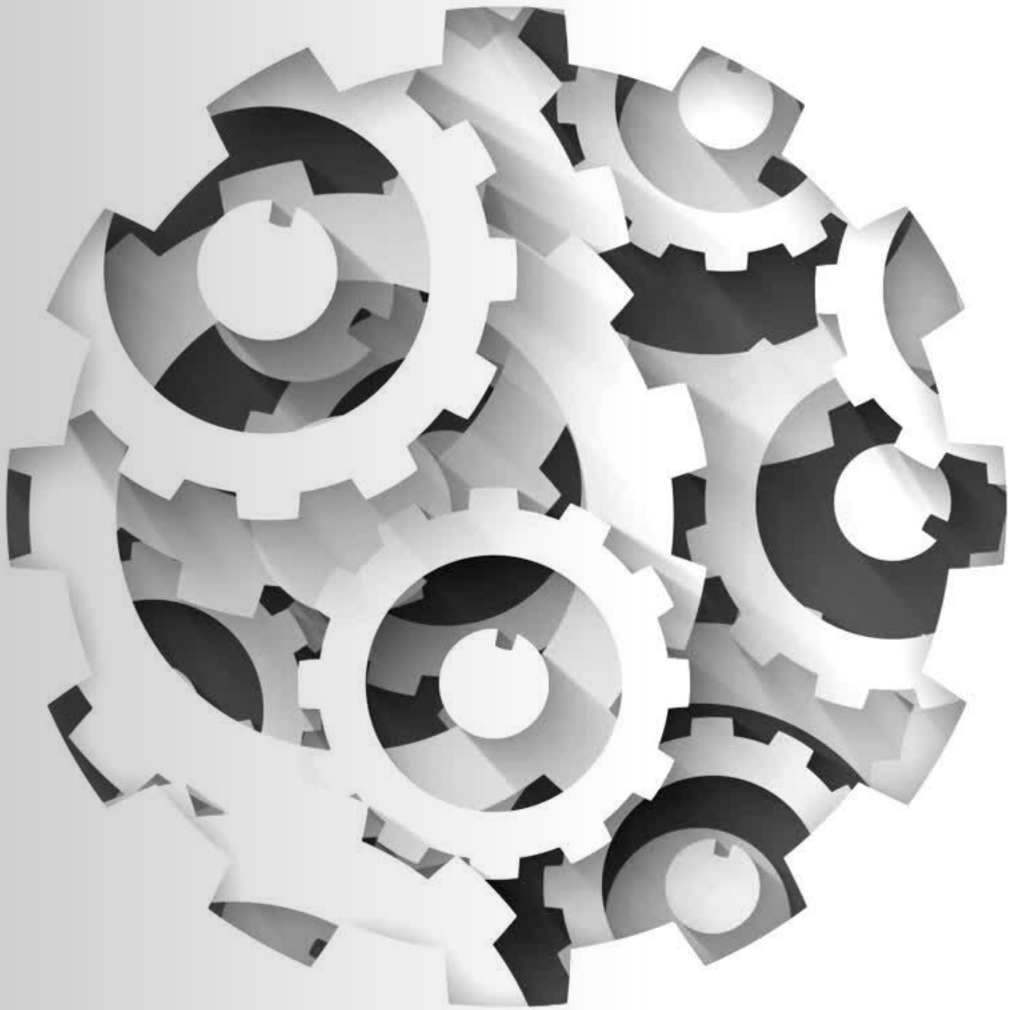
