Présentation 3

Constance Bau

2024-02-13

Construction d'une forêt aléatoire

- Au départ on a un échantillon de données (appelé échantillon d'entraînement initial) de taille n $D = \{(x_1, y_1), ...(x_n, y_n)\}$ ou les x_i sont des vecteurs de tailles p (i.e on a p covariables d'entrée) et les y_i sont des scalaires.
- On choisit ensuite le nombre n_{arbres} d'arbres de régression que l'on souhaite construire et on répète alors n_{arbres} fois la construction d'un arbre présentée ci-dessous.

Construction d'un arbre :

1) Pour le j-ième arbre on prend un échantillon de boostrap D_j de taille n dans l'échantillon d'entraînement D, c'est-à-dire que l'on tire aléatoirement, uniformément et avec remise n vecteurs x_i dans D. Les vecteur x_i qui n'ont pas été tiré (il y en a sûrement grâce au tirage avec remise) sont gardés de côté et constituent l'échantillon OOB (Out Of Bag) de cet arbre.

Construction d'une forêt aléatoire 2

2) Ensuite, on construit l'arbre de régression en utlisant l'échantillon D_j comme données d'entraînement. Pour cela toutes les données sont d'abord placées dans le noeud initiale puis, à chaque noeud (et ce jusqu'à ce que le critère d'arrêt soit vérifié), on sélectionne aléatoirement et sans remise m (avec m<p) charactéristiques du vecteur x (m composantes du vecteur x). On cherche ensuite la meilleure division (i.e couple (j,s) vu dans abres de régression) parmi les m covariables tirées et non parmi toutes. On sépare le noeud en deux noeuds enfants selon le couple de division (j,s) précedemment sélectionné.

Estimation de l'erreur généralisée d'une forêt aléatoire - Explication générale

Pour estimer l'erreur généralisée d'une forêt aléatoire on va utiliser les prédictions des échantillons OOB. Pour chaque vecteur x_i de l'ensemble d'entraînement initial D, on calcule sa prédiction par chacun des arbres k dont il n'a pas fait partie de l'ensemble d'entraînement D_k et on moyenne ces prédictions pour avoir sa prédiction "par la forêt" (même si dans ce cas on ne l'a pas fait passer par tous les arbres). Ensuite, pour obtenir l'estimation généralisée de la forêt on fait la moyenne des différences au carré de la prévision de chaque x_i et de sa valeur de sortie présente dans l'échantillon d'entraînement D.

Estimation de l'erreur généralisée d'une forêt aléatoire - Algorithme et formule

Algorithme des prédictions OOB :

Soit D_j le jème échantillon bootstrap et $\hat{h}_j(x)$ la prédiction de x à partir du jème arbre, pour $j = 1, ..., J$. Pour $i = 1$ à n :

1. Soit $J_i = \{j : (x_i, y_i) \notin D_j\}$ et $|J_i|$ le cardinal de J_i (Algorithme 2).
2. Définissons la prédiction hors sac à x_i comme suit :
   -$\hat{f}_{oob}(x_i) = \frac{1}{|J_i|} \sum_{j \in J_i} \hat{h}_j(x_i)$ pour la régression

L'erreur de généralisation est généralement estimée en utilisant l'erreur quadratique moyenne (MSE) hors sac suivante :

$$\text{MSE}_{oob} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{f}_{oob}(x_i))^2$$

Implémentation Random Forest

- implémenté selon les méthodes de construction d'arbres de régression et de random forests présentées précedemment
- la plupart des random forests sont impléménetés selon les méthodes vues précedemment (bagging, critère de division des noeuds qui repose sur une diminution de l'impureté des noeuds mesurée avec la variance estimée de la réponse intra-groupe, erreur de prédiction obtenue à partir des données OOB en tant qu'erreur quadratique moyenne, etc) car c'est la méthode de base décrite par Leo Breiman puis reprise dans de nombreux articles. Les principales différences entre les implémentations reposent sur des choix algorithmes, de stockage des données, d'utilisation de la mémoire, multithreading, recompilation ou non à chaque modification d'un paramètre, etc

Implémentation ranger

- optimisé pour les données de grandes dimensions et le plus rapide (en 2017) pour les données de grandes dimensions
- Paramètres à entrer : nombre d'échantillon n, d'arbres, de charactéristiques, et nombre de charactéristiques testées à chaque division.

Sobol-MDA - introduction

- Sobol-MDA est capable de mesurer l'importance des covariables en estimant un indice qui converge vers l'indice de Sobol-total. Sobol-MDA est donc capable d'estimer la proportion de la variance de la réponse expliquée perdue quand une covariable est enlevée du modèle.

Sobol-MDA - introduction

- On note Θ le vecteur qui contient les indices des vecteurs de l'échantillon initial utilisé pour l'entraînement de l'arbre auquel il est associé
- En rappelant que l'on note $m_n(x, \Theta)$ l'estimation de la valeur de la sortie de x par un arbre entraîner avec un échantillon indiqué par Θ , Sobol-MDA prédit $m^{(-j)}(X^{(-j)}) = E[m(X)|X^{(-j)}]$.

