II : Apprentissage par Ensemble : Random Forest

Qu'est-ce qu'une Random Forest ? (1/2)

Définition : Une **Random Forest** est un ensemble de **plusieurs arbres de décision** construits à partir de :

- **Bootstrap** (Bagging) :
  - Création de plusieurs échantillons de taille égale à celle de l'ensemble initial, tirés **avec remise**.
  - Chaque échantillon peut ainsi contenir des doublons et ignorer certains exemples originaux.
- **Sélection aléatoire d'attributs** à chaque division (Random Subspace) :
  - Au lieu de tester tous les attributs, on en choisit **un sous-ensemble** aléatoire pour chaque nœud.
  - Cela augmente la **diversité** entre les arbres.

Qu'est-ce qu'une Random Forest ? (2/2)

Vote majoritaire ou moyenne

- **Classification** : la prédiction finale est le *vote majoritaire* de tous les arbres.
- **Régression** : la prédiction finale est la *moyenne* des prédictions.

Pourquoi utiliser une Random Forest ? (1/2)

Avantages :

- **Meilleure robustesse** : la variance du modèle est réduite par le vote/moyenne.
- **Réduction du risque de sur-apprentissage** (overfitting) par rapport à un arbre unique.
- **Facile à utiliser** : peu d'hyperparamètres critiques (nombre d'arbres, nombre d'attributs aléatoires, etc.).

Pourquoi utiliser une Random Forest ? (2/2)

Limites :

- **Moins interprétable** qu'un arbre unique (il est plus complexe de visualiser une "forêt").
- Peut être **coûteux en mémoire** et en temps de calcul pour un très grand nombre d'arbres.

Idée générale

Plus on a d'arbres *indépendants*, plus la **moyenne** de leurs erreurs se compense, améliorant la qualité globale.

Bagging : Bootstrap Aggregation (1/2)

Étapes clés pour construire une Random Forest (exemple de classification) :

1. **Échantillons Bootstrap** :
   - À partir de l'ensemble d'origine (taille N), on forme M sous-échantillons également de taille N, mais tirés **avec remise**.
   - Chaque sous-échantillon est utilisé pour entraîner **un arbre de décision**.
2. **Arbres aléatoires** :
   - À chaque nœud, au lieu de tester **tous les attributs**, on en sélectionne **un sous-ensemble** (exemple : $\sqrt{d}$ parmi d attributs).
   - On choisit l'attribut qui maximise la réduction du Gini **parmi ceux sélectionnés**.

Bagging : Bootstrap Aggregation (2/2)

3. **Vote majoritaire** :
   - Pour prédire une classe, on **combine les prédictions de chaque arbre** par un vote.
   - En régression, on prend la **moyenne** des valeurs prédites.

Note

Le *Bagging* seul réduit déjà la variance. La **sélection aléatoire d'attributs** *ajoutée* évite que tous les arbres se ressemblent trop (cas de Bagging pur).

Exemple : Mini-données (construction d'une Forêt) (1/2)

Données simplifiées (8 exemples) :

ID	Taille (m)	Poids (kg)	Jouer (Oui/Non)
1	1.50	60	Oui
2	1.80	80	Non
3	1.65	70	Oui
4	1.70	75	Non
5	1.55	62	Oui
6	1.90	90	Non
7	1.60	68	Oui
8	1.75	72	Non

Table – Jeu de données fictif

- **Attributs :** Taille, Poids.

Exemple : Mini-données (construction d'une Forêt) (2/2)

Étape 1 : Échantillons Bootstrap
On forme 3 échantillons (puisqu'on veut 3 arbres). Chaque échantillon est obtenu par **tirage avec remise** de 8 exemples :

$$\textbf{Échantillon \#1 :} \{1, 2, 2, 3, 5, 5, 7, 8\}$$
$$\textbf{Échantillon \#2 :} \{2, 4, 4, 5, 6, 6, 7, 8\}$$
$$\textbf{Échantillon \#3 :} \{1, 1, 2, 3, 6, 7, 8, 8\}$$

Remarque

Chaque échantillon est de taille 8 (même que l'ensemble initial), mais contient des **doublons** et éventuellement **omet** certains exemples (OOB – Out Of Bag).

Exemple : Construction des Arbres (Étape 2)

- Pour chaque échantillon, on construit un **arbre de décision** en utilisant la **sélection aléatoire d'attributs**.
- Si on a 2 attributs (Taille, Poids) :
  - À chaque nœud, on peut en tirer 1 au hasard (ou parfois les 2).
  - On choisit celui qui **maximise** la réduction de l'impureté (Gini).
- Ainsi, on obtient 3 arbres différents, chacun "surapprenant" potentiellement à sa portion de données, **mais** de façon distincte.

