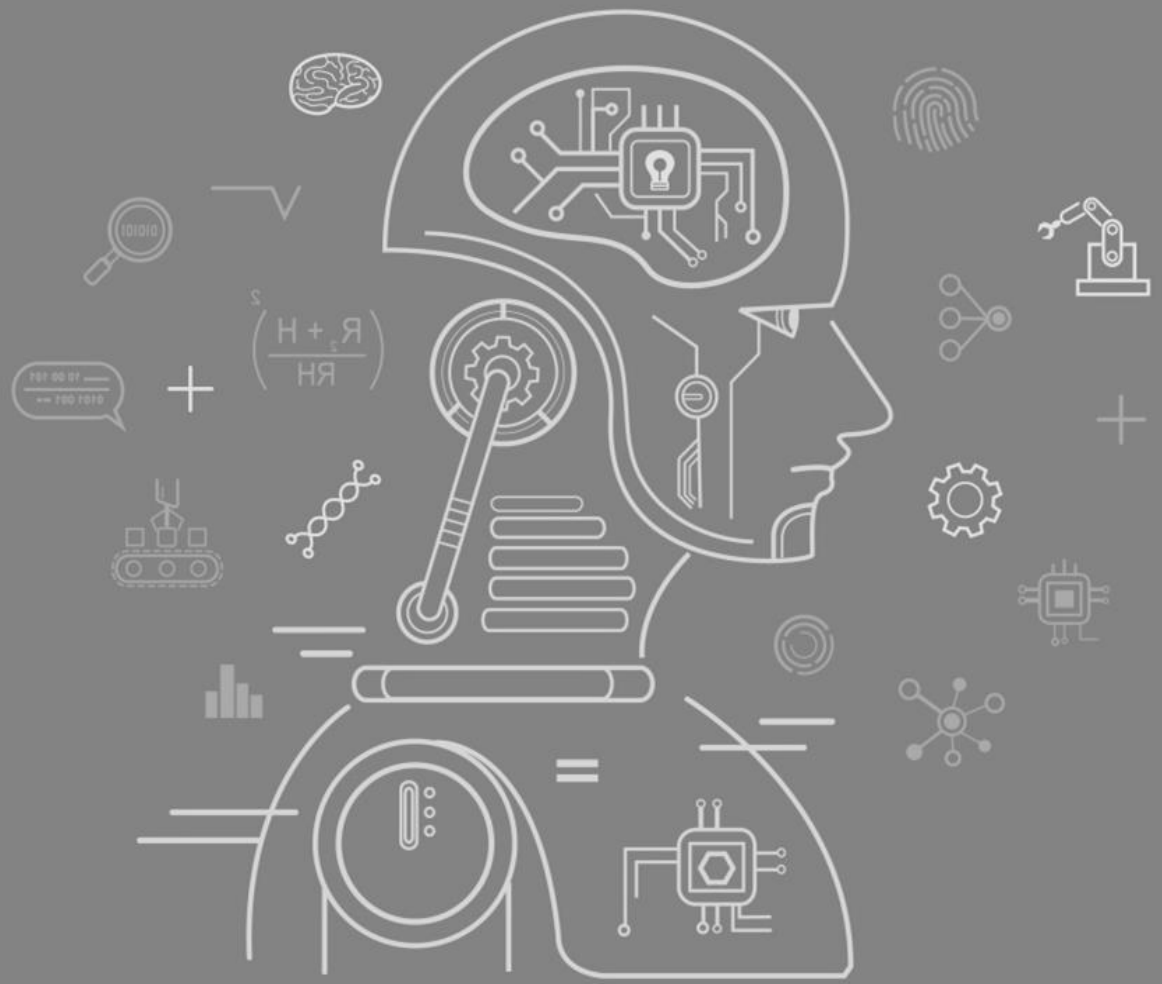
PYTHON
WEEKEND
16 de octubre

# BASES DE SCIKIT LEARN PARA ENTRENAR MODELOS DE MACHINE LEARNING & STATISTICAL LEARNING

## CONFERENCISTA:
## NESTOR MONTAÑO

Organiza

SOCIEDAD ECUATORIANA ESTADÍSTICA



# Bases de scikit learn para modelos de Machine & Statistical learning

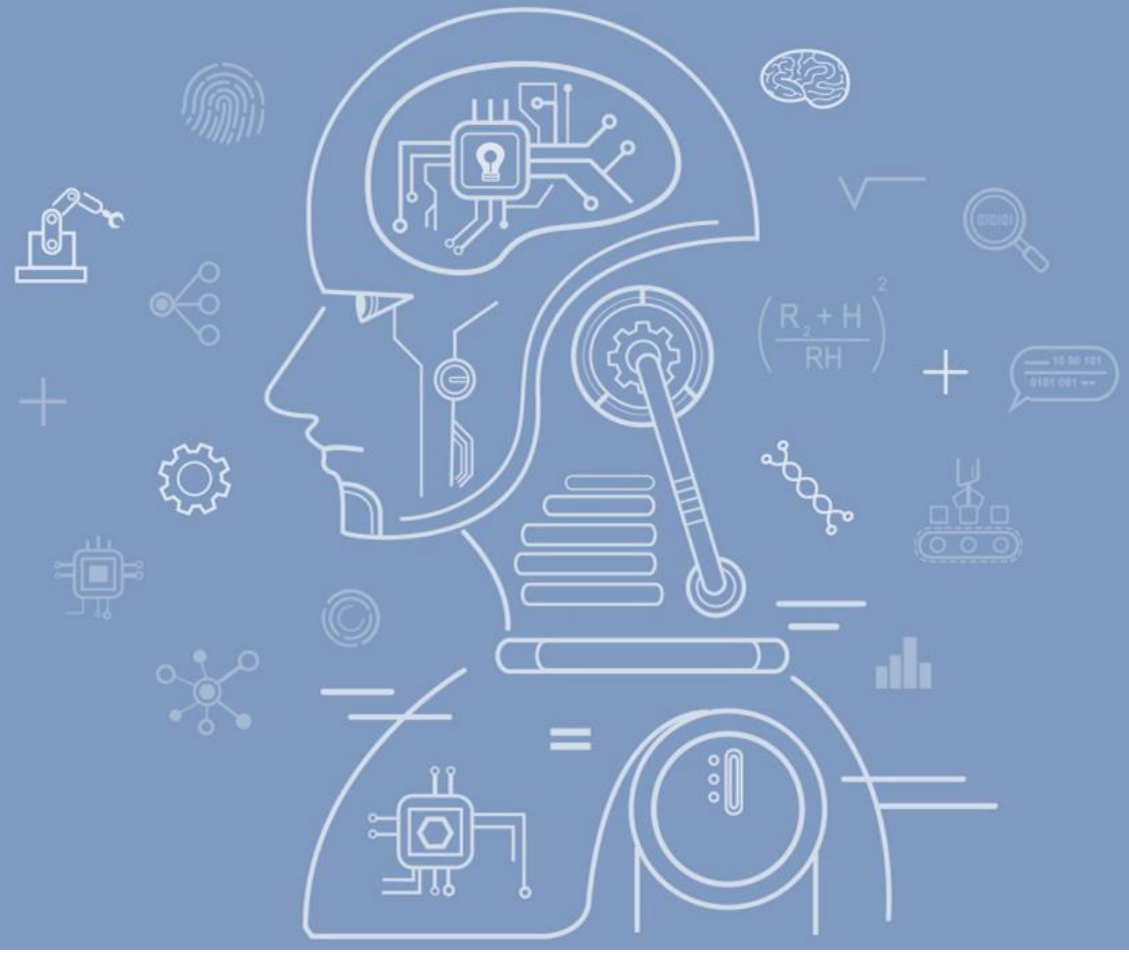
Por: **Néstor Montaño P.**
in/nestor-montaño/

