Knowledge-Based Systems 210 (2020) 106447

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

# Domain adaptation for regression under Beer–Lambert's law

Ramin Nikzad-Langerodi [a], Werner Zellinger [a,*], Susanne Saminger-Platz [b], Bernhard A. Moser [a]

[a] Software Competence Center Hagenberg GmbH (SCCH), Softwarepark 21, 4232 Hagenberg im Mühlkreis, Austria
[b] Department of Knowledge-Based Mathematical Systems, Johannes Kepler University Linz (JKU), Altenbergerstrasse 69 A-4040 Linz, Austria

ARTICLE INFO

*Article history:*
Received 15 January 2020
Received in revised form 7 September 2020
Accepted 8 September 2020
Available online 1 October 2020

*Keywords:*
Transfer learning
Domain adaptation
Moment alignment
Chemometrics
Calibration model adaptation
Partial least squares

ABSTRACT

We consider the problem of unsupervised domain adaptation (DA) in regression under the assumption of linear hypotheses (e.g. Beer–Lambert's law) – a task recurrently encountered in analytical chemistry. Following the ideas from the non-linear iterative partial least squares (NIPALS) method, we propose a novel algorithm that identifies a low-dimensional subspace aiming at the following two objectives: (i) the projections of the source domain samples are informative w.r.t. the output variable and (ii) the projected domain-specific input samples have a small covariance difference. In particular, the latent variable vectors that span this subspace are derived in closed-form by solving a constrained optimization problem for each subspace dimension adding flexibility for balancing the two objectives. We demonstrate the superiority of our approach over several state-of-the-art (SoA) methods on different DA scenarios involving unsupervised adaptation of multivariate calibration models between different process lines in Melamine production and equality to SoA on two well-known benchmark datasets from analytical chemistry involving (unsupervised) model adaptation between different spectrometers. The former dataset is published with this work[1]



## 1. Introduction

Unsupervised domain adaptation (DA) aims at learning a model on a labeled *source* sample and an unlabeled *target* sample that follow different distributions with the goal of achieving a high performance on the unlabeled sample's distribution [1–3]. Recently, DA techniques attracted considerable attention in analytical chemistry since adaptation of (multivariate) calibration models, model maintenance and calibration transfer between similar analytical devices are recurring tasks that still lack satisfactory "off-the-shelf" solutions [4–8]. Yet, the success of DA techniques on the type of data typically derived from chemical measurement systems has been limited indicating that the assumptions of the underlying models do not comply with the properties of the data. Primarily, most of the DA techniques developed over the past decade involve non-linear hypotheses, which is the natural choice for applications in e.g. computer vision, text mining or natural language processing. However, the data generating process underlying most setups in analytical chemistry is governed by *Beer–Lambert's* law, which describes the relationship between absorbance of electromagnetic radiation and analyte concentration [9], i.e.

$$A = -\log\left(\frac{I_0}{I}\right) = \varepsilon \cdot \zeta \cdot d. \tag{1}$$

$A$ in Eq. (1) denotes absorbance, $\varepsilon$ is the characteristic (substance specific) absorptivity of the analyte, $\zeta$ the concentration in solution and $d$ the optical path length. $I_0$ is the raw intensity for $\zeta = 0$ (i.e. the background signal) and $I$ is the attenuated signal (Fig. 1A). Although Beer–Lambert's law does not strictly hold in practice due to e.g. light scattering, (non-linear) interactions between different analytes (i.e. matrix effects), sample inhomogeneity *etc.*, the linear dependence of the measured signal on concentration still holds surprisingly well for a wide array of analytical techniques and sample types justifying the use of linear hypotheses when modeling concentration given absorbance (i.e. calibration) [10]. Further reasons for the limited success of current DA technique on Beer–Lambert type data and applications in analytical chemistry are that (i) sample size is often small (i.e. $N < 100$) and (ii) the number of predictors usually large in typical calibration settings and thus (iii) estimation of the underlying distributions without further assumptions on the data in general difficult. The typical assumptions are that the data has low rank and that it is approximately normal distributed. The former follows from the number of chemically distinct molecular species (i.e. the chemical rank) within a samples, which is usually small in relation to the number of predictors. Normal distribution,

* Corresponding author.
*E-mail address:* werner.zellinger@scch.at (W. Zellinger).

[1] https://github.com/RNL1/Melamine-Dataset

https://doi.org/10.1016/j.knosys.2020.106447
0950-7051/

on the other hand, is a reasonable assumption when the data lies on a low-dimensional, latent space since the corresponding linear combinations of the predictors converge to normal distributed random variables as the number of predictors grows [11].

To account for all the aforementioned peculiarities, in what follows we propose a new algorithm for multivariate regression that combines recent ideas from unsupervised domain adaptation with an old technique that strongly influenced the field of chemometrics: The non-linear iterative partial least squares (NIPALS) algorithm [12]. Along these lines, our algorithm aims at mapping the input data on a low-dimensional subspace explaining a high amount of variation in the output variable and at the same time a small difference between first and second order statistics of the domain-specific input samples (Fig. 1). The directions of this subspace are computed consecutively as closed-form solution of a convex optimization problem. Each iteration of our algorithm is followed by matrix deflation yielding orthogonal, domain-invariant latent variables with high predictive power w.r.t. the output variable in the source domain. We thus coin our method Domain-Invariant Iterative Partial Least Squares (DIPALS).[2]

Our main contributions are as follows:

- We propose a new deterministic learning algorithm for unsupervised domain adaptation of multivariate linear regression models.
- We provide a rigorous mathematical analysis of our approach including (a) a new upper bound on the target error which is minimized by our algorithm, (b) a proof of the uniqueness and a characterization of the projection vector computed in each iteration, and, (c) a heuristic for interpretable parameter setting.
- We provide a new dataset of real-world data for unsupervised domain adaptation of regression.[3]
- Our algorithm compares favourably to state-of-the-art methods on three benchmark datasets for analytical chemistry.

The rest of the paper is organized as follows: Section 2 gives a brief overview of previous work related to DA. Section 3 states the problem of unsupervised DA in regression. In Section 4 we develop the theoretical basis of our algorithm which is proposed in Section 5. Section 6 compares our algorithm with different DA techniques on three benchmark datasets from analytical chemistry and Section 7 concludes the paper.

## 2. Related work

Several strategies have been proposed in the past to overcome the problems arising when training and test data are sampled from different distributions: Instance weighting approaches reweigh the training instances to increase the similarity between training and test distributions [14–16]. Blitzer et al. suggested (heuristic) selection of so-called domain-invariant pivot variables in order to narrow the domain gap [17]. A different strategy to reduce the difference between domains involve minimization of some measure of domain discrepancy in latent spaces: In [18] the Maximum Mean Discrepancy [19] between the empirical domain-specific distributions is minimized subject to an orthogonality constraint on the latent variables. Scatter Component Analysis employs the Maximum Mean Discrepancy to minimize domain scatter while concomitantly maximizing total and between-class scatter [20]. Conceptually, these methods perform dimensionality reduction in a feature space (induced by the corresponding kernel) and thus introduce non-linearity, which might not be appropriate if the true relationship between input and output variables is approximately linear. Combination of pivot variable selection and distribution alignment has been proposed in [21, 22]. DA techniques that align distributions in linear subspaces have been proposed in [23–25]. However, these methods involve a numerical solution of non-convex optimization problems and eventually converge to sub-optimal solutions. The same is true for recent approaches aiming at (unsupervised) correction of changes in the conditional distribution [26,27]. Non-linear hypotheses and/or subspaces in combination with non-convex objectives are treated e.g. in [28–31].

## 3. Problem formulation

We follow the basic formal model of domain adaptation defined in [2], where a *domain* is defined as a pair $\langle P, l\rangle$ consisting of a distribution $P$ (in our case on $\mathbb{R}^K$) and a target function (in our case $l : \mathbb{R}^K \to [0, 1]$). We consider a *source* domain $\langle P, l\rangle$ and a *target* domain $\langle Q, l\rangle$ sharing the same labeling function, i.e. we follow the covariate-shift assumption [32].

