Computing in SCIENCE & ENGINEERING

IEEE Xplore® Digital Library
http://ieeexplore.ieee.org/Xplore

Department: Head
Editor: Name, xxxx@email

Discovering Geometry in Data Arrays

Eric C. Chi
Department of Statistics, Rice University

***Abstract*—Modern technologies produce a deluge of complicated data. In neuroscience, for example, minimally invasive experimental methods can take recordings of large populations of neurons at high resolution under a multitude of conditions. Such data arrays possess non-trivial interdependencies along each of their axes. Insights into these data arrays may lay the foundations of advanced treatments for nervous system disorders. The potential impacts of such data, however, will not be fully realized unless the techniques for analyzing them keep pace. Specifically, there is an urgent, growing need for methods for estimating the low-dimensional structure and geometry in big and noisy data arrays. This article reviews a framework for identifying complicated underlying patterns in such data and also recounts the key role that the Department of Energy Computational Sciences Graduate Fellowship played in setting the stage for this work to be done by the author.**

IT is a great privilege to have the opportunity to describe the impact that the Department of Energy (DOE) Computational Sciences Graduate Fellowship (CSGF) has had on my career. The fellowship has several distinguishing features, but I will highlight two features that have non-trivially influenced me during my formative years as a graduate student and beyond.

The first feature is the national laboratory practicum experience. Fellows are required to complete a practicum at a DOE national lab. The CSGF program generously funded me to complete three practica: one at Lawrence Berkeley National Laboratory and two at Sandia National Laboratories in Livermore, California. My experiences at Sandia, in particular, played a central role in introducing me to problems that I continue to study today. I owe a lot to my practica supervisor, Dr. Tamara Kolda. We worked on two variations on tensor decompositions; the latter one on nonnegative decompositions for sparse count data was published in the SIAM Journal of Matrix Analysis and Applications [1]. Tensor decompositions can be employed as a dimensionality reduction technique and are often computed as solutions to optimization problems. As I will describe in more detail in this article, major themes of my research interests and efforts continue to revolve around dimensionality reduction and optimization algorithms. Moreover, under Dr. Kolda's guidance (and high standards!), I acquired scientific computing and software development skills

Published by the IEEE Computer Society

Digital Object Identifier 10.1109/MCSE.2021.3120039

that are uncommon in a PhD statistics program. It was also through my collaborations with Dr. Kolda that I developed a taste for designing algorithms that not only behave well on paper but also admit practical implementations. Developing these skills under the supervision of a leading computational scientist and mathematician at a national lab was especially timely with a data science revolution just on the horizon.

The second feature is the CSGF's emphasis on building a community of computational scientists. For example, the CSGF holds an annual program review bringing together fellows and alumni working across an extremely wide spectrum of disciplines. The meeting is a unique melting pot experience with poster presentations by junior fellows and oral presentations by outgoing senior fellows that are aimed for the non-specialist. The program also regularly hosts gatherings for fellows and alumni at larger meetings, e.g., SIAM CS&E. Through these efforts, the CSGF program has created an environment where computational scientists can build and develop connections and collaborations across disciplines. In my case, I owe two such collaborations to the CSGF program. I am currently working with a fellow alum of the program and professor of biomedical engineering at Duke University, Dr. Amanda Randles, to develop machine learning algorithms to better characterize a poorly understood type of coronary artery disease so that more effective treatment plans can be devised. I have also worked with Dr. Mary Ann Leung, who is not only an alum of the program but was also its program manager during my tenure as a fellow, to develop an outreach program for high school students that is connected to my research. I will defer details on this outreach and its CSGF influences to the end of this article. I first describe the research I alluded to earlier.

The Data Revolution

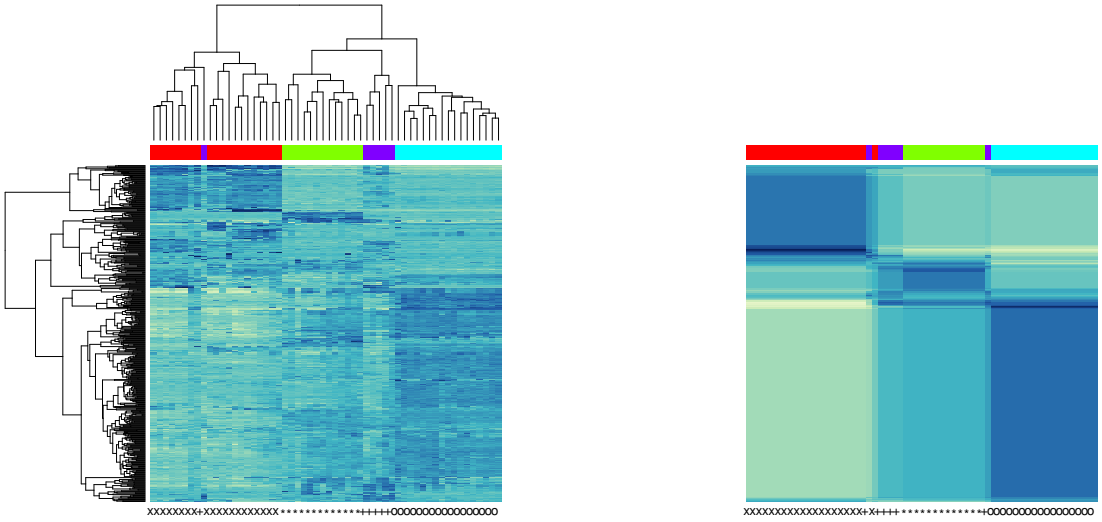
We are in the midst of a data revolution; data are being collected at not only increasingly higher volumes and speed, but also at previously unimaginable resolution. For example, minimally invasive experimental methods have enabled recordings of large populations of neurons at high resolution [2]. As another example, recent technological innovations have made it feasible to extract and amplify minuscule quantities of RNA, enabling the quantification of genome-wide transcriptional activity at the level of a single cell [3]. In the former case, such detailed physiological measurements hold the promise of understanding how information is represented, stored, and modified in cortical networks. In the latter case, the ability to finely discriminate between cell types based on their transcriptional profiles could yield insights that lay the foundations of advanced treatments for genetic disorders. In many cases, such high resolution measurements are being acquired under multiple combinations of different experimental conditions leading to data that consist of multiway arrays.

