

PLS regression methods & extended tools

Philippe Bastien

L'Oreal Research, Clichy, France (pbastien@rd.loreal.com)



Régression et multicolinéarité

La multicolinéarité se caractérise par:

- Une variabilité très grande des coefficients de régression
- Une incohérence des signes des coefficients instables d'un échantillon à l'autre
- Des modèles de régression qui fit bien l'échantillon d'apprentissage mais qui se généralisent très mal.

Régression et multicollinéarité

$$y = X\beta + \varepsilon$$

$$\hat{\beta}_{OLS} = \arg \min \|y - X\beta\|^2$$

$$\text{Si } X'X \text{ inversible} \rightarrow \hat{\beta}_{OLS} = (X'X)^{-1} X'y$$

$$\text{var}(\hat{\beta}_{OLS}) = \sigma^2 (XX')^{-1}$$

$$C = (X'X)^{-1}$$

$$C_{jj} = \frac{1}{1 - R_j^2} \rightarrow \infty \text{ si } R_j^2 \rightarrow 1$$

$$\text{var}(\hat{\beta}_{OLS}) \rightarrow \infty$$

Si $X'X$ est singulière comment l'inverser ?

➤ *1) Diagonaliser $X'X$*

$$X = U\Sigma V' = \sum_{i=1}^r \sqrt{\lambda_i} u_i v'_i$$

$$X = \sqrt{\lambda_1} \begin{matrix} | \\ | \\ | \end{matrix} \begin{matrix} \text{---} \\ \text{---} \\ \text{---} \end{matrix} + \dots + \sqrt{\lambda_r} \begin{matrix} | \\ | \\ | \end{matrix} \begin{matrix} \text{---} \\ \text{---} \\ \text{---} \end{matrix}$$

$u_1 \quad v'_1 \quad \quad \quad u_r \quad v'_r$

$$(X'X)^- = V\Sigma^{-2}V'$$

$$\hat{\beta} = (X'X)^- X'y = V\Sigma^{-2}V'V\Sigma U'y = V\Sigma^{-1}U'y$$

Inverse généralisé de Moore-Penrose, solution de norme minimum

→ PCR

Si $X'X$ est singulière comment l'inverser ?

2) *Tridiagonaliser $X'X$*

Méthode de Lanczos (1950) pour l'approximation des plus grandes valeurs propres de matrices symétriques.

Soit b un vecteur et A une matrice symétrique, Lanczos construit une séquence de matrices tridiagonales $W_j^T A W_j$

Les vecteurs qui composent W_j sont construits par orthogonalisation de Gram-Schmidt de la série de Krylov:
 $K(b, A, j) = (b, Ab, A^2b, \dots, A^j b)$

En particulier la régression PLS correspond à: $b = X'y$ et $A = X'X$

Lanczos C.. (1950). An iterative method for the solution of the eigenvalue problem of linear differential and integral operators. J. Res. Nat. Bur. Standards, Sect B.

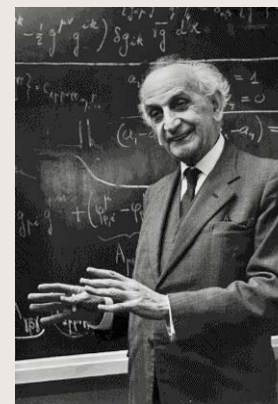
Si $X'X$ est singulière comment l'inverser ?

$$W_a = GS K(X'y, X'X, a) = GS(X'y, X'XX'y, \dots, (X'X)^{a-1} X'y)$$

$$\hat{\beta}_a^{PLS} = W_a (W_a' X'X W_a)^{-1} W_a' X'y$$

$$\hat{\beta}_a^{PLS} = W_a (W_a' A W_a)^{-1} W_a' A \hat{\beta}^{OLS}$$

Projection A orthogonale de $\hat{\beta}^{OLS}$ sur $\text{vect}(W_a)$



Cornelius Lanczos (1893-1974)

mathématicien et physicien hongrois

Lanczos C.. (1950). An iterative method for the solution of the eigenvalue problem of linear differential and integral operators. J. Res. Nat. Bur. Standards, Sect B.

Si $X'X$ est singulière comment l'inverser ?

3) Une factorisation particulière de X

$X = URW'$ avec $U'U = W'W = I$ et R inversible

$$\hat{\beta} = X^{-}y = WR^{-1}U'y$$

L'algorithme Bidiag2 de Golub et Kahan (1965) construit une matrice R bidiagonale à droite et donc facilement inversible.

$$u_0 = 0$$

$$w_1 = X'y / \|X'y\|$$

$$u_i = k_i[Xw_i - u_{i-1}(u'_{i-1}Xw_i)], \quad i \geq 1$$

$$w_{i+1} = k'_i[X'u_i - w_i(w'_iX'u_i)], \quad i \geq 1$$

Golub G. and Kahan W. (1965). Calculating the singular values and pseudo-inverse of a matrix, SIAM Journal on Numerical Analysis, 2: 205-224.

Exemple: Régression et multicollinéarité

Obs	Obs	AUTO	CYL	PUI	LON	LAR	POIDS	VITESSE	PRIX
1	1	ALFASUD-TI-1350	1350	79	393	161	870	165	30570
2	2	AUDI-100-L	1588	85	468	177	1110	160	39990
3	3	SIMCA-1307-GLS	1294	68	424	168	1050	152	29600
4	4	CITROEN-GS-CLUB	1222	59	412	161	930	151	28250
5	5	FIAT-132-1600GLS	1585	98	439	164	1105	165	34900
6	6	LANCIA-BETA-1300	1297	82	429	169	1080	160	35480
7	7	PEUGEOT-504	1796	79	449	169	1160	154	32300
8	8	RENAULT-16-TL	1565	55	424	163	1010	140	32000
9	9	RENAULT-30-TS	2664	128	452	173	1320	180	47700
10	10	TOYOTA-COROLLA	1166	55	399	157	815	140	26540
11	11	ALFETTA-1-66	1570	109	428	162	1060	175	42395
12	12	PRINCESS-1800-HL	1798	82	445	172	1160	158	33990
13	13	DATSUN-200L	1998	115	469	169	1370	160	43980
14	14	TAUNUS-2000-GL	1993	98	438	170	1080	167	35010
15	15	RANCHO	1442	80	431	166	1129	144	39450
16	16	MAZDA-9295	1769	83	440	165	1095	165	27900
17	17	OPEL-RKORD-L	1979	100	459	173	1120	173	32700
18	18	LADA-1300	1294	68	404	161	955	140	22100

Exemple: Régression et multicollinéarité

	PRIX	CYL	PUI	LON	LAR	POIDS	VITESSE
PRIX	1.00000	0.63858 0.0043	0.79870 <.0001	0.64376 0.0039	0.54665 0.0189	0.75329 0.0003	0.58176 0.0113
CYL	0.63858 0.0043	1.00000	0.79663 <.0001	0.70146 0.0012	0.62976 0.0051	0.78895 <.0001	0.66493 0.0026
PUI	0.79870 <.0001	0.79663 <.0001	1.00000	0.64136 0.0041	0.52083 0.0267	0.76529 0.0002	0.84438 <.0001
LON	0.64376 0.0039	0.70146 0.0012	0.64136 0.0041	1.00000	0.84927 <.0001	0.86809 <.0001	0.47593 0.0459
LAR	0.54665 0.0189	0.62976 0.0051	0.52083 0.0267	0.84927 <.0001	1.00000	0.71687 0.0008	0.47295 0.0475
POIDS	0.75329 0.0003	0.78895 <.0001	0.76529 0.0002	0.86809 <.0001	0.71687 0.0008	1.00000	0.47760 0.0450
VITESSE	0.58176 0.0113	0.66493 0.0026	0.84438 <.0001	0.47593 0.0459	0.47295 0.0475	0.47760 0.0450	1.00000

Toutes les variables apparaissent significatives et corrélées positivement avec le prix.

