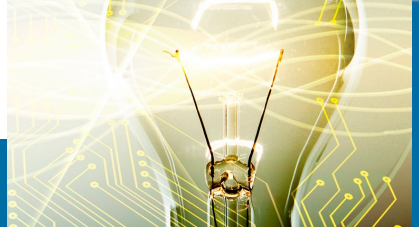


Máster Smart Energy

Postgrado de DIGITAL ENERGY

Sesión 5 – Modelos de Aprendizaje Supervisado (I): Regresión

Sara Barja: sara.barja@upc.edu
Marc Jené: marc.jene@upc.edu

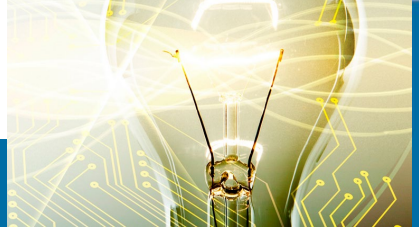
Calendario

	Lunes	Martes	Miércoles	Jueves
ABRIL	10	11	12 S1 – Introducción a Machine Learning	13
	17 S2 – Introducción a Python	18	19 S3 – Estadística descriptiva	20
	24 S4 – Modelos de aprendizaje supervisado (I): Clasificación	25	26 S5 – Modelos de aprendizaje supervisado (II): Regresión	27
MAYO	1	2	3 S6 – Aplicación de AI en el sector eléctrico: Odit-e	4
	8 S7 – Modelos de aprendizaje no supervisado	9	10 S8 – Examen final	

ONLINE!

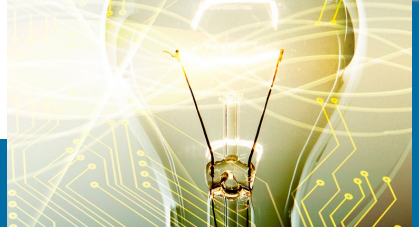
Sara Barja : sara.barja@upc.edu

Marc Jené: marc.jene@upc.edu



Objetivos de la sesión

- Conceptos generales.
- Comprender la regresión lineal.
- Aprender las métricas de evaluación usadas en regresión.
- Presentar el principio de la regularización para evitar overfitting.
- Explorar la visualización de la regresión lineal.
- Presentar la regresión polinomial.
- Repasar modelos de aprendizaje supervisado y la pipeline.
- Presentación de un ejemplo práctico de un modelo de regresión.



Conceptos generales.

Regresión lineal.

Métricas de evaluación.

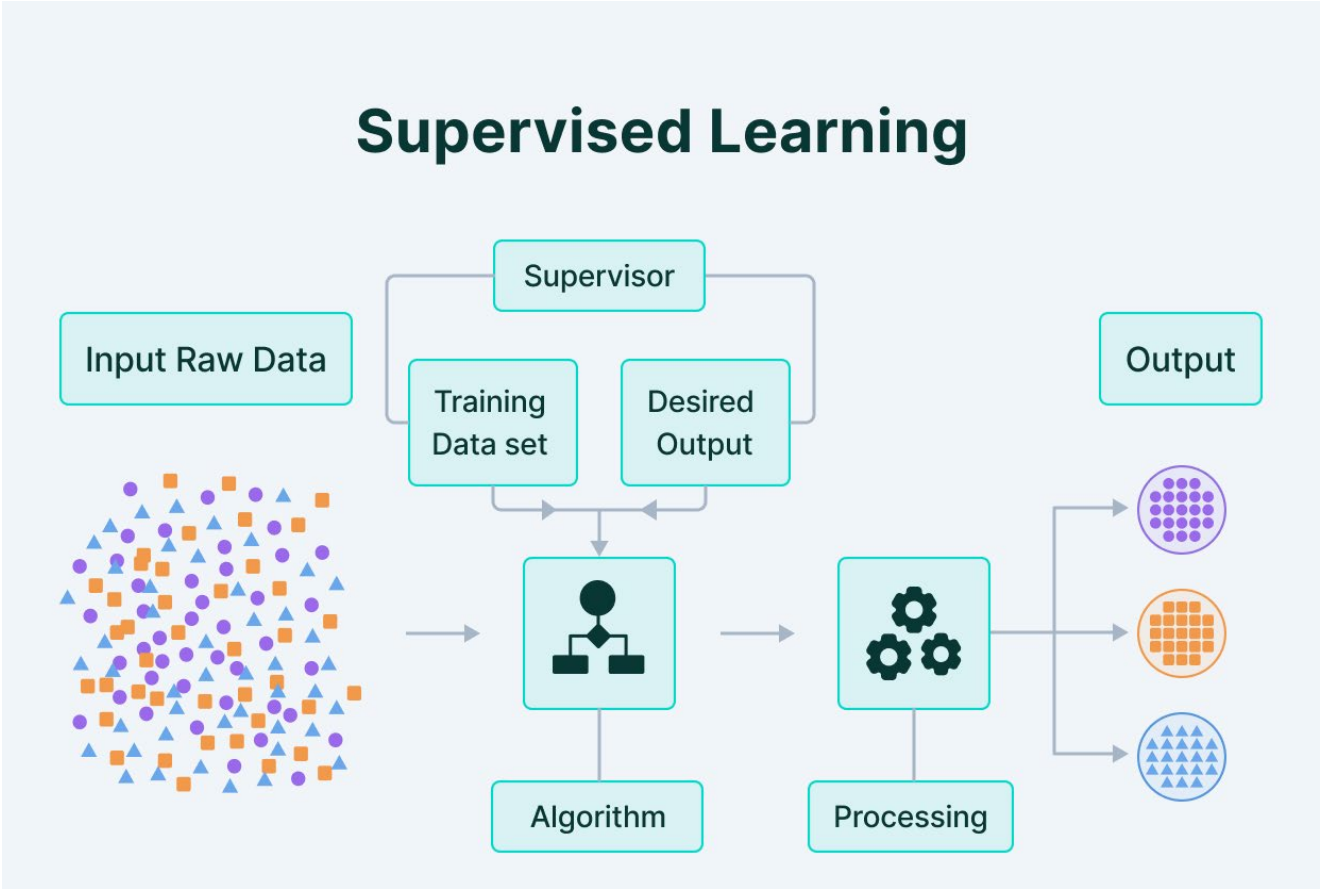
Regularización.

