Bagging et forêts aléatoires

Ghazi Bel Mufti

belmufti@yahoo.com

ESSAI-3 / DATA MINING

Plan

Bagging I

- Le bagging regroupe un ensemble de méthodes introduit par Léo Breiman en 1996.
- Le terme bagging vient de la contraction de Boostrap Aggregating. Nous présentons cette famille de méthodes dans un contexte de régression. Elles s'étendent aisément à la classification supervisée.
- On désigne par (X, Y) un vecteur aléatoire où X prend ses valeurs dans $\mathbb{R}^p$ et Y dans $\mathbb{R}$. On note $D_n = (X_1, Y_1), \ldots, (X_n, Y_n)$ un n-échantillon i.i.d. et de même loi que (X, Y) et $m(x) = E[Y|X = x]$ la fonction de régression.
- Pour $x \in \mathbb{R}^p$, on considère l'erreur quadratique moyenne d'un estimateur $\hat{m}$ et sa décomposition biais-variance :

$$E[(\hat{m}(x) - m(x))^2] = (E[\hat{m}(x)] - m(x))^2 + V(\hat{m}(x)).$$

Bagging II

- Les méthodes bagging sont des méthodes d'agrégation. Elles consistent à agréger un nombre B d'estimateurs $\hat{m}_1, ..., \hat{m}_B$:

$$\hat{m}(x) = \frac{1}{B} \sum_{k=1}^{B} \hat{m}_k(x).$$

- Remarquons que si on suppose les régrésseurs $\hat{m}_1, ..., \hat{m}_B$ i.i.d., on a :

$$E[\hat{m}(x)] = E[\hat{m}_1(x)] \text{ et } V(\hat{m}(x)) = \frac{1}{B} V(\hat{m}_1(x)).$$

Le biais de l'estimateur agrégé est donc le même que celui des $\hat{m}_k$ mais la variance diminue.

- Bien entendu, en pratique il est quasiment impossible de considérer des estimateurs $\hat{m}_k$ indépendants dans la mesure où ils dépendent tous du même échantillon D_n.

Bagging III

- L'approche bagging consiste à tenter d'atténuer la dépendance entre les estimateurs que l'on agrège en les construisant sur des échantillons bootstrap.

Algorithme I

- L'algorithme est simple à implémenter : il suffit de construire B estimateurs sur des échantillons bootstrap et de les agréger.
- Le fait de considérer des échantillons bootstrap introduit un aléa supplémentaire dans l'estimateur.
- Afin de prendre en compte cette nouvelle source d'aléatoire, on note $\theta_k = \theta_k(D_n)$ l'échantillon bootstrap de l'étape k et $\hat{m}(., \theta_k)$ l'estimateur construit à l'étape k.
- On écrira l'estimateur final

$$\hat{m}_B(x) = \frac{1}{B} \sum_{k=1}^{B} \hat{m}(x, \theta_k).$$

Algorithme II

Algorithme 1 Bagging

ENTRÉES :

- x l'observation à prévoir
- un régresseur (arbre CART, 1 plus proche voisin...)
- d_n l'échantillon
- B le nombre d'estimateurs que l'on agrège.

Pour $k = 1, \ldots, B$ **:**

1. Tirer un échantillon bootstrap d_n^k dans d_n
2. Ajuster le régresseur sur cet échantillon bootstrap : $\hat{m}_k$

SORTIES : L'estimateur $\hat{m}(x) = \frac{1}{B} \sum_{k=1}^{B} \hat{m}_k(x)$.

Algorithme III

- **Remarque :** Les tirages bootstrap sont effectués de la même manière et indépendamment les uns des autres. Ainsi, conditionnellement à D_n, les variables $\theta_1, \ldots, \theta_B$ sont i.i.d. et de même loi que θ (qui représentera la loi de la variable de tirage de l'échantillon bootstrap).

