**RANDOM FOREST**

Un rando forest es un conjunto (ensamble) de muchos árboles de decisión entrenamos de maneta independiente. La predicción final se obtiene mediante

* Clasificación: votación mayoritaria (la clase más votada entre los árboles)
* Regresión: promedio de las predicciones de los árboles

**Como funciona**

1. Boostraíng (bagging):
   1. Cada árbol se entrena con una muestra aleatoria (con reemplezo) del dataset
   2. Esto introduce variabilidad y evita que todos los árboles sean idénticos
2. Selección aleatoria de variables (feature randomness):
   1. En cada división (Split), el algoritmo no evalúa todas las variables, sino un subconjunto aleatorio.
   2. Esto reduce la correlación entre árboles y mejora la generalización
3. Agregación (ensemble):
   1. Para clasificación, se usa votación
   2. Para regresión, se toma el promedio

**Ventajas**

* Menos propenso al overfitting que un solo árbol
* Maneja datasets grandes y de alta dimensionalidad
* Funciona bien con datos faltantes o mezclas de variables categóricas y numéricas

**Desventajas**

* Difícil de interpretar
* Más costoso computacionalmente que un solo árbol

**GRADIENTE DE BOOSTING**

Mientras que random forest entrena arboles independientes en paralelo, gradiente boosting entrena árboles secuenciales, donde cada nuevo árbol corrige los errores de los anteriores.

Como funciona

1. Se entrena un árbol base (generalmente pequeño y débil, como un stump)
2. Se calculan los errores residuales (que tanto se equivocó el modelo)
3. Se entrena un nuevo árbol sobre esos residuales
4. Se actualiza el modelo combinando el árbol anterior + el nuevo árbol, ajustando con un learning rate (tasa de aprendizaje)
5. Se repite el proceso muchas veces

**Ventajas**

* Muy alta precisión en muchos problemas reales
* Permite controlar el bias-variance tradeoff con hiperparámetros (learning rate, profundidad, números de árboles)
* Es la base de algoritmos muy usados en la industria como XGBoost, LightGBM y CatBoost

Desventajas

* Más sensible al overfitting sino se ajusta bien los parámetros
* Computacionalmente más lento que el Random Forest (porque es secuencial)