Mach Learn (2006) 63: 69–101
DOI 10.1007/s10994-006-6130-8

Model-based transductive learning of the kernel matrix

Zhihua Zhang · James T. Kwok · Dit-Yan Yeung

Received: 11 March 2005 / Revised: 2 November 2005 / Accepted: 10 November 2005 /
Published online: 9 March 2006

Abstract This paper addresses the problem of transductive learning of the kernel matrix from a probabilistic perspective. We define the kernel matrix as a Wishart process prior and construct a hierarchical generative model for kernel matrix learning. Specifically, we consider the target kernel matrix as a random matrix following the Wishart distribution with a positive definite parameter matrix and a degree of freedom. This parameter matrix, in turn, has the inverted Wishart distribution (with a positive definite hyperparameter matrix) as its conjugate prior and the degree of freedom is equal to the dimensionality of the feature space induced by the target kernel. Resorting to a missing data problem, we devise an *expectation-maximization* (EM) algorithm to infer the missing data, parameter matrix and feature dimensionality in a *maximum a posteriori* (MAP) manner. Using different settings for the target kernel and hyperparameter matrices, our model can be applied to different types of learning problems. In particular, we consider its application in a semi-supervised learning setting and present two classification methods. Classification experiments are reported on some benchmark data sets with encouraging results. In addition, we also devise the EM algorithm for kernel matrix completion.

Keywords: Kernel learning · Wishart process · Bayesian inference · Transductive learning · EM algorithm · MAP estimation · Semi-supervised learning

Editor: Philip M. Long

Z. Zhang · J. T. Kwok · D.-Y. Yeung (✉)
Department of Computer Science, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong
e-mail: dyyeung@cs.ust.hk

Z. Zhang
e-mail: zhzhang@cs.ust.hk

J. T. Kwok
e-mail: jamesk@cs.ust.hk

1. Introduction

In recent years, kernel methods (Schölkopf & Smola, 2002; Vapnik, 1998) are increasingly popular in machine learning and data processing applications due to their benefits from conceptual simplicity and theoretical potentiality. Kernel machines, such as the support vector machines (SVM) (Cortes & Vapnik, 1995), kernel principal component analysis (PCA) (Schölkopf, Smola, & Müller, 1998) and kernel Fisher discriminant analysis (FDA) (Baudat & Anouar, 2000), work by mapping data nonlinearly into a high-dimensional feature space and then implementing some traditional linear algorithms in this feature space. This approach is attractive since feature vectors in the high-dimensional feature space are more likely to be linearly separable than data points in the original input space. Moreover, the so-called *kernel trick* makes the implementation of kernel methods efficient, since kernels can be used without explicit usage of the feature vectors themselves.

On the other hand, *Gaussian process* (GP), also known as "kriging" in geostatistics, has been widely used for interpolating and smoothing spatial data in spatial statistics (Cressie, 1991). In machine learning, GP is also a common Bayesian tool to assign prior distributions over functions, and has been successfully used in various nonlinear modeling tasks (Bishop, 1995) such as classification and regression. An important component in GP's is the covariance matrix. Usually, the covariance of the random field at any two index vectors is assumed to be a positive definite function of the distance between the vectors (Wahba, 1990). Thus, the covariance matrix in a GP can also be regarded as a kernel matrix, and this bridges the two techniques of GP's and kernel machines (Seeger, 2000; Smola & Schölkopf, 2002).

1.1. Related work

Because of the central role of the kernel, a poor kernel choice can lead to significantly impaired performance. Typically, the practitioner has to select the kernel before learning starts, with common choices being the polynomial kernel, Gaussian kernel, and Laplacian kernel. The associated kernel parameters, such as the order in the polynomial kernel and the width in the Gaussian or Laplacian kernel, can then be determined by the user using various heuristics. A more disciplined approach to set the parameters is by optimizing a quality functional of the kernel (Ong, Smola, & Willamson, 2003) such as some generalization error bound (Chapelle et al., 2002) or evidence (Kwok, 2000; Sollich, 2000). Instead of adapting only the kernel parameters, a recent development is to adapt also the form of the kernel itself. As in practice we are often interested in finite-sized data sets, almost all information in the kernel function can be encoded in a kernel matrix. Consequently, one could bypass the learning of the kernel function by just learning the kernel matrix instead.

Cristianini et al. (2002) introduced the notion of *alignment* to measure the similarity between two kernels or between a kernel and a target function. Based on this notion, they proposed a transductive learning method (Vapnik, 1998) for the kernel matrix by optimizing the coefficients (eigenvalues) for the spectral decomposition of the full kernel matrix on both training and test data. Kandola et al. (2002) extended this method to the inductive setting. Lanckriet et al. (2004) derived a generalization bound for choosing the kernel and formulated the kernel matrix learning problem as a convex optimization problem that is not prone to local minima. However, even with the recent advances in interior point methods, convex programming problems such as semi-definite programming (SDP) are still very computationally expensive on problems with large kernel matrices. Thus, instead of using SDP, Bousquet and Herrmann (2003) proposed a simple, efficient gradient-descent

algorithm that can be orders of magnitude faster than a typical SDP solver. Crammer et al. (2003), on the other hand, formulated this learning problem under the boosting paradigm, so that an accurate kernel is constructed from simple base kernels obtained from solving the generalized eigenvector problem. Recently, kernel matrix learning has been used to deal with the problem of missing data, giving a kernel matrix completion problem. For example, Graepel et al. (2002) considered kernel matrix completion by applying SDP. Based on information geometry (Amari, 1995), Tsuda et al. (2003) introduced the use of Kullback-Leibler (KL) divergence as a similarity measure between two positive definite matrices. They then devised an *em* algorithm for the kernel matrix completion problem.

Notice that among these methods, SDP (Vandenberghe & Boyd, 1996) and gradient descent (Bousquet & Herrmann, 2003) are algebraic, while boosting (Friedman, Hastie, & Tibshirani, 2000) can be regarded as statistical. Tsuda et al. (2003) also described an EM formulation for their *em* algorithm. However, as mentioned by the authors, this EM formulation does not in fact have any observed data nor does it have any prior distribution of missing data. Hence, this so-called EM formulation is only intended for interpreting the relationship between the equations in the E- and M-steps with those in the *e*- and *m*-steps. In summary, none of the above methods stems from a model-based perspective.

Due to the strong connection between the covariance matrix in a GP and the kernel matrix as discussed above, the problem of choosing the covariance matrix can also be regarded as a kernel matrix learning problem. Usually, the covariance matrix is first parameterized and then the associated hyperparameters are estimated using methods such as maximum likelihood estimation (MLE) (Mardia & Marshall, 1984) or Markov chain Monte Carlo (MCMC) (Diggle Tawn, & Mayeed, 1998; Neal, 1997a; Williams & Barber, 1998). These methods for learning the covariance matrix are based on the inductive setting.

1.2. Outline of our work

In this paper, we propose the notion of Wishart processes by treating a reproducing kernel as a stochastic process. Specifically, if each feature dimension follows a Gaussian process prior, then the corresponding random kernel matrix follows the Wishart distribution. Conversely, if we are given a kernel matrix following the Wishart distribution, then there exists a set of feature vectors with each feature dimension following the Gaussian process prior. Moreover, the dimensionality of the kernel-induced feature space is equal to the degree of freedom of the Wishart distribution. This provides a generative model of the kernel matrix and motivates us to view the kernel matrix learning problem from a model-based perspective. Moreover, this also reveals the intrinsic statistical mechanism of reproducing kernels with Wishart process priors, and inspires us to explore classification and regression problems using Wishart processes. We use a transductive learning setting (Joachims, 1999; Kandala, Shawe-Taylor & Cristianini, 2003; Vapnik, 1998) to achieve these goals simultaneously.

Based on the Wishart generative model of the kernel matrix, we first propose in this paper a hierarchical transductive learning framework for the kernel matrix. We consider the target kernel matrix as a random matrix distributed according to the Wishart distribution (Gupta & Nagar, 2000), whose parameter matrix in turn follows the conjugate prior of the Wishart distribution, which is the inverted Wishart distribution. As will be seen later, this prior has the effect of including a regularization term in the likelihood function. Under the *maximum a posteriori* (MAP) setting, we develop an *expectation-maximization* (EM) algorithm (Dempster, Laird, & Rubin, 1977) to infer the missing data and the model parameters for the corresponding learning problem. To our own surprise, not only the parameter matrix,

but also the dimensionality of the kernel-induced feature space as defined above, can be estimated through the proposed EM algorithm.

Since the kernel matrix is a positive semi-definite matrix, our transductive learning model based on Wishart processes has potential applications in many machine learning and pattern recognition problems. For example, we can consider an affinity matrix or similarity matrix as a kernel matrix and then learn it from data using our model. In this paper, we apply our hierarchical transductive learning model to the semi-supervised learning paradigm (Zhou et al., 2004), which has recently attracted a great deal of interest. By using different settings on the target kernel matrix, we present two semi-supervised learning methods.

The first method is derived from the equivalence outlined above, namely, the reproducing kernel follows a Wishart process and the dimensions of the feature vectors in the kernel-induced feature space are mutually independent Gaussian processes. This inspires us to define each feature dimension as a Gaussian process prior. Thus, the resultant method avoids the usage of the logistic function. Moreover, we shall see that the EM algorithm can be used to estimate the covariance matrix in a GP. In the second method, we use the discriminant kernel (Zhang, 2003) as the target kernel, and then construct a transductive discriminant analysis method for both classification and clustering problems. Our method differs from the generalized FDA in that the kernel matrix we use includes information from both the input vectors and the labels.

In addition, based on the Wishart generative model of the kernel matrix, we devise the EM algorithm for a kernel matrix completion problem (Tsuda, Akaho, & Asai, 2003), where a kernel matrix is defined over a data set with missing information. This problem can be formulated a transductive learning problem. Tsuda et al. (2003) devised an *em* algorithm for this and described its relationship with the EM formulation. Unfortunately, the derivation of the E-step in their EM algorithm is theoretically unclear because of the lack of a prior distribution on the missing part of the kernel matrix. Our work proposes a rigorous derivation of the EM algorithm, which also bears resemblance to the *em* algorithm.

1.3. Notations and organization of the paper

Throughout this paper, matrices and vectors are denoted by boldface uppercase letters and lowercase letters, respectively. Let $\mathbf{A} = [a_{ij}]$ be an $m \times n$ matrix. We denote the transpose of $\mathbf{A}$ by $\mathbf{A}'$ and $(a_{11}, \ldots, a_{m1}, a_{12}, \ldots, a_{mn})'$ by $\mathrm{vec}(\mathbf{A})$. Moreover, when $m = n$, the trace of $\mathbf{A}$ is denoted by $\mathrm{tr}(\mathbf{A})$, its determinant by $|\mathbf{A}|$, and its inverse (if exists) by $\mathbf{A}^{-1}$. In addition, we write $\mathbf{A} \succ 0$ if $\mathbf{A}$ is positive definite and $\mathbf{A} \succeq 0$ if $\mathbf{A}$ is positive semi-definite. Also, the Kronecker product of $\mathbf{A}$ and $\mathbf{B}$ is denoted by $\mathbf{A} \otimes \mathbf{B}$. To simplify our presentation, we will employ the notation of (Gupta & Nagar, 2000). Thus, for an $n \times n$ random matrix $\mathbf{W}$, $\mathbf{W} \sim \mathcal{W}_n(r, \mathbf{\Sigma})$, means that $\mathbf{W}$ follows a *Wishart distribution* with degree of freedom r and an $n \times n$ parameter matrix $\mathbf{\Sigma} \succ 0$. Finally, for an $n \times n$ random matrix $\mathbf{X}$, $\mathbf{X} \sim \mathcal{IW}_n(r, \mathbf{\Theta})$ means that $\mathbf{X}$ follows an *inverted Wishart distribution* with degree of freedom $r + n + 1$ and an $n \times n$ parameter matrix $\mathbf{\Theta} \succ 0$.

The paper is organized as follows. Section 2 presents a hierarchical Bayesian model for transductive learning of the kernel matrix and develops the EM algorithm for our model. In Section 3, we apply our transductive learning framework with the EM algorithm to the semi-supervised learning paradigm. An EM algorithm for the kernel matrix completion problem is then discussed in Section 4. Experimental results on classification applications are presented in Section 5, and the last section gives some concluding remarks. In order to facilitate

readers, brief introductions to certain topics of the matrix theory, including matrix variate distributions, matrix differentials and the Kronecker product, are given in Appendix A. Detailed derivation of the EM algorithm can be found in Appendix B.

2. Hierarchical transductive learning model of the kernel matrix

Let $\mathcal{I}$ denote a given space and $\mathcal{S} = \{\mathbf{t}_i\}_{i=1}^n \subset \mathcal{I}$ be a finite set of samples. Most existing kernel methods define the kernel function K on the Cartesian space $\mathcal{I} \times \mathcal{I}$, i.e.,

$$K : \mathcal{I} \times \mathcal{I} \to \mathbb{R}, \qquad K(\mathbf{t}_i, \mathbf{t}_j) = k_{ij} = F(t_i)' F(\mathbf{t}_j),$$

where $F : \mathcal{I} \to \mathcal{F}$ is a (usually nonlinear) mapping that relates $\mathcal{I}$ to a (possibly infinite-dimensional) feature space $\mathcal{F}$. The kernel trick allows us to compute the inner product of $F(\mathbf{t}_i)$ and $F(\mathbf{t}_j)$ in $\mathcal{F}$ without having to explicitly compute the mapping F. The kernel matrix (or Gram matrix) defined on all samples in $\mathcal{S}$ is denoted as $\mathbf{K} = [k_{ij}]_{n\times n}$. Our point of departure is to treat the feature vectors $\{F(\mathbf{t}); \mathbf{t} \in \mathcal{I}\}$ as a stochastic process. Then the kernel function $\{K(\mathbf{t}_i, \mathbf{t}_j); \mathbf{t}_i, \mathbf{t}_j \in \mathcal{I}\}$ also follows a stochastic process. First of all, we give the following definition.

Definition 1. *$\{K(\mathbf{s}, \mathbf{t}); \mathbf{s}, \mathbf{t} \in \mathcal{I}\}$ is said to be a* Wishart process *if for any $n \in \mathbb{N}$ and $\{\mathbf{t}_1, \ldots, \mathbf{t}_n\} \subseteq \mathcal{I}$, the $n \times n$ random matrix $\mathbf{K} = [K(\mathbf{t}_i, \mathbf{t}_j)]$ follows a Wishart distribution.*

Let us assume that the feature space $\mathcal{F}$ is of finite dimensionality r. For any input vector $\mathbf{t} \in \mathcal{I}$, we can express $F(\mathbf{t}) = (F_1(\mathbf{t}), \ldots, F_r(\mathbf{t}))'$ as an r-dimensional functional vector. Let us define $\mathbf{F}$ as

$$\mathbf{F} = \begin{bmatrix} F_1(\mathbf{t}_1) & F_2(\mathbf{t}_1) & \ldots & F_r(\mathbf{t}_1) \\ F_1(\mathbf{t}_2) & F_2(\mathbf{t}_2) & \ldots & F_r(\mathbf{t}_2) \\ \vdots & \vdots & \ddots & \vdots \\ F_1(\mathbf{t}_n) & F_2(\mathbf{t}_n) & \ldots & F_r(\mathbf{t}_n) \end{bmatrix}. \tag{1}$$

Then $\mathbf{K} = \mathbf{F}\mathbf{F}'$. In this paper, we formulate a probabilistic generative model of the kernel matrix $\mathbf{K}$ based on random matrix variate theory. Recall that $F_j(\mathbf{t})$ $(j = 1, \ldots, r)$ represents the jth coordinate of the feature vector $F(\mathbf{t})$ and $F_j(\mathbf{t})$ is itself a function from $\mathcal{I}$ to $\mathbb{R}$. Denote $\mathbf{f}^{(j)} = (F_j(\mathbf{t}_1), F_j(\mathbf{t}_2), \ldots, F_j(\mathbf{t}_n))'$, which contains the jth feature dimension in all n feature vectors $(j = 1, \ldots, r)$. From the dual relationship between the matrix-variate distribution and the Wishart distribution (Gupta & Nagar, 2000), we have the following theorem:

Theorem 1. *Let $\mathbf{f}^{(1)}, \mathbf{f}^{(2)}, \ldots, \mathbf{f}^{(r)}$ be r independent vectors from $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{0}$ is an n-dimensional zero vector and $\mathbf{\Sigma} \succ 0$ is $n \times n$. Then $\mathbf{K}$ is a random Wishart matrix $\mathbf{K} \sim \mathcal{W}_n(r, \mathbf{\Sigma})$. Conversely, given a kernel matrix $\mathbf{K} \sim \mathcal{W}_n(r, \mathbf{\Sigma})$, where r is an integer, then there exist r mutually independent n-dimensional vectors $\boldsymbol{f}^{(j)}$ from $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$.*

Thus, we can conclude that $\{K(\mathbf{s}, \mathbf{t}); \mathbf{s}, \mathbf{t} \in \mathcal{I}\}$ is a *Wishart process* if and only if each feature dimension follows a Gaussian process, or in other words, $\{F_j(\mathbf{t}); \mathbf{t} \in \mathcal{I}\}$ $(j = 1, \ldots, r)$ are r mutually independent *Gaussian processes*. Theorem 1 leads us to a generative model for

the kernel matrix $\mathbf{K}$. That is, we define the kernel matrix $\mathbf{K}$ as a random Wishart matrix from $\mathcal{W}_n(r,\mathbf{\Sigma})$ on which kernel learning can be performed. Furthermore, its degree of freedom r is equal to the dimensionality of the feature space induced by kernel K. This generative model provides a statistical basis for developing a Bayesian inference approach for learning the kernel matrix. Motivated by this idea, we seek to pursue this interesting direction in the current paper. Specifically, we shall present a hierarchical model for the transductive learning of the kernel matrix and then devise an EM algorithm to infer this model.

