Détection par boosting de données aberrantes en régression

Nathalie Chèze, Jean-Michel Poggi

Détection par Boosting de Données Aberrantes en Régression

Nathalie Cheze[*,**], Jean-Michel Poggi[*,***]

[*]Université Paris-Sud, Lab. Mathématique, Bât. 425, 91405 Orsay, France
jean-michel.poggi@math.u-psud.fr
[**]Université Paris 10-Nanterre, Modal'X, France
cheze@u-paris10.fr
[***]Université Paris 5 Descartes, France

Résumé. Nous proposons une méthode basée sur le boosting, pour la détection des données aberrantes en régression. Le boosting privilégie naturellement les observations difficiles à prévoir, en les surpondérant de nombreuses fois au cours des itérations. La procédure utilise la réitération du boosting pour sélectionner parmi elles les données effectivement aberrantes. L'idée de base consiste à sélectionner l'observation la plus fréquemment rééchantillonnée lors des itérations du boosting puis de recommencer après l'avoir retirée. Le critère de sélection est basé sur l'inégalité de Tchebychev appliquée au maximum du nombre moyen d'apparitions dans les échantillons bootstrap. Ainsi, la procédure ne fait pas d'hypothèses sur la loi du bruit. Des exemples tests bien connus sont considérés et une étude comparative avec deux méthodes classiques illustrent le comportement de la méthode.

1 Introduction

Rousseeuw et Leroy (1987) proposent un panorama très complet des problèmes de détection de données aberrantes en régression. Le modèle sous-jacent, la méthode d'estimation et le nombre de données aberrantes par rapport à la taille de l'échantillon conduisent à définir différents types de données aberrantes. Par exemple, plusieurs voies de contamination sont distinguées : dans l'espace de la variable réponse, dans celui des covariables ou dans les deux. De nombreuses méthodes ont été développées pour traiter ces situations.

Une idée majeure est néanmoins facile à dégager : la robustesse et la référence à un modèle paramétrique sous-jacent, le plus souvent linéaire. Par exemple, on peut citer outre Rousseeuw et Leroy (1987), Pena et Yohai (1999) et pour un rapide panorama, saisi au travers de la présentation d'un logiciel, on pourra consulter Verboven et Hubert (2005)). Parmi les méthodes classiques évoquées on peut en dégager deux dont les principes marquent fortement ce type de méthodes.

Le premier contexte est celui des méthodes factorielles robustes (voir Jolliffe (2002)) qui sont basées sur des estimateurs robustes de la matrice de covariance comme l'estimateur MCD (pour Minimum Covariance Determinant, voir Rousseeuw et Van Driessen (1999)). Les données (non aberrantes) sont assimilées à un vecteur gaussien dont les caractéristiques sont estimées sur un échantillon, éventuellement contaminé par des observations aberrantes, par des

méthodes robustes. On construit alors une région de confiance pour la moyenne à l'extérieur de laquelle sont détectées les données aberrantes.

Le second contexte s'insère dans le cadre de la régression linéaire où l'on considère des estimateurs de type moindres carrés robustes ou moindres écarts comme l'estimateur LTS (pour Least Trimmed Squares, voir Rousseeuw et Leroy (1987)) ou encore l'estimateur LMS (pour Least Median of Squares, voir Rousseeuw (1984)). Dans ce cas, au moyen d'une méthode robuste on ajuste un modèle linéaire. On construit alors, sous l'hypothèse du modèle linéaire gaussien une région de confiance pour les résidus à l'extérieur de laquelle sont détectées les données aberrantes, autrement dit, les données mal prédites par le modèle robuste sont considérées comme aberrantes.

Ainsi, ces approches sont limitées par la définition d'une donnée aberrante par les déviations par rapport à un modèle linéaire ou paramétrique donné. Ici, on préférera considérer un modèle de régression général de la forme $Y = f(X) + \xi$, en ne faisant pas d'hypothèse paramétrique ni sur la fonction f, ni sur la loi du bruit.

Notre objectif est de proposer une procédure de détection entièrement automatique dans le sens où les paramètres associés ne dépendent que des données. Les deux ingrédients clés sont le boosting d'une part, et l'usage d'une méthode d'estimation complètement non paramétrique d'autre part. Ceci permet à la fois d'être le moins lié possible à un modèle de la dépendance entre la variable réponse et les variables explicatives et d'être capable d'explorer les différents aspects des données par rééchantillonnage adaptatif.

L'article est organisé de la façon suivante. Dans la Section 2, les arbres CART en régression ainsi que le boosting dans ce contexte sont brièvement rappelés. La procédure de détection est ensuite introduite et motivée dans la Section 3. La Section 4 formule trois remarques méthodologiques. Les Sections 5 et 6 sont dédiées à une illustration expérimentale de la procédure de détection, qui est comparée avec deux méthodes classiques sur des données tant simulées que réelles. Enfin, la Section 7 propose quelques éléments de conclusion.