SOCIEDAD ECUATORIANA ESTADÍSTICA

# Conceptos claves

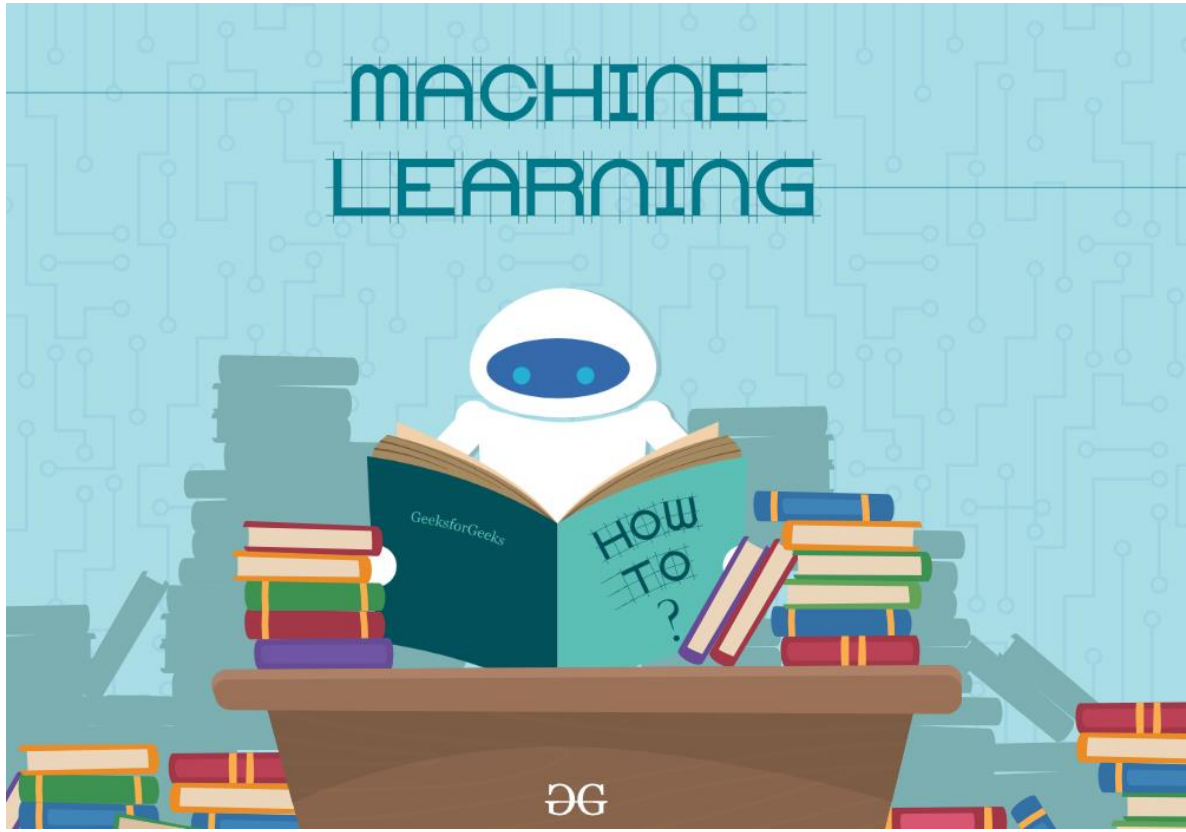
# Inteligencia Artificial

La **Inteligencia artificial** intenta automatizar tareas intelectuales normalmente realizadas por humanos.



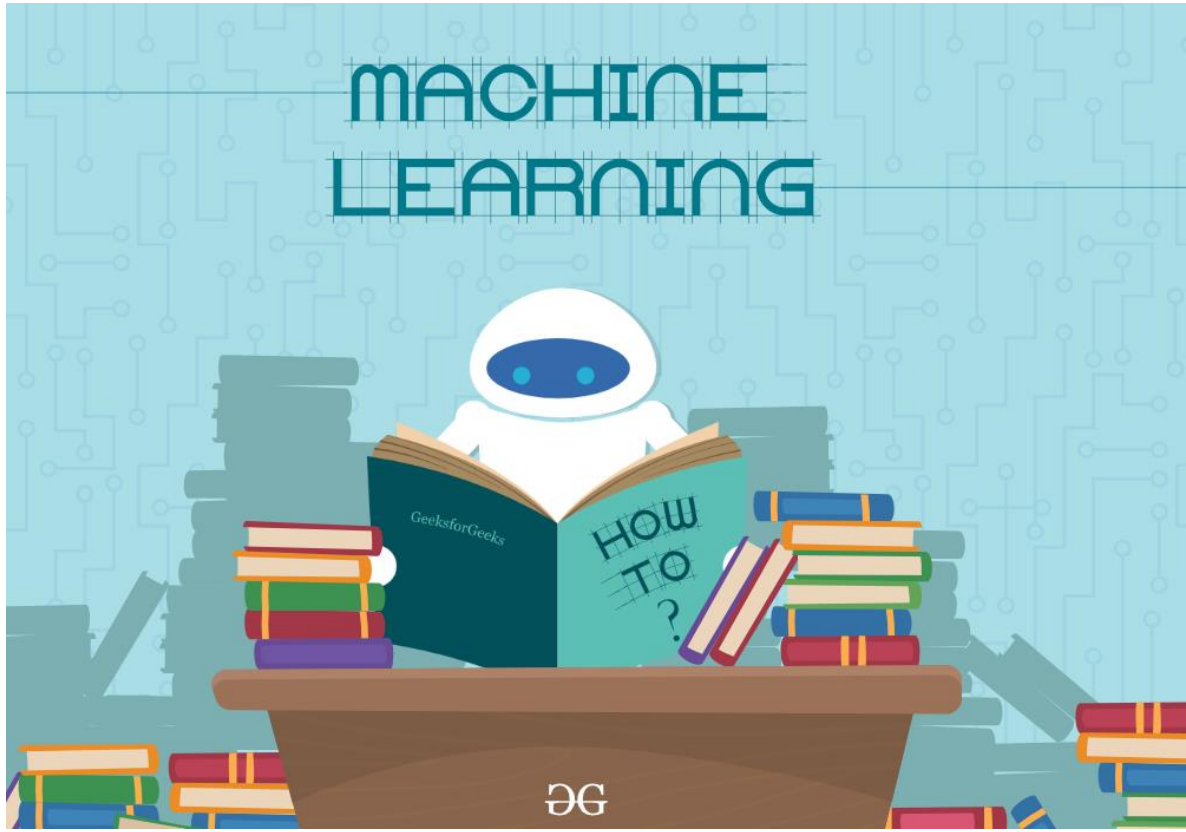
# Machine Learning

El objetivo del machine learning (aprendizaje automático) es extraer información o patrones (aspirando que sean útiles) de un conjunto de datos.



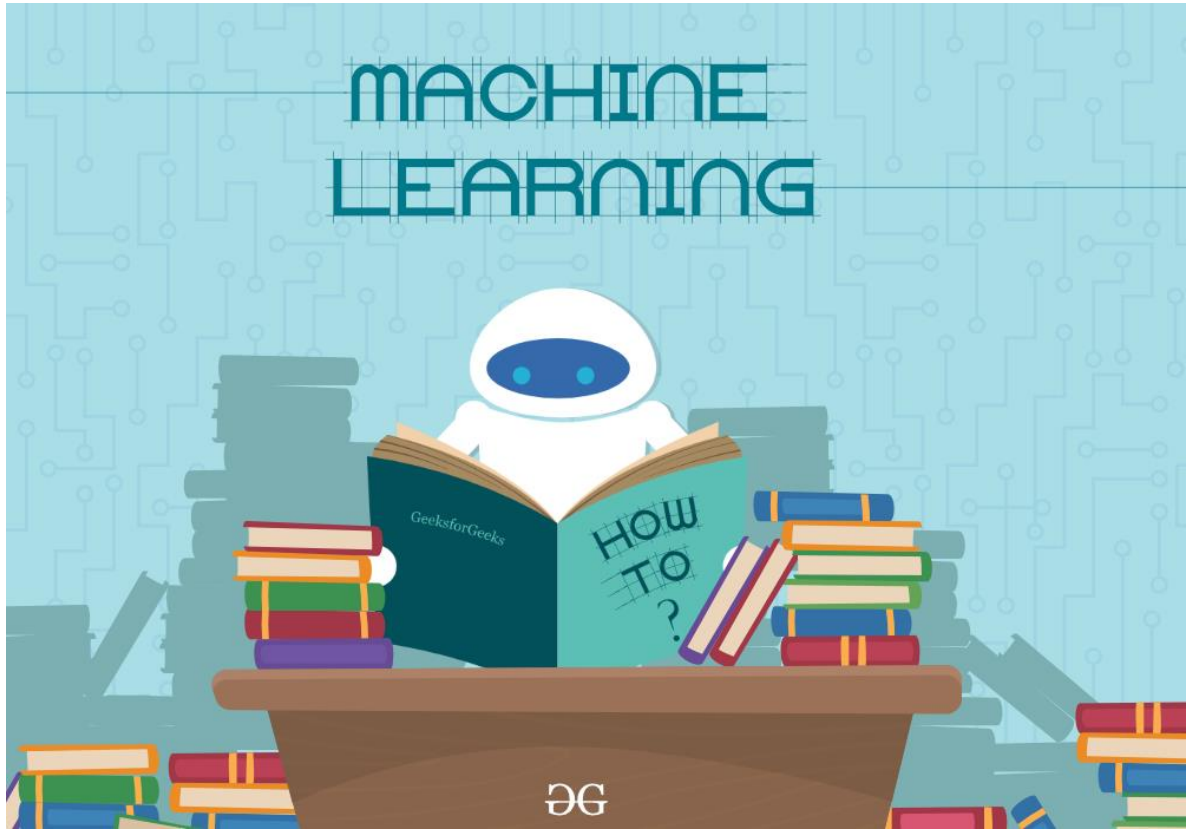
# Machine Learning

El objetivo del machine learning (aprendizaje automático) es extraer información o patrones (aspirando que sean útiles) de un conjunto de datos.



