Multi-Label Learning with Kernel Generalized Homogeneity Analysis

Hanchen Xiong Sandor Szedmak Justus Piater

{*hanchen.xiong,sandor.szedmak,justus.piater*}*@uibk.ac.at*

Institute of Computer Science, University of Innsbruck

Abstract

Canonical correlation analysis (CCA) and homogeneity analysis (HA) are two popular methods for analyzing multivariate data. Although they are applied on different data types – the former is used on two sets of variables while the latter operates on multivariate categorical variables – we reveal that they are actually closely related. Building on this relation, we generalize HA to handle continuous variables, which leads to a relaxed variant of multiple-set CCA. Furthermore, kernel functions are also utilized to enable generalized HA to learn nonlinear dependencies within data.

In present paper, we in particular investigate how kernel generalized HA (KGHA) can be applied to multi-label learning. We found that, for vector-valued functions, KGHA works as a learning method consisting of two advantageous components: low-rank output kernel learning and co-regularized multi-view learning. Low-rank output kernel learning coincides with lower-dimensional latent label space discovery, while co-regularized multi-view learning is related to multiple kernel learning for heterogeneous information fusion. Furthermore, a large-scale KGHA learning scheme is developed by employing a block-wise Nyström approximation. We evaluate KGHA on two multi-label classification applications, image annotation and protein function prediction. Our experimental results on several benchmark databases demonstrate that KGHA compares favorably to other state-of-the-art methods.

1 Introduction

The study of embedding complex data into a lower-dimensional space is an important task in machine learning. Relevant methods include *principal component analysis* (PCA), *canonical correlation analysis* (CCA), *homogeneity analysis* (HA, also known as multiple correspondence analysis), to name just a few. These methods are generally known as *multivariate analysis* (MVA; Izenman 2008). Originally, MVA methods were proposed with linear projections to satisfy different objective functions, in supervised or unsupervised contexts. For instance, the objective of PCA is to maximize the variances of linear projections of data onto a small number of principal bases. CCA seeks two lower-dimensional coordinate frames in which two sets of variables (*e.g.* input and output) are maximally correlated. HA, by contrast, operates on multivariate categorical data, and outcomes are a set of linear projections which can map both data instances

and categorical values to a low-dimensional space such that their consistency is preserved as much as possible. Although these methods have been successfully employed in various application domains, detecting linear patterns within data is rather limited in the face of increasingly more complicated data. Therefore, some nonlinear embedding techniques, *e.g. kernelized versions* of the above-mentioned MVA methods or *nonlinear manifold learning* methods (Ma & Fu, 2011), are increasingly used in modern data analysis.

We start with introductions to CCA and HA (Section 2), in which their corresponding objective functions, constraints, solutions and properties are explained. Similar to CCA, in a supervised-learning context, HA can be employed by considering one set of variables as outputs and the remaining sets as input features from heterogeneous information sources. Then, in section 3, by reformulating CCA on J sets of variables with $J > 2$, we arrive at an objective function of a form identical to HA. Building on this relationship, we generalize homogeneity analysis to handle continuous variables, which leads to a relaxed variant of multiple-set CCA. Furthermore, in section 4, we add a trade-off parameter to fit supervised-learning scenarios and kernel functions to enable generalized HA for learning nonlinear patterns. We refer to this novel HA as kernel generalized HA (KGHA). Similarly to regular HA, KGHA is trained via alternating least squares (ALS), which is more efficient than the multiple pair-wise eigenvalue computation in multiple-set CCA.

In section 5 we study KGHA in the multi-label learning case. We show that when used for learning vector-valued functions (*e.g.* multi-label, multi-task learning), KGHA is an elegant combination of low-rank output kernel learning and co-regularized multi-view learning. Low-rank output kernel learning coincides with multi-label dimensionality reduction (Ye et al., 2011), which enables learners to gain higher efficiency and accuracy (Ji & Ye, 2009) by exploiting more compact yet informative latent space. Also, co-regularized multi-view learning is related to multiple kernel learning (MKL; Bucak et al. 2014), in which heterogeneous information is encoded in an ensemble of kernels to match outputs. One feature worth noting is that, since multi-label is encoded in a lower-dimensional latent space, co-regularization in KGHA takes place in a subsapce of multi-view, which differs from conventional co-regularization (Rosenberg & Bartlett, 2007).

This paper makes four contributions. First, we reveal the close connections between HA and multiple-set CCA, which sheds light on new understanding and potential extensions of these two MVA techniques. Second, we propose a novel multi-label learning method, KGHA, which is composed of two advantageous components, low-rank output kernel learning and co-regularized multi-view learning. Third, we develop a large-scale learning scheme for KGHA by employing a block-wise Nyström method for approximating kernel matrices and conjugate gradient for solving ALS. Finally, according to our experimental results in image annotation and protein function prediction tasks, KGHA can improve performance on several benchmark databases.

1.1 Related Work

Our study can be connected to many other work in different respects. The following gives a short summary of recent advances of relevant research.

Variants of CCA. It has been shown that CCA is related to other MVA techniques, such as partial least square (PLS) (Sun et al., 2009) and Fisher linear discriminative analysis (Sun et al., 2011). More extensions of CCA for multi-label learning can be found in Hardoon et al. Hardoon et al. (2004) and Sun et al. Sun et al. (2011).

Multi-Label Prediction. Basically, multi-label learning has been studied with different "canonical" learning schemes , e.g. regression (Hsu et al., 2009; Lin et al., 2014) and ranking (Elisseeff & Weston, 2002). Recently, structured output learning has also been leveraged (Hariharan et al., 2010; Xiong et al., 2014) for this study. Our method belongs to the regression category. **Multi-Label Dimensionality Reduction.** Much effort has been put into learning a shared subspace for multi-label outputs (see a review by Ye et al. 2011). Other notable work includes projection via compressed sensing (Hsu et al., 2009) and feature-aware label encoding (Lin et al., 2014). **MKL**. MKL has been recruited as a framework for integrating multiple input features from heterogeneous information sources (Wang et al., 2008). Especially in computer vision and bioinformatics applications (Bucak et al., 2014; Mostafavi & Morris, 2010), since various visual features and bio-related features are available, MKL plays an important role in manipulating geometric structures of data in multiple features to fit certain applications. **Co-regularization for Multi-View Learning.** Co-regularization has been well investigated in multi-view learning (Rosenberg & Bartlett, 2007; Sridharan & Kakade, 2008), however, these studies focus on semi-supervised learning. A similar subspace co-regularization in supervised-learning circumstance was proposed in Guo & Xiao (Guo & Xiao, 2012), where nevertheless only two views are considered.

