SVM: Máquinas de Soporte Vectorial

Métodos Basados en Kernel /

Funciones de Base Radial (RBF)

Inteligencia Artificial y Aprendizaje Automático

Tecnológico de Monterrey

Luis Eduardo Falcón Morales

Producto Interior

Consideremos un espacio vectorial V.

Consideremos una función escalar $\langle \vec{u}, \vec{w} \rangle : V \times V \to \mathbb{R}$ con las siguientes propiedades:

- **Linealidad**: $\langle a\vec{u} + b\vec{w}, \vec{v} \rangle = a\langle \vec{u}, \vec{v} \rangle + b\langle \vec{w}\ \vec{v} \rangle$
- **Simétrica**: $\langle \vec{u}, \vec{w} \rangle = \langle \vec{w}, \vec{u} \rangle$
- **Definida Positiva**: $\langle \vec{u}, \vec{u} \rangle \geq 0$, y además $\langle \vec{u}, \vec{u} \rangle = 0$ sí y solo si $\vec{u} = 0$.

donde $\vec{u}, \vec{w}, \vec{v}$ son vectores del espacio vectorial V y a, b son escalares.

A dicha función escalar la llamaremos el **producto interno o interior** de dos vectores.

Al espacio con un producto interno le llamaremos espacio vectorial con producto interno.

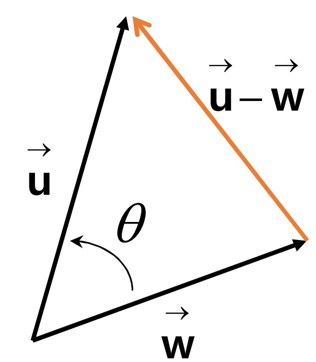
Notaciones equivalentes: $\langle \vec{u}, \vec{w} \rangle = \vec{u} \cdot \vec{w} = \vec{u}^T \vec{w}$

Geometría de un Espacio con Producto Interior

Longitud o norma de un vector: $\|\vec{u}\| = \sqrt{\vec{u} \cdot \vec{u}}$ de donde: $\|\vec{u}\|^2 = \vec{u} \cdot \vec{u}$

Distancia entre vectores: $d = \|\vec{u} - \vec{w}\|$

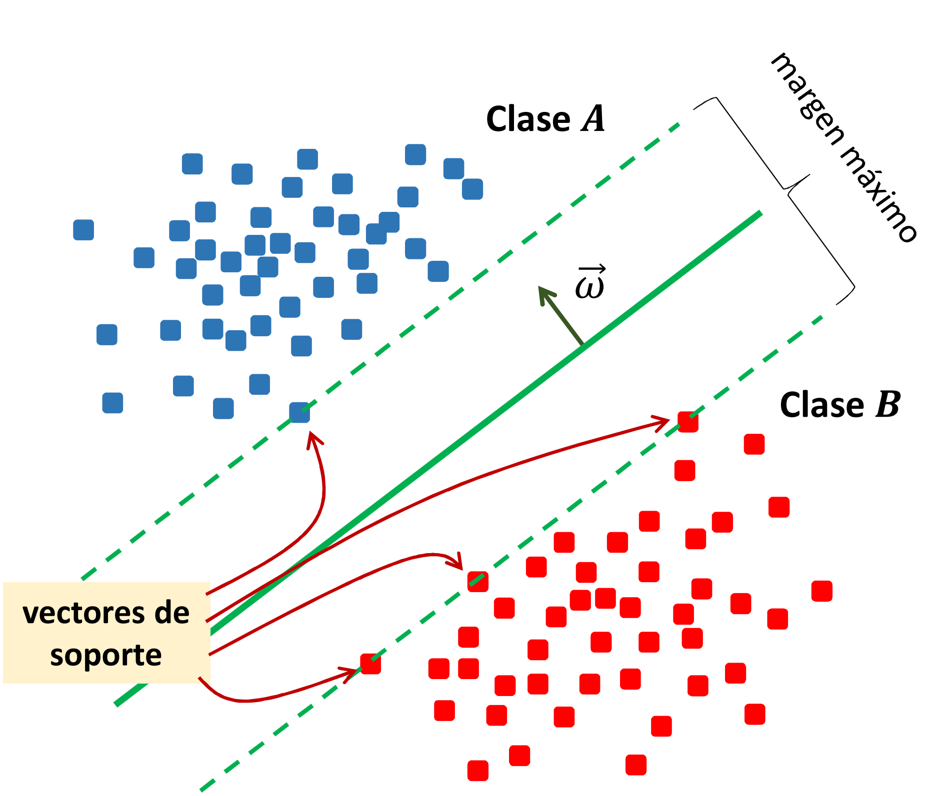
Ángulo entre vectores: $\vec{u} \cdot \vec{w} = \|\vec{u}\|\|\vec{w}\| cos\theta$



