Apprentissage supervisé à noyau basé sur la décomposition Canonique Polyadique (CP)

Ouafae KARMOUDA[1], Jérémie BOULANGER[1], Rémy BOYER[1]

[1]Université de Lille, CNRS, CRIStAL, 59655 Lille, France
ouafae.karmouda@univ-lille.fr, jeremie.boulanger@univ-lille.fr
remy.boyer@univ-lille.fr

Résumé – Les décompositions tensorielles sont un cadre très prometteur pour l'apprentissage automatique. Dans le contexte de l'apprentissage supervisé des données multidimensionnelles, certaines méthodes populaires de l'état de l'art utilisent la décomposition Canonique Polyadique (CP) de rang faible pour l'extraction de caractéristiques. L'une des méthodes la plus connue de la littérature, DuSK, exploite un noyau gaussien-euclidien (GE) entre les facteurs respectifs de la décomposition CP de 2 tenseurs et effectue une classification basée sur les machines à vecteurs de support (SVM). Malheureusement, dans cette méthode, le choix de la fonction noyau n'est pas en mesure de gérer les ambiguïtés inhérentes au modèle CP. Dans ce travail, nous montrons comment les ambiguïtés liées au modèle CP corrompent l'algorithme SVM. Cette faiblesse explique les mauvaises performances de classification du DuSK sur des jeux de données réels. Pour améliorer les performances de classification, nous proposons de modifier le choix du noyau en se basant sur la géométrie grassmannienne qui permet de gérer efficacement les ambiguïtés CP.

Abstract – Tensor decompositions are a very promising framework for machine learning. In the context of supervised learning of multi-dimensional data, state-of-art methods use the low rank Canonical Polyadic Decomposition (CPD) for feature extraction. One of the most prominent state-of-art method, called DuSK, exploits a Gaussian-Euclidean (GE) kernel between CP factors and performs classification based on the Support Vector Machines (SVM) approach. Unfortunately, in this method, the choice of the kernel function is not able to manage the scaling ambiguities inherent of the CP model. In this work, we show how the CP scaling ambiguities corrupt the decision function in the SVM algorithm. These weaknesses explain the poor classification performance of the DuSK on real datasets. To improve the classification performance, we propose to modify the kernel choice based on the Grassmannian geometry that allows to efficiently manage the CP ambiguities.

1 Introduction

Les machines à vecteurs de support (SVM) ont été largement utilisées dans le contexte de l'apprentissage automatique [15, 10]) en raison de leurs bases théoriques solides [1, 3]. On s'intéresse dans ce travail à l'extension de cet algorithme aux données multidimensionnelles. Vectoriser les données détruit leur structure et n'exploite pas l'information qu'ils contiennent. Par conséquence, comme le montre [2] par exemple, les performances de classification sont médiocres. De plus, la complexité de cette approche est très élevée.
Plusieurs travaux dans la littérature se sont intéressés aux SVM pour les données tensorielles. Citons [12] qui décompose le paramètre du tenseur de poids des SVM avec la décomposition de Tucker [16]. Pour les tenseurs cubiques, [5] utilise un noyau convolutif qui est un filtre cubique appris à partir des données. La méthode de l'état de l'art de ce papier est le schéma DuSK [6] qui généralise les méthodes à noyaux aux tenseurs en utilisant la décomposition CP. Cependant, le noyau considéré qui est une mesure de similarité échoue dans cette tâche en raison des ambiguïtés inhérentes à la décomposition CP. Notre but dans ce travail n'est pas de proposer un nouveau noyau pour les données tensorielles mais de mettre en lumière les faiblesses du schéma DuSK et de combler ses lacunes. Plus spécifiquement, on va s'intéresser à la non-invariance du DuSK aux ambiguïtés d'échelle inhérentes à la décomposition CP. En se basant sur la géométrie de Grasmann, nous proposons une approche permettant au noyau DuSK d'être robuste face aux ambiguïtés citées.

