



ARISA Learning Material

Educational Profile and EQF level: DATA SCIENTIST – EQF 6

PLO: 1, 2, 3, 4, 5

Learning Unit (LU): MACHINE LEARNING: SUPERVISED

Topic: 3. OVERFITTING



www.aiskills.eu

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them.

Copyright © 2024 by the Artificial Intelligence Skills Alliance

All learning materials (including Intellectual Property Rights) generated in the framework of the ARISA project are made freely available to the public under an open license [Creative Commons Attribution–NonCommercial](https://creativecommons.org/licenses/by-nc/4.0/) (CC BY-NC 4.0).

ARISA Learning Material 2024

This material is a draft version and is subject to change after review coordinated by the European Education and Culture Executive Agency (EACEA).

Authors: Universidad Internacional de La Rioja (UNIR)

Disclaimer: This learning material has been developed under the Erasmus+ project ARISA (Artificial Intelligence Skills Alliance) which aims to skill, upskill, and reskill individuals into high-demand software roles across the EU.



Co-funded by
the European Union

This project has been funded with support from the European Commission. The material reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

- About ARISA

- The Artificial Intelligence Skills Alliance (ARISA) is a four-year transnational project funded under the EU's Erasmus+ programme. It delivers a strategic approach to sectoral cooperation on the development of Artificial Intelligence (AI) skills in Europe.
- ARISA fast-tracks the upskilling and reskilling of employees, job seekers, business leaders, and policymakers into AI-related professions to open Europe to new business opportunities.
- ARISA regroups leading ICT representative bodies, education and training providers, qualification regulatory bodies, and a broad selection of stakeholders and social partners across the industry.

[ARISA Partners & Associated Partners](#) | [LinkedIn](#) | [Twitter](#)

Sobreajuste y subajuste en el aprendizaje automático

Terminología básica

- **Representación de características**
- **Instancias de datos/muestras/ejemplos (X)**
- **Valor objetivo (y)**
- Feature representation
- Data instances/samples/examples (X)
- Target value (y)

fruits							
	fruit_label	fruit_name	fruit_subtype	mass	width	height	color_score
0	1	apple	granny_smith	192	8.4	7.3	0.55
1	1	apple	granny_smith	180	8.0	6.8	0.59
2	1	apple	granny_smith	176	7.4	7.2	0.60
3	2	mandarin	mandarin	86	6.2	4.7	0.80
4	2	mandarin	mandarin	84	6.0	4.6	0.79
5	2	mandarin	mandarin	80	5.8	4.3	0.77
6	2	mandarin	mandarin	80	5.9	4.3	0.81
7	2	mandarin	mandarin	76	5.8	4.0	0.81
8	1	apple	braeburn	178	7.1	7.8	0.92
9	1	apple	braeburn	172	7.4	7.0	0.89
10	1	apple	braeburn	166	6.9	7.3	0.93
11	1	apple	braeburn	172	7.1	7.6	0.92
12	1	apple	braeburn	154	7.0	7.1	0.88
13	1	apple	golden_delicious	164	7.3	7.7	0.70
14	1	apple	golden_delicious	152	7.6	7.3	0.69
15	1	apple	golden_delicious	156	7.7	7.1	0.69
16	1	apple	golden_delicious	156	7.6	7.5	0.67
17	1	apple	golden_delicious	168	7.5	7.6	0.73
18	1	apple	cripps_pink	162	7.5	7.1	0.83

Terminología básica

- **Conjuntos de entrenamiento y pruebas**
- **Modelo/Estimador**
- **El ajuste del modelo produce un "modelo entrenado".**
- **El entrenamiento es el proceso de estimación de los parámetros del modelo.**
- **Método de evaluación**

```
%matplotlib notebook
import numpy as np
import pandas as pd
import seaborn as sn
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier

fruits = pd.read_table('fruit_data_with_colors.txt')

X = fruits[['height', 'width', 'mass', 'color_score']]
y = fruits['fruit_label']

X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)

knn = KNeighborsClassifier(n_neighbors = 5)
knn.fit(X_train, y_train)
print("Accuracy of K-NN classifier on test set: ", knn.score(X_test, y_test))

example_fruit = [[5.5, 2.2, 10, 0.70]]
print("Predicted fruit type for ", example_fruit, " is ", knn.predict(example_fruit))
```

Clasificación y regresión

- Tanto la clasificación como la regresión requieren un conjunto de entrenamiento para "asignarlo" a y (las etiquetas)

Para la clasificación, x y y son discretos (de los grupos que se van a clasificar)

Un caso particular es la clasificación binaria (positivo/negativo o sí/no)

Generalización a multiclase, por ejemplo, frutas

Regresión

- En regresión (aproximación de funciones) el valor objetivo "y" es un valor continuo (real)
- Ejemplo: predecir el precio de una casa, la nota de un examen, el nivel de criminalidad de una zona....
- En general, todo problema tiene un enfoque obvio, aunque es posible pasar de la regresión a la clasificación (agrupando las etiquetas) y, a veces, de la clasificación a la regresión (por interpolación, por ejemplo)
- En general, en los problemas es obvio si se debe aplicar uno u otro
- Los métodos supervisados generalmente cubren ambos enfoques (utilizando el mismo método).

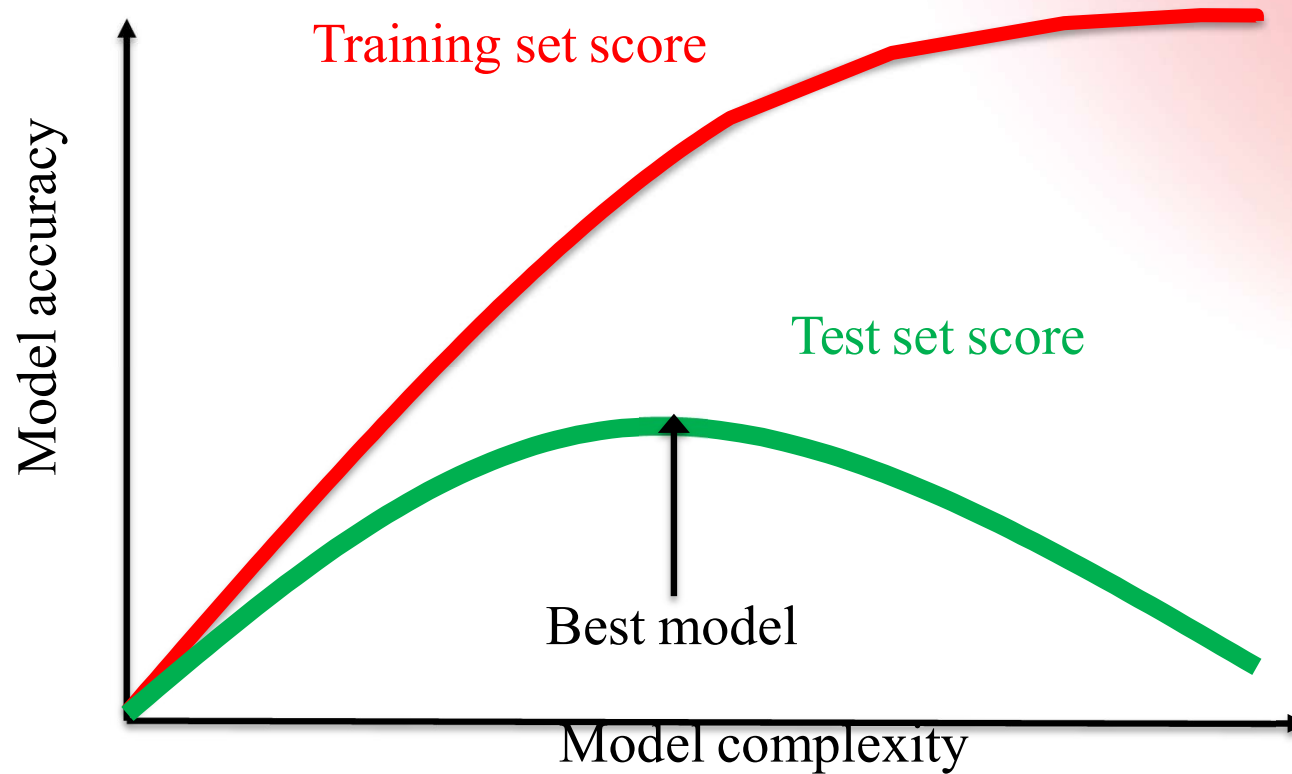
Métodos básicos para el aprendizaje supervisado

