Developmental Stage Annotation of Drosophila Gene Expression Pattern Images via an Entire Solution Path for LDA

JIEPING YE and JIANHUI CHEN
Arizona State University
RAVI JANARDAN
University of Minnesota
and
SUDHIR KUMAR
Arizona State University

4

Gene expression in a developing embryo occurs in particular cells (spatial patterns) in a time-specific manner (temporal patterns), which leads to the differentiation of cell fates. Images of a *Drosophila melanogaster* embryo at a given developmental stage, showing a particular gene expression pattern revealed by a gene-specific probe, can be compared for spatial overlaps. The comparison is fundamentally important to formulating and testing gene interaction hypotheses. Expression pattern comparison is most biologically meaningful when images from a similar time point (developmental stage) are compared. In this paper, we present LdaPath, a novel formulation of Linear Discriminant Analysis (LDA) for automatic developmental stage range classification. It employs multivariate linear regression with the L_1-norm penalty controlled by a regularization parameter for feature extraction and visualization. LdaPath computes an entire solution path for all values of regularization parameter with essentially the same computational cost as fitting one LDA model. Thus, it facilitates efficient model selection. It is based on the equivalence relationship between LDA and the least squares method for multiclass classifications. This equivalence relationship is established under a mild condition, which we show empirically to hold for many high-dimensional datasets, such as expression pattern images. Our experiments on a collection of 2705 expression pattern images show the effectiveness of the proposed algorithm. Results also show that

This research is supported in part by funds from the Arizona State University, the National Science Foundation (NSF) under Grant No. IIS-0612069, and the National Institutes of Health (NIH) under Grant No. HG002516.
Authors' addresses: J. Ye, J. Chen, Center for Evolutionary Functional Genomics and Department of Computer Science and Engineering, Arizona State University, Tempe, AZ 85287; email: {jieping.ye,jianhui.chen}@asu.edu; R. Janardan, Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455; email: janardan@cs.umn.edu; S. Kumar, Center for Evolutionary Functional Genomics and School of Life Sciences, Arizona State University, Tempe, AZ 85287; email: s.kumar@asu.edu.

the LDA model resulting from LdaPath is sparse, and irrelevant features may be removed. Thus, LdaPath provides a general framework for simultaneous feature selection and feature extraction.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications—*Data mining*

General Terms: Algorithms

Additional Key Words and Phrases: Gene expression pattern image, dimensionality reduction, linear discriminant analysis, linear regression

ACM Reference Format:

Ye, J., Chen, J. Janardan, R., and Kumar, S. 2008. Developmental stage annotation of drosophila gene expression pattern images via an entire solution path for LDA. ACM Trans. Knowl. Discov. Data. 2, 1, Article 4 (March 2008), 21 pages. DOI = 10.1145/1342320.1342324 http://doi.acm.org/10.1145/1342320.1342324

1. INTRODUCTION

Understanding the roles of genes and their interactions is one of the central themes of genome research. One popular approach, based on the analysis of microarray gene expression data [Golub et al. 1999; Gurunathan et al. 2004], often does not capture spatial patterns of expression. In contrast, the classic genetic analysis of spatial patterns of gene expression relies on the direct visualization of the presence or absence of gene products (mRNA or protein). Recent advances in the *in situ* hybridization technique allow us to localize specific mRNA sequences in morphologically preserved tissues/cells by hybridizing the complementary strand of a nucleotide probe to the sequence of interest. Large numbers of images of a *Drosophila melanogaster* embryo at a given developmental stage, showing a particular gene expression pattern revealed by a gene-specific probe, are now available [Tomancak et al. 2002]. It is thus possible to study and understand the interplay of genes in different stages of development through the examination of the spatial overlap of patterns of gene expression [Carroll et al. 2005; Gurunathan et al. 2004; Kumar et al. 2002; Peng and Myers 2004].

Estimation of the pattern overlap is most biologically meaningful when images from a similar time point (developmental stage) are compared. Stages in *Drosophila melanogaster* development denote the time after fertilization at which certain specific events occur in the developmental cycle. Embryogenesis is traditionally divided into a series of consecutive stages distinguished by morphological markers [Bownes 1975] (see Figure 1). The duration of developmental stages varies from 15 minutes to more than 2 hours; therefore, the stages of development are differentially represented in the embryo collections. The first 16 stages of embryogenesis are divided into six stage ranges (stages 1–3, 4–6, 7–8, 9–10, 11–12, and 13–16). We are interested in how image analysis can be used for automatic stage range annotation (classification). In recent high-throughput experiments [Tomancak et al. 2002], each image is assigned to one of the stage ranges manually.

It has been observed that across the various developmental stages, image textural properties at a subblock level are a distinguishing feature, because image texture at the subblock level changes as embryonic development progresses [Ye et al. 2006] (Figure 1). Gabor filters [Daugman 1988] were thus applied to

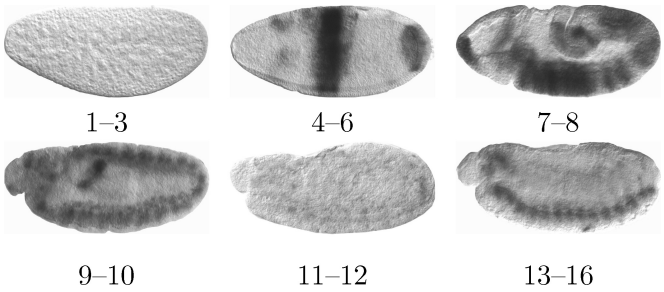


Fig. 1. Spatial and temporal view of *Drosophila* images across different stages (1–16) of development (of the same gene Kr). The textural features (based on the morphology of the embryo) are different from the gene expression, which is indicted by the blue staining.

extract the textural features of image subblocks. Since not all features were useful for stage range discrimination, Regularized Linear Discriminant Analysis (RLDA) [Guo et al. 2003; Hastie et al. 2001] was applied for the extraction of the most discriminant features, which are linear combinations of the textural features derived from the Gabor filters.

Linear Discriminant Analysis (LDA) is a well-known method for feature extraction and visualization that projects high-dimensional data onto a low-dimensional space where the data achieves maximum class separability [Duda et al. 2000; Fukunaga 1990; Hastie et al. 2001]. It has been shown to be particularly effective for applications involving image data, such as face recognition [Belhumeour et al. 1997]. However, features extracted via LDA are linear combinations of the original set of features, and the coefficients learned from LDA algorithms, such as RLDA, are typically nonzero. This often makes it difficult to interpret the derived features. It is therefore desirable to derive a sparse formulation of LDA, which contains only a small number of nonzero coefficients. Sparsity often leads to easy interpretation and a good generalization ability. It has been used successfully in several well-known algorithms, such as Principal Component Analysis [d'Aspremont et al. 2004] and SVM [Zhu et al. 2003]. Imposing the sparsity constraint in LDA is, however, challenging, as it involves a generalized eigenvalue problem [Duda et al. 2000; Fukunaga 1990; Hastie et al. 2001]. Furthermore, the RLDA algorithm [Ye et al. 2006] involves a regularization parameter, which is commonly estimated via cross-validation from a given candidate set. This parameter selection process is computationally expensive, especially when the size of the regularization candidate set is large.

In this paper, we present LdaPath for automatic developmental stage range classification. LdaPath overcomes the limitations of the RLDA algorithm, while maintaining competitive classification performance. LdaPath employs multivariate linear regression for feature extraction and visualization. It is based on the equivalence relationship between LDA and the least squares method for multiclass classifications. LDA in the binary-class case has been shown to be equivalent to linear regression with the class label as the output [Duda et al. 2000; Mika 2002]. This implies that LDA for binary-class classifications can be formulated as a least squares problem, which, however, does not extend to the multiclass case. We show in this paper that this equivalence relationship can

be established for the multiclass case under a mild condition, which we show empirically to hold for many high-dimensional datasets, such as expression pattern images.

