HAL open science

PlsRglm : Régression PLS et modèles linéaires généralisés sous R

Frédéric Bertrand, Myriam Maumy-Bertrand, Nicolas Meyer

To cite this version:

Frédéric Bertrand, Myriam Maumy-Bertrand, Nicolas Meyer. PlsRglm : Régression PLS et modèles linéaires généralisés sous R. 42èmes Journées de Statistique, 2010, Marseille, France, France. inria-00494857

HAL Id: inria-00494857

https://inria.hal.science/inria-00494857

Submitted on 24 Jun 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

`plsRglm`, modèles linéaires généralisés PLS sous R

`plsRglm`, PLS generalized linear models for R

Frédéric Bertrand[1], Myriam Maumy-Bertrand[2] & Nicolas Meyer[3]

[1] *Institut de Recherche Mathématique Avancée, Université de Strasbourg*
E-mail : *fbertran@math.u-strasbg.fr*

[2] *Institut de Recherche Mathématique Avancée, Université de Strasbourg*
E-mail : *mmaumy@math.u-strasbg.fr*

[3] *Laboratoire de Biostatistique, Faculté de Médecine, Université de Strasbourg*
E-mail : *nmeyer@unistra.fr*

Résumé
La finalité de la bibliothèque de fonctions `plsRglm` écrite par les auteurs et implémentée dans le logiciel R (R Development Core Team 2008) est multiple et s'organise principalement autour de deux thématiques : l'extension de la régression PLS au cas des modèles linéaires généralisés, en particulier celui des régressions logistiques (Bastien *et al.* 2005), et le traitement des jeux de données incomplets par validation croisée. Ces modèles ont été appliqués avec succès à des données de nature variée : par Bastien *et al.* (2005) à des problèmes de régression multiple en liaison avec les données de Cornell (Kettaneh-Wold 1992), à des problèmes de régression linéaire généralisée et en particulier à une étude de la qualité de vins de Bordeaux à l'aide d'un modèle de régression logistique ordinale. Plus récemment, les auteurs se sont servis de modèles de régression logistique binaire PLS pour étudier des données d'allélotypage (Meyer *et al.* 2009) qui interviennent dans la compréhension de mécanisme liés à l'évolution des cancers.
Mots-clés : Régression PLS, modèles linéaires généralisés, bootstrap, données de grande dimension, logiciel R.

Abstract
There are mainly two aims for the `plsRglm` library written by the authors for the R software (R Development Core Team 2008). The extension of PLS regression to generalized linear models, and for instance to logistic regression models (Bastien *and al.* 2005), and the need to provide tools to PLS users to deal with incomplete datasets using cross-validation. These models were successfully applied to datasets of various kind : by Bastien *and al.* (2005) to multiple regression problems linked to the famous mixture dataset of Cornell (Kettaneh-Wold 1992), to generalized linear regression and especially to a study of Bordeaux wine quality thanks to an ordinal logistic regression model. More recently, the authors carried out the analysis of allelotyping data thanks to PLS binary logistic regression models (Meyer *et al.* 2009) in order to enhance the understanding of mecanisms involved in the evolution of cancers.
Keywords : Partial least squares regression, generalized linear models, bootstrap, high dimensional data, R software package.

1 Introduction

Nous commençons par rappeler comment étendre la régression PLS au cas des modèles linéaires généralisés, puis nous proposons cinq exemples d'application de la régression PLS étendue aux modèles de régression logistique obtenus à l'aide de la bibliothèque de fonctions `plsRglm` disponible pour le logiciel R.

2 Régression PLS étendue aux modèles de régression linéaire généralisée

2.1 La régression PLS

Considérons les variables centrées y, x_1, ..., x_j, ..., x_p. Soit X la matrice des prédicteurs x_1, ..., x_j, ..., x_p. La régression PLS est bien connue et décrite de manière exhaustive notamment par Höskuldsson (1988) et Wold *et al.* (2001). La présentation classique de la régression PLS est sous forme algorithmique. Nous n'en rappellerons que les éléments utiles pour la suite. La régression PLS est un modèle non-linéaire qui permet de construire des composantes orthogonales t_h obtenues en maximisant les quantités $cov(y, t_h)$. Soit T la matrice formée de ces composantes, nous avons :