- Los métodos más simples son los K-vecinos más cercanos y los modelos lineales
- Hay dos enfoques complementarios:
- (1) K-NN: no hace ninguna suposición sobre las características de los datos de entrada y suele dar buenos resultados, aunque tiende a ser inestable: los datos son el modelo.
- (2) Los modelos lineales asumen una distribución sobre los datos: los datos se comprimen mucho en el modelo (solo parámetros)

¿Qué es un modelo?

- Descripción computacional que expresa la relación entre un conjunto de variables de entrada y una o más variables de resultado que se están estudiando o prediciendo
- En estadística, las variables de entrada se denominan variables independientes y las variables de resultado se denominan variables dependientes.
- En Machine Learning utilizamos el término características para referirnos a la entrada, o variables independientes y el valor objetivo o la etiqueta objetivo para referirnos a la salida, variables dependientes.
- En el caso de los métodos de aprendizaje supervisado, nuestro objetivo es desarrollar modelos predictivos que puedan predecir con precisión el resultado, es decir, el valor objetivo o la etiqueta para los datos de entrada no vistos anteriormente

Precisión vs. complejidad (accuracy vs. complexity)



Generalización, sobreajuste y subajuste

- La generalización se refiere a la capacidad de un algoritmo para dar predicciones precisas de datos nuevos y no vistos anteriormente.

Suposiciones:

Los datos no vistos futuros (conjunto de prueba) tendrán las mismas propiedades que los conjuntos de entrenamiento actuales.

Por lo tanto, se espera que los modelos que son precisos en el conjunto de entrenamiento sean precisos en el conjunto de prueba.

Pero es posible que eso no suceda si el modelo entrenado se ajusta demasiado específicamente al conjunto de entrenamiento.

Se dice que los modelos que son demasiado complejos para la cantidad de datos de entrenamiento disponibles se sobreajustan y no es probable que se generalicen bien a nuevos ejemplos.

Se dice que los modelos que son demasiado simples, que ni siquiera funcionan bien con los datos de entrenamiento, no se ajustan bien y tampoco es probable que generalicen bien.

Underfitting

Entreno al modelo con
1 sólo raza de perro



Muestra nueva:
¿Es perro?



La máquina fallará en reconocer al perro por falta de suficientes muestras. No puede generalizar el conocimiento.

Overfitting

Entreno al modelo con
10 razas de perro color marrón

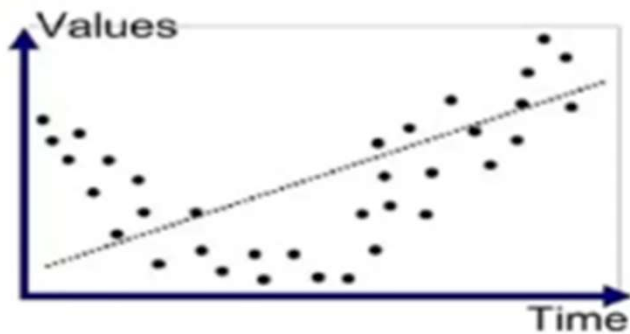


Muestra nueva:
¿Es perro?

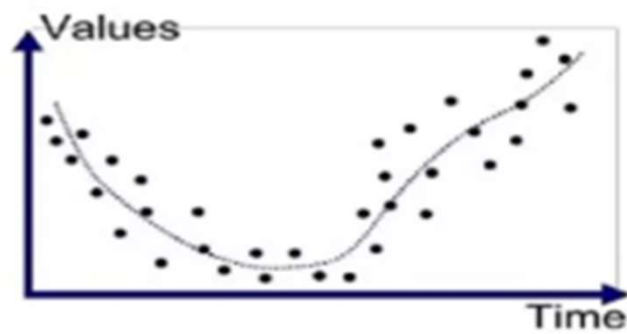


La máquina fallará en reconocer un perro nuevo porque no tiene estrictamente los mismos valores de las muestras de entrenamiento.

