Adaptation de domaine non supervisée par pondération d'importance pour des données tensorielles

Duc Hoan NGUYEN Ricardo BORSOI Sebastian MIRON David BRIE
Université de Lorraine, CNRS, CRAN, Vandoeuvre-lès-Nancy, France

Résumé – Les tenseurs permettent une représentation naturelle des données multidimensionnelles. Cependant, dans les tâches d'apprentissage automatique, le manque d'échantillons tensoriels étiquetés pour la phase d'entraînement rend souvent nécessaire l'utilisation de méthodes d'adaptation de domaine (DA). La majorité des approches existantes en adaptation de domaine non supervisée (UDA) apprennent les caractéristiques à partir des versions vectorisées des données tensorielles, sans exploiter leur structure multilinéaire. Dans cette communication, nous proposons une nouvelle méthode d'adaptation de domaine non supervisée pour les données tensorielles, en tirant parti de leur structure de rang faible. Cette approche repose sur un schéma de régularisation par pondération d'importance dans les espaces de Hilbert à noyau reproduisant (RKHS), et sur l'estimation de la dérivée de Radon-Nikodym (R-ND) pour les distributions marginales des domaines source et cible. L'efficacité de la méthode proposée est validée par des simulations numériques.

Abstract – Tensors provide a natural representation of multidimensional data. However, in machine learning tasks, the lack of labeled tensor samples for the training phase often necessitates the use of domain adaptation (DA) methods. Most existing approaches to unsupervised domain adaptation (UDA) learn features from vectorized versions of tensor data, failing to exploit their multilinear structure. In this work, we introduce a novel unsupervised domain adaptation method for tensor data that leverages their low-rank structure. This approach relies on a regularization scheme based on importance weighting in reproducing kernel Hilbert spaces (RKHS) and on the estimation of the Radon-Nikodym derivative (R-ND) to align the marginal distributions of the source and target domains. The effectiveness of the proposed method is validated through numerical simulations.

1 Introduction

Les décompositions tensorielles fournissent un cadre efficace pour représenter des données multidimensionnelles, en capturant des relations complexes avec un petit nombre de paramètres. Cet article s'intéresse à l'apprentissage non-supervisé de données tensorielles, qui est un problème intervenant dans de nombreuses applications, telles que l'imagerie médicale et la télédétection. Cependant, ces applications peuvent souffrir d'un manque de données d'apprentissage étiquetées. De plus, pour deux jeux de données différents, les distributions peuvent varier : c'est l'hypothèse de « décalage de covariables ». Dans ce type de situation, l'apprentissage sur un premier jeu de données (source) peut s'avérer inopérant sur l'autre jeu de données (cible). L'adaptation de domaine non-supervisée (UDA) est une approche visant à résoudre ce type de problème.

L'UDA repose essentiellement sur les notions de pondération d'importance et d'apprentissage de représentation [3] largement étudiées pour les données vectorielles. L'application de ces méthodes aux tenseurs repose souvent sur une vectorisation des données, entraînant d'une part, la perte de la structure multidimensionnelle des données et, d'autre part, une complexité importante. Ce travail propose d'utiliser les décompositions tensorielles de rang faible, telles que les décompositions Canonique Polyadique (CP) ou de Tucker (pour plus d'informations sur ces décompositions tensorielles, voir [5, 10]) qui préservent la structure multidimensionnelle et atténuent le « fléau de la dimension ». Cette idée a été explorée dans le contexte de l'UDA [1, 7] pour l'apprentissage de représentation. En revanche, il n'existe à notre connaissance aucun travaux traitant de l'utilisation des décompositions de rang faible pour la pondération d'importance qui est l'approche la plus fréquemment adoptée sous l'hypothèse de décalage des covariables.

Cet article propose une approche pour la pondération d'importance reposant sur une décomposition de rang faible de type Tucker. Elle généralise ce modèle tensoriel en exploitant le schéma de régularisation linéaire dans les espaces de Hilbert à noyau reproduisant (RKHS) [3]. La pondération de l'importance est fondamentalement liée à la dérivée de Radon-Nikodym (R–ND). Nous incluons également un estimateur régularisé de la R–ND inspiré de [8]. Différentes expériences numériques montrent que la prise en compte de la structure tensorielle des données permet une amélioration des performances de classification sur des données simulées et réelles.

