Introducción a

Ciencia de Datos

2025

Asignatura Optativa

LICENCIATURA EN INFORMÁTICA

FACET-UNT

CD2025

Adquisición

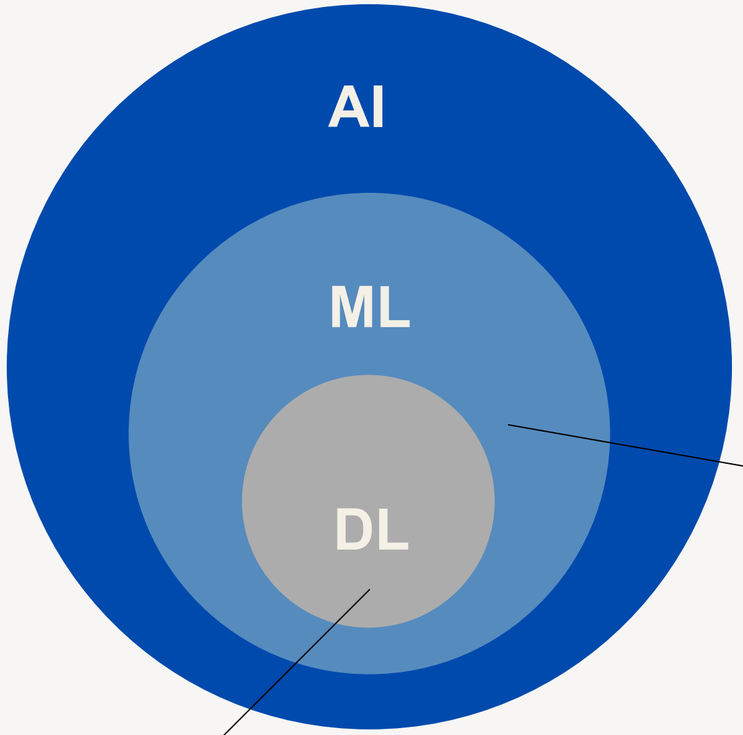
Pre-procesamiento

Almacenamiento

Procesamiento (modelado)

Deployment

- **Preparación**
- Selección
- Entrenamiento
- Evaluación
- Predicción



AI = area of computer science that emphasizes the creation of intelligent machines that work and react like humans.

ML = method of data analysis that automates data model building. ML uses algorithms that learn from data and can find insights without explicit programming.

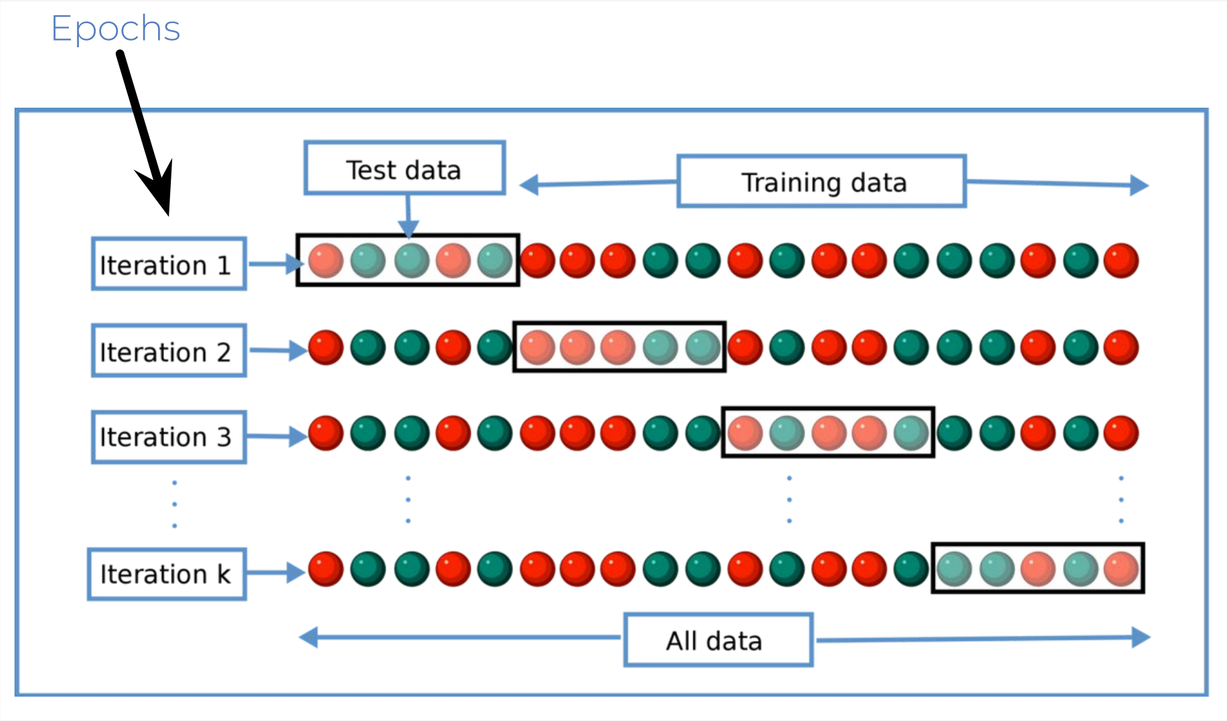
DL = a subset of ML and refers to artificial neural networks that are composed of many layers.

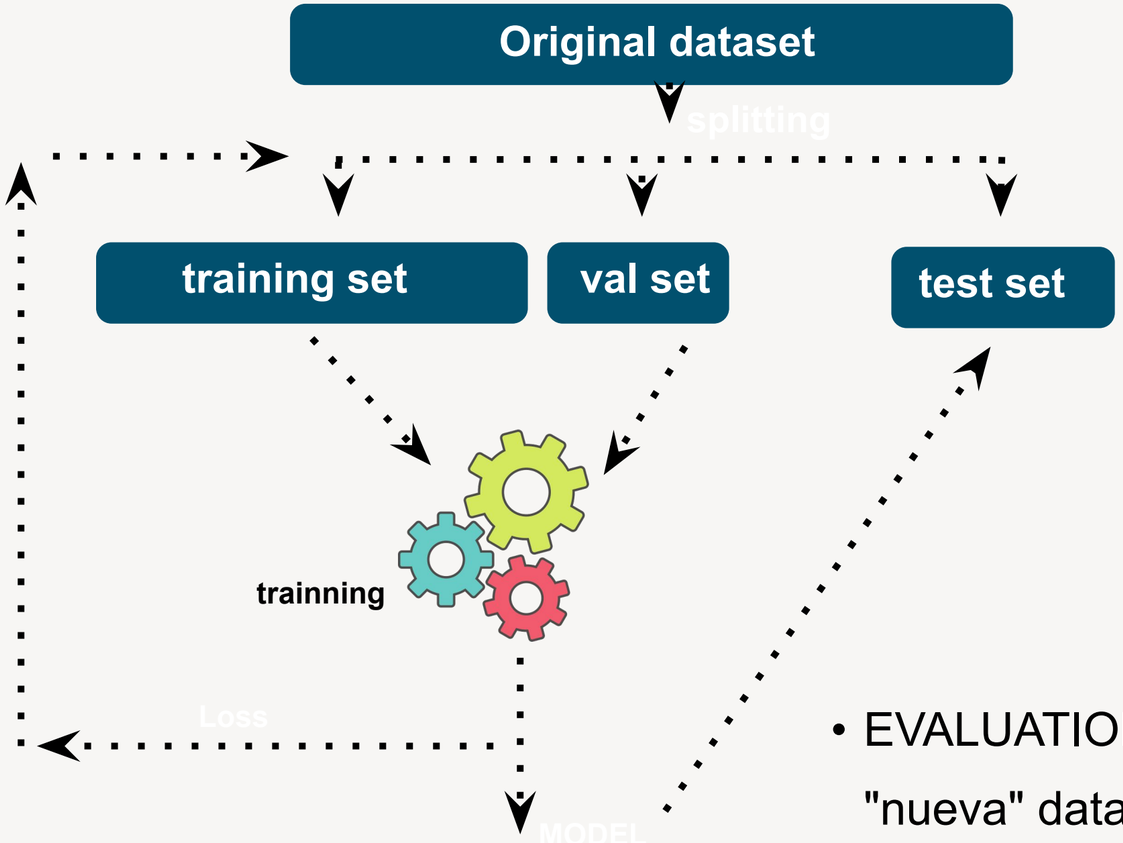

DATA
(input)
Programación
Tradicional
Output
Programa


DATA
(input)
Aprendizaje
Automático
o ML
modelo
(programa)
Output


YOU GET A MACHINE LEARNING
EVERYONE GETS A
MACHINE LEARNING
YOU GET A MACHINE LEARNING

Cómo es el proceso de aprendizaje?