$$y = T^t c + \epsilon, \tag{1}$$

où ϵ est le vecteur des résidus et ${}^t c$ le vecteur des coefficients des composantes, ^{t} désignant la transposée.
En posant $T = XW^*$, où W^* est la matrice des coefficients des variables x_j dans chaque composante t_h, nous avons l'expression directe de la réponse y à l'aide des prédicteurs x_j :

$$y = XW^{*t} c + \epsilon. \tag{2}$$

En développant le membre de droite de l'équation (2), nous obtenons pour chaque composante y_i de y :

$$y_i = \sum_{h=1}^{H} \left(c_h w^*_{1h} x_{i1} + \cdots + c_h w^*_{ph} x_{ip} \right) + \epsilon_i, \tag{3}$$

H étant le nombre de composantes retenues dans le modèle final avec $H \leqslant \mathrm{rg}(X)$, H étant en général très inférieur au rang de X et p étant égal au nombre de variables contenues dans la matrice X. Les coefficients $c_h w^*_{jh}$, où $1 \leqslant j \leqslant p$, suivant la notation avec $*$ de Wold *et al.* (2001), traduisent la relation entre le vecteur y et les variables x_j à travers les composantes t_h.

2.2 Extension de la PLS aux modèles de régression linéaire généralisée

La régression PLS étendue aux modèles de régression linéaire généralisée de la réponse y sur les variables x_1, ..., x_j, ..., x_p avec H composantes $t_h = w^*_{1h} x_{i1} + \cdots + w^*_{ph} x_{ip}$ (Bastien *et al.*

2005) s'écrit :

$$g(\theta)_i = \sum_{h=1}^{H} \left(c_h \sum_{j=1}^{p} w_{jh}^* x_{ij} \right), \tag{4}$$

où le paramètre θ peut être soit une espérance soit le vecteur des probabilités d'une loi discrète de support fini. La fonction de lien g est déterminée en fonction de la distribution de y et de la qualité de l'ajustement du modèle au données. Les composantes PLS t_h sont orthogonales. L'algorithme permettant de déterminer les composantes PLS t_h d'un modèle PLS-GLM est le suivant :

- Calcul de la première composante PLS t_1 :
  1. Calculer le coefficient a_{1j} de x_j dans la régression linéaire généralisée de y sur x_j pour chaque prédicteur x_j, $1 \leqslant j \leqslant p$.
  2. Normer le vecteur colonne a_1 : $w_1 = a_1/\|a_1\|$.
  3. Calculer la composante $t_1 = 1/({}^t w_1 w_1) X w_1$.
- Calcul de la seconde composante PLS t_2 :
  1. Calculer le coefficient a_{2j} de x_j dans la régression linéaire généralisée de y sur t_1 et x_j pour chaque prédicteur x_j, $1 \leqslant j \leqslant p$.
  2. Normer le vecteur colonne a_2 : $w_2 = a_2/\|a_2\|$.
  3. Calculer la matrice résiduelle X_1 de la régression linéaire de X sur t_1.
  4. Calculer la composante $t_2 = 1/({}^t w_2 w_2) X_1 w_2$.
  5. Exprimer la composante t_2 en termes de prédicteurs X : $t_2 = X w_2^*$.
- Nous supposons construites les $h-1$ composantes $t_1, \ldots, t_{h-1}$.
  Calcul de la $h-$ème composante PLS t_h :
  1. Calculer le coefficient a_{hj} de x_j dans la régression linéaire généralisée de y sur t_1, t_2, $\ldots$, t_{h-1} et x_j pour chaque prédicteur x_j, $1 \leqslant j \leqslant p$.
  2. Normer le vecteur colonne a_h : $w_h = a_h/\|a_h\|$.
  3. Calculer la matrice résiduelle X_{h-1} de la régression linéaire de X sur $t_1, t_2, \ldots, t_{h-1}$.
  4. Calculer la composante $t_h = 1/({}^t w_h w_h) X_{h-1} w_h$.
  5. Exprimer la composante t_h en termes de prédicteurs X : $t_h = X w_h^*$.

Il est possible de modifier l'algorithme précédent pour pouvoir traiter les jeux de données incomplets (Bastien *et al.* 2005).

