Régularisation implicite des factorisations de faible rang pénalisées

Jérémy E. COHEN

Univ Lyon, INSA-Lyon, UCBL, UJM-Saint Etienne, CNRS, Inserm, CREATIS UMR 5220, U1206, F-69100 Villeurbanne, France

Résumé – L'invariance d'échelle des modèles de factorisation matriciels et tensoriels est une propriété bien connue, habituellement perçue comme une source d'ambiguïté lors de l'inférence des paramètres de ces modèles. Cependant, lorsque ces modèles de factorisation sont employés dans une approche variationnelle, typiquement lorsque de la parcimonie est imposée, l'invariance d'échelle induit une régularisation implicite qui équilibre les solutions. En adaptant un algorithme d'optimisation classique, je montre empiriquement que l'estimation des paramètres pour des décompositions tensorielles pénalisées devient plus précise et fiable.

Abstract – Scale invariance is a well-known property of matrix and tensor factorization models. It is usually considered only as a source of ambiguity during inference. However, when regularizations which are not scale-invariant are added to the cost function, scale-invariance induces an implicit regularization that balances the estimated factors. These behaviors have been partially documented formally, but have not been accounted for practically. In this work, I further discuss this implicit regularization and show empirically how to adapt existing algorithms for improved robustness to hyperparameter choice and improved precision.

1 Introduction et formalisme

Intéressons-nous aux problèmes de la forme suivante pour des entiers m, n, r tels que $r \leq \min(m, n)$:

$$\inf_{\substack{X \in \mathbb{R}^{m\times r} \\ Y \in \mathbb{R}^{n\times r}}} f(XY^T) + \mu_X \sum_{i\leq r} g_X(X_i) + \mu_Y \sum_{j\leq r} g_Y(Y_j) \tag{1}$$

où f est une fonction de coût positive, $g_{X,Y}$ sont des fonctions homogènes définies positives de degrés respectifs p_X et p_Y, et $\mu_{X,Y}$ sont des hyperparamètres réels positifs. Les vecteurs X_i et Y_j sont les i-ème et j-ème colonnes des matrices X et Y.

Un exemple de problème couvert par (1) est la factorisation en matrices nonnégatives (NMF) parcimonieuses avec une fonction de perte euclidienne et une matrice de données $M \in \mathbb{R}^{m\times n}$:

$$\inf_{\substack{X \in \mathbb{R}_+^{m\times r} \\ Y \in \mathbb{R}_+^{n\times r}}} \|M - XY^T\|_F^2 + \mu_X\|X\|_1 + \mu_Y\|Y\|_2^2 \,. \tag{2}$$

Le terme d'attache aux données $f(XY^T)$ est invariant par une mise à l'échelle des colonnes des matrices facteurs X et Y. En effet $f(XY^T) = f(X\Lambda\Lambda^{-1}Y^T)$ pour une matrice diagonale Λ de rang plein. Cependant ce n'est pas le cas des termes de régularisation $g_{X,Y}$, et une solution optimale doit donc notamment optimiser les échelles Λ. Le reste de cet article est dédié à l'étude des solutions optimales à travers le prisme de cette mise à l'échelle. Je montre notamment que les colonnes des matrices facteurs sont équilibrées à l'optimum. Cet équilibre peut être atteint par une procédure simple de normalisation pour la décomposition Canonique Polyadique (CP) ou la NMF dont les bénéfices pratiques sont étudiés expérimentalement à la section 4.

2 Solutions d'échelle optimale

2.1 Lien entre invariance d'échelle et équilibre des solutions

Dans cette section nous étudierons un problème matriciel, et une mise à l'échelle pour une seule colonne. En effet le problème est séparable en Λ, et la généralisation au cas tensoriel est simple car les outils présentés plus bas fonctionnent pour un nombre arbitraire de facteurs. On note x et y respectivement la colonne de X et Y considérée, pour un couple (X, Y) admissible de colonnes non nulles.