Multiway arrays, also referred to as tensors, are the generalization of matrices, which are two-way arrays, to arrays that store values that are indexed along three or more ways or modes. A 3-way tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ is a data cube that contains IJK elements x_{ijk} where the indices i, j, and k run from 1 to I, J, and K respectively. For example, a data cube $\mathcal{X}$ may consist of neural activity recordings that are indexed three ways where x_{ijk} is the activity level of the ith neuron during the jth time frame of the kth experimental condition.

A major aim of my research is to develop tools for analyzing multiway data. In this article, I focus on the special case of two-way arrays or data matrices as an example for notational simplicity and for expositional clarity. In this context, my research seeks to answer the following basic question: Does a data matrix have a latent or intrinsic organization or geometry along its rows and columns? Sometimes the organization that we seek is precisely defined: Do the rows and columns of a data matrix form well-defined clusters? Other times the organization that we seek is more nuanced: Is there a pair of lower dimensional representations where the rows and columns of a data matrix lie on similarity continua? I have two interrelated lines of work in this major focus on multiway data: co-clustering and co-manifold learning.

Co-Clustering

The simplest version of the co-clustering problem is biclustering. Biclustering aims to simultaneously group observations (rows) and fea-



(a) Raw Data

(b) Convex Co-clustering Estimate

Figure 1: Heatmaps of the expression of 500 genes (rows) across 56 tissue samples (columns). Figure 1a depicts the clustered dendrogram applied to the raw data; Figure 1b depicts the smooth estimate computed by convex co-clustering. Tissue samples belong to one of four subgroups: Normal (o), Carcinoid (x), Colon (*), and Small Cell (+). Used, with permission, from Chi et al. [4]

tures (columns) in a data matrix and is a principal tool for visualization and exploratory analysis in a wide array of applications.

Figure 1a illustrates an example of a clustered dendrogram, a widely used method for biclustering in bioinformatics, applied to expression data from a lung cancer study. The clustered dendrogram constructs similarity trees, or dendrograms, on the rows and columns of the data matrix and displays these dendrograms along with a heatmap of the data matrix with its rows and columns permuted with respect to the dendrograms.

This luncer cancer data matrix $\mathbf{X} \in \mathbb{R}^{500\times 56}$ consists of the expression levels of 500 genes across 56 tissue samples, a subset of the data studied in Bhattacharjee et al. [5]. The matrix element x_{ij} quantifies the expression level or degree of activity of the ith gene in the jth individual. In Figure 1a darker colors in the heatmaps indicate greater expression levels. Tissue samples belong to one of four subgroups: Normal, Carcinoid, Colon, or Small Cell. The latter three subgroups are distinct subtypes of lung cancer. Cancer is the result of cellular dysfunction where some genes are more active than they should be while other genes are not as active as they should be. While a cancer, such as lung cancer, may appear homogenous macroscopically at the clinical level, it often consists of several distinct subtypes microscopically at the gene level. A fundamental task of cancer research is to identify subtypes of cancerous tumors that have similar expression profiles as well as the genes that characterize each of the subtypes. Identifying these patterns is the first step towards developing personalized treatment strategies targeted to a patient's particular cancer subtype.

We see in Figure 1a that the clustered dendrogram reveals a "checkerboard" biclustering pattern in the expression data which seems to reasonably recover the ground truth subtypes of tissues and identify groups of genes with similar expression levels that characterize the subgroups of tissues. Nonetheless, there are two non-trivial issues with the simple strategy. First, the clustered dendrogram clusters the rows and columns separately. We will discuss shortly why clustering rows and columns jointly is better, even though it may be computationally more expensive, than doing them separately. Second, as an algorithmic procedure, the clustered dendrogram is not stable in the sense that small perturbations in the

data can lead to large changes in the clustering assignments. Robustness or insensitivity to perturbations in the data is a critically missing property of existing biclustering methods as stability in this sense is a necessary building block for reproducible research.

First issue: To develop some intuition on why using the column clustering structure affects the row clustering structure, consider the following thought experiment. Imagine trying to cluster row vectors $\mathbf{x}_i \in \mathbb{R}^{100,000}$ for $i = 1, \ldots, 100$ drawn from the following two-component mixture of Gaussians, namely

$$\mathbf{x}_i \overset{iid}{\sim} \frac{1}{2}N(\boldsymbol{\mu}, \sigma^2\mathbf{I}) + \frac{1}{2}N(\boldsymbol{\nu}, \sigma^2\mathbf{I}),$$