2 Quelques notions en algèbre tensorielle

2.1 Notations et définitions

Dans ce travail, les scalaires seront désignés par des lettres minuscules (*eg* : x), les matrices seront désignées par des lettres majuscules (*eg* : X) tandis que les tenseurs seront désignés par des lettres calligraphiques (*eg* : $\mathcal{X}$)). La $(i_1, i_2, \cdots, i_Q)$-ième entrée du tenseur d'ordre Q $\mathcal{X}$ est notée $\mathcal{X}(i_1, i_2, \cdots, i_Q)$.
Le symbole "$\circ$" représente le produit extérieur vectoriel. Cela signifie que l'entrée $(i_1, i_2, \cdots, i_Q)$ du produit extérieur de Q vecteurs u_q est le produit des composantes $u_q(i_q)$.
Définition Un tenseur $\mathcal{X}$ d'ordre Q est un tableau multidimensionnel de taille $I_1 \times \ldots \times I_Q$. Dans la suite, nous supposons que tous les tenseurs sont réels.

Définition Le produit scalaire de deux tenseurs $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{I_1 \times \cdots \times I_Q}$ d'ordre Q est défini comme :

$$\langle \mathcal{X}, \mathcal{Y} \rangle = \sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \cdots \sum_{i_Q=1}^{I_Q} \mathcal{X}(i_1, \ldots, i_q)\mathcal{Y}(i_1, \ldots, i_q). \quad (1)$$

La norme de Frobenius d'un tenseur $\mathcal{X}$ est notée $||\mathcal{X}||_F$ et sa définition découle du produit scalaire définie dans l'eq. (1).

2.2 Décomposition Canonique Poyadic (CP) et ses propriétés

2.2.1 Définition

Un tenseur d'ordre Q de rang canonique R suit une décomposition CP s'il s'écrit comme la somme de R tenseurs de rang 1. Plus formellement, pour un tenseur $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_Q}$, sa décomposition CP est donnée par :

$$\mathcal{X} = \sum_{r=1}^{R} x_r^{(1)} \circ \cdots \circ x_r^{(Q)} = [\![X^{(1)}, \ldots, X^{(Q)}]\!], \quad (2)$$

où $x_r^{(q)} \in \mathbb{R}^{I_q}$ est la r-ième colonne de la q-ième matrice du facteurs $X^{(q)}$ avec $1 \leq q \leq Q$.
Le cout de stockage du tenseur $\mathcal{X}$ avant décomposition est de $O(I^Q)$. Avec la décomposition CP, le coût de stockage de $\mathcal{X}$ devient $O(QIR)$ où $I = \max\{I_1, \cdots, I_Q\}$. Pour le rang canonique R, nous supposons qu'il est *a priori* connu. Son estimation est un problème NP-difficile [7].

2.2.2 Unicité

L'une des propriétés les plus importantes de la décomposition CP est son unicité dans l'identifiabilité des facteurs CP. Une condition suffisante pour son unicité (voir eq. (2)) est [11] :

$$\sum_{q=1}^{Q} \mathrm{k}_{\mathrm{rank}}(X^{(q)}) \geq 2R + Q - 1, \quad (3)$$

où $\mathrm{k}_{\mathrm{rank}}$ (pour le rang kruskal) [13] est la valeur maximale k telle que toutes les colonnes k soient linéairement indépendantes [11]. En d'autres termes, la décomposition CP dans l'eq. (2) est unique aux deux ambiguïtés près suivantes :

— Permutation de colonnes : Les facteurs $X^{(q)}$ dans l'eq. (2) sont uniques à une permutation de colonne commune *i.e.*,

$$\mathcal{X} = [\![X^{(1)}, ..., X^{(Q)}]\!] = [\![X^{(1)}\Pi, ..., X^{(Q)}\Pi]\!], \quad (4)$$

pour toute matrice de permutation Π de taille $R \times R$.

