

Arbres de régression et méthodes d'ensembles : erreur quadratique moyenne et stabilité

Komi Ayi

2023-04-25

1. Introduction

L'arbre de régression, le bagging et les forêts aléatoires sont des techniques d'apprentissages basés sur des arbres de décision qui divisent un ensemble de données en sous-ensembles pour prédire des résultats, mais différent dans la manière dont ils sont construits. Un critère important pour évaluer la performance des techniques d'apprentissage est la stabilité. Plus une technique est stable, plus elle est capable de maintenir des performances élevées sur de nouveaux ensembles de données.

Pour expliquer une variable cible en fonction d'un nombre très important de variables explicatives, on peut utiliser les arbres de décisions (Breiman et al. 1984) à cause de leur simplicité d'interprétation et leur qualité de précision relative. Néanmoins, ils sont très sensibles aux changements mineurs dans les données, ce qui peut entraîner une instabilité dans les résultats de prédiction.

Pour améliorer la stabilité de l'arbre de régression, le bagging (Breiman 1996) a été introduit. Il crée plusieurs échantillons aléatoires à partir de l'ensemble de données d'apprentissages et entraîne un arbre de décision sur chaque échantillon. Les prévisions de chaque arbre sont ensuite moyennées pour obtenir une prédiction finale plus stable et robuste.

Pour augmenter la diversité entre les arbres individuels, ce qui conduit à des prévisions plus précises et plus stables, les forêts aléatoires (Breiman 2001) furent introduites. Ils sont une extension du bagging en introduisant un aléa supplémentaire. Au lieu de diviser les données d'apprentissage de manière aléatoire, les forêts aléatoires utilisent également des sous-ensembles aléatoires de caractéristiques pour construire chaque arbre de décision.

Dans ce travail, nous étudions la stabilité et comparons l'erreur quadratique moyenne des trois méthodes. Dans la première section, nous présentons les idées derrière chaque méthode et leurs algorithmes. Une étude de simulation et une application sur un jeu de donnée réelle sont menées pour illustrer la stabilité.

2. Méthodes

On suppose que l'on est dans un contexte de régression. Les méthodes détaillées s'étendent à la classification. Soit (\mathbf{x}, y) un vecteur aléatoire où $\mathbf{x} \in \mathbb{R}^p$ et $y \in \mathbb{R}$. On note $S = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ l'observation de n -échantillon d'une loi conjointe \mathcal{D} sur $\mathcal{X} \times \mathcal{Y}$, inconnue. Soit $f(\mathbf{x}) = \mathbb{E}[Y|\mathbf{x}]$ la fonction de régression. L'erreur de généralisation de f est définie par (Shalev-Shwartz and Ben-David 2014):

$$L_{\mathcal{D}}(f) := \mathbb{E}_{\mathcal{D}}[(y - f(\mathbf{X}))^2]. \quad (1)$$

En l'absence de toute information ou hypothèse sur la loi \mathcal{D} , on remplace \mathcal{D} par S , et on minimise le risque empirique :

$$L_S(f) = \frac{1}{n} \sum_{i=1}^n (y_i - f(\mathbf{X}_i))^2 \quad (2)$$

La minimisation du risque empirique permet d'avoir un estimateur qu'on note h de f :

$$h_S \in \operatorname{argmin}_{f \in \mathcal{F}} L_S(f)$$

Le choix d'un modèle f adéquat relève des méthodes de sélection de modèles (Vapnik 1999).

2.1 Arbres de décision

Le principe de base des arbres de décision est de diviser un ensemble de données en sous-ensembles en fonction des caractéristiques des données. Chaque subdivision est basée sur une caractéristique qui permet de séparer au mieux les données en groupes homogènes. Le processus se répète jusqu'à ce que les sous-ensembles ne puissent plus être subdivisés ou jusqu'à ce qu'un critère d'arrêt prédéfini soit atteint.