2.1. Hierarchical model

Let the training set be $\mathcal{T} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{n_1}, y_{n_1})\}$ and test set be $\tilde{\mathcal{T}} = \{(\mathbf{x}_{n_1+1}, y_{n_1+1}), \ldots, (\mathbf{x}_{n_1+n_2}, y_{n_1+n_2})\}$, where $\mathbf{x}_i \in \mathbb{R}^q$, $y_i \in \{1, 2, \ldots, c\}$ for $i = 1, \ldots, n_1$ and the y_i's are unavailable for $i = n_1 + 1, \ldots, n_1 + n_2$. Letting $n = n_2 + n_1$, we refer to $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_{n_1}, \mathbf{x}_{n_1+1}, \ldots, \mathbf{x}_n\}$ and $\mathcal{Y} = \{y_1, \ldots, y_{n_1}, y_{n_1+1}, \ldots, y_n\}$ as the input set and output set, respectively. We define a kernel matrix $\mathbf{K}$ on $(\mathcal{T} \cup \tilde{\mathcal{T}}) \times (\mathcal{T} \cup \tilde{\mathcal{T}})$ and partition it as

$$\mathbf{K} = \begin{bmatrix} \mathbf{K_{11}} & \mathbf{K_{12}} \\ \mathbf{K_{21}} & \mathbf{K_{22}} \end{bmatrix}, \tag{2}$$

where $\mathbf{K}_{11}$ and $\mathbf{K}_{22}$ are $n_1 \times n_1$ and $n_2 \times n_2$ matrices defined on the training and test sets, respectively, and $\mathbf{K}_{21} = \mathbf{K}'_{12}$ is an $n_2 \times n_1$ matrix characterizing the similarities between the training and test data.

We assume that $\mathbf{K}$ is distributed according to a Wishart distribution, i.e., $\mathbf{K} \sim \mathcal{W}_n(r, \mathbf{\Sigma}/r)$. Although it is allowed that either $n \leq r$ or $n > r$, we consider the case of $n \leq r < \infty$ in this paper. In other words, we assume $\mathbf{K} \succ \mathbf{0}$. In this case, we have

$$p(\mathbf{K} \mid \mathbf{\Sigma}, r) = \frac{r^{rn/2}}{C(n,r)} |\mathbf{\Sigma}|^{-r/2} |\mathbf{K}|^{(r-n-1)/2} \exp\left(-\frac{r}{2}\mathrm{tr}(\mathbf{\Sigma}^{-1}\mathbf{K})\right), \tag{3}$$

where $\mathbf{\Sigma} \succ 0$ is an $n \times n$ parameter matrix, which is left completely unspecified in the model, and its uncertainty is incorporated through a higher-level prior in this paper. Since the conjugate prior of a Wishart distribution is inverted Wishart, we assume that $\mathbf{\Sigma}$ is distributed according to the inverted Wishart distribution[1] $\mathcal{IW}_n(\eta r + n + 1, \eta r \mathbf{\Theta})$, where $\mathbf{\Theta}_{n\times n} \succ 0$ is called the *hyperparameter* matrix and $\eta > 0$ is a hyperparameter. From Theorem 4 in Appendix A.1, it also follows that $\mathbf{C} = \mathbf{\Sigma}^{-1}$ is distributed according to $\mathcal{W}_n(\eta r + n + 1, (\eta r \mathbf{\Theta})^{-1})$, as[2]

$$p(\mathbf{C} \mid \mathbf{\Theta}, r, \eta) = \frac{(\eta r)^{(\eta r+n+1)n/2}}{C(n, \eta r + n + 1)} |\mathbf{\Theta}|^{(\eta r+n+1)/2} |\mathbf{C}|^{\eta r/2} \exp\left(-\frac{\eta r}{2}\mathrm{tr}(\mathbf{\Theta C})\right). \tag{4}$$

Like $\mathbf{\Sigma}$, we could again define $\mathbf{\Theta}$ and η as a random matrix and a positive random variable, respectively, and then incorporate their uncertainties by some higher-level priors. However, for simplicity, $\mathbf{\Theta}$and η will be held fixed in this paper. Therefore, our probabilistic model

[1] As will be seen later, our choice of $\mathbf{K} \sim \mathcal{W}_n(r, \mathbf{\Sigma}/r)$ and $\mathbf{\Sigma} \sim \mathcal{IW}_n(\eta r + n + 1, \eta r \mathbf{\Theta})$ facilitates a simple iterative estimation procedure for the unknown parameters $\mathbf{\Sigma}$ and r.

[2] It is not too restrictive to set the degree parameter ρ to $\eta r + n + 1$. Indeed, for any $\rho > n$, we can write $\rho = \eta r + n + 1$ where $\eta = (\rho - n - 1)/r$.

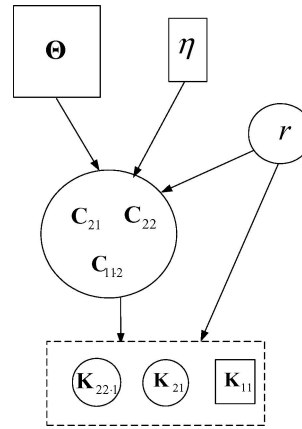

Fig. 1. A hierarchical model for transductive learning of the kernel matrix. Here, ◯ indicates unknown variables while □ indicates known variables.

is a hierarchical model with three levels. The first (lowest) level corresponds to a random Wishart matrix $\mathbf{K}$, the second level to the parameter matrix $\mathbf{C}$ of the Wishart matrix, and the third level to the hyperparameter matrix $\mathbf{\Theta}$ of the parameter matrix. Our model differs from existing kernel learning methods in that ours is based on a probabilistic generative model. Moreover, by using the hierarchical model, the hyperparameter matrix may be regarded as a regularization term to avoid the overfitting problem (Ong, Smola, & Williamson, 2003).

The observed data set provides a particular realization of $\mathbf{K}$. With an abuse of notation, we will denote this realization again by $\mathbf{K}$. Note that only the $\mathbf{K}_{11}$ part of $\mathbf{K}$ in (2) is available, while both $\mathbf{K}_{21}$ and $\mathbf{K}_{22}$ are missing. Hence, $\mathbf{K}$ represents the partially observed kernel matrix. We will formulate this as a missing data problem and then apply the EM algorithm. In other words, the incomplete (observable) data is $\mathbf{K}_{11}$, the complete data is $\{\mathbf{K}_{11}, \mathbf{K}_{21}, \mathbf{K}_{22}\}$, and the goal is to infer the missing data $\{\mathbf{K}_{21}, \mathbf{K}_{22}\}$ and the unknown model parameters $\{\mathbf{C}, r\}$.

As for $\mathbf{K}$ in (2), $\mathbf{\Sigma}$, $\mathbf{C}$ and $\mathbf{\Theta}$ are similarly partitioned as

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12} \\ \mathbf{\Sigma}_{21} & \mathbf{\Sigma}_{22} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix}, \quad \mathbf{\Theta} = \begin{bmatrix} \mathbf{\Theta}_{11} & \mathbf{\Theta}_{12} \\ \mathbf{\Theta}_{21} & \mathbf{\Theta}_{22} \end{bmatrix}. \tag{5}$$

Recall that $\mathbf{K} \succ 0$ if and only if $\mathbf{K}_{11} \succ 0$ and $\mathbf{K}_{22\cdot1} \succ 0$ (Horn & Johnson, 1985), where $\mathbf{K}_{22\cdot1} = \mathbf{K}_{22} - \mathbf{K}_{21}\mathbf{K}_{11}^{-1}\mathbf{K}_{12}$ is the Schur complement of $\mathbf{K}_{11}$. We take $\{\mathbf{K}_{11}, \mathbf{K}_{21}, \mathbf{K}_{22\cdot1}\}$ instead as the complete data to ensure that $\mathbf{K}$ is always positive definite. Moreover, we will use $\{\mathbf{C}_{11\cdot2}, \mathbf{C}_{21}, \mathbf{C}_{22}\}$, where $\mathbf{C}_{11\cdot2} = \mathbf{C}_{11} - \mathbf{C}_{12}\mathbf{C}_{22}^{-1}\mathbf{C}_{21}$, instead of $\mathbf{C}$ in our EM algorithm given in Section 2.3[3]. We will see that this representation can make the implementation of the EM algorithm simpler. Figure 1 shows a graphical model representing the hierarchical model for transductive learning of the kernel matrix.

[3] Alternatively, we often use $\{\mathbf{C}_{11\cdot2}, \mathbf{C}_{2|1}, \mathbf{C}_{22}\}$, where $\mathbf{C}_{2|1} = \mathbf{C}_{22}^{-1}\mathbf{C}_{21}$, which is based on the Bartlett decomposition $\begin{bmatrix} \mathbf{C}_{11\cdot2} + \mathbf{C}'_{2|1}\mathbf{C}_{22}\mathbf{C}_{2|1} & \mathbf{C}'_{2|1}\mathbf{C}_{22} \\ \mathbf{C}_{22}\mathbf{C}_{2|1} & \mathbf{C}_{22} \end{bmatrix}$ of $\mathbf{C}$. Moreover, we shall interchangeably employ either of $\mathbf{C}$, $\{\mathbf{C}_{11\cdot2}, \mathbf{C}_{21}, \mathbf{C}_{22}\}$ and $\{\mathbf{C}_{11\cdot2}, \mathbf{C}_{2|1}, \mathbf{C}_{22}\}$, depending upon the context.

2.2. Likelihood and inference

First of all, we give a lemma that will be useful in our later discussions.

Lemma 1. *With* $\mathbf{C} = \mathbf{\Sigma}^{-1}$ *as partitioned in (5), we have* $\mathbf{C}_{11} = \mathbf{\Sigma}_{11\cdot 2}^{-1}$, $\mathbf{C}_{11}^{-1}\mathbf{C}_{12} = -\mathbf{\Sigma}_{12}\mathbf{\Sigma}_{22}^{-1}$, $\mathbf{C}_{22} = \mathbf{\Sigma}_{22\cdot 1}^{-1}$ *and* $\mathbf{C}_{22}^{-1}\mathbf{C}_{21} = -\mathbf{\Sigma}_{21}\mathbf{\Sigma}_{11}^{-1}$.

From this lemma and Theorem 3 in Appendix A.1, we immediately have

Corollary 1. *Assume* $\mathbf{K} \sim \mathcal{W}_n(r,\ \mathbf{\Sigma}/r)$. *Then*
(i)

$$\begin{aligned} \mathbf{K}_{11} &\sim \mathcal{W}_{n_1}(r,\ (r\mathbf{C}_{11\cdot 2})^{-1}), \\ \mathbf{K}_{21} \mid \mathbf{K}_{11} &\sim \mathcal{N}(-\mathbf{C}_{2|1}\mathbf{K}_{11},\ (r\mathbf{C}_{22})^{-1} \otimes \mathbf{K}_{11}), \\ \mathbf{K}_{22\cdot 1} &\sim \mathcal{W}_{n_2}(r-n_1,\ (r\mathbf{C}_{22})^{-1}) \textit{ is independent of } \mathbf{K}_{11} \textit{ and } \mathbf{K}_{21}; \end{aligned}$$

(ii)

$$E(\mathbf{K}_{21} \mid \mathbf{K}_{11}) = -\mathbf{C}_{2|1}\mathbf{K}_{11}, \quad E(\mathbf{K}_{22\cdot 1}) = \frac{r-n_1}{r}\mathbf{C}_{22\cdot 1}^{-1}.$$

As mentioned above, $\{\mathbf{K}_{11}, \mathbf{K}_{21}, \mathbf{K}_{22\cdot 1}\}$ will be used as the complete data and hence we have to first obtain its density function from $p(\mathbf{K}) = p(\mathbf{K}_{11}, \mathbf{K}_{21}, \mathbf{K}_{22})$. This involves a standard transformation of variables: $\mathbf{K}_{22\cdot 1} = \mathbf{K}_{22} - \mathbf{K}_{21}\mathbf{K}_{11}^{-1}\mathbf{K}_{12}$, $\mathbf{B}_{21} = \mathbf{K}_{21}$ and $\mathbf{B}_{11} = \mathbf{K}_{11}$. Now, $(d\mathbf{K}) = (d\mathbf{K}_{11}) \wedge (d\mathbf{K}_{21}) \wedge (d\mathbf{K}_{22})$.[4] Since the Jacobian determinant involved is unity, we have

$$(d\mathbf{K}_{11}) \wedge (d\mathbf{K}_{21}) \wedge (d\mathbf{K}_{22}) = (d\mathbf{B}_{11}) \wedge (d\mathbf{B}_{21}) \wedge (d\mathbf{K}_{22\cdot 1}).$$

Thus, $p(\mathbf{K}) = p(\mathbf{K}_{11}, \mathbf{K}_{21}, \mathbf{K}_{22}) = p(\mathbf{K}_{11}, \mathbf{K}_{21}, \mathbf{K}_{22\cdot 1})$. This then follows from Corollary 1 that the log likelihood function $L(\mathbf{C}|\ \mathbf{K}, r)$ of the complete data is

$$\begin{aligned} L(\mathbf{C} \mid \mathbf{K}, r) &= \log p(\mathbf{K}_{22\cdot 1}) + \log p(\mathbf{K}_{11}) + \log p(\mathbf{K}_{21}|\mathbf{K}_{11}) \\ &= \frac{r-n-1}{2}\log|\mathbf{K}_{11}| + \frac{r-n-1}{2}\log|\mathbf{K}_{22\cdot 1}| + \frac{r}{2}\log|\mathbf{C}_{11\cdot 2}| + \frac{r}{2}\log|\mathbf{C}_{22}| \\ &\quad -\frac{r}{2}\mathrm{tr}(\mathbf{C}_{11\cdot 2}\mathbf{K}_{11}) - \frac{r}{2}\mathrm{tr}(\mathbf{C}_{12}\mathbf{C}_{22}^{-1}\mathbf{C}_{21}\mathbf{K}_{11}) - r\,\mathrm{tr}(\mathbf{C}_{12}\mathbf{K}_{21}) \\ &\quad -\frac{r}{2}\mathrm{tr}(\mathbf{C}_{22}\mathbf{K}_{22\cdot 1}) - \frac{r}{2}\mathrm{tr}(\mathbf{C}_{22}\mathbf{K}_{21}\mathbf{K}_{11}^{-1}\mathbf{K}_{12}) - \log C(n,r) + \frac{rn}{2}\log r. \end{aligned} \tag{6}$$

If we knew the complete matrix $\mathbf{K}$, it would be easy to determine the parameter matrix $\mathbf{C}$ by maximizing the (log) likelihood function. Similarly, if we knew the parameter matrix $\mathbf{C}$, we could determine the matrices $\mathbf{K}_{21}$ and $\mathbf{K}_{22\cdot 1}$. The problem is that we know neither.

[4] Here, we use the wedge product or exterior product. Definition 6 in Appendix A.2 gives a brief introduction.

However, by treating this as a missing data problem with complete data $\mathbf{K}$, observed data $\mathbf{K}_{11}$, and missing data $\mathbf{K}_{21}$ and $\mathbf{K}_{22\cdot1}$, we can make use of the EM algorithm (Dempster, Laird, & Rubin, 1977) to alternate estimations of $\{\mathbf{K}_{21}, \mathbf{K}_{22\cdot1}\}$ and $\{\mathbf{C}_{11\cdot2}, \mathbf{C}_{21}, \mathbf{C}_{22}, r\}$.

2.3. Learning with the EM algorithm

The EM algorithm consists of an E-step and an M-step. The E-step calculates the expectation of the complete data log-likelihood (with respect to the missing data) and the M-step maximizes this expectation with respect to the model parameters. With the availability of a prior distribution on the parameters, the EM algorithm can also be used to obtain the MAP estimate. In this Section, our EM algorithm will work in such a MAP setting. Thus, the EM algorithm computes the posterior estimates of the model parameters in two steps: Given the tth estimates, $\mathbf{C}(t)$ and $r(t)$, of $\mathbf{C}$ and r, the E-step computes

$$Q(\mathbf{C}, r \mid \mathbf{C}(t), r(t)) = E[\log p(\mathbf{K}_{11}, \mathbf{K}_{21}, \mathbf{K}_{22\cdot1} \mid \mathbf{C}, r) \mid \mathbf{K}_{11}, \mathbf{C}(t), r(t)],$$

and the M-step produces the new estimates as

$$\{\mathbf{C}(t+1), r(t+1)\} = \arg\max Q(\mathbf{C}, r \mid \mathbf{C}(t), r(t)) + \log p(\mathbf{C} \mid r).$$

Using the hierarchical model defined in Section 2.1, these two steps can be shown to be:

E-step: Given $\mathbf{K}_{11}$, $\mathbf{C}_{22}(t)$, $\mathbf{C}_{2|1}(t)$ and $r(t)$, compute

$$\begin{aligned} Q(\mathbf{C}, r \mid \mathbf{C}(t), r(t)) = {} & \frac{r}{2}\log|\mathbf{C}_{11\cdot2}| - \frac{r}{2}\mathrm{tr}(\mathbf{C}_{11\cdot2}\mathbf{K}_{11}) + r\,\mathrm{tr}\left(\mathbf{C}_{12}\mathbf{C}_{2|1}(t)\mathbf{K}_{11}\right) \\ & -\frac{r}{2}\mathrm{tr}\left(\mathbf{C}_{12}\mathbf{C}_{22}^{-1}\mathbf{C}_{21}\mathbf{K}_{11}\right) - \frac{r}{2}\mathrm{tr}\left(\mathbf{C}_{22}\mathbf{C}_{22}^{-1}(t)\right) + \frac{r}{2}\log|\mathbf{C}_{22}| \\ & -\frac{r}{2}\mathrm{tr}\left(\mathbf{C}_{22}\mathbf{C}_{2|1}(t)\mathbf{K}_{11}\mathbf{C}'_{2|1}(t)\right) \\ & +\frac{r-n-1}{2}\left(n_2\log\frac{2}{r(t)} + \sum_{j=0}^{n_2-1}\Psi\left(\frac{r(t)-n_1-j}{2}\right) - \log|\mathbf{C}_{22}(t)|\right) \\ & +\frac{rn}{2}\log r - \log C(n,r) + \frac{r-n-1}{2}\log|\mathbf{K}_{11}|. \end{aligned} \tag{7}$$

Here $\mathbf{C}_{2|1}(t) = \mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)$ and $\Psi(z) = \Gamma'(z)/\Gamma(z)$ is the *digamma function*.