2 CART et boosting en régression

Considérons le modèle de régression suivant :

$$Y = f(X) + \xi,$$

où $(X, Y) \in \mathbb{R}^p \times \mathbb{R}$, f est la fonction de régression inconnue et ξ un bruit additif inobservable centré conditionnellement à X et de variance σ_ξ^2 inconnue. On suppose disposer d'un échantillon L composé de n réalisations de la variable (X, Y), possiblement contaminé par des données aberrantes.

2.1 CART

On se restreint à des arbres de régression CART pour générer les estimateurs de f, notés génériquement $\hat{f}$ dans la suite. Comme nous sommes principalement intéressés par la suite des probabilités d'échantillonnage des observations produite par le boosting, une propriété particulièrement attractive de cette méthode d'estimation est, dans ce contexte, son instabilité. En effet, les arbres de régression booststrap n'ont pas le même nombre de nœuds terminaux,

et bien sûr ne sont pas basées sur les mêmes coupures pour des perturbations aléatoires de l'échantillon L. Ils permettent ainsi d'explorer différents aspects des données.

Dans ce travail, on utilise précisément la méthode CART en régression, introduite par Breiman et al. (1984)), permettant de construire à partir de L un estimateur $\hat{f}$ de f qui a une faible erreur de généralisation.

Le travail d'estimation est constitué lors d'une première étape en construisant un arbre maximal très profond et suradapté à L à partir duquel on déduit la famille de modèles au sein de laquelle on opère une sélection. Celle-ci est menée lors de la seconde étape, dite d'élagage, permettant de sélectionner le meilleur modèle au sens des moindres carrés pénalisés par le nombre de feuilles de l'arbre.

Comme la loi jointe est inconnue, l'erreur de resubstitution est utilisée pour générer les modèles et l'erreur de prédiction de $\hat{f}$ est évaluée (et l'arbre final choisi) en utilisant un schéma de validation croisée en 10 blocs, quand le nombre d'observations est suffisant (plus précisément lorsque la taille de l'échantillon est supérieure à 100). Dans le cas contraire, on retient simplement l'arbre maximal.

2.2 L'algorithme du boosting

Une remarque classique à propos de la procédure de boosting (introduite tout d'abord en classification supervisée par Freund et Schapire (1997), et appelée AdaBoost, puis adaptée au cas de la régression par Drucker (1997)) et ses variantes, est sa sensibilité aux données aberrantes. Cette propriété, en général considérée comme un inconvénient, peut être exploitée (voir Gey et Poggi (2006)) pour améliorer le modèle ajusté grâce à une méthode d'estimation donnée en s'adaptant mieux aux observations particulièrement difficiles à prévoir. Le but est ici de l'utiliser pour détecter les données aberrantes. Notre procédure est basée sur l'information fournie par le processus de rééchantillonnage adaptatif engendré par l'application du boosting aux arbres CART. Ce processus adaptatif contient beaucoup d'information, souvent négligée, sur les données et constitue, en dehors des remarquables performances du prédicteur agrégé, l'un des résultats intermédiaires les plus intéressant du boosting d'un point de vue de l'analyse des données.

L'algorithme du boosting utilisé dans cet article est rappelé dans la Table 1. Il a tout d'abord été proposé par Drucker (1997) puis a été étudié par Gey et Poggi (2006).

Ainsi, l'algorithme du boosting engendre une suite d'estimations de la fonction de régression dont les éléments sont individuellement ajustés à un échantillon bootstrap obtenu à partir de l'échantillon initial par un rééchantillonnage adaptatif qui met l'accent sur les observations mal prédites par son prédécesseur dans la suite. Par conséquent, un tel rééchantillonnage conduit à se focaliser naturellement sur les observations difficiles à prévoir en utilisant une méthode d'estimation donnée, c'est-à-dire à se concentrer sur les observations les plus souvent mal prédites. Bien sûr une donnée aberrante est une telle observation.

3 La procédure de détection des données aberrantes

La stratégie adoptée pour la détection des données aberrantes est donnée par la Table 2. Elle consiste en deux étapes : la première détecte les observations difficiles à prévoir et la seconde sélectionne parmi elles les données aberrantes.

