09481: Inteligencia Artificial

Profesor del curso: Breyner Posso, Ing. M.Sc.
e-mail: breyner.posso1@u.icesi.edu.co

Programa de Ingeniería de Sistemas.
Departamento TIC.
Facultad de Ingeniería.
Universidad Icesi.
Cali, Colombia.

Agenda

- Introducción
- Regresión
- Métricas
- Regresión Lineal
  - Regresión Lineal Simple
  - Regresión Lineal Múltiple
  - Regresión Polinomial

Introducción

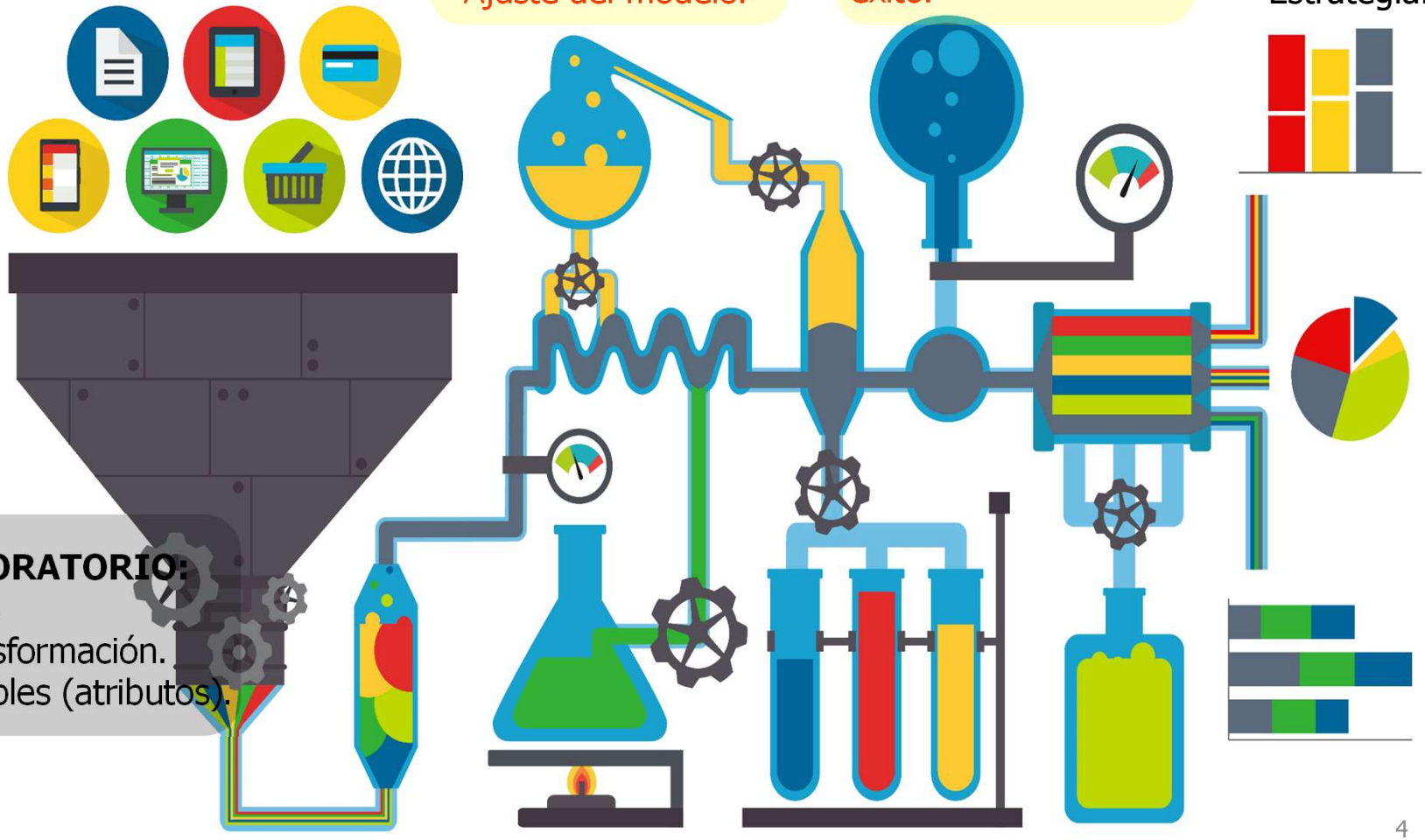
DATOS:
Materia prima.

MODELO:
Implementación del modelo de analítica.
Ajuste del modelo.

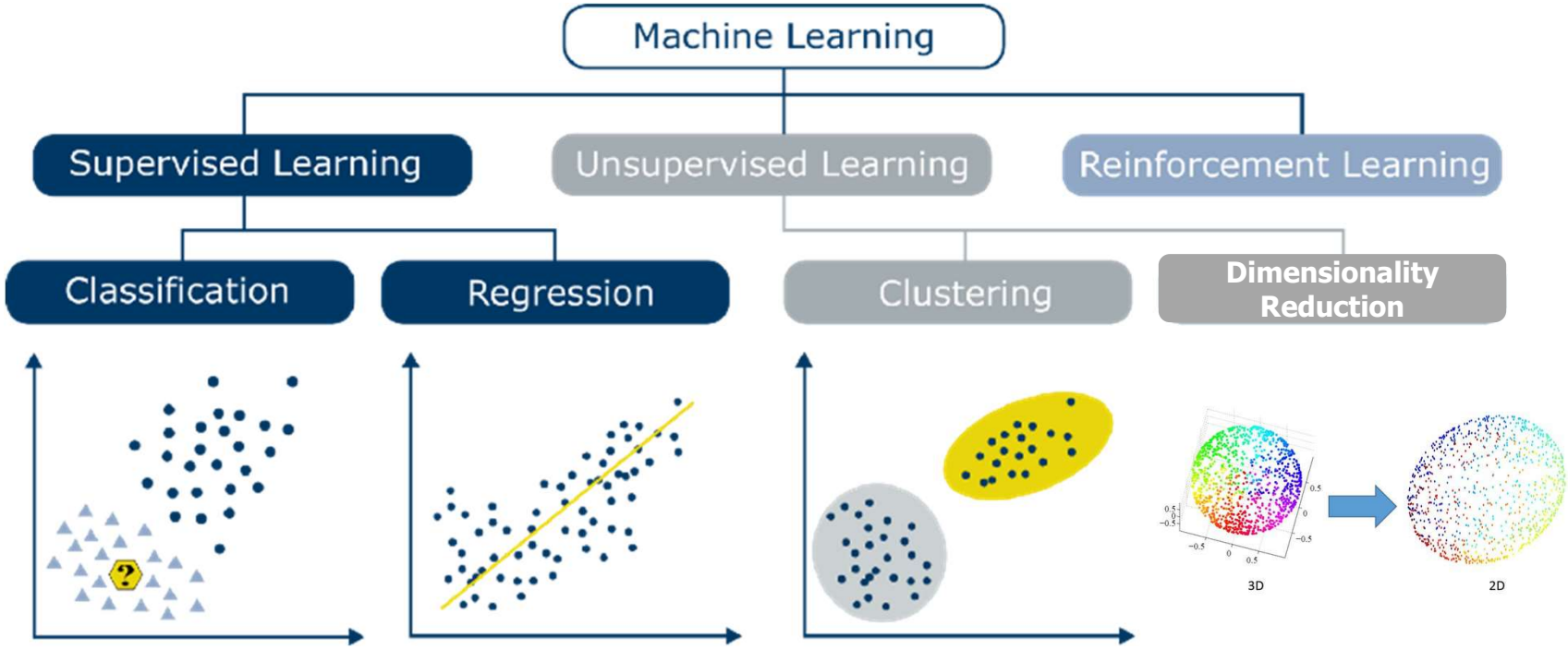
EVALUACION:
Viabilidad del negocio.
Validación criterios de éxito.

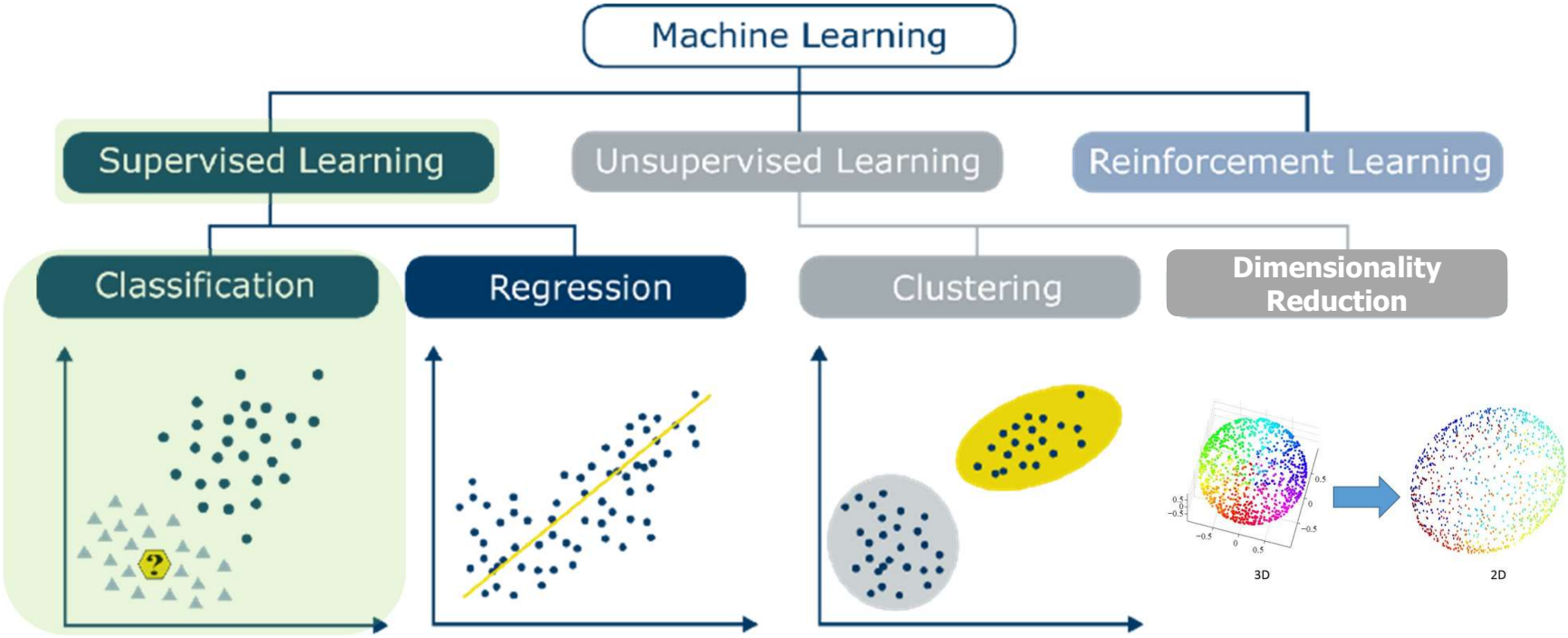
DESPLIEGUE:
Resultados.
Conocimiento.
Estrategia.

