MÁSTER EN TRATAMIENTO ESTADÍSTICO-COMPUTACIONAL DE LA INFORMACIÓN

TRABAJO FIN DE MÁSTER

Estudio de modelos de machine learning para clasificación

Fernando Sánchez Dorado

UNIVERSIDAD POLITÉCNICA DE MADRID
ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE TELECOMUNICACIÓN
&
UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE CIENCIAS MATEMÁTICAS

Tutores: Benjamin Ivorra, Ángel Manuel Ramos del Olmo, Miguel Carrasco Briones, Julio López

Madrid, Junio 2025

Resumen

Este Trabajo de Fin de Máster tiene como objetivo el estudio y la implementación en MATLAB de diversas variantes del modelo de aprendizaje automático Support Vector Machine (SVM). En primer lugar, se presentan los fundamentos teóricos del modelo, junto con sus variantes básicas, para posteriormente introducir versiones más complejas con el fin de profundizar en su análisis. Una vez implementadas las distintas variantes, se lleva a cabo un proceso de análisis numérico que permite evaluar su rendimiento en diferentes contextos. Finalmente, se extraen e interpretan las conclusiones obtenidas a partir de los resultados experimentales.

Abstract

This Master's Thesis aims to study and implement various variants of the Support Vector Machine (SVM) model in MATLAB. The work begins with an explanation of the theoretical foundations of the model, along with its basic variants, and then introduces more complex versions for further exploration. Once the models have been introduced and implemented, a numerical analysis is conducted to evaluate their performance in different scenarios. Finally, conclusions are drawn from the experimental results and are thoroughly interpreted.

Índice

1. Introducción

El presente Trabajo Fin de Máster tiene como objetivo analizar y comparar diversas variantes de modelos de clasificación en el ámbito del Machine Learning, con especial énfasis en el modelo Support Vector Machine (SVM).

El SVM, desarrollado por Vapnik [1] en 1995, se ha consolidado como una herramienta potente para tareas de clasificación, aunque también es aplicable a problemas de regresión (este último caso no se aborda en el presente estudio). Entre sus principales virtudes destacan su elevada capacidad de generalización y su eficacia en aplicaciones como la clasificación de imágenes, el reconocimiento de voz y la predicción de series temporales, entre otros.

El propósito de este trabajo es implementar, analizar y documentar en detalle diferentes variantes del modelo SVM, complementándolo con un proceso de validación experimental sobre un conjunto de datos. De esta manera, se pretende extraer conclusiones relevantes acerca del impacto de las modificaciones introducidas en el modelo.

1.1. Objetivos

Se han planteado los siguientes objetivos en este trabajo:

- Lectura y análisis de varios artículos sobre aprendizaje automático.
- Implementación en matlab de algoritmos relacionados con los temas estudiados.
- Utilización y adaptación de dichos algoritmos para resolver y simular casos de interés.
- Redacción de un informe.

1.2. Estructura

En la sección 1 se presentan la introducción, los objetivos del trabajo y el enfoque general del estudio, proporcionando al lector el contexto y la motivación de este Trabajo de Fin de Máster.

La sección 2 introduce una serie de conceptos teóricos fundamentales que sirven de base para el resto del documento. En particular, se repasan nociones de convexidad (conjuntos y funciones convexas) y fundamentos de optimización convexa, como la diferenciación entre problemas con y sin restricciones, así como el principio de dualidad en problemas de optimización. También se explican las condiciones de óptimo de Karush–Kuhn–Tucker (KKT), ya que estos conceptos resultan esenciales para comprender las formulaciones matemáticas de los modelos SVM desarrollados posteriormente.

En la sección 3 se presentan las formulaciones clásicas de los SVM. Inicialmente, se aborda el caso ideal de clasificación con margen duro (hard margin), en el que los datos son linealmente separables sin error. A continuación, se extiende el análisis al caso de margen suave (soft margin), que permite cierta superposición entre clases introduciendo tolerancia a errores de clasificación. Posteriormente, se introducirá el uso de kernels para proyectar los datos a espacios de mayor dimensión, lo que permite aplicar SVM en problemas no lineales. Para facilitar la comprensión de estos modelos, la exposición se acompaña de interpretaciones geométricas y ejemplos gráficos ilustrativos.

La sección 4 está dedicada al estudio de variantes del modelo SVM y a las funciones de pérdida asociadas. En primer lugar, se presenta el Probabilistic SVM (PSVM), una variante del modelo SVM, detallando su estructura y formulación, así como su relación con la clasificación probabilística. A continuación, se describirá el concepto de función de pérdida y se explicará las dos funciones de perdidas relevantes en estos modelos: la función Hinge, empleada en el SVM estándar, y la función Pinball, que se introduce como una alternativa novedosa. Se explica cómo esta última propone una manera diferente de penalizar los errores de clasificación en comparación con la Hinge tradicional, en el modelo PSVM, sentando las bases teóricas para su evaluación posterior.

En la sección 5 recogen los experimentos numéricos realizados para evaluar el rendimiento de los distintos modelos desarrollados. Se aplican los SVM clásicos y sus variantes (incluyendo PSVM con función Pinball) sobre varios conjuntos de datos, con el fin de comparar cuantitativamente su desempeño. Se analizan los resultados obtenidos y se discuten las diferencias en rendimiento entre las metodologías, lo que permite extraer conclusiones sobre el impacto de las modificaciones introducidas en el modelo.

Por último, la sección 6 se presentan las conclusiones de lo presentado en el trabajo. Primero, explicará lo estudiado en cada una de las secciones del trabajo; luego se discutirán las principales limitaciones detectadas durante su desarrollo; y, finalmente, se proponen líneas de investigación que puedan ampliar y profundizar este estudio.

2. Nociones básicas de convexidad y optimización convexa

2.1. Convexidad

En esta sección se presentan las definiciones de conjunto convexo y función convexa y cóncava. Además, se explica su papel en los modelos SVM.

Definición 1 (Conjunto convexo)**.** Un conjunto convexo $\Omega \subset \mathbb{R}^N$ es aquel que cumple que para cualquier $x_1, x_2 \in \Omega$ y $\lambda \in [0, 1]$ se verifica que:

$$\lambda x_1 + (1 - \lambda)x_2 \in \Omega. \tag{1}$$

Es decir que el segmento que une x_1 y x_2 está contenido en Ω.

Definición 2 (Funciones convexas)**.** Dado un conjunto convexo $\Omega \subset \mathbb{R}^N$ se dice que una función $f : \Omega \to \mathbb{R}$ es convexa si, para cualquier $x_1, x_2 \in \Omega$ y cualquier $\lambda \in [0, 1]$ se cumple que:

$$f(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda f(x_1) + (1 - \lambda)f(x_2). \tag{2}$$

Definición 3 (Función convexa estricta)**.** Dado un conjunto convexo $\Omega \subset \mathbb{R}^N$ se dice que una función $f : \Omega \to \mathbb{R}$ es convexa estricta si, para cualquier $x_1, x_2 \in \Omega$ tal que $x_1 \neq x_2$ y cualquier $\lambda \in (0, 1)$ se cumple que:

$$f(\lambda x_1 + (1 - \lambda)x_2) < \lambda f(x_1) + (1 - \lambda)f(x_2). \tag{3}$$

Definición 4 (Funciones cóncavas)**.** Dado un conjunto convexo $\Omega \subset \mathbb{R}^N$ se dice que una función $g : \Omega \to \mathbb{R}$ es cóncava si, para cualquier $x_1, x_2 \in \Omega$ y cualquier $\lambda \in [0, 1]$ se cumple que:

$$g(\lambda x_1 + (1 - \lambda)x_2) \geq \lambda g(x_1) + (1 - \lambda)g(x_2). \tag{4}$$

Nótese que g es cóncava si y solo si $-g$ es convexa.

Definición 5 (Función cóncava estricta)**.** Dado un conjunto convexo $\Omega \subset \mathbb{R}^N$ se dice que una función $g : \Omega \to \mathbb{R}$ es cóncava estricta si, para cualquier $x_1, x_2 \in \Omega$ tal que $x_1 \neq x_2$ y cualquier $\lambda \in (0, 1)$ se cumple que:

$$g(\lambda x_1 + (1 - \lambda)x_2) > \lambda g(x_1) + (1 - \lambda)g(x_2). \tag{5}$$

La convexidad es una propiedad importante en los modelos SVM, ya que garantiza que, al aplicar los algoritmos de optimización que se describirán en el siguiente apartado, se pueda encontrar valores óptimos.

2.2. Optimización

Una vez explicado la definición de un conjunto convexo y de una función cóncava y convexa se explicará algunas nociones necesarias de optimización.

Definición 6 (Mínimo y máximo global)**.** [2] Sea $C \subset \mathbb{R}^N$ y $f : C \to \mathbb{R}$ una función. Se dice que un punto $x^* \in C$ es:

$$\text{mínimo global si } f(\mathbf{x}^*) \leq f(\mathbf{x}), \forall \mathbf{x} \in C, \tag{6}$$
$$\text{mínimo global estricto si } f(\mathbf{x}^*) < f(\mathbf{x}), \forall \mathbf{x} \in C \text{ con } x \neq \mathbf{x}^*, \tag{7}$$
$$\text{máximo global si } f(\mathbf{x}^*) \geq f(\mathbf{x}), \forall \mathbf{x} \in C, \tag{8}$$
$$\text{máximo global estricto si } f(\mathbf{x}^*) > f(\mathbf{x}), \forall \mathbf{x} \in C \text{ con } \mathbf{x} \neq x^*. \tag{9}$$

Definición 7 (Mínimo y máximo local)**.** [2] Sea $C \subset \mathbb{R}^N$ y $f : C \to \mathbb{R}$. Se dice que $\mathbf{x}^* \in C$ es:

mínimo local si existe una bola $B(\mathbf{x}^*, r)$ centrada en $\mathbf{x}^*$ de radio $r > 0$ tal que:

$$f(\mathbf{x}^*) \leq f(\mathbf{x}), \forall \mathbf{x} \in C \cap B(\mathbf{x}^*, r), \tag{10}$$

mínimo local estricto si existe una bola $B(\mathbf{x}^*, r)$ centrada en $\mathbf{x}^*$ de radio $r > 0$ tal que:

$$f(\mathbf{x}^*) < f(\mathbf{x}), \forall \mathbf{x} \in C \cap B(\mathbf{x}^*, r) \text{ con } \mathbf{x} \neq \mathbf{x}^*, \tag{11}$$

máximo local si existe una bola $B(x^*, r)$ centrada en x^* de radio $r > 0$ tal que:

$$f(\mathbf{x}^*) \geq f(\mathbf{x}), \forall \mathbf{x} \in C \cap B(\mathbf{x}^*, r), \tag{12}$$

máximo local estricto si existe una bola $B(\mathbf{x}^*, r)$ centrada en $\mathbf{x}^*$ de radio $r > 0$ tal que:

$$f(\mathbf{x}^*) > f(\mathbf{x}), \forall \mathbf{x} \in C \cap B(\mathbf{x}^*, r) \text{ con } \mathbf{x} \neq \mathbf{x}^*. \tag{13}$$

Un mínimo global será un mínimo local, pero no se dará lo mismo al revés.

Una vez definidos estos conceptos se pasará a definir distintos métodos de encontrar máximos y mínimos.

2.2.1. Optimización sin restricciones

Supóngase que $\Omega \subset \mathbb{R}^N$ y se quiere encontrar el mínimo de una función $f : \Omega \to \mathbb{R}$, por tanto, se tiene el siguiente problema de optimización:

$$\min_{\mathbf{x} \in \Omega} f(\mathbf{x}). \tag{14}$$

Para resolver problemas de optimización sin restricciones se emplean principalmente dos tipos de métodos según [3]:

Definición 8 (Método analítico de primer orden)**.** [3] Consiste en encontrar $\mathbf{x}^* \in \Omega$ tal que

$$\nabla f(\mathbf{x}^*) = 0, \tag{15}$$

donde $\nabla f(\mathbf{x})$ es el vector de derivadas parciales de f.

Definición 9 (Método de descenso por gradiente)**.** [3] Parte de $x^{(0)} \in \Omega$ y define iterativamente

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha^{(k)} \nabla f\left(\mathbf{x}^{(k)}\right), \quad \alpha^{(k)} > 0, \tag{16}$$

donde $\alpha^{(k)}$ es el tamaño de paso, es decir la magnitud del desplazamiento desde $\mathbf{x}^{(k)}$ en la dirección del gradiente negativo, en la iteración k.

No obstante, estos enfoques no se emplean en los modelos SVM y no se analizarán en detalle en este trabajo, ya que la construcción de un SVM implica restricciones que requieren métodos de optimización con restricciones.

2.2.2. Optimización con restricciones

La inclusión de restricciones en la formulación de los modelos SVM da lugar a problemas de optimización con restricciones, en los que la función objetivo no puede minimizarse de forma directa mediante los métodos vistos anteriormente.

A continuación, se describirán las bases teóricas y las técnicas empleadas para resolver estos problemas en el contexto de los SVM. Supongamos que queremos resolver el siguiente problema:

$$\begin{aligned} &\min_{\mathbf{x}\in\Omega} f(\mathbf{x}) \\ &\text{s.a.} \quad g_i(\mathbf{x}) \leq 0, \quad i = 1, 2, \ldots, m, \\ &\qquad h_j(\mathbf{x}) = 0, \quad j = 1, 2, \ldots, p, \end{aligned} \tag{17}$$

donde, además de indicar la función que se busca minimizar f, se dan dos tipos de funciones: las funciones con restricciones de desigualdad g_i y funciones con restricciones de igualdad h_j.

Para minimizar estos problemas no basta únicamente con encontrar el valor mínimo de la función objetivo, también es necesario encontrar un $\mathbf{x}^*$ tal que cumpla con las distintas restricciones. Es decir, se buscan soluciones en el conjunto:

$$S = \{\mathbf{x} \in \Omega : g_i(\mathbf{x}) \leq 0 \quad i = 1, \ldots, m, h_j(\mathbf{x}) = 0 \quad j = 1, \ldots, p\}. \tag{18}$$

Es por esto que enfoques como los explicados en apartados anteriores, como el descenso del gradiente, no serán utilizados para minimizar estos problemas.

También para poder resolver este tipo de problemas conviene saber si el problema cumple ciertas propiedades de convexidad:

Lema 1 (Convexidad de una función cuadrática)**.** Sea $f : \mathbb{R}^N \to \mathbb{R}$ la función cuadrática

$$f(\mathbf{x}) \;=\; \tfrac{1}{2}\,\mathbf{x}^T Q\,\mathbf{x} + r^T\mathbf{x} + \delta,$$

con $Q \in \mathbb{R}^{N\times N}$ simétrica, $r \in \mathbb{R}^N$ y $\delta \in \mathbb{R}$. Si

$$Q \succeq 0,$$

es decir, si Q es semidefinida positiva, entonces f es una función convexa.

La demostración se encuentra en [4] en las secciones 1.2.5 y 1.2.6.

En todo lo que sigue vamos a suponer que f y todas las restricciones g_i son convexas.

Por último, vamos a suponer que las restricciones de igualdad son afines es decir, de la forma:

$$h_j(\mathbf{x}) = a_j^T\mathbf{x} + b_j = 0, \tag{19}$$

con $a_j \in \mathbb{R}^N$ y $b_j \in \mathbb{R}$ para $j = 1, \ldots, p$, pues un plano o subespacio afín preserva la convexidad al intersectarse con conjuntos convexos. Si h_j fuese no lineal, la intersección podría no ser convexa y el problema dejaría de ser convexo.

Si se cumplen las condiciones anteriores, se tiene que S es un conjunto convexo [4].

Para resolver este tipo de problemas, se presentará el enfoque basado en las condiciones de Karush–Kuhn–Tucker (KKT), las cuales constituyen un método fundamental para encontrar soluciones que no solo satisfacen las restricciones del problema, sino que además optimizan la función objetivo.

Antes de introducir las condiciones de KKT, se explicará el concepto de dualidad, ya que a partir de él se puede comprender con claridad la formulación del problema primal y su correspondiente problema dual.

2.2.3. Dualidad

Un problema de optimización con restricciones, como el que se presenta en (17), puede formularse de dos maneras. La primera es la denominada forma primal, que corresponde directamente con la expresión mostrada en (17).

La segunda es la forma dual, la cual se obtiene a partir de la primal y permite abordar el problema de manera equivalente, pero con frecuencia de forma más eficiente.