where $N(\boldsymbol{\mu}, \sigma\mathbf{I})$ denotes a multivariate Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\sigma\mathbf{I}$. We are attempting to solve an inverse problem using only 100 observations to estimate 200,000 parameters in the mean vectors $\boldsymbol{\mu} \in \mathbb{R}^{100,000}$ and $\boldsymbol{\nu} \in \mathbb{R}^{100,000}$. This is a hopelessly underdetermined clustering problem due to the glaringly small number of observations compared to the number of features. One can view this problem as a nonlinear version of seeking a least squares solution to a system 100 linear equations in 200,000 unknowns. Suppose, however, that we were told that $\mu_j = \mu_1$ and $\nu_j = \nu_1$ for $j = 1, \ldots, 50,000$, and $\mu_j = \mu_2$ and $\nu_j = \nu_2$ for $j = 50,001, \ldots, 100,000$. In other words, the features are clustered into two groups. So, we only need to estimate four parameters: μ_1, μ_2, ν_1, and ν_2. Now we have an abundance of observations compared to the number of effective features. One can view this latter problem as a nonlinear version of seeking a least squares solution to a system 100 linear equations in 4 unknowns. Thus, even if we lack a clear-cut clustering structure in the features, this exercise suggests that leveraging similarity structure along the columns can expedite identifying similarity structure along the rows, and vice versa. Another way to view jointly estimating structure in a matrix is as introducing regularization along the rows to improve estimation of column structure, and vice versa.

Second issue: To address the lack of stability guarantees in existing biclustering methods, I introduced a new convex formulation of the biclustering problem [4], as well as a generalization of this convex formulation to tensors [6]. The convex formulation not only produces co-clusterings with the desired stability properties but also admits practical computation via linear time and space complexity algorithms.

Figure 1b shows the solution to the convex optimization problem, which exhibits a visually sharper row and column clustering structure. In fact, row and column clusters can be automatically extracted by inspecting the pairwise differences of the rows and columns in the solution matrix. We first review how to cast the problem of clustering the columns of a matrix as a convex optimization problem before showing how to extend the formulation to co-clustering.

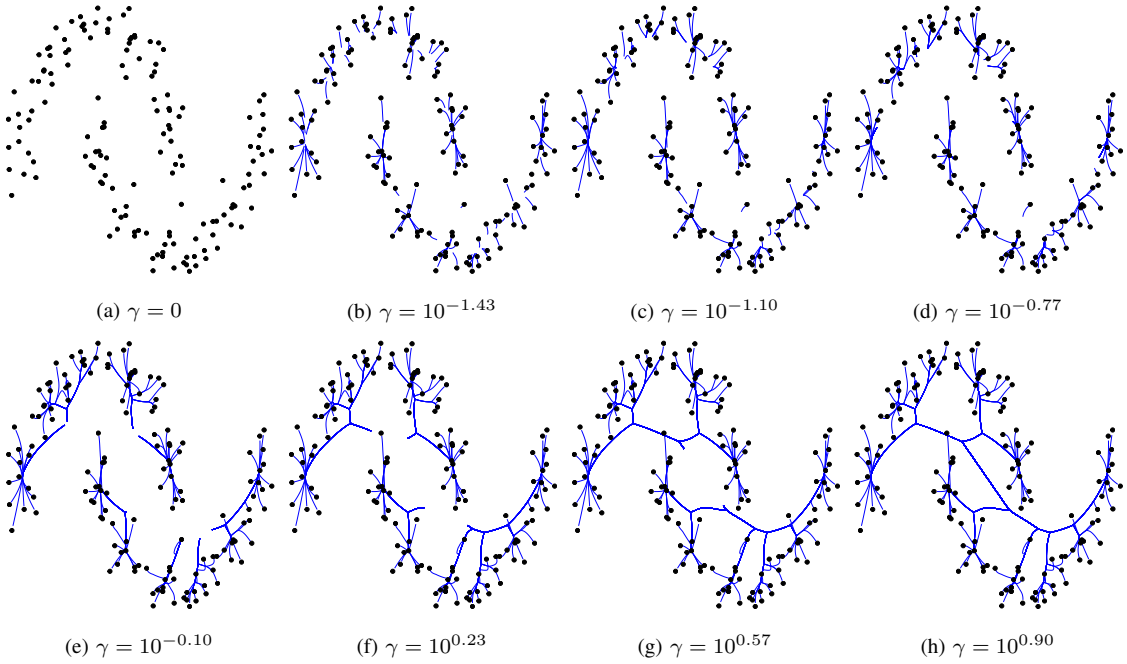
Convex Clustering

Following up on the initial proposal by Pelckmans et al. [7], several recent works have shown that solving a sequence of convex optimization problems can recover tree organizations [8], [9], [10]. Given m points $\mathbf{x}_1, \ldots, \mathbf{x}_m$ in $\mathbb{R}^n$, we seek cluster centers (centroids) $\mathbf{u}_i$ in $\mathbb{R}^n$ attached to point $\mathbf{x}_i$ that minimize the convex objective function

$$E_\gamma(\mathbf{u}) = \frac{1}{2}\sum_{i=1}^{m}\|\mathbf{x}_i - \mathbf{u}_i\|_2^2 + \gamma\sum_{i<j} w_{ij}\|\mathbf{u}_i - \mathbf{u}_j\|_2, \quad (1)$$

where γ is a nonnegative tuning parameter, w_{ij} is a nonnegative weight that quantifies how similar $\mathbf{x}_i$ and $\mathbf{x}_j$ are, and $\mathbf{u}$ is the vector in $\mathbb{R}^{mn}$ that follows from stacking the vectors $\mathbf{u}_1, \ldots, \mathbf{u}_m$ on top of each other. The sum of squares data-fidelity term in (1) quantifies how well the centroids $\mathbf{u}_i$ approximate the data $\mathbf{x}_i$, while the sum of norms regularization term penalizes the differences between pairs of centroids $\mathbf{u}_i$ and $\mathbf{u}_j$. The regularization term incentivizes sparsity in the pairwise differences of centroid pairs. The objective function $E_\gamma(\mathbf{u})$ is the energy of a configuration of centroids $\mathbf{u}$ for a given relative weighting γ between data-fidelity and model complexity as quantified by the regularization term.