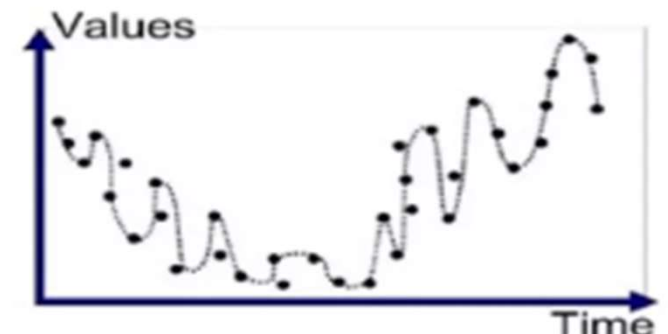
Generalización, sobreajuste y subajuste



Underfitted

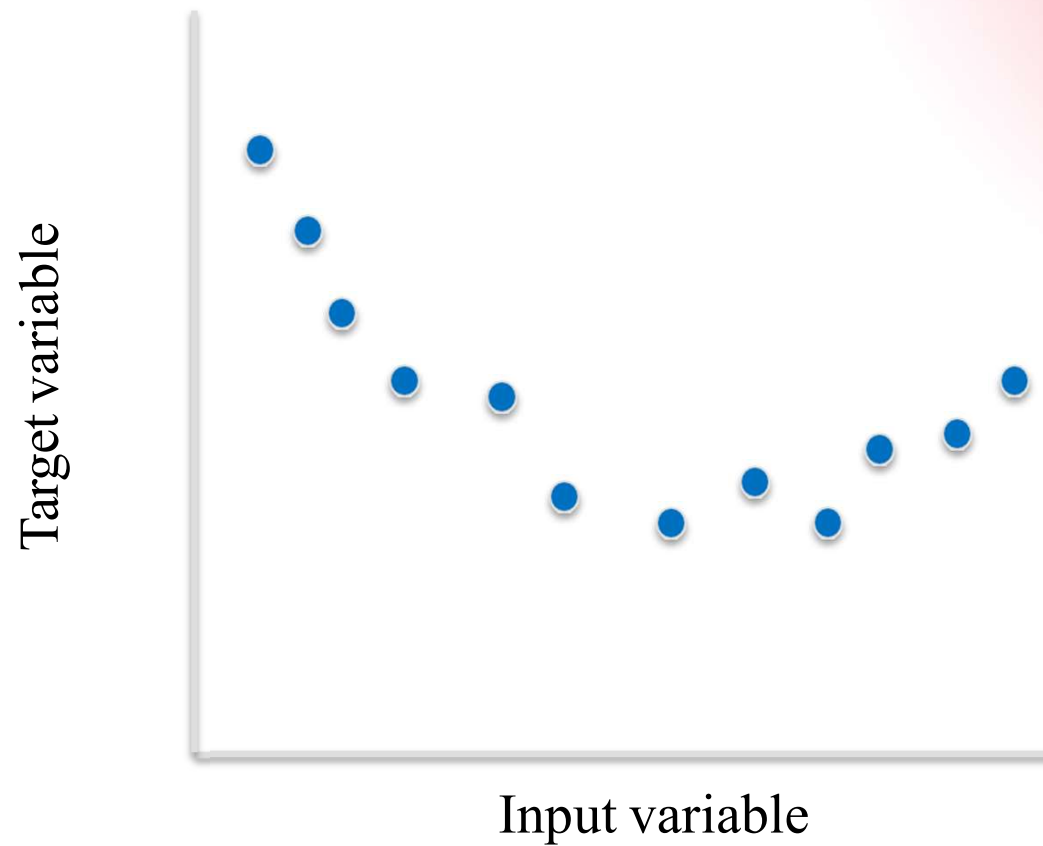


Good Fit/Robust

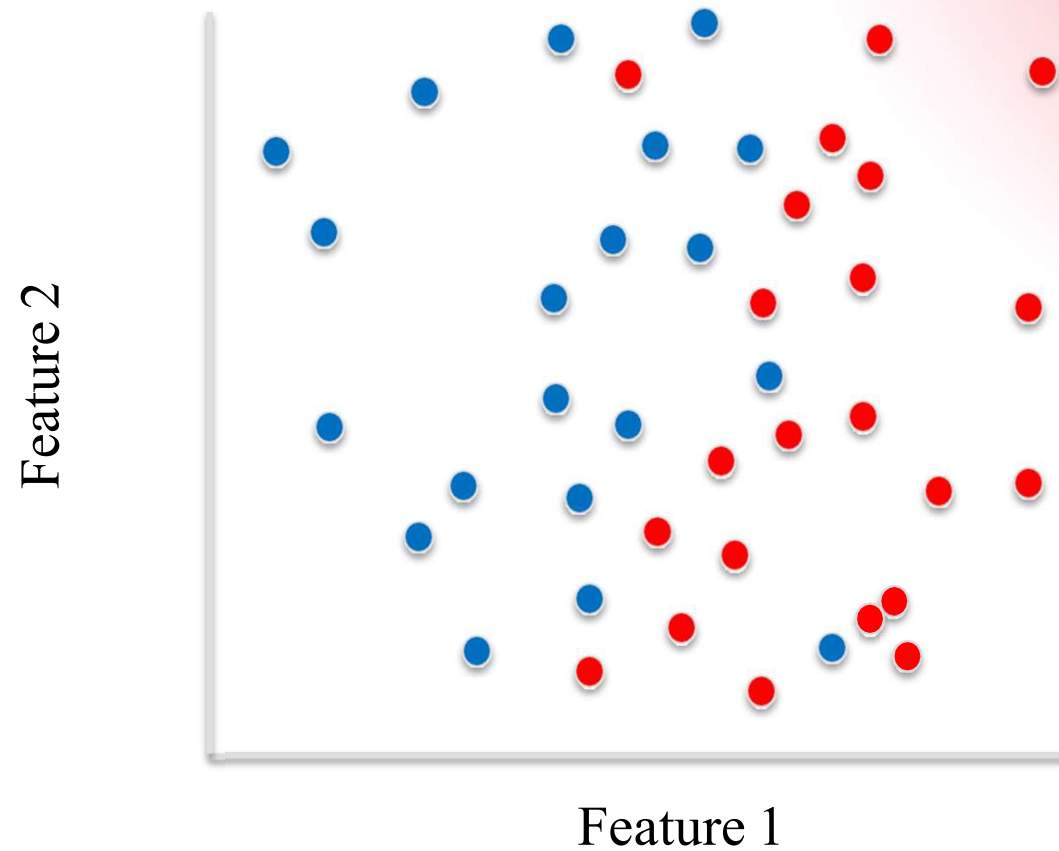


Overfitted

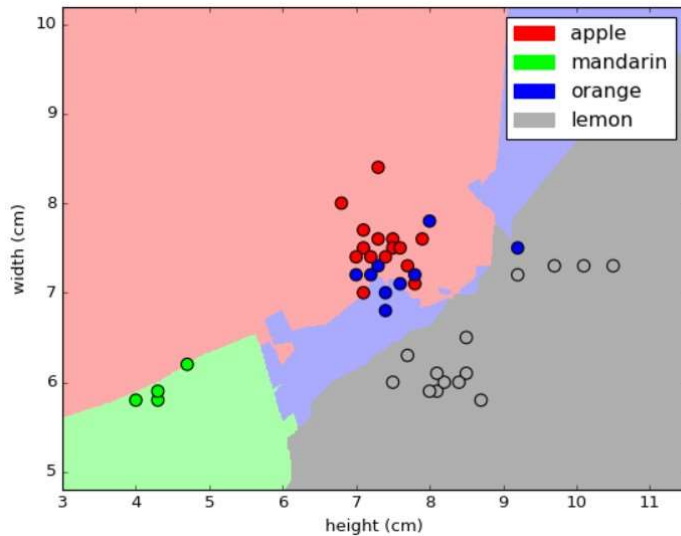
Overfitting in Regression



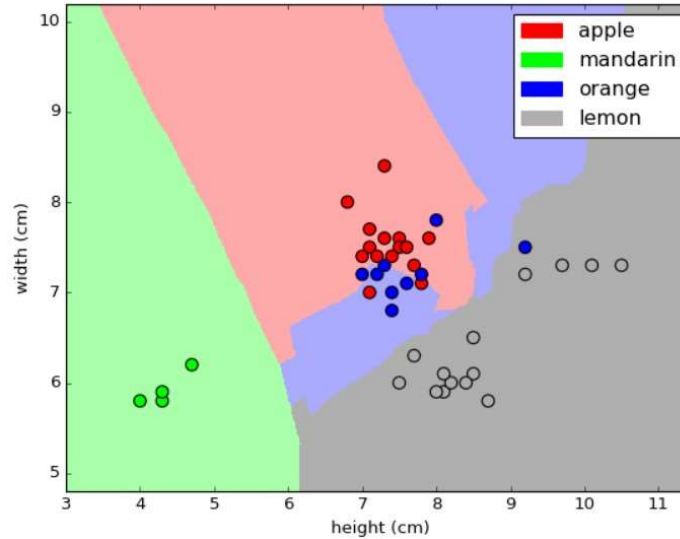
Overfitting in Classification



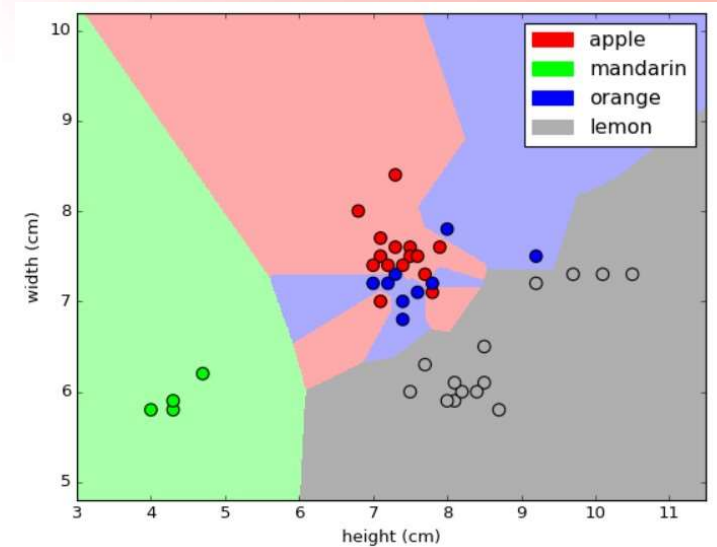
Overfitting with k-NN



K=10



K=5

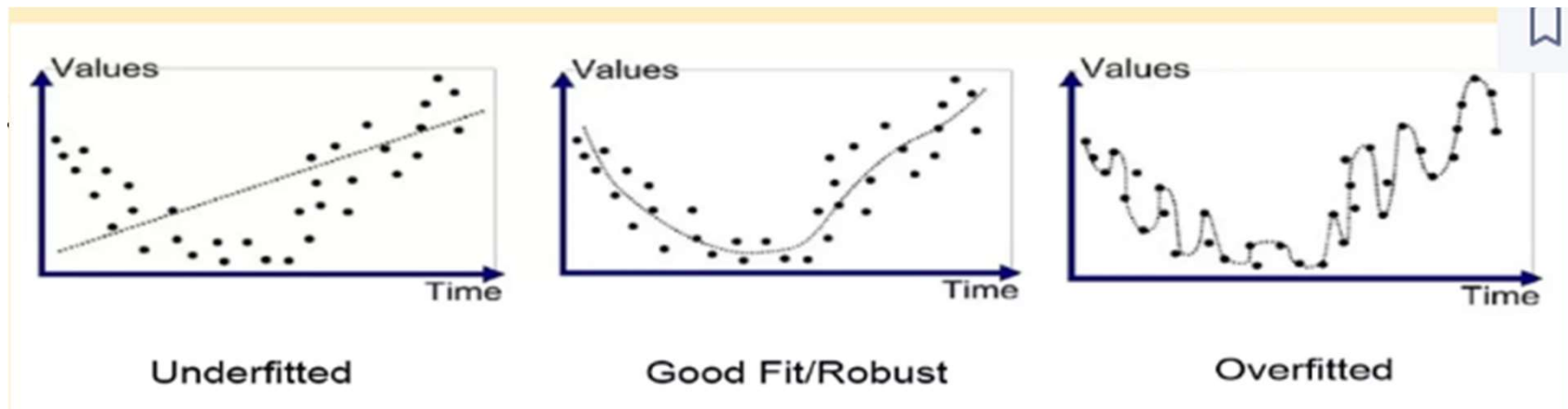


K=1

Sesgo y Varianza (Bias and variance)

- ① **Bias:** It shows the degree of randomness of the training data
 - (a) Based on the training data a suitable model may be created for regression or classification problem.
 - (b) **Regression:** Linear, logistic, polynomial
 - (c) **Classification:** Decision tree, random forest, naive bayes, KNN