M-step: Calculate

$$\mathbf{B} = (\mathbf{K}_{11} + \eta\boldsymbol{\Theta}_{11})^{-1}$$

and

$$\mathbf{C}_{11\cdot2} = (1+\eta)\mathbf{B}, \tag{8}$$

and perform the following two sub-steps:

(i) Given $\mathbf{C}_{22}(t)$, $\mathbf{C}_{2|1}(t)$ and $r(t)$, compute

$$\mathbf{C}_{2|1}(t+1) = (\mathbf{C}_{2|1}(t)\mathbf{K}_{11} - \eta\mathbf{\Theta}_{21})\mathbf{B}, \tag{9}$$

$$\begin{aligned}\mathbf{C}_{22}^{-1}(t+1) = \frac{1}{1+\eta}\big(&\mathbf{C}_{22}^{-1}(t) + \eta\mathbf{\Theta}_{22} + \mathbf{C}_{2|1}(t)\mathbf{K}_{11}\mathbf{C}_{2|1}'(t) \\ &- \mathbf{C}_{2|1}(t+1)(\mathbf{K}_{11} + \eta\mathbf{\Theta}_{11})\mathbf{C}_{2|1}'(t+1)\big).\end{aligned}$$

(ii) Given $r(t)$ and $\mathbf{C}(t+1)$, the $(t+1)$th estimate of r can be obtained by solving

$$Q_1(\mathbf{C}(t+1)|\mathbf{C}(t)) + \frac{dQ_2(r|\mathbf{C}(t), r(t))}{dr} = 0, \tag{10}$$

where

$$\begin{aligned}Q_1(\mathbf{C} \mid \mathbf{C}(t)) = &(1+\eta)\log|\mathbf{C}_{11\cdot 2}| - \mathrm{tr}\big(\mathbf{C}_{11\cdot 2}(\mathbf{K}_{11} + \eta\mathbf{\Theta}_{11})\big) \\ &+ 2\mathrm{tr}\left(\mathbf{C}_{12}\big(\mathbf{C}_{2|1}(t)\mathbf{K}_{11} - \eta\mathbf{\Theta}_{21}\big)\right) \\ &- \mathrm{tr}\big(\mathbf{C}_{12}\mathbf{C}_{22}^{-1}\mathbf{C}_{21}(\mathbf{K}_{11} + \eta\mathbf{\Theta}_{11})\big) - \mathrm{tr}\left(\mathbf{C}_{22}\big(\mathbf{C}_{22}^{-1}(t) + \eta\mathbf{\Theta}_{22}\big)\right) \\ &+ (1+\eta)\log|\mathbf{C}_{22}| - \mathrm{tr}\big(\mathbf{C}_{22}\mathbf{C}_{2|1}(t)\mathbf{K}_{11}\mathbf{C}_{2|1}'(t)\big),\end{aligned} \tag{11}$$

$$\begin{aligned}&Q_2(r \mid \mathbf{C}(t), r(t)) \\ &= (r-n-1)\left(n_2\log\frac{2}{r(t)} + \sum_{j=0}^{n_2-1}\Psi\left(\frac{r(t)-n_1-j}{2}\right) - \log|\mathbf{C}_{22}(t)|\right) \\ &+ rn\log r - 2\log C(n,r) + (r-n-1)\log|\mathbf{K}_{11}| \\ &+ (\eta r+n+1)\,n\log(\eta r) - 2\log C(n, \eta r+n+1) + (\eta r+n+1)\log|\mathbf{\Theta}|.\end{aligned} \tag{12}$$

In Appendix B, we show the detailed derivation of the above E-step and the first part of the M-step. From (8), we can see that $\mathbf{C}_{11\cdot 2}$ depends only on $\mathbf{K}_{11}$ and $\mathbf{\Theta}_{11}$, and so it can be computed *a priori* because $\mathbf{K}_{11}$ and $\mathbf{\Theta}_{11}$ are known. Moreover, instead of $\mathbf{C}_{21}$, we estimate $\mathbf{C}_{22}^{-1}\mathbf{C}_{21}$ directly. This makes the M-step more efficient. On the other hand, it is easy to see that $\mathbf{C}_{11\cdot 2} \succ 0$ and it is proved in Appendix B.3 that $\mathbf{C}_{22}(t+1)$ is also positive definite. Thus, $\mathbf{C}(t+1)$ is positive definite. For the second part of the M-step, we have the following theorem, whose proof is given in Appendix B.4.

Theorem 2. *Assume that $Q_1(\mathbf{C} \mid \mathbf{C}(t))$ and $Q_2(r \mid \mathbf{C}(t), r(t))$ are as defined in (11) and (12), respectively. Given $\mathbf{C}(t+1)$ in (9), the solution of (10) exists and is unique.*

Since it is based on the standard EM algorithm, it inherits its convergence property directly from (Dempster, Laird, & Rubin, 1977). It is worthy to note that the update of $\mathbf{C}$ is independent of r. In many cases, such as those in Section 3, it is not necessary to obtain an estimate of r, and so the update of r in M-step(ii) need not be performed. Since

M-step(i) involves only $\mathbf{C}_{22}^{-1}$ but not $\mathbf{C}_{22}$, the computational cost can be significantly reduced by avoiding matrix inversion at each iteration if the update is performed using $\mathbf{C}_{22}^{-1}$ directly (instead of $\mathbf{C}_{22}$).

After the algorithm has converged, then, depending upon the problem at hand, we can immediately compute

$$\mathbf{K}_{22\cdot 1} = \frac{r - n_1}{r}\mathbf{C}_{22}^{-1}, \quad \mathbf{K}_{21} = -\mathbf{C}_{2|1}\mathbf{K}_{11}, \quad \mathbf{K}_{22} = \mathbf{K}_{22\cdot 1} + \mathbf{C}_{2|1}\mathbf{K}_{11}\mathbf{C}'_{2|1} \tag{13}$$

using Corollary 1(ii), and from Lemma 1,

$$\mathbf{\Sigma}_{11} = \frac{\mathbf{K}_{11} + \eta\mathbf{\Theta}_{11}}{1+\eta}, \quad \mathbf{\Sigma}_{21} = -\mathbf{C}_{2|1}\mathbf{\Sigma}_{11}, \quad \mathbf{\Sigma}_{22} = \mathbf{C}_{22}^{-1} + \mathbf{C}_{2|1}\mathbf{\Sigma}_{11}\mathbf{C}'_{2|1}. \tag{14}$$

3. Applications in semi-supervised learning

Given $\mathbf{K}_{11}$ and $\mathbf{\Theta}$, we can now estimate the missing parts $\mathbf{K}_{21}$ and $\mathbf{K}_{22}$ from the EM algorithm. Transductive learning seeks to transfer the intrinsic attributes of $\mathbf{K}_{11}$ to $\mathbf{K}_{21}$ and $\mathbf{K}_{22}$ via the parameter kernel $\mathbf{\Sigma}$ with hyperparameter kernel $\mathbf{\Theta}$. The principal clue of transductive learning is the consistency assumption (Zhou et al., 2004), namely that a classification function should be sufficiently smooth with respect to the structure revealed by the training and test data.

In general, definitions of both the incomplete kernel matrix $\mathbf{K}_{11}$ and the hyperparameter matrix $\mathbf{\Theta}$ depend on the problem being considered and the prior knowledge available. While the kernel matrix learning framework presented above is not limited to the classification problem, the focus of this paper is the application of our model-based transductive learning framework to the semi-supervised learning problem by incorporating unlabeled data into labeled data for training the classifier. In particular, we will use $\mathbf{K}_{11}$ to capture class label information from the training data, so that after learning, we can obtain the kernel matrix $\mathbf{K}_{22}$, which then contains class label information on the test set, and $\mathbf{K}_{21}$, which measures the similarity between class labels on the training and test data. By using different settings on the kernel matrix $\mathbf{K}_{11}$, we present two methods for semi-supervised learning. In the first method, $\mathbf{K}$ is defined as a kernel matrix over the output data set $\mathcal{Y}$, while in the second method, $\mathbf{K}$ is defined as a kernel matrix over the joint set $(\mathcal{X} \times \mathcal{Y})$ of the input data set $\mathcal{X}$ and the output data set $\mathcal{Y}$. In both methods, the hyperparameter matrix $\mathbf{\Theta}$ is defined as a kernel matrix over the input data set $\mathcal{X}$. We can select any kernel defined over $\mathcal{X}$ for $\mathbf{\Theta}$. In particular, we use a Gaussian kernel for $\mathbf{\Theta}$ in our experiments. Now we can apply $\mathbf{K}_{11}$ and $\mathbf{\Theta}$ to our kernel learning framework, giving the estimates of $\mathbf{K}_{22}$ and $\mathbf{K}_{21}$, which can be used for classification.

3.1. A classifier using Wishart processes

Here, we follow the same notations in Section 2, and each input vector is assumed to belong to only one class. The first classification method is inspired by Theorem 1. Assume that the target kernel matrix $\mathbf{K}$ is defined on the output set $\mathcal{Y} = \{y_1, \ldots, y_{n_1}, y_{n_1+1}, \ldots, y_n\}$ where $y_i = j \in \{1, \cdots, c\}$ if the ith input vector belongs to the jth class. If $\mathbf{K} \sim \mathcal{W}_n(r, \mathbf{\Sigma}/r)$, then, according to Theorem 1, there exists a functional vector $F : \mathcal{Y} \to \mathbb{R}^r$. Our point of departure is to directly present an explicit form of the function $F(y) = (F_1(y), F_2(y), \ldots, F_r(y))'$, where $r = n + 1$. It is obvious that $c \leq r$ since $c \leq n$. First, we define r auxiliary functions

as,

$$\psi_j(y) = \begin{cases} \alpha & j = y, \\ \gamma & j \neq y \text{ and } j \leq c, \\ 0 & \text{otherwise} \end{cases} \qquad j = 1, \ldots, r$$

for $y \in \mathcal{Y}$, where, $\alpha, \gamma \in (0, 1)$ and $\alpha \gg \gamma$ are constants pre-specified by the user. For the experiments in Section 5, we will use $\alpha = 0.98$ and $\gamma = 0.01$. Another effective choice for ψ is

$$\psi_j(y) = \begin{cases} \frac{c-1}{c} & j = y, \\ -\frac{1}{c} & j \neq y \text{ and } j \leq c, \\ 0 & \text{otherwise.} \end{cases} \qquad j = 1, \ldots, r$$

Letting $\bar{\psi}(y) = \frac{1}{r} \sum_{j=1}^{r} \psi_j(y)$, we thus define

$$F_j(y) = \psi_j(y) - \bar{\psi}(y), \quad j = 1, \ldots, r.$$

For clarity, we again write out $\mathbf{F}$ defined in (1) as

$$\mathbf{F} = \begin{bmatrix} F_1(y_1) & F_2(y_1) & \ldots & F_r(y_1) \\ \vdots & \vdots & \ddots & \vdots \\ F_1(y_{n_1}) & F_2(y_{n_1}) & \ldots & F_r(y_{n_1}) \\ F_1(y_{(n_1+1)}) & F_2(y_{(n_1+1)}) & \ldots & F_r(y_{(n_1+1)}) \\ \vdots & \vdots & \ddots & \vdots \\ F_1(y_n) & F_2(y_n) & \ldots & F_r(y_n) \end{bmatrix}.$$

Furthermore, for simplicity of notation, we denote $\mathbf{f}_i = (f_{i1}, \ldots, f_{ir})$ as its ith row vector, and also $\mathbf{f}^{(j)} = (f_{1j}, \ldots, f_{n_1 j}, f_{(n_1+1)j}, \ldots, f_{nj})'$ as its jth column vector, where $f_{ij} = F_j(y_i)$. Let $\mathbf{a}^{(j)} = (f_{1j}, \ldots, f_{n_1 j})'$ and $\mathbf{b}^{(j)} = (f_{(n_1+1)j}, \ldots, f_{nj})'$ $(j = 1, \ldots, r)$. Then $\mathbf{f}^{(j)} = ((\mathbf{a}^{(j)})', (\mathbf{b}^{(j)})')'$ $(j = 1, \ldots, r)$. Clearly, for $i = 1, \ldots, n_1$, the $\mathbf{f}_i$'s are available, while for $i = n_1 + 1, \ldots, n$, the $\mathbf{f}_i$'s are missing because the corresponding labels y_i's are unknown. This gives a partially observed realization of $\mathbf{f}^{(j)}$ for $j = 1, \ldots, r$, i.e., $\mathbf{a}^{(j)}$ is available while $\mathbf{b}^{(j)}$ is missing. Moreover, we are given a realization of $\mathbf{K}_{11}$ on the output part of the training set,

$$\mathbf{K}_{11} = \begin{pmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_{n_1} \end{pmatrix} \left(\mathbf{f}'_1, \ldots, \mathbf{f}'_{n_1} \right) + \varepsilon \mathbf{I}_{n_1}, \tag{15}$$

where $\mathbf{I}_{n_1}$ is the $n_1 \times n_1$ identity matrix and ε is a small amount of jitter (e.g., $\varepsilon = 0.0001$) to prevent $\mathbf{K}_{11}$ from becoming singular.

According to Theorem 1, we have $\mathbf{f}^{(j)} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ $(j = 1, \ldots, r)$. This results in $\mathbf{b}^{(j)} \sim \mathcal{N}(\mathbf{\Sigma}_{21}\mathbf{\Sigma}_{11}^{-1}\mathbf{a}^{(j)}, \mathbf{\Sigma}_{22\cdot 1})$, conditioned on $\mathbf{a}^{(j)}$. Recall that $\mathbf{\Sigma}_{21}\mathbf{\Sigma}_{11}^{-1} = -\mathbf{C}_{22}^{-1}\mathbf{C}_{21}$ and

1. Compute the $n \times n$ hyperparameter kernel matrix $\mathbf{\Theta} = \left[\exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\beta})\right]$ and the $n_1 \times n_1$ target kernel $\mathbf{K}_{11}$ according to (15).
2. Run the kernel matrix learning algorithm in (9) to directly obtain $\mathbf{C}_{22}^{-1}\mathbf{C}_{21}$ and $\mathbf{C}_{22}^{-1}$.
3. Compute $\mathbf{b}^{(j)} = -\mathbf{C}_{22}^{-1}\mathbf{C}_{21}\mathbf{a}^{(j)}$, for $j = 1, \ldots, c$.
4. Label the unlabeled point $\mathbf{x}_i$ by $y_i = \arg\max_j \{f_{ij}\}_{j=1}^{c}$, for $i = n_1 + 1, \ldots, n$.

Fig. 2. A classifier using Wishart processes.

$\mathbf{C}_{22}^{-1} = \mathbf{\Sigma}_{22\cdot 1}$, then $\mathbf{b}^{(j)} \mid \mathbf{a}^{(j)} \sim \mathcal{N}(-\mathbf{C}_{22}^{-1}\mathbf{C}_{21}\mathbf{a}^{(j)}, \mathbf{C}_{22}^{-1})$. This leads us to a classification method, which is summarized in Fig. 2. Here, since $r = n + 1$ is pre-specified, there is no need to use EM to estimate r. Thus, the current M-step reduces to M-step(i) given in Section 2.3.

