Multi-Output Regularized Projection

Kai Yu
Corporate Technology
Siemens AG, Germany
kai.yu@siemens.com

Shipeng Yu
Institute for Computer Science
University of Munich, Germany
spyu@dbs.informatik.uni-muenchen.de

Volker Tresp
Corporate Technology
Siemens AG, Germany
volker.tresp@siemens.com

Abstract

Dimensionality reduction via feature projection has been widely used in pattern recognition and machine learning. It is often beneficial to derive the projections not only based on the inputs but also on the target values in the training data set. This is of particular importance in predicting multivariate or structured outputs which is an area of growing interest. In this paper we introduce a novel projection framework which is sensitive to both input features and outputs. Based on the derived features prediction accuracy can be greatly improved. We validate our approach in two applications. The first is to model users' preferences on a set of paintings. The second application is concerned with image categorization where each image may belong to multiple categories. The proposed algorithm produces very encouraging results in both settings.

1 Introduction

Consider the pattern recognition task of predicting an output quantity y given an input feature vector $\mathbf{x}$. If the input space is high-dimensional and contains irrelevant features, the design of an appropriate pattern recognition system becomes a non-trivial problem. Thus it is desirable to employ a preprocessing step in which input features are first *projected* into a new feature space that is compact, noise-free, and highly indicative. As an outcome, learning algorithms based on the new features are often efficient and effective. Projection methods such as principal component analysis (PCA), linear discriminant analysis (LDA) (see [2]), canonical correlation analysis (CCA) (e.g., [3, 1]) and partial least squares (PLS) [9, 4] have been applied successively in various applications.

Among all the algorithms, PCA is probably the most common choice, which aims to find the principle components preserving the covariance structure of input features. However, the *unsupervised* manner indicates the uncovered components not necessary helpful for predictions.

In this paper we are interested in *supervised* projection methods, since it is often beneficial to ensure feature projections sensitive to the predicted quantities. In particular we consider a very general setting where the outputs are *multivariate*, i.e., for an example $\mathbf{x}$ the corresponding output is a vector $\mathbf{y} = [y_1, \ldots, y_L]^\intercal$. Note that the usual univariate output is a special case of the framework.

Multi-output problem is very common in real-world applications, typically involving multiple predictive tasks based on the same input space. One example is to model people's preferences on a set of products. This is a typical multi-output problem since for each product many persons' preferences have to be estimated. Since people's tastes are usually correlated, it is desired to the interdependency between individuals. Another example is the problem of multi-label image categorization, where each image is allowed to be associated with multiple categories, which often have semantic correlations. There are many other multi-output problems where the dependency of outputs should be explored. For instance, tracking the positions of different parts of a person's hands or arms has multi-dimensional outputs which are dependent of each other, since the freedoms of different parts are mutually restricted.

In all these applications it is desired to exploit the dependency between the multiple outputs for prediction and multivariate data analysis. This paper introduces a novel framework, *multi-output regularized projection* (MORP), which maps the input features into a new feature space that not only retains the information of inputs, but also captures the dependency of outputs as well. The algorithm exposes the *inherent structure* of input features which are highly informative for predictions.

The paper is organized as follows. In Section 2 we formulate the data projection as an optimization problem in the linear case and then propose an regularized version to prevent overfitting, which is generalized to nonlinear mapping by using kernels. Then we discuss it connections to related work in Section 3. Finally we report the experiments in Section 4 and conclude the paper in Section 5.

2 Multi-Output Regularized Projection

We consider a set of N examples. For $i = 1, \ldots, N$, each example i is described by an M-dimensional feature vector $\mathbf{x}_i \in \mathcal{X}$, and is associated with an L-dimensional output vector $\mathbf{y}_i \in \mathcal{Y}$. We denote the input data as a matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^\intercal \in \mathbb{R}^{N \times M}$, and the output data as $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]^\intercal \in \mathbb{R}^{N \times L}$, where $[\cdot]^\intercal$ denotes matrix transpose. We aim to derive a mapping $\Psi : \mathcal{X} \mapsto \mathcal{V}$ that projects the input features into a K-dimensional latent space.

2.1 A Common Latent-Variable Model

We propose to project input features into a new feature space $\mathcal{V}$ that preserves the statistical structure of $\mathbf{X}$ as much as possible, and meanwhile explains $\mathbf{Y}$ very well. Thus we solve the following optimization problem:

$$\begin{aligned} \min_{\mathbf{A},\mathbf{B},\mathbf{V}} \quad & (1-\beta)\|\mathbf{X} - \mathbf{V}\mathbf{A}\|^2 + \beta\|\mathbf{Y} - \mathbf{V}\mathbf{B}\|^2 \quad (1) \\ \text{subject to:} \quad & \mathbf{V}^\intercal\mathbf{V} = \mathbf{I}, \end{aligned}$$

where $\mathbf{V} \in \mathbb{R}^{N \times K}$ gives the K-dimensional *projections* of examples. $\mathbf{A} \in \mathbb{R}^{K \times M}$, $\mathbf{B} \in \mathbb{R}^{K \times L}$ are the *factor loadings* for $\mathbf{X}$ and $\mathbf{Y}$, respectively. $0 \leq \beta \leq 1$ is a tuning parameter determining how much the indexing should be biased by the outputs. $\mathbf{V}^\intercal\mathbf{V} = \mathbf{I}$ restricts the K latent variables to be linearly independent and have identical variances. Clearly, the cost function is a trade-off between the *reconstruction error* of both $\mathbf{X}$ and $\mathbf{Y}$. The following proposition states the interdependency between $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{V}$ at the optimum.

