BIOINFORMATICS

Vol. 22 no. 14 2006, pages e417–e423
doi:10.1093/bioinformatics/btl256

# Semi-supervised analysis of gene expression profiles for lineage-specific development in the *Caenorhabditis elegans* embryo

Yuan Qi[1], Patrycja E. Missiuro[2], Ashish Kapoor[3], Craig P. Hunter[4], Tommi S. Jaakkola[1], David K. Gifford[1,*] and Hui Ge[2,*]

[1]Massachusetts Institute of Technology, Computer Science and Artificial Intelligence Laboratory, 32 Vassar Street, Cambridge, MA 02139, USA, [2]Whitehead Institute, 9 Cambridge Center, Cambridge, MA 02142, USA, [3]Microsoft Research, 1 Microsoft Way, Redmond, WA 98052, USA and [4]Department of Molecular and Cellular Biology, Harvard University, Cambridge, MA 02138, USA

## ABSTRACT

**Motivation:** Gene expression profiling is a powerful approach to identify genes that may be involved in a specific biological process on a global scale. For example, gene expression profiling of mutant animals that lack or contain an excess of certain cell types is a common way to identify genes that are important for the development and maintenance of given cell types. However, it is difficult for traditional computational methods, including unsupervised and supervised learning methods, to detect relevant genes from a large collection of expression profiles with high sensitivity and specificity. Unsupervised methods group similar gene expressions together while ignoring important prior biological knowledge. Supervised methods utilize training data from prior biological knowledge to classify gene expression. However, for many biological problems, little prior knowledge is available, which limits the prediction performance of most supervised methods.

**Results:** We present a Bayesian semi-supervised learning method, called BGEN, that improves upon supervised and unsupervised methods by both capturing relevant expression profiles and using prior biological knowledge from literature and experimental validation. Unlike currently available semi-supervised learning methods, this new method trains a kernel classifier based on labeled and unlabeled gene expression examples. The semi-supervised trained classifier can then be used to efficiently classify the remaining genes in the dataset. Moreover, we model the confidence of microarray probes and probabilistically combine multiple probe predictions into gene predictions. We apply BGEN to identify genes involved in the development of a specific cell lineage in the *C. elegans* embryo, and to further identify the tissues in which these genes are enriched. Compared to K-means clustering and SVM classification, BGEN achieves higher sensitivity and specificity. We confirm certain predictions by biological experiments.

**Availability:** The results are available at http://www.csail.mit.edu/~alanqi/projects/BGEN.html

**Contact:** hge@wi.mit.edu or gifford@mit.edu

*To whom correspondence should be addressed.

## 1 INTRODUCTION

Gene expression profiling is a powerful approach to probe global transcriptional programs underlying biological processes. However, it is a challenge to identify candidate genes with high sensitivity and specificity from large compendia of gene expression profiles. For example, in order to uncover transcriptional changes relevant to the development of certain cell types, gene expression profiles are often compared between wild-type animals and mutants that lack or contain an excess of the cell types (Reinke *et al*., 2000; Furlong *et al*., 2001; Gaudet & Mango, 2002; Robertson *et al*., 2004; Baugh *et al*., 2005). Genes that are spatially or temporally enriched can be identified in this way and then tested to confirm their expression patterns. In these cases, gene expression data are usually obtained from whole animals instead of single cells, so differential expression may be partially masked.

Unsupervised clustering methods have been applied to expression profiles to identify candidate genes (Eisen et al., 1998). Clustering methods group together genes with similar expression profiles by modeling the distribution of an entire dataset. However, they do not incorporate knowledge about genes that are already known to be differentially expressed. Consequently, genes clustered together are coherent in terms of expression profiles, yet they may have diverse biological functions.

Another approach to identify candidate genes is to use supervised classification methods. These methods train a model using prior biological knowledge of gene expression, including known regulators and experimentally confirmed candidate genes, and use the trained model for predictions on other genes. However, for many biological processes, either only a few key regulators have been identified, or only a few candidates are experimentally verified. Most classification methods, including Support Vector Machines (SVMs), use training data on known regulators and confirmed candidate genes. Therefore, with a limited amount of training data, it is difficult for supervised methods to achieve accurate predictions.

We propose a semi-supervised learning method that combines the advantages of supervised classification with the benefits of unsupervised clustering. We call this method *BGEN* (Bayesian

GENeralization from examples). By using information from both prior biological knowledge and the entire expression dataset, BGEN allows us to perform accurate predictions even when we only have scarce information about the known regulators. There have been a large number of approaches proposed in recent years for semi-supervised learning and the spectrum of these approaches include random walks, spectral methods (Belkin & Niyogi, 2004; Joachims, 2003; Zhou *et al*., 2004; Zhu *et al*., 2003), and information-regularization (Szummer & Jaakkola, 2003). BGEN differentiates itself from these previous semi-supervised learning approaches in the following ways. First, it provides a principled kernel classifier to classify new data points. Second, we offer a computationally efficient way to choose parameters of the method. Third, specific to microarray data, BGEN explicitly models probe confidence and probabilistically combines predictions from multiple probes corresponding to the same gene.

We apply BGEN to analyze development and differentiation of a specific cell lineage in the *C. elegans* embryo. *C. elegans* is a free-living soil nematode widely used in developmental biology. The adult nematode contains 959 somatic cells. Embryonic cell divisions from a fertilized egg have been traced by microscopy and the cell division patterns are invariant (Sulston *et al*., 1983). The early asymmetric divisions produce six founder cells: AB, MS, E, C, D and P4. Each of these founder cells maintain a distinct pace of cell divisions and produce a specific subset of tissues and cell types. In this paper, we focus on the differentiation of the C lineage, which mainly gives rise to epidermis and muscle cells.