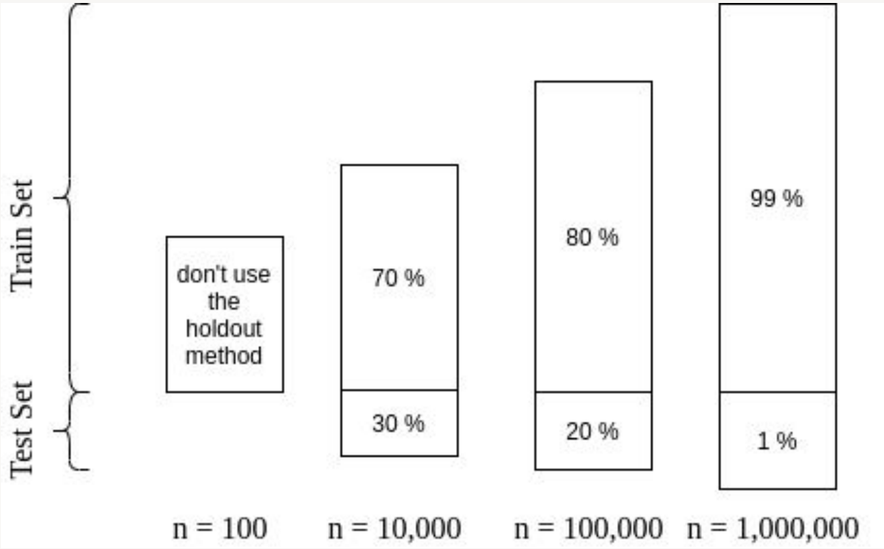
- EVALUATION: con "nueva" data

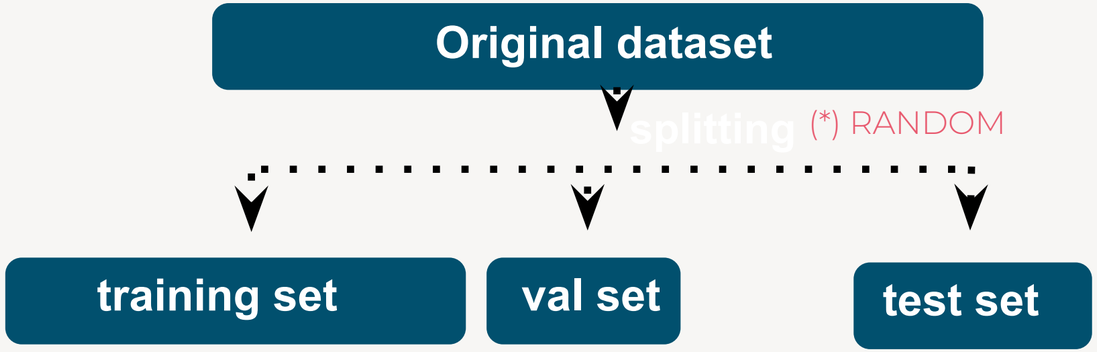
- TRAINING: aprendizaje a partir de los datos

Data splitting
(división del conj. de datos)

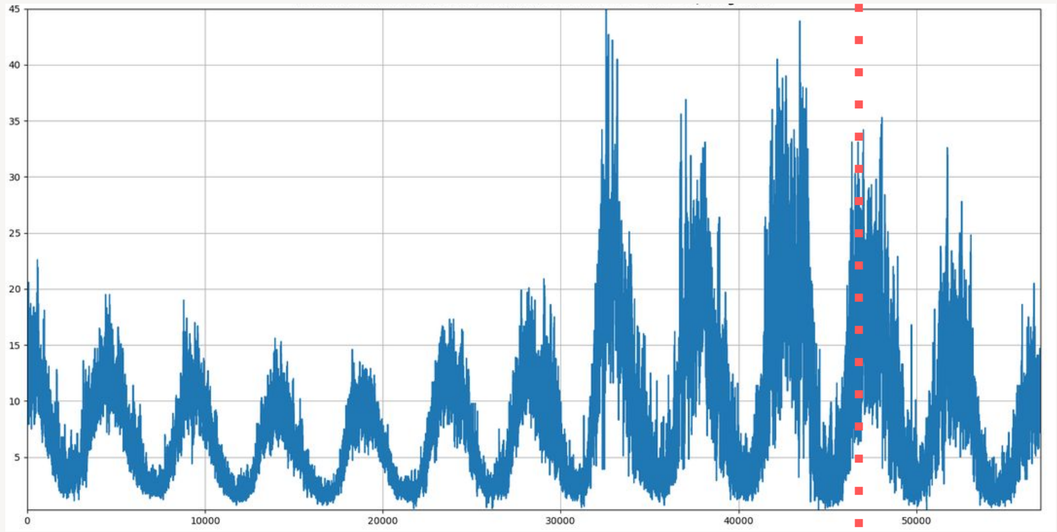
Validation strategies for target prediction methods

(Mathai et al, 2019)





Ejemplo: series temporales

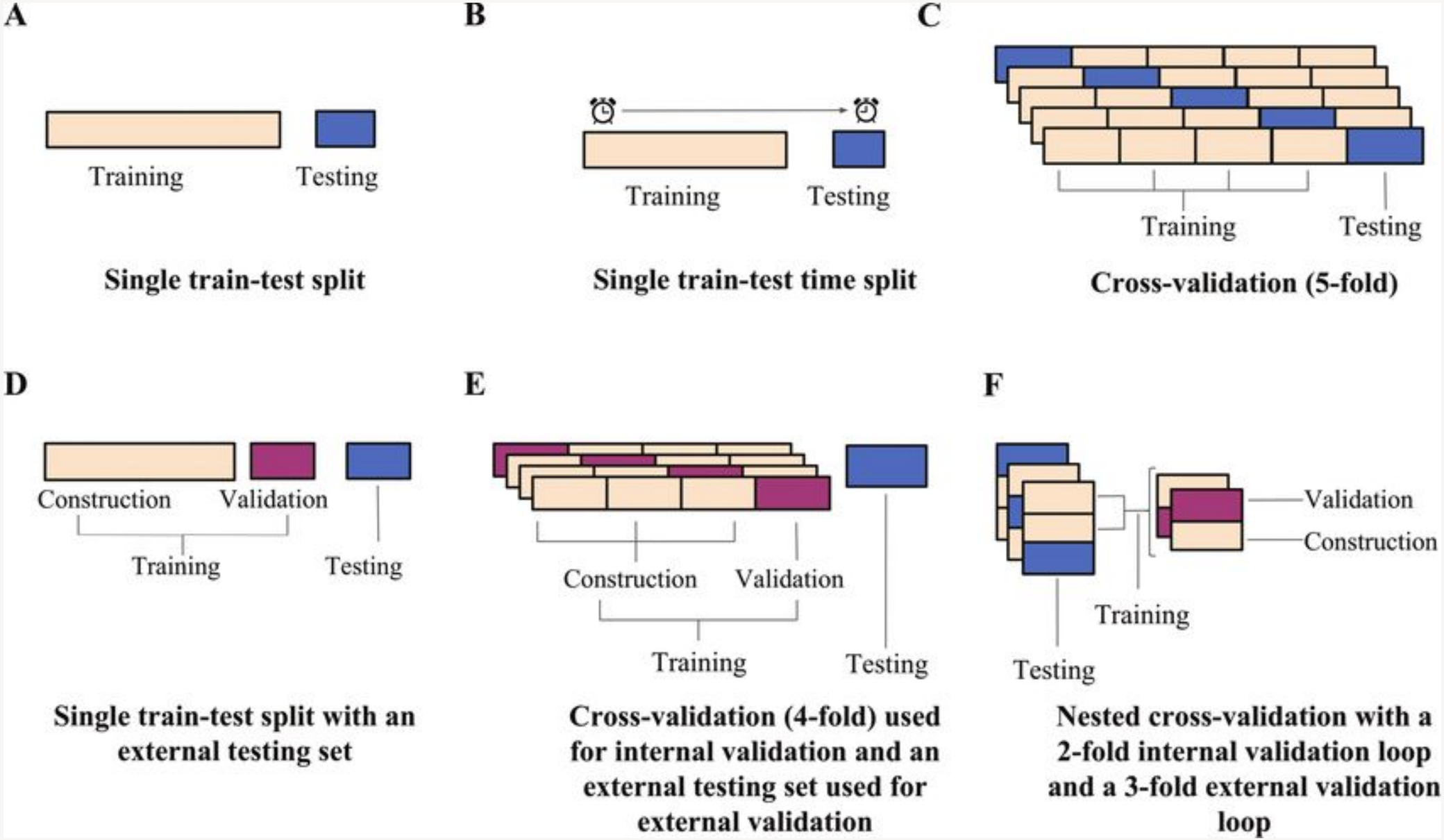


- Conjunto de datos balanceados: tener suficientes casos representativos a partir de los cuales pueda "aprender"
- "Suficientes" datos que describan el problema
- Diferentes técnicas para la división de los datos

Data splitting

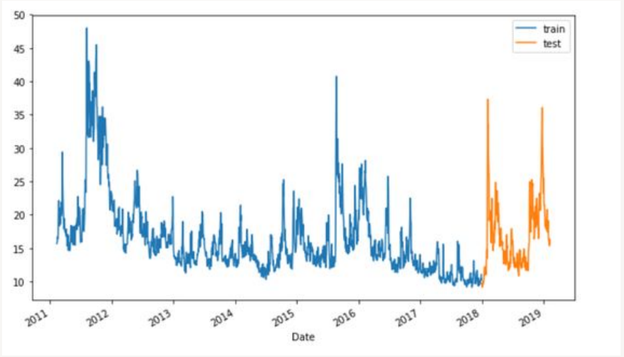
Validation strategies for target prediction methods

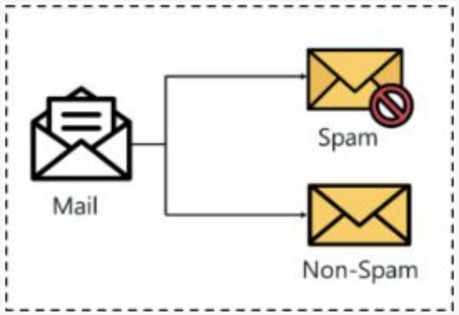
(Mathai et al, 2019)



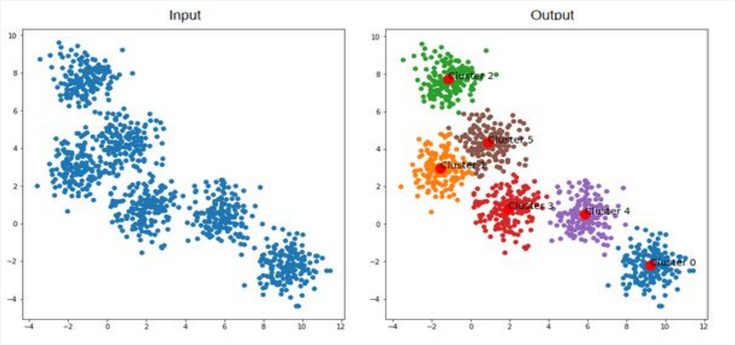
Tipos de aprendizaje

Supervisado: sabemos el valor de la función target. Tenemos un conjunto de datos etiquetados. Por ejemplo: regresión y clasificación