Puisque le terme $f(XY^T)$ a une invariance d'échelle, minimiser (1) par rapport à un équilibrage des colonnes x et y revient à résoudre le problème suivant :

$$\operatorname*{argmin}_{\lambda_X\geq 0,\ \lambda_Y\geq 0} \mu_X g_X(x)\lambda_X^{p_X} + \mu_Y g_Y(y)\lambda_Y^{p_Y} \text{ où } \lambda_X\lambda_Y = 1 \tag{3}$$

ou bien de façon équivalente,

$$\operatorname*{argmin}_{a\geq 0,\ b\geq 0} a + b \text{ où } a^{1/p_X}b^{1/p_Y} = d \tag{4}$$

avec $d = (\mu_X g_X(x))^{1/p_X}(\mu_Y g_Y(y))^{1/p_Y}$. On peut montrer[1] que ce problème, lié à la définition de la moyenne géométrique pondérée, a pour solution un couple (a^*, b^*) lorsque $p_X a^* = p_Y b^* = \beta$ où

$$\beta := \left(p_X^{1/p_X} p_Y^{1/p_Y} d\right)^{\frac{1}{1/p_X+1/p_Y}} \,. \tag{5}$$

On peut également remonter à la valeur optimale de λ dans l'équation (3), qui donne une mise à l'échelle optimale de x et y :

$$x^* = \left(\frac{\beta}{p_X\mu_X g_X(x)}\right)^{1/p_X} x \text{ et } y^* = \left(\frac{\beta}{p_Y\mu_Y g_Y(y)}\right)^{1/p_Y} y \,. \tag{6}$$

[1] La preuve est standard et différée à une version étendue du papier.

Pour un modèle tensoriel avec N facteurs $\{X^k\}_{k\leq N}$, on peut également montrer que le problème

$$\underset{a_k\geq 0}{\operatorname{argmin}} \sum_{k\leq N} a_k \text{ où } \prod_{k\leq N} a_k^{1/p_{X^k}} = d \tag{7}$$

avec $d = \prod_k \left(\mu_{X^k} g_{X^k}(x^k)\right)^{1/p_{X^k}}$ a une solution équilibrée. On aura le même effet d'équilibrage pour une colonne x^k du facteur X^k

$$x^{*k} = \left(\frac{\beta}{p_{X^k}\mu_{X^k} g_{X^k}(x^k)}\right)^{1/p_{X^k}} x^k \tag{8}$$

avec $\beta = \left(d \prod_k p_{X^k}^{1/p_{X^k}}\right)^{\frac{1}{\sum_k 1/p_{X^k}}}$.

2.2 Régularisation implicite

On peut tirer plusieurs conclusions de ces calculs. Tout d'abord à l'optimalité, on vérifiera nécessairement que pour chaque couple de colonnes X_i, Y_i, ces colonnes sont équilibrés : $p_X \mu_X g_X(X_i^*) = p_Y \mu_Y g_Y(Y_i^*)$. On peut donc dire que l'invariance d'échelle des modèles de factorisation matriciels et tensoriels induit une régularisation implicite sur les matrices facteurs de ces modèles qui conduit à les équilibrer.

De plus, la fonction minimisée en (3) a pour solution une solution particulière du problème suivant, obtenu en marginalisant Λ :

$$\inf_{\substack{X\in\mathbb{R}^{m\times r}\\ Y\in\mathbb{R}^{n\times r}}} f(XY^T) + \alpha \sum_{i\leq r} \left(g_X(X_i)^{1/p_X} g_Y(Y_i)^{1/p_Y}\right)^{\frac{1}{1/p_X+1/p_Y}} \tag{9}$$

avec α une constante dépendant de μ_X, μ_Y, p_X, p_Y. Cette écriture est intéressante notamment pour certains choix de fonctions de pénalité, donnons ici deux exemples :

- Pour $g_X = g_Y = \ell_1$. On a alors $\beta \propto \|x \otimes y\|_1$. En d'autres termes, si on introduit les matrices de rang un $L_i = X_i Y_i^T$, le coût minimisé en (9) est en fait

$$\inf_{\text{rang}(L_i)=1} f(\sum_{i\leq r} L_i) + \alpha \sum_{i\leq r} \sqrt{\|L_i\|_1}, \tag{10}$$

  c'est à dire une variante du Lasso par groupe avec une contrainte de rang, ou encore une variante de l'ACP parcimonieuse. Bien que ce modèle, à ma connaissance, n'ait pas été étudié, il devrait être possible de partir de ce point pour dériver des garanties de parcimonie sur les matrices L_i, ce qui reste présentement élusif.

