1 L'opérateur Gini covariance

Soit $\bar{x}_k$ la moyenne arithmétique de la variable x_k. L'opérateur de Gini covariance proposé par Schechtman and Yitzhaki (1987), encore appelé opérateur co-Gini est donné par:

$$\text{cog}(x_\ell, x_k) := \text{cov}(x_\ell, F(x_k)) = \frac{1}{N}\sum_{i=1}^{N}(x_{i\ell} - \bar{x_\ell})(\hat{F}(x_{ik}) - \bar{F}_{x_k}), \qquad (1.1)$$

où $\hat{F}(x_k)$ est la fonction de répartition de x_k, $\bar{F}_{x_k}$ sa moyenne, avec $\ell \neq k = 1, \ldots, K$. Lorsque $k = \ell$ le co-Gini mesure la variabilité entre une variable et elle-même (l'équivalent de la variance mesurée sur la norme ℓ_2). Le co-Gini est une mesure basée sur la distance de Manhattan (distance de métrique ℓ_1), en effet :

$$\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}|x_{ik} - x_{jk}| = 4\text{cog}(x_k, x_k).$$

D'autre part, lorsque $k \neq \ell$, le co-Gini produit une mesure de la variabilité jointe entre deux variables. Puisque le co-Gini n'est pas symétrique :

$$\text{cog}(x_k, x_\ell) := \text{cov}(x_k, F(x_\ell)) = \frac{1}{N}\sum_{i=1}^{N}(x_{ik} - \bar{x_k})(\hat{F}(x_{i\ell}) - \bar{F}_{x_\ell}).$$

Définissons les rangs croissants d'une variable alétoire afin de fournir un estimateur de F,

$$R_\uparrow(x_{i\ell}) := N\hat{F}(x_{i\ell)} = \begin{cases} \#\{x \leq x_{i\ell}\} & \text{si aucune observation similaire} \\ \frac{\sum_{i=1}^{p}\#\{x \leq x_{i\ell}\}}{p} & \text{s'il existe } p \text{ valeurs similaires } x_{i\ell}. \end{cases}$$

Alors, un estimateur du co-Gini est donné par,

$$\widehat{\text{cog}}(x_\ell, x_k) := \frac{1}{N}\sum_{i=1}^{N}(x_{i\ell} - \bar{x_\ell})(R_\uparrow(x_{ik}) - \bar{R}_{\uparrow_{x_k}}),\ \forall k, \ell = 1, \ldots, K, \qquad (1.2)$$

avec $\bar{R}_{\uparrow_{x_k}}$ la moyenne arithmétique du vecteur rang de la variable x_k.

2 Gini-PLS

Le premier algorithme Gini-PLS a été proposé par Mussard et Souissi-Benrejab (2018). Nous le décrivons dans les lignes qui suivent. Il s'agit d'une méthode de compression avec débruitage qui consiste à réduire les dimensions de

l'espace généré par X afin de trouver des composantes principales débruitées, dans le même esprit qu'une ACP débruitée, néanmoins l'approche est supervisée dans la mesure où une variable cible y est prise en compte dans le changement d'espace. Le sous-espace formé par les composantes principales $\{t_1, t_2, \cdots\}$ est construit de telle sorte que le lien entre les variables explicatives $X = [x_1, x_2, \cdots]$ et la cible y est maximisé.

• **Étape 1:** La régression Gini permet de concevoir un nouveau type de lien entre la variable expliquée et les variables explicatives tout en évitant l'influence des valeurs aberrantes. Ceci est permis grâce notamment à l'opérateur co-Gini dans lequel le rôle de la variable explicative est remplacé par celui de son vecteur rang dans un espace muni d'une métrique ℓ_1. Ainsi, il est possible de créer un nouveau vecteur de poids w_1 qui renforce le lien (co-Gini) entre la variable expliquée y et les régresseurs X dans le cadre d'une régression (linéaire ou non linéaire).
La solution du programme,

$$\max \operatorname{cog}(y, Xw_1) \ , \ \text{s.c.} \ \|w_1\| = 1 \ , \ \text{est}$$

$$w_{1j} = \frac{\operatorname{cog}(y, x_j)}{\sqrt{\sum_{j=1}^{p} cog^2(y, x_j)}} \ , \ \forall j = 1 \ldots, p \ .$$