Notations : Les tenseurs sont désignés par des caractères calligraphiques, par exemple $\mathcal{X}$; les matrices par des caractères majuscules gras, par exemple $\mathbf{X}$; et les vecteurs par des caractères minuscules gras, par exemple $\mathbf{x}$. L'ordre d'un tenseur est le nombre de dimensions (modes). L'élément $(i_1, i_2, \ldots, i_M)$ d'un tenseur $\mathcal{X}$ d'ordre M est noté par $x_{i_1 i_2 \ldots i_M}$. *La matricisation ou dépliement* d'un tenseur est le processus de réorganisation des éléments d'un tenseur en une matrice. Plus précisément, la matricisation selon le mode-m d'un tenseur $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M}$ est désignée par $\mathbf{X}_{(m)} \in \mathbb{R}^{I_m \times I_1 \cdots I_{m-1} I_{m+1} \cdots I_M}$, tandis que la vectorisation d'un tenseur est notée $\mathrm{vec}(\mathcal{X})$. Le produit mode-$m$ d'un tenseur $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_M}$ avec une matrice $\mathbf{U} \in \mathbb{R}^{J \times I_m}$ est noté $\mathcal{Y} = \mathcal{X} \times_m \mathbf{U}$ et peut être défini à l'aide de la matricisation mode-m selon $\mathbf{Y}_{(m)} = \mathbf{U}\mathbf{X}_{(m)}$. Nous utiliserons des lettres gothiques ($\mathfrak{A}$) pour désigner des ensembles ou des espaces

généraux, indépendamment de leur nature spécifique.

2 Espaces de Hilbert à noyaux reproduisants

La théorie des RKHS est l'un des domaines les plus développés de l'apprentissage statistique. Nous désignons un RKHS par $\mathfrak{H}$, et nous le représentons également par $(\mathfrak{H}, \langle\cdot,\cdot\rangle_{\mathfrak{H}})$ pour souligner le fait que $\mathfrak{H}$ est doté d'une structure d'espace de Hilbert, définie par le produit scalaire $\langle\cdot,\cdot\rangle_{\mathfrak{H}}$. Soit $\mathfrak{X}$ un sous-ensemble arbitraire de $\mathbb{R}^{I_1\times I_2\times\cdots\times I_M}$. Il est important de noter que $\mathfrak{X}$ peut représenter un sous-ensemble de scalaires, de vecteurs, de matrices ou, plus généralement, de tenseurs d'ordre $\leq M$. Un espace de Hilbert $(\mathfrak{H}, \langle\cdot,\cdot\rangle_{\mathfrak{H}})$ de fonctions $f : \mathfrak{X} \to \mathbb{R}$ est un RKHS s'il existe une fonction $k : \mathfrak{X} \times \mathfrak{X} \to \mathbb{R}$ satisfaisant les propriétés suivantes : (i) $k_{\mathcal{X}} := k(\mathcal{X},\cdot) \in \mathfrak{H}$ pour tout $\mathcal{X} \in \mathfrak{X}$, (ii) $f(\mathcal{X}) = \langle f, k(\mathcal{X},\cdot)\rangle_{\mathfrak{H}}$ pour tout $\mathcal{X} \in \mathfrak{X}$ et pour tout $f \in \mathfrak{H}$. Nous appelons k un noyau reproduisant pour $\mathfrak{H}$ et notons le RKHS correspondant par $\mathfrak{H}_k$.

Noyaux pour les données tensorielles

Le choix du noyau joue un rôle crucial car il a un impact direct sur la régularité, l'expressivité et la capacité de généralisation des fonctions apprises. Les travaux de [10] ont étudié différentes définitions de noyaux pour des données à valeurs tensorielles. Nous avons choisi de nous focaliser sur la définition d'un noyau qui prend en compte explicitement une structure de rang faible des données tensorielles par l'intermediaire d'une décomposition HOSVD (Higher-Order Singular Value Decomposition) tronquée de rang $(R_1,\ldots,R_M)$ [2]. Un tenseur $\mathcal{X} \in \mathbb{R}^{I_1\times\cdots\times I_M}$ de rang $(R_1,\ldots,R_M)$ peut s'écrire selon : $\mathcal{X} = \mathcal{C}\times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \times_3 \cdots \times_M \mathbf{U}_M$ où $\mathbf{U}_m \in \mathbb{R}^{I_m\times R_m}$ est une matrice orthogonale, $\mathcal{C} \in \mathbb{R}^{R_1\times\cdots\times R_M}$ est appelé tenseur coeur. En pratique, les matrices $\mathbf{U}_m$ sont obtenues par troncature de rang R_m de la décomposition en valeurs singulières (SVD) du tenseur $\mathcal{X}$ déplié selon le mode m :