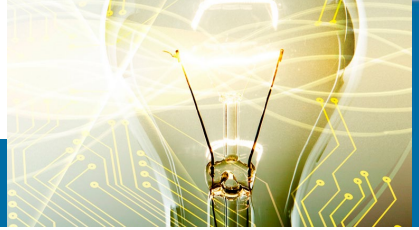
Visualización y regresión polinomial.

Repaso modelos y pipeline.

Caso práctico.

Recapitulemos…



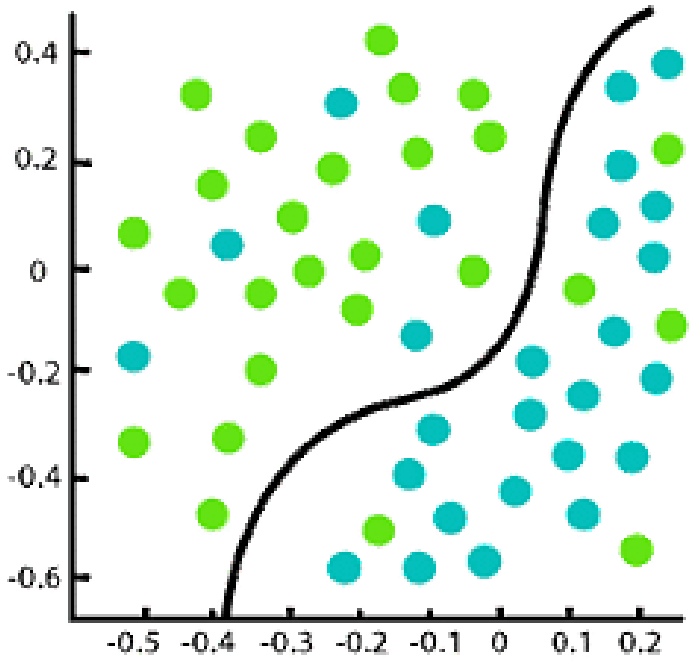


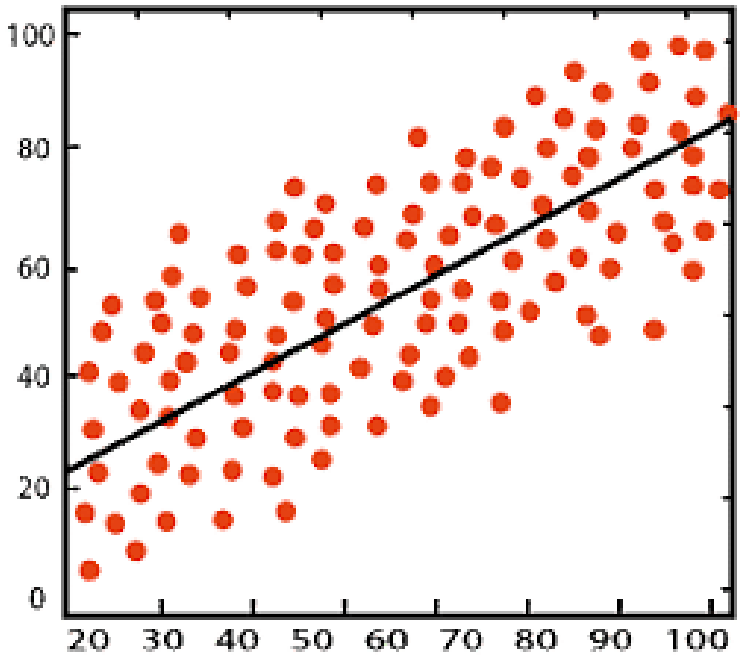
Recapitulemos…

MACHINE LEARNING

Aprendizaje Supervisado	Aprendizaje **NO** Supervisado	Reinforcement Learning
Clasificación	Clustering	
Regresión	Reducir dimensiones	

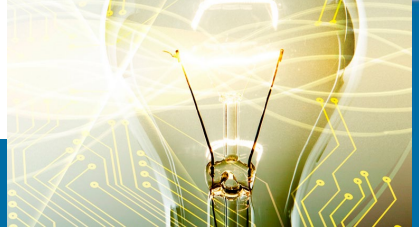
Clasificación vs Regresión



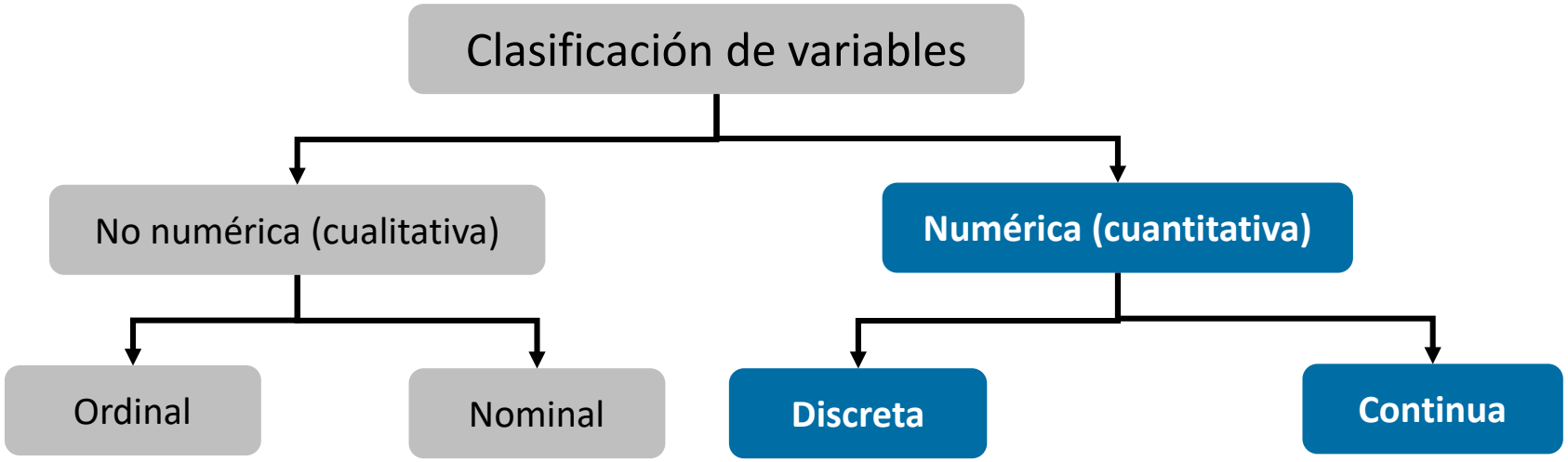


¿Qué es regresión en Machine Learning?