Given a source sample $\mathbf{X}_S \in \mathbb{R}^{N_S \times K}$ (with rows $\mathbf{x}^T$ corresponding to the input signals of individual samples) drawn from $P$ with corresponding continuous labels $\mathbf{y} = l(\mathbf{X}_S) \in [0, 1]^{N_S}$ and an unlabeled target sample $\mathbf{X}_T \in \mathbb{R}^{N_T \times K}$ drawn from $Q$, the goal of unsupervised domain adaptation is to find a function $h : \mathbb{R}^K \to [0, 1]$ with a small target error

$$\mathrm{E}_Q[|h - l|] = \int_{\mathbb{R}^K} |h - l| \, dQ. \tag{2}$$

## 4. Motivation

This section motivates our algorithm by means of a new learning bound under three typical characteristics of data derived from chemical measurement systems: Linear dependency between input and output (governed by Beer–Lambert's law), multicollinearity of input signals, and, approximately normally distributed data.

In the following, let us consider two domains $\langle P, l\rangle$ and $\langle Q, l\rangle$ and a function $h = f \circ g$ where $g : \mathbb{R}^K \to \mathbb{R}^L \in \mathcal{G}$ and $f : \mathbb{R}^L \to \mathbb{R} \in \mathcal{F}$ are two functions from appropriate classes $\mathcal{G}$ and $\mathcal{F}$. Denote by $\tilde{P} := P \circ g^{-1}$ and $\tilde{Q} := Q \circ g^{-1}$ the latent distributions w. r. t. $g$ (pushforward measures) and by $D(\mathcal{N}_{\tilde{P}} \parallel \mathcal{N}_{\tilde{Q}})$ the Kullback–Leibler divergence (KL-divergence) between the two Normal distributions $\mathcal{N}_{\tilde{P}}$ and $\mathcal{N}_{\tilde{Q}}$ with mean and covariance corresponding to $\tilde{P}$ and $\tilde{Q}$, respectively. Based on these definitions, we may state the following lemma.

**Lemma 1.** *Consider two domains $\langle P, l\rangle$, $\langle Q, l\rangle$ and the function $h = f \circ g$ which induces the latent distributions $\tilde{P}$, $\tilde{Q}$ and the Normal distributions $\mathcal{N}_{\tilde{P}}$, $\mathcal{N}_{\tilde{Q}}$ as defined above. Then the following holds:*

$$\mathrm{E}_Q[|h - l|] \leq \mathrm{E}_P[|h - l|] + \sqrt{2D(\mathcal{N}_{\tilde{P}} \parallel \mathcal{N}_{\tilde{Q}})} + \lambda(g) + \sqrt{8\epsilon} \tag{3}$$

*where*

$$\epsilon := \max\left\{D(\mathcal{N}_{\tilde{P}} \parallel \tilde{P}), D(\mathcal{N}_{\tilde{Q}} \parallel \tilde{Q})\right\} \tag{4}$$

*and*

$$\lambda(g) := \inf_{f \in \mathcal{F}} (\mathrm{E}_P[|f \circ g - l|] + \mathrm{E}_Q[|f \circ g - l|]). \tag{5}$$

**Proof.** Applying Thm.1 in [2] to the two domains $\langle \tilde{P}, l_P\rangle$ and $\langle \tilde{Q}, l_Q\rangle$ with (stochastic) labeling functions $l_P, l_Q : \mathbb{R}^N \to [0, 1]$, which are induced by the latent mapping $g$ and defined by $l_P(\mathbf{t}) := \mathrm{E}_P[l(\mathbf{x}) \mid g(\mathbf{x}) = \mathbf{t}]$ according to [33], yields

$$\mathrm{E}_{\tilde{Q}}[\left|f - l_Q\right|] \leq \mathrm{E}_{\tilde{P}}[|f - l_P|] + 2d_{\mathrm{TV}}(\tilde{P}, \tilde{Q})$$

[2] This document replaces preliminary work previously published in [13].

[3] https://github.com/RNL1/Melamine-Dataset

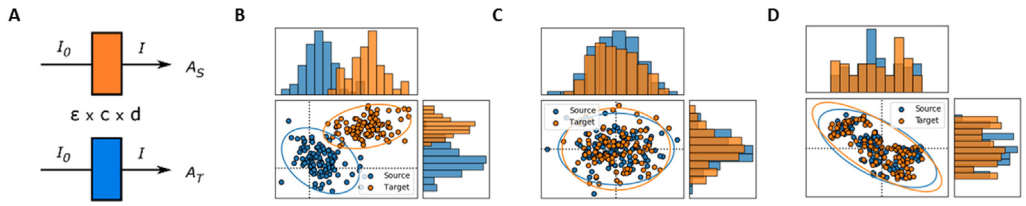


**Fig. 1.** Graphical abstract. (A) Beer–Lambert's law. (B) and (C) show projections of source and target domain data on the first two latent variables of a partial least squares (PLS) and a DIPALS model, respectively. (D) t-SNE embedding of C from higher-dimensional latent variable space.

$$+ \inf_{f:\mathbb{R}^L\to\mathbb{R}} (\mathrm{E}_{\tilde{P}}[|f - l_P|] + \mathrm{E}_{\tilde{Q}}[|f - l_Q|])$$

where $d_{\mathrm{TV}}$ refers to the Total-Variation distance. From the "change of variables" Thm. 4.1.11 in [34] we obtain

$$\mathrm{E}_{\tilde{P}}[|f - l_P|] = \int |f - l_P|\, d(P \circ g^{-1}) = \int |f - l_P| \circ g\, dP$$
$$= \mathrm{E}_P[|f \circ g - l_P|]$$

which, together with the application of the Triangle inequality for $d_{\mathrm{TV}}$, implies that

$$\mathrm{E}_Q[|h - l|] \leq \mathrm{E}_P[|h - l|] + \lambda(g) + 2d_{\mathrm{TV}}(\mathcal{N}_{\tilde{P}}, \mathcal{N}_{\tilde{Q}})$$
$$+ 2d_{\mathrm{TV}}(\tilde{P}, \mathcal{N}_{\tilde{P}}) + 2d_{\mathrm{TV}}(\tilde{Q}, \mathcal{N}_{\tilde{Q}}).$$

Eq. (3) then follows from Pinsker's inequality and the definition of $\epsilon$. □

Lemma 1 shows that the error in the target domain can be bounded in terms of the error in the source domain, the KL-divergence between Normal approximations of the latent distributions, a corresponding approximation error $\epsilon$ and the domain adaptation error $\lambda(g)$. For sample-based upper bounds on the target error in terms of moments, we refer to [35], and, for sample-based upper bounds on the distance $D(\mathcal{N}_P \parallel \mathcal{N}_Q)$ we refer to [36].

Lemma 1 suggests a small target error if the terms on the right-hand side of Eq. (3) are small. In the following, we motivate different algorithmic properties under which (in combination with the three observations stated in Section 1) each of these terms can be expected to be small.

**Domain Adaptation Error $\lambda(g)$:** Beer Lambert's law states a linear relationship between output variables and inputs. Therefore, we assume a target function $l : \mathbb{R}^K \to [0, 1]$ that is well approximable by a linear function, i.e. $l(\mathbf{x}) \approx \mathbf{x}^{\mathrm{T}}\mathbf{d}$ for some $\mathbf{d} \in \mathbb{R}^K$. For such a target function $l$ and each linear function $g : \mathbb{R}^K \to \mathbb{R}^L$ with $g(\mathbf{x}) = (\mathbf{x}^{\mathrm{T}}\mathbf{A})^{\mathrm{T}}$ and orthogonal matrix $\mathbf{A} \in \mathbb{R}^{K\times L}$, it always exits a linear function $f \in \mathcal{F}$, e.g. $f(\mathbf{x}) := x^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\mathbf{d}$, such that $l \approx f \circ g$ and $\lambda(g) \approx 0$. We therefore aim at finding a function $h = f \circ g$ with orthogonal projection $g : \mathbb{R}^K \to \mathbb{R}^L$ and linear function $f : \mathbb{R}^L \to \mathbb{R}$.

**Source Error $\mathrm{E}_P[|h - l|]$:** To overcome numerical instabilities caused by the observed high multicollinearity of the input data, the NIPALS algorithm has been proposed to find a linear latent variable model $f \circ g$ as defined above with a small source error. This algorithm serves as a starting point for our algorithm.

**Approximation Error $\epsilon$:** One implication of the assumption of approximately normally distributed input distributions $P$ and $Q$ is that the application of the linear transformation $g$ leads to latent distributions $\tilde{P}$ and $\tilde{Q}$ that are well approximable by Normal distributions. It is therefore reasonable to assume a small $\epsilon$ in Lemma 1. It is interesting to observe, that the term $\epsilon$ can be interpreted as an upper bound on the information stored in the distributions $P$ and $Q$ in addition to the first two moments [37].