For each value of γ, the objective function $E_\gamma(\mathbf{u})$ in (1) possesses a unique minimizer $\mathbf{u}(\gamma)$, whose m subvectors in $\mathbb{R}^n$ we denote by $\mathbf{u}_i(\gamma)$



(a) $\gamma = 0$ (b) $\gamma = 10^{-1.43}$ (c) $\gamma = 10^{-1.10}$ (d) $\gamma = 10^{-0.77}$

(e) $\gamma = 10^{-0.10}$ (f) $\gamma = 10^{0.23}$ (g) $\gamma = 10^{0.57}$ (h) $\gamma = 10^{0.90}$

Figure 2: Snapshots of the convex clustering solution path $\mathbf{u}(\gamma)$ of a point cloud $\mathbf{x}_1, \ldots, \mathbf{x}_m$ as the parameter γ increases. The path (blue lines) recovers a multiscale organization of the point cloud.

[10]. The tuning parameter γ trades off the relative emphasis between data fit and differences between pairs of centroids. When $\gamma = 0$, the minimum is attained when $\mathbf{u}_i = \mathbf{x}_i$, namely when each point occupies a unique cluster. As γ increases, the regularization term encourages cluster centroids to fuse together. Two points $\mathbf{x}_i$ and $\mathbf{x}_j$ with $\mathbf{u}_i = \mathbf{u}_j$ are said to belong to the same cluster. For sufficiently large γ, the $\mathbf{u}_i$ fuse into a single cluster, namely $\mathbf{u}_i = \bar{\mathbf{x}}$, where $\bar{\mathbf{x}}$ is the average of the data $\mathbf{x}_i$ [10]. Moreover, the unique global minimizer $\mathbf{u}(\gamma)$ is a continuous function of the tuning parameter γ [4]; we refer to the continuous paths $\mathbf{u}_i(\gamma)$, traced out from each $\mathbf{x}_i$ to $\bar{\mathbf{x}}$ as γ varies, collectively as the solution path. Thus, by computing $\mathbf{u}_i(\gamma)$ for a sequence of γ over an appropriately sampled range of values, we hope to recover a sensible tree organization of the data. Figure 2 shows snapshots of the solution path $\mathbf{u}(\gamma)$ computed on a point cloud consisting of two interlocking "half moons." Each half moon represents a cluster. The blue solid lines in Figures 2b to 2h show a linear interpolation of centroids $\mathbf{u}_i(\gamma)$ computed over a grid of 100 γ values. We see that there are only four unique centroid values $\mathbf{u}_i(10^{0.23})$ indicating that when $\gamma = 10^{0.23}$ the data points have fused into four clusters. There are only two unique centroid values of $\mathbf{u}_i(10^{0.57})$ indicating that when $\gamma = 10^{0.57}$ the data points have fused into two clusters. Finally, all the centroid values $\mathbf{u}_i(10^{0.90})$ are identical indicating that when $\gamma = 10^{0.90}$ the data points have fused into a single cluster. We see that the solution path $\mathbf{u}(\gamma)$ recovers a tree organization where the two main branches are subtrees that hierarchically organize the points in each half moon cluster.

The example shown in Figure 2 illustrates that remarkably, solving a sequence of convex optimization problems can recover a hierachical tree or multiscale organization of a point cloud - a problem often posed as a discrete optimization problem. From a computational perspective, solving a convex optimization problem is often preferable to solving a discrete one. Many convex optimization problems, including the convex clustering optimization problem, can be solved with algorithms that can scale to large problems. By contrast, the majority of discrete optimization problems are inherently combinatorial and

require searching through potential solution sets that grow exponentially fast as the problem size grows. Moreover, formulating this discrete organization task as a family of convex optimization problems produces solutions with desirable stability properties and also generalizes easily to the co-clustering task. A natural question is: Does the convex clustering solution path always return a tree organization of a point cloud? The answer to this question lies in the choice of the weights w_{ij}. Indeed, it is possible to choose the w_{ij} so that the solution path is not a tree. Fortunately, one has to make deliberate effort to engineer such pathological weights and simple and intuitive data-adaptive choices are guaranteed to ensure the recovery of a tree organization that respects the geometry of a point cloud [11].

Convex Co-Clustering

Extending convex clustering to convex co-clustering is straightforward. To bicluster a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$, we seek the unique global minimizer to the following convex objective function

$$E_\gamma(\mathbf{U}) = \frac{1}{2}\|\mathbf{X} - \mathbf{U}\|_{\mathrm{F}}^2 + \gamma \left[\Omega_{\mathbf{W}}(\mathbf{U}) + \Omega_{\tilde{\mathbf{W}}}(\mathbf{U}^{\mathsf{T}})\right], \tag{2}$$

where $\Omega_{\mathbf{W}}(\mathbf{U}) = \sum_{i<j} w_{ij}\|\mathbf{U}_{\cdot i} - \mathbf{U}_{\cdot j}\|_2$, and $\mathbf{U}_{\cdot i}$ ($\mathbf{U}_{i\cdot}$) denotes the ith column (row) of the matrix $\mathbf{U}$. The energy function incorporates a regularization term that includes both a penalty on the pairwise differences of columns $\Omega_{\mathbf{W}}(\mathbf{U})$ and rows $\Omega_{\tilde{\mathbf{W}}}(\mathbf{U}^{\mathsf{T}})$. Thus, the rows and columns of $\mathbf{U}$ are simultaneously shrunk towards each other as the parameter γ increases. The unique global minimizer, which we refer to as the Convex Co-clustering (CoCo) estimator, exhibits a checkerboard structure as seen in Figure 1b. The biclustering obtained by the CoCo estimator is fundamentally different from methods like the clustered dendrogram, which independently clusters the rows and columns. By coupling row and column clustering, our formulation explicitly seeks a solution with a "checkerboard" structure.

