JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 18, Number 11, 2011

Pp. 1709–1722
DOI: 10.1089/cmb.2011.0193

Learning Cellular Sorting Pathways Using Protein Interactions and Sequence Motifs

TIEN-HO LIN,[1] ZIV BAR-JOSEPH,[2] and ROBERT F. MURPHY[2,3]

ABSTRACT

Proper subcellular localization is critical for proteins to perform their roles in cellular functions. Proteins are transported by different cellular sorting pathways, some of which take a protein through several intermediate locations until reaching its final destination. The pathway a protein is transported through is determined by carrier proteins that bind to specific sequence motifs. In this article, we present a new method that integrates protein interaction and sequence motif data to model how proteins are sorted through these sorting pathways. We use a hidden Markov model (HMM) to represent protein sorting pathways. The model is able to determine intermediate sorting states and to assign carrier proteins and motifs to the sorting pathways. In simulation studies, we show that the method can accurately recover an underlying sorting model. Using data for yeast, we show that our model leads to accurate prediction of subcellular localization. We also show that the pathways learned by our model recover many known sorting pathways and correctly assign proteins to the path they utilize. The learned model identified new pathways and their putative carriers and motifs and these may represent novel protein sorting mechanisms. Supplementary results and software implementation are available from http://murphylab.web.cmu.edu/software/2010_RECOMB_pathways/.

Key words: gene expression, HMM, machine learning, pathways, protein motifs, subcellular localization, protein sorting.

1. INTRODUCTION

To perform their function(s), proteins usually need to be localized to the specific compartment(s) in which they operate. Subcellular localization of proteins is typically achieved by sorting pathways involving carrier proteins. Disruption of these pathways leading to inaccurate localization plays an important role in several diseases, including cancer (Cohen et al., 2008; Kau et al., 2004; Gladden and Diehl, 2005), Alzheimer's disease (De Strooper et al., 1997), hyperoxaluria (Purdue et al., 1990), and cystic fibrosis (Skach, 2000). Thus, an important problem in systems biology is to determine how proteins are localized to their target compartments, the carriers and motifs that govern this localization, and the pathways that are being used.

[1]Language Technology Institute, [2]Lane Center for Computational Biology and Machine Learning Department, School of Computer Science, and [3]Departments of Biological Sciences and Biomedical Engineering, Carnegie Mellon University, Pittsburgh, Pennsylvania.

Recent advances in fluorescent microscopy coupled with automated image-based analysis methods provide rich information about the compartments to which proteins are localized in yeast (Huh et al., 2003; Chen et al., 2007) and human (Osuna et al., 2007; Barbe et al., 2008; Newberg et al., 2009). Several computational methods have been developed to predict subcellular localization by integrating sequence data with other types of high-throughput data (Chou and Shen, 2008; Horton et al., 2007; Emanuelsson et al., 2007, 2000; Nair and Rost, 2005; Scott et al., 2005; Rashid et al., 2007; Bannai et al., 2002). These methods either treat the problem as a one versus all classification problem (Chou and Shen, 2008; Emanuelsson et al., 2007, 2000; Horton et al., 2007) or utilize a tree that corresponds to the current knowledge regarding intermediate compartments, for example, LOCtree (Nair and Rost, 2005), BaCelLo (Pierleoni et al., 2006), and discriminative HMMs (Lin et al., 2011). The tree-based methods were shown to be superior to the one versus all methods; however, these methods do not attempt to learn the sorting pathways, relying instead on current (partial) knowledge of protein sorting mechanism.

A number of methods have learned decision trees for predicting subcellular localization. These include PSLT2 (Scott et al., 2005), which refines the location into sub-compartments using a decision tree learned from data, and YimLOC (Shen and Burger, 2007), which learns a decision tree for the mitochondrion compartment only using features that include predictions from SherLoc (Shatkay et al., 2007), an abstract-based localization classifier. While the decision trees generated by these methods are often quite accurate, they are not intended to reflect sorting pathways, and they utilize features that, while useful for classification, are not related to the biochemical process of protein sorting.

In contrast to the global localization prediction methods, several experimental researchers have focused on trying to assign a specific sorting pathway to a small number of proteins. For example, proteins containing a signal peptide are exported through the secretory pathway (Lodish et al., 2003), while some proteins without a classical N-terminal signal peptide are found to be exported via the non-classical secretory pathway (Rubartelli and Sitia, 1997). A number of computational methods were developed to use this information to predict, for a given pathway, whether a protein goes through that pathway or not based on its sequence—for example, SignalP (Bendtsen et al., 2004b) and SecretomeP (Bendtsen et al., 2004a). However, these methods rely on the pathway as an input and cannot be used to infer new pathways.

There are many methods developed for reconstruction of pathways of other types, for example, for signaling pathways (Ruths et al., 2008; Bebek and Yang, 2007; Scott et al., 2006) and metabolic pathways (Dale et al., 2010; Fischer and Sauer, 2005; Covert et al., 2004). These pathways are used to describe information flow: one protein senses the environments and by activating a signaling or regulatory pathway passes that information along so that the cells can mount a response. We focused on a completely different meaning of pathway: physical movement of a specific protein. When referring to sorting pathways, we mean that a single protein is being carried from one location to another. Unlike information flow pathways, which involve different molecules along the way, physical sorting pathways always involve the same proteins interacting with a set of different proteins. This makes it much more complicated to infer the order in which this is performed (since it is always the same protein). In addition, the outcome of an information flow pathway is often a change in genes expression which can be readily measured using microarrays. In contrast, the outcome of a sorting pathway is the localization of a single (or a few) proteins to a compartment. Again, this requires different methods for inference. We are not aware of any prior article discussing computational methods for large scale inference of pathways describing physical movement of a protein.

While the above experimental methods provide some information on sorting pathways, no method exists to try and infer global sorting pathways from current localization information. In this article, we show that, by integrating sequence, motif, and protein interaction data, we can develop global models for the process in which proteins are localized to subcellular compartments. We use a hidden Markov model (HMM) to represent sorting pathways. Carrier proteins and motifs are used to define internal states in this model and the compartments serve as the final (goal) state. Using this model, we identified several sorting pathways, the carrier proteins that govern them, and the proteins that are being sorted according to these pathways. Simulation data indicates that the models learned are accurate (leading to 81% prediction accuracy with a noise level of 5%; see Fig. 3 below). Using data from yeast, we show that our model leads to accurate classification of protein compartments while at the same time enabling us to recover many known pathways and the proteins that govern these pathways. Several new predictions are provided by the model representing new putative sorting pathways.

2. METHODS

2.1. Input data

Our input data is composed of the localization of all proteins, their interactions, and their sequences. Each protein is labeled with one or more locations. Generative HMM search for motifs present in one compartment and discriminative HMM search for motifs present in one compartment but absent in other compartments. We also collected all interacting partners of the protein and the occurrences of a set of known motifs from public databases (denoted as deterministic motifs to distinguish from novel motifs extracted from sequence described below), specifically InterPro (Mulder et al., 2003) domains and three signal sequence feature from UniProt (Bairoch et al., 2005): signal peptides, transmembrane region, and GPI anchor (for more detail, see Section 3.2). We perform feature selection by a hypergeometric test to identify features with a significant association with a location before learning our model.