— Mise à l'échelle de la colonne : Pour un couple (r, q) donné, la colonne $x_r^{(q)}$ dans l'eq. (2) est unique à un facteur d'échelle près, noté $\lambda_r^{(q)} \in \mathbb{R} \setminus \{0\}$, *i.e.*,

$$\mathcal{X} = \sum_{r=1}^{R} (\lambda_r^{(1)} x_r^{(1)}) \circ \cdots \circ (\lambda_r^{(Q)} x_r^{(Q)}), \quad (5)$$

avec $\prod_{q=1}^{Q} \lambda_r^{(q)} = 1$.

Dans la suite, on va s'intéresser au problème de classification des données multidimensionnelles en utilisant les machines à vecteurs support à noyau. Cet algorithme se base sur la matrice noyau qui contient des valeurs décrivant les similitudes entre les différents tenseurs d'une base de données. Nous discutons alors la manière de définir un noyau tensoriel tenant compte de la structure multidimensionnelle des données.

3 La méthode DuSK

La méthode DuSK [6] est l'une des premières méthodes à généraliser les SVMs à noyaux pour l'apprentissage des tenseurs. L'étape d'extraction des "features" est basée sur la décomposition CP. La fonction de projection Φ proposée est la suivante :

$$\Phi : \sum_{r=1}^{R} x_r^{(1)} \circ ... \circ x_r^{(Q)} \rightarrow \sum_{r=1}^{R} \phi_1(x_r^{(1)}) \circ ... \circ \phi_Q(x_r^{(Q)}) \quad (6)$$

où ϕ_q est une fonction qui projette les éléments de $\mathbb{R}^{I_q}$ dans un espace pré-Hilbertien pour $1 \leq i_q \leq Q$.
Afin de calculer le noyau sur le couple $(\mathcal{X}_i, \mathcal{X}_j)$, la méthode DuSK considère leurs décomposition CP : $\mathcal{X}_i = \sum_r^R x_r^{(1)} \circ ... \circ x_r^{(Q)}$ et $\mathcal{X}_j = \sum_{r'}^R x_r'^{(1)} \circ ... \circ x_r'^{(Q)}$. Ensuite, le noyau k_{DuSK} proposé entre $\mathcal{X}_i$ et $\mathcal{X}_j$ est défini par :

$$k_{\text{DuSK}}(\mathcal{X}_i, \mathcal{X}_j) = \frac{1}{R} \sum_{r=1}^{R} \sum_{r'=1}^{R} \prod_{q=1}^{Q} \exp\left(-\gamma ||x_r^{(q)} - x'^{(q)}_{r'}||^2\right), \quad (7)$$

où γ est un hyperparamètre du noyau gaussien standard qui est utilisé pour définir une bande passante appropriée.

4 Effet des ambiguités de la décomposition CP sur DuSK

Le noyau Dusk évalue la similitude entre deux tenseurs à travers leurs décompositions CP. Plus spécifiquement, il évalue les similitudes entre les colonnes des facteurs CP des 2 tenseurs qui appartiennent au même mode à travers des sous noyaux gaussiens en utilisant la distance euclidienne. Or, les colonnes des facteurs CP sont obtenus à un facteur multiplicatif inconnu près du fait des ambiguïtés de la décomposition CP. Ceci entraine une mauvaise estimation de la similitude donnée par les sous noyaux qui, en se cumulant engendrent une mauvaise estimation de la similitude de deux tenseurs. En effet, cette dernière devrait être égale à 1 pour les tenseurs issus de la même classe et 0 si les tenseurs sont issus de classes différentes. Or, en utilisant l'eq. (7), on peut montrer que la similitude de deux tenseurs de la même classe ou encore plus, pour les mêmes tenseurs avec deux décompositions CP différentes tend vers 0 asymptotiquement avec l'ordre du tenseur. En effet, à cause des ambiguïtés liées à la CP, les colonnes des facteurs CP ne sont pas sur les mêmes