En régression, le but serait de découper l'espace des variables explicatives en région R_1, \dots, R_m qui minimisent la fonction de perte

$$\sum_{j=1}^m \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2, \quad (3)$$

où \hat{y}_{R_j} est la réponse moyenne sur les observations d'apprentissages qui tombent dans la région R_j . Malheureusement, il est impossible de traverser l'ensemble des partitionnements de l'espace des prédictors en m boîtes. Pour cela, on construit l'arbre binaire de façon récursive. L'algorithme récursif de l'arbre binaire se présente comme suit:

- *Initialisation Noeud racine.* On place l'ensemble de l'échantillon d'estimation à la racine de l'arbre.
- *Récurrence sur chaque noeud.* On partitionne chaque noeud en deux classes :

$$\mathcal{R}_1(j, s) = \{X : X_j \leq s\}; \quad \mathcal{R}_2(j, s) = \{X : X_j > s\}.$$

On construit la meilleur division en choisissant la coupe (j, s) qui minimise

$$\sum_{i \in \mathcal{R}_1(j; s)} (y_i - \hat{y}_1)^2 + \sum_{i \in \mathcal{R}_2(j; s)} (y_i - \hat{y}_2)^2 \quad (4)$$

la variance inter-groupe, où \hat{y}_l est la réponse moyenne des données d'apprentissage qui tombent dans la région $\mathcal{R}_l(j, s)$, $l = 1, 2$.

- *Règle d'arrêt.* Séparer les données et recommencer sur chaque sous-ensemble. On arrête si :
 - la variance inter-groupe ne diminue plus,
 - le nombre d'observations à un noeud est inférieur à une valeur prédéfinie,
 - la profondeur de l'arbre est supérieure à une valeur prédéfinie, etc...

Il est important de noter qu'un arbre parfait n'existe pas en général. Même si l'on arrive à construire un arbre, il faut tout de même éviter le surapprentissage, c'est-à-dire une capacité à décrire parfaitement bien le jeu de données d'apprentissage tout en ayant de faibles capacités de prédiction pour d'autres jeux de données. Ceci peut se traduire par un arbre trop profond, c-à-dire beaucoup de noeuds terminaux qui entraîne un petit biais, mais une grande variance. Également un arbre trop peu précis entraîne une grande erreur de prédiction. Donc pour cela, il faudra trouver un arbre optimal faisant un compromis entre le sur-aprentissage et un arbre trop peu précis, d'où l'*émondage*.

Soit T un arbre. La méthode d'émondage consiste à introduire un paramètre α (paramètre de complexité, CP) auquel correspond un sous-arbre de T_α qui minimise

$$\sum_{l=1}^{|T|} \sum_{i \in R_l} (y_i - \hat{y}_{R_l})^2 + \alpha |T|, \quad (5)$$

où $|T|$ est le nombre de feuilles de l'arbre, R_l est la région correspondant au l -ième feuille, \hat{y}_{R_l} est la valeur prédite au l -ième feuille (moyenne des observations de R_l). De l'expression, on peut remarquer que si $\alpha = 0$, l'arbre T est l'arbre maximal et si α croît ($\alpha > 0$), les branches sont progressivement émondées formant une suite de sous-arbres emboîtés.

Bien qu'étant faciles à comprendre, à interpréter et à être adaptés à une variété de problèmes, ils peuvent être sensibles aux variations mineures des données, ce qui peut entraîner une instabilité des modèles. Pour corriger cette instabilité, les méthodes d'ensembles furent développées.

2.2 Méthodes d'ensembles

Parfois, le même algorithme appliqué à un jeu de données d'apprentissage légèrement différent produira un modèle très différent (prédiction instable). Ce qui se passe, cependant, c'est que si différentes données d'entraînement sont collectées à partir du même échantillon, leur moyenne est beaucoup plus similaire que les valeurs individuelles d'un échantillon. Par conséquent, la moyenne des valeurs prédites par différents modèles est souvent plus précise et stable que la valeur prédite par un seul modèle. Dans cette section, nous présentons deux méthodes de configuration : **bagging** pour la synthèse bootstrap (Breiman 1996) et les forêts aléatoires (**forêt aléatoire**) de Breiman (2001) donnent une encapsulation améliorée. Nous mesurons la stabilité de l'arbre de régression dans les méthodes d'ensembles en utilisant l'erreur quadratique moyenne (MSE) entre les prédictions des différents arbres de décision.

