PROMiDAT
IBEROAMERICANO
Programa Iberoamericano de Formación en Minería de Datos

Aprendizaje Supervisado

Máquinas de Soporte Vectorial

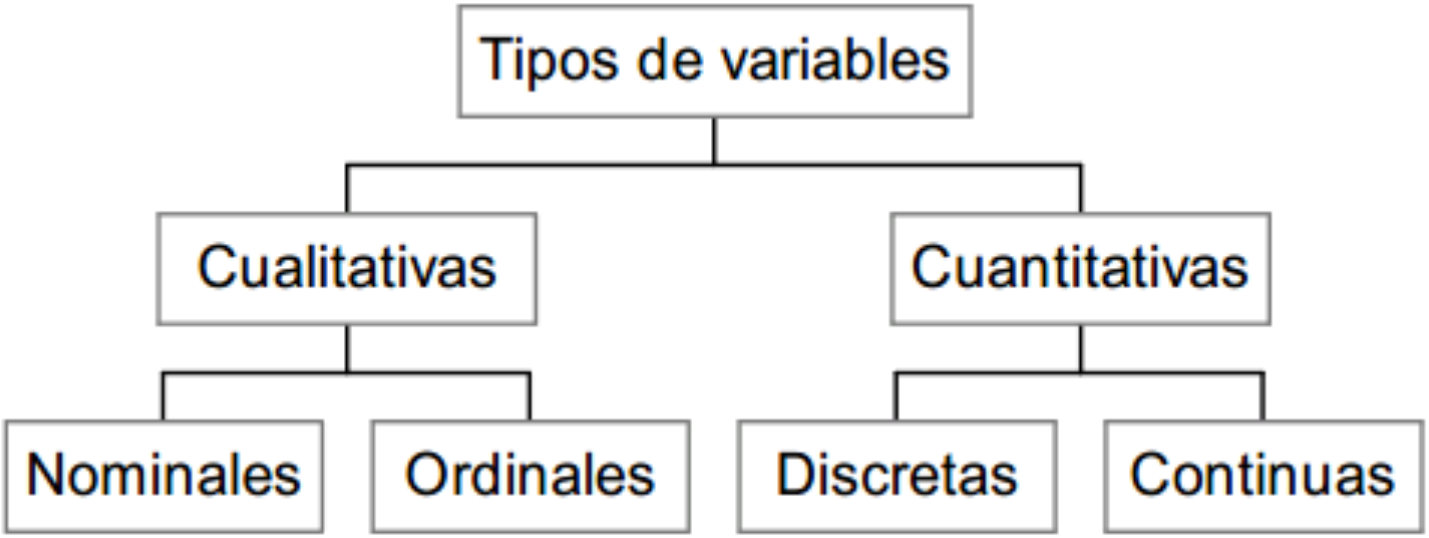
SVM

- Support Vector Machines is arguably the most important & interesting recent discovery in Machine Learning.
- Support vector machines were introduced by Vapnik .

Vladimir Naumovich Vapnik

[Wikipedia] Vladimir Vapnik was born in the Soviet Union. He received his master's degree in mathematics from the Uzbek State University, Samarkand, Uzbek USSR in 1958 and Ph.D in statistics at the Institute of Control Sciences, Moscow in 1964. He worked at this institute from 1961 to 1990 and became Head of the Computer Science Research Department. At the end of 1990, Vladimir Vapnik moved to the USA and joined the Adaptive Systems Research Department at AT&T Bell Labs in Holmdel, New Jersey.

Tipos de Variables



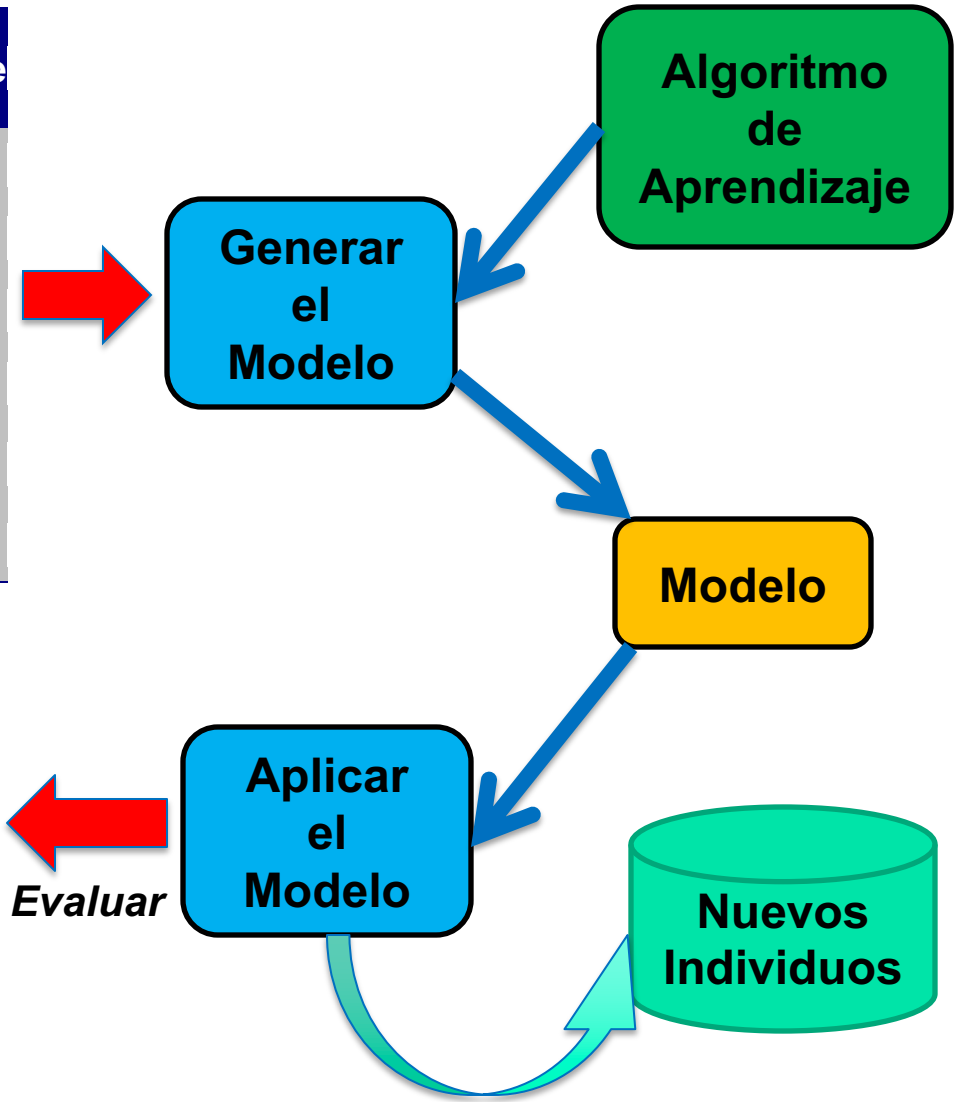
Modelo general de los métodos de Clasificación

Id	Reembolso	Estado Civil	Ingresos Anuales	Fraude
1	Sí	Soltero	125K	No
2	No	Casado	100K	No
3	No	Soltero	70K	No
4	Sí	Casado	120K	No
5	No	Divorciado	95K	Sí
6	No	Casado	60K	No

Tabla de Aprendizaje

Id	Reembolso	Estado Civil	Ingresos Anuales	Fraude
7	No	Soltero	80K	No
8	Si	Casado	100K	No
9	No	Soltero	70K	No

Tabla de Testing



Clasificación: Definición

- Dada una colección de registros (conjunto de entrenamiento) cada registro contiene un conjunto de variables (atributos) denominado x, con un variable (atributo) adicional que es la clase denominada y.

- El objetivo de la ***clasificación*** es encontrar un modelo (una función o algortimo) para predecir la clase a la que pertenecería cada registro, esta asignación una clase se debe hacer con la mayor precisión posible.

- Un conjunto de prueba (tabla de testing) se utiliza para determinar la precisión del modelo. Por lo general, el conjunto de datos dado se divide en dos conjuntos al azar de el de entrenamiento y el de prueba.

Definición de Clasificación

- Dada una base de datos $D = \{t_1, t_2, \dots, t_n\}$ de tuplas o registros (individuos) y un conjunto de clases $C = \{C_1, C_2, \dots, C_m\}$, el **problema de la clasificación** es encontrar una función $f: D \rightarrow C$ tal que cada t_i es asignada una clase C_j.

- $f: D \rightarrow C$ podría ser una Red Neuronal, un Árbol de Decisión, un modelo basado en Análisis Discriminante, o una Red Beyesiana.

Ejemplo: Créditos en un Banco

Tabla de Aprendizaje

Variable Discriminante

OLDEMARRR.DMEx...ditoViviendaPeq

	Id	MontoCredito	IngresoNeto	CoeficienteCre...	MontoCuota	GradoAcademico	BuenPagador
▶	1	2	4	3	1	4	1
	2	2	3	2	1	4	1
	3	4	1	1	4	2	2
	4	1	4	3	1	4	1
	5	3	3	1	3	2	2
	6	3	4	3	1	4	1
	7	4	2	1	3	2	2
	8	4	1	3	3	2	2
	9	3	4	3	1	3	1
	10	1	3	2	2	4	1
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Con la Tabla de Aprendizaje se entrena (aprende) el modelo matemático de predicción, es decir, a partir de esta tabla se calcula la función ***f*** de la definición anterior.

Ejemplo: Créditos en un Banco

Tabla de Testing

Variable Discriminante

OLDEMARRR.DME...iviendaPeqPRED | OLDEMARRR.DMEx...ditoViviendaPeq

	Id	MontoCredito	IngresoNeto	CoeficienteCre...	MontoCuota	GradoAcademico	BuenPagador
▶	11	3	3	3	3	1	2
	12	2	2	2	2	1	1
	13	2	2	3	2	1	1
	14	1	3	4	3	2	2
	15	1	2	4	2	1	1
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

- Con la Tabla de Testing se valida el modelo matemático de predicción, es decir, se verifica que los resultados en individuos que no participaron en la construcción del modelo es bueno o aceptable.
- Algunas veces, sobre todo cuando hay pocos datos, se utiliza la Tabla de Aprendizaje también como de Tabla Testing.