We extract novel motifs associated with a location using the generative and discriminative HMM motif finder we have previously described (Lin et al., 2011). We will compare two approaches to convert each sequence to motif features: sequence likelihood and binary occurrence. The first approach uses the sequence likelihood given the motif as feature, $\Pr(S|\lambda_k)$ where λ_k is the profile HMM of the motif. It represents how strong the instance matches the motif. Note that what really matters is the likelihood ratio of motif versus background, as described below. The second approach uses a binary value to represent whether a motif occurs in a sequence instead of a real value. Binary motif occurrence are determined by posterior decoding as described in our previous article (Lin et al., 2011).

2.2. Modeling sorting pathway by hidden Markov models

We used a HMM to model the process of sorting proteins to their compartments, determined by the interactions and sequence motifs. HMM is a generative model and thus provides the set of events that lead to the observed localization of the proteins (Fig. 1). An allowed pathway through the HMM state space structure represents a possible protein sorting pathway. All proteins start at the same start state, representing their translation in the cytoplasm. (While those few proteins that are translated in mitochondria would not begin in the cytoplasm, there were no mitochondrially-encoded proteins in our datasets and we can ignore this possibility.) The assigned (final) compartment of a protein is represented by a state in the model that

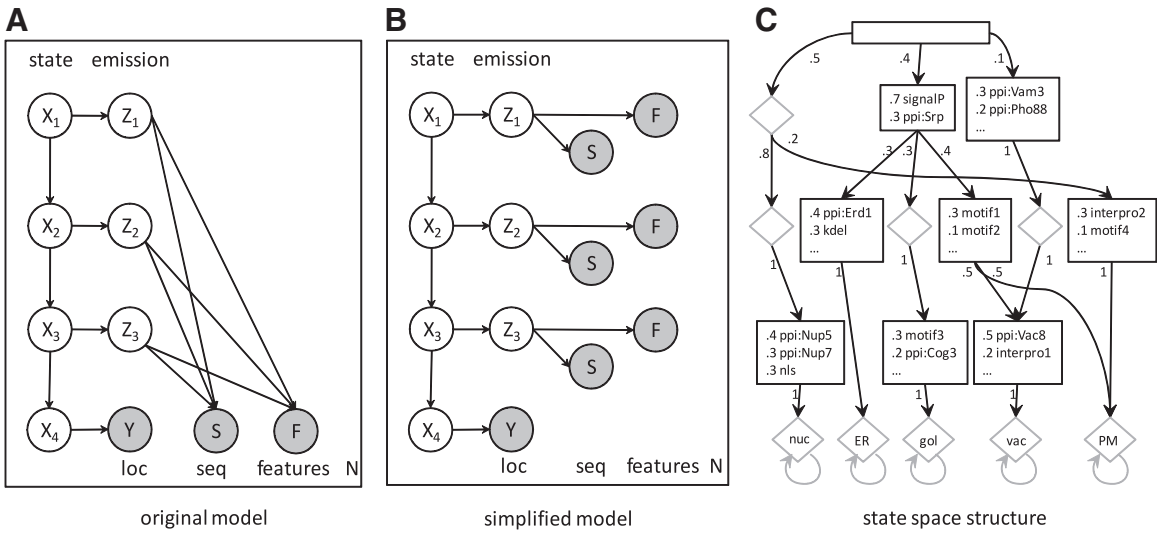

FIG. 1. **(A)** The graphical model representation of a sample HMM for sorting pathways. Variables $X_1 \cdots X_4$ are unobserved intermediate sorting states at each level or each step. $Z_1 \cdots Z_3$ are the emission responsible for protein sorting at each step. S is the sequence and F corresponds to the binary feature observations. **(B)** The simplified HMM that maintains conditional independence between steps. **(C)** A sample state space: The top block is the root and its outgoing arrows correspond to initial probabilities. Bottom nodes are compartment states. The blocks are states and the arrows are transitions, with transition probabilities labeled. The items listed inside a blocks are top features emitted by the states, and emission probabilities are given on the left. Diamond-shaped blocks are silent states that emit the background feature only.

does not have any outgoing transitions. Intermediate states correspond to intermediate compartments or to sorting events (e.g., interaction with a carrier protein). These internal states emit observed features that are related to the sorting events, namely motifs (implying that the targeted protein uses that motif to direct it to that state) and carrier proteins that target proteins to the state. The emitted features of a protein are observed and determine its path in the state space. Emission is probabilistic, and so certain proteins can pass through states even if they do not contain any of the motifs and do not interact with any of the carriers for that state. Note that while the compartment information is available during training, we do not know how many intermediate states should be included in the model (some sorting pathways may be short and others long, and several compartments can share parts of the pathways). Thus, unlike traditional HMM learning tasks that focus on learning the transition and emission probabilities, for our model we also need to learn the set of states that are used in the sorting HMM.

2.3. A HMM for the sorting pathways problem

We will discuss the likelihood of our HMM in detail here (Fig. 1). The following description applies to using likelihood for motif features, but can be easily adapted to the case of binary motif features by removing the sequence variable S and include motif occurrences in the binary feature variables F. As discussed above, in our HMM model all proteins move from a single start state to their final compartment. For reasons that will become clear when talking about learning the parameters of the model, we associate each state in our model with a specific level. The root state is level 0, all compartment states are associated with the final level (T) and each intermediate state is associated with a specific level t ($0 < t < T$). The number of levels T is inferred from the data during structure initialization as described in section 2.6. We require that a state at level t can be reached from the root after exactly t transitions; connections that are more than one level apart move through several ''silent'' states so that transitions are only between adjacent levels (diamond-shaped states in Figure 1). Silent states only emit a ''background'' feature. Let X_t denote a hidden state at level t, $t = 1, 2, \cdots, T$ in a T-level model. The value of X_t can be one of J possible states, $X_t \in \{1, 2, \cdots, J\}$.

In addition to transition probabilities states are associated with emission probabilities. State X_t emits a feature index Z_t. Z_t can either be one of M motifs (represented as a likelihood score for each protein), or one of K binary features which include interactions with selected carriers, selected deterministic motif occurrences based on UniProt, or the background feature emitted by silent states. Hence $Z_t \in \{1, 2, \cdots M + K + 1\}$, where the motifs are indexed from 1 to M and the features are indexed from $M + 1$ to $M + K$.