In mathematical programming, it is known that sparsity can often be achieved by constraining or penalizing the L_1-norm of the variables [Donoho 2006; Tibshirani 1996]. By casting LDA as a least squares problem, we can achieve the sparsity in LDA by employing multivariate linear regression with the L_1-norm penalty controlled by a regularization parameter, which leads to simultaneous feature selection and feature extraction in LdaPath. Furthermore, following the least squares formulation of LDA, we employ the Least Angle Regression algorithm (LARS) in Efron et al. [2004] to compute the entire solution path for LdaPath. The SvmPath algorithm developed in Hastie et al. [2004] has the same flavor as LdaPath in that it computes the entire solution path for SVM to speed up the model selection process. The key features of LdaPath are summarized as follows:

(1) LdaPath enforces the sparsity constraint in the formulation, which enhances the biological interpretability of the resulting model; and
(2) LdaPath computes the entire solution path for all values of regularization parameter, with essentially the same computational cost as fitting one LDA model. Thus, it facilitates efficient model selection.

Experiments on a collection of 2705 expression pattern images from early stages show that LdaPath is competitive with several other LDA-based dimensionality reduction algorithms in classification, while its resulting LDA model is much more sparse. Moreover, the sparsity constraint in LdaPath enhances the interpretability of the resulting model, which is critical for biologists.

A preliminary version of this article, which shows the equivalence relationship between LDA and linear regression in the multi-class case, appears in the Proceedings of the Twenty-Fourth International Conference on Machine Learning, 2007. This submission is substantially extended and contains: (1) the LdaPath algorithm; and (2) extensive experiments on the application of LdaPath to gene expression pattern image annotation.

2. METHODS

Expression pattern images used in this paper were collected from the Berkeley *Drosophila* Genome project (BDGP) [Tomancak et al. 2002]. Since images from BDGP were in different sizes and orientations, the image standardization procedure in Kumar et al. [2002] was applied, and all images were standardized to a size of 128×320. Gabor filters were then used to extract the textural features from the images, which were used to generate the feature vectors of size $d = 1280$ via a linear transformation (see Section 3.1 for details). Our dataset consists of $n = 2705$ gene expression pattern images represented as $\{(x_i, y_i)\}_{i=1}^{n}$ from early stages (1–8), where $x_i \in \mathbb{R}^d$ is the feature vector of the i-th image, and $y_i \in \{1, 2, \ldots, k\}$ ($k = 3$) denotes the class label of the i-th image. Let $X_i \in \mathbb{R}^{d \times n_i}$ be the data matrix of the i-th class, where n_i is the size of X_i and $\sum_{i=1}^{k} n_i = n$.

2.1 An Overview of Linear Discriminant Analysis

LDA is a well-known method for feature extraction and visualization. It projects high-dimensional data onto a low-dimensional space where the data achieves maximum class separability [Duda et al. 2000; Fukunaga 1990; Hastie et al. 2001]. LDA computes a linear transformation $G \in \mathbb{R}^{d \times d'}$ that maps x_i in the d-dimensional space to a vector in the d'-dimensional space as follows:

$$x_i \in \mathbb{R}^d \rightarrow G^T x_i \in \mathbb{R}^{d'} \ (d' < d).$$

In discriminant analysis [Fukunaga 1990], three scatter matrices (called *within-class*, *between-class* and *total* scatter matrices) are defined as follows:

$$S_w = \frac{1}{n} \sum_{i=1}^{k} \sum_{x \in X_i} \left(x - c^{(i)}\right)\left(x - c^{(i)}\right)^T, \tag{1}$$

$$S_b = \frac{1}{n} \sum_{i=1}^{k} n_i \left(c^{(i)} - c\right)\left(c^{(i)} - c\right)^T, \tag{2}$$

$$S_t = \frac{1}{n} \sum_{j=1}^{n} (x_j - c)(x_j - c)^T, \tag{3}$$

where $c^{(i)}$ is the *centroid* of the i-th class, and c is the *global centroid*. It follows from the definition that $S_t = S_b + S_w$. The *traces* [Golub and Van Loan 1996] of S_w and S_b, i.e., trace(S_w) and trace(S_b) measure the within-class cohesion and the between-class separation, respectively. In the lower-dimensional space resulting from the linear transformation, the scatter matrices become $G^T S_w G$, $G^T S_b G$, and $G^T S_t G$, respectively. An optimal transformation G maximizes trace($G^T S_b G$) and minimizes trace($G^T S_w G$) simultaneously, which is equivalent to maximizing trace($G^T S_b G$) and minimizing trace($G^T S_t G$) simultaneously, since $S_t = S_b + S_w$. The optimal transformation G^* is computed by solving the following optimization problem [Duda et al. 2000; Fukunaga 1990]:

$$G^* = \arg\max_G \{\text{trace}((G^T S_t G)^{-1} G^T S_b G)\}. \tag{4}$$

It can be shown that G^* consists of the top eigenvectors of $S_t^{-1} S_b$ corresponding to the nonzero eigenvalues [Fukunaga 1990], provided that S_t is nonsingular. Note that the traditional formulation of LDA in the multi-class case can not be solved by least-squares methods. In addition, the traditional LDA formulation fails when S_t is singular, which is the case for our expression pattern image data.

The Uncorrelated LDA algorithm (ULDA) [Ye 2005] is an extension of classical LDA for singular scatter matrices. The optimal transformation G^U of ULDA is given by maximizing the following objective function: trace $\left((G^T S_t G)^+ G^T S_b G\right)$, where M^+ denotes the pseudo-inverse of M [Golub and Van Loan 1996]. It has been shown that G^U is given by the top eigenvectors of $S_t^+ S_b$ [Ye 2005]. A key property of ULDA is that the features in the transformed space of ULDA are uncorrelated to each other, thus reducing the redundancy in the transformed (dimension reduced) space.

2.2 Multivariate Linear Regression with a Class Indicator Matrix

The LDA formulation in Section 2.1 is an extension of the original Fisher Linear Discriminant Analysis (FLDA) [Fisher 1936], which deals with binary-class problems. It has been shown [Duda et al. 2000; Mika 2002] that FLDA is equivalent to a least squares problem. Recall that gene expression pattern images are from multiple stage ranges (there are $k = 3$ classes in our image dataset), while the equivalence relationship between LDA and the least squares method does not extend to the multiclass case [Duda et al. 2000; Hastie et al. 2001; Zhang and Riedel 2005]. In this section, we study the least squares formulation of LDA in the multiclass case, which forms the basis for the LdaPath algorithm to be presented in the next section.

In multiclass classifications, it is common to apply linear regression of a class membership indicator matrix $Y \in \mathbb{R}^{n\times k}$, which applies a vector-valued class code for each of the samples [Hastie et al. 2001]. There are several well-known indicator matrices in the literature. Denote $Y_1 = (Y_1(ij))_{ij} \in \mathbb{R}^{n\times k}$ and $Y_2 = (Y_2(ij))_{ij} \in \mathbb{R}^{n\times k}$ as follows:

$$Y_1(ij) = \begin{cases} 1 & \text{if } y_i = j, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

$$Y_2(ij) = \begin{cases} 1 & \text{if } y_i = j, \\ -1/(k-1) & \text{otherwise.} \end{cases} \tag{6}$$

The first indicator matrix Y_1 is commonly used in connecting multiclass classification with linear regression [Hastie et al. 2001], while the second indicator matrix has recently been used in extending SVM to multiclass classifications [Lee et al. 2004]. A more general class indicator matrix has been studied in Park and Park [2005].