Ejemplo: Créditos en un Banco

Nuevos Individuos

Variable Discriminante

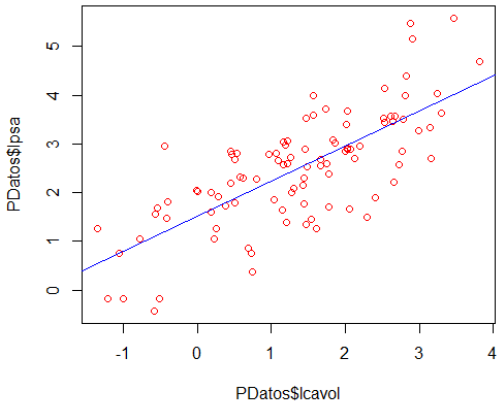
OLDEMARRR.DMEx ...editoViviendaNI

	Id	MontoCredito	IngresoNeto	CoeficienteCre...	MontoCuota	GradoAcademico	BuenPagador
	100	4	4	2	2	3	?
	101	1	4	3	2	4	?
	102	3	2	3	4	2	?
▶*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

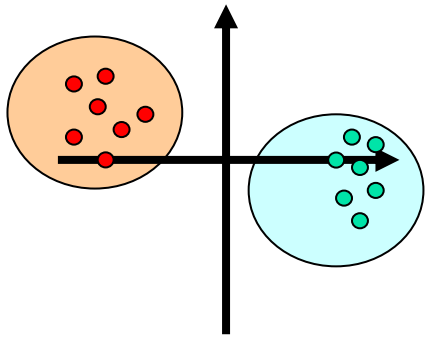
Con la Tabla de Nuevos Individuos se predice si estos serán o no buenos pagadores.

Regresión vs Clasificación

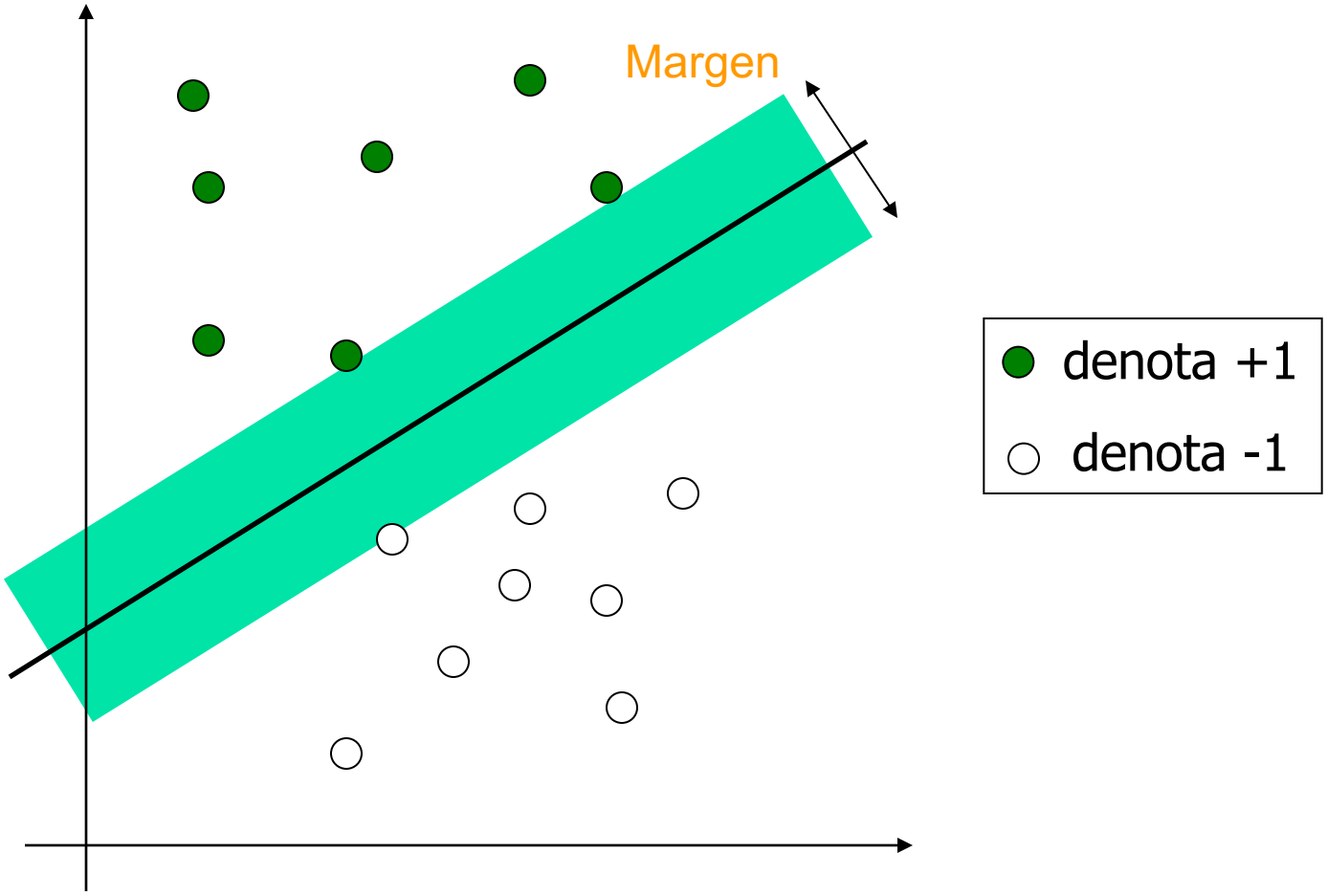
- Regresión:
  - La variable a predecir es cuantitativa
  - Por ejemplo predecir el salario de una persona



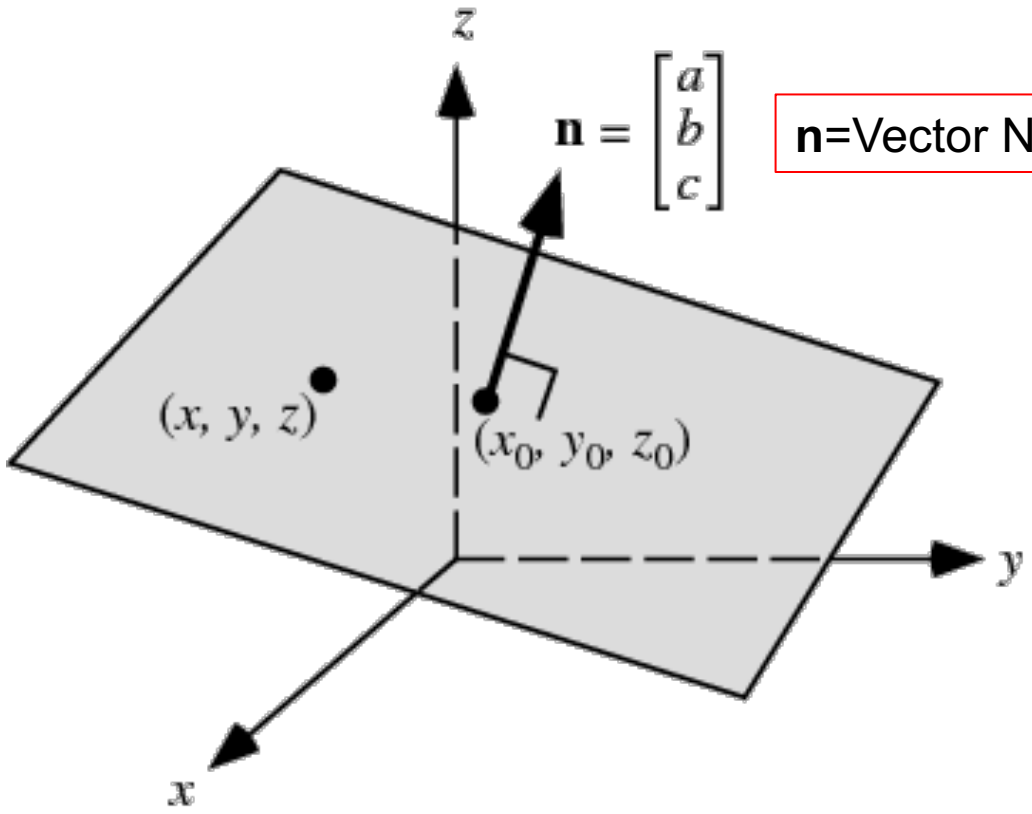
- Clasificación
  - La variable a predecir es cualitativa
  - Por ejemplo predecir si una transacción es fraude o no



Idea: Las Máquinas de Soporte Vectorial (Support Vector Machines) tratan de encontrar el hiperplano que separe a las clases con el mayor "margen" posible.

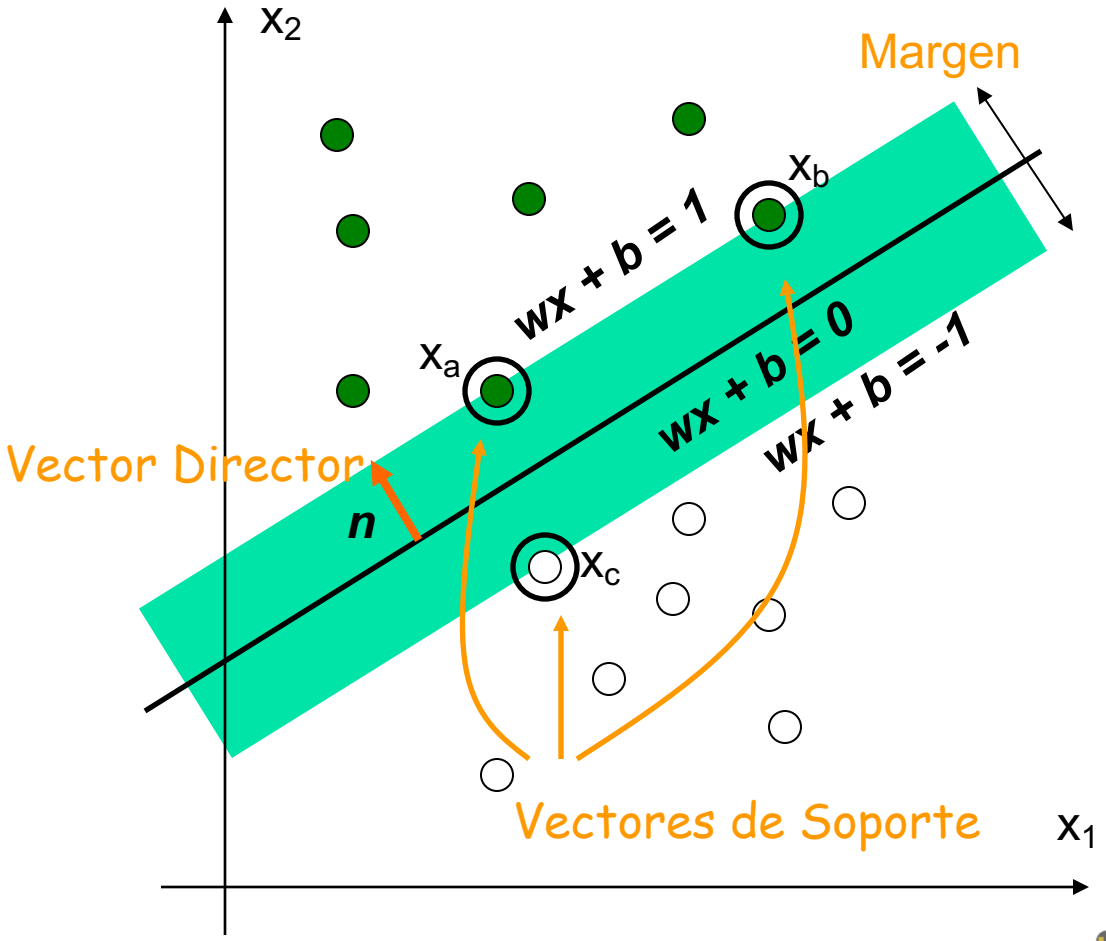


Como casos particulares en el plano el hiperplano es una recta y en el espacio es un plano como se muestra en la figura.



n=Vector Normal

¿Por qué se denominan Máquinas de Soporte Vectorial (Support Vector Machines)?

