A Projection free method for Generalized Eigenvalue Problem with a nonsmooth Regularizer

Seong Jae Hwang[a] Maxwell D. Collins[a] Sathya N. Ravi[b] Vamsi K. Ithapu[a]
Nagesh Adluru[e] Sterling C. Johnson[d] Vikas Singh[ca]

[a]Dept. of Computer Sciences, University of Wisconsin - Madison, Madison, WI
[b]Dept. of Industrial and Systems Engineering, University of Wisconsin - Madison, Madison, WI
[c]Dept. of Biostatistics and Med. Informatics, University of Wisconsin - Madison, Madison, WI
[d]William S. Middleton VA Hospital, Madison, WI
[e]Waisman Center, Madison, WI

Abstract

Eigenvalue problems are ubiquitous in computer vision, covering a very broad spectrum of applications ranging from estimation problems in multi-view geometry to image segmentation. Few other linear algebra problems have a more mature set of numerical routines available and many computer vision libraries leverage such tools extensively. However, the ability to call the underlying solver only as a "black box" can often become restrictive. Many 'human in the loop' settings in vision frequently exploit supervision from an expert, to the extent that the user can be considered a subroutine in the overall system. In other cases, there is additional domain knowledge, side or even partial information that one may want to incorporate within the formulation. In general, regularizing a (generalized) eigenvalue problem with such side information remains difficult. Motivated by these needs, this paper presents an optimization scheme to solve generalized eigenvalue problems (GEP) involving a (nonsmooth) regularizer. We start from an alternative formulation of GEP where the feasibility set of the model involves the Stiefel manifold. The core of this paper presents an end to end stochastic optimization scheme for the resultant problem. We show how this general algorithm enables improved statistical analysis of brain imaging data where the regularizer is derived from other 'views' of the disease pathology, involving clinical measurements and other image-derived representations.

1. Introduction

The explosion of photo or data sharing platforms in the last ten years has led to large and rich datasets where deriving a *single* all-encompassing representation for downstream statistical inference is challenging. Images often come with tags or user comments, and webpages can be characterized in terms of their textual content as well as the genre of related webpages. Even when working specifically with images, it is common to perform different feature extractions in the hope that all aspects of the image content are 'covered' by at least one feature type. Performing machine learning by fusing different views of the data is a well studied problem [3, 4, 6, 16, 26, 33].

Independent of the specific inference question of interest, observe that once the multiple views are in hand, practitioners often utilize off-the-shelf data exploration techniques to get a better sense of the derived representations and/or to identify reasonable parameter estimates for the subsequent components of the processing pipeline. To this end, spectral analysis is widely used for the evaluation of the heterogeneity in the groups and for feature selection [29]. In the latter setting, it is common to obtain the projection of the original distribution on the principal bases of the covariance and proceed with analyzing the embedded versions of the examples in the lower dimensional space instead. Frequently this may provide nicer affinity matrices which may be more suitable for machine learning tasks. When faced with multiple views, the above strategy can be applied to each view one by one, and the resultant affinity (or kernel) matrices can be averaged. But various recent results suggest that there is practical value in operating on each view separately and then enforcing consistency between the results obtained from each [24]. For example, in co-clustering, one imposes the constraint that leading eigenvectors across multiple views should be similar [4]. In the applied math literature, a more general version of the problems motivated from physics and engineering applications are studied as coupled eigenvalue problems [28]. From the perspective of the multi-view setup, this will entail solving a set of eigenvalue problems concurrently for the "primary"

and multiple "secondary" views. It turns out that when restricted to only two views, the formulation in some sense generalizes a very recent approach [14] for finding common eigenbases computed independently on different shapes.

The multiple view and co-clustering discussion above, while interesting, is not entirely essential to motivate eigenvalue problems in vision. Instances of eigen-decomposition are ubiquitous in computer vision in applications ranging from face recognition, indexing/hashing, registration, shape analysis to segmentation [10, 12, 22, 31, 34]. As soon as a formulation reduces to the eigenvalue form, a mature set of numerical analysis tools can be deployed directly. Their numerical behavior is well understood, and when faced with degenerate cases, it is also relatively easy to find robust preconditioners from the literature. That is, a black-box solver suffices. On the other hand, when a practitioner has additional supplementary information available for data, the existing solvers provide very little guidance on how such regularizers can be incorporated within the numerical optimization. In practice, such meta information may correspond to noisy labels in a semi-supervised setting, shape priors in segmentation, partial knowledge of a few eigen bases and so on [15]. In fact, we can also think of additional views of the data as regularizers on the primary eigen-decomposition. As we gradually move to systems where both the human and the statistical model mutually cooperate, it is important to derive end to end frameworks that offer such flexibility, yet retain much of the attractive numerical properties of their black-box counterparts.

With the foregoing motivation in mind, the main goal of this paper is to derive efficient numerical optimization schemes to solve a generalized eigenvalue problem with a nonsmooth regularizer, where few (if any) alternatives are currently available. We assume that the "mass matrix" in the eigenvalue formulation either comes naturally from the basic design (e.g., generalized Rayleigh [2]) or is a representation of the secondary views of the data. Separately, our formulation permits a fairly general (i.e., nonsmooth) regularizer. This may encode either partially observed or noisy meta knowledge about the data, common in crowd-sourced deployments or applications where a specific type of information is more expensive to obtain. Since a large majority of the data may be unobserved, standard imputation techniques are not applicable. The **contribution** of this work is to derive efficient numerical optimization schemes which solve the above problem as a trace minimization with generalized Stiefel constraints. We derive the update schemes and provide a detailed description of its properties. As an example, we show the applicability of these ideas to a statistical inference problem on brain imaging data, where we work with multiple derived representations of the image as well as measurements which are available only on a small subset of the participants.

2. Useful manifolds in numerical optimization

First, we present an overview of some manifolds that appear often in numerical optimization problems, which will serve as background material for much of the technical description that follows.

