

Clase 11: Clasificador Lineal

Laboratorio de Datos, FCEyN, 30/04/2021

A dark blue diagonal gradient bar that starts from the bottom left corner and extends towards the top right corner, covering the lower half of the slide.

Introducción

La clase de hoy probablemente sea más corta que las anteriores y eso está bueno porque las últimas clases venimos a full y terminando sobre la hora o nos falta tiempo

No hay mucho más para decir de un Clasificador lineal que no hayamos dicho hasta ahora

Los conceptos nuevos de la clase de hoy son el discriminador lineal de Fisher el cual es un ejemplo particular de LDA (análisis discriminante lineal) y el perceptrón

La novedad de la notebook que vamos a ver hoy es el dataset y un pispéo a redes neuronales. Vamos a trabajar con un dataset de imágenes de distintas prendas de vestir (pantalones, remeras, zapatos, etc.). Al final de la clase voy a comentar cómo está compuesto el dataset y cómo vamos a trabajar con las imágenes.

Resumen de lo visto

Vimos modelos de regresión: lineal, polinomial y logística [clases del 13-16-20/04]
donde optimizamos los pesos del modelo para obtener y números reales que mejor se ajusten a los y datos

Vimos clasificación simple (una sola clase binaria/dos clases) usando regresión logística [clase del 20/04]
donde ahora los y que queremos obtener tienen un rango acotado $[0,1]$ porque queremos identificarlos con una determinada clase

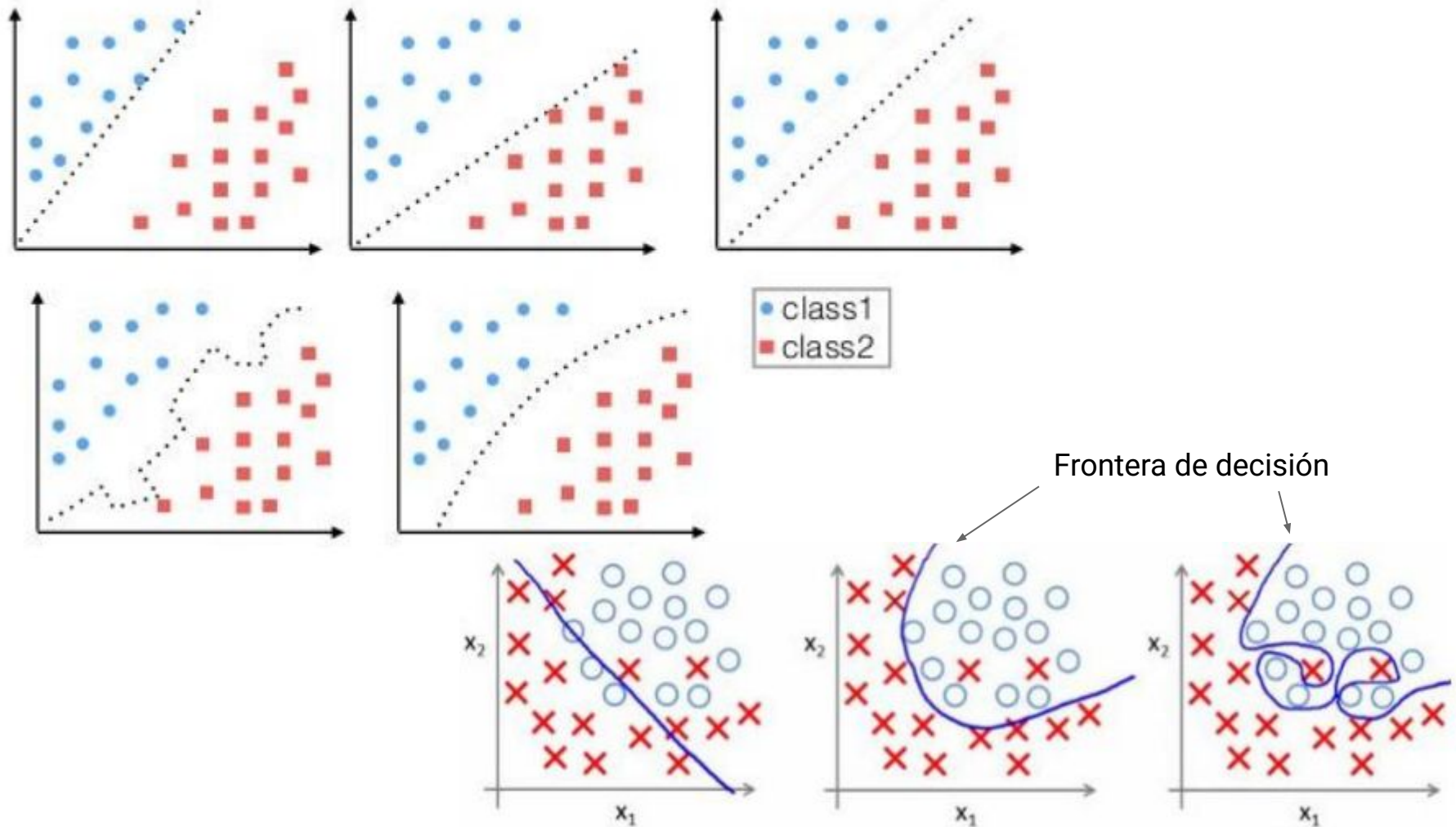
Vamos a generalizar (y repetir) el concepto de clasificación y ver distintos tipos de clasificadores