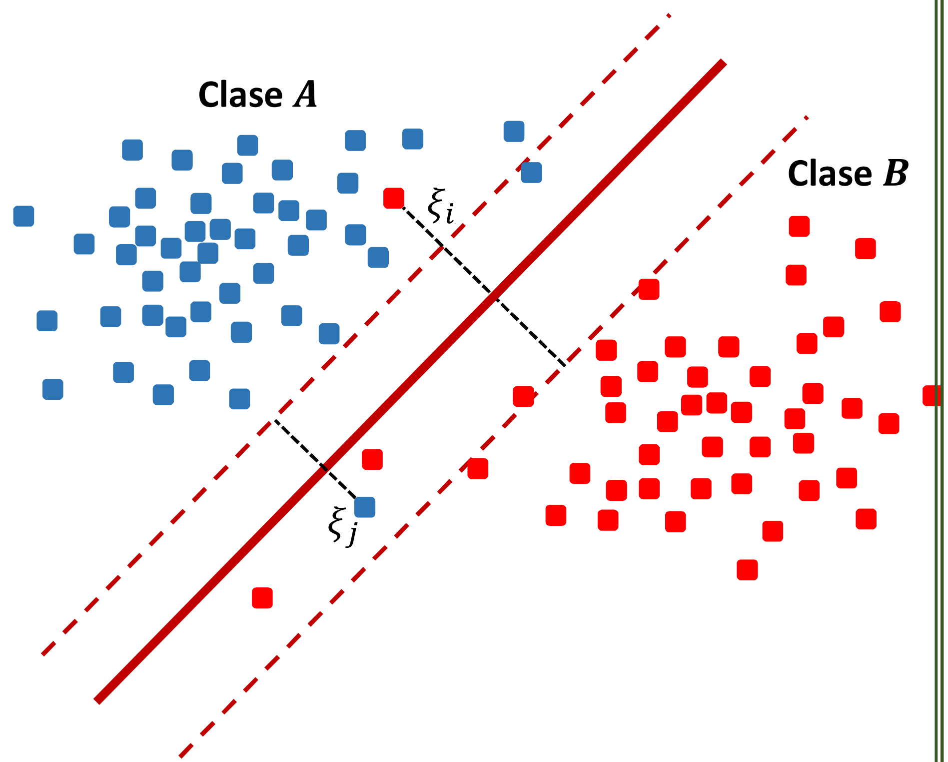
Vectores ortogonales: $\vec{u} \cdot \vec{w} = 0$

