

¿Qué es un *kernel* en Machine Learning?

En el ámbito del aprendizaje automático, los núcleos desempeñan un papel fundamental, especialmente en algoritmos diseñados para tareas de clasificación y regresión, como las Máquinas de Soporte Vectorial (SVMs). La función de núcleo es el corazón de estos algoritmos, hábil en simplificar la complejidad inherente en los datos. Transforma relaciones no lineales en un formato lineal, haciéndolas accesibles para algoritmos que tradicionalmente solo manejan datos lineales.

¿Qué es un *kernel* en Machine Learning?

Esta transformación es importante para permitir que las SVMs desentrañen y comprendan patrones y relaciones complejas. Los núcleos logran esto sin la intensidad computacional de mapear datos a dimensiones más altas de manera explícita. Su eficiencia y efectividad en revelar patrones ocultos los convierten en una piedra angular en el aprendizaje automático moderno.

A medida que exploramos más los núcleos, descubrimos su importancia en mejorar el rendimiento y la aplicabilidad de las SVMs en diversos escenarios.

¿Qué es un *kernel* en Machine Learning?

El concepto de un núcleo en el aprendizaje automático ofrece una manera convincente e intuitiva de entender esta poderosa herramienta utilizada en las Máquinas de Soporte Vectorial (SVMs). En su nivel más fundamental, un núcleo es una función relativamente sencilla que opera sobre dos vectores del espacio de entrada, comúnmente referido como el espacio X . El papel principal de esta función es devolver un valor escalar, pero el aspecto fascinante de este proceso radica en lo que este escalar representa y cómo se calcula.

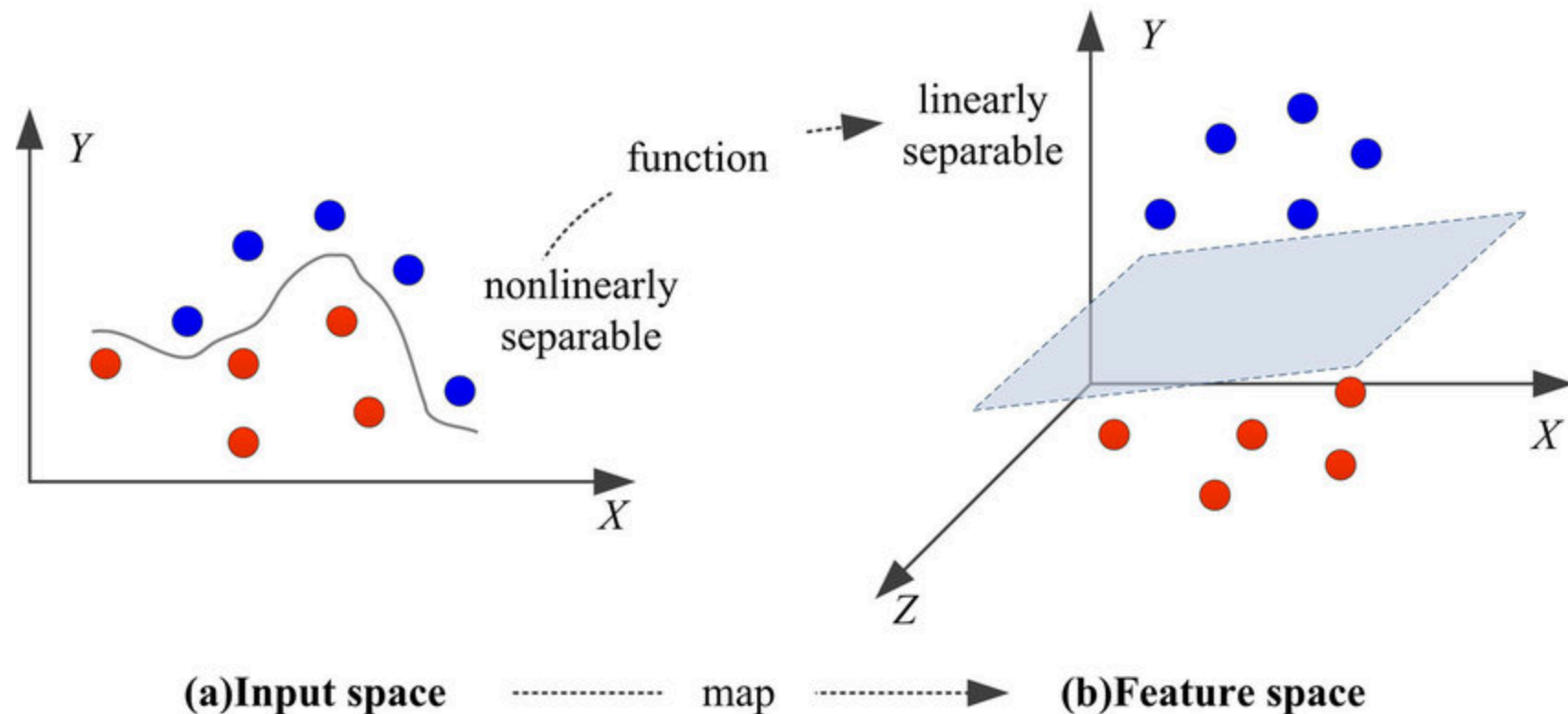
¿Qué es un *kernel* en Machine Learning?