La pondération est équivalente à :

$$w_{1j} = \frac{\operatorname{cov}(y, R(x_j))}{\sqrt{\sum_{j=1}^{p} \operatorname{cov}^2(y, R(x_j))}} \ , \ \forall j = 1 \ldots, p \ .$$

Comme dans la régression PLS, on régresse y sur la composante t_1 qui est construite de la manière suivante :

$$t_1 = \sum_{j=1}^{p} w_{1j} x_j \implies y = \hat{c}_1 t_1 + \hat{\varepsilon}_1 \ .$$

• **Étape 2:** On régresse le vecteur rang de chaque régreseur $R(x_j)$ sur la composante t_1 par moindres carrés ordinaires afin de récupérer les résidus $\hat{U}_{(1)j}$:

$$R(x_j) = \hat{\beta} t_1 + \hat{U}_{(1)j} \ , \ \forall j = 1, \cdots, p \ .$$

On construit le nouveau vecteur de pondération en utilisant les rangs des résidus des régressions partielles :

$$\max \operatorname{cog}(\hat{\varepsilon}_1, \hat{U}_{(1)} w_2) \ , \ \text{s.c.} \ \|w_2\| = 1 \implies w_{2j} = \frac{\operatorname{cog}(\hat{\varepsilon}_1, \hat{U}_{(1)j})}{\sqrt{\sum_{j=1}^{p} \operatorname{cog}^2(\hat{\varepsilon}_1, \hat{U}_{(1)j})}}.$$

On utilise à présent les composantes t_1 et t_2 pour établir un lien entre y et les régresseurs x_j :

$$t_2 = \sum_{j=1}^{p} w_{2j} \hat{U}_{(1)j} \implies y = \hat{c}_1 t_1 + \hat{c}_2 t_2 + \hat{\varepsilon}_2 \ .$$

La validation croisée permet de savoir si t_2 est significative.

• **Étape 3:** Les régressions partielles sont réitérées en rajoutant l'influence de t_2 :

$$R(x_j) = \beta t_1 + \gamma t_2 + \hat{U}_{(2)j} \ , \ \forall j = 1, \cdots, p.$$

D'où, après maximisation :

$$w_{3j} = \frac{\mathrm{cog}(\hat{\varepsilon}_2, \hat{U}_{(2)j})}{\sqrt{\sum_{j=1}^{p} \mathrm{cog}^2(\hat{\varepsilon}_2, \hat{U}_{(2)j})}} \ ,$$

$$t_3 = \sum_{j=1}^{p} w_{3j} \cdot \hat{U}_{(2)j} \implies y = \alpha_2 + c_1 t_1 + c_2 t_2 + c_3 t_3 + \varepsilon_3 \ .$$

La procédure s'arrête lorsque la validation croisée indique que la composante t_l n'est pas significative. L'algorithme Gini-PLS1 est valable si toutes les composantes t_h et t_l sont orthogonales, $\forall h \neq l$.

La validation croisée permet de trouver le nombre optimal $h > 1$ de composantes à retenir. Pour tester une composante t_h, on calcule la prédiction du modèle avec h composantes comprenant l'observation i, $\hat{y}_{h_i}$, puis sans l'observation i, $\hat{y}_{h_{(-i)}}$. L'opération est répétée pour tout i variant de 1 à n : on enlève à chaque fois l'observation i et on ré-estime le modèle.[1] Pour mesurer la robustesse du modèle, on mesure l'écart entre la variable prédite et la variable observée :

$$PRESS_h = \sum_i \left(y_i - \hat{y}_{h_{(-i)}} \right)^2 \ .$$

La somme des carrés résiduels obtenue avec le modèle à $(h-1)$ composantes est :

$$RSS_{h-1} = \sum \left(y_i - \hat{y}_{(h-1)_i} \right)^2 \ .$$

[1] Les observations peuvent être éliminées bloc par bloc au lieu de l'être une à une, *Cf.* Tenenhaus (1998), p. 77.