For vector spaces V and W denote by $L(V, W)$ the vector space of linear maps from V to W. Thus, the space of $L(\mathbb{R}^N, \mathbb{R}^p)$ may be identified with the space $\mathbb{R}^{N\times p}$ of $N\times p$ matrices. An injective linear map $u : \mathbb{R}^N \to V$ is called a $N-$frame in V. The set $\text{GF}_{N,p} = \{u \in L(\mathbb{R}^N, \mathbb{R}^p) : \text{rank}(u) = N\}$ of $N-$frames in $\mathbb{R}^p$ is called the Stiefel manifold. As a special case, when $N = p$, $\text{GF}_{N,N} :=\text{GF}_N$ is the General Linear group or the set of $N \times N$ matrices with nonzero determinant. In short, a Stiefel manifold is the set of $N\times p$ orthonormal matrices (with a Riemannian structure). The set of all $N-$dimensional (vector) subspaces $\alpha \subseteq \mathbb{R}^p$ is called the Grassmann manifold of $N-$planes in $\mathbb{R}^p$ and denoted by $\text{GR}_{N,p}$. With these definitions it is easy to see that the Grassmann manifold is just the Stiefel manifold quotiented by the Orthogonal group (set of orthogonal matrices) in $N-$dimensions. Let S_n be the set of $n \times n$ symmetric projection matrices with trace equal to p. Then we have that S_n is homeomorphic to $\text{GR}_{N,p}$ where the homeomorphism sends each element of S_n to its column space. Hence one may consider optimizing over S_n instead of $\text{GR}_{N,p}$ and vice-versa. Readers can see [1] for more details on these topics such as exponential map, tangent space and retraction.

Now, we will look at one prominent application of the manifolds described above in the context of computer vision, namely, *Spectral clustering*. Spectral clustering refers to a popular graph partitioning technique that analyzes the eigen structure of a matrix derived from the pairwise similarities of nodes, to identify clusters inherent in the data. The nodes in the graph represent individual data examples such as pixels in an image or vectors in a distribution $\mathcal{X}$. The algorithm, however, does not make use of the native space of $\mathcal{X}$, but rather the space induced by the chosen measure of similarity or the kernel matrix M. This works well because with a proper choice of M, the cohesiveness of clusters of points can be characterized via stability of the eigenvectors of its Laplacian matrix associated with the graph. Ordinary spectral clustering is formulated as

$$\min_{V\in\mathbb{R}^{N\times p}} \text{tr}(V^T MV) \quad \text{s.t.} \quad V^T V = I \tag{1}$$

where $\text{tr}(\cdot)$ denotes the trace functional. Observe that this is actually an implicit optimization over the Grassmann manifold rather than the Stiefel manifold. This is because, the objective function is invariant to a rotation in $\mathbb{R}^p$ of the decision variables, that is, replacing V with VQ so that $Q \in \mathbb{R}^{p\times p}, Q^T Q = I$, we have that,

$$\text{tr}((VQ)^T M(VQ)) = \text{tr}(Q^T(V^T MV)Q) = \text{tr}(V^T MV)$$

where the second equality is due to the similarity invariance property of the trace functional.

3. Regularized Generalized Eigenvalue Problem (R-GEP)

The Generalized Eigenvalue Problem (GEP) is a very well studied problem, particularly in finite element analysis, control theory, etc. [27]. In computer vision and machine learning, GEP can be used for binary classification [7] and face recognition tasks [8], among others. This problem constitutes the key computational phase of the Heat Kernel Smoothing procedure used in [23] to smooth signals over anatomical surfaces in 3D medical images. A relaxed version of the Normalized cuts problem used widely in image segmentation applications can also be formulated as a GEP [25]. It is expressable as the following numerical optimization problem,

$$\min_{V \in \mathbb{R}^{N \times p}} f(V) := \mathrm{tr}(V^T M V) \quad \text{s.t.} \quad V^T D V = I \tag{2}$$

where the decision variable of the optimization problem V is the matrix containing the first p eigenvectors of the matrix M which are the eigenvectors corresponding to the largest p eigenvalues of the matrix M with respect to another arbitrary matrix D. The pair $\{M, D\}$ is also commonly referred to as the matrix pencil. When D is the identity matrix, this problem reduces to the standard eigenvalue problem hence we note that Principal Component Analysis (PCA) is a special case of this problem by setting M to be the similarity matrix $Y^T Y$. While D can be singular, it is assumed to be a positive definite (p.d) matrix in many applications.

Now, we motivate the regularization part of the problem. Let $n = \{1, ..., N\}$ be the set of subjects and suppose that we are given *supplementary* information for a subset $n' \subseteq n, |n'| = N'$. One can also think of the supplementary information as data procured from more expensive sources. For instance, in our applications some modalities are expensive ($5000+) or may involve invasive procedures so not all participants will opt in. Another example is in various crowd sourced platforms where expert level annotation may be available only for few examples due to high acquisition cost. Let the data associated with n' be $\mathcal{S} \in \mathbb{R}^{s \times N'}$ where s is the number of supplementary features for each subject in n'. The key assumption is that $\mathcal{S}$ contains complementary information which captures the underlying pattern among the subjects, hence helping our primary goal. Practical aspects of this setup are further explained in (5). Let $\Gamma = \mathcal{S}^T \mathcal{S} \in \mathbb{R}^{N' \times N'}$ be the corresponding similarity matrix and $\alpha \in \mathbb{R}^{N'}$ be its leading eigenvector. We can think of the magnitude of coordinates of α as weights on the subjects in n'. Let $V_{\cdot 1} \in \mathbb{R}^N$ denote the first column of V and $V_{\cdot 1|n'}$ be the restriction of $V_{\cdot 1}$ to the set n' (the notation is suppressed when the context is clear). The simplest way to take advantage of the complementary information of α in our model is to use the ℓ_0 norm (which counts the number of nonzero entries) of the *difference* between $V_{\cdot 1}$ and α which seeks fidelity between them while keeping the number of places they are different small. It is well known that this gives us a computationally intractable problem but can be approximated for practical purposes by its best convex surrogate, the ℓ_1 norm. Hence the optimization problem is

$$\min_{V \in \mathbb{R}^{N \times p}} \quad \mathrm{tr}(V^T M V) + \lambda ||V_{\cdot 1} - \alpha||_1 \quad \text{s.t.} \; V^T D V = I \tag{3}$$

where $\lambda > 0$ is the regularization parameter. Even though in principle one can add $|n'|$ regularization terms, this generally does not provide significant improvements as shown empirically (see supplement). In the next section we explain how this optimization problem can be solved efficiently to exploit the structure of the problem. Note that the regularization term in problem (3) is specifically chosen with the application in mind, but the algorithm described in the following section can be used for any nonsmooth function, say $g : \mathbb{R}^{N \times p} \to \mathbb{R}$ with the following properties. We assume that g is a real valued convex (nonsmooth) function on $\{V \in \mathbb{R}^{N \times p} : V^T D V = I\}$ and that at least one element $s_g \in \partial g(V)$ can be computed efficiently for every V in the feasible set. Note that outside of the feasible set we do not have any assumptions on g unlike most projection based algorithms.