TAB. 1 – *Algorithme boosting :* $[M, i_0] = boost(L, K)$**.**

Entrées :	L'échantillon L de taille n et le nombre d'itérations K
Initialisation :	Poser $p_1 = D$ la distribution uniforme sur $\{1, \ldots, n\}$
Boucle :	pour $k = 1$ à K faire
étape 1	- tirer au hasard dans L avec remise, suivant la loi p_k, un échantillon L_k de taille n,
étape 2	- construire, grâce à CART, un estimateur $\hat{f}_k$ de f à partir L_k,
étape 3	- calculer sur l'échantillon d'origine L : $i = 1, \ldots, n$ $l_k(i) = \left(Y_i - \hat{f}_k(X_i)\right)^2$ et $\epsilon_{p_k} = \sum_{i=1}^{n} p_k(i) l_k(i)$, $\beta_k = \frac{\epsilon_{p_k}}{\max\limits_{1 \le i \le n} l_k(i) - \epsilon_{p_k}}$ et $d_k(i) = \frac{l_k(i)}{\max\limits_{1 \le i \le n} l_k(i)}$, $p_{k+1}(i) = \beta_k^{1-d_k(i)} p_k(i)$, normaliser p_{k+1} de sorte que la somme soit égale à 1
étape 4	- calculer $I_{i,k}$ le nombre d'apparitions de l'observation i dans L_k
Sorties :	calculer $S_i = \frac{1}{K} \sum_{k=1}^{K} I_{i,k}$ et $M = max_{i \in L}\, S_i, \quad i_0 = argmax_{i \in L}\, S_i$

L'idée clé de la première étape consiste à retenir l'observation la plus fréquemment ré-échantillonnée au long des itérations du boosting et de réitérer après l'avoir ôtée. Ainsi l'ensemble final H de la Table 2 contient les J observations d'indice $i(j)$ et qui sont apparues en moyenne M_j fois dans les échantillons bootstrap.

La seconde étape définit une région de confiance entièrement dirigée par les données pour sélectionner les données aberrantes dans H. La justification de la règle de sélection est la suivante. Pour chaque $j \in (1, ..., J)$, assimilons le problème de la détection des valeurs aberrantes à un problème de test individuel de l'hypothèse nulle : H_0 : *l'observation* $i(j)$ *n'est pas aberrante*, contre l'hypothèse alternative : H_1 : *l'observation* $i(j)$ *est aberrante*. Puisque si $i(j)$ est associée à une donnée aberrante alors M_j est grand, il est naturel de choisir la région de rejet de la forme suivante :

$$W = (M_j > C_\alpha)$$

pour un niveau de signification α donné.

En appliquant l'inégalité de Tchebychev à M_j, on obtient :

$$P_{H_0}\left(\frac{M_j - m}{\sigma} > \sqrt{\frac{1}{\alpha}}\right) \le P_{H_0}\left(\frac{|M_j - m|}{\sigma} > \sqrt{\frac{1}{\alpha}}\right) \le \alpha,$$

d'où l'on déduit C_α :

TAB. 2 – *Algorithme de détection des données aberrantes.*

Entrées :	J le nombre d'applications du boosting, L l'échantillon initial, α le niveau de signification de l'intervalle de confiance, K le nombre d'itérations de chaque boosting.
Initialisation :	Poser $L^1 = L$
Étape 1 :	pour $j = 1$ à J faire $[M_j, i(j)] = boost(L^j, K)$; $L^{j+1} = L^j \setminus i(j)$; $H = L \setminus L^J$
Étape 2 :	Les données aberrantes sont celles d'indice $i(j) \in H$ tel que $(M_j > C_\alpha)$

$$C_\alpha = \hat{m}_{rob} + \sqrt{\frac{\hat{\sigma^2}_{rob}}{\alpha}}$$

où $\hat{m}_{rob}$ et $\hat{\sigma^2}_{rob}$ sont des estimateurs robustes de m et σ^2 l'espérance et la variance de M_j sous l'hypothèse H_0. Le "trou" entre M_j et $m = E_{H_0}(M_j)$ sous H_1 permet de contourner l'aspect conservatif, défaut bien connu, de l'inégalité de Tchebychev. En effet, même si

$$P_{H_0}\left(|M_j - m|/ > \sigma\alpha^{-1/2}\right) << \alpha,$$

conduit à rétrécir la région de rejet, les hypothèses à tester sont suffisamment éloignées par la réitération du boosting pour que W suffise à sélectionner correctement les données aberrantes. Aussi, on utilisera $\alpha = 5\%$ pour tous les calculs dans la suite. L'avantage majeur de cette procédure est qu'elle ne fait pas d'hypothèses sur la loi du bruit et qu'elle ne requiert pas de choix de paramètres *a priori*, sinon J et K qui peuvent être choisis sur des considérations très générales (détaillées dans la section suivante).

4 Remarques sur la procédure de détection

Formulons quelques remarques sur la procédure de détection proposée sous la forme de quelques questions.

4.1 Pourquoi réitérer le boosting ?

La première d'entre elles : pourquoi réitérer le boosting alors même que celui-ci a naturellement tendance à se concentrer sur les observations les plus difficiles à prévoir ?

Il peut exister un effet de masquage, c'est-à-dire qu'une ou plusieurs observations peuvent empêcher la détection des autres. Mais ce défaut n'existe pas seulement dans des cas extrêmes mais aussi dans des situations plus standards : ainsi les j_0 observations les plus fréquemment rééchantillonnées au cours d'un boosting simple sont, en général, différentes des j_0 premières observations sélectionnées pas à pas par J réitérations du boosting. La Figure 1 illustre ce phénomène sur des données simulées avec 4 données aberrantes (indicées de 11 à 14), sur $n = 75$ observations. On y trouve les graphes des plus grands M_j : à droite ceux obtenus par un boosting simple, à gauche par un boosting itéré.