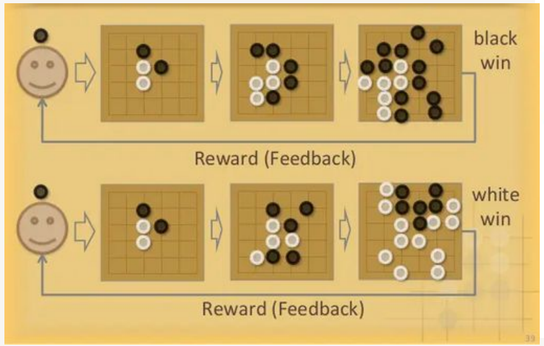




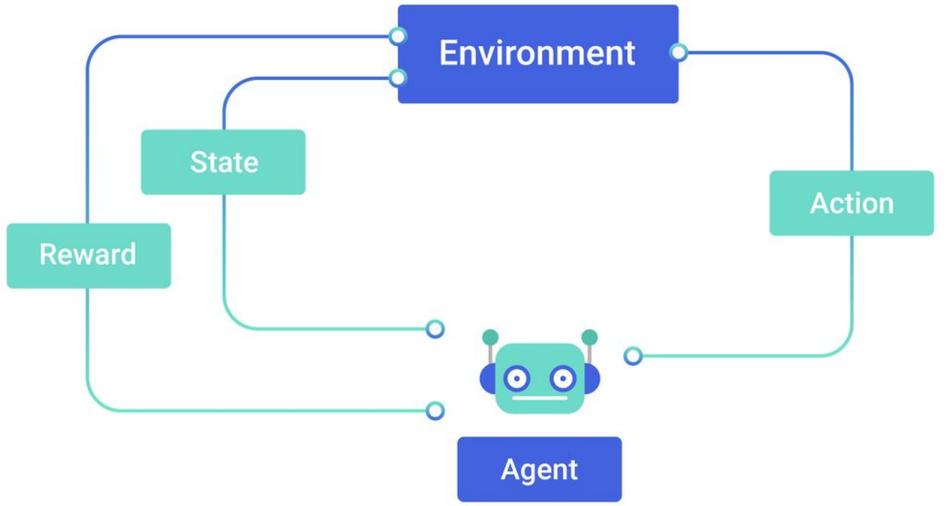
No Supervisado: Conjunto de datos sin etiquetas. Por ejemplo: clustering o agrupamiento.



Semisupervisado: Datos parcialmente etiquetados. Combinan los dos casos anteriores.



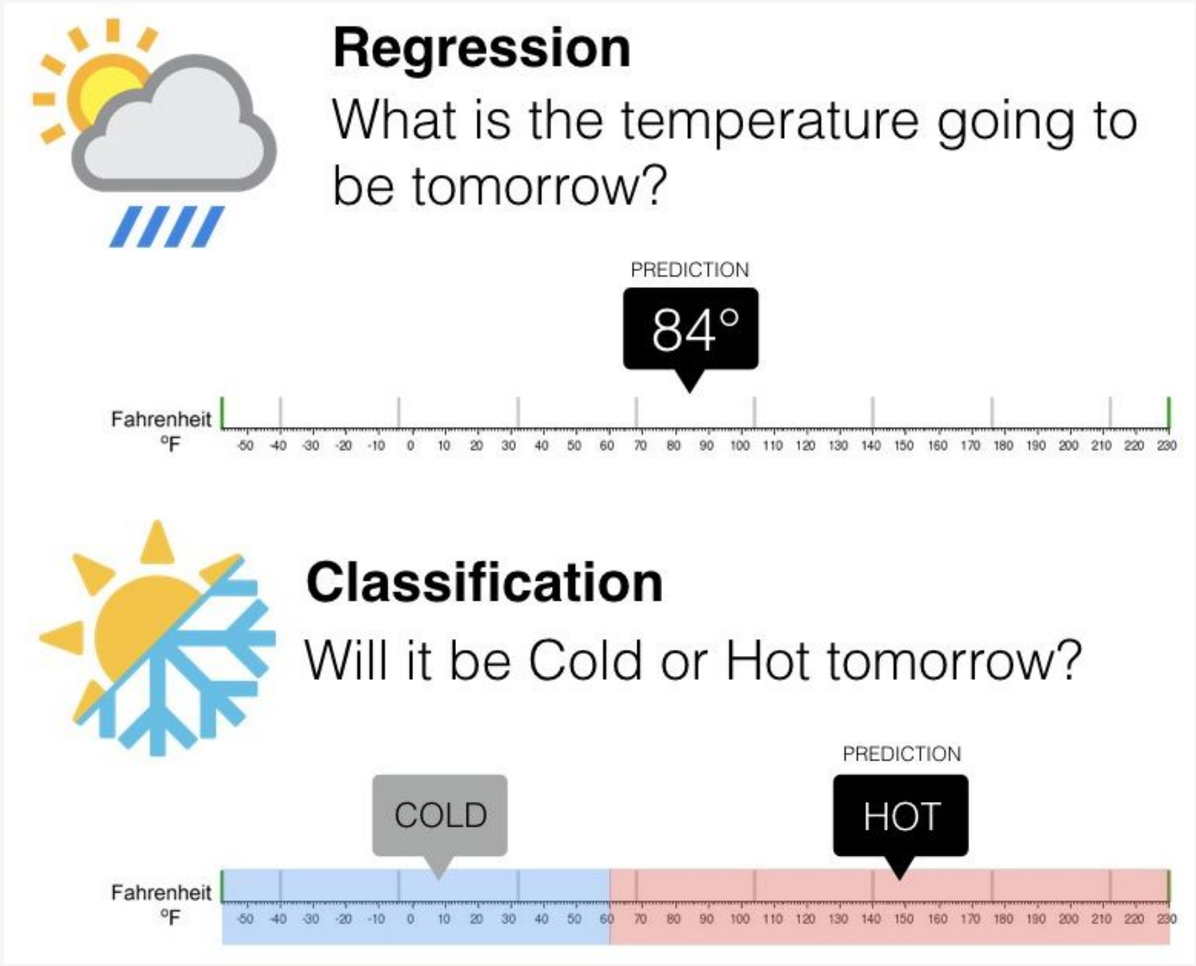
Por refuerzo: El ML aprende de su entorno y se corrige mediante penalizaciones y recompensas.



Tipos de predicciones

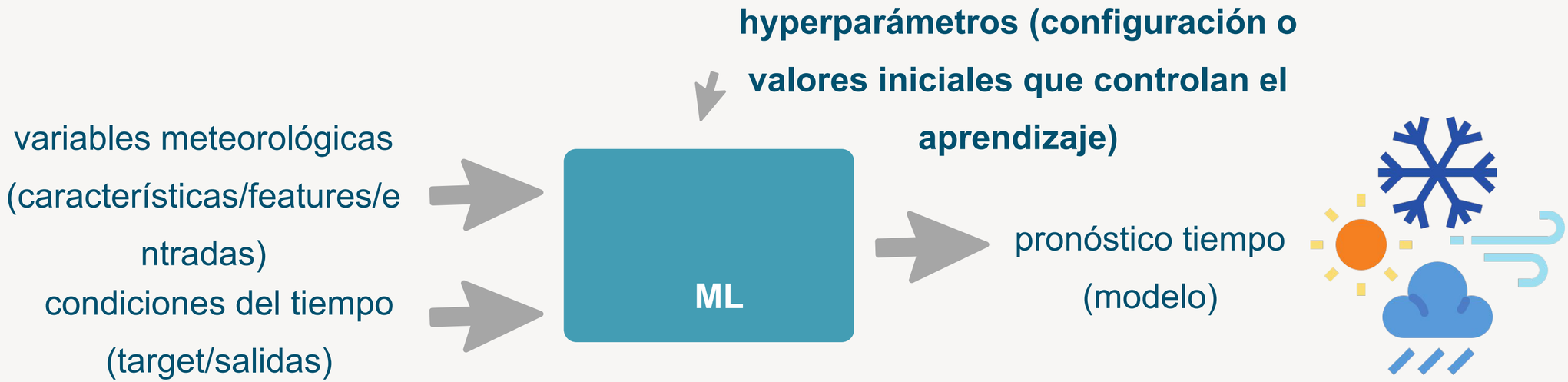
- Regresión: En este tipo de problemas se trata de mapear las entradas a las salidas para el caso que la salida sea un valor real

- Clasificación: El algoritmo aprende una función que mapea las entradas con las salidas donde la salida es un valor discreto (clase).



Ejemplo

https://datos.gob.ar/dataset?organization=smn

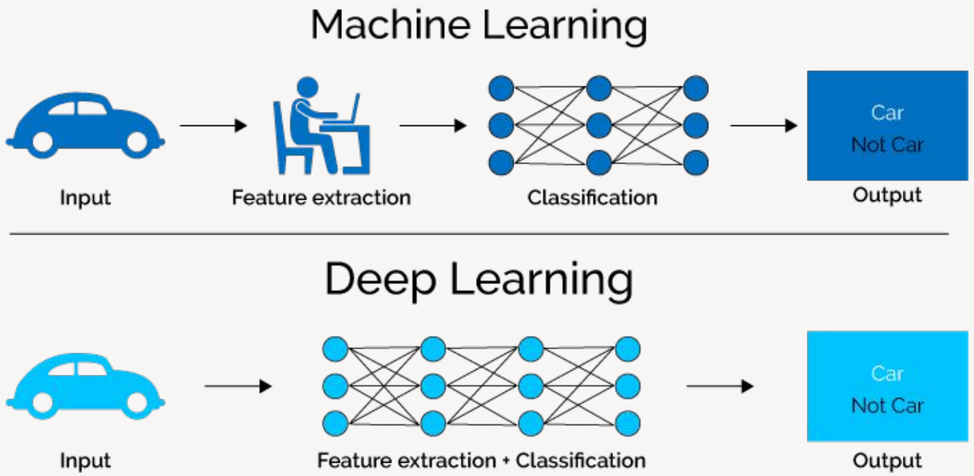


- Cuales son las variables que necesito?
- Podría haber alguna otra que no consideramos o que sea derivada?