- Similairement, pour $g_X = g_Y = \ell_2^2$, on a $\beta \propto \|x \otimes y\|_2$. En ajoutant à l'invariance d'échelle l'invariance par rotation propre à la norme euclidienne, Srebro *et. al.* ont montré [9] que la double pénalité ℓ_2^2 induit en fait une pénalité sur la norme nucléaire de $L := \sum_{i\leq r} L_i$:

$$\inf_{L\in\mathbb{R}^{m\times n}, \text{rang}(L)\leq r} f(L) + \alpha \|L\|_* \,. \tag{11}$$

  Si ce résultat est relativement bien connu dans la communauté des approximations de rang faible matricielles, il l'est beaucoup moins dans la littérature de la NMF et des décompositions tensorielles, alors qu'il permet de régulariser le nombre de composantes sans introduire des pénalités du type parcimonie de groupe sur les facteurs.

2.3 Résultats similaire dans la littérature

Les interactions entre régularisation et invariance d'échelle sont, à ma connaissance, peu étudiées dans la littérature sur la séparation de sources. Benichoux, Vincent et Gribonval [2] ont tout de même montré en 2013 que pour un mélange convolutif, l'invariance d'échelle conduit un modèle régularisé par une norme ℓ_1 à avoir une solution dégénérée. De plus, comme cité plus haut, Srebro, Rennie et Jaakkola ont montré en 2004 que la minimisation de pénalités ℓ_2^2 conduit à implicitement pénaliser le rang de l'approximation.

Cependant, des phénomènes similaires sont observables lors de l'entraînement des réseaux de neurones. La littérature sur ce sujet est beaucoup plus fournie. On peut citer par exemple les travaux récents de Ergen et Pilanci [4] ou bien de Tibshirani [11] qui établissent un lien entre régularisation explicite des couches d'un réseau et, entre autres, le LASSO. Sur le plan pratique, Neyshabur et co-auteurs [7], ainsi que Stock et co-auteurs [10], ont montré l'intérêt d'équilibrer explicitement les couches d'un réseau lors de l'entraintement. Dans ce qui suit, je poursuis la même logique et propose d'équilibrer explicitement les colonnes des matrices facteurs.

3 Algorithme d'équilibrage des facteurs

Dans cette section, je propose d'équilibrer les colonnes des facteurs comme vu précédemment au sein d'un algorithme d'optimisation usuel de factorisation matriciel ou tensoriel afin de se rapprocher plus rapidement de l'espace des solutions. L'idée sous-jascente est que, lorsque le terme d'attache au données est bien minimisé mais que les facteurs ne sont pas encore équilibrés, un algorithme de type alterné a des difficultés à converger vers une solution équilibrée rapidement car il ne peut pas modifier simultanement les facteurs à équilibrer, et doit donc pour ce faire dégrader le terme d'attache aux données. L'algorithme 1 définit une procédure pour explicitement équilibrer les colonnes des facteurs selon la règle de mise à jour (6) au sein d'un algorithme alterné générique.

La formulation du problème (1) a été pensée pour que les pénalités soient séparables colonne par colonne. Cela permet de dériver des algorithmes de minimisation alternés travaillant colonne par colonne, matrice facteur par matrice facteur ou globalement. L'algorithme proposé minimise alternativement le coût par rapport à chaque facteur, et équilibre tous les facteurs après chaque mise à jour. Cette opération est peu couteuse et ne ralentit pas l'algorithme. De plus elle décroit nécessairement la fonction de coût et l'on peut donc préserver les garanties de convergence sur la fonction du coût.

Je propose également une étape d'initialisation dans laquelle les facteurs ne sont pas uniquement équilibrés (lignes 5 à 7), mais également remis à l'échelle (lignes 3 et 4). Cette remise à l'échelle est importante pour éviter un phénomène de "blocage en zéro". En effet, si l'on considère par exemple le problème de NMF parcimonieuse (2), lors de la mise à jour de X à la première itération, la ligne 9 de l'algorithme 1 revient à résoudre un LASSO de matrice de mélange Y^T. Il est connu que si $\|MY\|_\infty \leq \mu_X$, la solution est nécessairement $X = 0$. Dans ce cas, $(0, 0)$ étant un point selle du problème (2), l'algorithme retournera forcement $X = 0$ et $Y = 0$. La mise