# Machine Learning

El objetivo del machine learning (aprendizaje automático) es extraer información o patrones (aspirando que sean útiles) de un conjunto de datos.

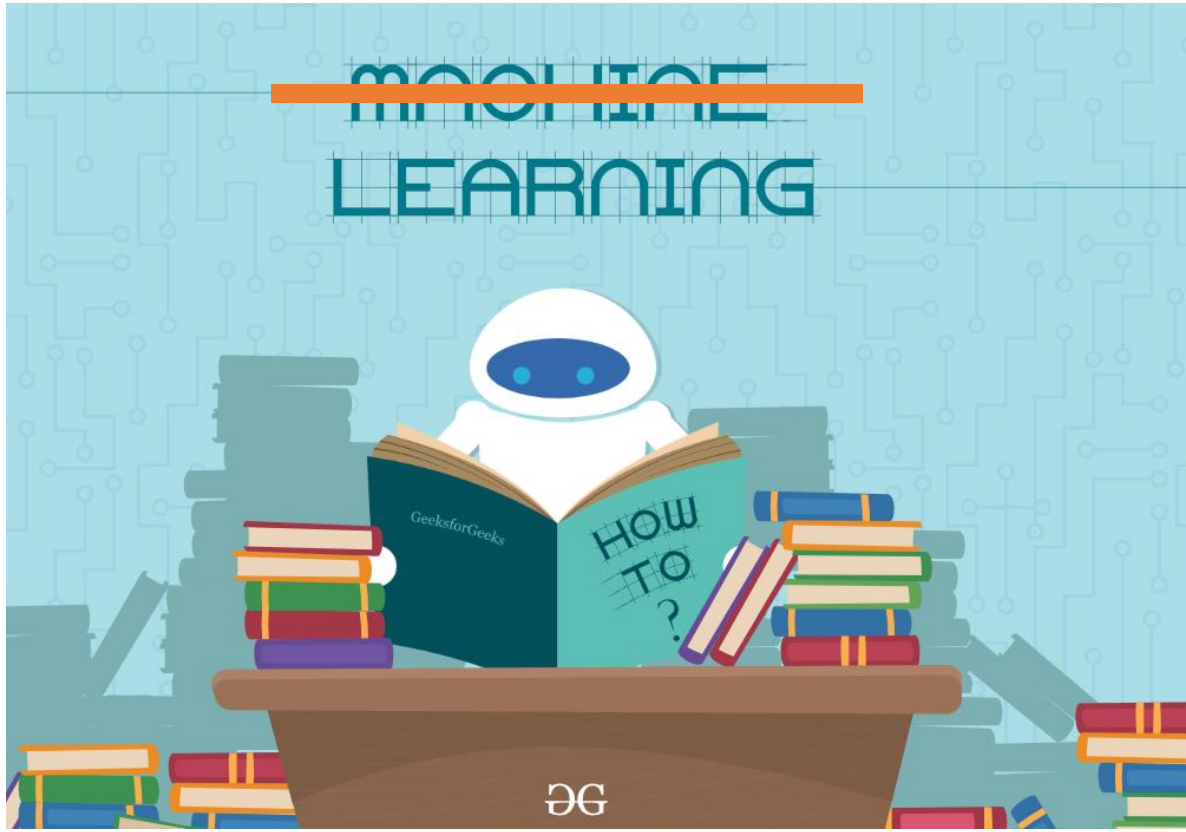



*Esto lo viene haciendo la estadística desde siempre, por ello se tiene el* ***statistical learning***

# El Aprendizaje

Al proceso de pasar de [datos] a [información] se le conoce como **aprendizaje**

De los datos se *aprende* patrones, reglas, etc.



# Tipos de Aprendizaje

## Supervisado

Input: X
Variables independientes o descriptoras

Output: Y
Variable dependiente

Regresión (predecir valor),

Clasificación (Predecir clase)

Optimiza en base al error de predicción

# Tipos de Aprendizaje

## Supervisado

Input: X
Variables independientes o descriptoras

Output: Y
Variable dependiente

Regresión (predecir valor),

Clasificación (Predecir clase)

Optimiza en base al error de predicción

## No Supervisado

Input: X
Variables descriptoras

Cluster,

Reducción de dimensiones

Reglas de asociación (*)

# Tipos de Aprendizaje

## Supervisado

Input: X
Variables independientes o descriptoras

Output: Y
Variable dependiente

Regresión (predecir valor),

Clasificación (Predecir clase)

Optimiza en base al error de predicción

## No Supervisado

Input: X
Variables descriptoras

Cluster,

Reducción de dimensiones

Reglas de asociación (*)

## Por Refuerzo

Input: Estado y acciones posibles

Output: Decisión / acción

Auto conducción,

Navegación,

Realizar de tareas

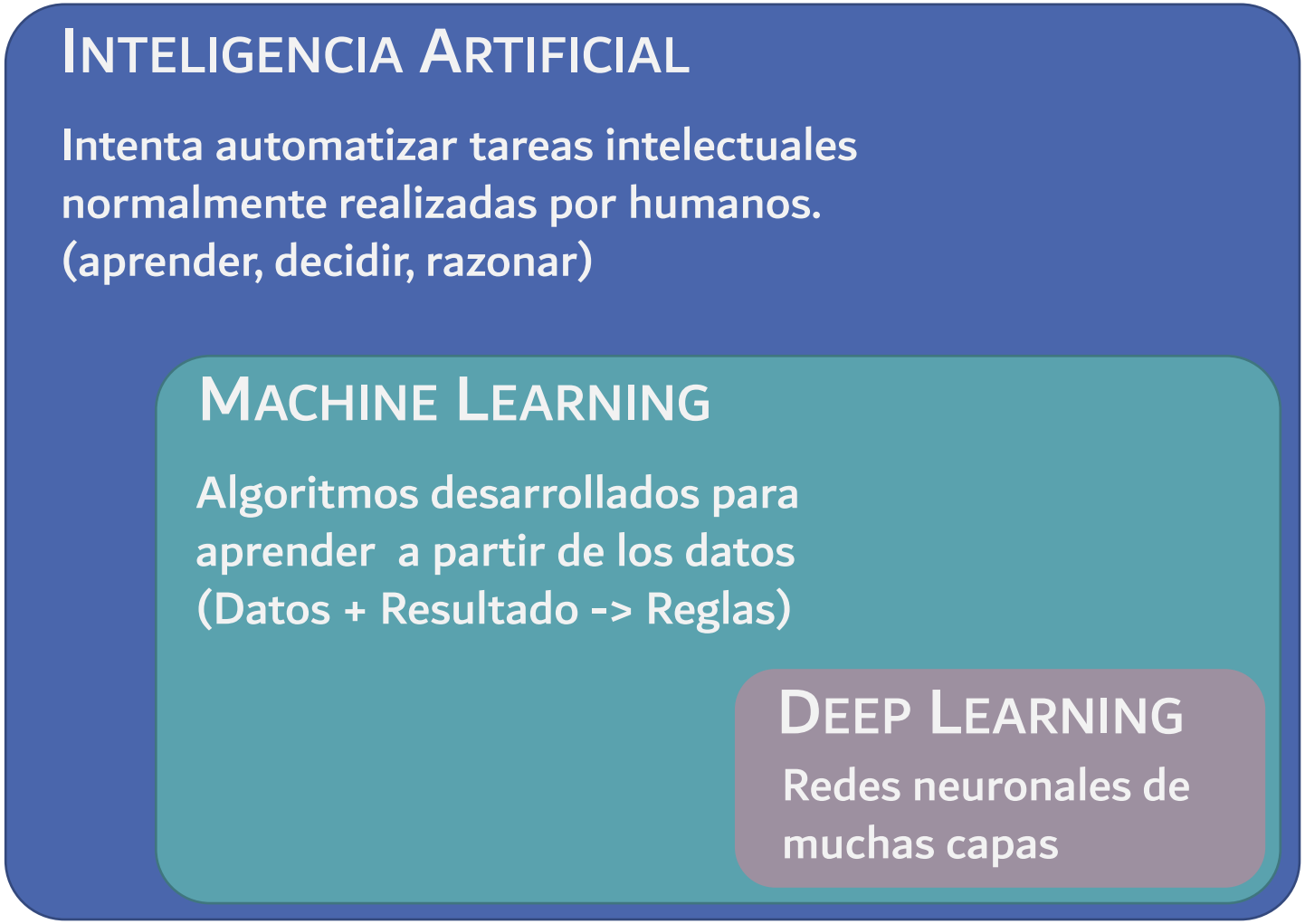
Optimiza en base al “premio” (refuerzo)

Ojo, existe también aprendizaje semi-supervisado.

# Machine Learning e Inteligencia Artificial

En Machine & Statistical learning entran los algoritmos que permitirán automatizar la tarea intelectual...