2.2.1 Bagging

Le bagging est une méthode d'agrégation simple dans laquelle les prédicteurs individuels sont de même type et sont construits sur différents échantillons bootstrap à partir des données d'apprentissage. Il combine les techniques *bootstrapping* et *aggregation*.

Le principe fondamental du bootstrap est de substituer, à la distribution de probabilité inconnue \mathcal{D} , dont est issu l'échantillon d'apprentissage, la distribution empirique S qui donne un poids $\frac{1}{n}$ à chaque réalisation. Ainsi on obtient un échantillon de taille n dit échantillon bootstrap selon la distribution empirique S par n tirages aléatoires avec remise parmi les n observations initiales. Ainsi, on construit un grand nombre d'échantillons bootstrap B sur lesquels calculer les estimateurs $h_{S^j}^j, j = 1, \dots, B$. L'estimateur bagging est obtenue par la moyenne des fonctions de régressions $h_{S^j}, j = 1, \dots, B$:

$$h_{bag}(\mathbf{x}) = \frac{1}{B} \sum_{j=1}^B h_{S^j}(\mathbf{x}), \quad (6)$$

et l'estimation bootstrap de l'erreur moyenne de prévision est approchée par simulation par:

$$\frac{1}{nB} \sum_{j=1}^B \sum_{i=1}^n (y_i - h_{S^j}(\mathbf{x}_i))^2 \quad (7)$$

Proposition 0.1 Si $h_{S^1}(\mathbf{x}), \dots, h_{S^B}(\mathbf{x})$ sont i.i.d alors,

$$\mathbb{E}(h_{bag}(\mathbf{x})) = \mathbb{E}(h_{S^1}(\mathbf{x})) \text{ et } \text{Var}(h_{bag}(\mathbf{x})) = \frac{1}{B} \text{Var}(h_{S^1}(\mathbf{x})). \quad (8)$$

Preuve. D'après l'Équation (6),

$$\begin{aligned} \mathbb{E}(h_{bag}(\mathbf{x})) &= \frac{1}{B} \sum_{j=1}^B \mathbb{E}(h_{S^j}(\mathbf{x})) \quad \text{par linéarité de l'espérance} \\ &= \frac{1}{B} \sum_{j=1}^B \mathbb{E}(h_{S^1}(\mathbf{x})) \quad \text{car } h_{S^1}(\mathbf{x}), \dots, h_{S^B}(\mathbf{x}) \text{ sont i.i.d} \\ \mathbb{E}(h_{bag}(\mathbf{x})) &= \mathbb{E}(h_{S^1}(\mathbf{x})), \end{aligned}$$

et

$$\begin{aligned}
\text{Var}(h_{bag}(\mathbf{x})) &= \frac{1}{B^2} \text{Var}\left(\sum_{j=1}^B h_{S^j}(\mathbf{x})\right) \\
&= \frac{1}{B^2} \sum_{j=1}^B \text{Var}(h_{S^j}(\mathbf{x})) \quad \text{car } h_{S^1}(\mathbf{x}), \dots, h_{S^B}(\mathbf{x}) \text{ sont i.i.d} \\
&= \frac{1}{B^2} \sum_{j=1}^B \text{Var}(h_{S^1}(\mathbf{x})) \\
\text{Var}(h_{bag}(\mathbf{x})) &= \frac{1}{B} \text{Var}(h_{S^1}(\mathbf{x})).
\end{aligned}$$

□

Le fait de considérer des échantillons bootstrap introduit un aléa supplémentaire dans l'estimateur (Breiman 1996). Pour cela, on note $\theta_j = \theta_j(S)$ l'échantillon bootstrap de l'étape j et $h_{S^j}(\cdot, \theta_j)$ l'estimateur construit à l'étape j . Alors, l'estimateur bagging est définie par

$$h_{bag}(\mathbf{x}) = \frac{1}{B} \sum_{j=1}^B h_{S^j}(\mathbf{x}, \theta_j). \quad (9)$$

Puisque, conditionnellement à S , les variables $\theta_1, \dots, \theta_B$ sont i.i.d, alors d'après la loi des grands nombres on a :

$$\lim_{B \rightarrow \infty} h_{bag}(\mathbf{x}) = \lim_{B \rightarrow \infty} \frac{1}{B} \sum_{j=1}^B h_{S^j}(\mathbf{x}, \theta_j) = \mathbb{E}_\theta[h_S(\mathbf{x}, \theta)|S] \quad ps. \quad (10)$$

On déduit donc de l'Équation (10) que prendre B trop grand ne va pas surajuster l'échantillon, donc recommandé de prendre B le plus grand possible.