Support Vector Machine – SVM

Clases Linealmente Separables:
Hiperplano de Margen Máximo



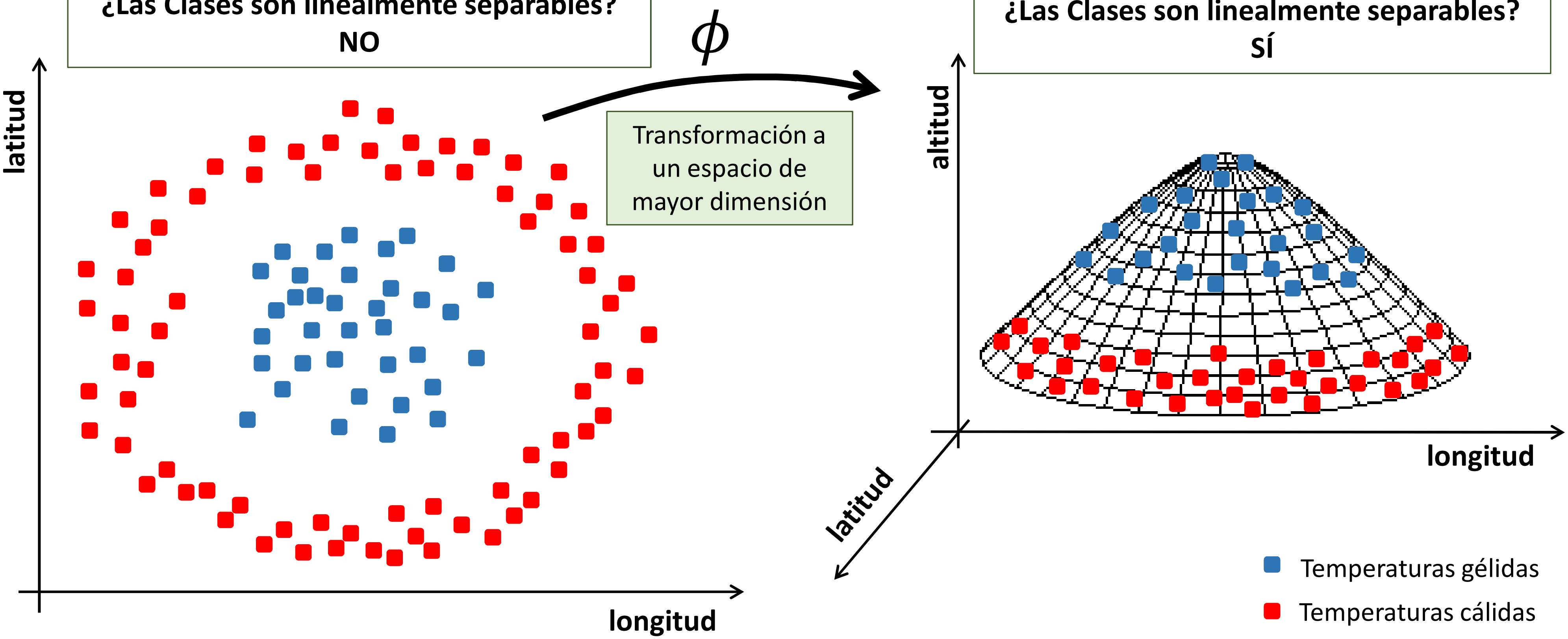
Clases No Linealmente Separables:
Clasificador de Vectores de Soporte