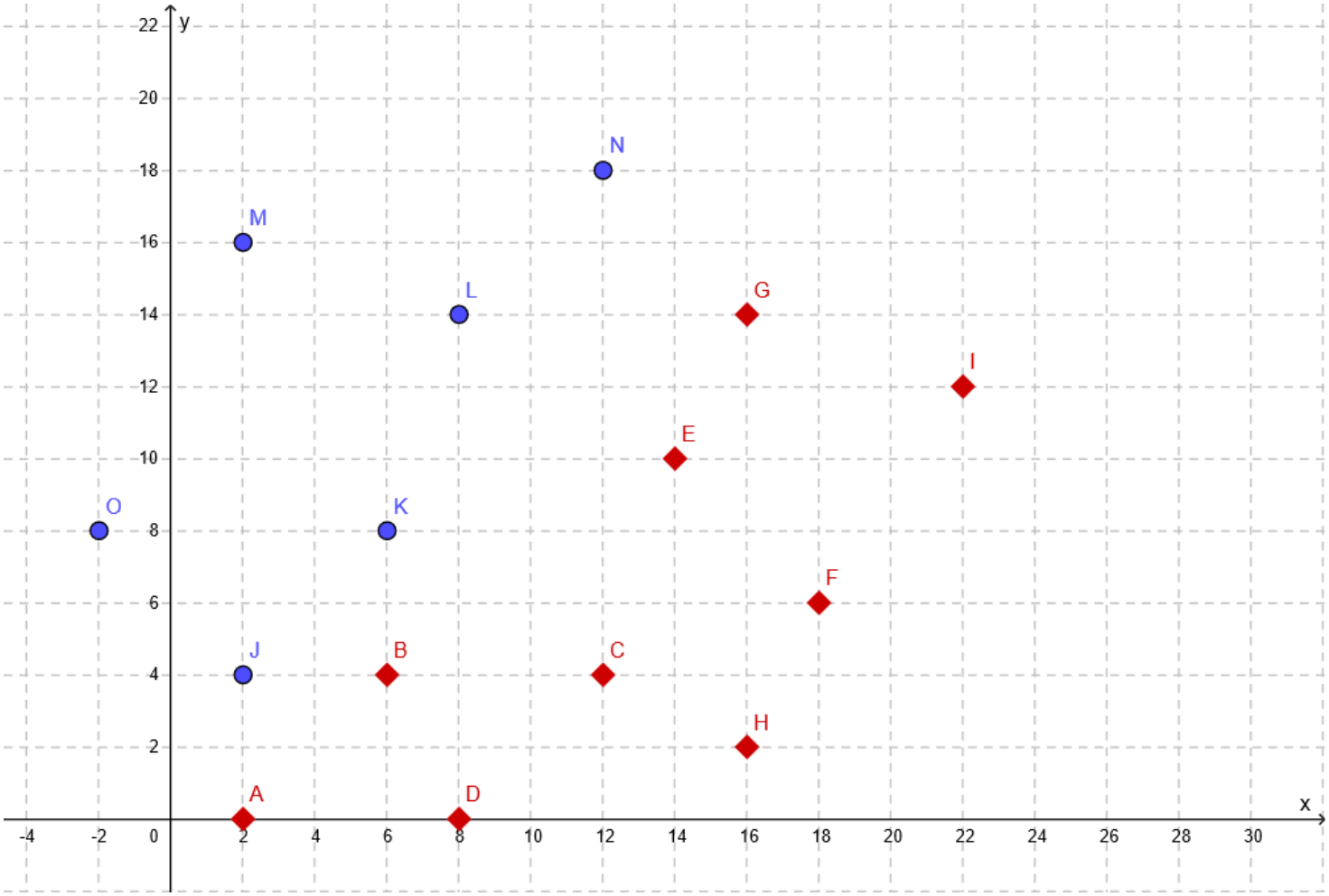


Ejemplo en dos dimensiones

Considere los siguientes datos:

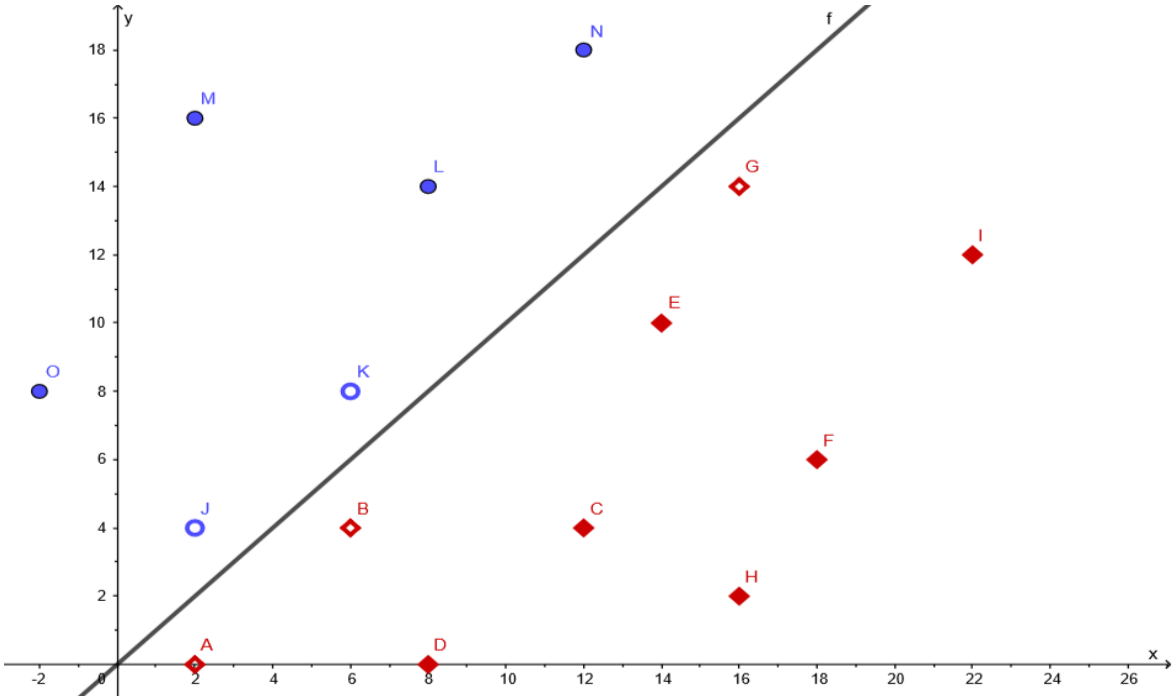
X1	*X2*	*Y*
2	0	Rojo
6	4	Rojo
12	4	Rojo
8	0	Rojo
14	10	Rojo
18	6	Rojo
16	14	Rojo
16	2	Rojo
22	12	Rojo
2	4	Azul
6	8	Azul
8	14	Azul
2	16	Azul
12	18	Azul
-2	8	Azul

Puntos en el plano



Hiperplano de separación

En este caso, haciendo uso de álgebra básica podemos encontrar el hiperplano de separación (en el caso de dos dimensiones una recta) sin necesidad de derivar, usando el punto medio entre los puntos J y A que es P1=(2,2) y el punto medio entre K y B que es P2=(6,6) con la fórmula usual de la pendiente y la recta *y=mx+b*, se puede concluir que es: *y=x o f(x)=x*.



Procedimiento para encontrar f(x): Puntos P1=(2,2) y P2=(6,6)

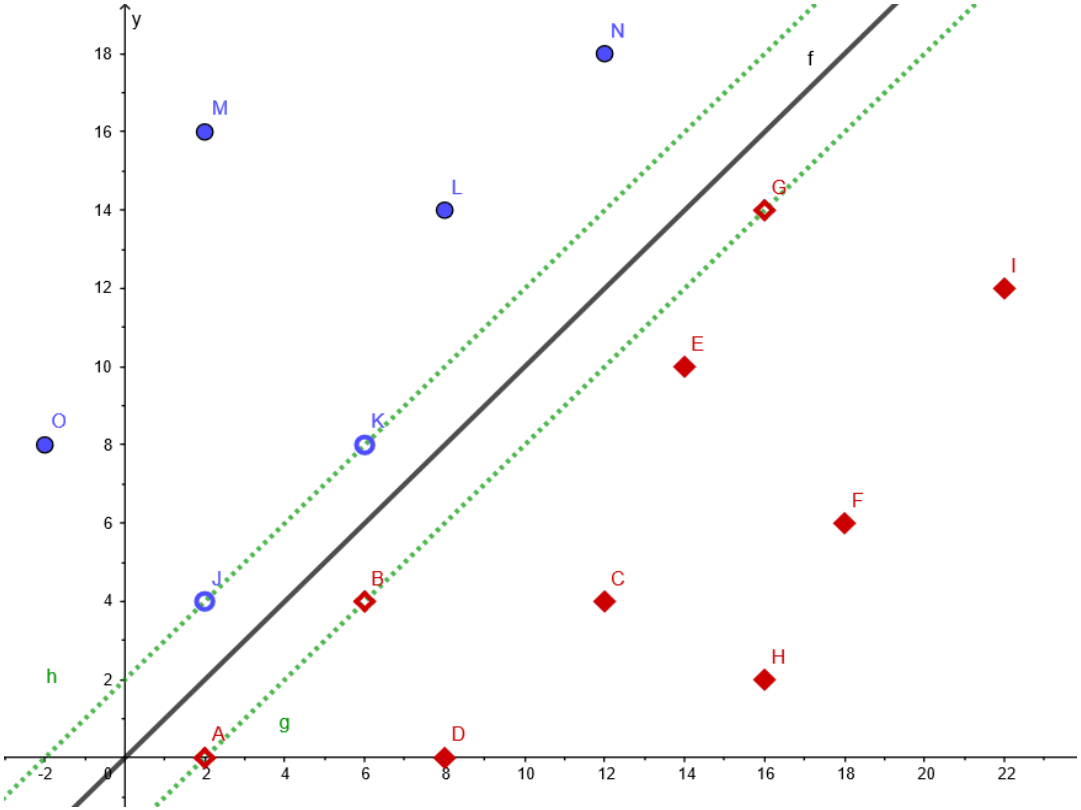
Pendiente m= (Y2-Y1)/(X2-X1) =(6-2)/(6-2)=1