Two pieces of work closely related to KGHA are FaIE (Lin et al., 2014) and MultiK-MHKS (Wang et al., 2008) respectively. First, in FaIE, lower-dimensional projections of multi-label data are found by jointly optimizing the correlations between input features and projections and recoverability of projections back to the original output data. From a different perspective, KGHA can be formulated as an objective function rather similar to FaIE. KGHA goes beyond FaIE by considering multiple features from heterogeneous sources of information. Secondly, in MultiK-MHKS, an extra regularization is used to encourage consensus among predictions from multiple kernels, which is identical to our co-regularized multi-view learning. Our work differs from MultiK-MHKS in that we learn multiple kernels in a non-binary subspace, and thus use least-squares loss instead of misclassification loss. In this sense, KGHA can be considered a combination of FaIE and MultiK-MHKS.

2 Preliminaries

2.1 Canonical Correlation Analysis

Canonical correlation analysis (CCA) (Hardoon et al., 2004) was developed to find the correlations between two sets of variables. The essence of CCA is to seek a pair of linear transformations, one for each set, such that the correlation of transformed variables is maximized. Assume that a data instance is composed of two set of variates, $[g_1^\top, g_2^\top]$, of which the dimensions are d_1 and d_2 respectively. A dataset $\mathcal{D}$ consisting of M such instances can be represented as a $M \times (d_1 + d_2)$ matrix of the form $\mathcal{D} = [G_1, G_2]$. By using two matrices $\mathbf{w}_1 \in \mathbb{R}^{d_1 \times p}$ and $\mathbf{w}_2 \in \mathbb{R}^{d_2 \times p}$ with $p < \min(d_1, d_2)$, we can project the data into a lower, p-dimensional space:

$$\hat{\mathcal{D}} = [G_1\mathbf{w}_1, G_2\mathbf{w}_2] \tag{1}$$

Assuming the original data are already centered, $\hat{\mathcal{D}}$ will be centered as well, and the covariance of $\hat{\mathcal{D}}$ is $\mathbf{w}_1 G_1 G_2 \mathbf{w}_2$. The objective of CCA is to select $\mathbf{w}_1$ and $\mathbf{w}_2$ to

maximize the correlation between $G_1\mathbf{w}_1$ and $G_2\mathbf{w}_2$:

$$\begin{aligned}\{\mathbf{w}_1^*, \mathbf{w}_2^*\} &= \underset{\mathbf{w}_1,\mathbf{w}_2}{\operatorname{argmax}} \frac{\mathbf{w}_1^\top G_1^\top G_2 \mathbf{w}_2}{||\mathbf{w}_1 G_1|| ||\mathbf{w}_2 G_2||} \\ &= \underset{\mathbf{w}_1,\mathbf{w}_2}{\operatorname{argmax}} \frac{\mathbf{w}_1^\top C_{12} \mathbf{w}_2}{\sqrt{\mathbf{w}_1^\top C_{11} \mathbf{w}_1 \mathbf{w}_2^\top C_{22} \mathbf{w}_2}}\end{aligned} \tag{2}$$

where C_{12}, C_{11}, C_{22} are blocks within the covariance matrices of $\mathcal{D}$:

$$\operatorname{cov}(\mathcal{D}) = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \tag{3}$$

(2) can be rewritten as

$$\begin{aligned}\{\mathbf{w}_1^*, \mathbf{w}_2^*\} &= \underset{\mathbf{w}_1,\mathbf{w}_2}{\operatorname{argmax}} \mathbf{w}_1^\top C_{12} \mathbf{w}_2 \\ s.t. \qquad & \mathbf{w}_1^\top C_{11} \mathbf{w}_1 = 1,\ \mathbf{w}_2^\top C_{22} \mathbf{w}_2 = 1\end{aligned} \tag{4}$$

It has been shown (Bie et al., 2005; Hardoon et al., 2004) that the solution to (4) can be obtained by solving following generalized eigenvalue problem:

$$\begin{pmatrix} \mathbf{0} & C_{12} \\ C_{21} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{pmatrix} = \lambda \begin{pmatrix} C_{11} & \mathbf{0} \\ \mathbf{0} & C_{22} \end{pmatrix} \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{pmatrix} \tag{5}$$

There can be many solutions for (5), which correspond to different eigenvectors. One important property of different solution pairs (*e.g.* when we have p solution pairs $\mathbf{W}_1 = [\mathbf{w}_1^1, \ldots, \mathbf{w}_1^p]$, $\mathbf{W}_2 = [\mathbf{w}_2^1, \ldots, \mathbf{w}_2^p]$) is that the projections onto different $\mathbf{w}_{i=1,2}^{k\in[1,p]}$ are uncorrelated to each other:

$$\begin{aligned}\forall i = 1, 2, \qquad & \mathbf{W}_i^\top C_{ii} \mathbf{W}_i = I_p \\ \forall k \neq h, \qquad & \mathbf{w}_1^{k\top} C_{12} \mathbf{w}_2^h = 0\end{aligned} \tag{6}$$

It was also shown that the solutions of (5) $\mathbf{w}_1, \mathbf{w}_2$ lie in the span of G_1 and G_2 respectively, *i.e.* $\mathbf{w}_1 = G_1^\top \boldsymbol{\alpha}_1$, $\mathbf{w}_2 = G_2^\top \boldsymbol{\alpha}_2$, $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2 \in \mathbb{R}^M$. By substituting the alternative form of $\mathbf{w}_1, \mathbf{w}_2$ into the primal form of CCA (5), we can write out the dual form of CCA (Bie et al., 2005; Hardoon et al., 2004) as

$$\begin{pmatrix} \mathbf{0} & K_1 K_2 \\ K_2 K_1 & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha}_1 \\ \boldsymbol{\alpha}_2 \end{pmatrix} = \lambda \begin{pmatrix} K_1^2 & \mathbf{0} \\ \mathbf{0} & K_2^2 \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha}_1 \\ \boldsymbol{\alpha}_2 \end{pmatrix} \tag{7}$$

where $K \in \mathbb{R}^{M\times M}$ is the *Gram matrix* of the data, *i.e.* $K_{i=1,2} = G_i G_i^\top$. Therefore, by comparing (5) and (7), we can see that when $M < d_1 + d_2$, the dual form can be used to accelerate computing. The value of the dual form is even more significant when the *kernel method* is used on the data and the Gram matrix is replaced with a kernel matrix:

$$\mathcal{K}(G_i^{(m)}, G_i^{(n)}) = \left\langle \phi_i(G_i^{(m)}), \phi_i(G_i^{(n)}) \right\rangle \tag{8}$$

where $\phi_i : \mathbb{R}^{d_i} \to \mathcal{H}$ is a feature map from original data space to a reproducing kernel Hilbert space (RKHS). Kernel methods are of great help in detecting nonlinear patterns within the data.