Biais et variance I

- Dans cette partie, nous comparons le biais et la variance de l'estimateur agrégé à ceux des estimateurs que l'on agrège. On note :
  - $\hat{m}_B(x) = \frac{1}{B}\sum_{k=1}^{B} \hat{m}(x, \theta_k)$ et $\hat{m}(x) = \lim_{B \to \infty} \hat{m}_B(x)$;
  - $\sigma^2(x) = V(\hat{m}(x, \theta_k))$ la variance des estimateurs que l'on agrège ;
  - $\rho(x) = corr[\hat{m}(x, \theta_1), \hat{m}(x, \theta_2)]$ le coefficient de corrélation entre deux estimateurs que l'on agrège (calculés sur deux échantillons bootstrap).
- La variance $\sigma^2(x)$ et la corrélation $\rho(x)$ sont calculées par rapport aux lois de D_n et de θ.
- On suppose que les estimateurs $\hat{m}(x, \theta_1), ..., \hat{m}(x, \theta_B)$ sont identiquement distribués. Cet hypothèse n'est pas contraignante puisque généralement les θ_i sont i.i.d.

Biais et variance II

- Il est alors facile de voir que le biais de l'estimateur agrégé est le même que le biais des estimateurs que l'on agrège. Par conséquent, agréger ne modifie pas le biais.
- Pour la variance, on a le résultat suivant :
  Propostion. On a

$$V(\hat{m}_B(x)) = \rho(x)\sigma^2(x) + \frac{1-\rho(x)}{B}\sigma^2(x).$$

  Par conséquent :

$$V(\hat{m}(x)) = \rho(x)\sigma^2(x).$$

Biais et variance III

Preuve. On note $T_k = \hat{m}(x, \theta_k)$. Les T_k étant identiquement distribuées, on a

$$\begin{aligned} V(\hat{m}_B(x)) &= V\Big[\frac{1}{B}\sum_{k=1}^{B} T_k\Big] = \frac{1}{B^2}\Big[\sum_{k=1}^{B} V(T_k) + \sum_{1\leq k\neq k'\leq B} cov(T_k, T_{k'})\Big] \\ &= \frac{1}{B}\sigma^2(x) + \frac{1}{B^2}[B^2 - B]\rho(x)\sigma^2(x) \\ &= \rho(x)\sigma^2(x) + \frac{1-\rho(x)}{B}\sigma^2(x). \end{aligned}$$

Discussion I

- Ainsi, si $\rho(x) < 1$, l'estimateur baggé a une variance plus petite que celle des estimateurs que l'on agrège (pour B suffisamment grand).
- A la lueur de ce résultat, on pourrait être tenté de se dire que la bonne stratégie consiste à bagger des estimateurs ayant un biais le plus faible possible. Ceci n'est clairement pas acceptable en effet :
  - Le compromis biais-variance est un problème central en apprentissage supervisé. Idéalement, on veut choisir un modèle qui reflète avec précision les régularités dans les données d'apprentissage, mais qui se généralise aussi aux données tests (données n'ayant pas servi à apprendre le modèle).
  - Malheureusement, il est généralement impossible de faire les deux en même temps.

Discussion II

- Les méthodes d'apprentissage avec une variance élevée peuvent assez bien représenter l'échantillon d'apprentissage, mais il existe un risque de surapprentissage sur des données tests ou bruitées.
- En revanche, les algorithmes avec un biais élevé produisent généralement des modèles plus simples qui n'ont pas tendance au sur-apprentissage, mais peuvent être en sous-apprentissage sur le jeu de données d'apprentissage.

- Ainsi, prendre des estimateurs ayant un biais faible implique que leur variance $\sigma^2(x)$ sera forte.
- Certes, le fait de bagger permettra de réduire dans une certaine mesure cette variance. Cependant, rien ne garantit qu'au final l'estimateur agrégé sera performant (si $\sigma^2(x)$ est très élevée alors $\rho(x)\sigma^2(x)$ sera également élevée !).

En pratique...

- On déduit de la proposition précédente que c'est la corrélation $\rho(x)$ entre les estimateurs que l'on agrège qui quantifie le gain de la procédure d'agrégation : la variance diminuera d'autant plus que les estimateurs que l'on agrège seront décorrélés.
- Le fait de construire les estimateurs sur des échantillons bootstrap va dans ce sens. Il faut néanmoins prendre garde à ce que les estimateurs que l'on agrège soient sensibles à des perturbations de l'échantillon par bootstrap.
- Si ces estimateurs sont robustes à de telles perturbations, les bagger n'apportera aucune amélioration.
- Les arbres de régression et de classification sont des estimateurs connus pour être instables. Par conséquent, ce sont des bons candidats pour être baggé.
- La plupart des logiciels utilisent des arbres dans leurs procédures bagging.