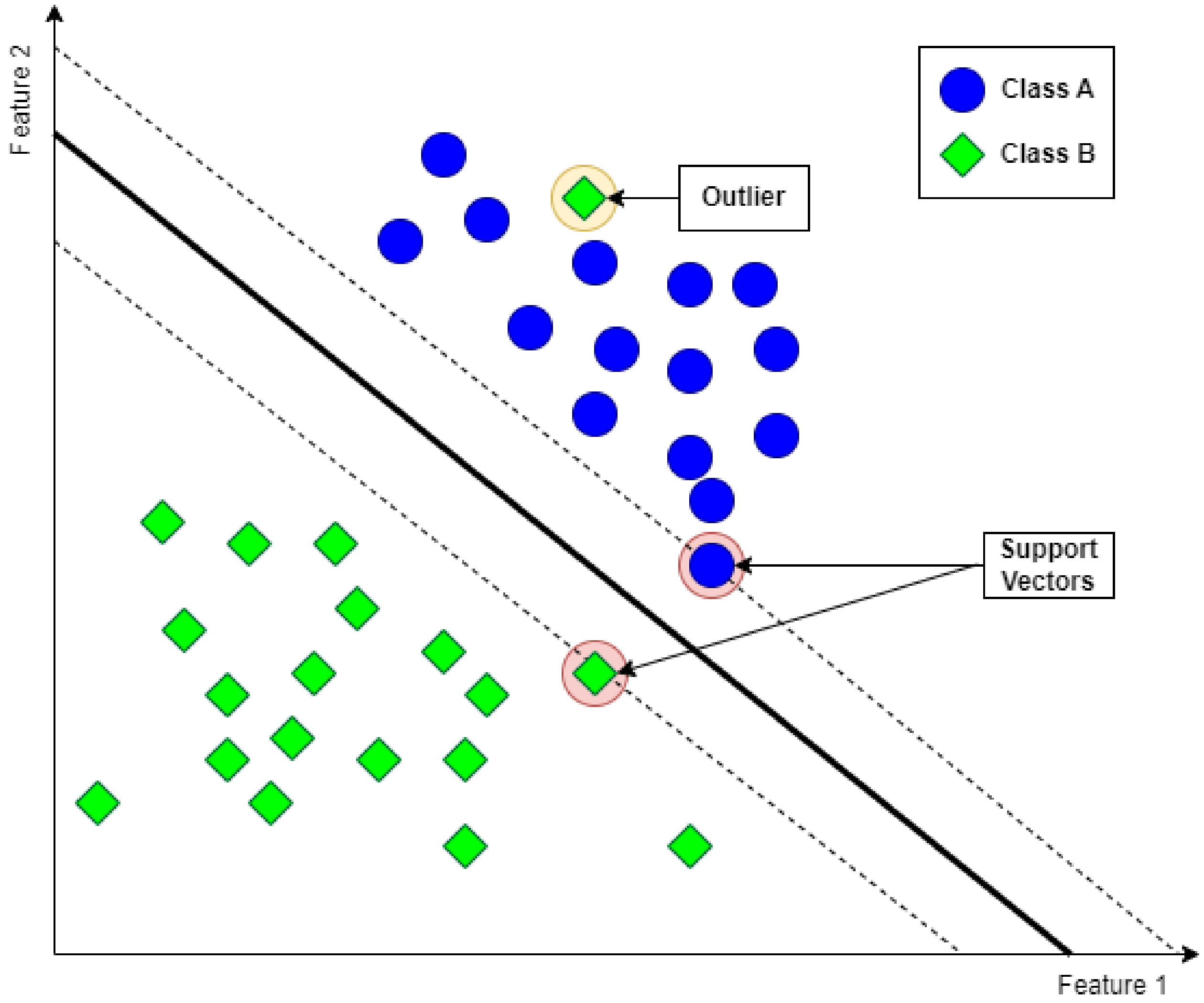
SEMANA 5