Mientras que Deep learning es una red neuronal, la cual tiene múltiples (muchas) capas en su arquitectura.



# Data Science vs Machine Learning vs Deep Learning

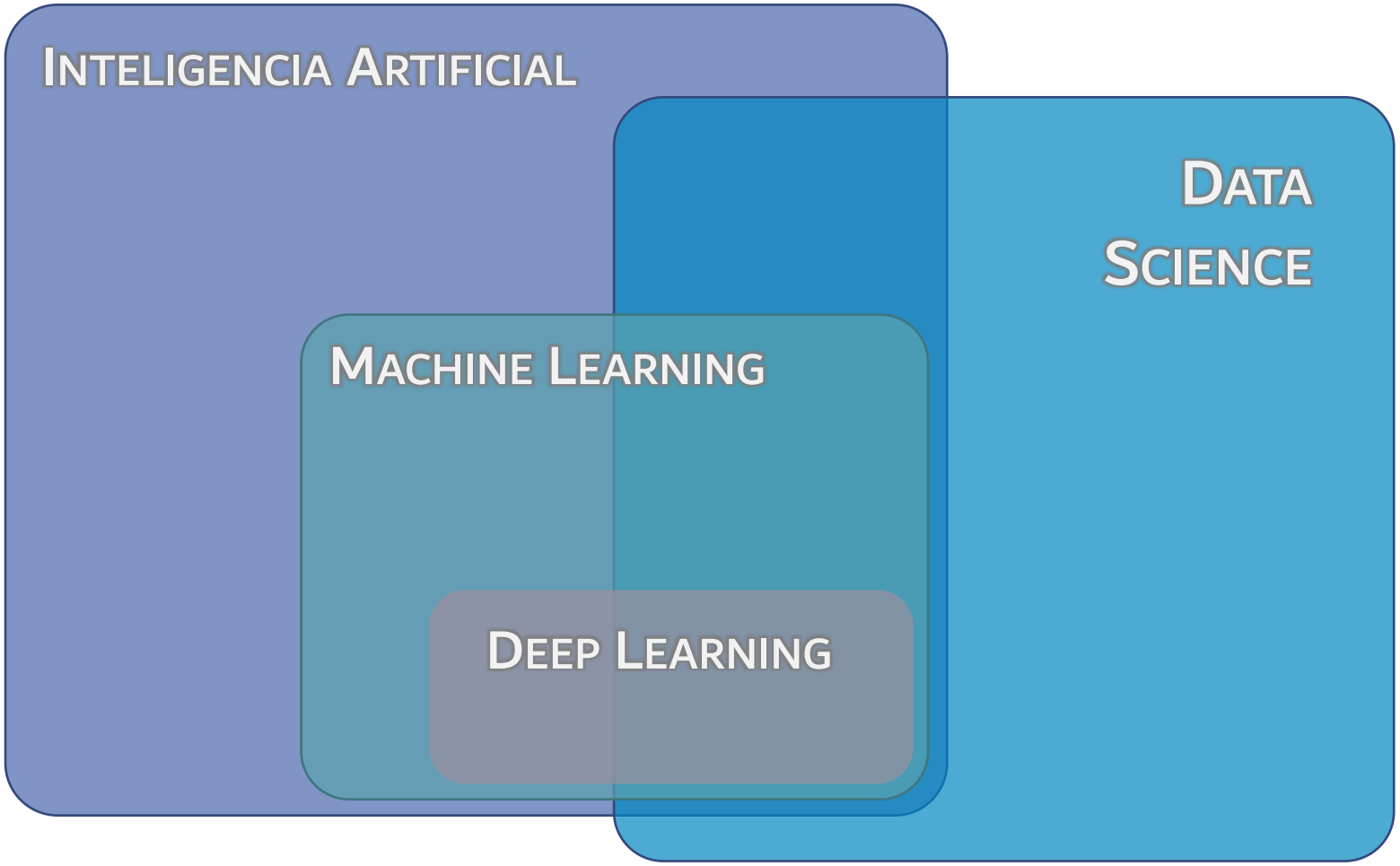
La Ciencia de datos tiene 4 objetivos:

**Descriptivo:** Comprender un set de datos.

**Exploratorio:** Cuando queremos encontrar comportamientos o relaciones desconocidas.

**Inferencial:** Aquí usamos una muestra para concluir sobre la población u otra muestra.

**Predicción:** Usamos una muestra para estimar los valores para nuevas observaciones.



# Data Science vs Machine Learning vs Deep Learning

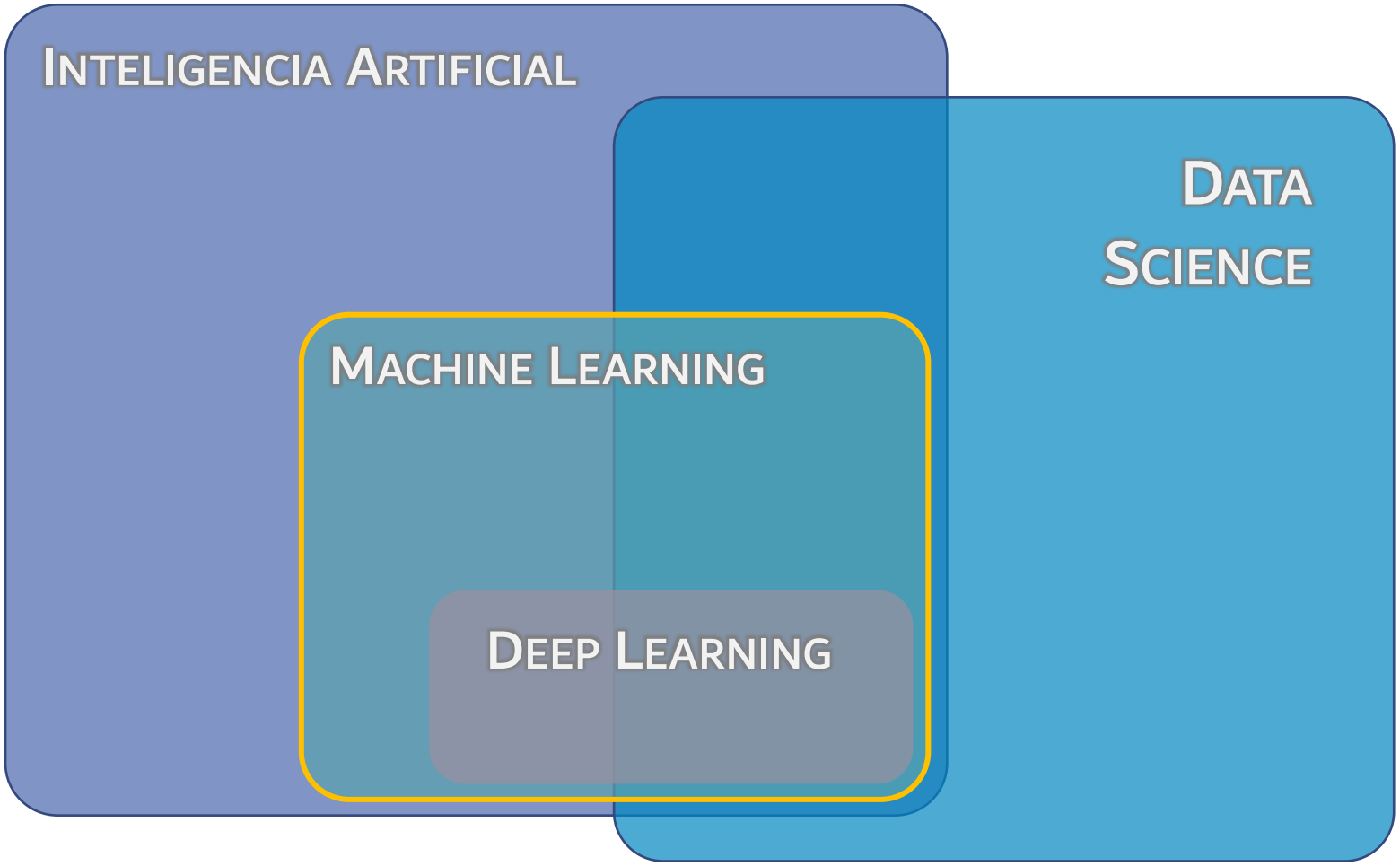
La Ciencia de datos tiene 4 objetivos:

**Descriptivo:** Comprender un set de datos.

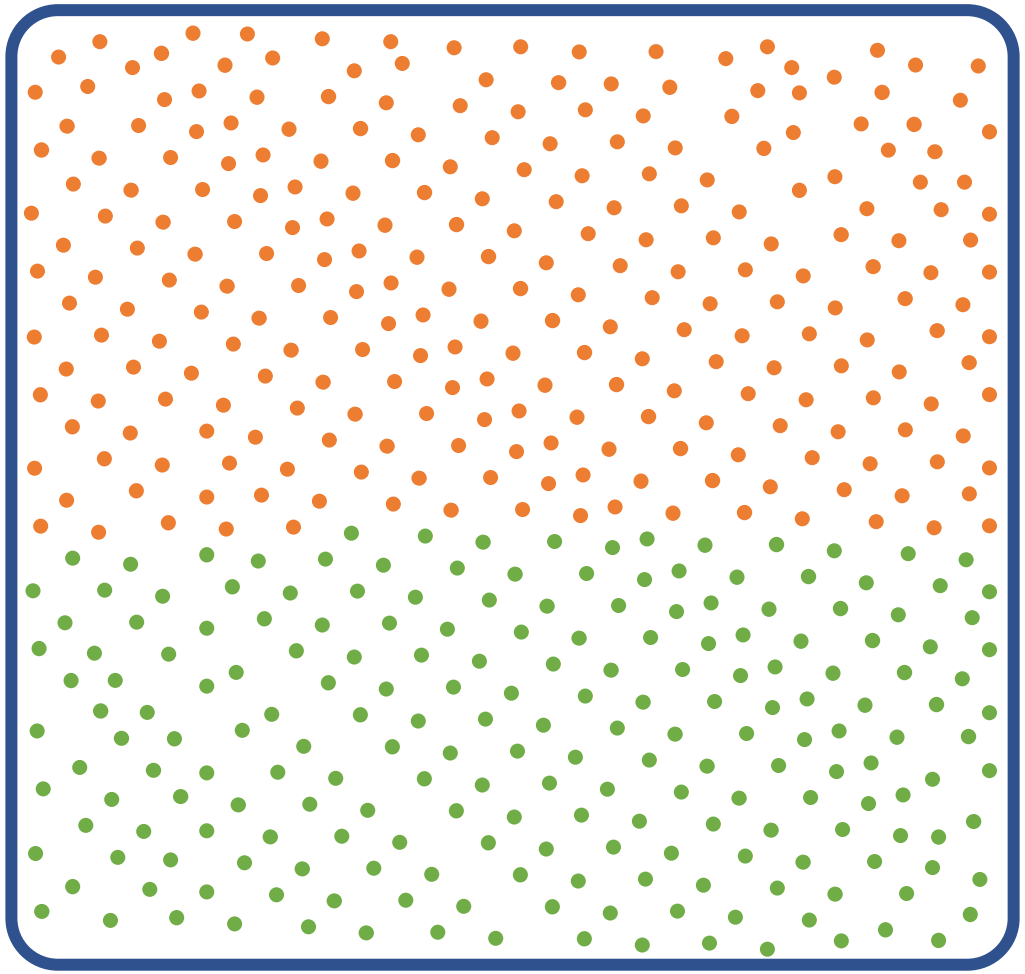
**Exploratorio:** Cuando queremos encontrar comportamientos o relaciones desconocidas.

**Inferencial:** Aquí usamos una muestra para concluir sobre la población u otra muestra.

**Predicción:** Usamos una muestra para estimar los valores para nuevas observaciones.



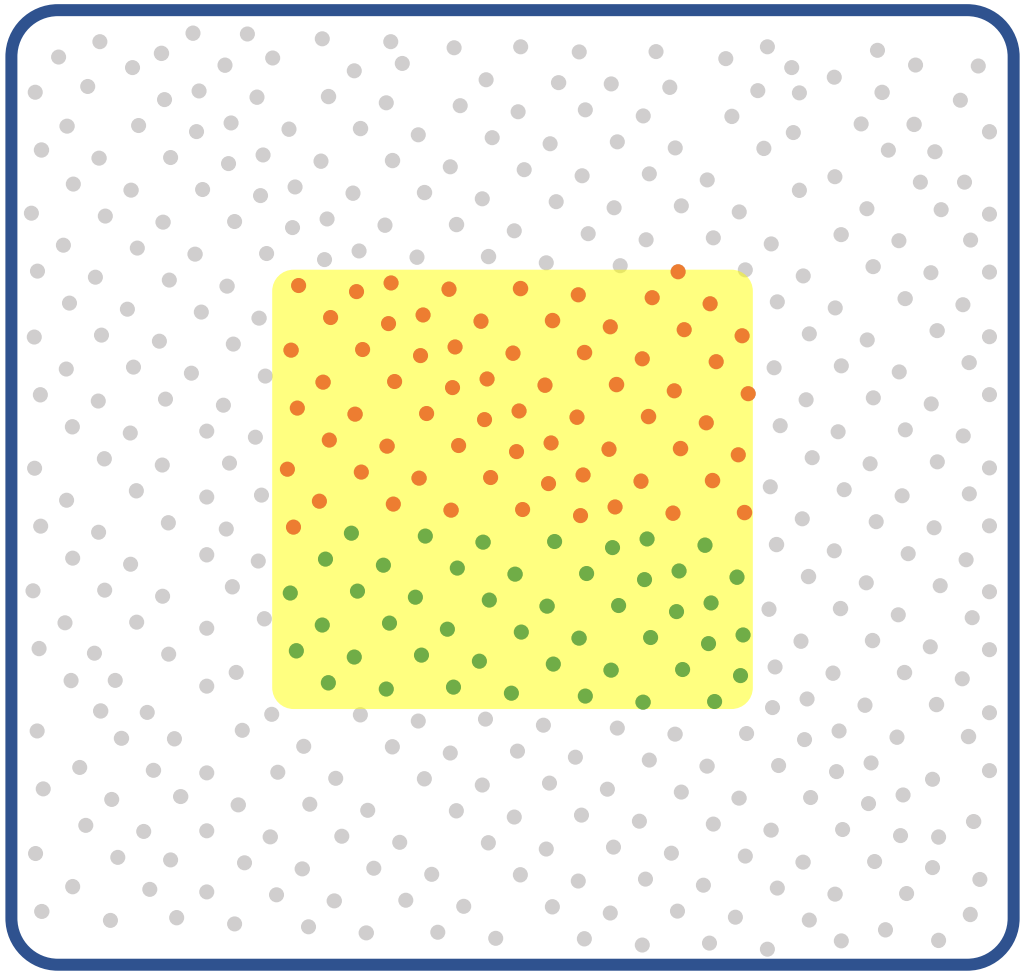
# Inferencia vs Predicción



**Suponga que desea determinar qué equipo de fútbol es más seguido en una ciudad**

**Población:** Los fanáticos de futbol de la ciudad

# Inferencia

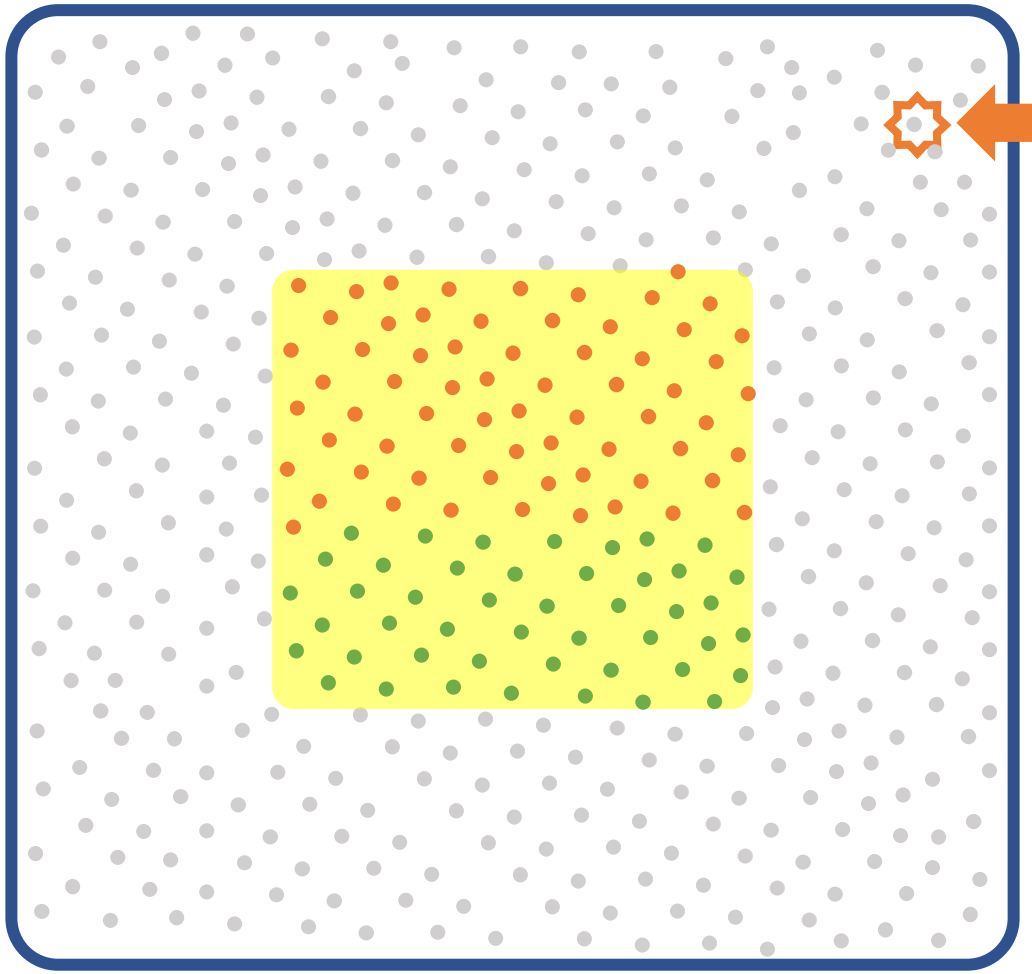


**Suponga que desea determinar qué equipo de fútbol es más seguido en una ciudad**

Como es imposible preguntar a todos, entonces de dicha **población** se obtiene una **muestra "representativa"** a quienes se les pregunta su preferencia.