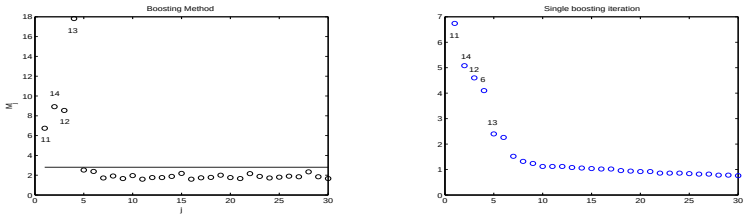

FIG. 1 – *Données simulées avec 4 données aberrantes sur $n = 75$ observations. Graphes des plus grands M_j : à droite, boosting simple ; à gauche, boosting itéré.*

Si le boosting simple permet bien de sélectionner les quatre observations aberrantes, il promeut aussi deux observations supplémentaires dans les six premières qui se détachent. En outre les autres valeurs de M_j associées à des observations standards décroissent lentement sans qu'une cassure nette soit patente. En revanche, sur le graphe de gauche, le boosting itéré permet à la fois de sélectionner les quatre observations aberrantes mais aussi de les détacher nettement des autres. La séparation entre les deux hypothèses est particulièrement nette. La hauteur de la droite horizontale est donnée par C_α. Le comportement des M_j qui ne sont pas associés à des données aberrantes semble parfaitement compatible avec la réalisation d'une suite de variables aléatoires indépendantes (ou faiblement dépendantes) et identiquement distribuées, et donc de même espérance m et même variance σ^2.

4.2 Sur le nombre d'itérations du boosting

Deux questions naturelles concernent le choix du nombre d'itérations K de chaque boosting et le nombre de fois J qu'il convient de réitérer le boosting.

La première de ces questions, qui porte en fait sur la stabilisation de l'estimateur agrégé produit par Adaboost, a été examinée par Breiman et al. (1984) pour la classification et par Gey et Poggi (2006) pour la régression. Ces résultats expérimentaux permettent de considérer que $K = 50$ constitue une valeur suffisante. Il faut noter de plus que dans le contexte qui nous occupe ici, la qualité de l'estimation n'est pas primordiale, il suffit qu'elle permette de mettre en valeur les observations difficiles à prévoir.

Une réponse à la seconde question portant sur le choix du nombre d'itérations J du boosting est qu'il faut réitérer suffisamment pour que non seulement toutes les données aberrantes soient sélectionnées mais aussi que les observations non contaminées par les données aberrantes soient assez nombreuses pour estimer convenablement l'espérance m et la variance σ^2

sous H_0 afin de pouvoir les injecter dans l'inégalité de Tchebichev. Typiquement, lorsque n n'est pas trop grand ou que le coût algorithmique n'est pas trop important (rappelons que la complexité théorique de notre procédure se situe, à une constante près, entre $KJp\log(n)$ et $KJpn^2$) le choix effectué ici est $J = 0.75n$. Sinon, il convient de prendre n de l'ordre de deux fois un majorant plausible du nombre de données aberrantes.

4.3 Pourquoi ne pas utiliser le bagging au lieu du boosting ?

La procédure de bagging (pour bootstrap and aggregating, voir Breiman (2001)) peut être définie comme un cas particulier du boosting en posant $L_k \equiv L$ et $p_k \equiv D$ la distribution uniforme sur $\{1, \ldots, n\}$ dans l'algorithme rappelé dans la Table 1. Ainsi le prédicteur agrégé mélange les estimateurs fournis par des échantillons bootstrap sans adaptation de la loi de rééchantillonnage.

De ce point de vue, le bagging semble un bon candidat pour construire un estimateur robuste de la fonction de régression. En fait, cela n'est vrai que lorsque le nombre de données aberrantes est faible vis-à-vis de la taille de l'échantillon, mais c'est généralement le cas. Une idée peut être alors de sélectionner les observations mal prédites par cet estimateur robuste en seuillant les résidus convenablement par rapport aux caractéristiques supposées du bruit. Bien sûr, cette stratégie requiert, comme ces analogues linéaires évoquées dans l'introduction, des hypothèses sur la loi du bruit.

5 Application à des données tests

Nous avons examiné de nombreux jeux de données tests pour étudier le comportement de la méthode proposée pour des données aberrantes de différents types : plusieurs modes de contamination (dans la direction de X, de Y ou dans les deux), pour des tailles d'échantillon variées : des petites tailles (qui auraient pu se révéler critiques pour une procédure basée sur une méthode d'estimation non paramétrique) aussi bien que des grandes.