Using previously published expression profiles of wild-type and mutant *C. elegans* embryos (Baugh *et al*., 2005), we identify genes enriched in C lineage and compare the prediction results of BGEN to those of K-means clustering and SVM classification. BGEN outperforms them with improved sensitivity and specificity. We further classify the candidate C-lineage genes from the whole genome into two sub-categories: epidermis enriched genes and muscle enriched genes. The classification is validated by the experimental results obtained by Baugh *et al*. (2005). To further validate our methodology, we experimentally test one gene predicted to be enriched in C-lineage epidermis cells and one gene predicted to be enriched in C-lineage muscle cells. Our experimental results are consistent with our predictions.

## 2 APPROACH

We begin with a gene expression compendium, $\mathbf{X} = \{\mathbf{x}_1,..,\mathbf{x}_{n+m}\}$ where $\mathbf{x}_i$ is the feature vector extracted from the gene expression of probe $i$. We also have a few (n) labeled genes and their corresponding probes, for which $\mathbf{X}_L = \{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ are labeled as $\mathrm{t}_L=\{t_1,\ldots,t_n\}$, and many unlabeled probes $\mathbf{X}_U = \{\mathbf{x}_{n+1},\ldots,\mathbf{x}_{n+m}\}$. Each label $t_i$ is a binary variable. For identification of C-lineage specific genes, labels 1 and $-1$ correspond to C-lineage and non-C-lineage genes, respectively. For classification among C-lineage candidate genes, labels 1 and $-1$ correspond to epidermis and muscle enriched genes, respectively.

Similar to traditional classification methods, we will classify a data point $x_i$ based on a classifier w. Given w, the probability of the label $t_i = 1$ for $\mathbf{x}_i$ in $\mathbf{X}$ is

$$p(t_i \mid \mathbf{x}_i, \mathbf{w}) = \Theta(t_i \mathbf{w}^{\mathrm{T}} \phi(\mathbf{x}_i)) \tag{1}$$

where $\Theta(\cdot)$ is a link function that maps a continuous unbounded value into a value between 0 and 1, and $\phi(\cdot)$ is a basis function, allowing nonlinear separation of data points. Equation (1) is known as the likelihood function of the data $(t_i, \mathbf{x}_i)$. We assume that the data labels are conditionally independent of each other given the input and the classifier, such that $p(\boldsymbol{t}_L \mid \mathbf{X}_L, \mathbf{w}) = \prod_{i:i\in\{1,2,\ldots,n\}} \Theta(t_i \mathbf{w}^{\mathrm{T}} \phi(\mathbf{x}_i))$. Later, we will discuss the likelihood function in more detail.

What distinguishes BGEN from traditional classification or clustering methods is the following: while traditional methods uses either labeled or unlabeled information, BGEN employs the information in both labeled and unlabeled data points. We achieve this by both assigning a data dependent prior $p(\mathbf{w} \mid \mathbf{X})$, which contains the information in *unlabeled* data points $\mathbf{X}_U$, and using the likelihood $p(\mathbf{t}_L \mid \mathbf{X}_L, \mathbf{w})$, which encodes labeled information. We fuse the information in labeled and unlabeled data points by the Bayes rule to compute the posterior distribution $p(\mathbf{w} \mid \mathbf{X}, \boldsymbol{t}_\mathbf{L})$.

Unlike the maximum likelihood or maximum a posteriori approach, which are both point estimates of w for prediction, we average our predictions for $t_i$ based on the posterior distribution $p(\mathbf{w} \mid \mathbf{X}, \mathbf{t}_L)$ to classify unlabeled data points. Note that when given a new data point that is not in the training set $\mathbf{X}$, we can easily classify it based on the classifier posterior $p(\mathbf{w} \mid \mathbf{X}, \mathbf{t}_L)$.

Moreover, in microarray datasets, a gene often corresponds to multiple probes. Therefore, we combine probabilistic predictions of multiple probes to classify their corresponding gene as well as to obtain classification confidence.

In the following subsections we present the prior and the likelihood distributions, describe how to compute the posterior distributions for classifier $\mathbf{w}$ and for label $t_i$, and show how to combine multiple probe predictions for gene classification, and describe experimental approaches to confirm our predictions.

### 2.1 From graph regularization to prior on classifiers

The prior plays a significant role in semi-supervised learning, especially when there is only a small amount of labeled data. In those cases, the prior greatly influences the posterior distribution, since the information from the data likelihood is relatively weak.

It is not an easy task to design a sensible prior on $\mathbf{w}$ that incorporates the information in the data $\mathbf{X}$. So instead of finding a good prior on $\mathbf{w}$ directly, we first introduce a latent vector to $\mathbf{w}$, for which it is relatively easy to assign a prior that contain the data information. Specifically, we introduce a latent vector $\boldsymbol{y} = [y_1,\ldots,y_{n+m}]^{\mathrm{T}}$:

$$y_i = \mathbf{w}^T \phi(\mathbf{x}_i)$$

where $y_i$ can be viewed as a soft label for the data point $\mathbf{x}_i$ and can be converted into the hard label $t_i$ through the link function $\Theta(\cdot)$. Setting $\mathbf{H} = [\phi(\mathbf{x}_1),\ldots,\phi(\mathbf{x}_{n+m})]$ yields

$$\mathbf{y} = \mathbf{H}^T \mathbf{w} \tag{2}$$

If we give a prior on the label $\mathbf{y}$ conditional on the data $\mathbf{X}$, we can then transform the prior $p(\mathbf{y} \mid \mathbf{X})$ to the prior $p(\mathbf{w} \mid \mathbf{X})$ on the classifier $\mathbf{w}$.