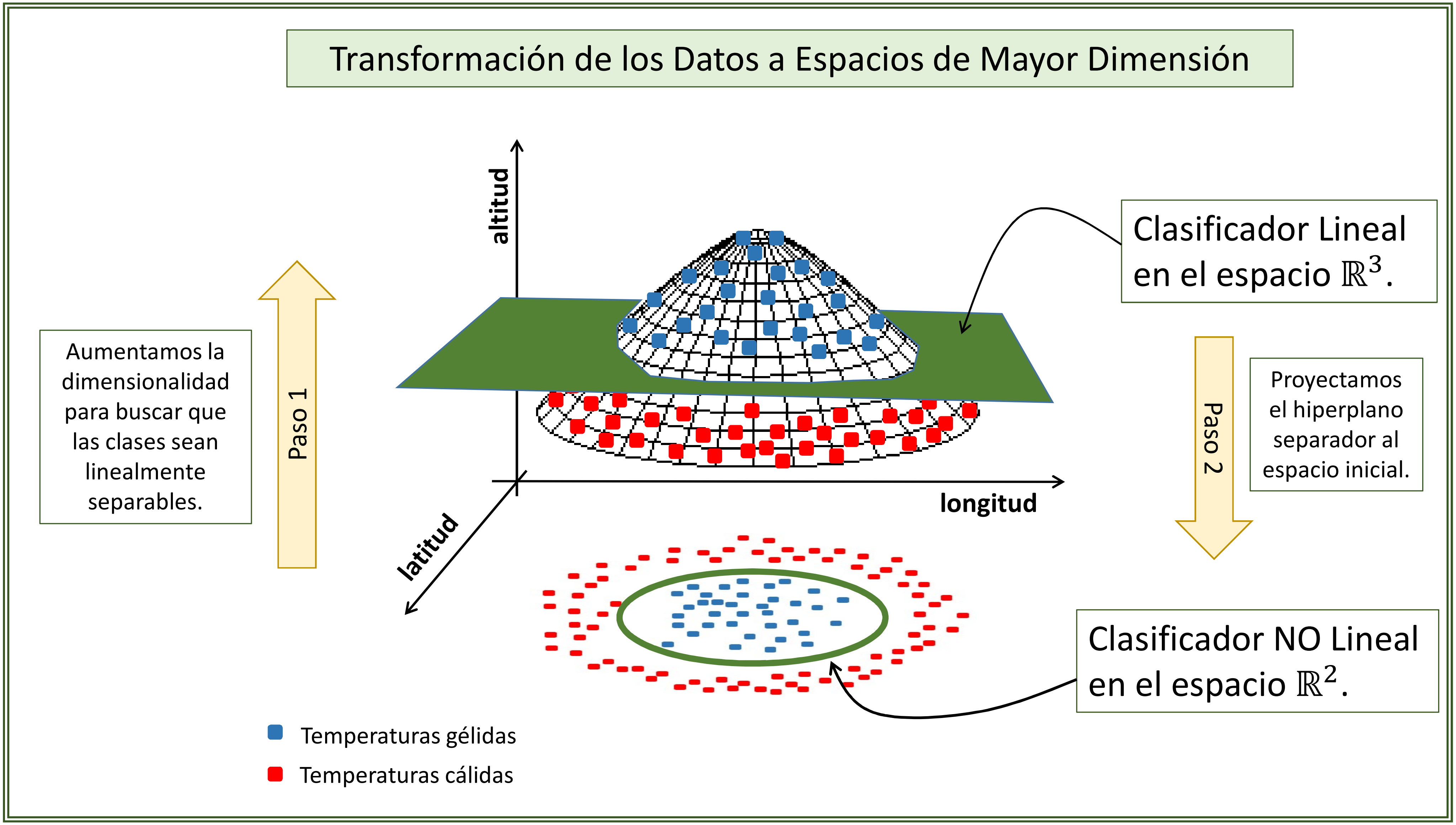


Transformación de los Datos a Espacios de Mayor Dimensión

Espacio original de los datos:
¿Las Clases son linealmente separables?
NO

Espacio de datos transformados:
¿Las Clases son linealmente separables?
SÍ

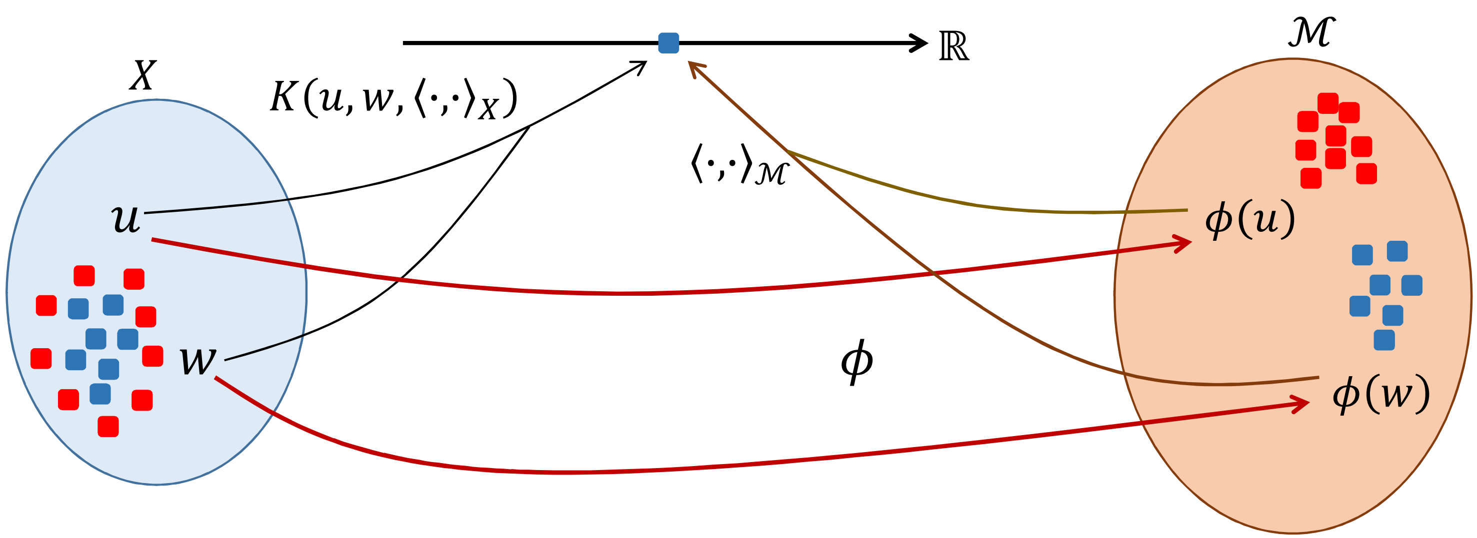



Transformación de los Datos a Espacios de Mayor Dimensión
altitud
longitud
latitud
Aumentamos la dimensionalidad para buscar que las clases sean linealmente separables.
Paso 1
Clasificador Lineal en el espacio $\mathbb{R}^3$.
Paso 2
Proyectamos el hiperplano separador al espacio inicial.
Clasificador NO Lineal en el espacio $\mathbb{R}^2$.
Temperaturas gélidas
Temperaturas cálidas