Proposition 2.1. *If $\mathbf{V}, \mathbf{A}$ and $\mathbf{B}$ are the optimal solutions to the problem* (1)*, then (i) $\mathbf{A} = \mathbf{V}^\intercal\mathbf{X}$, $\mathbf{B} = \mathbf{V}^\intercal\mathbf{Y}$; (ii) At the optimum, the objective function in* (1) *equals to $(1-\beta)\|\mathbf{X}\|_F^2 + \beta\|\mathbf{Y}\|_F^2 - Tr\big[\mathbf{V}^\intercal\mathbf{K}\mathbf{V}\big]$, where $Tr[\cdot]$ is the trace of a matrix, and $\mathbf{K} = (1-\beta)\mathbf{X}\mathbf{X}^\intercal + \beta\mathbf{Y}\mathbf{Y}^\intercal$.*

To improve readability, we put all proofs into the appendix. Since $\|\mathbf{X}\|_F^2$ and $\|\mathbf{Y}\|_F^2$ are both fixed, and the objective function is convex, Proposition 2.1 suggests that the problem (1) can be considered to be an optimization problem only with respect to $\mathbf{V}$:

$$\begin{aligned} \max_{\mathbf{V}} \quad & \mathrm{Tr}\big[\mathbf{V}^\intercal\mathbf{K}\mathbf{V}\big] \quad (2) \\ \text{subject to:} \quad & \mathbf{V}^\intercal\mathbf{V} = \mathbf{I} \end{aligned}$$

Note an ambiguity arises in (1) and (2). If $\mathbf{V}$ is the solution, then $\mathbf{V}' = \mathbf{V}\mathbf{R}$ is also a solution, given an arbitrary rotation matrix $\mathbf{R}$. The following theorem summarizes the situation.

Theorem 2.2. *If $[\tilde{\mathbf{v}}_1, \ldots, \tilde{\mathbf{v}}_N]$ are the eigenvectors of matrix $\mathbf{K}$, and $\lambda_1 \geq \ldots \geq \lambda_N$ are the corresponding eigenvalues , then (i) the maximum of the objective function* (2) *is $\sum_{i=1}^K \lambda_i$; (ii) $\mathbf{V}$ has the form $[\tilde{\mathbf{v}}_1, \ldots, \tilde{\mathbf{v}}_K]\mathbf{R}$, where $\mathbf{R}$ is an arbitrary $K \times K$ orthogonal rotation matrix.*

To remove the ambiguity, we are focusing on the solutions given by the eigenvectors without any rotation, i.e. $\mathbf{v}_j = \tilde{\mathbf{v}}_j, j = 1, \ldots, N$. Thus the original optimization problem (1) has an *equivalent* form: [1]

$$\begin{aligned} \max_{\mathbf{v} \in \mathbb{R}^N} \quad & \mathbf{v}^\intercal\mathbf{K}\mathbf{v} \quad (3) \\ \text{subject to :} \quad & \mathbf{v}^\intercal\mathbf{v} = 1, \end{aligned}$$

By setting the Lagrange's derivative to be zero, we obtain the standard form of an eigenvalue problem $\mathbf{K}\mathbf{v} = \lambda\mathbf{v}$. Let $\mathbf{v}_1, \ldots, \mathbf{v}_N$ be the eigenvectors of $\mathbf{K}$ with the eigenvalues sorted in a *non-increasing* order, then *an* optimal solution to (1) is given as $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_K]$, $\mathbf{A} = \mathbf{X}\mathbf{V}$, and $\mathbf{B} = \mathbf{Y}\mathbf{V}$.

2.2 Multi-Output Regularized Projection

Instead of uncovering the latent projections of observed examples, this paper focuses on learning the *mapping functions* $\Psi : \mathcal{X} \mapsto \mathcal{V}$ that are able to map *new* input features into a meaningful space, thus we restrict the latent variables as *linear mappings* of $\mathbf{X}$, and solve the following problem

$$\begin{aligned} \min_{\mathbf{A},\mathbf{B},\mathbf{V}} \quad & (1-\beta)\|\mathbf{X} - \mathbf{V}\mathbf{A}\|^2 + \beta\|\mathbf{Y} - \mathbf{V}\mathbf{B}\|^2 \quad (4) \\ \text{subject to:} \quad & \mathbf{V}^\intercal\mathbf{V} = \mathbf{I}, \quad \mathbf{V} = \mathbf{X}\mathbf{W} \end{aligned}$$

where $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_K] \in \mathbb{R}^{M \times K}$. Plugging $\mathbf{v} = \mathbf{X}\mathbf{w}$ into (3), we have an optimization problem with respect to $\mathbf{w}$

$$\begin{aligned} \max_{\mathbf{w} \in \mathbb{R}^M} \quad & \mathbf{w}^\intercal\mathbf{X}^\intercal\mathbf{K}\mathbf{X}\mathbf{w} \quad (5) \\ \text{subject to :} \quad & \mathbf{w}^\intercal\mathbf{X}^\intercal\mathbf{X}\mathbf{w} = 1 \end{aligned}$$