échelles, ce qui implique que leur différence en norme est largement impactée. Cette dernière peut être largement grande (resp.petite) pour les mêmes vecteurs (resp. des vecteurs non similaires), ce qui pousse le terme avec l'exponentielle dans l'eq. (7) à être nul (resp. égale à 1) et donc la similitude pour les mêmes tenseurs à être nulle (resp. très grande pour des tenseurs différents). Comme la fonction noyau est utilisée pour entrainer les SVM, les performances de classification seront négativement impactées. L'effet de ces ambiguités peut être diminué en normalisant les colonnes des facteurs CP sans les éliminer. On appellera cette méthode par NDuSK. Par contre, l'ambiguité du signe sera toujours présente, ce qui impacte significativement les résultats de classification.

5 Approche proposée

Afin que le noyau défini à l'eq. (7) soit robuste aux ambiguïtés de la décomposition CP vues à la section 2.2.2, les vecteurs $\lambda_r^q x_r^q$ *i.e.* les vecteurs de la droite vectorielle engendrée par les x_r^q devraient être considérés similaires par la fonction noyau. De ce fait, on veut comparer les droites vectoriels plutôt que les facteurs colonnes. Pour cela, l'idée est de considérer dans l'eq. (7) les distances entre les sous espaces qu'engendrent $x_r^{(q)}$ et ${x'}_{r'}^{(q)}$. Ces sous espaces définissent ce qu'on appelle par une variété grassmannienne définie dans la section suivante :

5.1 Manifold Grassmannien

Definition Pour les entiers $m > n \geq 0$, la variété de Grassmann [17] notée $\mathbb{G}(m,n)$ est l'ensemble des sous-espaces linéaires de dimension n de $\mathbb{R}^m$. Formellementt :

$$\mathbb{G}(m,n) = \{span(X) : X \in \mathbb{R}^{m\times n}, X^T X = I_n\}. \quad (8)$$

5.2 Noyau Grassmannien

Une distance appropriée entre $span(x_r^{(q)})$ et $span({x'}_{r'}^{(q)})$ qui donne lieu à un noyau gaussien défini positif sur $\mathbb{G}(q,1)$ est la distance chordale [9]. Ceci est donné par la norme de Frobenius de la différence entre les projecteurs sur $span(x_r^{(q)})$ et $span(x_{r'}^{\prime(q)})$ qui s'avère être la norme l_2 du vecteur formé par l'angle principal entre les deux sous-espaces [9, 8] :

$$\begin{aligned} d_c(span(x_r^{(q)}), span({x'}_{r'}^{(q)})) =& ||\Pi_{x_r^{(q)}} - \Pi_{{x'}_{r'}^{(q)}}||_F \\ =& \sqrt{2}|\sin(\theta)|, \end{aligned}$$

où θ est l'angle principal entre $span(x_r^{(q)})$ et $span(x_{r'}^{\prime(q)})$ et $\Pi_{x_r^{(q)}}$ (*resp.* $\Pi_{x_{r'}^{\prime(q)}}$ est le projecteur orthogonal sur $span(x_r^{(q)})$ (resp. $span(x_{r'}^{\prime(q)})$) défini comme :

$$\Pi_{x_r^{(q)}} = \frac{x_r^{(q)} x_r^{(q)^T}}{||x_r^{(q)}||^2} \quad \left(resp.\ \Pi_{{x'}_{r'}^{(q)}} = \frac{{x'}_{r'}^{(q)} {x'}_{r'}^{(q)^T}}{||{x'}_{r'}^{(q)}||^2} \right).$$

Il est à noter que le projecteur ci-dessus est invariant à toute échelle multiplicative non nulle $\lambda_r^{(q)}$ agissant sur $x_r^{(q)}$ (*resp.* $x_{r'}^{\prime(q)}$), ce qui signifie que

$$span(\lambda_r^{(q)} x_r^{(q)}) = span(\Pi_{\lambda_r^{(q)} x_r^{(q)}}) = span(\Pi_{x_r^{(q)}}) = span(x_r^{(q)}).$$