Este escalar es, en esencia, el producto punto de los dos vectores de entrada. Sin embargo, no se calcula en el espacio original de estos vectores. En cambio, es como si este producto punto se calculase en un espacio de dimensiones mucho más altas, conocido como el espacio Z . Aquí es donde reside el verdadero poder y elegancia del núcleo. Logra transmitir cuán cercanos o similares son estos dos vectores en el espacio Z sin la sobrecarga computacional de mapear los vectores a este espacio de dimensiones más altas y calcular su producto punto allí.

¿Qué es un *kernel* en Machine Learning?

El núcleo, por lo tanto, sirve como una especie de guardián del espacio Z . Permite obtener la información necesaria sobre los vectores en este espacio más complejo sin tener que acceder a dicho espacio directamente. Este enfoque es particularmente útil en las SVMs, donde comprender la relación y posición de los vectores en un espacio de mayor dimensión es crucial para las tareas de clasificación.

La razón por la que se llama *trick* (truco) es porque ingeniosamente evita la tarea computacionalmente intensiva de mapear puntos de datos en un espacio de mayor dimensionalidad, lo cual a menudo es necesario para realizar clasificaciones complejas y no lineales.



¿Cómo seleccionar un *kernel*?

Elegir el kernel adecuado para una tarea de aprendizaje automático, como en las Máquinas de Vectores de Soporte (SVMs), es una decisión crítica que puede impactar significativamente el rendimiento del modelo. El proceso de selección implica comprender tanto la naturaleza de los datos como los requisitos específicos de la tarea en cuestión.

¿Cómo seleccionar un *kernel*?

En primer lugar, es importante considerar la distribución y estructura de los datos. Si los datos son linealmente separables, un kernel lineal puede ser suficiente. Sin embargo, para datos más complejos y no lineales, un kernel polinomial o de función de base radial (RBF) podría ser más apropiado.

El kernel polinomial, por ejemplo, es efectivo para conjuntos de datos donde la relación entre variables no es meramente lineal, sino que involucra interacciones de mayor grado. Por otro lado, el kernel RBF, a menudo una opción preferida, es particularmente útil para conjuntos de datos donde el límite de decisión no está claro y los puntos de datos forman una estructura más dispersa.

¿Cómo seleccionar un *kernel*?

Otro aspecto crucial es el ajuste de los parámetros del kernel, que puede influir drásticamente en la precisión del modelo. Por ejemplo, en el kernel RBF, el parámetro gamma define hasta qué punto llega la influencia de un solo ejemplo de entrenamiento, con valores bajos que significan lejos y valores altos que significan cerca. El ajuste correcto de estos parámetros a menudo requiere experimentación iterativa y validación cruzada para evitar el sobreajuste y el subajuste.

En escenarios prácticos, también es aconsejable considerar la eficiencia computacional. Algunos kernels pueden llevar a una convergencia más rápida y menos sobrecarga computacional, lo cual es esencial en aplicaciones a gran escala o al trabajar con conjuntos de datos vastos.

Maquina de vector de soportes

Una Máquina de Vectores de Soporte (Support Vector Machine, SVM) es un modelo de aprendizaje supervisado utilizado tanto para clasificación como para regresión. Sin embargo, su uso principal es en problemas de clasificación. La teoría detrás de las SVM se basa en encontrar el hiperplano que mejor separa un conjunto de datos en diferentes clases.

Maquina de vector de soportes

- **Espacio de características:** en un problema de clasificación, los datos se representan como puntos en un espacio de características de n dimensiones. Cada dimensión representa una característica.
- **Hiperplano:** un hiperplano es un subespacio de dimensión $(n - 1)$ que divide el espacio de características en dos partes. En el caso de SVM, buscamos el hiperplano que mejor separa los datos de diferentes clases.

Maquina de vector de soportes

Separación óptima

- Margen máximo: la SVM busca maximizar el margen entre las clases. El hiperplano óptimo es aquel que tiene la mayor distancia al punto de datos más cercano de cualquier clase.
- Vectores de Soporte: son los puntos de datos que están más cerca del hiperplano y son cruciales para determinar su posición. Estos puntos *soportan* el hiperplano, de ahí el nombre de la técnica.

Maquina de vector de soportes

Separación óptima

- Para conjuntos de datos que son linealmente separables, hay una solución clara donde un hiperplano puede dividir las dos clases sin errores. La SVM encuentra el hiperplano que maximiza el margen entre las dos clases.
- En muchos casos, los datos no son linealmente separables. Para manejar estos casos, las SVM utilizan el truco del kernel.

Maquina de vector de soportes

Para un conjunto de datos (x_i, y_i) donde x_i es el vector de características y y_i es la etiqueta de la clase ± 1 , la SVM resuelve el siguiente problema de optimización:

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

sujeto a:

$$y_i(w \cdot x_i + b) \geq 1, \quad \forall i$$

donde w es el vector de pesos y b es el sesgo. El objetivo es maximizar el margen $\frac{2}{\|w\|}$.

Teorema de Mercer

Sea $K(x, y)$ una función continua, simétrica y positiva semi-definida sobre un espacio compacto $X \times X$. Entonces, existe una serie ortonormal de funciones $\phi_i(x)$ y una serie de valores reales no negativos λ_i tales que:

$$K(x, y) = \sum_{i=1}^{\infty} \lambda_i \phi_i(x) \phi_i(y)$$

donde la serie converge absolutamente y uniformemente.