**Distribution Divergence $D(\mathcal{N}_{\tilde{P}} \parallel \mathcal{N}_{\tilde{Q}})$:** It is well known that the convergence $D(\mathcal{N}_{P_n} \parallel \mathcal{N}_{P_\infty}) \to 0$ for $n \to \infty$ of some zero mean centered distributions $P_n$, $n \in \mathbb{N}$ and $P_\infty$ is implied by the convergence of the respective covariances $\Sigma_n \to \Sigma_\infty$. This motivates us to learn a transformation $g$ such that the transformations $\tilde{P}$ and $\tilde{Q}$ of the distributions $P$ and $Q$ show zero means and similar covariance matrices. It is interesting to note that, in the case of Normal distributions, the convergence in mean and covariance is equivalent to the convergence in most other probability metrics, see e.g. [35,38].

## 5. DIPALS algorithm

As described in Section 3, given two matrices $\mathbf{X}_S \in \mathbb{R}^{N_S\times K}$, $\mathbf{X}_T \in \mathbb{R}^{N_T\times K}$ with rows $\mathbf{x}^{\mathrm{T}}$ representing input signals $\mathbf{x}$ and a vector $\mathbf{y} \in \mathbb{R}^{N_S}$ of corresponding outputs, we aim at computing a linear function $f \circ g : \mathbb{R}^K \to \mathbb{R}$ with a small error on the distribution of the target sample $\mathbf{X}_T$.

As motivated in Section 4, we aim at computing linear functions $f : \mathbb{R}^L \to \mathbb{R}$ with $f(\mathbf{t}) = \mathbf{t}^{\mathrm{T}}\mathbf{c}$ for $\mathbf{c} \in \mathbb{R}^L$ and $g : \mathbb{R}^K \to \mathbb{R}^L$ with $g(\mathbf{x}) = (\mathbf{x}^{\mathrm{T}}\mathbf{A})^{\mathrm{T}}$ for orthogonal $\mathbf{A} \in \mathbb{R}^{K\times L}$ such that the source error is minimized and the sample covariance matrices of the latent samples $\mathbf{X}_S\mathbf{A}$ and $\mathbf{X}_T\mathbf{A}$ are similar. To handle collinearity in the inputs, we rely on a regularized version of the NIPALS algorithm.

### 5.1. Step-by-step description

The standard implementation of the NIPALS algorithm [12] contains four basic steps: Initialization, Projection, Regression and Deflation. In the following, these steps are adapted for unsupervised domain adaptation.

**Step 0 (Initialization):** The first step of our algorithm consists of zero mean centering of the inputs and outputs such that $\mathrm{E}[\mathbf{X}_S] = \mathrm{E}[\mathbf{X}_T] = \mathrm{E}[\mathbf{y}] = 0$ where $\mathrm{E}[\mathbf{X}]$ refers to the column-wise empirical mean of the matrix $\mathbf{X}$. Then, we follow the basic ideas of the NIPALS algorithm by iterating over the following steps to compute one direction of the latent mapping (and a corresponding regression coefficient):

**Step 1 (Domain-Invariant Projection):** The following objective function is considered:

$$\min_{\mathbf{w}^{\mathrm{T}}\mathbf{w}=1} \|\mathbf{X}_S - \mathbf{y}\mathbf{w}^{\mathrm{T}}\|_{\mathrm{F}}^2 + \gamma\mathbf{w}^{\mathrm{T}}\boldsymbol{\Lambda}\mathbf{w} \tag{6}$$

where $\|.\|_{\mathrm{F}}$ refers to the Frobenius norm, $\gamma$ is the domain-regularization parameter and

$$\boldsymbol{\Lambda} := \mathbf{K}\mathrm{diag}(|\lambda_1|, \ldots, |\lambda_K|)\mathbf{K}^{\mathrm{T}} \tag{7}$$

is the matrix obtained by taking the absolute value of all eigenvalues $\lambda_1, \ldots, \lambda_K$ in the eigendecomposition

$$\mathbf{K}\mathrm{diag}(\lambda_1, \ldots, \lambda_K)\mathbf{K}^{\mathrm{T}} = \frac{1}{N_S - 1}\mathbf{X}_S^{\mathrm{T}}\mathbf{X}_S - \frac{1}{N_T - 1}\mathbf{X}_T^{\mathrm{T}}\mathbf{X}_T \tag{8}$$

with corresponding eigenvector matrix $\mathbf{K}$ of the difference of the domain-specific covariance matrices. The first term in Eq. (6) corresponds to the ordinary NIPALS objective and its minimum is

**Algorithm 1** DIPALS

**Input:** Source sample $\mathbf{X}_S \in \mathbb{R}^{N_S \times K}$, labels $\mathbf{y} \in \mathbb{R}^{N_S}$, target sample $\mathbf{X}_T \in \mathbb{R}^{N_T \times K}$, number of latent variables $L \in \mathbb{N}$, domain-regularization parameter $\gamma \in \mathbb{R}$
**Output:** Regression vector $\mathbf{b} \in \mathbb{R}^K$ of the function $h(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{b}$

**Step 0 (Initialization):** $\mathbf{P} := [\,]$, $\mathbf{W} := [\,]$, $\mathbf{c} := [\,]$, $\mathbf{y} := \mathbf{y} - \mathrm{E}[\mathbf{y}]$, $\mathbf{X}_S := \mathbf{X}_S - \mathrm{E}[\mathbf{X}_S]$, $\mathbf{X}_T := \mathbf{X}_T - \mathrm{E}[\mathbf{X}_T]$
**for** $i = 1$ **to** $L$ **do**
  **Step 1 (Domain-Invariant Projection):**
  Compute eigenvalues $\lambda_1, \ldots, \lambda_K$ and eigenvector matrix $\mathbf{K}$ of $\frac{1}{N_S-1}\mathbf{X}_S^{\mathrm{T}}\mathbf{X}_S - \frac{1}{N_T-1}\mathbf{X}_T^{\mathrm{T}}\mathbf{X}_T$
  $\boldsymbol{\Lambda} := \mathbf{K}\mathrm{diag}(|\lambda_1|, \ldots, |\lambda_K|)\mathbf{K}^{\mathrm{T}}$
  $\mathbf{w}^{\mathrm{T}} := \frac{\mathbf{y}^{\mathrm{T}}\mathbf{X}_S}{\mathbf{y}^{\mathrm{T}}\mathbf{y}}\left(\mathbf{I} + \frac{\gamma}{\mathbf{y}^{\mathrm{T}}\mathbf{y}}\boldsymbol{\Lambda}\right)^{-1}$
  $\mathbf{w} := \mathbf{w}/\|\mathbf{w}\|$
  **Step 2 (Regression):**
  $\mathbf{t}_S := \mathbf{X}_S\mathbf{w}$, $\mathbf{t}_T := \mathbf{X}_T\mathbf{w}$
  $\tilde{c} := (\mathbf{t}_S{}^{\mathrm{T}}\mathbf{t}_S)^{-1}\mathbf{t}_S{}^{\mathrm{T}}\mathbf{y}$
  **Step 3 (Deflation):**
  $\mathbf{p}^{\mathrm{T}}{}_S := (\mathbf{t}_S{}^{\mathrm{T}}\mathbf{t}_S)^{-1}\mathbf{t}_S{}^{\mathrm{T}}\mathbf{X}_S$, $\mathbf{p}^{\mathrm{T}}{}_T := (\mathbf{t}_T{}^{\mathrm{T}}\mathbf{t}_T)^{-1}\mathbf{t}_T{}^{\mathrm{T}}\mathbf{X}_T$
  $\mathbf{X}_S := \mathbf{X}_S - \mathbf{t}_S\mathbf{p}_S{}^{\mathrm{T}}$, $\mathbf{X}_T := \mathbf{X}_T - \mathbf{t}_T\mathbf{p}_T{}^{\mathrm{T}}$
  $\mathbf{y} := \mathbf{y} - \tilde{c}\mathbf{t}_S$
  $\mathbf{P} := [\mathbf{P}, \mathbf{p}_S]$, $\mathbf{W} := [\mathbf{W}, \mathbf{w}]$, $\mathbf{c} := [\mathbf{c}, \tilde{c}]$
**end for**
$\mathbf{b} = \mathbf{W}(\mathbf{P}^{\mathrm{T}}\mathbf{W})^{-1}\mathbf{c}$

obtained by the direction $\mathbf{w}$ where $\mathbf{X}_S$ has maximum sample covariance with $\mathbf{y}$. The second term in Eq. (6) is our contribution and represents an upper bound on the absolute difference between the source sample variance and the target sample variance in the direction $\mathbf{w}$ (see Section 5.3). The (unique) solution of Eq. (6) is achieved by the vector

$$\mathbf{w}^{\mathrm{T}} = \frac{\mathbf{y}^{\mathrm{T}}\mathbf{X}_S}{\mathbf{y}^{\mathrm{T}}\mathbf{y}}\left(\mathbf{I} + \frac{\gamma}{\mathbf{y}^{\mathrm{T}}\mathbf{y}}\boldsymbol{\Lambda}\right)^{-1} \tag{9}$$

divided by its length $\mathbf{w}^{\mathrm{T}}\mathbf{w}$ (see Section 5.3). The coordinates $\mathbf{t}_S$ and $\mathbf{t}_T$ of the (domain-invariant) projections corresponding to the direction $\mathbf{w}$ can be computed by

$$\mathbf{t}_S = \mathbf{X}_S\mathbf{w} \quad \text{and} \quad \mathbf{t}_T = \mathbf{X}_T\mathbf{w}. \tag{10}$$