PREGUNTAS:
¿Cómo?
¿Cuáles?
¿Cuándo?
¿Por qué? *



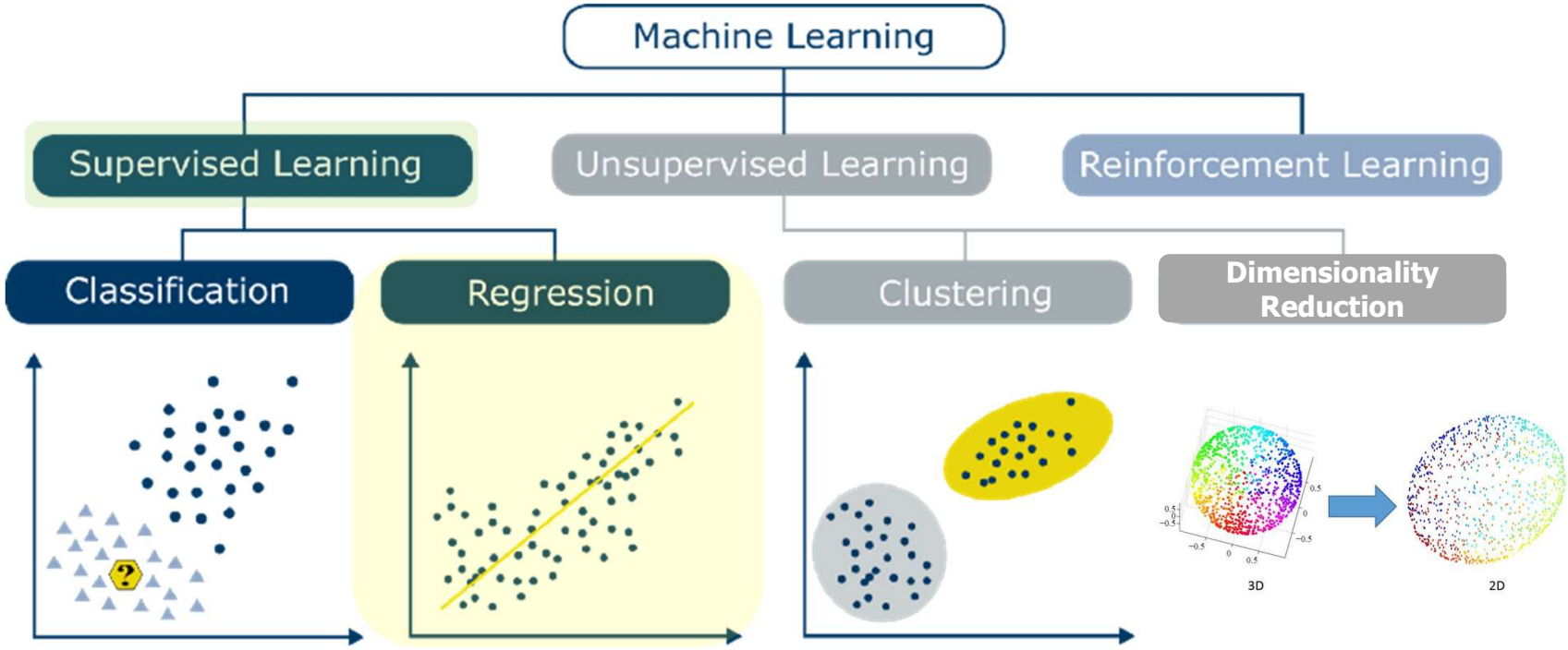
ANALISIS EXPLORATORIO:
Limpieza de datos.
Preparación / transformación.
Selección de variables (atributos).


Machine Learning
Supervised Learning
Unsupervised Learning
Reinforcement Learning
Classification
Regression
Clustering
Dimensionality Reduction
3D
2D



Métodos

- KNN (K-vecinos más cercanos)



Métodos

- KNN (K-vecinos más cercanos)

- Regresión Lineal Simple
- Regresión Lineal Múltiple
- Regresión Polinomial

Aprendizaje supervisado

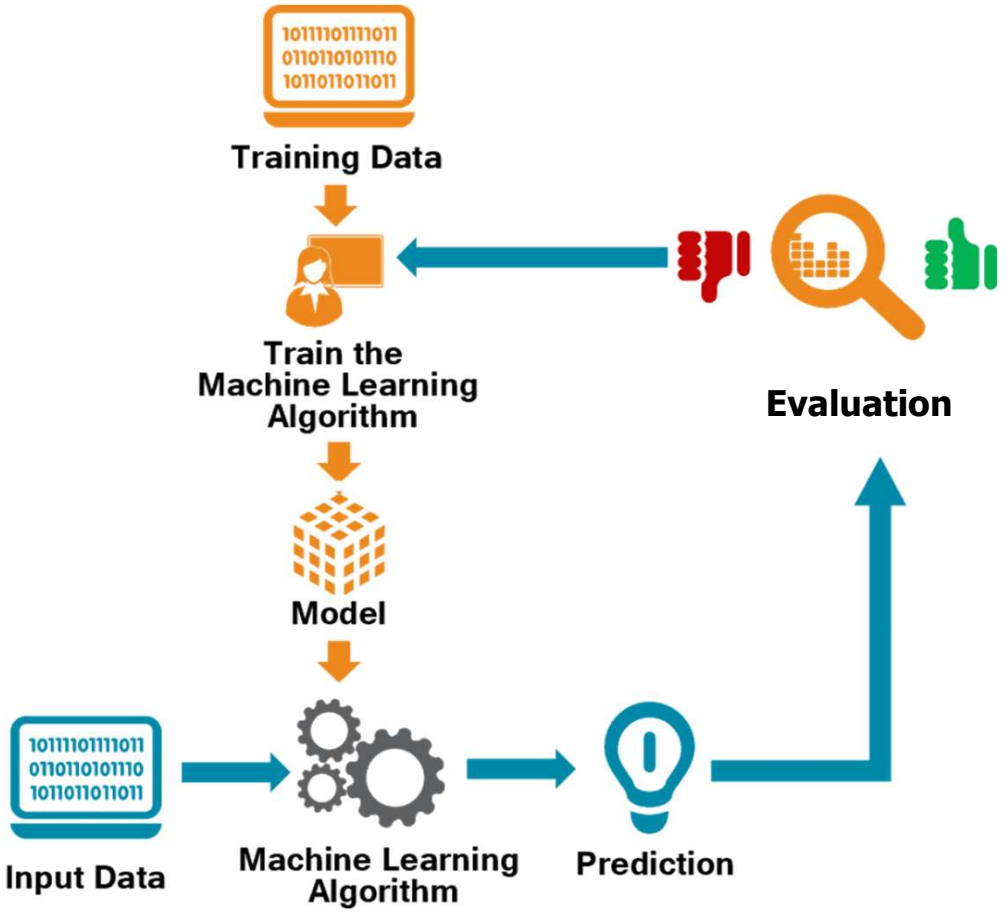
- Conjunto de datos **etiquetados** con una clase o valor:

$(x_1, x_2, \ldots, x_n, \mathbf{y})$

Predictores, variables explicativas, características, atributos o variables independientes.

Variable dependiente, objetivo, o Salida deseada.

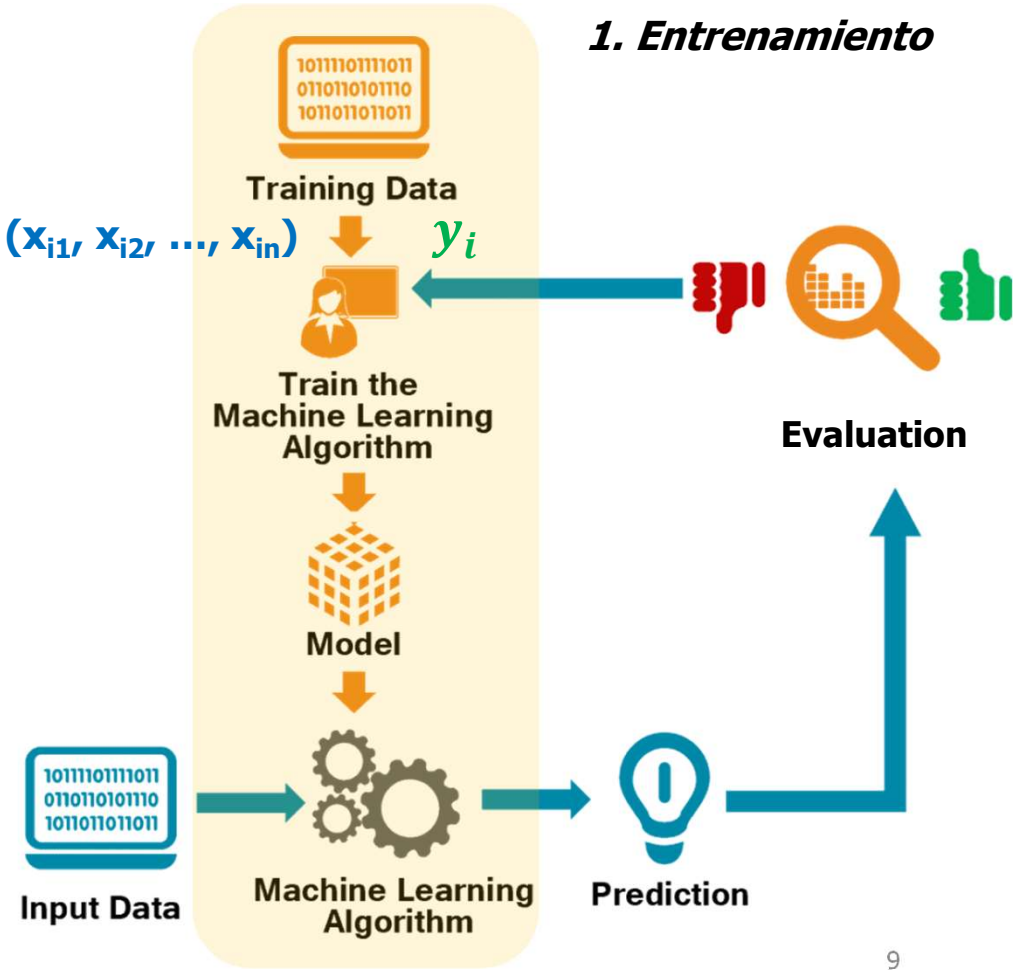
- **Objetivo**: predecir un valor (*Regresión*) o predecir una clase (*Clasificación*).



