Supplemental material to High-order Tensor Regularization with Application to Attribute Ranking

Kwang In Kim
University of Bath

Juhyun Park
Lancaster University

James Tompkin
Brown University

This supplemental material provides a brief introduction to vectors and tensors on Riemannian manifolds (Sec. 1); compares the ranking algorithms addressed in the main paper from the manifold regularization perspective to highlight challenges on high-order tensor regularization (Sec. 2); demonstrates the approximation capability of our low-rank ranking matrix representation (Sec. 3); presents details of reconstructing linear rankings from the estimated ranking matrices (Sec. 4); and presents additional experimental results (Sec. 5). Some contents from the main paper are reproduced so that this document is self-contained.

1. Vectors & tensors on Riemannian manifolds

A d-dimensional (real topological) manifold M is a space that *looks* locally Euclidean. That is, at each point p, there is an open neighborhood U containing p, and a continuous map x with a continuous inverse to an open subset $\tilde{U}$ of Euclidean space $\mathbb{R}^d$.[1]

The pair (U, x) is called a *coordinate chart*. A coordinate chart provides *representations* of points on $U \subset M$ to facilitate numerical operations. In general, multiple coordinate charts include a single point in their domains, with each providing its own coordinate representation:

$$\begin{aligned} x(p) &\sim (x^1(p), \dots x^d(p)), \\ y(p) &\sim (y^1(p), \dots y^d(p)). \end{aligned} \tag{1}$$

An atlas $\{(U_\alpha, x_\alpha)\}$ is a family of charts where $\{U_\alpha\}$ constitute an open covering of M. By selecting two coordinate charts (U_α, x_α) and (U_β, x_β) from an atlas and combining the corresponding chart maps x_α and x_β, one can define a *chart transition* as a map on $\mathbb{R}^d$:

$$x_\beta \circ x_\alpha^{-1} : x_\alpha(U_\alpha \cap U_\beta) \to x_\beta(U_\alpha \cap U_\beta). \tag{2}$$

An atlas is *smooth* (or infinitely differentiable) if all compatible ($U_\alpha \cap U_\beta \neq 0$) chart transitions are infinitely differentiable (or of class C^∞). A topological manifold M is smooth if the union of all possible atlas on M is smooth.

[1] In addition to the local Euclidean structure, a topological M further satisfies the conditions of being *Hausdorff* and *second countable*, which are automatically met when M is presented as a submanifold of a Euclidean space.

At each point p in a d-dimensional differentiable manifold M, the tangent space $T_p(M)$ is a d-dimensional vector space defined as the set of equivalence classes of curves passing through p on M: Two curves are equivalent when they are tangent at p. This equivalence class is represented by a vector $Y \in T_pM$ tangent to these curves. Given such a (geometric) vector Y, one could uniquely define a partial derivative operator that takes the derivative of an input function along the direction of Y. This establishes a connection between a vector and a directional derivative operator.

From now on, we will assume that all discussed functions are smooth, i.e., $f \in C^\infty(M)$. We will denote the union of T_pM over $p \in M$ as TM, which is called the *vector bundle*, and the dual of T_pM over $p \in M$ as T^*M, which is called the *covector bundle*. The term *vector* is used to denote an element of a tangent space T_pM, as well as an element in a vector bundle TM. In the latter case, it is also called a vector field. We will adopt this convention for tensors as well. We refer the reader to textbooks on this topic for a more systematic introduction [6, 7].

Given a coordinate representation $(x^1, \dots, x^d)$ around a point $p \in M$, we can represent a vector $Y \in TM$ as a first-order tensor:

$$(y^1, \dots, y^d) \leftrightarrow Y = \sum_{i=1,\dots,d} y^i \partial_i, \tag{3}$$

where the partial derivative operators $\{\partial_i := \frac{\partial}{\partial x^i}\}_{i=1}^d$ constitute a basis of TM at the vicinity of p. Similarly, a dual vector (or *covector*) $\omega \in T^*M$ is represented as:

$$(\omega_1, \dots, \omega_d) \leftrightarrow \omega = \sum_{i=1,\dots,d} \omega_i dx^i, \tag{4}$$

with the dual basis:

$$\{dx^i : dx^i(\partial_j) = \delta_j^i\}_{i=1}^d. \tag{5}$$

Using the duality (Eq. 5), we can identify a vector $Y \in TM$ with a linear function on T^*M:

$$Y \leftrightarrow Y(\cdot) = \sum_{i=1,\dots,d} y^i \partial_i(\cdot). \tag{6}$$

Similarly, a covector $\omega \equiv \omega(\cdot)$ can be regarded as a linear function on TM. Furthermore, as shown shortly, on a Riemannian manifold (M, g), we can establish a direct correspondence between a vector and a covector (a linear function on TM) using the metric g:

$$Y \in TM \leftrightarrow g(Y, \cdot) \in T^*M, \tag{7}$$

as $g(Y, \cdot)$ is a linear function on TM.