Algorithme 1 : Algorithme générique équilibré

Input : fonction f, paramètres de régularisation $\mu_{X,Y}$, fonctions $g_{X,Y}$, valeurs initiales X, Y.
Initialisation :
$\eta \in \text{argmin}_{\eta \geq 0} f(\eta^2 XY^T) + \eta^{p_X} \mu_X \sum_i g_X(X_i) + \eta^{p_Y} \mu_Y \sum_i g_Y(Y_i)$
$X = \eta X, Y = \eta Y$
pour $i \in [1, r]$ **faire**
  X_i et Y_i normalisés selon (6)
fin
tant que *convergence n'est pas observée* **faire**
  $X \in \text{argmin}_X f(XY^T) + \mu_X \sum_{i \leq r} g_X(X_i)$
  pour $i \in [1, r]$ **faire**
    X_i et Y_i normalisés selon (6)
  fin
  $Y \in \text{argmin}_Y f(XY^T) + \mu_Y \sum_{i \leq r} g_Y(Y_i)$
  pour $i \in [1, r]$ **faire**
    X_i et Y_i normalisés selon (6)
  fin
fin

à l'échelle mitige ce problème en évitant de commencer avec des valeurs pour X et Y qui soient inutilement grandes.

4 Expériences

J'ai testé les effets de la procédure d'équilibrage sur l'inférence des paramètres de deux modèles CP nonnégative d'ordre 3 :

- setup 1 : un facteur parcimonieux pénalisé par la norme ℓ_1 (le premier facteur) et deux facteurs pénalisés par la norme ℓ_2^2. Dans ce cas, pour un tenseur de données T décomposé au rang r la fonction de coût minimisée est

$$\underset{A\geq 0,\ B\geq 0,\ C\geq 0}{\text{argmin}} \frac{1}{2}\|T - \sum_{q\leq r} A_q \otimes B_q \otimes C_q\|_F^2 + \mu_A\|A\|_1 + \mu_B\|B\|_F^2 + \mu_C\|C\|_F^2 \quad (12)$$

- setup 2 : tous les facteurs sont pénalisés par la norme ℓ_2^2. Le tenseur estimé devrait être de rang plus faible que le rang de la décomposition fixé par l'utilisateur.

avec f une fonction de coût euclidienne, pour un tenseur de données $Y \in \mathbb{R}^{30\times 30\times 30}$. L'algorithme choisi pour instancier l'algorithme générique 1 est HALS (Hierarchical Alternating Least Squares) [3]. C'est un algorithme état de l'art pour estimer les paramètres d'un modèle CP nonnégatif, et il s'adapte facilement pour prendre en compte des pénalités ℓ_1 et ℓ_2^2.

Pour mieux mesurer l'effet des régularisations, le rang des données $r = 4$ est surestimé par le modèle qui a un rang $\hat{r} = 6$. Les données sont générées aléatoirement en tirant les facteurs selon des lois uniformes sur $[0, 1]$. Le premier facteur, pour le 1er setup uniquement, est tronqué afin que seules 30% des entrées, les plus grandes, soient conservées. Les données sont ensuite bruitées par un tenseur dont les entrées sont tirées selon des Gausiennes i.i.d. centrées, avec un écart-type $\sigma = 0.001$. Le niveau de bruit est très faible en pratique pour ce choix de dimensions. L'initalisation des facteurs est aléatoire et suit la même procédure que pour générer les données (sans la troncature).

Les expériences menés ci-dessous considèrent différentes métriques en fonction du paramètre de régularisation μ, choisi tel que $\mu = \mu_A = \mu_B = \mu_C$. Ces différentes métriques sont 1) La valeur prise par la fonction de coût (1) après 30 itérations, normalisée par $\|T\|_F^2$. 2) Le degré de parcimonie du premier facteur. Pour le setup 1 il doit avoir environ 30% d'éléments non-nuls. Pour le setup 2 il doit avoir exactement deux colonnes nulles. 3) La précision de l'estimation de facteurs, en utilisant le "factor match score" (FMS)[1].