Luego se podría aplicar estadística para inferir intervalos de confianza para el porcentaje de fanáticos de cada equipo en la población

# Predicción



**Suponga que desea saber de quién es fanático esta persona**

Ahora usaremos esta muestra para encontrar patrones entre los fanáticos de cada equipo, de tal manera que podamos predecir con “buen grado” de aceptividad de quién es fanático cualquier persona.

# Recordemos los tipos de Aprendizaje

en particular nos centraremos en el caso del aprendizaje supervisado.

## Supervisado

Input: X
Variables independientes o descriptoras

Output: Y
Variable dependiente

Regresión (predecir valor),

Clasificación (Predecir clase)

Optimiza en base al error de predicción

## No Supervisado

Input: X
Variables descriptoras

Cluster,

Reducción de dimensiones

Reglas de asociación (*)

## Por Refuerzo

Input: Estado y acciones posibles

Output: Decisión / acción

Auto conducción,

Navegación,

Realizar de tareas

Optimiza en base al "premio" (refuerzo)

Ojo, existe también aprendizaje semi-supervisado.

# Aprendizaje supervisado

Respuesta = **f(**variables explicativas o inputs**)** + ruido o error

**inputs**: Datos

**f( )**: Modelo

# Proyecto de Ciencia de datos: Proceso

Ojo, que en un proyecto de ciencia de datos se siguen muchos pasos, nos estamos centrando sólo en el modelamiento:

- **Entender el problema:** Esto tiene que ver más con entender el proceso o el negocio
- **Definir la tarea** de DS a realizarse: ¿Necesitamos inferencia? predecir? clasificar? Qué vamos a predecir? Cómo se establece que "x" observación pertenece a la clase "c"?
- **Obtener datos:** ¿Qué datos puedo usar? Cómo los obtengo? Tienen sesgo?
- **Exploración y preprocesamiento** de datos: Donde terminan de conocer los datos y comprender las relaciones entre variables
- **Modelamiento:** ¿qué modelo(s) aplicar? Cómo saber cuál es el mejor modelo?
- **Evaluar el modelo en "la vida real"**: Medir su efectividad en un escenario real
- **Poner el modelo en producción:** Ponerlo a disposición de los usuarios
- **Monitorear, reentrenar y/o redefinir:** Se repite el cliclo, a veces incluso desde el primer paso!


www.scikit-learn.org

https://scikit-learn.org/stable/about.html#history

Biblioteca open source (Python) que facilita y estandariza el entrenamiento de un gran número de modelos de aprendizaje supervisado y no supervisado

- Preprocesamiento de datos,
- Ajuste de modelos,
- Selección y evaluación de modelos,

Entre otras utilidades.


# Usemos scikit-learn para entrenar un modelo supervisado

# CASO

El Banco SEE desea mejorar los tiempos de atención al cliente en ventanilla, la estrategia requiere **predecir el tiempo en segundos que demora la atención para cada cliente.**

Se cuenta con datos recolectados anónimamente para cada cajero y transacción realizada, suministrados en un excel con tres hojas:

- Hoja con los datos de las transacciones, columnas: Sucursal, Cajero, ID_Transaccion, Transaccion, Tiempo_Servicio_seg, Nivel de satisfacción, Monto de la transaccion.
- Hoja que indica si en la sucursal se ha puesto o no el nuevo sistema.
- Hoja con los datos de los cajeros: Edad, sexo, año de ingreso a la empresa.

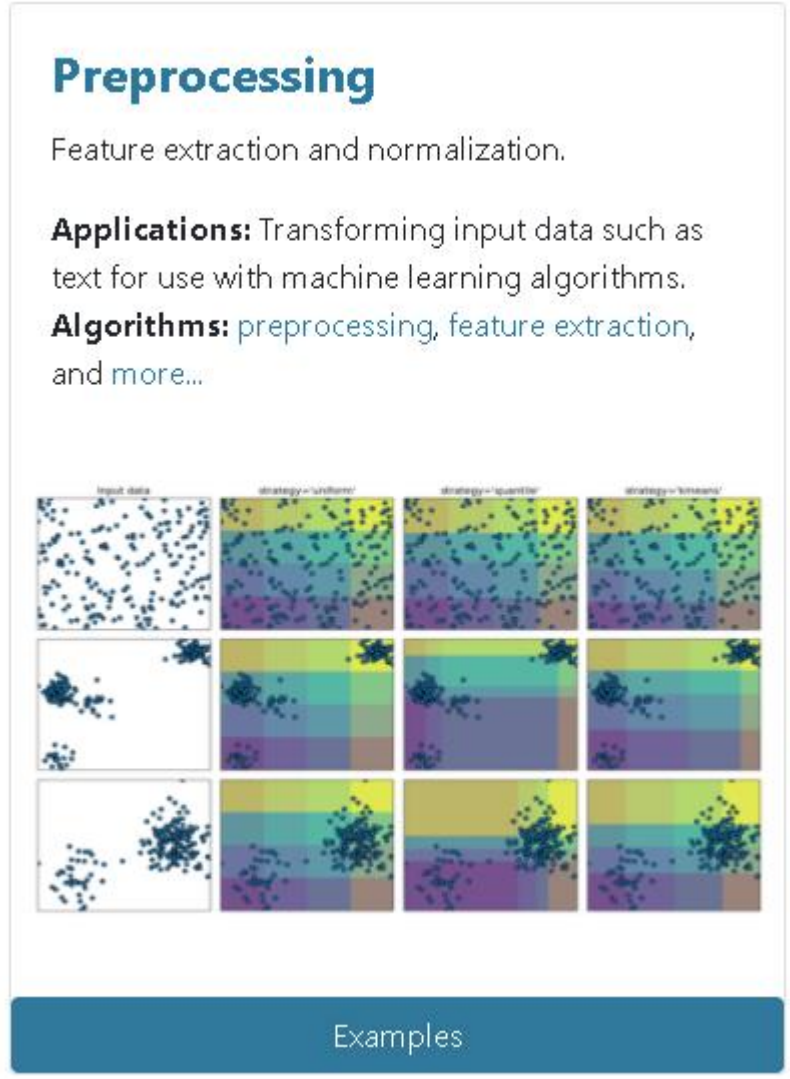
Abrir y copiar en su drive este Jupyter notebook de Google Colab:
https://colab.research.google.com/drive/1B5k8QYPt7m7tE6LXSi6oYF2RjgKLFtNk

# Preprocesamiento

Scikit-learn tiene su módulo de preprocesamiento que permite entre otras cosas:

- Estandarizar variables numéricas
- Imputación de datos
- Aplicar transformaciones a variables numéricas
- Codificar variables categóricas
- Crear nuevas variables
- etc.

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



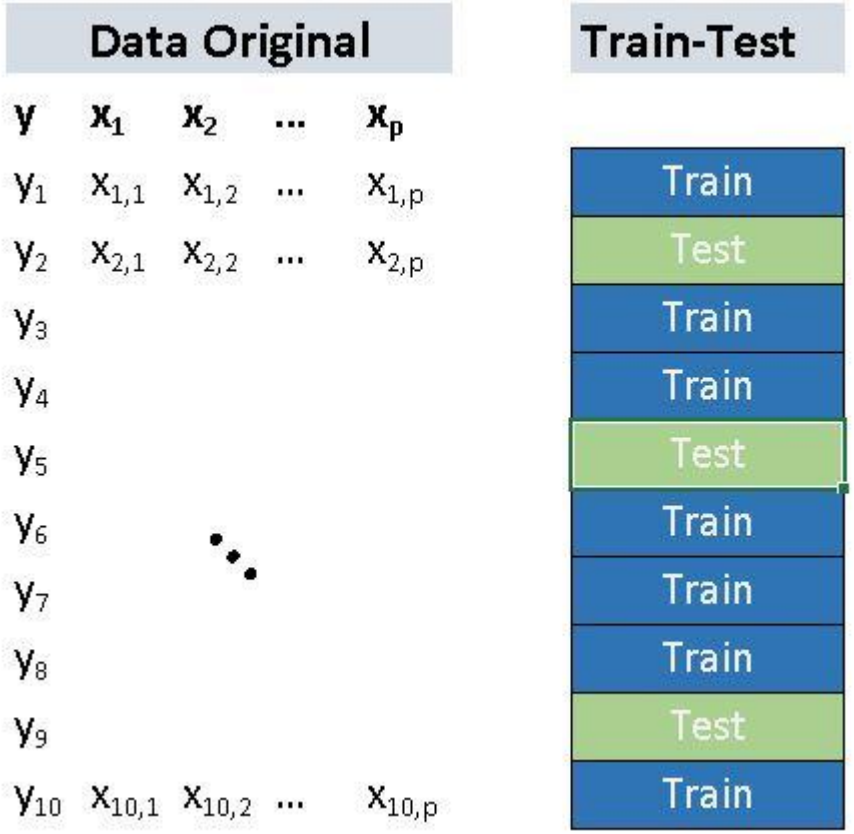
# Particionar datos: Train - Test

Recordemos que nuestro objetivo es poder aplicar el modelo a futuras observaciones, entonces verificar si el modelo da "buenos resultados" en datos no vistos antes parece ser lo más natural; nace entonces la **estrategia train - test**.

