Clasificación Lineal

Dr. Mauricio Toledo-Acosta
mauricio.toledo@unison.mx

Diplomado Ciencia de Datos con Python

Table of Contents

1 Introducción

2 Modelos Lineales de Clasificación
- Funciones de perdida
- Métricas de desempeño

3 Clasficación Multiclase

4 Mínimos cuadrados

¿Qué es la clasificación?

¿Qué tienen en común las siguientes tareas?

(a)

(b)

Tarea de Clasificación

Clasificación

Problema de modelación en el cual se predice una etiqueta para cada dato de entrada.

Es decir, asignar etiquetas a puntos.

- Clasificación Binaria: Dos etiquetas, mutuamente exclusivas.

Tarea de Clasificación

Clasificación

Problema de modelación en el cual se predice una etiqueta para cada dato de entrada.

Es decir, asignar etiquetas a puntos.

- Clasificación Binaria: Dos etiquetas, mutuamente exclusivas.
- Clasificación Multi-clase: Varias etiquetas mutuamente excluyentes.

Tarea de Clasificación

Clasificación

Problema de modelación en el cual se predice una etiqueta para cada dato de entrada.

Es decir, asignar etiquetas a puntos.

- Clasificación Binaria: Dos etiquetas, mutuamente exclusivas.
- Clasificación Multi-clase: Varias etiquetas mutuamente excluyentes.
- Clasificación Multi-etiqueta: Cada instancia tiene varias etiquetas.

Tarea de Clasificación

Datos de entrada:

$$X = \underbrace{\{x_1, ..., x_n\}}_{\text{Datos de entrada}} \subset \mathbb{R}^D, \quad Y = \underbrace{\{y_1, ..., y_n\}}_{\text{Etiqueta de cada dato}}$$

Tarea de Clasificación

Datos de entrada:

$$X = \underbrace{\{x_1, \ldots, x_n\}}_{\text{Datos de entrada}} \subset \mathbb{R}^D, \quad Y = \underbrace{\{y_1, \ldots, y_n\}}_{\text{Etiqueta de cada dato}}$$

Un clasificador asigna etiquetas a cada dato de entrada. Es decir, separa los datos de entrada en regiones de decisión cuyos límites se llaman fronteras de decisión.

Tarea de Clasificación

Datos de entrada:

$$X = \underbrace{\{x_1, ..., x_n\}}_{\text{Datos de entrada}} \subset \mathbb{R}^D, \quad Y = \underbrace{\{y_1, ..., y_n\}}_{\text{Etiqueta de cada dato}}$$

Un clasificador asigna etiquetas a cada dato de entrada. Es decir, separa los datos de entrada en regiones de decisión cuyos límites se llaman fronteras de decisión. Hay varios métodos:

- SVM (Support Vector Machine)
- Regresión Logística
- Naive-Bayes
- Perceptron
- Redes Neuronales

Table of Contents

1. Introducción
2. Modelos Lineales de Clasificación
   - Funciones de perdida
   - Métricas de desempeño
3. Clasficación Multiclase
4. Mínimos cuadrados

Modelos Lineales de Clasificación

$$X = \{x_1, ..., x_n\} \subset \mathbb{R}^D.$$

El clasificador tiene la forma

$$g(X) = w^T \cdot x + w_0$$
$$y(x) = f(g(x))$$

$w \in \mathbb{R}^D$ es el vector de pesos y $w_0 \in \mathbb{R}$ el sesgo (bias). La función f es la función de activiación.

Modelos Lineales de Clasificación

$$X = \{x_1, ..., x_n\} \subset \mathbb{R}^D.$$

El clasificador tiene la forma

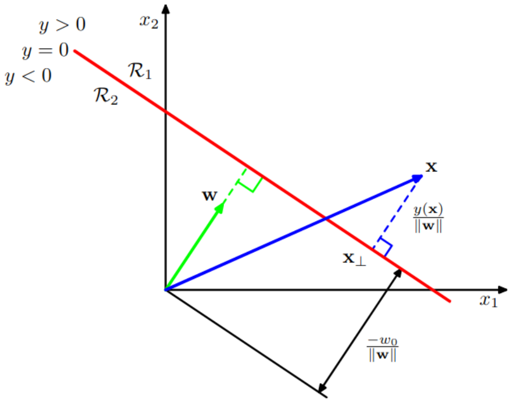
$$g(X) = w^T \cdot x + w_0$$
$$y(x) = f(g(x))$$