Now, we can build higher-order tensors upon the vector and covector bundle structure. For instance, the *Riemannian metric* g, as a second order tensor, is an element of $T^*M \times T^*M$, and it is represented in coordinates (x) as:[2]

$$g = \sum_{ij=1,\dots,d} g_{ij} dx^i \otimes dx^j, \tag{8}$$

where $\otimes$ is the tensor product.

The Riemannian metric introduces an *inner-product* on TM and T^*M and it enables measuring the length $\|Y\|_g$ of a vector $Y = \sum_i y^i \partial_j \in T_pM$:

$$\begin{aligned} \|Y\|_g^2 &:= \langle Y, Y \rangle = g(Y, Y) \\ &= \sum_{ij=1,\dots,d} g_{ij} dx^i(Y) \otimes dx^j(Y) \\ &= \sum_{ij=1,\dots,d} g_{ij} y^i y^j. \end{aligned} \tag{9}$$

This inner-product structure can be extended to higher-order tensors, e.g., an inner-product of two second-order tensors $h = \sum_{ij} h^{ij} \partial_i \otimes \partial_i$ and $q = \sum_{ij} q^{ij} \partial_i \otimes \partial_i$ is defined as [7]:

$$\langle h, g \rangle_g = \sum_{ijkl=1}^{d} g_{ij} g_{kl} h^{ik} q^{jl}. \tag{10}$$

This construction allows us to define the symmetric metric integral (Eqs. 6-7 of the main paper) and the induced anti-symmetric tensor integral (Eq. 9 of the main paper). Based on them, we introduce our high-order tensor extension of the harmonic regularization energy (Eq. 11 of the main paper).

2. Comparison of RankSVM, semi-supervised RankSVM, and transductive ranking.

Having constructed the notion of Riemannian manifolds, and vectors and tensors that lie on the bundle structure of a manifold, we are now ready to discuss traditional regularization techniques applied to (zeroth-order) functions, and why it is not straightforward to extend these techniques to tensors.

[2]Equivalently, g is a bilinear function on $TM \times TM$. Further, there are three other types of second order tensors as defined as elements of $TM \times TM$, $T^*M \times TM$, and $TM \times T^*M$, respectively. On Riemannian manifolds, they are all equivalent due to the duality induced by the metric g, Eq. 7.

Suppose that we are given a set of data points $\mathcal{X} = \{\mathbf{x}(1), \dots, \mathbf{x}(n)\} \subset \mathbb{R}^m$, along with pairwise inequality and equality relationships, respectively, $\mathcal{P} = \{(i, j)\}$ and $\mathcal{O} = \{(i, j)\}$, where $(i, j) \in \mathcal{P}$ implies that the rank of i-th data point is higher than j-th data point (as denoted as $\text{Rank}(\mathbf{x}(i)) > \text{Rank}(\mathbf{x}(j))$). Similarity, $(i, j) \in \mathcal{O}$ means $\text{Rank}(\mathbf{x}(i)) = \text{Rank}(\mathbf{x}(j))$.

RankSVM (RS). In the original Relative Attributes work [9], the desired ordering is obtained by applying an ordering function $f : \mathbb{R}^m \to \mathbb{R}$ to $\mathcal{X}$:

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}, \tag{11}$$

where the parameter vector $\mathbf{w}$ is estimated as the minimizer of the *RankSVM* (RS) energy functional [2, 9]:

$$\mathcal{E}_{RS}(\mathbf{w}) = \sum_{(i,j)\in\mathcal{P}} l_P(\mathbf{f}_i - \mathbf{f}_j) + \sum_{(i,j)\in\mathcal{O}} l_O(\mathbf{f}_i - \mathbf{f}_j) + \lambda \|\mathbf{w}\|^2, \tag{12}$$

where λ is a hyper-parameter. The inequality loss l_P and equality loss l_O are given respectively as

$$l_P(a) = \max(0, 1 - a)^2 \text{ and } l_O(a) = a^2, \tag{13}$$

while other loss (or inverse likelihood) functions are also possible.

Semi-supervised RankSVM (SSR). This extension of RankSVM can be obtained by replacing the ambient regularizer ($\|\mathbf{w}\|^2$ in Eq. 12) with a manifold regularizer:

$$\begin{aligned} \mathcal{E}_{SSR}(\mathbf{w}) = &\sum_{(i,j)\in\mathcal{P}} l_P(\mathbf{f}_i - \mathbf{f}_j) \\ &+ \sum_{(i,j)\in\mathcal{O}} l_O(\mathbf{f}_i - \mathbf{f}_j) + \lambda \mathbf{f}^\top L \mathbf{f}, \end{aligned} \tag{14}$$

where $\mathbf{f} = f_{|\mathcal{X}}$ with f given as Eq. 11, and L is the graph Laplacian constructed from $\mathcal{X}$: $L = D - W$, where

$$W_{ij} = \begin{cases} \exp\left(-\frac{\|\mathbf{x}(i) - \mathbf{x}(j)\|^2}{\sigma^2}\right) & \text{if } \mathbf{x}(i) \in \mathcal{N}(\mathbf{x}(j)) \\ & \wedge\, \mathbf{x}(j) \in \mathcal{N}(\mathbf{x}(i)) \\ 0 & \text{otherwise,} \end{cases} \tag{15}$$

We use the k-nearest neighborhood for $\mathcal{N}$, with scale parameter σ^2 and number of neighbors k as hyper-parameters.