Definición

- Una definición de **clasificación**:

“Usar las CARACTERÍSTICAS de un objeto para identificar a qué CLASE/CATEGORÍA pertenece”
- Ejemplos:
 - Si un mail es “spam” o no
 - Qué tipo de objeto está presente en una imagen
 - Diagnosticar una enfermedad según los síntomas de la persona
 - El tipo celular según el transcriptoma de la célula en un experimento sc-RNAseq
- La clasificación es un ejemplo de **reconocimiento de patrones**

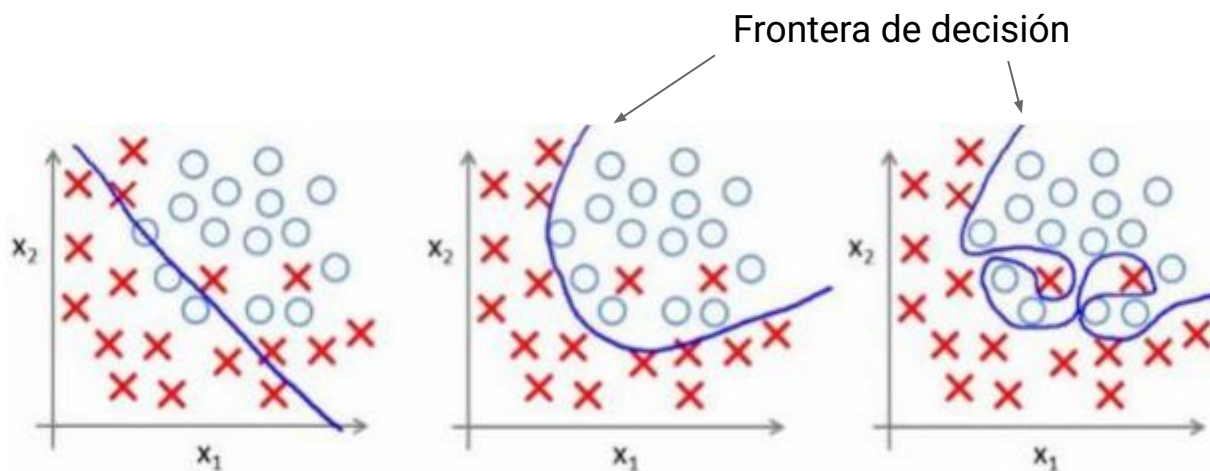
Definición



Definición

En modelos lineales de clasificación las fronteras de decisión son funciones lineales del input \mathbf{X} (vector de D -características), es decir están definidas por hiperplanos $D-1$ dimensionales

Datasets son *linealmente separables* si sus clases están bien separadas por fronteras de decisión lineales



Generalización de la regresión logística

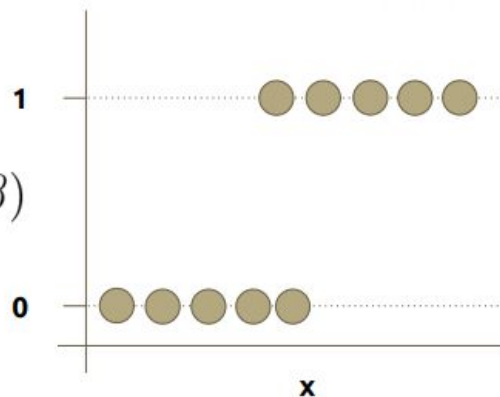
La hipótesis de representación

Pensemos a f como una probabilidad y usemos una función con imagen acotada entre cero y uno