Les forêts aléatoires

- Comme son nom l'indique, une forêt aléatoire consiste à agréger des arbres de discrimination ou de régression.

Définition. Soit $\{h(x, \theta_1), ..., h(x, \theta_B)\}$ une collection de prédicteurs par arbre où $(\theta_1, ..., \theta_B)$ est une suite de variables aléatoires i.i.d. Le prédicteur des forêts aléatoires est obtenu par agrégation de cette collection d'arbres.

- Une forêt aléatoire n'est donc ni plus ni moins qu'une agrégation d'arbres dépendants de variables aléatoires.
- Par exemple, bagger des arbres (construire des arbres sur des échantillons bootstrap) définit une forêt aléatoire ou Random Forests (voir Breiman & Cutler (2005)).
- Nous présentons dans cette partie la famille des random forests RI. Pour ce type de forêt aléatoire, les arbres sont construits avec l'algorithme CART.

Algorithme

- Nous avons vu dans la partie précédente que le bagging est d'autant plus performant que la corrélation entre les prédicteurs est faible.
- Afin de diminuer cette corrélation, Breiman propose de rajouter une couche d'aléa dans la construction des prédicteurs (arbres).
- Plus précisément, à chaque étape de CART, m variables sont sélectionnées aléatoirement parmi les p et la meilleure coupure est sélectionnée uniquement sur ces m variables.

Algorithme 2 Forêts aléatoires

ENTRÉES :

- x l'observation à prévoir ;
- un régresseur (arbre CART, 1 plus proche voisin...) ;
- d_n l'échantillon ;
- B le nombre d'estimateurs que l'on agrège ;
- $m \in \mathcal{N}^*$ le nombre de variables candidates pour découper un noeud.

Pour $k = 1, \ldots, B$ **:**

1. Tirer un échantillon bootstrap d_n^k dans d_n
2. Construire un arbre CART sur cet échantillon bootstrap, chaque coupure est sélectionnée en minimisant la fonction de coût de CART sur un ensemble de m variables choisies au hasard parmi les p. On note $h(., \theta_k)$ l'arbre construit.

SORTIES : L'estimateur $h(x) = \frac{1}{B} \sum_{k=1}^{B} h(x, \theta)$.

Discussion I

- Si nous sommes dans un contexte de discrimination, l'étape finale d'agrégation dans l'algorithme consiste à faire voter les arbres à la majorité.
- On retrouve un compromis biais-variance dans le choix de m :
  - lorsque m diminue, la tendance est à se rapprocher d'un choix "aléatoire" des variables de découpe des arbres. Dans le cas extrême où $m = 1$, les axes de la partition des arbres sont choisies au "hasard", seuls les points de coupure utiliseront l'échantillon.
  - Ainsi, si m diminue, la corrélation entre les arbres va avoir tendance à diminuer également, ce qui entraînera une baisse de la variance de l'estimateur agrégé. En revanche, choisir les axes de découpe des arbres de manière (presque) aléatoire va se traduire par une moins bonne qualité d'ajustement des arbres sur l'échantillon d'apprentissage, d'où une augmentation du biais pour chaque arbre ainsi que pour l'estimateur agrégé.
  - Lorsque m augmente, les phénomènes inverses se produisent.

Discussion II

- On déduit de cette remarque que le choix de m est lié aux choix des paramètres de l'arbre, notamment au choix du nombre d'observations dans ses noeuds terminaux.
  - En effet, si ce nombre est petit, chaque arbre aura un biais faible mais une forte variance. Il faudra dans ce cas là s'attacher à diminuer cette variance et on aura donc plutôt tendance à choisir une valeur de m relativement faible.
  - A l'inverse, si les arbres ont un grand nombre d'observations dans leurs noeuds terminaux, ils posséderont moins de variance mais un biais plus élevé. Dans ce cas, la procédure d'agrégation se révélera moins efficace.

En pratique...

- le nombre maximum d'observations dans les noeuds est par défaut pris relativement petit (5 en régression, 1 en classification).
- Concernant le choix de m, `randomForest` propose par défaut $m = p/3$ en régression et $m = \sqrt{p}$ en classification. Ce paramètre peut également être sélectionné via des procédures apprentissage-validation ou validation croisée.