Figure 3 shows snapshots of the CoCo solution path $\mathbf{U}(\gamma)$ on the lung data set, as the parameter γ takes on an increasing sequence of values. The path captures biclustering organizations of the data over a wide range of scales and resolutions from under-smoothed estimates of the mean structure (small γ), where each element of the data matrix $\mathbf{X}$ is assigned its own bicluster, to over-smoothed estimates (large γ), where all elements of the data matrix $\mathbf{X}$ are assigned to a single bicluster. In between these extremes, we see rows and columns "fusing" together as γ increases. Thus we have visual confirmation that minimizing (2) over a range of γ yields a convex formulation of the clustered dendrogram.

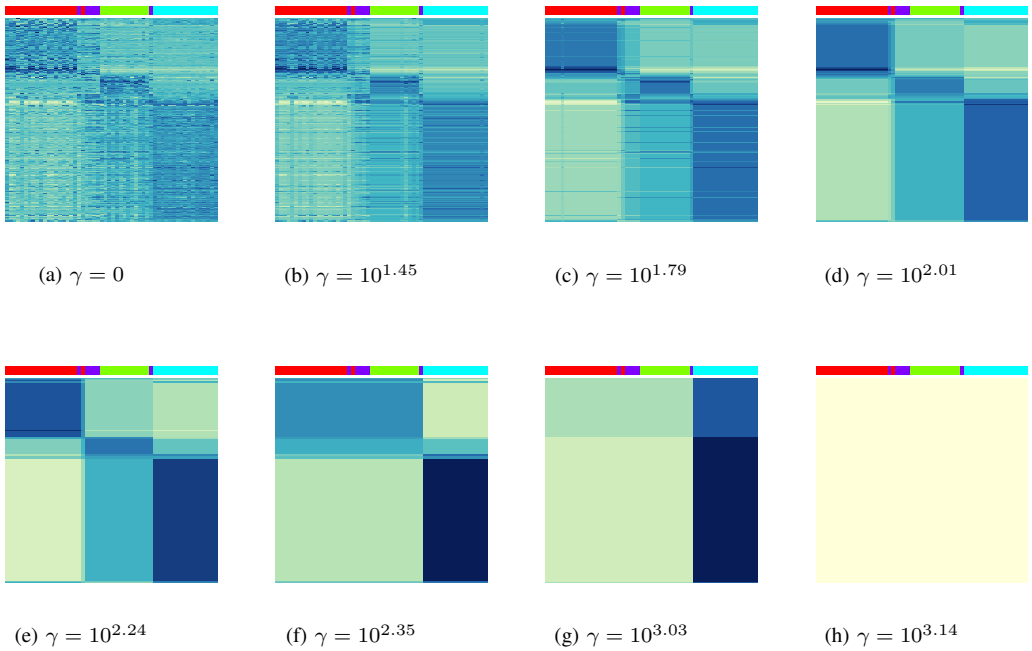
The CoCo estimator has several notable properties. First, the CoCo estimator is jointly continuous in all input parameters: γ, row and column weights w_{ij} and $\tilde{w}_{ij}$, and data $\mathbf{X}$ [4], [6] and is 1-Lipschitz in the data $\mathbf{X}$ [6]. This latter property warrants further explanation. Suppose we compute the CoCo estimator $\mathbf{U}(\mathbf{X})$ using the data $\mathbf{X}$ and compute the CoCo estimator $\mathbf{U}(\mathbf{X} + \Delta\mathbf{X})$ on the perturbed data $\mathbf{X} + \Delta\mathbf{X}$. Then

$$\|\mathbf{U}(\mathbf{X}) - \mathbf{U}(\mathbf{X} + \Delta\mathbf{X})\|_{\mathrm{F}} \leq \|\Delta\mathbf{X}\|_{\mathrm{F}}. \tag{3}$$

The above inequality tells us that the CoCo estimator is stable in the sense that a small perturbation $\Delta\mathbf{X}$ in the data $\mathbf{X}$ is guaranteed to not lead to disproportionately wild variations in the output. In fact, the change in the CoCo estimator cannot exceed the change in the input data.

Second, for a D-way tensor with $D \geq 3$ modes or ways, the solution to the optimization problem will recover a "co-clusterable" underlying tensor, in the sense that the underlying tensor has a "checkerbox" pattern under some permutation or reshuffling of its elements, with high probability even if the number of clusters along each mode is diverging [6]. The remarkable part of this result is a "Blessings of Dimensionality" phenomenon where the prediction error still vanishes with high probability even if the number of clusters grows at the rate $o(n^{(D-2)/(D-1)})$, where n is the number of observations along each mode, or almost as fast as new observations are observed along each mode [6]. This result gives us confidence in applying the method in practice, as the rationale for co-clustering is a prior belief that there are relatively fewer co-clusters than there are observations.