3.2. Kernel transductive discriminant analysis

The second semi-supervised learning method is motivated by a distance-based classifier using the discriminant kernel (Zhang, 2003). We use the Gaussian kernel matrix on $\mathcal{X} \times \mathcal{X}$ as the $n \times n$ hyperparameter matrix $\mathbf{\Theta}$ and the discriminant kernel on $\mathcal{I} \times \mathcal{I}$ to define the $n_1 \times n_1$ target kernel matrix $\mathbf{K}_{11}$. In other words, its (k, l)th element, $K((\mathbf{x}_k, y_k), (\mathbf{x}_l, y_l))$, is defined as

$$K((\mathbf{x}_k, y_k), (\mathbf{x}_l, y_l)) = \begin{cases} \frac{1}{2}\exp\left(-\frac{\|\mathbf{x}_k - \mathbf{x}_l\|^2}{\beta}\right) + \frac{1}{2} & y_k = y_l \\ \frac{1}{2}\exp\left(-\frac{\|\mathbf{x}_k - \mathbf{x}_l\|^2}{\beta}\right) & y_k \neq y_l \end{cases}, \quad k, l = 1, \ldots, n_1. \tag{16}$$

Since the discriminant kernel guarantees that all between-class distances must be larger than all within-class distances, this makes it desirable for distance-based classification or clustering methods. The discriminant kernel $\mathbf{K}$ employs information from both the input vector $\mathbf{x}$ and its associated label y. So the nonlinear mapping, which induces $\mathbf{K}$, should also be a joint function of $\mathbf{x}$ and y, and we will denote it by $F(\mathbf{x}, y)$. After obtaining the complete kernel $\mathbf{K}$, we use a distance-based classification method that utilizes the property of the discriminant kernel. Assume that N_j points in the training set belong to the jth class $\mathcal{C}_j$, and the class mean of $\mathcal{C}_j$ (in the feature space) is $\mathbf{m}_j = \frac{1}{N_j}\sum_{\mathbf{x}_i \in \mathcal{C}_j} F(\mathbf{x}_i, y_i)$. We then assign point $\mathbf{x}$ to $\mathcal{C}_i$ if $\|F(\mathbf{x}, y) - \mathbf{m}_i\|^2 \leq \|F(\mathbf{x}, y) - \mathbf{m}_j\|^2$ for all $j \neq i$, where

$$\begin{aligned} \|F(\mathbf{x}, y) - \mathbf{m}_i\|^2 &= F(\mathbf{x}, y)'F(\mathbf{x}, y) + \mathbf{m}_i'\mathbf{m}_i - 2F(\mathbf{x}, y)'\mathbf{m}_i \\ &= 1 + \frac{1}{N_i^2}\sum_{\mathbf{x}_j, \mathbf{x}_l \in \mathcal{C}_i} K((\mathbf{x}_j, y_j), (\mathbf{x}_l, y_l)) - \frac{2}{N_i}\sum_{\mathbf{x}_j \in \mathcal{C}_i} K((\mathbf{x}, y), (\mathbf{x}_j, y_j)) \end{aligned} \tag{17}$$

Here, $K(\cdot, \cdot)$ is the corresponding element of the target kernel $\mathbf{K}$. As can be seen from (17), this classification method works with $\mathbf{K}_{11}$ and $\mathbf{K}_{21}$, and from (13), we have $\mathbf{K}_{21} =$

1. Compute the $n \times n$ hyperparameter matrix $\mathbf{\Theta} = \left[\exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\beta})\right]$ and the $n_1 \times n_1$ target kernel $\mathbf{K}_{11}$ according to (16).
2. Run the learning algorithm in (9). After obtaining $\mathbf{C}_{2|1}$ and $\mathbf{C}_{22}^{-1}$, calculate $\mathbf{K}_{21} = -\mathbf{C}_{22}^{-1}\mathbf{C}_{21}\mathbf{K}_{11}$.
3. For every $n_1 + 1 \leq k \leq n$, compute $\|F(\mathbf{x}_k, y_k) - \mathbf{m}_i\|^2$ from (17), assign the test point $\mathbf{x}_k$ to $\mathcal{C}_i$ if $\|F(\mathbf{x}_k, y_k) - \mathbf{m}_i\|^2 \leq \|F(\mathbf{x}_k, y_k) - \mathbf{m}_j\|^2$ for all $j \neq i$.

Fig. 3. Kernel transductive discriminant analysis.

$-\mathbf{C}_{2|1}\mathbf{K}_{11}$. Therefore, the classification method is independent of r. In other words, we can drop M-step(ii) for updating r. The proposed procedure is summarized in Fig. 3. Clearly, this classification method is a nearest mean classifier with the target kernel. We also note that our classifier is similar to kernel FDA. Both are motivated by the Fisher discriminant criterion, and seek to obtain discriminant feature vectors such that between-class distances are larger than within-class distances. However, the procedures for achieving this goal are different. By employing joint information from both the input and output spaces, we first define an inner product over the training set such that the distance induced by the inner product satisfies the Fisher criterion, and then seek to transfer this distance measure to the test set through transductive learning. Kernel FDA, on the other hand, tries to find maximally separable feature vectors by optimizing the Fisher discriminant criterion using spectral decomposition.

4. EM algorithm for kernel matrix completion

In practical applications, it is possible that the observed data are available only for a subset of samples. Thus, when we work with a kernel matrix derived from such data, we are required to first complete the missing entries in this kernel matrix (Graepel, 2002; Tsuda, Akaho, & Asai, 2003; Kin et al., 1954; Smola, Vishwanathan, & Hoffman, 2004). Specifically, given an incomplete kernel matrix $\mathbf{K}$, we partition it as $\mathbf{K} = \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix}$ where $\mathbf{K}_{11}$ is available, and $\mathbf{K}_{12}$ $(= \mathbf{K}'_{21})$ and $\mathbf{K}_{22}$ are missing. Then, our goal is to complete $\mathbf{K}_{12}$ and $\mathbf{K}_{22}$. This so-called kernel matrix completion problem can be regarded as a special case of kernel matrix learning and can be included under the transductive learning framework of the kernel matrix. A common approach to restoring $\mathbf{K}_{12}$ and $\mathbf{K}_{22}$ is through use of an *auxiliary* kernel matrix.

Recently, Tsuda et al. (2003) devised an *em* algorithm for this problem. Moreover, they also described an EM formulation, where the E- and M-steps are equivalent to the *e*- and *m*-steps, respectively. However, the model in (Tsuda, Akaho, & Asai, 2003) does not have any observed data nor does it use any prior distribution of the missing data $\{\mathbf{K}_{12}, \mathbf{K}_{22}\}$. It is necessary to assign a prior for the missing data to compute the expectation of the missing data in the E-step. Thus, it is not really clear how to perform this EM algorithm (Tsuda, Akaho, & Asai, 2003). In this Section, by assigning a Wishart process prior to the kernel matrix $\mathbf{K}$, we demonstrate a rigorous derivation of the EM algorithm. First, if we let the auxiliary matrix to be the parameter matrix $\mathbf{\Sigma}$ of our model in Fig. 1, which is associated with the hyperparameter

matrix $\mathbf{\Theta}$, then our model and the EM algorithm devised in Section 2.3 can be easily used for this problem.

Now along the line in (Tsuda, Akaho, & Asai, 2003), the auxiliary matrix $\mathbf{\Sigma}$ is defined as $\mathbf{\Sigma} = \sum_{i=1}^{n} \lambda_i \boldsymbol{\mu}_i \boldsymbol{\mu}_i'$ with $\lambda_i > 0$. Denote $\mathbf{U} = [\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_n]'$ and $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$. Here, $\mathbf{U}$ is assumed to be known, while $\mathbf{\Lambda}$ is unknown and has to be estimated. Usually, $\boldsymbol{\mu}_i \boldsymbol{\mu}_i'$'s are also called the base kernel matrices (Cristianini et al., 2002; Lanckriet et at., 2004; Crammer, Keshet, & Singer, 2003). Thus, we seek to use a weighted combination of these fixed base matrices to approximate $\mathbf{K}$. Consequently, the problem is to estimate the weighting coefficients λ_i's and the missing data $\{\mathbf{K}_{12}, \mathbf{K}_{22}\}$. As in Section 2.1, we assume that the kernel matrix $\mathbf{K}$ is distributed according to a Wishart distribution $\mathcal{W}_n(r, \mathbf{\Sigma}/r)$. Similar to Section 2.1, we formulate it as a missing data problem where $\{\mathbf{K}_{11}, \mathbf{K}_{21}, \mathbf{K}_{22\cdot1}\}$ represents the complete data, $\mathbf{K}_{11}$ represents the (incomplete) observed data, and $\{\mathbf{\Lambda}, r\}$ represents the unknown parameters. Denote $\mathbf{C} = \mathbf{\Sigma}^{-1} = \sum_{i=1}^{n} \lambda_i^{-1} \boldsymbol{\mu}_i \boldsymbol{\mu}_i'$ and partition $\boldsymbol{\mu}_i$ as $\boldsymbol{\mu}_i' = (\mathbf{a}_i', \mathbf{b}_i')$, where $\mathbf{a}_i$ and $\mathbf{b}_i$ are n_1- and n_2-dimensional vectors, respectively. Then $\mathbf{C}_{11} = \sum_{i=1}^{n} \lambda_i^{-1} \mathbf{a}_i \mathbf{a}_i'$, $\mathbf{C}_{22} = \sum_{i=1}^{n} \lambda_i^{-1} \mathbf{b}_i \mathbf{b}_i'$, and $\mathbf{C}_{21} = \sum_{i=1}^{n} \lambda_i^{-1} \mathbf{b}_i \mathbf{a}_i'$. The log-likelihood function $L(\mathbf{\Lambda}, r \mid \mathbf{K})$ can be expressed as

$$
\begin{aligned}
L(\mathbf{\Lambda}, r \mid \mathbf{K}) &= \frac{rn}{2} \ln r - \ln C(n, r) + \frac{r}{2} \sum_{i=1}^{n} \ln \lambda_i^{-1} + \frac{r-n-1}{2} \ln |\mathbf{K}| - \frac{r}{2} \sum_{i=1}^{n} \lambda_i^{-1} \boldsymbol{\mu}_i' \mathbf{K} \boldsymbol{\mu}_i \\
&= \frac{rn}{2} \ln r - \ln C(n, r) + \frac{r}{2} \sum_{i=1}^{n} \ln \lambda_i^{-1} + \frac{r-n-1}{2} \Big(\ln |\mathbf{K}_{11}| + \ln |\mathbf{K}_{22\cdot1}| \Big) \\
&\quad - \frac{r}{2} \sum_{i=1}^{n} \lambda_i^{-1} \Big(\mathbf{a}_i' \, \mathbf{K}_{11} \, \mathbf{a}_i + 2 \mathbf{b}_i' \, \mathbf{K}_{21} \, \mathbf{a}_i + \mathbf{b}_i' \, \mathbf{K}_{22\cdot1} \, \mathbf{b}_i + \mathbf{b}_i' \, \mathbf{K}_{21} \, \mathbf{K}_{11}^{-1} \, \mathbf{K}_{21}' \, \mathbf{b}_i \Big). \quad (18)
\end{aligned}
$$

From (18) and the relation (Lutkepohl, 1996)

$$
\mathbf{b}_i' \mathbf{K}_{21} \mathbf{K}_{11}^{-1} \mathbf{K}_{21}' \mathbf{b}_i = (\mathrm{vec}(\mathbf{K}_{21}'))' (\mathbf{b}_i \mathbf{b}_i' \otimes \mathbf{K}_{11}^{-1}) \mathrm{vec}(\mathbf{K}_{21}'),
$$

we have $\{\ln |\mathbf{K}_{22\cdot1}|, \mathbf{K}_{22\cdot1}, \mathbf{K}_{21}, \mathrm{vec}(\mathbf{K}_{21}')(\mathrm{vec}(\mathbf{K}_{21}'))'\}$ as complete-data sufficient statistic for $\{\mathbf{\Lambda}, r\}$. Given the tth estimates, $\mathbf{\Lambda}(t)$ and $r(t)$, of $\mathbf{\Lambda}$ and r, by using the properties of Wishart distributions and matrix variate normal distributions, we obtain

$$
\begin{aligned}
&E\left(\ln |\mathbf{K}_{22\cdot1}| \middle| \mathbf{K}_{11}, \mathbf{\Lambda}(t), r(t) \right) = n_2 \ln \frac{2}{r(t)} + \sum_{j=0}^{n_2-1} \Psi\left(\frac{r(t) - n_1 - j}{2} \right) - \ln |\mathbf{C}_{22}(t)|, \\
&E\left(\mathbf{K}_{22\cdot1} \middle| \mathbf{K}_{11}, \mathbf{\Lambda}(t), r(t) \right) = \frac{r(t) - n_1}{r(t)} \mathbf{C}_{22}^{-1}(t), \\
&E\left(\mathbf{K}_{21} \middle| \mathbf{K}_{11}, \mathbf{\Lambda}(t), r(t) \right) = -\mathbf{C}_{22}^{-1}(t) \mathbf{C}_{21}(t) \mathbf{K}_{11}, \\
&E\left(\mathrm{vec}(\mathbf{K}_{21}')(\mathrm{vec}(\mathbf{K}_{21}'))' \middle| \mathbf{K}_{11}, \mathbf{\Lambda}(t), r(t) \right) \\
&= \left(\mathbf{C}_{22}^{-1}(t) \otimes \mathbf{K}_{11} \right) \mathrm{vec}(\mathbf{C}_{12}(t)) \left(\mathrm{vec}(\mathbf{C}_{12}(t)) \right)' \left(\mathbf{C}_{22}^{-1}(t) \otimes \mathbf{K}_{11} \right) + \frac{1}{r(t)} \mathbf{C}_{22}^{-1}(t) \otimes \mathbf{K}_{11}.
\end{aligned}
$$

Thus, for the E-step, we obtain the expectation of $L(\mathbf{\Lambda}, r|\mathbf{K})$ w.r.t. $p(\mathbf{K}_{22\cdot1}, \mathbf{K}_{21}|\mathbf{K}_{11}, \mathbf{\Lambda}(t), r(t))$ as

$$Q(\mathbf{\Lambda}, r|\mathbf{\Lambda}(t), r(t)) = \frac{rn}{2}\ln r - \ln C(n,r) + \frac{r}{2}\sum_{i=1}^{n}\left(\ln\lambda_i^{-1} - \lambda_i^{-1}\boldsymbol{\mu}_i'\mathbf{D}(t)\boldsymbol{\mu}_i\right) + \frac{r-n-1}{2}\left[\ln\frac{|\mathbf{K}_{11}|}{|\mathbf{C}_{22}(t)|} + n_2\ln\frac{2}{r(t)} + \sum_{j=0}^{n_2-1}\Psi\left(\frac{r(t)-n_1-j}{2}\right)\right].$$

Here

$$\mathbf{D}(t) = \begin{bmatrix}\mathbf{D}_{11}(t) & \mathbf{D}_{21}'(t)\\ \mathbf{D}_{21}(t) & \mathbf{D}_{22}(t)\end{bmatrix},$$

where $\mathbf{D}_{11}(t) = \mathbf{K}_{11}$ and

$$\mathbf{D}_{22\cdot1}(t) = \mathbf{C}_{22}^{-1}(t), \quad \mathbf{D}_{21}(t) = -\mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)\mathbf{K}_{11}. \tag{19}$$

The M-step consists of two parts:

(i) To compute the $(t+1)$th estimate of λ_i as

$$\lambda_i(t+1) = \boldsymbol{\mu}_i'\,\mathbf{D}(t)\,\boldsymbol{\mu}_i. \tag{20}$$

(ii) To compute the $(t+1)$th estimate of r by solving the following equation

$$n\ln\frac{r}{2} - \sum_{j=0}^{n-1}\Psi\left(\frac{r-j}{2}\right) = \sum_{i=1}^{n}\ln\boldsymbol{\mu}_i'\,\mathbf{D}(t)\,\boldsymbol{\mu}_i - \ln|\mathbf{D}(t)| + n_2\ln\frac{r(t)}{2} - \sum_{j=0}^{n_2-1}\Psi\left(\frac{r(t)-n_1-j}{2}\right). \tag{21}$$

Since $l\ln\frac{z}{2} - \sum_{j=0}^{l-1}\Psi(\frac{z-j}{2})$ is a positive monotonic decreasing function of z for $z \geq l$ (Chen, 1979), $n_2\ln\frac{r(t)}{2} - \sum_{j=0}^{n_2-1}\Psi(\frac{r(t)-n_1-j}{2})$ is always positive since $n_2\ln\frac{r(t)}{2} \geq n_2\ln\frac{r(t)-n_1}{2}$. Furthermore, applying Hadamard's inequality (Lütkepohl, 1996) to the positive definite matrix $\mathbf{UD}(t)\mathbf{U}'$ and considering that $\boldsymbol{\mu}_i$'s are mutually orthonormal, we have

$$|\mathbf{D}(t)| = |\mathbf{UD}(t)\mathbf{U}'| \leq \prod_{i=1}^{n}\boldsymbol{\mu}_i'\,\mathbf{D}(t)\,\boldsymbol{\mu}_i.$$

Namely, $\sum_{i=1}^{n}\ln\boldsymbol{\mu}_i'\,\mathbf{D}(t)\,\boldsymbol{\mu}_i - \ln|\mathbf{D}(t)|$ is also nonnegative. Hence the right-hand side of (21) is always positive. As a result, the solution of (21) is unique and may be obtained numerically through solving the equation. Essentially, the EM algorithm alternately works with (19), (20) and (21). Obviously, (19) and (20) correspond to the *e*- and *m*-steps, respectively, in the *em* algorithm of (Tsuda, Akaho, & Asai, 2003).

Table 1. Test set accuracies (in %) obtained from the classification experiments (60% for training and 40% for testing). The highest accuracies are shown in boldface.

Method	Breast cancer	Ionosphere	Sonar	Wine
GWPC	95.73 (±0.96)	92.44 (±1.92)	**87.60** (±3.85)	96.59 (±1.75)
KTDA	96.00 (±0.95)	**94.58** (±1.50)	87.40 (±3.61)	**98.04** (±1.41)
KFDA	**96.58** (±0.97)	92.34 (±1.99)	83.24 (±3.71)	96.04 (±2.55)
SVM	96.04 (±1.15)	92.06 (±2.08)	83.33 (±3.80)	96.92 (±1.95)
KNM	90.89 (±1.54)	68.58 (±4.55)	77.37 (±5.86)	94.39 (±3.10)
1-NN	95.14 (±1.00)	85.82 (±2.07)	84.18 (±3.77)	95.00 (±2.52)

5. Experiments

In this Section, we present some experiments to illustrate the two classification methods devised in Section 3. For the sake of easy reference, we refer to the classification methods in Sections 3.1 and 3.2 as GWPC and KTDA, respectively. In all our experiments, the initialization of $\mathbf{C}$ is $\mathbf{C}(0) = 0.8\boldsymbol{\Theta}^{-1}$ and the maximum number of iterations is set to 100. Once the maximum number of iterations is reached or the difference between the log likelihoods of two successive iterations is smaller than a threshold value of 0.00001, the EM algorithm will stop.