Setting the derivative of its Lagrange with respect to $\mathbf{w}$ to be zero, we reach a generalized eigenvector problem[2]:

$$\mathbf{X}^\intercal\mathbf{K}\mathbf{X}\mathbf{w} = \lambda\mathbf{X}^\intercal\mathbf{X}\mathbf{w} \quad (6)$$

which produces M generalized eigenvectors $\mathbf{w}_1, \ldots, \mathbf{w}_M$, as well as the eigenvalues $\lambda_1 \geq \ldots \geq \lambda_M$. The first K eigenvectors are used to form the mapping functions

$$\psi_j(\mathbf{x}) = \sqrt{\lambda_j}\mathbf{w}_j^\intercal\mathbf{x}, \quad j = 1, \ldots, K \quad (7)$$

where the scaling with $\sqrt{\lambda_j}$ reflects the relative importance of projection dimensions. Finally $\Psi(\mathbf{x}) = [\psi_1(\mathbf{x}), \ldots, \psi_K(\mathbf{x})]^\intercal$ maps $\mathbf{x}$ into a K-dimensional space.

[1] Solving the problem (3) itself only gives the first eigenvector $\mathbf{v}_1$ of $\mathbf{K}$. The full optimization problem should be recursively computing $\mathbf{v}_j$ by maximizing $\mathbf{v}^\intercal\mathbf{K}\mathbf{v}$ with the constraint $\mathbf{v}^\intercal\mathbf{v} = 1$ and $\mathbf{v} \perp \mathrm{span}\{\mathbf{v}_1, \ldots, \mathbf{v}_{j-1}\}$. Here we state the problem as (3) for simplicity and also because its Lagrange directly leads to the eigenvalue problem.

[2] In this paper we abuse the notation λ in all the eigenvalue problems. However their meanings are clear in the respective contexts.

2.3 Overfitting and Regularization

However, similar to other linear systems, the learned mapping functions can be ill-posed when $\mathbf{X}$ has the rank lower than M, which typically happens when the dimensionality of input features is very high, namely $N \ll M$. Under a mild assumption[3] $\text{rank}(\mathbf{K}) = N$, the maximization in (3) is equivalent to minimizing $\mathbf{v}^\intercal \mathbf{K}^{-1} \mathbf{v}$. Then we regularize the problem (5) as the following

$$\begin{aligned} \min_{\mathbf{w} \in \mathbb{R}^M} \quad & \mathbf{w}^\intercal \mathbf{X}^\intercal \mathbf{K}^{-1} \mathbf{X} \mathbf{w} + \gamma \|\mathbf{w}\|^2 \\ \text{subject to :} \quad & \mathbf{w}^\intercal \mathbf{X}^\intercal \mathbf{X} \mathbf{w} = 1 \end{aligned} \tag{8}$$

where $\|\mathbf{w}\|^2 = \mathbf{w}^\intercal \mathbf{w}$ is the Tickhonov regularizer [7] typically applied in ill-posed problems, and γ is a nonnegative scalar which is usually very small. The corresponding generalized eigenvalue problem is

$$\left[\mathbf{X}^\intercal \mathbf{K}^{-1} \mathbf{X} + \gamma \mathbf{I}\right] \mathbf{w} = \lambda \mathbf{X}^\intercal \mathbf{X} \mathbf{w} \tag{9}$$

which gives generalized eigenvectors $\mathbf{w}_1, \ldots, \mathbf{w}_M$ with eigenvalues $\lambda_1 \leq \ldots \leq \lambda_M$. Note since the objective is the inverse of some maximization problem, we sort eigenvalues in a *non-decreasing* order, and take the first K eigenvectors to form the mapping.

2.4 Nonlinear Projections

The following theorem implies that we can also derive a nonlinear mapping Ψ using the *kernel trick*.

Theorem 2.3. *If $\mathbf{w}$ is an eigenvector of the generalized eigenvalue problem* (8)*, then there exists $\boldsymbol{\alpha} \in \mathbb{R}^N$ such that $\mathbf{w} = \mathbf{X}^\intercal \boldsymbol{\alpha} = \sum_{i=1}^N (\boldsymbol{\alpha})_i \mathbf{x}_i$. If $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ are linearly independent, such an $\boldsymbol{\alpha}$ is unique.*