- Temperatura (estación, fecha,hora)
- Nubosidad (estación, fecha, hora, observador)
- Humedad
- Radiación solar
- etc ...

Características (features)

Ingeniería y selección de características



- Extraer y elegir las características (features) más relevantes para que el ML aprenda de los datos.
- Incluye la creación de nuevos features que no existían en el set de datos
- Combinación de experiencia y conocimiento del dominio.
- Hacer ingeniería de características a mano en la práctica es difícil, lento, no muy robusto, no escalable. DL permite encontrar las features directamente a partir de los datos

Características (features)

- Más allá de las características explícitas en los datos (col de dato, valores de la ST, etc).

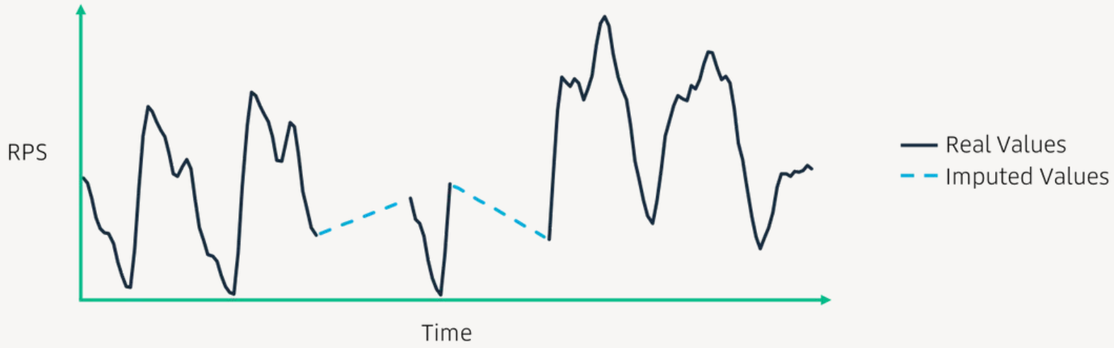
- Basadas en tiempo(lagged feature): cuando valores pasados con un determinado salto de tiempo dan información.

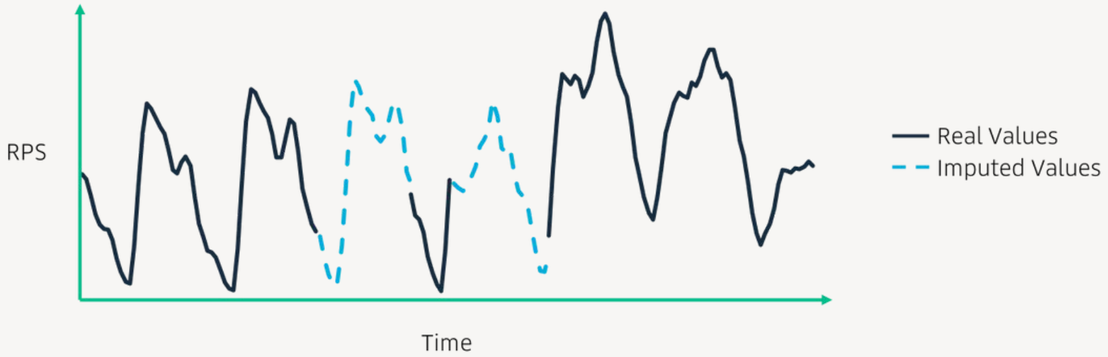
- Estadísticos: MEAN, STD, MIN, MAX, MEDIAN sobre el dataset

- Agregar información mediante transformaciones en los datos: imputación de datos, manejo de outliers, log transform, binning, escalado, etc

Características (features)

- Qué tanto importa que tengamos en cuenta los datos faltantes?
- Porqué hay datos faltantes en el dataset?
- Qué hacer con los datos faltantes?! eliminación vs imputación.





A survey on missing data in machine learning

Tlamelo et al. (2021)

IMPUTACIÓN

→ Imputación de datos numéricos: NaN o directamente no está el dato -> ponerlo a cero? poner un valor promedio o la media? o interpolar?

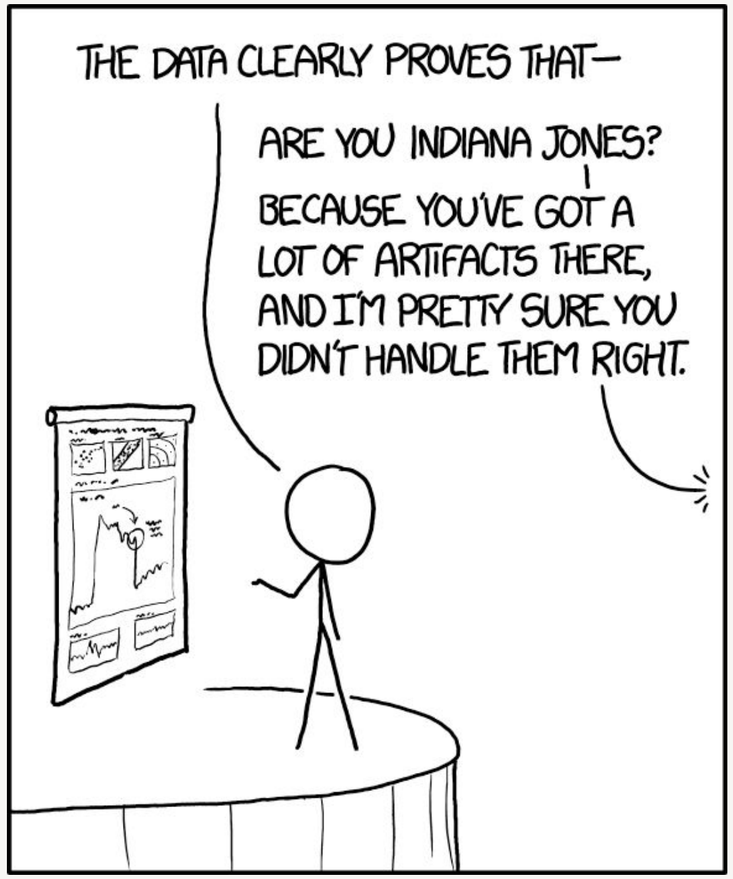
→ Imputación de datos categóricos: reemplazar por el máximo valor o crear una categoría "otros"

Características (features)

- Qué hacer con los OUTLIERS? tirarlos o ponerle un tope.

Determinación de outliers:

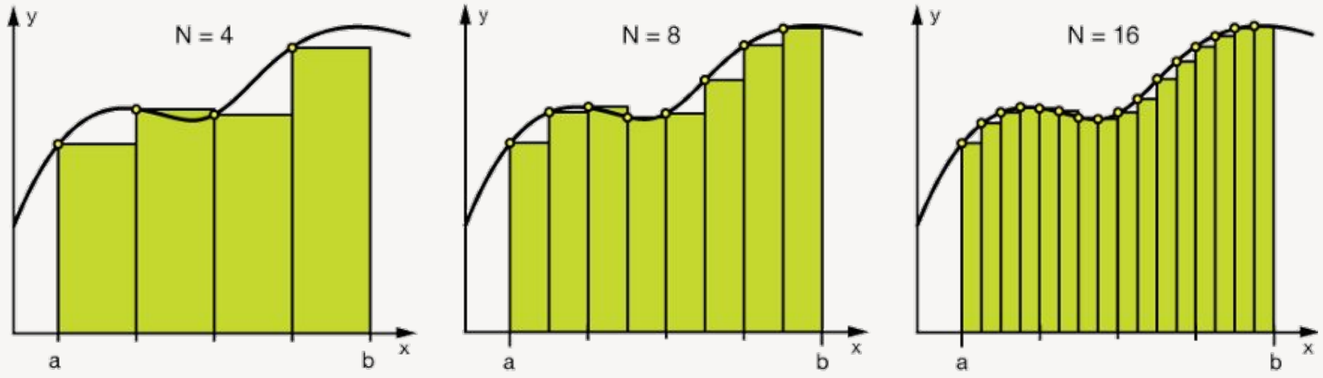
- Mediante visualización
- Mediante la desviación estándar: si un valor es mayor que factor*std se puede decir que es un outlier (factor ~2-4)
- Mediante percentiles



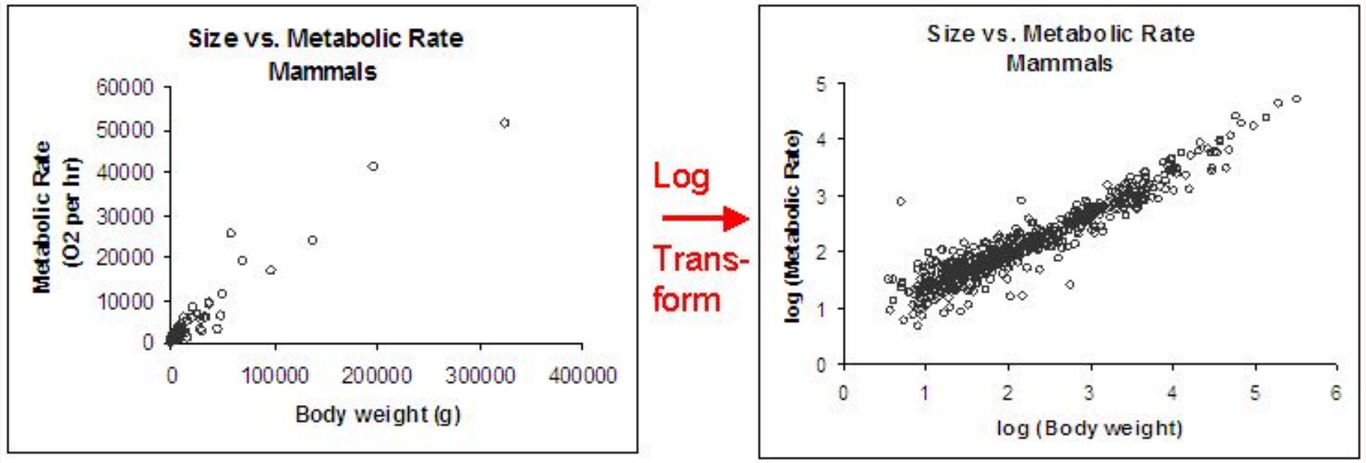
Características (features)