Conséquence

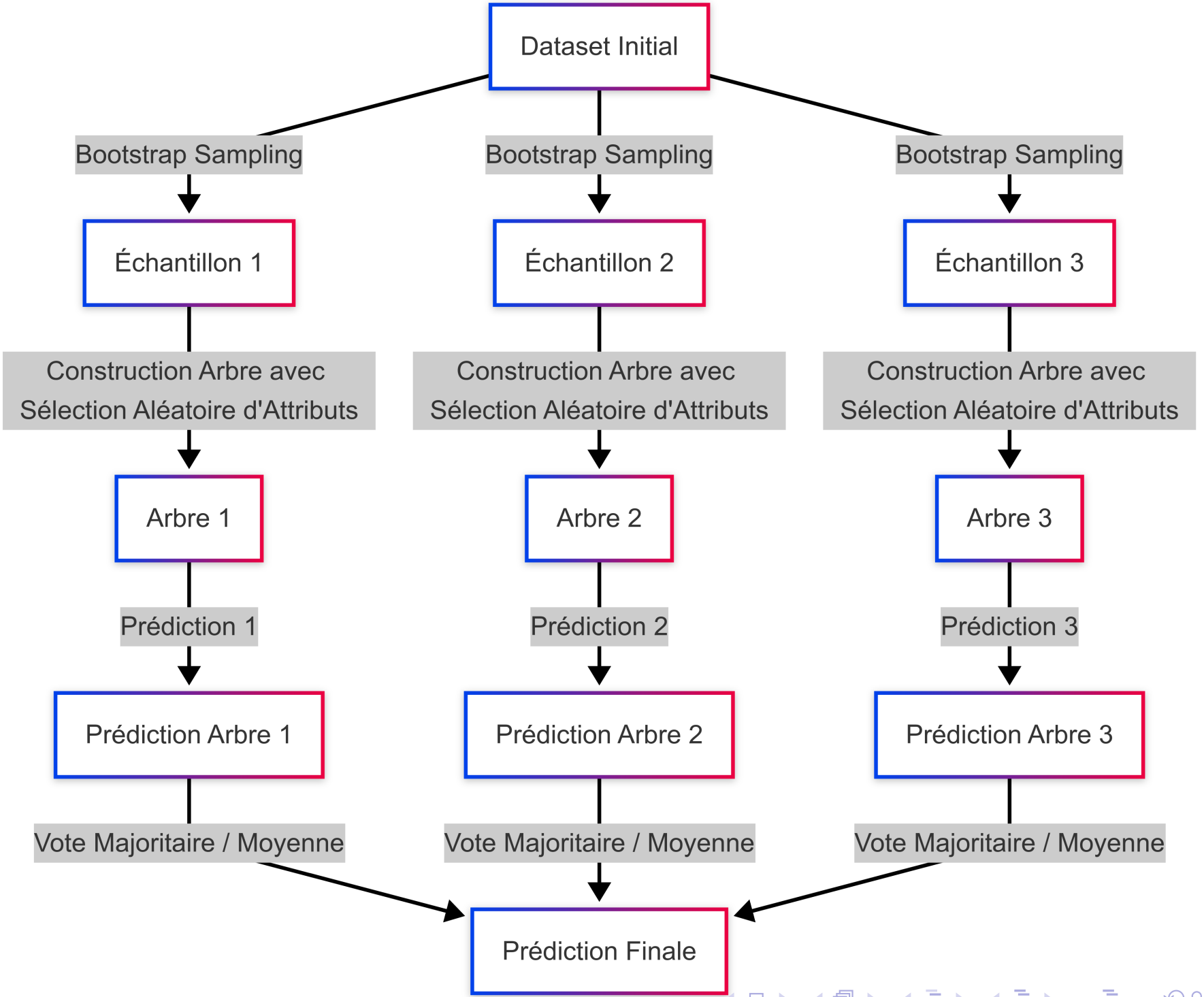
Les **corrélations** entre arbres diminuent (ils ne sont pas "clones"), améliorant la robustesse du vote final.

Exemple : Prédiction (Étape 3)

Pour un **nouvel exemple** ($Taille = 1.65, Poids = 72$) :

- **Arbre 1** prédit "Oui".
- **Arbre 2** prédit "Non".
- **Arbre 3** prédit "Oui".

