05MIAR-Aprendizaje supervisado

Tema 4 – Algoritmos de Regresión

Curso Abril 23/24

Universidad Internacional de Valencia

18/07/2023

De:
Planeta Formación y Universidades

¿Dónde estudiar el Tema 4?

Manual de la asignatura – Capítulo 3

Scripts del Tema 4

Índice

Índice

Introducción

“Todos los algoritmos de regresión lineal se basan en **encontrar relaciones lineales entre los atributos y la clase**. Dado que **la linealidad es una característica inherente a muchísimos problemas en la naturaleza**, las técnicas de regresión lineal han sido utilizadas con éxito en numerosos problemas. “

- **Regresión Simple:** un atributo de entrada
- **Regresión Múltiple:** admite varios atributos.

“El algoritmo clásico de **regresión** lineal, y el más utilizado, se denomina **ordinary least squares (OLS)**. Es apto tanto para regresión simple como múltiple … “ “… un modelo que consta de una función matemática muy sencilla: una **combinación lineal de los atributos.** ”

Atributos

objetivo

$$\boxed{\hat{y}} = w_0 + w_1 \cdot x_1 + \ldots + w_p \cdot x_p$$

Coeficientes

Entrenamiento de OLS

Ordinary least squares (OLS): el objetivo es **estimar los coeficientes**. "OLS comienza, entonces, un proceso **iterativo** de búsqueda de los valores óptimos para los coeficientes y el término independiente del modelo. "

$$\hat{y} = w_0 + w_1 \cdot x_1 + \dots + w_p \cdot x_p$$

"Todo proceso de optimización iterativo requiere de una **función objetivo** (también función de bondad o de **coste**) para evaluar la bondad de una posible solución, y de un esquema de búsqueda que genere nuevos valores para evaluar en la siguiente iteración. En el caso de OLS, la función objetivo es la **suma de errores al cuadrado** (**MSE** de mean squared error en inglés)… " "… **minimizan** su valor."

$$f(w_0, \dots, w_p) = \frac{1}{2n} \cdot \sum_{i=1}^{n} \left(\left(w_0 + \sum_{j=1}^{p} w_j \cdot x_{i,j} \right) - y_i \right)^2$$

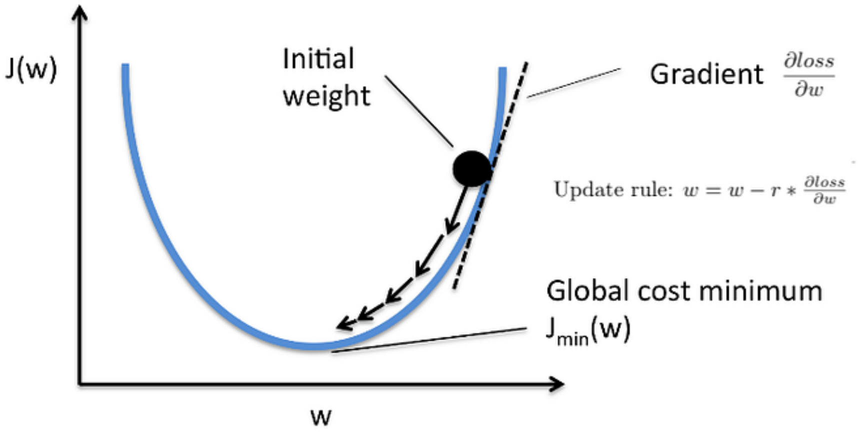
"Cuanto **menor** sea el valor de la función ***f***, **mejor** será el **ajuste** del modelo a los datos de **entrenamiento**, y se espera que tendrá mayor capacidad predictiva en ejemplos de test."