2.2 Homogeneity Analysis

Homogeneity analysis (Michailidis & de Leeuw, 1998) is a popular tool for analyzing and visualizing multivariate categorical data. Assume that there are M data instances

in a dataset $\mathcal{D} = \{O_m\}_{m=1}^{M}$, and each data instance is represented by a J-dimensional vector $O_m = [v_1, v_2, \ldots, v_J]^\top$ $(m = 1, \ldots, M)$. Variable v_j takes on n_j categorical values. Here we briefly review the procedure of homogeneity analysis with its application to this simple dataset. Since the data is represented in a categorical space, we need to convert them to a vector space. To this end, we list n_j categorical values of v_j over all M data instances into an $M \times n_j$ binary indicator matrix G_j. The set of indicator matrices can be gathered in a block matrix

$$G = [G_1|G_2|\cdots|G_J]. \tag{9}$$

The key feature of homogeneity analysis is that it simultaneously produces two projections into the same Euclidean space $\mathbb{R}^p$, one from J-dimensional data instances O_i, the other from the M-dimensional categorical attribute indicator vectors (columns of G). These projections are referred to as *object scores* and *category quantifications*, respectively (Michailidis & de Leeuw, 1998). In addition, these two projections are intended to preserve the consistency among data instances and attribute values as closely as possible to the data in the original categorical space:

- data instances that exhibit similar attribute values are located closely together;
- data instances are close to their attribute category values.

Suppose that the collection of data instances is represented by an $M \times p$ matrix X, and category quantifications for variable v_j are represented by a $n_j \times p$ matrix Y_j. Then, the cost function of projections can be formulated as:

$$f(X, Y_1, \ldots, Y_J) = \frac{1}{J}\sum_{j=1}^{J} ||X - G_j Y_j||_F^2 \tag{10}$$

where $||\cdot||_F$ denotes the Frobenius norm. Two extra constraints are added to avoid the trivial solution ($X = \mathbf{0}, \forall j \in [1, J] \quad Y_j = \mathbf{0}$):

$$\mathbf{1}_{M\times 1}^\top X = \mathbf{0} \tag{11}$$

$$X^\top X = I_p \tag{12}$$

The first constraint (11) essentially normalizes the projected object scores to be centered around the origin. The second restriction (12) standardizes all p dimensions of the object score by rescaling the square length of each dimension to M. In addition, another effect of (12) is that the p columns of X are imposed to be orthogonal to each other.

To minimize the cost function (10) under these constraints (11, 12), usually the alternating least squares (ALS) algorithm (Michailidis & de Leeuw, 1998) is used. The basic idea of ALS is to iteratively optimize with respect to X or to $[Y_1, \ldots, Y_M]$ with the other held fixed. Assuming $X^{(0)}$ is provided arbitrarily at iteration $t = 0$, each iteration of ALS can be summarized as:

1. $\forall j \in [1, J]$,update Y_j:

$$Y_j^{(t)} = (G_j^\top G_j)^{-1} G_j^\top X^{(t)} \tag{13}$$

2. update X:

$$X^{(t+1)} = J^{-1}\sum_{j=1}^{J} G_j Y_j^{(t)} \tag{14}$$

3. normalize X:

$$X^{(t+1)} = \textit{Gram-Schmidt}(X^{(t+1)}) \tag{15}$$

It can be seen (13) that the category quantification of Y_j is computed as the centroid of the object scores that belong to it. Step 2 (14) updates object scores X by taking the average of the quantifications of the categories it belongs to. In step 3 (15) a *Gram-Schmidt* procedure is used to find the normalized and orthogonal basis of updated object scores from the previous step. In this way, the object scores will be located close to the category quantifications they fall in, and category quantifications will be close to the object scores belonging in them.

3 Linking CCA and HA

Based on the previous section, we can see that CCA and HA are used in two different data types: the former operates on two sets of variables while the latter is used on multivariate categorical variables. We reveal that they are closely related when CCA is generalized to multiple sets of variables. Suppose we want to find 2 sets of p linear projections $\mathbf{W}_1 = [\mathbf{w}_1^1, \ldots, \mathbf{w}_1^p]$, $\mathbf{W}_2 = [\mathbf{w}_2^1, \ldots, \mathbf{w}_2^p]$ for regular CCA. We can rewrite (2) as

$$\begin{aligned} &\{\mathbf{W}_1^*, \mathbf{W}_2^*\} = \underset{\mathbf{W}_1, \mathbf{W}_2}{\operatorname{argmin}} ||G_1\mathbf{W}_1 - G_2\mathbf{W}_2||_F^2 \\ &s.t. \quad \forall i \in \{1,2\}, \forall k, h \in [1,p], k \neq t \\ &\quad \mathbf{W}_i^\top C_{ii}\mathbf{W}_i = I_p, \quad \mathbf{w}_1^{k\top} C_{12}\mathbf{w}_2^h = 0 \end{aligned} \tag{16}$$

When we have $J > 2$ sets of variables, (16) will be:

$$\begin{aligned} &\{\mathbf{W}_1^*, \ldots, \mathbf{W}_J^*\} = \underset{\mathbf{W}_1, \ldots, \mathbf{W}_J}{\operatorname{argmin}} \sum_{i=1,j=1}^{J} ||G_i\mathbf{W}_i - G_j\mathbf{W}_j||_F^2 \\ &s.t. \quad \forall i, j \in [1,J], \forall k, h \in [1,p], k \neq h \\ &\mathbf{W}_i^\top C_{ii}\mathbf{W}_i = I_p, \quad \mathbf{w}_i^{k\top} C_{ij}\mathbf{w}_j^h = 0 \end{aligned} \tag{17}$$

Lemma 3.1 *The objective function in (17) is equivalent to*

$$\min_{X, \mathbf{W}_1, \ldots, \mathbf{W}_J} \frac{1}{J} \sum_{j=1}^{J} ||X - G_j\mathbf{W}_j||_F^2 \tag{18}$$

