Received: 30 April 2016 | Revised: 19 June 2016 | Accepted: 7 July 2016

DOI: 10.1002/cem.2818

WILEY Journal of CHEMOMETRICS

SPECIAL ISSUE ARTICLE

Penalized eigendecompositions: motivations from domain adaptation for calibration transfer

Erik Andries

Department of Mathematics, Science and Engineering, Central New Mexico Community College, Albuquerque, NM 87124, U.S.A.

Correspondence
Erik Andries, Department of Mathematics, Science and Engineering, Central New Mexico Community College, Albuquerque, NM 87124, U.S.A.
Email: eandries@cnm.edu

Maintaining multivariate calibrations involves keeping models developed on an instrument applicable to predicting new samples over time. Sometimes, a primary instrument model is needed to predict samples measured on secondary instruments. This situation is referred to as calibration transfer. Sometimes, a primary instrument model is needed to predict samples that have acquired new spectral features (chemical, physical, and environmental influences) over time. This situation is referred to as calibration maintenance. Calibration transfer and maintenance problems have a long history and are well studied in chemometrics and spectroscopy. In disciplines outside of chemometrics, particularly computer vision, calibration transfer and maintenance problems are more recent phenomena, and these problems often go under the umbrella term *domain adaptation*. Over the past decade, domain adaptation has demonstrated significant successes in various applications such as visual object recognition. Since domain adaptation already constitutes a large area of research in computer vision and machine learning, we narrow our scope and report on penalty-based eigendecompositions, a class of domain adaptation methods that has its motivational roots in linear discriminant analysis. We compare these approaches against chemometrics-based approaches using several benchmark chemometrics data sets.

KEYWORDS

calibration transfer and maintenance, domain adaptation, eigendecomposition

1 | INTRODUCTION

The normal process of building a calibration model typically requires a large set of samples such that all spectral variances are included in future predictions. However, once a calibration model has been built, circumstances can cause the model to become invalid. For example, instrumental drift or uncalibrated spectral features appearing in new samples can occur later in time. Alternatively, an unknown sample could be measured on an instrument other than the instrument that the calibration model was built on. For these and other situations, the instrument must be recalibrated to accommodate new conditions. In theory, the remedy would be to include a large number of new calibration samples. In practice, though, the inclusion of such samples is often prohibitively costly and lengthy in terms of laboratory time.

In chemometrics, calibration transfer is often synonymous with *instrument transfer*. Samples from the primary instrument are used to build the original model, while the secondary samples are from another instrument. However, appreciable instrument-to-instrument variations almost always exists, e.g., differences in wavelength resolution and detector sensitivity. The expression *calibration maintenance* corresponds to the scenario where the secondary instrument is in fact the primary instrument but the samples being obtained later in time are occurring under different measuring conditions. Depending on the instrument and sample type, other chemical, physical, and environmental influences can cause new spectral features to appear later in time. Hence, mechanisms are needed to update the current model to include the new spectral effects not in the current calibration domain.

In this paper, the aim of calibration transfer and maintenance is to make predictions on new secondary samples. If the secondary samples are similar to the primary samples, then one can simply pool all of the samples together and build a global calibration model. However, if the secondary samples

are radically different from the primary samples, then one can ignore the primary samples and build a calibration model solely on the secondary samples, provided that there are enough samples. The usual statistical assumption—that the secondary samples are drawn from the same probability distribution that governs the primary samples—is at best a useful fiction for many real-world applications. Although we expect the secondary samples to be dissimilar (but not that dissimilar) to the primary samples, the primary samples should provide additional leverage such that an improved prediction can be obtained for secondary samples.

Calibration transfer and maintenance problems have a long history and are well studied in chemometrics and spectroscopy; see literature[1–18] and references therein. In disciplines outside of chemometrics (e.g., computer vision, image processing, text mining, audio, and language processing), calibration transfer and maintenance problems are more recent phenomena, and the techniques used to solve these problems go by different names: domain adaptation, transfer learning, concept drift, or covariate shift; see literature[19–28] and references therein. For brevity and clarity, we will heretofore use the following acronyms to refer to the following:

- CU: Calibration updating. CU will refer to both calibration transfer and maintenance approaches specific to chemometrics and spectroscopy.
- DA: Domain adaptation. This acronym will be used as an umbrella term for all CU approaches used in disciplines outside of chemometrics and spectroscopy.

In DA, data set bias naturally occurs, e.g., recognizing objects under poor lighting conditions while algorithms are trained on well-illuminated objects, and facial recognition when images are trained from frontal poses while the test set consists of side poses. Although CU and DA share many of the same problems, the research efforts of these communities have been largely unaware of each other.

Classification or retrieval algorithms dominate in DA applications, while regression dominates in most CU settings, e.g., prediction of analyte concentrations. Also, the size of the data sets in these communities is vastly different in scale. In DA, it is not uncommon for the number and dimension of the samples to be *massive*, whereas in CU, the size of data sets is usually much more modest. As a result, nonlinear methods are usually used in DA applications, while linear methods generally suffice in most CU scenarios. Moreover, discipline-specific priors used to eliminate poor or suboptimal solutions in DA often have no analogs in CU (and vice versa). Despite these differences, insights can be gleaned from the DA literature, providing promising avenues for further research and effort.

The CU and DA are large areas of research in their own spheres of influence. As a result, we intentionally narrow our scope to penalty-based eigendecompositions. Eigendecompositions describe a class of DA methods based upon generalized eigenvalue problems (GEPs), which amount to solving the equation $\mathbf{Tv} = \lambda\mathbf{Dv}$ with respect to the eigenvalue-eigenvector pair $(\lambda, \mathbf{v})$. Note that we are not investigating the following:

- Methods that require a standardization set.[1–5] A standardization set is a common set of samples measured across 2 or more instruments. (It could also be a common set of samples measured across 2 or more different measuring conditions). Since each standardization sample has the same reference value across instruments, the variability of the spectral measurements should largely reflect instrument-to-instrument difference. To establish transfer parameters, the standardization samples also need to be representative of the entire experimental regime and stable enough over time between situations in which the standardization is performed. For certain data sets examined here, the creation of a standardization set is not possible.
- Spectral preprocessing techniques.[7,8,11,29–32] These techniques refer to methods (e.g., wavelets and direct piecewise standardization) that transform the spectra to minimize domain differences between the primary and secondary samples without the use of a standardization set.

This is not to say that one cannot use standardization sets or preprocessing techniques by themselves or in tandem with eigendecompositions, but those investigations are outside the scope of this paper.

This paper is organized as follows. Section 2 reviews the maximization of Rayleigh quotients and the corresponding generalized eigenproblems that result from this optimization. We recast the eigenproblem framework of linear discriminant analysis (LDA) and reappropriate it for CU and DA purposes. Section 3 examines 3 DA-based eigendecompositions methods and their solution via a GEP. Section 4 describes the data sets used for algorithm assessment, and Section 5 discusses algorithm implementation and model selection procedures. Section 6 provides the analyses and comparative results. Finally, Section 7 concludes the paper.

We now discuss notation. Symbols that are not boldface represent scalars (x or P). Lowercase and uppercase boldface symbols represent column vectors ($\mathbf{x}$) or matrices ($\mathbf{X}$). All vectors are column vectors unless noted otherwise noted. The superscripted symbols ^{T} and $^{-1}$ indicate the transpose and inverse, respectively, of a vector or matrix. The matrices $\mathbf{I}$ and $\mathbf{0}$ and vectors $\mathbf{1}_n$ and $\mathbf{0}_n$ indicate the identity matrix, a matrix of 0's and a column vector of n ones and 0's, respectively. The comma and semicolon indicate the horizontal and vertical concatenation (or stacking) of matrix/vector entries. For example, it will be convenient to represent an $m \times n$ matrix of spectra $\mathbf{X}$ by concatenating column vectors $\mathbf{X} = [\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_m]^T$ such that $\mathbf{x}_j=[x_{j1};x_{j2};\ldots;x_{jn}]$ (or $\mathbf{x}_j = [x_{j1},x_{j2},\ldots,x_{jn}]^T$) corresponds to the jth spectrum. The vector $\mathbf{y} = [y_1;y_2;\ldots;y_m]$ (or $\mathbf{y} = [y_1,y_2,\ldots,y_m]^T$) represents the response variables (e.g., reference values such as analyte concentrations). In this paper, $\mathbf{X}^{(\mathrm{P})} = [\mathbf{x}_1^{(\mathrm{P})},\ldots,\mathbf{x}_{m_\mathrm{P}}^{(\mathrm{P})}]^T$ and $\mathbf{y}^{(\mathrm{P})} = [\mathbf{y}_1^{(\mathrm{P})},\ldots,\mathbf{y}_{m_\mathrm{P}}^{(\mathrm{P})}]^T$ correspond to m_P samples of

primary spectra and reference values, respectively. Similarly, $\mathbf{X}^{(S)} = [\mathbf{x}_1^{(S)}, \ldots, \mathbf{x}_{m_S}^{(S)}]^T$ and $\mathbf{y}^{(S)} = [\mathbf{y}_1^{(S)}, \ldots, \mathbf{y}_{m_S}^{(S)}]^T$ correspond to m_S samples of secondary spectra and their respective reference values.

2 | GENERALIZED EIGENPROBLEMS

Generalized eigenproblems naturally arise in many diverse fields such as signal processing, pattern recognition, and machine learning.[33–35] These problems involve 2 matrices, $\mathbf{T}$ and $\mathbf{D}$, that are referred to as scatter matrices. These scatter matrices occur in a Rayleigh quotient $\mathcal{R}(\mathbf{v})$ that is subsequently maximized:

$$\max_{\mathbf{v}} \ \mathcal{R}(\mathbf{v}) \qquad \text{where} \qquad \mathcal{R}(\mathbf{v}) = \frac{\mathbf{v}^T\mathbf{T}\mathbf{v}}{\mathbf{v}^T\mathbf{D}\mathbf{v}}. \tag{1}$$

Maximizing $\mathcal{R}(\mathbf{v})$ results in a vector $\mathbf{v}$ that projects the spectra $\mathbf{X}$ onto a desired subspace.