*Usando P1=(2,2) b=y-mx=2-(1*2)=0*

Entonces la recta de separación es f(x)=x

Vectores de soporte y margen

Los vectores de soporte son los puntos *J, K, A, B, y G*, y los denotamos con el centro en blanco, las rectas que delimitan el margen vienen dadas por *h(x)=x+2 y g(x)=x-2.*



Procedimiento para encontrar h(x): Puntos
J=(2,4) y K=(6,8)

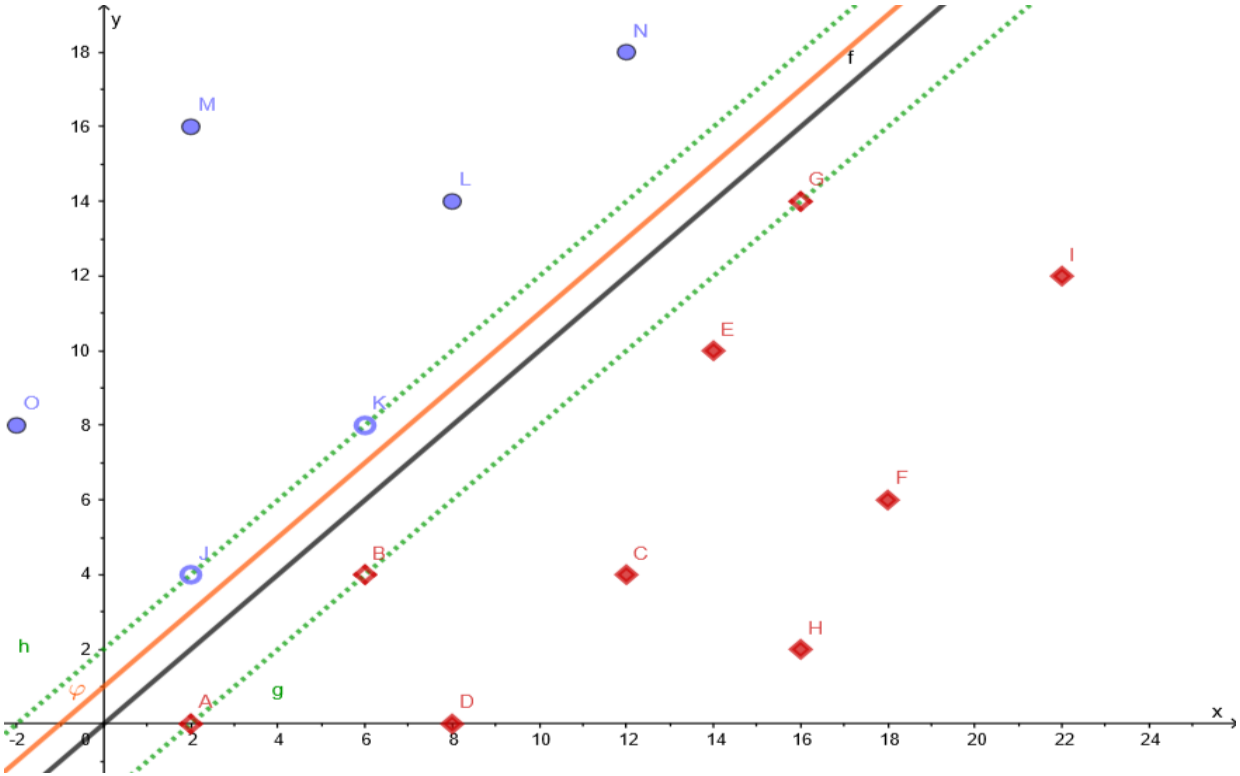
Pendiente
m= (Y2-Y1)/(X2-X1)
=(4-8)/(2-6)=1

b=y-mx
*=4-1*2=2*

g(x) se calcula análogamente

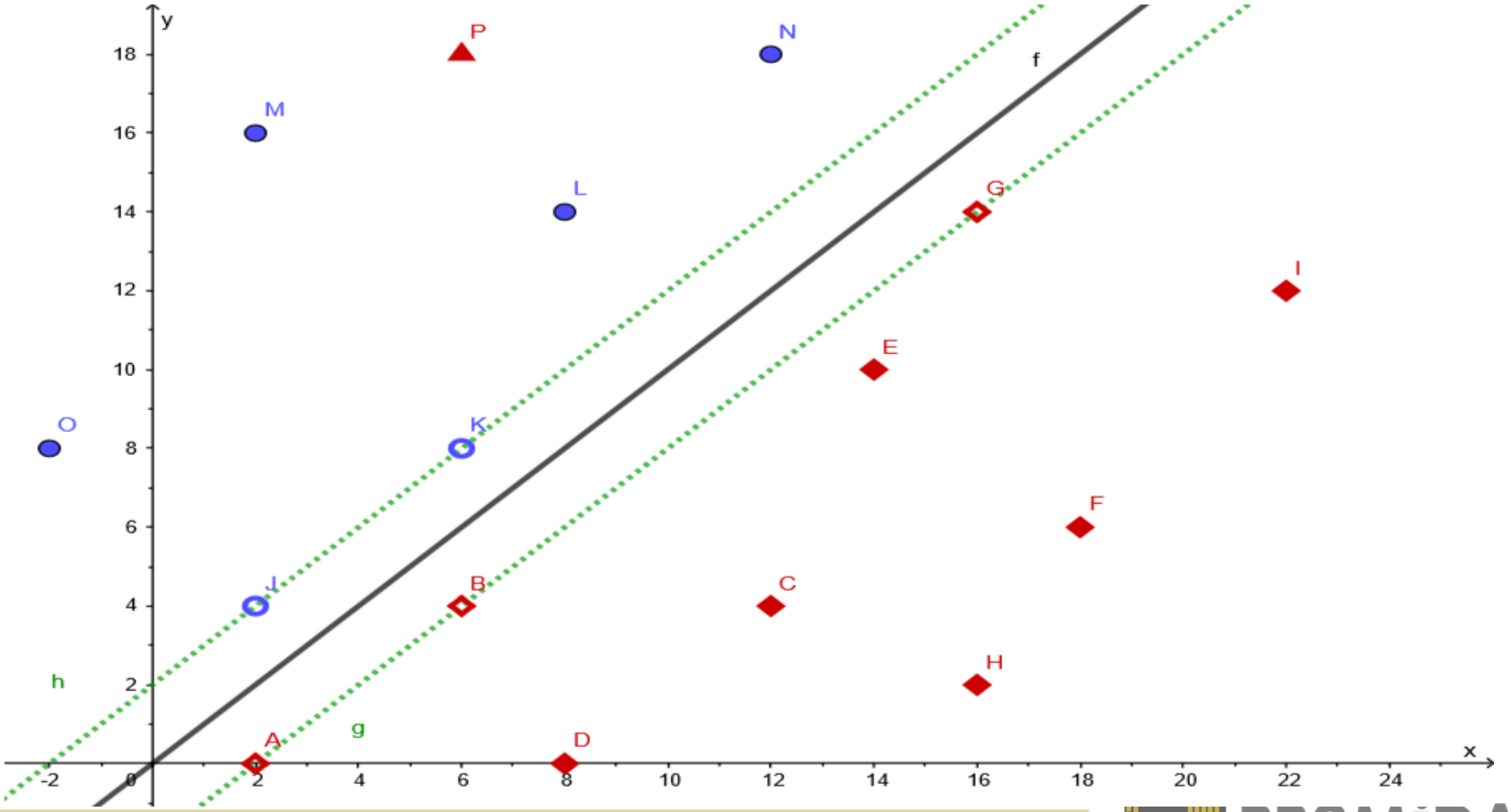
Hiperplano no óptimo

Ahora un hiperplano que separa pero no es óptimo con respecto al mejor margen es la recta naranja, con ecuación $\varphi(x)=x+1$ la cual aunque separa, no tiene margen máximo.



Problema no separable

Si se añade el punto rojo P=(6,18) este ya no es un problema linealmente separable.



Predicción de un nuevo individuo

Si tenemos un individuo nuevo, por ejemplo *Q=(a,b),* lo que hacemos para clasificarlo es seguir la siguiente regla:

- Azul si f(a) < b
- Rojo si f(a) ≥ b

Es decir que si esta por arriba del hiperplano de separación se clasifica como Azul y si esta por debajo del mismo se clasifica como Rojo.