La **regresión** es una subcategoría del **aprendizaje supervisado** cuyo objetivo es **establecer un método** para la relación entre un cierto número de características y una variable objetivo continua.



Variables objetivo en regresión



- Nivel Educación
- Talla camisetas

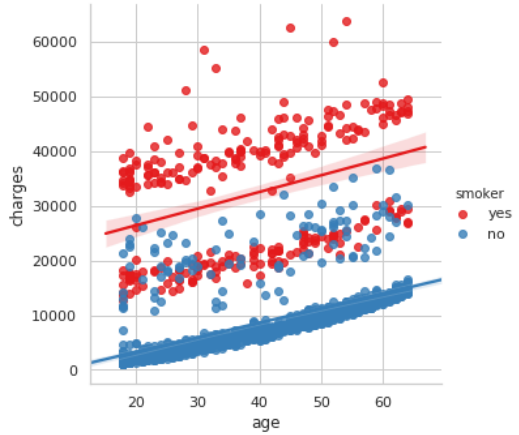
- Color de ojos
- Estado civil
- Nacionalidad

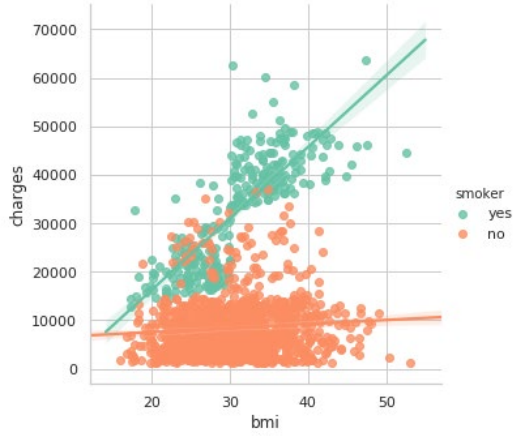
- Número de hijos
- Llamadas por hora

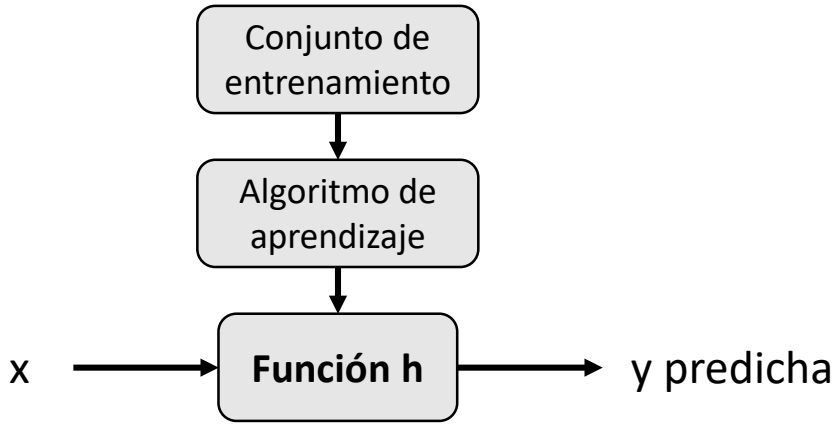
- Representan mediciones
- Los valores dentro de cualquier intervalo son posibles

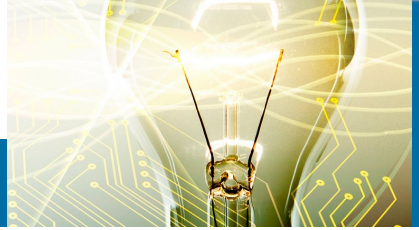
Ejemplo

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520









Conceptos generales.

Regresión lineal.

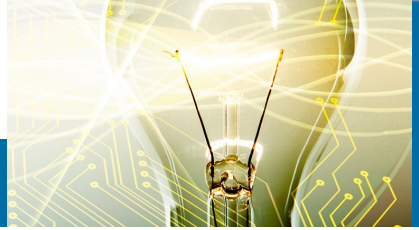
Métricas de evaluación.

Regularización.

Visualización y regresión polinomial.

Repaso modelos y pipeline.

Caso práctico.



Regresión lineal

Notación para el modelo de regresión lineal:

En un modelo lineal, tendremos un parámetro $\mathbf{y}$ que depende de manera lineal de varios covariantes $\mathbf{x}_i$:

$$\boldsymbol{y} = a_1 \cdot \boldsymbol{x_1} + a_2 \cdot \boldsymbol{x_2} + \cdots + a_m \cdot \boldsymbol{x_m}$$

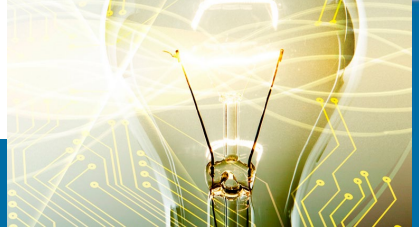
Los términos $\boldsymbol{a_i}$ serán los parámetros del modelo o coeficientes.

Si lo escribimos de forma matricial:

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{w}$$

donde

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \qquad X = \begin{pmatrix} x_{11} & \cdots & x_{1m} \\ x_{21} & & x_{2m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nm} \end{pmatrix}, \qquad w = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}$$



Regresión lineal

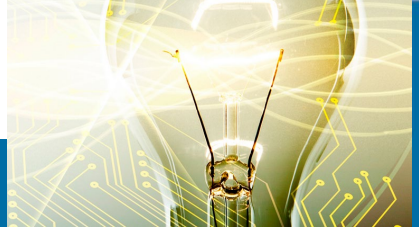
En un modelo de regresión lineal que solo dependa de una variable, tendremos:

$$\boldsymbol{y} = a_0 + a_1 \cdot \boldsymbol{x_1}$$

Con un parámetro $\boldsymbol{a_0}$ llamado constante o corte con el eje de ordenadas.

Si tenemos una regresión multivariable, tendremos:

$$\boldsymbol{y} = a_1 \cdot \boldsymbol{x_1} + a_2 \cdot \boldsymbol{x_2} + \cdots + a_m \cdot \boldsymbol{x_m} = \boldsymbol{Xw}$$



Regresión lineal

El objetivo siempre será minimizar la suma del cuadrado de la distancia entro los puntos reales y el valor de la función.

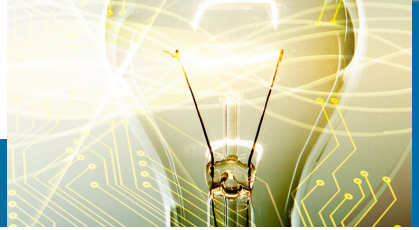
Si tenemos los datos $(\mathbf{x},\mathbf{y})$, queremos minimizar:

$$\|a_0 + a_1 \cdot \boldsymbol{x} - \boldsymbol{y}\|_2^2 = \sum_{j=1}^{n} (a_0 + a_1 \cdot x_j - y_j)^2$$

Esta expresión, se conoce como sum of squared errors of prediction (SSE).

