IAA-2023c1
Clase 3: Clasificación

UNSAM
UNIVERSIDAD
NACIONAL DE
SAN MARTÍN

Repaso: Modelo Lineal

Modelo: $y = \vec{w} \cdot \vec{x} + w_0$

Función de Pérdida: $L(y,t) = (y-t)^2$

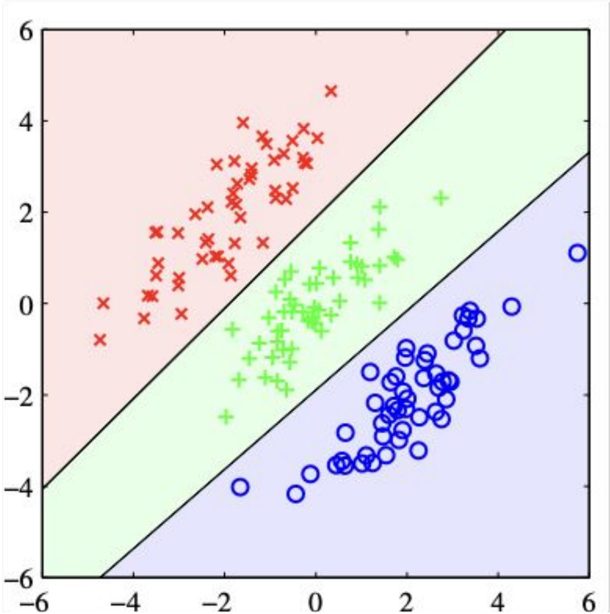
Target numérico: $y,\ t \in \mathbb{R}$

Optimizador:

- Solución algebraica (exacta si puedo invertir la matriz de diseño)
- Solución iterativa: Descenso por Gradiente

Clasificación

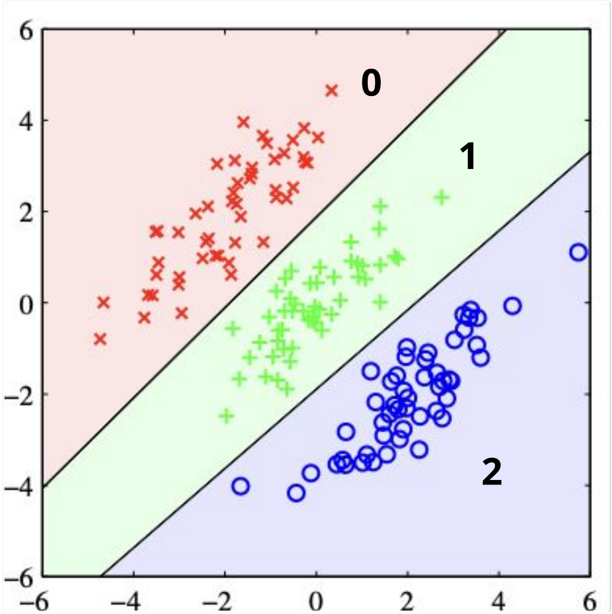
Un modelo de clasificación se caracteriza porque el objetivo (*target*) de una muestra x es un mapeo a una clase C_k (k=1, ... K).



Clasificación

Un modelo de clasificación se caracteriza porque el objetivo (*target*) de una muestra x es un mapeo a una clase C_k (k=0, … K-1).

Para esto, lo primero a definir es cómo codificamos esta clase de destino en un número: Para un punto del dataset (x, C_k)



- Label Encoding:
  Asignamos un número entero a cada clase de destino (las numeramos)

  $$t = k$$

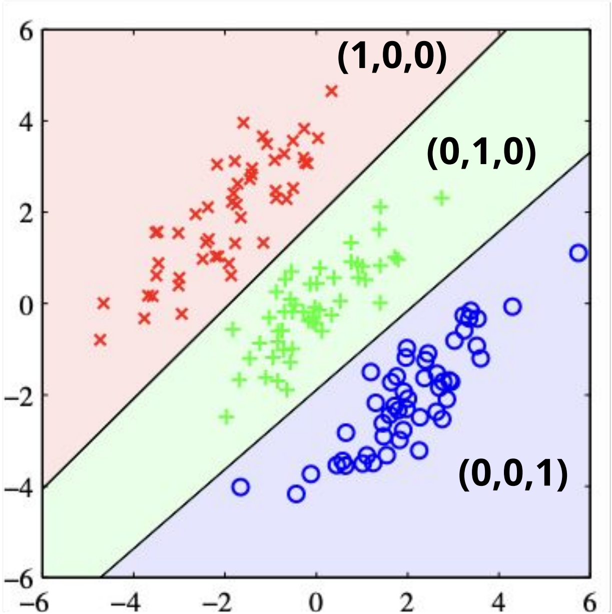
Clasificación

Un modelo de clasificación se caracteriza porque el objetivo (*target*) de una muestra x es un mapeo a una clase C_k (k=1, ... K).