$$\mathbf{X}_{(m)} \simeq \mathbf{U}_m \mathbf{\Sigma}_m \mathbf{V}_m^T . \tag{1}$$

L'ensemble des paramètres du tenseur original est de l'ordre de $\prod_m I_m$. Cependant, la décomposition HOSVD permet d'obtenir une représentation plus compacte avec seulement $\sum_m I_m R_m + \prod_m R_m$ paramètres. Lorsque les rangs sont faibles, c'est-à-dire lorsque $R_m \ll I_m$, la HOSVD entraîne une compression significative du nombre de paramètres.

Nous proposons d'utiliser une classe de fonctions noyaux basées sur la distance chordale sur les variétés grassmanniennes, définie en fonction des matrices $\mathbf{V}_m$ comme [10] :

$$k(\mathcal{X},\mathcal{X}') = \prod_{m=1}^{M} \exp\left(-\frac{1}{2\gamma_m^2}\|\mathbf{V}_m\mathbf{V}_m^T - \mathbf{V}'_m\mathbf{V}'^T_m\|_F^2\right),$$

où γ_m est un paramètre. La distance chordale mesure les angles principaux entre les sous-espaces. Plus précisément, elle mesure la distance euclidienne entre les matrices de projection représentant les sous-espaces. Ces matrices de projection restent inchangées en cas de rotations et de réflexions des sous-espaces à droite (des vecteurs ligne) de $\mathbf{X}_{(m)}$. Ce noyau possède donc des propriétés d'invariance de rotation et de réflexion pour les éléments de la variété grassmannienne [10].

3 UDA régularisée pour les tenseurs

UDA avec décalage de covariables

Considérons un tenseur d'entrée $\mathcal{X} \in \mathfrak{X} \subset \mathbb{R}^{I_1\times I_2\times\cdots\times I_M}$ d'ordre M et une variable/étiquette de sortie $y \in \mathfrak{Y} \subset \mathbb{R}$, que l'on suppose régies par des mesures de probabilité différentes $p(\mathcal{X},y)$ et $q(\mathcal{X},y)$ sur $\mathfrak{X}\times\mathfrak{Y}$. Dans le contexte de l'adaptation de domaine, p et q sont appelées respectivement la probabilité source et la probabilité cible. Nous nous concentrons sur l'apprentissage avec coût quadratique, où le risque moyen de prédire y à partir de $\mathcal{X}$ à l'aide d'une fonction $f : \mathfrak{X} \to \mathfrak{Y}$ est défini dans le domaine cible comme suit

$$\mathcal{R}_q(f) := \int_{\mathfrak{X}\times\mathfrak{Y}} (f(\mathcal{X}) - y)^2 \, dq.$$

On vérifie aisément que $\mathcal{R}_q(f)$ atteint son minimum pour $f_q(\mathcal{X}) = \int_{\mathfrak{Y}} y \, d\rho(y \mid \mathcal{X})$, où $\rho(y \mid \mathcal{X})$ est la probabilité conditionnelle de y sachant $\mathcal{X}$.

Cependant, dans le cadre de l'UDA, ni $\mathcal{R}_q(f)$ ni $f_q(\mathcal{X})$ ne peuvent être calculés directement, car la distribution cible q n'est pas entièrement accessible. Au lieu de cela, nous avons accès à un ensemble de données sources étiquetées $\mathbf{Z} = \{(\mathcal{X}^{(i)}, y^{(i)})\}_{i=1}^{N_s}$, où les échantillons suivent la mesure source p et un ensemble de données cible non étiqueté $\mathbf{Z}' = \{\mathcal{X}'^{(j)}\}_{j=1}^{N_t}$, dont les échantillons sont tirés de la mesure cible q. L'objectif est d'exploiter les échantillons cibles disponibles ainsi que les données sources étiquetées afin d'obtenir une approximation du minimiseur idéal f_q par l'estimateur empirique $f_{\mathbf{Z},\mathbf{Z}'}$. L'hypothèse de décalage des covariables garantit la faisabilité de la résolution du problème UDA sans les étiquettes du domaine cible.