Finally, both the computational and storage complexity of the CoCo estimator is linear in the size of the data using the commonly used data-adaptive and theoretically justified sparse



(a) $\gamma = 0$ (b) $\gamma = 10^{1.45}$ (c) $\gamma = 10^{1.79}$ (d) $\gamma = 10^{2.01}$

(e) $\gamma = 10^{2.24}$ (f) $\gamma = 10^{2.35}$ (g) $\gamma = 10^{3.03}$ (h) $\gamma = 10^{3.14}$

Figure 3: Snapshots of the CoCo solution path of the lung cancer data set as the parameter γ increases. The path captures a dynamic range in model resoluion between under-smoothed estimates of the mean structure (small γ), where each element of the matrix is assigned its own bicluster, to over-smoothed estimates (large γ), where all elements of the matrix are assigned to a single common bicluster. Used, with permission, from Chi et al. [4]

Gaussian kernel weights [11] in conjunction with a projected gradient method applied to the Lagrangian dual of the objective function given in Equation (2). Thus, doubling the size of the input data doubles the runtime and storage requirements. Moreover, continuity of the CoCo estimator $\mathbf{U}(\gamma)$ in γ can be leveraged to expedite computation through warm starts, namely using the solution $\mathbf{U}(\gamma)$ as the initial guess for iteratively computing $\mathbf{U}(\gamma')$ where γ' is slightly larger or smaller than γ.

Co-Manifold Learning

In some cases, seeking a co-clustering structure is overly simplistic; instead of a distinct and well-defined row and column grouping, the organizational structure along the rows and columns may be more continuous. Thus, our goal may be to identify new row and column representations that can reveal such structure – in other words, we seek to perform dimensionality reduction on the rows and columns of the data matrix. There are two main challenges in performing such dimensionality reductions: (i) For modern data matrices, measurements along the rows or columns reside in a high dimensional ambient spaces, and (ii) measurements along the rows and columns often exhibit non-trivial correlation structure. Tools exist for dealing with the first challenge since many high-dimensional datasets encountered in engineering and science can be approximated reliably by a lower dimensional representation. Indeed, manifold learning has proven to be effective as a nonlinear dimension reduction technique in many scientific domains where very high-dimensional measurements are recorded, such as the examples in neuroscience and bioinformatics described at the start of this article. With some reflection, this is not surprising since these high-dimensional data are generated from natural processes that are subject to physical constraints and are consequently intrinsically low-

dimensional. More concretely, conservation laws in physics represent lower-dimensional manifolds in the higher-dimensional state space of possible solutions. Returning to our two challenges, less progress has been made to deal with the second challenge. Naively applying existing nonlinear dimension reduction techniques separately along the modes of a tensor fails to take advantage of the rich correlation structure in many data arrays of interest. Consequently, I have been developing methods that leverage the correlations among the modes of a tensor to simultaneously learn coupled low-dimensional representations of each mode.

To illustrate the utility of learning a coupled set of representations, consider a dimensionality reduction problem in cheminformatics where we seek to identify groups of compounds with similar bioactivity towards a therapeutically-relevant target. Pharmaceutical companies may use this information to screen tens of thousands of lead compounds for desired activity and safety profiles. The resulting low-dimensional representations of compounds can highlight which novel compounds are most similar to known reference compounds. Simultaneously, the resulting low-dimensional representations of bioactivity assays can reveal redundancies in assays, providing feedback on how to streamline future studies. Figure 4a shows an example of raw cheminformatics data, where the rows are compounds and the columns are compound features, e.g., binding affinity to different proteins. Figure 4b shows the cheminformatics data matrix after reordering the rows and columns based on a novel multiscale distance that I developed with collaborators [12]. There are clearly two major groups of columns, while the rows exhibit more of a continuum of similarities than distinct groups. The two panels on the bottom of Figure 4 show the low-dimensional representations on rows and columns of the cheminformatics matrix recovered by co-manifold learning. The co-manifold learning framework based on the novel multiscale distance identifies a pair of two-dimensional coordinate systems that reveals unambiguous geometric relationships among the rows and among the columns. The compounds (rows) have a clear ordering along a 1-dimensional curve (the color coding of the row points in Figure 4c matches the row ordering in Figure 4b). The features (columns), meanwhile, have a clear clustered structure (the color coding of the column points in Figure 4d matches the column ordering in Figure 4b).

Interested readers are referred to Mishne et al. [12] for details, but I will briefly sketch how these coupled row and column low-dimensional representations are computed. There is a close connection between my work in co-manifold learning and co-clustering. The convex co-clustering estimator is the key building block for the co-manifold learning framework as it provides a way to simultaneously smooth rows and columns to different varying degrees. In other words, it provides a way to create a coupled pair of tree or multiscale organizations of the rows and columns of a data matrix.

The main work is to compute smooth estimates of the data matrix $\mathbf{X}$ along both the rows and columns, which is the minimizer $\mathbf{U}(\gamma_r, \gamma_c)$ of the objective function

