Descente de gradient stochastique sur le modèle de Cox : données longitudinales et coefficients dépendants du temps

Thibault Allart[a,b,c] and **Agathe Guilloux**[b,d]

[a]Ubisoft Production Internationale
126 rue de Lagny, 93100 Montreuil-Sous-Bois
[b]Université Pierre et Marie Curie
4, place Jussieu, 75252 Paris
[c]Conservatoire National des Arts et Métiers
292 rue Saint Martin, 75141 Paris
[d]CMAP, Ecole Polytechnique
Palaiseau, 91120 France

thibault.allart@gmail.com, agathe.guilloux@upmc.fr

Mots clefs : Analyse de survie, Données massives

En analyse de survie, les récentes avancées en aquisition de données ont amené des données de grande dimension (nombreuses variables) et de grande taille (grand nombre d'observations) : la santé [9], l'économie [6], etc.. Les algorithmes capables de travailler avec des données de grande taille sont pourtant encore peu nombreux, à l'exception de [8, 1]. Nous proposons ici un nouvel algorithme pour le modèle le plus utilisé de l'analyse de survie : le modèle de Cox [3]. Nous considérons le modèle de Cox

$$\lambda^\star(t|X(t)) = \exp\big(\beta^\star(t)X(t)\big) \tag{1}$$

dans lequel les covariables X (qui contiennent l'intercept) et les coefficients $\beta^\star$ peuvent dépendre du temps. La fonction $\exp(\beta^\star)$ est une fonction de $\mathbb{R}_+$ et $\mathbb{R}^{d+1}$ et sa première composante est généralement appelée risque de base.
Pour chaque individu $i = 1, \dots, n$, nous observons un processus de comptage N_i, un processus de covariables X_i et un processus de censure Y_i, observés sur un intervalle $[0, \tau]$. Dans ce modèle de Cox, l'opposé de la log-vraisemblance complète est donnée par

$$\ell_n(\beta) = -\frac{1}{n}\sum_{i=1}^{n}\Big\{\int_0^\tau X_i(t)\beta(t)dN_i(t) - \int_0^\tau Y_i(t)\exp\big(X_i(t)\beta(t)\big)dt\Big\}, \tag{2}$$

voir [7] pour des détails sur le modèle et sa vraisemblance.
L'algorithme actuellement disponible pour calculer les estimateurs dans ce modèle est implémenté dans le package `Timereg` [12] de R. Son implémentation repose sur une inversion matricielle et des itérés de lissage par noyaux et ne permet pas de traiter de grands volumes de données (grand nombre d'individus). Elle ne permet, par ailleurs, pas de considérer les pénalités classiques (de type lasso) et ne permet donc pas de considérer des données de grande dimension (grand nombre de covariables). Nous proposons ici une nouvelle implémentation, qui s'affranchit de ces deux limitations majeures.
Pour cela, nous considérons comme candidats à l'estimation des fonctions β constantes par morceaux sur des intervalles de temps $\{I_1, \dots, I_L\}$ de sorte que l'opposé de la log-vraisemblance s'écrit

$$\ell_n(\beta) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{l=1}^{L}\left(X_{i,l}\beta_l N_{i,l}(I_l) - \exp\left(X_{i,l}\beta_l\right)\int_{I_l} Y_i(t)ds\right). \tag{3}$$

Le problème de minimisation est alors séparable en n et L et permet de considérer des algorithmes de descente de gradient stochastique, tels que Adagrad [4] ou Adadelta [14]. Nous avons implémenté une variante de l'algorithme de LeCun et al. [11].
Pour introduire de la sparsité en temps et en covariables, nous avons proposé une nouvelle pénalité basée sur les pénalités TV [2] et L1 [13], définie ci-dessous. Le choix de L et des $\hat{\gamma}_{j,l}$ est fixé par les données; seul λ est estimé par validation croisée.

$$\|\beta\|_{\mathrm{gTV},\hat{\gamma}} = \lambda\sum_{j=0}^{d}\left(\hat{\gamma}_{j,1}|\beta_{j,1}| + \sum_{l=2}^{L}\hat{\gamma}_{j,l}|\beta_{j,l} - \beta_{j,l-1}|\right) \tag{4}$$

Nous avons implémenté ce modèle dans un package R [10]. Par soucis de performance, le code est écrit en c++ et interfacé avec R via RCpp [5]. Le package permet de travailler directement avec des fichiers de données, ce qui permet de faire l'estimation sur des données dont la taille dépasse celle de la mémoire vive (RAM) de l'ordinateur.

Références

[1] Massil Achab, Agathe Guilloux, Stéphane Gaïffas, and Emmanuel Bacry. Sgd with variance reduction beyond empirical risk minimization, 2015.

[2] Laurent Condat. A direct algorithm for 1d total variation denoising. *IEEE Signal Processing Letters*, 20(11):1054–1057, 2013.

[3] D. R. Cox. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society, Series B*, 74:187–220, 1972.

[4] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, 12:2121–2159, July 2011.

[5] Dirk Eddelbuettel and Romain François. Rcpp: Seamless R and C++ integration. *Journal of Statistical Software*, 40(8):1–18, 2011.

[6] Liran Einav and Jonathan Levin. Economics in the age of big data. *Science*, 346(6210):1243089, 2014.

[7] Torben Martinussen and Thomas H Scheike. *Dynamic regression models for survival data*. Springer Science & Business Media, 2007.

[8] S. Mittal, D. Madigan, J. Cheng, and R. S. Burd. Large-scale parametric survival analysis. *Statistics in medicine*, 32(23):3955–3971, 10 2013.

[9] Travis B Murdoch and Allan S Detsky. The inevitable application of big data to health care. *Jama*, 309(13):1351–1352, 2013.

[10] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016.

[11] T. Schaul, S. Zhang, and Y. LeCun. No more pesky learning rates. *arXiv preprint arXiv:1206.1106*, 2012.

[12] T Scheike, T Martinussen, and J Silver. timereg: timereg package for flexible regression models for survival data. *R package version*, pages 1–2, 2009.

[13] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[14] M. D. Zeiler. Adadelta: An adaptive learning rate method. *CoRR*, abs/1212.5701, 2012.