Intuitively, we want the prior $p(\boldsymbol{y} \mid \mathbf{X})$ to impose a smoothness constraint on the soft labels and to encourage similar labels between similar data points. Inspired by graph regularization (Zhou *et al*., 2004) we use similarity graphs and their transformed Laplacian to induce priors on the soft labels $\mathbf{y}$.

To construct the prior $p(\mathbf{y}\,|\,\mathbf{X})$, we first form an undirected similarity graph over the data points. The data points are the nodes of the graph and the edge-weights between the nodes are based on similarity. This similarity is usually captured using a kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$. Examples of kernels include Gaussian and polynomials kernels. For Gaussian kernels, $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(\|\mathbf{x}_i - \mathbf{x}_j\|^2/\sigma^2)$ where the kernel width $\sigma$ controls the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$. Given the dataset $\mathbf{X}$ and a kernel, we can construct an $(n+m)\times(n+m)$ kernel matrix $\mathbf{K}$, where $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ for all $i, j \in \{1,\ldots,n+m\}$. Note that the kernel matrix for semi-supervised learning involves both labeled and unlabeled data points. This is different from SVM kernels, which are defined by labeled data points only.

Given the similarity graph, we transform the kernel matrix $\mathbf{K}$ associated with the graph into the combinatorial Laplacian or the normalized Laplacian. Let us construct a matrix $\tilde{\mathbf{K}}$ the same as the matrix $\mathbf{K}$, except that the diagonal elements of $\tilde{\mathbf{K}}$ are set to zero, and define a diagonal matrix $\mathbf{G}$ where $\mathbf{G}_{ii} = \sum_j \tilde{\mathbf{K}}_{ij}$. The combinatorial Laplacian $\Delta$ and the normalized Laplacian $\tilde{\Delta}$ of the graph are defined as

$$\Delta = \mathbf{G} - \tilde{\mathbf{K}} \tag{3}$$

$$\tilde{\Delta} = \mathbf{I} - \mathbf{G}^{-\frac{1}{2}}\tilde{\mathbf{K}}\mathbf{G}^{-\frac{1}{2}} \tag{4}$$

where $\mathbf{I}$ is the identity matrix. Both the Laplacians are symmetric and positive semidefinite. For brevity, we slightly abuse the notation by using $\Delta$ for both the Laplacians. The construction of these Laplacian matrices are based on graph regularization theories. We impose a regularizer preferring soft labeling for which the norm $\mathbf{y}^{\mathrm{T}}\Delta\mathbf{y}$ is small. In a Bayesian framework, we assign a Gaussian prior distribution on y:

$$p(\mathbf{y}\,|\,\mathbf{X}) \propto e^{-\frac{1}{2}\mathbf{y}^{\mathrm{T}}\Delta\mathbf{y}} \propto \mathcal{N}(\mathbf{y}\,|\,0, \Delta^{-1}) \tag{5}$$

where $\mathcal{N}(\cdot\,|\,0,\ \Delta^{-1})$ denotes a Gaussian probability function with mean 0 and variance $\Delta^{-1}$. We can adjust the Laplacian matrices by changing their eigen-spectrum. Here, we use the normalized Laplacian matrices and add diagonal matrices with small values to them, avoiding the matrix inversion singularity.

Given the Gaussian prior on the labels $\mathbf{y}$, we construct the prior on the classifier w as follows:

$$\Sigma = (\mathbf{H}^{\mathbf{T}})^{-1}\Delta^{-1}(\mathbf{H}^{\mathbf{T}})^{-1} \tag{6}$$

$$p(\mathbf{w}\,|\,\mathbf{X}) = \mathcal{N}(\mathbf{w}\,|\,0, \Sigma) \tag{7}$$

where $(\mathbf{H}^{\mathbf{T}})^{-1}$ is the pseudo-inverse of $\mathbf{H}^{\mathbf{T}}$. This prior $p(\mathbf{w}\,|\,\mathbf{X})$ is consistent with the prior $p(\mathbf{y}\,|\,\mathbf{X})$ under the constraint between $\mathbf{y}$ and $\mathbf{w}$, i.e., $\mathbf{y} = \mathbf{H}^{\mathrm{T}}\mathbf{w}$. Again, we add some small positive values to the diagonal elements of $\Sigma$ to enhance its stability.

## 2.2 Modeling probe confidence by likelihood

Assuming conditional independence of the observed labels, we have the factorized likelihood function $p(\mathbf{t}_L\,|\,\mathbf{y}) = \prod_{i=1}^{n}\Theta(t_i\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i))$. The likelihood function $\Theta(t_i\phi(\mathbf{x}_i)^{\mathrm{T}}\mathbf{w})$ for each data point models the probabilistic relation between the observed label $t_i$ and the input feature vector $\phi(\mathbf{x}_i)$. Gene expression datasets often contain noise, which may lead to labeling errors. Also, the qualities of different probes may vary. To model the probe confidence, we adopt the following flipping-error likelihood:

$$\begin{aligned}\Theta(t_i\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i)) &= \epsilon_i(1 - \mathrm{step}(t_i\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i))) \\ &\quad + (1-\epsilon_i)\mathrm{step}(t_i\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i)) \\ &= \epsilon_i + (1-2\epsilon_i)\mathrm{step}(t_i\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i))\end{aligned} \tag{8}$$

where step$(\cdot)$ is a step function such that step$(t_i\mathbf{w}^{\mathrm{T}}(\phi\mathbf{x}_i)) = 1$ if $t_i\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i) >= 0$ and step$(t_i\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i)) = 0$ if $t_i\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i) < 0$, and $\epsilon_i$ models the uncertainty from the noise. This admits labeling errors with probability $\{\epsilon_i\}$. In our dataset, we have multiple probes that correspond to the same gene. The probe that is the closest to the most 3′ end of a gene more accurately measures the expression level of the given gene than the other probes, because the reverse transcription and amplification procedures introduce a bias against probes that are further away from the 3′ end. To model this effect, we set

$$\epsilon_{\mathrm{i}} = \begin{cases} e_l & \text{if probe i is most } 3' \\ e_h & \text{if probe i is not most } 3' \end{cases}$$

where $e_l > e_h$. By doing so, we give non-3′ probes a higher error rate than 3′ probes. Since this likelihood (8) explicitly models the labeling error rate, the model should be more robust to the presence of labeling noise in the data.