Proof For simplicity, we only consider one data instance $\mathcal{D} = [g_1^\top, \ldots, g_J^\top]$. $\forall i, j \in [1,J], i \neq j$, we denote $\mathbf{W}_i^\top g_i$ and $\mathbf{W}_j^\top g_j$ as v_i and v_j respectively, $v_i, v_j \in \mathbb{R}^p$. Then the objective function in (17) is

$$\begin{aligned} & \sum_{i=1,j=1}^{J} ||v_i - v_j||^2 \\ = & \sum_{i=j,j=1}^{J} \sum_{k=1}^{p} \left(v_{ik}^2 + v_{jk}^2 - 2v_{ik}v_{jk}\right) \\ = & \sum_{k=1}^{p} \left(\sum_{i=1,j=1}^{J} v_{ik}^2 + \sum_{i=1,j=1}^{J} v_{jk}^2 - \sum_{i=1,j=1}^{J} 2v_{ik}v_{jk} \right) \\ = & \sum_{k=1}^{p} \left(J\sum_{i=1}^{J} v_{ik}^2 + J\sum_{j=1}^{J} v_{jk}^2 - 2\sum_{i=1}^{J} v_{ik} \sum_{j=1}^{J} v_{jk} \right). \end{aligned} \tag{19}$$

In addition, by denoting $\mathcal{M}_1^k = \frac{1}{J}\sum_{j=1}^{J} v_{jk}$, $\mathcal{M}_2^k = \frac{1}{J}\sum_{j=1}^{J} v_{jk}^2$, (19) is equal to

$$J^2 \sum_{k=1}^{p} \left(\mathcal{M}_2^k - (\mathcal{M}_1^k)^2\right). \tag{20}$$

Since $\left(\mathcal{M}_2^k - (\mathcal{M}_1^k)^2\right)$ is the variance of the kth component in $\{v_i\}_{i=1}^{J}$, this is further equal to

$$J^2 \sum_{k=1}^{p}\sum_{j=1}^{J}(v_{ik} - \mathcal{M}_1^k)^2 = J^2 \sum_{j=1}^{J} ||v_i - \mathbf{M}_1||^2 \tag{21}$$

where $\mathbf{M}_1 = [\mathcal{M}_1^1, \ldots, \mathcal{M}_1^p]^\top$. (21) can be phrased as a rescaled optimization problem

$$\min_X \frac{1}{J}\sum_{j=1}^{J} ||v_i - X||^2 \tag{22}$$

with optimal solution $X = \mathbf{M}_1 = \frac{1}{J}\sum_{j=1}^{J} v_j$. When M data instances are considered, it is straightforward to extend (22) to

$$\min_X \frac{1}{J}\sum_{j=1}^{J} ||G_i\mathbf{W}_i - X||_F^2 \tag{23}$$

which completes the proof of the lemma. ∎

Comparing (10) and (23), we can see that multiple-set CCA has the same objective function as HA (by replacing Y_i with $\mathbf{W}_i$), yet with different constraints; see (11), (12) and (17). In the following, we will show some connections between constraints in multiple-set CCA Ω_{mCCA} and constraints in HA Ω_{HA}.

First, since in $\Omega_{mCCA}, \forall j \in [1, J], \mathbf{1}_{M\times 1}^\top G_j\mathbf{W}_j = 0, \mathbf{1}_{M\times 1}^\top X = \frac{1}{J}\sum_{j=1}^{J}\mathbf{1}_{M\times 1}^\top G_j\mathbf{W}_j = 0$, which coincides with the first constraint in Ω_{HA} (11). Secondly, in $\Omega_{mCCA}, \forall i, j \in [1, J], \forall k, h \in [1, p], k \neq h, \mathbf{W}_i^\top C_{ii}\mathbf{W}_i = I_p, \quad \mathbf{w}_i^{k\top} C_{ij}\mathbf{w}_j^h = 0$. Therefore, $X^\top X = \frac{1}{J^2}\left(\sum_{j=1}^{J}\mathbf{W}_j^\top C_{jj}\mathbf{W}_j + 2\sum_{i\neq j}\mathbf{W}_i^\top C_{ij}\mathbf{W}_j\right)$. We can see that when the correlation of projected data in every pair (i, j) are ideally maximized to 1, $X^\top X = I_p$, which is a rescaled version of the second constraint in Ω_{HA} (11). However, satisfying Ω_{HA} cannot ensure satisfaction of any constraint in Ω_{mCCA}. Therefore, roughly speaking, we can consider Ω_{mCCA} as a sufficient but not necessary condition for Ω_{HA}, or in other words, Ω_{HA} is a relaxed version of Ω_{mCCA}.

4 Kernel Generalized Homogeneity Analysis

Based on the analysis above, we can generalize HA as a relaxed variant of multiple-set CCA by replacing binary indicator matrices of J types of features. One strength we gain by using HA is that normalization constraints on J individual projections are eliminated. Therefore, by using ALS for training, multiple pair-wise eigenvalue computations can be avoided. In a supervised-learning context, we can assume that the Jth set of variables are outputs (denoted by $T = [t^{(1)}, t^{(2)}, \ldots, t^{(M)}]^\top \in \mathbb{R}^{M\times d_J}$) and the remaining $J-1$ sets of variables represent $J-1$ input features from heterogeneous

information sources. Then (10) is rewritten as

$$\begin{aligned} & f(X, \mathbf{W}, \ldots, \mathbf{W}_{J-1}, \mathbf{P}) \\ = \quad & \frac{1}{J}\left(\sum_{j=1}^{J-1} \underbrace{||X - G_j\mathbf{W}_j||_F^2}_{\rho_j} + \underbrace{||X - T\mathbf{P}||_F^2}_{\pi}\right) \end{aligned} \tag{24}$$

where $\mathbf{P}$ is the projection associated with outputs T[1]. Interestingly, ρ_j and π in (24) are identical to the predictability and recoverability of X respectively, which are two concepts recently introduced in FaIE (Lin et al., 2014). More concretely, predictability is measured by how much input features are correlated with lower-dimensional representations of multi-label outputs, while recoverability refers to how successfully the compact representations can be decoded back to binary vectors. It is worth noting that only one ρ_j was used in FaIE. Following the philosophy of Lin et al. (2014), we also introduce a trade-off parameter λ to balance $\sum_{j=1}^{J} \rho_j$ and π. After rescaling we can further rewrite (24) as:

$$f = \lambda \sum_{j=1}^{J-1} ||X - G_j\mathbf{W}_j||_F^2 + ||X - T\mathbf{P}||_F^2 \tag{25}$$

Similarly to kernel CCA and kernel FaIE, we can add a kernel function (8), for each feature, on a pair of data points, $\mathcal{K}_j(G_j^{(m)}, G_j^{(n)}), j \in [1, J], m, n \in [1, M]$. We refer to this novel learning method as kernel generalized HA (KGHA). Since updates of Y_j in (13) solve a multivariate linear regression (MLR), by replacing it with a dual form of kernel multivariate ridge regression (KMRR), we can develop a dual learning algorithm for KGHA by changing the first two steps in ALS to

1. $\forall j \in [1, J]$, update the dual matrix $\boldsymbol{\alpha}_j \in \mathbb{R}^{M \times p}$:

$$\boldsymbol{\alpha}_j^{(t)} = (K_j + c_j I_M)^{-1} X \tag{26}$$

2. update X:

$$X^{(t+1)} = \frac{1}{\lambda(J-1)+1}\left(\sum_{j=1}^{J-1} \lambda K_j \boldsymbol{\alpha}_j + K_J \boldsymbol{\alpha}_J\right) \tag{27}$$

where c_j is a ridge parameter for each feature . K_j denotes the kernel matrix of the data within the kth feature or the Gram matrix if no kernel function is applied.