- ② **Variance:** It shows the degree of randomness of the testing data
 - (i) Testing data validates the accuracy of a model, that has been trained with the help of training data set.
 - (ii) Testing data is nothing but the unlabeled or unknown data.

Se



Note:

- ⇒ The objective of ML algorithm not only fit for the training data but also fit for the testing data.
- ⇒ In other words, low bias and low variance is the appropriate solution.

Underfitting	Overfitting	Best fit
High bias	Very low bias	Low bias
High variance	High variance	Low variance

Generalización, sobreajuste y subajuste



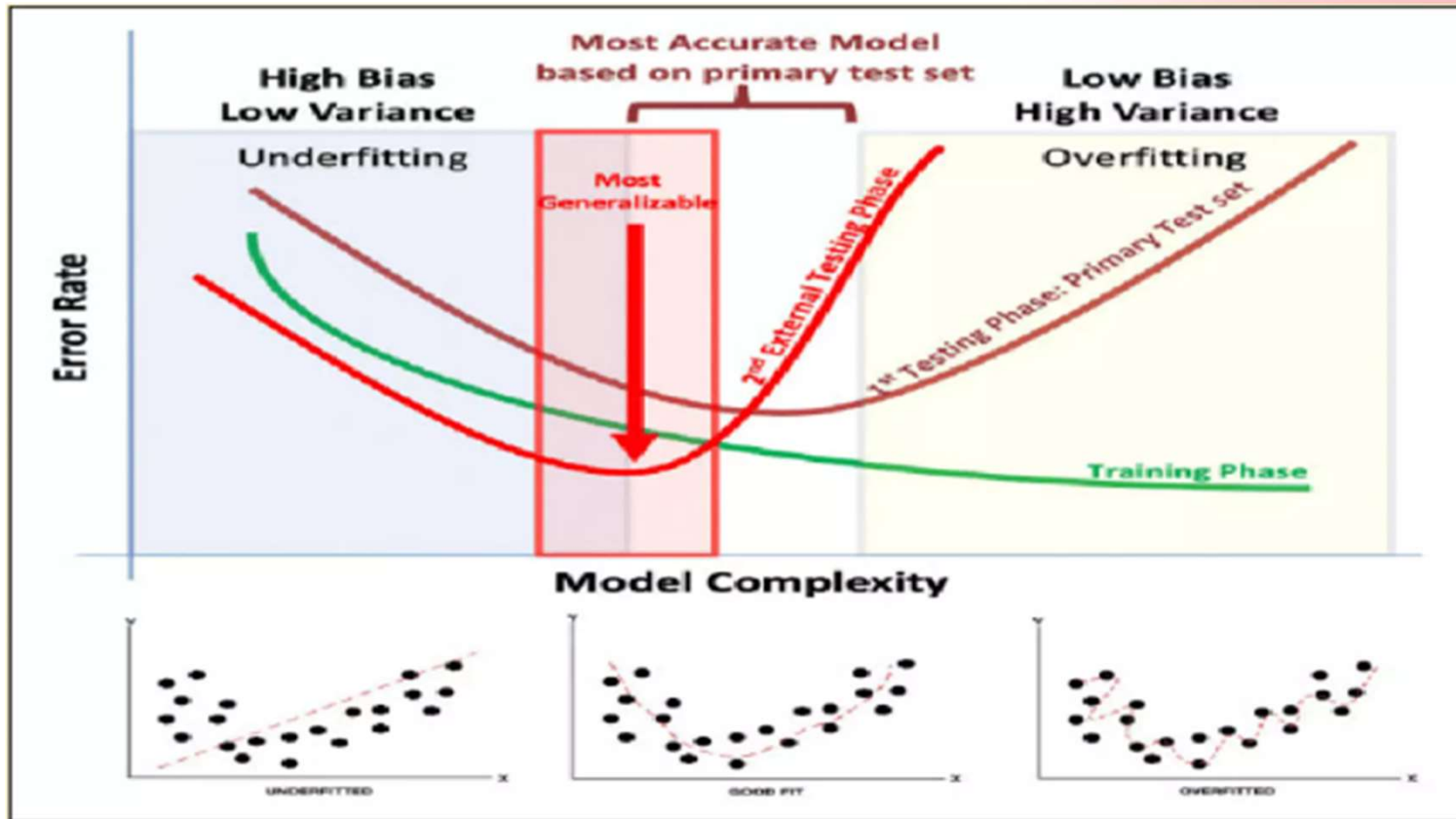
$$\text{bias}(\hat{f}(x)) = \mathbb{E}[\hat{f}(x)] - f(x) \quad (1)$$

$$\text{variance}(\hat{f}(x)) = \mathbb{E}[(\hat{f}(x) - \mathbb{E}[\hat{f}(x)])^2] \quad (2)$$