Ciertas transformaciones $K: X \times X \to \mathbb{R}$, se pueden expresar como el producto interior **en otro espacio** $\mathcal{M}$, es decir, queremos encontrar otra transformación $\phi: X \to \mathcal{M}$, tal que:

$$K(u, w) = \langle \phi\,(u), \phi\,(w) \rangle_{\mathcal{M}}$$

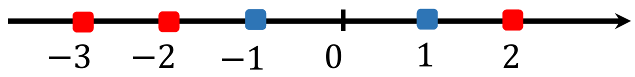
A dicha función/transformación K la llamaremos **kernel.**



En general se buscarán dichas funciones kernel que nos permita calcular el producto interior en $\mathcal{M}$, sin tener que estar calculando explícitamente las imágenes φ, mediante una función sobre el producto interior de X.

Ejemplo 1: Veamos nuevamente un ejemplo geométricamente:

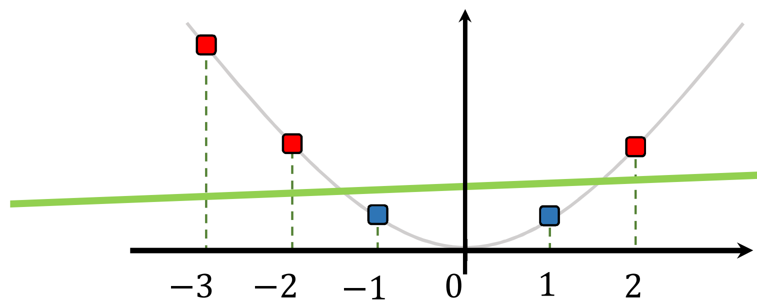
Clases linealmente <u>no</u> separables:
Espacio de dimensión 1



$\phi(x) = (x, x^2)$

Así, en lugar de aplicar SVM para un solo factor, x, ahora lo aplicamos para dos factores: x, x^2.

Clases linealmente separables:
Espacio de dimensión 2



Hiperplano separador:
$y = \omega x + b$

Se busca y se aplica una transformación ϕ de los datos originales a un espacio de mayor dimensión y en este nuevo espacio los nuevos datos ya son linealmente separables.

Recordemos que para resolver un problema bi-clase linealmente separable, el método de SVM busca el hiperplano de margen máximo (MMH) a través del siguiente problema dual de minimización:

$$\max_{\lambda_1,\dots,\lambda_n} \overbrace{\sum_{k=1}^{n} \lambda_k - \frac{1}{2}\left\{\sum_{k=1}^{n}\sum_{j=1}^{n} \lambda_k \lambda_j y_k y_j \vec{x}_k \cdot \vec{x}_j\right\}}^{\mathcal{L}(\vec{\lambda})\text{ : Lagrangiano}}$$

sujeto a las restricciones:

$\lambda_k \geq 0,$

para $k = 1,2,\dots,n$

$$\sum_{k=1}^{n} \lambda_k y_k = 0$$

a partir del conjunto de datos de entrada:

$$\{(x_{k1}, x_{k2}, \dots, x_{km}, y_k)\}_{k=1}^{n}$$

donde $\vec{x}_k = (x_{k1}, x_{k2}, \dots, x_{km})$

$$y_k = \begin{cases} +1 & si \quad \vec{x}_k \in A \\ -1 & si \quad \vec{x}_k \in B \end{cases}$$

Y una vez encontrados los multiplicadores de Lagrange podemos determinar a qué clase pertenecerá un nuevo dato de entrada $\vec{z} = (z_1, z_2, \dots, z_m)$ mediante la expresión:

$$g(\vec{z}) = signo\{\vec{\omega} \cdot \vec{z} + b\} = \begin{cases} \geq 0 & entonces & \vec{z} \in A \\ < 0 & entonces & \vec{z} \in B \end{cases}$$

donde

$$\vec{\omega} = \sum_{k=1}^{n} \lambda_k y_k \vec{x}_k \ , \qquad b = y_k - \vec{\omega} \cdot \vec{x}_k$$

Observa que al encontrar la solución del problema de clasificación, los datos de entrenamiento, los de prueba y los nuevos datos, participan siempre mediante el producto interior de dos de ellos.