5 Multi-Label Learning with KGHA

We now investigate the application of KGHA on multi-label learning, in which KGHA works as a learning framework with low-rank output kernel learning and subspace co-regularized multi-view learning. For the kernel on the jth feature ($j \in [1, J-1]$), the original data are mapped to a RKHS $\phi_j(G_j^{(m)}) \in \mathcal{H}_j, m \in [1, M]$. We define a linear kernel on multi-label outputs as did Hariharan et al. (2010) and Dinuzzo et al. (2011):

$$\mathcal{K}_T(t^{(m)}, t^{(n)}) = \left\langle \phi_T(t^{(m)}), \phi_T(t^{(n)}) \right\rangle = \left\langle \mathbf{Q}^\top t^{(m)}, \mathbf{Q}^\top t^{(n)} \right\rangle \tag{28}$$

[1]From now on, we refer to the same thing by using G_J or T, and similarly, $\mathbf{P}$ and $\mathbf{W}_J$ are equivalent.

where $t^{(m)}, t^{(n)} \in \mathbb{B}^{d_J} = \mathcal{T}$, $\mathbf{Q} \in \mathbb{R}^{d_J \times d_J}$ captures the pairwise dependencies between elements in $t^{(m)}$. Using a pairwise formulation as in (17), the objective function of KGHA is

$$\begin{aligned} &\sum_{j=1}^{J-1} \underbrace{||\phi_j(G_j)\mathbf{W}_j - T\mathbf{Q}\mathbf{P})||_F^2}_{\mathcal{A}_j} \\ &+ \lambda \sum_{i,j=1:i\neq j}^{J-1} \underbrace{||\phi_i(G_i)\mathbf{W}_i - \phi_j(G_j)\mathbf{W}_j||_F^2}_{\mathcal{B}_{ij}} \end{aligned} \tag{29}$$

where $\phi_j(G_j) = [\phi_j(G_j)^{(1)}, \ldots, \phi_j(G_j)^{(M)}]^\top$, $\sum_{j=1}^{J-1} \mathcal{A}_j$ corresponds to low-rank output kernel learning with $J-1$ features, while $\sum_{i,j=1:i\neq j}^{J-1} \mathcal{B}_{ij}$ corresponds to co-regularization for multi-view learning.

5.1 Low-Rank Output Kernel Learning

Let $\tilde{\mathbf{Q}} = \mathbf{Q}\mathbf{P}$ be a low-rank feature map $\tilde{\phi}_T$ for $\mathcal{T}$. Then, $\mathcal{K}_T(t^{(m)}, t^{(n)}) = t^{(m)\top}\tilde{\mathbf{Q}}\tilde{\mathbf{Q}}^\top t^{(n)} = t^{(m)\top}\mathbf{L}t^{(n)}$. In each $\mathcal{A}_j$, with a feature map defined on both input and output, a function to be learned is defined as $f_j(G_j^{(m)}, t^{(m)}) = \mathbf{W}_j^\top \left(\phi_j(G_j^{(m)}) \otimes \tilde{\phi}_T(t^{(m)})\right)$, where $\otimes$ denotes tensor product. Therefore, within the framework of regularization in reproducing kernel Hilbert spaces (RKHS) of vector-valued functions (Micchelli & Pontil, 2005), the unique kernel $\mathbf{H}_j$ associated with the RHKS of $\tilde{\phi}_T(\mathcal{T})$-valued function is

$$\begin{aligned} \mathbf{H}_j &= \left\langle \phi(G_j^{(m)}) \otimes \tilde{\phi}_T(t^{(m)}), \phi(G_j^{(n)}) \otimes \tilde{\phi}_T(t^{(n)}) \right\rangle \\ &= \left\langle \phi(G_j^{(m)}), \phi(G_j^{(n)}) \right\rangle \left\langle \tilde{\phi}_T(t^{(m)}), \tilde{\phi}_T(t^{(n)} \right\rangle \\ &= \left\langle t^{(m)}, \mathcal{K}_j^{m,n}\mathbf{L}t^{(n)} \right\rangle, \end{aligned}$$

where $\mathcal{K}_j^{m,n}$ is the kernel value $\mathcal{K}_j(G_j^{(m)}, \phi(G_j^{(n)})$. $\mathcal{K}_j^{m,n}\mathbf{L}$ defines an operator-valued, positive semidefinite $\mathcal{T}$-kernel: $\mathbb{R}^{d_j} \times \mathbb{R}^{d_j} \to \mathbb{R}^{d_J \times d_J}$. Because of the decomposability $\mathbf{H}_j = \mathcal{K}_j \cdot \mathbf{L}$ (Dinuzzo et al., 2011), $\mathbf{L}$ corresponds to a low-rank output kernel (Dinuzzo & Fukumizu, 2011). Since $\tilde{\mathbf{Q}}$ itself specifies a linear dimensionality reduction, a plain Gram matrix is used for T in (26) to learn $\tilde{\mathbf{Q}}$. Low-rank output kernel learning, to some extent, is equivalent to multi-label dimensionality reduction (see a review by Ye et al.2011), whose target is to find a lower-dimensional latent space for multi-label space so as to capture inter-label dependencies as well as to remove nuisance noise.

5.2 Co-regularized Multi-view Learning

Co-regularization has been popularly employed in multi-view learning (Farquhar et al., 2005; Brefeld et al., 2006; Rosenberg & Bartlett, 2007). Essentially, co-regularization works as an extra model-complexity controller by penalizing functions which tend to generate big disagreements among multiple views (see pairwise $\mathcal{B}_{ij}$ in (29)). In particular, an improved generalization bound of using co-regularization was presented by Rosenberg & Bartlett (2007) in terms of Rademacher complexities. While most co-regularization is for semi-supervised learning, quite similar to our work, a subspace co-regularised multi-view learning paradigm was proposed by Guo & Xiao (2012) for supervised learning. A similar regularization is also used in MultiK-MHKS (Wang et al., 2008) for multiple kernel learning, which strategically integrates heterogeneous

information with an ensemble of kernels. Since MultiK-MHKS works on the original binary output space, the squared misclassification loss is used. However, a least-squares loss is used in KGHA as a regression on lower-dimensional representations of multi-label outputs.