Entrenamiento de OLS

Ordinary least squares (OLS): el objetivo es **estimar los coeficientes**. "OLS comienza, entonces, un proceso **iterativo** de búsqueda de los valores óptimos para los coeficientes y el término independiente del modelo. "

$$\hat{y} = w_0 + w_1 \cdot x_1 + \ldots + w_p \cdot x_p \qquad f(w_0, \ldots, w_p) = \frac{1}{2n} \cdot \sum_{i=1}^{n} \left(\left(w_0 + \sum_{j=1}^{p} w_j \cdot x_{i,j} \right) - y_i \right)^2$$

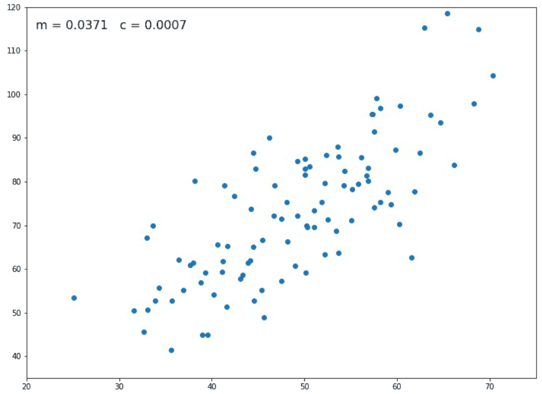
Una forma popular de realizar dicha aproximación es por **descenso de gradientes**.



$$\frac{\partial}{\partial m} = \frac{2}{N} \sum_{i=1}^{N} -x_i(y_i - (mx_i + b))$$

$$\frac{\partial}{\partial b} = \frac{2}{N} \sum_{i=1}^{N} -(y_i - (mx_i + b))$$

caso con solo 1 atributo



Entrenamiento de OLS

Ordinary least squares (OLS): el objetivo es **estimar los coeficientes**. "OLS comienza, entonces, un proceso **iterativo** de búsqueda de los valores óptimos para los coeficientes y el término independiente del modelo. "

$$\hat{y} = w_0 + w_1 \cdot x_1 + \ldots + w_p \cdot x_p$$

$$f(w_0, \ldots, w_p) = \frac{1}{2n} \cdot \sum_{i=1}^{n} \left(\left(w_0 + \sum_{j=1}^{p} w_j \cdot x_{i,j} \right) - y_i \right)^2$$

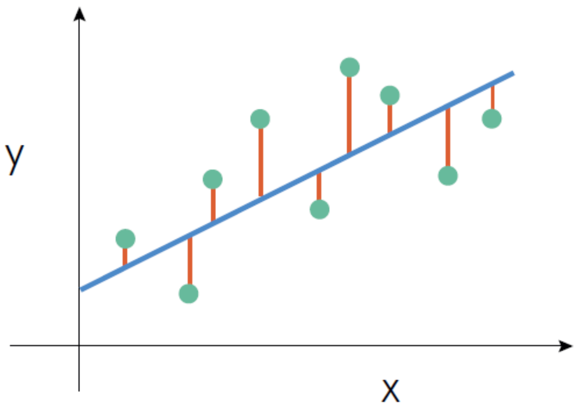


Figura 5. Escenario de cálculo de la función objetivo en regresión lineal.

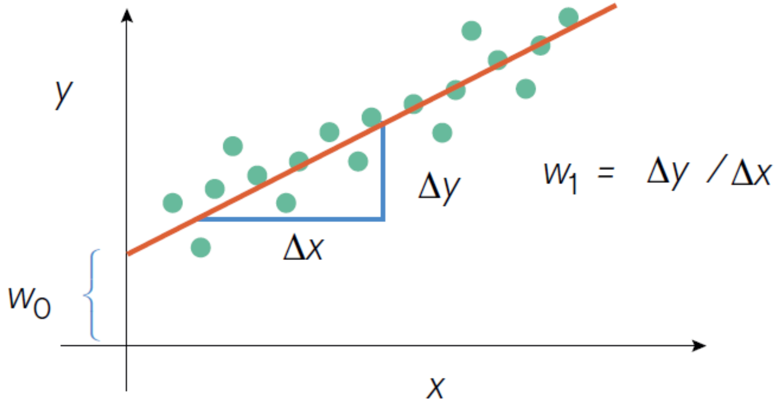


Figura 6. Interpretación geométrica de los coeficientes del modelo de regresión lineal.