función sigmoidea

$$P(y|x, \beta) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 \cdot x)}}$$

$$y = f(x, \beta) \equiv P(y|x, \beta)$$



$$y = f(\vec{w} \cdot \vec{x})$$

$$f(\vec{w} \cdot \vec{x}) = f\left(\sum_j w_j \cdot x_j\right) = f(\beta_0 + \beta_1 \cdot x)$$

f es la **función de activación** y puede ser una función no lineal

Identidad del clasificador lineal

- Clasificación lineal:
 - Un algoritmo de clasificación (*clasificador*) que hace sus clasificaciones (*predicciones*) basadas en una función de predicción lineal combinando un set de pesos, **W**, con el input **X**

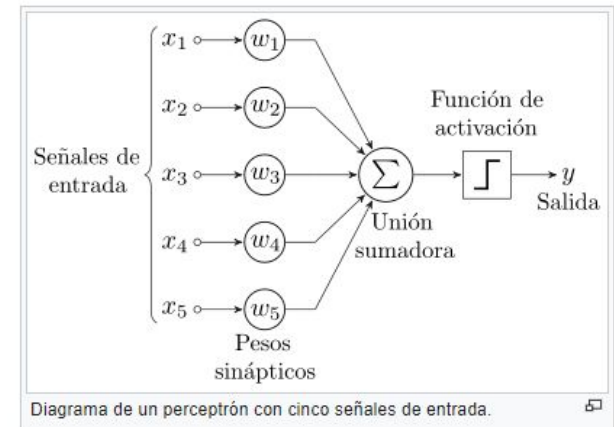
$$y = f(\vec{w} \cdot \vec{x}) = f\left(\sum_j w_j \cdot x_j\right)$$

- Fronteras de decisión planas:
 - líneas, planos, ...

Diferentes “approaches”

- Creando explícitamente la **función discriminante**: (función que le asigna a cada vector **X** una clase y, y se optimizan los coeficientes **w** para disminuir el error)

- Perceptrón
- Support Vector Machines (SVM)
[Clase que viene]



- Approach probabilístico:
 - Modelar la probabilidad condicional (Bayesiana)
 - Algoritmos

- Regresión logística

$$\begin{aligned} p(C_1|\mathbf{x}) &= \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_1)p(C_1) + p(\mathbf{x}|C_2)p(C_2)} \\ &= \frac{1}{1 + \exp(-a)} = \sigma(a) \end{aligned}$$

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

$$p(\mathbf{x}) = \sum_k p(\mathbf{x}|C_k)p(C_k)$$

Función Discriminante

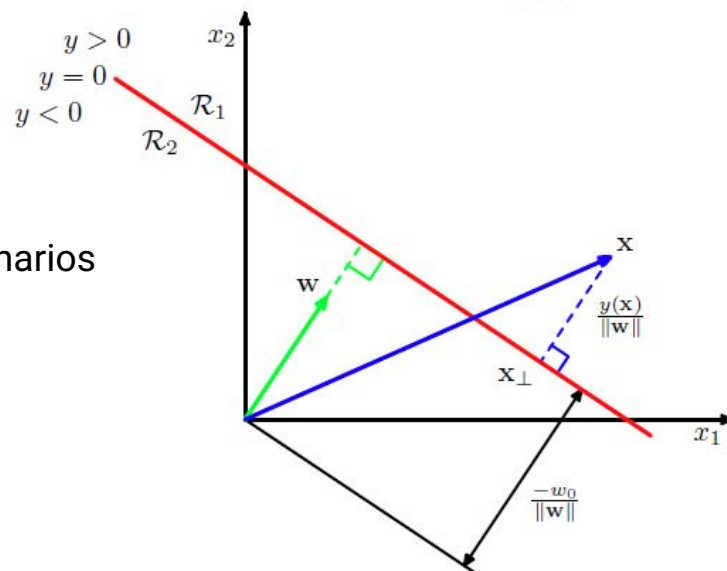
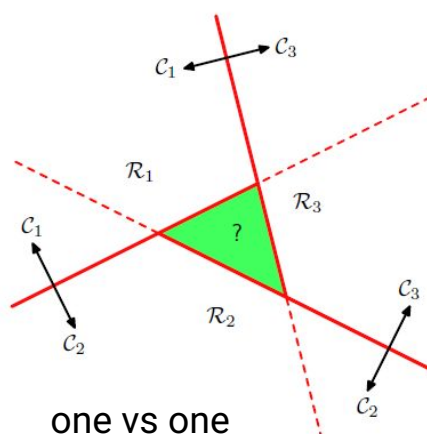
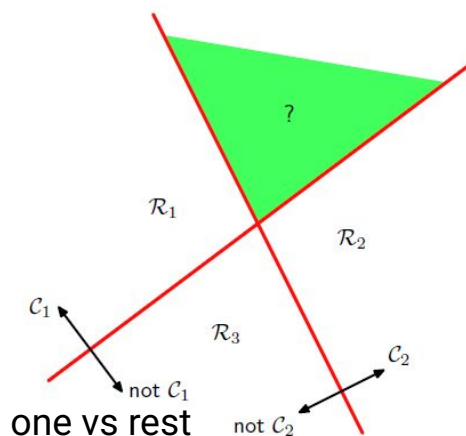
- Dos clases (K=2):