La manera más fácil de encontrar estos dos parámetros es usando el algoritmo OLS (Ordinary Least Squares).

$$\boldsymbol{y} = a_0 + a_1 \cdot \boldsymbol{x_1}$$



OLS (Ordinary Least Squares)

La **regresión por mínimos cuadrados ordinarios** (OLS en inglés) es una técnica habitual para estimar los coeficientes de las ecuaciones de regresión lineal que describen la relación entre una o varias variables cuantitativas independientes y una variable dependiente (regresión lineal simple o múltiple).

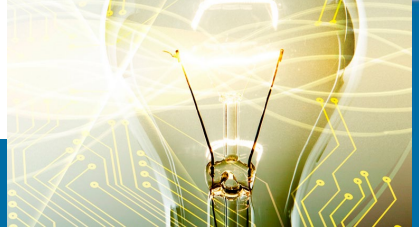
Como funciona el método?
El método OLS corresponde a la **minimización de la suma de las diferencias cuadradas entre los valores observados y los predichos**.

$$\min\left(\sum_{j=1}^{n}\left(a_0 + a_1 \cdot x_j - y_j\right)^2\right)$$

También podríamos minimizar otros valores como la **suma del valor absoluto de las diferencias.**

$$\min\left(\sum_{j=1}^{n}\left|a_0 + a_1 \cdot x_j - y_j\right|\right)$$

jupyter



OLS (Ordinary Least Squares)

Ventajas OLS

- Fácil de calcular computacionalmente cuando son datasets pequeños. Para conjuntos más grandes el cálculo de una inversa provoca un aumento del tiempo de cómputo.

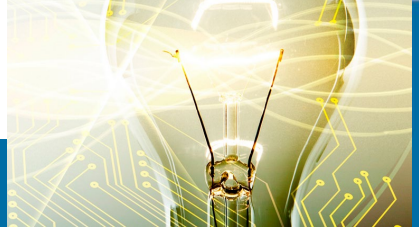
$$\hat{\beta} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y}$$

$\hat{\beta}$ = ordinary least squares estimator
X = matrix regressor variable X
$\top$ = matrix transpose
y = vector of the value of the response variable

- Fácil de interpretar

El modelo obtenido es $\hat{\boldsymbol{y}} = \widehat{a_0} + \widehat{a_1} \cdot \boldsymbol{x_1}$, donde los sombreros indican que son valores estimados

En el caso de la suma del valor absoluto de las diferencias se penalizan menos los valores lejanos.



Gradient Descent

Otra manera de calcular los coeficientes es el método conocido como **Gradient Descent,** que también se ha visto en otros algoritmos de machine learning.
Este proceso consta de los siguientes pasos:

- Calcular el error (MSE) del modelo con un parámetro inicial:

$$MSE(a_1) = \frac{1}{n}\sum_{i=1}^{n}(\widehat{y^i} - y^i)^2$$

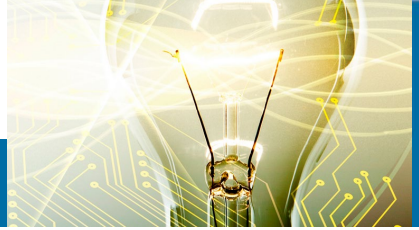
- Calcular la derivada del error en aquel punto.

$$\frac{d}{da_1}MSE(a_1)$$

- Actualizar el parámetro moviéndolo α veces la dirección de la derivada (learning rate)

$$a_1 = a_1 - \alpha\frac{d}{da_1}MSE(a_1)$$

jupyter

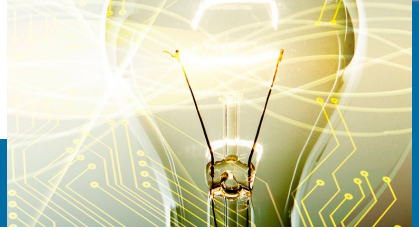


Regresión lineal con Sklearn

Por suerte no tenemos que desarrollar estos algoritmos desde cero. Para esto trabajamos con librerías de Machine Learning que ya están desarrolladas!

Por ejemplo, para crear un modelo de regresión lineal en Sklearn, seguiríamos estos pasos:

- Cargar dataset (dividir en X e y).
- Dividir dataset en entreno y test.
- Crear modelo de regresión lineal (*regr=LinearRegression()*)
- Entrenar (*regr.fit(X_train, y_train)*)
- Obtener modelo:
  - *a1 = regr.coef_*
  - *a0 = regr.intercept*
- Hacer predicciones (*y_pred = regr.predict(X_test)*)



Conceptos generales.

Regresión lineal.

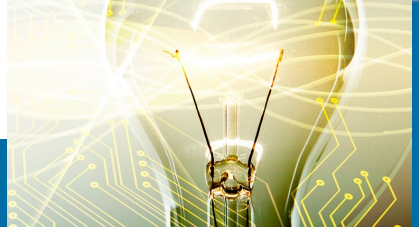
Métricas de evaluación.

Regularización.

Visualización y regresión polinomial.

Repaso modelos y pipeline.

Caso práctico.



Métricas de evaluación

Se puede evaluar el modelo obtenido calculando el **mean squared error (MSE)** y el **coeficiente de determinación** R^2 .

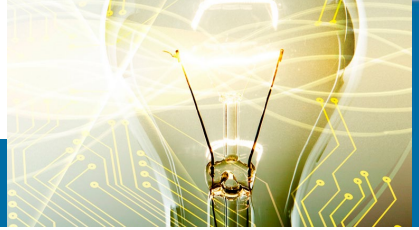
El error cuadrático medio mide la cantidad de error que hay entre dos conjuntos de datos. Compara un valor predicho y un valor observado o conocido.

El MSE, o error cuadrático medio, se calcula como:

$$\boldsymbol{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\widehat{y^i} - y^i)^2$$

El error cuadrático medio no es del todo intuitivo, pues nos da el error medio al cuadrado. Más intuitivo es la raíz cuadrada del error cuadrático medio (RMSE en inglés). A pesar de eso, no se usa todo el tiempo ya que:

- es más costoso computacionalmente.
- la forma del error cuadrático medio hace que cierto tipos de algoritmos de optimización (e.g., el Gradient Descent), encuentre la mejor solución más rápido.