## 2.3 Computing the classifier posterior

Given the prior and the likelihood, the classifier posterior is

$$p(\mathbf{w}\,|\,\mathbf{X}, \mathbf{t}_L) \propto p(\mathbf{w}\,|\,\mathbf{X}_L, \mathbf{X}_U)\ \prod_{i=1}^{n}\Theta(t_i\phi(\mathbf{x}_i)^{\mathrm{T}}\mathbf{w}) \tag{9}$$

Because of the nonlinear likelihood terms, we can not compute the exact posterior in a closed form. Instead of using computationally expensive Monte Carlo methods, we apply an efficient deterministic Bayesian approximation technique, expectation propagation (EP) (Minka, 2001; Qi, 2004), to obtain a Gaussian approximation of the posterior $p(\mathbf{w}\,|\,\mathbf{X}, \mathbf{t}_L)$. By exploiting the multiplication form (9) of the posterior, we iteratively refine the approximation of each likelihood term, eventually achieving an accurate approximate posterior. The algorithmic details for EP approximation of Gaussian classifiers can be found in Minka (2001).

## 2.4 Computing and combining probe predictions

As mentioned before, multiple probes are used to measure the expression levels of the same gene in the dataset we analyze. BGEN can classify each probe based on the classifier posterior $p(\mathbf{w}\,|\,\mathbf{X}, \mathbf{t}_L)$. To combine multiple probe predictions, we use a soft decision procedure. Instead of simply averaging the binary probe classification results, we compute the predictive posterior probability for each probe and average these predictive posteriors for all corresponding probes to obtain the prediction for each gene. Specifically, given the approximate classifier posterior $p(\mathbf{w}\,|\,\mathbf{X}, \mathbf{t}_L) \equiv \mathcal{N}(\mathbf{w}\,|\,\mathbf{m}_w, \mathbf{V}_w)$, where $\mathbf{m}_w$ and $\mathbf{V}_w$ are obtained from the EP approximation, we compute the predictive posterior for a probe as follows:

$$p(t_i\,|\,\mathbf{X}, \mathbf{t}_L) = \int p(t_i\,|\,\mathbf{w})p(\mathbf{w}\,|\,\mathbf{X}, \mathbf{t}_L)\mathrm{d}\mathbf{w} \tag{10}$$

$$= \epsilon_i + (1-2\epsilon_i)\varphi(z) \tag{11}$$

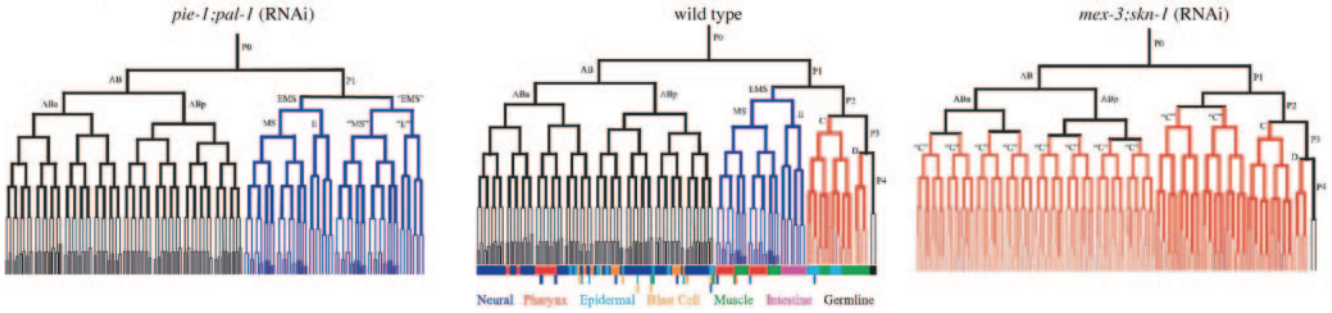


**Fig. 1.** Use of wild-type and mutant embryos to identify genes enriched in C lineage (adapted from Baugh *et al*. (2005)). Cell lineages are illustrated for wild-type embryos (middle), embryos of *pie-1*;*pal-1* (RNAi) genotype (left), and embryos of *mex-3*;*skn-1* (RNAi) genotype (right). C and EMS lineages are shown in red and purple, respectively.

where

$$z = \frac{t_i \phi(\mathbf{x}_i)^{\mathrm{T}} \mathbf{m}_w}{\sqrt{\phi(\mathbf{x}_i)^{\mathrm{T}} \mathbf{V}_w \phi(\mathbf{x}_i)}} \tag{12}$$

and $\varphi(\cdot)$ is the cumulative distribution function of a Gaussian with mean 0 and variance 1. Equation (12) shows that the predictive posterior is controlled not only by the posterior mean $\mathbf{m}_w$ of the classifier, but also by the uncertainty, the variance $\mathbf{V}_w$ for the trained classifier. We average the predictive posteriors of the probes corresponding to the same gene $k$ to obtain a gene predictive probability $p(\text{gene}_k \mid \mathbf{X}, \text{t}_L)$. Note that non-3′ probes contribute less to the gene prediction, since with a larger $\epsilon_i$ their predictive posteriors are less informative than the predictive posteriors of 3′ probes.