5.1. Results on UCI benchmark data sets

First, experiments are performed on four benchmark data sets (`Wisconsin breast cancer, ionosphere, sonar,` and `wine`) from the UCI Machine Learning Repository. In our experiments, we compare GWPC and KTDA with kernel FDA (KFDA), SVM, kernel nearest mean classifier (KNM) and 1-NN (i.e., k-NN with $k = 1$). The hyperparameter kernel $\boldsymbol{\Theta}$ is based on the Gaussian kernel. KNM allocates a data point x to $\mathcal{C}_i$ if $\|F_h(\mathbf{x}) - \mathbf{u}_i\|^2 \leq \|F_h(\mathbf{x}) - \mathbf{u}_j\|^2$, for all $j \neq i$, where

$$\|F_h(\mathbf{x}) - \mathbf{u}_i\|^2 = K_h(\mathbf{x}, \mathbf{x}) + \frac{1}{N_i^2} \sum_{\mathbf{x}_j, \mathbf{x}_l \in \mathcal{C}_i} K_h(\mathbf{x}_j, \mathbf{x}_l) - \frac{2}{N_i} \sum_{\mathbf{x}_j \in \mathcal{C}_i} K_h(\mathbf{x}, \mathbf{x}_j), \tag{22}$$

with $\mathbf{u}_j = \frac{1}{N_j} \sum_{\mathbf{x}_i \in \mathcal{C}_j} F_h(\mathbf{x}_i)$, and $F_h(\cdot)$ and $K_h(\cdot, \cdot)$ are the nonlinear mapping and kernel function, respectively, corresponding to $\boldsymbol{\Theta}$.

Experiments on these classifiers are performed with the same setting. Specifically, we set $\beta = 18.5$ for the `Wisconsin breast cancer` and `sonar` data sets, and $\beta = 2.5$ for the `ionosphere` and `wine` data sets. In addition, we use the public Matlab package *SVMlight* to implement SVM, where the regularization parameter C is set to 300 for all four data sets. Results are averaged over 100 random splits of the data, one with 60% for training and 40% for testing, and another with 10% for training and 90% for testing.

Tables 1–2 and Figs. 4–5 show the results. The standard deviations with respect to 100 random splits are also given inside brackets. As can be seen, the classification accuracies of GWPC, KTDA, KFDA and SVM are almost the same. Moreover, they always outperform

Table 2. Test set accuracies (in %) obtained from the classification experiments (10% for training and 90% for testing). The highest accuracies are shown in boldface.

Method	Breast cancer	Ionosphere	Sonar	Wine
GWPC	**94.58** (±1.42)	85.58 (±5.63)	**70.45** (±4.73)	93.79 (±2.14)
KTDA	94.47 (±1.47)	**87.56** (±5.73)	70.22 (±4.59)	**94.59** (±2.00)
KFDA	93.30 (±1.82)	77.06 (±10.12)	67.07 (±5.55)	85.37 (±7.83)
SVM	93.35 (±2.05)	77.37 (±10.00)	67.07 (±5.50)	92.59 (±3.29)
KNM	90.89 (±1.51)	76.99 (±7.78)	65.63 (±5.81)	87.22 (±5.32)
1-NN	92.94 (±1.59)	81.14 (±4.27)	69.18 (±4.60)	91.71 (±3.00)

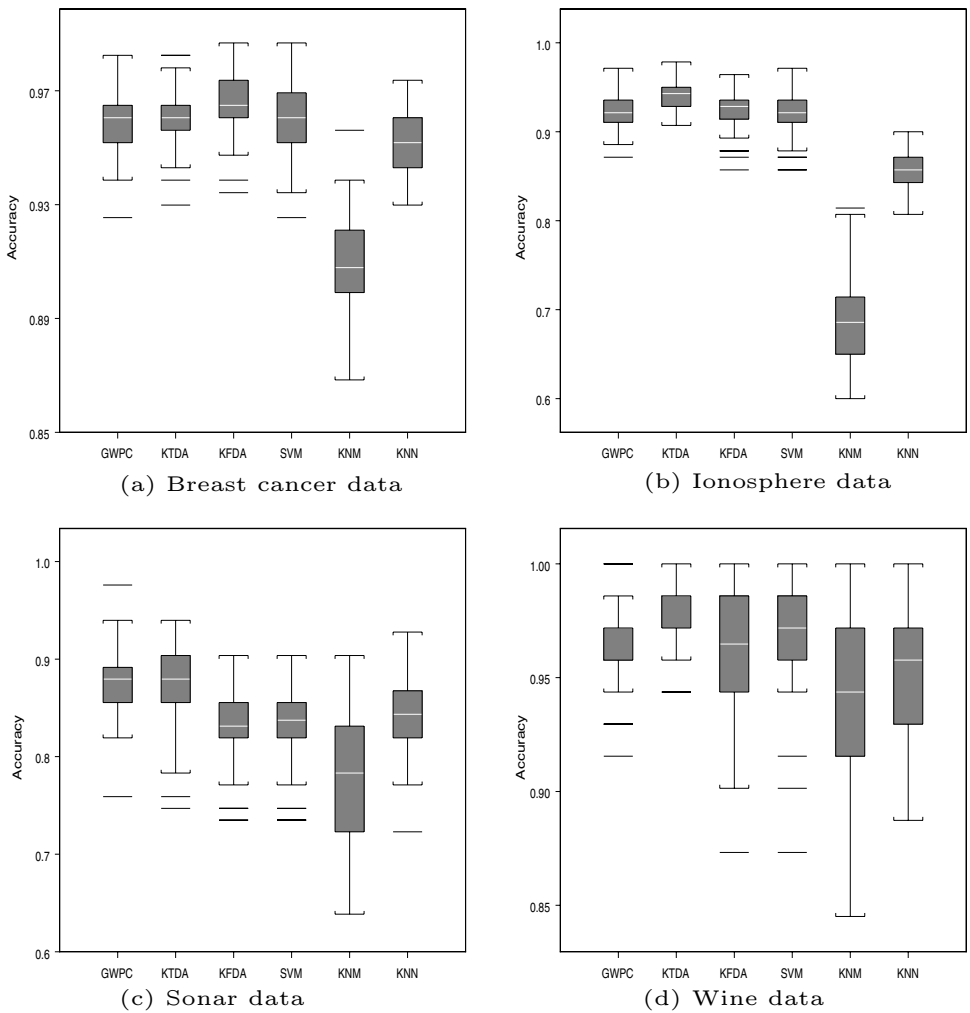

Fig. 4. Box plots of the classification results for GWPC, KTDA, KFDA, SVM, KNM and 1-NN (60% for training and 40% for testing).

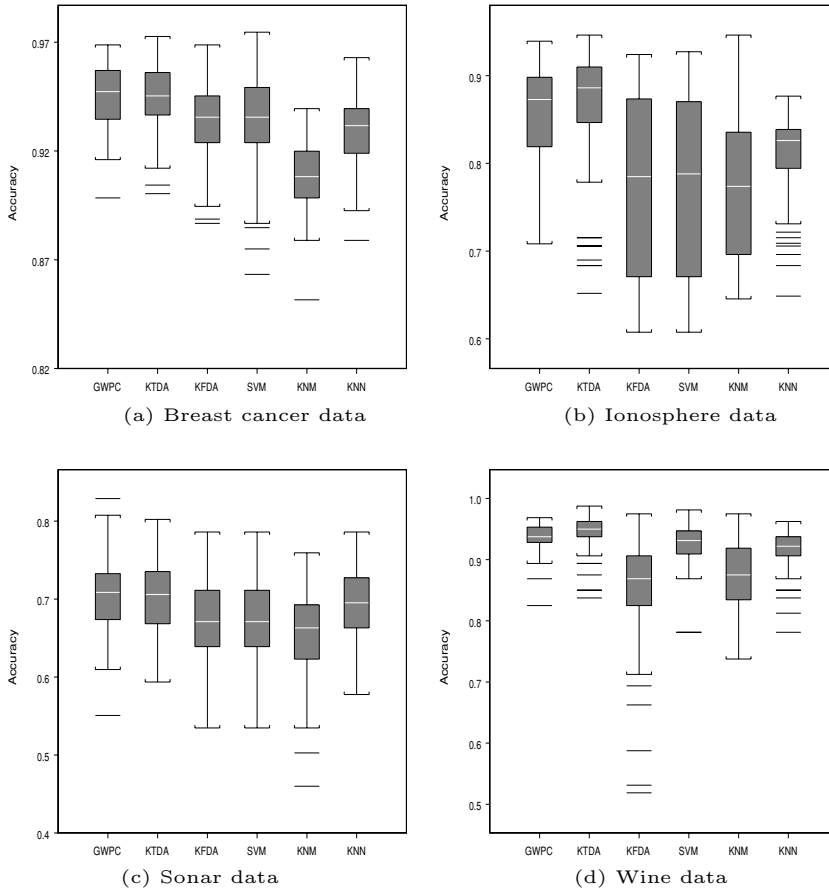

Fig. 5. Box plots of the classification results for GWPC, KTDA, KFDA, SVM, KNM and 1-NN (10% for training and 90% for testing).

KNM because they all utilize class label information from the training data during training but KNM does not. However, compared with KFDA and SVM, GWPC and KTDA are relatively insensitive to the training set size. Moreover, we find that KTDA generally outperforms GWPC, though we think that GWPC can be improved significantly by incorporating active learning.

As mentioned in Section 3.2, it is unnecessary to perform M-step(ii) for updating r in KTDA. However, in order to illustrate the dimensionality of the learned feature space, we also implement M-step(ii) in our experiments, where we initialize $r = n + 1$. Since $\eta = (\rho - n - 1)/r$ and $\rho > n$, one better choice for η is that $\eta \in [0, 1]$. In order to study the effect of η on r, we try both $\eta = 1.0$ and $\eta = 0.5$ in the experiments. For each of the 100 random data splits, r converges to a fixed point. As an illustrative example, we take one of the splits to demonstrate the convergence of r (Fig. 6). After the EM algorithm has converged, the average estimated values of r in the 100 random data splits for different values of n_1 (number of training examples) and η are shown in Table 3, showing that the

feature spaces are indeed of very high dimensionality. We find that for $\eta = 1.0$ and $\eta = 0.5$, the classification accuracy is insensitive, so we only report the classification results with $\eta = 0.5$. As can be seen, with a decrease in n_1 or η, the value of r increases. When n_1 gets smaller, the known part $\mathbf{K}_{11}$ of the kernel matrix becomes smaller. As a result, information from the hyperparameter matrix $\mathbf{\Theta}$, which is defined via the Gaussian kernel, will dominate. We know that the dimensionality of the feature space induced by the Gaussian kernel is infinite. This probably explains why r increases as a result. As for the relationships between r and η, recall that $\mathbf{C}$ is distributed according to $\mathcal{W}_n(\eta r + n + 1, (\eta r \mathbf{\Theta})^{-1})$ in our graphical model. Thus, there exists a tradeoff between r and η.

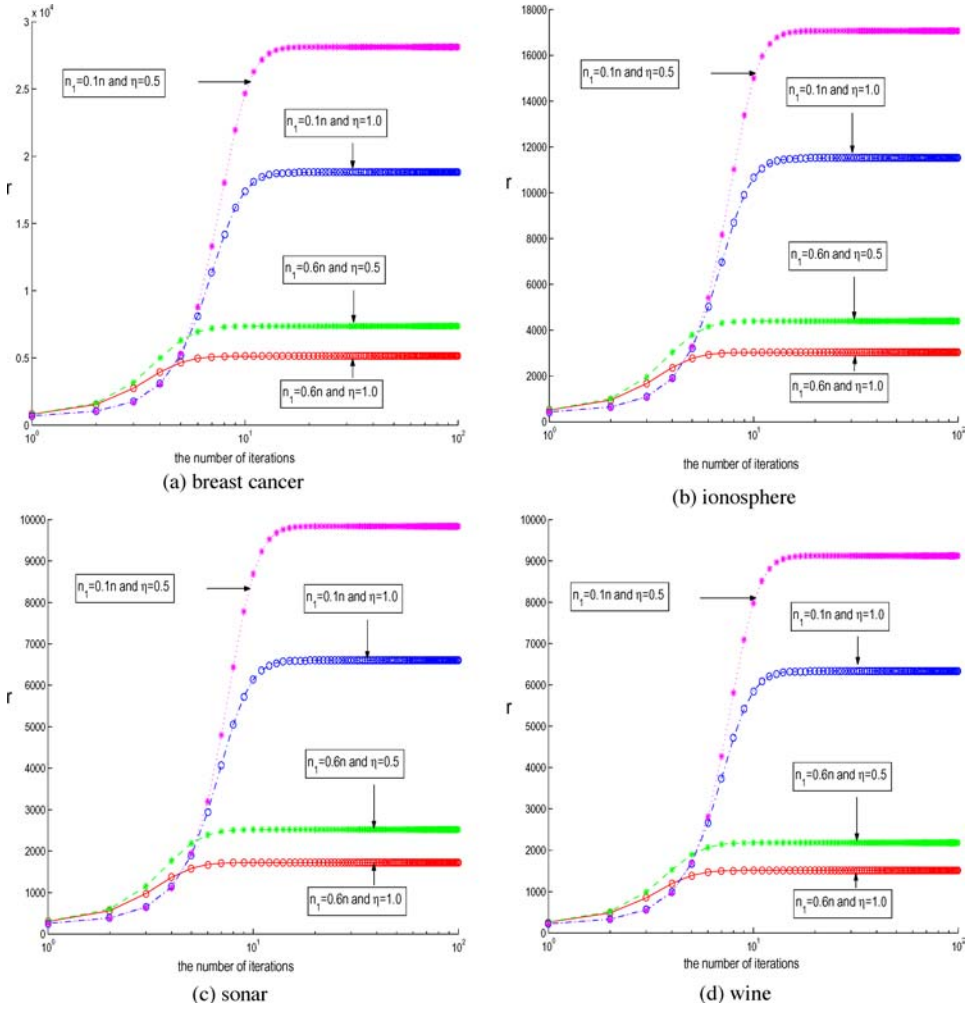

Fig. 6. Change of estimated value of r with the number of learning iterations for different settings.

Table 3. Average estimated values of r in the four benchmark data sets.

# training examples (n_1) in terms of data set size n	η	breast cancer ($n = 569$)	ionosphere ($n = 351$)	sonar ($n = 208$)	wine ($n = 178$)
	1.0	6,761 (±7)	3,040 (±10)	1,723 (±5)	1,502 (±8)
0.6 n	0.5	7,366 (±7)	4,407 (±13)	2,521 (±5)	2,187 (±7)
	1.0	19,015 (±159)	11,447 (±226)	6,569 (±128)	6,445 (±195)
0.1 n	0.5	28,118 (±195)	16,997 (±226)	9,796 (±155)	9,422 (±223)

Table 4. Average test set accuracies obtained from the classification experiments on the USPS database. The highest accuracies are shown in boldface.

# training examples	GWPC	KTDA	KFDA	SVM	KNM	1-NN
10%	**97.74** (± 0.31)	97.63 (± 0.34)	97.59 (± 0.36)	97.04 (± 0.49)	94.55 (± 0.45)	96.58 (± 0.37)
1%	**93.05** (± 1.27)	92.45 (± 1.36)	91.47 (± 3.83)	90.45 (± 1.65)	92.08 (± 1.30)	88.98 (± 1.75)

5.2. Results on USPS digit recognition

In this set of experiments, we use GWPC and KTDA to classify handwritten digits of size 16×16 from the USPS database. For simplicity, we only use digits 1, 2, 3 and 4 as four classes, comprising of 1269, 929, 824 and 852 examples, respectively. We set $\eta = 0.5$. The results are averaged over 100 random splits of the data, one with 10% for training and 90% for testing and the other with 1% for training and 99% for testing. Table 4 shows the averages and standard deviations of the test set accuracies over 100 random splits of the data. The corresponding box plots are shown in Fig. 7. Here, the regularization parameter C in the SVM is set to 300 and β in the Gaussian kernel is set to the average Euclidean distance between training examples. We see that with decreasing size of the training data set, both GWPC and KTDA outperform KFDA and SVM. Considering that the USPS data possesses good local consistency, we implemented the consistency method of Zhou et al. (2004) for comparison. The method was initialized with the classification result of 1-NN and ω in it is fixed at 0.95. When we used the value of β reported above, the classification accuracy of the consistency method (Zhou et al., 2004) is very low ($< 50\%$). We found that the β in this method prefers smaller values. Thus, we used $\beta = 10$. In this case, the consistency method obtained better accuracy, which is given in Table 5. However, the classification accuracies of SVM and KFDA are very low ($< 60\%$). Both GWPC and KTDA, whose accuracies are given in Table 5, are only slightly affected by changes in the value of β.