In multivariate linear regression (MLR), a k-tuple of separating functions

$$f(x) = \big(f_1(x), f_2(x), \ldots, f_k(x)\big), \tag{7}$$

for any $x \in \mathbb{R}^d$ is considered. Denote $\tilde{X} = [\tilde{x}_1, \ldots, \tilde{x}_n] \in \mathbb{R}^{d\times n}$, and $\tilde{Y} = (\tilde{Y}_{ij}) \in \mathbb{R}^{n\times k}$ as the centered data matrix X and the centered indicator matrix Y, respectively. That is, $\tilde{x}_i = x_i - \bar{x}$ and $\tilde{Y}_{ij} = Y_{ij} - \bar{Y}_j$, where $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ and $\bar{Y}_j = \frac{1}{n}\sum_{i=1}^{n} Y_{ij}$. Then, MLR determines the weight vectors, $\{w_j\}_{j=1}^{k} \in \mathbb{R}^d$, of the k linear models, $f_j(x) = x^T w_j$, for $j = 1, \ldots, k$, via the minimization of the following objective function:

$$L(W) = \frac{1}{2}\|\tilde{X}^T W - \tilde{Y}\|_F^2 = \frac{1}{2}\sum_{j=1}^{k}\sum_{i=1}^{n} \| f_j(\tilde{x}_i) - \tilde{Y}_{ij}\|^2, \tag{8}$$

where $W = [w_1, w_2, \ldots, w_k]$ is the weight matrix, and $\|\cdot\|_F$ denotes the *Frobenius* norm of a matrix [Golub and Van Loan 1996]. The optimal W is given by [Hastie et al. 2001]

$$W = \big(\tilde{X}\tilde{X}^T\big)^{+}\tilde{X}\tilde{Y}, \tag{9}$$

which is determined by $\tilde{X}$ and $\tilde{Y}$. Both Y_1 and Y_2 defined above, as well as the one in Park and Park [2005], could be used to define the centered indicator

matrix $\tilde{Y}$. However, the resulting linear regression models using these indicator matrices are not, in general, equivalent to LDA. A natural question is whether there exists a class indicator matrix $\tilde{Y} \in \mathbb{R}^{n\times k}$, with which the multivariate linear regression is equivalent to LDA. If this is the case, then LDA can essentially be formulated as a least squares problem in the multiclass case.

Note that in multivariate linear regression, each $\tilde{x}_i$ is transformed to

$$(f_1(\tilde{x}_i), \ldots, f_k(\tilde{x}_i))^T = W^T \tilde{x}_i,$$

and the centered data matrix $\tilde{X} \in \mathbb{R}^{d\times n}$ is transformed to $W^T\tilde{X} \in \mathbb{R}^{k\times n}$, thus achieving dimensionality reduction if $k < d$. In the following, we construct a specific class indicator matrix Y_3. We show in Section 2.3 the equivalence relationship between multivariate linear regression using indicator matrix Y_3 and LDA. The indicator matrix $Y_3 = (Y_3(ij))_{ij} \in \mathbb{R}^{n\times k}$ is constructed as follows:

$$Y_3(ij) = \begin{cases} \sqrt{\frac{n}{n_j}} - \sqrt{\frac{n_j}{n}} & \text{if } y_i = j, \\ -\sqrt{\frac{n_j}{n}} & \text{otherwise,} \end{cases} \tag{10}$$

where n_j is the sample size of the j-th class, and n is the total sample size. It can be shown that Y_3 defined above has been centered (in terms of rows), and thus $\tilde{Y}_3 = Y_3$.

2.3 The LdaPath Algorithm

Recall from Section 2.1 that the optimal transformation matrix G^U of ULDA consists of the top eigenvectors of $S_t^+S_b$ corresponding to the nonzero eigenvalues. With $\tilde{Y} = Y_3$ chosen as the class indicator matrix, the optimal weight matrix W_3 for multivariate linear regression in Equation (9) becomes

$$W_3 = \left(\tilde{X}\tilde{X}^T\right)^+ \tilde{X}\tilde{Y} = (nS_t)^+ nH_b = S_t^+ H_b. \tag{11}$$

We will first study the relationship between W_3 in Equation (11) and the eigenvectors of $S_t^+S_b$. It is based on the decomposition of the scatter matrices as follows.

Define matrices H_w, H_b, and H_t as follows:

$$H_w = \frac{1}{\sqrt{n}}\left[X_1 - c^{(1)}\left(e^{(1)}\right)^T, \ldots, X_k - c^{(k)}\left(e^{(k)}\right)^T\right], \tag{12}$$

$$H_b = \frac{1}{\sqrt{n}}\left[\sqrt{n_1}\left(c^{(1)} - c\right), \ldots, \sqrt{n_k}\left(c^{(k)} - c\right)\right], \tag{13}$$

$$H_t = \frac{1}{\sqrt{n}}(X - ce^T), \tag{14}$$

where X_i is the data matrix of the i-th class, X is the data matrix, $e^{(i)}$ is the vector of all ones of length n_i, and e is the vector of all ones of length n. Then S_w, S_b, and S_t can be expressed as follows: $S_w = H_wH_w^T$, $S_b = H_bH_b^T$, and $S_t = H_tH_t^T$. It is easy to check that $\tilde{X}Y_3 = nH_b$. Let $H_t = U\Sigma V^T$ be the Singular Value Decomposition (SVD) [Golub and Van Loan 1996] of H_t, where H_t is defined in Equation (14), U and V are orthogonal, $\Sigma = \mathrm{diag}(\Sigma_t, 0)$,

$\Sigma_t \in \mathbb{R}^{t\times t}$ is diagonal, and $t = \text{rank}(S_t)$. Then

$$S_t = H_t H_t^T = U\Sigma\Sigma^T U^T = U\,\text{diag}(\Sigma_t^2, 0)U^T. \tag{15}$$

Let $U = (U_1, U_2)$ be a partition of U, such that $U_1 \in \mathbb{R}^{d\times t}$ and $U_2 \in \mathbb{R}^{d\times(d-t)}$. Since $S_t = S_b + S_w$, we have

$$U^T S_b U = \text{diag}(U_1^T S_b U_1, 0). \tag{16}$$

Let

$$B = \Sigma_t^{-1} U_1^T H_b \in \mathbb{R}^{t\times k}, \tag{17}$$

where H_b is defined as in Equation (13) and let

$$B = P\hat{\Sigma}Q^T \tag{18}$$

be the SVD of B, where P and Q are orthogonal and $\hat{\Sigma} \in \mathbb{R}^{t\times k}$ is diagonal. Since $S_b = H_b H_b^T$, we have

$$\Sigma_t^{-1} U_1^T S_b U_1 \Sigma_t^{-1} = BB^T = P\Sigma_b P^T, \tag{19}$$

where

$$\Sigma_b = \hat{\Sigma}\hat{\Sigma}^T = \text{diag}(\alpha_1^2, \ldots, \alpha_t^2), \tag{20}$$

$$\alpha_1^2 \geq \cdots \geq \alpha_q^2 > 0 = \alpha_{q+1}^2 = \cdots = \alpha_t^2, \tag{21}$$

and $q = \text{rank}(S_b)$.

The relationship between ULDA and multivariate linear regression is summarized as follows: (see Appendix A for a detailed proof)

LEMMA 2.1. *Let $W_3 = S_t^+ H_b$ be defined as in Equation* (11)*, and let G^U be the optimal transformation matrix of ULDA, which consists of the top eigenvectors of $S_t^+ S_b$. Then $W_3 = [G^U \Sigma_{bq}^{0.5}, 0]Q^T$, where $\Sigma_{bq} \in \mathbb{R}^{q\times q}$ consists of the first q rows and the first q columns of Σ_b defined in Equation* (20)*, $q = rank(S_b)$, and Q defined in Equation* (18) *is orthogonal.*

The Nearest-Neighbor (NN) algorithm [Duda et al. 2000] based on the Euclidean distance is commonly applied as the classifier in the dimensionality reduced space of LDA. If the weight matrix W_3 is applied for dimensionality reduction before NN, it is invariant of an orthogonal transformation, since any orthogonal transformation preserves all pairwise distances. From Lemma 2.1, W_3 is essentially equivalent to $[G^U \Sigma_{bq}^{0.5}, 0]$ or $G^U \Sigma_{bq}^{0.5}$, as the removal of zero columns does not change the pairwise distance, either. The essential difference between W_3 and G^U is therefore the diagonal matrix $\Sigma_{bq}^{0.5}$.

Next, we show that matrix Σ_{bq} is an identity matrix of size q, that is, W_3 and G^U are essentially equivalent, under a mild condition C1: $\text{rank}(S_t) = \text{rank}(S_b) + \text{rank}(S_w)$, which has been shown to hold in many applications involving high-dimensional data [Ye and Xiong 2006]. This is also the case for our expression pattern image data. It can be shown [Ye and Xiong 2006] that if the data points in the training set are linearly independent, then condition C1 holds. The main result of this section is summarized in the following theorem: (see Appendix B for a detailed proof)

THEOREM 2.1. *Let $\Sigma_{bq} \in \mathbb{R}^{q\times q}$ consist of the first q rows and the first q columns of Σ_b, where Σ_b is defined in Equation* (20)*. Assume condition C1: $rank(S_t) = rank(S_b) + rank(S_w)$ holds. Then $\Sigma_{bq} = I_q$, where I_q is the identity matrix of size q.*

Theorem 2.1 implies that under condition C1, W_3 is equivalent to G^U, that is, multivariate linear regression with Y_3 as the class indicator matrix is equivalent to ULDA. Thus, ULDA, an extension of classical LDA for singular scatter matrices, can be formulated as a least squares problem.