Let $\mathcal{X}$ be a reproducing kernel Hilbert space (RKHS) with the *kernel* function $\kappa_x(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$, then based on Theorem 2.3 we have $\mathbf{v} = \mathbf{X}\mathbf{w} = \mathbf{X}\mathbf{X}^\intercal \alpha = \mathbf{K}_x \boldsymbol{\alpha}$ where $\mathbf{K}_x$ is the $N \times N$ kernel matrix with $(\mathbf{K}_x)_{i,j} = \kappa_x(\mathbf{x}_i, \mathbf{x}_j)$. Then an equivalent form of (8) is

$$\begin{aligned} \min_{\boldsymbol{\alpha} \in \mathbb{R}^N} \quad & \boldsymbol{\alpha}^\intercal \mathbf{K}_x \mathbf{K}^{-1} \mathbf{K}_x \boldsymbol{\alpha} + \gamma \boldsymbol{\alpha}^\intercal \mathbf{K}_x \boldsymbol{\alpha} \\ \text{subject to :} \quad & \boldsymbol{\alpha}^\intercal \mathbf{K}_x^2 \boldsymbol{\alpha} = 1 \end{aligned} \tag{10}$$

where $\mathbf{K} = (1-\beta)\mathbf{K}_x + \beta \mathbf{Y}\mathbf{Y}^\intercal$. The corresponding generalized eigenvalue problem is

$$\left[\mathbf{K}_x \mathbf{K}^{-1} \mathbf{K}_x + \gamma \mathbf{K}_x\right] \boldsymbol{\alpha} = \lambda \mathbf{K}_x^2 \boldsymbol{\alpha} \tag{11}$$

With the eigenvalues sorted as $\lambda_1 \leq \ldots \leq \lambda_N$, the first K eigenvectors $[\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_K]$ give the mappings, where the j-th function is $\psi_j(\mathbf{x}) = \mathbf{w}_j^\intercal \mathbf{x} = \sum_{i=1}^N (\boldsymbol{\alpha}_j)_i \kappa_x(\mathbf{x}_i, \mathbf{x})$. Now the algorithm is readily able to deal with *nonlinear* mappings. We consider a nonlinear function $\Phi : \mathcal{X} \mapsto \mathcal{F}$, which maps $\mathbf{x}$ into a high-dimensional or even infinite-dimensional feature space $\mathcal{F}$, and let $\mathbf{X} = [\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_N)]$, then the kernel is accordingly defined as $\kappa_x(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$. Finally, we can directly work with kernels (e.g. Gaussian kernel), without knowing $\phi(\cdot)$ explicitly.

[3] The assumption is particular true when $\mathbf{x}$ are in a reproducing kernel Hilbert space (RKHS) such that the inner product $\langle \cdot, \cdot \rangle$ defines a positive definite kernel.

2.5 Structured Outputs

Sometimes the outputs are not just vector-valued, but also have some complex structure like sequences or graphs. Similar to the case of $\mathbf{X}$, one can consider an proper kernel $\kappa_y(\mathbf{y}_i, \mathbf{y}_j) = \langle \varphi(\mathbf{y}_i), \varphi(\mathbf{y}_j) \rangle$ to characterize the structure of outputs, where $\varphi(\cdot)$ maps output vectors $\mathbf{y}$ into a RKHS space. Let $\mathbf{Y} = [\varphi(\mathbf{y}_1), \ldots, \varphi(\mathbf{y}_N)]$, then

$$\mathbf{K} = (1-\beta)\mathbf{K}_x + \beta \mathbf{Y}\mathbf{Y}^\intercal = (1-\beta)\mathbf{K}_x + \beta \mathbf{K}_y \tag{12}$$

In various problems, we can design the $\kappa_y(\cdot, \cdot)$ tailored to the nature of data. This is a very general setting where *nonlinear* dependency of outputs can be explored. The methods discussed in the previous sections are special cases which use the linear kernel $\kappa_y(\mathbf{y}_i, \mathbf{y}_j) = \langle \mathbf{y}_i, \mathbf{y}_j \rangle$.

2.6 The Algorithm

It is usually convenient to seek for the eigenvectors with the largest eigenvalues, which is numerically stabler and more efficient. Thus we transform the optimization problem (10) to obtain another equivalent form. Let $\mathbf{v} = \mathbf{K}_x \boldsymbol{\alpha}$, then problem (10) becomes

$$\begin{aligned} \min_{\mathbf{v}} \quad & \mathbf{v}^\intercal (\mathbf{K}^{-1} + \gamma \mathbf{K}_x^{-1}) \mathbf{v} \\ \text{subject to :} \quad & \mathbf{v}^\intercal \mathbf{v} = 1 \end{aligned} \tag{13}$$

After some matrix derivation, we can get $(\mathbf{K}^{-1} + \gamma \mathbf{K}_x^{-1})^{-1} = \mathbf{K}(\gamma \mathbf{K} + \mathbf{K}_x)^{-1} \mathbf{K}_x$. Then the objective in (13) becomes maximizing $\mathbf{v}^T \mathbf{K} (\gamma \mathbf{K} + \mathbf{K}_x)^{-1} \mathbf{K}_x \mathbf{v}$, which leads to the following standard eigenvalue problem:

$$\mathbf{K}(\gamma \mathbf{K} + \mathbf{K}_x)^{-1} \mathbf{K}_x \mathbf{v} = \lambda \mathbf{v} \tag{14}$$