Métricas de evaluación

El **coeficiente de determinación** es la proporción de la varianza total de la variable explicada por la regresión. $R^2 \approx 1$, mayor será el ajuste del modelo a la variable que pretende aplicara para el caso en concreto. $R^2 \approx 0$, menor será el ajuste, y menos fiable el modelo.

El coeficiente R^2 se define

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - {}^{\boldsymbol{u}}/_{\boldsymbol{v}}$$

donde $\mathbf{u}$ es la **suma de los cuadrados de los errores**:

$$\boldsymbol{u} = \sum_{i=1}^{n} (\boldsymbol{y} - \hat{\boldsymbol{y}})^2$$

$\mathbf{y}$: **valores observados** y

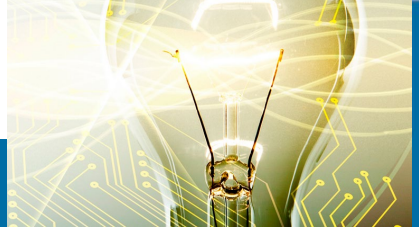
$\hat{\boldsymbol{y}}$: **valores de la predicción**.

$\boldsymbol{v}$ es **el total de la suma de los cuadrado**:

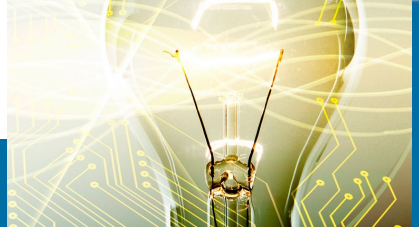
$$\boldsymbol{v} = \sum_{i=1}^{n} (\boldsymbol{y} - \bar{\boldsymbol{y}})^2$$

$\bar{\boldsymbol{y}}$: media de los datos observados.

Sklearn también incluye estas métricas de evaluación.



Conceptos generales.
Regresión lineal.
Métricas de evaluación.
Regularización.
Visualización y regresión polinomial.
Repaso modelos y pipeline.
Caso práctico.



Regularización (opcional)

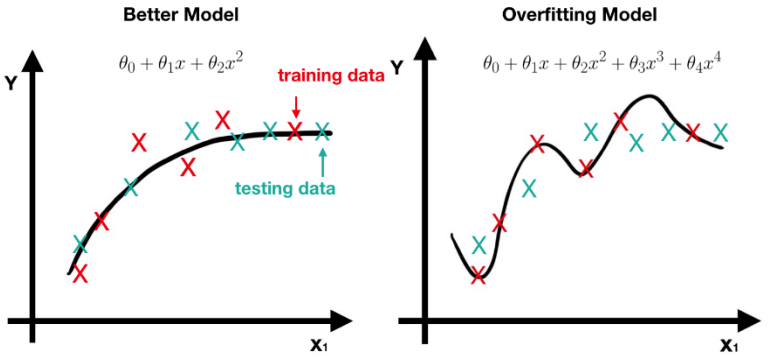
La **regularización** penaliza aquellos coeficientes que son muy grandes, **reduciendo la complejidad del modelo** y puede ser de **gran ayuda cuando se tengan problemas de overfitting**.

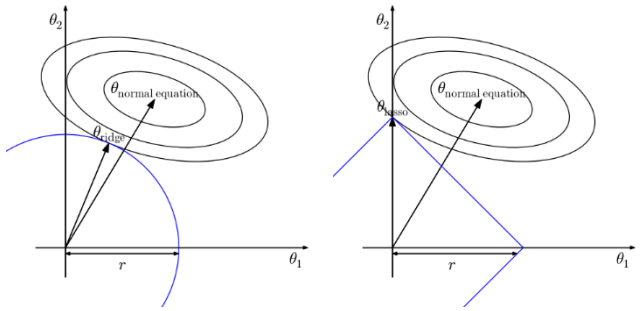
Para ello, al término a minimizar se añade un termino más. Así, a la suma de los cuadrados de los errores (SSE), ahora el objetivo es minimizar:

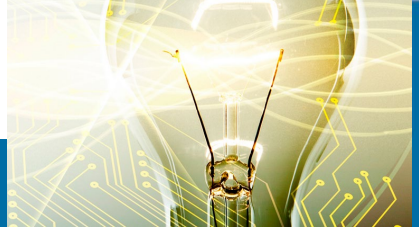
$$argmin_{\boldsymbol{w}} = \left(\frac{1}{2n} \|X\boldsymbol{w} - \boldsymbol{y}\|_2^2 + \boldsymbol{\alpha}\|\boldsymbol{w}\|_{\boldsymbol{p}} \right)$$

donde $\|\boldsymbol{w}\|_{\boldsymbol{p}}$ es la $\ell_{\boldsymbol{p}}$ - norma del vector de parámetros.

p = 2 cuando se usa Ridge y p = 1 cuando se usa Lasso.



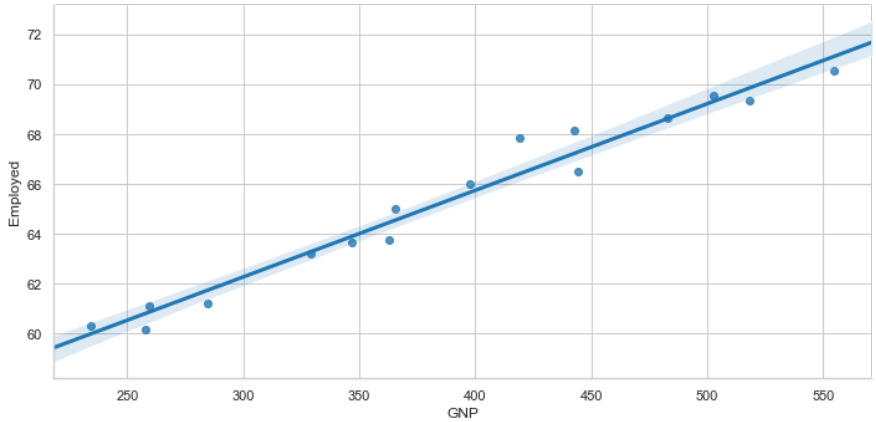




Conceptos generales.
Regresión lineal.
Métricas de evaluación.
Regularización.
Visualización y regresión polinomial.
Repaso modelos y pipeline.
Caso práctico.

Visualización

La librería Seaborn ofrece funciones muy interesantes para visualizar, como por ejemplo la función *lmplot()*, que representa las relaciones lineales de datasets multidimensionales. El input de esta función debe ser en pandas.