L'étude expérimentale exhaustive est menée dans Cheze et Poggi (2005) où l'on s'intéresse en particulier à un ensemble de données réelles (que l'on trouvera dans Rousseeuw et Leroy (1987)) très intéressantes puisque de petite taille et très étudiées par de nombreux auteurs pendant plus de vingt ans.

On applique à chacun de ces exemples notre méthode et l'on prend comme référence les résultats obtenus par la méthode basée sur l'estimateur LTS (Least Trimmed Squares) considérée comme particulièrement performante sur ce type de jeux de données de petite taille et justiciable d'une modélisation linéaire.

La principale conclusion est que, dans quasiment tous les cas, on obtient des résultats très voisins de ceux obtenus par les méthodes basées sur l'estimateur MCDCOV et sur LTS en dépit du très faible nombre de données (autour de 20 observations pour la plupart des jeux considérés) et du modèle paramétrique. Plus précisément, on obtient des mauvais résultats sur seulement trois exemples parmi dix-huit. Pour les autres, on obtient toujours de bons résultats de détection avec une sélection partielle ou totale.

Nous nous contenterons ici d'examiner trois situations typiques. Chacune d'elles est illustrée par une figure composée de quatre graphiques : en haut à gauche, les données ; en haut à droite, les valeurs de M_j pour $1 \leq j \leq J$ (défini dans la Table 2) obtenue par notre méthode

(en utilisant $\alpha = 5\%$) ; en bas, les résultats obtenus grâce aux deux méthodes alternatives basées sur les résidus standardisés du prédicteur LTS d'une part et d'autre part les distances de Mahalanobis robustes. Dans les légendes, n_{out}^{LTS} est le nombre de données aberrantes détectées par la méthode LTS. Les estimations $\hat{m}_{rob}$ et $\hat{\sigma}_{rob}$ nécessaires pour calculer le seuil de détection sont obtenus en utilisant les estimateurs MCD appliqués à $(M_j)_{1 \leq j \leq J}$. Ces estimateurs ainsi que les résultats de ces méthodes alternatives ont été obtenus grâce à la librairie LIBRA (Verboven et Hubert, 2005) développée en MATLAB®, en utilisant les valeurs par défaut des paramètres.

Pour notre méthode ainsi que pour MCDCOV, les données aberrantes sont celles dont les indices sont associés aux points situés au dessus de la droite horizontale alors que pour la méthode LTS, les données aberrantes sont repérées par des points situés à l'extérieur de l'intervalle délimité par les deux droites horizontales. En outre, on indique les indices des données aberrantes pour les données simulées et, pour les données réelles, ceux de certaines observations choisies de façon à faciliter la lecture et l'interprétation des graphiques. Remarquons enfin que, pour le graphique associé à notre méthode, les J points correspondent aux J réitérations du boosting alors que pour les deux autres, n points sont figurés.

5.1 Un exemple de sélection partielle

Tout d'abord concentrons-nous sur l'un des trois exemples pour lesquels notre méthode échoue. En fait, un examen attentif des arbres de régression construits explique la situation. Le défaut de détection est due à la flexibilité de CART et à sa capacité à s'adapter aux données, conséquence de l'absence de modèle dans ce type de méthode locale. En effet, lorsque le modèle CART crée un noeud contenant toutes les données aberrantes, la méthode ne peut les mettre en lumière. La Figure 2 illustre une telle situation.

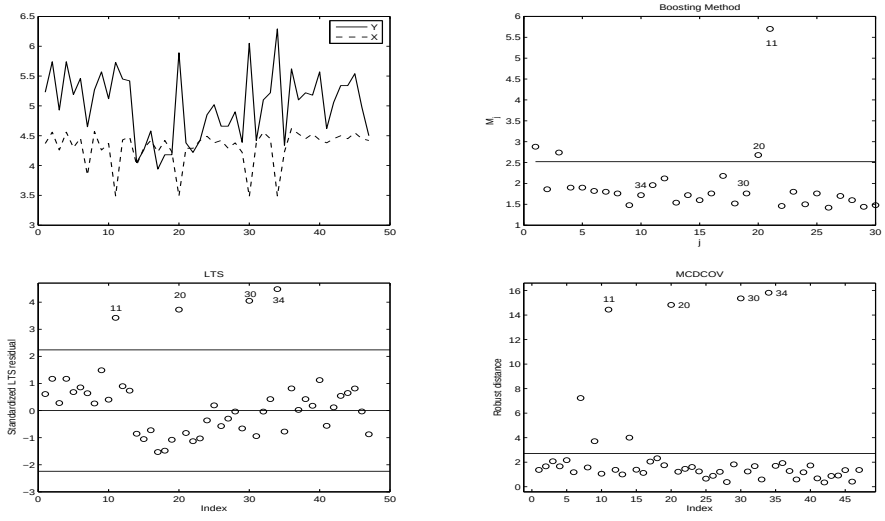

FIG. 2 – *Association stellaire CYG OB1, données p. 27 de Rousseeuw, Leroy (1987),* $n = 47$, $p = 1$, $n_{out}^{LTS} = 4$.