Vote majoritaire = Oui (2votes sur 3)


Dataset Initial
Bootstrap Sampling
Bootstrap Sampling
Bootstrap Sampling
Échantillon 1
Échantillon 2
Échantillon 3
Construction Arbre avec Sélection Aléatoire d'Attributs
Construction Arbre avec Sélection Aléatoire d'Attributs
Construction Arbre avec Sélection Aléatoire d'Attributs
Arbre 1
Arbre 2
Arbre 3
Prédiction 1
Prédiction 2
Prédiction 3
Prédiction Arbre 1
Prédiction Arbre 2
Prédiction Arbre 3
Vote Majoritaire / Moyenne
Vote Majoritaire / Moyenne
Vote Majoritaire / Moyenne
Prédiction Finale

Conclusion sur les Random Forests

- Une **Random Forest** est une **forêt** d'arbres de décision entraînés sur des **échantillons bootstrap**, avec une **sélection aléatoire d'attributs**.
- Chaque arbre est "instable" mais le **vote** ou la **moyenne** confère une grande **stabilité** à la forêt.
- Réduction de l'overfitting, bonne performance pratique.

Exercice : Construire Trois Arbres d'une Forêt Aléatoire

Objectif : Comprendre le fonctionnement des forêts aléatoires en construisant **trois arbres** à partir de sous-échantillons tirés par Bootstrap.

Données initiales :

ID	Revenu mensuel	Historique de crédit	Décision
1	Faible	Mauvais	Refusée
2	Moyen	Bon	Approuvée
3	Élevé	Mauvais	Refusée
4	Faible	Bon	Approuvée
5	Élevé	Bon	Approuvée
6	Moyen	Mauvais	Refusée
7	Faible	Mauvais	Refusée
8	Élevé	Bon	Approuvée

Table – Données simplifiées pour une demande de prêt (8 exemples, 2 attributs).

Trois Échantillons Bootstrap

Échantillon 1 = {1, 2, 3, 4, 4, 7, 8, 8}

ID	Revenu mensuel	Historique de crédit	Décision
1	Faible	Mauvais	Refusée
2	Moyen	Bon	Approuvée
3	Élevé	Mauvais	Refusée
4	Faible	Bon	Approuvée
4	Faible	Bon	Approuvée
7	Faible	Mauvais	Refusée
8	Élevé	Bon	Approuvée
8	Élevé	Bon	Approuvée

Table – Échantillon 1 (tirage avec remise)

Échantillon 2 = {1, 2, 2, 5, 5, 5, 6, 8}

ID	Revenu mensuel	Historique de crédit	Décision
1	Faible	Mauvais	Refusée
2	Moyen	Bon	Approuvée
2	Moyen	Bon	Approuvée
5	Élevé	Bon	Approuvée
5	Élevé	Bon	Approuvée
5	Élevé	Bon	Approuvée
6	Moyen	Mauvais	Refusée
8	Élevé	Bon	Approuvée

Table – Échantillon 2 (tirage avec remise)

Échantillon 3 = {1, 3, 3, 4, 6, 6, 7, 8}

ID	Revenu mensuel	Historique de crédit	Décision
1	Faible	Mauvais	Refusée
3	Élevé	Mauvais	Refusée
3	Élevé	Mauvais	Refusée
4	Faible	Bon	Approuvée
6	Moyen	Mauvais	Refusée
6	Moyen	Mauvais	Refusée
7	Faible	Mauvais	Refusée
8	Élevé	Bon	Approuvée

Table – Échantillon 3 (tirage avec remise)

Étape 2 : Construction de 3 Arbres de Décision

1. **Indice de Gini initial** : Calculez le Gini de chacun des trois échantillons 1, 2, 3 (présentés auparavant).
2. **Sélection Aléatoire d'Attributs** :
   - À chaque nœud, choisissez **au hasard** l'un des deux attributs (*Revenu mensuel* ou *Historique de crédit*).
   - Calculez le **Gain de Gini** et effectuez la division si elle **réduit** l'impureté.
3. **Compléter les trois arbres** : Continuez les divisions jusqu'à obtenir des feuilles pures (classe "Approuvée" ou "Refusée") ou presque pures.

Rappel

La Random Forest utilise **Bagging** (tirages avec remise) et la **sélection aléatoire d'attributs** pour construire des arbres variés (réduisant le sur-apprentissage).

Étape 3 : Décision Majoritaire

Nouveau point à prédire :

(**Revenu mensuel** = Moyen, **Historique de crédit** = Bon)

1. **Arbre 1, Arbre 2, Arbre 3** : Déterminez la classe prédite **(Approuvée ou Refusée)** par chacun des trois arbres.
2. **Vote majoritaire** :

   Décision finale = *majorité*(votes "Approuvée", votes "Refusée").

3. Comparez la décision finale à ce que donnerait **un seul arbre** pris isolément.

Note

La Random Forest combine les prédictions pour **réduire l'instabilité** d'un arbre unique et améliorer la **robustesse** du modèle.