Following the equivalence relationship established above, we next develop a sparse formulation of LDA. Sparsity has recently received much attention to improve model interpretability and generalization ability. It is known that sparsity can often be achieved by constraining or penalizing the L_1-norm of the variables [Donoho 2006; Tibshirani 1996; Zhu et al. 2003]. Based on the equivalence relationship between LDA and linear regression for the multiclass problems established above, we propose to develop sparse LDA by minimizing the following Lasso-type objective function [Tibshirani 1996]:

$$L_1(W,\gamma) = \frac{1}{2}\sum_{j=1}^{k}\left(\sum_{i=1}^{n}\left\|\tilde{x}_i^T w_j - \tilde{Y}_{ij}\right\|_2^2 + \gamma\|w_j\|_1\right) \tag{22}$$

where $\|w_j\|_1 = \sum_{i=1}^{d}|w_{ji}|$ denotes the 1-norm [Golub and Van Loan 1996] of w_j, and $\gamma > 0$ is a penalty (regularization) parameter. The optimal w_j^*, for $1 \le j \le k$, is given by

$$w_j^* = \arg\min_{w_j}\left(\sum_{i=1}^{n}\left(\tilde{x}_i^T w_j - \tilde{Y}_{ij}\right)^2 + \gamma\|w_j\|_1\right), \tag{23}$$

which can be reformulated as:

$$w_j^* = \arg\min_{w_j:\|w_j\|_1\le\tau}\sum_{i=1}^{n}\left(\tilde{x}_i^T w_j - \tilde{Y}_{ij}\right)^2, \tag{24}$$

for some tuning parameter $\tau > 0$ [Tibshirani 1996]. The optimal w_j^* in Equation (24) can be readily computed by applying the Least Angle Regression algorithm (LARS) in Efron et al. [2004]. One key feature of LARS is that it computes the entire solution path for all values of τ, with essentially the same computational cost as fitting one linear regression model. We thus call the proposed algorithm LdaPath.

Note that when τ is large enough, the constraints in Equation (24) are not effective, which leads to the following unconstrained optimization problem: $\tilde{w}_j^* = \arg\min_{\tilde{w}_j}\sum_{i=1}^{n}(\tilde{x}_i^T\tilde{w}_j - \tilde{Y}_{ij})^2$. Denote $T = \max\{\|\tilde{w}_1^*\|_1, \ldots, \|\tilde{w}_k^*\|_1\}$, which defines an upper bound for τ, that is, $0 \le \tau \le T$. Define $s = \tau/T$. It follows that $\tau = Ts$, for $0 \le s \le 1$. The estimation of τ is equivalent to the estimation of s, called the "Lasso coefficient" in the following discussion. Estimation of the Lasso coefficient s is the key to the performance of LdaPath. Cross-validation is commonly used to estimate the optimal value from a large candidate set $S = \{s_1, s_2, \ldots, s_p\}$, where $p = |S|$ is the size of S. An important property of the Lasso constraint is that making s sufficiently small will shrink some of the

coefficients of the weight vectors to be exactly zero, which results in a sparse model. We show in the following proposition that a small value of s is also necessary to overcome the over-fitting problem: (see Appendix C for a detailed proof)

PROPOSITION 2.1. *Let $W^* = [w_1^*, \ldots, w_k^*]$ be the weight matrix of LdaPath using Lasso coefficient $s = 1$, and let x be a data point from the i-th class. Assume condition C1 holds. Then, $(W^*)^T x = (W^*)^T c^{(i)}$, where $c^{(i)}$ is the centroid of the i-th class. That is, all data points from the i-th class are mapped to a common vector $(W^*)^T c^{(i)}$.*

Proposition 2.1 above shows that under condition C1, LdaPath with $s = 1$ maps all points from the same class to a common point. This leads to a perfect separation between different classes; however, this may also lead to over-fitting. A small value of s is therefore desirable in order to alleviate this problem, provided that a good Lasso coefficient can be estimated. In the following experimental studies, we will examine these issues in detail.

3. EXPERIMENTS

In this section, we apply LdaPath for *Drosophila* embryonic developmental stage range classification. A collection of 2705 expression pattern images from early stages was used in the experiments.

3.1 Data Preprocessing

We worked with expression pattern images in different sizes and orientations from BDGP [Tomancak et al. 2002]. The image standardization procedure in Kumar et al. [2002] was applied, and all images were standardized to a size of 128×320. Then, Gabor filters were used to extract the texture information from the images. The main steps for the data pre-processing include:

—Step 1: The image standardization procedure in Kumar et al. [2002] was employed, and all images were standardized to a size of 128×320. *Histogram Equalization* [Gonzalez and Woods 1993] was applied to improve the contrast and to obtain an approximately uniform histogram distribution, while the detailed information of the processed images was retained.

—Step 2: Each image was divided into 640 subblocks of size 8×8. Log Gabor Filters were applied on each of the subblocks to extract the texture features [Daugman 1988]. Gabor filters are the product of a complex sinusoidal function and a Gaussian-shaped function. We used Log Gabor filters with 4 different wavelet scales and 6 different filter orientations to extract the texture information. Hence, 24 Gabor images were obtained from the filtering operation. Note that all 24 Gabor images have the same size (i.e., 128×320) as the original one.

—Step 3: For each of the Gabor images, the mean value was used to represent each of the subblocks. The averaged Gabor images were of size 16×40, and were reshaped to form feature vectors of length 640. By concatenating all

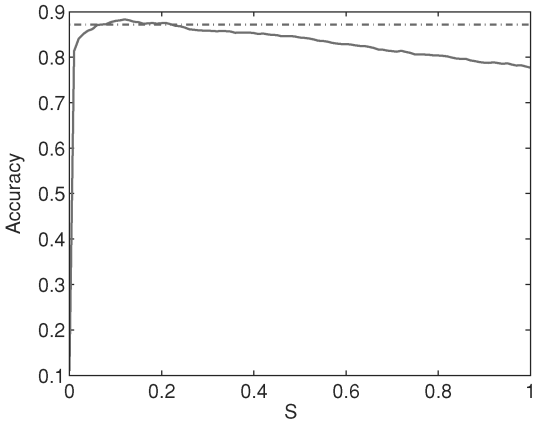


Fig. 2. Effectiveness of cross-validation on Lasso coefficient estimation. The test accuracy (corresponding to the dashdot horizontal line) using the Lasso coefficient estimated from cross-validation is close to the maximum possible accuracy when all possible Lasso coefficients are considered (shown as a solid curve).

of the 24 feature vectors together, we obtain a new feature matrix of size 24×640 for each of the embryo images.

—Step 4: All 24×640 feature matrices were projected to a lower dimensional space via a linear transformation so that the data dimensionality was further reduced. To achieve maximum separability after the projection, the transformation was computed based on a variant of LDA. With the projection, each of the feature matrices was reduced to a size of 2×640, which were reshaped to feature vectors of length 1280. Note that the entries in these feature vectors correspond to specific subblocks (regions) of the images (there are 640 subblocks in total). Each subblock of the images corresponds to 2 (instead of 24) entries in the feature vector. More details on the computation of the projection can be found in Appendix D.