**Estrategia Train & Test:** El modelo se construye con una parte del set de datos y se evalúa con la otra parte.

- Datos de entrenamiento (train-set)
- Datos de prueba (test-set)

Con el módulo model_selection la función train_test_split nos permitirá crear nuestra partición Train-Test

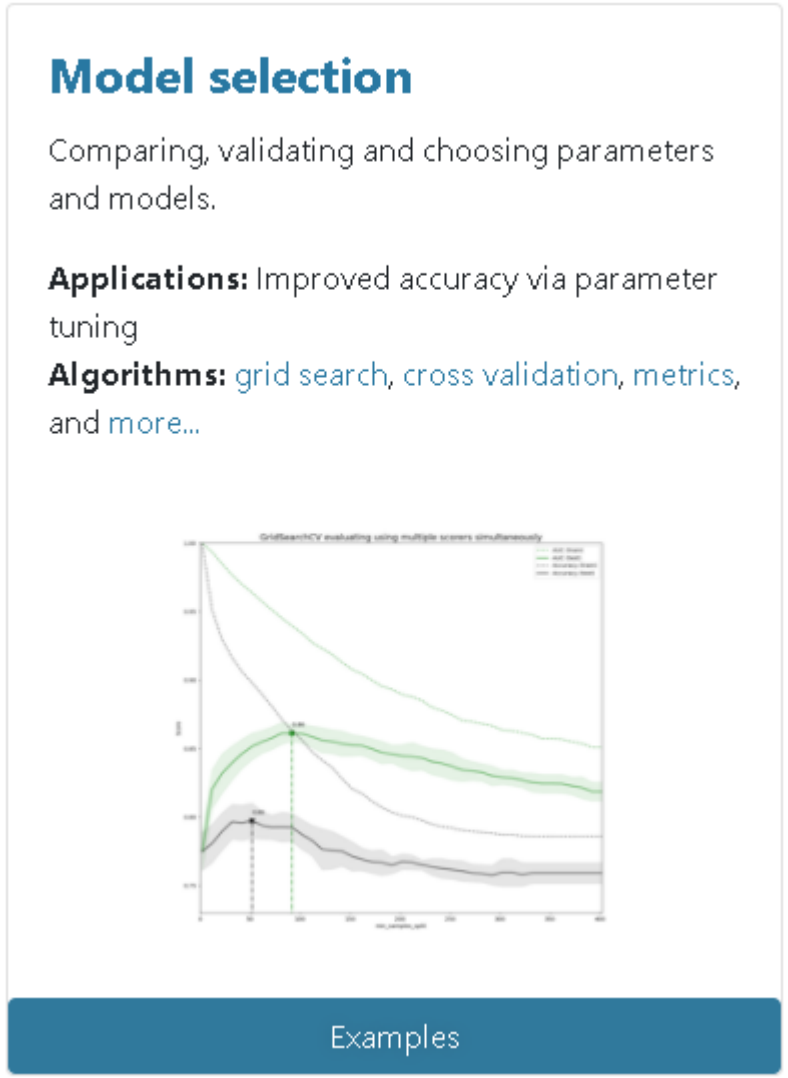


# Model Selection

El módulo de selección de modelo de Scikit-learn permite los siguientes pasos del modelamiento:

- Remuestro para definir la validación
- Métricas de evaluación
- Tuning de hyperparametros
- Y más

https://scikit-learn.org/stable/model_selection.html



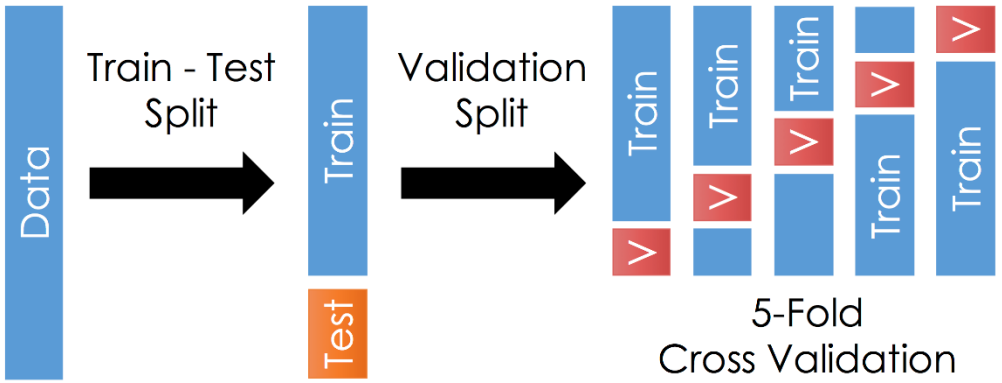
# Estrategia de Remuestreo

Además del test, se suele definir una validación que sin "topar el test-set" permita:

- Evaluar la estabilidad del modelo,
- Estimar el error del modelo.

Comúnmente se define una estrategia de remuestreo como Cross-validation

Además, la validación facilita el **afinamiento (tunning) de los hiperparámetros** (ya se explicará esto)



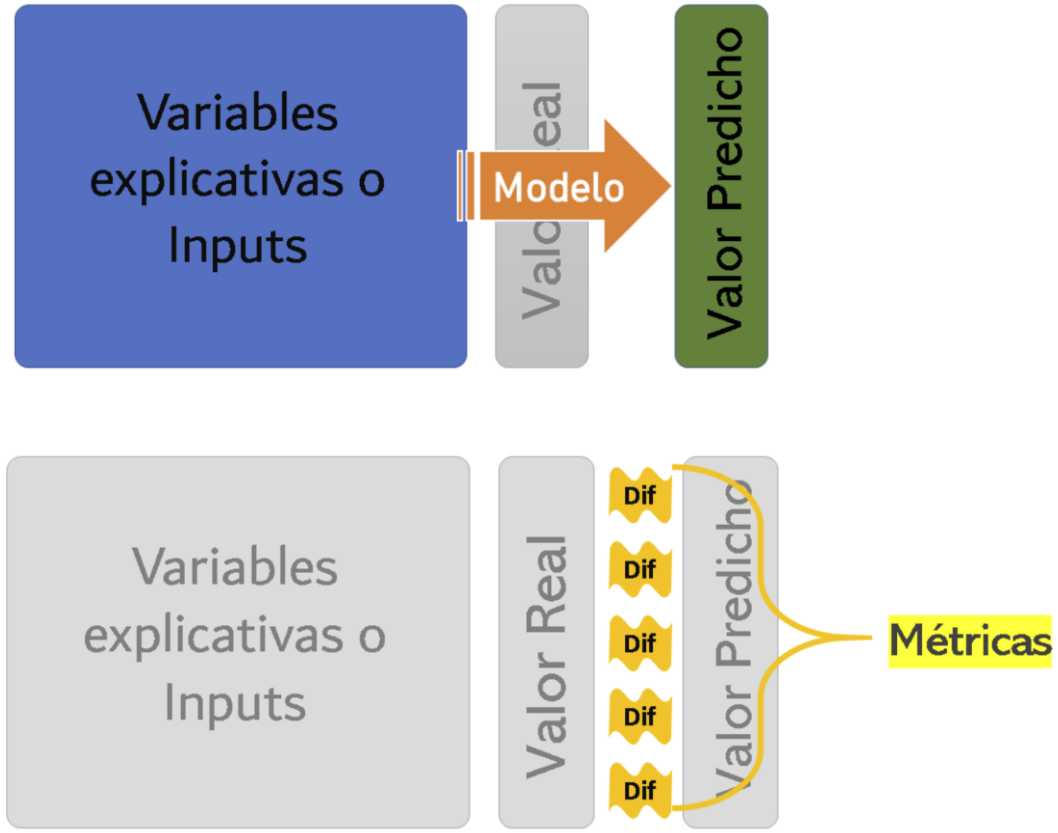

https://scikit-learn.org/stable/modules/cross_validation.html

# Métricas y evaluación del modelo

Estamos buscando el mejor modelo, ok, ¿pero... **¿"mejor" en función de qué?**.