### 2.5 Automatic hyperparameter tuning

BGEN has a few hyperparameters, including kernel width $\sigma$ and probe confidence levels $e_l$ and $e_h$. To achieve a good test performance, we need to tune these hyperparameters. Here we adopt an automatic procedure to estimate them in a principled way. As a side-product of EP for our Bayesian learning, we estimate the approximate-leave-one-out error count or probability without carrying out leave-one-out cross-validation. The details can be found in Qi *et al*. (2004). We use the approximate leave-one-out error probability to estimate these hyperparameters.

### 2.6 Experimental validation of gene expression patterns

We examine gene expression patterns by using a reporter assay. We fuse selected gene promoters to yellow fluorescence protein (YFP) and a dominant *rol-6* gene by PCR (Hobert, 2002). 5′ genomic sequences up to the next upstream gene are used as promoters. YFP is amplified from pPD132.112 (Fire *et al*., 1990). The *rol-6* gene, a co-transformation marker, is amplified from pRF4 (Mello *et al*., 1991). Transgenic lines are obtained by injection of the reporter constructs. Chromosomal integration is performed by gamma irradiation. Using fluorescence microscopes we observe expression patterns of reporter genes in embryos from integrated transgenic lines.

## 3 RESULTS

This section describes the expression profile dataset used for our task, presents our prediction results for genes enriched in the C lineage, and compare the prediction accuracy of BGENs with those of K-means and SVMs. Finally, we confirm some predictions with biological experiments.

### 3.1 Summary of expression dataset

Baugh *et al*. (2005) profiled global gene expression for wild-type *C. elegans* embryos and two types of mutant embryos at 0, 23, 41, 53, 66, 83, 101, 122, 143, and 186 minutes after 4-cell stage. Embryos of the *pie-1*;*pal-1* (RNAi) genotype lack C-lineage cells, while embryos of the *mex-3*;*skn-1* (RNAi) genotype bear excess C-lineage cells (Figure 1).

Expression patterns of selected reporter genes in *C. elegans* embryos reflected whether these candidates were specific to the C lineage, and the confirmed candidates could be further classified as epidermis or muscle enriched (Baugh *et al*., 2005). Among the 40 candidates tested, 25 were confirmed to be C-lineage enriched. A non-specific gene list comes from an RNAi screen that identified 661 genes required for the first two cell divisions of the *C. elegans* embryo (Sonnichsen *et al*., 2005). The first two cell divisions occur well before the development of C lineage and these genes are believed to encode proteins for the basic mitotic machinery. Therefore, these genes are likely not to be specific to any lineage development.

### 3.2 Semi-supervised learning and comparison with K-means clustering and SVM classification

We use experimentally confirmed C-lineage genes reported by Baugh *et al*. (2005) as labeled positive examples, and use the non-specific genes required for early cell divisions as labeled negative examples.

For each gene, we calculate the difference of its expression levels in *mex-3*;*skn-1* (RNAi) embryos and *pie-1*;*pal-1* (RNAi) embryos at each time point, and use the ratios of this difference over the expression level in wild-type embryos as extracted features for clustering and classification. The maximum value of the ratios during development is also used as an extracted feature.

We compare BGEN with K-means clustering and SVM classification. First, we perform K-means clustering, which does not use the labeled information at all. The performance of K-means depends on the number of clusters which is unknown a priori. We use Silhouette scores to determine the optimal number of clusters (Kaufman & Rousseeuw, 1990). The Silhouette scores measure the tightness of a cluster and the separation of the given cluster from other clusters. More specifically, the Silhouette scores show how close a data point in one cluster is to data points in the neighboring clusters. The score ranges from +1, indicating that data points in one cluster are close to one another and are distant from data points in neighboring clusters, to −1, indicating the opposite. We compute the average Silhouette scores for all genes in the dataset. K-means with 2 clusters has the highest average score 0.8481. This score suggests that the two clusters obtained by K-means are coherent among themselves and well-separated from each other. To evaluate the capability of K-means to detect C-lineage genes, we designate a cluster to be C-lineage cluster if the ratio of labeled C-lineage genes to all genes in that cluster exceeds a specified threshold between 0 and 1; otherwise we designate it as a non C-lineage cluster. Genes in a C-lineage cluster are predicted to be C-lineage genes, and vice versa. We vary the threshold value and average the detection results over 200 runs with random initializations. The Receiver Operating Characteristic (ROC) curve from the averaged detection results is shown in Figure 2. K-means clustering performs poorly in terms of detecting C-lineage genes, though the clustering achieves a high average Silhouette score. The underlying reason may come from the fact that K-means clustering ignores any prior biological knowledge and purely depends on the expression dataset, and that C-lineage expression profiles are diverse.

For BGEN and SVM, we use experimentally confirmed C-lineage genes reported by Baugh *et al*. (2005), excluding genes used as positive training data, to evaluate the sensitivity. We use the non-specific genes required for early cell divisions, excluding genes used as negative training data, to assess the specificity.

For SVM training, we construct a pool of representative positive labels: *pal-1*, *vab-7*, *cwn-1*, *elt-1*, *elt-3*, *mab-21*, *hnd-1* and *hlh-1*. Each time 4 genes are randomly selected from this pool and serve as positive training examples. We randomly select 20 genes as negative training examples from the non-specific genes. We test the SVM prediction performance on the rest of the labeled data points. For BGEN, we use the same labeled examples, as well as about 900 unlabeled examples for training. We repeat this training and prediction procedure 10 times. We use Gaussian kernels for both SVM and BGEN. The regularization and kernel widths of SVM are tuned by leave-one-out cross-validations. For BGEN, both the kernel width and probe confidence levels are tuned based on the approximate leave -one-out error probability without actually carrying out leave-one-out cross -validations, as described in section 2.5. Based on the averaged prediction results, we plot ROC curves for BGEN and SVM (Figure 2). Overall BGEN performs significantly better than SVM. For example, with the same 80% specificity (i.e.,20% false positive rate), BGEN achieves 99% sensitivity (i.e., true positive rate), while SVM achieves only 82% sensitivity. Moreover, BGEN clearly outperforms K-means clustering in terms of detecting C-lineage genes as shown in Figure 2.