In practice, since γ is usually very small, the eigenvalue problem (14) can be approximated as $\mathbf{K}\mathbf{v} = \lambda \mathbf{v}$. Compared to (14), the simplified version is much more efficient, since the multiplication of matrices and matrix inverse are both removed. After obtaining the leading eigenvectors $\mathbf{v}_j$, $j = 1, \ldots, k$ with the *largest* eigenvalues λ_j (due to the maximization), we can recover the coefficient vectors as

$$\boldsymbol{\alpha}_j = \mathbf{K}_x^{-1} \mathbf{v}_j, \quad j = 1, \ldots, K \tag{15}$$

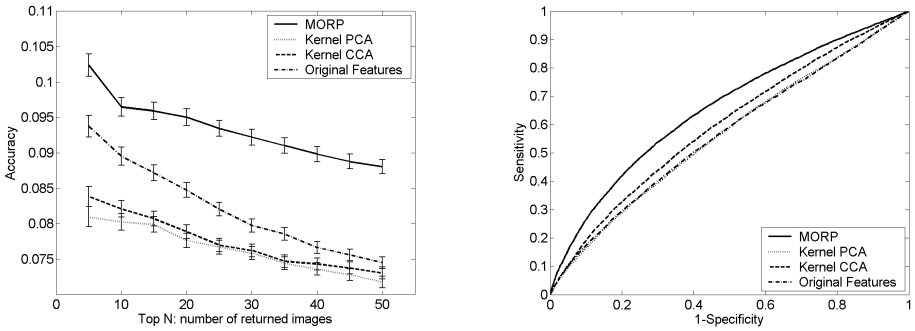

Figure 1. Comparison of feature projection methods in predicting user preferences: Top-N accuracy (right) and ROC curve(left).

Finally, incorporating the eigenvalues that reflect the relative importance of latent dimensions, we obtain the final feature mapping functions

$$\psi_j(\mathbf{x}) = \sqrt{\lambda_j} \sum_{i=1}^{N} (\boldsymbol{\alpha}_j)_i \kappa_x(\mathbf{x}_i, \mathbf{x}), \quad j = 1, \ldots, K. \quad (16)$$

3 Discussions and Related Work

The proposed framework MORP becomes identical to PCA if $\beta = 0$. This connection is also clear from (11), which becomes identical to kernel PCA [5] when $\beta = 0$. In the other extreme case $\beta = 1$, the feature mapping is enforced to entirely explain the dependency of outputs. Then the MORP algorithm is in spirit similar to kernel dependency estimation (KDE) [8], which first performs PCA on $\mathbf{K}_y$, and then uses input features to regress the eigenvectors. Due to the regularization in the post-regression phase, the uncovered projections are usually not orthogonal. In contrast to KDE's two-step strategy, our algorithm is derived in a single optimization framework. The proposed MORP is a very general framework. Compared with PCA, it makes the projections sensitive to output quantities, which is desired in supervised learning tasks. Compared with KDE, MORP retains the structure of the input features and thus prevents to be overfitted by the outputs, which makes the derived mapping functions more stable and potentially generalizable to new output dimensions.

In the literature there are some other supervised projection methods, like linear discriminant analysis (LDA) (e.g., [6]), canonical correlation analysis (CCA) (e.g., [3, 1]) and partial least squares (PLS) [9, 4]. MORP substantially differs from them. LDA is focusing on single classification problem where the output is one-dimensional. CCA finds the correlations between two representatives of the same examples (e.g., inputs $\mathbf{X}$ and outputs $\mathbf{Y}$ in our setting) by minimizing $\|\mathbf{v}_x - \mathbf{v}_y\|^2$ subject to both $\mathbf{v}_x$ and $\mathbf{v}_y$ being unitary and linear mappings of $\mathbf{x}_i$ and $\mathbf{y}_i$ (see a recent discussion in [1]). However, it does not require the projections $\mathbf{v}_x$ and $\mathbf{v}_y$ to promise low-reconstruction error of $\mathbf{x}$ and $\mathbf{y}$ and thus ignores the *intra* correlation of either. Instead, MORP takes into account all the inter and intra dependencies, since the projections minimize the reconstruction error of inputs and outputs simultaneously. PLS can be seen as penalized CCA (see [6]), which purely focuses on the regression of known output quantities, while does not consider the generalization for new dimension of outputs.

4 Empirical Study

We evaluate the proposed MORP on preference prediction and image categorization. In both settings each example is an image, from which *color histogram* (216-dim.), *correlagram* (256-dim.), *first and second color moments* (9-dim.) and *Pyramid wavelet texture* (10-dim.) are extracted to form a 491-dimensional feature vector $\mathbf{x}$. In both settings $\mathbf{K}_x$ is based on RBF kernel while $\mathbf{K}_y$ is based on linear kernel, and both matrices are re-scaled to ensure equal traces. For MORP $\beta = 0.5$ and $\gamma = 0.001$. Note that γ is found uncritical as long as it is very small.