Définition 1 *([4]) L'hypothèse de décalage des covariables est valide si les distributions conjointes $p(\mathcal{X},y)$, $q(\mathcal{X},y)$ peuvent être factorisées selon :*

$$p(\mathcal{X},y) = \rho(y|\mathcal{X})\rho_S(\mathcal{X}), \quad q(\mathcal{X},y) = \rho(y|\mathcal{X})\rho_T(\mathcal{X}), \tag{2}$$

où $\rho_S(\mathcal{X})$ et $\rho_T(\mathcal{X})$ sont les distributions marginales des entrées dans les domaines source (S) et cible (T), et $\rho(y|\mathcal{X})$ est la distribution conditionnelle ***commune aux deux domaines****.*

Conformément au cadre proposé dans [4], nous supposons l'existence d'une fonction $\beta : \mathfrak{X} \to \mathbb{R}_+$ telle que $d\rho_T(\mathcal{X}) = \beta(\mathcal{X}) d\rho_S(\mathcal{X})$. Ici, $\beta(\mathcal{X})$ représente le R-ND $\frac{d\rho_T}{d\rho_S}$, qui caractérise le rapport de densité entre les distributions de la cible et de la source.

Supposons que la fonction de régression f_q appartienne à un RKHS $\mathfrak{H}_k$ et que nous avons accès aux valeurs $\beta_i = \beta(\mathcal{X}^{(i)}), i = 1,2,\ldots,N_s$. Dans le cadre de la méthode des moindres carrés régularisés pondérés par β_i [3], l'approximant $f^{\lambda}_{\mathbf{Z},\mathbf{Z}'}$ de f_q est obtenu en tant que minimiseur du risque empirique pondéré régularisé

$$\mathcal{R}_{\mathbf{Z},\mathbf{Z}',\lambda,\beta}(f) = \frac{1}{N_s}\sum_{i=1}^{N_s} \beta_i \left(f(\mathcal{X}^{(i)}) - y_i\right)^2 + \lambda\|f\|^2_{\mathfrak{H}_k},$$

où λ est le paramètre de régularisation. Consdérons les deux opérateurs définis selon

$$S_{X_T} f = (f(\mathcal{X}'^{(1)}), f(\mathcal{X}'^{(2)}), \ldots, f(\mathcal{X}'^{(N_t)})) \in \mathbb{R}^{N_t},$$
$$S_{X_S} f = (f(\mathcal{X}^{(1)}), f(\mathcal{X}^{(2)}), \ldots, f(\mathcal{X}^{(N_s)})) \in \mathbb{R}^{N_s},$$

agissant de $\mathfrak{H}_k$ vers $\mathbb{R}^{N_t}$ et $\mathbb{R}^{N_s}$. La R–ND β_i étant non négative, le risque empirique $\mathcal{R}_{\mathbf{Z},\mathbf{Z}',\lambda,\beta}$ s'écrit :

$$\mathcal{R}_{\mathbf{Z},\mathbf{Z}',\lambda,\beta}(f) = \left\| \mathbf{B}^{\frac{1}{2}} S_{X_S} f - \mathbf{B}^{\frac{1}{2}} \mathbf{y} \right\|^2_{\mathbb{R}^{N_s}} + \lambda \|f\|^2_{\mathfrak{H}_K},$$