Para calcular la forma dual del problema (17), se define la función Lagrangiana $L : \Omega \times \{\boldsymbol{\lambda} \in \mathbb{R}^{\mathbf{m}} : \boldsymbol{\lambda} \geq \mathbf{0}\} \times \mathbb{R}^p \to \mathbb{R}$, que incorpora las restricciones al problema de optimización (17) en una única función de la siguiente forma:

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i g_i(\mathbf{x}) + \sum_{j=1}^{p} \mu_j h_j(\mathbf{x}). \tag{20}$$

Así:

$$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \leq f(\mathbf{x}) \quad \forall x \in S. \tag{21}$$

Entonces, si denotamos por p^* a:

$$p^* = \inf_{\mathbf{x} \in S} f(\mathbf{x}), \tag{22}$$

se tiene que:

$$\inf_{\mathbf{x} \in \Omega} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \leq \inf_{\mathbf{x} \in S} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \leq \inf_{\mathbf{x} \in S} f(\mathbf{x}) = p^*. \tag{23}$$

La segunda desigualdad refleja que, al cumplirse $L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \leq f(\mathbf{x})$ para todo $\mathbf{x} \in S$, su ínfimo en S no puede superar a p^*. Por tanto, si se define:

$$f_d(\boldsymbol{\lambda}, \boldsymbol{\mu}) = \inf_{\mathbf{x} \in \Omega} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}), \tag{24}$$

para cualquier $\boldsymbol{\lambda}$ y $\boldsymbol{\mu}$, con $\boldsymbol{\lambda} \geq 0$ se sabe por (23), que:

$$f_d(\boldsymbol{\lambda}, \boldsymbol{\mu}) \leq p^*, \tag{25}$$

es decir, puesto que la función dual proporciona siempre una cota inferior de p^*, el objetivo es encontrar el mayor de esos valores maximizando $f_d(\mathbf{x})$. Para ello se plantea el siguiente problema de optimización:

$$\begin{aligned} &\max_{\boldsymbol{\lambda} \in \mathbb{R}^m, \boldsymbol{\mu} \in \mathbb{R}^p} \quad f_d(\boldsymbol{\lambda}, \boldsymbol{\mu}) \\ &\text{s.a.} \quad \boldsymbol{\lambda} \geq 0. \end{aligned} \tag{26}$$

Esta es la forma dual del problema (17). Tanto el problema dual como el primal son convexos (ver teorema 1.2.17 [4]).

En lo que sigue usaremos la siguiente notación:

$$d^* = \sup\{f_d(\boldsymbol{\lambda}, \boldsymbol{\mu}), \boldsymbol{\lambda} \in \mathbb{R}^m, \boldsymbol{\mu} \in \mathbb{R}^p, \boldsymbol{\lambda} \geq 0\}. \tag{27}$$

A partir de esta formulación se dan las siguientes definiciones.

En las siguientes definiciones se dice que el ínfimo y el supremo son "valores óptimos"del problema. Sin embargo, el *ínfimo* de $\{f(\mathbf{x}) : \mathbf{x} \in S\}$ es la mayor cota inferior y no implica que exista $\mathbf{x}^* \in S$ con $f(\mathbf{x}^*) = \text{ínf}\, f$. El *mínimo* sí requiere que se alcance ese valor.

Análogamente, el *supremo* es la menor cota superior (puede no lograrse) y el *máximo* exige existir x^* con $f(x^*) = \sup f$. Por tanto, cuando se afirma que p^* o d^* son "óptimos" en los apartados siguientes, se está usando "óptimo" en el sentido de ínfimo o supremo. Solo si se demuestra que existe un punto que lo cumple, se podrá hablar de "mínimo" o "máximo" propiamente dichos.

Observación 1 (Dualidad débil)**.** Dado un problema convexo como el expresado en la ecuación (17), donde $p^* = \text{ínf}\{f(\mathbf{x}) | \mathbf{x} \in S\}$ representa la solución óptima de este y d^* la solución óptima de su problema dual (26), se cumple, tal y como se ha mostrado anteriormente, que:

$$p^* \geq d^*. \tag{28}$$

Además de la dualidad débil, existe la dualidad fuerte, la cual se satisface si se cumplen las siguientes condiciones.

Definición 10 (Condición de Slater)**.** Un problema convexo, como el planteado en (17), y sabiendo por (19) que h_j es afín (es decir, $h_j(\mathbf{x}) = a_j^T \mathbf{x} + b_j, \quad j = 1, \ldots, p, \quad a_j \in \mathbb{R}^N, \ b_j \in \mathbb{R}$), satisface la condición de Slater si existe $\mathbf{x} \in \Omega$ tal que:

$$g_i(\mathbf{x}) < 0, \forall i = 1, \ldots, m, \quad a_j^T \mathbf{x} + b_j = 0, \forall j = 1, \ldots, p \text{ con } a_j \in \mathbb{R}^N, b_j \in \mathbb{R} \tag{29}$$

o si, las primeras k condiciones de desigualdad son lineales, entonces:

$$\begin{aligned} &g_i(\mathbf{x}) \leq 0, i = 1, \ldots, k, \\ &g_i(\mathbf{x}) < 0, i = k+1, \ldots, m, \\ &a_j^T \mathbf{x} + b_j = 0, j = 1, \ldots, p. \end{aligned} \tag{30}$$

A partir de la condición de Slater se puede definir la dualidad fuerte:

Proposición 1 (Dualidad fuerte)**.** Dado un problema convexo como (17) que satisface la condición de Slater, con $p^* = \text{ínf}\{f(\mathbf{x})|\mathbf{x} \in S\}$ como su solución óptima y por d^* como la solución óptima de su problema dual (26), se cumple que:

$$p^* = d^*. \tag{31}$$

La demostración se puede encontrar en la sección 1.2.3.2 de [4].

2.2.4. Condiciones de KKT

Como se explicó en secciones anteriores, un problema de optimización con restricciones no se puede resolver utilizando esquemas como los indicados en el apartado de optimización sin restricciones; para ello se usan otros enfoques como las condiciones de KKT, que se explican a continuación:

Definición 11 (Condiciones de KKT)**.** Si se tiene un problema de optimización como (17),se dice que un punto $\mathbf{x}^*$ cumple las condiciones de KKT si existe $\boldsymbol{\lambda} \in \mathbb{R}^m$ y $\boldsymbol{\mu} \in \mathbb{R}^p$ tal:

$$g_i(\mathbf{x}^*) \leq 0, \quad i = 1, \ldots, m, \tag{32}$$

$$h_j(\mathbf{x}^*) = 0, \quad j = 1, \ldots, p, \tag{33}$$

$$\lambda_i \geq 0, \quad i = 1, .., m, \tag{34}$$

$$\lambda_i g_i(\mathbf{x}^*) = 0, \quad i = 1, .., m, \tag{35}$$

$$\nabla_x L(\mathbf{x}^*, \boldsymbol{\lambda}, \boldsymbol{\mu}) = \nabla f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i \nabla g_i(\mathbf{x}^*) + \sum_{j=1}^{p} \mu_j \nabla h_j(\mathbf{x}^*) = 0. \tag{36}$$

En problemas convexos que satisfacen la condición de Slater, las condiciones KKT son necesarias y suficientes para garantizar que un punto es solución óptima de (17). Esto permite caracterizar completamente la solución a través de dichas condiciones [4].

Un aspecto importante es que, si un problema como (17) cumple la condición de Slater y $\mathbf{x}^* \in \mathbb{R}^N$ es una solución óptima, entonces $\mathbf{x}^*$ satisface las condiciones de Karush–Kuhn–Tucker (KKT). La demostración puede consultarse en [4].

A continuación se explicará qué simboliza cada una de las condiciones que se han explicado en la definición 11 de las condiciones de KKT.

La última condición (36) es la condición de estacionariedad, implica que, en el punto óptimo, el gradiente de la función objetivo puede escribirse como combinación lineal de los gradientes de todas las restricciones activas. Es precisamente el equivalente, en presencia de restricciones, a imponer $\nabla f(x^*) = 0$ cuando no hubiera ninguna restricción.

Las expresiones (32) y (33) corresponden a las condiciones de factibilidad del problema, es decir las restricciones de igualdad y desigualdad explicadas al introducir (17).

Para el multiplicador $\boldsymbol{\lambda}$ deben cumplirse las condiciones de positividad (34) ($\lambda_i \geq 0$) y de complementariedad (35). La restricción de complementariedad se sigue de (1.2.52) de [4].

Ahora estas dos restricciones tienen el siguiente comportamiento conjunto, si en un punto $\mathbf{x}^*$ se cumple $g_i(\mathbf{x}^*) < 0$ (la restricción está inactiva), la condición de complementariedad $\lambda_i\, g_i(\mathbf{x}^*) = 0$ obliga a que $\lambda_i = 0$. En cambio, si $g_i(\mathbf{x}^*) = 0$ (la restricción está activa), entonces $\lambda_i \geq 0$ puede ser estrictamente positivo ya que se sigue cumpliendo la condición $\lambda_i g_i(\mathbf{x}^*) = 0$.

3. Formulaciones Preliminares y Conceptos Básicos de Modelos SVM

3.1. Conceptos básicos de modelos SVM

A continuación se explicarán las distintas variantes de modelos SVM; primero se introducirán conceptos importantes para entender los distintos modelos SVM que existen.

Definición 12 (Problema de clasificación)**.** Un problema de clasificación con 2 clases viene determinado por un conjunto de datos $(\mathbf{x}, \mathbf{y})$, donde $\mathbf{x}$ representa las características de entrada e $\mathbf{y}$ las etiquetas correspondientes, es decir:

$$\{(\mathbf{x}, \mathbf{y}) \mid \mathbf{x} \in \mathbb{R}^N,\ \mathbf{y} \in \{+1, -1\}\}. \tag{37}$$

Un problema de clasificación consiste en determinar un hiperplano capaz de separar de manera óptima las dos clases $\{+1, -1\}$.

Dicho hiperplano se conoce como el hiperplano separador, y es fundamental para asignar correctamente la etiqueta correspondiente a cada dato, en función de su posición relativa en el espacio de características:

Definición 13 (Hiperplano de separación)**.** Un *hiperplano de separación*, es una superficie en $\mathbb{R}^N$ cuya finalidad es separar entre sí las distintas clases del conjunto $\{+1, -1\}$. Su objetivo es establecer fronteras claras que permitan distinguir correctamente los elementos pertenecientes a cada clase. Cada uno de los planos de separación está determinado por $\mathbf{w}$, que es la orientación del hiperplano y $\mathbf{b}$ que es la posición del hiperplano con respecto al origen. Se tiene que un punto $\mathbf{x}$ está en dicho plano sí:

$$f(\mathbf{x}) = \mathbf{w}^{\mathbf{T}}\mathbf{x} + b = 0, \quad \mathbf{w} \in \mathbb{R}^N,\ b \in \mathbb{R}. \tag{38}$$

Para definir la clase de un dato $\mathbf{x}^*$, se considera:

$$f(\mathbf{x}^*) = \mathbf{w}^\top \mathbf{x}^* + b, \tag{39}$$

y luego, se asigna al dato $\mathbf{x}^*$ la clase, según el siguiente criterio:

$$y^* = \operatorname{sign}\big(f(\mathbf{x}^*)\big) = \begin{cases} +1, & f(\mathbf{x}^*) > 0, \\ -1, & f(\mathbf{x}^*) \leq 0. \end{cases} \tag{40}$$

Otro concepto importante es el de separabilidad entre clases, algunas veces no existe hiperplano separador esto sucede cuando:

Definición 14 (Separabilidad entre clases)**.** Dadas dos clases,$\{C_1, C_2\}$ que tienen los siguientes hiperplanos de separación $f_j \to \mathbb{R} \quad j = [1, 2]$. Se considera que son separables si:

$$y_i f_j(\mathbf{x}) > 0 \quad \forall (\mathbf{x}, \mathbf{y}), \mathbf{x} \in C_i. \tag{41}$$

Otros conceptos importantes son los Kernels. Los modelos Kernel se explicarán más adelante, pero antes se introducirán algunas definiciones:

Definición 15 (Función Kernel)**.** Sea $\Omega \subseteq \mathbb{R}^N$ y sea $\Omega : \Omega \times \Omega \to \mathbb{R}$. Decimos que Ω es un *kernel* si existe un espacio de Hilbert H y un mapeo (o *mapa de características*) $\phi : \Omega \to H$ tales que, para cualesquiera $\mathbf{x}, \mathbf{y} \in \Omega$,

$$\Omega(\mathbf{x}, \mathbf{y}) \;=\; \langle \phi(\mathbf{x}), \phi(\mathbf{y}) \rangle,$$

donde $\langle \cdot, \cdot \rangle$ denota el producto interno en H.

A continuación se presenta un ejemplo de una función Kernel para ilustrar el concepto: Se tiene la siguiente función de mapeo ϕ:

$$\phi(\mathbf{x}) : (x_1, x_2) \to (x_1^2, \sqrt{2} x_1 x_2, x_2^2). \tag{42}$$

Por lo que se puede formular el Kernel de la siguiente manera:

$$\Theta(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x})^T \phi(\mathbf{y}) = x_1^2 x_2^2 + 2 x_1 x_2 y_1 y_2 + x_2^2 y_2^2. \tag{43}$$

3.2. Modelos SVM: Hard margin

Este esquema representa la forma más elemental de los modelos SVM. En él se asume que las dos clases son linealmente separables (véase la definición 14) y se enmarca en un problema de clasificación binaria, es decir, con únicamente dos clases.

El objetivo es encontrar un hiperplano de separación, tal como el definido en la ecuación (13), que clasifique correctamente un conjunto de datos

$$\mathcal{C} = \{\, \mathbf{x}_i \mid i = 1, \ldots, m \},$$

asignando cada punto $\mathbf{x}_i$ a una de las dos etiquetas $y_i \in \{+1, -1\}$ siendo $\mathcal{C}_1 = +1$ y $\mathcal{C}_2 = -1$. Dicho hiperplano debe, además, maximizar la distancia (margen) hasta el punto más cercano de cada clase, garantizando así una mayor capacidad de generalización. A continuación se formula el problema de optimización con restricciones correspondiente:

$$\begin{aligned} \max_{\mathbf{w},b} \quad & \min\{||\mathbf{x} - \mathbf{x_i}||^2, \mathbf{w^T x} + b = 0, i = 1, .., m\} \\ s.a. \quad & y_j(\mathbf{w^T x}_j + b) \geq 1, \quad j = 1, \ldots, m, \end{aligned} \tag{44}$$

donde $||.||$ denota la norma Euclidiana.

A partir de este problema se pueden buscar reformulaciones más simples, lo primero que se puede hacer es que a partir del hiperplano $\mathbf{w^T x} + b = 0$ se pueden construir otros dos hiperplanos.

Uno de estos hiperplanos es el correspondiente a la clase 1,

$$\mathbf{w^T x}_j + b = 1, j = 1, \ldots, m_1, \tag{45}$$

y el otro sería el correspondiente a la clase 2

$$\mathbf{w^T x}_j + b = -1, j = 1, \ldots, m_2, \tag{46}$$

siendo m_1 y m_2 el número de elementos de la clase positiva y negativa, respectivamente.

Estos dos hiperplanos se conocen como márgenes; si se calcula la distancia del uno al otro, y dado que estos planos son paralelos se tendría:

$$\frac{1 - (-1)}{||\mathbf{w}||} = \frac{2}{||\mathbf{w}||}. \tag{47}$$

Con estos márgenes se puede simplificar el problema (44) a uno más simple donde se quiera maximizar la distancia que tienen los márgenes entre sí, esto además es una característica muy útil ya que permite que el modelo generalice mejor nuevos datos de esta manera, se plantea el siguiente problema de optimización:

$$\begin{aligned} \max_{\mathbf{w},b} \quad & \frac{2}{||\mathbf{w}||} \\ s.a. \quad & y_j(\mathbf{w^T x_j} + b) \geq 1, \quad j = 1, \ldots, m. \end{aligned} \tag{48}$$

Sin embargo, el inconveniente es que el problema original no es convexo, por lo que resulta necesario reformularlo. Para ello, se requiere invertir la función objetivo.

Dicha inversión conduce a la expresión $\frac{||\mathbf{w}||}{2}$, lo cual es equivalente a minimizar $\frac{||\mathbf{w}||^2}{2}$, lo que origina:

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}||\mathbf{w}||^2 \\ s.a. \quad & y_j(\mathbf{w^T x_j} + b) \geq 1, \quad j = 1, \ldots, m. \end{aligned} \tag{49}$$

Este problema es conocido como la forma primal del problema SVM de margen duro. A partir del concepto de dualidad, se puede entonces plantear la forma dual de este problema.