Aprendizaje supervisado

Fase 1: Entrenamiento

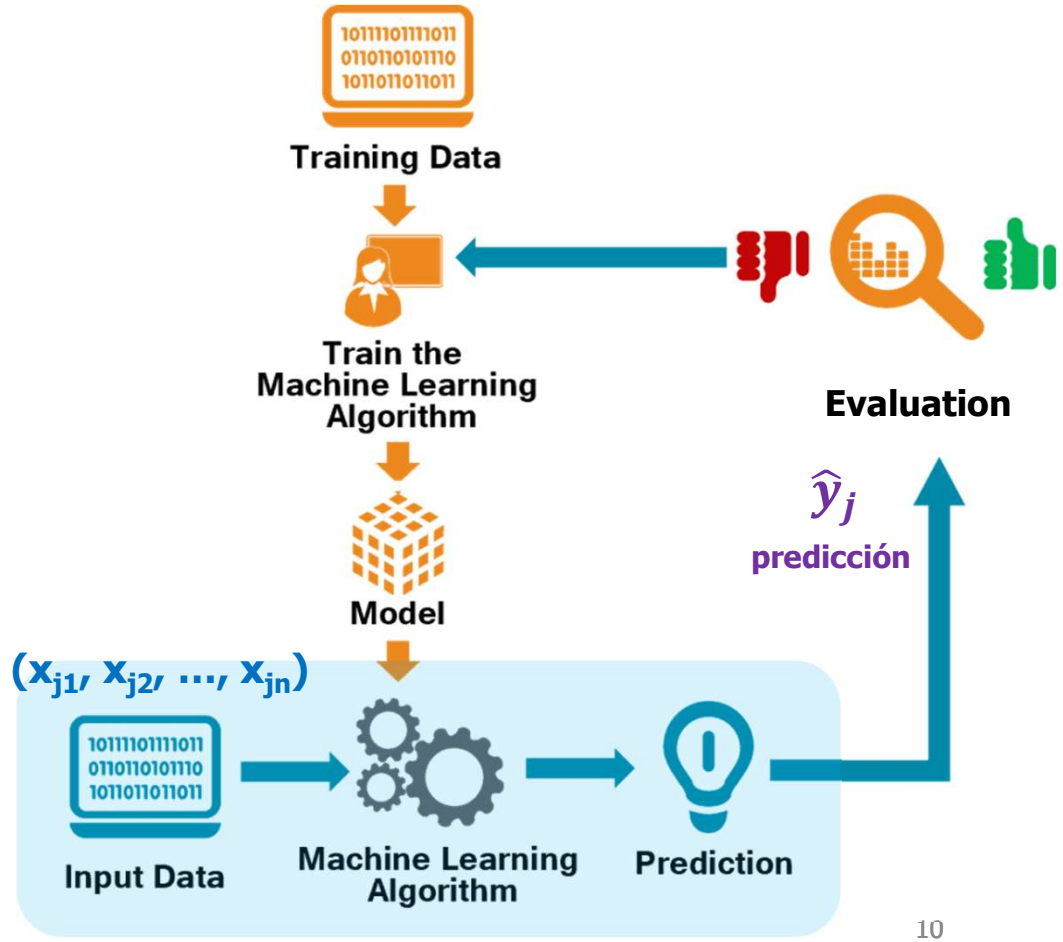
a. ¿Clasificación o regresión?
b. ¿Qué atributos voy a utilizar?
c. ¿Cómo particionar los datos?
d. ¿Qué modelo voy a usar?



Aprendizaje supervisado

Fase 2: Prueba

- Se utilizan datos que no fueron usados para entrenamiento.
- Para cada conjunto de atributos de entrada, se genera una predicción.



2. Prueba

Aprendizaje supervisado

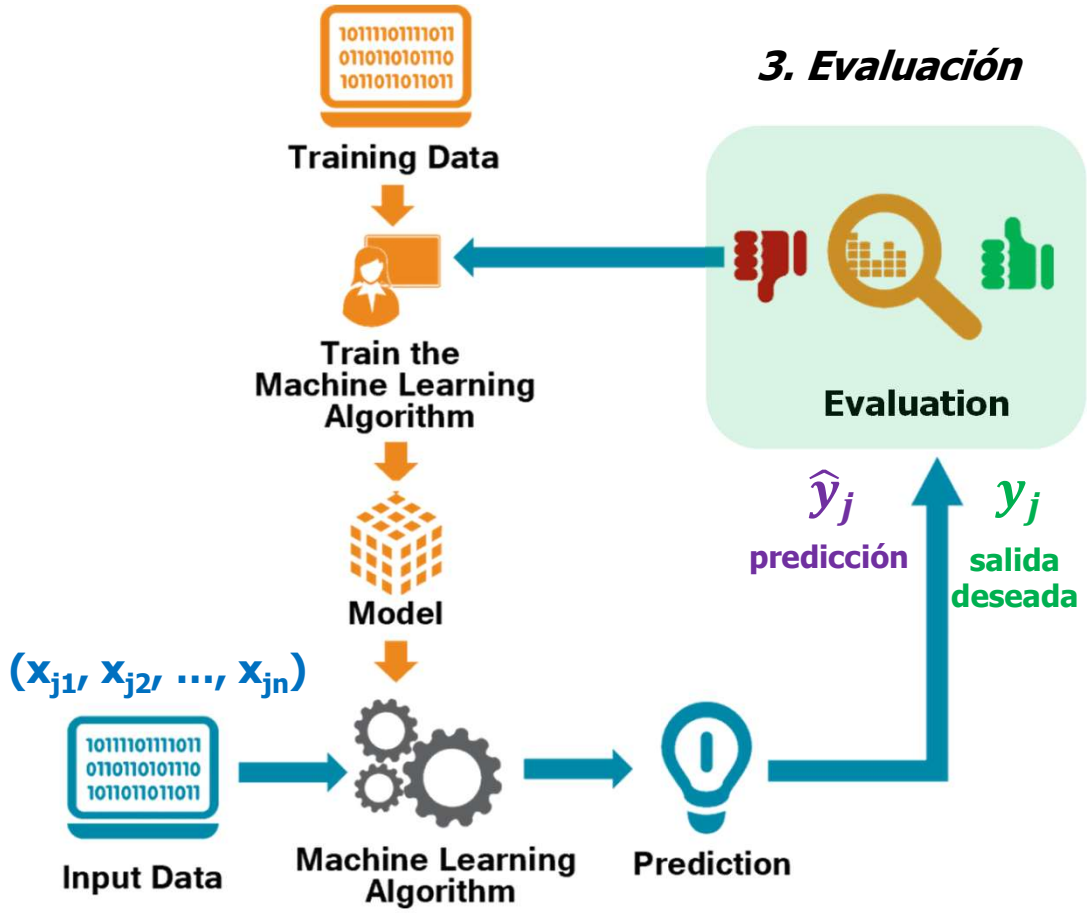
Fase 3: Evaluación

a. ¿Qué métricas voy a utilizar?

b. ¿Cuáles son los criterios para considerar que los resultados son exitosos?

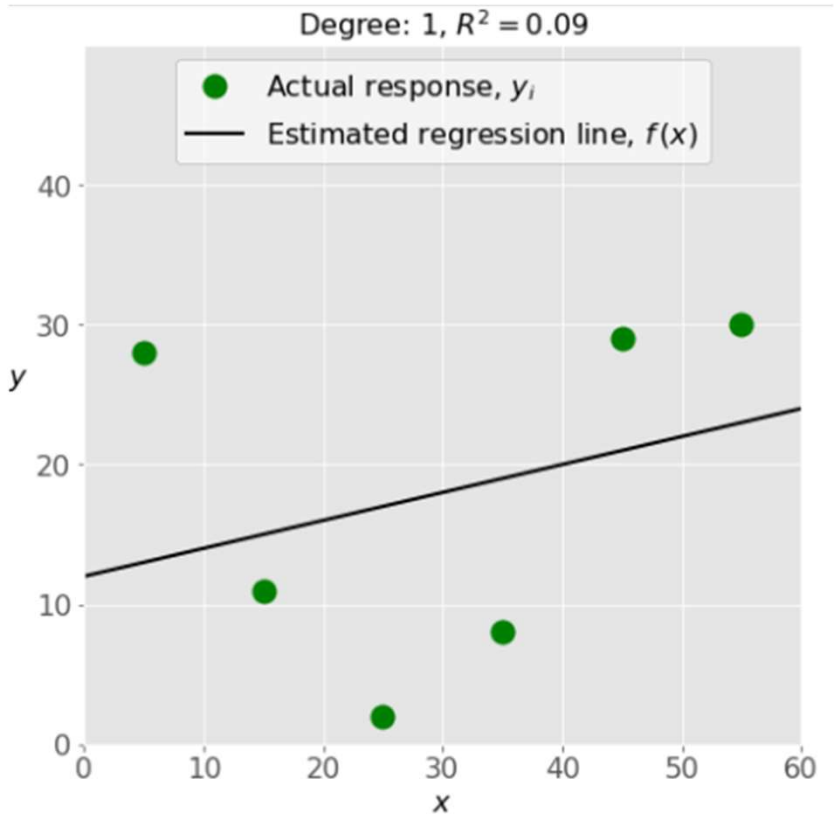
c. ¿Se presenta underfitting (sub entrenamiento) u overfitting (sobre entrenamiento)?

d. Si el modelo no es exitoso, podemos modificar los hiperparámetros o usar otro modelo.



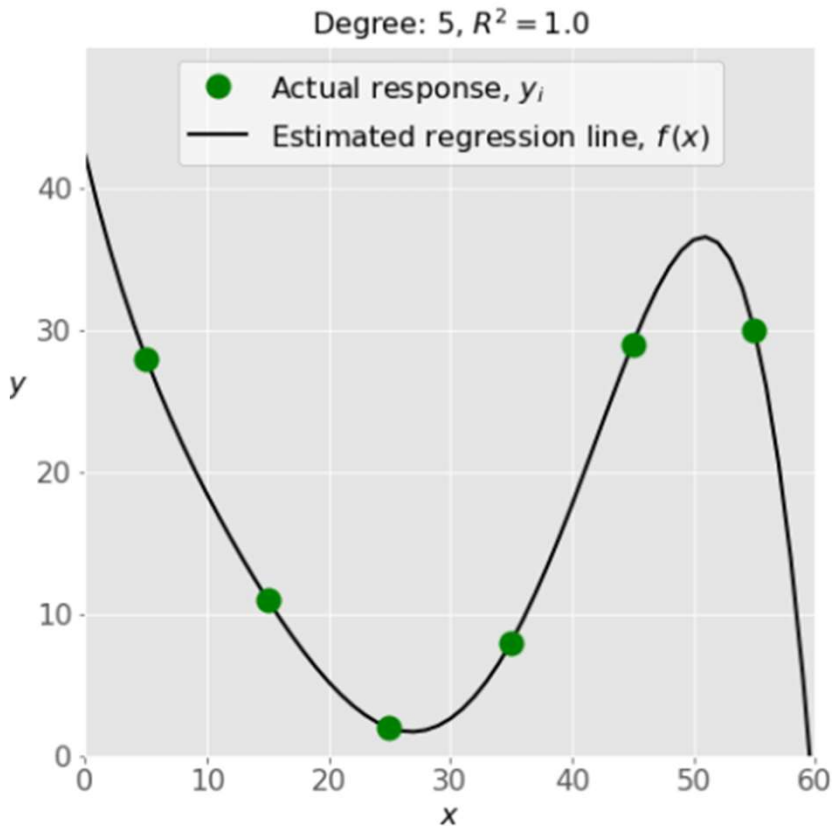
Underfitting and Overfitting

Underfitting: Ocurre cuando el modelo no puede capturar la relación entre la variable objetivo y los predictores. Generalmente, se detecta cuando se obtiene un error alto para los datos de entrenamiento y también para los de prueba.