- ⇒ $\hat{f}(x)$ → output observed through the training model
- ⇒ For linear model $\hat{f}(x) = w_1x + w_0$
- ⇒ For complex model $\hat{f}(x) = \sum_{i=1}^P w_i x^i + w_0$
- ⇒ We don't have idea regarding the true $f(x)$.
- ⇒ **Simple model:** Low bias & high variance
- ⇒ **Complex model:** High bias & low variance

$$\mathbb{E}[(y - \hat{f}(x))^2] = \text{bias}^2 + \text{Variance} + \sigma^2 \text{ (Irreducible error)} \quad (3)$$

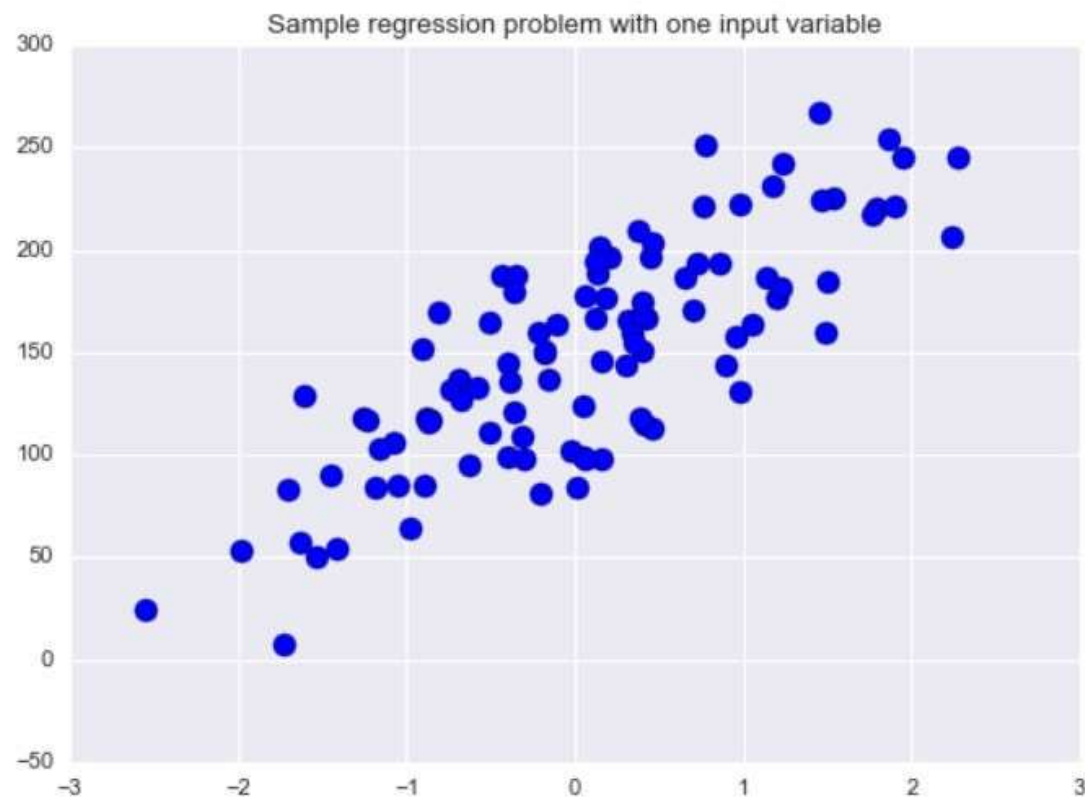
Generalización, sobreajuste y subajuste



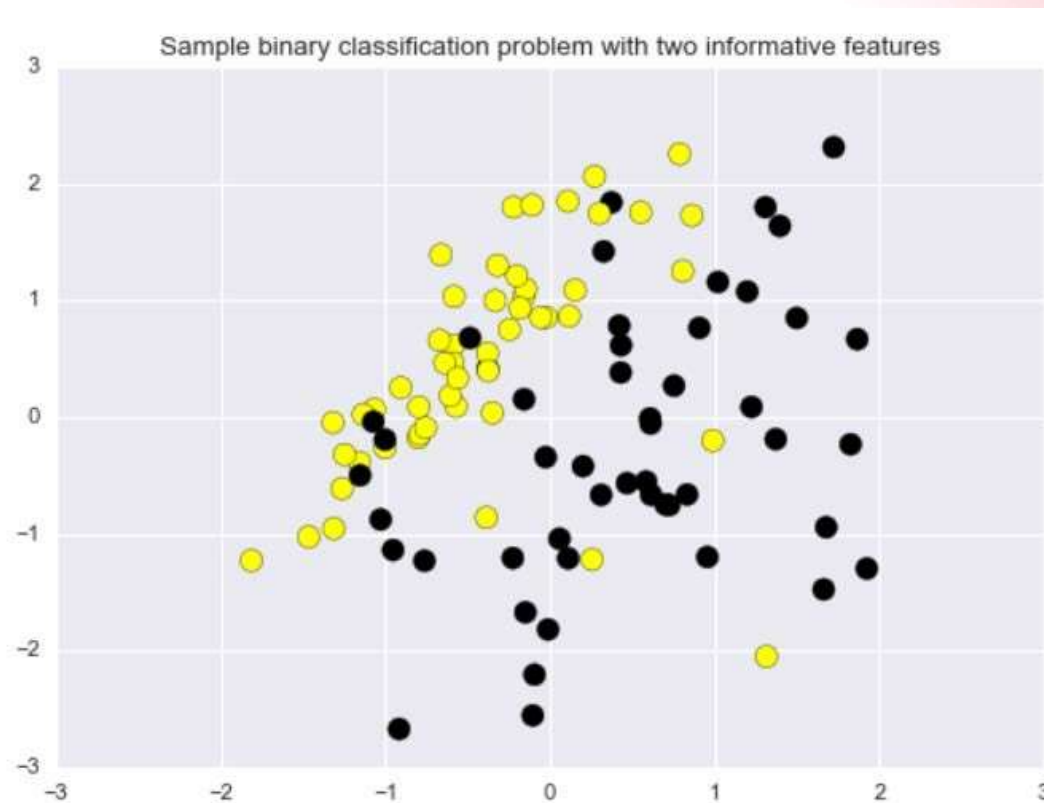
Conjuntos de datos estándar y su relevancia

- Muchos conjuntos de datos del mundo real tienen un espacio de entidades de alta dimensión: tienen docenas, cientos, incluso miles o millones de características.
- Parte de la intuición que obtenemos al observar ejemplos de baja dimensión no siempre se traduce en conjuntos de datos de alta dimensión,
- También los conjuntos de datos de alta dimensión, tienen la mayoría de sus datos en esquinas con mucho espacio vacío, y eso es un poco difícil de visualizar.
- Para explorar diferentes algoritmos de aprendizaje supervisado, vamos a usar una combinación de pequeños conjuntos de datos sintéticos o artificiales como ejemplos, junto con algunos conjuntos de datos más grandes del mundo real
- El conjunto de datos sintéticos utilizará, a título ilustrativo, ejemplos de baja dimensión. Debido a que solo usan un pequeño número de entidades, generalmente una o dos, esto los hace fáciles de explicar y visualizar.