INTRODUCCIÓN A MACHINE LEARNING

Universidad Tecnológica de Panamá
Facultad de Ingeniería Eléctrica
Maestría en Ingeniería Eléctrica

SUPPORT VECTOR MACHINE





SUPPORT VECTOR MACHINE (SVM)

Un **Support Vector Machine (SVM)** es un algoritmo de **clasificación supervisada** (aunque también puede adaptarse para regresión) que busca encontrar el **hiperplano óptimo** que **separa dos clases** de datos con el mayor **margen posible**.

¿QUÉ SIGNIFICA ESO?

Imagina que tienes puntos de dos clases (por ejemplo, positivos y negativos) en un plano. El SVM intenta encontrar una línea (en 2D) o un plano (en dimensiones mayores) que:

- **Separe correctamente** los puntos de ambas clases.
- Lo haga de forma que **la distancia entre los puntos más cercanos de cada clase y la línea sea máxima**. Esa distancia se llama **margen**.

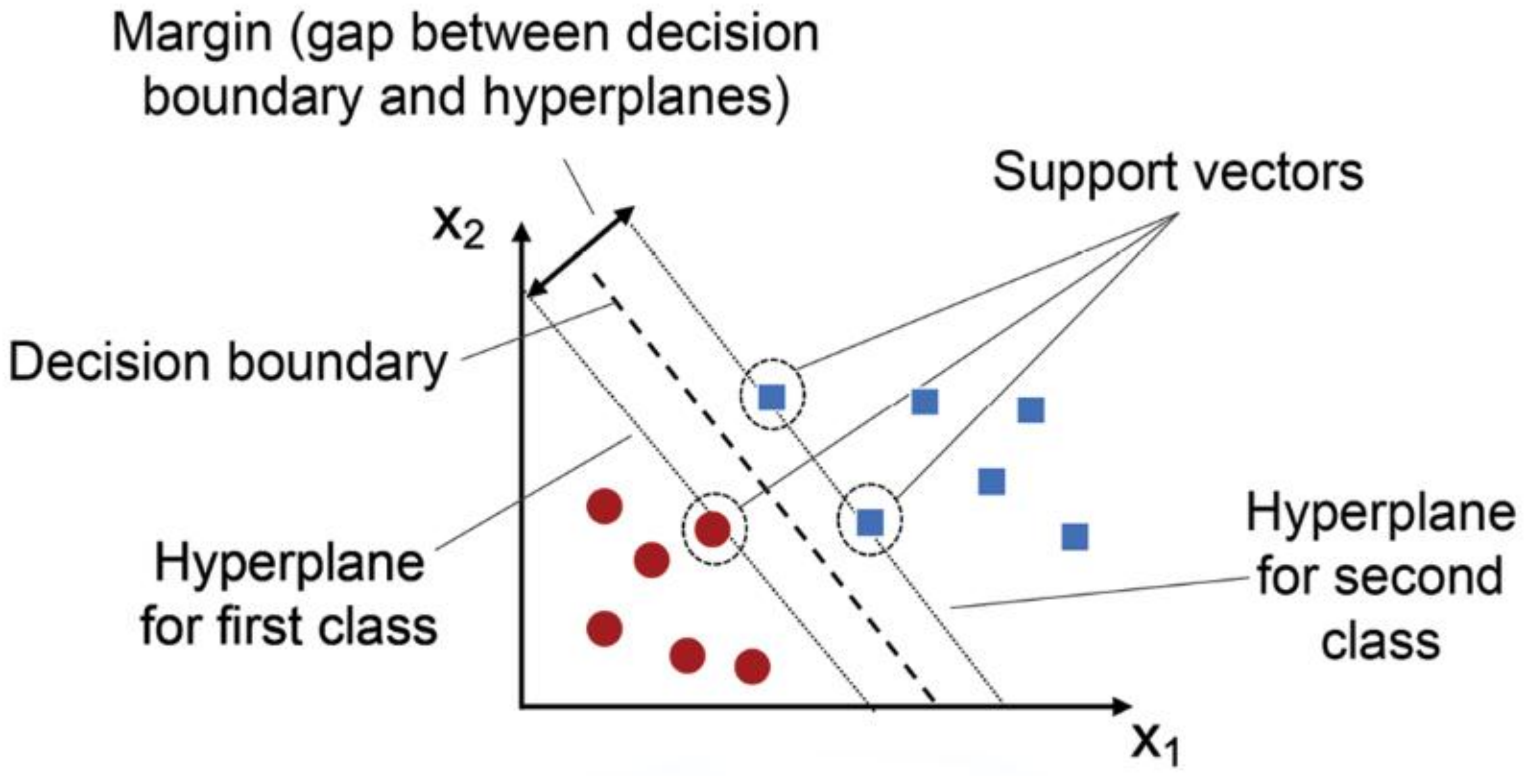
ELEMENTOS CLAVE

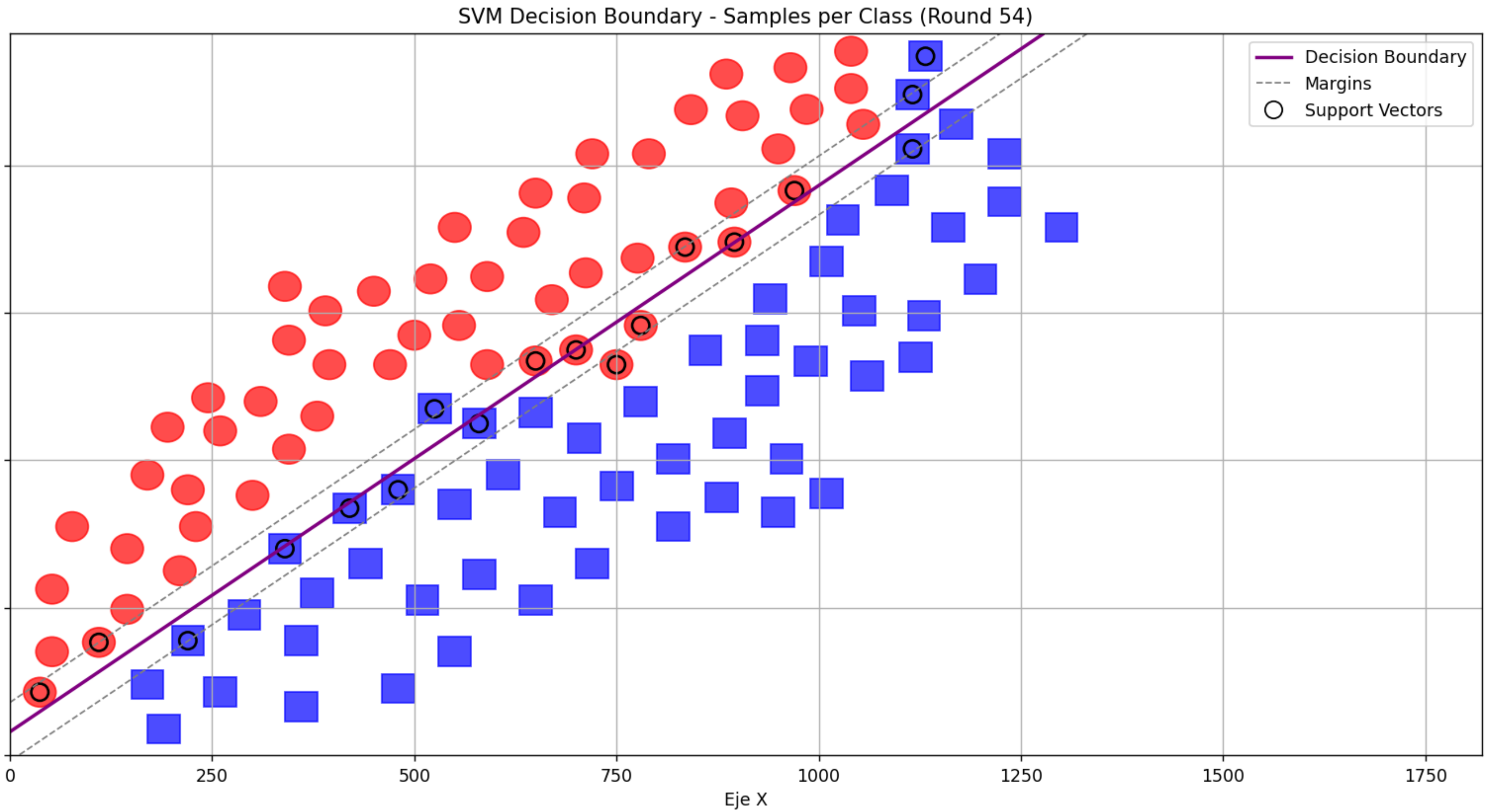
- **Hiperplano**: en 2D es una línea recta, en 3D es un plano, en más dimensiones es un hiperplano. Es la frontera de decisión.
- **Vectores de soporte**: los puntos más cercanos al hiperplano. Son los que “definen” la frontera.
- **Margen**: la distancia desde el hiperplano hasta los vectores de soporte. El SVM busca maximizar este margen.

TIPOS DE SVM

- **Lineal (margen duro)**: se aplica cuando los datos son perfectamente separables. No permite errores.
- **Lineal (margen blando)**: permite que algunos puntos se clasifiquen mal, ideal para datos no perfectamente separables.
- **No lineal**: usa el **truco del kernel** para proyectar los datos a un espacio de mayor dimensión donde sí pueden separarse linealmente.