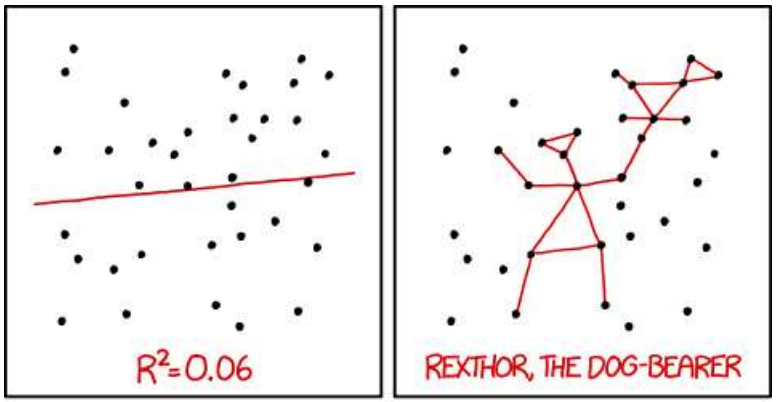
Underfitting and Overfitting

Overfitting: Ocurre cuando el modelo captura demasiado bien la relación entre la variable objetivo y los predictores en el conjunto de entrenamiento. Los modelos complejos que tienen muchos términos tienden al overfitting. Generalmente, se detecta cuando se obtiene un error muy bajo para los datos de entrenamiento y alto para los de prueba.

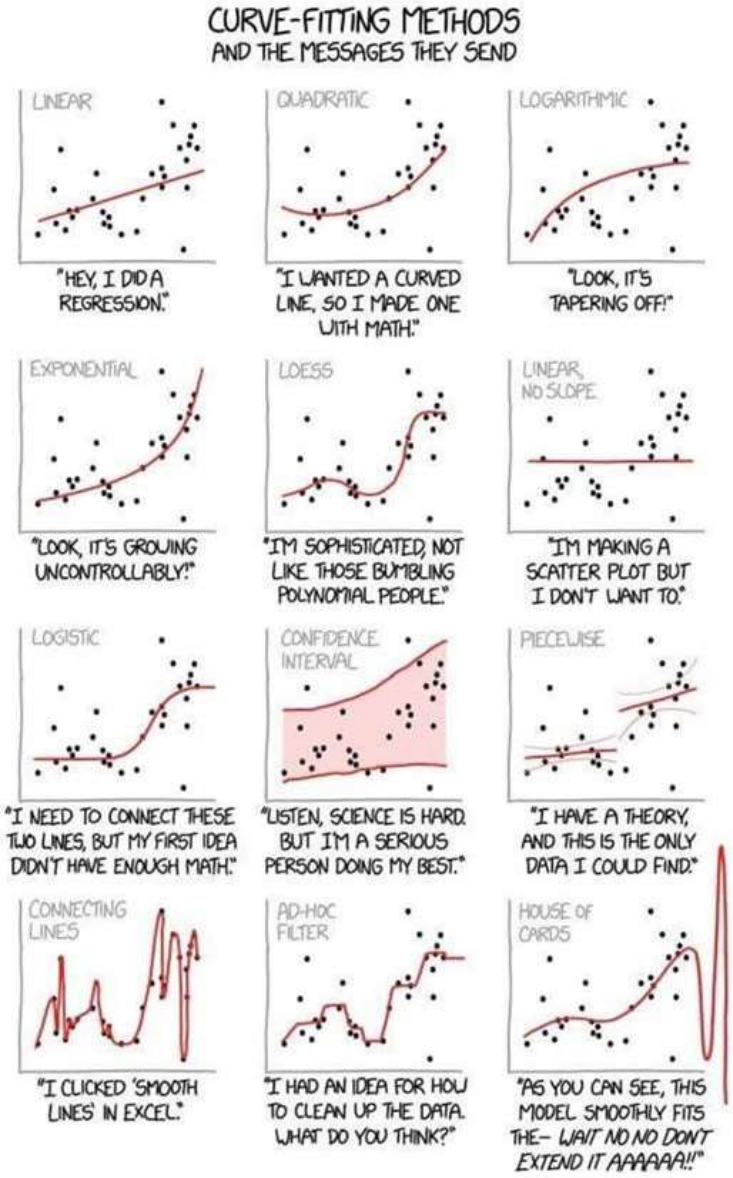


Regresión

Regresión



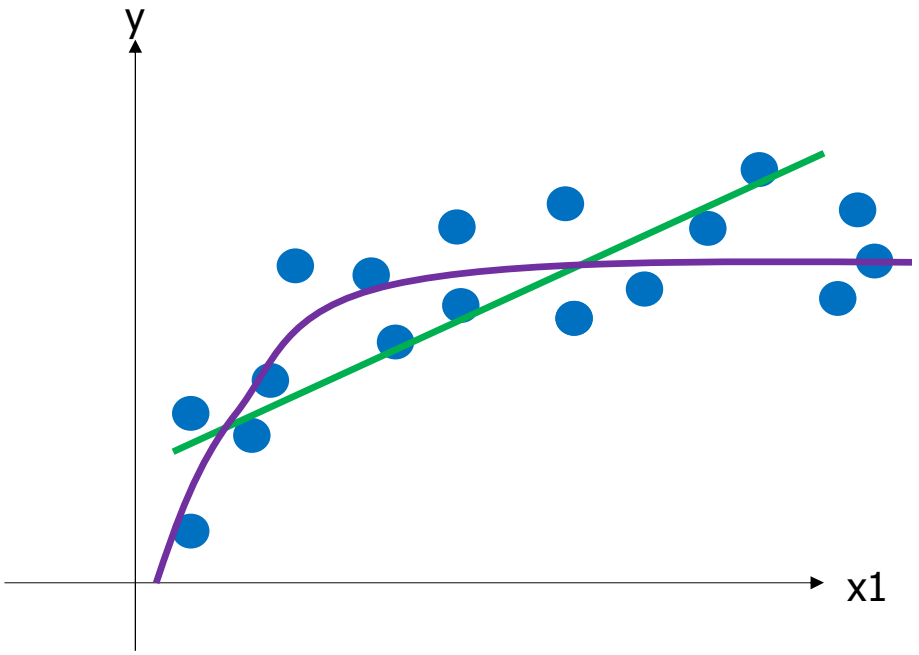
I DON'T TRUST LINEAR REGRESSIONS WHEN IT'S HARDER TO GUESS THE DIRECTION OF THE CORRELATION FROM THE SCATTER PLOT THAN TO FIND NEW CONSTELLATIONS ON IT.



Regresión

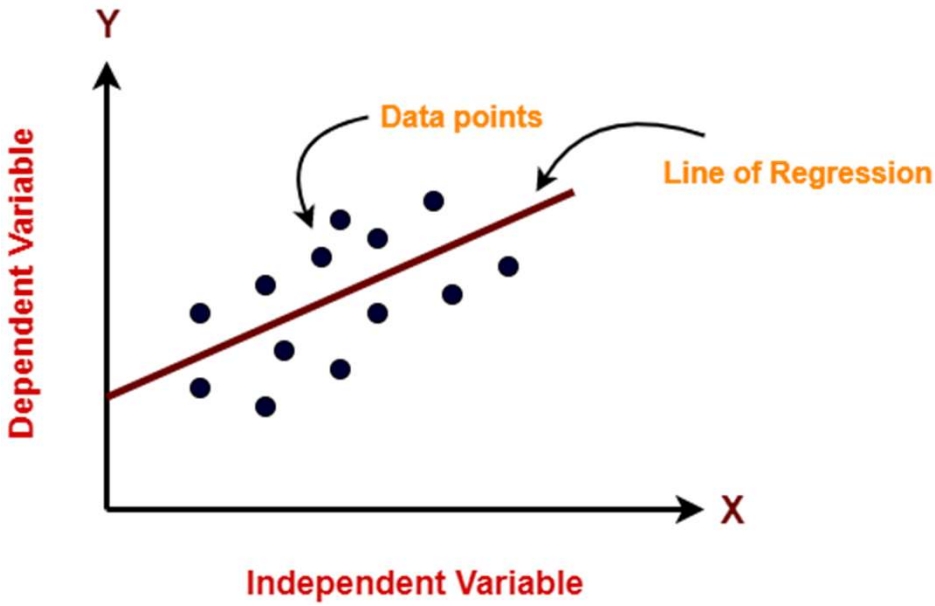
- **Objetivo:** Ajustar modelos para predecir valores **continuos** de la variable objetivo (target) con respecto a una o varias variables independientes (predictores).
- **Métodos:**
  - Regresión lineal (simple y múltiple).
  - Regresión polinomial.
  - RANSAC.
  - KNN.
  - Árboles de regresión.

  …
- **Línea base (*baseline*)**: evaluación dada por un modelo que predice una medida de tendencia central (e.g.: el promedio).



Regresión

- "Regression builds a line or curve that passes through all the datapoints on target-predictor graph in such a way that the vertical distance between the datapoints and the regression curve is minimum."
- La distancia entre los datos y la curva construida indica si el modelo ha capturado la relación entre los predictores y la variable objetivo.
- Esta distancia se denomina Residuo.

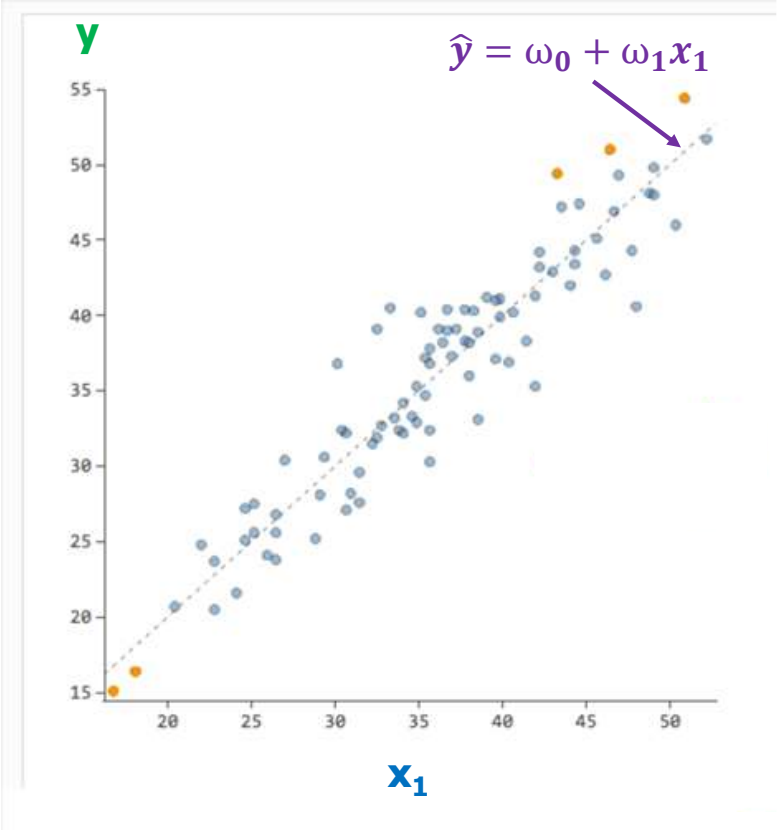


Regresión

- Residuo (e) = valor observado de salida – valor predicho:

$$\boldsymbol{e = y - \hat{y}}$$

- Los modelos de regresión buscan minimizar el valor de $\boldsymbol{e}$ para todo el conjunto de predictores de entrenamiento.