4.1 Preference Prediction

We collected 190 users' ratings on 642 paintings in a survey, where each user expresses "like" and "dislike" for some randomly presented paintings. On average each user had rated 89 paintings, thus there are missing entries in $\mathbf{Y}$. This is a typical multi-output classification problem, since for each paining many users' opinions need to be predicted. We examine the performance of various feature projection algorithms that map the original features into a 20-dim. space, where the new features are fed into SVMs. In the experiment, a set of users are selected as *test users*. For each test user, we withdraw some ratings so that 20 ratings are left, then a SVM trained on the 20 examples is employed to predict the rest of ratings. We compare MORP with kernel CCA and kernel PCA. The two supervised methods,

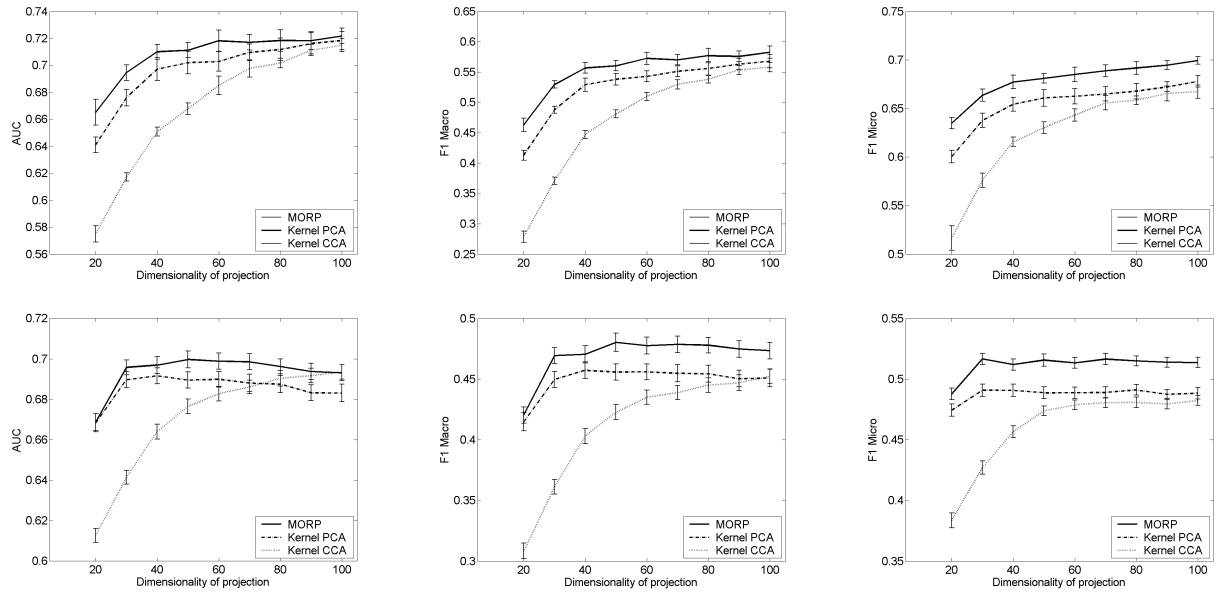

Figure 2. Image categorization accuracy under various dimensionality of feature projections. The upper panels correspond to the setting (I) where feature projection and categorization are based on the same set of categories; The lower panels present the setting (II) where feature projection and categorization are respectively based on two different sets of categories.

i.e., MORP and CCA, make use of the 190-dimensional outputs, with the missing and withdrawn entries filled with zeros. The derived new features are fed into SVMs with linear kernels. We also perform SVMs employing the same RBF kernel with the original 491-dim. features.

The first metric for evaluation is *Top-N accuracy*, i.e. the proportion of truly liked paintings among the N top-ranked paintings. Due to the missing entries in $\mathbf{y}$, we actually count the fraction of *known* liked paintings in the top ranked N paintings. The quantity is smaller than the true accuracy because *unknown* liked paintings are missing in the measurement. However, in our survey, the presenting of paintings to users is completely random, thus the distributions of rated/unrated paintings in both unranked and ranked lists are also random. This randomness dose not change the relative performances of the studied methods and thus the comparison still makes sense. The other metric is the *ROC curve*, which reflects the ranking quality of predictions and *insensitive* to the missing entries.

The experiment employs 10-fold cross validation, in which each fold is set as active users. For each active user the accuracy is averaged over 10 tests—in each time the 20 seen ratings are randomized. Finally the mean and variance over the 10 folds are presented in Figure 1. MORP significantly outperforms others in terms of both accuracy and ROC, because it explores the dependency between users. We also found that supervised projections, i.e., MORP and CCA, are generally better than unsupervised PCA. Note that in the left panel the ROC curves of PCA and original features are almost overlapped.

4.2 Image Categorization

The experiment is based on a subset of Corel image database, containing 1021 images that have been manually assigned into 35 categories based on their contents. In average, each image belongs to 3.6 categories and each category contains 98 positive examples. We treat each category as a binary classification problem. We employ AUC score and macro/micro F1 value to measure the accuracy. AUC is the size of area under the ROC curve, ranging from 0 to 1. F1 measures have been widely used in text categorization which combines precision and recall and is suitable when positive examples are much less than negative ones. Macro F1 is the simple average over all the categories while micro F1 is average weighted by the size of positive examples in each category. In all the cases larger values indicating better performances.