3 Bootstrap dans les modèles de régression PLS et PLS-GLM

3.1 Cas de la régression PLS

Nous supposons avoir retenu le nombre adéquat de composantes d'un modèle de régression PLS1 de Y sur $x_1, \ldots, x_j, \ldots, x_p$. Lazraq *et al.* (2003) ont proposé l'algorithme suivant pour

construire des intervalles de confiance et des tests de significativité pour les prédicteurs x_j, $1 \leqslant j \leqslant p$, d'un modèle de régression PLS1 à l'aide de techniques de bootstrap.
Commençons par quelques notations. Soit L le nombre d'échantillons bootstraps. Soit $w_l^{(b)}$: $\binom{Y_l^{(b)}}{X_l^{(b)}} = \left\{ \binom{Y_\alpha^{(b)}}{X_\alpha^{(b)}}, \alpha = 1, \ldots, n \right\}$ le $l-$ème échantillon bootstrap, tiré avec remise, de taille n, $l = 1, 2, \ldots, L$ avec $Y_\alpha^{(b)}$, de taille $1 \times n$, $X_\alpha^{(b)}$, de taille $p \times n$, et $w_\alpha^{(b)}$, de taille $(1+p) \times n$. Pour chaque échantillon bootstrap l, le résultat de la régression PLS1 de $Y_l^{(b)}$ sur $X_l^{(b)}$ est noté $B_l^{(b)}$ avec $B_l^{(b)} = (b_{1l}^{(b)}, b_{2l}^{(b)}, \ldots, b_{pl}^{(b)})$ de taille $1 \times p$.

Étape 1. Répéter pour $l = 1, 2, \ldots, L$.

1. Tirer, avec remise, un échantillon de taille n : $w_l^{(b)} = \binom{Y_l^{(b)}}{X_l^{(b)}}$.
2. Calculer $B_l^{(b)}$ le résultat de la régression PLS1 de $Y_l^{(b)}$ sur $X_l^{(b)}$.

Étape 2. Répéter pour $j = 1, 2, \ldots, p$.

1. Soit E_j le vecteur $(b_{j1}^{(b)}, b_{j2}^{(b)}, \ldots, b_{jL}^{(b)})$ de taille $1 \times L$, où E_j est un échantillon bootstrap de taille L de b_j, le coefficient de x_j, le $j-$ème prédicteur, dans la régression PLS1 de Y sur X.
2. Obtenir un intervalle de confiance $I_j^{(b)}$ pour b_j. Plusieurs constructions sont possibles : normaux, percentiles ou BC_a (Efron et Tibshirani 1993 ou Davison et Hinkley 1997). Les intervalles ainsi obtenus ne sont pas conçus pour servir à réaliser des comparaisons multiples ou deux à deux et doivent être interprétés séparément.
3. Si $0 \in I_j^{(b)}$, supprimer la variable x_j.

Étape 3. Renvoyer la liste des prédicteurs significatifs.

3.2 Cas de la régression PLS-GLM

Nous supposons avoir retenu le nombre m adéquat de composantes d'un modèle de régression PLS-GLM de Y sur x_1, ..., x_j, ..., x_p. Bastien *et al.* (2005) ont proposé l'algorithme suivant pour construire des intervalles de confiance et des tests de significativité pour les prédicteurs x_j, $1 \leqslant j \leqslant p$, à l'aide de techniques de bootstrap.
Soit $\widehat{F}_{(T|y)}$ la fonction de répartition empirique étant données la matrice T formées des m composantes PLS et la réponse y.

Étape 1. Tirer B échantillons de $\widehat{F}_{(T|y)}$.

Étape 2. Pour tout $b = 1, \ldots, B$, calculer :

$$c^{(b)} = ({}^tT^{(b)}T^{(b)})^{-1}{}^tT^{(b)}y^{(b)} \quad \text{et} \quad b^{(b)} = W^* c^{(b)},$$

où $[T^{(b)}, y^{(b)}]$ est le $b-$ème échantillon bootstrap, $c^{(b)}$ est le vecteur des coefficients des composantes et $b^{(b)}$ est le vecteur des coefficients des p prédicteurs d'origine pour cet échantillon et enfin W^* est la matrice fixe des poids des prédicteurs dans le modèle d'origine comportant m composantes.