Para esto, lo primero a definir es cómo codificamos esta clase de destino en un número: Para un punto del dataset (x, C_k)

- Label Encoding:
  Asignamos un número entero a cada clase de destino (las numeramos)
  $$t = k$$

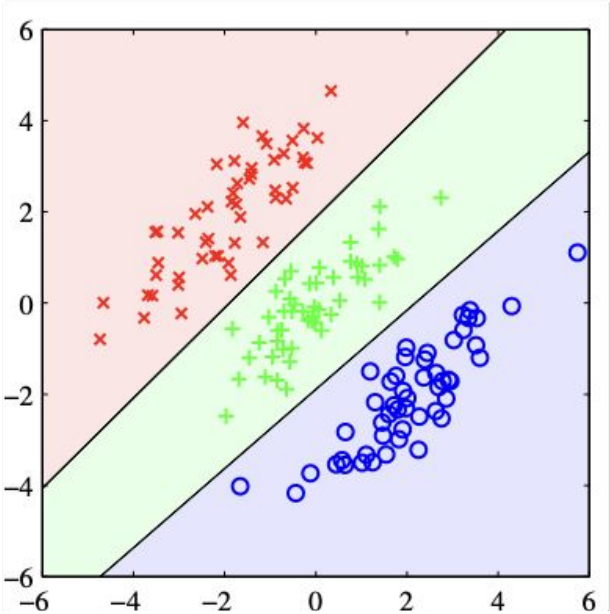
- One-Hot Encoding:
  Asignamos un vector de componentes nulas, salvo por la k-ésima que vale 1
  $$t = (0, \ldots, 0, 1, 0, \ldots, 0)$$



Clasificación

Un modelo de clasificación se caracteriza porque el objetivo (*target*) de una muestra x es un mapeo a una clase C_k (k=1, … K).

Para esto, lo primero a definir es cómo codificamos esta clase de destino en un número: Para un punto del dataset (x, C_k)



- Label Encoding:
  Asignamos un número entero a cada clase de destino (las numeramos)

  $t = k$ → *Implica un **orden** en las clases*

- One-Hot Encoding:
  Asignamos un vector de componentes nulas, salvo por la k-ésima que vale 1

  $t = (0, \ldots, 0, 1, 0, \ldots, 0)$ → *Todas las clases son ortogonales entre sí*

Clasificación Binaria

Clasificación binaria es el caso particular en que hay solo 2 clases (K=2).
En este caso se suele usar label encoding ya que es equivalente al one-hot.

	Caso General	Caso Binario (K=2)
Label Encoding	$t \in [0, 1, ..., k-1]$	$t = 0$ ó 1
One-Hot Encoding	$t = (0,..,0,1,0,...0)$	$t = (1, 0)$ ó $(0,1)$

Algoritmos de Función Discriminante

Intentamos predecir el target, es decir la clase a la que pertenece:

- Cuadrados Mínimos: $y = \vec{w} \cdot \vec{x} + w_0 \quad L(y,t) = (y-t)^2$
- Perceptrón: $y = \Theta(\vec{w} \cdot \vec{x} + w_0) \qquad \Theta(z) = 1 \text{ si } z > 0, \text{ sino } 0$

  (más un algoritmo especial para optimizar los pesos)

Algoritmos Probabilísticos Discriminativos

Intentamos predecir *la probabilidad* de que pertenezca a una clase.
(En el fondo, lo reducimos a un problema de regresión)

$$y_k = p(C_k|x)$$

El caso paradigmático es el Regresor Logístico

- **Regresor Logístico:**

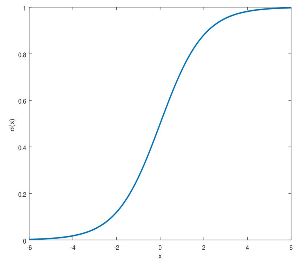
$$y = \sigma(\vec{w} \cdot \vec{x} + w_0)$$

optimizando la Entropía Cruzada:

$$L(y;t) = -\left[t \log(y)(1-t)\log(1-y)\right]$$

$$\left(= -\sum_k t_k \log(p_k)\right)$$

$$\sigma(z) = \frac{1}{1+e^{-z}}$$



Algoritmos Probabilísticos Generativos

Intentamos predecir *la probabilidad de observar el punto si perteneciera a cada clase*.
Es decir, la probabilidad de observar x condicionada sobre que x pertenece a la clase k. Lo que modelamos es el proceso que genera los datos.