Le critère RSS_h (Residual Sum of Squares) du modèle à h composante et $PRESS_h$ (PRedicted Error Sum of Squares) sont comparés. Leur ratio permet afin de savoir si le modèle avec la composante t_h améliore la prédictibilité du modèle. La statistique suivante est alors calculée :

$$Q_h^2 = 1 - \frac{PRESS_h}{RSS_{h-1}} \ .$$

La composante t_h est retenue si : $\sqrt{PRESS_h} \leq 0,95\sqrt{RSS_h}$. Autrement dit, lorsque $Q_h^2 \geq 0,0975 = (1-0,95^2)$, la nouvelle composante t_h est significative, elle améliore la prévision de la variable y. Pour la significativité de de la première composante t_1, on utilise :

$$RSS_0 = \sum_{i=1}^{n} (y_i - \bar{y})^2 \ .$$

3 Propositions : régressions Gini-PLS généralisée

Schechtman and Yitzhaki (2003) ont récemment généralisé l'opérateur co-Gini afin d'imposer plus ou moins de poids en queue de distribution. Notons $r_k = (R_\downarrow(x_{1k}), \ldots, R_\downarrow(x_{Nk}))$ le vecteur rang décroissant de la variable x_k, autrement dit, le vecteur qui assigne le rang le plus petit (1) à l'observation dont la valeur est la plus importante (et positive) x_{ik} :

$$R_\downarrow(x_{ik}) := \begin{cases} N+1-\#\{x \leq x_{ik}\} & \text{pas d'observation similaire} \\ N+1-\frac{\sum_{i=1}^{p}\#\{x \leq x_{ik}\}}{p} & \text{si } p \text{ observations similaires } x_{ik}. \end{cases}$$

L'opérateur co-Gini est généralisé grâce au paramètre ν :

$$\widehat{\text{cog}}_\nu(x_\ell, x_k) := -\nu\widehat{\text{cov}}(x_\ell, r_k^{\nu-1});\ \nu > 1. \tag{3.1}$$

Afin de bien comprendre le rôle de l'opérateur co-Gini, revenons sur la mesure du coefficient de corrélation linéaire généralisé au sens de Gini :

$$GC_\nu(x_\ell, x_k) := \frac{-\nu\widehat{\text{cov}}(x_\ell, r_k^{\nu-1})}{-\nu\widehat{\text{cov}}(x_\ell, r_\ell^{\nu-1})} \ ; \ GC_\nu(x_k, x_\ell) := \frac{-\nu\widehat{\text{cov}}(x_k, r_\ell^{\nu-1})}{-\nu\widehat{\text{cov}}(x_k, r_k^{\nu-1})}.$$

Property 1 – Schechtman et Yitzhaki (2003):
(i) $GC_\nu(x_\ell, x_k) \leq 1$.
(ii) *Si les variables x_ℓ et x_k sont indépendantes, pour tout $k \neq \ell$, alors* $GC_\nu(x_\ell, x_k) = GC_\nu(x_k, x_\ell) = 0$.

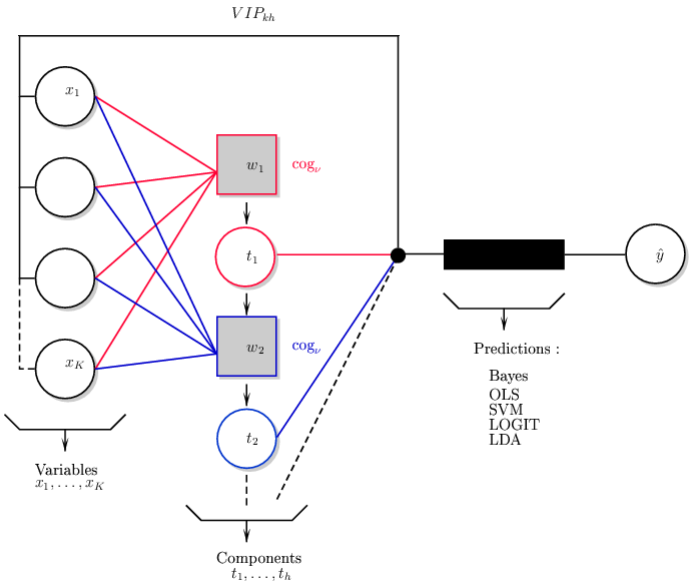
(iii) *Une transformation monotone des données φ n'affecte pas le coefficient de corrélation,* $GC_\nu(x_\ell, \varphi(x_k)) = GC_\nu(x_\ell, x_k)$.
(iv) *Pour une transformation linéaire φ,* $GC_\nu(\varphi(x_\ell), x_k) = GC_\nu(x_\ell, x_k)$ [*comme le coefficient de corrélation de Pearson*].
(v) *Si x_k et x_ℓ sont deux variables échangeables à une transformation linéaire près, alors* $GC_\nu(x_\ell, x_k) = GC_\nu(x_k, x_\ell)$.