Sobol-MDA - explications

- Pour retirer une variable j du processus de prédiction de l'arbre on procède comme suit. La partition de l'espace de covariables obtenu avec les feuilles terminales de l'arbre d'origine est projeté selon la j-ième direction et les sorties des cellules de cette nouvelle partition projetée sont recalculées avec les données d'entraînement, ce procédé permet de retirer la variable j de l'estimation de l'arbre. Ensuite, il est possible de calculer la précision de l'estimation de la forêt projetée grâce aux échantillons OOB, de soustraire cette précision de la précision initiale et de normaliser la différence obtenue par $V[Y]$ pour obtenir le Sobol-MDA pour X_j.

Sobol-MDA - explications 2 - explication de l'algorithme

Les données d'entraînement et les échantillons OOB sont mis dans l'arbre et envoyés à droite et gauche du noeud si celui fait une division sur la covariable j. A la fin chaque donnée peut donc se retrouver dans plusieurs feuille terminales. Pour chaque donnée OOB, la prédiction associée est donc la moyenne des sorties des données d'entraînement qui sont tombées sur les mêmes feuilles terminales. En d'autres termes, on calcule l'intersection de ces feuilles terminales pour sélectionner les observations d'entraînement appartenant à chaque cellule de cette collection afin d'estimer la prédiction. Cette intersection donne la cellule projetée. Ce mécanisme est équivalent à projeter la partition de l'arbre sur le sous-espace engendré par $X^{(-j)}$

Sobol-MDA - explications - notations mathématiques

On note $A_n(X, \Theta)$ la cellule de la partition de l'arbre d'origine où X tombe et on note $A_n^{(-j)}(X^{(-j)}, \Theta)$ la cellule associé de la partition projetée. On note l'estimation par l'arbre projeté associé $m_n^{(-j)}(X^{(-j)}, \Theta)$ et l'estimation par la forêt projeté associée $m_{M,n}^{(-j,OOB)}(X_i^{(-j)}, \Theta_{(M)})$. Ils sont définis de la manière suivante :

$$m_n^{(-j)}(X^{(-j)}, \Theta) = \frac{\sum_{i=1}^{a_n} Y_i 1_{X_i \in A_n^{(-j)}(X^{(-j)}, \Theta)}}{\sum_{i=1}^{a_n} 1_{X_i \in A_n^{(-j)}(X^{(-j)}, \Theta)}}$$

$$m_{M,n}^{(-j,OOB)}(X_i^{(-j)}, \Theta_{(M)}) = \frac{1}{|\Lambda_{n,i}|} \sum_{l \in \Lambda_{n,i}} m_n^{(-j)}(X_i^{(-j)}, \Theta_l) 1_{|\Lambda_{n,i}|>0}$$

Sobol-MDA indice

L'indice de Sobol-MDA est donné par la différence normalisée de l'erreur carrée (calculée grâce aux OOB) de la forêt projetée et l'erreur carrée (calculée grâce aux OOB) de la forêt initial, Sobol-MDA est donc défini de la manière suivante :

$$\widehat{S-MDA}_{M,n}(X^{(j)}) = \frac{1}{\hat{\sigma}_Y^2}\frac{1}{n}\sum_{i=1}^{n}\{Y_i - m_{M,n}^{(-j,OOB)}(X_i^{(-j)}, \Theta_{(M)})\}^2$$

$$-\{Y_i - m_{M,n}^{(OOB)}(X_i, \Theta_{(M)})\}^2$$

où $\hat{\sigma}_Y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \bar{Y})^2$ est la variance standard estimée de la réponse Y.

Sobol-MDA - idée de la preuve de convergence

Tout d'abord l'indice de Sobol total $S_j^T = \frac{E[Var(G(X)|X^{(-j)})]}{VarG(X)}$, s'écrit, dans le cas du métamodèle, en remplaçant G(X) par l'estimation obtenue par la forêt, lindice de Sobol total s'écrit donc $S_j^T = E[(m(X) - E[m(X)|X^{(-j)}])^2]$. On cherche donc à montrer que

$$\widehat{S - MDA}_{M,n}(X^j) \xrightarrow{p} E[(m(X) - E[m(X)|X^{(-j)}])^2]$$

Pour cela, on majore $E[\widehat{S - MDA}_{M,n}(X^j) - E[(m(X) - E[m(X)|X^{(-j)}])^2]]$ et on montre que, sous certaines hypothèse sur la construction du random-forest, cette majoration tend vers 0 lorsque le nombre d'échantillon n tend vers l'infini.

Sobol-MDA - avantages

- L'algorithme de force brute de Williamson et al.(2021) estime les indices de Sobol totaux avec entrées dépendantes les indices de Sobol totaux en réentraînant une forêt aléatoire sans la covariable dont on souhaite connaître l'importance et en comparant son erreur généralisée avec celle de la forêt initial. Cet algorihtme a une complexité en $O\{Mp^2 nlog^2(n)\}$ qui est quadratique avec les dimension.
- L'algorithme Sobol-MDA estime la même chose mais il a une complexité en $O\{Mnlog^3(n)\}$ ce qui est un grand avantage lorsque l'on travaille avec un grand nombre de varaibles d'entrée.