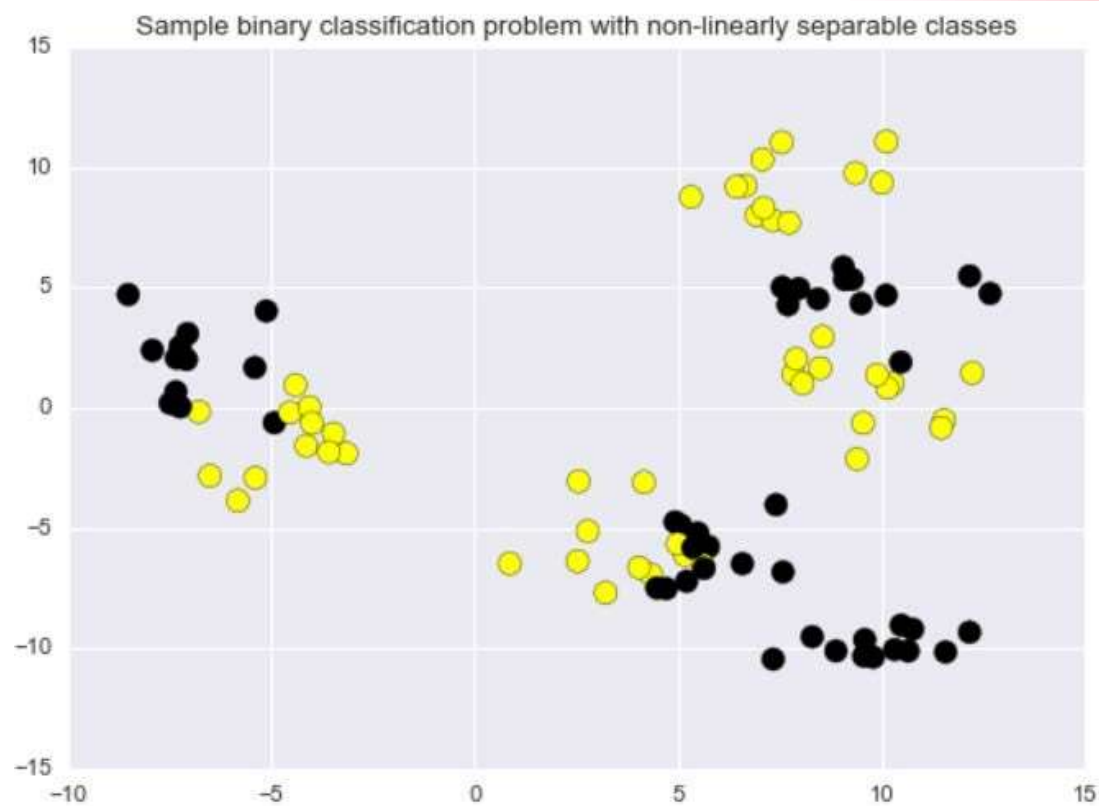
Conjunto de datos de regresión simple



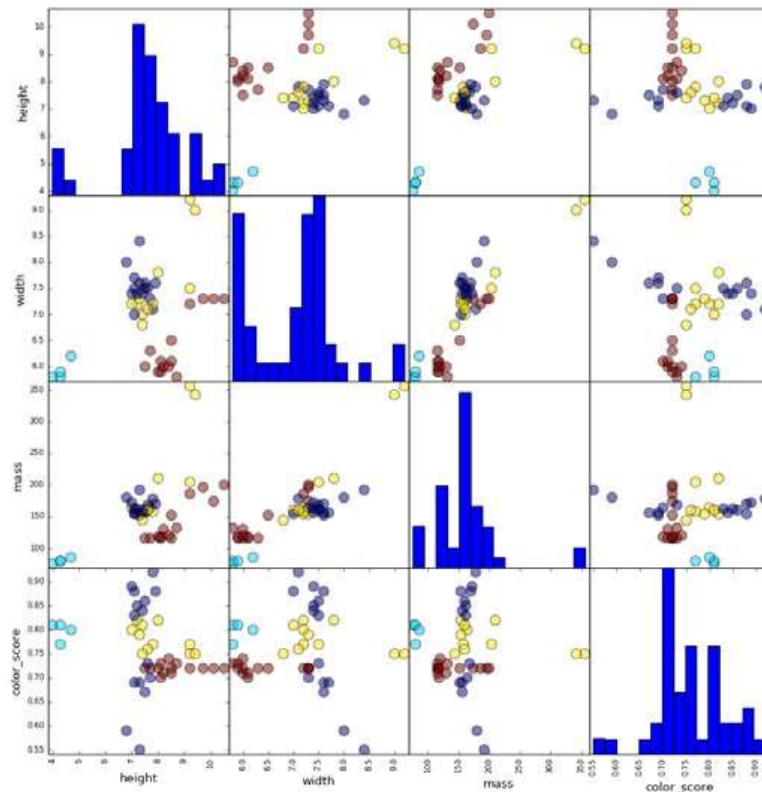
Conjunto de datos de clasificación binaria simple