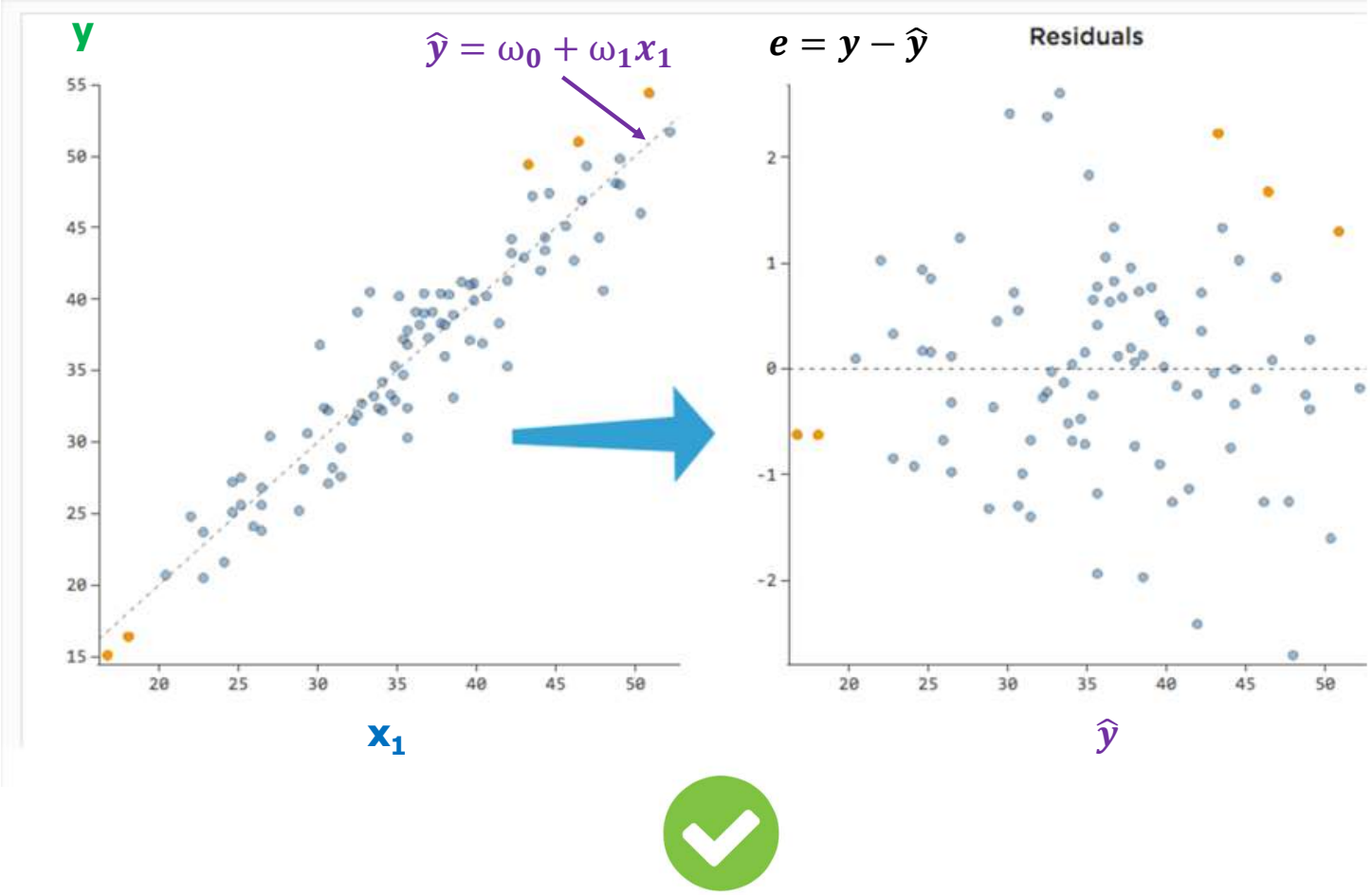


Regresión

- Residuo (e) = valor observado de salida – valor predicho:

$$e = y - \hat{y}$$

- La gráfica de los residuos puede ayudarnos a identificar si el modelo de regresión ha capturado la relación entre la variable objetivo y los predictores.

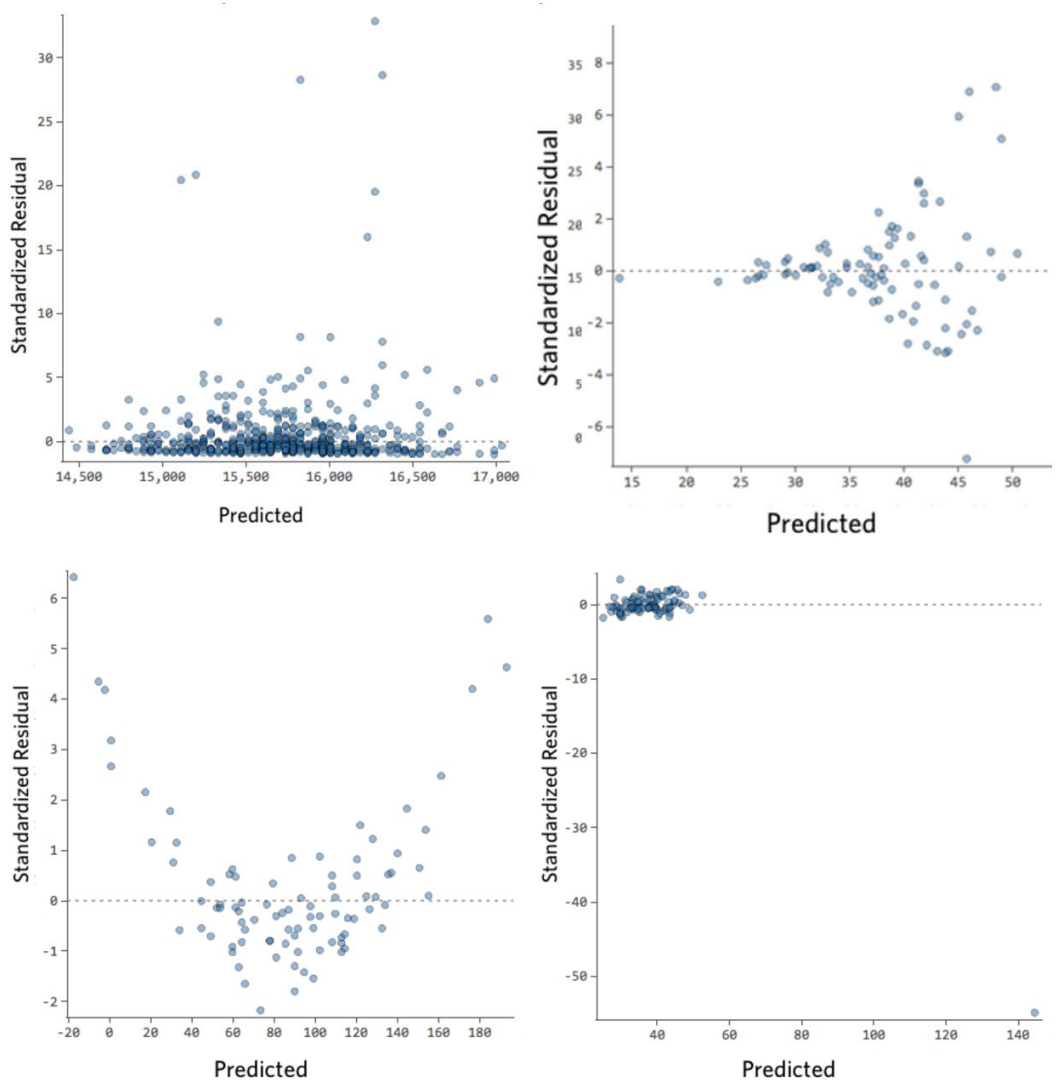


Regresión

- Residuo (e) = valor observado de salida – valor predicho:

$$\boldsymbol{e = y - \hat{y}}$$

- La gráfica de los residuos puede ayudarnos a identificar si el modelo de regresión ha capturado la relación entre la variable objetivo y los predictores.



Regresión: métricas de evaluación

1. **Coeficiente de correlación de Pearson** ($\rho \in [-1,1]$).

- Es una medida estadística de dependencia lineal entre dos variables aleatorias cuantitativas. A diferencia de la covarianza, la correlación de Pearson es independiente de la escala de medida de las variables.

Regresión: métricas de evaluación

1. Coeficiente de correlación de Pearson ($\rho \in [-1,1]$).

Sean X e Y dos variables aleatorias sobre una población:

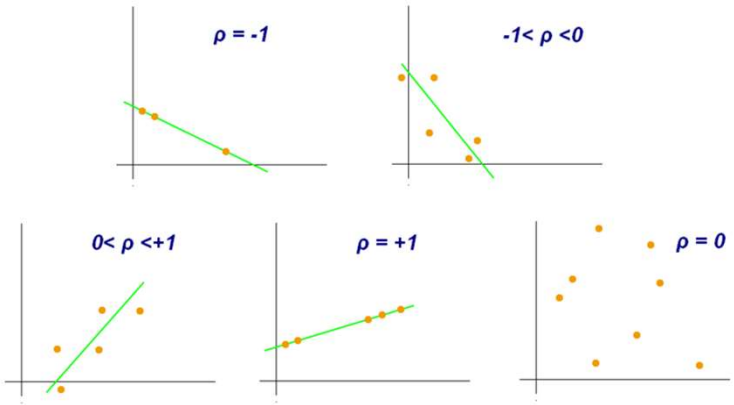
$$\rho_{X,Y} = \frac{Cov(X,Y)}{\sigma_X \sigma_Y}$$

Donde:

$$\operatorname{Cov}(X,Y) = \operatorname{E}\left[(X - \operatorname{E}[X])(Y - \operatorname{E}[Y])\right]$$

$$\sigma = \sqrt{\operatorname{E}[(X - \operatorname{E}[X])^2]}$$

- $|\rho| = 0$ no hay correlación lineal.
- $|\rho| = 0.10$ correlación muy débil.
- $|\rho| = 0.25$ correlación débil.
- $|\rho| = 0.50$ correlación media.
- $|\rho| = 0.75$ correlación fuerte.
- $|\rho| = 0.90$ correlación muy fuerte.
- $|\rho| = 1$ correlación perfecta.



Fuente: wikipedia.org

Regresión: métricas de evaluación

1. **Coeficiente de correlación de Pearson** ($\rho \in [-1,1]$).