Changing the scale of $\mathbf{v}$ (e.g., substituting $\mathbf{v}$ with $\tilde{\mathbf{v}} = \alpha\mathbf{v}$) does not change the value of the Rayleigh quotient $\mathcal{R}(\mathbf{v})$ in Equation 1. The norm of $\mathbf{v}$ is not as important as the direction in which $\mathbf{v}$ points in. To avoid the trivial solution $\mathbf{v} = \mathbf{0}$, one can impose a scalar constraint on $\mathbf{v}$ without a qualitative change in solution. Typically, this constraint imposes a unit norm on the inner product in the denominator in Equation 1, and as a result, the optimization problem in Equation 1 becomes

$$\max_{\mathbf{v}} \ \mathbf{v}^T\mathbf{T}\mathbf{v} \quad \text{subject to} \quad \mathbf{v}^T\mathbf{D}\mathbf{v} = 1. \tag{2}$$

Using the standard Lagrange multiplier approach from calculus, the corresponding Lagrangian function $\mathscr{L}(\mathbf{v})$ associated with Equation 2 results in the following maximization problem (constants omitted):

$$\max_{\mathbf{v}} \ \mathscr{L}(\mathbf{v}) \qquad \text{where} \qquad \mathscr{L}(\mathbf{v}) = \mathbf{v}^T\mathbf{T}\mathbf{v} - \lambda\mathbf{v}^T\mathbf{D}\mathbf{v}. \tag{3}$$

Setting the gradient of $\mathscr{L}(\mathbf{v})$ equal to $\mathbf{0}$, we obtain the following GEP[36,37]:

$$\mathbf{T}\mathbf{v} = \lambda\mathbf{D}\mathbf{v}. \tag{4}$$

2.1 | Eigenpairs

The Lagrange multiplier λ in Equation 4 corresponds to a generalized eigenvalue of the matrix pair ($\mathbf{T}$, $\mathbf{D}$). Although the maximal eigenvalue λ also maximizes the Rayleigh quotient, Equation 4 does have other solutions—the other eigenvalue-eigenvector pairs (λ, $\mathbf{v}$) of ($\mathbf{T}$, $\mathbf{D}$). However, these other eigenpairs do not correspond to the maximum of the Rayleigh quotient. In some applications, only the solution pair (λ,$\mathbf{v}$) associated with the maximal eigenvalue is of interest. For CU purposes, multiple eigenpairs will be of interest,

$$\mathbf{T}\mathbf{V}_k = \mathbf{D}\mathbf{V}_k\mathbf{\Lambda}_k, \tag{5}$$

where $\mathbf{\Lambda}_k = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_k)$ is a diagonal matrix containing k eigenvalues (usually the largest eigenvalues but not always) and $\mathbf{V}_k$=[$\mathbf{v}_1$,$\mathbf{v}_2$, … ,$\mathbf{v}_k$] is a matrix containing the corresponding eigenvectors. The term *eigendecomposition* will refer to the numerical process that extracts the eigenpairs (λ_j, $\mathbf{v}_j$), $j = 1, \ldots, k$.

2.2 | Motivation from LDA

The eigendecomposition strategies we seek for CU purposes are inspired by the minimization and maximization strategies of LDA. For example, in the binary classification setting where the samples in $\mathbf{X}$ belong to either the positive class $\mathcal{C}_+$ or negative class $\mathcal{C}_-$, LDA seeks to minimize within-class scatter and maximize between-class scatter,

$$\text{maximize}\left\{\frac{\text{between-class scatter}}{\text{within-class scatter}}\right\} = \max_{\mathbf{v}} \frac{\mathbf{v}^T\mathbf{D}\mathbf{v}}{\mathbf{v}^T\mathbf{T}\mathbf{v}}. \tag{6}$$

Here, in the LDA context, $\mathbf{D}$ and $\mathbf{T}$ indicate the between-class scatter and within-class scatter matrices, respectively[38]:

$$\mathbf{D} = (\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)(\boldsymbol{\mu}_+ - \boldsymbol{\mu}_-)^T \quad \text{where} \quad \begin{cases} \boldsymbol{\mu}_+ = \frac{1}{m_+}\sum_{k\in\mathcal{C}_+}\mathbf{x}_k \\ \boldsymbol{\mu}_- = \frac{1}{m_-}\sum_{k\in\mathcal{C}_-}\mathbf{x}_k \end{cases} \tag{7}$$

$$\mathbf{T} = \mathbf{T}_+ + \mathbf{T}_- \quad \text{where} \quad \begin{cases} \mathbf{T}_+ = \sum_{k\in\mathcal{C}_+}(\mathbf{x}_k - \boldsymbol{\mu}_+)(\mathbf{x}_k - \boldsymbol{\mu}_+)^T \\ \mathbf{T}_- = \sum_{k\in\mathcal{C}_-}(\mathbf{x}_k - \boldsymbol{\mu}_-)(\mathbf{x}_k - \boldsymbol{\mu}_-)^T. \end{cases} \tag{8}$$

The scalars m_+ and m_- indicate the number of samples in $\mathcal{C}_+$ and $\mathcal{C}_-$, respectively, such that $m_+ + m_- = m$ denotes the total number of samples in $\mathbf{X}$. The vectors $\boldsymbol{\mu}_+$ and $\boldsymbol{\mu}_-$ are the mean spectra or centroids associated with the samples in the positive and negative classes, respectively. The response variables $\mathbf{y} = [y_1, y_2, \ldots, y_m]$, $y_i = \{-1, +1\}$, are the class labels. The label y_i=+1 (y_i=−1) indicates that the ith sample $\mathbf{x}_i$ belongs to the positive (negative) class. Once the scatter matrices $\mathbf{D}$ and $\mathbf{T}$ are constructed from the samples in $\mathbf{X}$ (using the class information provided by $\mathbf{y}$), one then finds the vector $\mathbf{v} = [v_1, v_2, \ldots, v_n]^T$ that maximizes Equation 6. The optimal $\mathbf{v}$ is an eigenvector of the corresponding GEP $\mathbf{D}\mathbf{v} = \lambda\mathbf{T}\mathbf{v}$ and is normal (or orthogonal) to the discriminant hyperplane separating classes $\mathcal{C}_+$ and class $\mathcal{C}_-$. For LDA classification, only this 1 solution—the eigenpair (λ,$\mathbf{v}$) associated with the maximal eigenvalue—is of interest.

2.3 | Penalized eigendecompositions

In CU or DA applications, one naturally has 2 classes $\mathcal{C}_P$ and $\mathcal{C}_S$—the classes whose labels are associated with the primary and secondary samples, respectively. Unlike the LDA scenario, however, we want to *minimize domain scatter* while maximizing total scatter. In the eigendecomposition context, we want to avoid confusion with LDA-type class separation since samples could have class associations other than the

domain associations, ie, within the primary or secondary sets, samples can have other class associations that one would want to separate. For example, scatter component analysis (SCA) combines both class separation and domain scatter for object and image recognition[39]:

$$\max\left\{\frac{\text{Total Scatter} + \text{Between-Class Scatter}}{\text{Domain Scatter} + \text{Within-Class Scatter}}\right\}. \quad (9)$$

However, in this paper where regression largely dominates the chemometrics setting, we ignore the terms associated with class separation and instead solve

$$\max\left\{\frac{\text{Total Scatter}}{\text{Domain Scatter}}\right\} = \max_{\mathbf{v}} \frac{\mathbf{v}^T\mathbf{T}\mathbf{v}}{\mathbf{v}^T\mathbf{D}\mathbf{v}}. \quad (10)$$

In this paper, the matrices $\mathbf{T}$ and $\mathbf{D}$ will heretofore indicate total scatter and domain scatter, respectively. The methods that we explore next come from the DA literature, but they can readily be recast into a CU framework. In these methods, the data matrix consists of both primary and secondary spectra stacked on top of each other where $\mathbf{X} = [\mathbf{X}^{(\mathrm{P})};\mathbf{X}^{(\mathrm{S})}]$. Recall that $m = m_{\mathrm{P}} + m_{\mathrm{S}}$, where m_{P} and m_{S} are the number of samples in the primary and secondary sets.

2.3.1 | Total scatter

Most DA algorithms frame total scatter as principal component analysis: find an orthogonal transformation matrix $\mathbf{V}_k$ such that variance is maximized and the eigenvectors (loading vectors using CU nomenclature) have unit length

$$\max_{\mathbf{v}} \mathbf{v}^T\mathbf{T}\mathbf{v} \quad \text{subject to} \quad \mathbf{v}^T\mathbf{v} = 1 \quad (11)$$

across k eigenpairs $(\lambda_i, \mathbf{v}_i)$, $i = 1, \ldots, k$. The matrix $\mathbf{T}$ is proportional to the covariance matrix of $\mathbf{X}$:

$$\mathbf{T} = \mathbf{X}_c^T\mathbf{X}_c \quad \text{where} \quad \mathbf{X}_c = \mathbf{H}\mathbf{X}, \quad \mathbf{H} = \mathbf{I}_m - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^T. \quad (12)$$

The matrix $\mathbf{H}$ is referred to as the centering matrix and is symmetric and idempotent, ie, $\mathbf{H}^T = \mathbf{H}$ and $\mathbf{H}^2 = \mathbf{H}$. Maximizing total scatter in DA algorithms effectively involves the following expression for $\mathbf{T}$ in Equation 10:

$$\mathbf{T} = \mathbf{X}_c^T\mathbf{X}_c = (\mathbf{H}\mathbf{X})^T(\mathbf{H}\mathbf{X}) = \mathbf{X}^T\mathbf{H}\mathbf{X}. \quad (13)$$

2.3.2 | Domain scatter

In LDA, we seek a projection $\mathbf{v}$ that pushes apart samples from the positive and negative classes. In CU applications, we seek projections $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$ that pull samples together from different domain classes, ie, the primary and secondary samples. Hence, we want to minimize the inner product $\mathbf{v}^T\mathbf{D}\mathbf{v}$ associated with domain scatter:

$$\mathbf{D} = (\boldsymbol{\mu}^{(\mathrm{P})} - \boldsymbol{\mu}^{(\mathrm{S})})(\boldsymbol{\mu}^{(\mathrm{P})} - \boldsymbol{\mu}^{(\mathrm{S})})^T. \quad (14)$$