SUPPORT VECTOR MACHINE





SUPPORT VECTOR MACHINE

¿CÓMO SE ELIGEN LOS VECTORES DE SOPORTE EN UN SVM?

Los **vectores de soporte** no se eligen manualmente: el propio algoritmo de SVM los **determina automáticamente** durante el proceso de **entrenamiento**, a través de la **optimización matemática**.

¿QUÉ BUSCA HACER EL SVM?

En un **SVM lineal**, el objetivo es encontrar el **hiperplano**:

$$w^\top x + b = 0$$

que:

- **Separe correctamente** las clases,
- **Maximice el margen** entre las clases (la distancia entre el hiperplano y los puntos más cercanos),
- Permita ciertos errores si se usa un **margen blando** (cuando los datos no son perfectamente separables).

¿QUÉ ES EL MARGEN?

El **margen** es la distancia entre el hiperplano y los puntos más cercanos de cada clase. El SVM intenta **maximizar ese margen**, buscando la frontera que más se aleje de los datos de ambas clases.

El SVM resuelve un problema de **optimización convexa**.

- **Caso ideal (margen duro, datos separables):**

$$\min_{w,b} \frac{1}{2}\|w\|^2 \quad \text{sujeto a} \quad y_i(w^\top x_i + b) \geq 1 \quad \forall i$$

- **Caso realista (margen blando, permite errores):**

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i \quad \text{sujeto a} \quad y_i(w^\top x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0$$

Donde:

- x_i son los vectores de entrenamiento,
- $y_i \in \{-1, +1\}$ son las clases,
- ξ_i son los márgenes de error (permiten que se violen algunas condiciones),
- C controla cuánto penalizamos esos errores.