Sean X e Y dos variables aleatorias sobre una población:

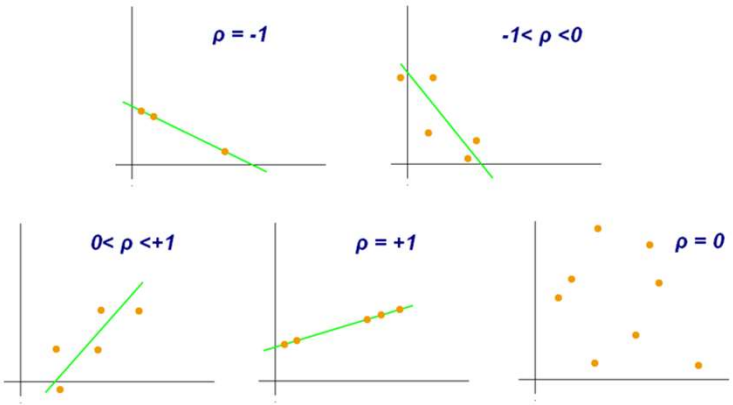
$$\rho_{X,Y} = \frac{Cov(X,Y)}{\sigma_X \sigma_Y}$$

Donde:

$$\operatorname{Cov}(X,Y) = \operatorname{E}\left[(X - \operatorname{E}[X])(Y - \operatorname{E}[Y])\right]$$

$$\sigma = \sqrt{\operatorname{E}[(X - \operatorname{E}[X])^2]}$$

- $|\rho| = 0$ no hay correlación lineal.
- $|\rho| = 0.10$ correlación muy débil.
- $|\rho| = 0.25$ correlación débil.
- $|\rho| = 0.50$ correlación media.
- $|\rho| = 0.75$ correlación fuerte.
- $|\rho| = 0.90$ correlación muy fuerte.
- $|\rho| = 1$ correlación perfecta.

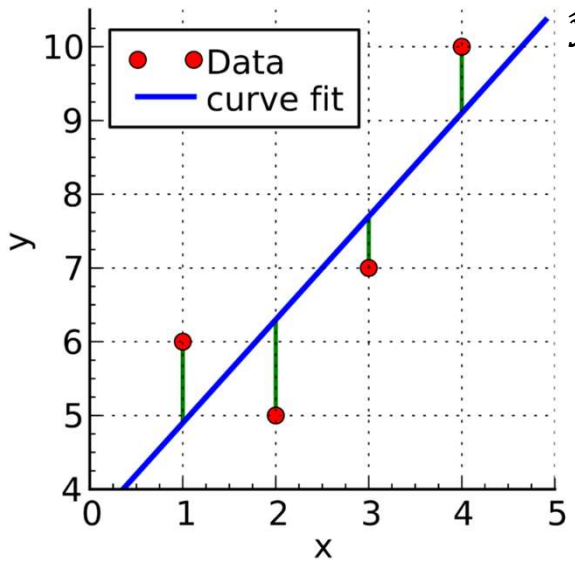


Fuente: wikipedia.org

- **En la regresión lineal, el coeficiente de determinación** R^2 es igual a ρ^2, e indica el porcentaje de la varianza que pudo ser explicada por los predictores a partir de la relación lineal.

Regresión: métricas de evaluación

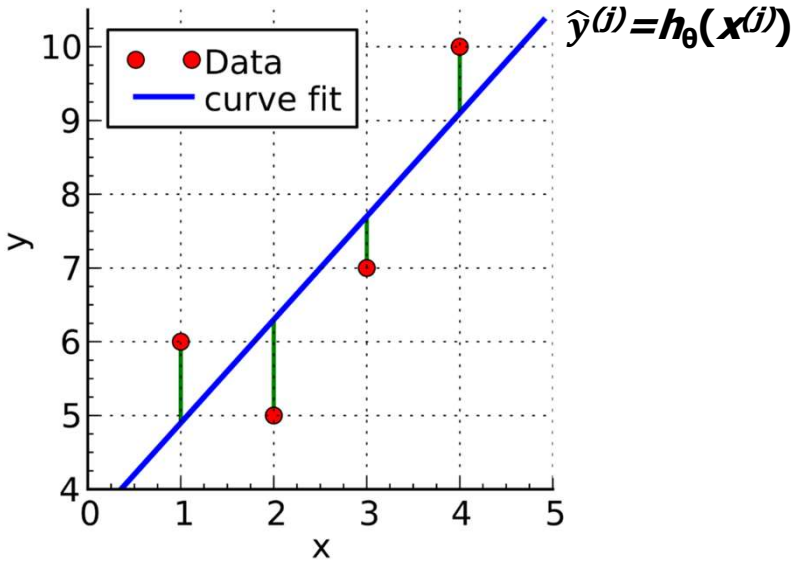
Sea:
$\boldsymbol{m}$: el número de observaciones o ejemplos.
$\boldsymbol{x^{(j)}}$: la j-ésima observación de entrada.
$\hat{y}^{(j)} = \boldsymbol{h_\theta(x^{(j)})}$: la j-ésima predicción de salida.
$\boldsymbol{y^{(j)}}$: el j-ésimo valor de la salida observada.



$\hat{y}^{(j)} = \boldsymbol{h_\theta(x^{(j)})}$

Regresión: métricas de evaluación

Sea:
m: el número de observaciones o ejemplos.
$\boldsymbol{x^{(j)}}$: la *j*-ésima observación de entrada.
$\hat{y}^{(j)}=\boldsymbol{h_\theta(x^{(j)})}$: la *j*-ésima predicción de salida.
$\boldsymbol{y^{(j)}}$: el *j*-ésimo valor de la salida observada.



$\hat{y}^{(j)}=\boldsymbol{h_\theta(x^{(j)})}$

2. MAE (*Mean Absolute Error*):

$$MAE = \frac{1}{m}\sum_{j=1}^{m}\left|y^{(j)} - h_\theta\left(x^{(j)}\right)\right|$$

3. MSE (M*ean Square Error*):

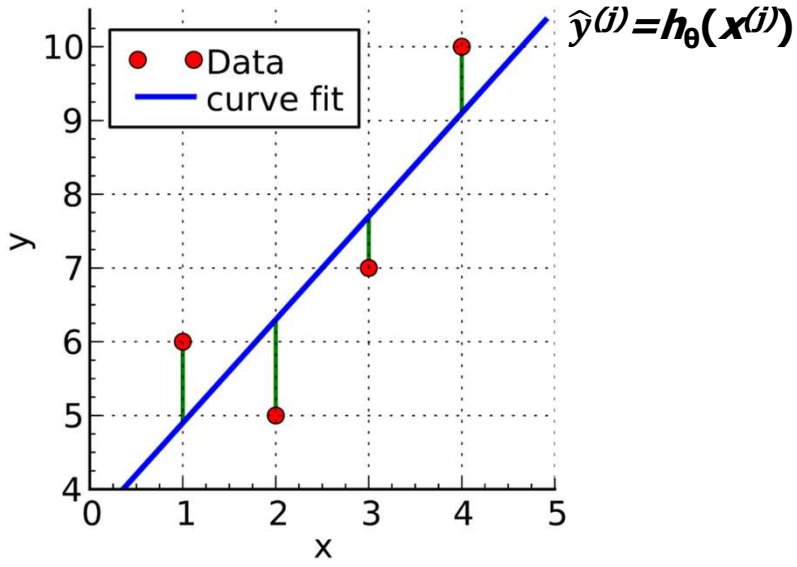
$$MSE = \frac{1}{m}\sum_{j=1}^{m}\left[\left(y^{(j)} - h_\theta\left(x^{(j)}\right)\right)^2\right]$$

4. RMSE (*Root Mean Square Error*):

$$RMSE = \sqrt{\frac{1}{m}\sum_{j=1}^{m}\left[\left(y^{(j)} - h_\theta\left(x^{(j)}\right)\right)^2\right]}$$

Regresión: métricas de evaluación

Sea:
m: el número de observaciones o ejemplos.
$\boldsymbol{x^{(j)}}$: la *j*-ésima observación de entrada.
$\hat{y}^{(j)} = \boldsymbol{h_\theta(x^{(j)})}$: la *j*-ésima predicción de salida.
$\boldsymbol{y^{(j)}}$: el *j*-ésimo valor de la salida observada.



$\hat{y}^{(j)} = \boldsymbol{h_\theta(x^{(j)})}$

5. R^2 (*coeficiente de determinación*), donde $\bar{y}$ es la media de las salidas:

$$R^2 = 1 - \frac{\sum_{j=1}^{m}\left[\left(y^{(j)} - h_\theta\left(x^{(j)}\right)\right)^2\right]}{\sum_{j=1}^{m}\left[\left(y^{(j)} - \bar{y}\right)^2\right]}$$

6. R^2 ajustado (penaliza el número *p* de <u>variables</u> independientes):

$$R^2_{adj} = 1 - \left[\left(1 - R^2\right)\frac{(m-1)}{(m-p-1)}\right]$$

Regresión: variables categóricas

- En muchas librerías, las variables predictoras deben ser numéricas. Por lo tanto, debemos convertir las variables categóricas a variables numéricas, para lo cual existen varias opciones:

Regresión: variables categóricas

- En muchas librerías, las variables predictoras deben ser numéricas. Por lo tanto, debemos convertir las variables categóricas a variables numéricas, para lo cual existen varias opciones:

1. **Reemplazar valores:** reemplazar las categorías con números "deseados". Usar *replace* en pandas: https://pandas.pydata.org/pandasdocs/stable/reference/api/pandas.DataFrame.replace.html

Regresión: variables categóricas

- En muchas librerías, las variables predictoras deben ser numéricas. Por lo tanto, debemos convertir las variables categóricas a variables numéricas, para lo cual existen varias opciones:

1. **Reemplazar valores:** reemplazar las categorías con números "deseados". Usar *replace* en pandas: https://pandas.pydata.org/pandasdocs/stable/reference/api/pandas.DataFrame.replace.html

2. **Codificar etiquetas (*label encoding*):** convierte cada valor de una columna en un número. Las etiquetas numéricas están siempre entre 0 y (número de categorías - 1). Usar *LabelEncoder* en pandas: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html

Regresión: variables categóricas

- En muchas librerías, las variables predictoras deben ser numéricas. Por lo tanto, debemos convertir las variables categóricas a variables numéricas, para lo cual existen varias opciones:

1. **Reemplazar valores:** reemplazar las categorías con números "deseados". Usar *replace* en pandas: https://pandas.pydata.org/pandasdocs/stable/reference/api/pandas.DataFrame.replace.html

2. **Codificar etiquetas (*label encoding*):** convierte cada valor de una columna en un número. Las etiquetas numéricas están siempre entre 0 y (número de categorías - 1). Usar *LabelEncoder* en pandas: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html

3. **Codificación 1 de n (*one-hot encoding*):** convierte cada valor de la categoría en una nueva columna y le asigna un valor de 1 o 0 (verdadero / falso). Tiene como ventaja no ponderar un valor de forma incorrecta. Usar *get_dummies()* en pandas: https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.get_dummies.html