Evaluación

“Definimos a continuación una nueva métrica de evaluación para regresión que debería usarse solo para evaluar modelos de lineales, como OLS. “

“EL coeficiente de terminación, R2, recoge la cantidad de variabilidad de la clase (con respecto a su media aritmética) que el modelo es capaz de predecir con respecto al total de variabilidad de la clase. ”

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

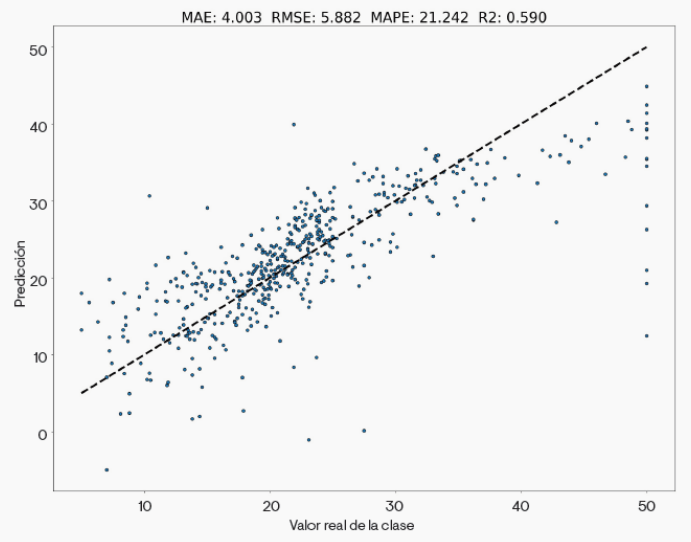


Figura 7. Gráfica generada por el Programa 12.

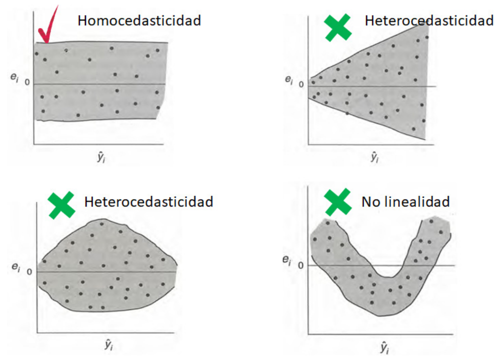
Ventajas e inconvenientes

Ventajas

- Simplicidad del modelo.
- Interpretabilidad del modelo.
- Gran eficacia en problemas con naturaleza lineal.
- Tiempo de ejecución de entrenamiento bajo.
- Tiempos de inferencia instantáneos.

Limitaciones:

- Independencia de los atributos.
- Distribución normal de los datos.
- Relación lineal entre atributos y clase.
- La homocedasticidad de los datos.



Universidad Internacional de Valencia

Mejoras en el entrenamiento

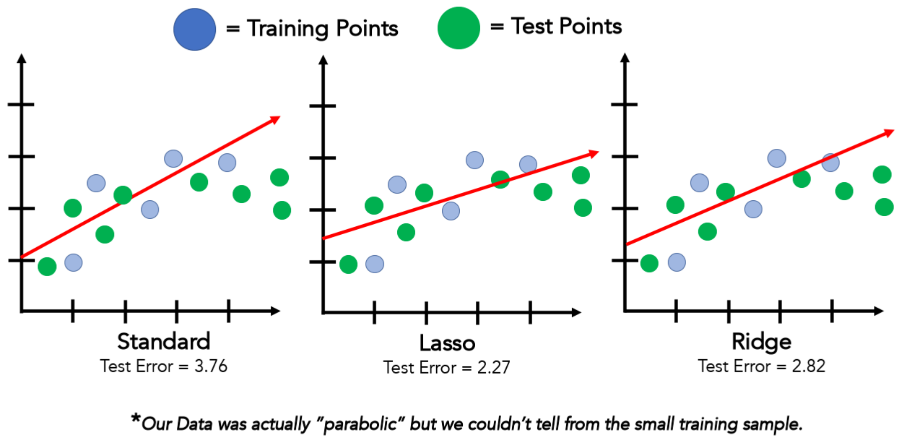
“Existen en la literatura varios algoritmos que pueden **mejorar** el algoritmo **OLS** en **determinados escenarios**. En concreto, las variantes **Ridge y Lasso** incorporan **nuevos términos a la función objetivo** utilizada para **ajustar** los **coeficientes del modelo lineal**. Estos nuevos términos permiten **evitar el sobreajuste** del modelo a los datos, especialmente cuando estos poseen **valores anómalos**.”