BINNING

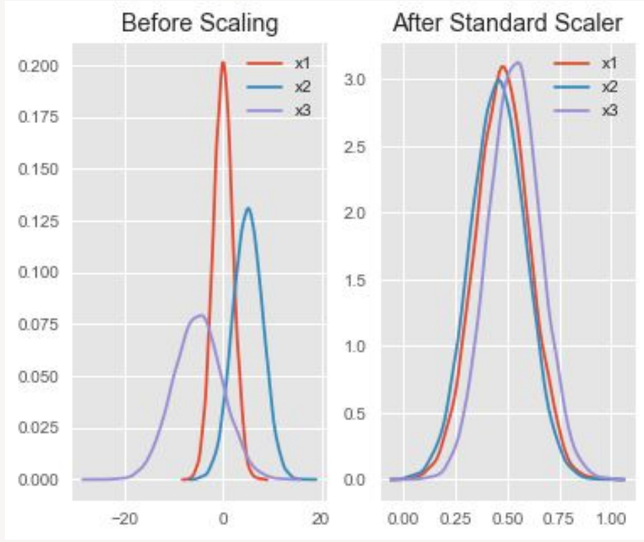
- Transformar datos continues en datos categóricos
- < performance (se sacrifica información)
- Ayuda a evitar el overfitting
- Cuantos bins?

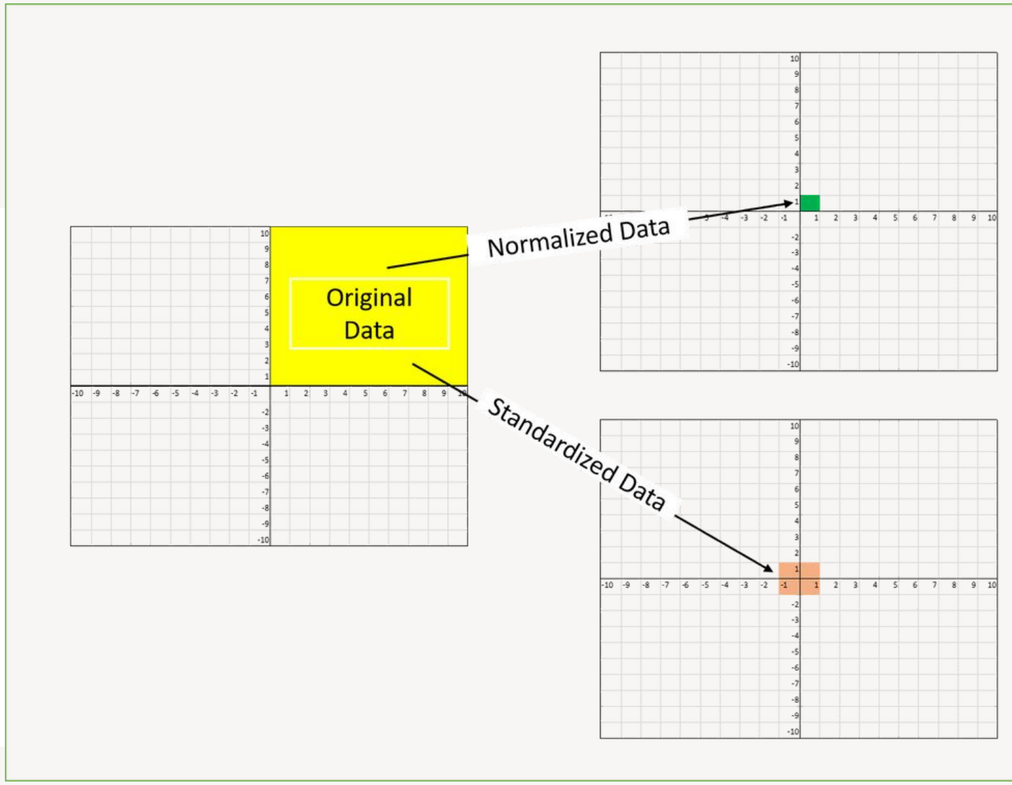


Características (features)



- Transformar datos usando por ejemplo Log (la más usada) pero hay otras.
  - Ayuda a evitar sesgo en los datos, luego de la transformación la distribución de los mismos se aproxima un poco más a la normal
  - Robustez del modelo





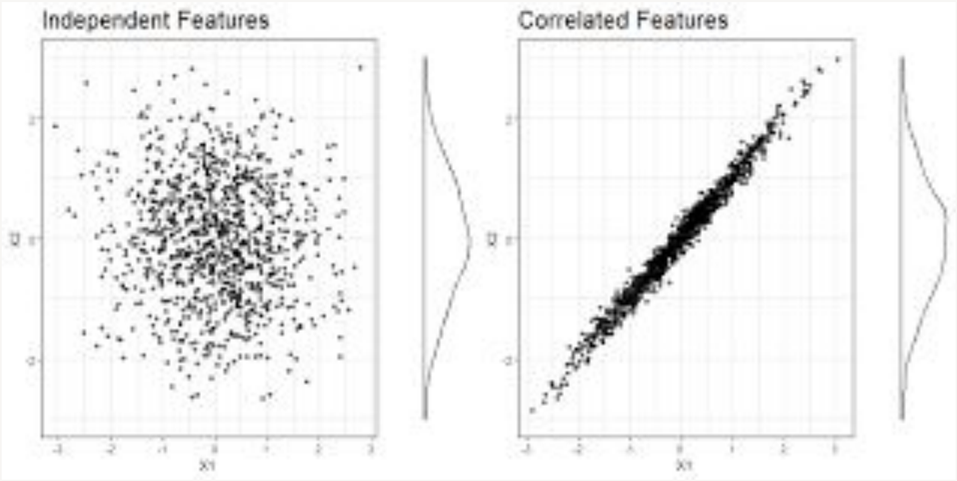
Escalar features: Normalización / Estandarización de datos

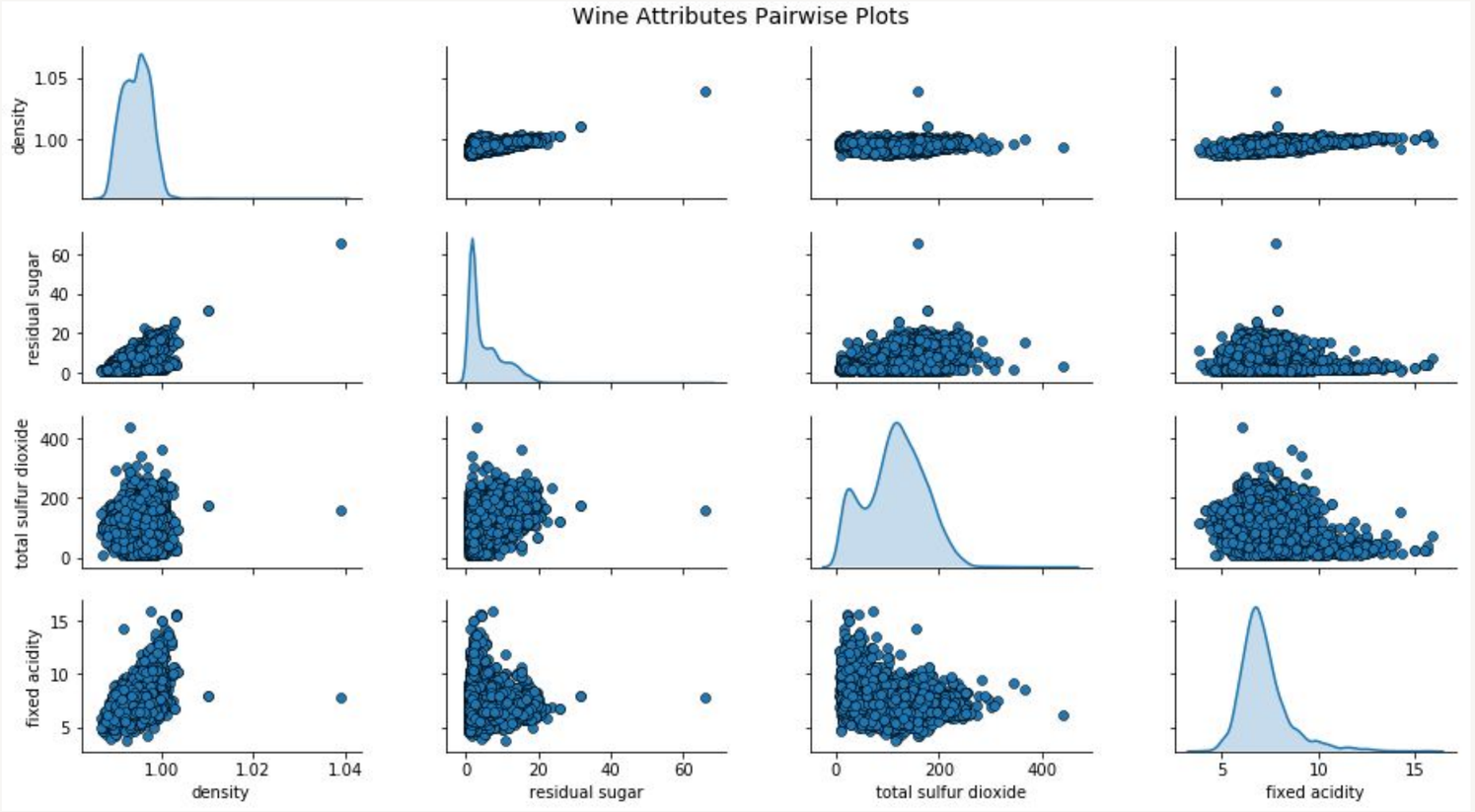
Cuántas features???