In each run of the experiment, we randomly pick up 25 categories and have 500 examples labeled. Projection methods with RBF kernels are trained on the 500 examples to learn the mapping functions, which are then employed to compute new features for all the 1021 images. In setting (I) we train linear SVM classifiers to predict the rest 521 images' labels, while in setting (II) we perform classification with 5-fold cross validation on the unlabeled 521 images with respect to the remaining 10 categories (one fold training and 4 folds test). Note that the second setting examines

the generalization of supervised projection methods on new output dimensions. The whole experiments are repeated by 10 runs with randomization, and the classification accuracy under different dimensionality of projections are shown in Figure 2. We can see that MORP outperforms CCA and PCA in all the cases. In particular, the results in the lower panels indicate that the features derived by MORP are generalized well to new predictive problems.

5 Summary and Conclusions

In this paper we propose a novel feature projection algorithm for predicting multivariate outputs. The projections retain the statistical structure of not only input features but also the outputs. We present the kernel version of the mappings such that nonlinear dependency can be captured. The algorithm achieves very good results in user preference prediction and image categorization. Currently we mainly exploit the linear dependency of outputs in the empirical study. As suggested in Section 2.5, the algorithm is generally applicable for outputs with richer structures, like sequences or graphs. In the future its applications to modeling structured outputs should be further studied.

Appendix

Proof. (**Proposition 2.1**) Applying the rule $\|\mathbf{C}\|^2 = \mathrm{Tr}\left[\mathbf{C}\mathbf{C}^\intercal\right]$ for an arbitrary matrix $\mathbf{C}$, we obtain

$$\begin{aligned} J(\mathbf{A},\mathbf{B},\mathbf{V}) &= (1-\beta)\|\mathbf{X}-\mathbf{V}\mathbf{A}\|^2 + \beta\|\mathbf{Y}-\mathbf{V}\mathbf{B}\|^2 \\ &= (1-\beta)\mathrm{Tr}\left[\mathbf{X}\mathbf{X}^\intercal - 2\mathbf{V}\mathbf{A}\mathbf{X}^\intercal + \mathbf{V}\mathbf{A}\mathbf{A}^\intercal\mathbf{V}^\intercal\right] \\ &\quad + \beta\mathrm{Tr}\left[\mathbf{Y}\mathbf{Y}^\intercal - 2\mathbf{V}\mathbf{B}\mathbf{Y}^\intercal + \mathbf{V}\mathbf{B}\mathbf{B}^\intercal\mathbf{V}^\intercal\right]. \end{aligned}$$

Setting the partial derivative of J with respect to $\mathbf{A}$ and $\mathbf{B}$ be zero respectively, we have $\mathbf{A} = \mathbf{V}^\intercal\mathbf{X}$ and $\mathbf{B} = \mathbf{V}^\intercal\mathbf{Y}$, which proves (i). Then we use the results (i) to replace $\mathbf{A}$ and $\mathbf{B}$ in J and obtain $J_{\mathrm{opt}} = \mathrm{Tr}\left[\mathbf{K}\right] - \mathrm{Tr}\left[\mathbf{V}^\intercal\mathbf{K}\mathbf{V}\right]$, which concludes (ii). □

Proof. (**Theorem 2.2**) The Lagrange of problem (2) is

$$L(\mathbf{V},\tilde{\mathbf{\Lambda}}) = \sum_{i=1}^{K}\mathbf{v}_i^\intercal\mathbf{K}\mathbf{v}_i - 2\sum_{i\neq j}\tilde{\lambda}_{i,j}\mathbf{v}_i^\intercal\mathbf{v}_j - \sum_{j=1}^{K}\tilde{\lambda}_{j,j}(\mathbf{v}_i^\intercal\mathbf{v}_i - 1)$$

where $(\tilde{\mathbf{\Lambda}})_{i,j} = \tilde{\lambda}_{i,j}$ and $\mathbf{V} = [\mathbf{v}_1,\ldots,\mathbf{v}_K]$. Setting its derivative with respect to $\mathbf{v}_j$ to be zero, we obtain

$$\frac{\partial L}{\partial \mathbf{v}_j} = 2\mathbf{K}\mathbf{v}_j - 2\sum_{i=1}^{K}\tilde{\lambda}_{i,j}\mathbf{v}_j = 0, \quad j = 1,\ldots,K,$$