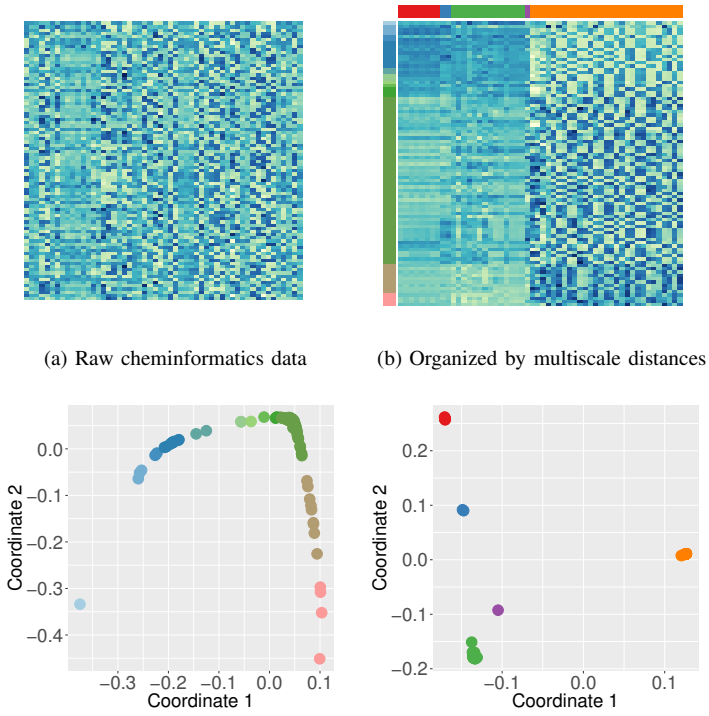
$$E(\mathbf{U}; \gamma_r, \gamma_c) = \frac{1}{2}\|\mathbf{X} - \mathbf{U}\|_{\mathrm{F}}^2 + \gamma_r J_r(\mathbf{U}) + \gamma_c J_c(\mathbf{U}), \tag{4}$$

where γ_r and γ_c are nonnegative tuning parameters, and $J_r(\mathbf{U})$ and $J_c(\mathbf{U})$ are regularization terms that impose smoothness in $\mathbf{U}$ along its rows and columns similar to $\Omega_{\mathbf{W}}$ and $\Omega_{\tilde{\mathbf{W}}}$ in Equation (2). Varying the parameters γ_r and γ_c trades off how well the estimate $\mathbf{U}$ agrees with $\mathbf{X}$ against how smooth $\mathbf{U}$ is along its rows and column. Smaller γ_r and γ_c enforce less smoothness on rows and columns of the data matrix.

Smooth estimates $\mathbf{U}(\gamma_r, \gamma_c)$ are computed over a grid of values for γ_r and γ_c by computing a sequence of CoCo estimators. Next a distance $d_r(i, j)$ between the ith and jth rows is computed by taking a weighted average over the pairwise difference over different smoothed estimates $\mathbf{U}(\gamma_r, \gamma_c)$,

$$d_r(i,j) = \sum_{\gamma_r, \gamma_c} \sqrt{\gamma_r \gamma_c}\, \Delta_{ij}(\gamma_r, \gamma_c),$$
$$\Delta_{ij}(\gamma_r, \gamma_c) = \|\mathbf{U}_{i\cdot}(\gamma_r, \gamma_c) - \mathbf{U}_{j\cdot}(\gamma_r, \gamma_c)\|_2. \tag{5}$$

Greater weight is given to the smoothed estimates corresponding to larger parameters γ_r and γ_c; these estimates are more heavily smoothed. Thus,



(a) Raw cheminformatics data

(b) Organized by multiscale distances

(c) Co-Manifold: Row

(d) Co-Manifold: Column

Figure 4: (a) Unordered cheminformatics matrix, darker colors indicate greater affinity. (b) Cheminformatics matrix with rows and columns ordered according to their low-dimensional representations discovered by co-manifold learning. (c, d) Low-dimensional representations of the rows and columns of the cheminformatics matrix that are recovered by co-manifold learning.

the distance places progressively less weight on discrepancies at lower levels of smoothing. If the ith and jth rows are similar at all smoothing levels, they will be close in the weighted distance given in (5). If they are different at all but the most smoothed scale, they will still be close in the weighted distance given in (5). Only if the two rows are different at the most smoothed scales will the two rows be apart in the weighted distance given in (5). Thus, small perturbations in the rows that would be amplified if a standard Euclidean distance were used are washed out, whereas only truly material differences in pairs of rows will persist in the multiscale distance.

The distance $d_c(i, j)$ between the ith and jth columns is computed analogously using the same collection of smoothed estimates $\mathbf{U}(\gamma_r, \gamma_c)$. Thus, the multiscale distances $d_r(i, j)$ and $d_c(i, j)$ both take into account the correlation structure among the rows and columns. The final lower dimensional representations can be obtained by applying standard spectral embedding techniques on the row and column distances respectively.

BROADER IMPACTS

Much of the work presented in this article is part of my NSF CAREER award. As part of this award, I also a run a year-long outreach program, Data Scientists in Training (DST), for high school students including those from underrepresented minorities in STEM. The goal of the DST program is to introduce students to careers in data science through hands-on experience with projects as well as mentoring and career guidance.

I owe much of the design and conceptualization of the DST program to guidance and input from Dr. Mary Ann Leung, currently Founder and President of The Sustainable Horizons Institute. While there were many elements to its execution as I will describe below, the common

theme and rationale behind these elements was to implement structures and collaborative activities to build community in a similar spirit to the CSGF among these high school students, who share a common curiosity in data science but come from different backgrounds. My goal is to create a supportive environment that would lower barriers to entry into this important and exciting field of data science and also nurture connections with peers and mentors who could help students buffer challenges that they might encounter while pursuing careers in data science.

As part of the DST program, I designed and taught a week-long bootcamp curricula on statistical concepts, coding practices, and data analysis. Students also interviewed data scientists in industry (Netflix, Google, Microsoft, SAS, HEB), DOE national labs (Lawrence Livermore and Pacific Northwest), and academia (NC Central, Johns Hopkins, Harvard) and presented what they learned about the ways this diverse group of individuals arrived at their careers.