3.2 Estimation of the Lasso Coefficient s

In this experiment, we investigated the estimation of the Lasso coefficient, denoted as s. We applied K-fold cross-validation ($K = 5$) for the estimation. Recall from Section 2.3 that the Lasso coefficient lies in the range [0, 1]. We chose a candidate set $S = \{s_1, \ldots, s_p\}$, with $p = 1000$, for the Lasso coefficient with $s_i = (i-1)/(p-1)$, for $i = 1, \ldots, p$. We ran the experiment using 300 samples in the training set, and the rest in the test set. The experimental result is presented in Figure 2, where the accuracy corresponding to the dashdot horizontal line denotes the test accuracy using the optimal Lasso coefficient estimated via cross-validation. To examine the effectiveness of the cross-validation estimation, we also plotted the test accuracies for all Lasso coefficients (shown as a solid curve). We can observe from Figure 2 that the test accuracy using the Lasso coefficient estimated from cross-validation is close to the maximum possible accuracy when all possible Lasso coefficients are considered. Note that it is possible to use a large set S in cross-validation, due to the efficient model

Table I. Comparison of Three LDA-Based Dimensionality Reduction Algorithms on Mean Classification Accuracy (%) and Standard Deviation (in parenthesis) using Different Numbers of Training Sample Size

	training sample size				
algorithm	90	180	300	480	540
RLDA	82.71	85.74	86.60	87.48	87.24
	(3.09)	(1.82)	(1.18)	(0.99)	(1.26)
ULDA	80.68	77.22	66.30	68.29	73.80
	(2.11)	(2.62)	(2.67)	(2.19)	(2.01)
LdaPath	82.23	85.36	87.14	87.72	87.82
	(2.14)	(1.67)	(0.90)	(0.74)	(0.87)

selection procedure in LdaPath. The experimental result confirms the effectiveness of cross-validation in estimating the optimal Lasso coefficient from a given candidate set.

3.3 Classification Performance

In this experiment, we compared LdaPath with two other LDA-based dimensionality reduction algorithms including Regularized LDA (RLDA) and Uncorrelated LDA (ULDA). We performed our comparative study by repeating random splitting of the whole dataset into training and test sets. The whole dataset was randomly partitioned into a training set consisting of n samples ($n = 90, 180, 300, 480$, and 540) and a test set consisting of the rest of the samples. To reduce the variability, the splitting was repeated 20 times, and the mean accuracy was reported. For LdaPath, the classification performance depends on the choice of the Lasso coefficient s. We chose the best s from a given candidate set via 5-fold cross-validation as in Section 3.2. We observed that condition C1 held in all cases, and LdaPath achieved the same classification performance as ULDA when the Lasso coefficient s is set to 1. This confirms the theoretical results in Section 2.3.

The classification result is summarized in Table I. We can observe from the table that LdaPath is competitive with RLDA in terms of classification accuracy, while they both perform much better than ULDA. Note that Nearest-Neighbor is employed as the classifier in LdaPath. The classification accuracy for LdaPath will be slightly higher, if other more sophisticated classifiers such as Support Vector Machines (SVM) are used instead. However, the difference is not statistically significant. The transformation matrices in RLDA and ULDA are typically quite dense (very small number of zeros). The main advantage of LdaPath over both RLDA and ULDA lies in the sparseness of the resulting model, which will be studied in the next experiment below.

3.4 Sparseness in LdaPath

In this experiment, we investigated the sparseness of the LDA model in LdaPath. Recall that the sparseness of the weight vectors w_i, for $i = 1, \ldots, k$, depends on the Lasso coefficient s. It has been observed [Tibshirani 1996] that a small value of s shrinks many coefficients to be exactly zero. Figure 2 shows that

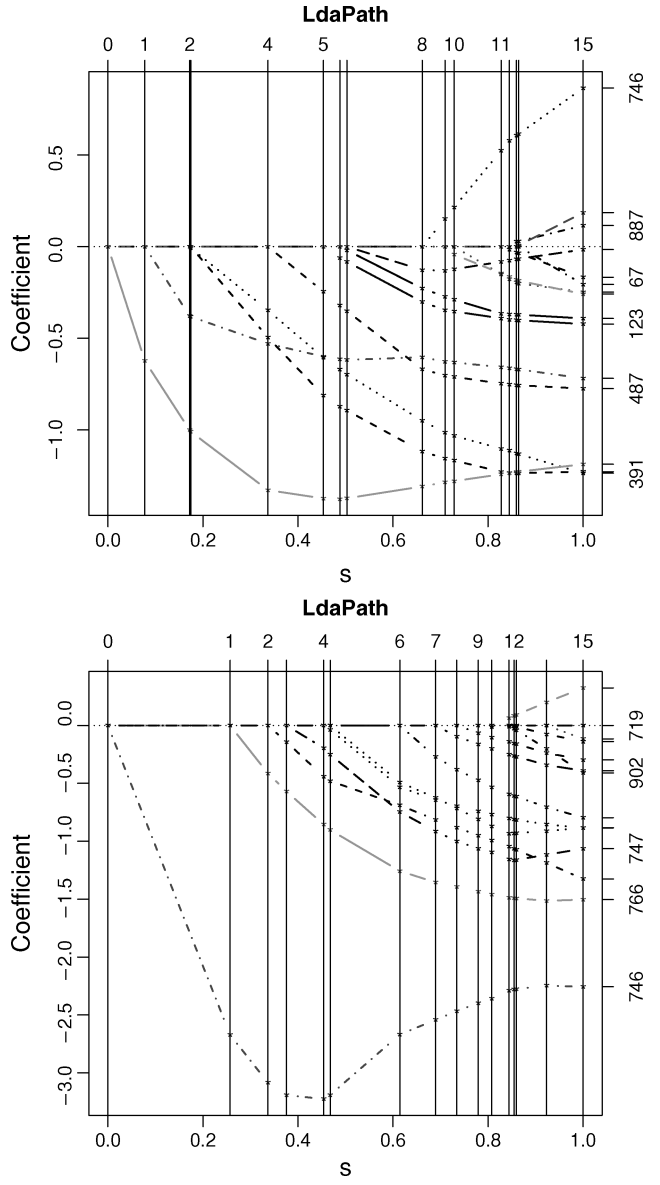


Fig. 3. The entire collection of solution paths for a subset of the coefficients from w_2 (top graph) and w_3 (bottom graph). The x-axis denotes the Lasso coefficient s, and the y-axis denotes the value of the coefficients.

the optimal Lasso coefficient estimated from cross-validation is very small. We therefore expect the corresponding weight vectors to be sparse.

Figure 3 shows the entire collection of solution paths for a subset of the coefficients from two weight vectors, w_2 and w_3. We have observed the same trend in w_1 (results omitted due to space constraint). The x-axis denotes the Lasso coefficient s, and the y-axis denotes the value of the coefficients. The

Table II. Percentage of the Nonzero Entries in the Weight Vectors, w_1, w_2, and w_3 of LdaPath

	training sample size					
vector	180	300	480	540	750	900
w_1	6.79	8.52	16.48	19.29	21.09	16.40
w_2	7.58	9.84	18.59	22.03	25.31	22.81
w_3	6.33	9.92	17.58	21.95	23.36	19.77

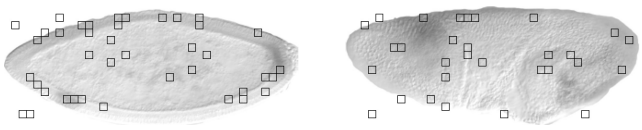


Fig. 4. Plot of discriminant features corresponding to stage range 4–6 (left graph) and stage range 7–8 (right graph), where rectangular blocks indicate the regions on the images with nonzero coefficients.

vertical lines denote (a subset of) the turning point of the path, as the solution path for each of the coefficients is piecewise linear [Efron et al. 2004]. We can observe from Figure 3 that when $s = 1$, most of the coefficients are nonzero, that is, the model is dense. When the value of the Lasso coefficient s decreases (from the right to the left side along the x-axis), more and more coefficients from both w_2 and w_3 become exactly zero. All coefficients become zero when $s = 0$.

Table II shows the percentage of the nonzero entries in the three weight vectors for various training sample sizes, when the optimal Lasso coefficient estimated from cross-validation is used. We can observe that when the sample size is small, about 90% of the coefficients in the weight vectors are zero, while for a larger sample size, about 80% of the coefficients are zero. Together with the classification result in Section 3.3, we conclude that LdaPath uses a much smaller number of features than RLDA, while it is competitive with RLDA in classification. Consequently, it is expected that the features eliminated in LdaPath may be irrelevant for classification, which will be examined in the next experiment.