This type of semi-supervised ranking extension has been commonly used in data retrieval applications [8, 5, 11] and demonstrated its superior performance over purely supervised ranking approaches.

Transductive ranking (TR). If our goal is to introduce an ordering to a given fixed dataset $\mathcal{X}$, as is typical in making inferences on graph-structured data, then semi-supervised ranking can be formulated as *transductive* learning, thereby eliminating the model assumption on f (Eq. 11). In this case, the learning algorithm directly estimates the ranking evaluations $\mathbf{f}$ but not f itself. The corresponding energy functional can be presented as:

$$\mathcal{E}_{TR}(\mathbf{f}) = \sum_{(i,j)\in\mathcal{P}} l_P(\mathbf{f}_i - \mathbf{f}_j) + \sum_{(i,j)\in\mathcal{O}} l_O(\mathbf{f}_i - \mathbf{f}_j) + \lambda \mathbf{f}^\top L \mathbf{f}. \tag{16}$$

Roughly, minimizing the regularizer $\mathbf{f}^\top L\mathbf{f}$ implies that if $\mathbf{x}(i)$ and $\mathbf{x}(j)$ are *similar* in the input space $\mathbb{R}^m$, the corresponding rank estimates $\mathbf{f}_i$ and $\mathbf{f}_j$ should also be similar. This framework has been proven to be effective in many semi-supervised learning and spectral clustering applications. Furthermore, it provides a very intuitive explanation for data retrieval applications: "if $\mathbf{x}(i)$ and $\mathbf{x}(j)$ are similar, their relevance to the query $\mathbf{x}$ should be similar as well".

Kernel-based transductive ranking (KR). In data retrieval applications of ranking, we care about the relevance of each data point to a single query point, often to build a binary classifier. However, in applications with pairwise relations, we may care about the relative comparisons of all possible pairs of data points in $\mathcal{X}$ (equivalent to a linear ordering of $\mathcal{X}$). We exploit the rich structure of all joint relationships to build a new regularizer. To facilitate this process, we introduce an antisymmetric kernel $K : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ which contains relative ordering information:

$$K(\mathbf{x},\mathbf{y}) = -K(\mathbf{y},\mathbf{x}) \begin{cases} > 0 & \text{if Rank}(\mathbf{x}) > \text{Rank}(\mathbf{y}), \\ < 0 & \text{if Rank}(\mathbf{x}) < \text{Rank}(\mathbf{y}). \end{cases}$$

A simple example of K is:

$$K(\mathbf{x},\mathbf{y}) = f(\mathbf{x}) - f(\mathbf{y}) \tag{17}$$

assuming that an underling linear ordering function f exists. Given the kernel function K, our new kernel-based ranking energy functional (KR) is obtained as:

$$\mathcal{E}_{KR}(K) = \sum_{(i,j)\in\mathcal{P}} \max(0, 1 - K_{ij})^2 + \sum_{(i,j)\in\mathcal{O}} (K_{ij})^2 + \lambda tr[K^\top L K], \tag{18}$$

where $tr[A]$ is the trace of A, and $K_{ij} := K(\mathbf{x}(i), \mathbf{x}(j))$. We abuse notation and use K to denote a function and a matrix as its sample evaluation.

Comparison of RS, SSR, and TR. The energy functionals of semi-supervised RankSVM (SSR) and transductive ranking (TR) are almost identical (Eqs. 14 and 16, respectively). The only difference is the model assumption of SSR given in Eq. 11, which can be replaced by any smooth functions if desired (e.g., neural networks [11]). An important advantage of this *explicit* function representation is that, since its domain is the entire input space $\mathbb{R}^m$, it can be directly applied to new data points $\mathcal{X}^* \notin \mathcal{X}$ and produce the output estimates $\mathbf{f}^* := f|_{\mathcal{X}^*}$. For TR, this is not directly possible since only the function evaluations $\mathbf{f}$ are estimated.[3]

On the other hand, when the problem is truly transductive, indirect TR provides a theoretically more sound justification: It is an instance of learning a function on Riemannian manifolds. One of the fundamental assumptions in many successful semi-supervised learning and spectral clustering algorithms is that data are generated from an underlying Riemannian manifold (M, g) (g is the metric of M): The probability distribution $p(x)$ is supported in M [13, 1, 3, 10, 8, 12]. More precisely, the dataset $\mathcal{X}$ is originally presented as a subset of $\mathbb{R}^m$ but it is assumed as a sample from M that is a d-dimensional embedded submanifold of $\mathbb{R}^m$. In this case, even though the data points are realized as elements of $\mathbb{R}^m$, their effective degrees of freedom are limited to $d \leq m$. This assumption leads to a regularization approach that measures and enforces the smoothness of the function of interest f only along the manifold M instead of the ambient Euclidean space $\mathbb{R}^m$.