- Qué pasa si una variable (feature) en mi dataset está relacionada fuertemente con otra? Cuanta información aporta al modelo usar las dos variables?

- Cuanta más features (dimensiones) tengamos, más datos necesitamos para entrenar satisfactoriamente a nuestro modelo

- Cómo analizamos si dos o más variables están relacionadas y cómo?





Cuántas features???

- Necesitamos entender (estadísticamente) la relación entre las features. Puede ser positiva, neutral, negativa

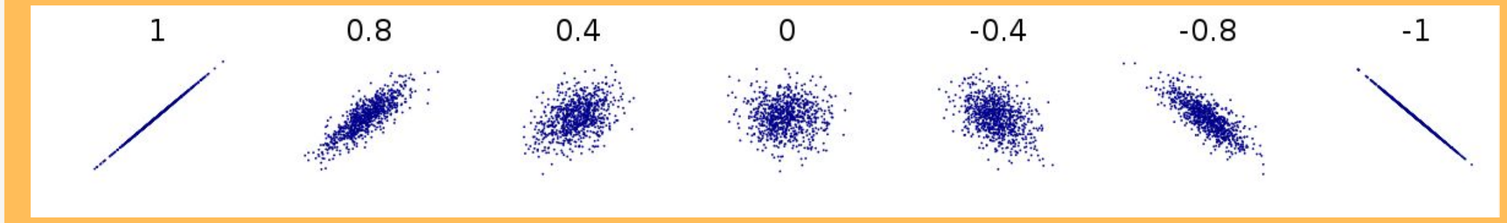
- Cuando hay variables muy relacionadas entre sí, la performance del modelo se deteriora. Sacar una de esas variables puede mejorar mucho la performance del modelo

COVARIANZA: Si las variables estan relacionadas de forma lineal, se puede evaluar mediante la covarianza entre ellas.

```
cov(X, Y) = (sum (x - mean(X)) * (y - mean(Y)) ) * 1/(n-1)
```

COEF CORRELACION PEARSON: indica que tan fuerte es la relación entre las features. Se puede calcular como sigue:

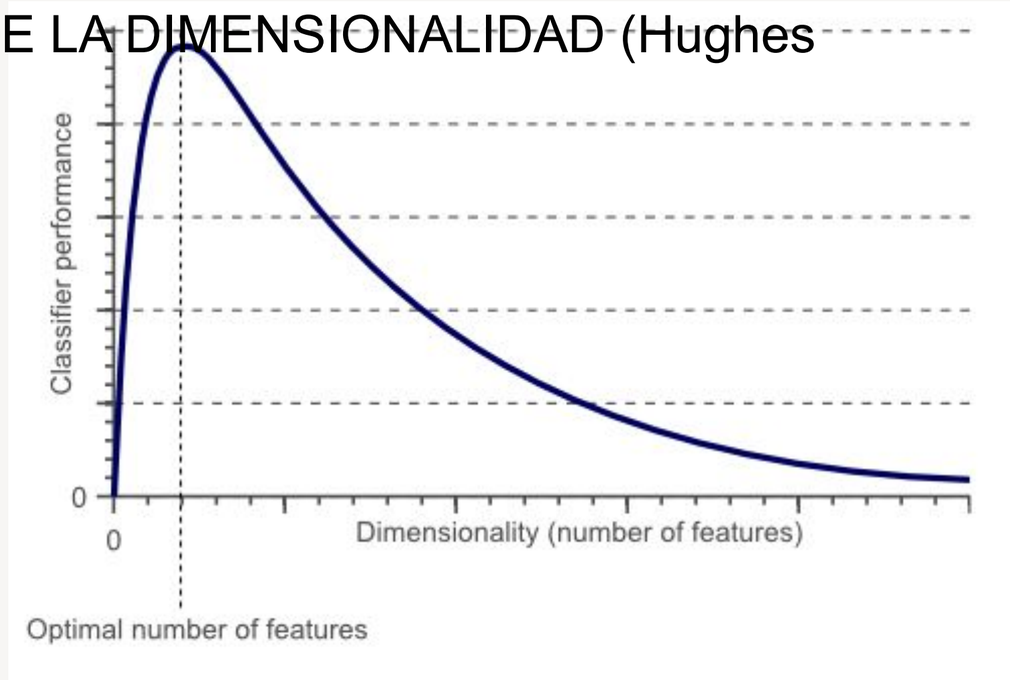
```
covariance(X, Y) / (stdv(X) * stdv(Y))
```

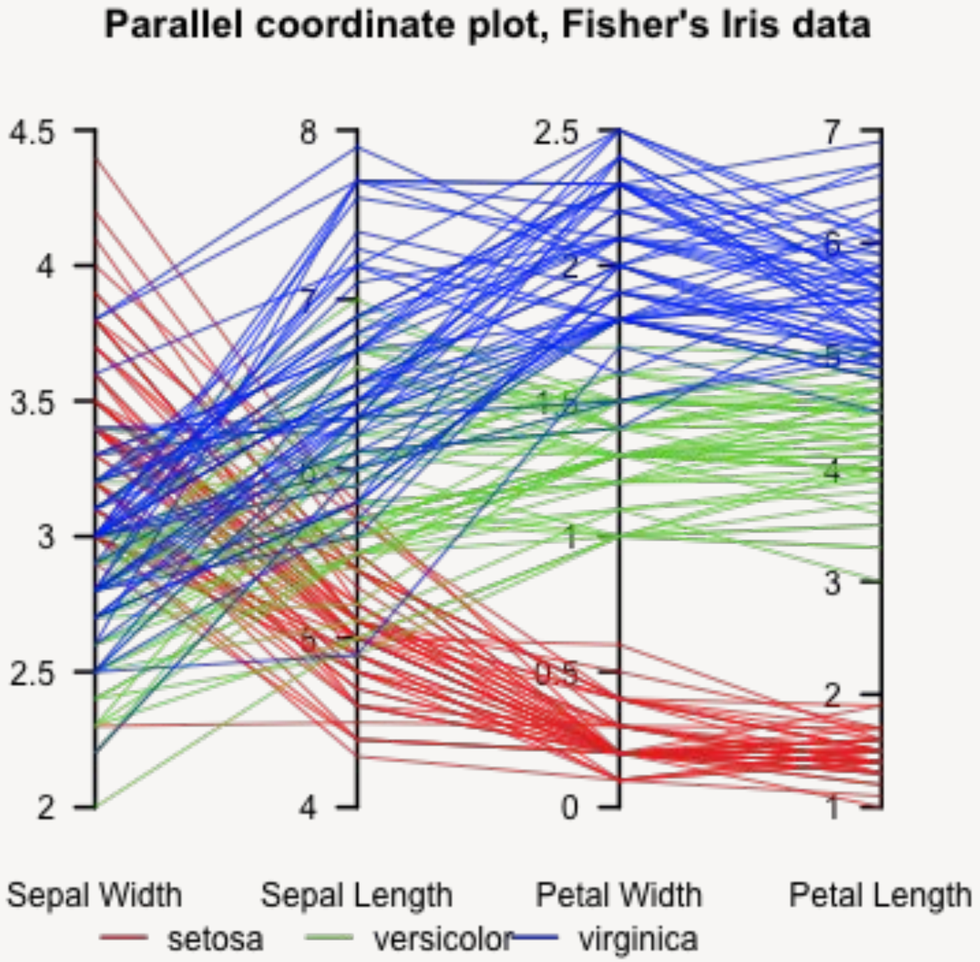


Cuántas features???