Plusieurs méthodes sont comparées : HALS avec équilibrage, HALS sans équilibrage mais avec initialisation de l'Algorithme 1, HALS sans équilibrage, HALS avec pénalité ℓ_1 mais pas ℓ_2 ($\mu_B = \mu_C = 0$), et enfin HALS sans régularisation. Les résultats moyennés pour $N = 5$ réalisations sont présentés par la Figure 1. L'HALS équilibré a été implémenté à partir de Tensorly [6], et les expériences conduites avec la boite à outils Shootout[2] Le FMS est calculé avec Tensorly-viz[3]. Le code est rendu disponible : `https://github.com/cohenjer/Gretsi2023_scale_invariance_lra`. L'HALS utilise exactement vingt itérations internes sans critère d'arrêt dynamique.

Observations : Pour les deux expériences, l'algorithme HALS équilibré maintient la fonction de coût proche de la valeur non pénalisée, tout en atteignant la valeur de parcimonie désirée pour une plage de valeurs de μ plus large que l'HALS non équilibré. Les variations de la parcimonie autour de la valeur attendue sont par ailleurs plus faibles pour l'algorithme équilibré. Les facteurs sont également plus précisement estimés. On notera que l'HALS pas équilibré atteint des valeurs de fonction de coût beaucoup plus élevées que la version initialisée avec l'équilibrage. L'algorithme pénalisé uniquement avec la norme ℓ_1 (setup 1 uniquement) induit une parcimonie qui n'améliore pas le FMS, le problème étant en fait mal posé. Cet aspect sera précisé sur le plan théorique dans une version étendue.

5 Conclusions et perspectives

Dans la littérature sur les factorisations matricielles et tensorielles régularisées, les facteurs non-pénalisés sont généralement normalisés. Cette normalization n'est pas anodine car la contrainte imposée est généralement non-convexe et non séparable. Dans ce court article j'étudie une alternative à la normalisation : pénaliser tous les facteurs. Je montre que l'invariance d'échelle du terme d'attache aux données induit une régularisation implicite qui tend en pratique à équilibrer les facteurs. Je montre empiriquement qu'il est bénéfique d'imposer explicitement cette normalisation dans un algorithme de décomposition tensorielle CP nonnégative. L'algorithme obtenu est également plus robuste au choix du paramètre de régularisation. Comme présenté dans l'article, ce travail fait également écho à des travaux récents sur l'équilibrage des couches cachées des réseaux de neurones[11].