Comme le montre le graphique en haut à gauche, les quatre données aberrantes (identifiées en utilisant LTS) d'indice 11, 20, 30 et 34, sont atypiques autant dans la direction de la variable réponse que dans celle de la variable explicative. Ces données concernent l'association stellaire CYG OB1, et sont constituées du logarithme de la température à la surface de l'étoile (X) et du logarithme de l'intensité lumineuse de l'étoile (Y). Les quatre étoiles identifiées sont les plus grosses.

Notre méthode détecte seulement deux d'entre elles, alors que d'une part, LTS capture les quatre comme MCDCOV qui en met en lumière trois autres. L'explication est que CART est suffisamment flexible pour créer un nœud contenant les quatre données aberrantes qui sont atypiques d'une façon semblable : leurs valeurs de X sont très proches et très éloignées des autres, et leurs valeurs de Y sont les quatre premiers maxima.

Observons cependant qu'au cours des itérations de l'algorithme de détection par boosting, dès que les observations 34 et 30 sont ôtées de l'échantillon, les observations aberrantes d'indice 20 et 11 sont alors aisément détectées.

5.2 Un exemple de détection correcte

Le deuxième exemple montre, au contraire, l'apport de la flexibilité de CART dans la détection et met en lumière une différence importante entre les méthodes MCDCOV et LTS. La Figure 3 détaille les résultats obtenus.

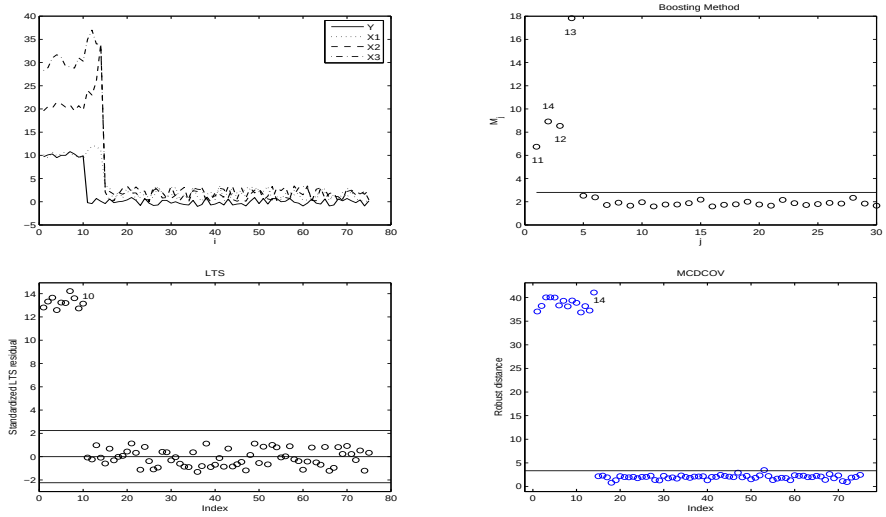

FIG. 3 – *Données simulées p. 94 de Rousseeuw, Leroy (1987),* $n = 75$, $p = 3$, $n_{out} = 4$.

Cet exemple, utilisé plus haut (pour obtenir la Figure 1), étant le seul exemple de données simulées de Rousseeuw et Leroy (1987), le "vrai" nombre de données aberrantes est connu et vaut 4. Le graphique en haut, à gauche montre que l'échantillon peut être divisé en trois parties. Deux sous-populations différentes et les données aberrantes : les observations d'indice compris

entre 1 et 10, celles d'indice plus grand que 15 et les quatre données aberrantes numérotées de 11 à 14.

Notre méthode les détecte correctement sans aucune fausse détection alors que les deux autres méthodes assimilent la première population aux observations aberrantes. Plus précisément, MCDCOV détecte les quatre données aberrantes ainsi que les premières observations puisqu'elle détecte les données aberrantes séparément dans chacune des deux directions, celles de X et Y. Quant à LTS, il échoue car il ajuste un modèle linéaire unique à toutes les données. Ce modèle robuste délivre une prédiction robuste proche de 0, ce qui explique que seule la sous-population majoritaire est détectée sans les données aberrantes puisque celles-ci on une réponse presque nulle.

5.3 Un exemple de sélection correcte mais de détection partielle

Le troisième exemple illustre la situation de sélection correcte mais de détection partielle : les données aberrantes sont bien promues dans l'ensemble H mais le seuil est trop élevé pour les en extraire toutes automatiquement.