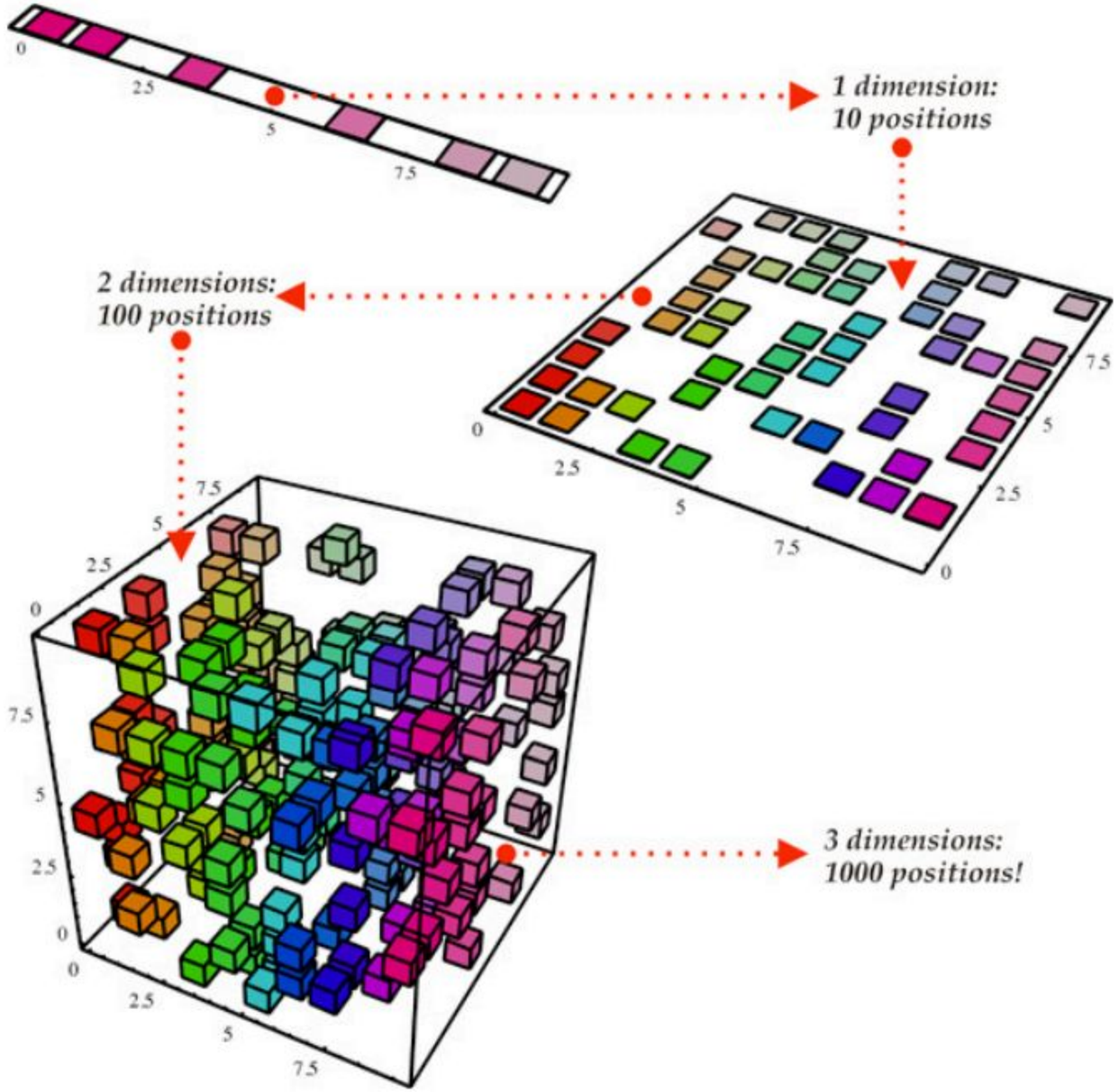
Various dimension reduction techniques for high dimensional data analysis: a review. Papia R et al. (2021)

- Nos referimos a datos de alta dimensionalidad cuando el número de características (p) de un dataset es mayor que el número de observaciones (N)
- Tratar con muchas dimensiones es difícil!
- MALDICIÓN DE LA DIMENSIONALIDAD (Hughes Phenomenon)





Cuántas features???



La **maldición de la dimensionalidad**

+ features + data + computo

A medida que crecen las dimensiones, la cantidad de datos que necesito para generalizar mejor crece exponencialmente!

+ datos dispersos

Distance concentration: muchos algoritmos de ML se basan en cálculo de distancia y, en el caso de altas dimensiones, la distancia entre dos puntos tiende a crecer (se hacen más disimiles)

https://www.nature.com/articles/s41592-018-0019-x

Reducir la dimensionalidad!

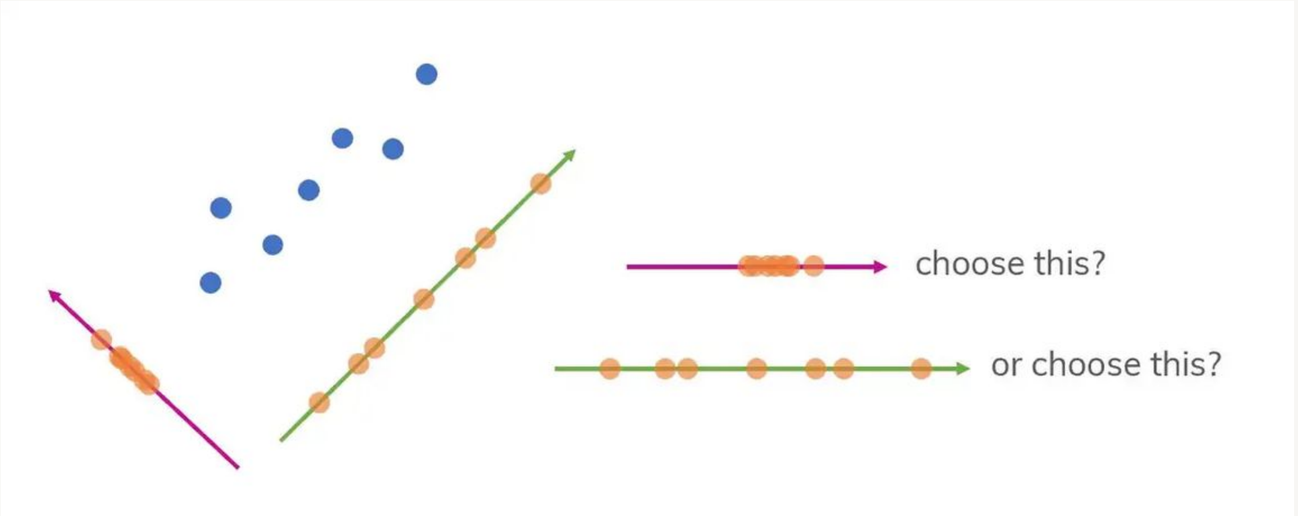
Reducción de dimensionalidad (PCA)

(existen otros métodos)

Principal Component Analysis (PCA):

Permite determinar cuales son las features que explican mejor el problema

Encuentra un nuevo set de dimensiones tal que todas las dimensiones son ortogonales (linealmente independientes, no correlacionadas) y las rankea de acuerdo a la varianza de los datos a lo largo de estas.

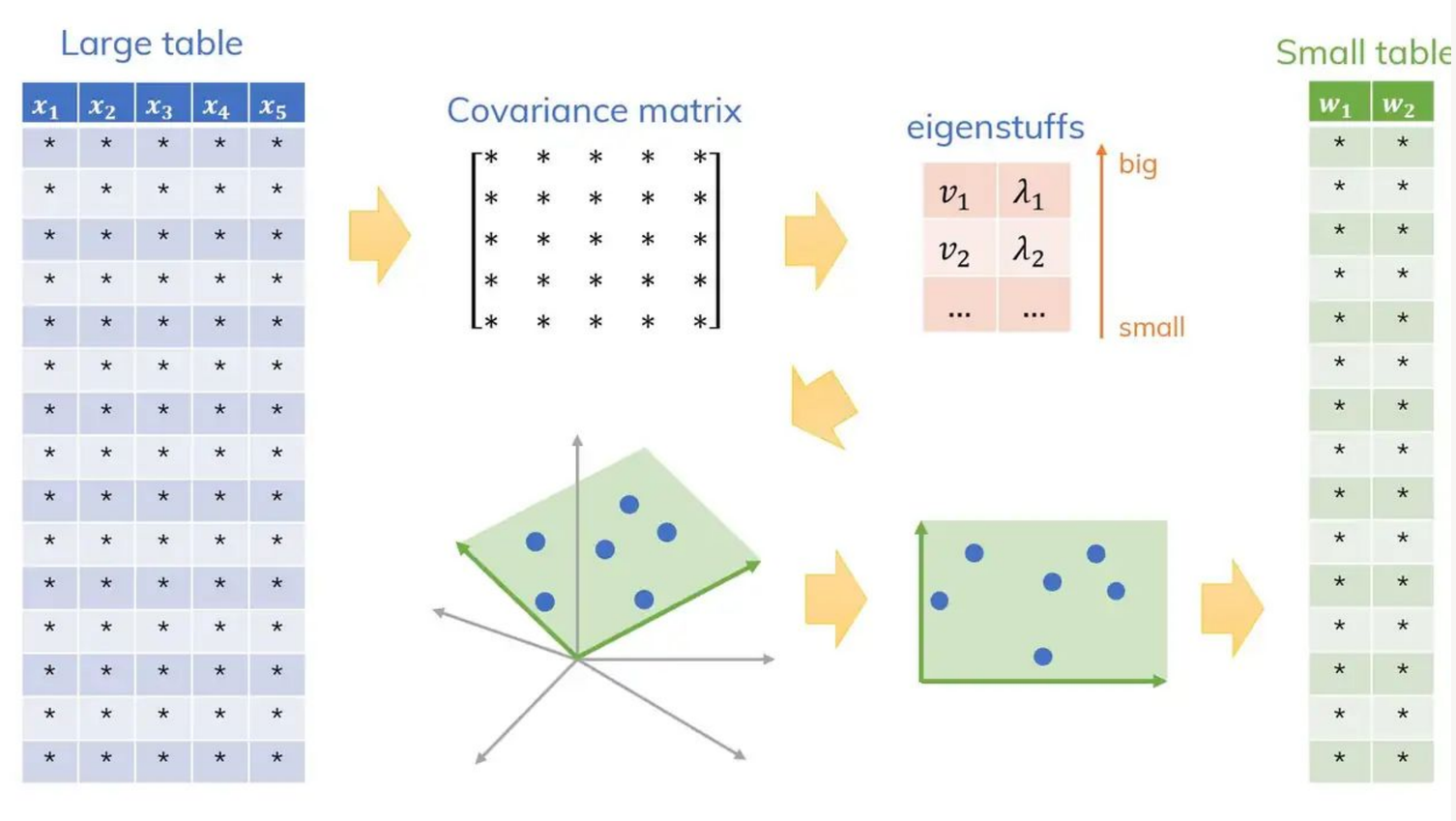


Significa que la nueva dimensión más importante aparece primero (+ importante

PCA

Algoritmo:

- Calcula la matriz X de cov de los datos
- Calcula los vectores y valores propios
- Ordena los vectores propios de acuerdo a sus valores propios en orden desc
- Elige los primeros k vectores propios, que serán las nuevas k dimensiones
- Transforma los datos originales de n-dimensiones en k-dimensiones



Má detalles en el TP

https://setosa.io/ev/principal-component-analysis/

Cuales???