4. Algorithm

We solve the optimization problem (3) with a coordinate descent method over the generalized Stiefel manifold. The main intuition of our algorithm is to decrease the function by finding the next iterate along a *curve* that lies in the feasible set. The constraints in (2) and (3) describe a manifold over the decision variables, specifically the generalized Stiefel manifold $\mathrm{GF}_{N,p}$. We can therefore construct curves in this manifold using the exponential map, or constructions such as Cayley curves [32]. In the text below, we describe an algorithm that constructs *descent curves* on the generalized Stiefel manifold. These curves are constructed to have two key properties. First, the curves only vary along a subset of the dimensions/decision variables, so that methods such as coordinate descent can be used to parallelize or reduce the problem [21]. Second, the directional derivative of the objective along the tangent to the curve will be negative, meaning that an iterate chosen from a suitable distance along this curve will have decreased objective values relative to the current iterate.

To simplify calculations, we describe the update steps for the unregularized in problem (2). This can be extended to the regularized problem in (3) by adding the subdifferential of the regularization function to the subdifferential used here.

Algorithm 1 Stochastic coordinate descent on $\mathrm{GF}_{N,p}$

Require: $f : \mathrm{GF}_{N,p} \to \mathbb{R}$, $D \in \mathbb{R}^{N\times N}$, $V_0 \in \mathrm{GF}_{N,p}(D)$

for $t = 1, \ldots, T$ **do**
  Select rows $\mathcal{I} \subseteq \{1, \ldots, N\}$
  $U_0 \leftarrow V_{\mathcal{I}\cdot} - D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}} V_{\bar{\mathcal{I}}\cdot}$
  $[Q\ QR] \leftarrow U_0$ for Q nonsingular
  Take $G \in \partial_{V_{\mathcal{I}}\cdot} f(V)$
  $G' \leftarrow G_{\mathcal{I}\mathcal{J}} + G_{\mathcal{I}\bar{\mathcal{J}}} R^T$
  Construct a descent curve Y on $\mathrm{GF}_{i,p}(D_{\mathcal{I}\mathcal{I}})$ through U_0 in the direction of $-G'$ (22)
  Pick step size τ_t satisfying Armijo-Wolfe condition [18]
  $V_{t+1} \leftarrow Y(\tau_t)$
end for

We start by describing a constructive way of dividing the optimization problem into smaller subproblems while still maintaining the orthogonality constraints with respect to the given positive definite matrix D. Note that if D is the identity matrix this reduces to the usual Stiefel constraints.

Suppose we have a subset $\mathcal{I}$ of i row indices, corresponding to rows of V. The submatrix consisting only of these rows is denoted by $V_{\mathcal{I}\cdot} \in \mathbb{R}^{i\times p}$. We seek to construct a descent curve by reducing (2) to the subproblem over only this submatrix. We are given a feasible iterate V, and seek to compute the next iterate W such that it also lies in the generalized Stiefel manifold $\mathrm{GF}_{N,p}$ and is thus feasible for the problem in (3), and W only differs from V in the rows selected by $\mathcal{I}$. To start, assume w.l.o.g. that $\mathcal{I}$ selects the first i rows of V. Then we write the constraint $V^T D V = I$ as

$$\begin{bmatrix} V_{\mathcal{I}\cdot} \\ V_{\bar{\mathcal{I}}\cdot} \end{bmatrix}^T \begin{bmatrix} D_{\mathcal{I}\mathcal{I}} & D_{\bar{\mathcal{I}}\mathcal{I}}^T \\ D_{\bar{\mathcal{I}}\mathcal{I}} & D_{\bar{\mathcal{I}}\bar{\mathcal{I}}} \end{bmatrix} \begin{bmatrix} V_{\mathcal{I}\cdot} \\ V_{\bar{\mathcal{I}}\cdot} \end{bmatrix} = I. \tag{4}$$

We are interested in the case that the rows *not* selected, with indices in the complement $\bar{\mathcal{I}}$, are fixed. Writing the constraints only the free variable $V_{\mathcal{I}\cdot}$, we have:

$$V_{\mathcal{I}\cdot}^T D_{\mathcal{I}\mathcal{I}} V_{\mathcal{I}\cdot} + V_{\bar{\mathcal{I}}\cdot}^T D_{\bar{\mathcal{I}}\mathcal{I}} V_{\mathcal{I}\cdot} + V_{\mathcal{I}\cdot}^T D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot} + V_{\bar{\mathcal{I}}\cdot}^T D_{\bar{\mathcal{I}}\bar{\mathcal{I}}} V_{\bar{\mathcal{I}}\cdot} = I. \tag{5}$$

On the subproblems, it will be sufficient to choose new iterates which preserve the equality. This is a general quadratic equality constraint, so it will be more difficult than a Stiefel constraint. Note that this constraint also includes rows *not* in the selected set, i.e., $V_{\bar{\mathcal{I}}\cdot}$. However, we can ignore rows which are not neighbors of $\mathcal{I}$ in the graph representation of nonzeros of D. As a result, when D is sparse, this computations below will still be of order $\ll N$.