Proposition 0.2 *Soit, $\rho(\mathbf{x}) = \text{cor}(h_{S^j}(\mathbf{x}, \theta_j), h_{S^{j'}}(\mathbf{x}, \theta_{j'}))$ le coefficient de corrélation entre deux estimateurs calculés sur deux échantillons bootstrap. On a:*

$$\text{Var}(h_{bag}(\mathbf{x})) = \rho(\mathbf{x}) \text{Var}(h_{S^1}(\mathbf{x}, \theta_1)) + \frac{1 - \rho(\mathbf{x})}{B} \text{Var}(h_{S^1}(\mathbf{x}, \theta_1)). \quad (11)$$

Preuve. Puisque les θ_j sont i.i.d, les estimateurs $h_{S^1}(\mathbf{x}, \theta_1), \dots, h_{S^B}(\mathbf{x}, \theta_B)$ sont identiquement distribués et donc

$$\begin{aligned}
\text{Var}(h_{bag}(\mathbf{x})) &= \frac{1}{B^2} \left[\sum_{j=1}^B \text{Var}(h_{S^j}(\mathbf{x})) + \sum_{1 \leq j \neq j' \leq B} \text{cov}(h_{S^j}(\mathbf{x}, \theta_j), h_{S^{j'}}(\mathbf{x}, \theta_{j'})) \right] \\
&= \frac{1}{B^2} \left[\sum_{j=1}^B \text{Var}(h_{S^1}(\mathbf{x})) + \sum_{1 \leq j \neq j' \leq B} \text{cov}(h_{S^1}(\mathbf{x}, \theta_1), h_{S^2}(\mathbf{x}, \theta_2)) \right] \\
&= \frac{1}{B^2} \left[B \text{Var}(h_{S^1}(\mathbf{x})) + \sum_{1 \leq j \neq j' \leq B} \rho(\mathbf{x}) \text{Var}(h_{S^1}(\mathbf{x})) \right] \\
&= \frac{1}{B^2} \left[B \text{Var}(h_{S^1}(\mathbf{x})) + B(B-1) \rho(\mathbf{x}) \text{Var}(h_{S^1}(\mathbf{x})) \right] \\
&= \frac{1}{B} \text{Var}(h_{S^1}(\mathbf{x})) + \frac{B-1}{B} \rho(\mathbf{x}) \text{Var}(h_{S^1}(\mathbf{x})).
\end{aligned}$$

□

À partir des théorèmes 0.1 et 0.2, nous pouvons conclure que la moyenne du prédicteur obtenu avec un arbre est égale à celle obtenue par le bagging, mais la variance obtenue par le bagging est plus petite et réduit ainsi l'erreur de prédiction. De plus, la variance de $h_{bag}(\mathbf{x})$ diminue lorsque les estimateurs agrégés $h_{S^j}(\mathbf{x})$ ne sont pas corrélés. Par conséquent, la synthèse des estimateurs sensibles aux perturbations de l'échantillon à l'aide du bootstrap apportera une amélioration.