Once the regularization energy (or the smoothness measure) on M is defined, learning a function f is facilitated by combining it with the training error functions (e.g. l_P and l_O; Eq. 13). In practice, we do not have access to M directly. Instead a sample $\mathcal{X}$ is presented and therefore, a sample-based approximation or discretization is used. One of the best established sample-based regularizers in this aspect is the graph Laplacian L which is instantiated as a discretization of the density (p)-weighted Laplace-Beltrami operator $\Delta_p := \frac{1}{p}\nabla^* p \nabla$ on M [4, 1]. Applied to a smooth function $f \in C^\infty(M)$ on a compact manifold M, Δ_p can measure the first-order variation of f as weighted by p: Applying Stokes' theorem on M, the *Harmonic energy* of f is represented in terms of $\Delta_p f$:

$$\begin{aligned} \mathcal{E}_H(f) &:= -\int_M f(x)[\Delta_p f](x) dV(x) \\ &= \int_M \|\nabla^g f(x)\|_g^2 p(x) dV(x), \end{aligned} \tag{19}$$

where dV is the volume form of g. Furthermore, as $|\mathcal{X}| \to \infty$, $\mathbf{f}$ converges to a function f on M ($\mathbf{f}$ is regarded as $f|_{\mathcal{X}}$) and, in this case, the graph Laplacian regularizer corresponds to a sample-based approximation of $\mathcal{E}_H(f)$ [1, 4]:

$$C_{(M,g)} \mathbf{f}^\top L \mathbf{f} \to \mathcal{E}_H(f) \text{ as } n \to \infty, \tag{20}$$

where $C_{(M,g)}$ is a positive constant depending only on (M, g). This result provides a theoretical justification of graph Laplacian-based regularization approaches.

[3] However, once $\mathbf{f}$ is estimated, estimating $\mathbf{f}^*$ subsequently is not difficult. For instance, one could estimate $\mathbf{f}^*$ by applying the k-nearest neighbors regression algorithm using $\mathbf{f}$ as labels.

The rest of this subsection shows that this interpretation is not directly applicable to RS and SSR. Take the simplest case: When M itself is a compact domain D in Euclidean space and the data distribution p is uniform, the (Euclidean) derivative $\nabla f(x)$ of a linear function f at x (Eq. 11) is given as coefficients $\mathbf{w}$ independent of x. The corresponding regularization term $\|\mathbf{w}\|^2$ (Eq. 12) is simply a constant multiple (by the volume of D) of the Harmonic energy $\int_D \|\nabla f(x)\|^2 p(x)dx$ on D. In this special case, all three algorithms (RS, SSR, TR) essentially use the same regularizer. When p is non-uniform, SSR and TR can be interpreted as density-adaptive extensions of RS in the Euclidean space.[4]

The main difficulty to extend this perspective to general manifolds is that the vector ∇f and its coefficients $\mathbf{w} = (w^1, \ldots, w^d)$ cannot be naturally identified. In general, a vector X is represented by a set of coefficients $\mathbf{x} = (x^1, \ldots, x^d)$ once a basis $\{E_1, \ldots, E_d\}$ is fixed: $X = \sum_{i=1,\ldots,d} x^i E_i$. The geometric significance of X (e.g., its length) is independent of its coordinate representation $\mathbf{x}$ and its basis E. However, in the above example, we identified the gradient vector ∇f with its coordinate representation $\mathbf{w}$. This can be interpreted such that $\mathbf{w}$ is assumed to be the representation of ∇f in *canonical coordinates* (with respect to a canonical basis) in D. This special coordinate system is commonly used in Euclidean geometry as the squared norm $\|\nabla f\|^2$ of ∇f (which is a geometric quantity) is the same as the squared sum of the coefficients $\|\mathbf{w}\|^2 = \mathbf{w}^\top \mathbf{w} = \sum_{i=1,\ldots,d} (w^i)^2$ (which is an algebraic quantity). This justifies the identification of $\mathbf{w}$ and ∇f.

However, this identification does not always apply to other coordinate systems even in Euclidean space. For instance, given the polar coordinate representation ($\mathbf{w}' = [w'^1, \ldots, w'^d]$) of $\|\nabla f\|$, $\|\nabla f\|^2 \neq \mathbf{w}'^\top \mathbf{w}'$. In Euclidean space, this distinction between a vector X and its coordinate representation $\mathbf{x}$ is insignificant as one can always choose canonical coordinates which are defined over the entire space. However, on a general manifold, no single coordinate system covers the entire manifold and therefore canonical coordinates are not defined. Here, the explicit function representation (Eq. 11) does not lead to a geometric interpretation.

