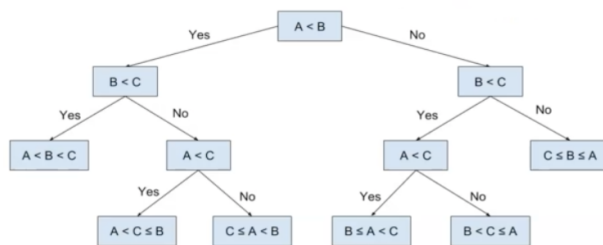


# Introducción a XGBoost - Mirada macro

## What is XGBoost? Why should you use it?

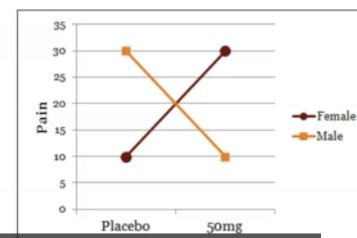
- One of the most powerful AI & Machine learn algorithms based on the gradient boosted decision trees



	Drink	
	YES	NO
Pill	YES 5lbs	2lbs
NO	2lbs	1 lb

Weight loss after one week

interactions



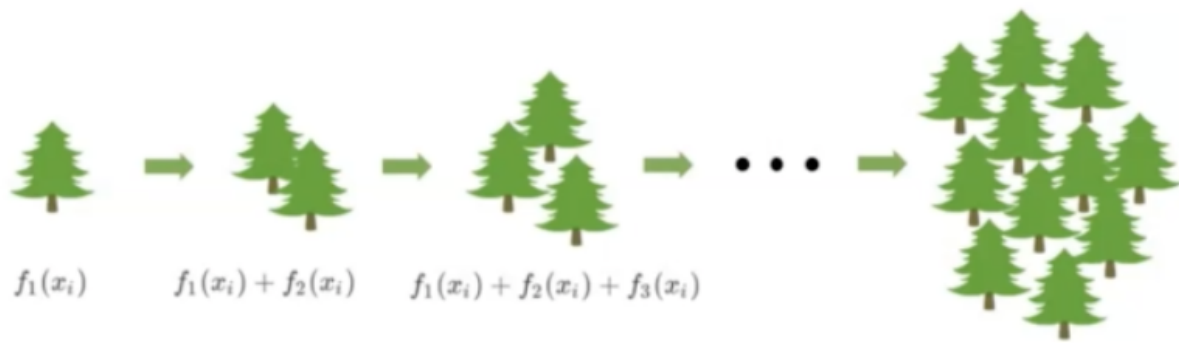
Es una especie de iteración y modelo iterativo, en donde busca corregir el error del modelo anterior.



A cada paso voy a predecir el error generado por el modelo anterior, usamos la iteración para corregir la predicción previa.

Usamos el metodo iterativo porque es un proceso granular y así aprendiendo de los errores de los modelos anteriores.

En cada paso aplicaremos un arbol y al final habrán tantos arboles que en cada paso se corregirá la predicción del anterior, predicción secuencial.



Los árboles de decisión pueden resolver patrones no lineales entre los datos, también lidiar con valores perdidos (missing data),

Por qué se usa de manera secuencial? si usamos un método secuencial aditivo (o iterativo) para hacer una predicción combinada, será un modelo extremadamente accurate.

## Entendiendo el gradient boosting

### Gradient Boost Tree Algorithm

$$Y = F(x) + \varepsilon_0$$

$$\hat{Y} = F(x)$$

$$\varepsilon_0 = F_1(x) + \varepsilon_1$$

$$\hat{\varepsilon}_0 = F_1(x)$$

$$\varepsilon_1 = F_2(x) + \varepsilon_2$$

$$\hat{\varepsilon}_1 = F_2(x)$$

$$\text{Predict} = \hat{Y} + \hat{\varepsilon}_0 + \hat{\varepsilon}_1 + \hat{\varepsilon}_2 + \dots$$