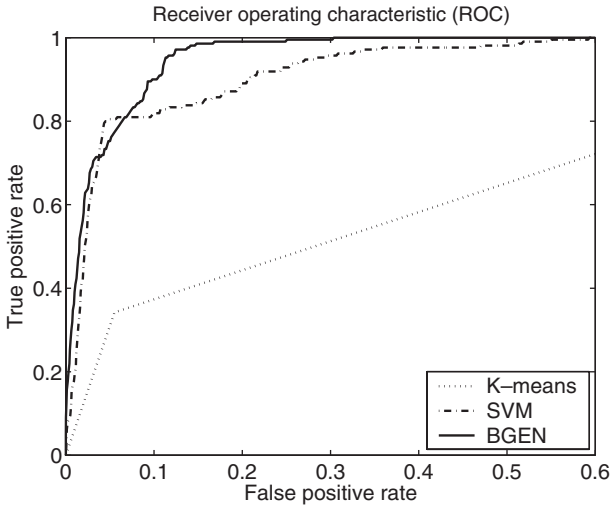


**Fig. 2.** Receiver Operating Characteristic (ROC) curves of BGEN, SVM, and K-means. Our semi-supervised learning method BGEN outperforms both SVM and K-means.

### 3.3 Whole genome prediction of C-lineage genes

Having tested the efficacy of BGEN, we predict C-lineage genes in the whole genome. We use 20 negative examples and all positive examples except for *pal-1*, because *pa1-1* is a maternally-supplied regulator while we are interested in identifying genes which are active in zygotic transcription during development. With 97% specificity evaluted by the non-specific gene set, we predicted 317 genes as enriched in C lineage, in addition to the previously confirmed C-lineage genes.

Our whole genome prediction is highly efficient in the sense that we use a kernel classifier pre-trained in a semi-supervised fashion to classify whole genome. This is different from many previous semi-supervised learning methods (Joachims, 2003; Zhou *et al*., 2004; Zhu *et al*., 2003), where either a re-training or a simple nearest-neighbor classifier is needed to classify new data points in addition to the training set.

BGEN may reduce potential false-positives from the original analysis. For example, F45E4.9(*hmg-5*), a HMG-box containing protein, which was previously predicted to be enriched in C lineage while our method classifies it as a non-C- lineage gene with a probabilistic confidence of 0.10. The experimental result showed that the expression pattern of F45E4.9 is not specific to the C lineage. This is also consistent with other reports in the literature that F45E4.9 is ubiquitously expressed in *C. elegans* embryos (Im & Lee, 2003). Another example is Y71F9AL.17, an uncharacterized gene that may be involved in intracellular trafficking and vesicular transport. Y71F9AL.17 was previously identified as a C-lineage candidate gene. In our analysis this gene receives a probabilistic confidence of 0.46 and is classified as non-C-lineage (Figure 3). The result of biological experiment was consistent with our prediction.

To visualize our predictions, we plot representative expression profiles for C-lineage genes and non-C-ineage genes with high prediction confidence (Figure 3). D1005.2 and F54D7.4 (the first column), two high-confidence C-lineage genes, are up- regulated