Exemple: Régression et multicolinéarité

Procédure REG
Modèle : MODEL1
Variable dépendante : PRIX

Nombre d'observations lues 18
Nombre d'observations utilisées 18

Analyse de variance

Source	DDL	Somme des carrés	Moyenne quadratique	Valeur F	Pr > F
Modèle	6	520591932	86765322	4.47	0.0156
Erreur	11	213563858	19414896		
Total sommes corrigées	17	734155790			

Root MSE 4406.23379 R carré 0.7091
Moyenne dépendante 34159 R car. ajust. 0.5504
Coeff Var 12.89934

Valeurs estimées des paramètres

Variable	DDL	Valeur estimée des paramètres	Erreur type	Valeur du test t	Pr > t
Intercept	1	-8239.36268	42718	-0.19	0.8506
CYL	1	-3.50518	5.55060	-0.63	0.5406
PU1	1	282.16880	174.88297	1.61	0.1349
LON	1	-15.03766	129.74749	-0.12	0.9098
LAR	1	208.69377	412.04788	0.51	0.6225
POIDS	1	12.57468	24.62219	0.51	0.6197
VITESSE	1	-111.11355	222.25657	-0.50	0.6270

Exemple: Régression et multicolinéarité : PLSR

The PLS Procedure

Variation en pourcentage expliquée
par Partial Least Squares Factors

Nombre de facteurs extraits	Effets du modèle		Variables dépendantes	
	Actuel	Total	Actuel	Total
1	73.6230	73.6230	60.8374	60.8374

Parameter Estimates for Centered and Scaled Data

PRIX

Intercept	0.0000000000
CYL	0.1457852413
PUI	0.1823397520
LON	0.1469668392
LAR	0.1247976334
POIDS	0.1719738622
VITESSE	0.1328131564

Parameter Estimates

PRIX

Intercept	-39940.36629
CYL	2.56208
PUI	58.80660
LON	43.68699
LAR	154.34048
POIDS	8.25174
VITESSE	71.89164

Exemple: Régression et multicolinéarité

Obs	Obs	AUTO	CYL	PUI	LON	LAR	POIDS	VITESSE	PRIX	Pred
1	1	ALFASUD-TI-1350	1350	79	393	161	870	165	30570	29223
2	2	AUDI-100-L	1588	85	468	177	1110	160	39990	37553
3	3	SIMCA-1307-GLS	1294	68	424	168	1050	152	29600	31418
4	4	CITROEN-GS-CLUB	1222	59	412	161	930	151	28250	28038
5	5	FIAT-132-1600GLS	1585	98	439	164	1105	165	34900	35354
6	6	LANCIA-BETA-1300	1297	82	429	169	1080	160	35480	33445
7	7	PEUGEOT-504	1796	79	449	169	1160	154	32300	35649
8	8	RENAULT-16-TL	1565	55	424	163	1010	140	32000	29384
9	9	RENAULT-30-TS	2664	128	452	173	1320	180	47700	44692
10	10	TOYOTA-COROLLA	1166	55	399	157	815	140	26540	24734
11	11	ALFETTA-1-66	1570	109	428	162	1060	175	42395	35521
12	12	PRINCESS-1800-HL	1798	82	445	172	1160	158	33990	36407
13	13	DATSUN-200L	1998	115	469	169	1370	160	43980	41322
14	14	TAUNUS-2000-GL	1993	98	438	170	1080	167	35010	37219
15	15	RANCHO	1442	80	431	166	1129	144	39450	32577
16	16	MAZDA-9295	1769	83	440	165	1095	165	27900	35059
17	17	OPEL-RKORD-L	1979	100	459	173	1120	173	32700	39443
18	18	LADA-1300	1294	68	404	161	955	140	22100	27817

Exemple: Régression et multicollinéarité: PLSR

The PLS Procedure

Variation en pourcentage expliquée
par Partial Least Squares Factors

Nombre de facteurs extraits	Effets du modèle		Variables dépendantes	
	Actuel	Total	Actuel	Total
1	73.6230	73.6230	60.8374	60.8374
2	9.0379	82.6609	6.2420	67.0794
3	9.2571	91.9181	2.3752	69.4546
4	4.6704	96.5885	0.9020	70.3566
5	2.4807	99.0691	0.4523	70.8090
6	0.9309	100.0000	0.1013	70.9103

Parameter Estimates for Centered and Scaled Data

PRIX

Intercept	0.0000000000
CYL	-.1994486089
PUI	0.8749117597
LON	-.0505879950
LAR	0.1687469748
POIDS	0.2620680197
VITESSE	-.2052720153

Parameter Estimates

PRIX

Intercept	-8239.362677
CYL	-3.505182
PUI	282.168803
LON	-15.037660
LAR	208.693773
POIDS	12.574678
VITESSE	-111.113551

Exemple: Régression et données manquantes

Obs	Obs	AUTO	CYL	PUI	LON	LAR	POIDS	VITESSE	PRIX
1	1	ALFASUD-TI-1350	.	79	393	161	870	165	30570
2	2	AUDI-100-L	1588	.	468	177	1110	160	39990
3	3	SIMCA-1307-GLS	1294	68	.	168	1050	152	29600
4	4	CITROEN-GS-CLUB	1222	59	412	.	930	151	28250
5	5	FIAT-132-1600GLS	1585	98	439	164	.	165	34900
6	6	LANCIA-BETA-1300	1297	82	429	169	1080	.	35480
7	7	PEUGEOT-504	1796	79	449	169	.	154	32300
8	8	RENAULT-16-TL	1565	55	424	.	1010	140	32000
9	9	RENAULT-30-TS	2664	128	.	173	1320	180	47700
10	10	TOYOTA-COROLLA	1166	.	399	157	815	140	26540
11	11	ALFETTA-1-66	.	109	428	162	1060	175	42395
12	12	PRINCESS-1800-HL	1798	.	445	172	1160	158	33990
13	13	DATSUN-200L	1998	115	.	169	1370	160	43980
14	14	TAUNUS-2000-GL	1993	98	438	.	1080	167	35010
15	15	RANCHO	1442	80	431	166	.	144	39450
16	16	MAZDA-9295	1769	83	440	165	1095	.	27900
17	17	OPEL-RKORD-L	1979	100	459	173	.	173	32700
18	18	LADA-1300	1294	68	404	.	955	140	22100

```
228 proc pls data=lib.voitureswithmissing METHOD=PLS (ALGORITHM=NIPALS)CV=BLOCK details;
229     model PRIX=CYL PUI LON LAR POIDS VITESSE/solution;
230 run;
```

ERROR: No valid observations.

NOTE: 18 observation(s) lue(s) dans la table LIB.VOITURESWITHMISSING.

NOTE: Procédure PLS a utilisé (Durée totale du traitement) :

temps réel 0.00 secondes
temps UC 0.00 secondes

Exemple: Régression et multicollinéarité

PLS-NIPALS

Var ID	Coeff (PRIX)
constant	-33851,4
CYL	2,6477
PUI	56,8296
LON	44,5767
LAR	122,162
POIDS	6,98538
VITESSE	73,6172

Régularisation

Nécessité de régularisation afin d'obtenir un modèle prédictif stable

- *Régularisation par réduction de la dimensionnalité: PCR/PLS*

- *Régularisation par pénalisation L1 (Lasso), L2 (Ridge), L1&L2 (Elastic net)*

Remarques:

- *Ridge = OLS stabilisé*

- *Elastic net = Lasso stabilisé*

Régularisation et biais

- Théorème de Gauss-Markov:



Gauss : 1777-1855

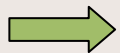


Markov : 1856-1922

$\hat{\beta}_{OLS}$ est de tous les estimateurs sans biais celui de variance minimale (BLUE)



On ne peut diminuer la variance qu'en biaisant l'estimateur



Compromis biais-variance

Compromis biais-variance

$$\begin{aligned}MSE(\hat{\beta}) &= E \left[(\hat{\beta} - \beta)' (\hat{\beta} - \beta) \right] \\&= (E(\hat{\beta}) - \beta)' (E(\hat{\beta}) - \beta) + E \left[(\hat{\beta} - E(\hat{\beta}))' (\hat{\beta} - E(\hat{\beta})) \right] \\&= \text{biais}^2 + \text{variance}(\hat{\beta})\end{aligned}$$

Quel est le biais de PLS ?