5.3 Prediction

With training data $\mathcal{D} = [G_1, G_2, \ldots, G_{J-1}; T]$, we can obtain $J-1$ dual matrices $\{\boldsymbol{\alpha}\}_{j=1}^{J-1}$ and one linear output decoding matrix $\tilde{\mathbf{Q}} = T^\top \boldsymbol{\alpha}_J$. Then, given a test inputs $\mathcal{D}^{test} = [\dot{G}_1^\top, \dot{G}_2^\top, \ldots, \dot{G}_{J-1}^\top]$, the predicted lower-dimensional representation is

$$\dot{X} = \frac{1}{J-1} \sum_{j=1}^{J-1} \mathcal{K}_j(\dot{G}_j, G_j)\boldsymbol{\alpha}_j \tag{30}$$

where $\mathcal{K}_j(G_j, \dot{G}_j)$ is a cross kernel matrix. Then the score values of labels are computed as

$$\dot{T} = \dot{X}\tilde{\mathbf{Q}}^\top(\tilde{\mathbf{Q}}\tilde{\mathbf{Q}}^\top)^\dagger \tag{31}$$

where † denotes the Moore-Penrose pseudoinverse. Finally, labels can be predicted by retrieving top$-l$ (l is the desired number of labels) ranked score values.

6 Large-Scale KGHA Learning

The computation in each ALS iteration is dominated by the matrix inversion in (26); thus the time complexity of training KGHA is $O(JM^3)$. On the other hand, the space complexity is $O(JM^2)$. In modern machine learning tasks, it is not uncommon to come across databases with large numbers (*e.g.* millions) of training instances. Like other kernel-based methods, learning on such large-scale databases is challenging since the storage and computation of large kernel matrices will go beyond the memory of normal PCs. To enable KGHA for large-scale learning, in our experiments we used low-rank approximations of kernel matrices with Memory Efficient Kernel Approximation (MEKA, Si et al.2014). MEKA is essentially a block-wise Nyström algorithm by first clustering instances to obtain dense diagonal kernel blocks, and then obtaining rank-k (k is small) approximations of all diagonal blocks with Nyström algorithm and also off-diagonal blocks with regression. For the kernel matrix of the j-th feature,

$$\tilde{K}_j \approx W_j L_j W_j^\top \tag{32}$$

where $W_j = \oplus_{s=1}^{S} W_j^{(s)}$ (*i.e.* the direct sum of $W_j^{(s)}$ in S blocks, $W_j \in \mathbb{R}^{Sk \times Sk}$) and $L_j \in \mathbb{R}^{Sk \times Sk}$ consists of S^2 block-linking matrices. MEKA was reported to outperform other low-rank approximations by exploiting the block structure in kernel matrices (Si et al., 2014). By using MEKA, the space complexity of training KGHA decreases to $O(J(Mk + (ck)^2))$. In addition, to avoid the matrix inverse in (26), we employed *conjugate gradient* (CG) to solve the linear equation for each feature:

$$(K_j + c_j I_M)\boldsymbol{\alpha}_j = X \tag{33}$$

and consequently, time complexity of solving ALS is reduced to $O(JMk)$.

Dataset	#labels	#training instances	#test instances	#average labels
Corel5k	260	4500	500	3.3965
Espgame	268	18689	2081	4.6859
Iaprtc12	291	17665	1962	5.7187

Table 1: Statistics of three image-annotation benchmark datasets.

Feature	Dim	Source	Descriptor	Location
DenseHueV3H1	300	texture	Hue	dense
DenseSiftV3H1	3000	texture	Sift	dense
Gist	512	-	Holistic	-
HarrisHueV3H1	300	texture	Hue	Harris
HarrisSiftV3H1	3000	texture	Sift	Harris
HsvV3H1	5184	color	HSV	-
LabV3H1	5184	color	LAB	-
RgbV3H1	5184	color	RGB	-

Table 2: A summary of 8 heterogeneous visual features.

	Corel5K			Espgame			Iaprtc12		
p/d_J	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
0.2	26.1	30.7	28.2	30.1	18.8	23.1	35.9	24.7	29.3
0.4	30.8	35.1	32.8	33.7	24.4	28.3	38.2	25.2	30.4
0.5	**33.7**	**42.5**	**37.6**	**37.8**	**27.3**	**31.7**	40.1	**29.7**	**34.1**
0.6	29.1	38.6	33.2	33.1	26.8	29.6	40.7	26.4	32.0
0.8	28.5	38.1	32.6	31.9	25.5	28.3	**41.3**	26.5	32.2

Table 3: Performance of KGHA with different p values.

	Corel5K			Espgame			Iaprtc12		
Method	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)	P(%)	R(%)	F1(%)
MBRM	24.0	25.0	24.0	18.0	19.0	18.0	24.0	23.0	23.0
JEC	27.0	32.0	29.0	24.0	19.0	21.0	29.0	19.0	23.0
TagProp	**33.0**	42.0	**37.0**	39.0	**27.0**	**32.0**	45.0	**34.0**	**39.0**
FastTag	32.0	**43.0**	**37.0**	**46.0**	22.0	30.0	**47.0**	26.0	34.0
r-MLR	27.7	29.3	28.5	24.3	19.3	21.5	34.8	19.5	25.0
r-KMLR	31.7	35.1	33.3	24.8	26.6	25.7	36.6	20.1	26.0
L-HA	30.0	27.5	28.7	25.1	22.4	23.7	38.1	22.5	28.3
KGHA−d	33.1	38.1	37.5	32.7	27.2	29.7	**43.8**	21.3	28.7
KGHA−r	31.3	32.6	32.0	28.8	18.6	22.6	31.2	22.6	26.2
KGHA	**33.7**	**42.5**	**37.6**	**37.8**	**27.3**	**31.7**	40.1	**29.7**	**34.1**

Table 4: Comparison between KGHA and other related methods on three image-annotation benchmark databases. The results in the upper panel were reported by Chen et al. (2013).

7 Experiments

To evaluate the proposed KGHA for multi-label learning, we test it on image annotation and protein function prediction tasks.