6. Conclusion

In this paper, we have proposed a model-based approach for transductive learning of the kernel matrix. Formulated as a missing value problem, we devise an EM algorithm for learning

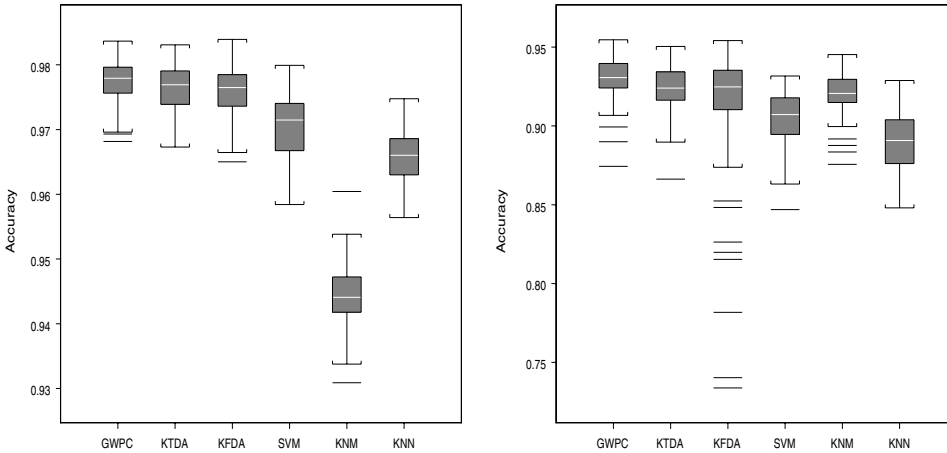

Fig. 7. Box plots of the classification results for GWPC, KTDA, KFDA, SVM, KNM and 1-NN on the USPS database.

Table 5. Average test set accuracies obtained from the classification experiments on the USPS database when $\beta = 10$. The highest accuracies are shown in boldface.

# training examples	GWPC	KTDA	Consistency method
10%	96.73 ($\pm$ 0.29)	96.61 ($\pm$ 0.42)	**98.04** ($\pm$ 0.29)
1%	90.84 ($\pm$ 1.32)	90.44 ($\pm$ 1.38)	**96.17** ($\pm$ 1.27)

the missing entries of the kernel matrix and the unknown parameters of the underlying distribution. We have demonstrated the efficacy of this approach by proposing two semi-supervised learning methods. In particular, we have studied our hierarchical transductive learning framework with the EM algorithm under the classification setting. In another work (Zhang et al., 2004), based on this same framework, we also devised the Tanner-Wong data augmentation algorithm (Tanner and Wong, 1987) which is a variant of MCMC. It is also possible to apply the framework to regression problems with Gaussian processes. This direction will be pursued in our future work.

Recall that in most existing kernel-based methods, only the kernel on the input set is used. However, in our first method, the target kernel $\mathbf{K}$ and the hyperparameter kernel $\boldsymbol{\Theta}$ are defined on the output set and input set, respectively. Their relationship is established through the parameter matrix $\boldsymbol{\Sigma}$ (or $\mathbf{C}$). So the parameter matrix plays a role similar to the cross-covariance kernel (Baker, 1973). In the second method, since the discriminant kernel $\mathbf{K}$ is the direct sum of the ideal kernel (Cristianini et al., 2002) on the output set and the Gaussian kernel on the input set, it can also be regarded as a cross-covariance kernel that relates the input space with the output space. In fact, our proposed methods for semi-supervised learning consist of two separate processes: the first explores the mutual relationship between kernels on the input and output sets through a hierarchical model, and the second implements the classification task with the target kernel or discriminant kernel.

A. Matrix theory

A.1 Matrix variate distributions

In the following, we will briefly introduce the concept of random matrices and some common matrix variate distributions. Interested readers are referred to (Gupta and Nagar, 2000) for more details.

Definition 2 *An $m \times n$* random matrix $\mathbf{X} = [x_{ij}]$ *is a matrix of random variables* $x_{11}, \ldots, x_{mn}$.

Obviously, random vectors and random variables are special cases of random matrices. Analogous to random vectors and random variables, random matrices also follow some distributions, called matrix variate distributions, with common examples including the normal, Wishart, and inverted Wishart distributions.

Definition 3. *An $s \times t$ random matrix* $\mathbf{X}$ *is said to follow the matrix variate* normal distribution *with mean matrix* $\mathbf{M}$ *and covariance matrix* $\mathbf{A} \otimes \mathbf{B}$ *(denoted* $\mathbf{X} \sim \mathcal{N}_{s,t}(\mathbf{M}, \mathbf{A} \otimes \mathbf{B})$*), where* $\mathbf{A}(s \times s) \succ 0$ *and* $\mathbf{B}(t \times t) \succ 0$*, if* $\text{vec}(\mathbf{X}') \sim \mathcal{N}(\text{vec}(\mathbf{M}'), \mathbf{A} \otimes \mathbf{B})$*. The corresponding p.d.f. is*

$$p(\mathbf{X}) = (2\pi)^{-st/2}|\mathbf{A}|^{-t/2}|\mathbf{B}|^{-s/2} \exp\left[-\frac{1}{2}\text{tr}\left(\mathbf{A}^{-1}(\mathbf{X}-\mathbf{M})\mathbf{B}^{-1}(\mathbf{X}-\mathbf{M})'\right)\right].$$

Definition 4. *An $m \times m$ symmetric positive definite random matrix* $\mathbf{W}$ *is said to follow the* Wishart distribution *(denoted* $\mathbf{W} \sim \mathcal{W}_m(\rho, \mathbf{S})$*), if*

$$p(\mathbf{W}) = \frac{1}{C(m,\rho)}|\mathbf{S}|^{-\rho/2}|\mathbf{W}|^{(\rho-m-1)/2} \exp\left(-\frac{1}{2}\text{tr}(\mathbf{S}^{-1}\mathbf{W})\right).$$

Here, $\rho \geq m$ is the *degree of freedom*, $\mathbf{S}(m \times m) \succ 0$ is the *parameter matrix*, and $C(m,\rho) = 2^{\rho m/2}\pi^{m(m-1)/4} \cdot \prod_{j=1}^{m} \Gamma(\frac{\rho+1-j}{2})$ is a normalization term with $\Gamma(\cdot)$ being the Gamma function.

Definition 5. *An $m \times m$ symmetric positive definite random matrix* $\mathbf{V}$ *is said to follow the* inverted Wishart distribution *(denoted* $\mathbf{V} \sim \mathcal{IW}_{\mathbf{m}}(\rho, \mathbf{T})$*) if*

$$p(\mathbf{V}) = \frac{1}{C(m,\rho)}|\mathbf{T}|^{\rho/2}|\mathbf{V}|^{-(\rho+m+1)/2} \exp\left(-\frac{1}{2}tr\,(\mathbf{T}\mathbf{V}^{-1})\right).$$

Some properties of these distributions are given in the subsequent part. In particular, the first moments of $\mathbf{W} \sim \mathcal{W}_m(\rho, \mathbf{S})$ and $\mathbf{V} \sim \mathcal{IW}_m(\rho, \mathbf{T})$ are $E(\mathbf{W}) = \rho\mathbf{S}$ and $E(\mathbf{V}) = \mathbf{T}/(\rho - m - 1)$, respectively.

Proposition 1

(1) *If* $\mathbf{X} \sim \mathcal{N}_{s,t}(\mathbf{M}, \mathbf{A} \otimes \mathbf{B})$*, then*

$$E(\mathbf{X}) = \mathbf{M},$$
$$E\left((\mathbf{X}-\mathbf{M})(\mathbf{X}-\mathbf{M})'\right) = \mathbf{A} \otimes \mathbf{B}.$$

(2) *If* $\mathbf{W} \sim \mathcal{W}_m(\rho, \mathbf{S})$, *then*

$$E\left(\log|\mathbf{W}|\right) = \log|\mathbf{S}| + m\log 2 + \sum_{j=0}^{m-1} \Psi\left(\frac{\rho - j}{2}\right).$$

Here, $E(\cdot)$ *denotes the expectation and* $\Psi(z) = \Gamma'(z)/\Gamma(z)$ *is the* digamma function.

Proof. The moments of $\mathbf{X}$ are given in (Gupta and Nagar, 2000). To obtain $E(\log|\mathbf{W}|)$, consider

$$\int |\mathbf{S}|^{-\rho/2}|\mathbf{W}|^{(\rho-m-1)/2}\exp\left(-\frac{1}{2}\mathrm{tr}\left(\mathbf{S}^{-1}\mathbf{W}\right)\right)(d\mathbf{W}) = C(m,\rho),$$

and take the derivatives of both sides with respect to ρ:

$$\frac{1}{2}\int \left(\log|\mathbf{W}| - \log|\mathbf{S}|\right)|\mathbf{S}|^{-\rho/2}|\mathbf{W}|^{(\rho-m-1)/2}\exp\left(-\frac{1}{2}\mathrm{tr}\left(\mathbf{S}^{-1}\mathbf{W}\right)\right)(d\mathbf{W}) = C'(m,\rho).$$

Take log of $C(m, \rho)$ as

$$\log C(m,\rho) = \frac{\rho m}{2}\log 2 + \frac{m(m-1)}{4}\log\pi + \sum_{j=0}^{m-1}\log\Gamma\left(\frac{\rho-j}{2}\right).$$

Then take the derivative of $\log C(m, \rho)$ with respect to ρ:

$$\frac{C'(m,\rho)}{C(m,\rho)} = \frac{m}{2}\log 2 + \frac{1}{2}\sum_{j=0}^{m-1}\frac{\Gamma'\left(\frac{\rho-j}{2}\right)}{\Gamma\left(\frac{\rho-j}{2}\right)} = \frac{m}{2}\log 2 + \frac{1}{2}\sum_{j=0}^{m-1}\Psi\left(\frac{\rho-j}{2}\right).$$

Thus,

$$E\left(\log|\mathbf{W}|\right) = \int \log|\mathbf{W}|\,p(\mathbf{W})(d\mathbf{W}) = \log|\mathbf{S}| + m\log 2 + \sum_{j=0}^{m-1}\Psi\left(\frac{\rho-j}{2}\right).$$

□

The following results, which can be found in (Gupta and Nagar, 2000), follow easily from the definitions.

Theorem 3. *Suppose* $\mathbf{W} \succ 0$ *and* $\mathbf{W} \sim \mathcal{W}_m(\rho, \mathbf{S})$. *Partition* $\mathbf{W}$ *and* $\mathbf{S}$ *as* $\begin{bmatrix}\mathbf{W}_{11} & \mathbf{W}_{12}\\ \mathbf{W}_{21} & \mathbf{W}_{22}\end{bmatrix}$ *and* $\begin{bmatrix}\mathbf{S}_{11} & \mathbf{S}_{12}\\ \mathbf{S}_{21} & \mathbf{S}_{22}\end{bmatrix}$ *respectively, where* $\mathbf{W}_{11}$ *and* $\mathbf{S}_{11}$ *are of size* $k \times k$. *Let* $\mathbf{W}_{22\cdot1} =$

$\mathbf{W}_{22} - \mathbf{W}_{21}\mathbf{W}_{11}^{-1}\mathbf{W}_{12}$ *and* $\mathbf{S}_{22\cdot 1} = \mathbf{S}_{22} - \mathbf{S}_{21}\mathbf{S}_{11}^{-1}\mathbf{S}_{12}$ *be the Schur complements of* $\mathbf{W}_{11}$ *and* $\mathbf{S}_{11}$*, respectively. Then*

(i) $\mathbf{W}_{11} \sim \mathcal{W}_k(\rho, \mathbf{S}_{11})$ *and* $\mathbf{W}_{22} \sim \mathcal{W}_{m-k}(\rho, \mathbf{S}_{22})$*;*
(ii) $\mathbf{W}_{21} \mid \mathbf{W}_{11} \sim \mathcal{N}(\mathbf{S}_{21}\mathbf{S}_{11}^{-1}\mathbf{W}_{11}, \mathbf{S}_{22\cdot 1} \otimes \mathbf{W}_{11})$*; and*
(iii) $\mathbf{W}_{22\cdot 1} \sim \mathcal{W}_{m-k}(\rho - k, \mathbf{S}_{22\cdot 1})$ *and is independent of* $\mathbf{W}_{21}$ *and* $\mathbf{W}_{11}$.

Theorem 4. *If* $\mathbf{W} \sim \mathcal{W}_m(\rho, \mathbf{S})$*, then* $\mathbf{W}^{-1} \sim \mathcal{IW}_m(\rho, \mathbf{S}^{-1})$.

A.2 Wedge product and matrix differentials

For any matrix $\mathbf{X}$, we denote the matrix of differentials (dx_{ij}) by $d\mathbf{X}$.

Definition 6. *For an arbitrary* $m \times n$ *matrix* $\mathbf{X}$*,* $(d\boldsymbol{X})$ *denotes the wedge product (or exterior product) of all mn elements of* $d\boldsymbol{X}$*:*

$$(d\mathbf{X}) \equiv dx_{11} \wedge \cdots \wedge dx_{1n} \wedge \cdots \wedge dx_{m1} \wedge \cdots \wedge dx_{mn}.$$

If $\mathbf{X}$ *is a symmetric* $m \times m$ *matrix,* $(d\mathbf{X})$ *is the wedge product of the* $\frac{1}{2}m(m+1)$ *distinct elements of* $d\mathbf{X}$*:*

$$(d\mathbf{X}) \equiv dx_{11} \wedge \cdots \wedge dx_{1m} \wedge dx_{22} \wedge \cdots \wedge dx_{2m} \wedge \cdots \wedge dx_{mm}.$$

We list below some results of matrix calculus that will be used in the sequel.

Proposition 2

(a) *If* $\mathbf{X}$ *and* $\mathbf{A}$ *are* $p \times q$ *and* $q \times p$*, then*

$$\frac{\partial \operatorname{tr}(\mathbf{XA})}{\partial \mathbf{X}} = \mathbf{A}';$$

(b) *If* $\mathbf{X}$*,* $\mathbf{A}$ *and* $\mathbf{B}$ *are* $p \times q$*,* $q \times q$ *and* $p \times p$*, then*

$$\frac{\partial \operatorname{tr}(\mathbf{AX}'\mathbf{BX})}{\partial \mathbf{X}} = \mathbf{BXA} + \mathbf{B}'\mathbf{XA}';$$

(c) *If* $\mathbf{X}$ *is a* $p \times p$ *symmetric positive definite matrix, then*

$$\frac{\partial \log \mid \mathbf{X} \mid}{\partial \mathbf{X}} = 2\mathbf{X}^{-1} - \operatorname{diag}(\mathbf{X}^{-1}),$$

$$\frac{\partial \operatorname{tr}(\mathbf{XA})}{\partial \mathbf{X}} = \mathbf{A} + \mathbf{A}' - \operatorname{diag}(\mathbf{A});$$

(d) *If* $\mathbf{X}$ *is a* $p \times p$ *symmetric positive definite matrix and* $\mathbf{A}$ *and* $\mathbf{B}$ *are* $q \times p$ *and* $p \times q$*, then*

$$\frac{\partial \operatorname{tr}(\mathbf{AX}^{-1}\mathbf{B})}{\partial \mathbf{X}} = -\mathbf{X}^{-1}(\mathbf{BA} + \mathbf{A}'\mathbf{B}')\mathbf{X}^{-1} + \operatorname{diag}(\mathbf{X}^{-1}\mathbf{BAX}^{-1}).$$

Proof. Here we only prove (d). As $\mathbf{I} = \mathbf{X}\mathbf{X}^{-1}$, we have

$$0 = \partial \mathbf{I}/\partial x = \partial/\partial x(\mathbf{X}\mathbf{X}^{-1}) = \partial/\partial x(\mathbf{X})\mathbf{X}^{-1} + \mathbf{X}\partial/\partial x(\mathbf{X}^{-1}).$$

If $i \neq j$, since $\mathbf{X}$ is symmetric, then $\partial \mathbf{X}/\partial x_{ij} = \mathbf{e}_i\mathbf{e}'_j + \mathbf{e}_j\mathbf{e}'_i$, where $\boldsymbol{e}_\mathrm{i}$ is the ith column of $\mathbf{I}$. It then follows that $\partial/\partial x_{ij}(\mathbf{X}^{-1}) = -\mathbf{X}^{-1}(\mathbf{e}_i\mathbf{e}'_j + \mathbf{e}_j\mathbf{e}'_i)\mathbf{X}^{-1}$. Thus,

$$\begin{aligned}
\partial/\partial x_{ij}(\mathrm{tr}\,(\mathbf{A}\mathbf{x}^{-1}\mathbf{B})) &= -\mathrm{tr}\,\left(\mathbf{A}\mathbf{x}^{-1}(\mathbf{e}_i\mathbf{e}'_j + \mathbf{e}_j\mathbf{e}'_i)\mathbf{X}^{-1}\mathbf{B}\right) \\
&= -\mathrm{tr}\,\left(\mathbf{A}\mathbf{X}^{-1}\mathbf{e}_i\mathbf{e}'_j\mathbf{X}^{-1}\mathbf{B}\right) - \mathrm{tr}\,\left(\mathbf{A}\mathbf{X}^{-1}\mathbf{e}_j\mathbf{e}'_i\mathbf{X}^{-1}\mathbf{B}\right) \\
&= -\mathbf{e}'_j\mathbf{X}^{-1}\mathbf{B}\mathbf{A}\mathbf{X}^{-1}\mathbf{e}_i - \mathbf{e}'_i\mathbf{X}^{-1}\mathbf{B}\mathbf{A}\mathbf{X}^{-1}\mathbf{e}_j \\
&= -(\mathbf{X}^{-1}\mathbf{B}\mathbf{A}\mathbf{X}^{-1})_{ji} - (\mathbf{X}^{-1}\mathbf{B}\mathbf{A}\mathbf{X}^{-1})_{ij} \\
&= -(\mathbf{X}^{-1}\mathbf{A}'\mathbf{B}'\mathbf{X}^{-1})_{ij} - (\mathbf{X}^{-1}\mathbf{B}\mathbf{A}\mathbf{X}^{-1})_{ij}.
\end{aligned}$$

On the other hand, as $\partial \mathbf{X}/\partial x_{ii} = \mathbf{e}_i\mathbf{e}'_i$, thus

$$\partial/\partial x_{ii}(\mathrm{tr}(\mathbf{A}\mathbf{X}^{-1}\mathbf{B})) = -(\mathbf{X}^{-1}\mathbf{A}'\mathbf{B}'\mathbf{X}^{-1})_{ii}.$$

Result follows from combining the two. □

A.3 The kronecker product of matrices

Definition 7. *Let* $\mathbf{A} = (a_{ij})$ *be a* $p \times q$ *matrix and* $\mathbf{B} = (b_{ij})$ *be an* $s \times t$ *matrix. The Kronecker product of* $\mathbf{A}$ *and* $\mathbf{B}$*, denoted by* $\mathbf{A} \otimes \mathbf{B}$*, is the* $ps \times qt$ *matrix*

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \dots & a_{1q}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \dots & a_{2q}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p1}\mathbf{B} & a_{p2}\mathbf{B} & \dots & a_{pq}\mathbf{B} \end{bmatrix}.$$

Some important properties of the Kronecker product are listed in the following.