The constraint on $V_{\mathcal{I}\cdot}$ will be of the form

$$V_{\mathcal{I}\cdot}^T D_{\mathcal{I}\mathcal{I}} V_{\mathcal{I}\cdot} + V_{\bar{\mathcal{I}}\cdot}^T D_{\bar{\mathcal{I}}\mathcal{I}} V_{\mathcal{I}\cdot} + V_{\mathcal{I}\cdot}^T D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot} = P_1 \tag{6}$$

for a matrix P_1 that is constant w.r.t. $V_{\mathcal{I}\cdot}$. If we assume that $D_{\mathcal{I}\mathcal{I}}$ is full-rank, we can complete the square:

$$\left(D_{\mathcal{I}\mathcal{I}}^{\frac{1}{2}} V_{\mathcal{I}\cdot} + D_{\mathcal{I}\mathcal{I}}^{-\frac{1}{2}} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot} \right)^T \left(D_{\mathcal{I}\mathcal{I}}^{\frac{1}{2}} V_{\mathcal{I}\cdot} + D_{\mathcal{I}\mathcal{I}}^{-\frac{1}{2}} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot} \right) = P \tag{7}$$

where the matrix $P = P_1 + V_{\bar{\mathcal{I}}\cdot}^T D_{\bar{\mathcal{I}}\mathcal{I}} D_{\mathcal{I}\mathcal{I}}^{-1} D_{\mathcal{I}\bar{\mathcal{I}}} V_{\bar{\mathcal{I}}\cdot}$ is still constant with respect to the selected submatrix.

Note that $D \succ 0$ implies $D_{\mathcal{I}\mathcal{I}} \succ 0$, so we can assume the inverse matrices above exist when D is positive definite.

We next describe the constraints over subproblem decision matrix U. If we take any orthogonal U, and say

$$V_{\mathcal{I}\cdot} = D_{\mathcal{I}\mathcal{I}}^{-\frac{1}{2}} U P^{\frac{1}{2}} - D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot}, \tag{8}$$

this provides a new iterate that satisfies the constraints in (4) and subsequent equations.

The descent curve will then be computed around the point:

$$U_0 = \left(D_{\mathcal{I}\mathcal{I}}^{\frac{1}{2}} V_{\mathcal{I}\cdot} + D_{\mathcal{I}\mathcal{I}}^{-\frac{1}{2}} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot} \right) P^{-\frac{1}{2}} \tag{9}$$

given V is the previous iterate. Here, we note that for the regularized problem (3), we simply add $\lambda \mathrm{sign}(V_{\cdot 1} - \alpha)$ to the first column of the subdifferential.

4.1. Alternate Form

The previous derivation provides the most general means to construct the subproblem over U, and would be used e.g., if the chosen descent curve is a geodesic constructed from the exponential map of a subgradient around U_0. We can in general perform optimization on this subproblem using any choice of *retraction*. This is a general class of mappings from the tangent space of a manifold to the manifold and preserves the key properties of the exponential function necessary to perform feasible descent on a manifold, for more details, see [1]. A computationally efficient retraction on the Stiefel manifold is given by the Cayley transform. A form of this transformation suitable for the generalized Stiefel manifold is given by Equation (1.2) and Lemma 4.1 of [32]. This allows us to eliminate the potentially expensive computation of matrix square roots. Here we would instead consider

$$V_{\mathcal{I}\cdot} = U - D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot}, \tag{10}$$

which will satisfy the constraint in (5) if $U^T D_{\mathcal{I}\mathcal{I}} U = P$. Note the construction from [32] still assumes that $D_{\mathcal{I}\mathcal{I}} \succ 0$ and P is nonsingular.

4.2. Singularity Correction

We can relax the assumptions in the above subproblem construction, in that we do not necessarily require that the constraint matrix P to be nonsingular. This section describes a transformation of the subproblem that allows us to consider singular P.

First rewrite (7) as:

$$\left(V_{\mathcal{I}\cdot} + D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot}\right)^T D_{\mathcal{I}\mathcal{I}} \left(V_{\mathcal{I}\cdot} + D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot}\right) = P. \tag{11}$$

Assume, as above, that $D_{\mathcal{I}\mathcal{I}} \succ 0$. Then for any matrix that satisfy this equation, P will be nonsingular iff $V_{\mathcal{I}\cdot} + D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot}$ is nonsingular. We achieve the "singularity correction" by transforming the subproblem into a problem over only a maximal set of linearly independent columns of the latter matrix. Assume w.l.o.g. that

$$V_{\mathcal{I}\cdot} + D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot} = \begin{bmatrix} Q & QR \end{bmatrix} \tag{12}$$

for a $i \times r$ nonsingular matrix Q and a $r \times (p-r)$ matrix R. Let $\mathcal{J}$ be the indices of the columns corresponding to Q. Then

$$Q = V_{\mathcal{I}\mathcal{J}} + D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\mathcal{J}} \tag{13}$$

and

$$\begin{bmatrix} Q & QR \end{bmatrix}^T D_{\mathcal{I}\mathcal{I}} \begin{bmatrix} Q & QR \end{bmatrix} = \begin{bmatrix} Q^T D_{\mathcal{I}\mathcal{I}} Q & Q^T D_{\mathcal{I}\mathcal{I}} QR \\ R^T Q^T D_{\mathcal{I}\mathcal{I}} Q & R^T Q^T D_{\mathcal{I}\mathcal{I}} QR \end{bmatrix} = P.$$

Taking R to be fixed, and expressing the constraint only on the submatrix Q of linearly independent columns, we can expect the equality to be true iff

$$\left(V_{\mathcal{I}\mathcal{J}} + D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\mathcal{J}}\right)^T D_{\mathcal{I}\mathcal{I}} \left(V_{\mathcal{I}\mathcal{J}} + D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\mathcal{J}}\right) = P_{\mathcal{J}\mathcal{J}}. \tag{14}$$

So given $U \in \mathbb{R}^{i \times r}$ such that $U^T D_{\mathcal{I}\mathcal{I}} U = P_{\mathcal{J}\mathcal{J}}$, we let

$$V_{\mathcal{I}\mathcal{J}} = U - D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\mathcal{J}}, \tag{15}$$

$$V_{\mathcal{I}\bar{\mathcal{J}}} = UR - D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\bar{\mathcal{J}}}. \tag{16}$$

We now show feasibility after performing the singularity correction.