En primer lugar hay que plantear el Lagrangiano del problema primal:

$$L(\mathbf{w}, b, \boldsymbol{\lambda}) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{m} \lambda_i (y_i(\mathbf{w}^{\mathbf{T}}\mathbf{x_i} + b) - 1). \tag{50}$$

Sabiendo que la función objetivo dual tiene la forma (26), es necesario calcular el ínfimo del Lagrangiano. Para ello, se calcularán las derivadas parciales con respecto a $\mathbf{w}$ y con respecto a b:

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\lambda})}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{m} y_i \mathbf{x_i} \lambda_i = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{m} \lambda_i y_i \mathbf{x_i}, \tag{51}$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\lambda})}{\partial b} = \sum_{i=1}^{m} \lambda_i y_i = 0. \tag{52}$$

A continuación, se sustituyen estos resultados en el Lagrangiano del problema primal y se realiza la correspondiente simplificación para obtener la formulación dual:

$$\begin{aligned}
L(\mathbf{w}, b, \boldsymbol{\lambda}) &= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m} \lambda_i \left(y_i(\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b) - 1\right) \\
&= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m} \lambda_i y_i \mathbf{w}^{\mathbf{T}}\mathbf{x}_i - \sum_{i=1}^{m} \lambda_i y_i b + \sum_{i=1}^{m} \lambda_i \\
&= \frac{1}{2}\|\mathbf{w}\|^2 - \mathbf{w}^{\mathbf{T}} \sum_{i=1}^{m} \lambda_i y_i \mathbf{x}_i - b \sum_{i=1}^{m} \lambda_i y_i + \sum_{i=1}^{m} \lambda_i \quad (\text{se sustituye } \sum \lambda_i y_i = 0) \\
&= \frac{1}{2}\|\mathbf{w}\|^2 - \mathbf{w}^{\mathbf{T}} \sum_{i=1}^{m} \lambda_i y_i \mathbf{x}_i + \sum_{i=1}^{m} \lambda_i \quad (\text{se sustituye } \mathbf{w} = \sum \lambda_i y_i \mathbf{x}_i) \\
&= \frac{1}{2}\mathbf{w}^{\mathbf{T}}\mathbf{w} - \mathbf{w}^{\mathbf{T}}\mathbf{w} + \sum_{i=1}^{m} \lambda_i \\
&= -\frac{1}{2}\mathbf{w}^{\mathbf{T}}\mathbf{w} + \sum_{i=1}^{m} \lambda_i \quad (\text{se sustituye } \mathbf{w} = \sum \lambda_i y_i \mathbf{x}_i) \\
&= -\frac{1}{2}\left(\sum_{i=1}^{m} \lambda_i y_i \mathbf{x}_i\right)^T \left(\sum_{j=1}^{m} \lambda_j y_j \mathbf{x_j}\right) + \sum_{i=1}^{m} \lambda_i \\
&= \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i y_j \lambda_i \lambda_j \mathbf{x}_i^T \mathbf{x_j}
\end{aligned} \tag{53}$$

A partir de esta expresión se tiene el problema de optimización dual:

$$\begin{aligned} \max_{\boldsymbol{\lambda}} \quad & \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i y_j \lambda_i \lambda_j \mathbf{x}_i^T \mathbf{x}_j \\ s.a. \quad & \sum_{i=1}^{m} \lambda_i y_i = 0, \\ & \lambda_i \geq 0, \quad i = 1, \ldots, m. \end{aligned} \tag{54}$$

Tal como se deriva del modelo dual, no es posible obtener directamente $\mathbf{w}$ ni b para definir el hiperplano. En cambio, se sigue este procedimiento:

El vector de pesos $\mathbf{w}$ se calcula a partir de los multiplicadores λ_i (ver ecuación (51)):

$$\mathbf{w} \; = \; \sum_{i=1}^{m} \lambda_i \, y_i \, \mathbf{x}_i. \tag{55}$$

A continuación, b se puede obtener de cualquiera de las restricciones activas:

$$b \; = \; y_i \; - \; \mathbf{w}^{\mathbf{T}} \mathbf{x}_i, \quad i \in S, \tag{56}$$

donde $S = \{i \mid \lambda_i > 0\}$ es el conjunto de índices de los vectores soporte. Para mejorar la robustez numérica, es habitual promediar sobre todos los vectores soporte:

$$b \; = \; \frac{1}{|S|} \sum_{i \in S} \big(y_i \; - \; \mathbf{w}^{\mathsf{T}} \mathbf{x}_i \big), \tag{57}$$

donde |S| denota la cardinalidad del conjunto.S.

A continuación, en la Figura 1 se presenta un gráfico en donde se muestra el hiperplano construido por el modelo SVM hard margin.

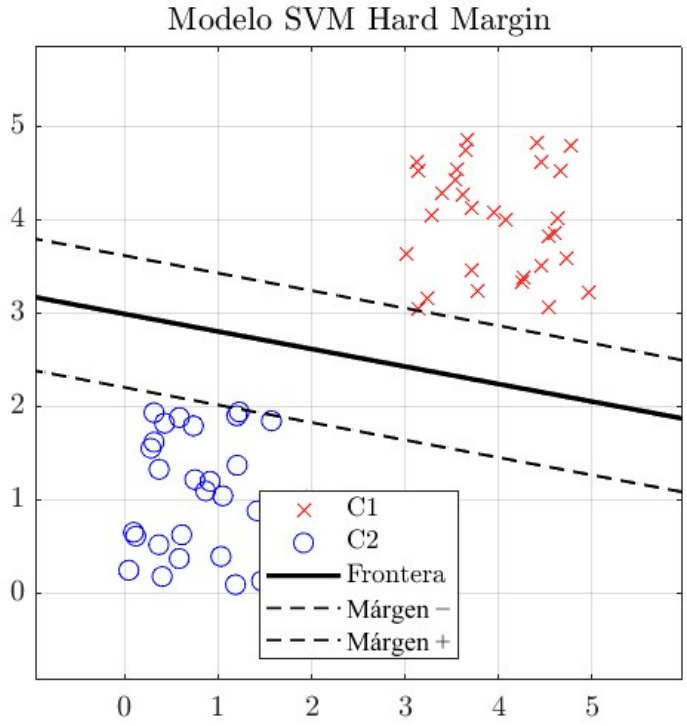

Figura 1: SVM hard margin

3.3. Modelos SVM: Soft margin

En determinadas situaciones los datos no son linealmente separables (véase la definición 14). Un ejemplo de ello se presenta en la Figura 2:

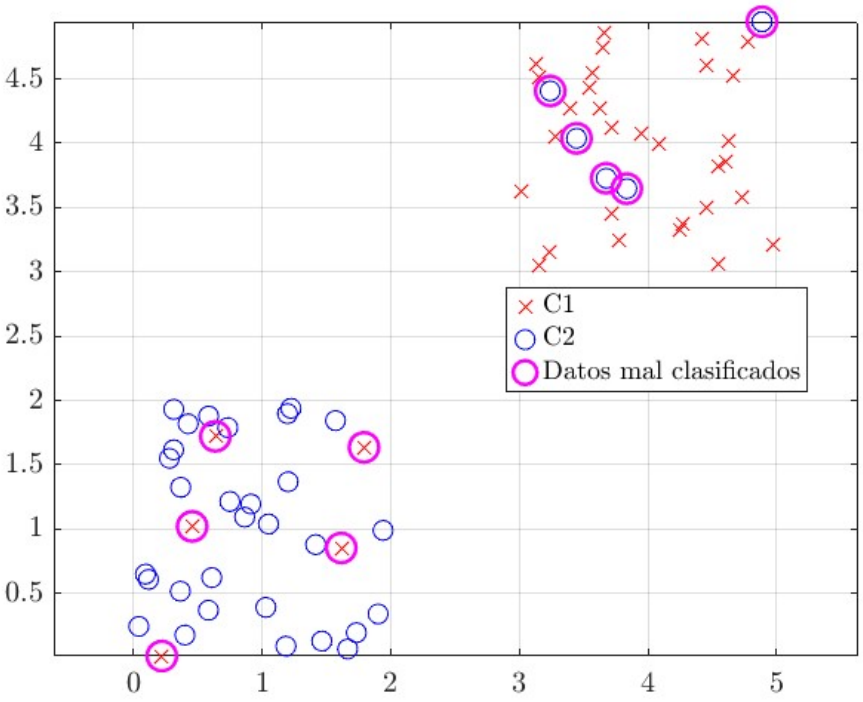

Figura 2: Puntos no separables

En este caso no existe ningún hiperplano separador capaz de distinguir $\mathcal{C}_1$ y $\mathcal{C}_2$; de intentar aplicarse el modelo de margen duro, el problema divergiría y no admitía solución.

El problema de soft margin parte de la explicación dada para el problema de hard margin (44).

En estos casos hay varias opciones: primero se puede utilizar un hiperplano que no sea lineal utilizando una función de Kernel, como se explicará en siguientes apartados, pero para casos donde los puntos son no separables como en (2) no sería una solución factible. En estos casos se suele relajar las condiciones del modelo de hard-margin.

El modelo de margen duro puede reescribirse incorporando el parámetro de relajación $C > 0$ y el parámetro ξ. Este último actúa como un término de penalización asociado a cada dato x_i, y su valor representa el grado de error de clasificación correspondiente.

Con estos dos nuevos parámetros y partiendo del modelo (44) se puede formular de la siguiente forma el modelo SVM soft margin:

$$\begin{aligned} \min_{\mathbf{w},b,\xi_i} \quad & \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{m}\xi_i \\ s.a. \quad & y_i(\mathbf{w^T x_i} + b) \geq 1 - \xi_i, \quad i = 1,\dots,m, \\ & \xi_i \geq 0, \quad i = 1,\dots,m. \end{aligned} \tag{58}$$

A partir de esta forma se puede expresar el problema dual de la misma forma que con el caso del margen duro:

Primero se plantea el Lagrangiano del problema (58):

$$L(\mathbf{w}, b, \xi, \lambda, \mu) = \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}\lambda_i[y_i(\mathbf{w^T x_i} + b) - 1 + \xi_i] - \sum_{i=1}^{m}\mu_i\xi_i. \tag{59}$$

A continuación se calcula el ínfimo del Lagrangiano, calculando las derivadas parciales de este con respecto a $\mathbf{w}$,b y ξ_i.

$$\frac{\partial L(\mathbf{w}, b, \xi_i, \boldsymbol{\lambda}, \boldsymbol{\mu})}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{m}\lambda_i y_i x_i = 0, \tag{60}$$

$$\frac{\partial L(\mathbf{w}, b, \xi_i, \boldsymbol{\lambda}, \boldsymbol{\mu})}{\partial b} = \sum_{i=1}^{m}\lambda_i y_i = 0, \tag{61}$$

$$\frac{\partial L(\mathbf{w}, b, \xi_i, \boldsymbol{\lambda}, \boldsymbol{\mu})}{\partial \xi_\mathbf{i}} = C - \lambda_i - \mu_i = 0, i = 1 \ldots, m. \tag{62}$$

A partir de estas derivadas parciales se puede llegar a la forma dual del problema de margen suave:

$$\begin{aligned}
L(\mathbf{w}, b, \xi, \lambda, \mu) &= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}\lambda_i\Big[y_i\big(\mathbf{w^T x}_i + b\big) - 1 + \xi_i\Big] - \sum_{i=1}^{m}\mu_i\xi_i \\
&= \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}\lambda_i y_i\big(\mathbf{w^T x}_i + b\big) + \sum_{i=1}^{m}\lambda_i - \sum_{i=1}^{m}\lambda_i\xi_i - \sum_{i=1}^{m}\mu_i\xi_i \\
&= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m}\lambda_i y_i\big(\mathbf{w^T x}_i + b\big) + \sum_{i=1}^{m}\lambda_i + \sum_{i=1}^{m}\xi_i\Big(C - \lambda_i - \mu_i\Big) \\
&\quad \text{(se sustituye (62))} \\
&= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m}\lambda_i y_i\,\mathbf{w^T x}_i + \Big(\sum_{i=1}^{m}\lambda_i y_i\Big)b + \sum_{i=1}^{m}\lambda_i \quad \text{(se sustituye (61))} \\
&= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m}\lambda_i y_i\,\mathbf{w^T x}_i + \sum_{i=1}^{m}\lambda_i \quad \text{(se sustituye (60))} \\
&= \frac{1}{2}\left(\sum_{i=1}^{m}\lambda_i y_i \mathbf{x}_i\right)^T\left(\sum_{j=1}^{m}\lambda_j y_j \mathbf{x}_j\right) - \sum_{i=1}^{m}\lambda_i y_i\left(\sum_{j=1}^{m}\lambda_j y_j \mathbf{x}_j\right)^T \mathbf{x}_i + \sum_{i=1}^{m}\lambda_i \\
&= \frac{1}{2}\left(\sum_{i=1}^{m}\lambda_i y_i \mathbf{x}_i\right)^T\left(\sum_{j=1}^{m}\lambda_j y_j \mathbf{x}_j\right) - \sum_{i=1}^{m}\sum_{j=1}^{m}\lambda_i\lambda_j y_i y_j\,\mathbf{x}_i^T\mathbf{x}_j + \sum_{i=1}^{m}\lambda_i \\
&= \sum_{i=1}^{m}\lambda_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} y_i y_j\,\lambda_i\lambda_j\,\mathbf{x}_i^T\mathbf{x}_j.
\end{aligned} \tag{63}$$

Así se llega a la expresión dual y como no se puede expresar con μ_i esa parte se puede eliminar y se tendría la siguiente forma para el problema dual:

$$\begin{aligned} \max_{\lambda} \quad & \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i y_j \lambda_i \lambda_j \mathbf{x}_i^T \mathbf{x}_j \\ s.a. \quad & \sum_{i=1}^{m} \lambda_i y_i = 0, \\ & 0 \leq \lambda_i \leq C, \quad i = 1, \ldots, m. \end{aligned} \tag{64}$$

Es muy similar a la versión dual del modelo hard margin (54) la única diferencia es que se le añade una condición de que tiene que ser menor o igual que C.

El vector de pesos $\mathbf{w}$ se calcula a partir de los multiplicadores λ_i (ver ecuación (60)):

$$\mathbf{w} \; = \; \sum_{i=1}^{m} \lambda_i \, y_i \, \mathbf{x_i}. \tag{65}$$

A continuación, b se puede obtener de cualquiera de las restricciones activas:

$$\begin{aligned} \mathbf{w^T x_i} + b = \frac{1 - \xi_i}{y_i} = y_i - y_i \, \xi_i, \\ \Longrightarrow \quad b = y_i - y_i \, \xi_i - \mathbf{w^T x_i}. \end{aligned} \tag{66}$$

Dado que los valores de ξ_i no se conocen, se realiza la iteración sobre los vectores soporte. Para ello, se considera el conjunto $\hat{S} = \{i \mid 0 < \lambda_i < C\}$, que representa los índices correspondientes a los vectores soporte. En este conjunto, se cumple que $\xi_i = 0$.

Para mejorar la robustez numérica, es habitual promediar sobre todos los vectores soporte:

$$b \; = \; \frac{1}{|\hat{S}|} \sum_{i \in \hat{S}} \big(y_i \; - \; \mathbf{w^T x_i} \big). \tag{67}$$

Gracias a la variable de penalización ξ_i pueden definirse múltiples variantes del modelo de margen suave. Por ejemplo, incluir un término cuadrático ξ_i^2 en la función de coste intensifica la penalización de los puntos mal clasificados (para mas detalles ver [5]).

Mediante el modelo de margen suave (64) resulta posible tratar casos con puntos no separables (véase la Figura 2), pues, aunque existan observaciones que no puedan separarse perfectamente, el clasificador no intentará forzar un hiperplano separador ideal, sino que simplemente penalizará esas violaciones y permitirá clasificaciones erróneas.

A continuación se presenta un gráfico en donde se muestra el hiperplano construido por el modelo SVM soft margin.

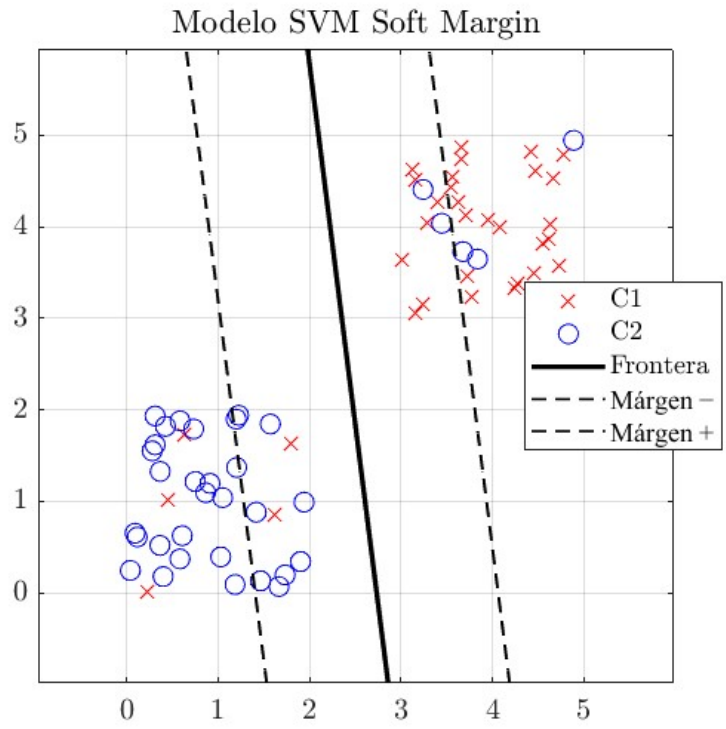

Figura 3: SVM soft margin

3.4. Modelos SVM: Kernels

A continuación, se explicarán los modelos SVM no lineales, también conocidos como modelos con función Kernel.