In this perspective, a major advantage of the transductive ranking (TR) setting over the model-based approaches (Eq. 14) is that it facilitates intrinsic (geometric) regularization: Explicitly calculating the gradient $\nabla^g f$ of f from only sampled data points $\mathcal{X}$ is a challenging problem. However, using the Stokes' theorem (Eq. 19) one can approximate the Harmonic energy of f without having to calculate the gradient vectors but instead using the the graph Laplacian.

Unfortunately, this approach cannot simply be extended to regularizing high-order tensors on manifolds (a function is a 0-th order tensor). Sections 2 and 3 of the main paper show that our kernel-based transductive ranking algorithm is obtained as a practical algorithm for intrinsic tensor regularization.

[4]For simplicity, we will henceforth assume that the density p is uniform. However, the convergence results presented in this paper extends to non-uniform probability distributions p using the results of Hein *et al.* [4].

3. Reconstruction capability of low-rank K approximations

A major drawback of our original kernel-based ranking approach (KR) is its high computational and memory complexities: It requires explicitly optimizing an $n \times n$-sized kernel matrix K. Therefore, directly applying KR to large-scale problems is infeasible. We overcome this limitation by adopting a low-rank factorized approximation of K: Given a factor matrix $B \in \mathbb{R}^{n \times p}$ ($p \ll n$), an antisymmetric kernel matrix $\widetilde{K} \in \mathbb{R}^{n \times n}$ of rank p is constructed as:

$$\begin{aligned}\widetilde{K} &= \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(B_{[:,i]} B_{[:,j]}^\top - B_{[:,j]} B_{[:,i]}^\top \right) \\ &= BQB^\top, \end{aligned} \tag{21}$$

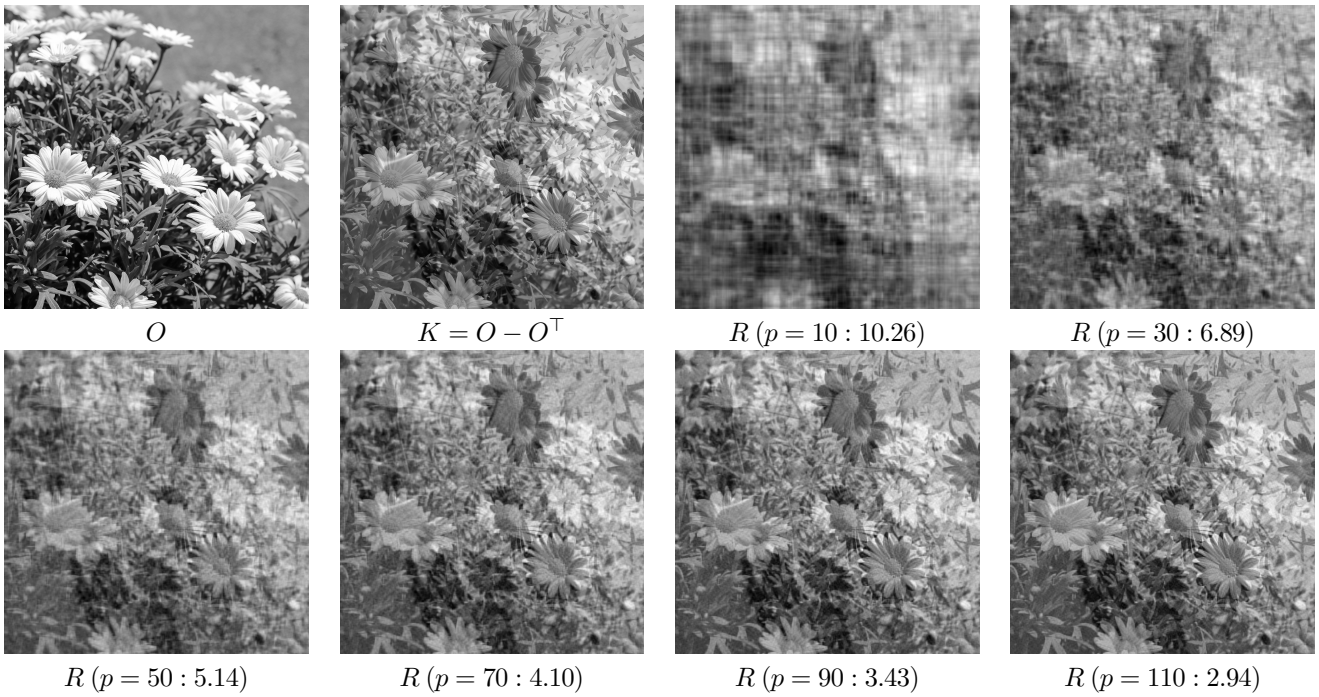
where $Q = R^\top - R$ with R being the lower triangular matrix of ones. By regarding $\widetilde{K}$ as an approximation of K, we take the low-rank matrix B as a new variable to optimize. Unfortunately, reformulating the KR optimization problem (Eq. 18) based on this factorization:

$$\begin{aligned}\mathcal{E}_{KR}(B) &= \mathcal{L}_P(B) + \mathcal{L}_O(B) + \lambda \mathcal{R}(B) \\ &= \sum_{(i,j) \in \mathcal{P}} \max(0, 1 - [BQB^\top]_{ij})^2 \\ &+ \sum_{(i,j) \in \mathcal{O}} ([BQB^\top]_{ij})^2 + \lambda tr[BQ^\top B^\top L B Q B^\top], \end{aligned} \tag{22}$$

renders the energy functional $\mathcal{E}_{KR}$ non-convex with respect to the parameter matrix B. However, we empirically observed that when B is initialized with all ones (i.e. $B = [\mathbf{1}]_n [\mathbf{1}]_p^\top$ with $\mathbf{1} = [1, \ldots, 1]^\top$), the resulting optimized solutions lead to competitive ranking results. This is further supported by evaluating the pure reconstruction capability of this factorization and the optimization initialization in image reconstruction as an example (Fig. 1): From the original $1,280 \times 1,280$-sized gray-level image (Fig. 1.O), we constructed an antisymmetric matrix K as $O - O^\top$ after normalizing the gray-level range of O to $[0, 1]$. The parameter matrix B is then obtained by minimizing the squared deviation $\|K - BQB^\top\|_F^2$ from K with B being initialized with ones. Even with the rank of B as low as 30, our reconstruction $BQB^\top$ already shows a perceivable image. Furthermore, the reconstruction error constantly decreased as the rank p increases.

4. Reconstruction of $\mathbf{f}$ given K

While the estimated kernel matrix K may not satisfy the reconstruction constraint of $\mathbf{f}$ (Eq. 17) for all pairs $(\mathbf{x}(i), \mathbf{x}(j)) \in \mathcal{X} \times \mathcal{X}$, $\mathbf{f}$ can be easily identified as the least-square approxi-



O | $K = O - O^\top$ | $R\ (p = 10 : 10.26)$ | $R\ (p = 30 : 6.89)$

$R\ (p = 50 : 5.14)$ | $R\ (p = 70 : 4.10)$ | $R\ (p = 90 : 3.43)$ | $R\ (p = 110 : 2.94)$

Figure 1. Low-rank approximation $R = BQB^\top$ (Eq. 21) of antisymmetric matrix K with varying rank p. For each reconstruction, we show the root-mean-squared deviation from K ($\times 100$). The approximation quality is good even for low-rank approximations (e.g., $p = 30$).

mation:[5] The reconstruction cost $\mathcal{C}(\mathbf{f})$ of $\mathbf{f}$ given K is given as:

$$\begin{aligned}\mathcal{C}(\mathbf{f}) &= \frac{1}{2}\sum_{i,j}(\mathbf{f}_i - \mathbf{f}_j - K_{ij})^2 \\ &= \mathbf{f}^\top(D - \mathbf{1}\mathbf{1}^\top)\mathbf{f} - 2\mathbf{f}^\top K\mathbf{1} + c,\end{aligned} \tag{23}$$

where $c = \frac{1}{2}\sum_{ij} K_{ij}^2$, $D = nI$ and we used $K^\top = -K$. Since $D - \mathbf{1}\mathbf{1}^\top$ is conditionally positive definite, $\mathcal{C}$ is not strictly convex. Therefore, we uniquely identify the optimal reconstruction solution $\mathbf{f}^*$ by adding a ridge regularizer:

$$\mathcal{C}_R(\mathbf{f}) := \mathcal{C}(\mathbf{f}) + \epsilon\mathbf{f}^\top\mathbf{f}, \tag{24}$$

where ϵ is a small positive value fixed at 10^{-8}. Since $\mathcal{C}_R$ is now strictly convex, $\mathbf{f}^*$ is obtained by equating the derivative of $\mathcal{C}_R$ with zero:

$$\begin{aligned}0 &= 2(D - \mathbf{1}\mathbf{1}^\top)\mathbf{f} - 2K\mathbf{1} + 2\epsilon\mathbf{f} \\ \Leftrightarrow \mathbf{f}^* &= ((D - \mathbf{1}\mathbf{1}^\top) + \epsilon I)^{-1}K\mathbf{1}.\end{aligned}$$

Now re-arranging the summands in the system matrix $G = ((D - \mathbf{1}\mathbf{1}^\top) + \epsilon I)^{-1}$ using the Sherman-Morrison-Woodbury formula, we obtain:

$$G = H - \frac{hh^\top}{1 + \mathbf{1}^\top h},$$

where $H = (D + \epsilon I)^{-1}$ and $h = H\mathbf{1}$. Therefore, $\mathbf{f}^*$ is obtained based on matrix-vector multiplications:

$$\mathbf{f}^* = HK\mathbf{1} - \left(\frac{h[h^\top K\mathbf{1}]}{1 + \mathbf{1}^\top h}\right). \tag{25}$$

Note that H and h can be calculated before K is optimized. When the low-rank approximation $BQB^\top$ of K is adopted (Eq. 21), each occurrence of K in Eq. 25 can be replaced by $BQB^\top$ in Eq. 25.