Borrowing notation from the DA literature, the difference between the mean vectors $\boldsymbol{\mu}^{(\mathrm{P})}$ and $\boldsymbol{\mu}^{(\mathrm{S})}$ is often expressed in terms of the data matrix $\mathbf{X}$:

$$\mathbf{e}^{(\mathrm{P})} = \tfrac{1}{m_{\mathrm{P}}}\mathbf{1}_{m_{\mathrm{P}}}, \quad \mathbf{e}^{(\mathrm{S})} = \tfrac{1}{m_{\mathrm{S}}}\mathbf{1}_{m_{\mathrm{S}}}, \quad \mathbf{e} = \begin{bmatrix}\mathbf{e}^{(\mathrm{P})} \\ -\mathbf{e}^{(\mathrm{S})}\end{bmatrix}, \quad \mathbf{L} = \mathbf{e}\mathbf{e}^T$$
$$\boldsymbol{\mu}^{(\mathrm{P})} = (\mathbf{X}^{(\mathrm{P})})^T\mathbf{e}^{(\mathrm{P})}, \quad \boldsymbol{\mu}^{(\mathrm{S})} = (\mathbf{X}^{(\mathrm{S})})^T\mathbf{e}^{(\mathrm{S})}, \quad \boldsymbol{\mu}^{(\mathrm{P})} - \boldsymbol{\mu}^{(\mathrm{S})} = \mathbf{X}^T\mathbf{e}. \quad (15)$$

The expression for the domain scatter matrix $\mathbf{D}$ in Equation 14 can now be more compactly expressed as

$$\mathbf{D} = (\boldsymbol{\mu}^{(\mathrm{P})} - \boldsymbol{\mu}^{(\mathrm{S})})(\boldsymbol{\mu}^{(\mathrm{P})} - \boldsymbol{\mu}^{(\mathrm{S})})^T = \mathbf{X}^T\mathbf{L}\mathbf{X}. \quad (16)$$

The structure of $\mathbf{D}$ in Equation 16 has consequences with respect to how one mean-centers the data. In many CU applications, one commonly mean-centers the primary and secondary samples separately:

$$\begin{aligned}\mathbf{X}_c^{(\mathrm{P})} &= \mathbf{X}^{(\mathrm{P})} - \mathbf{1}_{m_{\mathrm{P}}}(\boldsymbol{\mu}^{(\mathrm{P})})^T, & \mathbf{y}_c^{(\mathrm{P})} &= \mathbf{y}^{(\mathrm{P})} - \mathbf{1}_{m_{\mathrm{P}}}\bar{\mathbf{y}}^{(\mathrm{P})}, \\ \mathbf{X}_c^{(\mathrm{S})} &= \mathbf{X}^{(\mathrm{S})} - \mathbf{1}_{m_{\mathrm{S}}}(\boldsymbol{\mu}^{(\mathrm{S})})^T, & \mathbf{y}_c^{(\mathrm{S})} &= \mathbf{y}^{(\mathrm{S})} - \mathbf{1}_{m_{\mathrm{S}}}\bar{\mathbf{y}}^{(\mathrm{S})}.\end{aligned} \quad (17)$$

As a result, one can solve the following augmented linear system of equations for the vector $\mathbf{b}$:

$$\begin{bmatrix}\mathbf{X}_c^{(\mathrm{P})} \\ \tau\mathbf{X}_c^{(\mathrm{S})}\end{bmatrix}\mathbf{b} = \begin{bmatrix}\mathbf{y}_c^{(\mathrm{P})} \\ \tau\mathbf{y}_c^{(\mathrm{S})}\end{bmatrix}. \quad (18)$$

This linear system has been previously proposed for model updating in CU applications[14,40,41] and is well known as generalized Tikhonov regularization in the numerical analysis literature.[42,43]

In Equation 18, the reweighting is performed on the mean-centered secondary samples. That is, model updating is largely achieved by first moving the centroids of the primary and secondary spectra (via separate mean centering) independently to the origin and then reweighting the secondary samples to have the same commensurate scale as the primary samples. However, in the context of DA applications, if $\mathbf{X} = [\mathbf{X}_c^{(\mathrm{P})};\mathbf{X}_c^{(\mathrm{S})}]$, then the domain scatter matrix in Equation 16 would trivially yield $\mathbf{X}^T\mathbf{L}\mathbf{X} = \mathbf{0}$. Hence, local mean centering (LMC) cannot be coupled with the minimization of domain scatter in DA applications. Instead, we globally mean-center $\mathbf{X}$ via the centering matrix $\mathbf{H}$ such that $\mathbf{X}_c = \mathbf{H}\mathbf{X}$ where $\mathbf{H}$ is defined in Equation 12.

At this stage, it will be convenient to introduce 2 acronyms: LMC and GMC. Local mean centering is already defined in Equations 17 and 18, where $\{\mathbf{X}^{(\mathrm{P})}, \mathbf{y}^{(\mathrm{P})}\}$ and $\{\mathbf{X}^{(\mathrm{S})}, \mathbf{y}^{(\mathrm{S})}\}$ are separately mean-centered by their respective means. Global mean centering (GMC), on the other hand, is the same as LMC except that the spectral and reference means for both the primary and secondary samples are the same,

$$\begin{aligned}\mathbf{X}_c^{(\mathrm{P})} &= \mathbf{X}^{(\mathrm{P})} - \mathbf{1}_m\boldsymbol{\mu}^T, & \mathbf{y}_c^{(\mathrm{P})} &= \mathbf{y}^{(\mathrm{P})} - \mathbf{1}_m\bar{\mathbf{y}}, \\ \mathbf{X}_c^{(\mathrm{S})} &= \mathbf{X}^{(\mathrm{S})} - \mathbf{1}_m\boldsymbol{\mu}^T, & \mathbf{y}_c^{(\mathrm{S})} &= \mathbf{y}^{(\mathrm{S})} - \mathbf{1}_m\bar{\mathbf{y}},\end{aligned} \quad (19)$$

where

$$\boldsymbol{\mu} = \frac{1}{m}\mathbf{X}^T\mathbf{1}_m \quad \text{and} \quad \bar{\mathbf{y}} = \frac{1}{m}\mathbf{y}^T\mathbf{1}_m \quad (20)$$

are derived from the pooled primary and secondary samples $\mathbf{X} = [\mathbf{X}^{(\mathrm{P})};\mathbf{X}^{(\mathrm{S})}]$ and $\mathbf{y} = [\mathbf{y}^{(\mathrm{P})};\mathbf{y}^{(\mathrm{S})}]$. For both LMC and GMC,

we solve the same augmented linear system in Equation 18 for the vector **b** using partial least squares (PLS). Both LMC and GMC are the CU methods that we will compare against the DA methods to be defined later on (namely, transfer component analysis [TCA] and SCA).

An open question remains though: Does minimizing domain scatter on globally mean-centered data outperform LMC and reweighting where separate means are used for the primary and secondary spectra? The answer to this question is one of the main thrusts of this paper and is a question that is rarely addressed in the DA literature. It is important to note that if one is uncertain about the domain membership or labeling of a novel spectrum as being primary or secondary, then LMC is not feasible.

2.3.3 | Kernelized Rayleigh quotients

Most DA applications use kernel formulations of the Rayleigh quotient where the vector **v** in Equation 10 is expressed as a linear combination of the spectra

$$\mathbf{v} = \mathbf{X}^T\mathbf{u} \qquad \text{where} \qquad \mathbf{u} = [u_1, u_2, \ldots, u_m]^T. \tag{21}$$

In the optimization literature, the variables **v** and **u** are referred to as the primal and dual variables, respectively. As a result, if we substitute Equation 21 into Equation 10, then we obtain a Rayleigh quotient in terms of the dual variables:

$$\max_{\mathbf{u}} \mathcal{R}(\mathbf{u}) \quad \text{where} \quad \mathcal{R}(\mathbf{u}) = \frac{\mathbf{u}^T\mathbf{T}\mathbf{u}}{\mathbf{u}^T\mathbf{D}\mathbf{u}} \quad \text{and} \quad \begin{cases} \mathbf{T} = \mathbf{K}\mathbf{H}\mathbf{K} \\ \mathbf{D} = \mathbf{K}\mathbf{L}\mathbf{K} \\ \mathbf{K} = \mathbf{X}\mathbf{X}^T \end{cases}. \tag{22}$$

The matrix $\mathbf{K} = \mathbf{X}\mathbf{X}^T$ is called the kernel matrix. In actuality, **K** is a *linear* kernel matrix. In the entire discussion that follows, one can replace the linear kernel matrix with a variety of nonlinear kernel matrices that are commonly used in the DA literature. However, this paper will use linear kernels since the number of samples in most chemometrics data sets is typically not large enough to justify the use of nonlinear methods.

The GEP associated with the maximization of the Rayleigh quotient in Equation 22 is formed by setting the corresponding Lagrangian function $\mathscr{L}(\mathbf{u})$ (as defined in Equation 3) to 0. The resulting GEP is

$$\mathbf{T}\mathbf{U}_k = \mathbf{D}\mathbf{U}_k\mathbf{\Lambda}_k. \tag{23}$$

The eigenvector in the primal space can then be recovered via the relation $\mathbf{v}_i = \mathbf{X}^T\mathbf{u}_i$, $i = 1, \ldots, k$.

3 | DA-BASED EIGENDECOMPOSITION METHODS

In this section, we will discuss 3 DA methods,

- TCA,[44]
- SCA,[39] and
- primal SCA (PSCA),

which use penalty terms or constraints to create penalized variants of GEPs. In the discussion to follow, the total scatter matrix **T** and domain scatter matrix **D** in TCA and SCA are defined using the dual variables $\mathbf{U}_k = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_k]^T$ and the linear kernel matrix **K** as in Equation 22. Primal SCA, on the other hand, uses the primal counterparts of the total and scatter matrices as defined in Equations 13 and 16.

TCA and SCA follow the same basic protocol for obtaining the eigenvector matrix $\mathbf{U}_k$: solve the GEP across k eigenpairs $\{(\lambda_i, \mathbf{u}_i)\}_{i=1}^k$ associated with maximizing its corresponding Rayleigh quotient $\mathcal{R}(\mathbf{u})$. After obtaining $\mathbf{U}_k$, the primal variables $\mathbf{V}_k$ are recovered via the relation $\mathbf{V}_k = \mathbf{X}^T\mathbf{U}_k$ and $\mathbf{V}_k$ serves as the transformation matrix that maps the n-dimensional calibration spectra onto a lower k-dimensional representation

$$\mathbf{Z}_k = \mathbf{K}\mathbf{U}_k = \mathbf{X}\mathbf{V}_k, \tag{24}$$

where domain scatter is minimized. In the case of PSCA, obtaining $\mathbf{V}_k$ is more direct: obtain the eigenpairs $\{(\lambda_i, \mathbf{v}_i)\}_{i=1}^k$ by maximizing $\mathcal{R}(\mathbf{v})$ and use $\mathbf{V}_k = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k]^T$ to map the spectra **X** onto a lower dimension via $\mathbf{Z} = \mathbf{X}\mathbf{V}_k$.