où $\mathbf{B}^{\frac{1}{2}} = diag(\sqrt{\beta_1}, \sqrt{\beta_2}, \ldots, \sqrt{\beta_n})$ et $\mathbf{y} = (y_i)_{i=1}^{N_s}$. Le minimiseur de $\mathcal{R}_{\mathbf{Z},\mathbf{Z}',\lambda,\beta}$ est de la forme $f^{\lambda}_{\mathbf{Z},\mathbf{Z}'} = (\lambda \mathbf{I} + S^*_{X_S} \mathbf{B} S_{X_S})^{-1} S^*_{X_S} \mathbf{B}\mathbf{y}$ et peut être considéré comme une solution approchée de l'équation normale $S^*_{X_S} \mathbf{B} S_{X_S} f = S^*_{X_S} \mathbf{B}\mathbf{y}$, régularisée par la perturbation $\lambda \mathbf{I} f$, puisque $S^*_{S_X} B S_{S_X}$ est un opérateur auto-adjoint, non négatif et compact sur $\mathfrak{H}_k$. Ici, $\mathbf{I}$ est la matrice identité de taille $N_s \times N_s$ et $S^*_{X_S}$ est l'opérateur adjoint de S_{X_S}. Compte tenu du théorème de la représentation [9], il apparaît que $f^{\lambda}_{\mathbf{Z},\mathbf{Z}'}$ s'écrit selon

$$f^{\lambda}_{\mathbf{Z},\mathbf{Z}'}(\cdot) = \sum_{i=1}^{N_s} c_i k(\cdot, \mathcal{X}^{(i)}) \tag{3}$$

où les coefficients $\mathbf{c} = (c_1, c_2, ..., c_{N_s})$ sont solutions du système linéaire $N_s^{-1}(\lambda \mathbf{I} + \mathbf{B}\mathbf{K})\mathbf{c} = \mathbf{B}\mathbf{y}$ avec $\mathbf{K}$ la matrice de Gram d'élément $k_{ij} = k(\mathcal{X}^{(i)}, \mathcal{X}^{(j)}), i, j = 1, 2, \ldots, N_s$.

Estimation régularisée de la dérivée de Radon-Nikodym

Dans la section précédente, nous avons supposé connues les valeurs de R–ND β_i, pour $i = 1, 2, \ldots, N_s$. Cependant, dans la pratique, ces valeurs ne sont pas directement accessibles. Il s'agit donc d'obtenir une estimation de la R–ND $\beta(\mathcal{X}) = \frac{d\rho_T}{d\rho_S}$ pour ensuite l'intégrer dans l'estimation de $f^{\lambda}_{\mathbf{Z},\mathbf{Z}'}$ (3). Nous supposons que β appartient à un RKHS (éventuellement différent), toujours désigné par $\mathfrak{H}_k$ pour simplifier la notation. Supposons que $\mathfrak{H}_k$ contient toutes les fonctions constantes.

En pratique, ρ_S et ρ_T sont inconnus et nous n'avons accès aux observations empiriques qu'à travers les échantillons $X_S = \{\mathcal{X}^{(i)}\}_{i=1}^{N_s}$ et $X_T = \{\mathcal{X}'^{(j)}\}_{j=1}^{N_t}$. Ainsi, en suivant [3,8], nous considérons le problème de dimension finie

$$S^*_{X_S} S_{X_S} \beta = S^*_{X_T} S_{X_T} \mathbf{1}, \tag{4}$$

qui sert d'équation empirique pour estimer β. Ici, $\mathbf{1}$ désigne le vecteur constant égal à 1. En utilisant la régularisation itérative de Laverentiev (voir, e.g., [3, 8]) pour résoudre l'équation (4), l'approximation de R–ND $\hat{\beta}_\alpha$ peut être obtenue de manière itérative [8] avec le paramètre de régularisation α.

4 Expériences numériques

L'objectif de cette section est d'évaluer les performances de l'approche proposée et de les comparer aux méthodes de l'état de l'art sur des données de simulation contrôlées ainsi que sur des données réelles classiquement utilisées en adaptation de domaine.

Méthodes évaluées

Deux principaux types de méthodes sont considérées : (i) le modèle est entraîné sur les données sources étiquetées et appliqué directement aux données cibles sans aucune adaptation de domaine - moindres carrés régularisé (RLS) (3) avec $\mathbf{B} = \mathrm{diag}(1, 1, \ldots, 1)$ — ou par régression fondée sur la décomposition de Tucker (TTR) [6] ; (ii) avec UDA appliquée soit directement aux données vectorisées (Standard UDA) [3], soit aux données tensorielles (TUDA) avec noyau gaussien et noyau chordal. Notons que le noyau gaussien opère directement sur les données tensorielles sans réduction de dimension alors que le noyau chordal opère sur les sous-espaces de dimensions réduites. Pour l'ensemble des méthodes, les hyperparamètres ont été choisis par recherche sur une grille de façon à minimiser l'erreur sur les données source et maximiser la précision sur les données cible.