Let S denote the sequence observed for each protein, F be the binary features from interaction databases and UniProt, and Y be the compartment assignments for a protein. The data likelihood of our HMM model (Fig. 1), is defined as:

$$\Pr(S, F, Y|\Theta) = \sum_{X_1} \cdots \sum_{X_T} \sum_{Z_1} \cdots \sum_{Z_{T-1}} \Pr(S, F, Y, X_1, \cdots X_T, Z_1, \cdots Z_{T-1}|\Theta)$$

These joint probabilities can be decomposed based on the HMM independence assumptions as follows:

$$\begin{aligned} &\Pr(S, F, Y, X_1, \cdots X_T, Z_1, \cdots Z_{T-1}|\Theta) \\ &= \Pr(X_1) \prod_{t=1}^{T-1} \Pr(X_{t+1}|X_t) \Pr(Z_t|X_t) \Pr(S|Z_1, \cdots Z_{T-1}) \\ &\quad \Pr(F|Z_1, \cdots Z_{T-1}) \Pr(Y|Z_T). \end{aligned} \tag{1}$$

The parameters of our HMM are the initial, transition and emission probabilities, $\Theta = (\pi, A, B)$, defined as

$$\pi_i = \Pr(X_1 = i), \; A_{ij} = \Pr(X_{t+1} = j|X_t = i), \; B_{ik} = \Pr(Z_t = k|X_t = i).$$

where π_i is the initial probability of transition from the root to state i, A_{ij} is the transition probability between state i and state j, and B_{ik} is the emission probabilities from state i to emission k. Since each state only transits to a small number of states and emits a small number of features, these matrices are sparse.

2.4. Defining the emission and transition probabilities for our model

As indicated above the feature observation includes the sequences and interactions selected carriers inferred by feature selection described above. Note that these observations are static and so may depend on all levels in the HMM. The emission probability for the sequence S is thus $\Pr(S|Z_1, \cdots Z_{T-1})$. Since probability depends on several motif models (one per level), which may be dependent (for example for overlapping motifs) and is thus computationally intractable given many combinations of motifs. As is commonly done (Sinha, 2006), we approximate this term by the product of the conditional probabilities of the sequence given an individual emission at each level: $\prod_{t=1}^{T-1} Pr(S|Z_t)$. Similarly we calculate the conditional probability of the binary features $\Pr(F|Z_1, \cdots Z_{T-1})$ using the product of the conditional probabilities of individual emissions (unlike for the sequence data this computation is exact since they are provided as independent events): $\prod_{t=1}^{T-1} Pr(F|Z_t)$. This leads to the more typical HMM model shown in Figure 1B.

To translate the sequence information to a probability we use the likelihood of the sequence given the motif, $\Pr(S|\lambda_k)$, where λ_k is the motif mode. We use a profile HMM model in this article, but any other probabilistic models would also work, for example, a position weight matrix (PWM) which specifies a weight for each amino acid at each motif position, assuming independence between positions. This likelihood is termed the motif score and indicates how well the sequence agrees with the motif model. For states emitting one of the binary features or the background feature, the likelihood of the sequence is $\Pr(S|\lambda_0)$, where λ_0 is the background model for which we use a 0th-order Markov model, which assumes that each position in the sequence are generated independently according to amino acid frequencies. Combined, the sequence likelihood is given by

$$\Pr(S|Z_t = k) = \begin{cases} \Pr(S|\lambda_k) & \text{if } 1 \le k \le M \\ \Pr(S|\lambda_0) & \text{if } M+1 \le k \le M+K+1 \end{cases} \tag{2}$$

The binary features observations, $F = (F_1, F_2, \cdots, F_K), F_k \in \{0, 1\}$ correspond to observed protein interactions and deterministic motifs as discussed above. As mentioned above, we assume independence in noisy observation of these features, which is a necessary simplification. This leads to

$$\Pr(F|Z_t = k) = \prod_{j=1}^{K} \Pr(F_j|Z_t = k)$$

The conditional probability of observing a feature F_j given an emission Z_t is

$$\Pr(F_j = 1|Z_t = k) = \begin{cases} \nu_j & \text{if } k \ne M+j \\ \nu_0 & \text{if } k = M+j \end{cases}, \quad 1 \le j \le K \tag{3}$$

where v_j is the probability of observing this interaction across all proteins in our dataset (background distribution), and $1 - v_0$ is the probability of false negatives (i.e., proteins that should go through this state but do not have this interaction/motif). Note that we need to use v_j since an interaction or a motif may be observed even if the corresponding feature is not emitted by one of the states since many interactions are not related to protein sorting but rather to another pathway in which this protein is a member.

The conditional probability of the compartment given the final state is denoted by: $\Pr(Y|X_T)$. If a single compartment is given for a protein, the bottom state X_T is known for that protein and so this probability is 1 for that compartment and 0 for others. If the training data contains multiple compartments for a protein, it is reflected by the given compartment likelihood $\Pr(Y = y|X_T = c)$, which is assumed to be uniform for all compartments listed for that protein. In other words, we consider multiple localization as uncertainty. For example, a protein might be considered to be 50% certain as one compartment and 50% certain as another compartment.

2.5. Approximation and feature levels

Unlike a typical HMM learning problem, the emission data we observe (sequence and interaction data) is static and so cannot be directly associated with any sequence of events. In addition, since our features are static, they can be emitted multiple times along the *same* path. However, if this happens the independence assumptions of HMMs are violated. Specifically, if a feature is emitted by a state in level t and then again by a

state in level $t + 1$ then it is not true anymore that the probability of emitting the feature given the state is independent of any emission events in previous states (since if it was emitted before, the protein can still emit it again). We thus constrain all features in our model so that each is only associated with a specific level and can only be emitted by states on that level. The level is determined in the initial structure estimation step discussed in the next section. Since no transitions are allowed between states on the same level no feature can thus be emitted more than once along the path and so the independence assumption holds. This requirement guarantees that the likelihood function obtained from the model presented in Figure 1B is a constant factor approximation of the likelihood function of our original model (Fig. 1A). See detailed proof at http://murphylab.web.cmu.edu/software/2010_RECOMB_pathway/.

2.6. Structure learning

In addition to learning the parameters (emission and transition probabilities) we also need to learn the set of states that should be included in our model. The learning algorithm is formally presented in Figure 2. We start by associating potential features (protein interactions and known motifs) with compartments. For a potential feature, we use the hypergeometric distribution to determine the significance of this association (by looking at the overlap between proteins assigned to each compartment and proteins that are associated with each of the features). We next identify a set of significantly associated compartments (p-value < 0.01 with Bonferroni correction) for each potential feature. Features that are significantly associated with at least one compartment are selected and the remaining features are removed.

After feature selection, we estimate an initial structure by using the association between features and compartments. All features that correspond to the same set of associated compartments are grouped and assigned to a single state, such that this state emits these features with uniform probability. These features are fixed to the level corresponding to the number of compartments they are significantly associated with and can only be emitted by states on that level (we tried optimizing these feature levels as part of the iterative learning process but this did not improve performance while drastically increasing run time). Initial transition between states is determined from the inclusion relationship of the set of compartments (states for which features are associated with more compartments are assigned to higher levels). We initially only allow transitions between two states where the second state contains features that are associated with a subset of the compartments of the first state. That is, the initial structure resembles a partially ordered set when the states are ordered by inclusion. The transition probability out of a state is also set to the uniform distribution. The number of levels of this structure, T, will be fixed throughout the structure search process.