Regresión: variables categóricas

- En muchas librerías, las variables predictoras deben ser numéricas. Por lo tanto, debemos convertir las variables categóricas a variables numéricas, para lo cual existen varias opciones:

Codificar etiquetas

Breakfast
Every day
Never
Rarely
Most days
Never

→

Breakfast
3
0
1
2
0

Codificación *one-hot*

Color
Red
Red
Yellow
Green
Yellow

→

Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0

Regresión: variables categóricas

- En muchas librerías, las variables predictoras deben ser numéricas. Por lo tanto, debemos convertir las variables categóricas a variables numéricas, para lo cual existen varias opciones:

Codificar etiquetas

Breakfast
Every day
Never
Rarely
Most days
Never

→

Breakfast
3
0
1
2
0

Codificación *one-hot*

Color
Red
Red
Yellow
Green
Yellow

→

Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0

4. Codificación binaria: primero las categorías se codifican como ordinales, luego esos enteros se convierten en código binario, luego los dígitos de esa cadena binaria se dividen en columnas separadas. Esto codifica los datos en menos dimensiones que one-hot. Usar BinaryEncoder de la librería category_encoders: https://contrib.scikit-learn.org/category_encoders/binary.html

Regresión: variables categóricas

- En muchas librerías, las variables predictoras deben ser numéricas. Por lo tanto, debemos convertir las variables categóricas a variables numéricas, para lo cual existen varias opciones:

Codificar etiquetas

Breakfast
Every day
Never
Rarely
Most days
Never

→

Breakfast
3
0
1
2
0

Codificación *one-hot*

Color
Red
Red
Yellow
Green
Yellow

→

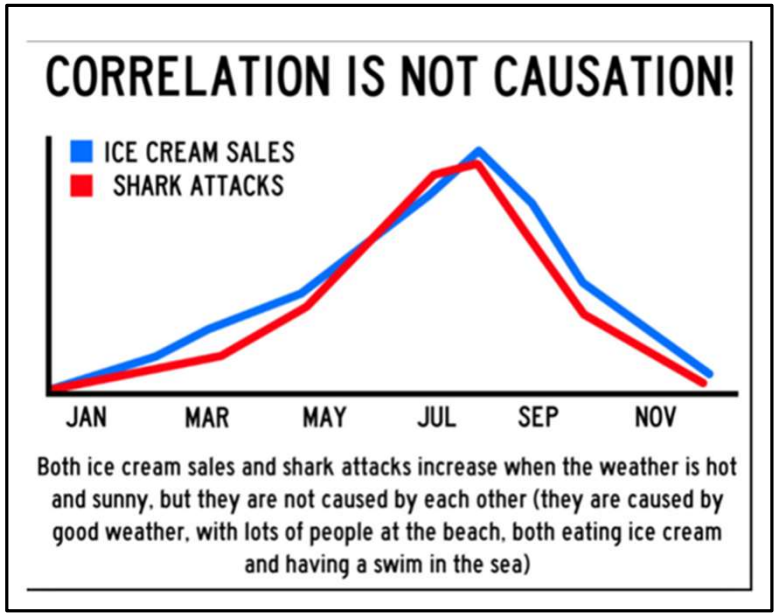
Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0

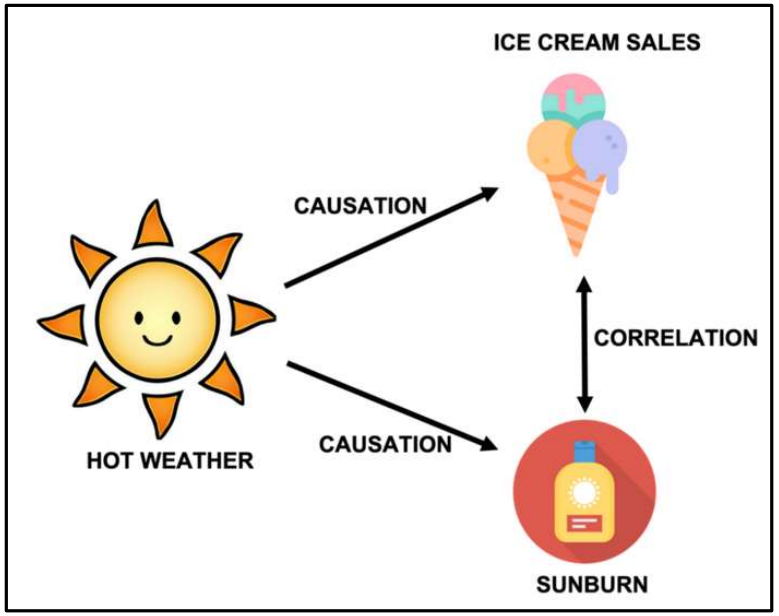
4. Codificación binaria: primero las categorías se codifican como ordinales, luego esos enteros se convierten en código binario, luego los dígitos de esa cadena binaria se dividen en columnas separadas. Esto codifica los datos en menos dimensiones que one-hot. Usar BinaryEncoder de la librería category_encoders: https://contrib.scikit-learn.org/category_encoders/binary.html

Nota: en Python, es una buena práctica cambiar el *dtype* a "*category*" en las columnas que representan atributos categóricos, dado que las operaciones sobre este tipo de columnas son más rápidas que cuando tienen "*object*" como su *dtype*. Puede hacerlo usando el método *.astype()* en las columnas que lo requieran.

Regresión

La correlación no implica causalidad. Son conceptos diferentes!

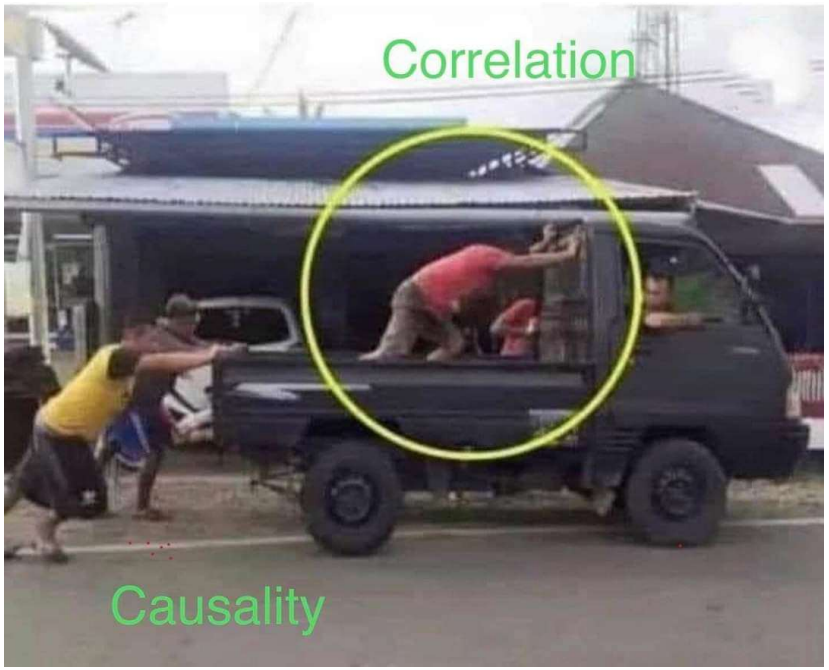


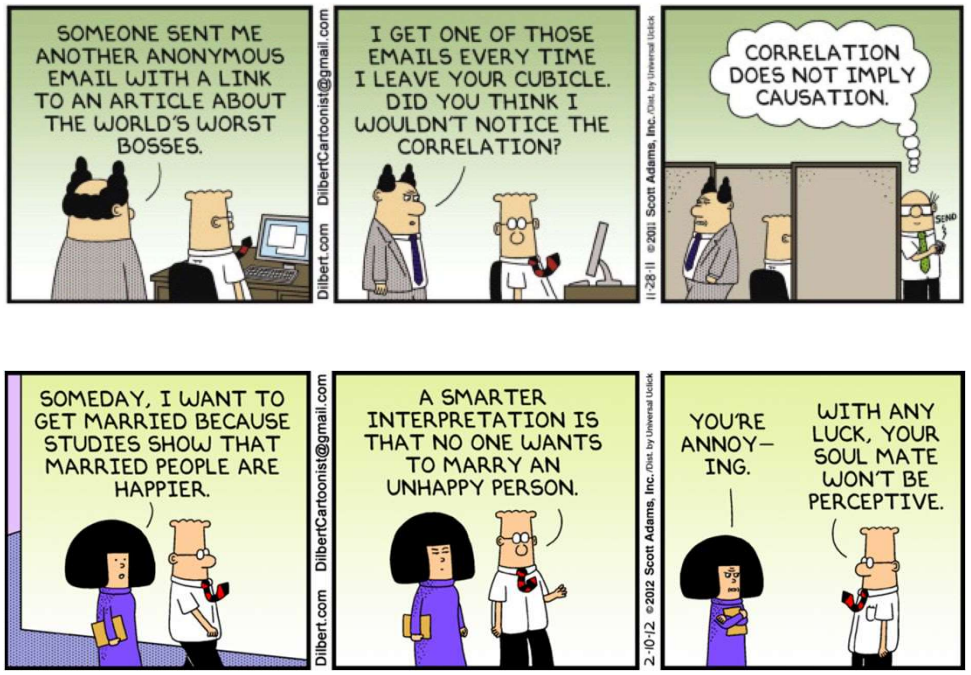


Fuentes: Quora, towards data science.

Regresión

La correlación no implica causalidad. Son conceptos diferentes!





Regresión Lineal: simple, múltiple y polinomial

Regresión Lineal Simple

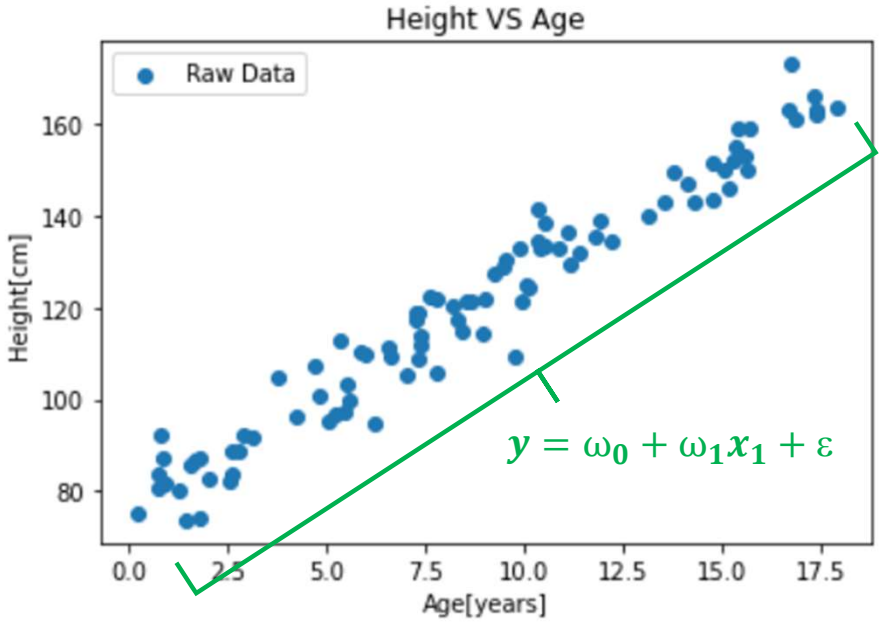
- Es un modelo de regresión que estima una relación lineal entre una variable objetivo (Target) y una sola variable independiente o predictor.