Conjunto de datos de clasificación binaria compleja



Conjunto de datos multiclase: Frutas



Features

- **width**
- **height**
- **mass**
- **color_index**

Classes

- 0: apple
- 1: mandarin orange
- 2: orange
- 3: lemon

Algoritmo clasificador K-NN

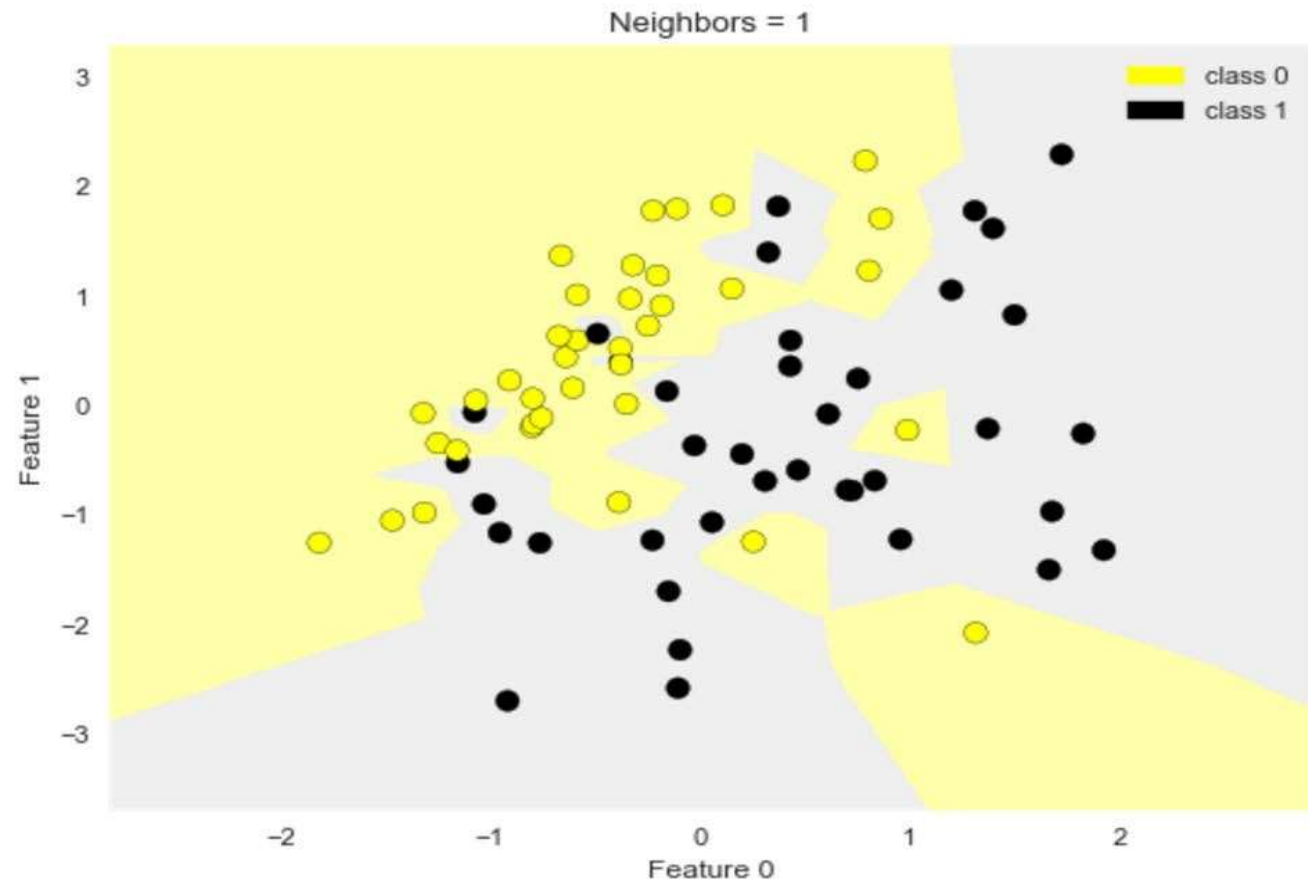
Dado un conjunto de entrenamiento X_{train} con etiquetas y_{train} y dada una nueva instancia x_{test} se va a clasificar:

Encuentre las instancias (X_{NN}) más similares a x_{test} que están en X_{train} .

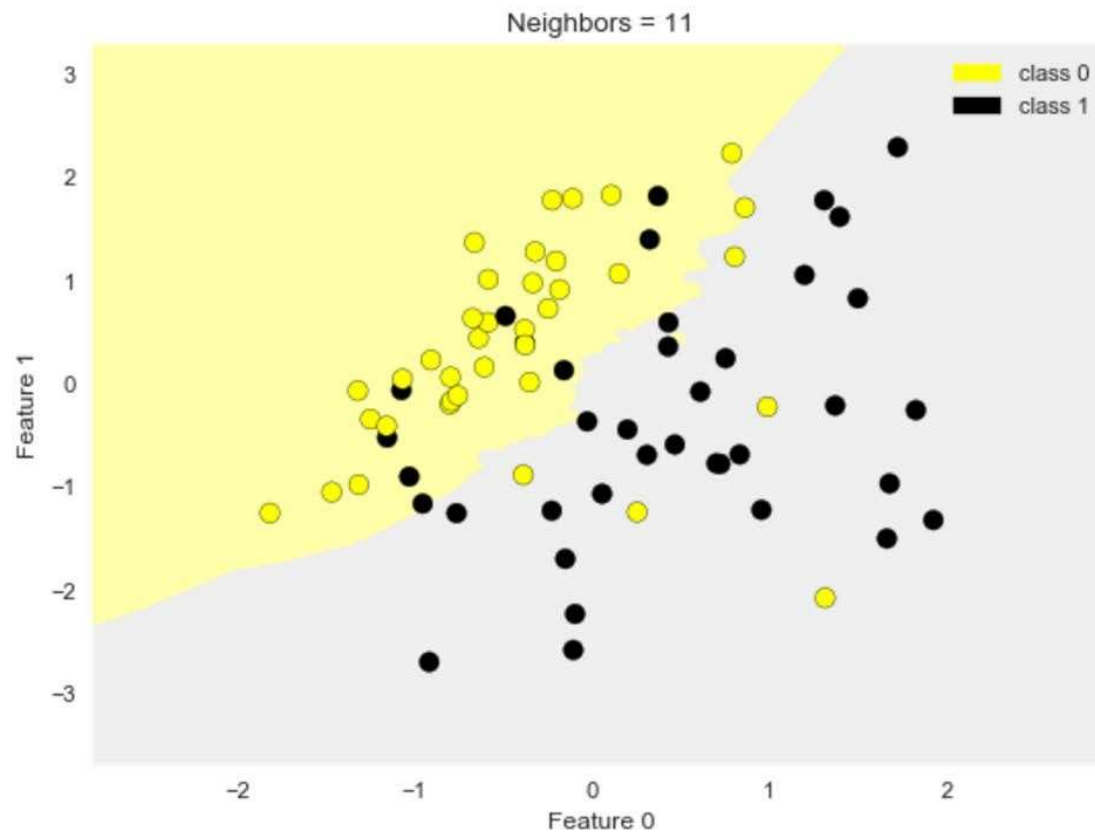
Obtenga las etiquetas y_{NN} para las instancias de X_{NN}

Prediga la etiqueta para x_{test} combinando las etiquetas y_{NN} por ejemplo, mayoría simple de votos