Étape 3. Pour chaque j, notons Φ_{b_j} l'approximation de Monte-Carlo de la fonction de répartition de la statistique bootstrap de b_j.
Pour chaque b_j, des boîtes à moustaches et des intervalles de confiance peuvent être construits à l'aide des percentiles de Φ_{b_j}. Un intervalle de confiance peut être défini par $I_j(\alpha) = \Phi_{b_j}^{-1}(\alpha), \Phi_{b_j}^{-1}(1-\alpha)$ où $\Phi_{b_j}^{-1}(\alpha)$ et $\Phi_{b_j}^{-1}(1-\alpha)$ sont les valeurs obtenues à partir de la fonction de répartition de la statistique bootstrap de telle sorte qu'un niveau nominal de confiance de niveau $100(1-2\alpha)\%$ soit atteint. Afin d'améliorer la qualité de l'intervalle de confiance en termes de taux de couverture, c'est-à-dire la capacité de $I_j(\alpha)$ à fournir les taux de couverture attendus, il est possible d'utiliser plusieurs techniques de construction : normale, percentile ou BC_a (Efron et Tibshirani 1993 ou Davison et Hinkley 1997). Les intervalles ainsi obtenus ne sont pas conçus pour servir à réaliser des comparaisons multiples ou deux à deux et doivent être interprétés séparément.

4 Points forts de l'implémentation

La bibliothèque de fonctions `plsRglm` possède plusieurs points forts.

- Modèles de régression PLS1 et PLS-GLM avec des données complètes ou incomplètes.
- Choix du nombre de composantes grâce à différents critères AIC, BIC, arrêt de significativité de la composante t_{m+1} lorsqu'aucun des coefficients a_{m+1} n'est plus significatif dans le modèle (Bastien *et al.* 2005) ou en utilisant un critère Q^2 (Bastien *et al.* 2005) ou le nombre de mal classés tous les deux estimés par validation croisée.
- Validation croisée « repeated k-fold cross-validation » avec des données complètes ou incomplètes.
- Bootstrap des coefficients des prédicteurs pour des modèles PLS1 (Lazraq *et al.* 2003) et PLS-GLM (Bastien *et al.* 2005) avec des données complètes ou incomplètes. Différentes constructions d'intervalles, détaillées dans Efron et Tibshirani (1993) ou Davison et Hinkley (1997), sont disponibles et reposent sur la bibliothèque de fonction `boot` (Canty et Ripley 2009).

5 Application à cinq exemples

Nous proposerons des exemples d'application à cinq jeux de données classiques.

5.1 Vins de Bordeaux (`Bordeaux`)

Une régression PLS multinomiale sera appliquée au jeu de données sur la qualité des vins de Bordeaux étudié dans (Bastien *et al.* 2005), jeu de données `Bordeaux` de la bibliothèque `plsRglm`. Il s'agit d'un exemple classique d'application de la PLS-GLM.

5.2 Microarray (Cancer du colon)

Alon et al. (1999) ont analysé 62 échantillons (40 d'une tumeur, 22 d'une partie saine) prélevés dans le colon de 62 patients atteints du cancer du colon. 2000 parmi les 6500 gènes exprimés ont été sélectionnés par les auteurs (Alon *et al.* 1999), jeu de données `colonCA` de la bibliothèque

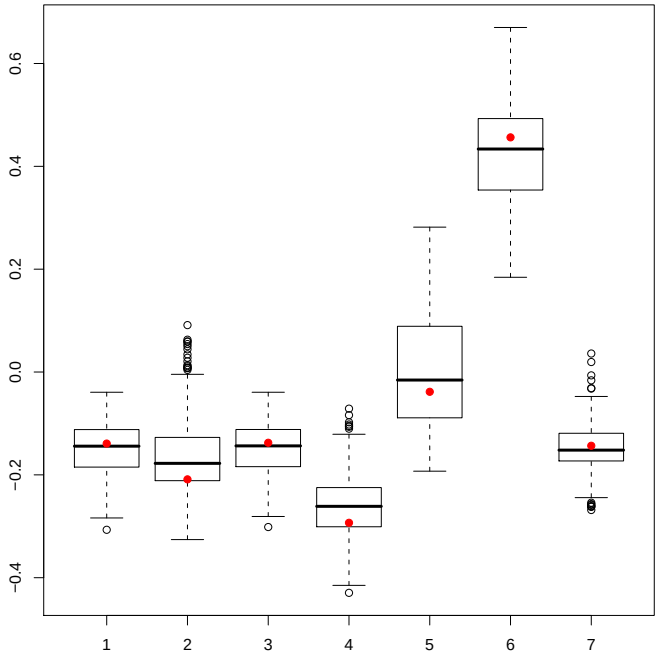

FIGURE 1 – Distribution bootstrap des coefficients centrés-réduits b_j, $(j = 1, \ldots, 7)$.