Proposition 3

(a) $(\mathbf{A} \otimes \mathbf{B})' = \mathbf{A}' \otimes \mathbf{B}'$.
(b) *If* $\mathbf{A}$ *and* $\mathbf{B}$ *are both* $n \times n$*, then* $\mathrm{tr}(\mathbf{A} \otimes \mathbf{B}) = \mathrm{tr}(\mathbf{A})\mathrm{tr}(\mathbf{B})$.
(c) *If* $\mathbf{A}$ *and* $\mathbf{B}$ *are both* $n \times n$*, then* $|\mathbf{A} \otimes \mathbf{B}| = |\mathbf{A}|^n \cdot |\mathbf{B}|^n$.
(d) *If* $\boldsymbol{A}$ *is* $k \times l$, $\mathbf{B}$ *is* $p \times q$, $\mathbf{X}$ *is* $l \times s$ *and* $\boldsymbol{Y}$ *is* $q \times t$, *then* $(\mathbf{A} \otimes \mathbf{B})(\mathbf{X} \otimes \mathbf{Y}) = \mathbf{A}\mathbf{X} \otimes \mathbf{B}\mathbf{Y}$.
(e) *If* $\mathbf{A}$ *and* $\mathbf{B}$ *are nonsingular, then* $(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1}$.

The following proposition (Horn and Johnson, 1991) shows the connection between Kronecker product and the vec of a matrix.

Proposition 4 *If $\mathbf{A}$ is $t \times k$, $\mathbf{X}$ is $k \times l$, $\mathbf{B}$ is $l \times s$, $\mathbf{Y}$ is $l \times l$ and $\mathbf{D}$ is $l \times t$, then*

(i) $\text{vec}(\mathbf{AXB}) = (\mathbf{B}' \otimes \mathbf{A})\text{vec}(\mathbf{X})$
(ii) $\text{tr}(\mathbf{AXD}) = (\text{vec}(\mathbf{A}'))'(\mathbf{I} \otimes \mathbf{X})\text{vec}(\mathbf{D})$
(iii) $\text{tr}(\mathbf{AXYX}'\mathbf{D}) = (\text{vec}(\mathbf{X}'))'(\mathbf{DA} \otimes \mathbf{Y}')\text{vec}(\mathbf{X}') = (\text{vec}(\mathbf{X}'))'(\mathbf{A}'\mathbf{D}' \otimes \mathbf{Y})\text{vec}(\mathbf{X}')$

A.4 Proof of Lemma 1

Lemma 2 *With $\mathbf{C} = \sum^{-1}$ as partitioned in (5), we have $\mathbf{C}_{11} = \mathbf{\Sigma}_{11\cdot 2}^{-1}$, $\mathbf{C}_{11}^{-1}\mathbf{C}_{12} = -\mathbf{\Sigma}_{12}\mathbf{\Sigma}_{22}^{-1}$, $\mathbf{C}_{22} = \mathbf{\Sigma}_{22\cdot 1}^{-1}$ and $\mathbf{C}_{22}^{-1}\mathbf{C}_{21} = -\mathbf{\Sigma}_{21}\mathbf{\Sigma}_{11}^{-1}$.*

Proof. As

$$\mathbf{\Sigma} \cdot \mathbf{C} = \begin{bmatrix} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12} \\ \mathbf{\Sigma}_{21} & \mathbf{\Sigma}_{22} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix} = \mathbf{I},$$

we have

$$\begin{cases} \mathbf{\Sigma}_{11}\mathbf{C}_{11} + \mathbf{\Sigma}_{12}\mathbf{C}_{21} = \mathbf{I}, \\ \mathbf{\Sigma}_{11}\mathbf{C}_{12} + \mathbf{\Sigma}_{12}\mathbf{C}_{22} = \mathbf{0}, \\ \mathbf{\Sigma}_{21}\mathbf{C}_{11} + \mathbf{\Sigma}_{22}\mathbf{C}_{21} = \mathbf{0}, \\ \mathbf{\Sigma}_{21}\mathbf{C}_{12} + \mathbf{\Sigma}_{22}\mathbf{C}_{22} = \mathbf{I}. \end{cases}$$

Thus, $\mathbf{C}_{21} = -\mathbf{\Sigma}_{22}^{-1}\mathbf{\Sigma}_{21}\mathbf{C}_{11}$, $\mathbf{C}_{12} = -\mathbf{\Sigma}_{11}^{-1}\mathbf{\Sigma}_{12}\mathbf{C}_{22}$, $(\mathbf{\Sigma}_{11} - \mathbf{\Sigma}_{12}\mathbf{\Sigma}_{22}^{-1}\mathbf{\Sigma}_{21})\mathbf{C}_{11} = \mathbf{I}$ and $(\mathbf{\Sigma}_{22} - \mathbf{\Sigma}_{21}\mathbf{\Sigma}_{11}^{-1}\mathbf{\Sigma}_{12})\mathbf{C}_{22} = \mathbf{I}$, and result follows. □

B. Details of the proposed EM algorithm

B.1 Derivation of the E-step

The E-step is equivalent to computing the expectation $Q(\mathbf{C}, r \mid \mathbf{C}(t), r(t))$ of the complete data log-likelihood function:

$$\begin{aligned} &Q(\mathbf{C}, r \mid \mathbf{C}(t), r(t)) \\ &= E\left[\log p\left(\mathbf{K} \mid \mathbf{C}, r\right) \mid \mathbf{K}_{11}, \mathbf{C}(t), r(t)\right] \\ &= \int L\left(\mathbf{C}, r \mid \mathbf{K}\right) p(\mathbf{K}_{22\cdot 1} \mid \mathbf{C}(t), r(t)) p(\mathbf{K}_{21} \mid \mathbf{K}_{11}, \mathbf{C}(t), r(t))(d\mathbf{K}_{22\cdot 1}) \wedge (d\mathbf{K}_{21}). \quad (23) \end{aligned}$$

Substituting (6) into (23), we obtain

$$\begin{aligned} Q(\mathbf{C}, r \mid \mathbf{C}(t), r(t)) = {} & \frac{rn}{2}\log r - \log C(n, r) + \frac{r}{2}\log|\mathbf{C}_{11\cdot 2}| + \frac{r}{2}\log|\mathbf{C}_{22}| \\ & - \frac{r}{2}\text{tr}(\mathbf{C}_{11\cdot 2}\mathbf{K}_{11}) - \frac{r}{2}\text{tr}(\mathbf{C}_{12}\mathbf{C}_{22}^{-1}\mathbf{C}_{21}\mathbf{K}_{11}) + \frac{r-n-1}{2}\log|\mathbf{K}_{11}| \end{aligned}$$

$$+\frac{r-n-1}{2}\int \log|\mathbf{K}_{22\cdot1}|p(\mathbf{K}_{22\cdot1}\mid \mathbf{C}(t),r(t))(d\mathbf{K}_{22\cdot1})$$

$$-\frac{r}{2}\int \mathrm{tr}(\mathbf{C}_{22}\mathbf{K}_{22\cdot1})p(\mathbf{K}_{22\cdot1}\mid \mathbf{C}(t),r(t))(d\mathbf{K}_{22\cdot1})$$

$$-r\int \mathrm{tr}(\mathbf{C}_{12}\mathbf{K}_{21})p(\mathbf{K}_{21}\mid \mathbf{K}_{11},\mathbf{C}(t),r(t))(d\mathbf{K}_{21})$$

$$-\frac{r}{2}\int \mathrm{tr}(\mathbf{C}_{22}\mathbf{K}_{21}\mathbf{K}_{11}^{-1}\mathbf{K}_{12})p(\mathbf{K}_{21}\mid \mathbf{K}_{11},\mathbf{C}(t),r(t))(d\mathbf{K}_{21}). \quad (24)$$

Using Corollary 1 and Proposition 1 in Appendix A.1, we have:

$$\int \mathrm{tr}(\mathbf{C}_{22}\mathbf{K}_{22\cdot1})p(\mathbf{K}_{22\cdot1}\mid \mathbf{C}_{22}(t),r(t))(d\mathbf{K}_{22\cdot1})=\frac{r(t)-n_1}{r(t)}\mathrm{tr}\left(\mathbf{C}_{22}\mathbf{C}_{22}^{-1}(t)\right),$$

$$\int \log|\mathbf{K}_{22\cdot1}|p(\mathbf{K}_{22\cdot1}\mid \mathbf{C}(t),r(t))(d\mathbf{K}_{22\cdot1})=n_2\log\frac{2}{r(t)}+\sum_{j=0}^{n_2-1}\Psi\left(\frac{r(t)-n_1-j}{2}\right)-\log|\mathbf{C}_{22}(t)|,$$

$$\int \mathrm{tr}(\mathbf{C}_{12}\mathbf{K}_{21})p(\mathbf{K}_{21}\mid \mathbf{K}_{11},\mathbf{C}(t),r(t))(d\mathbf{K}_{21})=-\mathrm{tr}\left(\mathbf{C}_{12}\mathbf{C}_{2|1}(t)\mathbf{K}_{11}\right). \quad (25)$$

As a result of using the statement "$\mathbf{Y}$ is $\mathcal{N}(\mathbf{M},\mathbf{C}\otimes\mathbf{D})$" is equivalent to the statement that "$\mathbf{y}$ is $\mathcal{N}(\mathbf{m},\mathbf{C}\otimes\mathbf{D})$" with $\mathbf{y}=\mathrm{vec}(\mathbf{Y}')$ and $\mathbf{m}=\mathrm{vec}(\mathbf{M}')$, we obtain

$$E\left(\mathrm{vec}(\mathbf{K}_{12})|\mathbf{K}_{11}\right)=-\mathrm{vec}\left(\mathbf{K}_{11}\mathbf{C}_{12}\mathbf{C}_{22}^{-1}\right)=-\left(\mathbf{C}_{22}^{-1}\otimes\mathbf{K}_{11}\right)\mathrm{vec}(\mathbf{C}_{12}),$$

$$E\left(\mathrm{vec}(\mathbf{K}_{12})(\mathrm{vec}(\mathbf{K}_{12}))'|\mathbf{K}_{11}\right)=\left(\mathbf{C}_{22}^{-1}\otimes\mathbf{K}_{11}\right)\mathrm{vec}(\mathbf{C}_{12})(\mathrm{vec}(\mathbf{C}_{12}))'\left(\mathbf{C}_{22}^{-1}\otimes\mathbf{K}_{11}\right)+\frac{1}{r}\mathbf{C}_{22}^{-1}\otimes\mathbf{K}_{11}.$$

On the other hand,

$$\mathrm{tr}\left(\mathbf{C}_{22}\mathbf{K}_{21}\mathbf{K}_{11}^{-1}\mathbf{K}_{12}\right)=(\mathrm{vec}(\mathbf{K}_{12}))'\left(\mathbf{I}\mathbf{C}_{22}\otimes\mathbf{K}_{11}^{-1}\right)\mathrm{vec}(\mathbf{K}_{12})=\mathrm{tr}\left(\left(\mathbf{C}_{22}\otimes\mathbf{K}_{11}^{-1}\right)\mathrm{vec}(\mathbf{K}_{12})(\mathrm{vec}(\mathbf{K}_{12}))'\right).$$

It then follows from Propositions 3 and 4 in Appendix A.3 that

$$\int \mathrm{tr}(\mathbf{C}_{22}\mathbf{K}_{21}\mathbf{K}_{11}^{-1}\mathbf{K}_{21}')p(\mathbf{K}_{21}\mid \mathbf{K}_{11},\mathbf{C}(t),r(t))(d\mathbf{K}_{21})$$

$$=\int \mathrm{tr}\left((\mathbf{C}_{22}\otimes\mathbf{K}_{11}^{-1})\mathrm{vec}(\mathbf{K}_{12})(\mathrm{vec}(\mathbf{K}_{12}))'\right)p\left(\mathrm{vec}(\mathbf{K}_{12})\mid \mathbf{K}_{11},\mathbf{C}(t),r(t)\right)d\mathrm{vec}(\mathbf{K}_{12})$$

$$=\frac{n_1}{r(t)}\mathrm{tr}\left(\mathbf{C}_{22}\mathbf{C}_{22}^{-1}(t)\right)+\mathrm{tr}\left(\mathbf{C}_{22}\mathbf{C}_{2|1}(t)\mathbf{K}_{11}\mathbf{C}_{2|1}'(t)\right). \quad (26)$$

So we obtain (7) through substituting (25) and (26) back into (24). It is worthy to note that

$$\{\mathbf{K}_{22\cdot1}, \log|\mathbf{K}_{22\cdot1}|, \mathbf{K}_{21}, \text{vec}(\mathbf{K}'_{21})(\text{vec}(\mathbf{K}'_{21}))'\}$$

are complete-data *sufficient statistic* for $\{\mathbf{C}_{11\cdot2}, \mathbf{C}_{21}, \mathbf{C}_{22}, r\}$.

B.2 Derivation of the M-step

After some calculations, we have

$$\begin{aligned}\log p(\mathbf{C} \mid \mathbf{\Theta}, r) = {} & \frac{(\eta r + n + 1)n}{2}\log(\eta r) - \log C(n, \eta r + n + 1) + \frac{\eta r + n + 1}{2}\log|\mathbf{\Theta}| \\ & + \frac{\eta r}{2}\log|\mathbf{C}_{11\cdot2}| - \frac{\eta r}{2}\text{tr}(\mathbf{C}_{11\cdot2}\mathbf{\Theta}_{11}) - \frac{\eta r}{2}\text{tr}\left(\mathbf{C}_{12}\mathbf{C}_{22}^{-1}\mathbf{C}_{21}\mathbf{\Theta}_{11}\right) \\ & - \eta r\,\text{tr}(\mathbf{C}_{21}\mathbf{\Theta}_{12}) - \frac{\eta r}{2}\text{tr}(\mathbf{C}_{22}\mathbf{\Theta}_{22}) + \frac{\eta r}{2}\log|\mathbf{C}_{22}|. \end{aligned} \tag{27}$$

Our M-step is now to maximize $Q(\mathbf{C}, r \mid \mathbf{C}(t), r(t)) + \log p(\mathbf{C} \mid \mathbf{\Theta}, r, \eta)$ with respect to $\mathbf{C}$ and r, and then obtain $\mathrm{C}(t+1)$ and $r(t+1)$. Letting $F(\mathbf{C}, r \mid \mathbf{C}(t), r(t)) = Q(\mathbf{C}, r \mid \mathbf{C}(t), r(t)) + \log p(\mathbf{C} \mid \mathbf{\Theta}, r)$, we reformulate it as

$$F(\mathbf{C}, r \mid \mathbf{C}(t), r(t)) = \frac{r}{2}Q_1(\mathbf{C} \mid \mathbf{C}(t)) + \frac{1}{2}Q_2(r \mid \mathbf{C}(t), r(t)),$$

where $Q_1(\mathbf{C} \mid \mathbf{C}(t))$ and $Q_2(r \mid \mathbf{C}(t), r(t))$ are defined in (11) and (12), respectively. As

$$\begin{cases} \frac{\partial F(\mathbf{C}, r \mid \mathbf{C}(t), r(t))}{\partial \mathbf{C}} = 0 \\ \frac{\partial F(\mathbf{C}, r \mid \mathbf{C}(t), r(t))}{\partial r} = 0 \end{cases} \Longleftrightarrow \begin{cases} \frac{\partial Q_1(\mathbf{C} \mid \mathbf{C}(t))}{\partial \mathbf{C}} = 0 \\ \frac{\partial Q_2(r \mid \mathbf{C}(t), r(t))}{\partial r} + Q_1(\mathbf{C} \mid \mathbf{C}(t)) = 0, \end{cases}$$

our M-step can be separated into two parts: first, obtain the $(t+1)$th estimate of $\mathbf{C}$ by solving $\frac{\partial Q_1(\mathbf{C} \mid \mathbf{C}(t))}{\partial \mathbf{C}} = 0$; then, obtain the $(t+1)$th estimate of r by solving (10). For the first part, using Proposition 2 in Appendix A.2, we obtain the derivatives of $Q_1(\mathbf{C} \mid \mathbf{C}(t))$ with respect to $\mathbf{C}_{11\cdot2}$, $\mathbf{C}_{21}$ and $\mathbf{C}_{22}$ as

$$\begin{aligned} \frac{\partial Q_1}{\partial \mathbf{C}_{11\cdot2}} = {} & 2(1+\eta)\mathbf{C}_{11\cdot2}^{-1} - (1+\eta)\text{diag}\left(\mathbf{C}_{11\cdot2}^{-1}\right) - 2(\mathbf{K}_{11} + \eta\mathbf{\Theta}_{11}) \\ & + \text{diag}(\mathbf{K}_{11} + \eta\mathbf{\Theta}_{11}), \\ \frac{\partial Q_1}{\partial \mathbf{C}_{21}} = {} & 2\mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)\mathbf{K}_{11} - 2\eta\mathbf{\Theta}_{21} - 2\mathbf{C}_{22}^{-1}\mathbf{C}_{21}(\mathbf{K}_{11} + \eta\mathbf{\Theta}_{11}), \\ \frac{\partial Q_1}{\partial \mathbf{C}_{22}} = {} & 2(1+\eta)\mathbf{C}_{22}^{-1} - (1+\eta)\text{diag}\left(\mathbf{C}_{22}^{-1}\right) - 2\left(\mathbf{C}_{22}^{-1}(t) + \eta\mathbf{\Theta}_{22}\right) \\ & + \text{diag}\left(\mathbf{C}_{22}^{-1}(t) + \eta\mathbf{\Theta}_{22}\right) - 2\mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)\mathbf{K}_{11}\mathbf{C}_{12}(t)\mathbf{C}_{22}^{-1}(t) \\ & + \text{diag}\left(\mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)\mathbf{K}_{11}\mathbf{C}_{12}(t)\mathbf{C}_{22}^{-1}(t)\right) \end{aligned}$$

$$+ 2\mathbf{C}_{22}^{-1}\mathbf{C}_{21}(\mathbf{K}_{11} + \eta\boldsymbol{\Theta}_{11})\mathbf{C}_{12}\mathbf{C}_{22}^{-1}$$
$$- \operatorname{diag}\left(\mathbf{C}_{22}^{-1}\mathbf{C}_{21}(\mathbf{K}_{11} + \eta\boldsymbol{\Theta}_{11})\mathbf{C}_{12}\mathbf{C}_{22}^{-1}\right).$$