¿QUÉ PROBLEMA RESUELVE EL SVM?

¿QUÉ PUNTOS SE CONVIERTEN EN VECTORES DE SOPORTE?

Durante la optimización, se resuelve una versión dual del problema usando multiplicadores de Lagrange.

Los vectores de soporte son aquellos puntos para los cuales el multiplicador de Lagrange $\alpha_i > 0$.

Intuitivamente:

- Son los **puntos que están exactamente sobre el margen**, o incluso dentro del margen si hay error.
- Son los **únicos que influyen en la posición del hiperplano**.
- Todos los demás puntos **no afectan** la solución: el SVM solo “se apoya” en estos pocos puntos para construir la frontera de decisión.

MARGIN VIOLATORS

Son los puntos del conjunto de entrenamiento que no respetan el margen definido por el SVM. Es decir, están dentro del margen o incluso mal clasificados.

- En los modelos con **margen flexible** (*soft margin*), estos puntos son permitidos y penalizados durante el entrenamiento. Su presencia permite al modelo adaptarse mejor a datos reales que no son perfectamente separables.
- Incluso con un **margen rígido** (*hard margin*), si los datos no son perfectamente separables, pueden surgir *margin violators* y provocar que el modelo sea inestable o no encuentre solución.

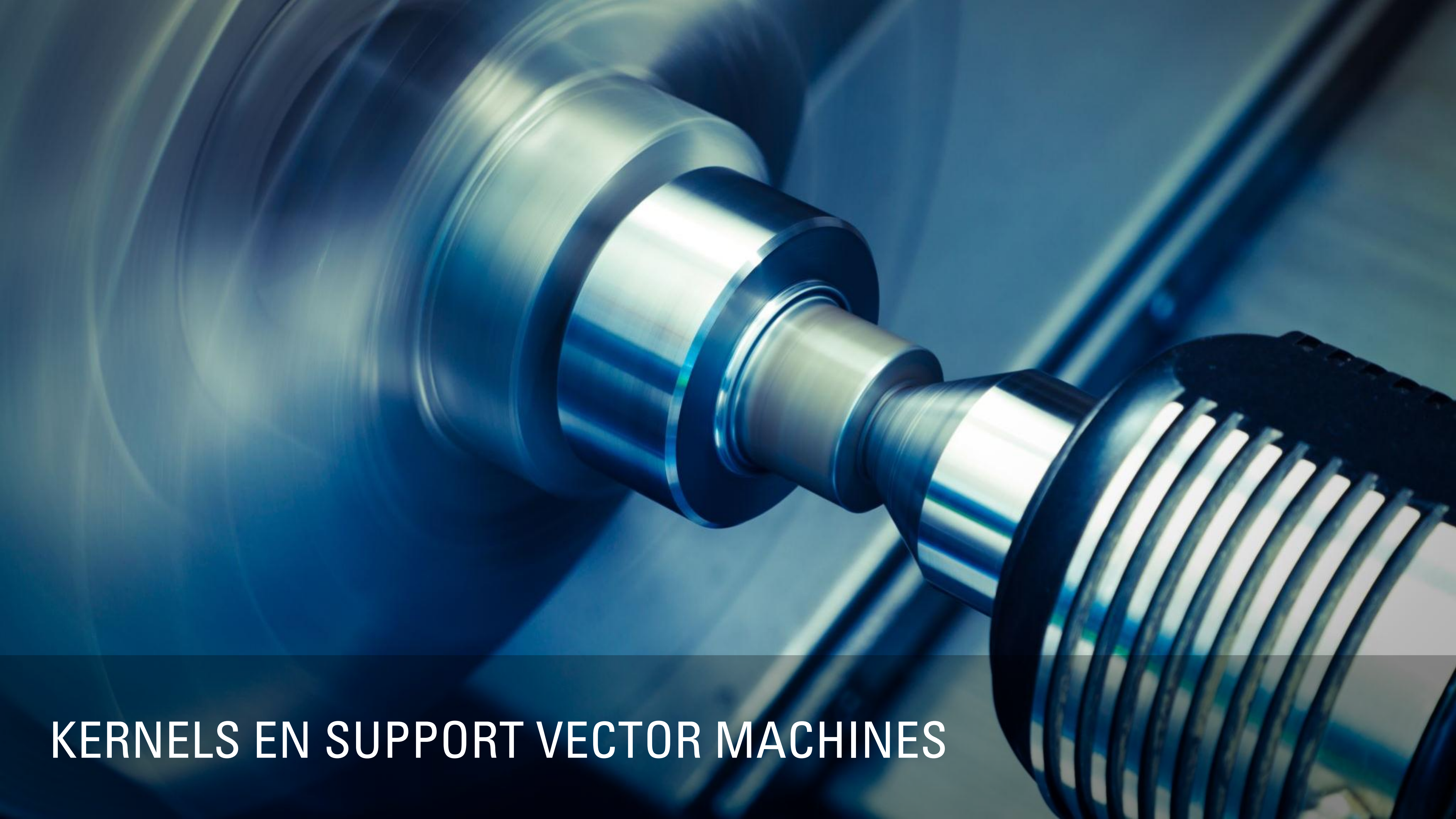
PARÁMETRO C

El parámetro C es el **parámetro de penalización** en las Máquinas de Vectores de Soporte (SVM). Controla el equilibrio entre tener un **margen amplio** y cometer **pocos errores de clasificación**.

¿Qué hace C?

- Un valor **alto de C** penaliza fuertemente a los *margin violators*. El modelo intentará clasificar todos los puntos correctamente, aunque eso signifique un margen más estrecho.