Lemma 1. *With the above notations, $V_{\mathcal{I}\cdot}$ constructed is feasible.*

Proof. The proof consists of simple linear algebraic calculations, that is, first observe that,

$$\begin{aligned} &V_{\mathcal{I}\mathcal{J}} + D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\cdot} \\ &= \begin{bmatrix} U - D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\mathcal{J}} & UR - D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\bar{\mathcal{J}}} \end{bmatrix} \\ &\quad + \begin{bmatrix} D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\mathcal{J}} & D_{\mathcal{I}\mathcal{I}}^{-1} D_{\bar{\mathcal{I}}\mathcal{I}}^T V_{\bar{\mathcal{I}}\bar{\mathcal{J}}} \end{bmatrix} = \begin{bmatrix} U & UR \end{bmatrix}. \end{aligned}$$

Now it is enough to show that this block matrix produces P when multiplied with the square matrix $D_{\mathcal{I}\mathcal{I}}$ as

$$\begin{bmatrix} U^T \\ R^T U^T \end{bmatrix} D_{\mathcal{I}\mathcal{I}} \begin{bmatrix} U & UR \end{bmatrix} = \begin{bmatrix} U^T D_{\mathcal{I}\mathcal{I}} U & U^T D_{\mathcal{I}\mathcal{I}} UR \\ R^T U^T D_{\mathcal{I}\mathcal{I}} U & R^T U^T D_{\mathcal{I}\mathcal{I}} UR \end{bmatrix} = \begin{bmatrix} P_{\mathcal{J}\mathcal{J}} & P_{\mathcal{J}\mathcal{J}} R \\ R^T P_{\mathcal{J}\mathcal{J}} & R^T P_{\mathcal{J}\mathcal{J}} R \end{bmatrix} = P.$$

□

As a footnote, while we can allow P to be singular, it is still necessary for the correctness of our method that $D_{\mathcal{I}\mathcal{I}} \succ 0$ for any choice of $\mathcal{I}$. However, it is sufficient to show that $D \succ 0$:

$$\mathbf{x}^T D_{\mathcal{I}\mathcal{I}} \mathbf{x} \overset{\text{w.l.o.g.}}{=} \begin{bmatrix} \mathbf{x} \\ 0 \end{bmatrix}^T \begin{bmatrix} D_{\mathcal{I}\mathcal{I}} & D_{\bar{\mathcal{I}}\mathcal{I}}^T \\ D_{\bar{\mathcal{I}}\mathcal{I}} & D_{\bar{\mathcal{I}}\bar{\mathcal{I}}} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ 0 \end{bmatrix} \geq 0. \tag{17}$$

This derivation therefore produces valid subproblems of (2) as long as the constraint matrix D is positive definite.

4.3. Computing a Descent Curve

A descent direction for the subproblem will come from differentiating $f \circ V(U)$ w.r.t. U, where V is related to U by (8):

$$\frac{\partial}{\partial U} f \circ V(U) = 2(D_{\mathcal{I}\cdot} V_{\cdot\mathcal{J}} + (D_{\mathcal{I}\mathcal{I}} V_{\mathcal{I}\mathcal{J}} R + D_{\mathcal{I}\bar{\mathcal{I}}} V_{\bar{\mathcal{I}}\bar{\mathcal{J}}}) R^T) P^{1/2}. \tag{18}$$

We pick a subgradient $G \in \partial_{V_{\mathcal{I}\cdot}} f \circ V(U)$ and then perform the singularity correction on G with the same R in (12):

$$G' = G_{\mathcal{I}\mathcal{J}} + G_{\mathcal{I}\bar{\mathcal{J}}} R^T. \tag{19}$$

To generate a descent curve, we can project a subgradient of $f \circ W$ onto the tangent space of the manifold $\mathrm{GF}_{i,p}(D_{\mathcal{I}\mathcal{I}})$ at U_0, where W is the next feasible point for any orthonormal U such that

$$W(U) = \begin{bmatrix} UP^{1/2} & UP^{1/2} R \\ V_{\bar{\mathcal{I}}\mathcal{J}} & V_{\bar{\mathcal{I}}\bar{\mathcal{J}}} \end{bmatrix} \in \mathbb{R}^{N \times p} \tag{20}$$

assuming w.l.o.g. that $\mathcal{I}$ selects the first $|\mathcal{I}|$ rows of the matrix. This construction preserves the constraints while leaving the complement $\bar{\mathcal{I}}$ unchanged, so it is clear that W is also feasible. Then, a skew-symmetric matrix is defined as

$$A = G' U_0^T - U_0 G'^T, \tag{21}$$

and the curve Y as a function of τ by the Crank-Nicolson-like design as in [32] is

$$Y(\tau) = \left(I + \frac{\tau}{2} A D_{\mathcal{I}\mathcal{I}}\right)^{-1} \left(I - \frac{\tau}{2} A D_{\mathcal{I}\mathcal{I}}\right) U_0. \tag{22}$$

So one can think of Y as a function of a single parameter τ on which we perform a linear search over the descent curve with sufficient decrease in the objective value in each iteration.

Theorem 2. *Let $F := f + g$ and V_t be a point V at iteration t. $F(V_t)$ is a monotonically nonincreasing sequence for (3) and hence for (2).*

Proof. Note that from lemma (1), at every iteration t we produce a feasible point and from section (4.3) they satisfy the strong Wolfe conditions. Combining both gives us the desired result. □

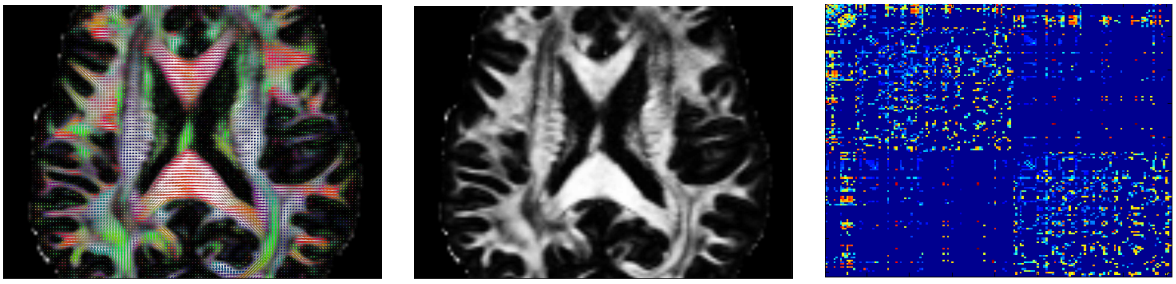

Figure 1: DTI image showing tensor directionality, followed by the FA image and the connectivity matrix.