Starting with this initial model, we use a greedy search algorithm which attempts to optimize the Bayesian information criterion (BIC), which is the negative data log likelihood plus a penalty term for model selection.

```
1. Estimate the associations between features and compartments
   using a hypergeometric test.
2. Select features significantly associated with at least one compartment.
2. Start with an initial structure estimated from associations
   between features and compartments.
3. While BIC score improves do
   a. For each level, create a candidate structure as follows.
      i. Add a node (state) at this level.
      ii. Link from all upper nodes and link to all lower nodes.
      iii. Run EM to optimize parameters.
      iv. Prune edges (transitions) rarely visited based on the parameters.
      v. Prune emissions rarely used based on the parameters.
      vi. Run EM again to adjust parameters.
   b. Create candidate structures by randomly splitting the state
      with largest number of out-transitions.
      i. Create a new state at the same level.
      ii. Each out-transition has 1/2 probability to be moved
          to the new state.
      iii. Copy the in-transitions to the new state.
      iv. Run EM to optimize parameters.
      v. Prune transitions and emissions rarely visited.
      vi. Repeat for a fixed number of times, e.g. the number of levels.
   c. Choose the candidate structure with highest BIC score.
   d. If improving, update to that structure; otherwise stop.
```

FIG. 2. Algorithm for structure search.

$$\mathrm{BIC} = -2 \log \Pr(\mathbf{S}, \mathbf{F}, \mathbf{Y}|\Theta) + |\Theta| \log N$$

where **S**, **F**, **Y** are the collection of sequences, feature observations, and compartments of the proteins in the training data. $\Theta = \pi, A, B)$ denote the parameters of the HMM. $|\Theta|$ is the number of parameters according to the structure, which is a function of the number of states and the number of transitions and emissions of each state. Complicated structures will have large $|\Theta|$ while simple structures will have small ones. N is the number of proteins in our training data. BIC is asymptotically consistent while Akaike information criterion (AIC) is not, and BIC is chosen particularly because we prefer sparser structures (Hastie et al., 2003). Since use of BIC can sometimes lead to overfitting, we compared the use of BIC to fourfold internal cross-validation for model selection. BIC is faster than internal cross validation and performed better on simulated data (see Section 3.1).

To improve the initial structure described above we perform two types of local moves at each search iteration: adding a new state and splitting the largest state. For each level, we try adding a state which is fully connected to all states in levels above and below it and emits all features on that level. We run standard EM algorithm (Dempster et al., 1977) to optimize the parameters of the model for all states (transition and emission probabilities). Transitions and emissions with probabilities lower than a specific threshold are pruned. Features not emitted by any states are also pruned, so the feature set becomes smaller and smaller. Then we run EM algorithm again because the parameters are changed. A candidate model and structure is created by this process for each level. We also try splitting the largest state, defined as the state with the largest number of out-transitions. A randomly chosen half of the out-transitions will be moved to a newly created state which shares the same in-transitions and emissions. As above we run EM algorithm, prune transitions and emission, and run EM algorithm again to obtain a candidate structure. We try this for a fixed number of times, usually the number of levels so that half of the local moves are adding and half are splitting. Among all candidate structures obtained by adding and splitting, the one with the highest BIC score is chosen. This procedure is repeated until the BIC score no longer improves.

3. RESULTS

3.1. Simulated data

We first tested our method using simulated data in order to determine how well it can recover a known underlying structure given only information on destinations, carriers and motifs. We manually created structures with 7, 14, 23, 25, and 31 states with multiple emitted features per state (for the structure of these models, see http://murphylab.web.cmu.edu/software/2010_RECOMB_pathway/). For each structure we simulate the probabilistic generative procedure and record the emitted features. 1,200 proteins are generated from the model, with varying levels of noise (leading to false positive and false negative features for proteins). We also tested various sizes of input sets with a fixed noise level.

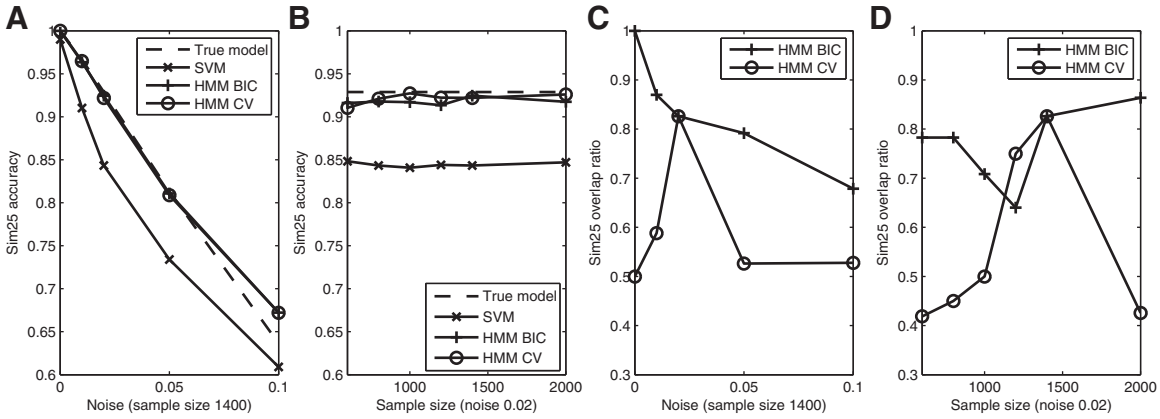

FIG. 3. **(A)** Testing error of simulated dataset generated from a structure with 25 states with varying levels of noise (false positive and false negative in features). The training sample size was fixed at 1400. **(B)** Testing error versus different training sample sizes. The noise level was fixed at 2%. **(C)** The ratio of overlapping nodes and edges between the learned model and the true model with varying levels of noise. The training sample size was fixed at 1400. **(D)** The ratio of overlapping nodes and edges with varying training sample sizes. The noise level was fixed at 2%.

3.1.1. Predicting protein locations. While it is not its primary goal, our method can provide predictions regarding the final localization of each protein. For each training dataset, we therefore generated a test dataset with 4,000 proteins from the same model and evaluated the accuracy of predicting protein localization for the test data using the structure and model learned by our method. Our method is compared to predictions made by the true model (note that due to noise, the true model can make mistakes as well) and by a linear support vector machine (SVM) learned from the training data using the features associated with each protein. Prediction accuracy on the 25-states dataset is shown in Figure 3, and the accuracy of other simulated datasets are available at http://murphylab.web.cmu.edu/software/2010_RECOMB_pathways/. As can be seen, when noise levels are low, our model performs well and its accuracy is similar to that obtained by the true model for both simple and more complicated models. Both the learned model and the true model outperform SVM which does not try to model the generative process in which proteins are sorted in cells relying instead on a one versus all classification strategy. We compare model selection based on BIC versus fourfold internal cross validation. BIC achieved similar accuracy with less computation and matched the true structure better.