En utilisant la distance définie dans l'eq. (9), le noyau Gaussien-Grassmann (GG) est défini par :

$$k_{GG}(\mathcal{X}_i, \mathcal{X}_j) := \sum_{r=1}^{R}\sum_{r'=1}^{R}\prod_{q=1}^{Q} \exp\left(-\gamma d_c^2(span(x_r^{(q)}), span({x'}_{r'}^{(q)})\right). \quad (9)$$

Il est à noter que le noyau k_{GG} proposé est invariant :

— A la permutation (commune) des colonnes entre facteurs du fait de la commutativité de la somme exactement comme le noyau k_{DuSK}.

— Au facteur multiplicatif propre à chaque colonne des facteurs du fait de l'invariance d'échelle des projecteurs, ce qui n'est pas géré par le noyau k_{DuSK}.

6 Expériences numériques

6.1 Bases de données

— **UCF11** : Cette base de données [14] contient 1600 clips vidéo contenant 11 actions humaines telles que : *plonger*, *sauter*, *marcher*. Deux actions humaines sont choisies : *saut en trampoline* et *marche*. Elles représentent 2 classes pour la tache de classification considérée. Pour chaque clip vidéo, nous considérons une séquence de 240 images RBG où la résolution de chaque image est de 320 $\times$240 $\times$ 3. Ces clips vidéos peuvent être interprétés comme des tenseurs d'ordre 4 avec des dimensions 240 $\times$240 $\times$ 320 $\times$ 3. Un total de 109 tenseurs sont présents dans chaque classe.

— **Extended Yale B** : Cette base de données [4] contient 28 images de visgaes différentes. Pour chaque sujet, il y a 576 images de taille 480×640 prises sous 9 poses différentes. Chaque pose est prise sous 64 illuminations différentes. Dans ce cas, 3 sujets sont arbitrairement choisis. Ceci représente 3 classes pour un problème de classification. Afin de construire l'ensemble d'entraînement et l'ensemble de test, nous décomposons le tenseur de chaque sujet en 16 tenseurs en considérant chaque 4 illuminations dans un tenseur de taille $9 \times 480 \times 640 \times 4$.

6.2 Performances de classification

L'hyper-paramètre des SVM qui sert à contrôler le compromis entre nombre d'erreurs de classification et la largeur de la marge avec le plan séparateur ainsi que le γ défini dans l'éq. (7) sont sélectionnés dans la grille de valeurs

$\{2^{-9}, 2^{-8}, ..., 2^8, 2^9\}$ par une validation croisée. Le rang R du modèle CP est un hyper-paramètre qui est inférieur ou égal à la plus petite dimension des tenseurs d'entrée.Pour les deux bases de données considérées, on considère de façon aléatoire 60 % de la taille de la base de donnée pour constituer l'ensemble d'apprentissage et 40 % pour le test. Cette expérience est effectuée 5 fois on fournit alors le score moyen avec l'écart type.

Method/R	1	2	3
DuSK	$0.28(10^{-2})$	$0.27(10^{-2})$	$0.28(10^{-2})$
NDuSK	$0.88(0.12)$	$0.9(10^{-2})$	$0.77(10^{-2})$
Notre approche	$\mathbf{1}(0)$	$\mathbf{1}(0)$	$\mathbf{1}(0)$

TABLE 1 – Scores moyens de classification des différentes méthodes sur l'Extended Yale B par rapport à R : moyenne (écart type)

Method/R	1	2	3
DuSK	$0.48(10^{-2})$	$0.54(10^{-2})$	$0.52(10^{-2})$
NDuSK	$0.65(10^{-2})$	$0.57(10^{-2})$	$0.63(10^{-2})$
Notre approche	$\mathbf{0.82}(10^{-2})$	$\mathbf{0.84}(10^{-2})$	$\mathbf{0.72}(10^{-2})$

TABLE 2 – Scores moyens de classification des différentes méthodes sur l'UCF11 par rapport à R : moyenne (+/- écart type)

— D'après les Tables. 1 et 2, on constate la supériorité de notre approche pour les différentes valeurs possibles de R sur les deux jeux de données considérés.