Once the dimension-reduced data set $\mathbf{Z}_k$ is obtained, it can be fed into any multivariate calibration algorithm to construct a regression model for subsequent prediction on novel secondary spectra $\bar{\mathbf{X}}^{(S)}$ (which will be similarly dimension reduced via $\bar{\mathbf{Z}}_k = \bar{\mathbf{X}}^{(S)}\mathbf{V}_k$). In this paper, PLS is used as the multivariate calibration method on $\mathbf{Z}_k$, and all k PLS latent vectors will be used. Numerically, since $\mathbf{Z}_k$ has k columns (and presumably not rank deficient), then a subsequent PLS regression on $\{\mathbf{Z}_k, \mathbf{y}\}$ using *all* k latent vectors is theoretically equivalent to ordinary least squares regression on $\{\mathbf{Z}_k, \mathbf{y}\}$. We want to distinguish between (*a*) PLS using fewer than k latent vectors and (*b*) PLS using all k latent vectors (equivalent to ordinary least squares). Using option (*a*) would entail a further reduction to k' dimensions ($k' < k$), and as a result, 3 parameters would have to be tuned (τ; k, the number of eigenvectors kept from the eigendecomposition; and k', the number of latent vectors associated with PLS regression). Using option (*b*), the option we use here, maintains consistency across both the CU and DA methods. That is, the only 2 parameters of interest are τ (the penalty parameter) and k (the number of dimensions in the reduced subspace). In the case of the DA-based eigendecomposition, k is the number of eigenvectors kept in the transformation matrix $\mathbf{V}_k$. In the case of the CU-based linear system of Equation 18, k is the number of PLS latent vectors.

3.1 | Transfer component analysis

TCA was originally proposed for DA applications in Wi-Fi location and text classification.[44] The Wi-Fi data contain some labeled Wi-Fi data collected in period A (the primary domain) and a large amount of unlabeled Wi-Fi data collected in period B (the secondary domain). In the text classification

scenario, business text data set is categorized to a hierarchical structure. Data from different subcategories under the same parent category are considered to be from different but related domains. The task is to predict the labels of the parent category.

The maximization of the TCA-based Rayleigh quotient $\mathcal{R}(\mathbf{u})$

$$\max_{\mathbf{u}} \mathcal{R}(\mathbf{u}): \quad \mathcal{R}(\mathbf{u}) = \frac{\mathbf{u}^T\mathbf{T}\mathbf{u}}{\mathbf{u}^T\mathbf{D}\mathbf{u} + \tau\mathbf{u}^T\mathbf{u}} = \frac{\mathbf{u}^T\mathbf{T}\mathbf{u}}{\mathbf{u}^T(\mathbf{D} + \tau\mathbf{I})\mathbf{u}} \quad (25)$$

results in the following GEP:

$$\mathbf{T}\mathbf{U}_k = (\mathbf{D} + \tau\mathbf{I})\mathbf{U}_k\mathbf{\Lambda}_k. \quad (26)$$

The penalty term $\tau\mathbf{u}^T\mathbf{u} = \tau\,||\mathbf{u}||_2^2$ in Equations 25 and 26 mitigates the rank deficiency of the domain scatter matrix $\mathbf{D}$.

3.2 | Scatter component analysis

The SCA is another eigendecomposition approach developed for image classification and object recognition.[39] However, the optimization of its Rayleigh quotient

$$\max_{\mathbf{u}} \mathcal{R}(\mathbf{u}): \quad \mathcal{R}(\mathbf{u}) = \frac{\mathbf{u}^T\mathbf{T}\mathbf{u}}{\mathbf{u}^T\tau\mathbf{D}\mathbf{u}} \quad \text{subject to} \quad \mathbf{u}^T\mathbf{K}\mathbf{u} = 1 \quad (27)$$

involves a hard constraint on the eigenvectors $\mathbf{u}$ instead of the soft constraint via a penalty term used by TCA. The SCA also uses an additional parameter τ that controls the trade-off between total and domain scatter. The corresponding SCA-based GEP becomes

$$\mathbf{T}\mathbf{U}_k = (\tau\mathbf{D} + \mathbf{K})\mathbf{U}_k\mathbf{\Lambda}_k. \quad (28)$$

Note that $\mathbf{u}^T\mathbf{K}\mathbf{u} = 1$ in Equation 27 actually imposes a unit length constraint on the primal vector $\mathbf{v} = \mathbf{X}^T\mathbf{u}$ since $\mathbf{u}^T\mathbf{K}\mathbf{u} = \mathbf{u}^T\mathbf{X}\mathbf{X}^T\mathbf{u} = \mathbf{v}^T\mathbf{v} = ||\mathbf{v}||_2^2 = 1$, whereas TCA in Equation 25 actually penalizes the size of dual variables where $\mathbf{u}^T\mathbf{u} = ||\mathbf{u}||_2$.

Primal SCA: The observation that SCA imposes a unit length constraint on the primal eigenvectors gives rise to the primal formulation of SCA. (This was never explored in the DA literature since nonlinear kernels were only of interest.) If we rewrite Equation 27 in terms of the primal vectors $\mathbf{v}$ and the primal scatter and domain matrices using Equations 13 and 16, we obtain

$$\begin{aligned}\mathcal{R}(\mathbf{u}) &= \frac{\mathbf{u}^T\mathbf{T}\mathbf{u}}{\mathbf{u}^T\tau\mathbf{D}\mathbf{u}} = \frac{\mathbf{u}^T(\mathbf{X}\mathbf{X}^T\mathbf{H}\mathbf{X}\mathbf{X}^T)\mathbf{u}}{\mathbf{u}^T\tau(\mathbf{X}\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{X}^T)\mathbf{u}} \\ &= \frac{\mathbf{v}^T(\mathbf{X}^T\mathbf{H}\mathbf{X})\mathbf{v}}{\mathbf{v}^T\tau(\mathbf{X}^T\mathbf{L}\mathbf{X})\mathbf{v}} = \mathcal{R}(\mathbf{v}).\end{aligned} \quad (29)$$

Although maximizing $\mathcal{R}(\mathbf{u})$ and $\mathcal{R}(\mathbf{v})$ superficially appears to be the same, numerically, the dual and primal maximizations can be quite different, especially if the data matrix $\mathbf{X} = [\mathbf{X}^{(P)};\mathbf{X}^{(S)}]$ is quite ill conditioned, e.g., the spectra are highly collinear. The GEP associated with PSCA becomes

$$\mathbf{T}\mathbf{V}_k = (\tau\mathbf{D} + \mathbf{I})\mathbf{V}_k\mathbf{\Lambda}_k. \quad (30)$$

4 | DATA SETS

We explore 4 spectroscopic data sets for purposes of model updating. Each data set is divided into 3 subsets: a *calibration set* consisting of primary samples $\{\mathbf{X}^{(P)}, \mathbf{y}^{(P)}\}$ and a small number of secondary samples $\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$, a *validation set* containing a small number of secondary samples $\{\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val}\}$, and a test set of secondary samples $\{\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test}\}$. We use the validation set for model selection, ie, the selection of the penalty parameter τ and number of latent vectors k. This topic will be discussed in greater detail in Section 5.2.

We intentionally restrict the number of secondary samples $\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$ in the calibration set to be small relative to the number of secondary samples in the test set $\{\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test}\}$. Although not all sources of variation and information can probed with such a small sample size, we do want to reflect the reality that reference values are often difficult and/or expensive to obtain. Plus, having a large pool of labeled secondary samples in the calibration set obviates the need for using CU or DA strategies for modeling updating in the first place.

For each data set, the calibration, validation, and test sets describe just 1 partition or split of the data. Results from 1 partition are often just anecdotal. Instead, we will examine 250 additional random splits of the data. In this way, we can see how robust a given method is against sample perturbations in the data, and how large the performance spread is. The reshuffling protocol is described in Section 4.5.

4.1 | Tablet instrument

The tablet instrument data set consists of near-infrared (NIR) spectra obtained from pharmaceutical tablets from 2 spectrometers.[45] All 650 wavelengths between the spectral region of 1100 to 1898 nm are used. There is a designated split of the data whereby the calibration, validation, and test sets contain 155, 40, and 460 samples, respectively. Each sample is measured on both spectrometers. The primary and secondary samples correspond to the samples drawn from the first and second spectrometers, respectively. For every tablet, 3 response variables are measured: weight, hardness, and the amount of active ingredient (nominally 200 mg/tablet). We use the third response variable: typically, one wants to estimate the amount of active ingredient from NIR spectra.

The primary calibration samples $\{\mathbf{X}^{(P)}, \mathbf{y}^{(P)}\}$ consist of the 155 samples measured on the first spectrometer in the calibration set. Since the designated number of samples in the validation set is 40, we restrict the number of secondary samples $\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$ in the calibration set to be 40 (of 155) as well. Indices 1, 2, … ,40 from the second spectrometer were used from the calibration set. (No attempt was made to judiciously select a few representative samples, as is often the

case with methods that use a standardization set). The validation set $\{\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val}\}$ consists of all 40 samples measured on the second spectrometer in the designated validation set. The test set $\{\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test}\}$ consist of all 460 samples from the second spectrometer.

4.2 | Corn instrument

The corn instrument data set consists of NIR spectra measured from 3 instruments labeled m5, mp5, and mp6.[46] Each sample was measured on all 3 instruments. Only instruments m5 and mp5 are used here. Model updating was performed using all 700 wavelengths between the spectral region of 1100 to 2498 nm. The primary and secondary samples are drawn from the m5 and mp5 instruments, respectively. For every sample, 4 response variables were measured: moisture, oil, protein, and starch. We used moisture as the response variable.

Since there is no default or designated split of the data into calibration, validation, and test sets, one was created. The calibration set consists of the first 40 samples measured on instrument m5 (indices 1,2, … ,40)—the primary calibration set $\{\mathbf{X}^{(P)}, \mathbf{y}^{(P)}\}$—and the first 4 samples measured on instrument mp5 (indices 1,2,3,4)—the secondary calibration set $\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$. The validation set $\{\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val}\}$ contains 4 samples (indices 41 through 44) measured on instrument mp5. The test set $\{\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test}\}$ contains 36 samples (indices 45 through 80) from instrument mp5.