5. Experiments

Figure 1 shows a slice of an example pair of a DTI image, the corresponding FA image and the connectivity matrix (with 160 regions of interest).

Our experiments evaluate the efficacy of R-GEP in fusing multiple sources via measuring performance improvement for downstream statistical analysis tasks. We also discuss about running time for Alg. 1.

5.1. Data

The dataset for our experiments is comprised of brain imaging data, cognitive test scores and other demographic data from 102 middle-aged and older adults. In this cohort, 58 of the subjects are healthy (according to a dementia rating scale [17]), while the rest are diseased. Recall that the data used in our model come from three sources. The primary source is 3D volumetric Fractional Anisotropy (FA) imaging data, while the single secondary source is connectivity information derived from the corresponding 3D Diffusion Tensor Images (DTI). For each voxel in the brain image space, a DTI image provides the rate and directionality of diffusion of water. The two sources are related in the sense that FA summarizes the degree of diffusion of water within each voxel (i.e., 3D pixel) of a DTI. However, there is information loss in this summarization, and hence using DTI-derived connectivity information as a secondary source for any statistical analysis performed in FA space is expected to increase the statistical power. Using the DTI data and performing a pre-processing step such as tractography, one can construct a connectivity matrix that corresponds to an adjacency graph where the nodes represent anatomical regions of interest and the edges weights (non-negative) give the strength of their connection (e.g., derived using fiber counting procedures [19]).

Note that the secondary source in this case is a third order tensor where each slice i corresponds to a subject's adjacency matrix. Using Canonical Polyadic decomposition [13] on this tensor, we can then compute the subject space factor matrix $C^{N\times r}$, where r represents the tensor decomposition rank. The resulting factor matrix C will respect the structure of the adjacency graph, and hence the mass matrix D in (3) is given by CC^T. The incomplete priors n' from which we derive α for the regularization term as described in Section 3, include 7 different cerebrospinal fluid (CSF) scores that measure specific types of protein levels in the brain that may be related to the disease [30]. These measures are positive scalars and are generally available for a smaller subset of the cohort (in our case, 60 out of 102) because it is a relatively more involved procedure.

5.2. Evaluations setup

Our evaluations are two-fold. Recall that the embeddings V learned by our model in (3) should, as a first order requirement, retain the structural and group-level characteristics of the input data, for example, the healthy versus diseased discrimination power. If such sanity checks are satisfied, we can evaluate improvements obtained in downstream statistical analysis. Therefore, using V as the feature representations for the inputs, we first check for changes in our ability to classify the healthy versus diseased subjects using an off the shelf machine learning library.

The comparison is performed against three models of incremental complexities. First, we compare the results to a baseline model which relies *only* on the primary source/view (FA data). Second, we also compare the results to 'intermediate' models that include a PCA based approach on FA data and a GEP (2) setup which does not use any regularizer. Lastly, a PCA-avg model is also evaluated where the primary and secondary source kernel are averaged. See supplement for the extended versions of Table 1 and 2.

We further repeat the same set of experiments to evaluate the power of these representations in replicating the disease progression. This is achieved via regressing the representations using existing disease markers as an outcome/dependent variable (example, a cognitive score like MMSE [5]). We used linear-SVM for both classification and regression setups. For the baseline model, the input features are FA and for the other models, the inputs are V. All results are 10-fold cross validated.

5.3. Results

Table 1 and 2 present the classification and regression results respectively. In either case, the rows correspond to PCA rank (p). The columns represent the baseline model,

p	Baseline	PCA	PCA-avg	GEP 2			R-GEP 3		
				$r=1$	5	10	$r=1$	5	10
3	63.4	85.7	85.6	86.6	85.7	85.7	**90.3**	88.5	86.5
5	63.4	82.6	81.8	85.4	85.4	86.3	**89.5**	89.3	87.3
7	63.4	84.3	80.7	84.3	84.3	84.3	86.4	**88.4**	87.5
10	63.4	82.4	83.5	82.4	84.2	86.2	86.5	86.4	**89.3**
13	63.4	83.3	85.6	86.2	84.2	88.1	89.2	**91.2**	88.2

Table 1: Healthy versus diseased classification accuracy (10-fold cross validated) using GEP and R-GEP, compared to the baseline linear classifier and the PCA setup. p denotes the PCA rank and r is tensor rank.

p	Baseline	PCA	PCA-avg	GEP 2			R-GEP 3		
				$r=1$	5	10	$r=1$	5	10
3	0.679	0.718	0.647	0.719	0.718	0.718	0.745	**0.771**	0.758
5	0.679	0.719	0.614	0.726	0.737	0.735	**0.769**	0.746	0.749
7	0.679	0.707	0.610	0.707	0.707	0.713	0.763	**0.785**	0.734
10	0.679	0.656	0.622	0.656	0.742	0.719	0.741	**0.762**	0.754
13	0.679	0.717	0.654	0.730	**0.765**	0.745	0.737	0.757	0.754

Table 2: Healthy versus diseased regression correlation coefficient (10-fold cross validated) using GEP and R-GEP, compared to the baseline linear classifier and the PCA setup. p denotes the PCA rank and r is tensor rank.