Algorithm 1 Bagging

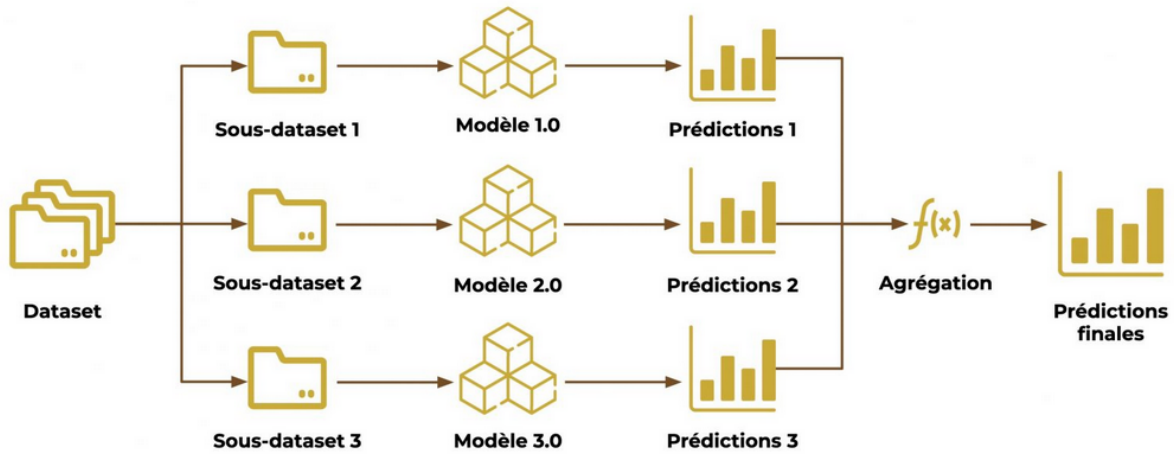
Entrées:

- un régresseur (arbre de régression, par exemple)
- un échantillon S
- le nombre B de réplication bootstrap

Pour $j = 1, \dots, B$:

- Tirer un échantillon bootstrap de S^j de S
- Apprentissage du modèle sur cet échantillon bootstrap: h_{S^j}

Sortie: Estimateur $h_{bag}(\mathbf{x}) = \frac{1}{B} \sum_{j=1}^B h_{S^j}(\mathbf{x})$

Figure 1: Bagging¹

2.2.2 Random Forest

L'algorithme de Random Forest pour l'arbre de régression est une extension de l'algorithme de l'arbre de décision pour la régression. Random Forest utilise une approche d'ensemble pour construire plusieurs arbres de décision et combiner leurs prédictions pour obtenir une prédiction plus robuste et précise. Au lieu de construire des arbres en utilisant toutes les données (p valeurs explicatives), Random Forest sélectionne aléatoirement un sous-ensemble de données (m variables explicatives) pour chaque arbre.

L'intuition derrière cette méthode est que le bagging étant plus efficace lorsque la corrélation entre les prédicteurs est faible, donc la sélection d'un sous-ensemble peut isoler les variables fortement corrélées et réduire la variance. En effet, l'utilisation d'un ensemble de variables explicatives aléatoires pour chaque arbre de décision et la construction de plusieurs arbres de décision avec différentes combinaisons de variables explicatives vont permettre à réduire le risque de surajustement et augmenter la précision des prédictions.