$w \in \mathbb{R}^D$ es el vector de pesos y $w_0 \in \mathbb{R}$ el sesgo (bias). La función f es la función de activiación. Un ejemplo básico de f es la función signo:

$$f(z) = \begin{cases} 1, & z \geq 0 \\ -1, & z < 0. \end{cases}$$

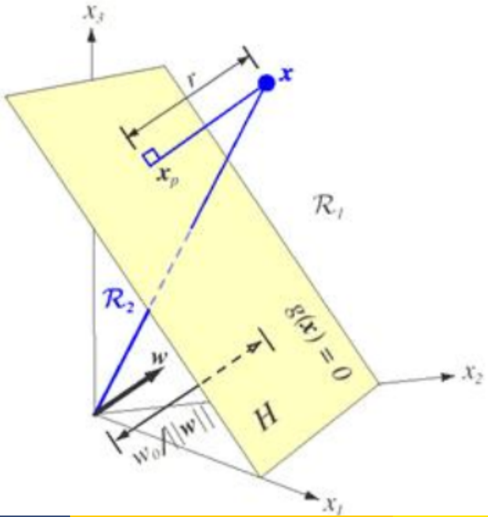
Modelos Lineales de Clasificación

Los puntos x que satisfacen $g(x) = 0$ forman un hiperplano en $\mathbb{R}^D$.



Modelos Lineales de Clasificación

Los puntos x que satisfacen $g(x) = 0$ forman un hiperplano en $\mathbb{R}^D$.



Observación sobre la dimensión

Datos son D-dimensionales

Observación sobre la dimensión

Datos son D-dimensionales

$$\downarrow$$

$g(x)$ representa un hiper-plano en $D + 1$-dimensiones

Observación sobre la dimensión

Datos son D-dimensionales

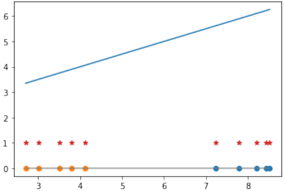
$$\downarrow$$

$g(x)$ representa un hiper-plano en $D+1$-dimensiones

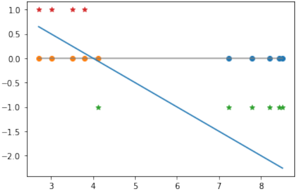
$$\downarrow$$

$g(x) = 0$ es un hiper-plano en D-dimensiones que divide a $\mathbb{R}^D$ en dos regiones.

Modelos Lineales de Clasificación: Ejemplo 1D

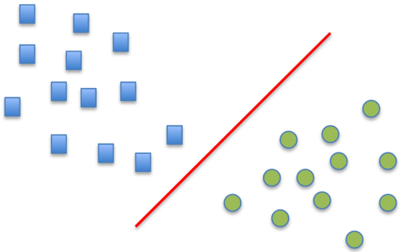


(a) Mala clasificación



(b) Mejor Clasificación

Modelos Lineales de Clasificación: Ejemplo 2D



Aprendiendo el clasificador

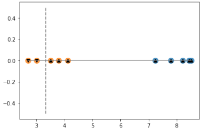
- Aprender consiste en estimar una *buena* frontera de decisión (FD).

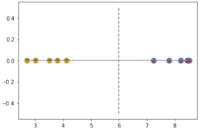
Aprendiendo el clasificador

- Aprender consiste en estimar una *buena* frontera de decisión (FD).
- Es necesario encontrar una dirección w y una ubicación w_0.

Aprendiendo el clasificador

- Aprender consiste en estimar una *buena* frontera de decisión (FD).
- Es necesario encontrar una dirección w y una ubicación w_0.
- Es necesario definir que quiere decir que la FD sea *buena*.





Aprendiendo el clasificador

- Aprender consiste en estimar una *buena* frontera de decisión (FD).
- Es necesario encontrar una dirección w y una ubicación w_0.
- Es necesario definir que quiere decir que la FD sea *buena*.

- Una vez que hemos hecho una estimación, ¿cuál es el costo de equivocarnos?

Funciones de perdida

Una función de perdida $L(y, t)$ cuantifica la perdida en la que se incurre por predecir y cuando la respuesta correcta es t. Se usa como medida de cuán bueno es un modelo de clasificación en términos de poder predecir el resultado esperado.

- 0-1

$$L(y, t) = \begin{cases} 1, & y \neq t \\ 0, & y = t. \end{cases}$$

Funciones de perdida

Una función de perdida $L(y,t)$ cuantifica la perdida en la que se incurre por predecir y cuando la respuesta correcta es t. Se usa como medida de cuán bueno es un modelo de clasificación en términos de poder predecir el resultado esperado.