**Step 2 (Regression):** Ordinary least squares regression of $\mathbf{y}$ on $\mathbf{t}_S$ yields

$$c = (\mathbf{t}_S^{\mathrm{T}}\mathbf{t}_S)^{-1}\mathbf{t}_S^{\mathrm{T}}\mathbf{y}. \tag{11}$$

**Step 3 (Deflation):** Following the Gram–Schmidt process, our algorithm removes the variation in $\mathbf{X}_S$ explained by the current latent variable by subtracting the projection of $\mathbf{X}_S$ along $\mathbf{t}_S$, i.e. the following update is performed

$$\mathbf{X}_S := \mathbf{X}_S - \mathbf{t}_S(\mathbf{t}_S^{\mathrm{T}}\mathbf{t}_S)^{-1}\mathbf{t}_S^{\mathrm{T}}\mathbf{X}_S. \tag{12}$$

The matrix $\mathbf{X}_T$ is updated analogously by means of $\mathbf{t}_T$.

After each iteration, the coordinates of the vectors are properly aggregated to obtain the final regression vector $\mathbf{b}$ such that $h(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{b}$, see Algorithm 1 and [39] for detailed derivations of these standard PLS steps. In particular, the projection matrix $\mathbf{A}$ such that $g(\mathbf{x}) = (\mathbf{x}^{\mathrm{T}}\mathbf{A})^{\mathrm{T}}$ can be computed by the relationship [40]:

$$\mathbf{A} = \mathbf{W}(\mathbf{P}^{\mathrm{T}}\mathbf{W})^{-1}, \tag{13}$$

where the projection matrix $\mathbf{P}$ and the weight matrix $\mathbf{W}$ are computed as described in Algorithm 1.

### 5.2. Regularization parameter heuristic

Consider the matrix $\boldsymbol{\Lambda}$ from Eq. (7) and the vector $\mathbf{w}_0$ corresponding to the unconstrained ($\gamma = 0$) NIPALS solution. We propose to iteration-wise fix the value of gamma as

$$\gamma := \frac{\|\mathbf{X}_S - \mathbf{y}\mathbf{w}_0^{\mathrm{T}}\|_{\mathrm{F}}^2}{\mathbf{w}_0^{\mathrm{T}}\boldsymbol{\Lambda}\mathbf{w}_0}. \tag{14}$$

This setting leads to equal weighting of the terms in the objective Eq. (6) in the direction $\mathbf{w}_0$.

### 5.3. Discussion

*Multicollinearity.* The optimum of the first term in the objective function Eq. (6) is achieved by the direction $\mathbf{w}_0$ where the sample covariance $\frac{1}{N_S-1}\mathbf{w}_0^{\mathrm{T}}\mathbf{X}_S^{\mathrm{T}}\mathbf{y}$ between $\mathbf{X}_S$ and the output vector $\mathbf{y}$ is maximal. As a result, the NIPALS algorithm well handles multicollinearity of the input sample (compare also [39]).

*Uniqueness of solution.* The value of our regularizer $\mathbf{w}^{\mathrm{T}}\boldsymbol{\Lambda}\mathbf{w}$ (for $\mathbf{w}^{\mathrm{T}}\mathbf{w} = 1$) is nothing but the value of the Rayleigh quotient of the positive semi-definite matrix $\boldsymbol{\Lambda}$. It is therefore convex and its summation preserves the convexity of the original NIPALS objective, i.e. the first term in Eq. (6). As a result, the unique solution of the objective function can be obtained as the root of its derivative and has the form of Eq. (9).

*Interpretation of proposed regularizer.* Our regularizer is an upper bound on the absolute difference

$$\left|\frac{1}{N_S-1}\mathbf{w}^{\mathrm{T}}\mathbf{X}_S^{\mathrm{T}}\mathbf{X}_S\mathbf{w} - \frac{1}{N_T-1}\mathbf{w}^{\mathrm{T}}\mathbf{X}_T^{\mathrm{T}}\mathbf{X}_T\mathbf{w}\right| \tag{15}$$

between the domain-specific sample variances in the direction $\mathbf{w}$. To see this, consider the eigenvector matrix $\mathbf{K}$ and the eigenvalues $\lambda_1, \ldots, \lambda_K$ as in Eq. (8). Then, by letting $\mathbf{v} := (v_1, \ldots, v_K) := \mathbf{K}^{\mathrm{T}}\mathbf{w}$, Eq. (15) is equal to

$$\begin{aligned}
&|\mathbf{w}^{\mathrm{T}}\mathbf{K}\mathrm{diag}(\lambda_1, \ldots, \lambda_K)\mathbf{K}^{\mathrm{T}}\mathbf{w}| \\
&\quad = |v_1^2\lambda_1 + \cdots + v_K^2\lambda_K| \\
&\quad \le |v_1^2\lambda_1| + \cdots + |v_K^2\lambda_K| \\
&\quad = v_1^2|\lambda_1| + \cdots + v_K^2|\lambda_K| \\
&\quad = \mathbf{v}^{\mathrm{T}}\mathrm{diag}(|\lambda_1|, \ldots, |\lambda_K|)\mathbf{v} \\
&\quad = \mathbf{w}^{\mathrm{T}}\boldsymbol{\Lambda}\mathbf{w}.
\end{aligned}$$

This shows, that the proposed regularizer corresponds to an upper bound on the difference between the source sample variance and the target sample variance in the direction $\mathbf{w}$. It can therefore be interpreted as biasing the NIPALS objective towards directions with a low variance difference between the domains in the projection space.

*Interpretation of proposed heuristic.* The derivations above allow to interpret the regularization strength $\gamma$ as trade-off between attaining high input–output covariance in the source domain and low variance difference between the domains. This leads to intuitive heuristics for default values of $\gamma$ as proposed in Section 5.2. With the value of $\gamma$ as in Eq. (14) we articulate our preference of treating regression and domain alignment as equal important in a range around the optimal NIPALS solution.

*Computational complexity.* The computational complexity of our DIPALS algorithm is $\mathcal{O}(LK^3)$ where $L$ is the number of latent variables and $K$ is the input dimension. The most time consuming step is the eigendecomposition in Eq. (7) which can be implemented in $\mathcal{O}(K^3)$ time-steps, see e.g. [41].

## 6. Experiments

In this section we compare our method (DIPALS) with several state-of-the-art DA techniques on three benchmark data sets from analytical chemistry. We consider: Correlation alignment (CORAL) for the fact that — similar to our method — it can handle domain shifts in terms of covariance differences [42]; Transfer component analysis (TCA) for the capability of addressing both, covariate shift and multicollinearity [18]; Joint distribution optimal transportation (JDOT), which in contrast to most other DA techniques proposed so far can handle shifts not only in the marginal but also in the conditional distributions [27]; Domain-invariant PLS (di-PLS), which similar to the here proposed method aims at modeling the response in terms of domain-invariant LVs but solves a non-convex optimization problem to do so [24] and ordinary partial least squares (PLS) regression on the source and applied in the target domain, which is used as baseline in all experiments.