- Más simple -> función lineal del input $y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$

$$\begin{aligned}\tilde{\mathbf{W}} &= (w_0, \mathbf{w}) \\ \tilde{\mathbf{X}} &= (x_0, \mathbf{x}) \\ x_0 &= 1 \\ y(\mathbf{x}) &= \tilde{\mathbf{W}}^T \tilde{\mathbf{X}}\end{aligned}$$

- Más de dos clases (K>2):

- Tentados a usar K-1 clasificadores lineales binarios



Función Discriminante

- Dos clases (K=2):

- Más simple -> función lineal del input: $y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$

$$\tilde{\mathbf{W}} = (w_0, \mathbf{w})$$

$$\tilde{\mathbf{x}} = (x_0, \mathbf{x})$$

$$x_0 = 1$$

$$y(\mathbf{x}) = \tilde{\mathbf{W}}^T \tilde{\mathbf{x}}$$

- Más de dos clases (K>2):

- Función discriminante de K-clases: $y_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x} + w_{k0}$
(K funciones lineales)

- Fronteras de decisión: $y_k(\mathbf{x}) = y_j(\mathbf{x})$

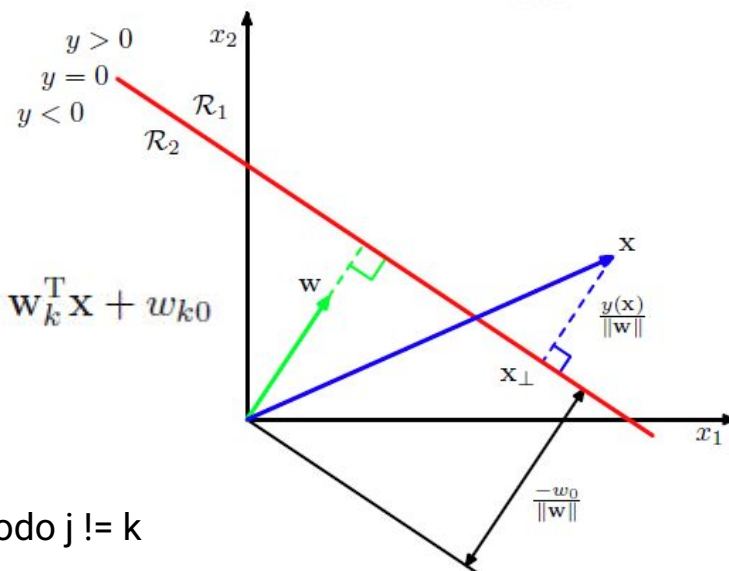
- asigno X a la clase K si $y_k(\mathbf{x}) > y_j(\mathbf{x})$ para todo $j \neq k$

- One-hot encoding: $y = (y_1, y_2, \dots, y_k)$

- w ahora es una matriz de pesos cuya k-ésima columna es $\tilde{\mathbf{w}}_k = (w_{k0}, \mathbf{w}_k^T)^T$

$$y(\mathbf{x}) = \tilde{\mathbf{W}}^T \tilde{\mathbf{x}}$$

$$\tilde{\mathbf{x}} = (1, \mathbf{x}^T)^T$$



Aprendizaje de parámetros de función discriminante

1. Método basado en cuadrados mínimos
2. Discriminante lineal de Fisher
 - Método de reducción de dimensionalidad
3. Perceptrón