3.1.2. Recovering the true structure. To quantitatively evaluate how well a learned structure resembles the true structure, we use the graph edit distance to measure their topological similarity (Gao et al., 2010). First we need to match the nodes in a learned structure to a node in the true structure. We run the Viterbi algorithm on proteins in the testing data, and count the state co-occurrence matrix W whose elements W_{ij} is the co-occurrence of state i in the learned model and state j in the true model (i.e., the number of proteins in which the two states i and j occur in the Viterbi path inferred by the two models). The optimal one-to-one matching M, denoted as a set containing pairs of matched state indexes, can be found by running the Hungarian algorithm on the co-occurrence matrix W optimizing the objective function $\sum_{(i,j)\in M} W_{ij}$.

With the optimal matching, we use the maximum common subgraph (MCS) and minimum common supergraph in the graph edit distance methodology to quantify similarity between two structures. Given two graphs G_1 and G_2, let $\hat{G}$ and $\check{G}$ be the MCS and minimum common supergraph of G_1 and G_2. Denote $|G|$ as the size, or the number of edges and nodes of a graph, we define the overlap rate as $|\hat{G}|/|\check{G}|$ (i.e., the percentage of overlapping edges and nodes). The overlap rate comparing to the true model on the 25-states dataset is shown in Figure 3C. Structural comparison on other datasets is available on the supporting website. As can be seen, our algorithm successfully recovers the correct structure in all cases with 0% noise. As the noise increases the accuracy decreases. However, even for very high levels of noise the two models share a substantial overlap (around 40% of states and transitions could be matched).

3.2. Yeast data

We next evaluated our method using subcellular locations of yeast proteins derived from fluorescence microscopy (the UCSF yeast GFP dataset [Huh et al., 2003]). This dataset contains 3,914 proteins that were manually annotated, based on imaging data, to 22 compartments. We collected the features from the following sources. Protein-protein interaction (PPI) data was downloaded from BioGRID (BiG) (Stark et al., 2006). For deterministic motifs we use the annotated occurrences of InterPro (Mulder et al., 2003) domains and the following three signal sequences listed on UniProt (Bairoch et al., 2005):

1. *Signal peptides:* UniProt defines this sequence feature based on the literature or consensus vote of four programs, SignalP, TargetP, Phobius and Predotar.
2. *Transmembrane region:* UniProt annotates a sequence with this feature either based on literature or consensus vote of four programs, TMHMM, Memsat, Phobius and Eisenberg.
3. *GPI anchor:* UniProt annotation for this feature either relies on literature or prediction by the program big-PI.

The above features are filtered by a hypergeometric test to identify features with a significant association with a final destination (p-value < 0.01 with Bonferroni correction) before learning the model.

To extract novel motifs associated with localization, we downloaded protein sequences from UniProt (Bairoch et al., 2005) and ran generative and discriminative HMM motif finder (Lin et al., 2011). We extract 20 motifs for each compartment, and compared setting all to length 4 versus setting the length to range from 3 to 7. The performance in all following evaluations are similar and we show results based on

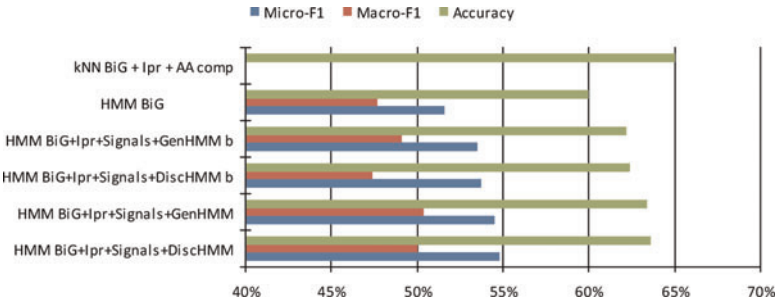

FIG. 4. The accuracy of predicting the final subcellular location. For kNN, we use the reported accuracy based on PPI information from BiG, deterministic InterPro motif annotation from UniProt, and amino acid composition of different length, gaps, and chemical properties using leave one out cross validation (Lee et al., 2008). For HMM, we also show micro-averaging and macro-averaging F1 score in 10-fold cross validation. The features for HMM include InterPro and BiG, and three signal sequences from UniProt. The novel motifs are learned using generative or discriminative HMM of length 4, represented by likelihood and binary features (GenHMM/DiscHMM b).

motif length as 4. We will compare using likelihood and binary occurrence for motif features. For binary motif occurrence, a motif is considered present if posterior probabilities of the begin state and the end state of the motif are both greater than 0.9 (detail in Lin et al., 2011).

3.2.1. Predicting protein locations. As with the simulated data, we first evaluated the accuracy of predicting the final subcellular location for each protein. This provides a useful benchmark for comparison to all other computational methods for which this is the end result. The performance is evaluated by 10-fold cross-validation. In each fold both feature selection and motif finding are restricted to the training data without accessing the testing data. We use three conventional measure in information retrieval: the accuracy, micro-averaging F1 and macro-averaging F1 (Yang and Liu, 1999). For the accuracy, a prediction is considered correct if it matches any of the true locations. The F1 score is the harmonic mean of precision and recall (Van Rijsbergen, 1979). Micro-averaging takes the average of the F1 score over all proteins, giving each protein an equal weight; in other words, the classes are weighted by their sizes. Macro-averaging takes the average of the score over classes, giving each class an equal weight. Including macro-averaging F1 ensures smaller classes are not ignored since other measures are dominated by large classes. The result is shown in Figure 4. We compared our method with the k-Nearest Neighbors (kNN) from Lee et al. (2008) which was shown by the authors to outperform other methods. As can be seen in Figure 4, PPI information (BiG) provides the major contribution for accurate predictions while InterPro motifs do not contribute as much. This agrees with previous studies (Scott et al., 2005; Lee et al., 2008). When adding more features, the performance improves and the best result is achieved using all features. Note that the accuracy of our method is very close to that of the kNN method. However, it is important to note that our method performs the much harder task of simultaneously learning the sorting pathways as well as predicting locations. Unlike these prior methods, our method correctly determines pathways and not just end points. This is an important contribution of the method that is achieved while not compromising prediction accuracy.

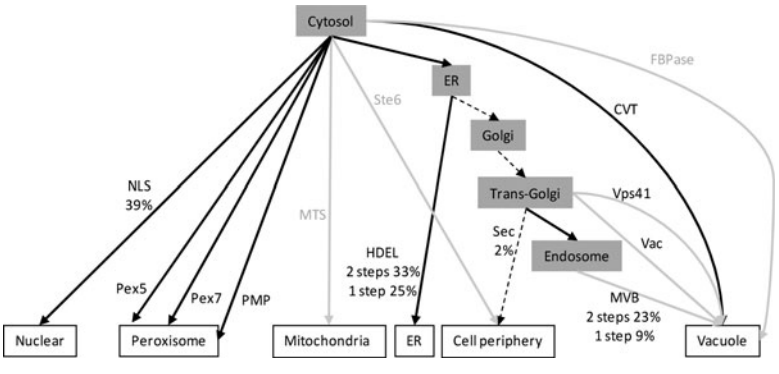

FIG. 5. Protein sorting pathways collected from the literature. Each pathway is a path from cytosol to a compartment at the bottom, consisting of one or more steps (the links) that transport proteins between intermediate locations. Each step has a list of carriers and motifs responsible for the transportation by which we can verify whether the pathway is recovered. Shaded links denote steps whose carriers are underrepresented on BiG (covering less than 5% of proteins transported to the corresponding compartment in the GFP dataset). Dashed lines denote steps taken by default without specific carriers. The percentage under pathway name is the protein sorting precision when the pathway is recovered, as described in Table 2.