4.3 | Tablet batch

The tablet batch data set consists of NIR spectra measured across 4 different dose types (tablets of 5, 10, 15, and 20 mg) of a pharmaceutical drug.[47] All 404 wavenumbers between the spectral region of 7400 to 10 507 cm^{-1} are used. There are 310 samples: 31 batches and 10 tablets per batch. The batches can be classified into 3 production scales: laboratory, pilot, and full. In this paper, we restrict our attention to 2 dose types (tablets of 5 and 10 mg) and to 2 batch scales (laboratory and full). The primary and secondary samples are drawn from the laboratory and full batches, respectively. As a result, there are 60 primary samples and 60 secondary samples. Unlike the tablet instrument and corn instrument data sets, none of the samples were measured on multiple instruments, ie, there is no possibility to create a standardization set.

As in the corn instrument data set, there is no default or designated split of the data. The primary calibration samples $\{\mathbf{X}^{(P)}, \mathbf{y}^{(P)}\}$ consist of 60 samples from the laboratory batches: 30 samples from type 1 (5 mg) and 30 samples from type 2 (10 mg). The 60 secondary samples from the full batches were split into 6 calibration samples $\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$ (3 samples from type 1 and 3 samples from type 2), 6 validation samples $\{\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val}\}$ (3 samples from type 1 and 3 samples from type 2), and 48 test set samples $\{\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test}\}$ (24 samples from type 1 and 24 samples from type 2).

4.4 | Wheat kernel

The wheat data set consists of NIR transmittance spectra.[48] The calibration set comprises 415 wheat kernels samples representing 43 varieties or variety mixtures from 2 different locations in Denmark. The test set consists of 108 samples representing 11 varieties from 1 location. (The test samples were actually acquired from the calibration samples, but these samples were stored for 2 additional months before measurement so as to provide a check for temporal drift in the samples and instrumentation). All 100 wavelengths between the spectral region of 850 to 1048 nm are used. The reference values correspond to protein content percentage. The primary and secondary samples are drawn from the 415 calibration and 108 test samples, respectively.

The primary calibration set $\{\mathbf{X}^{(P)}, \mathbf{y}^{(P)}\}$ contains the 415 wheat kernels in the calibration set. The partitioning of the 108 test set samples into 3 secondary sample sets ($\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$, $\{\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val}\}$, and $\{\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test}\}$) is governed by the fact that the test set reference values are already sorted in ascending order. As a result, the subset partitioning will be staggered. The secondary calibration set $\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$ consists of 11 samples (samples 1, 11, 21, … ,91, 101) of the 108 test set samples. The secondary validation set $\{\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val}\}$ similarly consists of 11 samples (samples 2, 12, 22, … ,92, 102). The secondary test set $\{\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test}\}$ consists of the remaining samples.

4.5 | Random splits

In addition to the designated split just outlined for each data set, 250 additional random splits of the data will be generated. To facilitate the subsequent discussion of data splitting, Table 1 gives the number of samples in the calibration, validation, and test sets across each data set.

For the tablet instrument data set, all of the samples—separately for each spectrometer—will be pooled together. The samples for each spectrometer will then be reshuffled. For the first spectrometer, samples 1 through 155 will be assigned to the primary calibration set $\{\mathbf{X}^{(P)}, \mathbf{y}^{(P)}\}$. For the second spectrometer, samples 1 through 40, samples 156 through 195, and samples 196 through 655 will be assigned to $\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$, $\{\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val}\}$, and $\{\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test}\}$, respectively. For the corn instrument data set, all of the samples—separately for instrument m5 and mp5—will be pooled together. The samples for each instrument will then be separately reshuffled. For instrument m5, samples 1 through 40 will be assigned to the primary calibration set

TABLE 1 Number of samples in calibration, validation, and test sets

		Tablet Instrument	Corn Instrument	Tablet Batch	Wheat Kernel
Calibration set	$(\mathbf{X}^{(P)}, \mathbf{y}^{(P)})$	155	40	60	415
	$(\mathbf{X}^{(S)}, \mathbf{y}^{(S)})$	40	4	6	11
Validation set	$(\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val})$	40	4	6	11
Test set	$(\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test})$	460	36	48	86

$\{\mathbf{X}^{(P)}, \mathbf{y}^{(P)}\}$. For instrument mp5, samples 1 through 4, samples 41 through 44, and samples 45 through 80 will be assigned to $\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$, $\{\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val}\}$, and $\{\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test}\}$, respectively.

For the tablet batch and wheat kernel data sets, the primary and secondary samples will be separately reshuffled. For the primary samples, the first 80% of the samples (samples 1 through 48 for tablet batch and samples 1 through 332 for wheat kernel) will be assigned to $\{\mathbf{X}^{(P)}, \mathbf{y}^{(P)}\}$. As in the tablet instrument and corn instrument data sets, we want each split to contain a different subset of $\{\mathbf{X}^{(P)}, \mathbf{y}^{(P)}\}$. For the secondary samples, the first 10% (samples 1 through 6 for tablet batch and samples 1 through 11 for wheat kernel), the second 10% (samples 7 through 12 for tablet batch and samples 12 through 22 for wheat kernel), and the remaining 80% of the samples will be assigned to $\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$, $\{\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val}\}$ and $\{\mathbf{X}^{(S)}_{test}, \mathbf{y}^{(S)}_{test}\}$, respectively.

Each random split preserves the proportion of primary and secondary samples. In the case of the tablet batch data set, the proportion of 5- and 10-mg sample types are also preserved.

5 | METHODS AND MODEL SELECTION

5.1 | Regression methods

In examining each data set, 6 regression methods will be compared against other: primary predicting secondary (PPS), LMC, GMC, PSCA, dual TCA (DTCA), and dual scatter component analysis (DSCA). The PPS is simply the prediction on $\{\mathbf{X}^{(S)}, \mathbf{y}^{(S)}\}$ by a PLS model built solely from primary calibration samples $\{\mathbf{X}^{(P)}, \mathbf{y}^{(P)}\}$. LMC and GMC are defined in Section 2.3.2. Using the spectral and reference means defined in Equations 15 and 20, the prediction on a novel secondary spectrum $\mathbf{x}^{(S)}$ for PPS, LMC, and GMC is expressed as $\hat{\mathbf{y}} = (\mathbf{x}^{(S)} - \boldsymbol{\mu}^{(P)})^T\mathbf{b} + \bar{\mathbf{y}}^{(P)}$, $\hat{\mathbf{y}} = (\mathbf{x}^{(S)} - \boldsymbol{\mu}^{(S)})^T\mathbf{b} + \bar{\mathbf{y}}^{(S)}$, and $\hat{\mathbf{y}} = (\mathbf{x}^{(S)} - \boldsymbol{\mu})^T\mathbf{b} + \bar{\mathbf{y}}$, respectively.

In summary, the penalized eigendecomposition methods PSCA, TCA, and SCA use the globally centered data matrices $\mathbf{X}^{(P)}_c$ and $\mathbf{X}^{(S)}_c$ in Equation 20 to solve a GEP and subsequently obtain a transformation matrix of eigenvectors $\mathbf{V}_k$. The matrix $\mathbf{V}_k$ then dimension-reduces $\mathbf{X}_c = [\mathbf{X}^{(P)}_c; \mathbf{X}^{(S)}_c]$ via $\mathbf{Z}_k = \mathbf{X}_c\mathbf{V}_k$, where $\mathbf{Z}_k$ and $\mathbf{y}_c$ are subsequently fed into a multivariate calibration method (in this case, PLS) to obtain a vector $\mathbf{b}$. As in the GMC case, prediction on a novel secondary spectrum $\mathbf{x}^{(S)}$ is given by $\hat{\mathbf{y}} = (\mathbf{x}^{(S)} - \boldsymbol{\mu})^T\mathbf{b} + \bar{\mathbf{y}}$.

5.2 | Model selection

All of the methods involve 2 tuning parameters: τ (the penalty parameter) and k (the number of latent vectors). Following the guidelines set out by Hansen,[42] the τ values are chosen in an exponentially decaying fashion where the minimal and maximal τ values (τ_{min} and τ_{max}) are the smallest and largest singular values of the coefficient matrix. For example, in Equation 18, the coefficient matrix is $[\mathbf{X}^{(P)}_c; \mathbf{X}^{(S)}_c]$. Sixty τ values of are used such that $\tau_1 > \tau_2 > \ldots > \tau_{60}$. The number of latent vectors k ranges at 1, 2, … ,50. Hence, there are $N = (60)(50) = 3000$ possible solutions.

We use a fusion rule to combine similarity rankings to select the tuning parameters.[49] Here, the term fusion indicates the combination of many model quality measures to produce a final ranking of models. A number of bias and variance model quality measures are evaluated with the fusion rule so as to determine a good updated calibration model. For this study, bias refers to model prediction error, and variance corresponds to prediction uncertainty. The goal of the model quality measures is to assess the degree of overfitting (lower bias but greater variance or model complexity) versus underfitting (greater bias but less variance or model complexity). Thus, the task of the fusion rule is to identify those models with an acceptable bias/variance trade-off.

Seven model quality measures are computed on the validation set $\{\mathbf{X}^{(S)}_{val}, \mathbf{y}^{(S)}_{val}\}$. First, we examine a bias error known as the root-mean-square error

$$\text{RMSE}_i = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}, \quad i = 1, \ldots, N. \tag{31}$$

The corresponding R^2, slope, and y-intercept obtained from plotting predicted values against reference values are also evaluated. We also want to measure model complexity. For this task, we measure the 2 norm $||\mathbf{b}||_2$ and jaggedness

$$J_i = \sqrt{\sum_{j=2}^{n}\left(b_{(i,j)} - b_{(i,j-1)}\right)^2}, \quad i = 1, \ldots, N, \tag{32}$$

where $b_{(i,j)}$ is the jth regression coefficient of the ith model. Another quality measure used is the U-curve defined by the following:

$$U_i = \frac{||\mathbf{b}_i||_2 - ||\mathbf{b}||_2^{\min}}{||\mathbf{b}||_2^{\max} - ||\mathbf{b}||_2^{\min}} + \frac{\text{RMSE}_i - \text{RMSE}^{\min}}{\text{RMSE}^{\max} - \text{RMSE}^{\min}}, \; i = 1, \ldots, N, \tag{33}$$

where $||\mathbf{b}||_2^{\min}$ and $||\mathbf{b}||_2^{\max}$ indicate the minimal and maximal 2-norms across all N models (the same holds for $\text{RMSE}^{\min}$ and $\text{RMSE}^{\min}$).