¹source

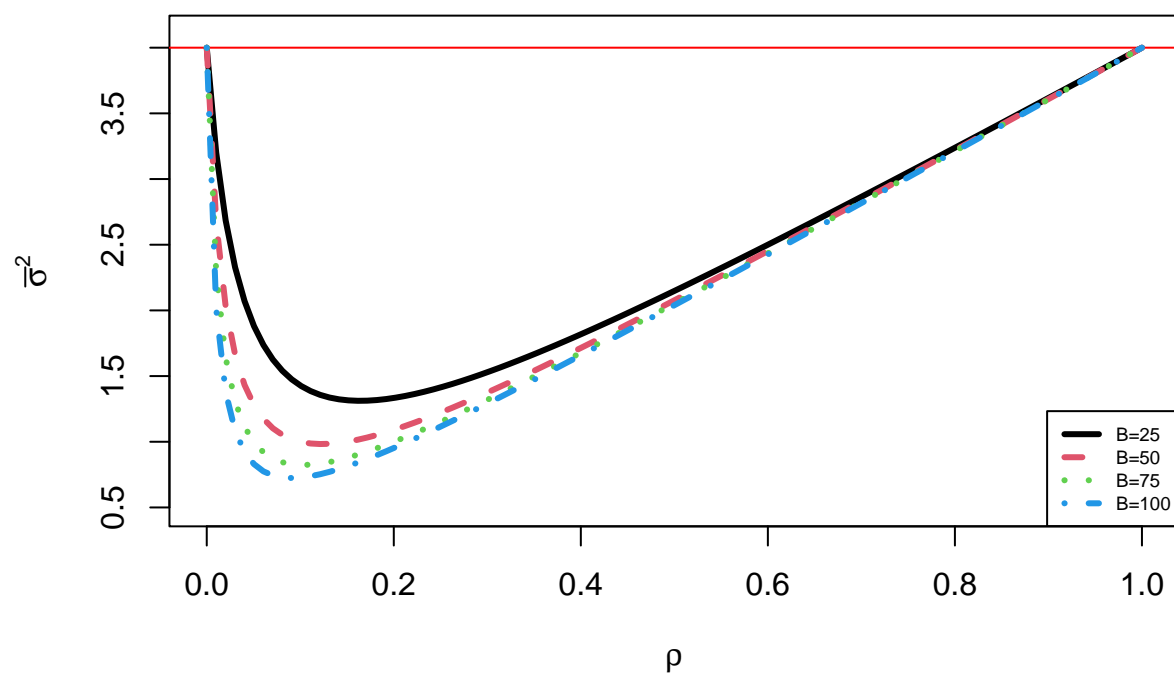


Figure 2: Variance du prédicteur baggé en fonction de la corrélation.

Ci-dessous l'algorithme (algorithme 2) d'un Random forest. Particulièrement utile lorsque les données contiennent des valeurs aberrantes, il est robuste aux valeurs manquantes et peut gérer des jeux de données de grande dimension. Cependant, si le nombre d'arbres de décision est trop grand, il est coûteux en temps de calcul et en mémoire.

Algorithme 2 Random Forest (arbre de régression)

Entrées:

- un échantillon S
- le nombre B de réplication bootstrap
- m le nombre de variables candidates pour découper un noeud.

Pour $j = 1, \dots, B$:

- Tirer un échantillon bootstrap S^j de S
- Apprentissage du modèle sur cet échantillon bootstrap avec randomisation des variables: la coupure optimale est effectuée en minimisant le critère de variabilité (variance) sur un sous-ensemble de variables choisies (m) parmi les p possibles: $h_{S^j}(\cdot, \theta_j)$

Sortie: Estimateur $h_{rf}(\mathbf{x}) = \frac{1}{B} \sum_{j=1}^B h_{S^j}(\mathbf{x}, \theta_j)$

Comment choisir m ? Breiman (2001) dit que pour $m = 1$, il y a un choix complètement aléatoire de variables, mais pas dans le seuil de la coupe. Par conséquent, à mesure que m diminue, la corrélation entre les arbres diminue, de même que la variance de l'estimateur h_{rf} . La sélection aléatoire, en revanche, reflète la mauvaise qualité d'ajustement des échantillons d'apprentissage, augmentant ainsi le biais pour chaque arbre et h_{rf} . Augmenter m produit le résultat opposé. La sélection de m est donc liée à la sélection des paramètres dans l'arbre. En fait Breiman (2001) propose $m \approx \log(p)$. Dans notre travail, nous allons sélectionner le paramètre m par validation croisée (k-fold cross-validation).