Régression biaisée

$$\hat{\beta}_{OLS} = \sum_{i=1}^r \frac{u_i' y}{\sqrt{\lambda_i}} v_i = \sum_{i=1}^r \hat{\beta}_i$$

$$\hat{\beta}_{\text{biaisé}} = \sum_{i=1}^r f(\lambda_i) \hat{\beta}_i$$

$f(\lambda_i) < 1 \rightarrow$ diminution de la variance de la $i^{\text{ème}}$ composante

Régression ridge et biais

$$\hat{\beta}_{RR} = \arg \min_{\beta} \|y - X\beta\|^2 + \lambda \|\beta\|^2 \rightarrow \hat{\beta}_{RR} = (X'X + \lambda I)^{-1} X'y$$

$$\hat{\beta}_{RR} = \sum_{i=1}^r \frac{\lambda_i}{\lambda_i + \lambda} \hat{\beta}_i \quad f(\lambda_i) < 1 \quad \hat{\beta}_{RR} = \frac{1}{1 + \lambda} \begin{pmatrix} 1 & \frac{\rho_{12}}{1 + \lambda} & \frac{\rho_{13}}{1 + \lambda} \\ \frac{\rho_{21}}{1 + \lambda} & 1 & \frac{\rho_{23}}{1 + \lambda} \\ \frac{\rho_{31}}{1 + \lambda} & \frac{\rho_{32}}{1 + \lambda} & 1 \end{pmatrix}^{-1} X'y$$

Remarque: Un effet subtil de la ridge est de contraindre à tendre vers des valeurs voisines les coefficients associés à des variables corrélées. On retrouve cet effet avec PLS.

Régression PLS et biais

$$\hat{\beta}_{PLS} = \sum_{i=1}^r \left[1 - \prod_{j=1}^p \left(1 - \frac{\lambda_i}{\mu_j} \right) \right] \hat{\beta}_i$$

$\mu_j =$ vecteur de Ritz

$=$ approximation de λ_j au sous-espace de Krylov

$\text{span} \{ X'y, X'XX'y, \dots, (X'X)^p X'y \}$

Relation non claire en terme de « shrinkage » car

$|f(\lambda_i)|$ peut aussi être > 1 !

NIPALS (Wold H., 1966)



- L'algorithme de régression PLS classique, comme cas particulier de l'approche PLS, est basé sur des développements de l'algorithme NIPALS, acronyme de Nonlinear estimation by Iterative Partial Least Squares.
- L'algorithme NIPALS est une méthode robuste pour déterminer de façon itérative les valeurs et vecteurs propres d'une matrice.
- Elle permet en particulier de faire de l'analyse en composantes principales en présence de valeurs manquantes.
- Cet algorithme est directement inspiré de la méthode de la puissance itérée (Hotelling 1936)

Wold H. Estimation of principal components and related models by iterative least squares. *In Krishainaah, P.R. (ed), Multivariate Analysis*. New Academic Press, New York 1966, pp. 391-420.

Méthode de la puissance itérée (Hotelling, 1936)

- On initialise l'algorithme avec un vecteur w aléatoire, et on itère jusqu'à convergence :

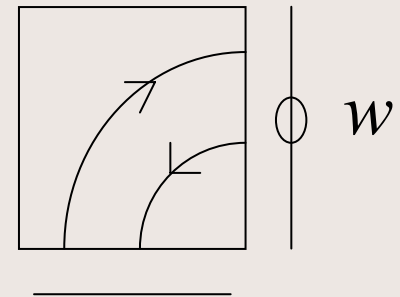
$$(1) \begin{cases} w = X'Xw \\ w = w / \|w\| \end{cases}$$

On obtient ainsi un vecteur w_1 vecteur propre de $X'X$ associé à la plus grande valeur propre λ_1^2

En effet, soit un vecteur b quelconque que l'on décompose dans la base canonique des vecteurs propres (w_1, \dots, w_r) sous la forme $b = \sum_i \alpha_i w_i$

$$(X'X)^n b = \sum_i \alpha_i \lambda_i^{2n} w_i \rightarrow \alpha_1 \lambda_1^{2n} w_1 \propto w_1$$

avec $\lambda_1 > \lambda_2 > \dots > \lambda_r$



Méthode de la puissance itérée (Hotelling, 1936)

- Lorsque l'algorithme converge, on a la relation :

$$X'Xw_1 = \lambda_1^2 w_1$$

A partir de laquelle on dérive l'expression de : $\lambda_1^2 = \frac{\|w_1^{(iteration\ n+1)}\|}{\|w_1^{(iteration\ n)}\|}$

On déflate ensuite la matrice $X'X$:

$$(X'X)^1 = X'X - \lambda_1^2 w_1 w_1'$$

$$(X'X)^i = X'X - \sum_{j=1}^i \lambda_j^2 w_j w_j'$$

Cette méthode extrait ainsi les vecteurs propres l'un après l'autre dans l'ordre décroissant des valeurs propres.

Méthode de la puissance itérée: représentation géométrique

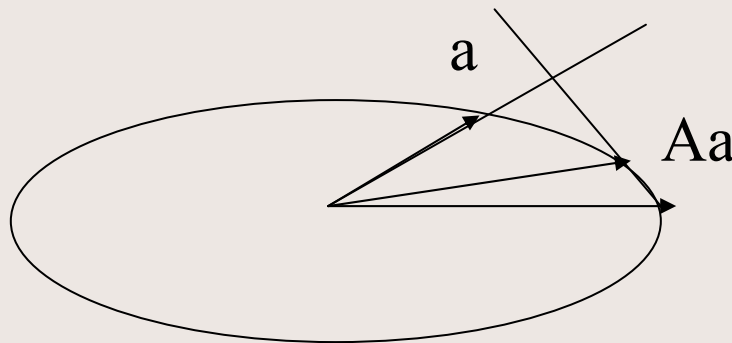
Soit $A(p \times p)$ une matrice symétrique ex : $A = X'X$

$A_c = \{z / z'Az = c\} = \text{ellipsoïde en } p \text{ dimension}$

$b = Aa = \text{rotation tangente}$

$b = \text{point de tangence entre l'ellipsoïde } A_c$

et l'hyperplan \perp à a



Algorithme NIPALS

- C'est un algorithme inspiré de la méthode de la puissance itérée appliqué à la matrice X , où l'on a remplacé le produit matrice vecteur par des régressions simples.

1) $X_0 = X$

2) pour $h = 1, a$

2.1) $t_h = \text{colonne de } X_{h-1}$

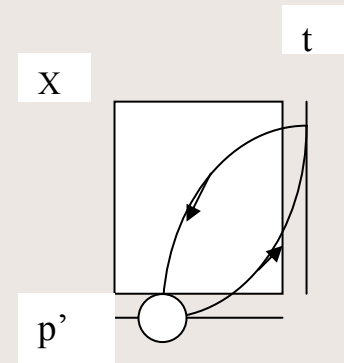
2.2) répéter jusqu'à convergence de p_h

2.2.1) $p_h = X'_{h-1}t_h / t'_h t_h$

2.2.2) $p_h = p_h / \|p_h\|$

2.2.3) $t_h = X_{h-1}p_h / p'_h p_h$

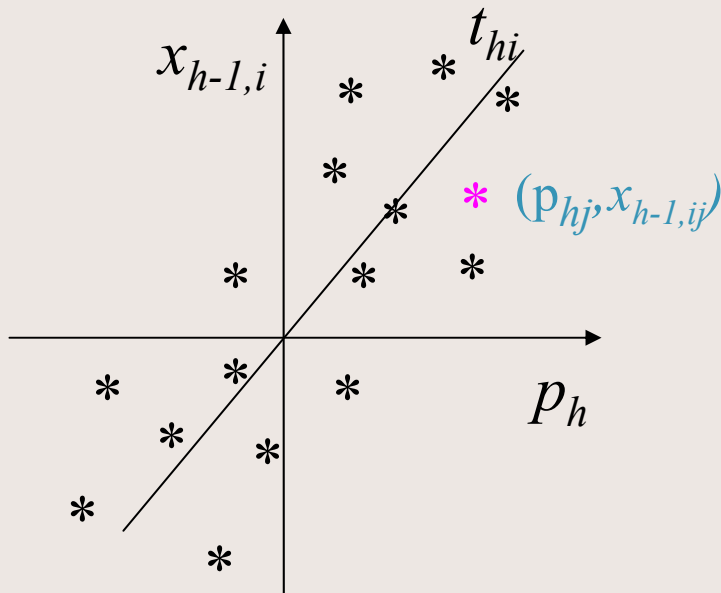
2.3) $X_h = X_{h-1} - t_h p'_h$



NIPALS et données manquantes

L'algorithme peut fonctionner en présence de données manquantes.