### 6.1. Hyperparameter selection

The parameter $\gamma$ of our method (DIPALS) was set for each latent variable using the parameter heuristic described in Sub Section 5.2. Alternatively, we employ 10-fold cross-validation on the source data for $\gamma$ in the range $\{0.1, 1, \ldots, 10^{10}\}$ and set $\gamma_i$ to the value exhibiting the lowest cross-validation error on the source task for $i = \{1, \ldots, L\}$ latent variables. Note that in both cases, no labels from the target domain were used for finding the optimal parameter(s). The number of latent variables was chosen based on 10-fold cross-validation on the source task employing the one-standard-error rule as described elsewhere [43]. CORAL was applied to the (training set) projections of the baseline (PLS) model followed by ordinary least squares regression. For the other methods we report the best parameter (combination) yielding the lowest mean squared error on the test set (i.e. supervised tuning): For (linear) TCA we explore $\{1, \ldots, 30\}$ latent variables and $\mu \in \{10^{-10}, 10^{-9}, \ldots, 1\}$. For JDOT we either set $\alpha = 1/\max_{i,j} \mathrm{d}(\mathbf{x}_i^S, \mathbf{x}_j^T)$ or vary $\alpha \in \{10^{-10}, 10^{-9}, \ldots, 1\}$ (whichever yielded better results in the target domain) as described previously [27]. In addition, we vary the (linear) kernel ridge regression parameter $\lambda$ in the range $\{10^{-10}, 10^{-9}, \ldots, 1\}$. The regularization parameter $\lambda$ for di-PLS was tuned in the range $10^{-9}, 10^{-8}, \ldots, 10^{9}$ as described in [24].

### 6.2. Datasets

*Melamine.* The Melamine dataset originates from a batch condensation process and consists of near infrared (NIR) absorbance spectra of 4 different Melamine resins recorded at different degrees of polymerization. The analytical goal is to predict the turbidity point (expressed in °C), which is related to the degree of polymerization, from the corresponding NIR spectra.[4] We herein report the results of inductive DA between different Melamine resins using the entire set of input features, i.e. both wavenumber ranges from 5546–6254 $\mathrm{cm}^{-1}$ and 6596–6975 $\mathrm{cm}^{-1}$. To this end, we randomly split the target domain data for each scenario into an unlabeled training set (60%), which is used along with the source domain data for training, and a test set (40%) used to estimate the generalization error in the target domain. The test set was re-centered to the training set and all experiments were repeated 10 times.

*Corn.* The Corn dataset is a well established dataset used to benchmark instrument standardization algorithms in analytical chemistry and comprises NIR spectra from a set of 80 corn samples measured on 3 similar spectrometers (m5,mp5 and mp6).[5] The analytical goal is to predict oil, water, starch and protein contents from the corresponding spectra. In the present contribution we consider inductive DA between the different instruments by defining the source domain as the first 40 samples and the target domain as the following 40 samples of the dataset. Given the 4 output variables, this translates into 24 DA scenarios. We proceed similar to the experiments on the Melamine dataset and split the target domain data randomly into an (unlabeled) training and a test set comprising 24 and 16 samples, respectively.

*Tablets.* The Tablets dataset was originally published by the international diffuse reflectance conference (IDRC) in 2002 and consists of NIR spectra of 654 pharmaceutical tablets recorded on 2 spectrometers at 650 individual wavelengths.[6] The analytical goal is to predict the active pharmaceutical ingredient (API) concentration from the NIR spectra. The dataset is divided into a calibration, a validation and a test set comprising 155, 40 and 460 samples measured on both instruments. We consider inductive DA between calibration and test sets from the two instruments including the wavelength range 600–1600 nm and proceed in analogy with the experiments on the Corn and Melamine datasets.

### 6.3. Results

*Melamine dataset.* Table 1 shows the prediction errors on the 12 DA scenarios of the Melamine dataset. Application of CORAL on the source and target projections of the baseline model did not yield any improvements on the target task. In contrast, TCA and JDOT significantly reduced the target error, with the former achieving significantly lower errors on most scenarios. However, TCA and JDOT yielded reasonable performance in the target domain only when the corresponding hyperparameters were set in a supervised way (i.e. making use of label information in the target domain). In contrast, our method (DIPALS) clearly outperformed TCA, JDOT and di-PLS in most scenarios using the hyperparameter heuristic in Eq. (14). Indeed the parameter heuristic yielded overall good performance on the target tasks, whereas for some scenarios (e.g. 862 → 562) setting $\gamma$ such the CV error on the source task was minimized yielded superior accuracy. We found that the improvement over the baseline model is not only due to reduction of domain discrepancy in the latent variable space but can also be attributed to the fact that domain regularization reduced the error on the source task in all scenarios (Fig. 2). Given the increase in the variance of the regression vector (i.e. $\|\mathbf{b}\|_2^2$) with increasing $\gamma$ this observations indicates that alignment of source and target data in fact reduces the bias of the (source) model.

*Corn dataset.* In contrast to the Melamine dataset, where more complicated distribution shifts occur between domains due to qualitative changes in sample composition, the changes observed in the Corn dataset occur due to changes in the instruments' response and are mostly manifested in offsets between the corresponding spectra. All in all, we found similar performance of TCA and DIPALS with slightly better results with the former for prediction of oil content and with the latter when predicting moisture (Table 2). Although JDOT and di-PLS could improve the accuracy on the target task for determination of protein and starch (compared to the baseline), accuracy was significantly lower in most scenarios compared to DIPALS. Finally, no improvement of the baseline model could be achieved with CORAL.

[4] https://github.com/RNL1/Melamine-Dataset

[5] http://www.eigenvector.com/data/Corn/ (accessed April 11, 2018).

[6] http://www.eigenvector.com/data/tablets/ (accessed January 14, 2019).

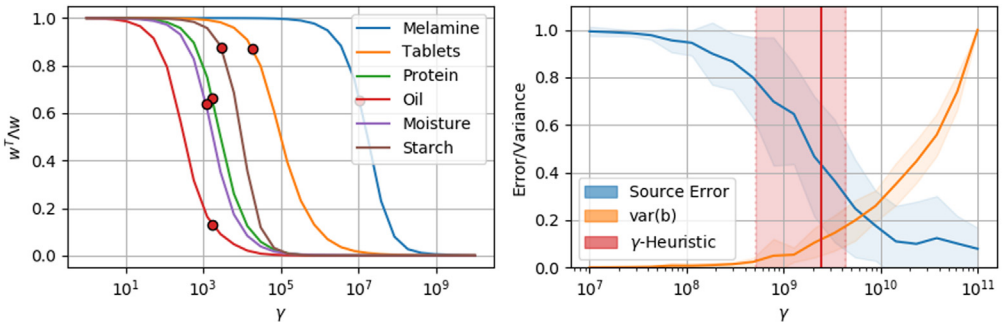


**Fig. 2.** The distance measure changed here: DIPALS hyperparameter heuristic. Left: Magnitude of the regularization term against increasing values of the regularization parameter $\gamma$ shown for different datasets. The red dots indicate the heuristic choice of $\gamma$ according to Eq. (14). Right: The average mean squared error on the source task and variance of the regression vector **b** at increasing $\gamma$ computed from all DA scenarios of the Melamine dataset. The red vertical line indicates the average $\gamma$ obtained using Eq. (14). Note that all trajectories have been normalized to lie in [0, 1] for better comparability. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

**Table 1**
Accuracy on the Melamine dataset. The average root mean squared errors across 10 experiments with standard deviations are shown. The best value for each scenario is indicated in bold. (Sup) indicates supervised hyperparameter selection using target domain labels, (Heur) indicates that the regularization parameter $\gamma$ was set according to Eq. (14) and (S) indicates that $\gamma$ was set such that the CV Error on the source task was minimized.

