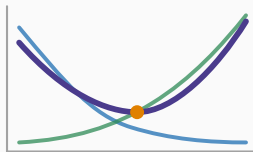


Apprentissage statistique

Chapitre 10 : Le boosting de gradient



Jaouad Madkour

Faculté d'Économie et de Gestion de Tanger · 27 août 2026

Apprendre de ses erreurs

Les trois réglages

Les implémentations modernes

Ce que l'on perd

Ce qu'il faut retenir

Apprendre de ses erreurs

Le renversement par rapport au chapitre 9

Deux stratégies d'ensemble

	Bagging / forêt	Boosting
Construction	en parallèle	en séquence
Chaque modèle	ignore les autres	corrige le précédent
Modèles de base	arbres profonds	arbres très courts
Cible principale	la variance	le biais

Le principe du boosting

1. Partir d'une prédiction constante.
2. Calculer les **résidus** — ce que le modèle actuel se trompe.
3. Ajuster un petit arbre **sur ces résidus**.
4. L'ajouter au modèle, avec un poids faible.
5. Recommencer.

Intuition

Ajuster un arbre sur les résidus revient à faire un pas dans la direction qui réduit le plus la perte : c'est une **descente de gradient**, où chaque pas est un petit arbre.

Le cours *Apprentissage profond* reprendra exactement cette idée, avec des neurones à la place des arbres.

Les trois réglages

Ils interagissent, et c'est le point délicat

Les hyperparamètres essentiels

Paramètre	Rôle	Valeur usuelle
Taux d'apprentissage η	poids de chaque arbre	0,01 à 0,1
Nombre d'arbres B	longueur de la séquence	des centaines à des milliers
Profondeur	complexité de chaque arbre	3 à 8

La règle qui gouverne les deux premiers

η et B sont liés : diviser η par deux exige de doubler B pour parcourir la même distance.

Un η faible avec un B élevé donne presque toujours de meilleurs résultats — chaque pas est prudent, et l'ensemble converge mieux. Le prix est le temps de calcul.

On fixe donc η petit, et on choisit B par arrêt précoce.

Définition

Surveiller l'erreur sur un ensemble de validation à chaque itération, et s'arrêter quand elle cesse de baisser.

C'est indispensable, car contrairement à la forêt, **le boosting sur-ajuste** si on le laisse continuer.

Application en gestion

	Augmenter le nombre d'arbres
Forêt aléatoire	l'erreur se stabilise — aucun risque
Boosting	l'erreur baisse, puis remonte — sur-ajustement

C'est ce qui rend le boosting plus performant et plus dangereux. La forêt pardonne les réglages approximatifs; le boosting non.

Les implémentations modernes

Ce qu'elles ajoutent

Trois bibliothèques répandues

XGBoost — régularisation explicite sur la structure des arbres, gestion native des valeurs manquantes.

LightGBM — croissance par feuille plutôt que par niveau, très rapide sur gros volumes.

CatBoost — traitement soigné des variables qualitatives à nombreuses modalités.

Les régularisations disponibles

- Pénalités L_1 et L_2 sur les valeurs des feuilles — le chapitre 4, transposé.
- **Sous-échantillonnage** des lignes et des colonnes à chaque arbre — le chapitre 9, transposé.
- Nombre minimal d'observations par feuille.

Pourquoi le boosting domine sur données tabulaires

Sur des tableaux structurés — clients, transactions, dossiers de crédit — le boosting de gradient bat régulièrement les réseaux de neurones, et de loin.

Les raisons tenues pour principales : il gère nativement les variables hétérogènes et de natures mélangées, il ne demande aucune normalisation, et il capte les interactions sans architecture à concevoir.

Ce que l'on perd

Le prix de la performance

Trois coûts

1. **Aucune interprétabilité directe** — mille arbres ne se lisent pas.
2. **Réglage exigeant** — plusieurs paramètres qui interagissent.
3. **Risque de sur-ajustement** réel, contrairement à la forêt.

La trajectoire du cours, en un regard

Chapitre	Modèle	Interprétabilité	Performance
4	Régression pénalisée	élevée	modérée
8	Arbre unique	excellente	faible
9	Forêt aléatoire	faible	élevée
10	Boosting	très faible	très élevée

Le cours a systématiquement échangé de la lisibilité contre de la précision. Le chapitre 12 tentera d'en récupérer une part.

Quand cet arbitrage n'est pas permis

Ce qu'il faut retenir

Cinq points

1. Le **boosting** construit les arbres en **séquence**, chacun ajusté sur les **résidus** du précédent.
2. Il attaque le **biais**, là où la forêt attaquait la variance ; ses arbres sont **courts**.
3. η faible et B élevé donnent les meilleurs résultats ; les deux sont liés.
4. Contrairement à la forêt, **il sur-ajuste** : l'**arrêt précoce** est indispensable.
5. Sur données **tabulaires**, il reste la référence — devant les réseaux de neurones.

La suite

Le chapitre 11 présente la dernière famille du cours, issue d'une intuition géométrique entièrement différente : les machines à vecteurs de support.