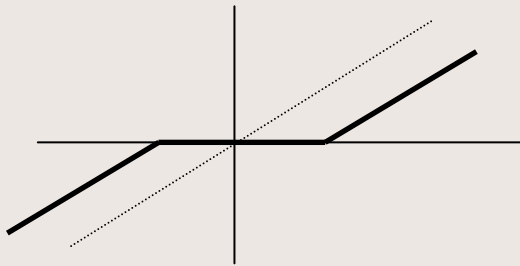
$$t_{hi} = \frac{x_{h-1,i} p_h}{p_h' p_h}$$



Sparse PCA

Soft thresholding function

$$g_{\lambda}(x) = (|x| - \lambda)_+ \text{sig}(x)$$



Sparse PCA : Algorithme

Décompose $X=U\Sigma V'$, $X_0 = X$

Pour $h = 1, H$

1) $v_{old} = \delta_h v_h$, $u_{old} = u_h$

2) Jusqu'à convergence de u_{new} et v_{new}

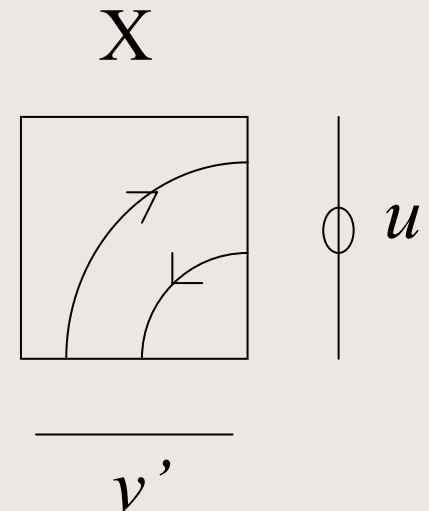
a) $v_{new} = g_\lambda (X'_{h-1} u_{old})$

b) $u_{new} = X_{h-1} v_{new} / \|X_{h-1} v_{new}\|$

c) $u_{old} = u_{new}$

3) $v_{new} = v_{new} / \|v_{new}\|$

4) $X_h = X_{h-1} - \delta_h u_{new} v'_{new}$



Algorithme PLS1-NIPALS

1) X, y centrés

2) pour $h = 1, a$

$$2.1) w_h = X'_{h-1}y / y'y$$

$$2.2) \|w_h\| = 1$$

$$2.3) t_h = X_{h-1}w_h / w'_h w_h$$

$$2.4) p_h = X'_{h-1}t_h / t'_h t_h$$

$$2.5) X_h = X_{h-1} - t_h p'_h$$

3) $T = [t_1, \dots, t_a], \hat{y}_{PLS} = T(T'T)^{-1}T'y$

$$\hat{y}^{PLS} = \sum_{i=1}^a \frac{y'_i t_i}{t'_i t_i} t_i = \sum_{i=1}^a \frac{y'_i t_i}{t'_i t_i} X_{i-1} w_i = \sum_{i=1}^a \frac{y'_i t_i}{t'_i t_i} X w_i^*$$

$$W^* = W(P'W)^{-1} \quad w_h^* = \left(\prod_{i=1}^{h-1} (I - w_i p'_i) \right) w_h$$

$$W = [w_1, \dots, w_a]$$

$$P = [p_1, \dots, p_a]$$

Sparse PLS: Algorithme

$$X'Y = U\Delta V'$$

$$1/ X_0 = X, Y_0 = Y$$

2/ pour $i = 1:H$

a) décompose $X'_{h-1}Y_{h-1}$ et extraire la première paire de vecteurs singuliers

$$u_{old} = u_h, v_{old} = v_h$$

b) jusqu'à convergence de u_{new} et v_{new}

$$i) u_{new} = g_{\lambda_2}(X'_{h-1}Y_{h-1}v_{old}), \|u_{new}\| = 1$$

$$ii) v_{new} = g_{\lambda_1}(Y'_{h-1}X_{h-1}u_{old}), \|v_{new}\| = 1$$

$$iii) u_{old} = u_{new}, v_{old} = v_{new}$$

$$c) \xi_h = X_{h-1}u_{new} / u'_{new}u_{new}$$

$$w_h = Y_{h-1}v_{new} / v'_{new}v_{new}$$

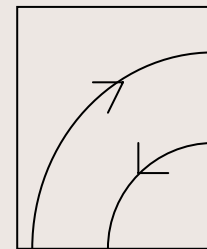
$$d) c_h = X'_{h-1}\xi_h / \xi'_h\xi_h$$

$$d_h = Y'_{h-1}\xi_h / \xi'_h\xi_h$$

$$e) X_h = X_{h-1} - \xi_h c'_h$$

$$f) Y_h = Y_{h-1} - \xi_h d'_h$$

$$X'_{h-1}Y_{h-1}$$



u

v

Régression PLS et très grande dimension

Kernel PLS

Quand $n \gg p$ ou $p \gg n$ des algorithmes rapides ont été proposés dans les années 90 consistant à effectuer l'essentiel des calculs sur des matrices de taille réduite appelées « kernel ».

$n \gg p \rightarrow w_1$ premier vecteur propre de $X'YY'X$ (pxp)

$p \gg n \rightarrow t_1$ premier vecteur singulier à gauche de $XX'YY'$ (nxn)

De Jong et ter Braak ont proposés lorsque $p \gg n$ d'effectuer la régression PLS sur la matrice des composantes principales Z , la régression PLS étant invariante par transformation orthogonale.

$$t_1 \prec XX'y = ZZ'y \rightarrow PLS(X) = PLS(Z)$$

Lindgren F., Geladi P., and Wold S., (1993). The kernel algorithm for PLS. *Journal of chemometrics*.

Rännar S., Geladi P., Lindgren F., Wold S., (1994). A PLS kernel algorithm for data sets with many variables and fewer objects. *Journal of Chemometrics*.

De Jong, and ter Braak C. (1994). Comments on the PLS kernel algorithm. *Journal of Chemometrics*.

Régression PLS et très grande dimension

Canonical PLS

Avec l'algorithme « Canonical Partial Least Squares » en 2001, De Jong et al. proposent d'effectuer l'essentiel des calculs dans la base formée par les composantes principales (base canonique).

$$\begin{aligned} T &= (t_1, t_2, \dots, t_a) = GS(XX'y, (XX')^2 y, \dots, (XX')^a y) \\ &= GS(ULU'y, UL^2U'y, \dots, UL^aU'y) \end{aligned}$$

$$\hat{\beta}_{PLS} = VL^{-1/2}U'TT'y$$

De Jong S., Wise B., and Ricker N., (2001). Canonical partial least squares and continuum power regression. Journal of Chemometrics,

De Jong, (1993). SIMPLS: an alternative approach to partial least squares regression. Chemometrics and Intelligent Laboratory Systems.