Aprendizaje de parámetros de función discriminante

$$y(x) = \widetilde{W}^T \widetilde{X}$$

1. Método basado en cuadrados mínimos

- Ajusta simultáneamente un modelo de regresión lineal a cada una de las columnas de $y = (y_1, y_2, \dots, y_k)$

- Los pesos van a tener esta pinta: $\widetilde{W} = (\widetilde{X}^T \widetilde{X})^{-1} \widetilde{X}^T \mathbf{T} = \widetilde{X}^\dagger \mathbf{T}$

donde \mathbf{T} son las etiquetas reales y \widetilde{X} una matriz con los datos de entrada

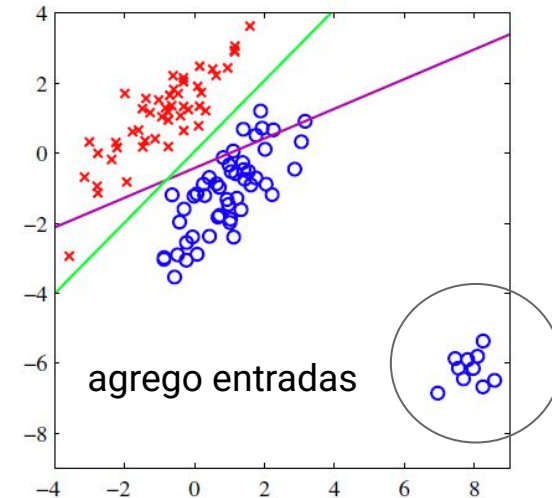
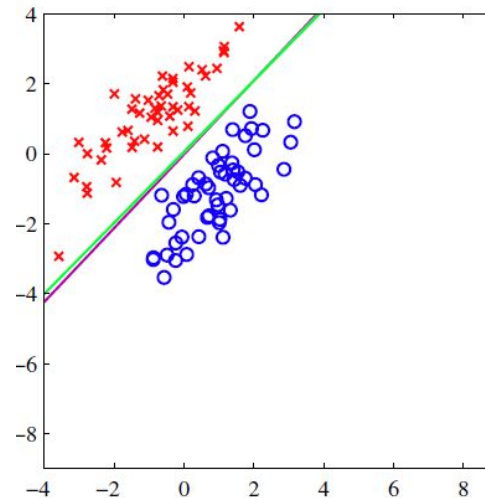
- Clasificar una nueva observación x :

- Calculo la función discriminante para cada clase $y_i(x) = W \cdot X$
- Me quedo con la clase que tenga mayor valor de $y_i(x)$

Aprendizaje de parámetros de función discriminante

1. Método basado en cuadrados mínimos

- Funciona bien:
 - linealmente separables
 - pocos outliers
 - $K = 2$



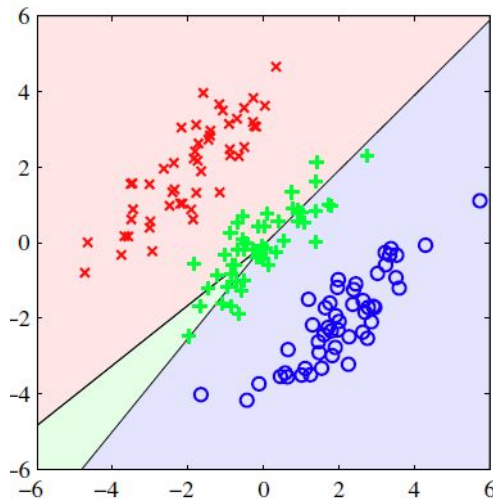
- Frontera de decisión de la función logística
- Frontera de decisión por cuadrados mínimos

Aprendizaje de parámetros de función discriminante

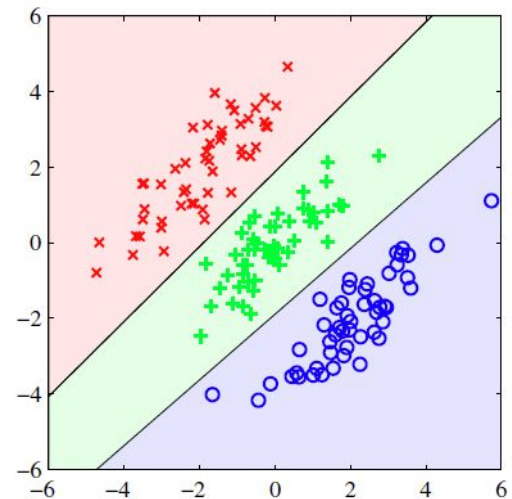
1. Método basado en cuadrados mínimos

- No funciona bien:

- $K > 2$



cuadrados mínimos



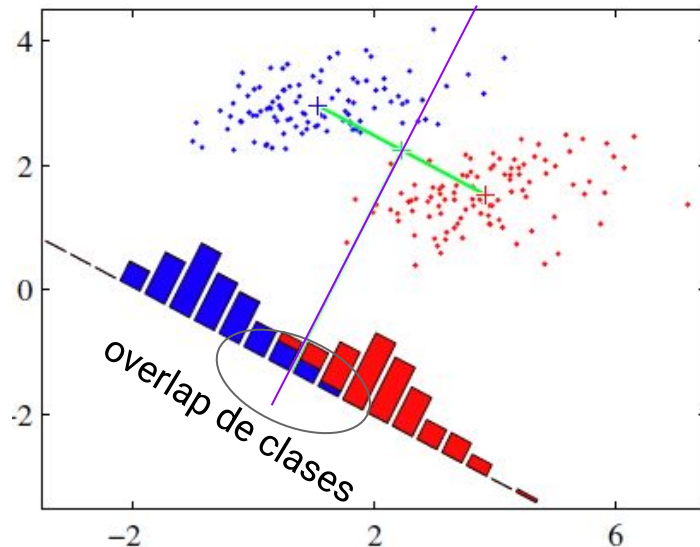
función logística

Aprendizaje de parámetros de función discriminante

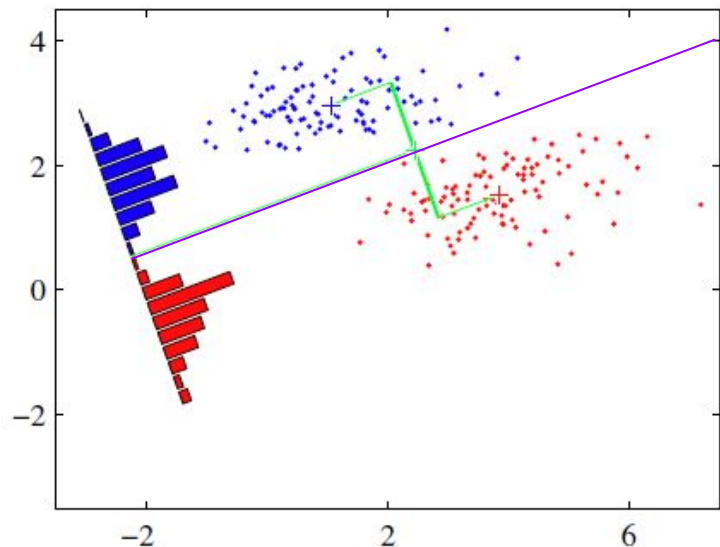
2. Discriminante lineal de Fisher

- Reducción de dimensionalidad

Criterio de Fisher: $J(w) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2}$ $J(w) = \frac{w^T S_B w}{w^T S_W w}$



Histogramas de la proyección de las clases sobre la línea que pasa por sus medias



Histogramas de la proyección de las clases sobre la línea determinada por el discriminante lineal de Fisher

- Al proyectar estamos usando $W \rightarrow$ para que no se disparen los pesos pedimos $\sum_i w_i^2 = 1$

Aprendizaje de parámetros de función discriminante

3. Perceptrón

$$y = f(\vec{w} \cdot \vec{x}) = f\left(\sum_j w_j x_j\right), f(a) = \begin{cases} +1, & a \geq 0 \\ -1, & a < 0. \end{cases}$$

¿Cómo encontramos los pesos?