Generalización de los Clasificadores Lineales a No Lineales con un Kernel

Así, una forma de generalización del método SVM mediante un kernel K y mediante el problema dual de los coeficientes de Lagrange para el caso bi-clase linealmente separable, sería como sigue:

A partir de un conjunto de datos de entrada:

$$\{(x_{k1}, x_{k2}, \dots, x_{km}, y_k)\}_{k=1}^{n}$$

donde $\vec{x}_k = (x_{k1}, x_{k2}, \dots, x_{km})$

$$y_k = \begin{cases} +1 & si \quad \vec{x}_k \in A \\ -1 & si \quad \vec{x}_k \in B \end{cases}$$

se desea optimizar el Lagrangiano $\mathcal{L}\left(\vec{\lambda}\right)$:

$$\max_{\lambda_1,\dots,\lambda_n} \sum_{k=1}^{n} \lambda_k - \frac{1}{2}\left\{\sum_{k=1}^{n}\sum_{j=1}^{n} \lambda_k \lambda_j y_k y_j \underbrace{\phi(\vec{x}_k)\cdot\phi\left(\vec{x}_j\right)}_{K\left(\vec{x}_k, \vec{x}_j\right)}\right\}$$

sujeto a las restricciones:

$\lambda_k \geq 0,$ para $k = 1,2,\dots,n$

$$\sum_{k=1}^{n} \lambda_k y_k = 0$$

Y una vez resuelto el problema de optimización podemos determinar a qué clase pertenecerá un nuevo dato de entrada $\vec{z} = (z_1, z_2, \dots, z_m)$ mediante cualquiera de los vectores de soporte $\phi\left(\vec{x}_{j_0}\right)$:

$$g(\vec{z}) = signo\left\{\sum_{k=1}^{n} \lambda_k y_k \underbrace{\phi(\vec{x}_k)\cdot\phi(\vec{z})}_{K(\vec{x}_k, \vec{z}\,)} + y_{j_0} - \sum_{k=1}^{n} \lambda_k y_k \underbrace{\phi(\vec{x}_k)\cdot\phi\left(\vec{x}_{j_0}\right)}_{K\left(\vec{x}_k, \vec{x}_{j_0}\right)}\right\}$$

$$= \begin{cases} \geq 0 & entonces \quad \vec{z} \in A \\ < 0 & entonces \quad \vec{z} \in B \end{cases}$$

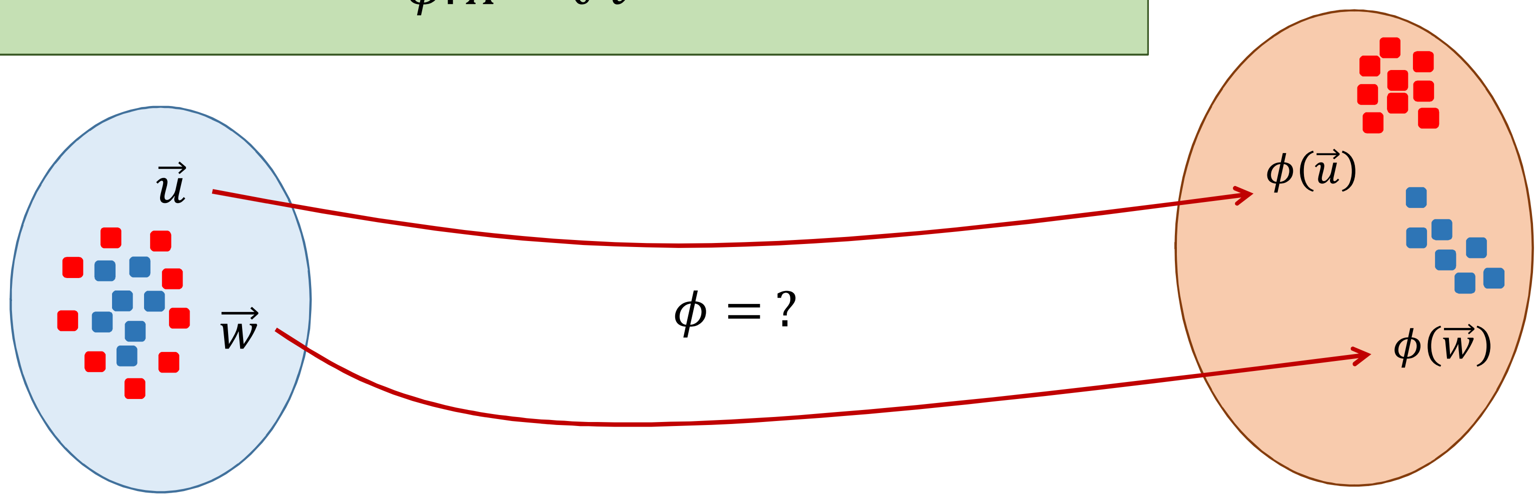
Lo importante de esto es que podremos calcular los productos internos de las imágenes de ϕ, ¡sin calcular o conocer explícitamente las coordenadas de dichas imágenes!