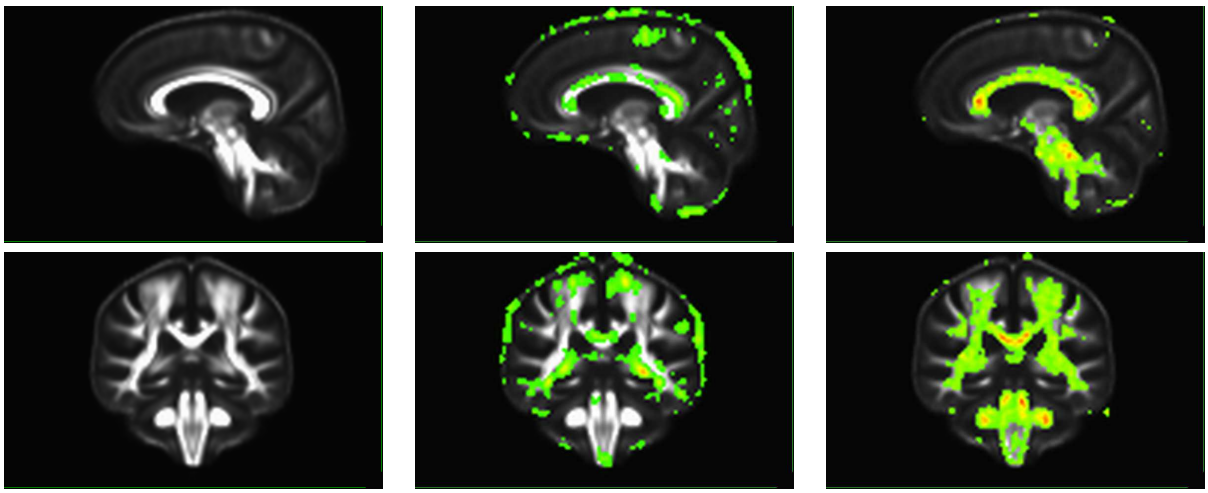

Figure 2: Feature sensitivity. First column shows the FA image. Second column shows overlays of the weights assigned by baseline linear kernel on this FA image. Last column shows overlays from the base R-GEP case in Table 1. Green (Red) corresponds to smaller (larger) weights.

PCA on primary source, PCA on primary and secondary source (PCA-avg) and the GEP and R-GEP models (with different choices of tensor decomposition rank r). Refer to the supplement for an expanded version of these tables. It is clear from the accuracy results in Table 1 that introducing additional sources of information always increases the performance (63.4% accuracy for baseline to $> 91\%$ for R-GEP). Same is the case with regression results in Table 2 (0.68 correlation coefficient from baseline to > 0.78 for R-GEP). R-GEP outperforms the rest (especially GEP) across multiple choices of p and r. These trends support the hypothesis that our incomplete priors (CSF measures) are predictive of the disease [9]. It is interesting to see that even when only the primary source is used, the performance improves from baseline to PCA (second to third columns), which is perhaps due to nature of the imaging data itself.

As the length of the embeddings p increases, both the accuracies and correlations for the R-GEP model are not necessarily monotonic. This implies that for the statistical task of interest (e.g., discriminating healthy versus diseased in Table 1), there may be a 'sweet spot' for p. Smaller values of p seem to perform better. It should be noted that all these interpretations are sensitive to the number of data instances, the specific choices of data sources, and the chosen task at

hand. The results for GEP and R-GEP (last six columns in Tables 1 and 2) for a given p show that the performance changes only marginally (in most of the cases) for different t. More precisely, there seems to be no single t which gives best set of accuracies and/or correlations across all the p. This is ideal because r is not an outcome of the model, and it only governs the way we compute the mass matrix. Note that the two sources do provide complementary information, which can be seen by the performance differences of the PCA-avg model to that of the PCA.

An interesting exploratory tool is to compute the sensitivity (or weight) of each feature (or voxel) in classifying the healthy versus diseased subjects. Computing these weights is straight forward for the baseline case since it corresponds to a linear SVM. However, for R-GEP the feature space is V and not the voxel space, see Figure 1. We used a trick from [20], where results from a SVM method can be used to assess sensitivities in the original feature space. Figure 2 shows two pairs of these feature sensitivity maps of the baseline model to the *best* case of R-GEP in the classification case. Sensitivity of a voxel is proportional to the absolute value of the weights (here, green is smaller and red is larger). The regions selected by R-GEP are different from the baseline, and more importantly, R-GEP assigned weights more contiguously compared to the baseline. It should be noted that the baseline is a simple linear SVM and so unsatisfactory sensitivity maps are expected. These results support the premise that incorporating secondary and incomplete priors increase performance, and our R-GEP model combines these information sources in a meaningful way offering good improvements. Additional experiments using positron emission tomography (PET) images from a study on pre-clinical Alzheimer's disease are available on the project webpage.

We note that there is a broad spectrum of ways in which information from disparate sources can be combined, e.g., multiple kernel learning with data imputation for incomplete features [11]. The purpose of these experiments is not to claim that the proposed ideas are the *best* means for multi-view data fusion. Instead, the experiments suggest that independent of which statistical machinery we choose to deploy, methods such as the one presented here can be used as a pre-processing step to harmonize information across the views to construct meaningful low dimensional embeddings that can then be fed to the downstream analysis.

5.4. Discussion

Table 3 shows the runtime of Alg. 1 versus the condition number (denoted by κ) of D. We note two aspects of our algorithm. Firstly, as the problem size N increases, the increase in the runtime is not significant implying that the algorithm is scalable to large datasets. Secondly, we see that κ has a significant impact on the convergence (Table 3). Intuitively, this means that when the data matrix consists of points that are similar in some sense, κ of the similarity matrix induced increases. In these cases, as expected, finding a *good* descent direction becomes harder, and we tend to make very little progress towards the optimal (local) solution at each iteration. Recall that this issue is very common in most numerical optimization algorithms, and the solution involves applying either standard (or specialized) preconditioning techniques (refer to [18]). The results presented here do not utilize any preconditioning. For reasonable values of κ, the runtime scales approximately linearly. For $\kappa = 3$, the solver returns the correct solution for $N = 1000$ in $\approx$ 5s, $N = 5000$ in $\approx$ 2min and $N = 10000$ in $\approx$ 7min.

Condition	Problem size (N)			
number κ	10	30	50	100
1	0.04	0.06	0.27	0.46
5	0.04	2.91	36.18	91.5
10	0.05	8.00	71.55	514.2
20	0.35	75.86	324.2	>1000

Table 3: Effect of condition number κ on the runtime (in seconds) of Alg. 1.

6. Conclusion

This paper describes a manifold optimization framework to obtain solutions to generalized eigenvalue problems with a nonsmooth regularizer. Given (i) the numerous problems in vision that involve GEP and (ii) a practical need to incorporate various forms of meta knowledge or supervision into such formulations, our algorithm addresses an important gap where few alternatives are available currently. As long as the inputs are well conditioned, the method is scalable and efficient. We show a concrete application to a brain imaging problem where the framework helps improve standard statistical machine learning experiments which seek to utilize diverse types of imaging modalities for disease diagnosis. In this case, incorporating a nonsmooth regularizer has the direct consequence that it yields higher sensitivity/specificity and arguably more interpretable visual results. Our solver can be used in a plug and play manner in various other settings in vision where a regularization is expected to meaningfully bias and improve the performance. The extended version of this paper, the supplementary material and the code are available at http://pages.cs.wisc.edu/~sjh/.