which can be rewritten as $\mathbf{K}\mathbf{V} = \mathbf{V}\tilde{\mathbf{\Lambda}}$. Since $\tilde{\mathbf{\Lambda}}$ is a symmetric matrix, we have $\tilde{\mathbf{\Lambda}} = \mathbf{R}^\intercal\mathbf{\Lambda}\mathbf{R}$ where $\mathbf{\Lambda}$ is a diagonal matrix and $\mathbf{R}$ is an orthogonal rotation matrix satisfying $\mathbf{R}\mathbf{R}^\intercal = \mathbf{R}^\intercal\mathbf{R} = \mathbf{I}$. Then $\mathbf{K}\mathbf{V} = \mathbf{V}\mathbf{R}^\intercal\mathbf{\Lambda}\mathbf{R}$ yields $\mathbf{K}\mathbf{V}\mathbf{R} = \mathbf{V}\mathbf{R}\mathbf{\Lambda}$. Since $\mathbf{\Lambda}$ is diagonal, it is easy to see that the columns of $\tilde{\mathbf{V}} = \mathbf{V}\mathbf{R}^\intercal$ are the eigenvectors of $\mathbf{K}$. Thus the optimal $\mathbf{V}$ is formed by an arbitrary rotation of $\mathbf{K}$'s eigenvectors, i.e. $\mathbf{V} = \tilde{\mathbf{V}}\mathbf{R}$. Inserting $\mathbf{V}$ back to the objective function, then the value of objective function are $\mathrm{Tr}(\mathbf{\Lambda})$, i.e., sum of the K corresponding eigenvalues of $\mathbf{K}$. It is easy to see the maximal $\mathrm{Tr}(\mathbf{\Lambda})$ is the sum of the K largest eigenvalues, which proofs (i). In this case, $\mathbf{V}$ is an arbitrary rotation of the K largest eigenvectors, thus conclusion (ii) holds. □

Proof. (**Theorem 2.3**) Let $J(\mathbf{w})$ denote the cost function in (8). Obviously $J(\mathbf{w})$ achieves the minimum at the first eigenvector $\mathbf{w} = \mathbf{w}_1$ of the generalized eigenvalue problem (9). Consider $\mathbf{w}_\parallel$ as the projection of $\mathbf{w}_1$ on the subspace span$\{\mathbf{x}_1,\ldots,\mathbf{x}_N\}$. Then we can write $\mathbf{w}_1 = \mathbf{w}_\parallel + \mathbf{w}_\perp$, where $\mathbf{w}_\perp$ is orthogonal to the subspace spanned by $\mathbf{x}_i$. Then we have $\mathbf{w}_1^\intercal\mathbf{x}_i = \mathbf{w}_\parallel^\intercal\mathbf{x}_i + \mathbf{w}_\perp^\intercal\mathbf{x}_i$. Since $\|\mathbf{w}_1\|^2 = \|\mathbf{w}_\parallel\|^2 + \|\mathbf{w}_\perp\|^2 \geq \|\mathbf{w}_\parallel\|^2$, then $J(\mathbf{w}_1) \geq J(\mathbf{w}_\parallel)$. However, $J(\mathbf{w}_1)$ achieves the minimum, meaning $J(\mathbf{w}_1) \leq J(\mathbf{w}_\parallel)$. Therefore $J(\mathbf{w}_1) = J(\mathbf{w}_\parallel)$, and $\mathbf{w}_\perp = 0$. So far we have proofed that the first eigenvector (with the smallest eigenvalue) is a linear combination of $\mathbf{x}_i$. Given eigenvectors $\mathbf{w}_j$, $j = 1,\ldots,n-1$, it is known that the n-th eigenvector is obtained by first deflating the matrix $\mathbf{K}^\dagger = \mathbf{K}^{-1} - \sum_{j=1}^{n-1}\lambda_j\mathbf{X}^\intercal\mathbf{w}_j\mathbf{w}_j^\intercal\mathbf{X}$ and then solving the problem $\min_{\mathbf{w}\in\mathbb{R}^M} \quad \mathbf{w}^\intercal\mathbf{X}\mathbf{K}^\dagger\mathbf{X}^\intercal\mathbf{w} + \gamma\|\mathbf{w}\|^2$, subject to: $\mathbf{w}^\intercal\mathbf{X}^\intercal\mathbf{X}\mathbf{w} = 1$. Following the same procedure as before we proof that the eigenvector $\mathbf{w}_n$ also lies in the span of $\mathbf{x}_i$. □

References

[1] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis; an overview with application to learning methods. Technical Report CSD-TR-03-02, Royal Holloway University of London, 2003.

[2] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Satatistical Learning*. Springer Verlag, 2001.

[3] H. Hotelling. Relations between two sets of variables. *Biometrika*, 28:321–377, 1936.

[4] R. Rosipal and L. J. Trejo. Kernel partial least squares regression in reproducing kernel hilbert space. *Journal of Machine Learning Research*, 2(12):97–123, 2001.

[5] B. Schölkopf, A. Smola, and K.-R. Müller. Kernel principal component analysis. In *Advances in Kernel Methods - Support Vector Learning*, pages 327–352, 1999.

[6] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge Univeristy Press, 2004.

[7] A. N. Tikhonov and V. Y. Arsenin. *Solutions of Ill-Posed Problems*. Wiley, New York, 1977.

[8] J. Weston, O. Chapelle, A. Elisseeff, B. Schölkopf, and V. Vapnik. Kernel dependency estimation. In S. T. S. Becker and K. Obermayer, editors, *Advances in Neural Information Processing Systems 15*. MIT Press.

[9] H. Wold. Soft modeling by latent variables; the nonlinear iterative partial least squares approach. *Perspectives in Probability and Statistics, Papers in Honour of M.S. Bartlett*, 1975.