Régression PLS et très grande dimension

CANPLS

CANPLS revient à remplacer la déflation implicite de Canonical PLS résultant de l'orthogonalisation de Gram-Schmidt:

$$(t_1, t_2, \dots, t_a) = GS(ULU'y, UL^2U'y, \dots, UL^aU'y)$$

par une déflation explicite des vecteurs singuliers

$$(t_1, t_2, \dots, t_a) \propto (ULU'y, U_1LU'_1y, \dots, U_{a-1}LU'_{a-1}y)$$

CANPLS algorithm

let $X = UL^{1/2}V'$ the SVD of X (or X')

$$t_1 = ULU'y = \sum_i \lambda_i^2 (u_i'y) u_i$$
$$\|t_1\| = 1$$

$$U_{h-1} = (I - t_{h-1}t_{h-1}')U_{h-2}$$

$$t_h = U_{h-1}LU'_{h-1}y = \sum_i \lambda_i^2 (u_{i,h-1}'y) u_{i,h-1}$$
$$\|t_h\| = 1$$

$$\hat{\beta}_{PLS} = VL^{-1/2}U'TT'y$$

Modified Continuum power regression

Le compromis réalisé par PLS entre fit et stabilité qui donne le même poids aux deux termes peut être modifié et CanPLS peut être généralisé dans la même logique que la Continuum Power Regression de de Jong et al.

$$\tilde{t}_1 \propto UL^{\gamma}U' y = \sum_{i=1}^r \lambda_i^{\gamma} (u_i' y) u_i$$

$$\tilde{t}_k \propto U_{k-1} L^{\gamma} U'_{k-1} y = \sum_{i=1}^r \lambda_i^{\gamma} (u_{k-1,i}' y) u_{k,i}$$

Modified Continuum power regression

$$\tilde{t}_k \propto U_{k-1} L^\gamma U'_{k-1} y = \sum_{i=1}^r \lambda_i^\gamma (u'_{k-1,i} y) u_{k,i}$$

Définit une trajectoire entre OLS et PCR

OLS for $\gamma = 0$,

PLS for $\gamma = 1$,

PCR for $\gamma \rightarrow \infty$

PLS(X) = PLS(Z)

PCR(X) = PLS sur un sous ensemble de CP standardisés

Régression PLS et très grande dimension

Sample-based PLS

Bush et Nachbar (1993) ont montré que la régression PLS dépend de la distance entre les observations à travers le produit XX' plus que des valeurs individuelles des descripteurs.

Approche très intéressante lorsque les individus sont difficiles voire même impossibles à caractériser vectoriellement.

Bush R. and Nachbar B., (1993). J. Comput.-Aided Mol. Design

Lewi, P.J. (1995). Pattern recognition, reflection from a chemometric point of view. Chemometrics and Intelligent laboratory System

Sample-based PLS

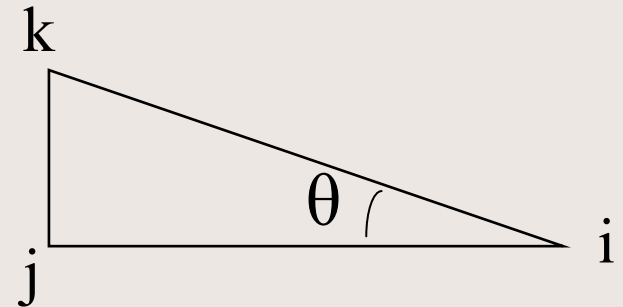
➤ *Torgerson (1952) détermine de manière analytique les coordonnées des points à partir de la distance entre ces points*

Loi du cosinus

$$\cos(\theta) = \frac{d_{ij}^2 + d_{ik}^2 - d_{jk}^2}{2d_{ij}d_{ik}}$$
$$b_{ijk} = d_{ij}d_{ik} \cos(\theta) = \frac{d_{ij}^2 + d_{ik}^2 - d_{jk}^2}{2}$$

Young & Householder 1938

$$B = UVU' = U\sqrt{V}(U\sqrt{V})' = XX'$$



Sample-based PLS

$$D_{ij}^2 = \sum_k (X_{ik} - X_{jk})^2 \Rightarrow C_{ij} = -1/2 (D_{ij}^2 - D_{i.}^2 - D_{.j}^2 + D_{..}^2)$$

$$C_{ij} = XX' \text{ si } X \text{ centré}$$

→ *SAMPLS algorithm*

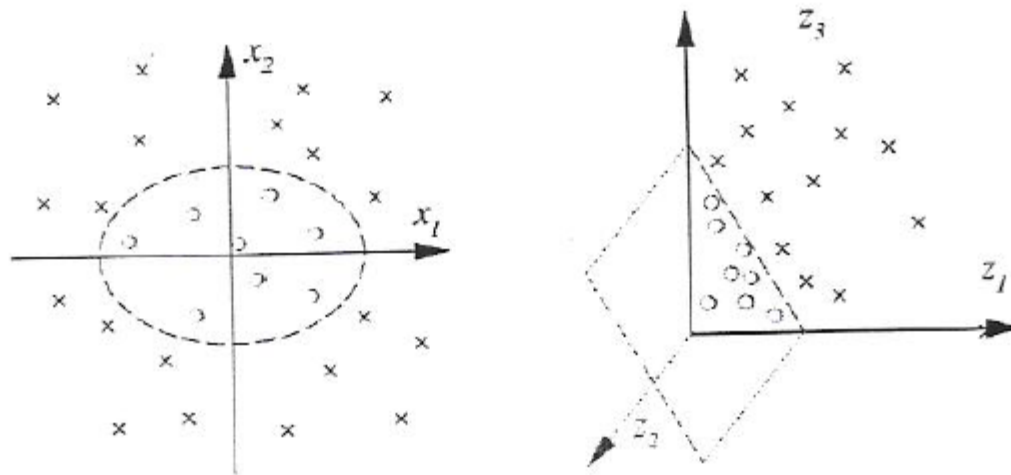
La généralisation à des distances non euclidiennes ouvre la voie vers le non-linéaire

Bush R. and Nachbar B., 1993. J. Comput.-Aided Mol. Design **1993**, 7, 587)

Kernel methods

$$l = 2, p = 3$$

$$\phi : (x_1, x_2) \rightarrow (x_1^2, \sqrt{2}x_1x_2, x_2^2)$$



En 2D : une surface de décision non linéaire est nécessaire pour séparer les données

En 3D : un hyperplan est suffisant.

Kernel methods

Soit $x = (x_1, x_2)$ et $\Phi(x) = (x_1^2, \sqrt{2}x_1x_2, x_2^2)$

$$\begin{aligned}\langle \Phi(x), \Phi(y) \rangle &= x_1^2x_2^2 + 2x_1x_2y_1y_2 + y_1^2y_2^2 \\ &= (x_1y_1 + x_2y_2)^2 = \langle x, y \rangle^2\end{aligned}$$

Noyau polynomial : $K(x, y) = (\langle x, y \rangle + C)^d$

Noyau Gaussien : $K(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right)$

Régression PLS et optimisation

La régression PLS peut aussi être vue comme un problème d'optimisation avec un critère à optimiser, le critère de Tucker.

$$T_1 = Xw_1 \text{ avec } w_1 \text{ solution de } \max_{w_1'w_1=1} (\text{cov}^2(Xw_1, y))$$

$$\text{cov}^2(Xw_1, y) \prec \text{var}(Xw_1)\text{cor}^2(Xw_1, y)$$

La solution w_1 est le vecteur propre normalisé de $X'y y' X$

$$X'y y' X X'y \propto X'y$$

$$w_1^j = \frac{\text{cov}(x_j, y)}{\sqrt{\sum_j \text{cov}^2(x_j, y)}} \longrightarrow t_1 = \frac{1}{\sqrt{\sum_{j=1}^p \text{cov}(y, x_j)^2}} \sum_{j=1}^p \text{cov}(y, x_j) x_j$$