¿A qué espacio y con qué producto interior debería mandar los datos originales para que ahí ya sean linealmente separables?

$$\phi: X \to \mathcal{M}$$

NOTA: por simplicidad denotamos $\vec{u}$ como u.

X
con el producto interior Euclidiano: $\langle u, w \rangle_X$



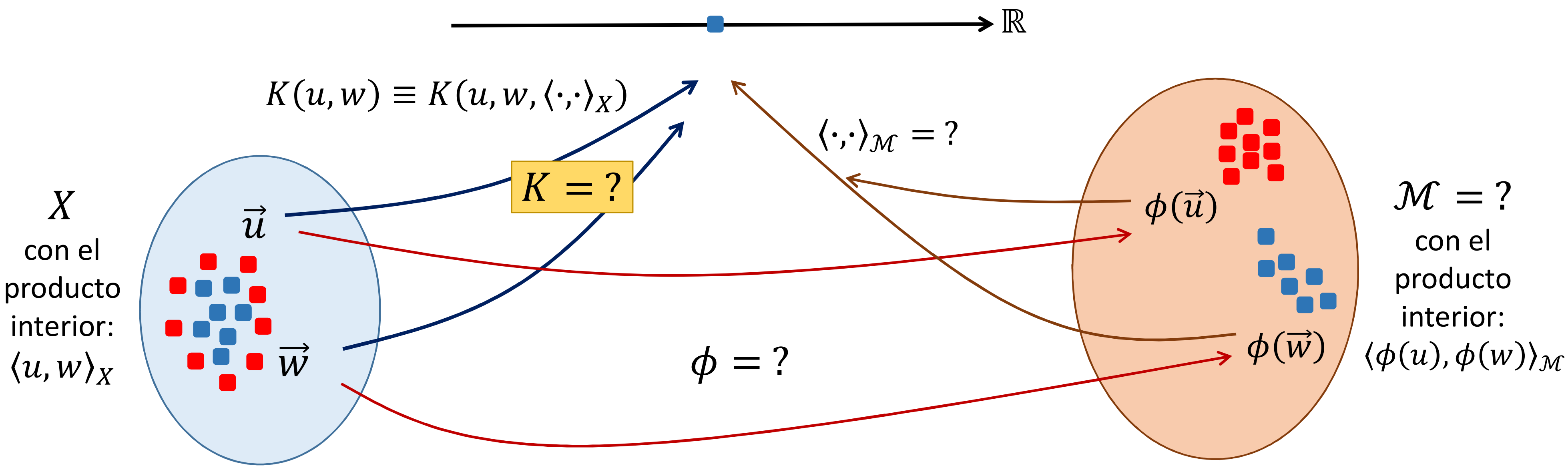
$\mathcal{M} = ?$
con el producto interior: $\langle \phi(u), \phi(w) \rangle_{\mathcal{M}}$

- Se inicia con un conjunto de puntos de clases no linealmente separables en un espacio vectorial X con un producto interior $\langle \cdot,\cdot \rangle_X$.
- Se desea encontrar una transformación ϕ y un espacio vectorial $\mathcal{M}$ con un producto interior $\langle \cdot,\cdot \rangle_{\mathcal{M}}$, tal que ahí los nuevos puntos sean linealmente separables.