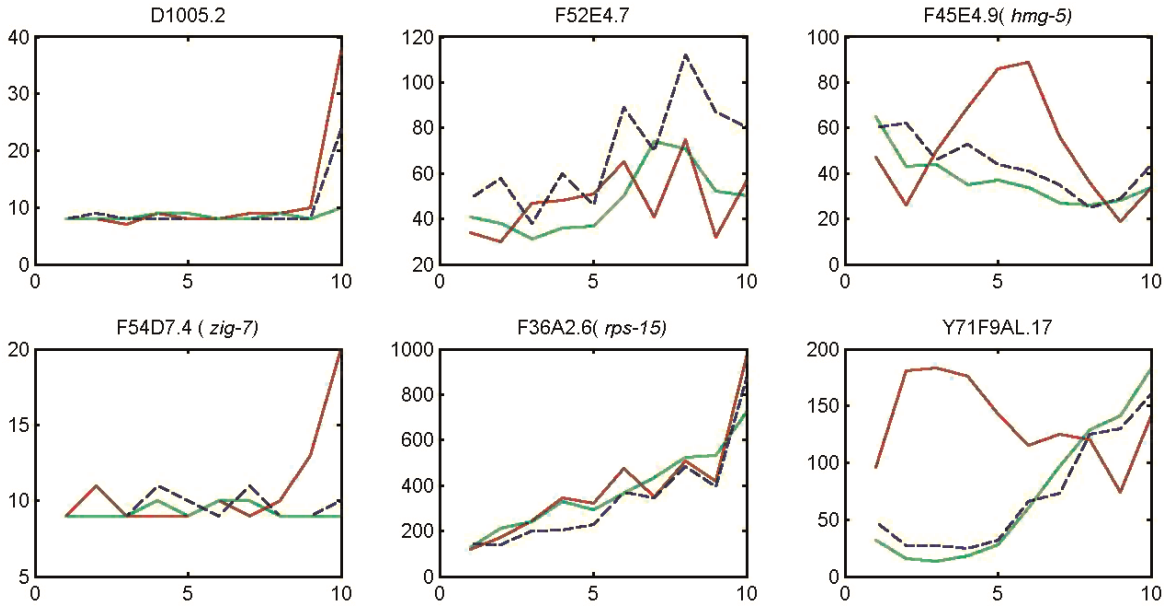


**Fig. 3.** Expression profiles of prediction examples. Red lines represent expression profiles in *mex-3*;*skn-1* (RNAi) embryos. Green lines represent expression profiles in *pie-1*;*pal-1* (RNAi)embryos. Blue dotted lines represent expression profiles in wild-type embryos. D1005.2 and F54D7.4 are high-confidence predictions of C-lineage genes. They receive confidence scores of 0.99 and 0.98, respectively. F52E4.7 and F36A2.6 are high-confidence predictions of non-C-lineage genes. They both receive confidence scores of 0.01. F45E4.9 and Y71F9AL.17 are less obvious examples. They receive confidence scores of 0.10 and 0.46, respectively, and are classified as non-C-lineage specific genes. Baugh *et al*. (2005) identified F45E4.9 and Y71F9AL.17 as C -lineage genes in their data analysis, but subsequent experimental results showed that these two genes were not specific to the C lineage.

in *mex-3*; *skn-1* (RNAi) embryos during development. F52E4.7 and F36A2.6 (the second column), two high-confidence non-C-lineage genes, do not exhibit such up-regulation of expression. The two examples of false-positives (F45E4.9 and Y71F9AL.17) by the previous analysis are also plotted. These two genes are prone to misprediction since they are up-regulated in *mex-3*; *skn-1* (RNAi) embryos. These examples illustrate the capability of BGEN to distinguish C-lineage genes from non-C-lineage genes even in some subtle cases.

### 3.4 Predictions of C epidermis and C muscle genes

During embryonic development, C-lineage cells differentiate into epidermis and muscle cells. Epidermis and muscle enriched genes are likely to exhibit slightly different expression profiles in wild-type and mutant embryos. Given our whole genome predictions of C-lineage genes, we apply BGEN to further distinguish the C- lineage genes as epidermis or muscle enriched. Baugh *et al*. (2005) showed by reporter assay that among the confirmed C-lineage genes, 15 were specifically expressed in epidermis cells and 4 were specifically expressed in muscle cells. We use this information to train and evaluate K-means, SVM, and BGEN. In addition to the normalized features used in 3.3, 2-level Daubechies wavelet decomposition of the difference features that explicitly represents the temporal and frequency information in the data is also computed as features.

Similar to what we have done before, we use the Silhouette scores to determine the number of clusters for K-means. For SVM and BGEN, we randomly select 6 epidermis and 2 muscle-genes and use them as training data. We use the rest of experimentally confirmed genes as the test set, which includes 9 epidermis genes and 2 muscle genes for each run. We repeat this procedure 5 times.

We evaluate the average area under the ROC curves for these three methods. For K-means, we compute the ROC curve using the same method as in the previous section. The average area under the ROC curve of BGEN is 0.80, indicating its prediction potential. The average areas achieved by K-means and SVM are only 0.56 and 0.50 respectively, indicating the failure of the K-means and SVM predictions. This further demonstrates the advantage of our semi- supervised learning method. For the run in which BGEN achieves the largest area under the ROC curve, we correctly predict all 9 epidermis genes and 2 muscle genes in the test set. The prediction accuracy achieved by BGEN suggests the epidermis genes and muscle genes may be separable from each other in terms of expression profiles. However, this prediction accuracy should not be over- interpreted, because both the training and testing datasets are small. In the future, more labeled data and additional microarray datasets may be integrated to improve the predictions. The lists of predicted C epidermis and C muscle genes can be downloaded at http://www.csail.mit.edu/~alanqi/~projects/BGEN.html.

### 3.5 Experimental verification of predictions

We predict K01A2.5 and R11A5.4, two uncharacterized genes, as enriched in C lineage. These two genes were also identified in previous analysis as C-lineage candidates but were not tested (Baugh *et al*., 2005). We further identify K01A2.5 as epidermis enriched and R11A5.4 as muscle enriched. We examine their expression patterns by reporter assay. The expression patterns

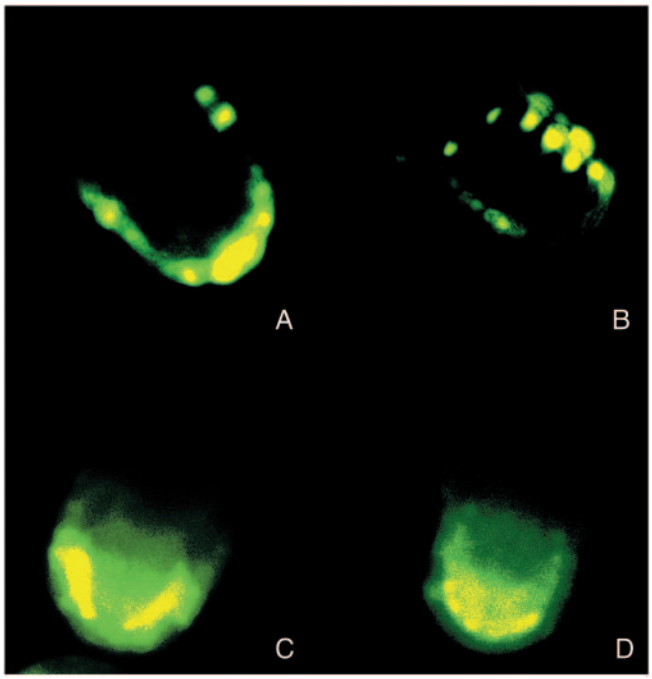


**Fig. 4.** Experimental validation of redictions. We predict K01A2.5 and R11A5.4, as enriched in C epidermis cells and enriched in C muscle cells, respectively. We examine expression patterns of K01A2.5 (A, B) and R11A5.4 (C, D) in developing *C. elegans* embryos. The experimental results are consistent with our predictions for both genes.

of reporter genes in *C. elegans* embryos are consistent with our predictions (Figure 4). The reporter gene that contains K01A2.5 promoter is expressed in C epidermis cells, and the reporter gene that contains R11A5.4 promoter is expressed in C muscle cells. The experimental results support that our methodology yields relevant biological insights to elucidate developmental processes.