Erreur Out Of Bag et importance des variables

Parmi les nombreuses sorties proposées par la fonction randomForest, deux se révèlent particulièrement intéressantes.

1. L'erreur Out Of Bag

- Il s'agit d'une procédure permettant de fournir un estimateur des erreurs :
  - $E[(\hat{m}(X) - Y)^2]$ en régression ;
  - $P(\hat{m}(X) \neq Y)$ en classification.
- De tels estimateurs sont souvent construits à l'aide de méthode apprentissage-validation ou validation croisée.
- L'avantage de la procédure Out Of Bag (OOB) est qu'elle ne nécessite pas de découper l'échantillon. Elle utilise le fait que les arbres sont construits sur des estimateurs baggés et que, par conséquent, ils n'utilisent pas toutes les observations de l'échantillon d'apprentissage.

- Nous la détaillons dans le cadre des forêts aléatoires mais elle se généralise à l'ensemble des méthodes bagging.
- Etant donné une observation (X_i, Y_i) de D_n, on désigne par $\mathcal{I}_{\mathcal{B}}$ l'ensemble des arbres de la forêt qui ne contiennent pas cette observation dans leur échantillon bootstrap.
  - Pour estimer, la prévision de la forêt sur Y_i on agrège uniquement ces arbres là :
$$\hat{Y}_i = \frac{1}{|\mathcal{I}_{\mathcal{B}}|} \sum_{k \in \mathcal{I}_{\mathcal{B}}} h(X_i, \theta_k).$$
  - Si on est dans un contexte de classification, la prévision s'obtient en faisant voter ces arbres à la majorité.
- L'erreur Out Of Bag est alors définie par :
  - $\frac{1}{n} \sum_{i=1}^{n} (\hat{Y}_i - Y_i)^2$ en régression ;
  - $\frac{1}{n} \sum_{i=1}^{n} 1_{\hat{Y}_i \neq Y_i}$ en classification.

2. Importance des variables

- Que ce soit en régression ou discrimination, il est souvent crucial de déterminer les variables explicatives qui jouent un rôle important dans le modèle.
- L'inconvénient des méthodes d'agrégation est que le modèle construit est difficilement interprétable, on parle souvent d'aspect "boite noire" pour ce type de méthodes.
- Pour le modèle de forêts aléatoires que nous venons de présenter, Breiman & Cutler (2005) proposent une mesure qui permet de quantifier l'importance des variables $X_j, j = 1, \ldots, p$ dans le modèle.

- On désigne par OOB_k l'échantillon Out Of Bag associé au k ème arbre de la forêt. Cet échantillon est formé par les observations qui ne figurent pas dans le k ème échantillon bootstrap. On note E_{OOB_k} l'erreur de prédiction de l'arbre $h(.,\theta_k)$ mesurée sur cet échantillon :

$$E_{OOB_k} = \frac{1}{|OOB_k|} \sum_{i \in OOB_k} (h(X_i, \theta_k) - Y_i)^2.$$

- On désigne maintenant par OOB_j^k l'échantillon OOB_k dans lequel on a perturbé aléatoirement les valeurs de la variable j et par $E_{OOB_j^k}$ l'erreur de prédiction de l'arbre $h(.,\theta_k)$ mesurée sur cet échantillon :

$$E^j_{OOB_k} = \frac{1}{|OOB^j_k|} \sum_{i \in OOB^j_k} (h(X^j_i, \theta_k) - Y_i)^2,$$

où les X_i^j désignent les observations perturbées de OOB_k^j.

- Heuristiquement, si la j ème variable joue un rôle déterminant dans la construction de l'arbre $h(., \theta_k)$, alors une permutation de ces valeurs dégradera fortement l'erreur.
- La différence d'erreur $E^j_{OOB_k} - E_{OOB_k}$ sera alors élevée. L'importance de la j ème variable sur la forêt est mesurée en moyennant ces différences d'erreurs sur tous les arbres :

$$Imp(X_j) = \frac{1}{B} \sum_{k=1}^{B} (E^j_{OOB_k} - E_{OOB_k}).$$