4.1 Expérience sur données synthétiques

Le problème traité est un problème de régression. Un ensemble de données $\{\mathcal{X}^{(n)}\}_{n=1}^{N}$ est engendré à partir des tenseurs $\mathcal{X}^{(n)}_{\text{sans\_bruit}} \in \mathbb{R}^{5\times5\times5}$ suivant un modèle CP de rang 2 :

$$\mathcal{X}^{(n)}_{\text{sans\_bruit}} = \mathcal{C}^{(n)} \times_1 \mathbf{U}^{(n)}_1 \times_2 \mathbf{U}^{(n)}_2 \times_3 \mathbf{U}^{(n)}_3, \tag{5}$$

où les entrées de $\{\mathbf{U}^{(n)}_1, \mathbf{U}^{(n)}_2, \mathbf{U}^{(n)}_3\}$ de dimension 5×2, ainsi que les entrées de la diagonale de $\mathcal{C}^{(n)}$ sont tirées suivant une loi normale. Les échantillons de sortie sont définis par $y_n = f(g(\mathcal{X}^{(n)}_{\text{sans\_bruit}})) + \epsilon$, où $\epsilon \sim \mathcal{N}(0, \sigma^2_\epsilon)$ est un bruit additif. La transformation multilinéaire g est défini selon : $g(\mathcal{X}^{(n)}_{\text{sans\_bruit}}) = \tilde{\mathbf{x}}^{(n)} = [\tilde{x}^{(n)}_1 \;\; \tilde{x}^{(n)}_2]^T \in \mathbb{R}^2$, $\tilde{x}^{(n)}_i = \mathcal{X}^{(n)}_{\text{sans\_bruit}} \times_1 \mathbf{w}^{(1)T}_i \times_2 \mathbf{w}^{(2)T}_i \times_3 \mathbf{w}^{(3)T}_i, i = 1, 2$ avec $\{\mathbf{w}^{(m)}_i\}_{m=1}^{3} \sim \mathcal{N}(0, 1)$. La transformation non linéaire est pour sa part définie selon $f(\tilde{\mathbf{x}}^{(n)}) = (\tilde{x}^{(n)}_1)^2 + 3(\tilde{x}^{(n)}_2)^2$. Ces deux transformations successives garantissent que y_n n'est ni exactement linéaire ni multilinéaire par rapport à $\mathcal{X}^{(n)}_{\text{sans\_bruit}}$. Pour évaluer la robustesse des différents algorithmes, un bruit blanc gaussien d'une variance de 0.9^2 a été rajouté sur les échantillons $\mathcal{X}^{(n)}_{\text{sans\_bruit}}$ pour obtenir les données $\mathcal{X}^{(n)}$ et $\mathcal{X}'^{(n)}$.

Source : Les données $\{\mathcal{X}^{(i)}, y_i\}_{i=1}^{N_s}$ sont générées selon (5), où $\{\mathcal{C}^{(i)}, \mathbf{U}^{(i)}_1, \mathbf{U}^{(i)}_2, \mathbf{U}^{(i)}_3\}$ sont tirées suivant une loi normale $\mathcal{N}(0, 2)$. La variable de sortie est définie par $y_i = f(g(\mathcal{X}^{(i)})) + \epsilon$, avec $\epsilon \sim \mathcal{N}(0, \sigma^2_\epsilon)$ et $\sigma_\epsilon = 0.3$.

Cible : Les données $\{\mathcal{X}'^{(j)}\}_{j=1}^{N_t}$ sont également générées selon (5), mais avec des paramètres $\{\mathcal{C}'^{(j)}, \mathbf{U}'^{(j)}_1, \mathbf{U}'^{(j)}_2, \mathbf{U}'^{(j)}_3\}$ tirés suivant $\mathcal{N}(1.5, 0.3)$.