¿A qué espacio y con qué producto interior debería mandar los datos originales para que ahí ya sean linealmente separables?

NOTA: por simplicidad denotamos $\vec{u}$ como u.

Kernel trick: $K(u, w, \langle\cdot,\cdot\rangle_X) = \langle\phi\,(u), \phi\,(w)\rangle_{\mathcal{M}}$



- **Kernel trick:** En lugar de buscar la función ϕ y el espacio $\mathcal{M}$, donde $\phi: X \to \mathcal{M}$, encontrar una función/kernel K tal que: $K(u, w, \langle\cdot,\cdot\rangle_X) = \langle\phi\,(u), \phi\,(w)\rangle_{\mathcal{M}}$

Kernel y la Generalización del Clasificador Lineal

$$K(\vec{x}_k, \vec{x}_j) = \phi(\vec{x}_k) \cdot \phi(\vec{x}_j)$$

Actualmente existe una gran variedad de kernels.
De hecho es un área de investigación abierta y actualmente se siguen proponiendo más.
Los siguientes son algunos de los más utilizados:

Kernel Lineal:

$$K(\vec{x}_k, \vec{x}_j) = \vec{x}_k \cdot \vec{x}_j$$

Kernel Polinomial:

$$K(\vec{x}_k, \vec{x}_j) = (\gamma\, \vec{x}_k \cdot \vec{x}_j + \delta)^d$$

Kernel (Sigmoide) Tangente Hiperbólico:

$$K(\vec{x}_k, \vec{x}_j) = tanh(\gamma \vec{x}_k \cdot \vec{x}_j + \delta)$$

Kernel Gaussiano (RBF):

$$K(\vec{x}_k, \vec{x}_j) = e^{-\frac{1}{2\sigma^2}\|\vec{x}_k - \vec{x}_j\|^2}$$

Kernel Exponencial:

$$K(\vec{x}_k, \vec{x}_j) = e^{-\frac{1}{2\sigma^2}\|\vec{x}_k - \vec{x}_j\|}$$

Llamados kernels/Funciones de Base Radial

Se puede considerar: $\gamma = \frac{1}{2\sigma^2}$

Hiperparámetros en SVM

La constante C, entre mayor es el valor, el margen más pequeño y por lo tanto variables de holgura más pequeños.

En cambio, para valores de C pequeños, las variables de holgura serán mayores y por lo tanto el margen será mayor.

El valor de C se puede buscar como el recíproco de la varianza del espacio de características (features).

En los modelos de Kernel **Gaussiano**, **Exponencial** o **Polinomial**, se tiene además el hiperparámetro sigma.

Usualmente dicho parámetro se define como el recíproco de la varianza. Es decir,

$$gamma = \gamma \propto \frac{1}{\sigma^2}$$

Kernel Trick

$$K\left(\vec{x}_k, \vec{x}_j, \langle\cdot,\cdot\rangle_X\right) = \left\langle \phi(\vec{x}_k), \phi\left(\vec{x}_j\right)\right\rangle_{\mathcal{M}}$$

- La transformación ϕ tiene como dominio el espacio de los vectores de entrenamiento X, y como imagen un nuevo espacio $\mathcal{M}$ de mayor dimensión que X. Generalmente permanecen desconocidos tanto este nuevo espacio como su producto interior.
- Formalmente la transformación K define un kernel, es decir, una transformación simétrica y semidefinida positiva.
- Aunque las imágenes de los vectores $\phi(\vec{x}_k)$, $\phi\left(\vec{x}_j\right)$ están en un espacio de dimensión mayor, que inclusive puede ser infinito, la clasificación mediante el kernel/producto interior sigue siendo bastante sencilla de calcular y utilizar.
- Se obtienen fronteras de decisión no-lineales y más complejas, pero sin un alto costo computacional.
- No se requiere encontrar las coordenadas de la imagen de los vectores de entrenamiento $\phi(\vec{x}_k)$ y $\phi\left(\vec{x}_j\right)$, ya que su producto interior $\left\langle \phi(\vec{x}_k), \phi\left(\vec{x}_j\right)\right\rangle_{\mathcal{M}}$ se obtiene mediante el kernel $K\left(\vec{x}_k, \vec{x}_j,, \langle\cdot,\cdot\rangle_X\right)$.