[2] `https://github.com/cohenjer/shootout`
[3] `http://tensorly.org/viz/stable/`

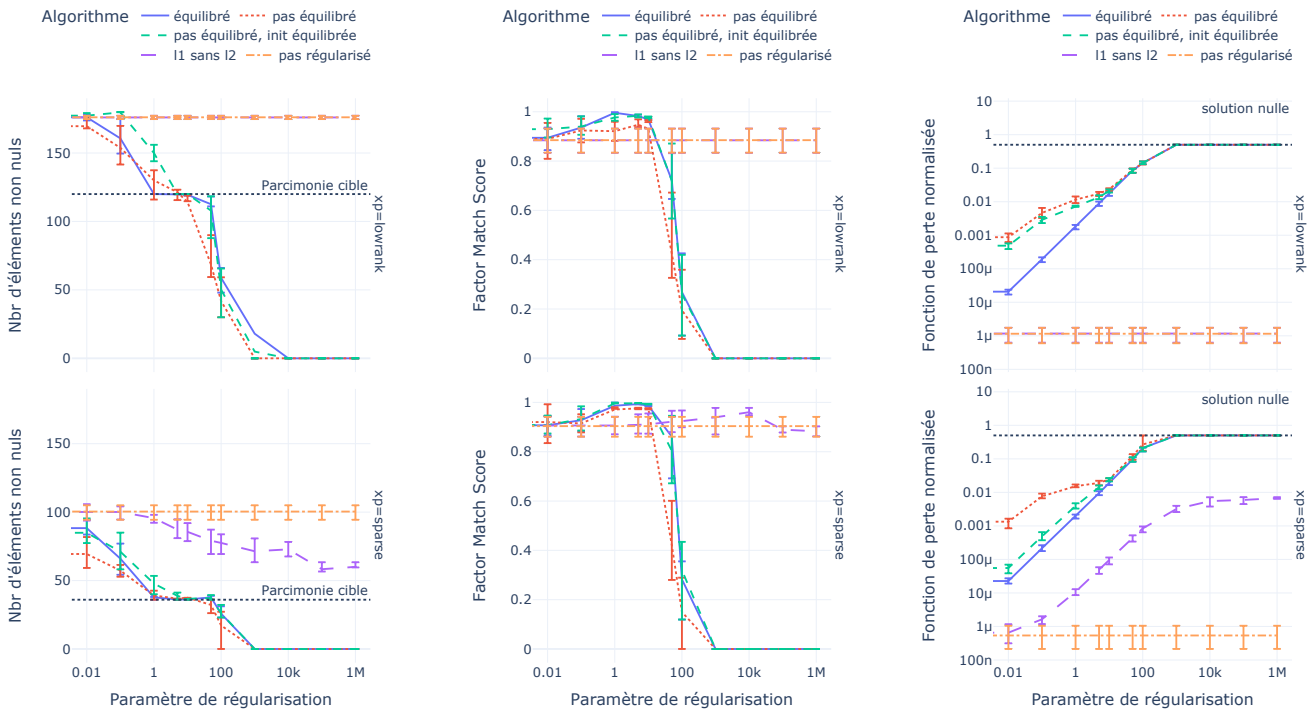

FIGURE 1 : (gauche) Parcimonie, (milieu) FMS et (droite) fonction de perte en fonction du paramètre de régularisation μ.

Les perspectives de ce travail sont diverses. Il devrait être possible de dériver des garanties sur la parcimonie des facteurs grâce à la formulation implicite. Une étude plus poussée de l'impact de la procédure d'équilibrage des facteurs devrait également être effectuée, par exemple sur données réelles, avec d'autres modèles notamment matriciels et pour d'autre algorithmes. De plus une comparaison avec d'autres approches (normalisation explicite des facteurs, pénalités 0-homogènes de type rapport de normes [5, 8]) devra être effectuée.

6 Remerciements

Merci à Valentin Leplat pour ses conseils sur la NMF parcimonieuse, et à Rémi Gribonval pour m'avoir orienté vers la littérature sur l'apprentissage profond. Ce travail a été financé par l'ANR JCJC LoRAiA ANR-20-CE23-0010.

Références

[1] Evrim ACAR, Tamara G KOLDA et Daniel M DUNLAVY : All-at-once optimization for coupled matrix and tensor factorizations. *arXiv preprint arXiv :1105.3422*, 2011.

[2] Alexis BENICHOUX, Emmanuel VINCENT et Rémi GRIBONVAL : A fundamental pitfall in blind deconvolution with sparse and shift-invariant priors. *In 2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013.

[3] Andrzej CICHOCKI, Anh Huy PHAN et Cesar CAIAFA : Flexible HALS algorithms for sparse non-negative matrix/tensor factorization. *In 2008 IEEE Workshop on Machine Learning and Signal Processing*, pages 73–78. IEEE, 2008.

[4] Tolga ERGEN et Mert PILANCI : Implicit convex regularizers of cnn architectures : Convex optimization of two-and three-layer networks in polynomial time. *arXiv preprint arXiv :2006.14798*, 2020.

[5] Patrik O HOYER : Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 5(9), 2004.

[6] Jean KOSSAIFI, Yannis PANAGAKIS, Anima ANANDKUMAR et Maja PANTIC : Tensorly : Tensor learning in python. *arXiv preprint arXiv :1610.09555*, 2016.

[7] Behnam NEYSHABUR, Russ R. SALAKHUTDINOV et Nati SREBRO : Path-SGD : Path-normalized optimization in deep neural networks. *Advances in Neural Information Processing Systems*, 28, 2015.

[8] Audrey REPETTI, Mai Quyen PHAM, Laurent DUVAL, Emilie CHOUZENOUX et Jean-Christophe PESQUET : Euclid in a taxicab : Sparse blind deconvolution with smoothed l1/l2 regularization. *IEEE Signal Processing Letters*, 22(5):539–543, 2014.

[9] Nathan SREBRO, Jason RENNIE et Tommi JAAKKOLA : Maximum-margin matrix factorization. *Advances in Neural Information Processing Systems*, 17, 2004.

[10] Pierre STOCK, Benjamin GRAHAM, Rémi GRIBONVAL et Hervé JÉGOU : Equi-normalization of neural networks. *International Conference on Learning Representations*, 2019.

[11] Ryan J. TIBSHIRANI : Equivalences between sparse models and neural networks. *Working Notes. URL https ://www. stat. cmu. edu/ryantibs/papers/sparsitynn. pdf*, 2021.