Tema 3: Regresión logística

Aprendizaje automático

Héctor Lacueva Sacristán

Índice

Problemas de clasificación

Son problemas en los que las salidas son discretas (clases).

Clasificación binaria

Dada una entrada, la salida puede ser de dos clases:

$$y \in \{0, 1\}$$

- Clase **positiva**: 1
- Clase **negativa**: 0

Puede servir para problemas muy sencillo de clasificación:

- Email: Spam/Ham
- Transacción Online fraudulenta: Si/No
- Tumor: Maligno/Benigno

Clasificación multiclase

Dada una entrada, la salida puede pertenecer a más de dos clases:

$$y \in \{1, \cdots, C\}$$

Puede servir para problemas un poco más complejos:

- Email: trabajo / amigos / familia /...
- Reconocimiento de dígitos: 1, 2, 3, ...
- Reconocimiento de personas: Pedro, Juan, María, ...

¿Sirve la regresión lineal umbralizada? NO

Dependiendo de la función de regresión, la misma entrada podría ser clasificada tanto como una clase como otra distinta.

Regresión Logística

Es una técnica de clasificación.

Regresión logística binaria

Distingue entre dos clases. Se emplea la función logística (o Sigmoidal).

$$sig(z) = \frac{1}{1 + e^{-\theta^T x}}$$

Esta función es muy fácil de derivar:

$$der(sig(z)) = sig(z)(1 - sig(z))$$

Nuestro modelo sería:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

Con $0 \leq h_\theta(x) \leq 1$.

Se interpreta como la probabilidad de que $\hat{y} = 1$ dado x parametrizada por θ.

$$h_\theta(x) = p(y = 1|x; \theta)$$

Frontera de decisión

Frontera (línea imaginaria) a partir de la cual el resultado pasa a interpretarse de una clase. Es lineal en el espacio de los atributos de x.

Frontera de decisión lineal Pueden ser lineales, p.ej.

$$h_\theta(x) = sig(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

Frontera de decisión no lineal Si se les aplica expansión de funciones base a los atributos originales:

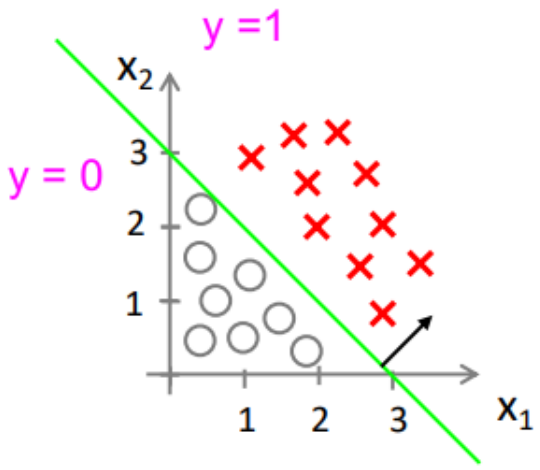
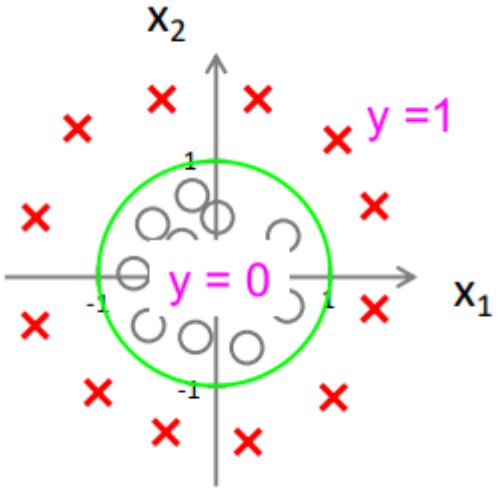
- Atributos originales:

$$x = (1, x_1, x_2, \cdots)^T$$

- Expansión de funciones base:

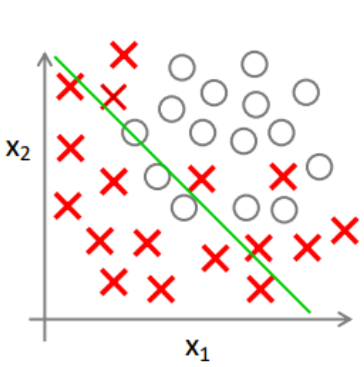
$$\phi(x) = (1, x_1, x_2, x_1^2, x_1 x_2, x_2^2, \cdots)^T$$

$$h_\theta(x) = sig(\theta^T \phi(x))$$

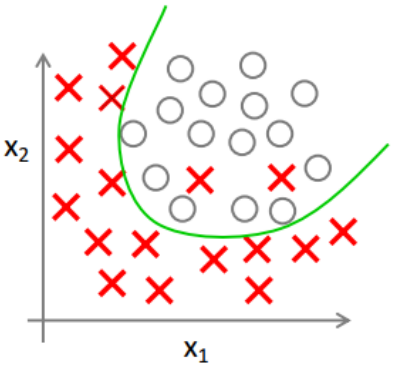
Frontera de decisión lineal	Frontera de decisión no lineal
	

Nota: La frontera de decisión es lineal en el espacio de los atributos expandidos $\phi(x)$. Pero **no lo es en el espacio original x**.