As $2\mathbf{A} - \operatorname{diag}(\mathbf{A}) = \mathbf{0}$ is equivalent to $\mathbf{A} = \mathbf{0}$, the saddle point equations of F with respect to $\mathbf{C}_{11\cdot 2}$, $\mathbf{C}_{21}$ and $\mathbf{C}_{22}$ are

$$\begin{aligned}
\mathbf{C}_{11\cdot 2} &= (1+\eta)(\mathbf{K}_{11} + \eta\boldsymbol{\Theta}_{11})^{-1},\\
\mathbf{C}_{22}^{-1}\mathbf{C}_{21} &= (\mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)\mathbf{K}_{11} - \eta\boldsymbol{\Theta}_{21})(\mathbf{K}_{11} + \eta\boldsymbol{\Theta}_{11})^{-1},\\
(1+\eta)\mathbf{C}_{22}^{-1} &= \mathbf{C}_{22}^{-1}(t) + \eta\boldsymbol{\Theta}_{22} + \mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)\mathbf{K}_{11}\mathbf{C}_{12}(t)\mathbf{C}_{22}^{-1}(t)\\
&\quad -\mathbf{C}_{22}^{-1}\mathbf{C}_{21}(\mathbf{K}_{11} + \eta\boldsymbol{\Theta}_{11})\mathbf{C}_{12}\mathbf{C}_{22}^{-1}.
\end{aligned}$$

Substituting the second equation into the third equation above, we obtain the M-step in (11).

B.3 Proof of $\mathbf{C}(t+1) \succ 0$

Assuming that $\mathbf{C}(t) \succ 0$, we now proceed to prove that $\mathbf{C}(t+1)$ given in (9) is also positive definite. Consider the following equality:

$$\begin{aligned}
&\mathbf{C}_{22}^{-1}(t) + \eta\boldsymbol{\Theta}_{22} + \mathbf{C}_{2|1}(t)\mathbf{K}_{11}\mathbf{C}_{2|1}'(t) - \mathbf{C}_{2|1}(t+1)(\mathbf{K}_{11} + \eta\boldsymbol{\Theta}_{11})\mathbf{C}_{2|1}'(t+1)\\
&= \mathbf{C}_{22}^{-1}(t) + \eta\boldsymbol{\Theta}_{22} + \mathbf{C}_{2|1}(t)\mathbf{K}_{11}\mathbf{C}_{2|1}'(t)\\
&\quad -(\mathbf{C}_{2|1}(t)\mathbf{K}_{11} - \eta\boldsymbol{\Theta}_{21})(\mathbf{K}_{11} + \eta\boldsymbol{\Theta}_{11})^{-1}(\mathbf{C}_{2|1}(t)\mathbf{K}_{11} - \eta\boldsymbol{\Theta}_{21})'\\
&= \mathbf{C}_{22}^{-1}(t) + \eta\boldsymbol{\Theta}_{22} + \mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)\mathbf{K}_{11}\mathbf{C}_{12}(t)\mathbf{C}_{22}^{-1}(t)\\
&\quad -(\mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)\mathbf{K}_{11} - \eta\boldsymbol{\Theta}_{21})(\mathbf{K}_{11} + \eta\boldsymbol{\Theta}_{11})^{-1}(\mathbf{K}_{11}\mathbf{C}_{12}(t)\mathbf{C}_{22}^{-1}(t) - \eta\boldsymbol{\Theta}_{12})\\
&= \mathbf{D}_{22}(t) + \eta\boldsymbol{\Theta}_{22} - (\mathbf{D}_{21}(t) + \eta\boldsymbol{\Theta}_{21})(\mathbf{D}_{11}(t) + \eta\boldsymbol{\Theta}_{11})^{-1}(\mathbf{D}_{12}(t) + \eta\boldsymbol{\Theta}_{12}),
\end{aligned}$$

where $\mathbf{D}_{11}(t) = \mathbf{K}_{11}$, $\mathbf{D}_{21}(t) = \mathbf{D}_{12}'(t) = -\mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)\mathbf{K}_{11}$ and $\mathbf{D}_{22}(t) = \mathbf{C}_{22}^{-1}(t) + \mathbf{C}_{22}^{-1}(t)\mathbf{C}_{21}(t)\mathbf{K}_{11}\mathbf{C}_{12}(t)\mathbf{C}_{22}^{-1}(t)$. Now we define a new matrix as

$$\mathbf{D}(t) = \begin{bmatrix} \mathbf{D}_{11}(t) & \mathbf{D}_{12}(t) \\ \mathbf{D}_{21}(t) & \mathbf{D}_{22}(t) \end{bmatrix}. \tag{28}$$

It is easy to obtain $\mathbf{D}_{22\cdot 1}(t) = \mathbf{C}_{22}^{-1}(t)$. So we have $\mathbf{D}(t) \succ 0$ and $\mathbf{D}(t) + \eta\boldsymbol{\Theta} \succ 0$. This then follows that $\mathbf{D}_{22}(t) + \eta\boldsymbol{\Theta}_{22} - (\mathbf{D}_{21}(t) + \eta\boldsymbol{\Theta}_{21})(\mathbf{D}_{11}(t) + \eta\boldsymbol{\Theta}_{11})^{-1}(\mathbf{D}_{12}(t) + \eta\boldsymbol{\Theta}_{12})$, the Schur complement of $\mathbf{D}_{11}(t) + \eta\boldsymbol{\Theta}_{11}$, is positive definite. Therefore, by (9), we obtain $\mathbf{C}_{22}(t+1) \succ 0$. Integrating $\mathbf{C}_{11\cdot 2}(t+1) \succ 0$, we have $\mathbf{C}(t+1) \succ 0$ as long as $\mathbf{C}(0) \succ 0$.

B.4 Proof of Theorem 2

Using the matrix $\mathbf{D}(t)$ defined in (28), we can re-express $Q_1(\mathbf{C}(t+1) \mid \mathbf{C}(t))$ as

$$Q_1(\mathbf{C}(t+1) \mid \mathbf{C}(t)) = (1+\eta)\log|\mathbf{C}(t+1)| - \operatorname{tr}\big(\mathbf{C}(t+1)(\mathbf{D}(t) + \eta\boldsymbol{\Theta})\big),$$

and compute

$$\frac{\partial Q_2}{\partial r} = n_2 \log \frac{2}{r(t)} + \sum_{j=0}^{n_2-1} \Psi\left(\frac{r(t)-n_1-j}{2}\right) + \log|\mathbf{D}(t)| + \eta \log|\mathbf{\Theta}|$$
$$+ n + \frac{(\eta r + n + 1)\eta n}{\eta r} + \eta n \log \frac{\eta r}{\eta r + n + 1} + n \log \frac{r}{2} - \sum_{j=0}^{n-1} \Psi\left(\frac{r-j}{2}\right)$$
$$+ \eta n \log \frac{\eta r + n + 1}{2} - \eta \sum_{j=0}^{n-1} \Psi\left(\frac{\eta r + n + 1 - j}{2}\right).$$

So we have

$$n \log \frac{r}{2} - \sum_{j=0}^{n-1} \Psi\left(\frac{r-j}{2}\right) + \eta n \log \frac{\eta r + n + 1}{2} - \eta \sum_{j=0}^{n-1} \Psi\left(\frac{\eta r + n + 1 - j}{2}\right)$$
$$+ \eta n \frac{n+1}{\eta r} - \eta n \log\left(1 + \frac{n+1}{\eta r}\right) = n_2 \log \frac{r(t)}{2} - \sum_{j=0}^{n_2-1} \Psi\left(\frac{r(t)-n_1-j}{2}\right) \tag{29}$$
$$+ \operatorname{tr}\big(\mathbf{C}(t+1)(\mathbf{D}(t) + \eta\mathbf{\Theta})\big) - \log|\mathbf{D}(t)| - \eta \log|\mathbf{\Theta}| - (1+\eta)\log|\mathbf{C}(t+1)| - (1+\eta)n.$$

It is clear that $\eta n \frac{n+1}{\eta r} - \eta n \log\left(1 + \frac{n+1}{\eta r}\right)$ is a positive decreasing function of r for $r \geq n$. From the Lemma in the Appendix of (Chen, 1979), we also obtain that both $n \log \frac{r}{2} - \sum_{j=0}^{n-1} \Psi(\frac{r-j}{2})$ and $\eta n \log \frac{\eta r+n+1}{2} - \eta \sum_{j=0}^{n-1} \Psi(\frac{\eta r+n+1-j}{2})$ are positive monotonic decreasing functions of r for $r \geq n$. Thus, the left-hand side of (29) is a positive monotonic decreasing function of r for $r \geq n$. Furthermore, as

$$\operatorname{tr}\big(\mathbf{C}(t+1)\mathbf{D}(t)\big) + \eta \operatorname{tr}\big(\mathbf{C}(t+1)\mathbf{\Theta}\big) \geq \log|\mathbf{C}(t+1)\mathbf{D}(t)| + n + \eta \log|\mathbf{C}(t+1)\mathbf{\Theta}| + \eta n,$$

together with $n_2 \log \frac{r(t)}{2} - \sum_{j=0}^{n_2-1} \Psi(\frac{r(t)-n_1-j}{2}) \geq 0$, which is due to $n_2 \log \frac{r(t)}{2} \geq n_2 \log \frac{r(t)-n_1}{2}$, the right-hand side of (29) is always nonnegative. Therefore the solution of (29) is uniquely determined.

Acknowledgments This research has been supported by two Competitive Earmarked Research Grants (CERG), HKUST6174/04E and HKUST6195/02E, from the Research Grants Council of the Hong Kong Special Administrative Region, China.

References

Amari, S. (1995). Information geometry of the EM and *em* algorithms for neural networks. *Neural Networks, 8:9*, 1379–1408.

Baker, R. C. (1973). Joint measures and cross-covariance operators. *Transactions of the American Mathematical Society, 186*, 273–289.

Baudat, G., & Anouar, F. (2000). Generalized discriminant analysis using a kernel approach. *Neural Computation, 12*, 2385–2404.

Bishop, M. C. (1995). *Neural networks for pattern recognition*. Oxford University Press, Oxford.

Bousquet, O., & Herrmann, D. J. L. (2003). On the complexity of learning the kernel matrix. In *Advances in neural information processing systems 15*. Cambridge, MA. MIT Press.
Chapelle, O., Vapnik, V., Bousquet, O., & Mukherjee, S. (2002). Choosing multiple parameters for support vector machines. *Machine Learning, 46:1-3*, 131–159.
Chen, C.-F. (1979). Bayesian inference for a normal dispersion matrix and its application to stochastic multiple regression analysis. *Journal of the Royal Statistical Society Series B, 41:2*, 235–248.
Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*, 273–297.
Crammer, K., Keshet, J., & Singer, Y. (2003). Kernel design using boosting. In *Advances in neural information processing systems 15*, Cambridge, MA, MIT Press.
Cressie, N. A. C. (1991). *Statistics for spatial data*. Wiley, New York.
Cristianini, N., Kandola, J., Elissee, A., & Shawe-Taylor., J. (2002). On kernel target alignment. In T.G. Dietterich, S. Becker, and Z. Ghahramani, (Eds.), *Advances in neural information processing systems 14*. Cambridge, MA, MIT Press.
Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977) Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B, 39:1*, 1–38.
Diggle, P. J., Tawn, J. A., & Moyeed, R. A. (1998) Model-based geostatistics (with discussions). *Applied Statistics. 47:3*, 299–350.
Friedman, J., Hastie, T., & Tibshirani, R. (2000) Additive logistic regression: A statistical view of boosting. *Annals of Statistics, 38:2*,337–374.
Graepel, T. (2002). Kernel matrix completion by semidefinite programming. In *Proceedings of the international conference on artificial neural networks* (pp. 687–693). Madrid, Spain.
Gupta, A. K., & Nagar, D. K. (2000). *Matrix variate distributions*. Chapman & Hall/CRC.
Horn, R. A., & Johnson, C. R. (1985). *Matrix analysis*. Cambridge University Press. Cambridge. UK.
Horn, R. A., & Johnson, C. R. (1991). *Topics in matrix analysis*. Cambridge University Press. Cambridge. UK.
Joachims, T. (1999). Transductive inference for text classification using support vector machines. In *Proceedings of the international conference on machine learning*.
Joachims, T. (2003) Transductive learning via spectral graph partitioning. In *Proceedings of the international conference on machine learning*.
Kandola, J., Shawe-Taylor, J., & Cristianini, N. (2002) On the extensions of kernel alignment. Technical Report 2002–120, NeuroCOLT.
Kandola, J, Shawe-Taylor, J., & Cristianini, N. (2002). Optimizing kernel alignment over combinations of kernels. Technical Report 2002–121, NeuroCOLT.
Kin, T., Kato, T., Tsuda, K., & Asai, K. (1954). Protein classification via kernel matrix completion. *Annals of Mathematical Statistics, 25*, 40–75.
Kwok, J. T. (2000). The evidence framework applied to support vector machines. *IEEE Transactions on Neural Networks. 11:5*, 1162–1173.
Lanckriet, G. R. N., Cristianini, G., Bartlett, P., El Ghaoui, L., & Jordan, M. I. (2004). Learning the kernel matrix with semide.nite programming. *Journal of Machine Learning Research, 5*, 27-72.
Lütkepohl, H. (1996). *Handbook of matrices*. John Wiley & Sons, New York.
Mardia, K. V., & Marshall, R. J. (1984). Maximum likelihood estimation for models of residual covariance in spatial regression. *Biometrika. 71:1*, 135–146.
Neal, R. M. (1999). Regression and classification using Gaussian process priors (with discussion). In J. M. Bernardo, J. O. Berger, A. P. Dawid, & A. F. M. Smith (Eds.), *Bayesian statistics*, volume 6, pages 475–501. Oxford University Press.
Ong, C. S., Smola, A. J., & Williamson, R. C. (2003). Hyperkernels. In *Advances in neural information processing systems 15*. Cambridge, MA, MIT Press.
Schölkopf, B., Smola, A., & Klaus-Robert Müller. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation, 10*, 1299–1319.
Schölkopf B., & Smola, A. J. (2002). *Learning with kernels*. The MIT Press.
Seeger, M. (2000). Relationships between Gaussian processes, support vector machines and smoothing splines. Technical report, Institute for Adaptive and Neural Computation, University of Edinburgh.
Smola, A. J., & Schölkopf, B. (2002). Bayesian kernel methods. In S. Mendelson and A. J. Smola (Eds.), *Advanced lectures on machine learning* (pp. 65–117). Springer.
Smola, A. J., Vishwanathan, S. V. N., & Hoffman, T. (2004) Kernel methods for missing variables. In *NIPS'04 Workshop on Graphical Models and Kernels*.
Sollich, P. (2000) Probabilistic methods for support vector machines. In *Advances in neural information processing systems 12*, pp. 349–355.
Tanner, M. A., & Wong, W. H. (1987) The calculation of posterior distributions by data augmentation (with discussion). *Journal of the American Statistical Association. 82:398*, 528–550.

Tsuda, K., Akaho, S., & Asai, K. (2003). The *em* algorithm for kernel matrix completion with auxiliary data. *Journal of Machine Learning Research, 4*, 67-81.

Vandenberghe, L., & Boyd, S. (1996). Semidefinite programming. *SIAM Review, 38:1*, 49–95.

Vapnik, V. (1998). *Statistical learning theory*. John Wiley and Sons, New York.

Wahba, G. (1990). *Spline models for observational data*. SIAM, Philadelphia.

Williams, C. K. I., & Barber, D. (1998). Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20:12*, 1342–1351.

Zhang, Z. (2003) Learning metrics via discriminant kernels and multidimensional scaling: Toward expected Euclidean representation. In *Proceedings of the 20th international conference on machine learning*.

Zhang, Z., Yeung, D. Y., & Kwok, J. K. (2004). Bayesian inference for transductive learning of kernel matrix using the Tanner-Wong data augmentation algorithm. In *Proceedings of the 21st international conference on machine learning*.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., & Schölkopf, B., (2004). Learning with local and global consistency. In *Advances in neural information processing systems 16*.