All dissimilarity measures are converted into similarity measures, e.g., R^2 becomes $1 - R^2$. Hence, each vector $\mathbf{b}_i$ has 7 model quality measures $[q_{1i}, q_{2i}, \ldots, q_{7i}]^T$ associated with it, where

$$\mathbf{Q} = \begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1N} \\ q_{21} & q_{22} & \cdots & q_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ q_{71} & q_{72} & \cdots & q_{7N} \end{bmatrix} \tag{34}$$

is a matrix of all model quality measures. For a given model quality metric (ie, for each row of **Q**), the model measures are ranked from best to worst. Each matrix entry q_{ij} in Equation 34 is assigned a rank r_{ij}, and a matrix of rankings is generated:

$$\begin{bmatrix} r_{11} & r_{12} & \dots & r_N \\ r_{21} & r_{22} & \dots & r_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ r_{71} & r_{72} & \dots & r_{7N} \end{bmatrix}. \quad (35)$$

The model sums $[S_1, S_2, \dots, S_N]$, where $S_i = \sum_{k=1}^{7} r_{\mathrm{ki}}$ are computed and the model vector with the lowest sum is deemed the overall best model.

6 | RESULTS

The results are categorized into 2 calibration updating scenarios: calibration transfer and calibration maintenance. For the tablet instrument and corn instrument data sets, the aim is to transfer a model developed from 1 instrument and transfer it to another instrument. The other data sets are more indicative of a calibration maintenance scenario. In the tablet batch data set, one builds a model off of the samples associated with the pilot batch. The aim is then to transfer this model to batches associated with full production. In the wheat kernel data set, the validation data set consists of 108 calibration samples that have been stored for 2 additional months. During this time, the spectral measurements will likely have acquired new characteristics from sample aging, in addition to possibly acquiring spectral artifacts from instrument drift.

6.1 | Performance metrics

There are many ways to characterize the performance of various regression methods across the 4 data sets. We will examine 2 performance merits: the root-mean-square error of validation (RMSEV) and the R^2 obtained by plotting predicted test set values against reference test set values. In short, we use RMSEV and R^2 as proxies for prediction accuracy and precision, respectively. As described in Section 4.5, there are 251 splits (the default split plus 250 additional random splits) of the data, and we compute the RMSEV and R^2 values for each split. As a result, there will naturally be a spread in the values of these performance metrics.

The motivation for doing many data splits is to ascertain how robust each method is to sample perturbations in the data. In particular, we are also interested in comparing the performance of the default split with the performance of the random splits. Oftentimes, the default split is chosen on the basis of certain design-of-experiment criteria, or perhaps the grouping inherent in the default split is simply a consequence of chronological ordering, e.g., the spectral measurements of the calibration set occurred earlier than those of the validation set.

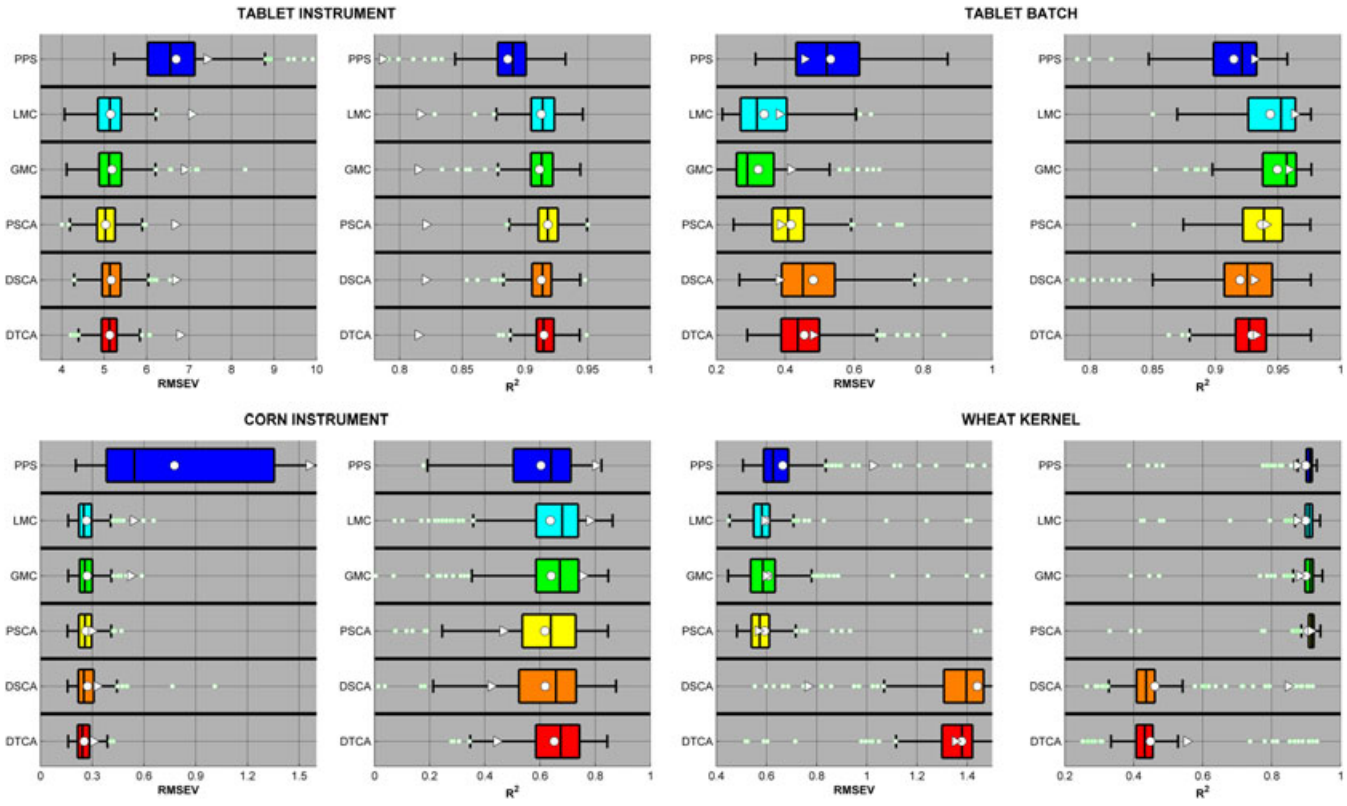

FIGURE 1 Boxplots for the root-mean-square error of validation (RMSEV) and R^2 values across 251 data splits for various regression methods and data sets. DSCA indicates dual scatter component analysis; DTCA, dual transfer component analysis; GMC, global mean centering; LMC, local mean centering; PPS, primary predicting secondary; PSCA, primal scatter component analysis

With the exception of the wheat kernel data set, the information detailing the sampling or collection protocol of the samples is missing. (This is often the case with most public domain data sets). For example, based upon prior experience with the corn instrument data set, we strongly suspect there is a temporal bias in this data. With very few exceptions, the prediction on the last n samples $(80 - n + 1, \ldots, 80)$ by a model built from the first $80 - n$ samples will be worse than the prediction on a random set of n samples from a model developed on the remaining samples. Hence, domain bias could possibly be confounded with the bias associated with sample acquisition and collection. We would like to measure the misfit in performance (if any) between the default split and the other 250 random splits. Recall that each random split preserves the proportion of primary and secondary samples, but the shuffling of the samples should eliminate any bias due to the sample acquisition in the default split.

Recall that we want to compare the model updating regression methods—LMC, GMC, PSCA, DSCA, and DTCA—against PPS (no model updating). Within the model updating methods, we want to compare the mean centering and sample reweighting CU schemes (LMC and GMC) against the newer DA-based penalized eigendecomposition methods (PSCA, DSCA, and DTCA). We want to answer the following question: Do these newer DA-based methods outperform established CU methods in terms of model updating?

6.2 | Performance results and spread

In Figure 1, a boxplot of the RMSEV and R^2 values is displayed for each regression method and for each data set. Each regression method has its own color. Overlaid on each boxplot are 2 additional white markers: a circle and a triangle. The white circle corresponds to the mean, and the white triangle corresponds to the performance value associated with the default split. Outliers are shown in small light green circles.

6.2.1 | Overall trends across regression methods

On average, all model updating methods outperform PPS (no updating). Within the model updating methods, LMC and GMC are noninferior to the eigendecomposition methods (PSCA, DSCA, and DTCA). The primal eigendecomposition method (PSCA) compares favorably with respect to LMC and GMC for the tablet instrument and wheat kernel data sets. For the tablet batch data set, PSCA performs on par with