$$y_k = p(x|C_k)$$

Esta probabilidad se relaciona con la probabilidad de pertenecer a cada clase por el Teorema de Bayes.

$$p(C_k|x) = p(x|C_k)\frac{p(C_k)}{p(x)} \sim p(x|C_k)p(C_k)$$

Pero además nos permite:

- Utilizar conocimiento sobre la distribución de clases para balancear nuestra predicción
- Generar data sintética.

Modelos Lineales Generalizados

- Perceptrón: $$y = \Theta(\vec{w} \cdot \vec{x} + w_0)$$
- Regresor Logístico $$y = \sigma(\vec{w} \cdot \vec{x} + w_0)$$
- En general: $$y = \ell(\vec{w} \cdot \vec{x} + w_0)$$

Función de “Link”

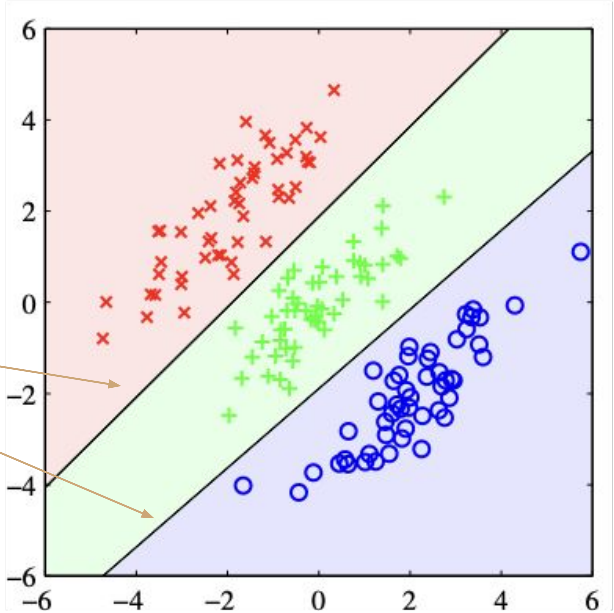
El output es constante cuando $$\vec{w} \cdot \vec{x} + w_0 = cte$$

Modelos Lineales Generalizados

El output es constante cuando $\vec{w} \cdot \vec{x} + w_0 = cte$

En particular la zona donde cambio mi predicción de clase también sigue la ecuación de una recta:

Frontera de Decisión Lineal

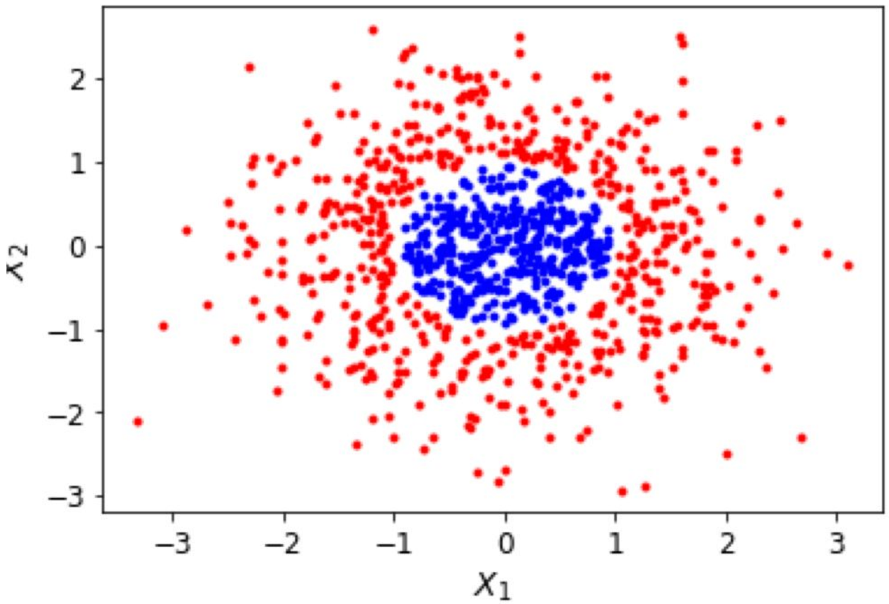


Separabilidad Lineal

Podemos separar las clases trazando una linea recta? (i.e. con un modelo lineal)