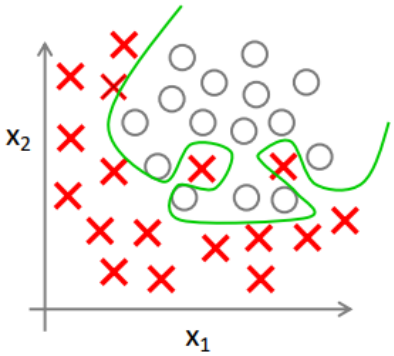
Se pueden construir fronteras arbitrariamente **complejas** pero hay que tener **cuidado con el sobre-ajuste**.



$sig(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$



$sig(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2)$



$sig(\theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^2 x_2 + \theta_4 x_1^2 x_2^2 + \theta_5 x_1^2 x_2^3 + \theta_6 x_1^3 x_2 + \dots)$

Aprendizaje en regresión logística

Dadas muestras de entrenamiento, variables de entrada, de salida, y una hipótesis, ¿Cómo aprender los valores de los parámetros θ?

Estimación por Máxima Verosimilitud (MLE) Consiste en buscar los parámetros que mejor explican los datos de entrenamiento, para ello se busca que:

$$\hat{\theta} = argmin_\theta J(\theta)$$

La función de coste tiene esta pinta:

$$J(\theta) = -\frac{1}{N}\sum_{i=1}^{N} y^{(i)} ln(h_\theta(x^{(i)})) + (1 - y^{(i)} ln(1 - h_\theta(x^{(i)})))$$

También se conoce como **binary cross-entropy**:

$$H_{ce}(p, q) = -[plogq + (1 - p)log(1 - q)]$$

Binary Cross-Entropy La función de coste para cada muestra se suele representar como:

$$J^{(i)}(\theta) = \begin{cases} -ln(h_\theta(x^{(i)})) \text{ if } y^{(i)} = 1 \\ -ln(1 - h_\theta(x^{(i)})) \text{ if } y^{(i)} = 0 \end{cases}$$

Algoritmo de descenso de gradiente Se debe de calcular el gradiente:

$$g(\theta) = der(J(\theta)) = \begin{pmatrix} \frac{1}{N}\sum_{i=1}^{N}(h_\theta(x^{(i)}))x_0^{(i)} \\ \vdots \\ \frac{1}{N}\sum_{i=1}^{N}(h_\theta(x^{(i)}))x_D^{(i)} \end{pmatrix}$$

Repetir

$$\theta_{k+1} := \theta_k - \alpha g(\theta_k) \text{ donde } \alpha \text{ es el Factor de aprendizaje.}$$

Hasta que converja.

Regresión logística regularizada Se puede aplicar regularización tanto L1 como L2 al problema para evitar problemas de sobre-ajuste o sub-ajuste. Se tomará como modelo el que menor error tenga en validación si estamos haciendo validación cruzada o test si estamos probando con entrenamiento y test.

Segundo orden vs Primer orden

El descenso de gradiente sólo utiliza la primera derivada, pero se puede calcular el Hessiano, derivada del gradiente, y calcular un mínimo. Se conoce como el Método de Paso de Newton.

Método de Newton

1. Calcular la dirección de avance d_k resolviendo:

$$H_k d_k = -g_k$$

2. Buscar el paso n_k que consiga el mayor descenso posible de J:

$$\theta_{k+1} := \theta_k + n_k d_k \text{ y además } J(\theta_{k+1}) \ll J(\theta_k)$$

Segundo orden Se usan para problemas no muy grandes:

- Newton (Calcula el Hessiano analítico)
- Quasi-Newton (Aproxima el Hessiano)
- BFGS (Aproximación de bajo rango del Hessiano)
- LBFGS (Con memoria limitada, se usa en D muy pequeños)

Primer orden Se usan para problemas muy grandes y RN:

- SGD (Descenso de gradiente estocástico)
- SAGA (Stochastic Averaged Gradient Accelerated)
- AdaGrad, RMSProp, AdaDelta, Adam (SGD precondicionado)

Regresión logística multiclase

Queremos predecir la probabilidad de cada clase.

One vs Rest (One vs All)

Para conseguir un modelo multi-clase entrenamos un clasificador para cada clase j, que estime la probabilidad de y = j. La clase resultante es la clase más probable para x. **Las probabilidades no están normalizadas**.