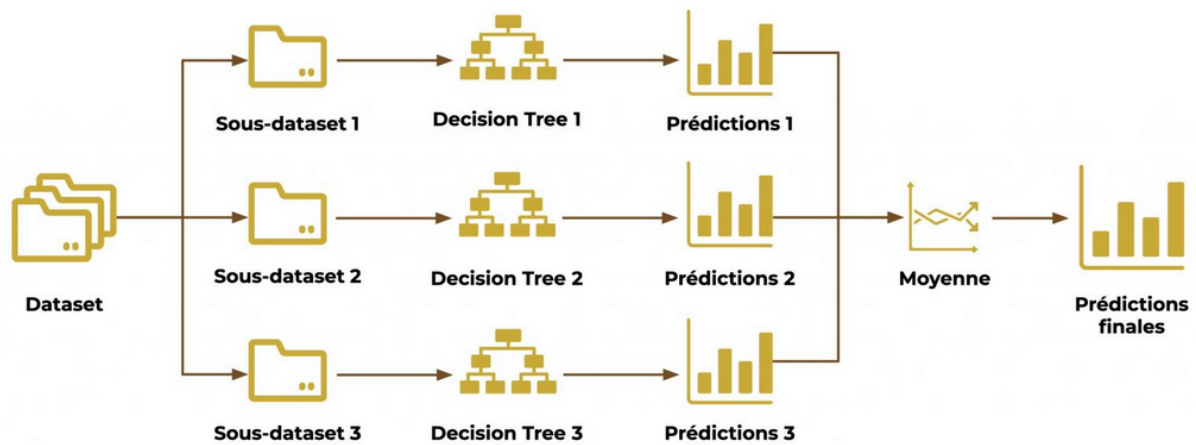


Figure 3: Random Forest²

3. Résultats

L'objectif de l'étude de simulation est d'étudier la stabilité de l'arbre de régression et de comparer les erreurs quadratiques moyennes des trois méthodes.

²source

3.1 Études de simulation

Nous utilisons le jeu de données de Friedman (1991) qui contient 10 variables explicatives.

On génère le terme d'erreur $\epsilon \sim \mathcal{N}(0, 1)$, les variables $x_p \sim \mathcal{U}(0, 1)$, $p = 1, \dots, 5$. Ensuite, on génère la réponse y en fonctions des variables explicatives et de l'erreur générés précédemment par:

$$y = 10 \sin(x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5 + \epsilon.$$

La taille des échantillons est de $n = 1200$ et le nombre de réplifications sont de $J = 1000$. On divise en deux les observations : 200 pour l'entraînement et 1000 pour le test. Le nombre de réplifications bootstrap est de $B = 100$. Pour chaque j ,

- Arbre: on détermine les paramètres optimaux `minsplit`, `maxdepth` (profondeur), `cp` (paramètre de complexité) pour construire un arbre optimal avec les données d'apprentissage et ensuite l'erreur quadratique moyenne à partir des données de test. Les paramètres de `minsplit` et `maxdepth` sont présenté dans la Table 1.
- Bagging: pour $b = 1, \dots, B$, on apprend le modèle sur les données d'apprentissage bootstrappé, moyenné pour obtenir le prédicteur bagging, et ensuite calculer l'erreur quadratique moyenne.
- Random Forest: on détermine le m optimal par validation croisée (m qui minimise le MSE), avec lequel on construit le modèle et l'erreur quadratique moyenne.

En moyennant sur $j = 1, \dots, J$, on obtient les estimations Monte-Carlo de MSE et de $\text{Var}(\text{MSE})$.

La stabilité de l'arbre de régression dans les modèles d'ensembles peut être mesurée en calculant l'erreur quadratique moyenne (MSE) entre les prédictions des différents arbres de décision. Pour cela, on crée un vecteur $Z' = (x_1, x_2, x_3, x_4, x_5, y)$. Pour un i , on remplace la ligne i de données d'apprentissage par Z' . On obtient ainsi une nouvelle base d'apprentissage. On utilise une méthode sur cette base pour apprendre le modèle et ainsi calculer l'erreur quadratique moyenne.

<code>minsplit</code>	3	9	24
<code>maxdepth</code>	1	10	15

Table 1: Paramètres de la profondeur et du nombre d'observations (Arbre)

3.2 Résultats de la simulation

La Table 2 présente, la valeur estimée et la variance empirique de l'erreur quadratique moyenne et le nombre de variables explicatives utilisées par l'algorithme de Random Forest dans la Table 3. Les résultats de l'étude de stabilité dans la Table 4.

L'erreur quadratique moyenne obtenue de l'algorithme Random Forest est 3.674, celle de bagging est 5.154 et celui obtenu de l'arbre de régression optimale est approximativement deux (2) fois supérieur à celui de Random Forest: 7.785. Ce résultat confirme la Proposition 0.2 et l'intuition derrière la construction d'un Random Forest. De plus, on remarque que la variabilité de MSE (0.107) estimé par Monte-Carlo est faible avec Random Forest comparé à l'arbre de régression (0.407) et au Bagging (0.192). Ce qui semble confirmer la robustesse de l'algorithme Random Forest. La construction de l'arbre de régression optimal dans Random Forest est effectuée la plupart du temps avec trois variables explicatives ($\approx 62\%$).

