SVM con Kernel

Héctor Maravillo

sea $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ un conjunto de patrones, tal que $\mathbf{x}_i \in \mathbb{R}^l$, para $i = 1, \ldots, n$. Sea $\mathbf{w} \in \mathbb{R}^l$ y $b \in \mathbb{R}$ parámetros y $y_i \in \{+1, -1\},\ i = 1, \ldots, n$ las etiquetas de los patrones.

La **Máquina de Soporte Vectorial (SVM)** busca una solución al siguiente problema de optimización con restricciones:

$$\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2 \tag{1}$$

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \qquad \forall i = 1, \ldots, n \tag{2}$$

Este es un problema de **optimización no lineal (cuadrático)** con un conjunto de **restricciones** formado por desigualdades lineales.

Su **función Lagrangiana asociada** es:

$$\begin{aligned} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\lambda}) &= \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{n} \lambda_i [y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1] \quad (3) \\ &= \frac{1}{2}\mathbf{w}^T \mathbf{w} - \sum_{i=1}^{n} \lambda_i [y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1] \quad (4) \end{aligned}$$

donde $\boldsymbol{\lambda} \in \mathbb{R}^n$ es el vector de **multiplicadores de Lagrange**

Las **condiciones de Karush-Kuhn-Tucker (KKT)**[1] aseguran que un mínimo del problema (1) y (2) debe satisfacer que:

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{w}} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\lambda}) &= 0 && (5)\\
\frac{\partial}{\partial b} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\lambda}) &= 0 && (6)\\
\lambda_i &\geq 0, \qquad i = 1, \ldots, n && (7)\\
\lambda_i [y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1] &= 0, \qquad i = 1, \ldots, n && (8)
\end{aligned}$$

[1] Estas condiciones aseguran condiciones necesarias y suficientes para problemas de optimización no lineal con restricciones de desigualdad.

Las restricciones anteriores, (5) y (6), equivalen a:

$$\begin{aligned} \mathbf{w} - \sum_{i=1}^{n} \lambda_i y_i \mathbf{x}_i &= 0 \\ \sum_{i=1}^{n} \lambda_i y_i &= 0 \end{aligned}$$

Las restricciones anteriores, (5) y (6), equivalen a:

$$\begin{aligned} \mathbf{w} &= \sum_{i=1}^{n} \lambda_i y_i \mathbf{x}_i \\ \sum_{i=1}^{n} \lambda_i y_i &= 0 \end{aligned}$$

- La condición (7) asegura que los multiplicadores deben ser cero o positivo, esto significa que el vector **w** de la solución óptima es una combinación lineal de $n_s \leq n$ vectores de características asociados con los multiplicadores $\lambda \neq 0$ (restricciones activas). Es decir,

$$\mathbf{w} = \sum_{i=1}^{n_s} \lambda_i y_i \mathbf{x}_i$$

  Estos vectores de característica se conocen como **vectores de soporte** y se encuentran en uno de los márgenes. Es decir, son los que se encuentran más cercanos al clasificador lineal.

- La función de costo (1) es **estrictamente convexa**, debido a que su matriz Hessiana H es **positiva definida**, esto es:

$$\mathbf{z}^T H\mathbf{z} > 0, \qquad \forall \mathbf{z} \in \mathbf{R}^l, \ \mathbf{z} \neq \mathbf{0}$$

además las restricciones (2) son definidas por desigualdades lineales. Por lo tanto, cualquier **mínimo local** es también es **mínimo global**, y éste es **único**.

Debido a que el problema de optimización (1), (2) es convexo, la **representación dual de Wolfe**[2] equivale a

$$\max_{\boldsymbol{\lambda}} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\lambda}) \tag{9}$$

$$\mathbf{w} = \sum_{i=1}^{n} \lambda_i y_i \mathbf{x}_i \tag{10}$$

$$\sum_{i=1}^{n} \lambda_i y_i = 0 \tag{11}$$

$$\boldsymbol{\lambda} \geq 0 \tag{12}$$

[2] El problema dual es una reformulación del problema original (primal) que se basa en el Lagrangiano e introduce variables asociadas a las restricciones.

Sustituyendo la ecuación (10) y (11) en la función objetivo, y usando un poco de algebra, el **dual Lagrangiano** del **SVM (hard-margin)** es:

$$\max_{\boldsymbol{\lambda}} \quad \left(\sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j) \right) \tag{13}$$

$$\sum_{i=1}^{n} \lambda_i y_i = 0 \tag{14}$$

$$0 \leq \lambda_i \qquad i = 1, \ldots, n \tag{15}$$

- Los multiplicadores de Lagrange óptimos $\boldsymbol{\lambda}^*$ se calculan mediante (13).
- Usando estos valores se puede calcular el vector del hiperplano óptimo $\mathbf{w}^*$ vía (10).
- El valor del umbral óptimo b^* se calcula usando los dos resultados anteriores en la restricción (7).

Hay dos elementos importantes de la formulación dual anterior:

- La función objetivo no depende explícitamente de la dimensión del espacio de entradas, sino del **número de datos** de entrenamiento, ya que $\boldsymbol{\lambda} \in \mathbb{R}^n$
- La función objetivo sólo depende de los datos de entrenamiento en términos de su **producto punto**.

En la Máquina de Soporte Vectorial con restricciones suaves, el problema primal es:

$$\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{n} \epsilon_i \tag{16}$$

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \epsilon_i \qquad \forall i = 1, \ldots, n \tag{17}$$

$$\epsilon_i \geq 0, \qquad \forall i = 1, \ldots, n \tag{18}$$

donde ϵ_i son conocidas como **variables de holgura**.

El **problema dual** correspondiente equivale a:

$$\max_{\boldsymbol{\lambda}} \quad \left(\sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j) \right) \tag{19}$$

$$\sum_{i=1}^{n} \lambda_i y_i = 0 \tag{20}$$

$$0 \leq \lambda_i \leq C \qquad i = 1, \ldots, n \tag{21}$$

En este caso los multiplicadores de Lagrange que corresponden a los puntos que están en la franja de los márgenes o en el lado incorrecto del clasificador (es decir, $\epsilon_i > 0$) son todos iguales al máximo valor C.

Truco del Kernel

Si el conjunto de patrones X no es linealmente separables, el SVM (hard-margin) será incapaz de encontrar un hiperplano óptimo.

Una idea es **mapear** los datos a una **dimensión más alta**, con la **esperanza** que en esa nueva dimensión los datos sean **linealmente separables**.

$$x \in \mathbb{R}^l \to z \in \mathbb{R}^k, \qquad l < k \tag{22}$$

Esta idea tiene dos problemas a resolver:

- ¿Que transformación es la más adecuada?
- ¿Cómo resolver eficientemente el SVM en una dimensión más grande?