| Scenario | NIPALS | CORAL | TCA (Sup) | JDOT (Sup) | di-PLS[a] | DIPALS (Heur) | DIPALS (S) |
|---|---|---|---|---|---|---|---|
| 562 → 568 | 4.83 ± 0.21 | 4.99 ± 0.24 | 2.04 ± 0.14 | 3.23 ± 0.30 | 2.47 | 1.75 ± 0.14 | **1.69 ± 0.23** |
| 562 → 861 | 5.65 ± 0.26 | 5.55 ± 0.18 | 2.46 ± 0.14 | 3.10 ± 0.15 | 3.28 | **2.13 ± 0.11** | 2.20 ± 0.13 |
| 562 → 862 | 3.22 ± 0.29 | 3.27 ± 0.21 | 2.59 ± 0.31 | 2.76 ± 0.20 | 2.23 | 2.09 ± 0.31 | **2.04 ± 0.34** |
| 568 → 562 | 3.96 ± 0.10 | 3.94 ± 0.05 | 2.27 ± 0.11 | 2.95 ± 0.15 | 2.58 | 2.02 ± 0.14 | **1.99 ± 0.12** |
| 568 → 861 | 5.13 ± 0.29 | 5.21 ± 0.26 | 2.44 ± 0.13 | 2.66 ± 0.09 | 2.92 | **2.13 ± 0.10** | 2.22 ± 0.22 |
| 568 → 862 | 3.39 ± 0.37 | 3.32 ± 0.19 | 2.71 ± 0.30 | 3.12 ± 0.23 | 2.38 | 2.26 ± 0.50 | **2.19 ± 0.31** |
| 861 → 562 | 4.22 ± 0.07 | 4.27 ± 0.14 | 2.35 ± 0.12 | 3.48 ± 0.15 | 3.29 | 2.15 ± 0.16 | **2.03 ± 0.18** |
| 861 → 568 | 4.64 ± 0.12 | 4.72 ± 0.18 | 2.16 ± 0.10 | 2.61 ± 0.31 | 3.01 | **1.85 ± 0.09** | 2.01 ± 0.37 |
| 861 → 862 | 3.69 ± 0.22 | 3.79 ± 0.20 | 2.76 ± 0.16 | 3.69 ± 0.19 | 2.81 | **1.94 ± 0.18** | 2.16 ± 0.53 |
| 862 → 562 | 4.74 ± 0.08 | 4.76 ± 0.11 | 3.25 ± 0.09 | 2.84 ± 0.10 | 3.52 | 3.07 ± 0.25 | **2.53 ± 0.13** |
| 862 → 568 | 5.28 ± 0.22 | 5.35 ± 0.20 | **2.85 ± 0.17** | 2.96 ± 0.38 | 4.11 | 3.10 ± 0.46 | 3.14 ± 0.38 |
| 862 → 861 | 6.17 ± 0.14 | 6.14 ± 0.16 | 3.90 ± 0.13 | **3.06 ± 0.08** | 4.90 | 3.64 ± 0.37 | 3.56 ± 0.21 |

[a]Values taken from [24].

**Table 2**
Accuracy on the Corn dataset. The average root mean squared errors across 10 experiments with standard deviations are shown. The best value for each scenario is indicated in bold. (Sup) indicates supervised hyperparameter selection using target domain labels, (Heur) indicates that the regularization parameter $\gamma$ was set according to Eq. (14) and (S) indicates that $\gamma$ was set such that the CV Error on the source task was minimized.

| Response | Scenario | NIPALS | CORAL | TCA (Sup) | JDOT (Sup) | di-PLS | DIPALS (Heur) | DIPALS (S) |
|---|---|---|---|---|---|---|---|---|
| Protein | m5→mp5 | 0.68 ± 0.13 | 0.65 ± 0.14 | 0.40 ± 0.03 | 0.61 ± 0.06 | 0.44 ± 0.07 | **0.38 ± 0.08** | 0.39 ± 0.09 |
| | m5→mp6 | 0.70 ± 0.12 | 0.77 ± 0.11 | 0.41 ± 0.05 | 0.57 ± 0.04 | 0.47 ± 0.13 | **0.41 ± 0.03** | 0.42 ± 0.10 |
| | mp5→m5 | 0.70 ± 0.11 | 0.71 ± 0.09 | **0.43 ± 0.09** | 0.57 ± 0.05 | 0.44 ± 0.04 | 0.44 ± 0.07 | 0.44 ± 0.08 |
| | mp5→mp6 | 0.65 ± 0.12 | 0.66 ± 0.12 | **0.36 ± 0.04** | 0.59 ± 0.07 | 0.50 ± 0.06 | 0.50 ± 0.08 | 0.49 ± 0.04 |
| | mp6→m5 | 0.69 ± 0.10 | 0.69 ± 0.08 | 0.43 ± 0.09 | 0.58 ± 0.07 | 0.46 ± 0.09 | **0.43 ± 0.07** | 0.43 ± 0.12 |
| | mp6→mp5 | 0.66 ± 0.11 | 0.61 ± 0.15 | 0.41 ± 0.05 | 0.44 ± 0.05 | 0.43 ± 0.07 | 0.40 ± 0.06 | **0.36 ± 0.03** |
| Starch | m5→mp5 | 1.19 ± 0.17 | 1.11 ± 0.19 | 0.70 ± 0.08 | 0.76 ± 0.08 | 0.68 ± 0.17 | 0.69 ± 0.18 | **0.66 ± 0.10** |
| | m5→mp6 | 1.08 ± 0.17 | 1.11 ± 0.21 | 0.67 ± 0.10 | 0.75 ± 0.08 | 0.65 ± 0.14 | **0.64 ± 0.15** | 0.68 ± 0.12 |
| | mp5→m5 | 1.38 ± 0.26 | 1.30 ± 0.17 | 0.68 ± 0.11 | 0.83 ± 0.06 | 0.82 ± 0.14 | 0.71 ± 0.08 | **0.67 ± 0.15** |
| | mp5→mp6 | 1.20 ± 0.16 | 1.27 ± 0.13 | **0.68 ± 0.09** | 0.82 ± 0.08 | 0.74 ± 0.13 | 0.80 ± 0.14 | 0.72 ± 0.15 |
| | mp6→m5 | 1.38 ± 0.17 | 1.48 ± 0.23 | **0.64 ± 0.10** | 0.82 ± 0.07 | 0.80 ± 0.14 | 0.76 ± 0.14 | 0.69 ± 0.15 |
| | mp6→mp5 | 1.21 ± 0.14 | 1.30 ± 0.11 | **0.55 ± 0.08** | 0.81 ± 0.06 | 0.71 ± 0.09 | 0.72 ± 0.18 | 0.89 ± 0.19 |
| Oil | m5→mp5 | 0.27 ± 0.03 | 0.27 ± 0.05 | **0.15 ± 0.02** | 0.20 ± 0.03 | 0.20 ± 0.02 | 0.19 ± 0.03 | 0.19 ± 0.01 |
| | m5→mp6 | 0.24 ± 0.05 | 0.30 ± 0.03 | **0.14 ± 0.01** | 0.22 ± 0.03 | 0.19 ± 0.03 | 0.17 ± 0.02 | 0.18 ± 0.04 |
| | mp5→m5 | 0.21 ± 0.02 | 0.24 ± 0.03 | **0.16 ± 0.02** | 0.22 ± 0.02 | 0.25 ± 0.03 | 0.26 ± 0.04 | 0.21 ± 0.02 |
| | mp5→mp6 | 0.21 ± 0.03 | 0.21 ± 0.03 | **0.15 ± 0.02** | 0.20 ± 0.03 | 0.21 ± 0.03 | 0.20 ± 0.02 | 0.21 ± 0.03 |
| | mp6→m5 | 0.22 ± 0.02 | 0.23 ± 0.03 | **0.17 ± 0.01** | 0.22 ± 0.03 | 0.22 ± 0.05 | 0.23 ± 0.05 | 0.19 ± 0.02 |
| | mp6→mp5 | 0.20 ± 0.03 | 0.22 ± 0.03 | **0.16 ± 0.02** | 0.21 ± 0.02 | 0.21 ± 0.04 | 0.21 ± 0.04 | 0.18 ± 0.01 |
| Moisture | m5→mp5 | 0.24 ± 0.03 | 0.28 ± 0.04 | 0.24 ± 0.02 | 0.30 ± 0.04 | 0.22 ± 0.05 | **0.22 ± 0.04** | 0.22 ± 0.08 |
| | m5→mp6 | 0.27 ± 0.03 | 0.27 ± 0.04 | 0.27 ± 0.02 | 0.31 ± 0.04 | 0.23 ± 0.05 | 0.25 ± 0.03 | **0.20 ± 0.02** |
| | mp5→m5 | 0.26 ± 0.03 | 0.27 ± 0.03 | 0.29 ± 0.03 | 0.27 ± 0.05 | 0.24 ± 0.05 | **0.23 ± 0.06** | 0.25 ± 0.06 |
| | mp5→mp6 | 0.27 ± 0.02 | 0.26 ± 0.04 | 0.28 ± 0.02 | 0.31 ± 0.03 | **0.18 ± 0.06** | 0.23 ± 0.04 | 0.20 ± 0.03 |
| | mp6→m5 | 0.28 ± 0.03 | 0.28 ± 0.03 | 0.31 ± 0.03 | 0.24 ± 0.01 | 0.24 ± 0.06 | **0.22 ± 0.03** | 0.22 ± 0.05 |
| | mp6→mp5 | 0.27 ± 0.03 | 0.26 ± 0.02 | 0.26 ± 0.03 | 0.31 ± 0.04 | 0.19 ± 0.04 | **0.17 ± 0.02** | 0.20 ± 0.03 |

