

# XGBoost : Extreme Gradient Boosting

Chaymae Gasmi

## 1 Definition

XGBoost est une bibliothèque optimisée de renforcement de gradient distribuée conçue pour être très efficace, flexible et portable. Il implémente des algorithmes d'apprentissage automatique dans le cadre de Gradient Boosting. XGBoost fournit un boost d'arborescence parallèle (également connu sous le nom de GBDT, GBM) qui résout de nombreux problèmes de science des données de manière rapide et précise.

## 2 Optimizing Tree structure

Pour  $\hat{y} = \sum_{b=1}^B f_b$ , une fonction coût général  $l$ , on veut minimiser la fonction objectif suivante :

$$\min \leftarrow \sum_{i=1}^n l(y_i, \hat{y}_i) + \sum_{b=1}^B \Omega(f_b)$$

avec  $\Omega(f_b) = \gamma|T| + \frac{1}{2}\lambda \sum_{j=1}^{|T|} w_j^2$ , et  $|T| = d + 1$  est le nombre de feuille de l'arbre de décision,  $w_j$  sont les poids de régression d'une feuille (région)  $R_j$

## 3 Additive training (Boosting)

On part d'une prédiction constante et on ajoute une nouvelle fonction à chaque fois :

$$\begin{aligned}\hat{y}_i^{(0)} &= 0 \\ \hat{y}_i^{(1)} &= f_1(x_i) = \hat{y}^{(0)} + f_1(x_i) \\ \hat{y}_i^{(2)} &= f_1(x_i) + f_2(x_i) = \hat{y}^{(1)} + f_2(x_i) \\ &\dots \\ \hat{y}_i^{(t)} &= \sum_{b=1}^t f_b(x_i) = \hat{y}^{(t-1)} + f_t(x_i)\end{aligned}$$

## 4 XGBoost implementation

- Comment décidons-nous quels  $f_t$  (fractionnements) ajouter? On Optimise l'objectif! - La prédiction au rang  $t$  est  $\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i)$ , so that

$$\begin{aligned}\text{Obj}^{(t)} &= \sum_{i=1}^n l(y_i, \hat{y}_i^{(t)}) + \sum_{b=1}^t \Omega(f_b) \\ &= \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) + C\end{aligned}$$

- On considère la fonction cout  $l_2$  :

$$\begin{aligned}\text{Obj}^{(t)} &= \sum_{i=1}^n \left( y_i - \left( \hat{y}_i^{(t-1)} + f_t(x_i) \right) \right)^2 + \Omega(f_t) + C_1 \\ &= \sum_{i=1}^n \left[ 2 \left( \hat{y}_i^{(t-1)} - y_i \right) f_t(x_i) + f_t(x_i)^2 \right] + \Omega(f_t) + C_2\end{aligned}$$

## 5 Developpement de Taylor

On rapelle le developpement de Taylor :  $-f(x + \Delta x) \approx f(x) + f'(x)\Delta x + \frac{1}{2}f''(x)\Delta x^2$  On définit  $g_i = \partial_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})$ ,  $h_i = \partial_{\hat{y}_i^{(t-1)}}^2 l(y_i, \hat{y}_i^{(t-1)})$  Donc :

$$\text{Obj}^{(t)} \approx \sum_{i=1}^n \left[ l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2 \right] + \Omega(f_t) + C$$

- On considère le cas de la fonction cout  $l_2$  :

$$\begin{aligned}g_i &= \partial_{\hat{y}_i^{(t-1)}} \left( \hat{y}_i^{(t-1)} - y_i \right)^2 = 2 \left( \hat{y}_i^{(t-1)} - y_i \right) \\ h_i &= \partial_{\hat{y}_i^{(t-1)}}^2 \left( \hat{y}_i^{(t-1)} - y_i \right)^2 = 2\end{aligned}$$

## 6 Retour sur les objectifs

- On définit  $J : \mathbb{R}^p \rightarrow T, x \mapsto R_j$  for  $j : x \in R_j$ , On Supprime les constantes et regroupe par feuilles :

$$\begin{aligned}\text{Obj}^{(t)} &\approx \sum_{i=1}^n \left[ g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \\ &= \sum_{i=1}^n \left[ g_i w_{J(x_i)} + \frac{1}{2} h_i w_{J(x_i)}^2 \right] + \gamma |T| + \frac{1}{2} \lambda \sum_{j=1}^{|T|} w_j^2 \\ &= \sum_{j=1}^{|T|} \left[ \left( \sum_{i: x_i \in R_j} g_i \right) w_j + \frac{1}{2} \left( \sum_{i: x_i \in R_j} h_i + \lambda \right) w_j^2 \right] + \gamma |T|\end{aligned}$$

- C'est la somme de  $|T|$  fonctions quadratiques indépendantes, on reconnait deux faits sur la fonction quadratique à variable unique

$$\arg \min_x Gx + \frac{1}{2} H x^2 = -\frac{G}{H}, H > 0, \quad \min_x Gx + \frac{1}{2} H x^2 = -\frac{1}{2} \frac{G^2}{H}$$

- On définit  $G_j = \sum_{i: x_i \in R_j} g_i$  and  $H_j = \sum_{i: x_i \in R_j} h_i$ , donc :

$$\text{Obj}^{(t)} \approx \sum_{j=1}^{|T|} \left[ G_j w_j + \frac{1}{2} (H_j + \lambda) w_j^2 \right] + \gamma |T|$$

- Supposons la structure de l'arbre  $(R_j)_{j=1}^{|T|}$  est fixée, le poids optimal de chaque feuille et la valeur objective résultante sont :

$$w_j = -\frac{G_j}{H_j + \lambda}, \quad \text{Obj}^{(t)} \approx -\frac{1}{2} \sum_{j=1}^{|T|} \frac{G_j^2}{H_j + \lambda} + \gamma |T|$$

## 7 XGBoost parameters

- ETA [par défaut = 0,3, alias: taux d'apprentissage] Réduction de la taille des pas utilisée dans la mise à jour pour éviter le surajustement. Après à chaque étape de boost, nous pouvons directement obtenir le poids des nouvelles fonctionnalités, et eta réduit les poids des fonctionnalités pour rendre le processus de renforcement plus conservateur.
- GAMMA [par défaut = 0, alias: minsplitloss] Réduction minimale des pertes requise pour effectuer une nouvelle partition sur un nœud feuille de l'arbre. Plus le gamma est grand, plus il est conservateur l'algorithme sera.
- Maxdepth [par défaut = 6] Profondeur maximale d'un arbre. Augmenter cette valeur rendra le modèle plus complexe et plus susceptible de dépasser t. 0 indique non limite.
- Colsamplebytree [par défaut = 1]
- Rapport de sous-échantillon des colonnes lors de la construction de chaque arbre.
- Le sous-échantillonnage se produira une fois dans chaque itération d'amplification.