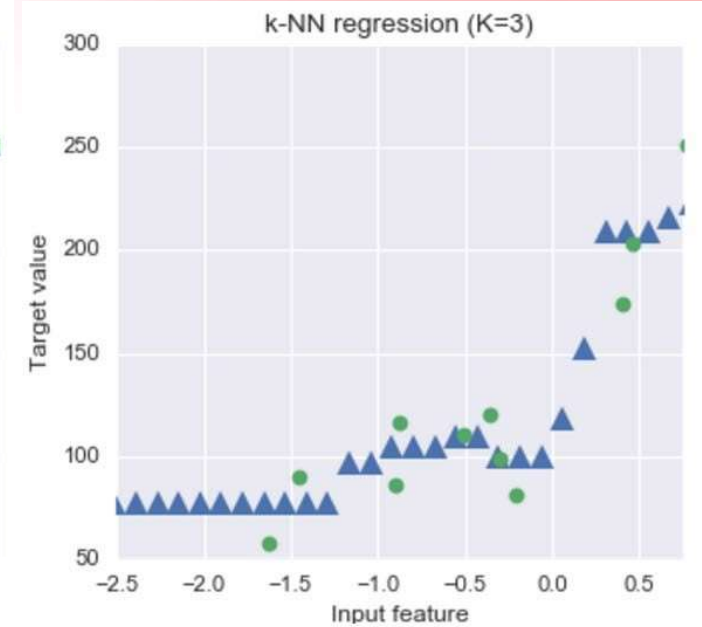
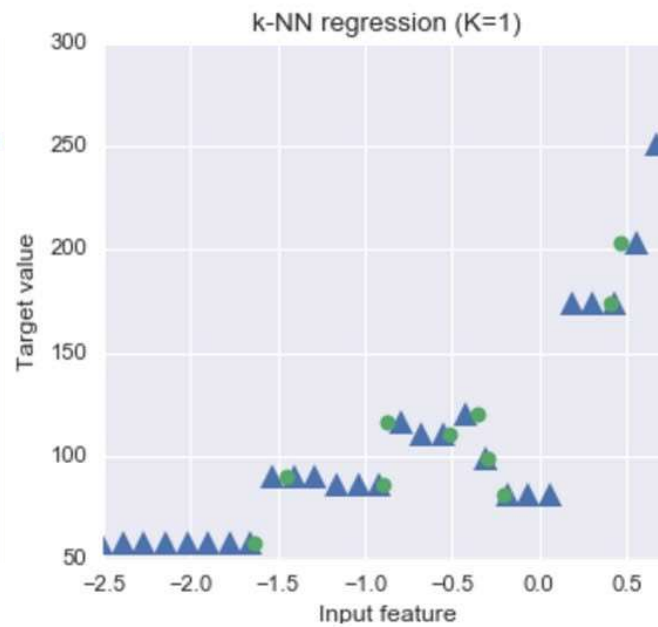
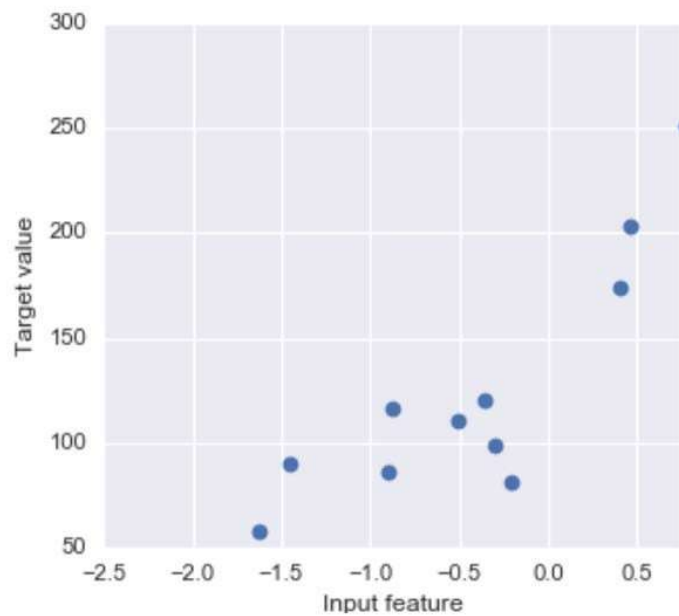
Clasificación K-NN con k=1



Clasificación K-NN con k=11



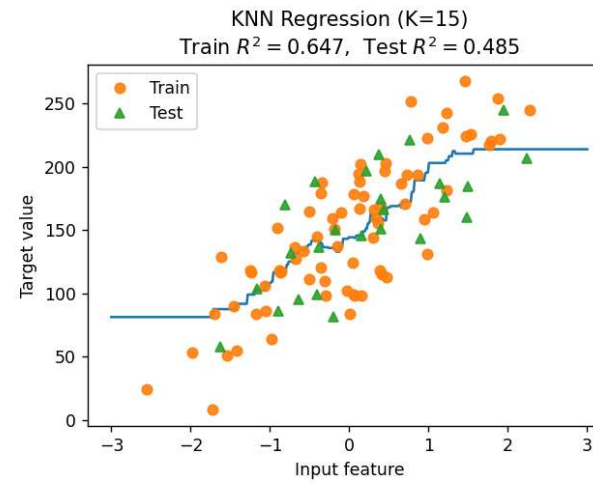
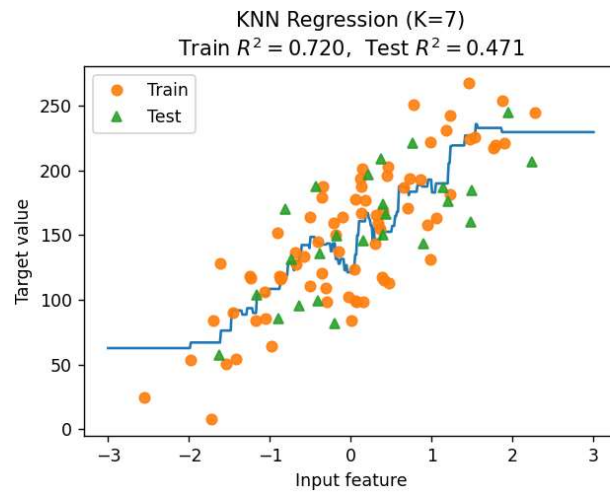
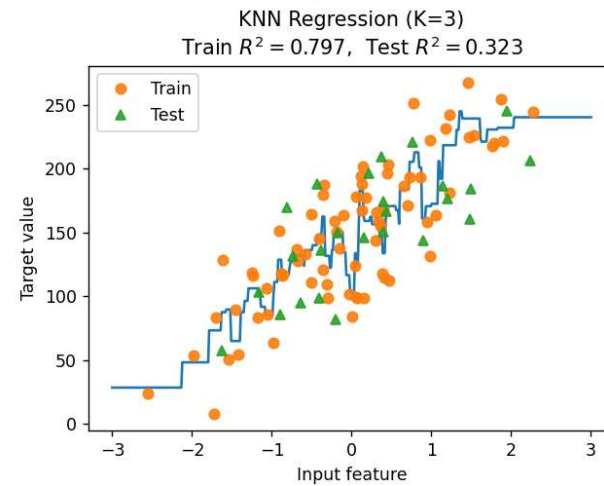
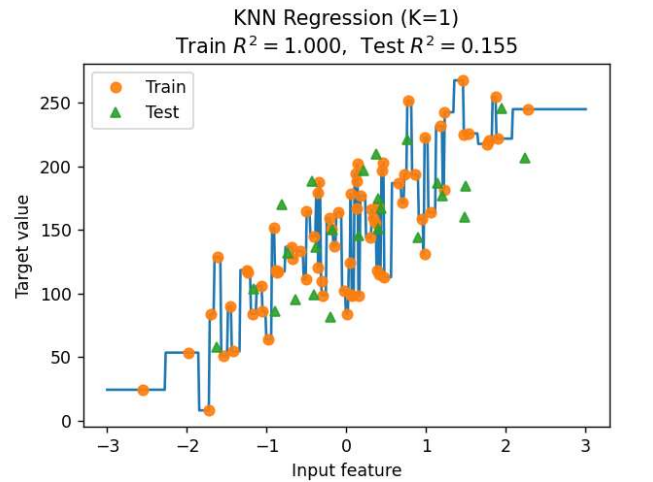
K-NN Regression



Puntuación de regresión R^2 ("r cuadrado")

- Mide el grado en que un modelo de predicción para la regresión se ajusta a los datos dados.
- **La puntuación está entre 0 y 1:** *Un valor de 0 corresponde a un modelo constante que predice el valor medio de todos los valores objetivo de entrenamiento.*
- *Un valor de 1 corresponde a la predicción perfecta*
- También conocido como "coeficiente de determinación"



Complexity ad unction of K



Parameters

- **Complejidad del modelo_{neighbors}** : número de vecinos más cercanos (k) a tener en cuenta
- **Por defecto = 5**
- **Ajuste del modelo**
- **Función de distancia métrica entre puntos de datos**
- **Predeterminado: Distancia de Minkowski con parámetro de potencia $p = 2$ (euclidiano)**



#AIskills4all |  @AIskillsEU |  [linkedin.com/AIskillsEU](https://www.linkedin.com/AIskillsEU)



www.aiskills.eu

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them.