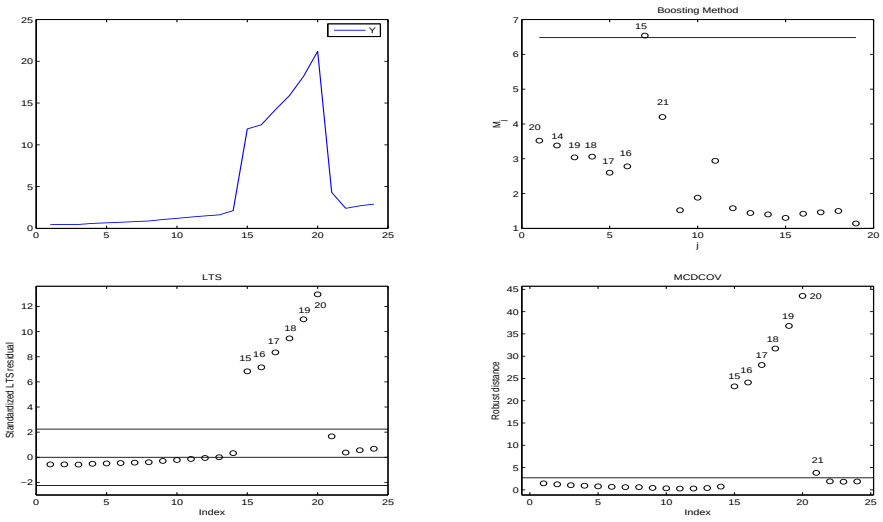

FIG. 4 – *Données p. 26 de Rousseeuw, Leroy (1987),* $n = 24$, $p = 1$, $n_{out}^{LTS} = 6$.

La Figure 4 montre une détection parfaite par les méthodes MCDCOV et LTS, alors que notre méthode échoue à correctement détecter les sept données aberrantes d'indice entre 15 et 20 (voir graphique en haut, à gauche). Néanmoins, la méthode basée sur le boosting sélectionne correctement les données aberrantes : les huit observations correspondant aux huit valeurs les plus élevées de l'ensemble H contiennent bien toutes les données aberrantes mais le seuil est trop élevé. Ceci vient du fait que $n = 24$ et $J - j_0 = 19 - 6$ sont bien trop faibles pour disposer d'un nombre suffisant d'observations pour estimer convenablement les paramètres inconnus requis pour la définition de la région de détection. Les estimations sont contaminées par un taux de données aberrantes (25%) particulièrement élevé.

6 Un exemple réel sans données aberrantes

Terminons par un dernier exemple traitant de données nombreuses (environ 1200) sans observations aberrantes. Il s'agit de données de pollution de l'air en région parisienne qui ont été utilisées, en particulier, pour l'analyse et la prévision de la concentration d'ozone (voir Chèze et al. (2003)). Les jours très pollués sont difficiles à prévoir et le modèle sous-jacent devient fortement non linéaire pour de telles observations. Ces données ne contiennent pas, à proprement parler de données aberrantes. Dans la Figure 5, on peut noter que, comme attendu, les méthodes LTS et MCDCOV conduisent à un grand nombre de fausses alarmes alors que celle basée sur le boosting ne fait ressortir qu'un seul jour comme aberrant.

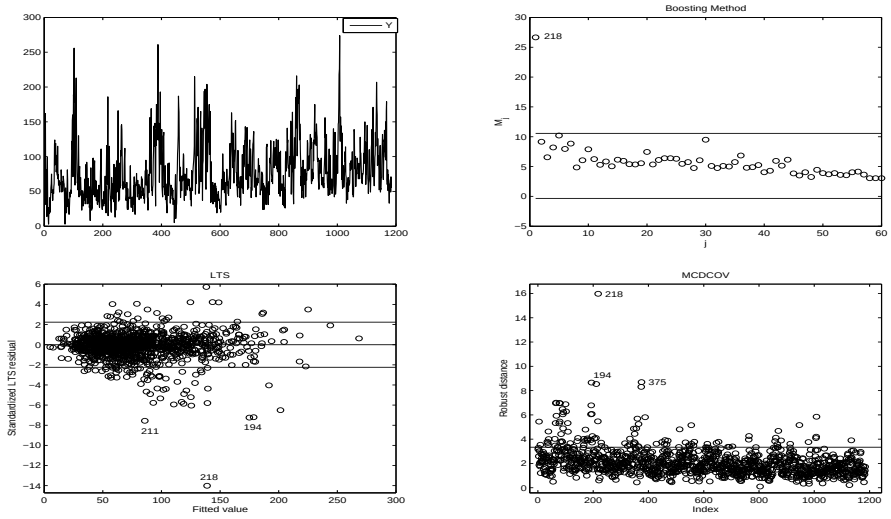

FIG. 5 – *Données de pollution de l'air par l'ozone.*

Un examen plus attentif de l'unique jour sélectionné par notre procédure montre qu'il correspond à un jour dont la température est élevée (autour de 28°C), dont le vent est faible et dont la veille très polluée (une concentration maximum autour de 126 $\mu g/m^3$). Ceci devrait normalement conduire à une concentration d'ozone autour de 120 $\mu g/m^3$ alors que l'on observe seulement 15 $\mu g/m^3$. Il s'agit donc d'un jour particulièrement difficile à prévoir, en fait impossible : c'est un jour atypique par rapport au faible nombre de variables explicatives considérées par ce modèle statistique simple.