- **Lasso – L1 penalty/regularization**

$$\sum_{i=1}^{M}(y_i-\hat{y}_i)^2=\sum_{i=1}^{M}\left(y_i-\sum_{j=0}^{p}w_j\times x_{ij}\right)^2+\lambda\sum_{j=0}^{p}|w_j|$$

- **Ridge – L2 penalty/regularization**

$$\sum_{i=1}^{M}(y_i-\hat{y}_i)^2=\sum_{i=1}^{M}\left(y_i-\sum_{j=0}^{p}w_j\times x_{ij}\right)^2+\lambda\sum_{j=0}^{p}w_j^2$$



https://thaddeus-segura.com/lasso-ridge/

Índice

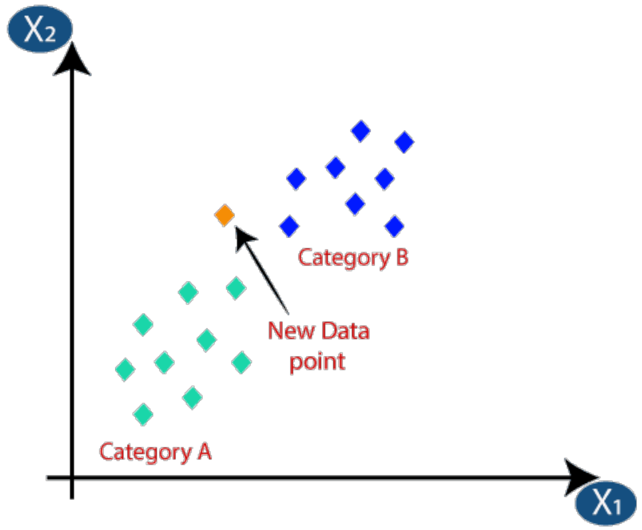
Introducción

"… **vecinos más cercanos** (**KNN** de *k nearest neighbors* en inglés) … premisa de que **ejemplos similares en sus atributos presentan también un comportamiento similar en el valor de sus clases**."

"… este algoritmo de aprendizaje automático para problemas de **regresión**, también puede ser aplicado tanto en **clasificación** como en problemas **no supervisados**."

"La idea fundamental de la técnica de los vecinos más cercanos para regresión consiste en **producir como predicción el promedio de las clases de los ejemplos de entrenamiento más parecidos (vecinos) al ejemplo de test que hay que predecir**..."

¿Cómo se define similitud?



Universidad Internacional de Valencia

Entrenamiento e inferencia en kNN

- **Entrenamiento**

"El proceso de entrenamiento de KNN es muy sencillo, pues tan solo consiste en **almacenar el conjunto de datos de entrenamiento**, para poder ser consultado posteriormente en la fase de predicción. KNN **no elabora ningún modelo a partir de los datos**. Por esta razón, el proceso de **entrenamiento apenas consume tiempo de ejecución**."

- **Inferencia**

Para llevar a cabo las predicciones, el algoritmo KNN clásico se basa en los dos siguientes elementos configurables: el número k de vecinos más cercanos y la función de distancia *d* (para poder evaluar la similitud entre los ejemplos).