7. Acknowledgment

SJH was supported by a University of Wisconsin CIBM fellowship (5T15LM007359-14). We acknowledge support from NIH grants AG040396 and AG021155, NSF RI 1116584 and NSF CAREER award 1252725, as well as UW ADRC (AG033514), UW ICTR (1UL1RR025011), Waisman Core grant (P30 HD003352-45), UW CPCP (AI117924) and NIH grant AG027161.

References

[1] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009. 2, 4

[2] P. Arbenz, D. Kressner, and D.-M. E. Zürich. Lecture notes on solving large scale eigenvalue problems. *D-MATH, EHT Zurich*, 2012. 2

[3] M. D. Collins, J. Liu, J. Xu, L. Mukherjee, and V. Singh. Spectral clustering with a convex regularizer on millions of images. In *ECCV*, pages 282–298. Springer, 2014. 1

[4] I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *ACM SIGKDD*, pages 269–274. ACM, 2001. 1

[5] M. F. Folstein, S. E. Folstein, and P. R. McHugh. mini-mental state: a practical method for grading the cognitive state of patients for the clinician. *Journal of psychiatric research*, 12(3):189–198, 1975. 6

[6] M. Gönen and E. Alpaydın. Multiple kernel learning algorithms. *The Journal of Machine Learning Research*, 12:2211–2268, 2011. 1

[7] M. R. Guarracino, C. Cifarelli, O. Seref, et al. A classification method based on generalized eigenvalue problems. *Optimisation Methods and Software*, 22(1):73–81, 2007. 3

[8] J. Gui, Z. Sun, W. Jia, et al. Discriminant sparse neighborhood preserving embedding for face recognition. *Pattern Recogn.*, 45(8):2884–2893, 2012. 3

[9] O. Hansson, H. Zetterberg, P. Buchhave, et al. Association between CSF biomarkers and incipient Alzheimer's disease in patients with mild cognitive impairment: a follow-up study. *The Lancet Neurology*, 5(3):228–234, 2006. 7

[10] X. He, S. Yan, Y. Hu, et al. Face recognition using Laplacianfaces. *IEEE PAMI*, 27(3):328–340, 2005. 2

[11] C. Hinrichs, V. Singh, G. Xu, et al. Predictive markers for AD in a multi-modality framework: an analysis of MCI progression in the ADNI population. *Neuroimage*, 55(2):574–589, 2011. 8

[12] W. H. Kim, M. K. Chung, and V. Singh. Multi-resolution shape analysis via non-euclidean wavelets: Applications to mesh segmentation and surface alignment problems. In *CVPR*, pages 2139–2146. IEEE, 2013. 2

[13] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009. 6

[14] A. Kovnatsky, M. M. Bronstein, A. M. Bronstein, et al. Coupled quasi-harmonic bases. In *Computer Graphics Forum*, volume 32, pages 439–448. Wiley Online Library, 2013. 2

[15] S. Maji, N. K. Vishnoi, and J. Malik. Biased normalized cuts. *CVPR*, 0:2057–2064, 2011. 2

[16] G. Monaci, P. Jost, P. Vandergheynst, et al. Learning multimodal dictionaries. *IEEE Trans. on Image Processing*, 16(9):2272–2283, 2007. 1

[17] J. C. Morris. Clinical dementia rating: a reliable and valid diagnostic and staging measure for dementia of the Alzheimer type. *International psychogeriatrics*, 9(S1):173–176, 1997. 6

[18] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006. 4, 8

[19] A. Raj, A. Kuceyeski, and M. Weiner. A network diffusion model of disease progression in dementia. *Neuron*, 73(6):1204–1215, 2012. 6

[20] P. M. Rasmussen, K. H. Madsen, T. E. Lund, et al. Visualization of nonlinear kernel models in neuroimaging by sensitivity maps. *NeuroImage*, 55(3):1120–1131, 2011. 8

[21] P. Richtárik and M. Takáč. Parallel coordinate descent methods for big data optimization. *Mathematical Programming*, pages 1–52. 3

[22] B. Schölkopf, A. Smola, and K. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319, 1998. 2

[23] S. Seo, M. K. Chung, and H. K. Vorperian. Heat kernel smoothing using Laplace-Beltrami eigenfunctions. In *MICCAI*, pages 505–512. Springer, 2010. 3

[24] A. Sharma, A. Kumar, H. Daume, and D. W. Jacobs. Generalized multiview analysis: A discriminative latent space. In *CVPR*, pages 2160–2167. IEEE, 2012. 1

[25] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE PAMI*, 22(8):888–905, 2000. 3

[26] M. Sugiyama. Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis. *The Journal of Machine Learning Research*, 8:1027–1061, 2007. 1

[27] F. Tisseur and K. Meerbergen. The quadratic eigenvalue problem. *SIAM review*, 43(2):235–286, 2001. 3

[28] D. Vanderbilt. Soft self-consistent pseudopotentials in a generalized eigenvalue formalism. *Physical Review B*, 41(11):7892, 1990. 1

[29] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007. 1

[30] A. Wallin, K. Blennow, N. Andreasen, et al. CSF biomarkers for Alzheimer's Disease: levels of beta-amyloid, tau, phosphorylated tau relate to clinical symptoms and survival. *Dementia and geriatric cognitive disorders*, 21(3):131–138, 2005. 6

[31] Y. Weiss, A. Torralba, and R. Fergus. Spectral hashing. In *NIPS*, pages 1753–1760, 2009. 2

[32] Z. Wen and W. Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1-2):397–434, 2013. 3, 4, 5

[33] M. White, X. Zhang, D. Schuurmans, et al. Convex multiview subspace learning. In *NIPS*, pages 1673–1681, 2012. 1

[34] H. Zhang, O. van Kaick, and R. Dyer. Spectral methods for mesh processing and analysis. In *Proceedings of Eurographics*, 2007. 2