Esta visualización incluye:

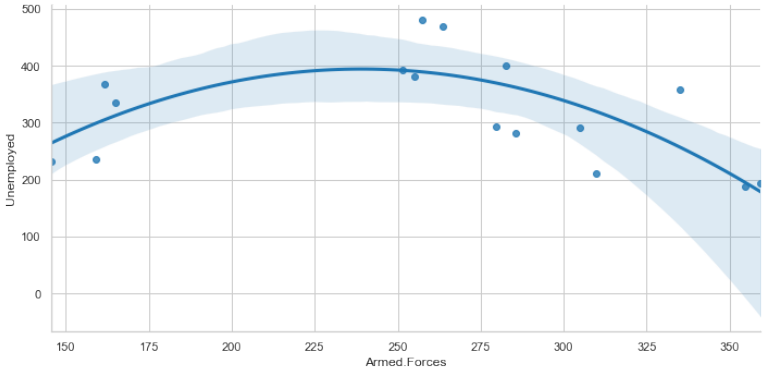
- los puntos observados en scatterplot.
- la línea de regresión obtenida con un IC del 95%.

Regresión múltiple y regresión polinomial

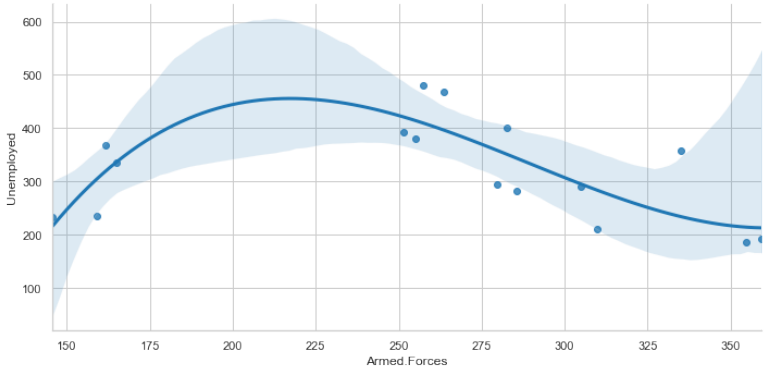
La regresión será lineal en sus parámetros no necesariamente en sus predictores. Si se añaden transformaciones no lineales al modelo de regresión lineal, el modelo puede pasar a ser no lineal

$$\boldsymbol{y} = a_1\phi(\boldsymbol{x_1}) + \cdots + a_m\phi(\boldsymbol{x_m})$$

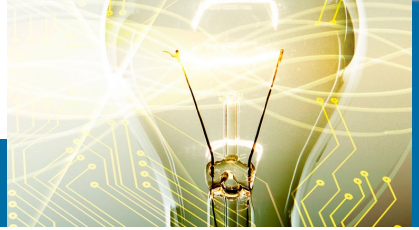
Esta técnica se conoce como **Polynomial Regression**, donde cuando más alto sea el grado del polinomio que se aplique más complejo puede ser el modelo (**vigilar con el overfitting y el tiempo de cómputo!!)**



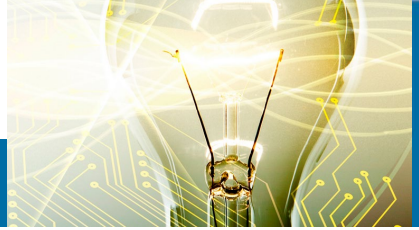
Modelo cuadrado



Modelo cúbico



Conceptos generales.
Regresión lineal.
Métricas de evaluación.
Regularización.
Visualización y regresión polinomial.
Repaso modelos y pipeline.
Caso práctico.



Modelos supervisados de Regresión

Scikit Learn ofrece varios modelos de regresión

Modelos de regresión lineal:

- Ridge Regression
- Lasso Regression

Support Vector Machine

- Support Vector Regression (SVR)

Nearest neighbour Regressor

- K-nearest neighbors (k-NN)

Ensemble models

- Decision Trees
- Random Forest Regressor
- Gradient Boosting Regressor

Neural Network

- Multi Layer Perceptron Regressor

También existen otros modelos de otras librerías

- **Redes Neuronales**
- **XGBoost**

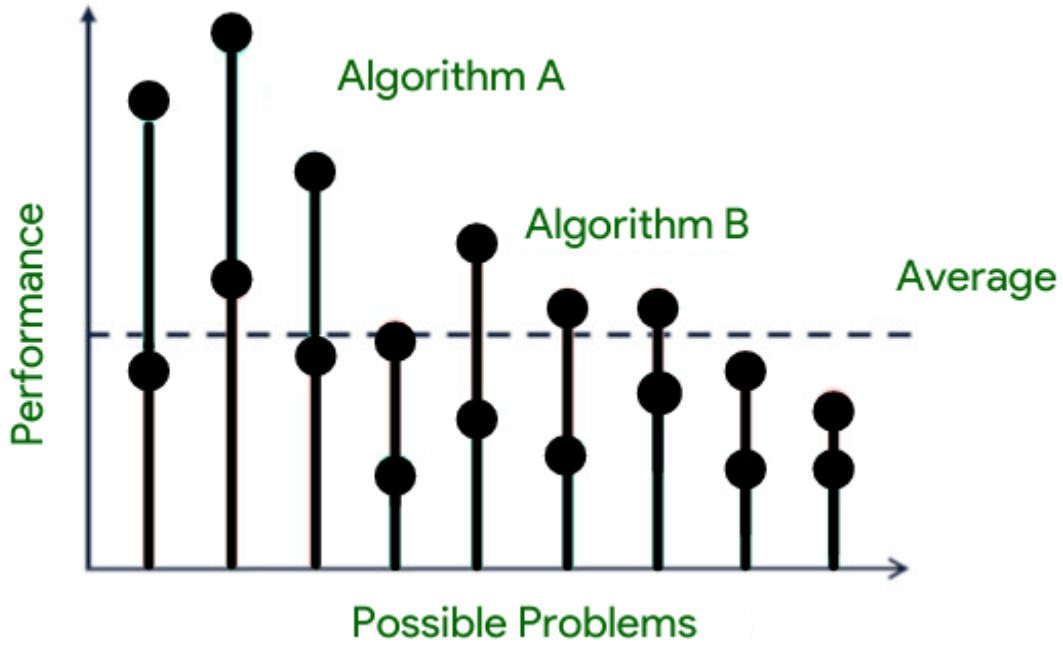
Modelos supervisados de Regresión

¿Qué modelo tengo que usar?