Predicción de un nuevo individuo

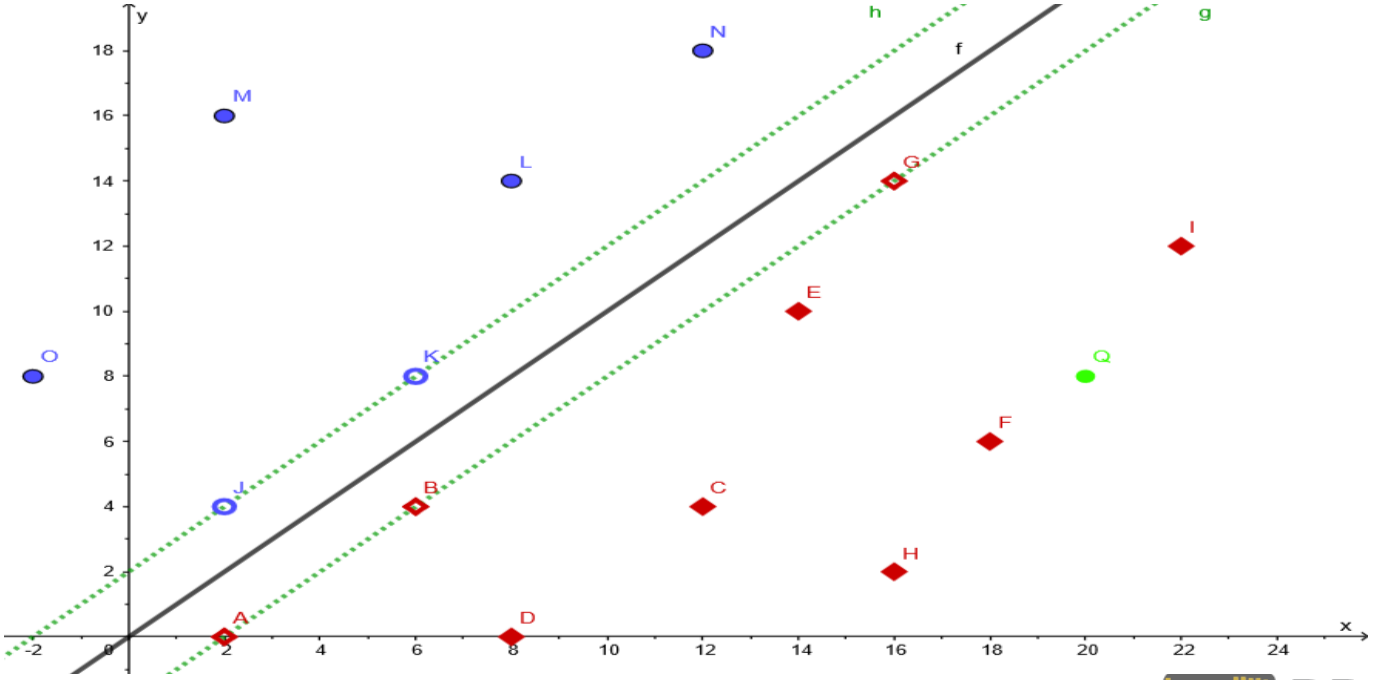
Por ejemplo si tenemos el punto *Q=(20,8)* entonces como *f(x)=x,* tenemos que:

f(20)=20, como 20 < 8 es falso

por lo tanto no es Azul.

f(20)=20, como 20 ≥ 8 es verdadero

Por lo tanto se clasifica como Rojo.



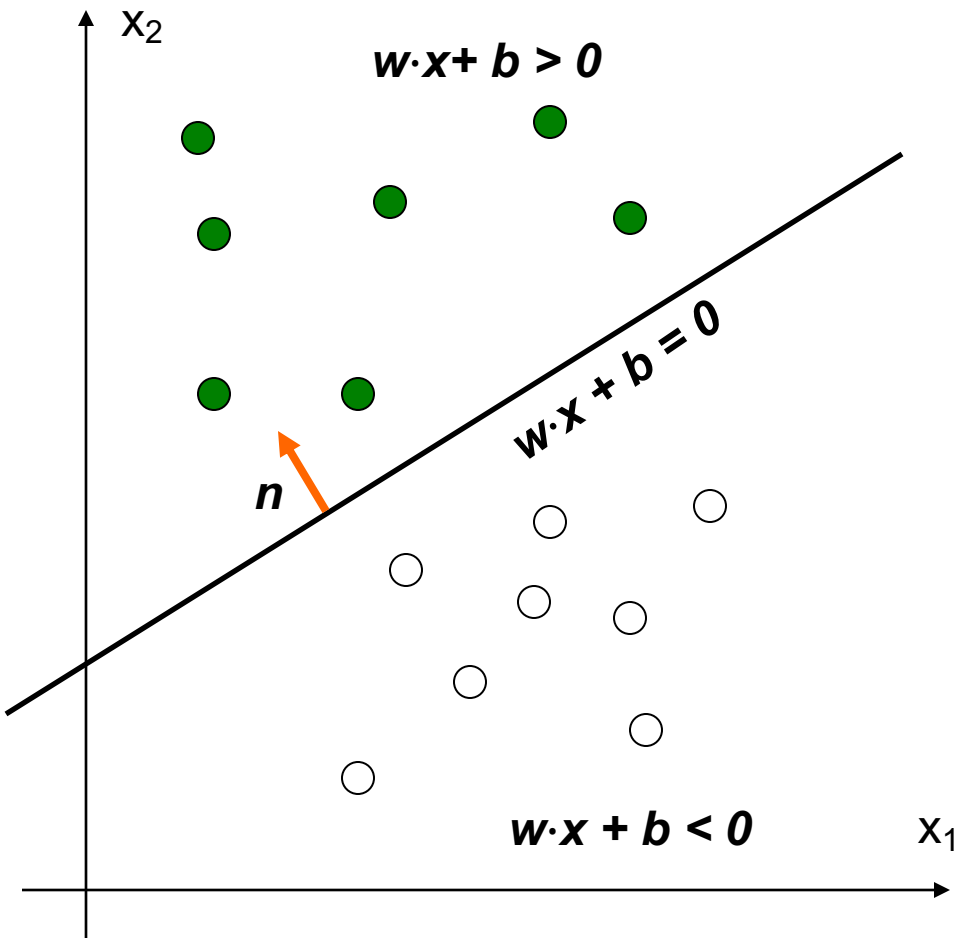
Función discriminante lineal

- g(x) es una función lineal:

$$g(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$$

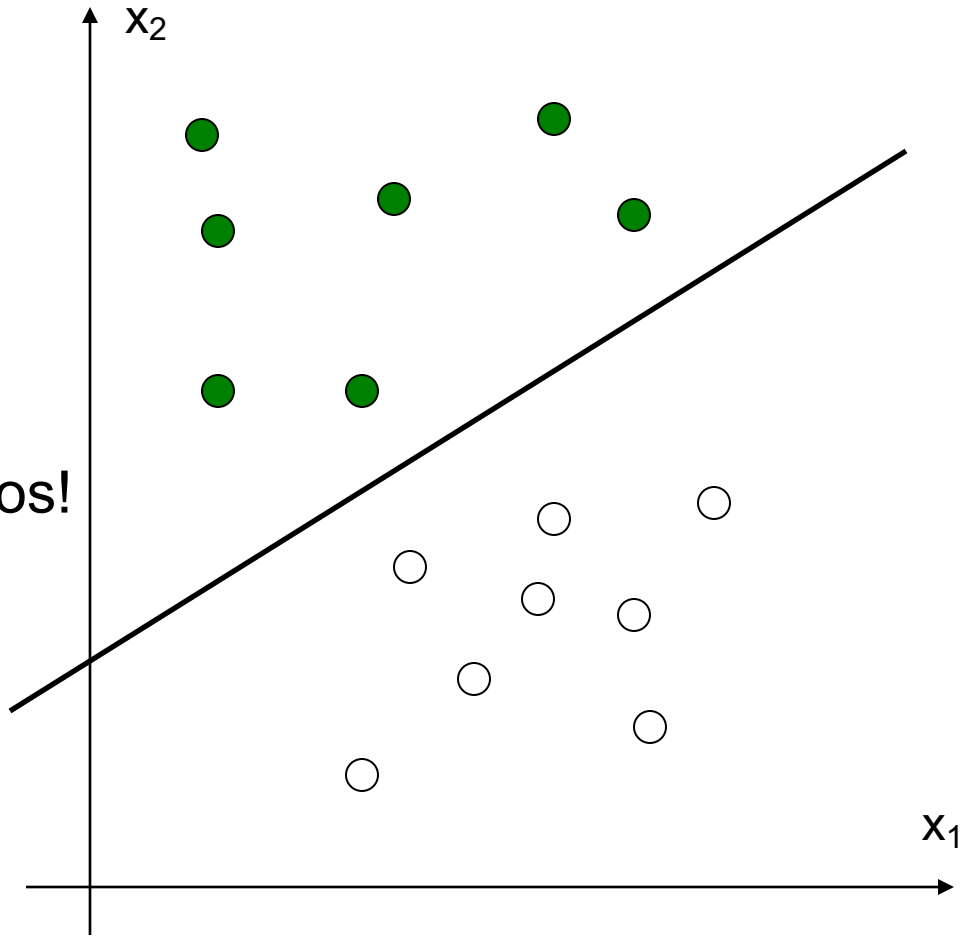
- Se busca un hiperplano en el espacio de las variables. En general **w** y **x** son vectores.
- ***n*** es el vector normal del hiperplano:

$$\mathbf{n} = \frac{\mathbf{w}}{\|\mathbf{w}\|}$$



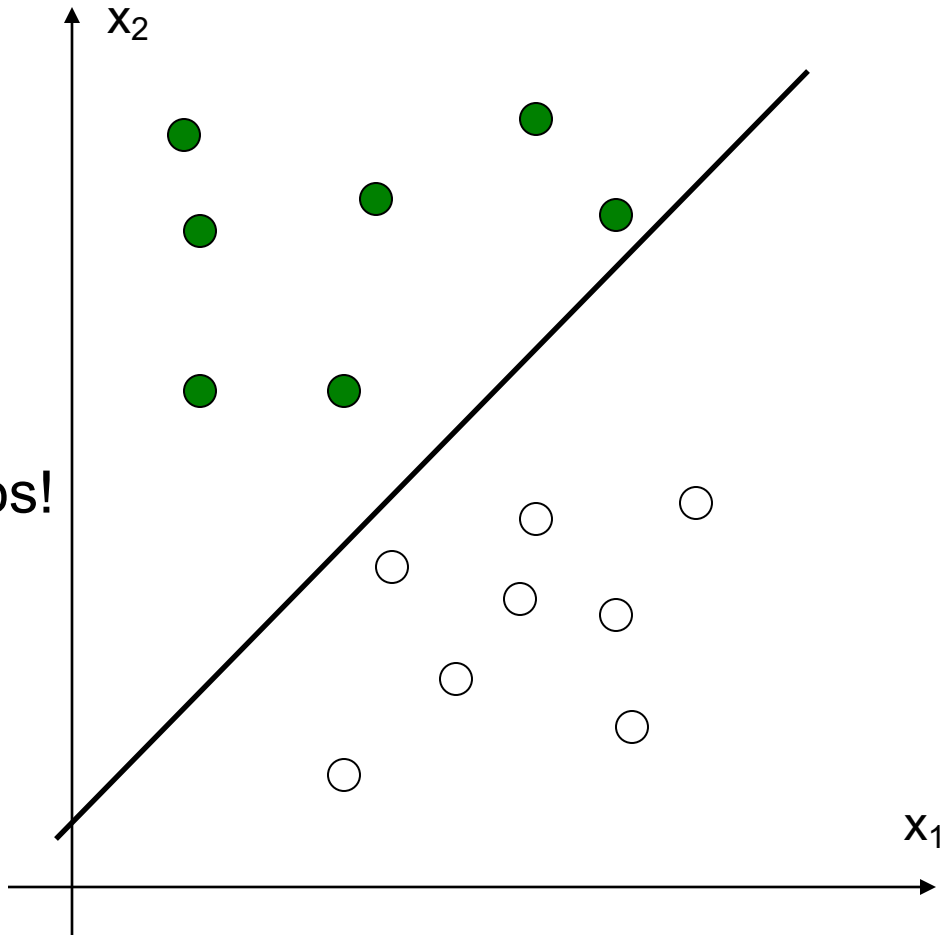
Función discriminante lineal

- ¿Cómo clasificar estos puntos mediante una función discriminante lineal reduciendo al mínimo el error?
- Podrían existir una cantidad infinita de posibles hiperplanos!