- 0-1

$$L(y,t) = \begin{cases} 1, & y \neq t \\ 0, & y = t. \end{cases}$$

- Binaria asimétrica

$$L(y,t) = \begin{cases} \alpha, & y = 1,\ t = 0 \\ \beta, & y = 0,\ t = 1 \\ 0, & y = t. \end{cases}$$

Funciones de perdida

- Perdida cuadrática (MSE)

$$L(y, t) = (t - y)^2.$$

Funciones de perdida

- Perdida cuadrática (MSE)

$$L(y,t) = (t-y)^2.$$

- Error absoluto (MAE)

$$L(y,t) = |t-y|.$$

Matriz de Confusión

		Predicted condition	
	Total population = P + N	**Positive (PP)**	**Negative (PN)**
Actual condition	**Positive (P)**	**True positive (TP)**	**False negative (FN)**
	Negative (N)	**False positive (FP)**	**True negative (TN)**

Métricas de desempeño

- **Accuracy**: De todos la población, ¿cuántos predije correctamente?

$$A = \frac{TP + TN}{\text{Total}}.$$

- **Recall**: De todos la población positiva, ¿cuántos predije correctamente como positivos?

$$R = \frac{TP}{TP + FN}.$$

- **Precision**: De todos los que predije como positivos, ¿cuántos son realmente positivos?

$$P = \frac{TP}{TP + FP}.$$

- **F1 score**: Media armónica de la precisión y el recall:

$$2\frac{P \cdot R}{P + R}$$

Ejemplo

Tenemos la siguiente población $\{++----\}$:

- El clasificador predice todo como $-$:

real	$+$	$+$	-	-	-	-
predicho	-	-	-	-	-	-

Accuracy: 0.66, Recall: 0, Precision: 0.

- El clasificador predice todo como $+$:

real	$+$	$+$	-	-	-	-
predicho	$+$	$+$	$+$	$+$	$+$	$+$

Accuracy: 0.33, Recall: 1, Precision: 0.33.

Table of Contents

1 Introducción

2 Modelos Lineales de Clasificación
- Funciones de perdida
- Métricas de desempeño

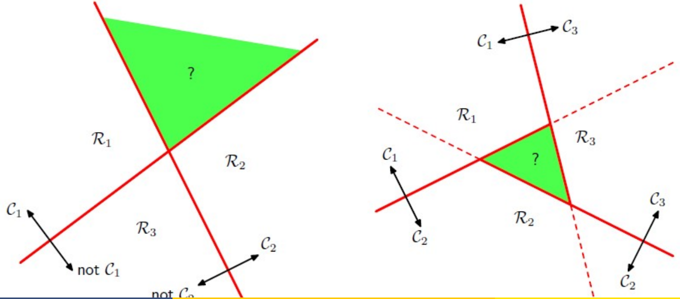
3 Clasficación Multiclase

4 Mínimos cuadrados

Clasificación Multiclase

Si tenemos k clases diferentes. Hay varios enfoques para lidiar con este problema usando discriminantes lineales:

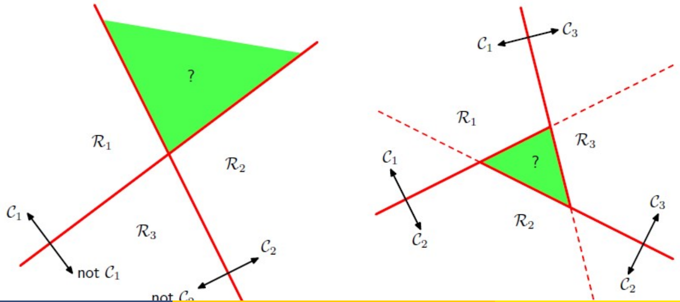
- *One vs all*. Considerar k problemas de clasificación binarias, el j-simo problema consiste en comparar la clase j contra lo que no pertenece a la clase j.



Clasificación Multiclase

Si tenemos k clases diferentes. Hay varios enfoques para lidiar con este problema usando discriminantes lineales:

- *One vs all*. Considerar k problemas de clasificación binarias, el j-simo problema consiste en comparar la clase j contra lo que no pertenece a la clase j.
- *One vs one*. Considerar todas las posibles comparaciones, clase i contra la clase j.



Clasficación Multiclase

Para evitar las regiones ambiguas hacemos:

$$g_i(x) = w_i^T \cdot x + w_{i,0}, \quad i = 1, ..., k$$

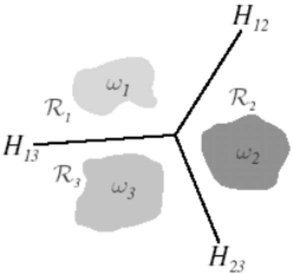
y asignamos x a la clase j si $g_j(x) > g_i(x)$ para todos $i = 1, ..., k$, $i \neq j$. Si hay ambigüedad, se deja sin asignar.