7.1 Image Annotation

7.1.1 Data and Evaluation

In this experiment, we used three benchmark datasets, Corel5k, Espgame and Iaprtc12. These three datasets have been widely used in image annotation studies (Guillaumin et al., 2009; Makadia et al., 2010; Chen et al., 2013) with performance evaluations reported therein. Therefore, we can easily compare our method with others. Statistics of the three benchmark datasets are summarized in Table 1. Readers are referred to Makadia et al. (2010) for more details of the three datasets. We worked with 8 visual features extracted by Guillaumin et al. (2009). They include one Gist descriptor, three global color histograms and four histograms of local bag-of-words texture features[2]. The descriptions of 8 features are summarized in Table 2. Readers are referred to Guillaumin et al. (2009) for more details on extracting these features. Our large-scale learning scheme (section 6) is applied on Espgame and Iaprtc12 since these contain large numbers of instances.

Following Chen et al. (2013), 5 labels with top prediction score values were annotated to each image. We evaluated annotation performance using *precision* (P), *recall* (R), and the *F1* measure (F). For each tag, the precision is computed as the ratio of the number of images assigned the tag correctly over the total number of images predicted to have the tag, while the recall is the number of images assigned the tag correctly divided by the number of images that truly have the tag. Then precision and recall are averaged across all tags. Finally, the F1 measure is calculated as $F = 2\frac{P \times R}{P+R}$.

7.1.2 Results and Comparison

First, KGHA was implemented and tested on three databases. A Gaussian kernel $\mathcal{K}_j^{\text{Gauss}} = \exp(-||G_j^{(m)} - G_j^{(n)}||_2^2/2\sigma_j^2)$ was used on all 8 visual features, with σ_j set to the average value of $||G_j^{(m)} - G_j^{(n)}||_2, m, n \in [1, M]$. The reduced dimension p is set to different 5 values ($p = \{0.2, 0.4, 0.5, 0.6, 0.8\} \times d_J$). Hyperparameters $(\lambda, \{c_j\}_{j=1}^{J-1})$ were selected by grid search with 4-fold cross validation from $\{10^{-5}, \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$. Here for simplicity we use a common ridge parameter c for all J features, and we found it almost does not affect performance. Experimental results are presented in Table 3. It can be seen that the best performance was achieved with $p/d_J = 0.5$. on all three datasets. To verify the significance of low-rank output kernel learning and co-regularization, we also implemented another five simplified methods for comparison: (1) multivariate linear ridge regression (r-MLR); (2) multivariate kernel ridge regression (r-KMLR); (3) linear HA (L-HA); (4) KGHA with $p = d_J$ (KGHA$-$d, no dimensionality reduction); (5) KGHA with extremely small λ (KGHA$-$r, no manifold regularization). To ensure fairness, the same Gaussian kernel construction and appropriate hyperparameter searching are used in all methods. The results of all six methods are presented in the lower panel of Table 4. We see that KGHA generally outperforms other methods, which empirically proves the importance of low-rank output kernel learning and co-regularization. In addition, the upper panel of Table 4 lists the results of some notable methods that were recently developed or surveyed (Guillaumin et al., 2009; Makadia et al., 2010; Chen et al., 2013). KGHA demonstrates promising capabilities by comparing favorably to the state-of-the-art.

[2] In the original dataset, two versions of color features and texture features are available, with and without spatial layout; here we use only those with layout.

Dataset	information sources for kernels	#labels	#instances	#ave. labels
Human	domain composition complexes protein-DNA/RNA-interaction p-transcriptional modification tissue expression protein interaction OMIM disease Su-tissue expression	254	3704	3.7268
Yeast	Pfam domain structure co-participation PPI network gene interaction network cell cycle&gene expression	13	3588	1.5279

Table 5: The statistics and heterogeneous features of two datasets.

7.2 Protein Function Prediction

7.2.1 Data and Evaluation

Two datasets were used in the second experiment. The first one is Human from Mostafavi & Morris (2010) and the second one is Yeast from Tsuda et al. (2005). Following the same preprocessing on Human dataset by Yu et al. (2013), we filtered out some protein instances with too general or too narrow functions. For both datasets, multiple kernels (similarity networks) were already pre-computed. The statistics and heterogeneous features of two datasets are listed in Table 5.

For better comparison with the state-of-the-art results in bioinformatics community, we evaluated performances with $1-$RankingLoss (Yu et al., 2012, 2013). *RankingLoss* is computed as the average fraction of label pairs which are incorrectly ordered. To align with Macro F1, $1-$RankingLoss was used instead. The score values of all labels in (31) were used for ordering. In the Human experiment, 60% of the instances were used for training and 40% for testing. On the other hand, 80% of instances were used for training and 20% for testing in Yeast experiment. In both cases, 5 independent trials were run with different random subsets, based on which the average and standard deviation of $1-$RankingLoss were computed and reported.

7.2.2 Results and Comparison

Similarly to our image-annotation experiment, KGHA with five different p values were tried on Human and Yeast databases. Also, the same hyper-parameter tuning was conducted. Results are presented in Table 6, from where we can see that optimal p is $0.5 \times d_J$ in the Human experiment and $0.6 \times d_J$ in Yeast. Three simplified methods (r-MLR and L-HA were not used since only kernel values are provided in original datasets) were also tried on two datasets with results listed in the lower panel of Table 7. KGAH is again superior to all others, which suggests its robustness and stability. In addition, four state-of-the-art results on these two datasets (see the upper panel of Table 7) were reported in Yu et al. Yu et al. (2012, 2013). It can be seen that KGHA yielded rather comparable performance.

p/d_J	Human	Yeast
0.2	70.03 ($\pm$0.69)	71.58 ($\pm$0.63)
0.4	78.63 ($\pm$0.53)	76.00 ($\pm$0.89)
0.5	**82.74** ($\pm$0.56)	81.13 ($\pm$0.75)
0.6	77.14 ($\pm$0.42)	80.06 ($\pm$0.83)
0.8	76.31 ($\pm$0.61)	**83.97** ($\pm$0.80)

Table 6: Evaluation of KGHA using $1-$RankingLoss (%) with different p values.

Method	Human	Yeast
SW	78.49 ($\pm$0.58)	**84.35** ($\pm$0.73)
GRF-MK	81.11 ($\pm$0.51)	80.93 ($\pm$0.99)
PfunBG-MK	81.60 ($\pm$0.50)	80.50 ($\pm$0.87)
TMEC	**83.40** ($\pm$0.46)	80.79 ($\pm$1.01)
r-KMLR	72.10 ($\pm$0.65)	73.11 ($\pm$0.94)
KGHA$-$d	78.69 ($\pm$0.57)	75.25 ($\pm$0.92)
KGHA$-$r	77.15 ($\pm$0.78)	72.69 ($\pm$1.24)
KGHA	**81.74** ($\pm$0.56)	**83.97** ($\pm$0.80)

Table 7: Comparison between KGHA and other related methods using $1-$RankingLoss (%) on three image-annotation benchmark databases. The results in the upper panel were reported by Yu et al. (2013) and Yu et al. (2012).