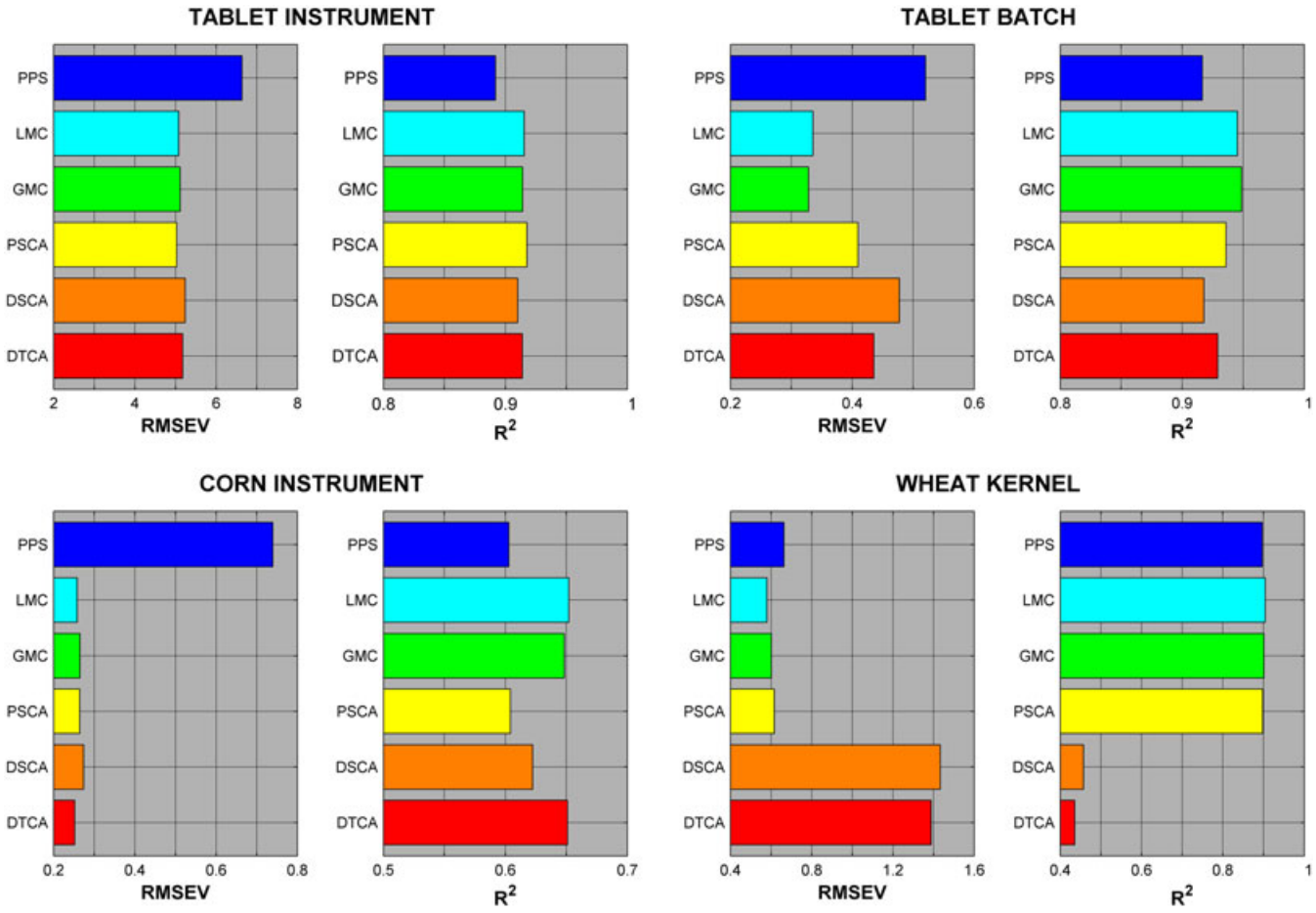

FIGURE 2 Mean root-mean-square error of validation (RMSEV) and R^2 values across 251 data splits for various regression methods and data sets. DSCA indicates dual scatter component analysis; DTCA, dual transfer component analysis; GMC, global mean centering; LMC, local mean centering; PPS, primary predicting secondary; PSCA, primal scatter component analysis

DSCA and DTCA, ie, worse than LMC and GMC. For the corn instrument data set, PSCA is slightly less precise (via the R^2 metric), on average, than GMC and LMC. Aside from the tablet batch data set, PSCA performs, on average, the same as (if not slightly better than) the CU-based methods LMC and GMC. A summary display of the mean values (the values associated with the white circles) is shown in Figure 2.

One possible explanation for the poor performance (e.g., tablet batch and wheat kernel) of the dual DA-based methods is numerical: both the total and domain scatter matrices **T** and

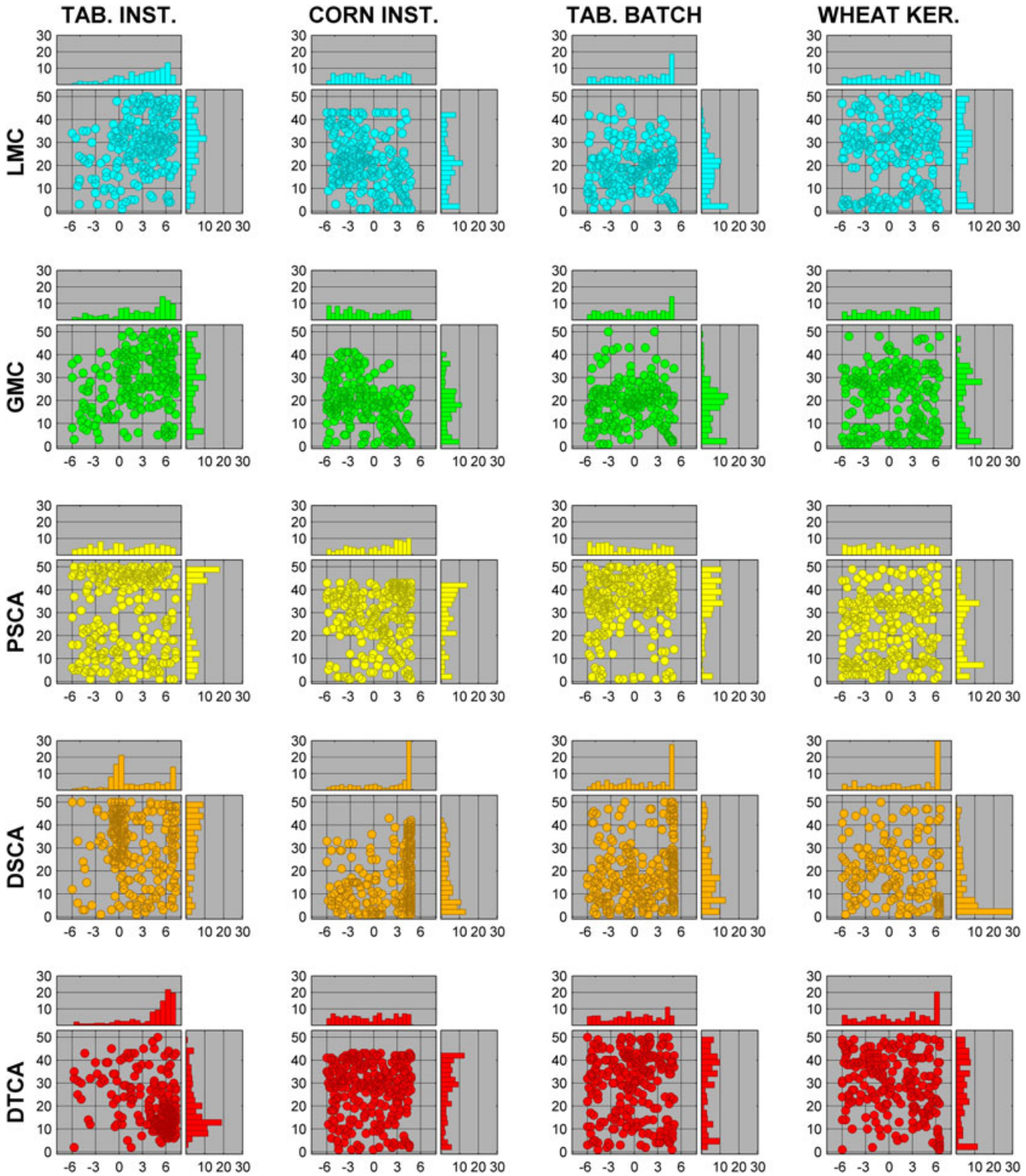

FIGURE 3 For each data set (column) and for each model updating method (row), a scatterplot and histogram of the tuning parameters ($\log_{10}$ and k) is shown across all data splits. In each scatterplot, the x-axis and y-axis correspond to the values of $\log_{10}(\tau)$ and k, respectively. Likewise, the top and right *percentage* histograms indicate the distribution of the $\log_{10}$ and k values, respectively. DSCA indicates dual scatter component analysis; DTCA, dual transfer component analysis; GMC, global mean centering; LMC, local mean centering; PSCA, primal scatter component analysis

D in Equation 22 use the kernel matrix $\mathbf{K} = \mathbf{X}\mathbf{X}^T$ as input (as opposed to **X** in the case of PSCA). The condition number of **K** is the square of that of **X**. In numerical analysis, the condition number is a measure of numerical instability associated with linear inversion and is bounded below by 1—the larger the condition number, the more numerically unstable the linear inversion. Hence, DSCA and DTCA are more prone to suffer from numerically instability, especially in highly collinear and low-sample-size-high-dimensional settings that characterize most chemometrics data sets.

6.2.2 | Outliers

There are a number of data splits in which performance is extremely poor across data sets. (For example, see the RMSEV results in the tablet instrument data set). With respect to the other methods, the presence of outliers tends to unsurprisingly skew the distribution of the performance metrics: slightly right skewed for RMSEV and left skewed for R^2. Moreover, in many instances, the performance value of the default split—as indicated by the white triangle—is likewise an outlier. This suggests that there could be a data set bias due to sample collection. In this respect, PSCA outperforms LMC and GMC. In the corn instrument and tablet batch data sets, PSCA appears to do a reasonable job in mitigating both domain and sample collection biases.

6.2.3 | Tuning parameters

We now examine the tuning parameters that were chosen by the fusion model selection process in Section 5.2. Recall that for each of the 251 data splits, 2 tuning parameters were chosen: τ, the penalty parameter, and k, the number of latent vectors used. As a result, we have 251 (τ,k) pairs, which can be displayed as coordinates. We are interested to see if certain tuning parameters are preferentially chosen. In Figure 3, a scatterplot of the values of $\log_{10}(\tau)$ and k is shown for each data set (column) and model updating method (row). Also, a percentage histogram indicating the density of each tuning parameter is shown: the top and right histograms correspond to $\log_{10}(\tau)$ and k densities, respectively.

Across data sets, there is no strong trend for any method to preferentially select certain tuning parameters within a given model updating method. For GMC and LMC, the distribution shape for the number of latent vectors k is fairly consistent within a data set. Moreover, the distribution shape for τ is mostly right skewed, indicating that a priority is placed upon reweighting the secondary samples. For the eigendecomposition methods, there is no consistent trend for τ distribution within a data set. Overall, the values of the tuning parameter (τ, k) pairs are quite diverse, indicating that parameter selection is highly sample dependent. Alternatively, this diversity may be explained by our attempt to simultaneously capture the variance/bias trade-off across 7 quality measures via the fusion rule. Capturing the trade-off can be self-conflicting. Using just 2 quality measures alone (e.g., minimizing Root Mean Square Error of Cross Validation (RMSECV) and maximizing R^2) would likely not be enough to achieve consensus among the models. Our rationale is to seek consensus across many quality measures rather than a few.

7 | CONCLUSION AND FUTURE WORK

Two model updating methods—both CU- and DA-based methods—were compared. Established CU-based methods that mean center and reweight the secondary samples are noninferior, on average, to DA-based penalized eigendecomposition methods. Within the CU-based strategies, there is no significant difference between LMC and GMC. As a result, one should opt for GMC since it uses a simpler mean-centering strategy. This is particularly important in cases where one does not know in advance (or has difficulty in assigning) the membership of a sample as being either primary or secondary in nature. Within the DA-based penalized eigendecomposition strategies, PSCA—the simplest of the eigendecomposition strategies—was easily the best performer. The dual eigendecomposition methods DTCA and DSCA are likely not warranted for small to medium data sets.

With the exception of 1 data set, PSCA performed on par with LMC and GMC and is a promising target for subsequent research. At the moment, PSCA is completely unsupervised in that it does not use the reference values $\mathbf{y}^{(P)}$ and $\mathbf{y}^{(S)}$ to construct the eigenvector matrix that transforms the spectra into a lower-dimensional subspace. (It does use domain labels, ie, samples are categorized into primary and secondary classes). Future work may incorporate reference values in the construction of the eigenvector matrix.