1) **Computar distancias entre muestras de entrenamiento y test**
2) **Seleccionar k vecinos que minimizan la distancia**
3) **Obtener como predicción el promedio en dichos vecinos**

$$d(e_i, e_{v_1}) \leq d(e_i, e_{v_2}) \leq \dots \leq d(e_i, e_{v_k}) \leq d(e_i, e_j)$$

$$1 \leq j \leq n \wedge j \neq v_l \wedge i \neq j \quad \forall l \in \{1, \dots, k\}$$

$$v_l \in \{1, \dots, n\} \quad v_l \neq i$$

$$\hat{y}_i = \frac{1}{k} \cdot \sum_{j=1}^{k} y_{v_j}$$

Universidad Internacional de Valencia

Entrenamiento e inferencia en kNN

- **Inferencia**

Para llevar a cabo las predicciones, el algoritmo KNN clásico se basa en los dos siguientes elementos configurables: el número k de vecinos más cercanos y la función de distancia *d* (para poder evaluar la similitud entre los ejemplos).

1) **Computar distancias entre muestras de entrenamiento y test**
2) **Seleccionar k vecinos que minimizan la distancia**
3) **Obtener como predicción el promedio en dichos vecinos**

$$d(e_i, e_{v_1}) \leq d(e_i, e_{v_2}) \leq \ldots \leq d(e_i, e_{v_k}) \leq d(e_i, e_j)$$

$$1 \leq j \leq n \wedge j \neq v_l \wedge i \neq j \quad \forall l \in \{1, \ldots, k\}$$

$$v_l \in \{1, \ldots, n\} \quad v_l \neq i$$

$$\hat{y}_i = \frac{1}{k} \cdot \sum_{j=1}^{k} y_{v_j}$$

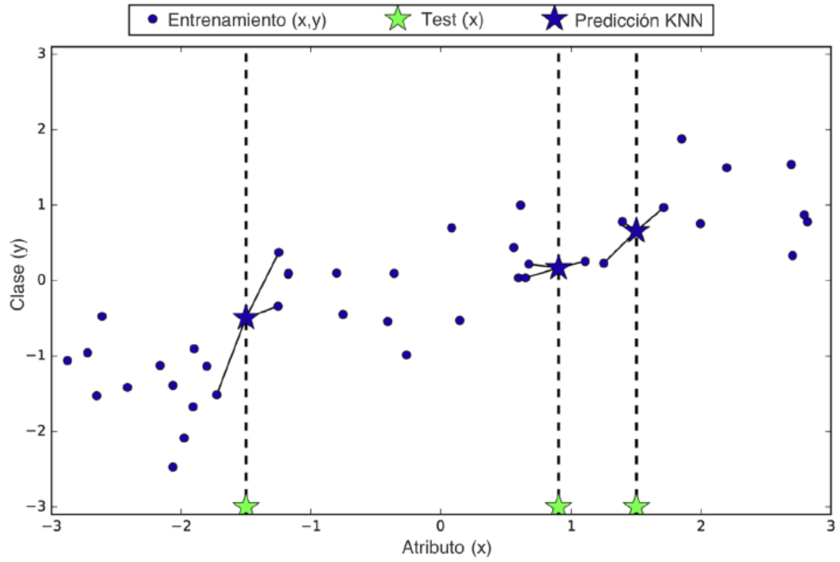


Figura 9. Predicción de KNN en regresión con un único atributo (*x*). Adaptado de *Introduction to Machine Learning with Python* (p. 42), por A. C. Mueller y S. Guido, 2016, Sebastopol: O'Reilly.

Acerca de la función de distancia

"… existen **numerosas funciones de distancia** propuestas en la literatura. En este capítulo usaremos la distancia de **Minkowski** … "

$$\text{Minkowski}(e_i, e_j) = \left(\sum_{d=1}^{p} |x_{i,d} - x_{j,d}|^q \right)^{1/q}$$

"Cuando ***q*** **= 2**, la distancia de Minkowski también se conoce como **distancia euclídea**, tal como se muestra en la siguiente ecuación."

$$\text{Euclidea}(e_i, e_j) = \sqrt{\sum_{d=1}^{p} |x_{i,d} - x_{j,d}|^2}$$

"La función de distancia es **un elemento crítico de KNN** que afecta en gran medida a la bondad de las predicciones. **Cada problema puede requerir una función de distancia más adecuada a la naturaleza y distribución de los datos."**