Les différentes méthodes testées ont été mises en œuvre pour $N_s = N_t \in \{10, 20, 50\}$. Pour évaluer les performances, nous avons calculé l'erreur quadratique moyenne $\mathrm{MRSE} = N_t^{-1} \sum_{j=1}^{N_t} \sqrt{\left(f^{\lambda}_{\hat{\mathbf{Z}},\mathbf{Z}'}(\mathcal{X}'^{(j)}) - y'_j\right)^2}$, où y'_j représente la vraie étiquette obtenue par $y'_j = f(g(\mathcal{X}'^{(j)}))$. Les résultats ont été moyennés sur 10 réalisations de $\{\mathcal{X}^{(i)}, y_i\}_{i=1}^{N_s}$ et $\{\mathcal{X}'^{(j)}\}_{j=1}^{N_t}$, et sont résumés dans la Fig. 1. On peut observer d'une part que les méthodes TUDA surpassent les méthodes de base y compris la méthode standard UDA. Pour les méthodes TUDA, le noyau chordal présente un léger avantage par rapport au noyau gaussien. En raison de la taille réduite des données, le taux de compression obtenu par la HOSVD reste relativement faible (3.67), ce qui limite l'amélioration des performances. Nous supposons cependant que ce gain en performance augmentera avec un taux de compression plus élevé.

4.2 Classification d'images

Cette expérience utilise deux ensembles de données extraits de MNIST et USPS pour évaluer la méthode proposée. MNIST

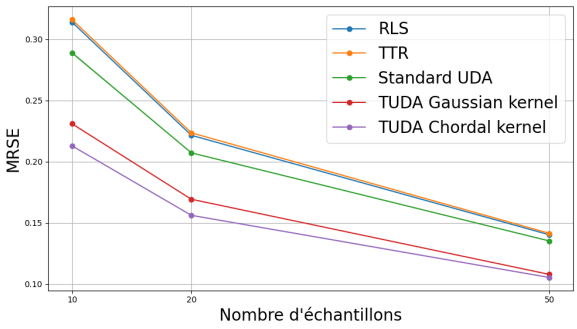

FIGURE 1 : MRSE pour les données synthétiques

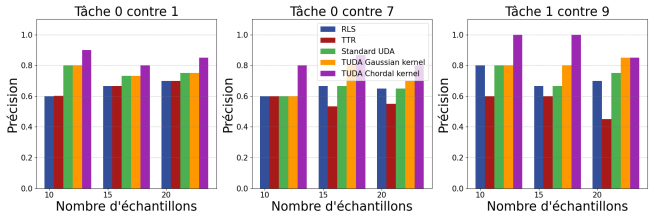

FIGURE 2 : Précision pour la tâche MNIST-to-USPS

contient des images en niveaux de gris, de chiffres manuscrits (0-9), et sert de référence standard pour les tâches de classification. En revanche, l'ensemble de données USPS (Kaggle) se compose d'images de chiffres RVB présentant des variations distinctes en termes de style, de résolution et de bruit. En raison de leurs distributions différentes, le transfert de connaissances entre MNIST et USPS pose un défi, ce qui fait de MNIST-to-USPS et USPS-to-MNIST de bonnes références pour évaluer les méthodes d'adaptation de domaine.

Dans cette expérience, nous utilisons des échantillons de taille $N_s = N_t \in \{10, 15, 20\}$ pour mettre en évidence les avantages de notre approche, en particulier dans le cas des échantillons de taille limitée. Les images en couleurs USPS sont représentées sous forme de tenseurs d'ordre trois, redimensionnés à $28 \times 28 \times 3$. Pour les images MNIST en niveaux de gris, nous les redimensionnons d'abord à 28×28 et concaténons ensuite trois images différentes avec la même étiquette pour former également un tenseur d'ordre trois. Le rang de l'HOSVD utilisé dans le noyau chordal a été choisi $(5, 5, 1)$, et correspond à la valeur qui maximise la performance sur l'ensemble de données d'entraînement. Pour chaque tâche de classification binaire, les performances sont évaluées en termes de précision, définie comme la proportion de toutes les classifications correctes, qu'elles soient positives ou négatives. Les résultats pour diverses paires de classes sont présentés sur les figures 2 et 3. On observe à nouveau que TUDA surpasse les méthodes de référence, ce qui confirme l'intérêt des approches tensorielles dans les méthodes UDA.