3.5 Biological Interpretability of LdaPath

In this experiment, we examined the discriminant features detected by LdaPath. Our experimental results in Table II have shown that the weight vectors in LdaPath are sparse. Sparsity may enhance the biological interpretability of the model resulting from LdaPath. In expression pattern image data, each feature corresponds to a specific region in the image. In this case, the discriminant features for each class correspond to the nonzero entries, and may carry biological significance. We found from our experiment (shown in Figure 4) that for images from stage range 4–6, many regions associated with the nonzero coefficients locate along the boundary of the embryo, especially near the two ending parts, while in comparison, more regions associated with the nonzero coefficients locate inside the embryo for images from stage range 7–8. These are consistent with the known knowledge from the developmental biology community [Hartenstein 1993]. That is, features from some regions of biological

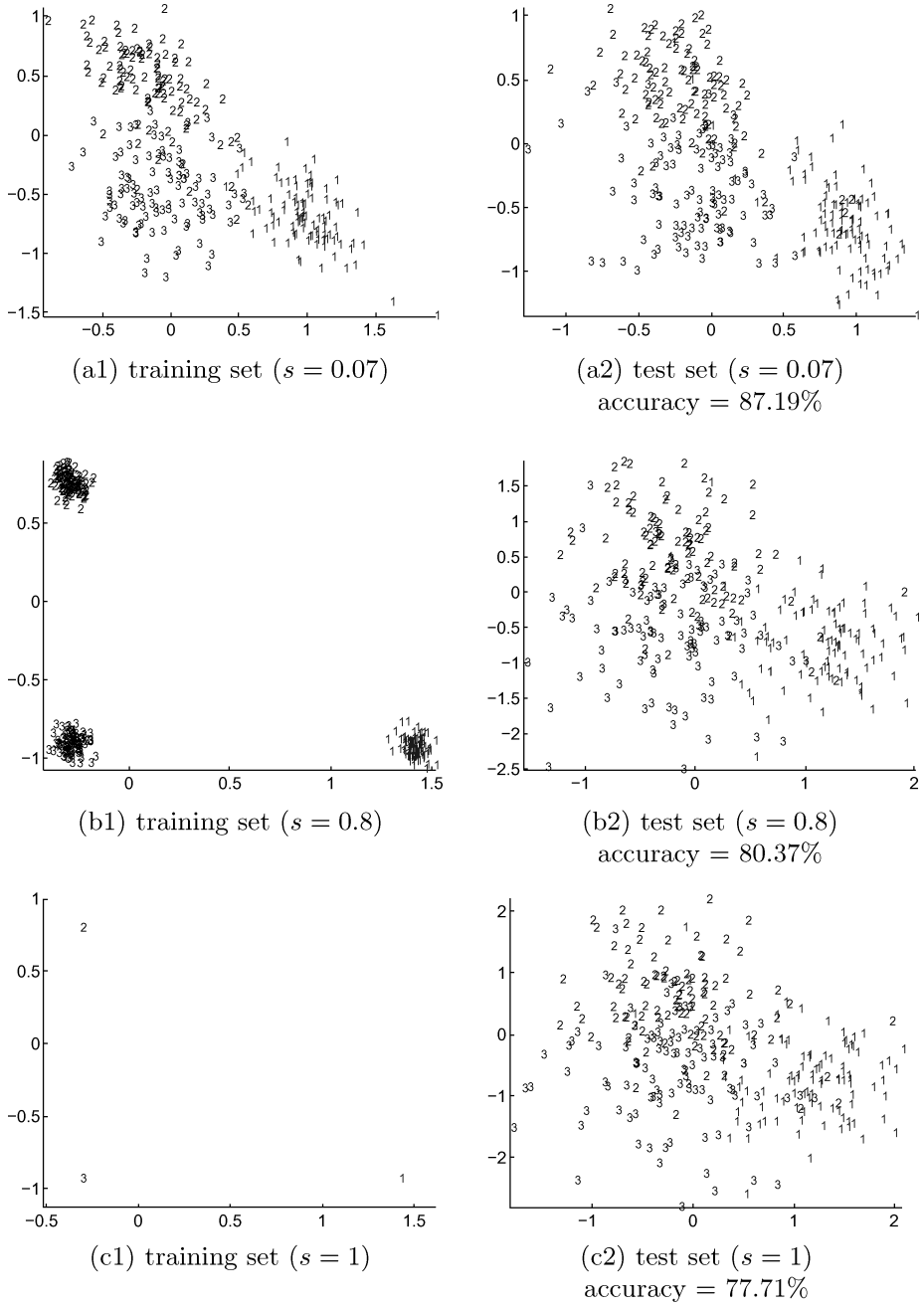
images provide most of the discriminant information in some stages of development. For example, morphological changes at the anterior and posterior end of the embryo occur during stages 4–6 (e.g., the formation and shifting of pole cells at the posterior end), and the prominent displacement of cell membranes at the anterior and posterior ends. Further changes occur in stages 7–8, which are mainly restricted to the interior of the embryo (e.g., the formation of amnioserosa and amnioproctodeal invagination [Hartenstein 1993]). We also observe some regions with nonzero entries outside the embryos. This is probably due to the presence of noises in the image data, which may come from the image generation or image standardization process.

3.6 Visualization

In this experiment, we visualized the effectiveness of LdaPath. To this end, we ran LdaPath with different values of s (0.07, 0.8, 1) on a training set of 300 images and applied the projection to a test set of 2405 images. There were $k = 3$ stage ranges (classes) in our experiments, and all images were projected onto the 3D space spanned by the three weight vectors. In Figure 5, we showed the projection of the training images (left column) and a subset of test images (right column) onto the 2D plane spanned by the first two weight vectors, w_1 and w_2, for clarity of presentation. We depicted each image by the corresponding stage range (1, 2, and 3). We can observe from Figure 5 (graph c1) that when $s = 1$, all training points from the same class are mapped to a common point, which leads to the perfect separation in the training set. However, the test data points are scattered around (graph c2), and the classification accuracy is about 77.71% only. When the value of s decreases, the diameter of each class in the training set increases, while the three classes in the test set are better separated. When $s = 0.8$ (graphs b1–b2), the classification accuracy is about 80.37%. We conducted further studies and found that the best accuracy (about 87.19%) estimated via cross-validation occurs when $s = 0.07$ (graphs a1–a2), and when the value of s further decreases, the accuracy starts to go down. The experimental studies show the effectiveness of the Lasso constraint in LdaPath, as well as the importance of model selection in estimating the optimal value of the Lasso coefficient s.

4. DISCUSSION

In this paper, we present LdaPath for automatic *Drosophila* embryonic developmental stage range classification based on gene expression pattern images. The key features of the proposed LdaPath algorithm include: (1) LdaPath enforces the sparsity constraint in the formulation, which enhances the biological interpretability of the resulting model, and (2) LdaPath computes the entire solution path for all values of regularization parameter, with essentially the same computational cost as fitting one LDA model. Thus LdaPath facilitates efficient model selection. Experiments on a collection of 2705 expression pattern images from early stages show the effectiveness of the LdaPath algorithm. The experimental results demonstrate the promise of the proposed computational approach for automatic embryonic developmental stage range classification.



(a1) training set ($s = 0.07$)

(a2) test set ($s = 0.07$)
accuracy = 87.19%

(b1) training set ($s = 0.8$)

(b2) test set ($s = 0.8$)
accuracy = 80.37%

(c1) training set ($s = 1$)

(c2) test set ($s = 1$)
accuracy = 77.71%

Fig. 5. Visualization of the training images (left row) and a subset of test images (right column) after the projection onto the 2D space spanned by the first two weight vectors with different values of s, i.e., $s = 0.07$ (top row a1–a2), $s = 0.8$ (middle row b1–b2), and $s = 1$ (bottom row c1–c2). The training sample size is 300. Images from the first range (1–3), the second range (4–6), and the third range (7–8) are depicted by 1, 2, and 3, respectively. The test accuracy for each value of s is reported.

We have focused on images from early stages (stage ranges 1–8) in this paper. Since there exist distinct morphological patterns for images from early stages, LdaPath is expected to work well in this case. However, the morphological patterns for images from late stages (stage ranges 9–16) are much more complex than those from early stages (see Figure 1). We plan to expand our data collection to include images from late stage ranges and examine the effectiveness of LdaPath in detecting morphological markers characterizing each of the late stage ranges.