Table 1. Pathway Recovery Results of Structure Learned from Different Feature Sets

Features	*Pathway recovery*	*Inferred protein path*
HMM BiG	5.9 (4.7–8.0)	7% (4–10%)
HMM BiG + Ipr + Signals	7.2 (5.7–8.7)	8% (6–11%)
HMM BiG + Ipr + Signals + GenHMM b	6.2 (4.3–7.7)	8.4% (6–11%)
HMM BiG + Ipr + Signals + DiscHMM b	6.2 (5.3–7.3)	8.4% (6–11%)
HMM BiG + Ipr + Signals + GenHMM	7.7 (6.7–8.7)	17.9% (13–23%)
HMM BiG + Ipr + Signals + DiscHMM	7.7 (6.7–8.7)	19% (15%–23%)

The precision of inferred protein path is also listed here. Mean, minimum, and maximum among the 10 folds are shown.

3.2.2. Evaluation of the learned structure. To evaluate the accuracy of the learned structure, we collected information about known sorting pathways from the literature. We were able to find information regarding 13 classical and non-classical sorting pathways (pathways followed by a minor fraction of proteins or that differ from the first discovered pathway are often referred to as non-classical pathways). For each of these pathways, we identified a set of carriers or motifs that govern the pathway and, when available, the set of proteins that are predicted to use this pathway. Figure 5 presents the pathways we collected from the literature. For example, the classical HDEL pathway into ER has two steps. In the first, proteins with signal peptide (SP) are introduced into this pathway by the SRP complex. In the second, proteins with the HDEL motif are retained in ER by interaction with proteins Erd1 and Erd2. The full list of carriers and motifs for these pathways is available at http://murphylab.web.cmu.edu/software/2010_RECOMB-pathways/.

We first wanted to check if the databases we used for obtaining features contain the carrier information for the literature pathway. We filtered pathways for which carrier information in the BIG database did not contain the genes associated with the pathway (and thus no method can identify this pathway based in this input data) leaving 10 pathways that could, in principal, be recovered by computational models. Sorting steps that were filtered out in this way are represented as shaded links in Figure 5.

To determine whether we accurately recovered a pathway in our model, we looked at the carriers and motifs that are associated with that pathway in the literature. A step in a literature pathway can be matched to a state if the state emits any carrier or motif in that step. A known pathway is considered recovered in a learned structure if its steps can be matched to the states along a path from the root to the compartment to which it leads. A pathway is partially recovered if only some of its steps can be matched. For example, the MVB pathway (Fig. 5) is only partially recovered (66.7%) because the third step does not have a well-represented carrier in the data sources. The numbers of recovered pathways for different sets of features are listed in Table 1. The ranges correspond to the different folds in our cross validation analysis. Fractions represent partial matches as discussed above. When using the full set of input features our algorithm is able

Table 2. Recovery and Protein Sorting Results of Each Pathway Using the Features BiG + InterPro + Signals + DiscHMM 4

Compartment	*Pathway (#proteins)*	*Recovery (folds)*	*Steps*	*Sorting*
Nucleus	NLS(15)	10/10	all	39%
Peroxisome	Pex5	1/10	all	
	Pex7	10/10	all	
	PMP	9/10	all	
ER	HDEL(11)	10/10	SP + HDEL	33%
			SP	25%
Cell periphery	Sec(28)	10/10	SP	2%
Vacuole	Vac	10/10	SP	
	MVB(9)	10/10	SP + MVB	23%
			SP	9%
	Vps41	10/10	SP	
	CVT	10/10	all	