- Un valor **bajo de C** permite que existan más errores (más *margin violators*), con tal de mantener un margen más amplio.

Esto puede ayudar a que el modelo generalice mejor, sobre todo en presencia de ruido.



KERNELS EN SUPPORT VECTOR MACHINES

¿QUÉ SON LOS KERNELS EN SVM?

En el contexto de las Máquinas de Vectores de Soporte (SVM), un **kernel** es una **función matemática** que permite al modelo trabajar en un **espacio de características diferente** sin tener que transformarlo explícitamente.

DEFINICIÓN SIMPLE

Un **kernel** calcula la **similitud** entre dos puntos como si estuvieran en un espacio de mayor dimensión, **sin necesidad de transformar los datos directamente**.

¿PARA QUÉ SIRVE?

Para que el SVM pueda encontrar una **frontera de decisión no lineal** en el espacio original.

Permite resolver problemas donde **las clases no se pueden separar con una línea recta**.

TIPOS DE KERNEL EN SVM

Kernel	Modelo matemático	Uso típico
Lineal	$K(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{x}'$	Datos linealmente separables. Rápido y simple.
Polinómico	$K(\mathbf{x}, \mathbf{x}') = (\mathbf{x}^\top \mathbf{x}' + c)^d$	Permite aprender curvas polinómicas.
RBF (Gaussiano)	$K(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2)$	Muy usado para patrones complejos no lineales.
Sigmoide	$K(\mathbf{x}, \mathbf{x}') = \tanh(\alpha \mathbf{x}^\top \mathbf{x}' + c)$	Inspirado en redes neuronales. Uso menos frecuente.

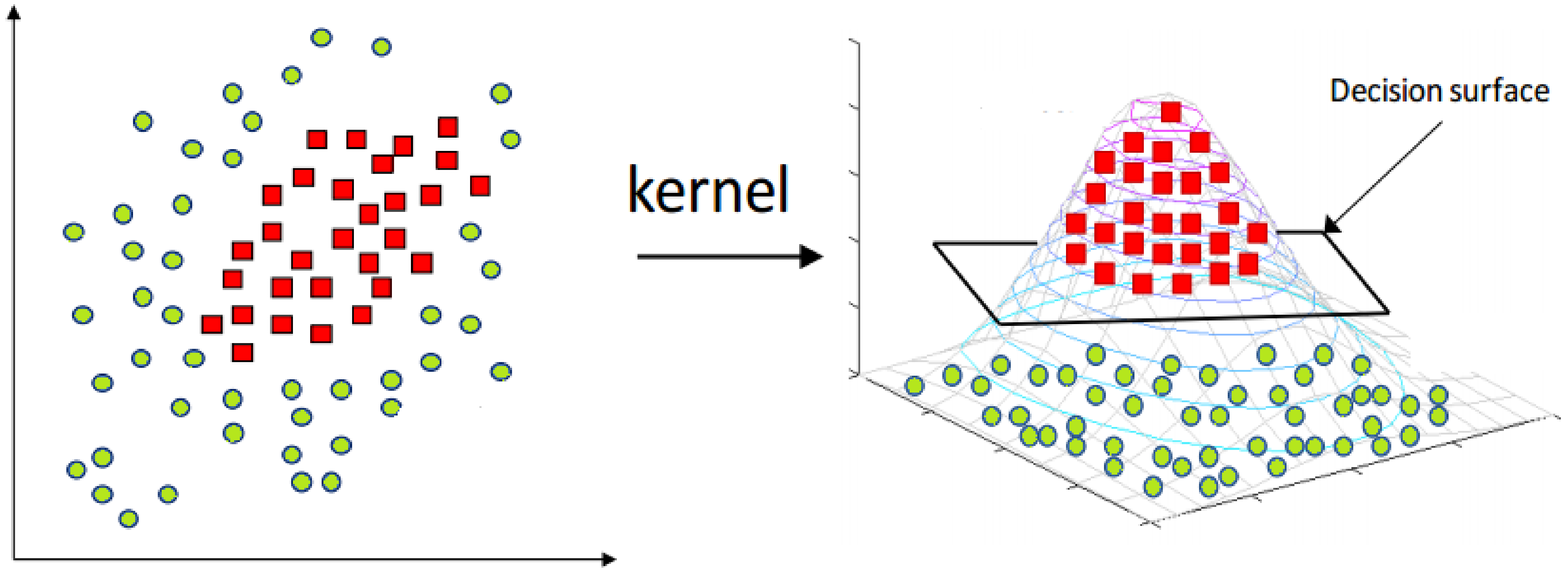
KERNEL RBF (RADIAL BASIS FUNCTION)