- Elegir unos pesos arbitrarios
- Para cada ejemplo \mathbf{x}_j del set de entrenamiento calculo y_j

$$\begin{aligned} y_j(t) &= f[\mathbf{w}(t) \cdot \mathbf{x}_j] \\ &= f[w_0(t)x_{j,0} + w_1(t)x_{j,1} + w_2(t)x_{j,2} + \cdots + w_n(t)x_{j,n}] \end{aligned}$$

- Actualizamos los pesos

$$w_i(t+1) = w_i(t) + \alpha(d_j - y_j(t))x_{j,i} \quad 0 \leq i \leq n$$

α es el "learning rate"

d_j es el vector de etiquetas reales

Aprendizaje de parámetros de función discriminante

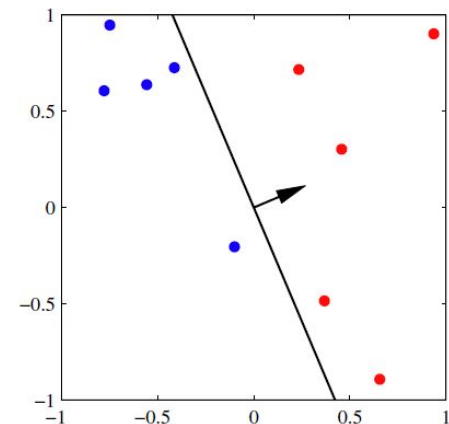
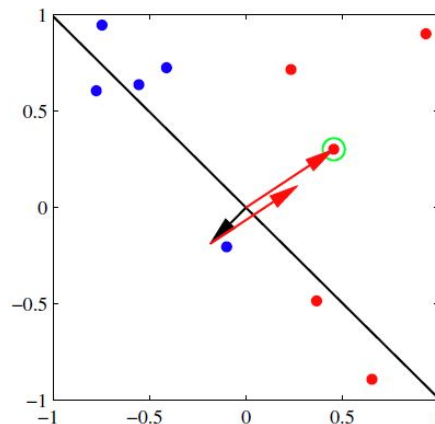
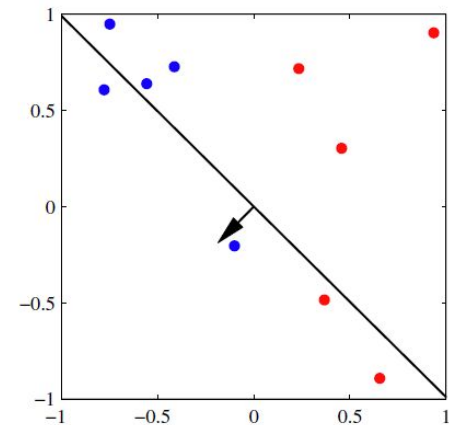
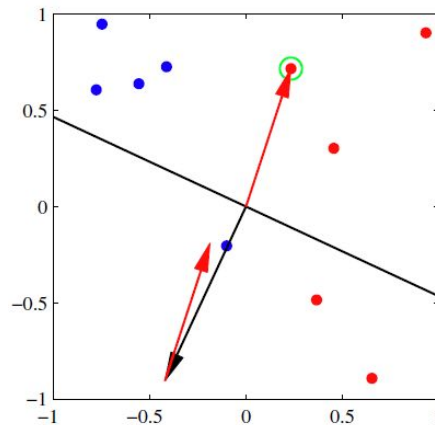
3. Perceptrón

la línea negra es el vector W

su punta indica dónde deberían estar los puntos rojos

Me paro en un dato (el verde) y me fijo si lo clasificó bien.
Si está bien clasificado no hago nada.
Si lo clasificó mal entonces le sumo su vector de features al vector W y así sucesivamente