LdaPath bears some resemblance to 1-norm SVM [Zhu et al. 2003]. LdaPath is natural for multiclass classifications. However, how to effectively combine binary classifications remains an important issue in multiclass 1-norm SVM [Wang and Shen 2006]. We plan to examine how the coding matrix Y_3 from LdaPath, as well as other coding methods [Ie et al. 2005; Wang and Shen 2006], may be used in 1-norm SVM for stage range classification and discriminant feature detection, especially when all six stage ranges are considered.

APPENDIX

A. Proof of Lemma 2.1

PROOF. From Equation (15), we can decompose matrix $S_t^+ S_b$ as follows:

$$\begin{aligned} S_t^+ S_b &= U \begin{pmatrix} (\Sigma_t^2)^{-1} & 0 \\ 0 & 0 \end{pmatrix} U^T H_b H_b^T \\ &= U \begin{pmatrix} (\Sigma_t^2)^{-1} U_1^T H_b H_b^T U_1 & 0 \\ 0 & 0 \end{pmatrix} U^T \\ &= U \begin{pmatrix} \Sigma_t^{-1} B B^T \Sigma_t & 0 \\ 0 & 0 \end{pmatrix} U^T \\ &= U \begin{pmatrix} \Sigma_t^{-1} P & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} \Sigma_b & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} P^T \Sigma_t & 0 \\ 0 & I \end{pmatrix} U^T, \end{aligned}$$

where the second equality follows from Equation (16), and the last two equalities follow since $B = \Sigma_t^{-1} U_1^T H_b = P \hat{\Sigma} Q^T$ is the SVD of B as in Equation (18) and $\Sigma_b = \hat{\Sigma} \hat{\Sigma}^T$. Thus, the transformation of ULDA is given by

$$G^U = U_1 \Sigma_t^{-1} P_q, \tag{25}$$

where P_q consists of the first q columns of P, since only the first q diagonal entries of Σ_b are nonzero. On the other hand,

$$\begin{aligned} S_t^+ H_b &= U_1 \Sigma_t^{-1} \big(\Sigma_t^{-1} U_1^T H_b \big) = U_1 \Sigma_t^{-1} P \hat{\Sigma} Q^T \\ &= U_1 \Sigma_t^{-1} P_q \big[\hat{\Sigma}_q, 0 \big] Q^T = \big[G^U \Sigma_{bq}^{0.5}, 0 \big] Q^T, \end{aligned} \tag{26}$$

where $\hat{\Sigma}_q, \Sigma_{bq} \in \mathbb{R}^{q \times q}$ consists of the first q rows and the first q columns of $\hat{\Sigma}$, Σ_b, respectively, the third equality follows since only the first q rows and the first q columns of $\hat{\Sigma}$ are nonzero and the last equality follows since $\Sigma_b = \hat{\Sigma} \hat{\Sigma}^T$.

It follows that

$$W_3 = \left[G^U \Sigma_{bq}^{0.5}, 0\right] Q^T,$$

where Q is orthogonal. □

B. Proof of Theorem 2.1

PROOF. Let matrix $H \in \mathbb{R}^{d \times d}$ be defined as follows:

$$H = U \begin{pmatrix} \Sigma_t^{-1} P & 0 \\ 0 & I_{d-t} \end{pmatrix}, \tag{27}$$

where U and Σ_t are defined in Equation (15), and P is defined in Equation (18). It follows from Equations (15)–(20) that

$$H^T S_b H = \begin{pmatrix} \Sigma_b & 0 \\ 0 & 0 \end{pmatrix}, \quad H^T S_t H = \begin{pmatrix} I_t & 0 \\ 0 & 0 \end{pmatrix}. \tag{28}$$

Since $S_w = S_t - S_b$, we have

$$H^T S_w H = \begin{pmatrix} \Sigma_w & 0 \\ 0 & 0 \end{pmatrix}, \tag{29}$$

for some diagonal matrix $\Sigma_w = I_t - \Sigma_b$.

From Equations (20), (21), (28), and (29), we have

$$\begin{aligned} H^T S_b H &= \operatorname{diag}(\alpha_1^2, \ldots, \alpha_t^2, 0 \ldots, 0), \\ H^T S_w H &= \operatorname{diag}(\beta_1^2, \ldots, \beta_t^2, 0 \ldots, 0), \end{aligned} \tag{30}$$

where $\alpha_1^2 \geq \cdots \geq \alpha_q^2 > 0 = \alpha_{q+1}^2 = \cdots = \alpha_t^2$, and $\alpha_i^2 + \beta_i^2 = 1$, for all i. Condition C1 implies that

$$t = \operatorname{rank}(H^T S_t H) = \operatorname{rank}(H^T S_b H) + \operatorname{rank}(H^T S_w H).$$

Since $\alpha_i^2 + \beta_i^2 = 1$, at least one of α_i and β_i is nonzero. Thus the following inequality:

$$\operatorname{rank}(H^T S_b H) + \operatorname{rank}(H^T S_w H) \geq t$$

always holds. The equality holds only when either α_i or β_i is zero, for all i. That is, $\alpha_i \beta_i = 0$, for all i. Hence, $\alpha_1^2 = \cdots = \alpha_q^2 = 1$, that is, Σ_{bq}, which consists of the first q rows and the first q columns of Σ_b, equals to I_q. □

C. Proof of Proposition 2.1

PROOF. When the Lasso coefficient s is set to 1, the weight matrix W^* in LdaPath equals to W_3. It follows from Equation (26) in Appendix A that

$$W^* = W_3 = U_1 \Sigma_t^{-1} P_q \left[\hat{\Sigma}_q, 0\right] Q^T.$$

From Equations (27) and (29) in Appendix B, we have

$$\left(U_1 \Sigma_t^{-1} P_q\right)^T S_w \left(U_1 \Sigma_t^{-1} P_q\right) = \Sigma_{wq}, \tag{31}$$

where Σ_{wq} consists of the first q rows and the first q columns of the diagonal matrix $\Sigma_w = I_t - \Sigma_b$. From Appendix B, Σ_{bq}, consisting of the first q rows and

the first q columns of Σ_b, is an identity matrix, when condition C1 holds. Thus $\Sigma_{wq} = I_q - \Sigma_{bq} = 0$. It follows from Equation (31) that $(U_1\Sigma_t^{-1}P_q)^T S_w = 0$, as S_w is positive semi-definite. Hence

$$(W^*)^T S_w = Q[\hat{\Sigma}_q, 0]^T \left(U_1\Sigma_t^{-1}P_q\right)^T S_w = 0, \tag{32}$$

and

$$0 = (W^*)^T S_w W^* = (W^*)^T H_w H_w^T W^*, \tag{33}$$

where H_w is defined as in Equation (12):

$$H_w = \left[X_1 - c^{(1)}\left(e^{(1)}\right)^T, \ldots, X_k - c^{(k)}\left(e^{(k)}\right)^T\right],$$

where X_i is the data matrix of the i-th class, and $e^{(i)}$ is the vector of all ones. It follows from Equation (33) that $(W^*)^T H_w = 0$. Considering the i-th block of $(W^*)^T H_w$, we have that

$$(W^*)^T \left(X_i - c^{(i)}(e^{(i)})^T\right) = \left((W^*)^T X_i - (W^*)^T c^{(i)}(e^{(i)})^T\right) = 0.$$

Hence, $(W^*)^T x = (W^*)^T c^{(i)}$, for each column x in X_i. This completes the proof of the proposition. □

D. Computation of the Projection Matrix

Let $A_i \in \mathbb{R}^{r\times c}$, for $i = 1, \ldots, n$, be the n feature matrices ($r = 24$, and $c = 640$). Let $M_i = \frac{1}{n_i}\sum_{y_j=i} A_j$ be the mean of the i-th class, $1 \le i \le k$, and $M = \frac{1}{n}\sum_{i=1}^{n} A_i$ be the global mean. We aim to find a transformation matrix $L \in \mathbb{R}^{r\times\ell}$ that maps each $A_i \in \mathbb{R}^{r\times c}$, for $1 \le i \le n$, to a matrix $B_i \in \mathbb{R}^{\ell\times c}$ such that $B_i = L^T A_i$. The optimal transformation (projection) L is computed based on a variant of Two-dimensional LDA [Ye et al. 2004] as follows.