Tree model to iteratively predict error

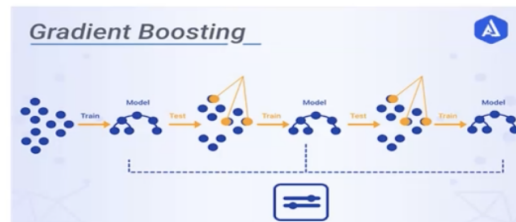
$F(x)$

$F_1(x)$

$F_2(x)$

Level-wise growth

Leaf-wise growth



Para cada modelo puedo definir una variable target  $Y$ , siendo  $F(x)$  cualquier modelo para predecir esa target, el modelo no es 100% accurate, por lo que añadimos  $e$  para representar el error:

$$Y = F(x) + e_0$$

Un modelo de regresión lineal es un modelo como ese, el gradient boosting Tree no se detiene ahí, por lo que, teniendo la predicción:  $\hat{Y} = F(x)$  por lo que en el segundo paso genero un modelo en el cual trataré de predecir el error del primer modelo, donde también agrego  $e_1$  asumiendo que tendré otro error.

$$\begin{aligned} e_0 &= Y - F_1(x) \\ \hat{e}_0 &= F_1(x) \end{aligned}$$

e iterativamente genero modelos para predecir los residuales

$$\begin{aligned} e_1 &= Y - F_2(x) \\ \hat{e}_1 &= F_2(x) \end{aligned}$$

este paso es el paso iterativo.

Sólo en la primera predicción se trata de predecir la target, en los demás modelos se intenta predecir el error.

Una vez que tengamos listo el proceso iterativo, se apunta a predecir la target con todos los árboles:

$$Predict = \hat{Y} + \hat{e}_0 + \hat{e}_1 + \dots + \hat{e}_n$$

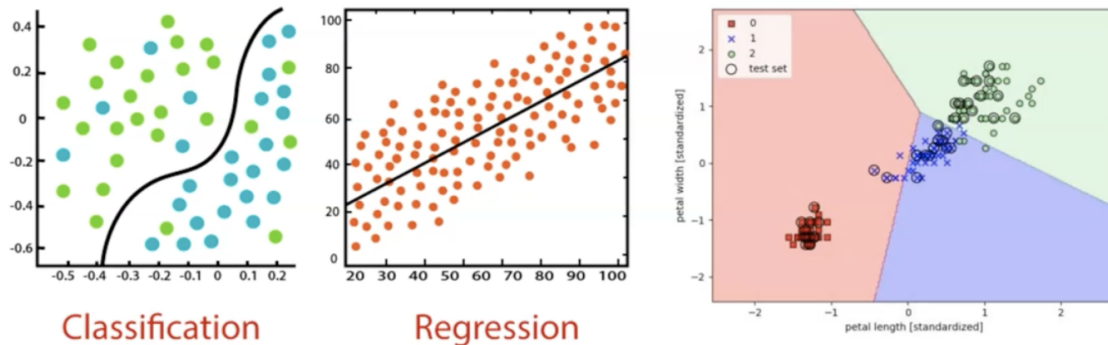
## Ventajas y aplicaciones de XGBoost

En cada iteración un decision tree es creado para predecir el error del modelo anterior, luego el error predecido (gradiente), es usado para mejorar la predicción final, entonces, el gradient boost machine es un modelo secuencial que busca llegar a una solución optima que es nuestra predicción final.

Este algoritmo puede ser usado para clasificación o regresión, y para varios tipos de datos.

## What is XGBoost? Why should you use it?

- It can be used to predict various types of data – continuous, binary, categorical...



Objective: what type of target or data type to be predicted  
num\_class: how many levels of target in classification problem

También puede manejar missing values, todos los missing values para los tipos de datos son ingestados como una clase en xgboost, pero no puede procesar valores categoricos (por lo que se recomienda hacer un encoding).

**Feature importance:** Una vez construido el modelo, todas las variables de entradas tienen cierto impacto en la variable target, y así podríamos entender la contribución de cada variable, XGboost tiene por defecto el index por feature importance, así nos podremos ayudar en la etapa de feature selection.

# What is XGBoost? Why should you use it?

- XGBoost provides various feature importance levels & plots