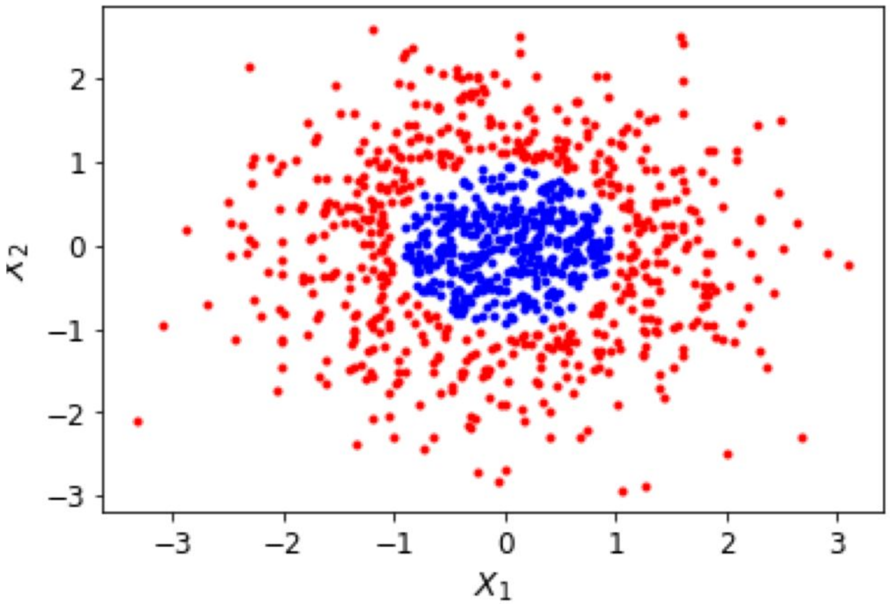
Separabilidad Lineal

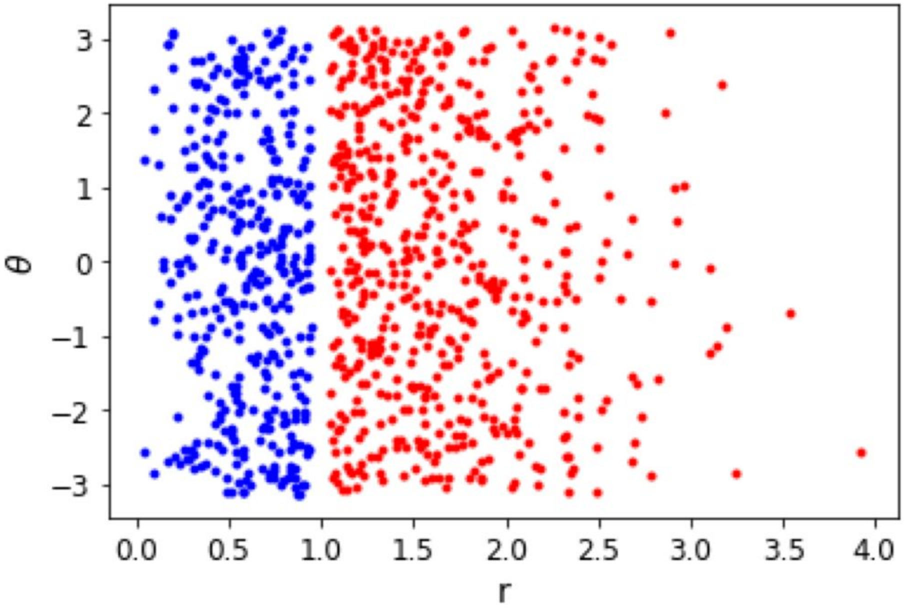
Podemos separar las clases trazando una linea recta? (i.e. con un modelo lineal)



Separabilidad Lineal

Podemos separar las clases trazando una linea recta? (i.e. con un modelo lineal)