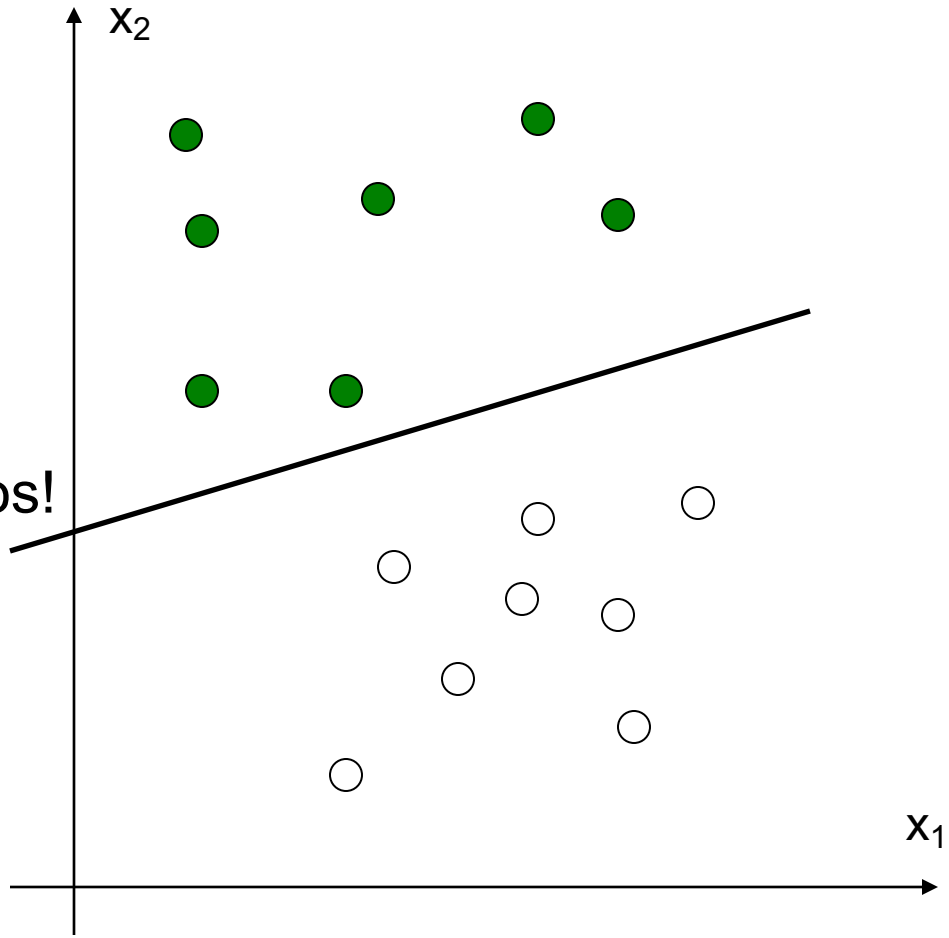
Función discriminante lineal

- ¿Cómo clasificar estos puntos mediante una función discriminante lineal reduciendo al mínimo el error?
- Podrían existir una cantidad infinita de posibles hiperplanos!



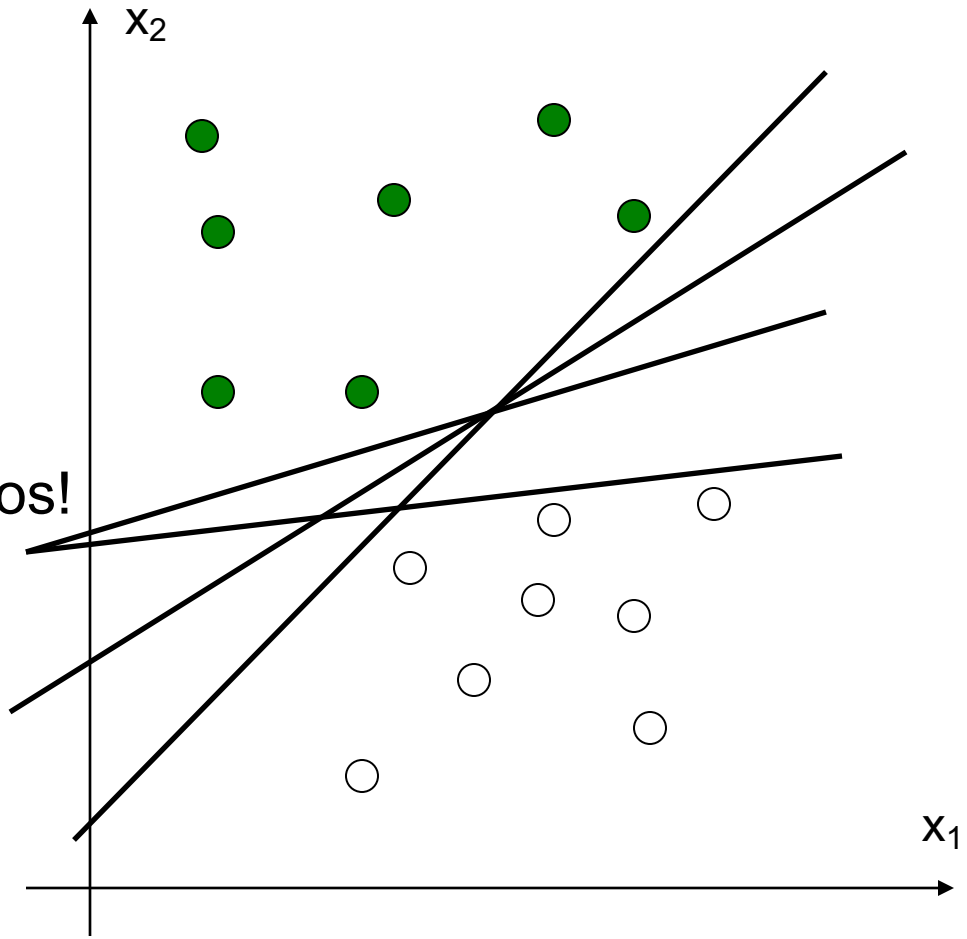
Función discriminante lineal

- ¿Cómo clasificar estos puntos mediante una función discriminante lineal reduciendo al mínimo el error?
- Podrían existir una cantidad infinita de posibles hiperplanos!



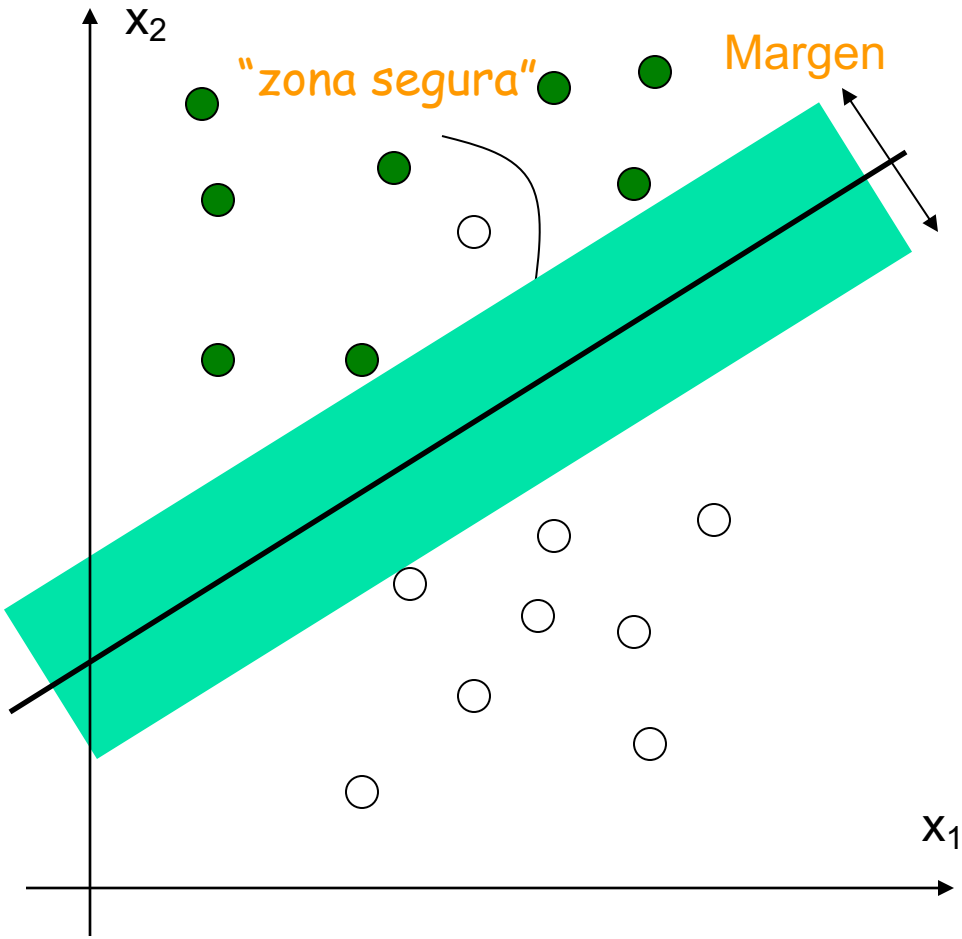
Función discriminante lineal

- ¿Cómo clasificar estos puntos mediante una función discriminante lineal reduciendo al mínimo el error?
- Podrían existir una cantidad infinita de posibles hiperplanos!
- ¿Cuál es el mejor?