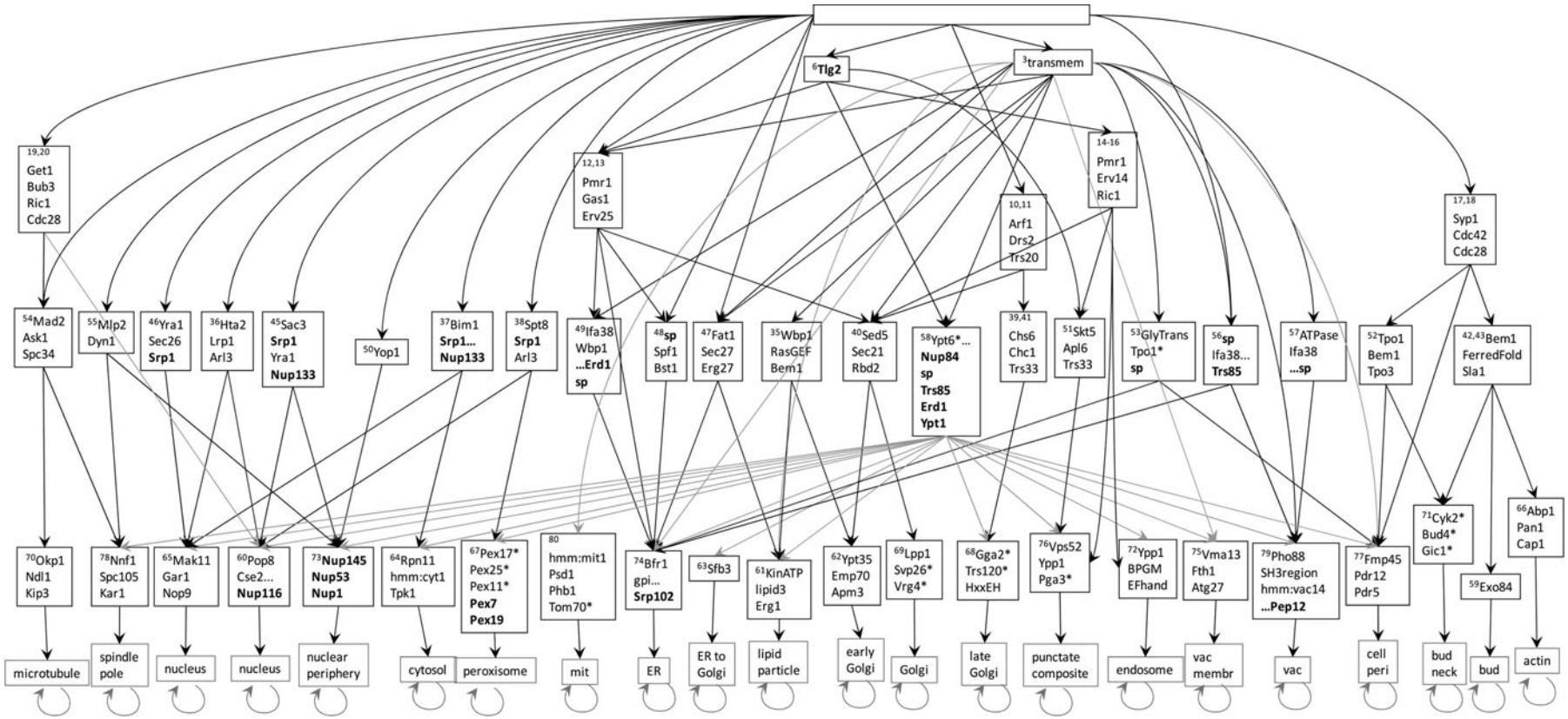

FIG. 6. The HMM state space structure learned by our method that corresponds to potential protein sorting pathways. A state is represented by a block; its transitions are shown as arrows and its top three emitting features are listed inside the block. The sparse transition and emission probabilities are omitted here. The initial state probabilities are denoted as arrows from the root block at the top. The bottom states are the final destination compartments. Some transitions are shaded only because of visual clarity, including transitions across levels or from and to the highly connected state (state 58). While silent states are not explicitly displayed (to remove clutter) they are actually implicitly present. Any time an edge jumps more than one level it is going through silent state(s). For example, the right most edge coming out of the root goes through a silent state in the first level. Carriers and motifs that matches our literature pathway collection are shown in boldface; other features potentially related to protein trafficking according to SGD are marked with an asterisk.

to recover roughly 80% of known pathways. Most of these pathways are recovered in all 10 folds (Table 1). Note that because some carriers do not appear in our database not all steps in all pathways can be matched and the best possible recovery is 8.7. Thus, the 7.7 recovery obtained is very close to optimal.

For example, because of lack of evidence (the motif and carrier detection steps did not find the Vam3, Vam7, or the Vps41 features), the classical vacuole import pathway (Vac in Fig. 5) and the alternative Vps41 pathway can only be 50% recovered (each missing a step). For both, the step of signal peptide (SP) is accurately found, but alternative motifs/carriers are selected to route proteins to the vacuole or cell periphery.

We further collected lists of proteins indicated as following specific pathways in the literature for four of the pathways, NLS, HDEL, Sec, and MVB, and tested whether the recovered pathways indeed sort proteins on the correct path to the correct destination (allowing close compartments as above). For each protein, we use the Viterbi algorithm to infer the highest probability path of states the protein is expected to follow according to our learned model, and compare the Viterbi path to the known pathways. Again counting partial match of a multi-step pathway as above, on average using all features results in correctly assigning 21% of 63 proteins. Focusing on a representative feature set, detailed protein path results for each pathway are also given in Table 2. The recovered NLS pathway sorted 39% of proteins correctly, and the recovered HDEL pathway sorted 33% correctly but sorted the other 25% via SP. Similarly the recovered MVB pathway sorted 23% to go through two of the three steps (SP and MVB) and other 9% to one of the three steps. The recovered Sec pathway only sorted 2% of the proteins to go through SP and end at cell periphery. However, this was due to the fact that while 17 of the 28 proteins collected from literature as being secreted were included in the GFP dataset, the majority are labeled as ER and vacule, and none are labeled as cell periphery. Overall the GFP dataset include 40 out of the 63 proteins whose pathway is known, of which only 28% are labeled in agreement with our lierature survey.

It is important to note that our analysis of the learned structure may underestimate its accuracy, since it may have recovered correct pathways that could not be verified due to insufficient detection of relevant motifs or carriers in the input data.

Figure 6 shows one of the learned structures obtained using all features. Besides carriers and motifs included in our literature pathway collection (marked as boldface), many other features were found that are also known to participate in protein trafficking as curated in SGD (Cherry et al., 1998) (marked with an asterisk). For those compartments not covered by our collection of known pathways, the general topology of this structure agrees with our basic understanding of cell biology. For example, microtubule share a step with spindle pole, which in turn share a step with nuclear periphery, and cell periphery share steps with bud neck, which in turn share steps with bud and actin.

4. DISCUSSION

The goal of this research is to propose hypotheses about protein sorting mechanisms, not just to make predictions. We propose, for what we believe is the first time, a method to learn sorting pathways from protein localization annotation, based on co-occurrence of interacting partner and sequence motif. Our method is able to recover a significant part of known pathways collected from the literature, and to infer the correct path of proteins known to follow these pathways.

Using a HMM, naturally simulates the transportation path of a protein among unobserved intermediate states. Although the path is unobserved, the most likely one can be inferred by the Viterbi algorithm of the HMM based on observed features. The model is probabilistic and returns a distribution of possible compartments, instead of a single predicted compartment. Proteins that are targeted to more than one compartment in the training data can be handled by treating multiple localization as uncertainty. While the method has been successful, an HMM-based approach also suffers from a number of limitations. The input data used by our method is static while HMM expects sequential data. This requires us to rely on a number of assumptions including limiting each of the features to a unique level, and assuming independence between the features. The structure search algorithm requires substantial computation, since the EM algorithm must be run every time a candidate structure is being tried. Improving the search strategy is a direction for future work. Another issue we would like to address is the inference of the actual location of the intermediate states. For example, we might associate an internal state with the ER or Golgi.

To determine such locations, we would need to tie the model to literature and try to identify overlaps which can be generalized.

Given that the sorting routes taken by many proteins are currently unknown, the most important part of our work is the potential to identify novel pathways. In this regard, we note that, just like hand-constructed pathways, any novel putative pathways contained in our learned model can be readily tested experimentally by perturbing motifs and/or carriers. An additional advantage of building comprehensive sorting models is that potential inconsistencies in canonical models can be identified and experiments performed to resolve them.

ACKNOWLEDGMENTS

We would like to thank Jennifer Bakal for programming support. This work was supported in part by the NIH (grant R01 GM075205).

DISCLOSURE STATEMENT

No competing financial interests exist.

REFERENCES

Bairoch, A., Apweiler, R., Wu, C.H., et al. 2005. The Universal Protein Resource (UniProt). *Nucleic Acids Res*. 33, D154–D159.

Bannai, H., Tamada, Y., Maruyama, O., et al. 2002. Extensive feature detection of N-terminal protein sorting signals. *Bioinformatics* 18, 298–305.

Barbe, L., Lundberg, E., Oksvold, P., et al. 2008. Toward a confocal subcellular atlas of the human proteome. *Mol. Cell Proteomics* 7, 499–508.

Bebek, G., and Yang, J. 2007. PathFinder: mining signal transduction pathway segments from protein-protein interaction networks. *BMC Bioinform.* 8, 335.

Bendtsen, J.D., Jensen, L.J., Blom, N., et al. 2004a. Feature-based prediction of non-classical and leaderless protein secretion. *Protein Eng. Des. Sel*. 17, 349–356.

Bendtsen, J.D., Nielsen, H., von Heijne, G., et al. 2004b. Improved prediction of signal peptides: SignalP 3.0. *J. Mol. Biol*. 340, 783–795.

Chen, S.C., Zhao, T., Gordon, G.J., et al. 2007. Automated image analysis of protein localization in budding yeast. *Bioinformatics* 23, i66–i71.