## 4 CONCLUSIONS

We have developed BGEN, a novel semi-supervised learning method, which utilizes both large-scale expression datasets and prior biological knowledge to identify target genes. Using BGEN, we have predicted genes enriched in C lineage during *C. elegans* embryonic development, and have further classified C-lineage candidate genes according to tissues where they are enriched. In comparison with unsupervised K-means clustering and supervised SVM classification, our semi-supervised learning method achieves higher sensitivity and specificity. We experimentally confirm two examples from our predictions, which further supports our methodology. As a powerful computational tool, BGEN can be used to refine target selection from large-scale expression datasets for many other biological problems in the future.

## ACKNOWLEDGEMENTS

We thank R. Dowell for critical reading of our manuscript. H.G. is supported by Whitehead Institute and supported in part by NIH GM 644429 to C.P.H.

## REFERENCES

Baugh,L.R., Hill,A.A., Claggett,J.M., Hill-Harfe,K., Wen,J.C., Slonim,D.K., Brown,E.L. and Hunter,C.P. (2005) The homeodomain protein PAL-1 specifies a lineage-specific regulatory network in the *C. elegans* embryo. *Development*, **132** (8), 1843–1854.

Belkin,M. and Niyogi,P. (2004) Semi-supervised learning on Riemannian manifolds. *Machine Learning, Special Issue on Clustering*, **56**.

Eisen,M.B., Spellman,P.T., Brown,P.O. and Botstein,D. (1998) Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci USA*, **95** (25), 14863–8.

Fire,A., Harrison,S.W. and Dixon,D. (1990) A modular set of lacz fusion vectors for studying gene expression in *Caenorhabditis elegans*. *Gene*, **93** (2), 189–98.

Furlong,E.E.M., Andersen,E.C., Null,B., White,K.P. and Scott,M.P. (2001) Patterns of gene expression during *Drosophila* mesoderm development. *Science*, **293** (5535), 1629–1633.

Gaudet,J. and Mango,S.E. (2002) Regulation of organogenesis by the *Caenorhabditis elegans* FoxA protein PHA-4. *Science*, **295** (5556), 821–825.

Hobert,O. (2002) PCR fusion-based approach to create reporter gene constructs for expression analysis in transgenic *C. elegans*. *Biotechniques*, **32** (4), 728–30.

Im,S.H. and Lee,J. (2003) Identification of HMG-5 as a double-stranded telomeric DNA-binding protein in the nematode *Caenorhabditis elegans*. *FEBS Lett*, **554** (3), 455–61.

Joachims,T. (2003) Transductive learning via spectral graph partitioning. *ICML*.

Kaufman,L. and Rousseeuw,P.J. (1990) *Finding Groups in Data: an Introduction to Cluster Analysis*. Wiley.

Mello,C.C., Kramer,J.M., Stinchcomb,D. and Ambros,V. (1991) Efficient gene transfer in *C. elegans*: extrachromosomal maintenance and integration of transforming sequences. *Embo J*, **10** (12), 3959–70.

Minka,T.P. (2001) Expectation propagation for approximate Bayesian inference. In *Uncertainty in Artificial Intelligence*.

Qi,Y. (2004) *Extending expectation propagation for graphical models*. Ph.D. thesis, MIT.

Qi,Y., Minka,T.P., Picard,R.W. and Ghahramani,Z. (2004) Predictive automatic relevance determination by expectation propagation. In *Proceedings of Twenty-first International Conference on Machine Learning*.

Reinke,V., Smith,H.E., Nance,J., Wang,J., Van Doren,C., Begley,R., Jones,S.J.M., Davis,E.B., Scherer,S., Ward,S. and Kim,S.K. (2000) A global profile of germline gene expression in *C. elegans*. *Molecular Cell*, **6** (3), 605–616.

Robertson,S.M., Shetty,P. and Lin,R. (2004) Identification of lineage -specific zygotic transcripts in early *Caenorhabditis elegans* embryos. *Dev Biol*, **276** (2), 493–507.

Sonnichsen,B., Koski,L.B., Walsh,A., Marschall,P., Neumann,B., Brehm,M., Alleaume,A.M., Artelt,J., Bettencourt,P., Cassin,E., Hewitson,M., Holz,C., Khan,M., Lazik,S., Martin,C., Nitzsche,B., Ruer,M., Stamford,J., Winzi,M., Heinkel,R., Roder,M., Finell,J., Hantsch,H., Jones,S.J., Jones,M., Piano,F., Gunsalus,K.C., Oegema,K., Gonczy,P., Coulson,A., Hyman,A.A. and Echeverri,C.J. (2005) Full-genome RNAi profiling of early embryogenesis in *Caenorhabditis elegans*. *Nature*, **434** (7032), 462–9.

Sulston,J.E., Schierenberg,E., White,J.G. and Thomson,J.N. (1983) The embryonic cell lineage of the nematode *Caenorhabditis elegans*. *Dev Biol*, **100** (1), 64–119.

Szummer,M. and Jaakkola,T. (2003) Information regularization with partially labeled data. *NIPS*.

Zhou,D., Bousquet,O., Lal,T.N., Weston,J. and Scholkopf,B. (2004) Learning with local and global consistency. *NIPS*, **16**.

Zhu,X., Ghahramani,Z. and Lafferty,J. (2003) Semi-supervised learning using Gaussian fields and harmonic functions. *ICML*.