La mesure de stabilité effectuée indique qu'un petit changement dans l'échantillon d'apprentissage en remplaçant une ligne par une nouvelle observation, conduite à une mesure de 225.311 pour Random Forest, de 1021.615 pour le Bagging et de 4422.368 pour l'arbre de régression (Table 4). Donc Random Forest est plus stable que Bagging qui est elle plus stable que l'arbre de régression. On remarque également que la stabilité de l'arbre de régression dans Random Forest n'est pas assez élevée. Ce qui semble garantir un modèle robuste et fiable.

	MSE	Var(MSE)
Arbre	7.785566	0.4073831
Bagging	5.153576	0.1919885
Random Forest	3.673866	0.1072489

Table 2: Erreur quadratique moyenne des différents modèles.

m optimal			
2	3	4	5
162	617	166	55

Table 3: Nombre de variables explicatives choisies par validation croisée (Random Forest)

methode	$\sum_{i=1}^n \{l(h_{S^i}, z') - l(h_S, z_i)\}$
Arbre	4422.368
Bagging	1021.615
Random Forest	225.3114

Table 4: Stabilité des différents modèles.

3.2 Applications à des données réelles

Nous avons appliqué notre méthode de stabilité sur des données réelles. Le jeu de données recense les coûts de construction (variable réponse), les variables financières et les indices économiques (27 variables explicatives continues) correspondant aux appartements résidentiels à Téhéran, en Iran (Rafiei 2018). Avec cette base, on construit un arbre de régression simple et dans les méthodes d'ensembles. Nous calculons et comparons le MSE obtenu de ces algorithmes. Une étude de stabilité est effectuée. L'échantillon $z' = (x_1, \dots, x_{27}, y)$ est généré comme suit: $\epsilon \sim \mathcal{N}(0, 1)$, $x_p \sim \mathcal{U}(0, 1)$, $p = 1, \dots, 27$ et la réponse $y = 100 \times \sum_{i=1}^{27} x_i + \epsilon$. Les résultats de stabilité et de l'erreur quadratique moyenne sont présentés respectivement dans les Table 6 et Table 5. L'algorithme Random Forest améliore la stabilité de l'arbre de régression.

	MSE
Arbre	2878.508
Bagging	2586.330
Random Forest	1733.553

Table 5: Comparaison des MSE.

methode	$\sum_{i=1}^n \{l(h_{S^i}, z') - l(h_S, z_i)\}$
Arbre	54248767
Bagging	7610564
Random Forest	4882612

Table 6: Stabilité.

Conclusion

En conclusion, la stabilité est un critère crucial pour évaluer les performances des techniques d'apprentissage statistique. L'arbre de régression peut être instable et sensible aux variations mineures des données d'entrée, tandis que le bagging et les forêts aléatoires ont été développés pour améliorer la stabilité et la précision des prévisions.

Le bagging améliore la stabilité en créant plusieurs échantillons aléatoires de l'ensemble de données d'origine, tandis que Random Forest introduit un aléa supplémentaire en utilisant des sous-ensembles aléatoires de caractéristiques pour construire chaque arbre de décision.

Cependant, le choix de la technique d'apprentissage peut dépendre des exigences spécifiques de chaque problème de régression, et il est important de considérer les avantages et les inconvénients de chaque méthode avant de prendre une décision.

References

- Breiman, Leo. 1996. "Bagging Predictors." *Machine Learning* 24: 123–40.
- . 2001. "Random Forests." *Machine Learning* 45: 5–32.
- Breiman, Leo, Jerome Friedman, Richard Olshen, and Charles Stone. 1984. "Cart." *Classification and Regression Trees*.
- Friedman, Jerome H. 1991. "Multivariate Adaptive Regression Splines." *The Annals of Statistics* 19 (1): 1–67.
- Rafiei, Mohammad. 2018. "Residential Building Data Set." UCI Machine Learning Repository.
- Shalev-Shwartz, Shai, and Shai Ben-David. 2014. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge university press.
- Vapnik, Vladimir. 1999. *The Nature of Statistical Learning Theory*. Springer science & business media.