Although eigendecomposition methods have had success in DA applications, it is arguable whether these improvements have actually come from better algorithms, or from the vast troves of data available to computer vision researchers and the improvements in computing power that makes analyzing these massive data sets feasible. Although the CU methods presented here are discipline agnostic (they can be applied to CU problems in chemometrics and spectroscopy or to DA problems in computer vision, bioinformatics, etc), these methods are not part of the standard repertoire of most DA practitioners, and they probably should be. What makes them relevant to CU problems is also what makes them relevant to DA problems: these reweighting methods do not require a standardization set. Using a standardization set requires that the same samples be measured in both the primary and secondary conditions, and this has limited utility for both CU and DA applications. Moreover, from the perspective of Occam's razor, the CU-based methods are also simpler to implement and are computationally faster than most DA methods. These methods can also be easily "kernelized" in a nonlinear fashion, making them relevant for DA problems involving large data sets.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under grant no. CHE-1506417 and is gratefully acknowledged by the author.

REFERENCES

1. Osborne B, Fearn T. Collaborative evaluation of universal calibrations for the measurement of protein and moisture in flour by near-infrared reflectance. *Int J Food Sci Technol.* 1983;18(4):453–460.
2. Osborne B, Fearn T. Collaborative evaluation of near-infrared reflectance analysis for the determination of protein, moisture and hardness in wheat. *J Sci Food Agric.* 1983;34(9):1011–1017.
3. Shenk J, Westerhaus M, Templeton W. Calibration transfer between near-infrared reflectance spectrophotometers. *Crop Sci*. 1985;25(1): 159–161.
4. Wang Y, Veltkamp D, Kowalski B. Multivariate instrument standardization. *Anal Chem.* 1991;63(23):2750–2756.
5. Wang Y, Kowalski B. Temperature-compensating calibration transfer for near-infrared filter instruments. *Anal Chem.* 1993;65(9):1301–1303.
6. de Noord O. Multivariate calibration standardization. *Chemom Intell Lab Syst.* 1994;25(2):85–97.
7. de Noord O. The influence of data preprocessing on the robustness and parsimony of multivariate calibration models. *Chemom Intell Lab Syst.* 1994;23(1):65–70.
8. Bouveresse E, Massart D. Improvement of the piecewise direct standardization procedure for the transfer of NIR spectra for multivariate calibration. *Chemom Intell Lab Syst.* 1996;32(2):201–213.
9. Feudale R, Woody N, Tan H, Myles A, Brown S, Ferré J. Transfer of multivariate calibration models: a review. *Chemom Intell Lab Syst.* 2002;64(2):181–192.
10. Anderson C, Kalivas J. Fundamentals of calibration transfer through Procrustes analysis. *Appl Spectrosc.* 1999;53(10):1268–1276.
11. Walczak B, Bouveresse E, Massart D. Standardization of near-infrared spectra in the wavelet domain. *Chemom Intell Lab Syst.* 1997;36(1): 41–51.
12. Sjöblom J, Svensson O, Josefson M, Kullberg H, Wold S. An evaluation of orthogonal signal correction applied to calibration transfer of near infrared spectra. *Chemom Intell Lab Syst.* 1998;44(1-2):229–244.
13. Fearn T. Standardisation and calibration transfer for near infrared instruments: a review. *J Near Infrared Spectrosc.* 2001;9(4):229–244.
14. Kalivas J, Siano G, Andries E, Coicoechea H. Calibration maintenance and transfer using Tikhonov regularization approaches. *Appl Spectrosc.* 2009;63(7):800–809.
15. Kunz M, Kalivas J, Andries E. Model updating for spectral calibration maintenance and transfer using 1-norm variants of Tikhonov regularization. *Anal. Chem*. 2010;82(9):3642–3649.
16. Sulub Y, Small G. Spectral simulation methodology for calibration transfer of near-infrared spectra. *Appl Spectrosc.* 2007;61(4):406–413.
17. Haaland D, Melgaard D. New prediction-augmented classical least squares (PACLS) methods: application to unmodeled interferents. *Appl Spectrosc.* 2000;54(9):1303–1312.
18. Igne B, Hurburgh C. Standardisation of near infrared spectrometers: evaluation of some common techniques for intra- and inter-brand calibration transfer. *J Near Infrared Spectrosc.* 2008;16(6):539–550.
19. Daumé IIIH, Marcu D. Domain adaptation for statistical classifiers. *J Artif Intell Res.* 2006;26:101–126.
20. Daumé HIII. Frustratingly easy domain adaptation. *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Prague, Czech Republic: Association of Computational Linguistics (ACL), 2007;256–263.
21. Jiang J. A Literature Survey on Domain Adaptation of Statistical Classifiers. http://www.mysmu.edu/faculty/jingjiang/papers/da_survey.pdf Accessed April 15, 2016.
22. Quinonero-Candela J, Sugiyama M, Schwaighofer A, Lawrence N. *Dataset Shift in Machine Learning*. Cambridge, MA, USA: MIT Press; 2009.
23. Pan S, Yang Q. A survey on transfer learning. *IEEE Trans Knowl Data Eng.* 2010;22(10):1345–1359.
24. Patel V, Gopalan R, Li R, Chellappa R. Visual domain adaptation: a survey of recent advances. *IEEE Signal Process Mag*. 2015;32(3):53–69.
25. Tommasi T, Patricia N, Caputo B, Tuytelaars T. A deeper look at dataset bias. *German Conference on Pattern Recognition*. Aachen, Germany: Springer International Publishing, 2015;504–516.
26. Fernando B, Habrard A, Sebban M, Tuytelaars T. Unsupervised visual domain adaptation using subspace alignment. *Proceedings of the 2013 IEEE International Conference on Computer Vision ICCV '13*. Sydney, NSW, Australia: Computer Vision Foundation, 2013;2960–2967.
27. Blitzer J, Dredze M, Pereira F. Biographies, Bollywood, boom-boxes and blenders: domain adaptation for sentiment classification. *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Prague, Czech Republic: Association of Computational Linguistics (ACL), 2007;440–447.
28. Yang J, Yan R, Hauptmann AG. Cross-domain video concept detection using adaptive SVMs. *Proceedings of the 15th ACM International Conference on Multimedia*. Augsburg, Germany: Association for Computing Machinery (ACM), 2007;188–197.
29. Wold S, Antti H, Lindgren F, Öhman J. Orthogonal signal correction of near-infrared spectra. *Chemom Intell Lab Syst.* 1998;44(1-2):175–185.
30. Tan H, Brown S. Wavelet analysis applied to removing non-constant, varying spectroscopic background in multivariate calibration. *J Chemom.* 2002;16(5):228–240.
31. Andrew A, Fearn T. Transfer by orthogonal projection: making near-infrared calibrations robust to between-instrument variation. *Chemom Intell Lab Syst.* 2004;72(1):51–56.
32. Igne B, Roger J, Roussel S, Bellon-Maurel V, Hurburgh C. Improving the transfer of near infrared prediction models by orthogonal methods. *Chemom Intell Lab Syst.* 2009;99(1):57–65.
33. Koren Y. Robust linear dimensionality reduction. *IEEE Trans Visual Comput Graphics*. 2004;10(4):459–470.
34. Ji S, Ye J. Generalized linear discriminant analysis: a unified framework and efficient model selection. *IEEE Trans Neural Networks*. 2008;19(10):1768–1782.
35. van der Maaten L, Postma E, van den Herik H. Dimensionality reduction: a comparative review. TiCC-TR 2009-005, Tilburg University; 2009.
36. Longsine D, McCormick S. Simultaneous Rayleigh-quotient minimization methods for $\mathbf{Ax} = \lambda\mathbf{Bx}$. *Linear Algebra Appl*. 1980;34:195–234.
37. Sameh A, Wisniewski J. A trace minimization algorithm for the generalized eigenvalue problem. *SIAM J Numer Anal.* 1982;19:1243–1259.
38. Duda R, Hart P, Stork D. *Pattern Classification*. 2nd ed. Hoboken: Wiley-Interscience; 2001.
39. Ghifary M, Balduzzi D, Kleijn W, Zhang M. Scatter Component Analysis: A Unified Framework for Domain Adaptation and Domain Generalization, 2015. arXiv:cs/1510.04373. arXiv.org e-Print archive. http://arxiv.org/abs/1510.04373. Accessed April 15, 2016.
40. Stork C, Kowalski B. Weighting schemes for updating regression models—a theoretical approach. *Chemom Intell Lab Syst.* 1999;48(2):151–166.
41. Brown C. Discordance between net analyte signal theory and practical multivariate calibration. *J Chemom.* 2004;76(15):4364–4373.
42. Hansen PC. *Rank-deficient and Discrete Ill-posed Problems: Numerical Aspects of Linear Inversion*. Philadelphia, PA, USA: SIAM Press; 1998.
43. Lawson C, Hanson R. *Solving Least Squares Problems*. Englewood Cliffs, NJ, USA: Prentice Hall Press; 1974.
44. Pan S, Tsang I, Kwok J, Yang Q. Domain adaptation via transfer component analysis. *IEEE Trans on Neural Networks*. 2009;22(2): 1187–1192.
45. International Diffuse Reflectance Conference (IDRC). Chambersburg, PA, USA: NIR Spectra of Pharmaceutical Tablets 2002. http:\ignorespaces//www.eigenvector.com/data/tablets/index.html. Accessed April 15, 2016.

46. NIR of Corn Samples for Standardization Benchmarking. http://www.eigenvector.com/data/Corn/ Accessed April 15, 2016.
47. Dyrby M, Engelsen S, Nørgaard L, Bruhn M, Nielsen L. Chemometric quantitation of the active substance in a pharmaceutical tablet using near infrared (NIR) transmittance and NIR FT Raman spectra. *Appl Spectrosc*. 2002;56(5):579–585. http://www.models.life.ku.dk/tablets Accessed April 15, 2016.
48. Wheat Kernels. http://www.models.life.ku.dk/wheat_kernels. Accessed April 15, 2016.
49. Willett P. Combination of similarity rankings using data fusion. *J Chem Inf Model*. 2013;53(1):1–10.

How to cite this article: Andries E. Penalized eigendecompositions: motivations from domain adaptation for calibration transfer. *Journal of Chemometrics*. 2017;31:e2818. https://doi.org/10.1002/cem.2818