Le rôle de l'opérateur co-Gini peut être expliqué de la manière suivante. Lorsque $\nu \to 1$, la variabilité des variables est atténuée de telle sorte que $\text{cog}_\nu(x_k, x_\ell)$ tend vers zéro (même si les variables x_k et x_ℓ sont fortement corrélées).Au contraire, si $\nu \to \infty$ alors $\text{cog}_\nu(x_k, x_\ell)$ permet de se focaliser sur les queues de distribution x_ℓ. Comme le montrent Olkin and Yitzhaki (1992), l'emploi de l'opérateur co-Gini atténue la présence de valeurs extrêmes, du fait que le vecteur rang agit comme un instrument dans la régression de y sur X (régression par variables instrumentales).

Ainsi, en proposant une régression Gini-PLS basée sur le paramètre ν, nous pouvons calibrer la puissance du débruitage grâce à l'opérateur co-Gini qui va localiser le bruit dans la distribution. Cette régression Gini-PLS généralisée devient une régression Gini-PLS régularisée où le paramètre ν joue le rôle de paramètre de régularisation.

3.1 L'algorithme Gini-PLS généralisé

Dans ce qui suit nous généralisons la régression Gini-PLS de Mussard et Souissi-Benrejab (2018) avec renforcement du pouvoir de débruitage par l'intermédiaire du paramètre nu.



La première étape consiste à trouver des poids de débruitage associés à chaque variable x_k afin d'en déduire la première composante t_1 (ou première variable latente). Cette opération est bouclée jusqu'à la composante t_{h^*}, où h^* est le nombre optimal de variable latentes. Ainsi, le modèle est estimé :

$$y = \sum_{h=1}^{h^*} c_h t_h + \varepsilon_h. \tag{3.2}$$

La statistique VIP_{hj} est mesurée afin de sélectionner la variable x_j qui a l'impact significatif le plus important sur $\hat{y}$. Les variables les plus significatives sont celles dont $VIP_{hj} > 1$ avec :

$$VIP_{hj} := \sqrt{\frac{p \sum_{\ell=1}^{h} Rd(y; t_\ell) w_{\ell j}^2}{Rd(y; t_1, \ldots, t_h)}}$$

et

$$Rd(y; t_1, \ldots, t_h) := \frac{1}{p} \sum_{\ell=1}^{h} \text{cor}^2(y, t_\ell) =: \sum_{\ell=1}^{h} Rd(y; t_\ell).$$

où $cor^2(y, t_\ell)$ est le coefficient de corrélation de Pearson entre y et la composante t_ℓ. Cette information est rétro-propagée dans le modèle (une seule fois) afin d'obtenir les variables latentes t_{h^*} et leurs coefficients estimés $\hat{c}_{h^*}$ sur les données d'entrainement. La variable cible y est ensuite prédite grâce à (3.2). Cette prévision est comparée aux modèles standards SVM, LOGIT,

Bayes et LDA lorsque les données tests sont projetées dans le sous-espace $\{t_1, \ldots, t_{h^*}\}$.