7 Conclusion

Nous nous sommes intéressés dans ce travail aux machines à vecteurs supports (SVM) pour les données multidimensionnelles. Nous avons montrés comment les ambiguïtés du modèle CP corrompe les performances de classification de la méthode DuSK. Nous avons alors proposé de modifier le noyau de telle façon à ce qu'il puisse être robuste à ces ambiguïtés. Pour ce faire, nous nous sommes basés sur la géométrie grassmannienne qui permet de bien gérer ces ambigüités. Notre approche a été validé par différentes expériences numériques sur des bases de données réelles.

Références

[1] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2) :121–167, 1998.

[2] Cong Chen, Kim Batselier, Wenjian Yu, and Ngai Wong. Kernelized support tensor train machines, 2020.

[3] N. Cristianini, C. Campbell, and C. Burges. Editorial : Kernel methods : Current research and future directions. *Mach. Learn.*, 46(1–3) :5–9, March 2002.

[4] A.S Georghiades, P.N. Belhumeur, and D.J. Kriegman. From few to many : Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. Pattern Anal. Mach. Intelligence*, 23(6) :643–660, 2001.

[5] Ashish Gupta, Murat Seçkin Ayhan, and Anthony S. Maida. Natural image bases to represent neuroimaging data. In *Proceedings of the 30th International Conference on International Conference on Machine Learning*, 2013.

[6] Lifang He, Xiangnan Kong, Philip S. Yu, Ann B. Ragin, Zhifeng Hao, and Xiaowei Yang. Dusk : A dual structure-preserving kernel for supervised tensor learning with applications to neuroimages. *CoRR*, abs/1407.8289, 2014.

[7] Christopher J. Hillar and Lek-Heng Lim. Most tensor problems are np-hard. *J. ACM*, 60(6), November 2013.

[8] Z. Huang, R. Wang, S. Shan, and X. Chen. Projection metric learning on grassmann manifold with application to video based face recognition. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 140–149, 2015.

[9] Sadeep Jayasumana, Richard I. Hartley, Mathieu Salzmann, Hongdong Li, and Mehrtash Tafazzoli Harandi. Kernel methods on riemannian manifolds with gaussian RBF kernels. *CoRR*, abs/1412.0265, 2014.

[10] Hyunsoo Kim, Peg Howland, and Haesun Park. Dimension reduction in text classification with support vector machines. *Journal of machine learning research*, 6(Jan) :37–53, 2005.

[11] Tamara G. Kolda and Brett W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3) :455–500, September 2009.

[12] I. Kotsia and I. Patras. Support tucker machines. In *CVPR 2011*, pages 633–640, 2011.

[13] J. B. Kruskal. Three-way arrays : rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra and its Applications*, 18(2) :95 – 138, 1977.

[14] J. Liu, Jiebo Luo, and M. Shah. Recognizing realistic actions from videos "in the wild". In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[15] Simon Tong and Edward Chang. Support vector machine active learning for image retrieval. In *Proceedings of the ninth ACM international conference on Multimedia*, 2001.

[16] Ledyard R Tucker. Implications of factor analysis of three-way matrices for measurement of change. *Problems in measuring change*, 15(122-137) :3, 1963.

[17] Jiayao Zhang, Guangxu Zhu, Robert W. Heath Jr., and Kaibin Huang. Grassmannian learning : Embedding geometry awareness in shallow and deep learning. *CoRR*, abs/1808.02229, 2018.