En muchos casos, como en la Figura 2, los datos no pueden separarse de manera lineal. En estos casos, es posible introducir un parámetro de relajación para encontrar un hiperplano separador lineal que divida las dos clases, pero esta solución no siempre es la más adecuada.

A menudo, los datos no son linealmente separables en el espacio original, y se puede lograr una mejor separación proyectando los datos a un espacio de mayor dimensión mediante una función kernel, lo que permite emplear hiperplanos de separación no lineales en el espacio transformado como el ejemplo de la Figura 4.

A continuación se presentará un ejemplo para entender mejor qué es una función de Kernel y cómo puede ser útil para solucionar problemas de este tipo.

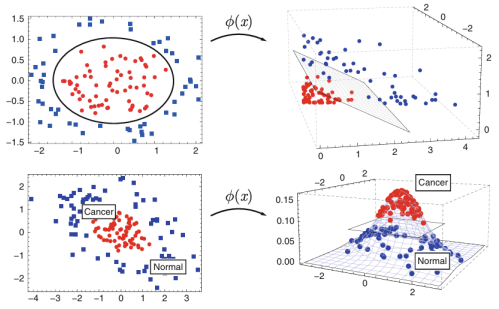

Figura 4: Ejemplos de uso de Kernels [5]

Para resolver casos como el de la Figura 4, se aplica un mapeo de los datos a un espacio de mayor dimensión mediante funciones kernel, de modo que en dicho espacio los ejemplos resulten linealmente separables por un hiperplano.

El objetivo es encontrar un hiperplano separador tal que:

$$H(\mathbf{x}) = \mathbf{w}^{\mathbf{T}}\phi(\mathbf{x}) + b, \tag{68}$$

donde $\phi : \mathbb{R}^N \to H$ un mapeo.

Para encontrar el hiperplano, se podría plantear el siguiente problema primal parecido a los problemas definidos en los apartados anteriores, (49) y (58):

$$\begin{aligned} &\min_{\mathbf{w},b} \quad \frac{1}{2}||\mathbf{w}||^2 \\ &s.a. \quad y_i(\mathbf{w}^{\mathbf{T}}\phi(\mathbf{x_i}) + b) \geq 1, \quad i = 1, \ldots, m. \end{aligned} \tag{69}$$

La principal dificultad de esta formulación es que exige conocer explícitamente la función de mapeo $\phi(\mathbf{x})$, lo cual suele resultar impracticable. Por ello, resulta habitual trabajar directamente con la forma dual del problema, ya que únicamente es necesario disponer de la función kernel correspondiente, que permite evaluar los datos directamente en el espacio $\mathbb{R}^N$.

Para derivar la forma dual de manera análoga a los casos de (54) y (64), se sigue el mismo procedimiento:

$$L(\mathbf{w}, b, \lambda) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{m} \lambda_i \left(y_i \left(\mathbf{w}^\top \phi(\mathbf{x_i}) + b \right) - 1 \right). \tag{70}$$

Se calculan las derivadas parciales respecto a $\mathbf{w}, b$:

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\lambda})}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{m} y_i \phi(\mathbf{x_i}) \lambda_i = 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{i=1}^{m} \lambda_i y_i \phi(\mathbf{x_i}), \tag{71}$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\lambda})}{\partial b} = -\sum_{i=1}^{m} \lambda_i y_i = 0. \tag{72}$$

Ahora se sustituyen estos resultados en el Lagrangiano del problema primal y se despeja para hallar el dual:

$$\begin{aligned}
L(\mathbf{w}, b, \lambda) &= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m} \lambda_i \left(y_i(\mathbf{w}^\mathbf{T}\phi(\mathbf{x_i}) + b) - 1\right) \\
&= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m} \lambda_i y_i \mathbf{w}^\mathbf{T}\phi(\mathbf{x_i}) - \sum_{i=1}^{m} \lambda_i y_i b + \sum_{i=1}^{m} \lambda_i \\
&= \frac{1}{2}\|\mathbf{w}\|^2 - \mathbf{w}^\mathbf{T} \sum_{i=1}^{m} \lambda_i y_i \phi(\mathbf{x_i}) - b\sum_{i=1}^{m} \lambda_i y_i + \sum_{i=1}^{m} \lambda_i \quad (\text{se sustituye } \sum \lambda_i y_i = 0) \\
&= \frac{1}{2}\|\mathbf{w}\|^2 - \mathbf{w}^\mathbf{T} \sum_{i=1}^{m} \lambda_i y_i \phi(\mathbf{x_i}) + \sum_{i=1}^{m} \lambda_i \quad (\text{se sustituye } \mathbf{w} = \sum \lambda_i y_i \phi(\mathbf{x_i})) \\
&= \frac{1}{2}\mathbf{w}^\mathbf{T}\mathbf{w} - \mathbf{w}^\mathbf{T}\mathbf{w} + \sum_{i=1}^{m} \lambda_i \\
&= -\frac{1}{2}\mathbf{w}^\mathbf{T}\mathbf{w} + \sum_{i=1}^{m} \lambda_i \quad (\text{se sustituye } \mathbf{w} = \sum \lambda_i y_i \phi(\mathbf{x_i})) \\
&= -\frac{1}{2}\left(\sum_{i=1}^{m} \lambda_i y_i \phi(\mathbf{x_i})\right)^T \left(\sum_{j=1}^{m} \lambda_j y_j \phi(\mathbf{x_j})\right) + \sum_{i=1}^{m} \lambda_i \\
&= \sum_{i=1}^{m} \lambda_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} y_i y_j \lambda_i \lambda_j \phi(\mathbf{x_i})^T \phi(\mathbf{x_j})
\end{aligned} \tag{73}$$

Aquí, como se puede observar en el último paso, se puede expresar como una función Kernel.

$$\sum_{i=1}^{m} \lambda_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} y_i y_j \lambda_i \lambda_j \, \Theta(\mathbf{x_i}, \mathbf{x_j}). \tag{74}$$

Ahora se puede reescribir el problema a la siguiente forma dual:

$$\begin{aligned}
\max_{\lambda} \quad & \sum_{i=1}^{m} \lambda_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} y_i y_j \lambda_i \lambda_j \Theta(\mathbf{x_i}, \mathbf{x_j}) \\
s.a. \quad & \sum_{i=1}^{m} \lambda_i y_i = 0, \\
& \lambda_i \geq 0, \quad i = 1, \ldots, m.
\end{aligned} \tag{75}$$

Las funciones kernel también permiten formular el modelo con margen suave, sustituyendo en la forma primal del modelo soft margin (58) el vector $\mathbf{x}$ por su correspondiente imagen mediante la función de mapeo $\phi(\mathbf{x})$:

$$\begin{aligned}
\min_{\mathbf{w}, b, \xi_i} \quad & \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{m} \xi_i \\
s.a. \quad & y_i(\mathbf{w}^\mathbf{T}\phi(\mathbf{x_i}) + b) \geq 1 - \xi_i, \quad i = 1, \ldots, m, \\
& \xi_i \geq 0, \quad i = 1, \ldots, m.
\end{aligned} \tag{76}$$

En este caso se calcularía $\mathbf{w}$ y b de una manera muy similar a como se calcula en (54), por este motivo, no se desarrollará el proceso para este modelo.

Para obtener la formulación dual parte del mismo razonamiento ya empleado en el modelo de soft margin lineal (64), y guarda una estructura similar a la del modelo hard margin con funciones kernel (75), es por esto que no se desarrollara el proceso para este modelo (76).

Se obtiene la siguiente formulación para el problema dual con Kernel:

$$\begin{aligned} \max_{\boldsymbol{\lambda}} \quad & \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i y_j \lambda_i \lambda_j \Theta(\mathbf{x_i}, \mathbf{x_j}) \\ s.a. \quad & \sum_{i=1}^{m} \lambda_i y_i = 0, \\ & 0 \leq \lambda_i \leq C, \quad i = 1, \ldots, m. \end{aligned} \tag{77}$$

En este caso se calcularía $\mathbf{w}$ y b de una manera muy similar a como se calcula en (64), por este motivo, no se desarrollará el proceso para este modelo.

Una vez establecidas las formulaciones duales anteriores, se describirán algunas de las funciones kernel más empleadas:

Definición 16 (Kernel polinómico)**.** [4] Dado $d \in \mathbb{N}$, se define el kernel polinómico de grado d como:

$$\Theta(\mathbf{x}, \mathbf{y}) = (\mathbf{x^T y} + 1)^d. \tag{78}$$

Definición 17 (Kernel de base radial gaussiana)**.** [4] Dado $\sigma \in \mathbb{N}$, se define el kernel de base radial gaussiana como:

$$\Theta(\mathbf{x}, \mathbf{y}) = \exp(-||\mathbf{x} - \mathbf{y}||^2 / 2\sigma^2). \tag{79}$$

Definición 18 (Kernel gaussiano)**.** [4] Dados $\sigma, p \in \mathbb{N}$, se define el kernel gaussiano como:

$$\Theta(\mathbf{x}, \mathbf{y}) = \frac{1}{(2\pi)^{\frac{p}{2}} \sigma} \exp\left\{ -\frac{1}{2\sigma^2} \|\mathbf{x} - \mathbf{y}\|^2 \right\}. \tag{80}$$

4. Variantes de modelos SVM y funciones de perdidas Hinge y Pinball

4.1. Modelos SVM: PSVM

Dentro de las distintas variantes existentes del modelo SVM, se encuentra la Probability Support Vector Machine (PSVM) [6], [7], la cual se abordará en esta sección.

Este modelo tiene por objetivo estimar la probabilidad

$$\Pr(y = +1 \mid \mathbf{x}),$$

donde $y \in \{-1, +1\}$ es la etiqueta de clase y $\mathbf{x} \in \mathbb{R}^N$. En otras palabras, se trata de inferir la probabilidad de que un punto $\mathbf{x}$ pertenezca a una u otra clase.

A diferencia de los modelos descritos en (49), (54), (58), (64), (69) y (76), que asignan directamente una etiqueta de clase, este enfoque estima la probabilidad de pertenencia a la clase:

$$\Pr\big(y = +1 \mid \mathbf{x}\big).$$

Este enfoque resulta especialmente útil para la toma de decisiones en contextos como los modelos de "churn" (que predicen la probabilidad de que un cliente cambie de compañía), pues la probabilidad estimada permite a las empresas diseñar intervenciones específicas para clientes con alto riesgo de abandono, optimizando la asignación de recursos y mejorando las estrategias de retención [6].

Esta sería la forma del modelo PSVM:

$$\begin{aligned} \min_{\mathbf{w},b,\xi} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{\varepsilon}\sum_{i=1}^{m}\xi_i \\ \text{s.a.} \quad & y_i\left(\mathbf{w}^{\mathbf{T}}\phi(\mathbf{x}_i) + b - 0.5\right) \geq 0.5\varepsilon - \xi_i, \quad i = 1, \ldots, m, \\ & 0 \leq \mathbf{w}^{\mathbf{T}}\phi(\mathbf{x}_i) + b \leq 1, \quad \xi_i \geq 0, \quad i = 1, \ldots, m. \end{aligned} \tag{81}$$

Este modelo se fundamenta en la versión de margen suave presentada en (58), con el mapeo ϕ, que incorpora las penalizaciones ξ_i y el coeficiente de regularización $C > 0$, a las que se añade un nuevo parámetro $\varepsilon > 0$ para ajustar la calibración probabilística.

La forma dual de este problema es la siguiente [6]:

$$\begin{aligned} \max_{\lambda,\mu,\alpha} \quad & \sum_{i=1}^{m}\left(0.5\lambda_i(y_i + \varepsilon) - \alpha_i\right) - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\lambda_i y_i + \mu_i - \alpha_i)(\lambda_j y_j + \mu_j - \alpha_j)\,\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j) \\ \text{s.a.} \quad & \sum_{i=1}^{m}(\lambda_i y_i + \mu_i - \alpha_i) = 0, \\ & 0 \leq \lambda_i \leq \frac{C}{\varepsilon}, \quad \mu_i, \alpha_i \geq 0, \quad i = 1, \ldots, m. \end{aligned} \tag{82}$$

Esta forma se puede reescribir con la función Kernel correspondiente sustituyendo $\phi(\mathbf{x_i})^T\phi(\mathbf{x_j}) = \Theta(\mathbf{x_i}, \mathbf{x_j})$.

Este modelo maximiza el margen asegurando que, con una pequeña tolerancia al error, las observaciones de la clase positiva se sitúen en el semiespacio.

$$\{\, \mathbf{x} \in \mathbb{R}^N : \mathbf{w}^{\mathbf{T}}\mathbf{x} + b > 0.5\},$$

y las de la clase negativa en su semiespacio complementario. De este modo, las salidas quedan acotadas en el intervalo $[0, 1]$ y pueden interpretarse probabilísticamente.

A continuación, para determinar el hiperplano separador que resulta de resolver el problema, es decir, obtener $\mathbf{w}$ y b asociado a la solución, se seguirá el siguiente procedimiento:

- Al resolver el problema con $\lambda, \mu, \alpha \neq 0$ y $\varepsilon \in (0, 1]$.
- Para hallar $\mathbf{w}$ asociada a la forma primal de los problemas (81) y (82):

$$\mathbf{w} = \sum_{i=1}^{m} \big(\lambda_i y_i + \mu_i - \alpha_i\big)\, \phi(\mathbf{x}_i).$$

- Para hallar b se pueden dar los siguientes casos:
  - Si existe un λ_i tal que $\lambda_i \in \big(0, \frac{C}{\varepsilon}\big)$, entonces

$$b = 0.5\, \varepsilon\, y_i + 0.5 - \mathbf{w}^\top \phi(\mathbf{x}_i).$$

    Para ese i, si $\xi_i = 0$ y $\varepsilon \in (0, 1)$, se tiene $\mu_i = \alpha_i = 0$.
  - Si existe un $\mu_i > 0$, entonces

$$b = -\,\mathbf{w}^\top \phi(\mathbf{x}_i), \quad \alpha_i = 0.$$

    Si además $y_i = -1$, entonces $\xi_i = 0$.
  - Si existe un $\alpha_i > 0$, entonces

$$b = 1 - \mathbf{w}^\top \phi(\mathbf{x}_i), \quad \mu_i = 0.$$

    Si además $y_i = 1$, entonces $\xi_i = 0$.

Al igual que en SVM, es preferible calcular el término independiente b como el promedio sobre todos los vectores de soporte (aquellos con $\xi_i = 0$):

$$b \;=\; \frac{1}{|S|} \sum_{i \in S} \Big(0.5\, \varepsilon\, y_i + 0.5 - \mathbf{w}^\top \phi(\mathbf{x}_i)\Big), \tag{83}$$

donde

$$S \;=\; \big\{\, i \in \{1, \ldots, m\} \mid 0 < \lambda_i < \tfrac{C}{\varepsilon} \big\},$$

y $|S|$ indica el número de elementos de S.

A continuación, se incluirá un gráfico en el que se representa el modelo PSVM utilizando los distintos kernels disponibles: lineal, gaussiano, de base radial y polinómico de grado 2:

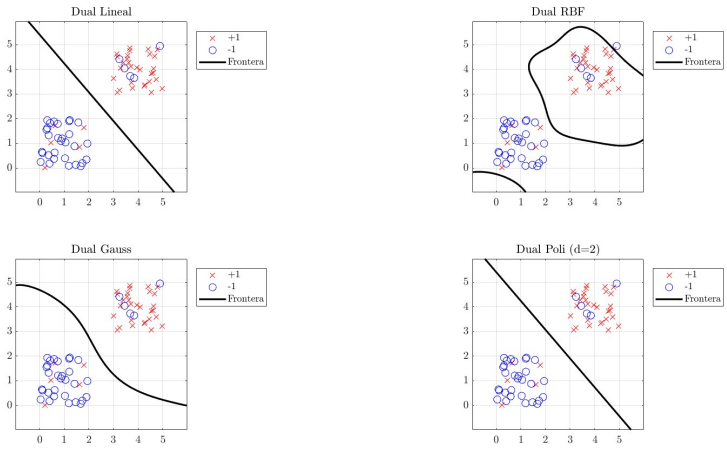

Figura 5: Modelo PSVM

4.2. Funciones de pérdidas: Hinge y Pinball

En esta sección se abordará el concepto de función de pérdida, presentando inicialmente la función de pérdida empleada en los modelos descritos hasta el momento.

Posteriormente, se introducirá la nueva función de pérdida que será objeto de estudio en este trabajo, detallando el procedimiento que permite extender un modelo SVM tradicional a un modelo PSVM basado en dicha función.