Il faudrait intégrer des variables à plus grande échelle permettant de rendre compte des phénomènes de transport d'ozone et de déplacement des masses d'air pour arriver à prévoir convenablement de tels épisodes.

De ce point de vue, on peut aussi constater sur la Figure 5 que la suite des M_j ne peut pas être assimilée à la réalisation d'une suite de variables indépendantes et de mêmes caractéristiques au second ordre. Il y a une légère tendance linéaire et la décroissance lente et régulière observée est la signature des variables omises dans la modélisation du phénomène. On pourra trouver des compléments dans Bel et al. (1999).

7 Conclusion

Notre procédure exploite l'information fournie par le boosting des arbres de régression CART qui, conjuguée à la réitération, permet de détecter les données aberrantes. Elle ne requiert le choix d'aucun paramètre critique car elle ne fait d'hypothèses ni sur la forme de la relation entre la réponse et les variables explicatives, ni sur la loi du bruit.

De ce point de vue, le boosting permet de s'affranchir des limitations usuelles de la définition d'une donnée aberrante et donc de rendre cette notion plus intrinsèque, moins dépendante d'un contexte particulier de modélisation. Ainsi, l'usage de méthodes issues de la statistique inférentielle (l'estimation non paramétrique) conjugué à des idées d'agrégation de prédicteurs récemment apparues en apprentissage statistique permet de revisiter un thème essentiel de l'analyse statistique des données.

Quelques exemples tests bien connus et une comparaison avec deux méthodes classiques illustrent le comportement de la méthode dans cet article. Le lecteur trouvera une étude plus complète dans Cheze et Poggi (2005).

Notons pour finir que, puisque CART permet aussi de construire des arbres de classification, on peut penser à étendre ce type d'algorithme pour étudier une situation analogue en classification en revenant à l'algorithme AdaBoost pour effectuer le boosting.

Références

Bel, L., L. Bellanger, V. Bonneau, G. Ciuperca, D. Dacunha-Castelle, C. Deniau, B. Ghattas, M. Misiti, Y. Misiti, G. Oppenheim, J.-M. Poggi, et R. Tomassone (1999). Eléments de comparaison de prévisions statistiques des pics d'ozone. *Revue de Statistique Appliquée XLVII* (3), 7–25.

Breiman, L. (2001). Using iterated bagging to debias regressions. *Machine Learning 45*(3), 261–277.

Breiman, L., J. H. Friedman, R. A. Olshen, et C. J. Stone (1984). *Classification And Regression Trees*. Chapman & Hall.

Cheze, N. et J.-M. Poggi (2005). Outlier detection by boosting regression trees. Technical Report 17, Univ. Orsay. 22 p.

Cheze, N., J.-M. Poggi, et B. Portier (2003). Partial and recombined estimators for nonlinear additive models. *Stat. Inf. for Stochastic Processes 6*(2), 155–197.

Drucker, H. (1997). Improving regressors using boosting techniques. In M. Kaufmann (Ed.), *Proc. of the 14th Int. Conf. on Machine Learning*, pp. 107–115.

Freund, Y. et R. E. Schapire (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences 55*(1), 119–139.

Gey, S. et J.-M. Poggi (2006). Boosting and instability for regression trees. *Computational Statistics & Data Analysis* (50), 533–550.

Jolliffe, I. (2002). *Principal Component Analysis*. Springer.

Pena, D. et V. Yohai (1999). A fast procedure for outlier diagnostics in large regression problems. *Journal of the American Statistical Association 94*(446), 434–445.

Rousseeuw, P. (1984). Least median of squares regression. *J. Amer. Statist. Assoc. 79*, 871–880.

Rousseeuw, P. et A. Leroy (1987). *Robust regression and outlier detection*. Wiley.

Rousseeuw, P. et K. Van Driessen (1999). A fast algorithm for the minimum covariance determinant estimator. *Technometrics 41*, 212–223.

Verboven, S. et M. Hubert (2005). LIBRA : a MATLAB library for robust analysis. *Chemometrics and Intelligent Laboratory Systems 75*, 127–136.

Summary

A procedure for detecting outliers in regression problems based on information provided by boosting trees is proposed. Boosting focuses on observations that are hard to predict, by giving them extra weights. In the present paper, such observations are considered to be possible outliers, and a procedure is proposed that uses the boosting results to diagnose which observations could be outliers. The key idea is to select the most frequently resampled observation along the boosting iterations and reiterate boosting after removing it. The selection criterion is based on Tchebychev's inequality applied to the maximum over the boosting iterations of the average number of appearances in bootstrap samples. So the procedure is noise distribution free. A lot of well-known bench data sets are considered and a comparative study against two classical competitors allows to show the value of the method.