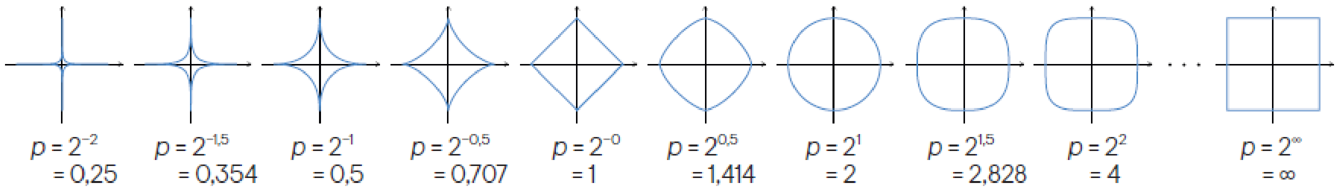


Figura 8. Distancia de Minkowski para diferentes valores de *q*. Por Waldir bajo licencia CC BY-SA 3.0. Recuperado de https://commons.wikimedia.org/wiki/File:2D_unit_balls.svg

Acerca de la función de distancia

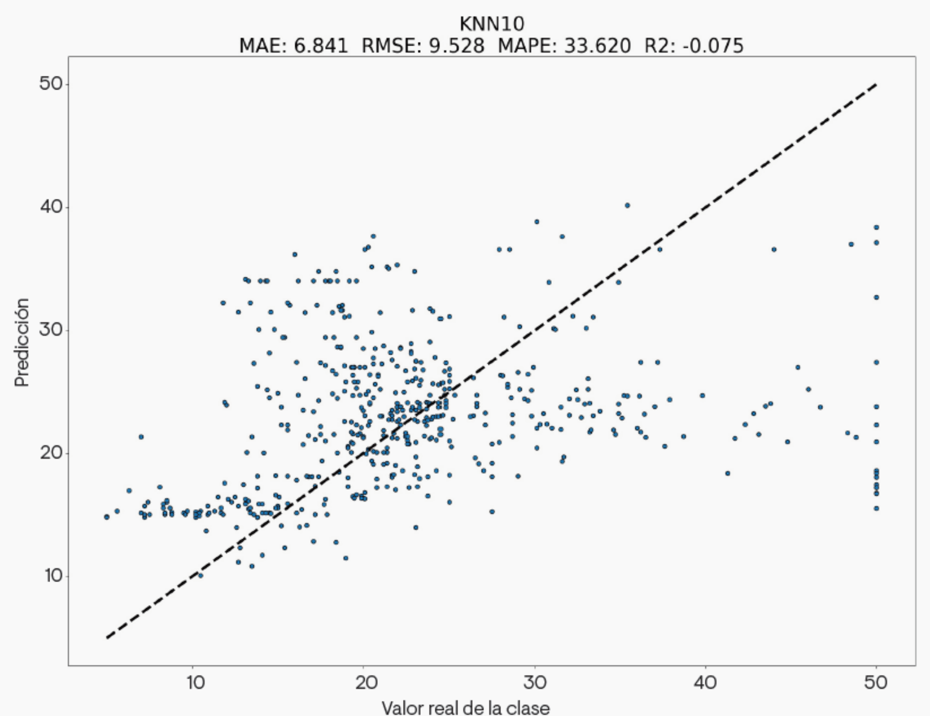
"Cualquier función de distancia que se pretenda usar debería cumplir las cuatro siguientes **propiedades matemáticas**:"

- **No negatividad** $d(e_i, e_j) \geq 0$
- **Identidad** $d(e_i, e_j) = 0 \Leftrightarrow e_i = e_j$
- **Simetría** $d(e_i, e_j) = d(e_j, e_i)$
- **Desigualdad triangular** $d(e_i, e_r) + d(e_r, e_j) \geq d(e_i, e_j)$