Régression PLS généralisée

Une réinterprétation de la régression PLS

En 99, Michel Tenenhaus a proposé une interprétation des composantes PLS en fonction de simples régressions linéaires dans une approche similaire à celle proposée par Garthwaite (1994), mais qui permet la prise en compte des valeurs manquantes dans l'esprit de NIPALS

$$y = a_{1j} \frac{x_j}{\text{var}(x_j)} + \varepsilon \quad \longrightarrow \quad a_{1j} = \frac{\text{cov}(y, \frac{x_j}{\text{var}(x_j)})}{\text{var}(\frac{x_j}{\text{var}(x_j)})} = \text{cov}(y, x_j)$$

$$x_j = p_{1j}t_1 + p_{2j}t_2 + \dots + p_{h-1j}t_{h-1} + x_{h-1j}$$

$$y = c_1t_1 + c_2t_2 + \dots + c_{h-1}t_{h-1} + a_{hj} \frac{x_{h-1j}}{\text{var}(x_{h-1j})} + \varepsilon \quad \longrightarrow \quad a_{hj} = \text{cov}(y, x_{h-1j})$$

→ PLS GLR

Tenenhaus M. (1999). *La regression logistique PLS in Proceedings of the 32èmes journées de Statistique de la Société française de Statistique, Fes*

Garthwaite P.H. (1994). *An interpretation of Partial Least Squares. Journal of the American Statistical Association, 425:122-127*

PLS-GLR algorithm

The PLS-GLR algorithm consists of four steps :

- (1) Computation of the m PLS components t_h*
- (2) GL regression on the m retained PLS components*
- (3) Expression of PLS-GLR in terms of the original explanatory variables*
- (4) Bootstrap validation of coefficients in the final model*



Available online at www.sciencedirect.com
SCIENCE @ DIRECT®
Computational Statistics & Data Analysis 48 (2005) 17–46

COMPUTATIONAL
STATISTICS
& DATA ANALYSIS
www.elsevier.com/locate/jcsda

PLS generalised linear regression

Philippe Bastien^a, Vincenzo Esposito Vinzi^{b,c,*}, Michel Tenenhaus^c

^a*L'Oréal Recherche, Aubay, France*

^b*Department of Mathematics and Statistics University "Federico II", Naples, Italy*

^c*HEC, School of Management Jussieu-Jussieu, France*

Accepted 6 February 2004

PLS-GLR algorithm

$$a_{1,j} = GLRfit(x_j), j = 1, p$$

$$w_1 = \frac{a_1}{\|a_1\|}$$

$$t_1 = \frac{Xw_1}{w_1'w_1}$$

$$a_{2,j} = GLRfit(x_j, t_1), j = 1, p$$

$$w_2 = \frac{a_2}{\|a_2\|}$$

$$x_{1,j} = \text{linfit}(x_j / t_1), j = 1, p$$

$$t_2 = \frac{X_1w_2}{w_2'w_2}$$

$$a_{h,j} = GLRfit(x_j, t_1, t_2, \dots, t_{h-1}), j = 1, p$$

$$w_h = \frac{a_h}{\|a_h\|}$$

$$x_{h-1,j} = \text{linfit}(x_j / t_1, t_2, \dots, t_{h-1}), j = 1, p$$

$$t_h = \frac{X_{h-1}w_h}{w_h'w_h}$$

PLS-GLR and sparsity

By taking advantage from the statistical tests associated with generalized linear regression, it is feasible to select the significant explanatory variables to include in the model.

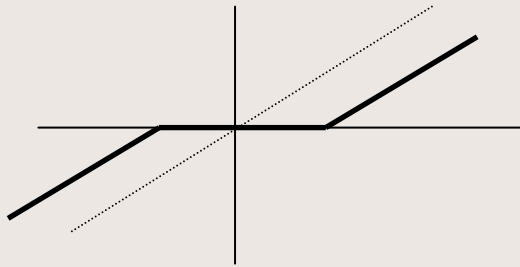
Computation of the PLS components t_h can thus be simplified by setting to 0 those regression coefficients $a_{h-1,j}$ that show to be not significant at a predefined threshold.

Only significant variables will then contribute to the computation of the PLS components.

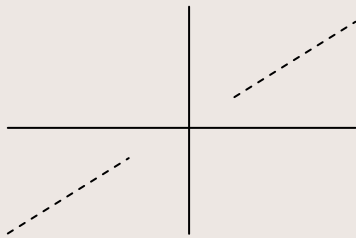
Moreover, the number m of PLS components to be retained may be chosen by observing that the component t_{m+1} is not significant because none of the coefficients $a_{m,j}$ is significantly different from 0.

Soft/Hard Thresholding

Soft thresholding exhibits continuous shrinkage but suffers from substantial bias for large coefficients.



Hard thresholding exhibits non continuous shrinkage but does not suffer from such a bias.



Régression logistique PLS

Table 8
Bordeaux wine data

OBS	Year	Temperature	Sunshine	Heat	Rain	Quality
1	1924	3064	1201	10	361	2
2	1925	3000	1053	11	338	3
3	1926	3155	1133	19	393	2
4	1927	3085	970	4	467	3
5	1928	3245	1258	36	294	1
6	1929	3267	1386	35	225	1
7	1930	3080	966	13	417	3
8	1931	2974	1189	12	488	3
9	1932	3038	1103	14	677	3
10	1933	3318	1310	29	427	2
11	1934	3317	1362	25	326	1
12	1935	3182	1171	28	326	3
13	1936	2998	1102	9	349	3
14	1937	3221	1424	21	382	1
15	1938	3019	1230	16	275	2
16	1939	3022	1285	9	303	2
17	1940	3094	1329	11	339	2
18	1941	3009	1210	15	536	3
19	1942	3227	1331	21	414	2
20	1943	3308	1366	24	282	1
21	1944	3212	1289	17	302	2
22	1945	3361	1444	25	253	1
23	1946	3061	1175	12	261	2
24	1947	3478	1317	42	259	1
25	1948	3126	1248	11	315	2
26	1949	3458	1508	43	286	1
27	1950	3252	1361	26	346	2
28	1951	3052	1186	14	443	3
29	1952	3270	1399	24	306	1
30	1953	3198	1259	20	367	1
31	1954	2904	1164	6	311	3
32	1955	3247	1277	19	375	1
33	1956	3083	1195	5	441	3
34	1957	3043	1208	14	371	3

The following variables (Table 8) were measured in 34 years (1924–1957):

Temperature: Sum of average day temperatures (°C)
 Sunshine: Duration of sunshine (h)
 Heat: Number of very warm days
 Rain: Rain height (mm)
 Wine quality: 1 = good, 2 = average, 3 = poor

Ordinal logistic regression

- Ordinal logistic regression of quality on the four standardised predictors corresponds to the following model

$$Prob(y \leq k) = \frac{e^{a_k + b_1 Temperature + b_2 Sunshine + b_3 heat + b_4 Rain}}{1 + e^{a_k + b_1 Temperature + b_2 Sunshine + b_3 heat + b_4 Rain}}$$

Wine quality (k): 1 = good; 2 = average; 3 = poor

Ordinal logistic regression

Table 9
Logistic regression of quality on the standardised meteorological variables

Score test for the proportional odds assumption
Chi-square = 2.9159 with 4 DF ($p = 0.5720$)

Analysis of maximum likelihood estimates

Variable	DF	Parameter estimate	Standard error	Wald chi-square	Pr > chi-square
INTERCP1	1	-2.6638	0.9266	8.2641	0.0040
INTERCP2	1	2.2941	0.9782	5.4998	0.0190
TEMPERA	1	3.4268	1.8029	3.6125	0.0573
SUNSHINE	1	1.7462	1.0760	2.6335	0.1046
HEAT	1	-0.8891	1.1949	0.5536	0.4568
RAIN	1	-2.3668	1.1292	4.3931	0.0361

Table 10
Prediction quality of model (27) by using classical logistic regression

Quality observed counts	Predicted			Total
	1	2	3	
1	8	3	0	11
2	2	8	1	11
3	0	1	11	12
Total	10	12	12	34

Régression logistique PLS

Régressions logistiques de la qualité du vin sur chaque prédicteur standardisé:

$$a_{1j} = GLRfit(x_j), j = 1, p$$

$$w_1 = \frac{a_1}{\|a_1\|}$$

$$t_1 = \frac{Xw_1}{w_1'w_1}$$

Temperature: 3.0117 (p = 0.0002)

Sunshine: 3.3401 (p = 0.0002)

Heat: 2.1445 (p = 0.0004)