Es una función de similitud que mide **qué tan cerca están dos puntos** en el espacio. Define una transformación **no lineal** que permite separar clases con fronteras curvas o complejas.

Ideal para:

- Datos **no linealmente separables**
- Fronteras de decisión **suaves o complejas**
- Casos donde los patrones están **agrupados por proximidad**

KERNEL TRICK



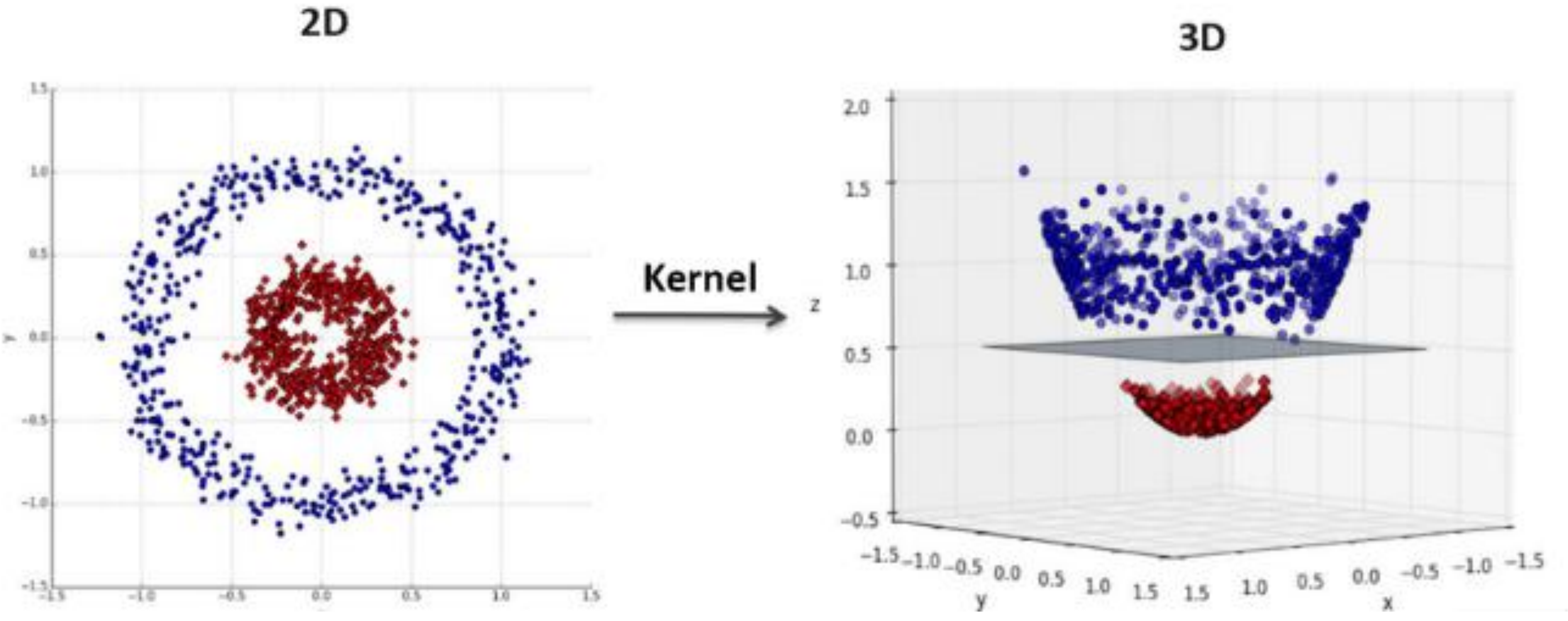
KERNEL TRICK

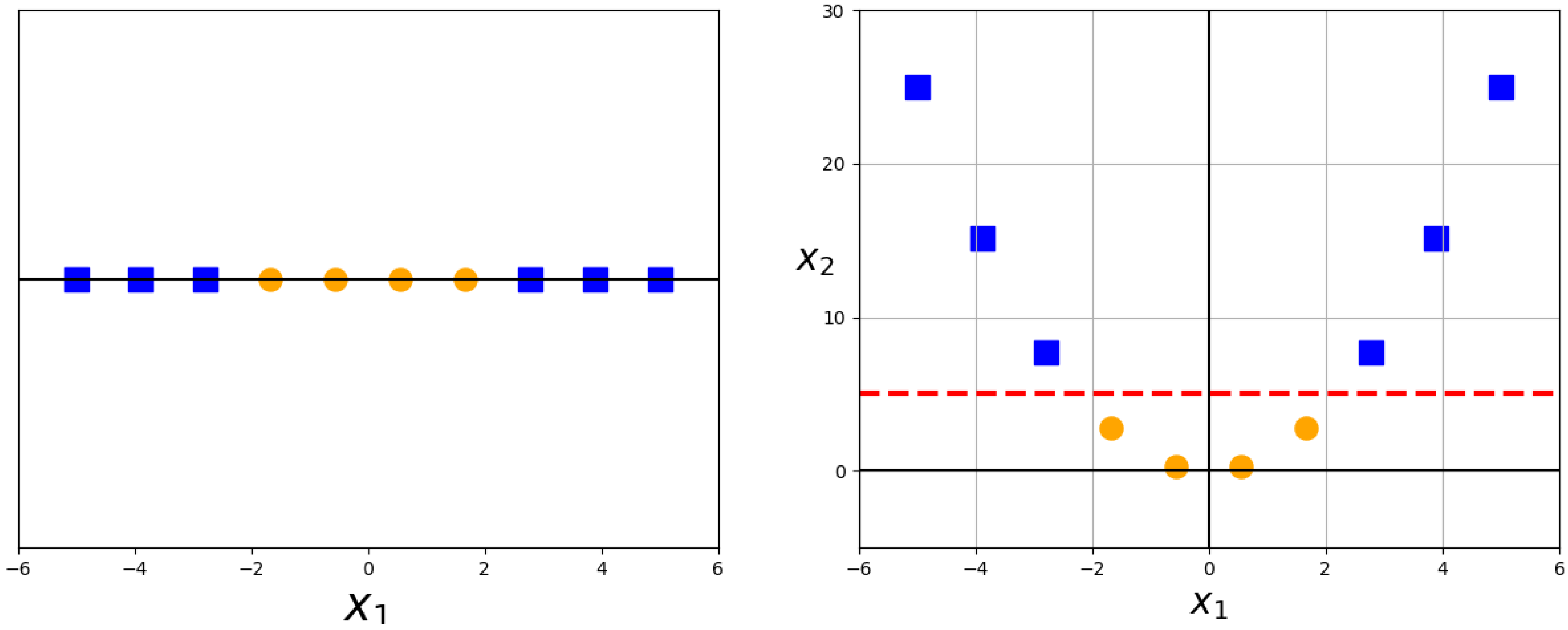
Es una técnica que permite aplicar un modelo lineal (como el SVM) en un espacio de características de mayor dimensión, sin necesidad de transformar explícitamente los datos.

¿QUÉ PROBLEMA RESUELVE?

- Muchos conjuntos de datos **no son linealmente separables** en su espacio original. Pero si los proyectamos a un espacio de mayor dimensión, **sí pueden separarse**.

Ejemplo: un conjunto con forma de anillos concéntricos **no puede separarse con una línea** en 2D, pero sí con un plano en 3D.


2D
Kernel
3D
z
y
x



−6
−4
−2
0
2
4
6
x_1
30
20
10
0
x_2
−6
−4
−2
0
2
4
6
x_1

¿QUÉ HACE EL *KERNEL TRICK*?

Evita calcular explícitamente la transformación:

- En lugar de transformar $\mathbf{x} \to \phi(\mathbf{x})$,
- se usa directamente el **producto interno en el espacio transformado**:

$$K(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$$

¿POR QUÉ ES ÚTIL?

- Reduce el **costo computacional**.
- Permite trabajar en **espacios de dimensión infinita** (como en el caso del RBF).
- Permite que SVM y otros métodos lineales **se vuelvan no lineales sin cambiar el algoritmo**.