Clasficación Multiclase

Para evitar las regiones ambiguas hacemos:

$$g_i(x) = w_i^T \cdot x + w_{i,0}, \quad i = 1, ..., k$$

y asignamos x a la clase j si $g_j(x) > g_i(x)$ para todos $i = 1, ..., k$, $i \neq j$. Si hay ambigüedad, se deja sin asignar.
Este clasificador forma k regiones



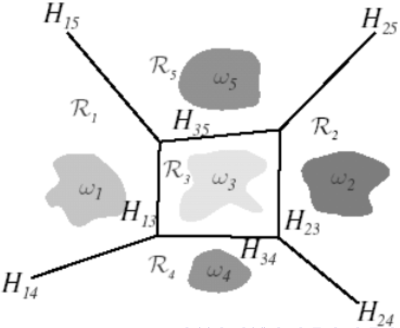


Table of Contents

1 Introducción

2 Modelos Lineales de Clasificación
- Funciones de perdida
- Métricas de desempeño

3 Clasficación Multiclase

4 Mínimos cuadrados

Planteamiento

Cada clase C_j se describe por su propio modelo lineal:

$$y_j(x) = w_j^T \cdot x + w_{j,0}$$

donde $j = 1, \ldots, k$. Podemos agrupar los términos para escribir usando notación vectorial:

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

Podemos encontramos $\mathbf{W}$ usando mínimos cuadrados.

Ejemplo

Si tenemos tres clases para un conjunto de datos en $\mathbb{R}^2$, con etiquetas $y = \{2, 0, 1, ...\}$, tenemos tres modelos

$$g_1(x) = \begin{pmatrix} w_{11} & w_{12} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + w_0^1 = \begin{pmatrix} w_{11} & w_{12} & w_0^1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

$$g_2(x) = \begin{pmatrix} w_{21} & w_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + w_0^2 = \begin{pmatrix} w_{21} & w_{22} & w_0^2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

$$g_3(x) = \begin{pmatrix} w_{31} & w_{32} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + w_0^3 = \begin{pmatrix} w_{31} & w_{32} & w_0^3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

$$\mathbf{g}(x) = \begin{pmatrix} w_{11} & w_{12} & w_0^1 \\ w_{21} & w_{22} & w_0^2 \\ w_{31} & w_{32} & w_0^3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ 1 \end{pmatrix}$$

Solución

Consideramos la matriz X de los N puntos del conjunto de entrenamiento en $\mathbb{R}^D$:

$$X = \begin{pmatrix} x_1 \\ \cdots \\ x_N \end{pmatrix} = \begin{pmatrix} x_1^{(1)} & \cdots & x_D^{(1)} \\ \cdots & \cdots & \cdots \\ x_1^{(N)} & \cdots & x_D^{(N)} \end{pmatrix}$$

Solución

Obtenemos la matriz $\tilde{X}$

$$\tilde{X} = \begin{pmatrix} x_1^{(1)} & \cdots & x_D^{(1)} & 1 \\ \cdots & \cdots & \cdots & \cdots \\ x_1^{(N)} & \cdots & x_D^{(N)} & 1 \end{pmatrix}$$

Solución

Las dimensiones son:

$$\tilde{X} = \underbrace{\left.\begin{pmatrix} x_1^{(1)} & \cdots & x_D^{(1)} & 1 \\ \cdots & \cdots & \cdots & \cdots \\ x_1^{(N)} & \cdots & x_D^{(N)} & 1 \end{pmatrix}\right\} N}_{D+1}$$

Solución

Las dimensiones son:

$$\tilde{X} = \underbrace{\left.\begin{pmatrix} x_1^{(1)} & \dots & x_D^{(1)} & 1 \\ \dots & \dots & \dots & \dots \\ x_1^{(N)} & \dots & x_D^{(N)} & 1 \end{pmatrix}\right\} N,}_{D+1} \quad t = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

Solución

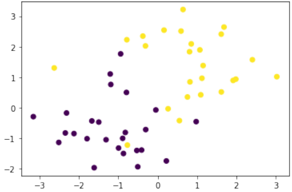
Las dimensiones son:

$$\tilde{X} = \underbrace{\left.\begin{pmatrix} x_1^{(1)} & \cdots & x_D^{(1)} & 1 \\ \cdots & \cdots & \cdots & \cdots \\ x_1^{(N)} & \cdots & x_D^{(N)} & 1 \end{pmatrix}\right\} N}_{D+1}, \quad t = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

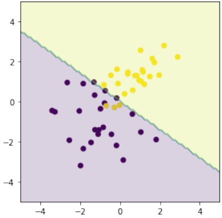
Usando OLS obtenemos la matriz de pesos $\tilde{W}$:

$$\tilde{W} = \left(\tilde{X}^T \tilde{X}\right)^{-1} \tilde{X}^T t$$

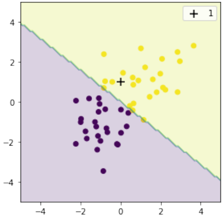
Ejemplo



(a) El conjunto de datos de entrenamiento



(b) La frontera de decisión y ambas regiones



(c) Clasificamos un nuevo punto