Regresión Lineal Simple

- Cuando se usa este modelo se supone que los datos tienen un comportamiento dado por:

$$\boldsymbol{y^{(j)} = \omega_0 + \omega_1 x_1^{(j)} + \varepsilon}$$

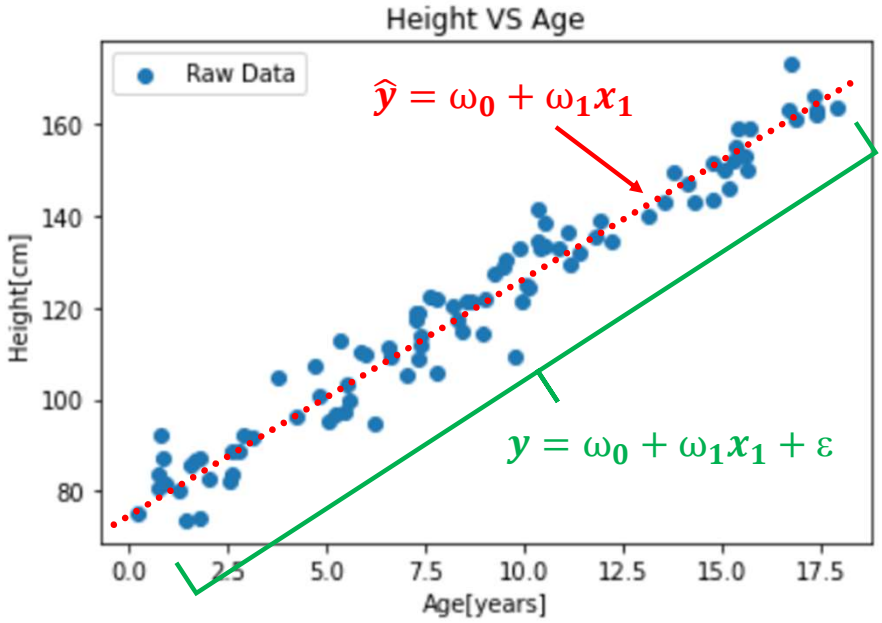
- Dónde ε es el ruido presente en los datos.
- $x_i^{(j)}$ se interpreta como el valor de la *i*-ésima variable independiente de la la *j*-ésima observación. Se asume que el conjunto de datos de entrenamiento tiene *m* observaciones.
- $y^{(j)}$se interpreta como el valor del Target de la la *j*-ésima observación.



Regresión Lineal Simple

- El modelo generado es una ecuación de una línea recta de la forma:

$$\hat{y}^{(j)} = \omega_0 + \omega_1 x_1^{(j)}$$



Regresión Lineal Simple

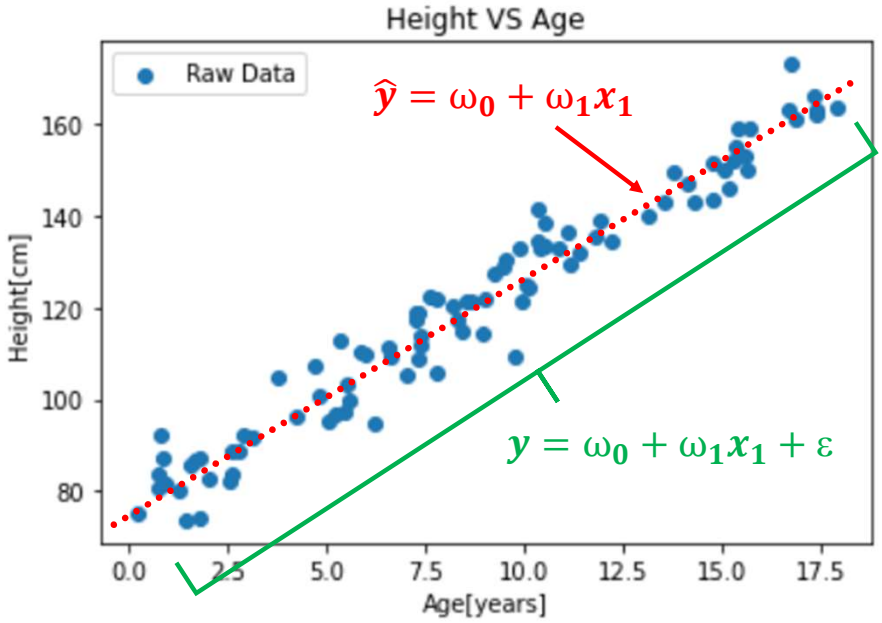
- El modelo generado es una ecuación de una línea recta de la forma:

$$\hat{y}^{(j)} = \omega_0 + \omega_1 x_1^{(j)}$$

- Se puede observar que los residuos:

$$e = y^{(j)} - \hat{y}^{(j)}$$
$$e = \omega_0 + \omega_1 x_1^{(j)} + \varepsilon - (\omega_0 + \omega_1 x_1^{(j)})$$
$$e = \text{?}$$

- Representan el ruido que tienen los datos.



Regresión Lineal Múltiple

- Es un modelo de regresión que estima una relación lineal entre una variable objetivo (Target) y una **varias variables** independientes o predictores.

Regresión Lineal Múltiple

- Cuando se usa este modelo se supone que los datos tienen un comportamiento dado por:

$$\boldsymbol{y^{(j)} = \omega_0 + \omega_1 x_1^{(j)} w_2 x_2^{(j)} + \omega_i x_i^{(j)} \ldots + \omega_n x_n^{(j)}} + \varepsilon$$

- Dónde ε es el ruido presente en los datos.

- $\boldsymbol{x_i^{(j)}}$ se interpreta como el valor de la *i*-ésima variable independiente de la la *j*-ésima observación. Se asume que el conjunto de datos de entrenamiento tiene *m* observaciones.

- $\boldsymbol{y^{(j)}}$se interpreta como el valor del Target de la la *j*-ésima observación.

- $\boldsymbol{n}$ es el número de predictores utilizado para construir el modelo.

Regresión Lineal Múltiple

- El modelo generado es una ecuación de la forma.

$$\hat{\boldsymbol{y}}^{(j)} = \boldsymbol{\omega_0} + \boldsymbol{\omega_1 x_1^{(j)} w_2 x_2^{(j)}} + \boldsymbol{\omega_i x_i^{(j)}} \ldots + \boldsymbol{\omega_n x_n^{(j)}}$$

- Los parámetros $\boldsymbol{\omega_i}$ se estiman teniendo como objetivo la minimización de la suma de los residuos al cuadrado. Donde los residuos son las diferencias entre los los valores reales ($\boldsymbol{y}^{(j)}$) y las predicciones ($\hat{\boldsymbol{y}}^{(j)}$):

$$\mathbf{arg\,min} \sum_{j=1}^{m} \underbrace{[y^{(j)} - (\omega_0 + \omega_1 x_1^{(j)} + \omega_2 x_2^{(j)} + \omega_i x_i^{(j)} \ldots + \omega_n x_n^{(j)})]^2}_{\textit{Residual Sum of Squares (RSS)}}$$

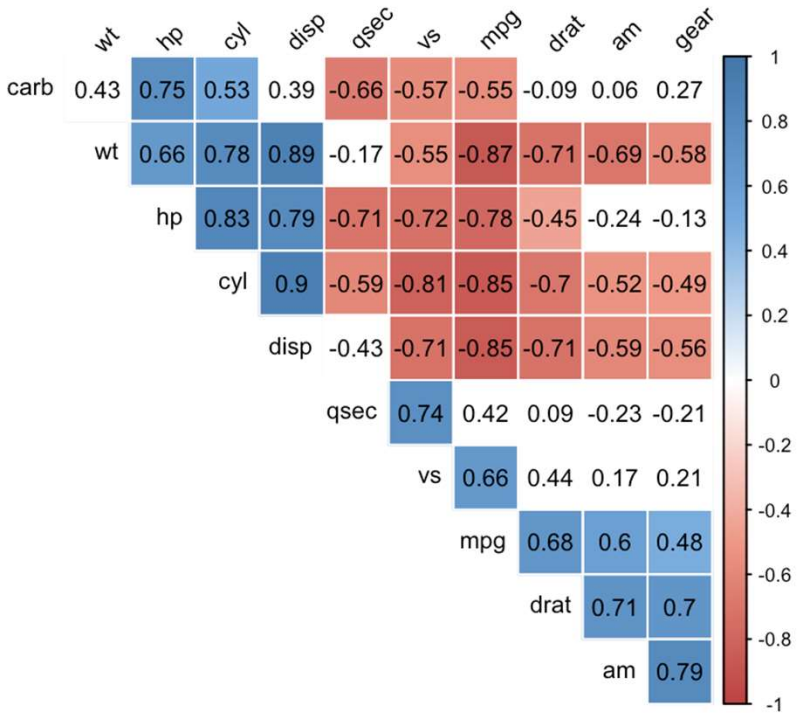
- La dimensión del vector $\boldsymbol{\omega_0} = [\boldsymbol{\omega_0}, \boldsymbol{\omega_1}, \boldsymbol{\omega_i}, \ldots, \boldsymbol{\omega_n}]$ es n+1.

Regresión Lineal Múltiple

- Supuestos estadísticos para la utilización de la regresión lineal múltiple:

  ✓**Utilice variables predictoras linealmente independientes entre ellas.**
  ✓Evitar el problema de la **multicolinealidad.**

- Para validar:
  ✓Correlograma, filtrar variables con correlaciones altas (e.g.: ρ > |0.85|).



Regresión Lineal Múltiple

Si existen varias variables independientes, se puede definir el modelo de regresión múltiple a utilizar, dada una medida de calidad del ajuste:

- **Completo**: se evalúan todas las posibles combinaciones de variables independientes, y se escoge la mejor.
- **Tamaño fijo**: se evalúan todas las posibles combinaciones de *K* variables independientes y se escoge la mejor.
- **Paso a paso (*stepwise*)**
  - ✓Hacia adelante (***forward***): se prueba una a una con las variables independientes que aún no se escogen y se evalúa el modelo conjuntamente con las variables seleccionadas previamente. Se detiene cuando la medida de la calidad del ajuste no mejore.
  - ✓Hacia atrás (***backward***): sigue un proceso contrario al método de búsqueda "hacia adelante", en este caso, se empieza con todas las variables y se va eliminando la variable que, cuando no se considera, optimiza la medida de calidad del ajuste.
- **PCA**: se transforman los datos a un nuevo espacio vectorial de menor dimensionalidad que el de entrada.

Lecturas Complementarias

Linear Regression
https://en.wikipedia.org/wiki/Linear_regression

What is Linear Regression
https://www.ibm.com/topics/linear-regression

Linear Regression for Machine Learning
https://machinelearningmastery.com/linear-regression-for-machine-learning/