Aquí entran las diferentes métricas que se han definido tanto para problemas de regresión (variable numérica) como para problemas de clasificación (variable categórica), scikit-learn tiene todo un módulo y pueden ser usadas tanto en el entrenamiento como para evaluar la bondad del modelo final.

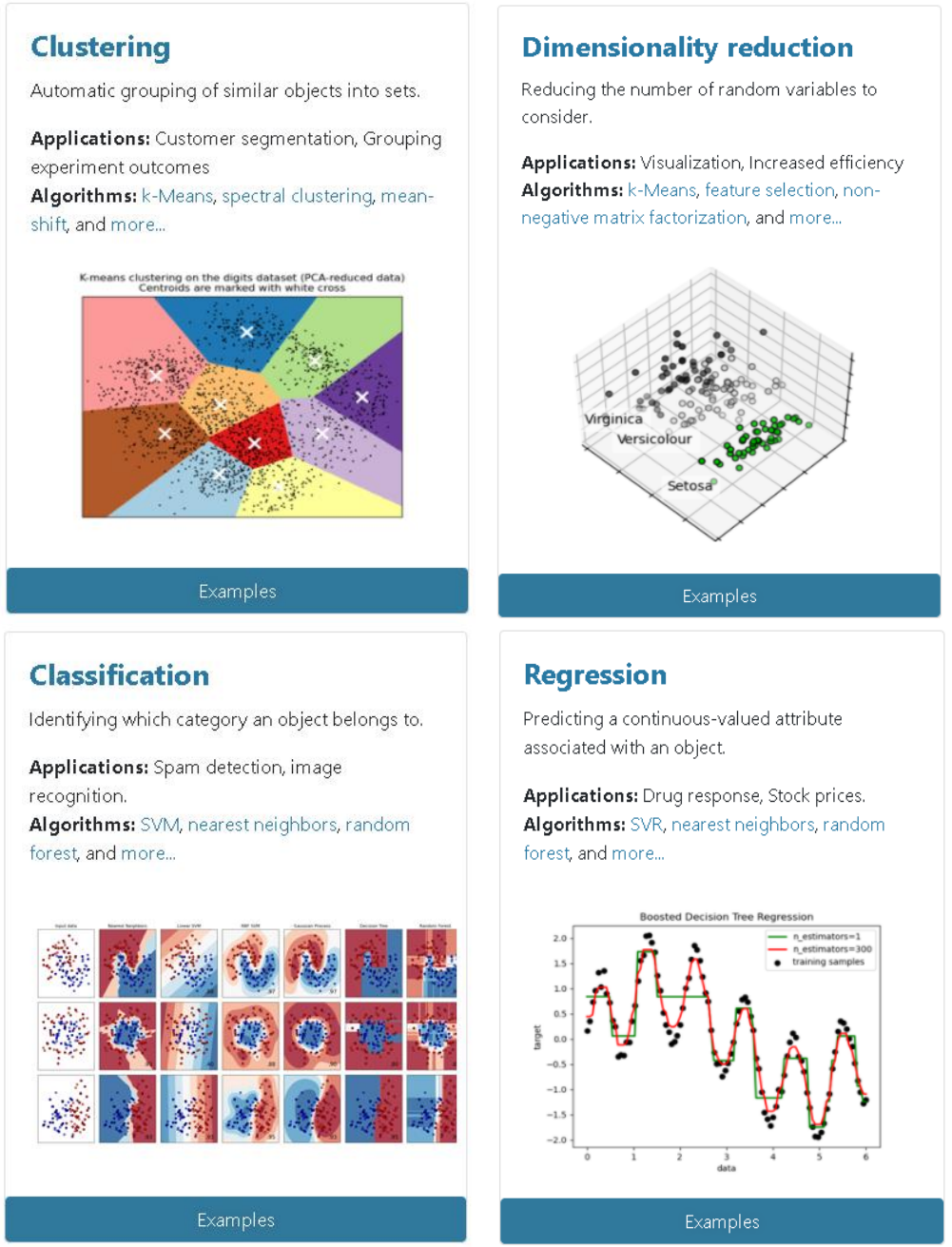
https://scikit-learn.org/stable/modules/model_evaluation.html



# Definir Modelo

De acuerdo a la tarea que deseamos realizar, Scikit-learn tiene implementado diversos tipos de modelos, este es en realidad uno de sus puntos fuertes: la gran cantidad de modelos disponibles.

Cada familia de modelos tendrá su modulo, (aparte de model-selection), sin embargo una vez definido el modelo, si este tiene hiperparámetros usamos model-selection para ajustar los mismos (*tuning de hiperparámetros*)



# Definir Modelo: KNN

Por facilidad vamos a usar uno de los algoritmos más sencillos, el de **K vecinos cercanos.**

La idea de este modelo es que observaciones cercanas deben ser parecidas, es decir, si estoy queriendo predecir la cantidad de visitas de los clientes y el cliente X es muy parecido (en variables como edad, ubicación, gustos, etc) a los clientes A, B, C; entonces la cantidad de visitas de X será parecida a la cantidad de visitas de A, B, C.

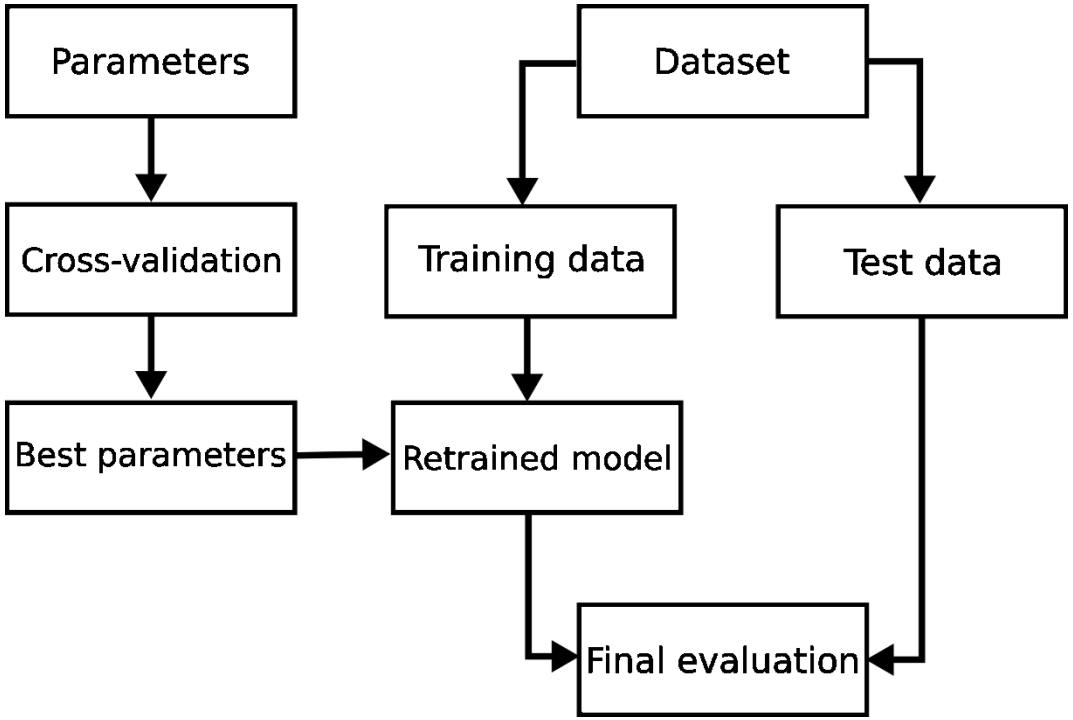
El hiperparámetro principal de este modelo sería: cantidad de vecinos usar

# Tuning de hiperparámetros

Recordemos, los parámetros de un modelo al estar en la “formula” del mismo se encuentran al optimizar la función de costo del modelo, pero ¿cómo optimizar los hiperparámetros si no forman parte de la función de costo?

Una de las técnicas más usadas es **Grid Search** con la que nosotros predefinimos el conjunto de valores a probar para cada hiperparámetro y se usa el remuestreo para estimar el error que tendría el modelo con esos valores de hiperparámetros.

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




Fuente imagen: Scikit-learn

# Mucho más!!

Scikit-learn tiene otras muchas utilidades como creación de pipelines que incluyan el modelo o postprocesamiento, preprocesamiento diferenciado por una columna "selectora" graficar métricas de evaluación, datasets integrados para probar modelos y aprendizaje, selección de variables, etc.

Para más información pueden seguir la guía oficial:
https://scikit-learn.org/stable/getting_started.html

# GRACIAS!

## Bases de scikit learn para modelos de Machine & Statistical learning

Por: **Néstor Montaño P.**
in/nestor-montaño/