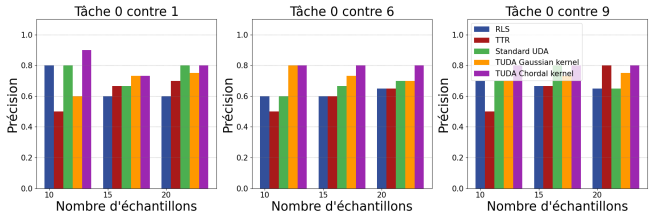

FIGURE 3 : Précision pour la tâche USPS-MNIST

5 Conclusion

Dans cette étude, nous avons proposé des algorithmes d'apprentissage dans le cadre de la pondération par l'importance pour l'UDA sous l'hypothèse de décalage des covariables, et introduit des techniques de régularisation pour l'estimation du R–ND pour les données tensorielles. A notre connaissance, il s'agit du premier travail appliquant conjointement la pondération par l'importance et l'estimation de la R–ND spécifiquement aux données tensorielles. En exploitant les noyaux tensoriels, et en particulier le noyau basé sur la distance chordale, notre approche capture efficacement les propriétés géométriques des espaces tensoriels tout en préservant leur structure intrinsèque. Les résultats expérimentaux démontrent les avantages de notre méthode, notamment sa robustesse au bruit, son efficacité sur de petits échantillons et sa capacité à s'adapter aux changements de distribution. Il est à noter que la méthode proposée se révèle particulièrement adaptée aux jeux de données d'entraînement de taille réduite. Les travaux futurs viseront à évaluer ses performances sur d'autres ensembles de données tensorielles réelles.

Remerciements

Ce travail a été soutenu par l'Agence Nationale de la Recherche (ANR) dans le cadre des projets ANR-23-CE94-0001 et ANR-23-CE23-0024.

Références

[1] Ali BRAYTEE, Mohamad NAJI et Paul J. KENNEDY : Unsupervised domain-adaptation-based tensor feature learning with structure preservation. *IEEE Transactions on Artificial Intelligence*, 3(3):370–380, 2022.

[2] Lieven DE LATHAUWER, Bart DE MOOR et Joos VANDEWALLE : A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[3] Elke R. GIZEWSKI, Lukas MAYER, Bernhard A. MOSER, Duc Hoan NGUYEN, Sergiy PEREVERZYEV, Sergei V. PEREVERZYEV, Natalia SHEPELEVA et Werner ZELLINGER : On a regularization of unsupervised domain adaptation in RKHS. *Applied and Computational Harmonic Analysis*, 57:201–227, 2022.

[4] Jiayuan HUANG, Alexander J. SMOLA, Arthur GRETTON et Karsten M. BORGWARDT : Correcting sample selection bias by unlabeled data. *Advances in neural information processing systems*, 19:601–608, 2006.

[5] Tamara G. KOLDA et Brett W. BADER : Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009.

[6] Xiaoshan LI, Da XU, Hua ZHOU et Lexin LI : Tucker tensor regression and neuroimaging analysis. *Statistics in Biosciences*, 10:520–545, 2018.

[7] Hao LU, Lei ZHANG, Zhiguo CAO, Wei WEI, Ke XIAN, Chunhua SHEN et Anton van den HENGEL : When unsupervised domain adaptation meets tensor representations. *In ICCV*, pages 599–608, 2017.

[8] Duc Hoan NGUYEN, Werner ZELLINGER et Sergei PEREVERZYEV : On regularized Radon-Nikodym differentiation. *Journal of Machine Learning Research*, 25(266):1–24, 2024.

[9] Bernhard SCHÖLKOPF, Ralf HERBRICH et Alex J. SMOLA : A generalized representer theorem. *In* David HELMBOLD et Bob WILLIAMSON, éditeurs : *Computational Learning Theory*, pages 416–426, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[10] Marco SIGNORETTO, Lieven DE LATHAUWER et Johan A.K. SUYKENS : A kernel-based framework to tensorial data analysis. *Neural Networks*, 24(8):861–874, 2011. Artificial Neural Networks : Selected Papers from ICANN 2010.

[11] Qibin ZHAO, Guoxu ZHOU, Tulay ADALI, Liqing ZHANG et Andrzej CICHOCKI : Kernelization of tensor-based models for multiway data analysis : Processing of multidimensional structured data. *IEEE Signal Processing Magazine*, 30(4):137–148, 2013.