Definición 19 (Función de pérdida)**.** Sea $L\colon \mathcal{Y}\times\mathcal{A}\to[0,\infty)$ la función que, dado un valor objetivo $y\in\mathcal{Y}$ y una predicción $\hat{y}\in\mathcal{A}$, mide su discrepancia. Para un modelo parametrizado $f(x;\theta)$, definimos

$$R(\theta) \;=\; \mathbb{E}_{(X,Y)\sim P}\left[L\left(Y,\; f(X;\theta)\right)\right],$$

que llamamos *riesgo teórico*, y su estimación sobre la muestra $\{(\mathbf{x_i}, y_i)\}_{i=1}^{m}$ es

$$\widehat{R}(\theta) \;=\; \frac{1}{m}\sum_{i=1}^{m} L\left(y_i,\; f(\mathbf{x_i};\theta)\right).$$

La función de pérdidas utilizada en todos los modelos que se han explicado hasta ahora es la Hinge loss, denotada por L_{hinge}. Esta función de pérdidas mide la menor distancia entre clases, $y\in\{+1,-1\}$ y se define como:

Definición 20 (Hinge loss)**.** Para $u\in\mathbb{R}$, la función de pérdida hinge viene dada por

$$L_{\text{hinge}}(u) = \text{máx}\{0,\, u\}, \tag{84}$$

de modo que penaliza únicamente los márgenes negativos o nulos [8].

Al adoptar la función de pérdida Hinge, la formulación del modelo SVM se simplifica (en su forma primal sin incluir restricciones explícitas) hasta

quedar como:

$$\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{m} L_{hinge}(1 - y_i(\mathbf{w^T x_i} + b)). \tag{85}$$

Este modelo, en el caso límite en que $C \to \infty$, se reduce al modelo clásico de margen duro (hard margin). Por otro lado, si C se mantiene finito, el modelo se transforma en el conocido modelo de margen suave (soft margin).

No obstante, esta formulación presenta una desventaja importante: al estar basada en la distancia mínima entre las clases, resulta sensible al ruido, es decir, a la presencia de observaciones mal clasificadas de manera aleatoria. Dicho ruido puede originarse tanto en las características de los datos $\mathbf{x}$ como en las etiquetas de clase $y \in \{+1, -1\}$.

Diversas soluciones a este problema han sido propuestas [8]. Sin embargo, en este artículo se investigó una nueva propuesta [8]: modificar la función de pérdida, sustituyendo la tradicional función L_{hinge} por una función de pérdida tipo pinball, denotada como L_τ.

La función de pérdida tipo pinball no calcula la distancia entre conjuntos basándose en los puntos más próximos de clases opuestas, enfoque que, como se ha mencionado anteriormente, resulta sensible al ruido. En su lugar, la función pinball emplea una medida de distancia cuartílica, la cual presenta una mayor robustez frente a la presencia de ruido en los datos.

Otra aportación novedosa del [8] consiste en la aplicación de la función de pérdida pinball a problemas de clasificación. Tradicionalmente, esta función ha sido utilizada en el ámbito de la regresión, por lo que su uso en clasificación representa una contribución relevante en el campo.

Definición 21 (Función de pérdida pinball)**.** Para $u \in \mathbb{R}$ la función de pérdida pinball se define como:

$$L_\tau(u) = \begin{cases} u, & \text{si } u \geq 0, \\ -\tau u, & \text{si } u < 0, \end{cases} \tag{86}$$

donde $\tau \in (0, 1)$ es un parámetro que controla el grado de penalización de los errores.

Al igual que en el caso del modelo basado en la función de pérdida Hinge, el objetivo consiste en minimizar una medida de dispersión, en este caso la *distancia cuartílica*. Formalmente, se busca minimizar:

$$t_{C_1}^{\frac{\tau}{1+\tau}} = \arg\min_{t\in\mathbb{R}} \sum_{i\in C_1} L_\tau\big(t - y_i(\mathbf{w}^\top \mathbf{x_i} + b)\big), \tag{87}$$

$$t_{C_2}^{\frac{\tau}{1+\tau}} = \arg\min_{t\in\mathbb{R}} \sum_{i\in C_2} L_\tau\big(t - y_i(\mathbf{w}^\top \mathbf{x_i} + b)\big), \tag{88}$$

donde C_1 y C_2 denotan los subconjuntos de datos correspondientes a cada clase.

A partir de la ecuación (87), fijando $\tau = \frac{q}{1-q}$ para un cierto $q \in (0,1)$, se impone que, para todo $t \in \mathbb{R}$,

$$\sum_i L_\tau\big(t - y_i(\mathbf{w}^\top \mathbf{x_i} + b)\big) \;\geq\; \sum_i L_\tau\big(1 - y_i(\mathbf{w}^\top \mathbf{x_i} + b)\big).$$

Minimizando el segundo sumatorio y añadiendo un término de regularización basado en la norma de $\mathbf{w}$, se obtiene el modelo SVM con función de pérdida pinball:

$$\min_{\mathbf{w},b} \; \frac{1}{2}\|\mathbf{w}\|^2 \;+\; C\sum_{i=1}^{m} L_\tau\big(1 - y_i(\mathbf{w}^\top \mathbf{x_i} + b)\big), \tag{89}$$

el cual resulta análogo a la formulación del SVM clásico basado en la función de pérdida hinge (85).

A partir de la expresión (89), se puede derivar, de manera similar al caso del modelo hinge, la siguiente formulación primal del modelo SVM:

$$\begin{aligned} \min_{\mathbf{w},b,\xi_i} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}\xi_i \\ \text{s.a.} \quad & y_i(\mathbf{w}^\top \mathbf{x_i} + b) \geq 1 - \xi_i, \quad i = 1,\dots,m, \\ & y_i(\mathbf{w}^\top \mathbf{x_i} + b) \leq 1 + \frac{1}{\tau}\xi_i, \quad i = 1,\dots,m. \end{aligned} \tag{90}$$

A partir de esta formulación, se puede construir el modelo dual. Primero se plantea el Lagrangiano asociado al problema (90):

$$\begin{aligned} L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu}) =& \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}\xi_i \\ & - \sum_{i=1}^{m}\lambda_i\left[y_i(\mathbf{w}^\top \mathbf{x_i} + b) - 1 + \xi_i\right] \\ & + \sum_{i=1}^{m}\mu_i\left[y_i(\mathbf{w}^\top \mathbf{x_i} + b) - 1 - \frac{1}{\tau}\xi_i\right]. \end{aligned} \tag{91}$$

A continuación, se calcula el ínfimo del Lagrangiano imponiendo las condiciones de estacionaridad, es decir, anulando las derivadas parciales respecto a $\mathbf{w}$, b y ξ_i:

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu})}{\partial \mathbf{w}} = 0 \quad \Rightarrow \quad \mathbf{w} = \sum_{i=1}^{m}(\lambda_i - \mu_i)y_i\mathbf{x}_i, \tag{92}$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu})}{\partial b} = 0 \quad \Rightarrow \quad \sum_{i=1}^{m}(\lambda_i - \mu_i)y_i = 0, \tag{93}$$

$$\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu})}{\partial \xi_i} = 0 \quad \Rightarrow \quad C - \lambda_i - \frac{1}{\tau}\mu_i = 0, \quad i = 1,\dots,m. \tag{94}$$

A partir de estas condiciones se deriva la forma dual del problema de margen suave. Tras el desarrollo correspondiente, se obtiene la siguiente formulación dual:

$$\begin{aligned} \underset{\boldsymbol{\lambda},\boldsymbol{\mu}}{\text{máx}} \quad & \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i y_i \mathbf{x_i^T} \mathbf{x_j} y_j \lambda_j, \\ \text{s.a.} \quad & \sum_{i=1}^{m} \lambda_i y_i = 0, \\ & \lambda_i + \left(1 + \frac{1}{\tau}\right) \mu_i = C, \quad i = 1, \ldots, m \\ & \lambda_i + \mu_i \geq 0, \quad \mu_i \geq 0, \quad i = 1, \ldots, m. \end{aligned} \tag{95}$$

La forma dual (95) puede generalizarse para ser usada con funciones kernels. Para ello, se introduce una función de mapeo no lineal $\phi : \mathbb{R}^N \to H$, sabiendo que $\phi(\mathbf{x_1})^T \phi(\mathbf{x_2}) = \Theta(\mathbf{x_1}, \mathbf{x_2})$. Así, se obtiene la formulación dual del SVM con pérdida pinball en su versión con kernel:

$$\begin{aligned} \underset{\boldsymbol{\lambda},\boldsymbol{\mu}}{\text{máx}} \quad & \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i y_i \Theta(\mathbf{x}_i, \mathbf{x}_j) y_j \lambda_j, \\ \text{s.a.} \quad & \sum_{i=1}^{m} \lambda_i y_i = 0, \\ & \lambda_i + \left(1 + \frac{1}{\tau}\right) \mu_i = C, \quad i = 1, \ldots, m, \\ & \lambda_i + \mu_i \geq 0, \quad \mu_i \geq 0, \quad i = 1, \ldots, m. \end{aligned} \tag{96}$$

La forma dual (95) se puede simplificar de la siguiente manera:

Primero la siguiente restricción se puede reescribir de la siguiente manera:

$$\begin{aligned} & \lambda_i + \left(1 + \frac{1}{\tau}\right) \mu_i = C \to \mu_i = \left(\frac{\tau}{1+\tau}\right) (C - \lambda_i) \\ & \xrightarrow{\mu_i \geq 0} \left(\frac{\tau}{1+\tau}\right) (C - \lambda_i) > 0 \xrightarrow{\frac{\tau}{1+\tau} > 0} C - \lambda_i \geq 0 \to \lambda_i \leq C. \end{aligned} \tag{97}$$

Ahora, usando la expresión para μ_i en la primera restricción de desigualdad se tiene lo siguiente:

$$\begin{aligned} & \lambda_i + \mu_i \geq 0 \to \lambda_i + (\frac{\tau}{1+\tau})(C - \lambda_i) \geq 0 \\ & \xrightarrow{\times(1+\tau)} \lambda_i + \tau\lambda_i + \tau C - \tau\lambda_i \geq 0 \to \lambda_i + \tau C \geq 0 \to \lambda_i \geq -\tau C. \end{aligned} \tag{98}$$

Entonces se deduce que $-\tau C \leq \lambda_i \leq C$ para $i = 1, \ldots, m$. Por lo tanto, podemos reescribir (96) como:

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{máx}} \quad & \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i y_i \Theta(\mathbf{x}_i, \mathbf{x}_j) y_j \lambda_j, \\ \text{s.a.} \quad & \sum_{i=1}^{m} \lambda_i y_i = 0, \\ & -\tau C \leq \lambda_i \leq C, \quad i = 1, \ldots, m. \end{aligned} \tag{99}$$

Para calcular los valores $\mathbf{w}$ y b se seguirán lo siguientes pasos:

Primero para calcular $\mathbf{w}$ se usará la ecuación (92) es decir:

$$\mathbf{w} = \sum_{i=1}^{m} (\lambda_i - \mu_i) y_i \mathbf{x}_i. \tag{100}$$

En el caso de utilizar la formulación (99) simplemente se sustituye $\mu_i = (\frac{\tau}{1+\tau})(C - \lambda_i), i = 1, \ldots, m$.

Segundo, para calcular b usaremos las siguientes dos condiciones de complementariedad:

$$\lambda_i \left[y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 + \xi_i \right] = 0, \quad i = 1, \ldots, m, \tag{101}$$

$$\mu_i \left[y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 - \tfrac{1}{\tau}\xi_i \right] = 0, \quad i = 1, \ldots, m. \tag{102}$$

Considerando los vectores que cumplan las condiciones $\lambda_i > 0$ y $\mu_i > 0$ se tiene:

$$y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 + \xi_i = 0, \quad i = 1, \ldots, m, \tag{103}$$

$$y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 - \tfrac{1}{\tau}\xi_i = 0 \quad i = 1, \ldots, m, \tag{104}$$

A continuación se tiene que multiplicando la segunda igualdad (104) por τ y luego sumándola a la primera (103) nos da:

$$(1 + \tau) y_i(\mathbf{w_i^T x_i} + b) - 1 - \tau = 0 \rightarrow b = y_i - \mathbf{w_i^T x_i} \tag{105}$$

Luego para hallar b se tiene que iterar sobre los vectores que cumplan las condiciones $\lambda_i > 0$, entendiendo que $\lambda_i = \alpha_i + \mu_i$, y $\mu_i > 0$ y calcular el promedio.

Otra característica importante es que si en este modelo $\tau = 0$ el modelo tiende a la versión con función de perdidas L_{hinge}, es decir, el modelo SVM de margen suave.

A continuación, se incluirá un gráfico en el que se representa el modelo SVM pinball utilizando los distintos kernels disponibles: lineal, gaussiano, de base radial y polinómico de grado 2:

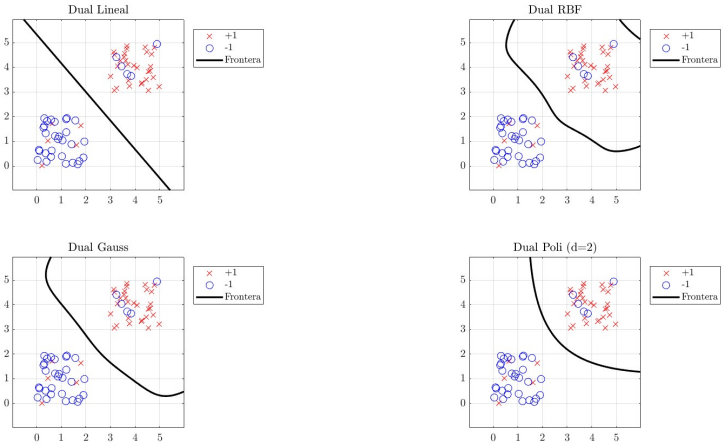

Figura 6: Modelo SVM pinball

4.3. PSVM con función de perdidas Pinball

A partir de la versión (95), se calculará el modelo PSVM con función de pérdida pinball, según lo explicado en apartados anteriores. El modelo resultante es el siguiente:

$$\begin{aligned} \min_{\mathbf{w},b,\xi_i} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{\varepsilon}\sum_{i=1}^{m}\xi_i \\ \text{s.a.} \quad & y_i\big(\mathbf{w^T x_i} + b - 0.5\big) \;\geq\; 0.5\,\varepsilon - \xi_i, \quad i = 1,\ldots,m, \\ & y_i\big(\mathbf{w^T x_i} + b - 0.5\big) \;\leq\; 0.5\,\varepsilon + \tfrac{1}{\tau}\,\xi_i, \quad i = 1,\ldots,m, \\ & 0 \;\leq\; \mathbf{w^T x_i} + b \;\leq\; 1, \quad i = 1,\ldots,m. \end{aligned} \tag{106}$$

Esta formulación conserva los elementos esenciales del PSVM original, garantiza que la función de decisión $f(\mathbf{x})$ tome valores en $[0,1]$ mediante la restricción

$$0 \;\leq\; \mathbf{w}^\top\mathbf{x} + b \;\leq\; 1,$$

y mantiene el control del ancho de margen a través del parámetro ε, además de que $C > 0$ y $\varepsilon > 0$.