Beyond the summer bootcamp, the DST program includes mentoring, communication, and teaching opportunities for undergraduate and PhD statistics students who serve as mentors to their juniors. The program culminates each year with data analysis presentations at the North Carolina Junior Science and Humanities Symposium (NCJHS) for the high school participants, and the NC State Undergraduate Research Symposium for the undergraduate participants. The 2020 NCJSHS poster competition took place virtually in March due to COVID-19. One team worked on a year-long project using tensor decompositions that I developed with Dr. Kolda [1] to perform exploratory analysis of crime incident report data obtained from the Raleigh Police Department. The team employed the computed tensor factors to identify four clusters of assault patterns in Raleigh that had distinct spatio-temporal patterns and received an honorable mention at the 2020 NCJSHS poster competition.

I aim to expose students to the research component of my award through the technical programming of DST. Although the work presented in this article is built upon mathematics that is beyond the background of the participants in the DST program, the basic ideas can be readily grasped by a curious student. As illustrated in the figures of this article, much of my research is highly visual and intuitive. Appreciating how optimization problems can be designed and engineered to have solutions with desired structure can also be readily grasped. My goal is to spark an interest in the power of data science tools to solve problems in science and engineering and also to give participants guidance on future choices in their education if this is something that they would like to pursue. I have had many wonderful mentors and role models, but if I had to credit a single person for setting me on my own career path, it would be my high school geometry teacher, Dr. Michael Keyton. Dr. Keyton shared his delight in elegant proofs with all of his students and helped develop my interests and tastes early on. My hope for the DST program is to create a similar environment of discovery and growth for future data scientists.

ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation under grant DMS-1752692 and the National Institutes of Health under grant R01GM135928.

Eric C. Chi is currently an associate professor in the Department of Statistics at Rice University, Houston, TX, USA. He received his B.A. degree in physics from Rice University, Houston, Texas; his M.S. degree in electrical engineering from the University of California, Berkeley; and his Ph.D. degree in statistics from Rice University. His PhD studies were funded through a Department of Energy Computational Sciences Graduate Fellowship (DOE CSGF). He received an NSF CAREER award in 2018 and an ORAU Ralph E. Powe Junior Faculty Enhancement Award in 2017. He has served as an Associate Editor of the Journal of Computational and Graphical Statistics since 2016, and has been on the Editorial Board of Statistical Methods in Medical Research since 2011. His research interests include statistical learning and numerical optimization and their application to analyzing large and complicated modern data in biological science and engineering applications. Contact him at echi@rice.edu.

REFERENCES

1. E. C. Chi and T. G. Kolda, “On tensors, sparsity, and nonnegative factorizations,” *SIAM Journal on Matrix*

Analysis and Applications, vol. 33, no. 4, pp. 1272–1299, 2012.

2. G. M. G. Shepherd, "Corticostriatal connectivity and its role in disease," *Nature Reviews Neuroscience*, vol. 14, no. 4, pp. 278–291, 2013. [Online]. Available: https://doi.org/10.1038/nrn3469
3. F. Tang, C. Barbacioru, Y. Wang, E. Nordman, C. Lee, N. Xu, X. Wang, J. Bodeau, B. B. Tuch, A. Siddiqui, K. Lao, and M. A. Surani, "mRNA-Seq whole-transcriptome analysis of a single cell," *Nature Methods*, vol. 6, pp. 377–382, 2009.
4. E. C. Chi, G. I. Allen, and R. G. Baraniuk, "Convex biclustering," *Biometrics*, vol. 73, no. 1, pp. 10–19, 2017.
5. A. Bhattacharjee, W. G. Richards, J. Staunton, C. Li, S. Monti, P. Vasa, C. Ladd, J. Beheshti, R. Bueno, M. Gillette, M. Loda, G. Weber, E. J. Mark, E. S. Lander, W. Wong, B. E. Johnson, T. R. Golub, D. J. Sugarbaker, and M. Meyerson, "Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses," *Proc Natl Acad Sci U S A*, vol. 98, no. 24, pp. 13 790–13 795, Nov 2001.
6. E. C. Chi, B. J. Gaines, W. W. Sun, H. Zhou, and J. Yang, "Provable convex co-clustering of tensors," *Journal of Machine Learning Research*, vol. 21, no. 214, pp. 1–58, 2020. [Online]. Available: http://jmlr.org/papers/v21/18-155.html
7. K. Pelckmans, J. De Brabanter, J. A. Suykens, and B. L. De Moor, "Convex clustering shrinkage," in *PASCAL Workshop on Statistics and Optimization of Clustering Workshop*, 2005.
8. Y. She, "Sparse regression with exact clustering," *Electronic Journal of Statistics*, vol. 4, pp. 1055–1096, 2010.
9. T. Hocking, J.-P. Vert, F. Bach, and A. Joulin, "Clusterpath: An algorithm for clustering using convex fusion penalties," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ser. ICML '11, L. Getoor and T. Scheffer, Eds. New York, NY, USA: ACM, June 2011, pp. 745–752.
10. E. C. Chi and K. Lange, "Splitting methods for convex clustering," *Journal of Computational and Graphical Statistics*, vol. 24, no. 4, pp. 994–1013, 2015.
11. E. C. Chi and S. Steinerberger, "Recovering trees with convex clustering," *SIAM Journal on Mathematics of Data Science*, vol. 1, no. 3, pp. 383–407, 2019.
12. G. Mishne, E. Chi, and R. Coifman, "Co-manifold learning with missing data," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 4605–4614. [Online]. Available: http://proceedings.mlr.press/v97/mishne19a.html