Modelos supervisados de Regresión

¿Qué modelo tengo que usar?

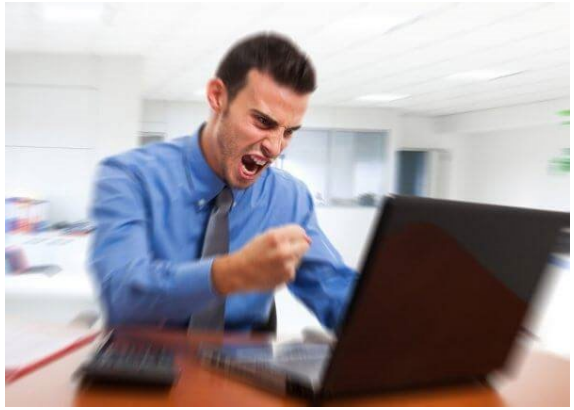
- Teorema **“no free lunch”**



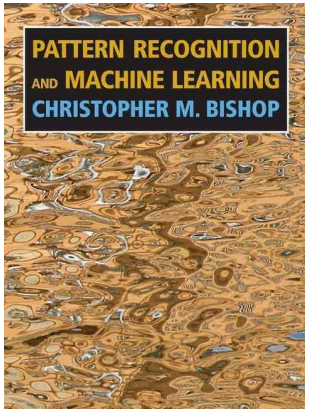
Modelos supervisados de Regresión

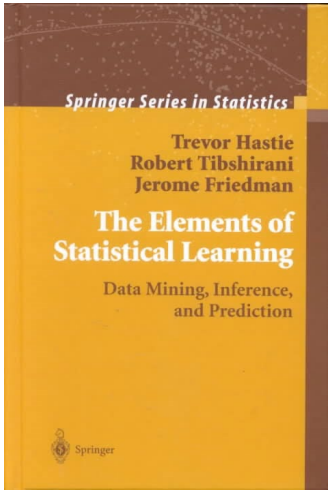
¿Cómo mejorar mi manera de escoger?

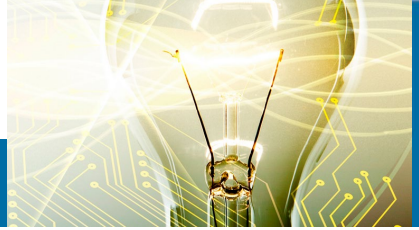
- **Experiencia**



- **Conocimiento sobre los modelos**

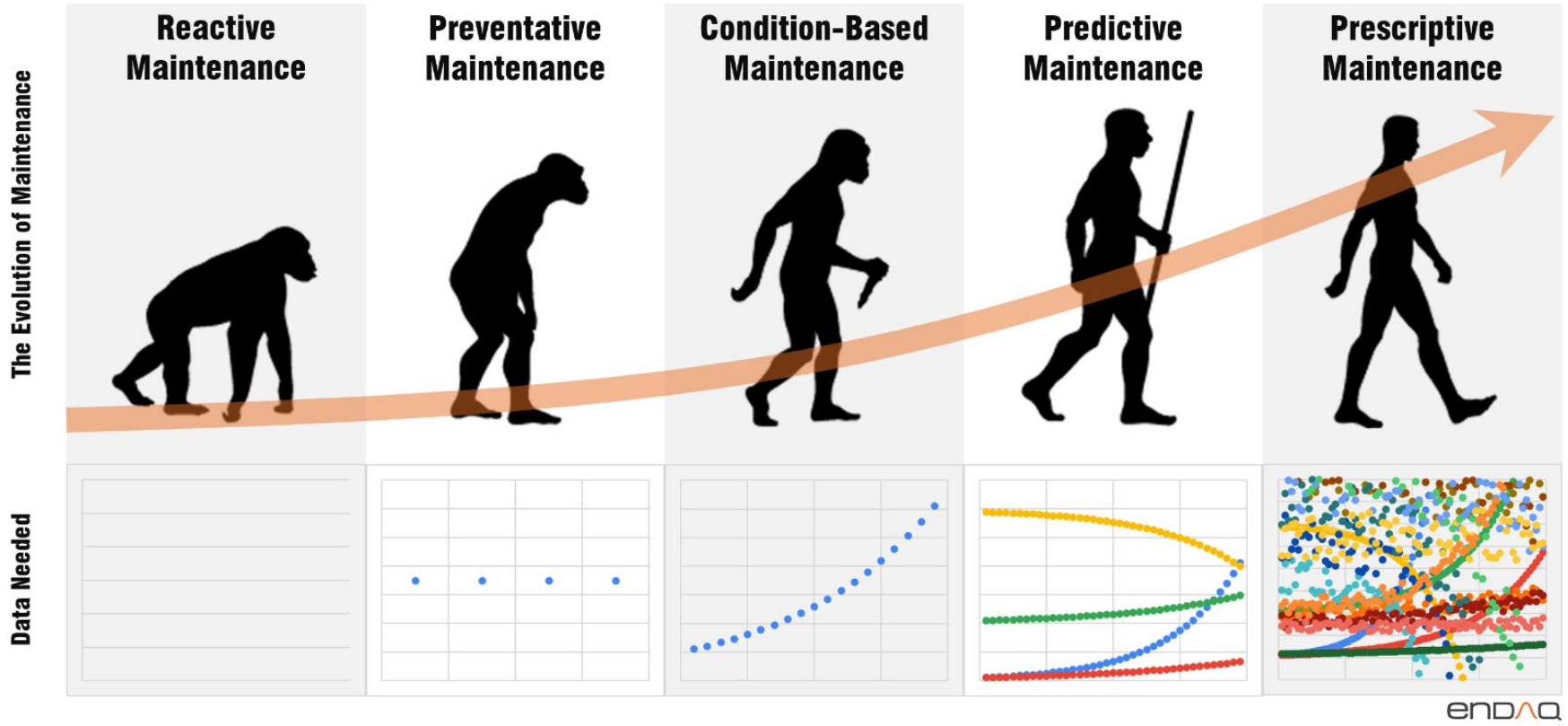




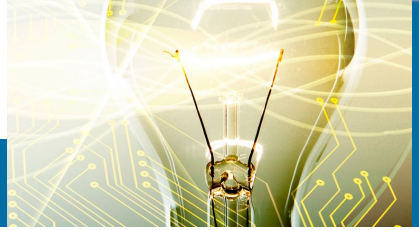


Conceptos generales.
Regresión lineal.
Métricas de evaluación.
Regularización.
Visualización y regresión polinomial.
Repaso modelos y pipeline.
Caso práctico.