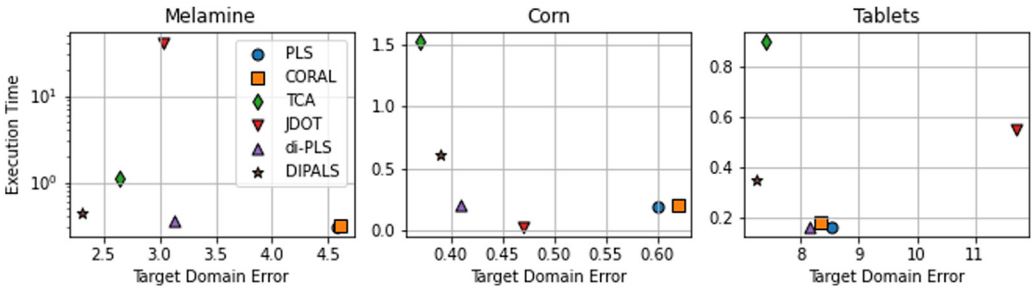


**Fig. 3.** Target domain error vs. CPU time.

**Table 3**
Accuracy on the Tablets dataset. The average root mean squared errors across 10 experiments with standard deviations are shown. The best value for each scenario is indicated in bold. (Sup) indicates supervised hyperparameter selection using target domain labels, (Heur) indicates that the regularization parameter $\gamma$ was set according to Eq. (14) and (S) indicates that $\gamma$ was set such that the CV Error on the source task was minimized.

| Scenario | NIPALS | CORAL | TCA (Sup) | JDOT (Sup) | di-PLS | DIPALS (Heur) | DIPALS (S) |
|---|---|---|---|---|---|---|---|
| cal1→test2 | 9.48 ± 0.64 | 9.56 ± 0.58 | 8.50 ± 0.59 | 12.86 ± 1.07 | 8.69 ± 1.06 | **7.69 ± 0.47** | 8.04 ± 0.53 |
| test2→cal1 | 8.18 ± 0.94 | 7.89 ± 1.01 | 7.23 ± 1.04 | 10.26 ± 0.67 | 7.77 ± 1.14 | **6.54 ± 1.25** | 7.58 ± 1.59 |
| cal2→test1 | 8.35 ± 0.58 | 7.51 ± 0.73 | **6.63 ± 0.92** | 13.46 ± 0.50 | 7.98 ± 0.83 | 7.12 ± 0.68 | 6.75 ± 0.48 |
| test1→cal2 | 8.12 ± 1.55 | 8.39 ± 1.33 | **7.26 ± 1.83** | 10.24 ± 0.55 | 8.20 ± 1.26 | 7.66 ± 1.66 | 8.43 ± 1.25 |

**Table 4**
Average execution times of the different domain adaptation methods in seconds.

| | PLS | CORAL | TCA | JDOT | di-PLS | DIPALS |
|---|---|---|---|---|---|---|
| Melamine | 0.30 | 0.32 | 1.11 | 41.5 | 0.36 | 0.45 |
| Corn | 0.19 | 0.20 | 1.53 | 0.02 | 0.20 | 0.61 |
| Tablets | 0.16 | 0.18 | 0.90 | 0.55 | 0.16 | 0.35 |

*Tablets dataset.* The Tablets dataset, similar to the Corn datasets, involves DA between similar NIR spectrometers. Accordingly, we found similar overall performance of DIPALS and TCA on the target tasks (Table 3). In contrast, JDOT could not surpass the accuracy of the baseline model, which can be explained with the fact that the Tablets dataset contains several **y**-direction outliers (i.e. spectra with wrongly assigned API values) that apparently lead to erroneous transport of the joint distribution.

*CPU time.* Table 4 shows the average execution times for training and testing of PLS, CORAL, TCA, JDOT, di-PLS and DIPALS models for fixed parameter sets on all domain adaptation scenarios from Tables 1–3 including the time required for pre-processing training and test data (where necessary). For better comparability, the number of LVs was fixed to 5 for PLS, CORAL, TCA, di-PLS and DIPALS. In particular, CORAL and di-PLS have execution times similar to the baseline PLS model. In contrast, DIPALS shows $1.5 - 3$ times higher execution times which can be attributed to the computational complexity of the convex relaxation in Eq. (7) including the eigendecomposition of a matrix, which is especially time-consuming for DA problems involving a large number of predictors (e.g. problems on the Corn data set). JDOT solves the Earth Movers' distance problem to come up with the optimal transport plan between source and target domain samples, which scales exponentially with the number of samples involved. Thus, for larger scale problems (e.g. the Melamine data set) the execution time of JDOT becomes very large. Finally, (primal) TCA solves a generalized eigenvalue problem, which scales similar to DIPALS with the number of predictors but involves factorization of two (i.e. total and domain scatter) matrices. Finally, Fig. 3 shows the average execution times vs. the target domain error, where DIPALS compares favorably with the competing methods.

## 7. Conclusion

We have here considered unsupervised DA for multivariate regression under linear input–output relationship, multicollinearity and approximately normally distributed domains – a situation frequently encountered in analytical chemistry. We proposed a novel algorithm that performs DA under the non-iterative partial least squares (NIPALS) framework by extending the NIPALS objective by a domain regularization term. Notably, the solution of the underlying convex optimization problem is obtained in closed-form for each latent variable yielding a linear subspace with high predictive power towards a (continuous) response and low domain discrepancy. This is in contrast to other DA techniques that rely on numerical optimization or employ dimensionality reduction on (non-linear) feature spaces. The latter can be considered inappropriate if (i) the underlying input–output relationship is linear and/or (ii) when aiming at interpretable models, which are usually preferred in analytical chemistry. Finally, we have demonstrated superiority of our approach over different state-of-the-art DA techniques for linear regression on the Melamine dataset and similar performance on the Corn and Tablets datasets.

## CRediT authorship contribution statement

**Ramin Nikzad-Langerodi:** Conceptualization, Methodology, Software, Validation, Investigation, Data curation, Writing - original draft, Writing - review & editing, Visualization. **Werner Zellinger:** Conceptualization, Methodology, Formal analysis, Writing - original draft, Writing review & editing, Visualization, Project administration. **Susanne Saminger-Platz:** Conceptualization, Formal analysis, Resources, Writing - review & editing, Supervision, Funding acquisition. **Bernhard A. Moser:** Conceptualization, Formal analysis, Resources, Writing - review & editing, Supervision, Funding acquisition.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

The first author acknowledges the COMET Centre CHASE which is funded within the framework of COMET - Competence Centers for Excellent Technologies by BMVIT, BMDW, the Federal Provinces of Upper Austria and Vienna run by the Austrian research funding association (FFG). The work of the second, third and fourth author was supported by the Austrian Ministry for Climate Action, Environment, Energy, Mobility, Innovation and Technology, and the Province of Upper Austria in the frame of the COMET center SCCH. The second and fourth author in addition acknowledge the Federal Ministry for Digital and Economic Affairs (BMDW), and the Province of Upper Austria in the frame of the COMET - Competence Centers for Excellent Technologies Programme and the COMET Module S3AI managed by FFG.