5. Experimental results

We compare our kernel-based transductive ranking algorithm (KR) to the relative attributes RankSVM approach (RS, Eq. 12 [9]), its model-based semi-supervised extension (SSR, Eq. 14) which can be regarded as an example of existing work in data retrieval applications [5, 11], and its straightforward transductive extension (TR). We also compare with deep neural networks that are optimized based on stochastic gradient descent (DR) [14].

Figure 2 shows the mean rank coefficients with corresponding error bars (with length twice the standard deviation). Deep learning algorithm (DR) outperformed RS for all datasets demonstrating the effectiveness of deep learning for ranking problems. Also, except for *PubFig*, the two transductive learning algorithms TR and KR constantly outperformed RankSVM (RS). This demonstrates the effectiveness of exploiting unlabeled data in relative attribute applications.

[5] It is possible to penalize the deviations from the equalities Eq. 17 as a new regularizer. This can be efficiently calculated in a similar way as $\mathbf{f}$ reconstruction. However, the resulting improvement in ranking performance is marginal and it requires tuning an additional regularization hyper-parameter. We abandoned this possibility for the sake of simplicity.

However, unlike TR and KR, performance of the model-based semi-supervised extension (SSR) is roughly on par with RS (it is better than RS on *ETH-80* and *DTD*, and worse on *OSR* and *PubFig*). Our kernel-based ranking algorithm (KR) significantly improves upon the other algorithms including the baseline transductive ranking (TR).

In particular, for *MNIST*, KR resulted in $\approx 40\%$ higher rank coefficients than other algorithms when the number of labels per class were less than 10. On *OSR*, DR and KR perform best. The improvement of KR over TR is especially significant when the number of labels l is limited. As l increases, the performance gap between these two algorithms narrows and eventually, they become almost identical as shown in the corresponding results of *DTD*.

Although the performances of KR and TR on this dataset are roughly equal, their performance variations across different attributes vary significantly. This suggests that, from the performance perspective, DR and KR are complementary. Figure 2 demonstrates this by simply choosing either the DR or KR results based on the validation error (DR+KR). This constructs a ranker that frequently outperforms other algorithms.

A notable exception to this tendency is *PubFig*, where DR is clear winner. This indicates that semi-supervised learning might not be always useful. One possible explanation is that *PubFig* has insufficient data points to reveal the underlying manifold structure upon which the semi-supervised algorithms build (only 772 data points, while other datasets are of order thousand or ten thousand). Another explanation is simply that the data do not lie on a low-dimensional manifold. Unfortunately, verifying these possibilities is a challenging problem. Furthermore, it is not straightforward to predict which (class of) algorithms would lead to better performances on specific datasets or problems. In practice, users would interact (provide labels) with data and be able to provide feedback on the *utility* of different algorithms. In this respect, the experiments demonstrate that our kernel-based ranking algorithm provides a good alternative to RankSVM and deep learning.

References

[1] M. Belkin and P. Niyogi. Towards a theoretical foundation for Laplacian-based manifold methods. *Journal of Computer and System Sciences*, 74(8):1289–1308, 2005. 3

[2] O. Chapelle and S. S. Keerthi. Efficient algorithms for ranking with SVMs. *Information Retrieval*, 13(3):201–215, 2010. 2

[3] O. Chapelle, B. Schölkopf, and A. Zien. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2010. 3

[4] M. Hein, J.-Y. Audibert, and U. von Luxburg. From graphs to manifolds - weak and strong pointwise consistency of graph Laplacians. In *Proc. COLT*, pages 470–485, 2005. 3, 4

[5] S. C. H. Hoi and R. Jin. Semi-supervised ensemble ranking. In *Proc. AAAI*, pages 634–639, 2008. 2, 5

[6] J. Jost. *Riemannian Geometry and Geometric Analysis*. Springer, New York, 6th edition, 2011. 1

[7] J. M. Lee. *Introduction to Smooth Manifolds*. Springer, 2003. 1, 2

[8] Y. Liu, Y. Liu, S. Zhong, and K. C. C. Chan. Semi-supervised manifold ordinal regression for image ranking. In *Proc. ACM Multimedia*, pages 1393–1396, 2011. 2, 3

[9] D. Parikh and K. Grauman. Relative attributes. In *Proc. IEEE ICCV*, pages 503–510, 2011. 2, 5