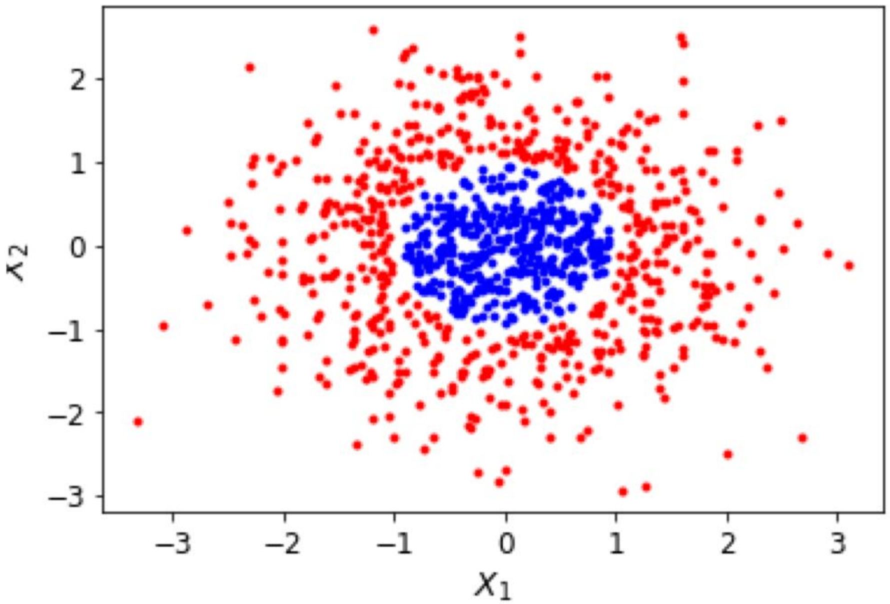


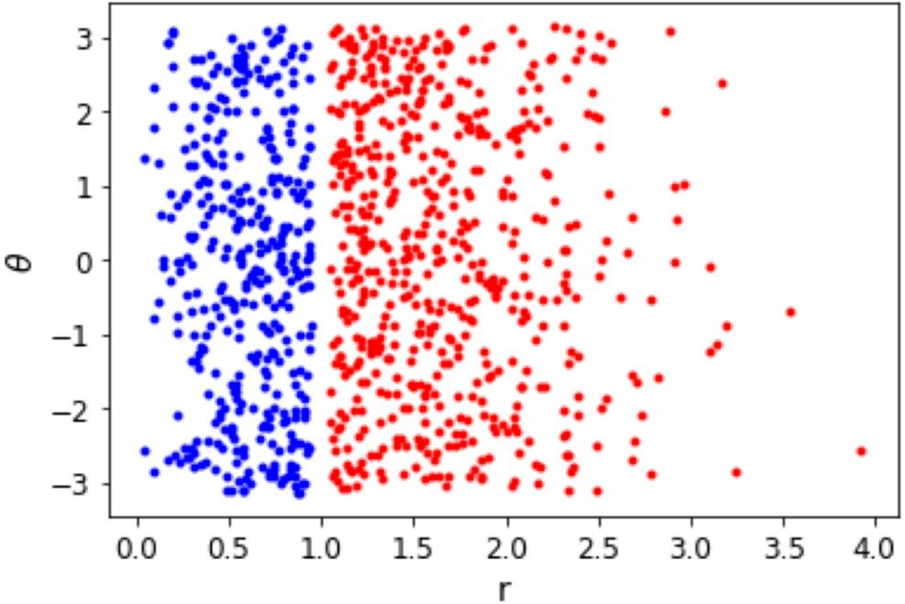
Coordenadas polares

Separabilidad Lineal

Podemos separar las clases trazando una linea recta? (i.e. con un modelo lineal)

Depende del las variables/coordenadas que usemos





Coordenadas polares

[Extra] Optimización Regresor Logístico

$$\frac{\mathrm{d}L(y;t)}{\mathrm{d}y} = -\frac{t}{y} + \frac{1-t}{1-y} = \frac{y-t}{y(1-y)}$$

$$\frac{\mathrm{d}\sigma(z)}{\mathrm{d}z} = \sigma(z)[1-\sigma(z)]$$

$$\frac{\mathrm{d}L(\sigma(z);t)}{\mathrm{d}z} = y - t$$

$$\nabla_{\vec{w}}(\vec{w}\cdot\vec{x} + w_0) = \vec{x}$$

$$\nabla_{\vec{w}} L = (y-t)\vec{x}$$

Fácil de optimizar