Selección de características

- Hay varias técnicas para seleccionar features adecuadas

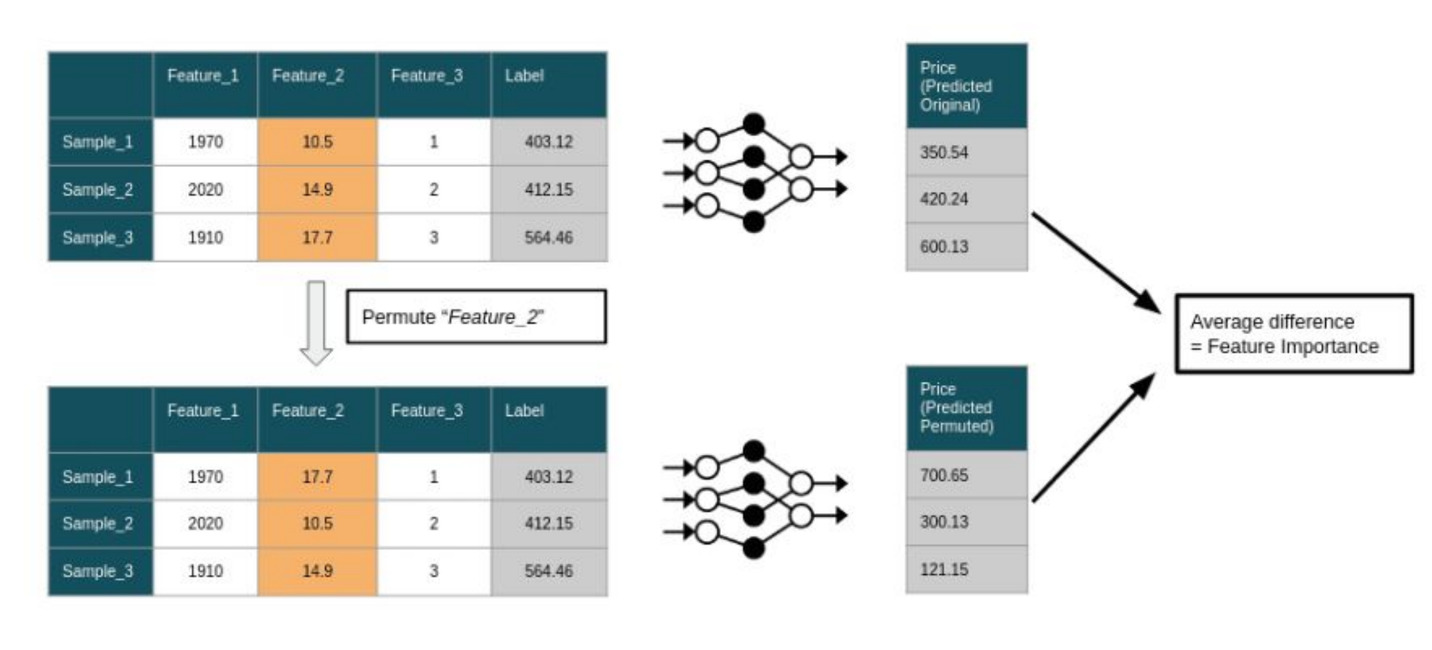
- Permutation feature importance (relaciones no-lineales, datos tabulares): refiere al decrecimiento de la performance de un modelo cuando una de las features es mezclada aleatoriamente. De esta manera, el procesamiento quiebra las relaciones que pueda haber entre la feature y el target. Si esto ocurre (baja el score del modelo), es un indicativo de cuánto depende ese modelo de esa feature (cuán importante es para el modelo).

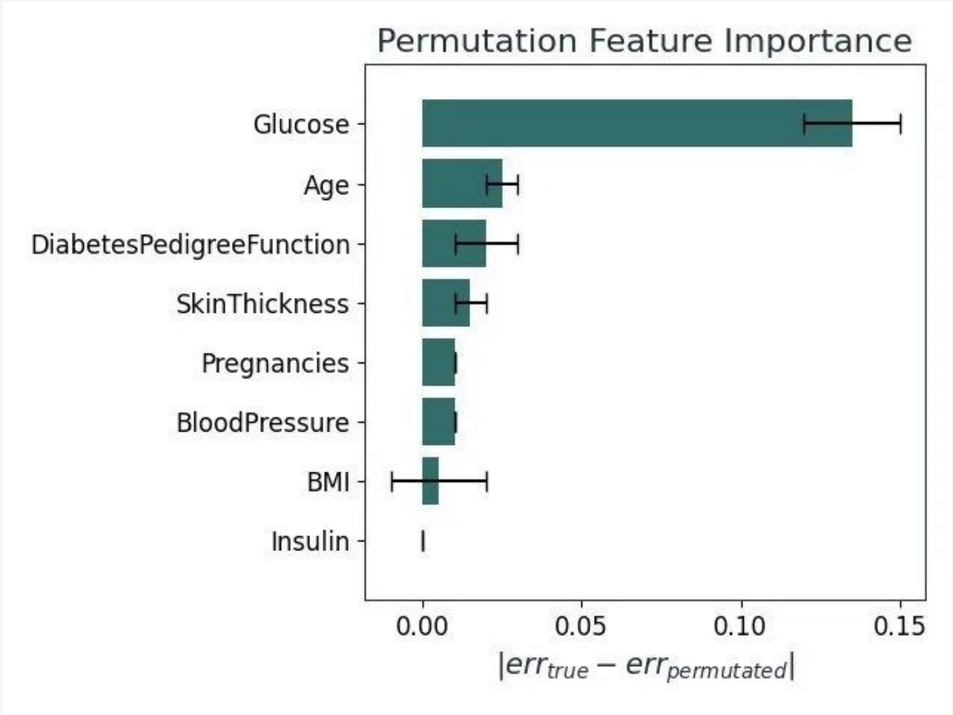
  Se calcula post-hoc (model agnostic) y colabora al concepto de interpretabilidad del resultado. Una vez entrenado el modelo, se corre con diferentes permutaciones de las features

Ayuda a seleccionar features

https://scikit-learn.org/stable/modules/permutation_importance.html

Permutation feature importance





Black box vs XAI

data
+
hyperparameters

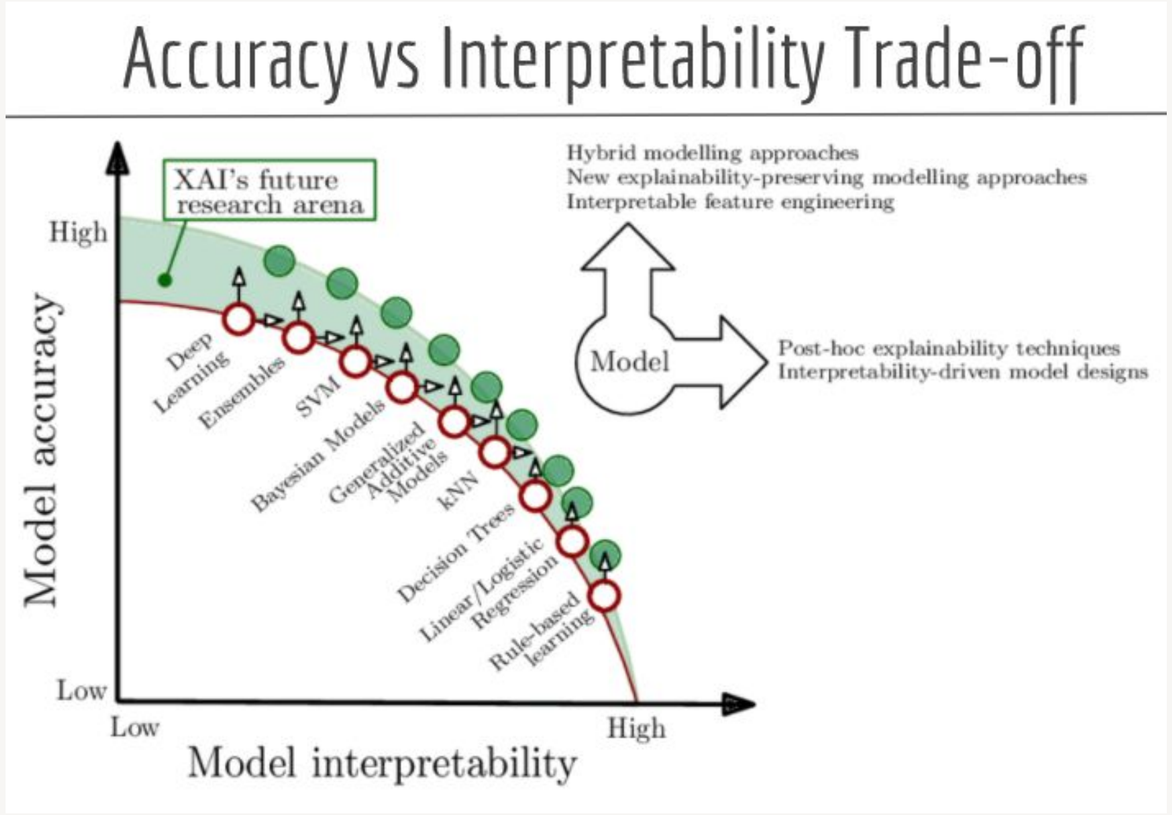
model
(parameters)

data
+
hyperparameters
+ physics/constraints/domain knowledge

model
(parameters)

- Physical consistency (definitions, conservation laws...)
- Ability to generalize outside of the training set
- Interpretability
- Stability
- Data limitations

- Interpretabilidad: transparencia.
- Explacabilidad: procedimiento para clarificar o dar detalles del funcionamiento interno de un modelo
- eXplainable AI (XAI): técnica ML -> confiabilidad



Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities a challenges toward responsible AI

Barredo Arrieta et.al. (2019)