8 Conclusion

A novel multi-label learning framework, kernel generalized homogeneity analysis (KGHA), was proposed. Starting from the connections between regular HA and multiple-set CCA, we revealed that HA can be generalized as a relaxed variant of multiple-set CCA to handle multiple heterogeneous features. By using kernel functions, we showed that KGHA, in multi-label learning, works as a method consisting of low-rank output kernel learning and co-regularized multi-view learning. We also presented some interesting links between low-rank output kernel learning and multi-label dimensionality reduction, co-regularization and multiple kernel learning, respectively. Promising results are achieved by using KGHA in our experiments on image annotation and protein function prediction.

References

Bie, Tijl De, Cristianini, Nello, and Rosipal, Roman. Eigenproblems in Pattern Recognition. In *Handbook of Geometric Computing: Applications in Pattern Recognition, Computer Vision, Neuralcomputing, and Robotics*, pp. 129–170. Springer, 2005.

Brefeld, Ulf, Gärtner, Thomas, Scheffer, Tobias, and Wrobel, Stefan. Efficient co–regularised least squares regression. In *Proceedings of the 23rd International Conference on Machine Learning*, 2006.

Bucak, Serhat Selcuk, Jin, Rong, and Jain, Anil K. Multiple kernel learning for visual object recognition: A review. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(7):1354–1369, 2014.

Chen, Minmin, Zheng, Alice, and Weinberger, Kilian Q. Fast image tagging. In *ICML*, 2013.

Dinuzzo, Francesco and Fukumizu, Kenji. Learning low-rank output kernels. In *ACML*, 2011.

Dinuzzo, Francesco, Ong, Cheng S., Gehler, Peter V., and Pillonetto, Gianluigi. Learning Output Kernels with Block Coordinate Descent. In *ICML*, 2011.

Elisseeff, André and Weston, Jason. A kernel method for multi-labelled classification. In *Advances in Neural Information Processing Systems (NIPS)*, 2002.

Farquhar, Jason D. R., Hardoon, David R., Meng, Hongying, Taylor, John S., and Szedmák, Sándor. Two view learning: SVM-2K, theory and practice. In *NIPS*, 2005.

Guillaumin, Matthieu, Mensink, Thomas, Verbeek, Jakob, and Schmid, Cordelia. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *ICCV*, 2009.

Guo, Yuhong and Xiao, Min. Cross language text classification via subspace co-regularized multi-view learning . In *ICML*, 2012.

Hardoon, David R., Szedmak, Sandor, and Shawe-Taylor, John. Canonical Correlation Analysis: An Overview with Application to Learning Methods. *Neural Computation*, 16(12):2639–2664, 2004.

Hariharan, Bharath, Zelnik-Manor, Lihi, Vishwanathan, S. V. N., and Varma, Manik. Large scale max-margin multi-label classification with priors. In *ICML*, 2010.

Hsu, Daniel, Kakade, Sham, Langford, John, and Zhang, Tong. Multi-label prediction via compressed sensing. In *NIPS*, 2009.

Izenman, Alan J. *Modern Multivariate Statistical Techniques : Regression, Classification, and Manifold Learning*. Springer New York, 2008.

Ji, Shuiwang and Ye, Jieping. Linear dimensionality reduction for multi-label classification. In *International Jont Conference on Artifical Intelligence*, IJCAI'09, 2009.

Lin, Zijia, Ding, Guiguang, Hu, Mingqing, and Wang, Jianmin. Multi-label classification via feature-aware implicit label space encoding. In *ICML-14*, 2014.

Ma, Yunqian and Fu, Yun. *Manifold Learning Theory and Applications*. CRC Press, Inc., Boca Raton, FL, USA, 1st edition, 2011. ISBN 1439871094, 9781439871096.

Makadia, Ameesh, Pavlovic, Vladimir, and Kumar, Sanjiv. Baselines for image annotation. *International Journal of Computer Vision*, 90:88–105, 2010.

Micchelli, Charles A. and Pontil, Massimiliano. On learning vector-valued functions. *Neural Computation*, 17(1):177–204, 2005.

Michailidis, George and de Leeuw, Jan. The Gifi System Of Descriptive Multivariate Analysis. *STATISTICAL SCIENCE*, 13:307–336, 1998.

Mostafavi, Sara and Morris, Quaid. Fast integration of heterogeneous data sources for predicting gene function with limited annotation. *Bioinformatics*, 26(14):1759–1765, 2010.

Rosenberg, David S. and Bartlett, Peter L. The rademacher complexity of co-regularized kernel classes. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics (AISTATS)*. Journal of Machine Learning Research - Proceedings Track, 2007.

Si, Si, jui Hsieh, Cho, and Dhillon, Inderjit. Memory efficient kernel approximation. In Jebara, Tony and Xing, Eric P. (eds.), *ICML-14*, 2014.

Sridharan, Karthik and Kakade, Sham M. An information theoretic framework for multi-view learning. In *21st Annual Conference on Learning Theory - COLT*, 2008.

Sun, Liang, Ji, Shuiwang, Yu, Shipeng, and Ye, Jieping. On the equivalence between canonical correlation analysis and orthonormalized partial least squares. In *IJCAI*, 2009.

Sun, Liang, Ji, Shuiwang, and Ye, Jieping. Canonical correlation analysis for multilabel classification: A least-squares formulation, extensions, and analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(1), January 2011.

Tsuda, Koji, Shin, Hyunjung, and Schölkopf, Bernhard. Fast protein classification with multiple networks. *Bioinformatics*, 21(2):59–65, January 2005.

Wang, Zhe, Chen, Songcan, and Sun, Tingkai. Multik-mhks: A novel multiple kernel learning algorithm. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(2), February 2008.

Xiong, Hanchen, Szedmák, Sándor, and Piater, Justus H. Joint SVM for accurate and fast image tagging. In *European Symposium on Artificial Neural Networks, ESANN*, 2014.

Ye, Jieping, Ji, Shuiwang, and Sun, Liang. *Multi-Label Dimensionality Reduction*. Chapman & Hall/CRC, 2011.

Yu, Guoxian, Domeniconi, Carlotta, Rangwala, Huzefa, Zhang, Guoji, and Yu, Zhiwen. Transductive multi-label ensemble classification for protein function prediction. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012.

Yu, Guoxian, Rangwala, Huzefa, Domeniconi, Carlotta, Zhang, Guoji, and Yu, Zhiwen. Protein function prediction using multi-label ensemble classification. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, 10(4), July 2013.