## References

[1] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2010) 1345–1359, http://dx.doi.org/10.1109/TKDE.2009.191.
[2] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, J.W. Vaughan, A theory of learning from different domains, Mach. Learn. 79 (1) (2010) 151–175, http://dx.doi.org/10.1007/s10994-009-5152-4.
[3] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, V. Lempitsky, Domain-adversarial training of neural networks, J. Mach. Learn. Res. 17 (Jan) (2016) 1–35, http://dx.doi.org/10.1007/978-3-319-58347-110.
[4] B. Malli, A. Birlutiu, T. Natschläger, Standard-free calibration transfer—An evaluation of different techniques, Chemometr. Intell. Lab. Syst. 161 (2017) 49–60, http://dx.doi.org/10.1016/j.chemolab.2016.12.008.
[5] R. Nikzad-Langerodi, E. Lughofer, C. Cernuda, T. Reischer, W. Kantner, M. Pawliczek, M. Brandstetter, Calibration model maintenance in melamine resin production: Integrating drift detection, smart sample selection and model adaptation, Anal. Chim. Acta 1013 (2018) 1–12, http://dx.doi.org/10.1016/j.aca.2018.02.003, (featured article).
[6] J.J. Workman, A review of calibration transfer practices and instrument differences in spectroscopy, Appl. Spectrosc. 72 (3) (2018) 340–365, http://dx.doi.org/10.1016/j.aca.2017.12.027.
[7] E. Andries, Penalized eigendecompositions: motivations from domain adaptation for calibration transfer, J. Chemometr. 31 (4) (2017) e2818.
[8] R. Nikzad-Langerodi, F. Sobieczky, Graph-based calibration transfer, 2020, arXiv:2006.00089.
[9] D.F. Swinehart, The Beer–Lambert law, J. Chem. Educ. 39 (7) (1962) 333, http://dx.doi.org/10.1021/ed039p333.
[10] H. Mark, J. Workman Jr., Chemometrics in Spectroscopy, Elsevier, 2010.
[11] R.C. Bradley Jr., Central limit theorems under weak dependence, J. Multivariate Anal. 11 (1) (1981) 1–16.
[12] H. Wold, Soft modelling by latent variables: the non-linear iterative partial least squares (NIPALS) approach, J. Appl. Probab. 12 (S1) (1975) 117–142, http://dx.doi.org/10.1017/S0021900200047604.
[13] R. Nikzad-Langerodi, W. Zellinger, S. Saminger-Platz, B. Moser, Domain-invariant regression under Beer–Lambert's law, in: 2019 18th IEEE International Conference on Machine Learning and Applications, ICMLA, 2019, pp. 581–586, http://dx.doi.org/10.1109/ICMLA.2019.00108.
[14] H. Shimodaira, Improving predictive inference under covariate shift by weighting the log-likelihood function, J. Stat. Plann. Inference 90 (2) (2000) 227–244, http://dx.doi.org/10.1016/S0378-3758(00)00115-4.
[15] J. Huang, A. Gretton, K.M. Borgwardt, B. Schölkopf, A.J. Smola, Correcting sample selection bias by unlabeled data, in: Advances in Neural Information Processing Systems, 2007, pp. 601–608.
[16] A. Gretton, A.J. Smola, J. Huang, M. Schmittfull, K.M. Borgwardt, B. Schölkopf, Covariate Shift by Kernel Mean Matching, MIT Press, 2009.
[17] J. Blitzer, R. McDonald, F. Pereira, Domain adaptation with structural correspondence learning, in: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, EMNLP '06, Association for Computational Linguistics, Stroudsburg, PA, USA, 2006, pp. 120–128, URL http://dl.acm.org/citation.cfm?id=1610075.1610094.
[18] S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain adaptation via transfer component analysis, IEEE Trans. Neural Netw. 22 (2) (2011) 199–210, http://dx.doi.org/10.1109/TNN.2010.2091281.
[19] A. Gretton, K.M. Borgwardt, M.J. Rasch, B. Schölkopf, A. Smola, A kernel two-sample test, J. Mach. Learn. Res. 13 (3) (2012) 723–773.
[20] M. Ghifary, D. Balduzzi, W.B. Kleijn, M. Zhang, Scatter component analysis: A unified framework for domain adaptation and domain generalization, IEEE Trans. Pattern Anal. Mach. Intell. 39 (2017) 1414–1430.
[21] F. Sun, H. Wu, Z. Luo, W. Gu, Y. Yan, Q. Du, Informative feature selection for domain adaptation, IEEE Access 7 (2019) 142551–142563, http://dx.doi.org/10.1109/ACCESS.2019.2944226.
[22] W. Deng, A. Lendasse, Y. Ong, I.W. Tsang, L. Chen, Q. Zheng, Domain adaption via feature selection on explicit feature map, IEEE Trans. Neural Netw. Learn. Syst. 30 (4) (2019) 1180–1190, http://dx.doi.org/10.1109/TNNLS.2018.2863240.
[23] M. Baktashmotlagh, M.T. Harandi, B.C. Lovell, M. Salzmann, Unsupervised domain adaptation by domain invariant projection, in: Proceedings of the 2013 IEEE International Conference on Computer Vision, ICCV '13, IEEE Computer Society, Washington, DC, USA, 2013, pp. 769–776, http://dx.doi.org/10.1109/ICCV.2013.100.
[24] R. Nikzad-Langerodi, W. Zellinger, E. Lughofer, S. Saminger-Platz, Domain-invariant partial-least-squares regression, Anal. Chem. 90 (11) (2018) 6693–6701, http://dx.doi.org/10.1021/acs.analchem.8b00498.
[25] W. Zellinger, T. Grubinger, M. Zwick, E. Lughofer, H. Schöner, T. Natschläger, S. Saminger-Platz, Multi-source transfer learning of time series in cyclical manufacturing, J. Intell. Manuf. 31 (3) (2020) 777–787.
[26] M. Gong, K. Zhang, T. Liu, D. Tao, C. Glymour, B. Schölkopf, Domain adaptation with conditional transferable components, in: International Conference on Machine Learning, 2016, pp. 2839–2848, URL http://dl.acm.org/citation.cfm?id=3045390.3045689.
[27] N. Courty, R. Flamary, A. Habrard, A. Rakotomamonjy, Joint distribution optimal transportation for domain adaptation, in: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4–9 December 2017, Long Beach, CA, USA, 2017, pp. 3733–3742, URL https://arxiv.org/abs/1705.08848.
[28] B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic flow kernel for unsupervised domain adaptation, in: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE, 2012, pp. 2066–2073.
[29] B. Sun, K. Saenko, Deep coral: Correlation alignment for deep domain adaptation, in: Computer Vision–ECCV 2016 Workshops, Springer, 2016, pp. 443–450.
[30] H. Zuo, G. Zhang, W. Pedrycz, V. Behbood, J. Lu, Fuzzy regression transfer learning in Takagi–Sugeno fuzzy models, IEEE Trans. Fuzzy Syst. 25 (6) (2017) 1795–1807.
[31] H. Zuo, G. Zhang, W. Pedrycz, V. Behbood, J. Lu, Granular fuzzy regression domain adaptation in Takagi–Sugeno fuzzy models, IEEE Trans. Fuzzy Syst. 26 (2) (2018) 847–858, http://dx.doi.org/10.1109/TFUZZ.2017.2694801.
[32] M. Sugiyama, M. Kawanabe, Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation, MIT Press, 2012.
[33] S. Ben-David, J. Blitzer, K. Crammer, F. Pereira, Analysis of representations for domain adaptation, in: Advances in Neural Information Processing Systems, 2007, pp. 137–144, URL http://papers.nips.cc/paper/2983-analysis-of-representations-for-domain-adaptation.pdf.
[34] R.M. Dudley, Real Analysis and Probability, Vol. 74, Cambridge University Press, 2002.
[35] W. Zellinger, Moment-Based Domain Adaptation: Learning Bounds and Algorithms (Doctoral thesis), Johannes Kepler University Linz, 2020, April.
[36] A.R. Barron, C.-H. Sheu, Approximation of density functions by sequences of exponential families, Ann. Statist. (1991) 1347–1369, http://dx.doi.org/10.1214/aos/1176348252.
[37] T.M. Cover, J.A. Thomas, Elements of Information Theory, John Wiley & Sons, 2012.
[38] A.L. Gibbs, F.E. Su, On choosing and bounding probability metrics, Int. Stat. Rev. 70 (3) (2002) 419–435.
[39] P. Geladi, B.R. Kowalski, Partial least-squares regression: a tutorial, Anal. Chim. Acta 185 (1986) 1–17.
[40] R. Manne, Analysis of two partial-least-squares algorithms for multivariate calibration, Chemometr. Intell. Lab. Syst. 2 (1) (1987) 187–197, http://dx.doi.org/10.1016/0169-7439(87)80096-5.
[41] V.Y. Pan, Z.Q. Chen, The complexity of the matrix eigenproblem, in: Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, 1999, pp. 507–516.
[42] B. Sun, J. Feng, K. Saenko, Return of frustratingly easy domain adaptation, in: AAAI, 2016, URL https://dl.acm.org/citation.cfm?id=3016186.
[43] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, in: Springer Series in Statistics, Springer New York Inc., 2001.