[10] F. Perbet, S. Johnson, M.-T. Pham, and B. Stenger. Human body shape estimation using a multi-resolution manifold forest. In *Proc. IEEE CVPR*, pages 668–675, 2014. 3

[11] M. Szummer and E. Yilmaz. Semi-supervised learning to rank with preference regularization. In *Proc. ACM CIKM*, pages 269–278, 2011. 2, 3, 5

[12] J. B. Tenenbaum, V. Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, 2000. 3

[13] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007. 3

[14] X. Yang, T. Zhang, C. Xu, S. Yan, M. S. Hossain, and A. Ghoneim. Deep relative attributes. *IEEE T-MM*, 18(9):1832–1842, 2016. 5

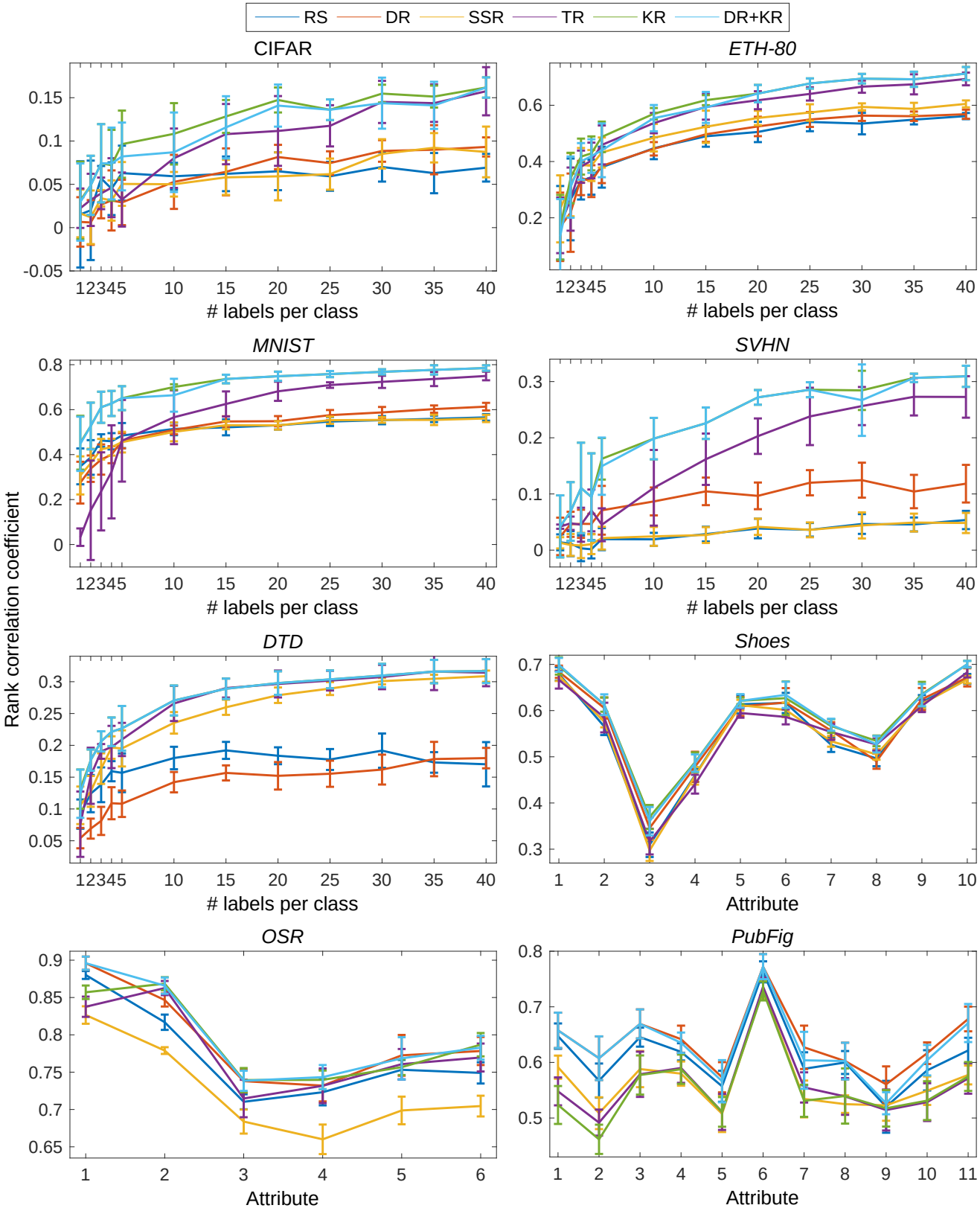


Figure 2. Performance of different ranking algorithms for eight datasets. In raster order, *first five:* x-axis shows the number of labels per class; *last three*: x-axis corresponds to the indices of attributes to learn. In all but PubFig, our KR is comparable or better. Furthermore, combining DR and KR, we can construct a ranker that frequently outperforms other algorithms (cyan line).