Regresión Logística Multinomial

Entrenamos un clasificador conjunto que estime las probabilidades de y = j. La clase resultante es la clase más probable para x. Las probabilidades ya están normalizada. Con dos clases equivale a la regresión logística binaria.

Estimación de Máxima Verosimilitud (MLE)

Como antes aparece Cross-Entropy

Algoritmo de Descenso de Gradiente

Igual que antes

Clasificación con RN

Clasificación binaria

Capa de salida sigmoidal. Aprende automáticamente atributos para la clasificación.

Clasificación multi-clase

Capa de salida softmax. Aprende automáticamente atributos para la clasificación.

Métricas para clasificación

Tasa de acierto (Accuracy)

$$A_{cv}(\theta) = \frac{aciertos}{total}$$

Tasa de error

$$E_{cv}(\theta) = 1 - A_{cv}(\theta) = \frac{fallos}{total}$$

Precision / Recall

Por convención: y = 1 es la clase rara que queremos detectar

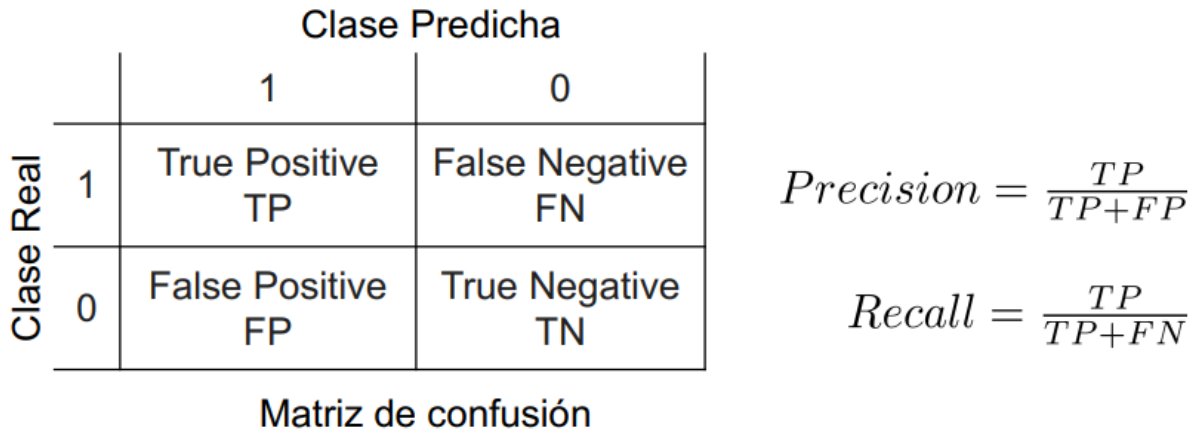


Figure 1: Matriz de confusión

Precision

De los que damos como positivos, qué fracción lo son realmente.

$$Precision = \frac{TP}{TP + FP}$$

Recall o TPR (True Positive Rate)

De los casos positivos, qué fracción detectamos.

$$Recall = \frac{TP}{TP + FN}$$

Specificity o TNR (True negative Rate)

$$Specificity = \frac{TN}{TN + FP}$$

Curva Precision-Recall

La salida predicha depende del umbral con el que se compare. Cambiando el umbral podemos mejorar la precisión a costa del recall, o viceversa.

Curva ROC

Enfrenta el TPR con el FPR, cuanto más se aleja de la diagonal (puntuación de un clasificador random) mejor es el modelo.

$F_1 Score$

Para seleccionar el mejor modelo en validación cruzada necesitamos una métrica única:

$$F_1 = 2\frac{P \times R}{P + R}$$

$F_\beta Score$

Si se le quiere dar más importancia a la precisión o el recall mejor este:

$$F_\beta = (1 + \beta^2)\frac{P \times R}{\beta^2 P + R}$$

- F_1 da el mismo peso a precision y recall.
- $F_{0.5}$ da más importancia a **precision**.
- F_2 da más importancia a **recall**.

Tasa de aciertos balanceada

Promedio del recall de las distintas clases

- Equivale a tasa de aciertos, si se ponderan las muestras con la inversa de la frecuencia de su clase verdadera.
- Si el dataset es balanceado, es igual a la tasa de aciertos.

Métricas para clasificación binaria

- Problema balanceado: Accuracy
- Problema desbalanceado: BalancedAccuracy

Para seleccionar modelos:

- Balanced accuracy.
- Sin preferencia de P/R: F1_score
- Con preferencia:
  - $F_{0.5}$ para más precision
  - F_2 para más recall

Métricas para clasificación multi-clase

- Matriz de confusión, junto a precision y recall.

Para selección de modelos:

- Pb. balanceado:
  - Accuracy.
- Pb. Desbalanceado:
  - Balanced_Accuracy.
  - F1 promedio de las clases.