Rain: -1.7906 (p = 0.0016)

$$t_1 = \frac{3.0117 * Temperature + 3.3401 * Sunshine + 2.1445 * Heat - 1.7906 * Rain}{\sqrt{(3.0117)^2 + (3.3401)^2 + (2.1445)^2 + (-1.7906)^2}}$$

$$= 0.5688 * Temperature + 0.6309 * Sunshine + 0.4050 * Heat - 0.3282 * Rain$$

Régression logistique PLS

Table 11
Results from logistic regression of quality on component t_1

Analysis of parameter	Maximum DF	Likelihood estimate	Estimates std.error	Chi-square	Pr > Chi sq
Intercept	1	-2.2650	0.8644	6.8662	0.0088
Intercept2	1	2.2991	0.8480	7.3497	0.0067
$t1$	1	2.6900	0.7155	14.1336	0.0002

Cross-table of observed and predicted quality

Quality observed Counts	Predicted			Total
	1	2	3	
1	9	2	0	11
2	2	8	1	11
3	0	1	11	12
Total	11	11	12	34

Régression logistique PLS

Détermination de t_2

$$\text{Prob}(y \leq k) = \frac{e^{\alpha_k + \beta_1 t_1 + \beta_2 x_j^*}}{1 + e^{\alpha_k + \beta_1 t_1 + \beta_2 x_j^*}}$$

$\rightarrow t_2$ NS

Expression du modèle final en fonction des descripteurs standardisés

$$\text{Prob}(y = 1) = \frac{e^{-2.265 + 1.53 \times \text{Temperature} + 1.70 \times \text{Sunshine} + 1.09 \times \text{Heat} - 0.91 \times \text{Rain}}}{1 + e^{-2.265 + 1.53 \times \text{Temperature} + 1.70 \times \text{Sunshine} + 1.09 \times \text{Heat} - 0.91 \times \text{Rain}}}$$

$$= 0.5688 \times \text{Temperature} + 0.6309 \times \text{Sunshine} + 0.4050 \times \text{Heat} - 0.3282 \times \text{Rain}$$

Régression logistique PLS

Balanced Bootstrap CI

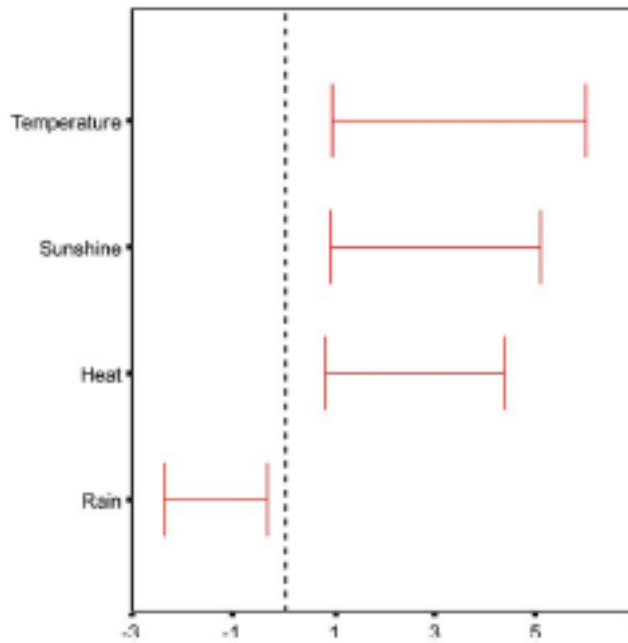


Fig. 3. 95% randomized/balanced bootstrap confidence intervals for Bordeaux wine data.

The Cox proportional hazard model

The model assumes the following hazard function for the occurrence of an event at time t in the presence of censoring:

$$\lambda(t) = \lambda_0(t) \exp(X\beta)$$

The Cox's partial likelihood can be written as :

$$PL(\beta) = \prod_{k \in D} \frac{\exp(\beta' x_k)}{\sum_{j \in R_k} \exp(\beta' x_j)}$$

When $p > n$, there is no unique β to maximize this partial likelihood.

Even when $p \leq n$, covariates could be highly correlated and regularization may still be required in order to reduce the variance of the estimates and to improve the prediction performance.

Cox D.R. 1972. Journal of the Royal Statistical Society B.; 74: 187-220

PLS-Cox regression

(Bastien & Tenenhaus, 2001)

- *Computation of the first PLS component \mathbf{t}_1*

$$a_{1i} = \text{Coxfit}(x_i) , i = 1, p$$

$$w_1 = \frac{a_1}{\|a_1\|}$$

$$t_1 = \frac{Xw_1}{w_1'w_1}$$

- *Computation of the second PLS component \mathbf{t}_2*

$$a_{2,i} = \text{Coxfit}(t_1, x_i) , i = 1, p$$

$$w_2 = \frac{a_2}{\|a_2\|}$$

$$x_{1,i} = \text{linfit}(x_i / t_1) , i = 1, p$$

$$t_2 = \frac{X_1w_2}{w_2'w_2}$$

PLS-Cox regression

- *Computation of the h^{th} PLS component \mathbf{t}_h*

$$a_{h,i} = \text{Coxfit}(t_1, t_2, \dots, t_{h-1}, x_{h-1,i}), \quad i = 1, p$$

$$w_h = \frac{a_h}{\|a_h\|}$$

$$x_{h-1,i} = \text{linfit}(x_i / t_1, t_2, \dots, t_{h-1}), \quad i = 1, p$$

$$t_h = \frac{X_{h-1} w_h}{w_h' w_h}$$

- *Coxfit(t_1, t_2, \dots, t_A)*
- *Expression of the Cox model in terms of the original explanatory variables*
- *Bootstrap validation*

L1 penalized Cox regression

In the context of censored data, Tibshirani (1997) extended the LASSO procedure to variable selection with the Cox model.

$$\hat{\beta} = \operatorname{argmin} -l(\beta), \text{ subject to } \sum_{j=1}^p |\beta_j| \leq s$$

Tibshirani proposed an iterative procedure which require the minimization of a one-term Taylor series expansion of the log partial likelihood.

$$\operatorname{argmin} (z - X\beta)^T A(z - X\beta) \text{ subject to } \sum_{j=1}^p |\beta_j| \leq s$$

$$\text{with } z = \eta + A^{-1}\mu, \quad \eta = X\beta, \quad \mu = \frac{\partial l}{\partial \eta}, \quad A = -\frac{\partial^2 l}{\partial \eta \eta'}$$

LARS procedure

Efron et al. (2004), proposed a highly efficient procedure, called Least Angle Regression for variable Selection which can be used to perform variable selection with very large matrices.

Using the connection between LARS and Lasso, Gui and Li (2005) proposed LARS-Cox for gene selection in high-dimension and low sample size settings.

Using a Choleski factorization they transform the minimization in a constrained version of OLS which can be solved by the LARS-Lasso procedure.

Efron B., Johnston I., Hastie T., and Tibshirani R., 2004. Annals of Statistics, 32:407-499

LARS-Cox

$$\arg \min (z - X\beta)^T A(z - X\beta) \text{ subject to } \sum_{j=1}^p |\beta_j| \leq s$$



$$\arg \min (y - \hat{X}\beta)^T (y - \hat{X}\beta) \text{ subject to } \sum_{j=1}^p |\beta_j| \leq s$$

with $y = Tz$, $\hat{X} = TX$, et $A = TT'$ (Choleski)

However, the IRWLS iterations performed in the LARS-Cox procedure counter balanced the efficiency of the LARS-Lasso algorithm and render the Gui and Li algorithm computationally costly.