Ejercicio de Regresión – Mantenimiento predictivo



https://blog.endaq.com/differences-between-condition-based-predictive-and-prescriptive-maintenance

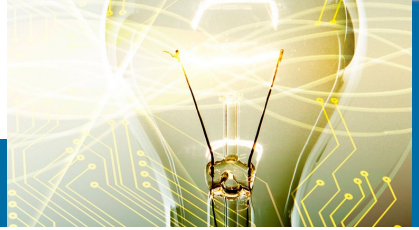


Ejercicio de Regresión – Mantenimiento predictivo

Dataset de la NASA

Enlace https://data.nasa.gov/Aerospace/CMAPSS-Jet-Engine-Simulated-Data/ff5v-kuh6

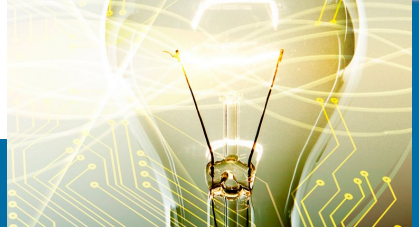
- El dataset consta de múltiples series temporales multivariantes.
- Ya vienen divididos en entrenamiento y prueba.
- Cada serie temporal procede de un motor diferente.
- Hay tres ajustes operativos que tienen un efecto sustancial en el rendimiento del motor (también se incluyen en los datos).
- El motor funciona con normalidad al principio de cada serie temporal y desarrolla un fallo en algún momento de la serie.



Ejercicio de Regresión – Mantenimiento predictivo

Dataset de la NASA

- Los datos se facilitan en forma de archivo de texto comprimido con 26 columnas de números, separadas por espacios.
- Cada fila es una instantánea de los datos tomados durante un único ciclo operativo, cada columna es una variable diferente.
- Las columnas corresponden a:
  1. Número de unidad
  2. Tiempo (en ciclos)
  3. Operational Setting 1 (2 y 3)
  4. Sensor measurement 1 (hasta 26)



Ejercicio de Regresión – Mantenimiento predictivo

Objetivo

Desarrollar un modelo que sea capaz de:

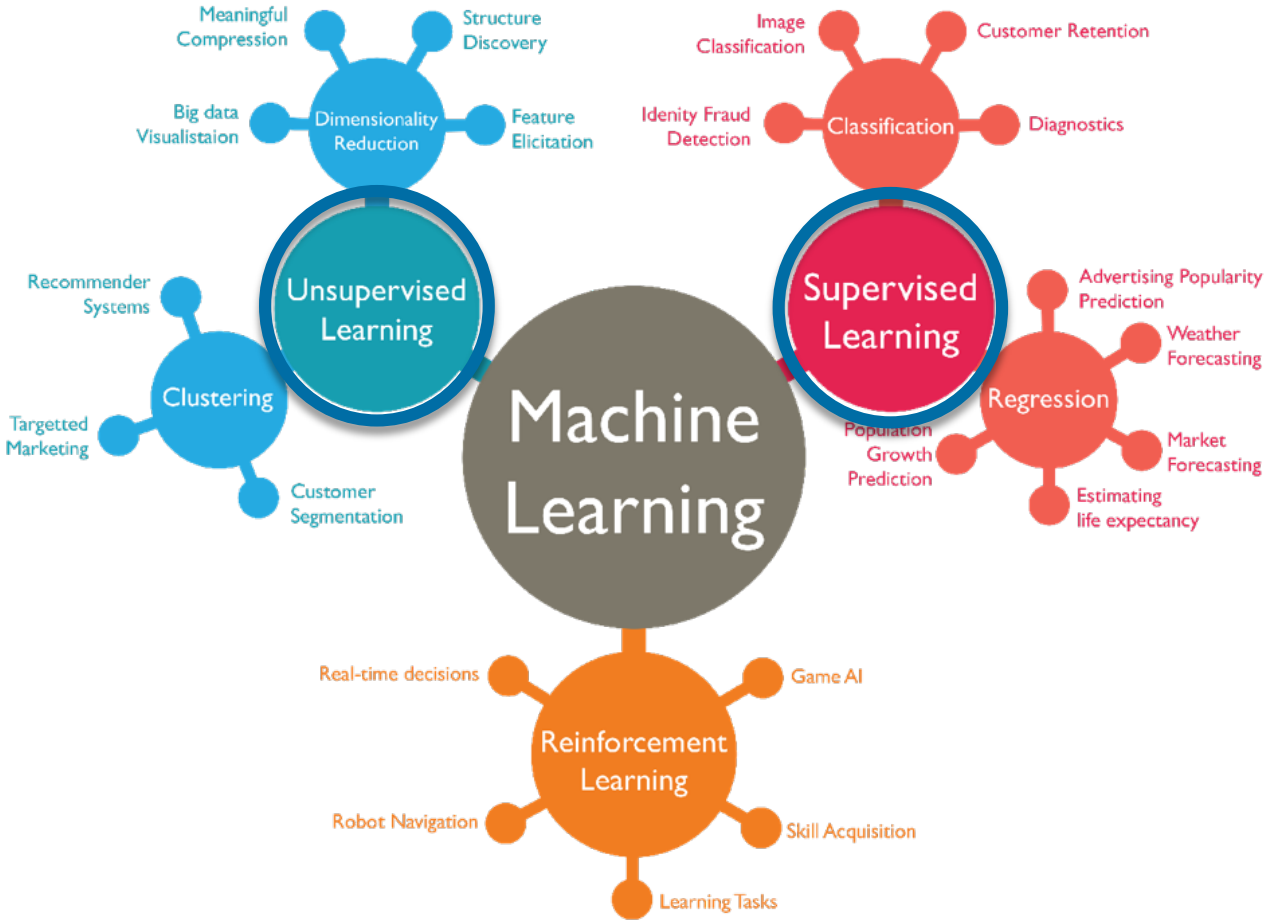
1. Predecir la RUL (en ciclos) según las variables disponibles.
2. Clasificar un punto con la variable objetivo binaria “Últimos 15 Ciclos” (SI o NO).

Preguntas:

¿Qué algoritmos debemos usar?

¿Qué pasos deberemos seguir?

Tipos de Machine Learning



Source: Oracle AI and Data Science Blog.
https://blogs.oracle.com/datascience/types-of-machine-learning-and-top-10-algorithms-everyone-should-know-v2

Machine Learning Pipeline