Clasificador lineal con el margen más amplio

- La función discriminante lineal con el máximo margen es la mejor
- El margen se define como el ancho que limita los datos (podría no existir)
- ¿Por qué es la mejor?
  - Generalización robusta y resistente a los valores atípicos



Formulación del Problema: Máquinas de Soporte Vectorial

- Supongamos que tenemos un problema de clasificación donde la variable a predecir es binaria y que tenemos *n* casos de entrenamiento (x_i, y_i) para $i = 1,2,\dots,n$ donde $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})$, es decir, los x_i son los predictores y y_i es la variable a predecir.
- Asumimos que $y_i \in \{-1,1\}$ denota la etiqueta de clase.
- La frontera de decisión de puede escribir como:

$$w \cdot x + b = 0$$

- Donde w y b son los parámetros del modelo.

Resolver un Problema Optimización

Un problema de programación cuadrática con restricciones lineales

$$\min_{w} \frac{\| w \|^2}{2}$$

Sujeto a: $y_i(w \cdot x_i + b) \geq 1$ para $i = 1,2, \ldots, n$

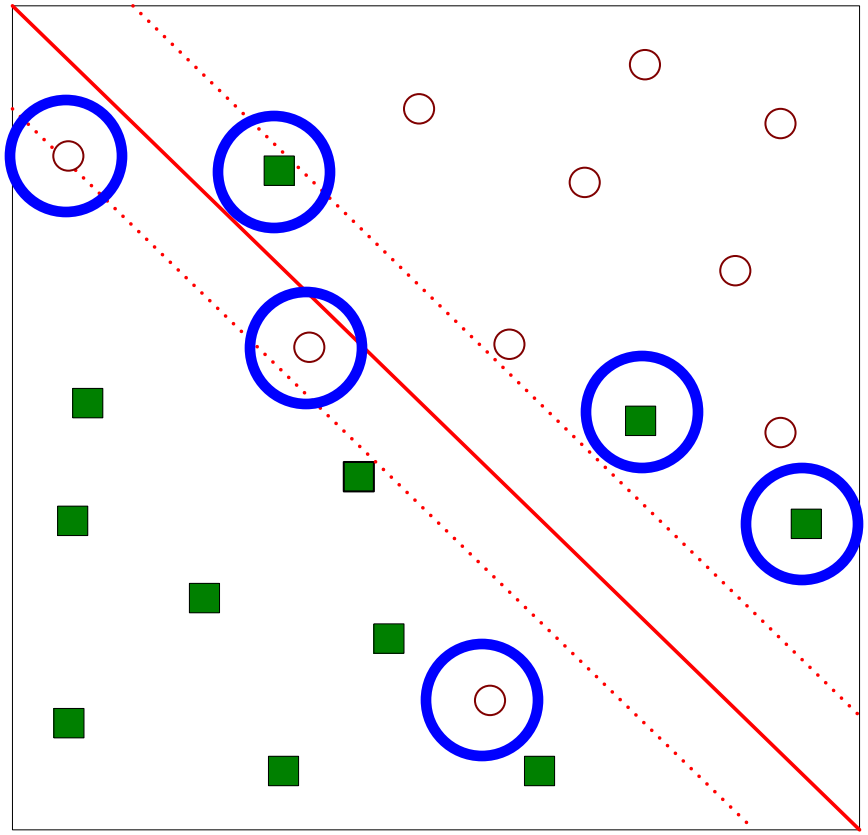
Minimización de Lagrange

$$L_P(w, b, \lambda_i) = \frac{\| w \|^2}{2} - \sum_{i=1}^{n} \lambda_i \left(y_i(w \cdot x_i + b) - 1\right)$$

con $\lambda_i \geq 0$ (los λ se llaman multiplicadores de Lagrange)

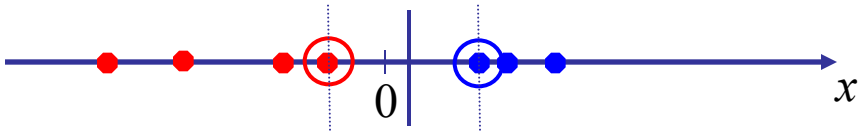
Máquinas de Soporte Vectorial

- ¿Qué pasa si el problema no es linealmente separable?



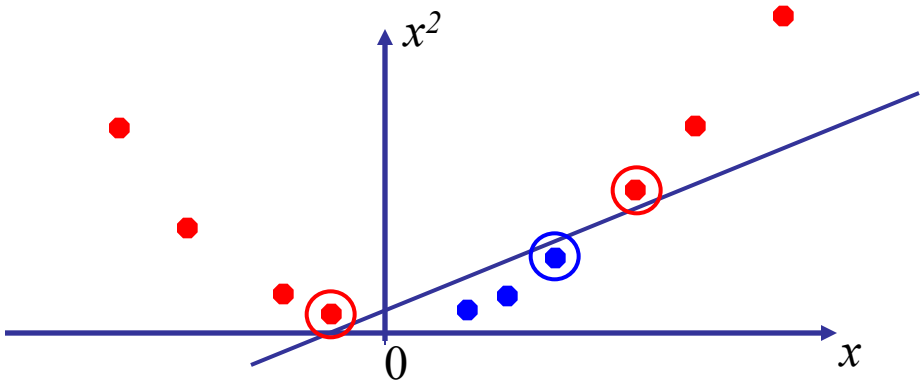
MVS no linealmente separables

- Datos linealmente separables:



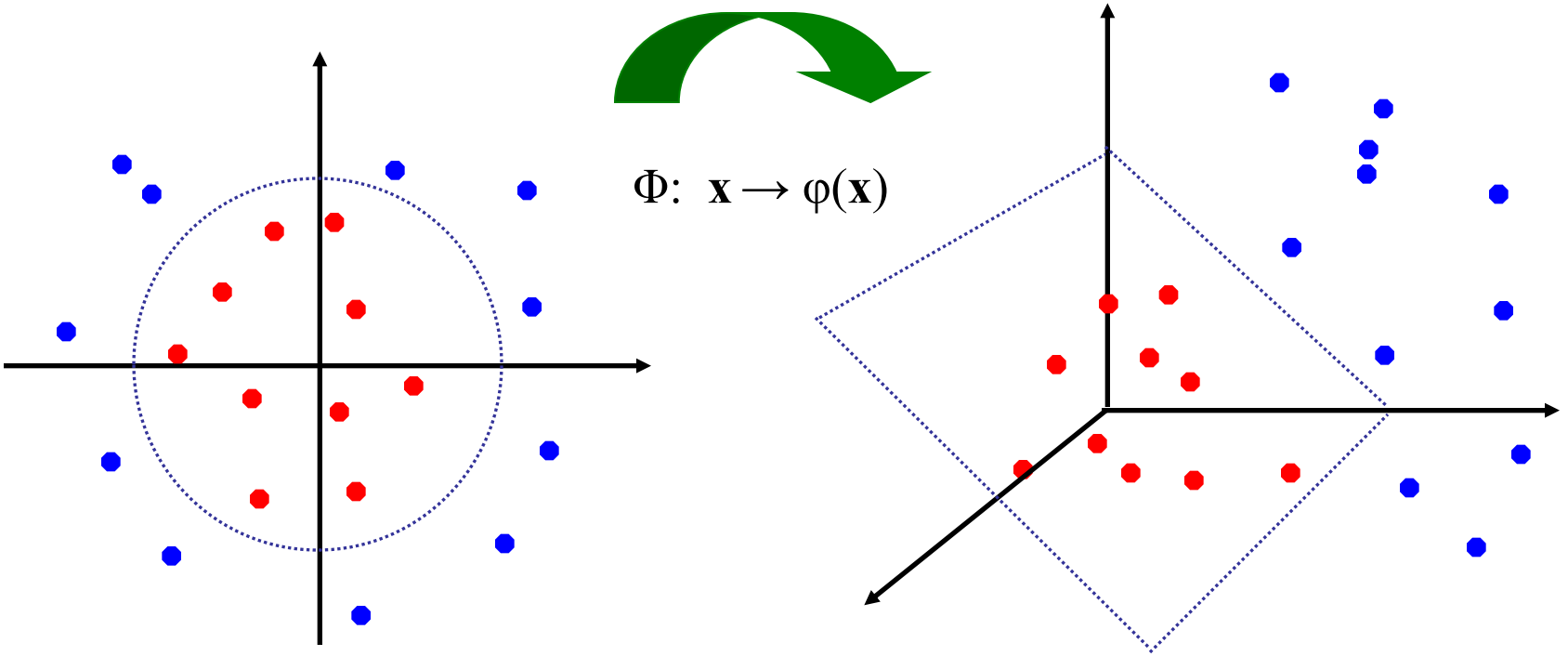
- Datos no linealmente separables:

- La idea es… Encontrar una función para trasladar los datos a un espacio de mayor dimensión:



MVS no linealmente separables

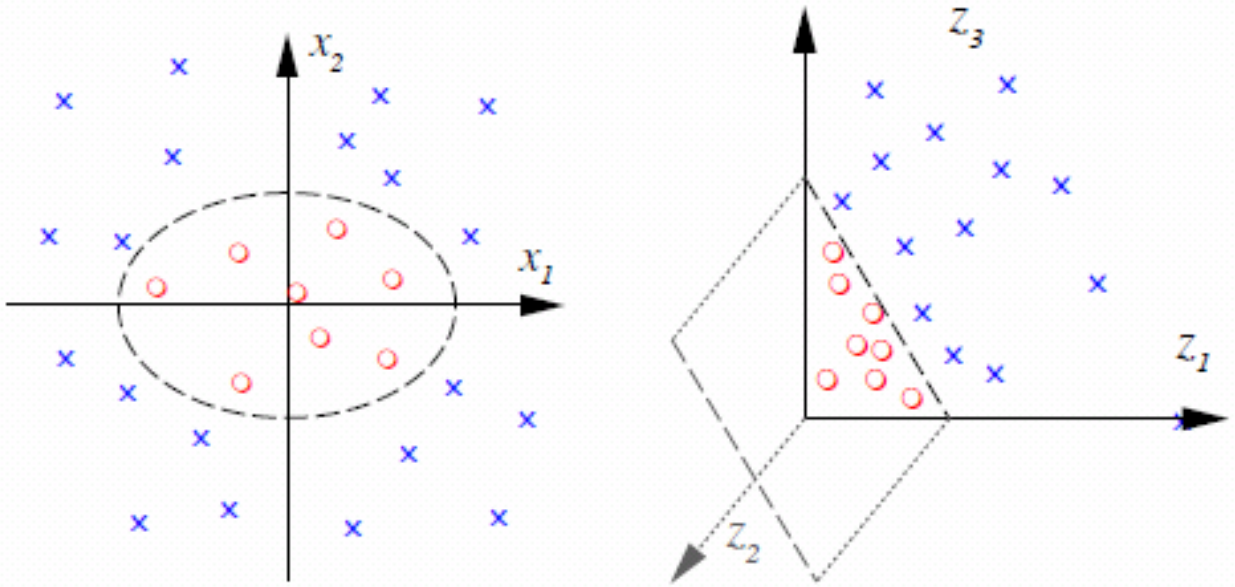
- Idea general: Los datos de entrada se puede trasladar a algún espacio de mayor dimensión en el que la Tabla de Entrenamiento sí sea separable:



MVS no linealmente separables

$$\Phi : R^2 \rightarrow R^3$$

$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{(2)} x_1 x_2, x_2^2)$$



El Truco del Núcleo (Kernel Trick)

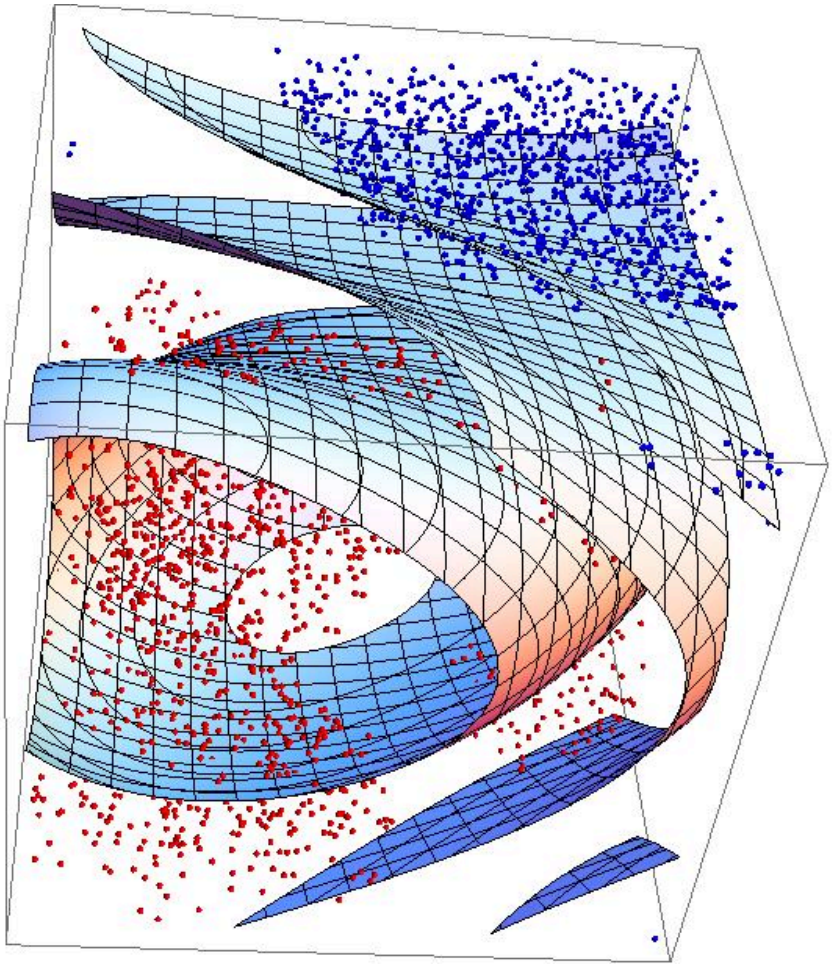
Ejemplos: Algunas funciones núcleo K usadas son:

$$K(x, y) = (x \cdot y + 1)^p$$

$$K(x, y) = e^{-\|x-y\|^2/(2\sigma^2)}$$

$$K(x, y) = \tanh(kx \cdot y - \delta)$$

El Truco del Núcleo (Kernel Trick)



Ejemplo 1: IRIS.CSV

Ejemplo con la tabla de datos IRIS
IRIS Información de variables:

1.sepal largo en cm
2.sepal ancho en cm
3.petal largo en cm
4.petal ancho en cm
5.clase:

- Iris Setosa
- Iris Versicolor
- Iris Virginica

	A	B	C	D	E
1	s.largo	s.ancho	p.largo	p.ancho	tipo
2	5.1	3.5	1.4	0.2	setosa
3	4.9	3.0	1.4	0.2	setosa
4	4.7	3.2	1.3	0.2	setosa
5	4.6	3.1	1.5	0.2	setosa
6	5.0	3.6	1.4	0.2	setosa
7	5.4	3.9	1.7	0.4	setosa
8	4.6	3.4	1.4	0.3	setosa
9	5.0	3.4	1.5	0.2	setosa
10	4.4	2.9	1.4	0.2	setosa
11	4.9	3.1	1.5	0.1	setosa
12	5.4	3.7	1.5	0.2	setosa
13	4.8	3.4	1.6	0.2	setosa
14	4.8	3.0	1.4	0.1	setosa
15	4.3	3.0	1.1	0.1	setosa
16	5.8	4.0	1.2	0.2	setosa
17	5.7	4.4	1.5	0.4	setosa
18	5.4	3.9	1.3	0.4	setosa
19	5.1	3.5	1.4	0.3	setosa
20	5.7	3.8	1.7	0.3	setosa
21	5.1	3.8	1.5	0.3	setosa
22	5.4	3.4	1.7	0.2	setosa
23	5.1	3.7	1.5	0.4	setosa
24	4.6	3.6	1.0	0.2	setosa
25	....	...	...	...	....

Descripción de Variables

MontoCredito

Numérica

MontoCuota

1=Muy Bajo
2=Bajo
3=Medio
4=Alto

IngresoNeto

1=Muy Bajo
2=Bajo
3=Medio
4=Alto

GradoAcademico

1=Bachiller
2=Licenciatura
3=Maestría
4=Doctorado

CoeficienteCreditoAvaluo

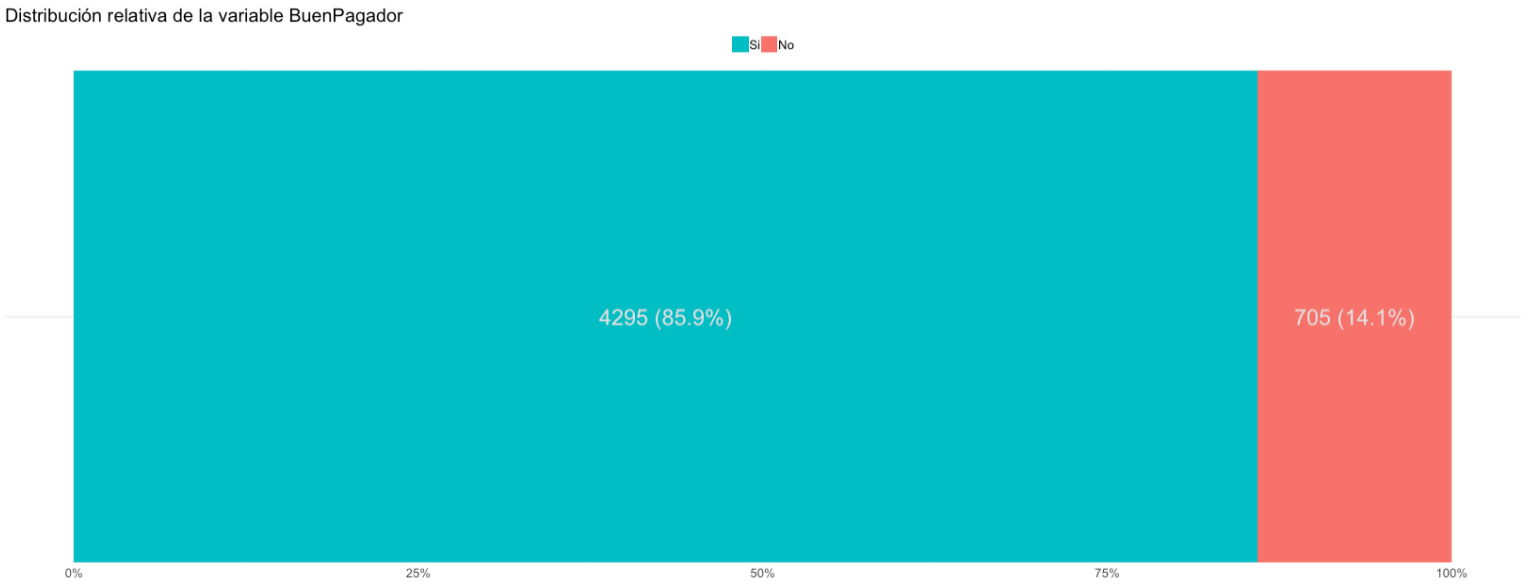
1=Muy Bajo
2=Bajo
3=Medio
4=Alto

BuenPagador

1=NO
2=Si

Problema desequilibrado

Distribución Variable a Predecir | Gráfico de Pares | Distribución Variables Categóricas Según Variable a Predecir | Densidad Variables Numéricas Según Variable a Predecir



Ejecutar

```
colores <- gg_color_hue(length(unique(datos[,'BuenPagador'])))
label.size <- 9.5 - length(unique(datos[,'BuenPagador']))
label.size <- ifelse(label.size < 3, 3, label.size)
data. <- dist.x.predecir(datos, 'BuenPagador','BuenPagador')
ggplot(data., aes(x='', y=prop, fill= data.[['BuenPagador']]))+
geom_bar(width = 1, stat = 'identity')+
geom_text(aes(label = paste0(count, ' (', scales::percent(prop), ')'), y = prop ), color = 'gray90',
position = position_stack(vjust = .5), size = label.size)+
theme_minimal() +
theme(text = element_text(size=15)) +
scale_fill_manual(values = colores) +
```

sklearn.svm.SVC

class `sklearn.svm.` **SVC** (*C=1.0, kernel='rbf', degree=3, gamma='auto_deprecated', coef0=0.0, shrinking=True, probability=False, tol=0.001, cache_size=200, class_weight=None, verbose=False, max_iter=-1, decision_function_shape='ovr', random_state=None*) [source]

PR0MiDAT

IBEROAMERICANO

Programa Iberoamericano de Formación en Minería de Datos

Gracias….