wang, “Texture Unit, Texture Spectrum, And Texture Analysis,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 28, n.^o 4, pp. 509-512, 1990. doi: [10.1109/TGRS.1990.572934](https://doi.org/10.1109/TGRS.1990.572934).
- [4] J. Frost, *How F-tests work in Analysis of Variance (ANOVA)*, 2020. [En línea]. Disponible en: <https://statisticsbyjim.com/anova/f-tests-anova/>.
- [5] T. Saito y M. Rehmsmeier, “The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets,” *PLOS ONE*, vol. 10, n.^o 3, pp. 1-21, mar. de 2015. doi: [10.1371/journal.pone.0118432](https://doi.org/10.1371/journal.pone.0118432). [En línea]. Disponible en: <https://doi.org/10.1371/journal.pone.0118432>.