Gui J. and Li H., (2005). Bioinformatics Penalized Cox regression analysis in the high-dimensional and low-sample size settings, with application to microarray gene expression data. Bioinformatics

Cox-Lasso procedure on deviance residuals

Segal (2006) showed that the formula to be minimized in the Cox-Lasso procedure can be approximate, at a first order Taylor approximation by the deviance residual sum of squares :

$$(z - X\beta)^T A(z - X\beta) \approx \text{RSS}(\hat{D})$$

$$\hat{M}_i = \delta_i - \hat{E}_i = \delta_i - \hat{H}_0(t_i) \exp \hat{\beta}' x_i$$

$$d_i = \text{sign}(\hat{M}_i) * \left[2 \left[-\hat{M}_i - \delta_i \ln \left(\frac{\delta_i - \hat{M}_i}{\delta_i} \right) \right] \right]^{1/2}$$

Segal M.R. (2006), Microarray Gene Expression Data with Linked Survival Phenotypes: Diffuse large-B-Cell Lymphoma Revisited, Biostatistics

Cox-Lasso procedure on deviance residuals

The deviance residual is a measure of excess of death and can therefore be interpreted as a measure of hazard.

Segal thus proposed to speed-up the calculations by replacing the survival times by the deviance residuals, a normalized version of Martingal residuals that result from fitting a null (intercept only) Cox regression model.

Therneau T.M., Grambsch P.M., and Fleming T.R. (1990). Martingale-based residuals for survival models. Biometrika

PLS on deviance residual

We have proposed to use the same idea in the setting of Partial Least Squares.

A very simple and fast alternative formulation of the PLS-Cox model could be derived by fitting the deviance residuals of a null Cox model with a simple PLS regression.



Available online at www.sciencedirect.com

 ScienceDirect

Chemometrics and Intelligent Laboratory Systems 91 (2006) 76–86

Chemometrics and
intelligent
laboratory systems

www.elsevier.com/locate/chemolab

Deviance residuals based PLS regression for censored data
in high dimensional setting

Philippe Bastien *

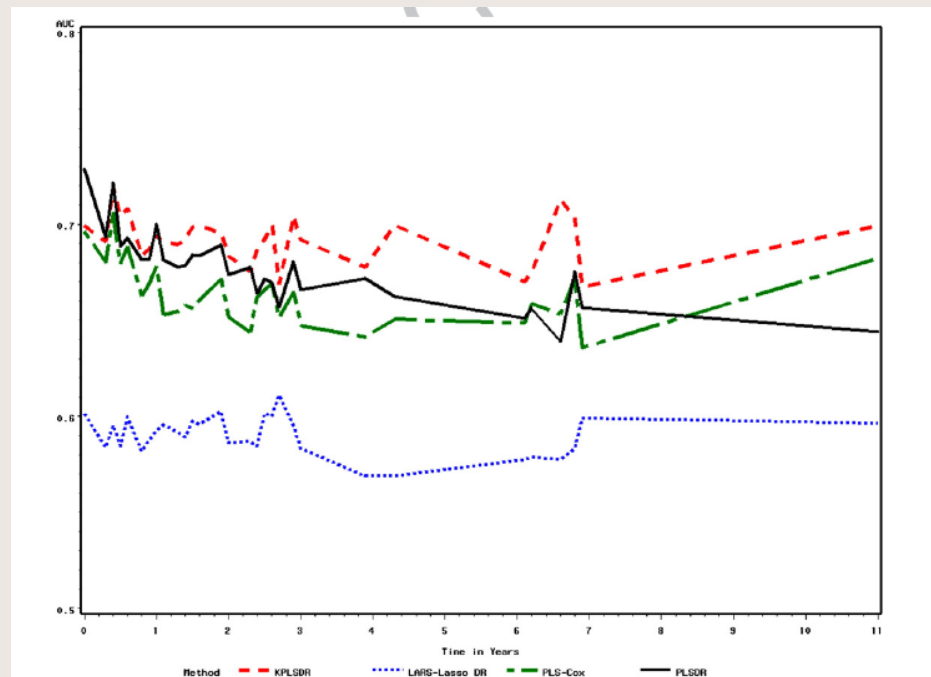
L'Oréal Recherche — Aulnay, 90 rue du Général Rogier, 92583 Clichy Cedex, France

Received 29 December 2006; received in revised form 20 September 2007; accepted 20 September 2007
Available online 5 October 2007

Application en transcriptomique

(<http://www-stat.stanford.edu/~tibs/superpc/standt.html>)

- 240 patients atteints de DLBCL.
- 138 décès / médiane de survie 2.8 ans / 30% de censure
- 7399 zones qui représentent 4128 gènes
- 160 échantillon d'apprentissage / 80 échantillon de test

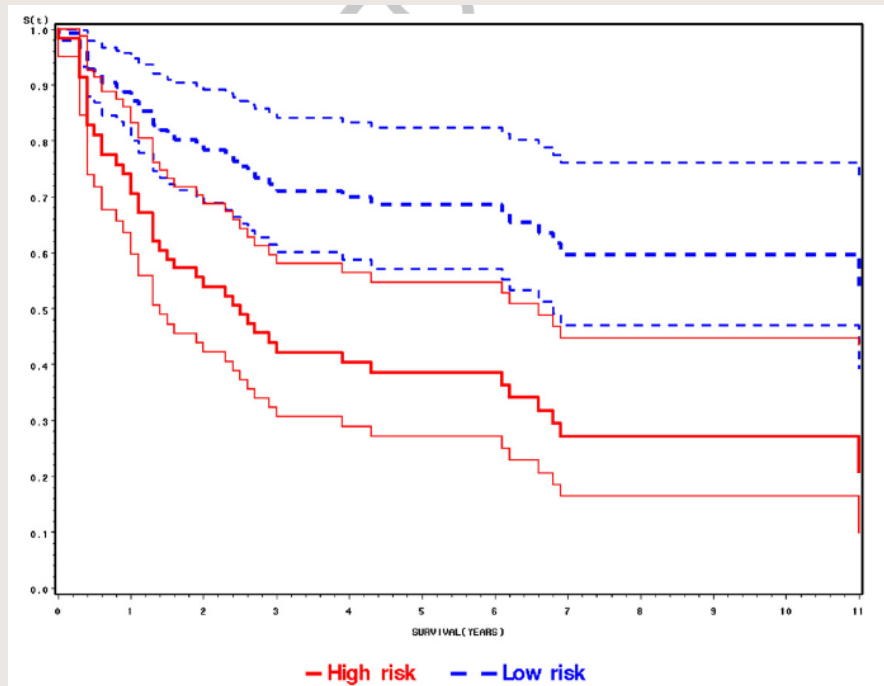


Rosenwald A. et al., 2002. *The new England Journal of medicine*, 346:1937-1947
Heagerty P.J., Lumley T., and Pepe M. 2000. *Biometrics* 56,337-344

DLBCL Rosenwald (2002)

$T\beta > 0 \Rightarrow \text{High Risk}$

$T\beta < 0 \Rightarrow \text{Low Risk}$



Wald test, $p=0.001$

plsRcox

- Récemment dans *Bioinformatics* (2009) Sohn et al. ont proposé un algorithme de type « gardient lasso » qui comme PLS ne requiert pas d'inversion de matrices.
- Bertrand et al ont montré sur les données de Rosenwald sur des données d'allélotypage que la PLSDR semble surperformer l'approche de Sohn et al.

Bertrand F., Maumey-Bertrand M., beau-Faller M., Meyer N. PLSRcox: modèles de Cox en présence d'un grand nombre de variables explicatives. Poster Chimométrie 2010, ENSPCI PARIS.

Packages sous R

- Package '**integrOmics**' : Regularized CCA and Sparse PLS
Author Sebastien Dejean, Ignacio Gonzalez and Kim-Anh Le Cao
- Package '**plsRglm**' : Partial least squares Regression for generalized linear models
Author Frederic Bertrand <frederic.bertrand@math.unistra.fr>, Nicolas Meyer
- Package '**plsRcox**' : Partial least squares Regression for Cox regression
Author Frederic Bertrand <frederic.bertrand@math.unistra.fr>, Maumy-Bertrand, Beau-Faller, Nicolas Meyer
- Package **PLSDOF** : Degrees of Freedom and Confidence Intervals for Partial Least Squares Regression
Author Nicole Kraemer, Mikio L. Braun

R Development Core Team : R : A language and environment for statistical computing, R Foundation for Statistical Computing, Vienna, Austria, 2008. <http://www.R-project.org>.