Algorithm 1: Gini-PLS Généralisé

Result: Prédiction du juge $y = 0; 1$

repeat
  repeat
    $\max \text{cog}_\nu(y, w_h X)$ s.t. $\|w_h\| = 1 \Longrightarrow$ poids w_h de X ;
    MCO équation: $y = \sum_h c_h t_h + \varepsilon_h$;
    MCO équation: $R(x_j) = \sum_h \beta_h t_h + \epsilon_k \ \forall k = 1, \ldots, K$;
    $X := (\hat{\epsilon}_1, \ldots, \hat{\epsilon}_K)$ $y := \hat{\varepsilon}_h$;
  until $h = 10$ $[h = h + 1]$;
  Mesurer VIP_{kh}, Q_h^2 ;
  Sélectionner le nombre optimal de composantes h^* ;
until $\nu = 14$ $[\nu = \nu + 2]$;
Déduire le paramètre optimal ν^* qui minimise l'erreur ;
return Prédiction $\hat{y}$ avec Gini-PLS (h^*, ν^*) ;
return Prédiction $\hat{y}$ avec SVM, LOGIT, Bayes, LDA sur les composantes $(t_1, \ldots, t_{h^*})$;

3.2 L'algorithme LOGIT-Gini-PLS généralisé

Comme nous le constatons dans l'algorithme Gini-PLS généralisé que nous avons proposé dans le section précédente, les poids w_j proviennent de l'opérateur co-Gini appliqué à une variable booléenne $y = 0; 1$. Afin de trouver les poids w_j qui maximisent le lien entre les variables x_j et la variable cible y, nous proposons d'utiliser la régression LOGIT, autrement dit, une sigmoïde qui est bien adapté à des variables boléennes. Ainsi, dans chaque étape de la régression Gini-PLS nous remplaçons la maximisation du co-Gini par la mesure de la probabilité conditionnelle suivante :

$$\mathbb{P}(y_i = 1/X = X_i) = \frac{\exp\{X_i\beta\}}{1 + \exp\{X_i\beta\}} \qquad \text{(LOGIT)}$$

où X_i est la ième ligne de la matrice X (observation des caractéristiques/dimensions de la décision juridique i). L'estimation du vecteur β se fait maximum de vraisemblance. On en déduit alors les pondérations w_j :

$$w_j = \frac{\beta_j}{\|\beta\|}$$

L'algorithme LOGIT-Gini-PLS généralise est donc le suivant :

Algorithm 2: LOGIT-Gini-PLS Généralisé

Result: Prédiction du juge $y = 0; 1$

repeat
  repeat
    LOGIT équation : $\Longrightarrow$ poids w_j de X ;
    MCO équation : $y = \sum_h c_h t_h + \varepsilon_h$;
    $X := (\hat{\epsilon}_1, \ldots, \hat{\epsilon}_K)$ $y := \hat{\varepsilon}_h$;
  until $h = 10$ $[h = h + 1]$;
  Mesurer VIP_{kh}, Q_h^2 ;
  Sélectionner le nombre optimal de composantes h^* ;
until $\nu = 14$ $[\nu = \nu + 2]$;
Déduire le paramètre optimal ν^* qui minimise l'erreur ;
return Prédiction $\hat{y}$ avec Gini-PLS (h^*, ν^*) ;
return Prédiction $\hat{y}$ avec SVM, LOGIT, Bayes, LDA sur les composantes $(t_1, \ldots, t_{h^*})$;

References

[1] Mussard, S. and F. Souissi-Benrejab (2018), Gini-PLS regressions, *Journal of Quantitative Economics*, 1-36, https://doi.org/10.1007/s40953-018-0132-9.

[2] Schechtman, E., Yitzhaki, S. (2003). A family of correlation coefficients based on extended Gini, *Journal of Economic Inequality*, 1, 129–146.

[3] Wold, S., Martens, H., Wold, H. (1983). The multivariate calibration problem in chemistry solved by the PLS method, in *Proc. Conf. Matrix Pencils*, Rnhe, A., and Kagstroem, B. (eds), Springer-Verlag: Berlin. pp. 286–293.

[4] Yitzhaki, S., Schechtman, E. (2013). *The Gini Methodology: A Primer on a Statistical Methodology*, Springer-Verlag: Berlin.