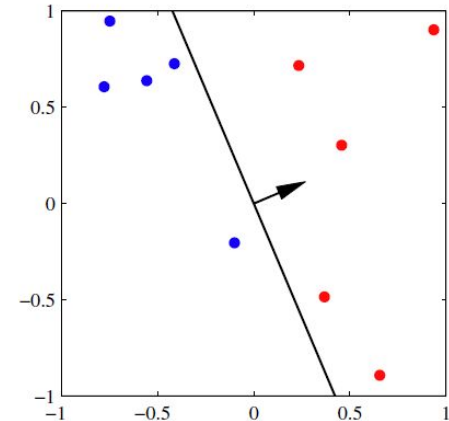
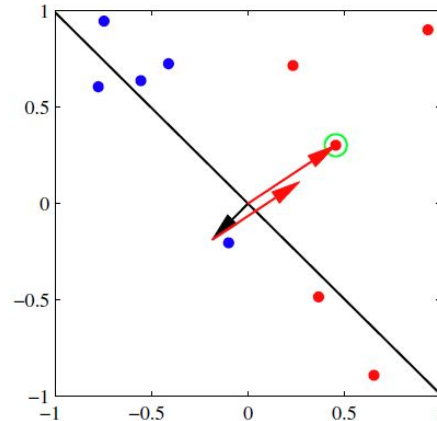
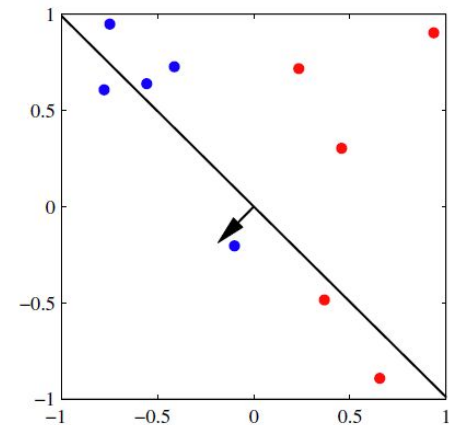
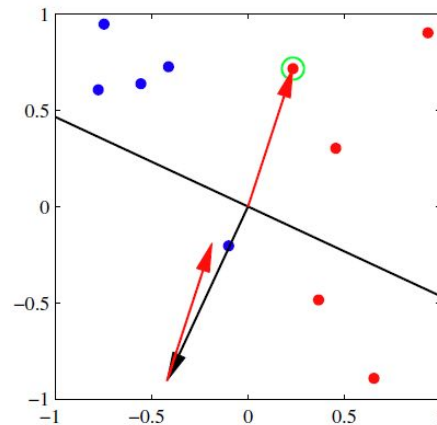
Termina cuando no queda ejemplos mal clasificados



Aprendizaje de parámetros de función discriminante

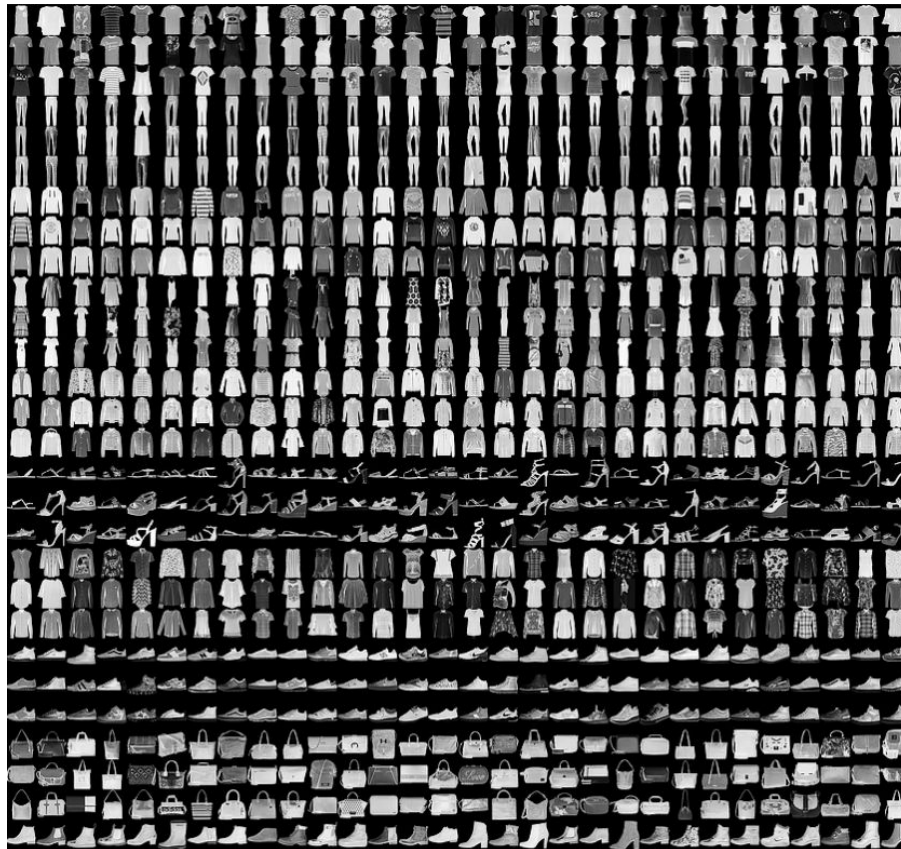
3. Perceptrón

- Funciona bien:
 - dataset linealmente separables
 - es rápido
- No tan bueno:
 - no elige las mejores fronteras de decisión



Notebook

Dataset fashion.mnist



Label	Description
0	T-shirt/top
1	Trouser
2	Pullover
3	Dress
4	Coat
5	Sandal
6	Shirt
7	Sneaker
8	Bag
9	Ankle boot