A continuación se explicará como se calcula la forma dual del modelo:

Primero se plantea el Lagrangiano del problema (106):

$$\begin{aligned} L(\mathbf{w},b,\boldsymbol{\xi},\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{\gamma}) =& \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{\varepsilon}\sum_{i=1}^{C}\xi_i \\ & - \sum_{i=1}^{C}\lambda_i\left[y_i(\mathbf{w}^\top\mathbf{x_i} + b - 0.5) - 0.5\varepsilon + \xi_i\right] \\ & + \sum_{i=1}^{C}\mu_i\left[y_i(\mathbf{w}^\top\mathbf{x_i} + b - 0.5) - 0.5\varepsilon - \frac{1}{\tau}\xi_i\right] \\ & - \sum_{i=1}^{C}\beta_i\left(\mathbf{w}^\top\mathbf{x_i} + b\right) \\ & + \sum_{i=1}^{C}\gamma_i\left(\mathbf{w}^\top\mathbf{x_i} + b - 1\right). \end{aligned} \tag{107}$$

A continuación se calcula el ínfimo del Lagrangiano, calculando las derivadas parciales de este con respecto a $\mathbf{w}$,b y ξ_i.

$$\begin{aligned} \frac{\partial L(\mathbf{w},b,\boldsymbol{\xi},\boldsymbol{\lambda},\boldsymbol{\mu},\boldsymbol{\beta},\boldsymbol{\gamma})}{\partial\mathbf{w}} =& \mathbf{w} - \sum_{i=1}^{m}\lambda_i y_i\mathbf{x_i} + \sum_{i=1}^{m}\mu_i y_i\mathbf{x_i} - \sum_{i=1}^{m}\beta_i\mathbf{x_i} + \sum_{i=1}^{m}\gamma_i\mathbf{x_i} = 0, \\ \Rightarrow\; & \mathbf{w} = \sum_{i=1}^{m}((\lambda_i - \mu_i)y_i + \beta_i - \gamma_i)\mathbf{x_i}, \end{aligned} \tag{108}$$

$$\begin{aligned}\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\gamma})}{\partial b} &= -\sum_{i=1}^{m} \lambda_i y_i + \sum_{i=1}^{m} \mu_i y_i - \sum_{i=1}^{m} \beta_i + \sum_{i=1}^{m} \gamma_i = 0, \\ &\Rightarrow \sum_{i=1}^{m} [(\lambda_i - \mu_i) y_i + \beta_i - \gamma_i)] = 0.\end{aligned} \tag{109}$$

$$\begin{aligned}\frac{\partial L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\gamma})}{\partial \xi} &= \frac{C}{\varepsilon} - \lambda_i - \frac{\mu_i}{\tau} = 0, \\ &\Rightarrow \lambda_i = \frac{C}{\varepsilon} - \frac{\mu_i}{\tau}, \quad i = 1, \ldots, m.\end{aligned} \tag{110}$$

Con estas restricciones se puede reescribir el Lagrangiano como:

$$\begin{aligned}L(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\beta}, \boldsymbol{\gamma}) =& \tfrac{1}{2}\|\mathbf{w}\|^2 - \mathbf{w}^\top \sum_{i=1}^{m} ((\lambda_i - \mu_i) y_i + \beta_i - \gamma_i)\, \mathbf{x_i}) \quad \text{Sustituyes por (108)} \\ &+ \sum_{i=1}^{m} \xi_i \Big(\tfrac{C}{\varepsilon} - \lambda_i - \tfrac{1}{\tau}\mu_i \Big) \quad \text{Sustituyes por (110)} \\ &- b \sum_{i=1}^{m} ((\lambda_i - \mu_i) y_i + \beta_i - \gamma_i) \quad \text{Sustituyes por (109)} \\ &+ \sum_{i=1}^{m} \big(0.5(\lambda_i - \mu_i) y_i + 0.5\, \varepsilon (\lambda_i - \mu_i) y_i - \gamma_i \big) \\ &= \tfrac{1}{2}\|\mathbf{w}\|^2 - \mathbf{w^T w} + \sum_{i=1}^{m} \big(0.5(\lambda_i - \mu_i) y_i + 0.5\, \varepsilon (\lambda_i - \mu_i) y_i - \gamma_i \big) \\ &= -\tfrac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{m} \big(0.5(\lambda_i - \mu_i) y_i + 0.5\, \varepsilon (\lambda_i - \mu_i) y_i - \gamma_i \big)\end{aligned} \tag{111}$$

Ahora sustituyendo $\mathbf{w}$ con (108) y a partir de (111) se llegaría a la siguiente forma dual:

$$\begin{aligned}&\max_{\boldsymbol{\lambda}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\mu}} \sum_{i=1}^{m} \Big(\tfrac{1}{2}(\lambda_i - \mu_i)(y_i + \varepsilon) - \gamma_i \Big) - \\ &\quad \tfrac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} ((\lambda_i - \mu_i) y_i + \beta_i - \gamma_i)\, ((\lambda_i - \mu_i) y_j + \beta_j - \gamma_j)\, \mathbf{x_i}^\top \mathbf{x}_j) \\ &\text{s.a.} \quad \sum_{i=1}^{m} ((\lambda_i - \mu_i) y_i + \beta_i - \gamma_i) = 0, \\ &\quad \lambda_i = \frac{C}{\varepsilon} - \frac{\mu_i}{\tau}, \quad i = 1, \ldots, m, \\ &\quad \lambda_i, \mu_i,\ \beta_i,\ \gamma_i \geq 0, \quad i = 1, \ldots, m,\end{aligned} \tag{112}$$

Si se crea una variable $\alpha_i = \lambda_i - \mu_i$, para $i = 1, \ldots, m$ con el propósito de

simplificar la forma dual, se tiene el siguiente modelo:

$$\begin{aligned}
&\max_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma},\boldsymbol{\mu}} \sum_{i=1}^{m}\Big(\tfrac{1}{2}\alpha_i(y_i+\varepsilon)-\gamma_i\Big) - \tfrac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_i y_i+\beta_i-\gamma_i)\,(\alpha_j y_j+\beta_j-\gamma_j)\,\mathbf{x_i}^{\top}\mathbf{x}_j \\
&\text{s.a.} \quad \sum_{i=1}^{m}(\alpha_i y_i+\beta_i-\gamma_i)=0, \\
&\quad \alpha_i+\Big(1+\tfrac{1}{\tau}\Big)\mu_i=\tfrac{C}{\varepsilon}, \quad i=1,\ldots,m, \\
&\quad \mu_i,\ \beta_i,\ \gamma_i \geq 0, \quad i=1,\ldots,m, \quad \alpha_i+\mu_i\geq 0, \quad i=1,\ldots,m.
\end{aligned} \tag{113}$$

Ahora con la misma idea que con el modelo SVM pinball (99), se buscará hacer una simplificación similar para el modelo PSVM pinball. Como antes se sustituye μ_i:

$$\begin{aligned}
&\alpha_i+\Big(1+\tfrac{1}{\tau}\Big)\mu_i=\tfrac{C}{\varepsilon}\rightarrow \mu_i=(\frac{\tau}{1+\tau})(\frac{C}{\varepsilon}-\alpha_i) \xrightarrow{\mu_i\geq 0} \\
&(\frac{\tau}{1+\tau})(\frac{C}{\varepsilon}-\alpha_i)\geq 0 \xrightarrow{\frac{\tau}{1+\tau}>0} \frac{C}{\varepsilon}-\alpha_i\geq 0\rightarrow \alpha_i\leq\frac{C}{\varepsilon}.
\end{aligned} \tag{114}$$

Luego a partir de la siguiente restricción se puede simplificar a la siguiente forma:

$$\begin{aligned}
&\alpha_i+\mu_i\geq 0\rightarrow \alpha_i+(\frac{\tau}{1+\tau})(\frac{C}{\varepsilon}-\alpha_i)\geq 0 \xrightarrow{\times(1+\tau)} \\
&\alpha_i+\tau\alpha_i+\tau\frac{C}{\varepsilon}-\tau\alpha_i\geq 0\rightarrow \alpha_i+\frac{\tau C}{\varepsilon}\geq 0\rightarrow \alpha_i\geq -\frac{\tau C}{\varepsilon}.
\end{aligned} \tag{115}$$

Con estas simplificaciones se puede obtener la versión simplificada del PSVM dual pinball:

$$\begin{aligned}
&\max_{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}} \sum_{i=1}^{m}\big(\tfrac{1}{2}\alpha_i(y_i+\varepsilon)-\gamma_i\big) - \tfrac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_i y_i+\beta_i-\gamma_i)(\alpha_j y_j+\beta_j-\gamma_j)\,\mathbf{x}_i^{\top}\mathbf{x}_j \\
&\text{s.a.} \quad \sum_{i=1}^{m}(\alpha_i y_i+\beta_i-\gamma_i)=0, \\
&\quad -\frac{\tau C}{\varepsilon}\leq\alpha_i\leq\frac{C}{\varepsilon}, \quad i=1,\ldots,m, \\
&\quad \beta_i,\gamma_i\geq 0, \quad i=1,\ldots,m.
\end{aligned} \tag{116}$$

Finalmente, al introducir el mapeo

$$\phi\colon \mathbb{R}^N \longrightarrow H \tag{117}$$

y emplear la identidad fundamental de los métodos kernel,

$$\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j) = \Theta(\mathbf{x}_i,\mathbf{x}_j), \tag{118}$$

las formas duales (112), (113) y (116) se obtiene sus correspondientes formas con kernel.

Para determinar el hiperplano separador asociado a la resolución de cualquiera de los problemas (113) o (116) es decir, para recuperar los parámetros $\mathbf{w}$ y b de la solución óptima, procederemos de la siguiente manera:

- Al resolver el problema con $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\mu} \neq 0$ y $\varepsilon \in (0, 1]$.
- Para hallar $\mathbf{w}$ se utiliza (108) con α_i:

$$\mathbf{w} = \sum_{i=1}^{m} (\alpha_i y_i + \beta_i - \gamma_i)\mathbf{x_i}.$$

- Se busca una frontera de clasificación $f : \mathbb{R}^N \to \{+1, -1\}$, para hallar la frontera tendra la siguiente forma:

$$f(\mathbf{x}) = sign(\mathbf{w^T x} + b - 0.5) = sign(\sum_{i=1}^{m} (\alpha_i y_i + \beta_i - \gamma_i)\mathbf{x_i^T x} + b - 0.5) \tag{119}$$

  Esta expresión se puede expresar como forma con kernel de la siguiente manera:

$$f(\mathbf{x}) = sign(\sum_{i=1}^{m} (\alpha_i y_i + \beta_i - \gamma_i)\Theta(\mathbf{x_i}, \mathbf{x}) + b - 0.5) \tag{120}$$

- Para hallar el b sabiendo que hay dos condiciones de complementariedad:

$$\lambda_i \big[y_i(\mathbf{w}^\top \mathbf{x}_i + b - 0.5) - 0.5\varepsilon + \xi_i \big] = 0, \quad i = 1, \ldots, m, \tag{121}$$

$$\mu_i \big[y_i(\mathbf{w}^\top \mathbf{x}_i + b - 0.5) - 0.5\varepsilon - \tfrac{1}{\tau}\xi_i \big] = 0, \quad i = 1, \ldots, m, \tag{122}$$

  Considerando los vectores que cumplan las condiciones $\lambda_i > 0$ y $\mu_i > 0$ se tiene:

$$y_i(\mathbf{w}^\top \mathbf{x}_i + b - 0.5) - 0.5\varepsilon + \xi_i = 0, \quad i = 1, \ldots, m, \tag{123}$$

$$y_i(\mathbf{w}^\top \mathbf{x}_i + b - 0.5) - 0.5\varepsilon - \tfrac{1}{\tau}\xi_i = 0, \quad i = 1, \ldots, m, \tag{124}$$

  A continuación se tiene que multiplicando la segunda igualdad (124) por τ y luego sumándola a la primera (123) nos da:

$$(1 + \tau) y_i(\mathbf{w_i^T x_i} + b) - 0.5\varepsilon(1 + \tau) = 0 \to b = 0.5\varepsilon y_i - \mathbf{w_i^T x_i} \tag{125}$$

  Para hallar b se tiene que iterar sobre los vectores que cumplan las condiciones $\lambda_i > 0$, entendiendo que $\lambda_i = \alpha_i + \mu_i$, y $\mu_i > 0$ y calcular el promedio.

A continuación, se incluirá un gráfico en el que se representa el modelo PSVM pinball utilizando los distintos kernels disponibles: lineal, gaussiano, de base radial y polinómico de grado 2:

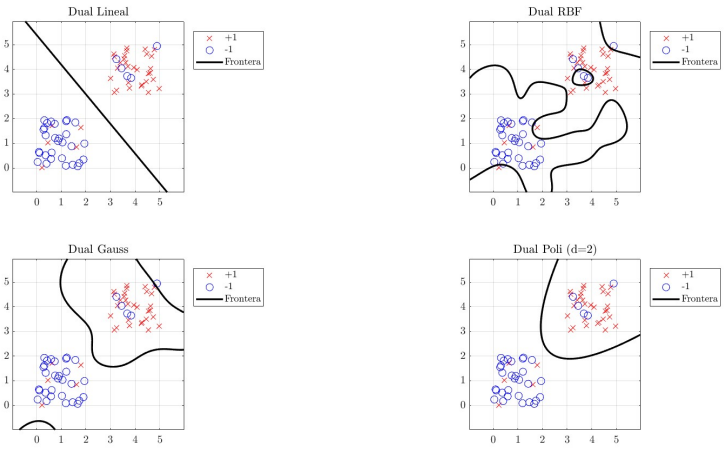

Figura 7: Modelo PSVM pinball

5. Pruebas numéricas

A continuación se evaluarán los distintos modelos descritos previamente utilizando tres conjuntos de datos de referencia de reducido tamaño, ampliamente empleados como bancos de prueba: **Sonar**, **Breast Cancer** e **Ionosphere**. Para más detalles de los datos utilizados para la evaluación ver [9].

El objetivo de este experimento es evaluar el comportamiento de los modelos pinball-SVM ante distintas configuraciones de sus hiperparámetros clave (C, τ y ε) como también del kernel no lineal. Para ello se explorarán rangos de valores propuestos en la literatura, siguiendo referencias como [8], [6], ver sección 5.2.

5.1. Medidas de evaluación de rendimiento

En primer lugar, se describirán las métricas empleadas para evaluar el rendimiento de los distintos modelos. En este trabajo, nos centraremos principalmente en el Accuracy y el Balanced Accuracy(también conocido como AUC).

Antes de introducir estas métricas, es necesario aclarar algunos conceptos básicos de validación de modelos. En tareas de clasificación, los resultados pueden agruparse en las siguientes categorías:

Definición 22 (Estados de clasificación)**.** Sea $\mathbf{x} \in \mathbb{R}^N$ con etiqueta $y \in \{+1, -1\}$. Sea

$$f : \mathbb{R}^N \to \mathbb{R}$$

una función de decisión cuya predicción es

$$\hat{y} = \operatorname{sign}\big(f(\mathbf{x})\big) \in \{+1, -1\}.$$

Los posibles resultados de la clasificación son:

- **Verdadero positivo (TP)**:

$$y = +1, \quad \hat{y} = +1. \tag{126}$$

- **Verdadero negativo (TN)**:

$$y = -1, \quad \hat{y} = -1. \tag{127}$$

- **Falso positivo (FP)**:

$$y = -1, \quad \hat{y} = +1. \tag{128}$$

- **Falso negativo (FN)**:

$$y = +1, \quad \hat{y} = -1. \tag{129}$$

Estos estados que se pueden dar al hacer una clasificación normalmente se suelen representar en un matriz, esta matriz se conoce como matriz de confusión:

Data class	Classified as *pos*	Classified as *neg*
pos	true positive (*tp*)	false negative (*fn*)
neg	false positive (*fp*)	true negative (*tn*)

$$\begin{bmatrix} tp & fn \\ fp & tn \end{bmatrix}$$

Figura 8: Matriz de confusión [10]

Según el contexto de los datos, puede resultar más crítico minimizar los falsos positivos que maximizar los verdaderos positivos. Esto ocurre, por ejemplo, en sistemas de detección de fraude o en diagnósticos médicos de enfermedades graves, donde un elevado número de falsos positivos puede acarrear consecuencias significativas, como altos costes económicos, daños a la reputación o riesgos para la salud.

A partir de las definiciones anteriores se pueden definir las siguientes medidas de evaluación de rendimiento:

Definición 23 (Accuracy)**.** Sea $\mathbf{x} \in \mathbb{R}^N$ con etiqueta $y \in \{+1, -1\}$. Sea

$$f : \mathbb{R}^N \to \mathbb{R}$$

una función de decisión cuya predicción es

$$\hat{y} = \operatorname{sign}\big(f(\mathbf{x})\big) \in \{+1, -1\}.$$

El Accuracy se define como:

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TP}. \tag{130}$$

El Accuracy mide la eficacia general del modelo simplemente analizando si ha clasificado todos lo datos de manera correcta.

Definición 24 (Balance Accuracy o AUC)**.** Sea $\mathbf{x} \in \mathbb{R}^N$ con etiqueta $y \in \{+1, -1\}$. Sea

$$f : \mathbb{R}^N \to \mathbb{R}$$

una función de decisión cuya predicción es

$$\hat{y} = \operatorname{sign}\big(f(\mathbf{x})\big) \in \{+1, -1\}.$$

El AUC se define como:

$$\text{AUC} = \frac{1}{2}\Big(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\Big). \tag{131}$$

El AUC cuantifica la capacidad del clasificador para discriminar correctamente entre las dos clases, es decir, su habilidad para minimizar tanto los falsos positivos como las falsos negativas.

5.2. Explicación de la evaluación

En esta sección se describen las distintas bases de datos que se utilizarán, así como el procedimiento seguido para evaluar los diferentes modelos.

Tal como se indicó anteriormente, se emplearán tres conjuntos de datos: **Sonar**, **Breast Cancer** e **Ionosphere**. A continuación, se presentan sus respectivas dimensiones:

Nombre	m	N
Sonar	208	60
Breast Cancer	569	30
Ionosphere	351	33

Tabla 1: Dimensiones de las bases de datos utilizadas

Se ha optado por utilizar conjuntos de datos de tamaño pequeño para mantener bajo control el coste computacional; de este modo, es posible centrar el esfuerzo en la evaluación de cada modelo.

Los modelos serán evaluados mediante un proceso de validación cruzada de diez particiones (10-fold cross-validation). En cada iteración, el conjunto de datos se dividirá en diez subconjuntos de tamaño aproximadamente igual.

Uno de estos subconjuntos se utilizará como conjunto de prueba, es decir, los datos sobre los que se evaluará el rendimiento del modelo, mientras que los nueve subconjuntos restantes se emplearán para el entrenamiento, es decir, para resolver el modelo.