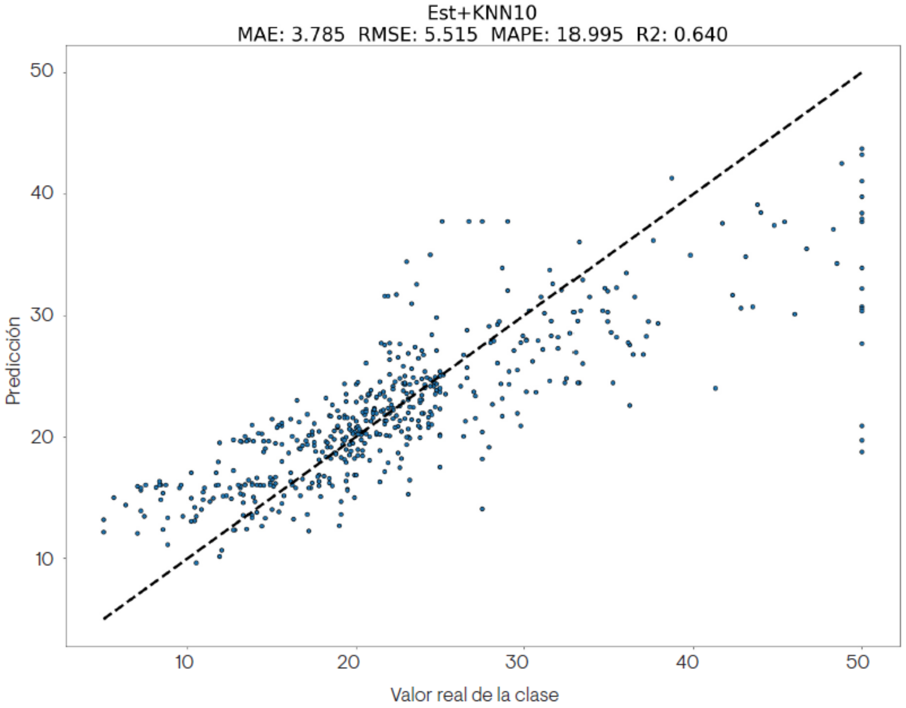
La distancia de **Minkowski** solo cumple las cuatro anteriores propiedades si **q > 1**.

Acerca de la normalización de las variables

- **Sin estandarizar:**



- **Estandarizando:**



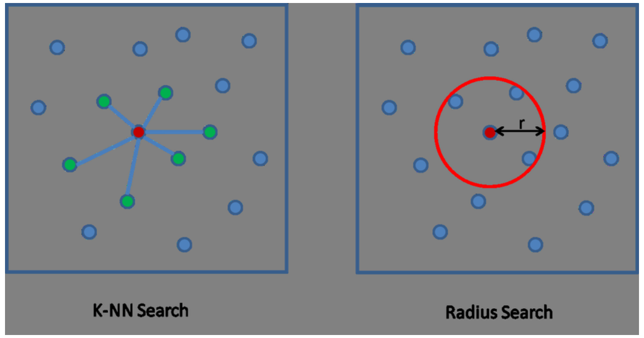
Factores de éxito en kNN

"… el algoritmo KNN se apoya en ciertos aspectos que afectan enormemente a su funcionamiento … "

- **La escala de valores de los atributos.**
- **La medida de similitud entre ejemplos:**
  - **La función de distancia.**
  - **La ponderación de los atributos.**

$$\text{Minkowski}(e_i, e_j) = \left(\frac{1}{\sum w} \cdot \sum_{d=1}^{p} w_d \left| x_{i,d} - x_{j,d} \right|^q \right)^{1/q}$$

- **La selección de vecinos más cercanos.**
  - **Ejemplos con menor distancia.**
  - **Todos los elementos con d < radio *r*.**



- **El cálculo de la predicción a partir de los vecinos más cercanos:**
  - **Función de resumen.**
  - **Ponderación de vecinos.**

$$\hat{y}_i = \frac{1}{\sum \mu} \cdot \sum_{j=1}^{k} \mu_j \cdot y_{v_j}$$

Ventajas e Inconvenientes

Ventajas

- Simplicidad del modelo.
- Interpretabilidad.
- Tiempo de entrenamiento.

Limitaciones:

- Tiempo de inferencia.
- Falta de generalización.
- Atributos relevantes y en escala.

Universidad Internacional de Valencia

universidadviu.com

De:
Planeta Formación y Universidades