A natural similarity metric between matrices is the Frobenius norm [Golub and Van Loan 1996]. Under this metric, the (squared) within-class and between-class distances D_w and D_b can be computed as follows:

$$D_w = \sum_{i=1}^{k}\sum_{y_j=i} \|A_j - M_i\|_F^2, \quad D_b = \sum_{i=1}^{k} n_i \|M_i - M\|_F^2,$$

where y_j is the class label of the j-th feature matrix A_j. Using the property of the *trace*, that is, $\text{trace}(MM^T) = \|M\|_F^2$, for any matrix M, we can rewrite D_w and D_b as follows:

$$D_w = \text{trace}\left(\sum_{i=1}^{k}\sum_{y_j=i}(A_j - M_i)(A_j - M_i)^T\right), \tag{34}$$

$$D_b = \text{trace}\left(\sum_{i=1}^{k} n_i (M_i - M)(M_i - M)^T\right). \tag{35}$$

In the low-dimensional space resulting from the linear transformation L, the within-class and between-class distances become

$$\tilde{D}_w = \text{trace}\left(\sum_{i=1}^{k}\sum_{y_j=i} L^T(A_j - M_i)(A_j - M_i)^T L\right), \tag{36}$$

$$\tilde{D}_b = \text{trace}\left(\sum_{i=1}^{k} n_i L^T (M_i - M)(M_i - M)^T L\right). \quad (37)$$

The optimal transformation L would maximize $\tilde{D}_b$ and minimize $\tilde{D}_w$ and is given by the eigenvectors of $\tilde{S}_w^{-1}\tilde{S}_b$, where $\tilde{S}_w$ and $\tilde{S}_b$ are defined as follows:

$$\tilde{S}_w = \sum_{i=1}^{k} \sum_{y_j=i} (A_j - M_i)(A_j - M_i)^T, \quad (38)$$

$$\tilde{S}_b = \sum_{i=1}^{k} n_i (M_i - M)(M_i - M)^T. \quad (39)$$

With the projection by L, each of the feature matrices is reduced to a size of $\ell \times 640$, where $1 \leq \ell \leq 24$. In general, a larger value of ℓ leads to a higher classification accuracy. Our experiments showed that the classification performance improved significantly when we increased the value of ℓ from 1 to 2, while any further increase of ℓ didn't improve the classification performance much. We thus used $\ell = 2$ in the our experiments, and the resulting feature matrices in the low dimensional space are of size 2×640, which were reshaped to feature vectors of length 1280.

REFERENCES

BELHUMEOUR, P., HESPANHA, J., AND KRIEGMAN, D. 1997. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell. 19*, 7, 711–720.

BOWNES, M. 1975. A photographic study of development in the living embryo of Drosophila melanogaster. *J. Embryol. Exp. Morphol. 33*, 789–801.

CARROLL, S., GRENIER, J., AND WEATHERBEE, S. 2005. *From DNA to Diversity: Molecular Genetics and the Evolution of Animal Design.* 2nd Ed. Blackwell Publishing.

D'ASPREMONT, A., GHAOUI, L., JORDAN, M., AND LANCKRIET, G. 2004. A direct formulation for sparse PCA using semidefinite programming. In *Proceedings of the Conference on Advance in Neural Information Processing Systems*. 41–48.

DAUGMAN, J. 1988. Complete discrete 2-d gabor transforms by neural networks for image analysis and compression. *IEEE Trans. Acoust. Speech Signal Proc. 36*, 7, 1169–1179.

DONOHO, D. 2006. For most large underdetermined systems of linear equations, the minimal 11-norm near-solution approximates the sparsest near-solution. *Comm. Pure Appl. Math. 59*, 7, 907–934.

DUDA, R., HART, P., AND STORK, D. 2000. *Pattern Classification*. John Wiley.

EFRON, B., HASTIE, T., JOHNSTONE, I., AND TIBSHIRANI, R. 2004. Least angle regression. *Annals Statis. 32*, 2, 407–499.

FISHER, R. 1936. The use of multiple measurements in taxonomic problems. *Annals Eugenics 7*, 179–188.

FUKUNAGA, K. 1990. *Introduction to Statistical Pattern Classification*. Academic Press.

GOLUB, G. H. AND VAN LOAN, C. F. 1996. *Matrix Computations*. The Johns Hopkins University Press.

GOLUB ET AL., T. 1999. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science 286*, 5439, 531–537.

GONZALEZ, R. AND WOODS, R. 1993. *Digital Image Processing*, 2nd Ed. Addison-Wesley.

GUO, Y., HASTIE, T., AND TIBSHIRANI, R. 2003. Regularized discriminant analysis and its application in microarrays. Tech. Rep. Stanford University.

GURUNATHAN, R., VAN EMDEN, B., PANCHANATHAN, S., AND KUMAR, S. 2004. Identifying spatially similar gene expression patterns in early stage fruit fly embryo images: Binary feature versus invariant moment digital representations. *BMC Bioinform. 5*, 1, 202.

HARTENSTEIN, V. 1993. *Atlas of Drosophila Development*. Cold Spring Harbor Laboratory Press.

HASTIE, T., ROSSET, S., TIBSHIRANI, R., AND ZHU, J. 2004. The entire regularization path for the support vector machine. *J. Mach. Learn. Resear. 5*, 1391–1415.

HASTIE, T., TIBSHIRANI, R., AND FRIEDMAN, J. 2001. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

IE, E., WESTON, J., NOBLE, W., AND LESLIE, C. 2005. Multi-class protein fold recognition using adaptive codes. In *Proceedings of the International Conference on Machine Learning*. 329–336.

KUMAR, S., JAYARAMAN, K., PANCHANATHAN, S., GURUNATHAN, R., MARTI-SUBIRANA, A., AND NEWFELD, S. 2002. BEST: A novel computational approach for comparing gene expression patterns from early stages of *Drosophila melanogaster* development. *Genetics 162*, 4, 2037–2047.

LEE, Y., LIN, Y., AND WAHBA, G. 2004. Multicategory Support Vector Machines, theory, and application to the classification of microarray data and satellite radiance data. *J. Amer. Statist. Assoc. 99*, 67–81.

MIKA, S. 2002. *Kernel Fisher Discriminants*. Ph.D. thesis, University of Technology, Berlin, Germany.

PARK, C. AND PARK, H. 2005. A relationship between LDA and the generalized minimum squared error solution. *SIAM J. Matrix Anal. Appl. 27*, 2, 474–492.

PENG, H. AND MYERS, E. 2004. Comparing in situ mRNA expression patterns of Drosophila embryos. In *Proceedings of the Conference on Research in Computational Molecular Biology*. 157–166.

TIBSHIRANI, R. 1996. Regression shrinkage and selection via the lasso. *J. Royal. Statist. Soc. B.* 1, 267–288.

TOMANCAK ET AL., P. 2002. Systematic determination of patterns of gene expression during Drosophila embryogenesis. *Genome Biol. 3*, 12, 1–14.

WANG, L. AND SHEN, X. 2006. On l_1-norm multi-class support vector machines: Methodology and theory. `http://www.stat.umn.edu/~xshen/paper/ msvm_jasa_rev2.pdf`.

YE, J. 2005. Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems. *J. Mach. Learn. Resear. 6*, 483–502.

YE, J., CHEN, J., LI, Q., AND KUMAR, S. 2006. Classification of *Drosophila* embryonic developmental stage range based on gene expression pattern images. In *Proceedings of the IEEE Computational Systems Bioinformatics Conference*. 293–298.

YE, J., JANARDAN, R., AND LI, Q. 2004. Two-dimensional linear discriminant analysis. In *Proceedings of the Conference on Advance in Neural Information Processing Systems*. 1569–1576.

YE, J. AND XIONG, T. 2006. Computational and theoretical analysis of null space and orthogonal linear discriminant analysis. *J. Mach. Learn. Res.*, 1183–1204.

ZHANG, P. AND RIEDEL, N. 2005. Discriminant analysis: A unified approach. In *ICDM*. 514–521.

ZHU, J., ROSSET, S., HASTIE, T., AND TIBSHIRANI, R. 2003. 1-norm support vector machines. In *Proceedings of the Conference on Advance in Neural Information Processing Systems*. 49–56.

Received June 2007; accepted October 2007