Este proceso se repetirá diez veces, alternando el subconjunto utilizado como prueba en cada iteración, el rendimiento final se calculará como la media de los resultados obtenidos en cada repetición.

Para los hiperparámetros se usarán los siguientes rangos de valores:

- Para C se usará el siguiente rango:

$$\{2^{-7}, \ldots, 2^{7}\}. \tag{132}$$

- Para ε se usará el siguiente rango:

$$\{2^{-7}, \ldots, 1\}. \tag{133}$$

- Para τ se usará el siguiente rango según [8]:

$$\{1, 0.5, 0.2, 0.1\}. \tag{134}$$

5.2.1. Pruebas numéricas en Bases de Datos sin ruido

En esta sección se evaluarán los modelos PSVM con funciones de pérdida pinball y hinge loss, así como el modelo SVM con pérdida pinball y hinge loss, aplicando diferentes funciones kernel.

Se presentarán los mejores resultados obtenidos por cada modelo en las distintas bases de datos, calculando los valores de Accuracy $\times$ 100 y AUC $\times$ 100.

A continuación, se presentan dos Tablas con los valores máximos y promedios obtenidos a partir de la validación cruzada. En dichas Tablas se destacan en negrita los valores más altos obtenidos entre los cuatro modelos analizados, tanto para los promedios como para los máximos.

La validación cruzada se realizó mediante un proceso iterativo anidado sobre bases de datos, kernels e hiperparámetros. Los resultados se almacenaron en matrices guardadas en formato `.mat`. Se empleó una validación cruzada tipo *K-fold* con $K = 10$.

Estos valores se consiguen de las matrices resultantes de hacer la validacion cruzada, mas la utilización de un script que sacaba en maximo el promedio y los intervalos de confianza.

Kernel	Métrica	Base	SVM-Pinball	PSVM-Pinball	SVM-Hinge	PSVM-Hinge
Linear	Accuracy	Sonar	72.30 ± 0.93 (77.09)	71.11 ± 0.33 (78.91)	71.58 ± 2.25 (76.50)	**72.50 ± 0.63 (78.91)**
		Breast Cancer	**92.88 ± 0.37 (95.45)**	88.94 ± 0.23 (92.64)	91.76 ± 0.38 (94.00)	92.06 ± 0.74 (96.36)
		Ionosphere	85.92 ± 0.33 (88.09)	**86.35 ± 0.20 (89.09)**	83.85 ± 0.53 (86.18)	86.24 ± 0.42 (89.09)
	AUC	Sonar	72.00 ± 0.96 (76.83)	70.97 ± 0.34 (79.00)	71.53 ± 2.27 (79.17)	**72.31 ± 0.68 (79.00)**
		Breast Cancer	**91.22 ± 0.45 (94.29)**	85.43 ± 0.32 (90.54)	90.26 ± 0.38 (91.07)	89.68 ± 1.01 (95.54)
		Ionosphere	**84.67 ± 0.36 (87.08)**	83.41 ± 0.28 (87.08)	83.16 ± 0.60 (85.54)	83.94 ± 0.64 (87.50)
RBF	Accuracy	Sonar	83.79 ± 1.09 (88.00)	**85.63 ± 0.71 (87.91)**	83.91 ± 2.22 (88.00)	84.09 ± 1.26 (88.00)
		Breast Cancer	93.78 ± 0.40 (95.45)	91.66 ± 0.18 (95.45)	**94.11 ± 0.66 (95.45)**	93.14 ± 0.47 (96.36)
		Ionosphere	87.60 ± 0.25 (89.18)	87.89 ± 0.69 (92.18)	86.92 ± 0.35 (88.18)	**95.81 ± 1.14 (98.00)**
	AUC	Sonar	83.96 ± 1.09 (88.17)	**85.46 ± 0.79 (88.00)**	84.09 ± 2.23 (88.17)	83.55 ± 1.38 (87.83)
		Breast Cancer	92.41 ± 0.45 (94.29)	89.80 ± 0.20 (94.29)	**92.92 ± 0.68 (94.29)**	91.42 ± 0.62 (95.00)
		Ionosphere	89.75 ± 0.21 (91.07)	88.65 ± 0.97 (93.57)	89.18 ± 0.30 (90.24)	**95.07 ± 1.44 (97.50)**
Gaussiano	Accuracy	Sonar	82.87 ± 1.40 (89.82)	**85.74 ± 0.78 (89.82)**	83.12 ± 2.87 (89.82)	81.34 ± 1.95 (88.09)
		Breast Cancer	93.58 ± 0.47 (95.45)	90.64 ± 0.32 (96.36)	**93.61 ± 0.91 (95.45)**	92.08 ± 1.16 (96.36)
		Ionosphere	89.70 ± 0.37 (93.18)	91.42 ± 0.78 (94.09)	89.45 ± 0.59 (91.18)	**92.87 ± 1.65 (98.00)**
	AUC	Sonar	82.75 ± 1.42 (88.83)	**85.26 ± 0.86 (89.67)**	83.12 ± 2.87 (89.00)	80.58 ± 2.13 (87.83)
		Breast Cancer	92.00 ± 0.56 (94.29)	87.98 ± 0.40 (95.00)	**92.25 ± 1.04 (94.29)**	89.62 ± 1.60 (95.00)
		Ionosphere	91.50 ± 0.30 (94.40)	**91.63 ± 1.04 (95.12)**	91.29 ± 0.49 (92.74)	91.41 ± 2.10 (97.50)
Polinómico	Accuracy	Sonar	**83.73 ± 0.75 (89.82)**	83.26 ± 0.17 (89.82)	82.22 ± 2.34 (87.09)	57.09 ± 0.88 (75.18)
		Breast Cancer	**93.08 ± 0.34 (95.45)**	89.26 ± 0.39 (94.55)	92.50 ± 0.74 (94.55)	72.21 ± 2.70 (96.36)
		Ionosphere	**86.84 ± 2.16 (97.00)**	75.59 ± 0.67 (97.00)	85.43 ± 4.60 (95.09)	1.00 ± 0.00 (1.00)
	AUC	Sonar	**83.60 ± 0.75 (89.67)**	83.07 ± 0.17 (88.83)	82.11 ± 2.30 (86.83)	54.51 ± 1.06 (88.91)
		Breast Cancer	**91.38 ± 0.40 (94.29)**	86.41 ± 0.41 (92.50)	91.32 ± 1.04 (94.64)	64.94 ± 3.01 (95.00)
		Ionosphere	84.27 ± 2.55 (96.67)	71.17 ± 0.76 (96.25)	83.05 ± 5.69 (95.00)	**86.19 ± 0.79 (97.50)**

Tabla 2: Resultados validación cruzada de los cuatro modelos. Se muestra el valor promedio ± intervalo de confianza y el valor máximo (entre paréntesis). La negrita indica el mejor rendimiento promedio por fila.

En la Tabla 2 se observa que la mayoria de los modelos obtienen unos Acurracy y AUC similares estos se verificara obteniendo los promedios:

Modelo	Promedio Máximo Accuracy (%)	Promedio Máximo AUC (%)
SVM-Pinball	91.17	90.96
PSVM-Pinball	91.49	91.03
SVM-Hinge	90.35	90.06
PSVM-Hinge	**91.98**	**91.67**

Tabla 3: Promedio de los valores máximos de Accuracy y AUC para cada modelo.

Se puede observar en la Tabla 3, que no necesariamente los modelos pinball obtienen mejores resultados, esto en principio es esperable por que el objetivo de la función pinball es disminuir la sensibilidad a el ruido no aumentar la eficiencia [8].

Una observación importante es que, al realizar la validación cruzada, se han presentado limitaciones en cuanto a la cantidad de valores diferentes validados. Por ejemplo, parámetros esenciales en varios kernels, como σ y d (grado polinómico), estaban fijados.

5.2.2. Pruebas numéricas en Bases de Datos con ruido

A continuación, a partir de los resultados anteriores, se cogerán para cada base de datos y para cada modelo los hiperparámetros que hayan conseguido

un mejor AUC y un mejor Accuracy con los datos de la Tabla 2.

Luego estos hiperparámetros óptimos se validarán con las mismas bases de datos, pero con un ruido gaussiano del 5 %. Este es el ruido máximo que se suele atribuir a una base de datos para considerarla aceptable.

Este caso sería similar al de la validación cruzada sin ruido, con la diferencia de que los hiperparámetros se fijan y se eliminan los bucles asociados a ellos. Además, al cargar la base de datos, se le añade el ruido correspondiente.

Modelo	**Base**	**Métrica**	**Kernel**	**C**	**epsilon**	**tau**	**Valor r=0.00 (%)**	**Valor r=0.05 (%)**
SVM Pinball	Breast Cancer	Accuracy	gaussian	2^{0}	–	0.1	95.4545	87.00
		AUC	gaussian	2^{0}	–	0.1	94.2857	86.130
	Ionosphere	Accuracy	polinómico	2^{-2}	–	1	97.0000	83.181
		AUC	polinómico	2^{-2}	–	1	96.6667	83.154
	Sonar	Accuracy	polinómico	2^{1}	–	0.2	88.9091	68.363
		AUC	polinómico	2^{1}	–	0.2	89.0000	68.166
SVM Hinge	Breast Cancer	Accuracy	gaussian	2^{2}	–	–	95.4545	90.633
		AUC	polinómico	2^{1}	–	–	94.6429	77.142
	Ionosphere	Accuracy	polinómico	2^{-2}	–	–	95.0909	85.00
		AUC	polinómico	2^{-4}	–	–	95.0000	89.285
	Sonar	Accuracy	gaussian	2^{2}	–	–	69.6360	66.727
		AUC	gaussian	2^{2}	–	–	70.5000	67.000
PSVM Hinge	Breast Cancer	Accuracy	gaussian	2^{-4}	2^{-4}	–	96.3636	83.454
		AUC	lineal	2^{-7}	2^{-7}	–	95.5357	84.940
	Ionosphere	Accuracy	gaussian	2^{-7}	2^{-5}	–	98.0000	96.000
		AUC	gaussian	2^{-7}	2^{-5}	–	97.5000	95.833
	Sonar	Accuracy	gaussian	2^{-6}	2^{-6}	–	88.0909	63.636
		AUC	rbf	2^{4}	2^{-1}	–	87.8333	65.833
PSVM Pinball	Breast Cancer	Accuracy	gaussian	2^{-7}	2^{-3}	0.1	96.3636	90.818
		AUC	gaussian	2^{-7}	2^{-3}	0.1	95.0000	88.035
	Ionosphere	Accuracy	polinómico	2^{-7}	2^{-1}	0.1	97.0000	90.181
		AUC	polinómico	2^{-7}	2^{-1}	0.1	96.2500	88.452
	Sonar	Accuracy	gaussian	2^{-7}	2^{-5}	0.2	89.8182	52.000
		AUC	gaussian	2^{-7}	2^{-5}	0.2	89.6667	55.454

Tabla 4: Mejores hiperparámetros (C, epsilon, tau) y métricas (valor en %) para cada modelo y base de datos con y sin ruido

De la Tabla 4 se pueden extraer varias conclusiones. En primer lugar, el conjunto de datos Sonar resulta el más sensible al ruido.

Además, contrariamente a lo esperado, los modelos basados en la pérdida Pinball no reducen el efecto del ruido tanto como cabría anticipar.

Estas discrepancias pueden explicarse por varios factores. Como se señaló anteriormente, las restricciones en el conjunto de hiperparámetros evaluados pudieron impedir la exploración de configuraciones de modelo con un rendimiento potencialmente superior.

A continuación se medirá la robustez de los distintos modelos, la robustez trata de medir cuánto de estable es, es decir, cuánto poco se ve afectado un modelo al introducir cambios en sus datos. Para ello, se usará la siguiente métrica de robustez [11]:

$$N_{\text{rob}} = \frac{1}{100} \sum_{i=1}^{100} \frac{\left\| [w_{p,i},\, b_{p,i}] - [w_u,\, b_u] \right\|_2}{\left\| [w_u,\, b_u] \right\|_2} \tag{135}$$

Con esta fórmula evaluamos la robustez del modelo a partir de los vectores de pesos y sesgos obtenidos al resolver los problemas duales de cada variante. En concreto, $[w_{p,i}, b_{p,i}]$ y $[w_u, b_u]$ representan los pares (peso, sesgo) correspondientes a los hiperplanos de los modelos "p" (con datos perturbados) y "u" (con datos no perturbados), respectivamente.

Esta métrica resulta especialmente adecuada porque permite cuantificar con precisión la variación entre los posibles hiperplanos. Además, al basarse directamente en el vector de pesos, que no está disponible en las versiones kernelizadas, su aplicación se limita al caso lineal con sus resultados óptimos por modelo base y métrica.

Para calcular la robustez, se seguirá el siguiente proceso: a diferencia del caso de validación cruzada con ruido, aquí no se realizará un proceso iterativo anidado sobre los hiperparámetros.

En cambio, primero se calcularán el vector de pesos $\mathbf{w}$ y el sesgo b utilizando los datos sin perturbar, es decir el modelo u.

Posteriormente, se ejecutará un proceso iterativo con cien repeticiones, generando en cada iteración un nuevo vector de pesos $\mathbf{w}$ y un sesgo b, al aplicar perturbaciones aleatorias mediante ruido gaussiano, es decir los modelos p.

Modelo	**Base**	**C**	**epsilon**	**tau**	**r=0.05**
SVM Pinball	Breast Cancer	2^{-2}	–	0.5	0.4683
	Ionosphere	2^{2}	–	0.1	0.5460
	Sonar	2^{0}	–	0.1	0.8791
SVM Hinge	Breast Cancer	2^{2}	–	–	0.9690
	Ionosphere	2^{-2}	–	–	0.5421
	Sonar	2^{2}	–	–	1.0043
PSVM Hinge	Breast Cancer	2^{-7}	2^{-7}	–	0.2891
	Ionosphere	2^{-7}	2^{-3}	–	0.2197
	Sonar	2^{4}	2^{-1}	–	0.9883
PSVM Pinball	Breast Cancer	2^{-5}	2^{-2}	0.1	0.7652
	Ionosphere	2^{-7}	2^{-2}	0.5	0.1590
	Sonar	2^{-7}	2^{0}	1	0.3825

Tabla 5: Robustez de los modelos por base de datos y mejores hiperparámetros.

En la Tabla 5 se observa la insensibilidad al ruido de los modelos con función de pérdida tipo Pinball, a partir de los dos apartados, lo siguiente: los modelos que emplean la función de pérdida *Pinball* son, como era de esperar [8], ligeramente más robustos que sus equivalentes con pérdida *Hinge*:

Para ilustrar esta robustez, se incluirá además un gráfico en $\mathbb{R}^2$ con datos

Modelo	Pérdida	Robustez media
SVM	Pinball	0.6311
SVM	Hinge	0.83703
PSVM	Pinball	0.4999
PSVM	Hinge	0.4355

Tabla 6: Robustez media alcanzada por cada modelo

sintéticos, en el cual se representarán los distintos hiperplanos separadores obtenidos al introducir ruido en los datos originales.

Para generar las gráficas de los hiperplanos, se utilizan parámetros fijos. En todos los casos, los datos se perturban con el mismo nivel de ruido ($r = 0.05$).

Hiperparámetro	Valor
Parámetro de regularización (C)	1
Parámetro de proximidad (ϵ)	0.1
Parámetro de pérdida Pinball (τ)	0.5
Tipo de Kernel	Lineal
Varianza del ruido (σ^2)	0.05

Tabla 7: Hiperparámetros fijos utilizados para los graficos.

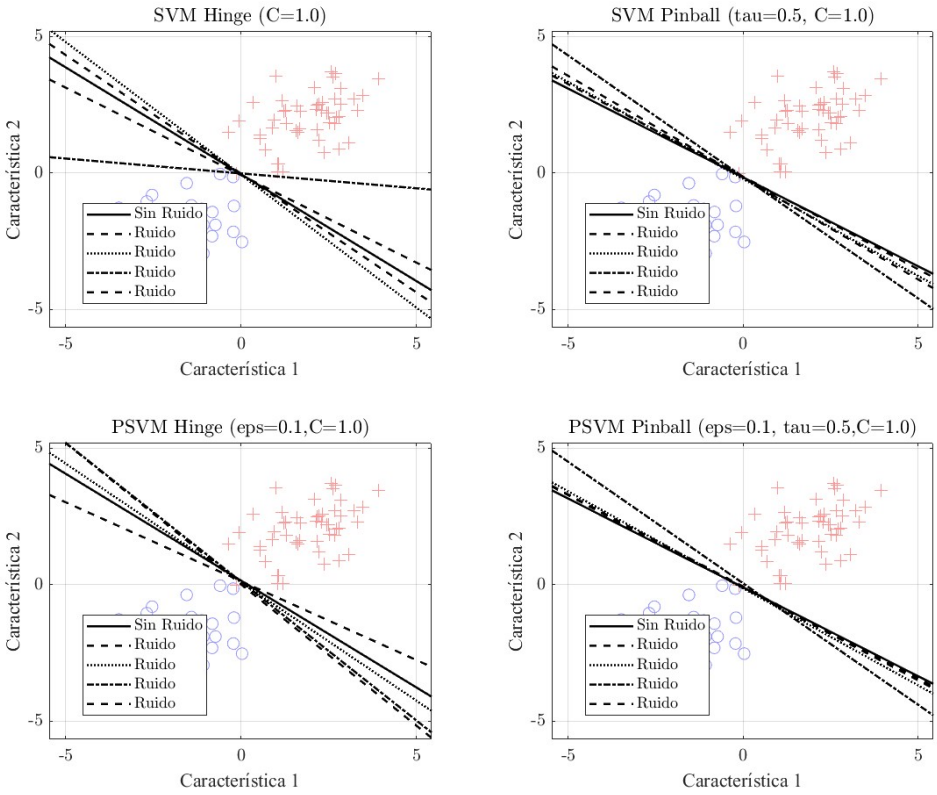

Figura 9: Robustibidad modelo SVM Hinge, PSVM Hinge, SVM Pinball y PSVM Pinball

En esta Figura 9 se puede apreciar mejor cómo los modelos Pinball tienden a ser más robustos en comparación con los modelos Hinge, aunque numericamente como se observa en la Tabla 6 no haya tanta diferencia.