Cherry, J.M., Adler, C., Ball, C., et al. 1998. SGD: *Saccharomyces* genome database. *Nucleic Acids Res.* 26, 73–79.

Chou, K.-C.C., and Shen, H.-B.B. 2008. Cell-PLoc: a package of Web servers for predicting subcellular localization of proteins in various organisms. *Nat. Protocols* 3, 153–162.

Cohen, A.A., Geva-Zatorsky, N., Eden, E., et al. 2008. Dynamic proteomics of individual cancer cells in response to a drug. *Science* 322, 1511–1516.

Covert, M.W., Knight, E.M., Reed, J.L., et al. 2004. Integrating high-throughput and computational data elucidates bacterial networks. *Nature* 429, 92–96.

Dale, J., Popescu, L., and Karp, P. 2010. Machine learning methods for metabolic pathway prediction. *BMC Bioinform.* 11, 15.

De Strooper, B., Beullens, M., Contreras, B., et al. 1997. Phosphorylation, subcellular localization, and membrane orientation of the Alzheimer's disease-associated presenilins. *J. Biol. Chem.* 272, 3590–3598.

Dempster, A.P., Laird, N.M., and Rubin, D.B. 1977. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Ser. B Methodol.* 39, 1–38.

Emanuelsson, O., Nielsen, H., Brunak, S., et al. 2000. Predicting subcellular localization of proteins based on their N-terminal amino acid sequence. *J. Mol. Biol*. 300, 1005–1016.

Emanuelsson, O., Brunak, S., von Heijne, G., et al. 2007. Locating proteins in the cell using TargetP, SignalP and related tools. *Nat. Protocols* 2, 953–971.

Fischer, E., and Sauer, U. 2005. Large-scale in vivo flux analysis shows rigidity and suboptimal performance of *Bacillus subtilis* metabolism. *Nat. Genet.* 37, 636–640.

Gao, X., Xiao, B., Tao, D., et al. 2010. A survey of graph edit distance. *Patt. Anal. Appl.* 13, 113–129.

Gladden, A.B., and Diehl, A.A. 2005. Location, location, location: the role of cyclin D1 nuclear localization in cancer. *J. Cell. Biochem.* 96, 906–913.

Hastie, T., Tibshirani, R., and Friedman, J.H. 2003. *The Elements of Statistical Learning*. Springer, New York.
Horton, P., Park, K.J., Obayashi, T., et al. 2007. WoLF PSORT: protein localization predictor. *Nucleic Acids Res*. 35, W585–W587.
Huh, W.K., Falvo, J.V., Gerke, L.C., et al. 2003. Global analysis of protein localization in budding yeast. *Nature* 425, 686–691.
Kau, T.R., Way, J.C., and Silver, P.A. 2004. Nuclear transport and cancer: from mechanism to intervention. *Nat. Rev. Cancer* 4, 106–117.
Lee, K., Chuang, H.-Y., Beyer, A., et al. 2008. Protein networks markedly improve prediction of subcellular localization in multiple eukaryotic species. *Nucleic Acids Res.* 36, e136.
Lin, T.H., Murphy, R.F., and Joseph, Z.B. 2011. Discriminative motif finding for predicting protein subcellular localization. *IEEE/ACM Trans. Comput. Biol. Bioinform.* 8, 441–451.
Lodish, H., Berk, A., Matsudaira, P., et al. 2003. *Molecular Cell Biology*, 5th ed. W.H. Freeman, New York.
Mulder, N.J., Apweiler, R., Attwood, T.K., et al. 2003. The InterPro database, 2003 brings increased coverage and new features. *Nucleic Acids Res*. 31, 315–318.
Nair, R., and Rost, B. 2005. Mimicking cellular sorting improves prediction of subcellular localization. *J. Mol. Biol*. 348, 85–100.
Newberg, J.Y., Li, J., Rao, A., et al. 2009. Automated analysis of human protein atlas immunofluorescence images. *Proc. 2009 IEEE Int. Symp. Biomed. Imaging* 1023–1026.
Osuna, E.G., Hua, J., Bateman, N.W., et al. 2007. Large-scale automated analysis of location patterns in randomly tagged 3T3 cells. *Ann. Biomed. Eng*. 35, 1081–1087.
Pierleoni, A., Martelli, P.L., Fariselli, P., et al. 2006. BaCelLo: a balanced subcellular localization predictor. *Bioinformatics* 22.
Purdue, P.E., Takada, Y., and Danpure, C.J. 1990. Identification of mutations associated with peroxisome-to-mitochondrion mistargeting of alanine/glyoxylate aminotransferase in primary hyperoxaluria type 1. *J. Cell Biol.* 111, 2341–2351.
Rashid, M., Saha, S., and Raghava, G.P. 2007. Support vector machine–based method for predicting subcellular localization of mycobacterial proteins using evolutionary information and motifs. *BMC Bioinform.* 8, 337.
Rubartelli, A., and Sitia, R. 1997. R. Secretion of mammalian proteins that lack a signal sequence. *Unusual Secretory Pathways: From Bacteria to Man.* R.G. Landes, Austin, *TX*.
Ruths, D., Nakhleh, L., and Ram, P.T. 2008. Rapidly exploring structural and dynamic properties of signaling networks using PathwayOracle. *BMC Syst. Biol.* 2.
Scott, J., Ideker, T., Karp, R.M., et al. 2006. Efficient algorithms for detecting signaling pathways in protein interaction networks. *J. Comput. Biol.* 13, 133–144.
Scott, M.S., Calafell, S.J., Thomas, D.Y., et al. 2005. Refining protein subcellular localization. *PLoS Comput. Biol*. 1, 6.
Shatkay, H., Höglund, A., Brady, S., et al. 2007. SherLoc: high-accuracy prediction of protein subcellular localization by integrating text and protein sequence data. *Bioinformatics* 23, 1410–1417.
Shen, Y.-Q., and Burger, G. 2007. ''Unite and conquer'': enhanced prediction of protein subcellular localization by integrating multiple specialized tools. *BMC Bioinform.* 8, 420.
Sinha, S. 2006. On counting position weight matrix matches in a sequence, with application to discriminative motif finding. *Bioinformatics* 22, e454–e463.
Skach, W.R. 2000. Defects in processing and trafficking of the cystic fibrosis transmembrane conductance regulator. *Kidney Int.* 57, 825–831.
Stark, C., Breitkreutz, B.J., Reguly, T., et al. 2006. BioGRID: a general repository for interaction datasets. *Nucleic Acids Res.* 34, D535–D539.
Van Rijsbergen, C.J. 1979. *Information Retrieval,* 2nd ed. Department of Computer Science, University of Glasgow.
Yang, Y., and Liu, X. 1999. A re-examination of text categorization methods. *Proc. SIGIR '99* 42–49.

Address correspondence to:
Dr. Robert F. Murphy
Lane Center for Computational Biology
Carnegie Mellon University
Pittsburgh, PA 15213

E-mail: murphy@cmu.edu