`colonCA`. Nous utilisons les fonctions de la bibliothèque `plsRglm` pour ajuster un modèle PLS-logistique qui permettra de modéliser la probabilité qu'un tissu soit sain ou cancéreux à l'aide de gènes bien choisis.

5.3 Allélotypage (`aze_compl`)

Meyer *et al.* (2009) ont appliqué la régression PLS-logistique binaire avec lien Logit à des données d'allélotypage.

6 Conclusion et perspectives

Notre objectif a été de mettre à la disposition des utilisateurs du logiciel libre R l'extension de la régression PLS au cas des modèles linéaires généralisés qui permet ainsi de faire bénéficier les régressions logistiques, aussi bien binaire, qu'ordinale ou multinomiale, des points forts de la régression PLS. En premier lieu, il s'agit de la possibilité de travailler avec des prédicteurs colinéaires, difficulté inévitable dans le cas de la modélisation des mélanges ou lors de l'analyse de spectres, de l'étude de données génétiques, protéomiques ou métabonomiques. En second lieu, la régression PLS, lorsqu'elle est réalisée par exemple avec l'algorithme NIPALS, peut être appliquée à des jeux de données incomplets.

Une seconde thématique concerne le traitement des jeux de données incomplets. La bibliothèque de fonctions plsRglm vise à palier à certains manques des bibliothèques de fonctions existantes concernant le traitement des jeux de données incomplets à l'aide de la régression PLS classique. Dans ce cas, par exemple, et contrairement au logiciel SIMCA (Wold *et al.* 2001), aucune bibliothèque de fonctions dans le logiciel R ne propose pour le moment la sélection du

nombre de composantes par validation croisée. Nous avons donc implémenté des fonctions permettant de choisir le nombre de composantes de régressions PLS classiques ou de régressions PLS-GLM par validation croisée « repeated k-fold cross-validation » dans toutes les situations.

Enfin, nous avons complété la bibliothèque par des techniques bootstrap (Lazraq *et al.* 2003, Bastien *et al.* 2005) pour par exemple tester la significativité de chacun des prédicteurs présents dans le jeu de données et ainsi valider les modèles construits.

Bibliographie

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D. & Levine A. J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissue probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA*, 96, 6745-6750.

Bastien, Ph., Esposito Vinzi, V., & Tenenhaus, M. (2005). PLS generalised linear regression. *Computational Statistics & Data Analysis*, 48(1), 17-46.

Canty, A., & Ripley, B. (2009). `boot`*: Bootstrap R (S-Plus) Functions.* R package version 1.2-37.

Davison, A.C., & Hinkley, D.V. (1997). *Bootstrap methods and their application.* Cambridge: Cambridge University Press.

Efron, B., & Tibshirani, R.J. (1993). *An Introduction to the Bootstrap.* New York: Chapman & Hall.

Eriksson, L., Johansson, E., Kettaneh-Wold, N., Trygg, J., Wikström, C., & Wold, S. (2001). *Multi- and Megavariate Data Analysis, Principles and Applications.* Umeå: Umetrics Academy.

Höskuldsson, A. (1988). PLS regression methods. *Journal of Chemometrics*, 2, 211-228.

Kettaneh-Wold, N. (1992). Analysis of mixture data with partial least squares. *Chemometrics & Intelligent Laboratory Systems*, 14, 57-69.

Lazraq, A., Cléroux, R., Gauchi, J.-P. (2003). Selecting both latent and explanatory variables in the PLS1 regression model. *Chemometrics and Intelligent Laboratory Systems*, 66, 117-126.

Meyer, N., Maumy-Bertrand, M. & Bertrand, F. (2009). Comparaison de variantes de régressions logistiques PLS et de régression PLS sur variables qualitatives : application aux données d'allélotypage. *Prépublication de l'IRMA.*

R Development Core Team (2008). *A language and environment for statistical computing.* Vienna: R Foundation for Statistical Computing. `http://www.R-project.org`.

Wold, S., Sjöström, M., & Eriksson, L. (2001). PLS-regression: a basic tool of Chemometrics. *Chemometrics and Intelligent Laboratory Systems*, 58, 109-130.