Para analizar la sensibilidad de los modelos se realizará un proceso de validación cruzada similar al efectuado con los modelos sin ruido.

Concretamente, se implementarán varios bucles anidados que recorrerán, en primer lugar, las diferentes bases de datos, las cuales serán perturbadas con ruido gaussiano de valor 0.05, luego se iterará sobre los distintos kernels y los rangos definidos para los hiperparámetros. Finalmente, se llevará a cabo una validación cruzada tipo *K-fold* con $K = 10$.

Kernel	Métrica	Base	SVM-Pinball	PSVM-Pinball	SVM-Hinge	PSVM-Hinge
Linear	Accuracy	Sonar	56.46 ± 1.25 (65.64)	**66.83 ± 0.22 (73.27)**	61.70 ± 2.05 (66.64)	61.33 ± 0.27 (66.73)
		Breast Cancer	**89.75 ± 0.29 (92.64)**	84.46 ± 0.13 (87.00)	86.37 ± 0.52 (87.00)	88.97 ± 0.48 (92.55)
		Ionosphere	85.52 ± 0.30 (88.09)	84.06 ± 0.18 (87.09)	83.18 ± 0.37 (85.09)	**86.86 ± 0.43 (89.09)**
	AUC	Sonar	56.12 ± 1.23 (65.17)	**66.21 ± 0.25 (72.83)**	61.81 ± 2.06 (66.83)	60.63 ± 0.29 (65.50)
		Breast Cancer	**88.33 ± 0.29 (91.61)**	80.08 ± 0.21 (83.57)	85.42 ± 0.35 (86.25)	85.80 ± 0.71 (90.54)
		Ionosphere	**84.15 ± 0.25 (86.25)**	81.62 ± 0.20 (85.12)	82.07 ± 0.30 (83.75)	84.11 ± 0.59 (86.67)
RBF	Accuracy	Sonar	51.73 ± 0.00 (51.73)	53.64 ± 0.00 (53.64)	**56.18 ± 0.00 (56.18)**	53.64 ± 0.00 (53.64)
		Breast Cancer	**91.74 ± 0.05 (91.82)**	74.10 ± 0.31 (76.09)	84.72 ± 0.22 (85.18)	84.59 ± 1.23 (87.00)
		Ionosphere	63.45 ± 0.00 (63.45)	60.36 ± 0.00 (60.36)	**71.36 ± 0.00 (71.36)**	60.36 ± 0.00 (60.36)
	AUC	Sonar	53.00 ± 0.00 (53.00)	50.00 ± 0.00 (50.00)	**57.33 ± 0.00 (57.33)**	50.00 ± 0.00 (50.00)
		Breast Cancer	**91.14 ± 0.21 (91.96)**	65.24 ± 0.44 (68.04)	84.40 ± 0.15 (85.24)	79.20 ± 1.69 (82.50)
		Ionosphere	69.76 ± 0.00 (69.76)	50.00 ± 0.00 (50.00)	**76.31 ± 0.00 (76.31)**	50.00 ± 0.00 (50.00)
Gaussiano	Accuracy	Sonar	60.36 ± 0.12 (60.91)	58.41 ± 0.17 (59.18)	**63.48 ± 0.24 (63.91)**	59.23 ± 0.39 (60.09)
		Breast Cancer	**91.73 ± 0.24 (93.55)**	82.12 ± 0.42 (85.18)	87.86 ± 0.51 (88.82)	89.56 ± 1.14 (91.64)
		Ionosphere	**81.98 ± 0.35 (84.18)**	62.94 ± 0.09 (64.36)	75.30 ± 0.13 (75.36)	66.30 ± 0.43 (67.36)
	AUC	Sonar	60.57 ± 0.13 (61.17)	55.17 ± 0.19 (56.00)	**63.94 ± 0.22 (64.33)**	56.07 ± 0.42 (57.00)
		Breast Cancer	**90.28 ± 0.20 (92.32)**	77.66 ± 0.62 (81.61)	85.90 ± 0.44 (87.14)	86.38 ± 1.58 (89.29)
		Ionosphere	**85.11 ± 0.28 (86.90)**	53.59 ± 0.13 (55.42)	79.59 ± 0.11 (79.64)	57.78 ± 0.56 (59.17)
Polinómico	Accuracy	Sonar	59.26 ± 0.20 (62.00)	**67.76 ± 0.07 (74.00)**	59.91 ± 2.32 (70.45)	66.72 ± 0.11 (68.45)
		Breast Cancer	82.21 ± 2.03 (91.64)	63.35 ± 0.68 (83.36)	73.79 ± 6.90 (89.73)	**86.99 ± 0.97 (90.73)**
		Ionosphere	86.54 ± 0.77 (93.09)	80.74 ± 0.20 (89.18)	86.14 ± 2.27 (94.00)	**87.90 ± 0.54 (91.09)**
	AUC	Sonar	58.75 ± 0.21 (62.00)	**67.29 ± 0.07 (73.50)**	60.27 ± 2.30 (70.83)	66.23 ± 0.12 (68.17)
		Breast Cancer	81.20 ± 2.02 (90.36)	61.86 ± 0.57 (78.57)	73.60 ± 6.28 (87.68)	**84.57 ± 1.01 (88.57)**
		Ionosphere	84.76 ± 0.81 (91.25)	78.13 ± 0.21 (87.20)	84.69 ± 2.50 (93.33)	**86.21 ± 0.59 (88.75)**

Tabla 8: Resultados promedio (± IC 95 %) y máximos (entre paréntesis) para los cuatro modelos con ruido.

En esta Tabla 8 se puede ver que los modelos Pinball de media tienen unos mejores resultados.

Para verificar, se calculará una media de los valores máximos de cada modelo, es decir, los indicados en el 8 y se compararán entre sí y con los de la Tabla 3 (datos sin ruido) :

Modelo	**Promedio Máximo Accuracy (%)**	**Promedio Máximo AUC (%)**
SVM-Pinball	**78.23**	**78.48**
PSVM-Pinball	74.39	70.16
SVM-Hinge	77.81	78.22
PSVM-Hinge	76.56	73.01

Tabla 9: Promedio de los valores máximos de Accuracy y AUC para cada modelo, calculado a través de todos los kernels y bases de datos.

En la Tabla 9, se observa que el modelo con mejores resultados en Accuracy y AUC es el SVM-Pinball.

Por otro lado, el modelo con peor desempeño en ambas métricas es el PSVM-Pinball, lo cual es un resultado inesperado.

Podría pensarse inicialmente que esta diferencia se debe a una posible menor eficacia del modelo PSVM-Pinball sobre las bases de datos utilizadas; sin embargo, al observar la Tabla 3, queda claro que todos los modelos presentan valores de Accuracy y AUC muy similares.

Este resultado puede deberse principalmente, como ya se ha señalado, a una exploración insuficiente del espacio de hiperparámetros.

Probablemente, si se hubieran evaluado rangos diferentes o se hubiese realizado una búsqueda más exhaustiva iterando sobre los hiperparámetros d y σ, se habrían obtenido resultados que reflejaran una ventaja clara de los modelos Pinball en presencia de ruido.

6. Conclusiones

A continuación se presentan las conclusiones obtenidas en este trabajo. Para situarlas en contexto, recordemos que el objetivo principal fue analizar y comparar distintas variantes del modelo SVM. Con ese fin se revisó la literatura especializada, se implementaron los modelos en Matlab y se describieron sus características. Por último, se llevó a cabo un análisis y una evaluación de su desempeño en casos de interés.

6.1. Resultados por sección

La sección 1, tal como se explicó al comienzo, es de carácter introductorio y en ella se presenta el contexto general del trabajo, sus objetivos y su estructura.

En la sección 2 se introdujeron los conceptos básicos necesarios para comprender los modelos PSVM. Primero se presentaron las nociones generales de convexidad y su utilidad; a continuación, se definieron los distintos tipos de óptimos que pueden existir en un problema de optimización. Seguidamente, se describieron de forma general los dos tipos de problemas de optimización con restricciones, aunque, como se indicó, estos métodos no se emplean en los modelos SVM, y se profundizó en los métodos de optimización con restricciones, explicando sus criterios de convexidad. Después, se abordó el concepto de dualidad y el procedimiento para obtener la forma dual de un problema, y, por último, se expusieron las condiciones KKT utilizadas para resolver problemas de optimización con restricciones.

En la sección 3 se implementaron las versiones básicas de los modelos SVM.

En primer lugar, se introdujeron los conceptos fundamentales para abordar este tipo de problemas, problema de clasificación, hiperplano de separación, separabilidad entre clases y funciones kernel, y, a partir de ellos, se desarrolló el modelo SVM Hard Margin, que requiere separabilidad de los datos. Se detalló su formulación primal, el paso a su forma dual y el cálculo de sus parámetros ($\mathbf{w}$ y b). A continuación, se presentó el modelo Soft Margin para datos no separables, explicando su motivación, diferencias con el Hard Margin, su forma dual y la obtención de sus parametros ($\mathbf{w}$ y b). Tras ello, se incorporó la versión kernel en los modelos anteriores y algunos de los kernels más comunes.

Paralelamente a la exposición teórica, se implementaron los modelos: primero el SVM Hard Margin, luego el Soft Margin y, por último, las variantes con kernel, todas ellas usando CVX. Finalmente, se desarrolló una implementación kernel dual, válida para datos separables y no separables, inicialmente en CVX pero por motivos de rendimiento, se hizo otra versión con la librería quadprog. Además, hicieron gráficos ilustrativos de los hiperplanos de separación de los modelos SVM Hard Margin y SVM Soft Margin.

En la sección 4 se presentaron variantes más avanzadas del modelo SVM, centrándonos en el PSVM. Primero, se expuso en qué se diferenciaba respecto a los modelos básicos tratados en la sección 3 y, a continuación, se explicó su formulación primal y dual, así como el cálculo del vector de pesos y del sesgo. En la fase de implementación, se implementó la forma dual del PSVM compatible con diversos kernels, tanto en CVX como en quadprog, y se generaron gráficos que ilustraron su comportamiento del modelo PSVM con distintos kernels.

Seguidamente, se analizaron dos funciones de pérdida: la clásica hinge, empleada en los modelos anteriores explicados en la sección 3, y la novedosa pinball, habitual en regresión, cuyo uso se explicó como estrategia para mitigar el ruido. A partir de su concepto, se derivó el modelo SVM Pinball, detallando su forma dual y el cálculo de sus parámetros, y se acompañó de las correspondientes representaciones gráficas.

Finalmente, se mostró la transformación del PSVM hinge al PSVM pinball, la aportación más innovadora de este trabajo. Para ello, se introdujo su formulación primal,luego se desarrollo su forma dual y se simplificó paso a paso, luego se definió su extensión con kernel, concluyendo con las gráficas que evidenciaron el comportamiento del PSVM Pinball con distintos kernels.

En la sección 5 se llevaron a cabo las pruebas numéricas. En primer lugar, se presentaron las bases de datos utilizadas y se describieron las métricas de evaluación para contextualizar los resultados. A continuación, se detalló cada conjunto —sus tamaños y la motivación de su elección— y se explicó el proceso de validación de los modelos, que finalmente se puso en práctica.

Los resultados mostraron que, en datos sin ruido, no existieron diferencias significativas entre las distintas variantes de SVM. Sin embargo, en presencia de ruido surgieron hallazgos inesperados: algunos modelos con función de

pérdida Pinball obtuvieron, de media, peores resultados que sus homólogos con Hinge-loss, aunque el mejor rendimiento global correspondió precisamente a los modelos Pinball. Por último, en los análisis de robustez se confirmó que las versiones con pérdida Pinball eran ligeramente más robustas que sus contrapartes Hinge.

6.2. Limitaciones

Una de las principales limitaciones que encontramos fue el rendimiento de los distintos modelos. En un primer momento, lo solucionamos reformulando los modelos con la librería quadprog, mucho más eficiente y rápida, aunque las validaciones seguían consumiendo un tiempo excesivo.

Esta demora impidió realizar pruebas adicionales con más bases de datos y variaciones de los hiperparámetros (d y σ), lo cual podría explicar los resultados inesperados de la sección 5 y condicionar la obtención de métricas superiores en los datos con ruido.

6.3. Lineas futuras

En futuras líneas de trabajo, sería recomendable realizar una exploración más exhaustiva de los hiperparámetros para evaluar con mayor claridad si los modelos Pinball mitigan el ruido de forma más eficaz que los modelos Hinge.

Asimismo, convendría investigar otras herramientas y bibliotecas que optimicen la implementación de los modelos, incrementando su eficiencia y facilitando las pruebas con más conjuntos de datos y un rango más amplio de hiperparámetros.

También como línea futura, se propone realizar pruebas numéricas utilizando mas bases de datos ya que en este trabajo solo se han utilizado tres bases de datos lo cual es un numero limitado a la hora de poder sacar conclusiones sobre el comportamiento de los distintos modelos.

Por último sería conveniente llevar a cabo un análisis estadístico más exhaustivo, que permita comparar de forma más rigurosa las ventajas y limitaciones de cada enfoque. Para ello, podrían emplearse estrategias de validación cruzada como Leave-One-Out [11] y pruebas estadísticas más robustas como el test de Nemenyi o de Friedman [6].

Bibliografía

[1] V. Vapnik y C. Cortes. "Support-Vector Networks". En: *AT&T Bell Labs., Holmdel, NJ 07733, USA* 1 (1995), págs. 1-25.

[2] A. Beck. *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB*. Ed. por K. Scheinberg. MOS–SIAM Series on Optimization. Philadelphia, PA: Society for Industrial, Applied Mathematics y Mathematical Optimization Society, 2014. ISBN: 978-1-611973-64-8.

[3] S. Boyd y L. Vandenberghe. *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004. ISBN: 978-0-521-83378-3.

[4] N. Deng, Y. Tian y C. Zhang. *Support Vector Machines: Optimization-Based Theory, Algorithms, and Extensions*. Data Mining and Knowledge Discovery Series. Taylor & Francis. Chapman & Hall/CRC, 2012. DOI: 10.1201/b14297. URL: https://doi.org/10.1201/b14297.

[5] L. Li. *Selected Applications of Convex Optimization*. Vol. 103. Springer Optimization and Its Applications. Berlin Heidelberg: Springer y Tsinghua University Press, 2015. ISBN: 978-3-662-46355-0. DOI: 10.1007/978-3-662-46356-7.

[6] M. Carrasco et al. "Machine Learning in Operations Research: New Insights into Conditional Probability Support Vector Machine Models". En: *Preprint* (2025). Available upon request.

[7] V. Vapnik y R. Izmailov. "Reinforced SVM method and memorization mechanisms". En: *Pattern Recognition* 119 (2021), pág. 108018. DOI: 10.1016/j.patcog.2021.108018.

[8] X. Huang, L. Shi y J. A. K. Suykens. "Support Vector Machine Classifier with Pinball Loss". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.5 (2014), págs. 984-996. DOI: 10.1109/TPAMI.2013.178.

[9] A. Asunción y D. Newman. *UCI Machine Learning Repository*. http://archive.ics.uci.edu/ml. Accedido: 11 de junio de 2025. 2007.

[10] Marina Sokolova y Guy Lapalme. "A systematic analysis of performance measures for classification tasks". En: *Information Processing & Management* 45.4 (2009), págs. 427-437. ISSN: 0306-4573. DOI: https://doi.org/10.1016/j.ipm.2009.03.002. URL: https://www.sciencedirect.com/science/article/pii/S0306457309000259.

[11] Miguel Carrasco et al. "Embedded feature selection for robust probability learning machines". En: *Pattern Recognition* 159 (2025), pág. 111157. ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2024.111157. URL: https://www.sciencedirect.com/science/article/pii/S0031320324009087.