BAYESIAN DATA INTEGRATION: A FUNCTIONAL PERSPECTIVE

Curtis Huttenhower and Olga G. Troyanskaya*

Department of Computer Science, Lewis-Sigler Institute for Integrative Genomics,Princeton University
Princeton, NJ 08544, USA
**Email: ogt@cs.princeton.edu*

Accurate prediction of protein function and interactions from diverse genomic data is a key problem in systems biology. Heterogeneous data integration remains a challenge, particularly due to noisy data sources, diversity of coverage, and functional biases. It is thus important to understand the behavior and robustness of data integration methods in the context of various biological functions. We focus on the ability of Bayesian networks to predict functional relationships between proteins under a variety of conditions. This study considers the effect of network structure and compares expert estimated conditional probabilities with those learned using a generative method (expectation maximization) and a discriminative method (extended logistic regression). We consider the contributions of individual data sources and interpret these results both globally and in the context of specific biological processes. We find that it is critical to consider variation across biological functions; even when global performance is strong, some categories are consistently predicted well, and others are difficult to analyze. All learned models outperform the equivalent expert estimated models, although this effect diminishes as the amount of available data decreases. These learning techniques are not specific to Bayesian networks, and thus our conclusions should generalize to other methods for data integration. Overall, Bayesian learning provides a consistent benefit in data integration, but its performance and the impact of heterogeneous data sources must be interpreted from the perspective of individual functional categories.

1. INTRODUCTION

As more sources of high-throughput biological data have become available, many efforts have been made to automatically integrate heterogeneous data types for the prediction of protein function and interactions[1-3]. Several of these systems have focused solely on data representation and presentation as a means of allowing efficient storage, retrieval, and manipulation by domain experts[4-6]. We look instead at the process of fully automated prediction of genetic interaction and functional linkages, which has to date been addressed by four primary methods: decision trees[7, 8], support vector machines[9], graph-based methods[10-13], and Bayesian networks[14, 15].

When applying any of these techniques to the problem of data integration, it is important to consider the effects of the method's parameters and assumptions on the resulting biological predictions. If two genes are predicted to be interacting or functionally related, are they necessarily related under all conditions, or do they interact only under specific circumstances or within one or two narrowly defined processes? Similarly, given some set of heterogeneous experimental data to be integrated, they may easily possess significant differences in reliability, magnitude, and coverage of various biological functions. Intuitively, one might expect correlations in microarray expression values to indicate a different, less reliable relationship between proteins than direct binding in an immunoprecipitation experiment. These differences could also be more pronounced in specific functional categories; for example, regulatory mechanisms such as phosphorylation signaling pathways will not be visible in microarray experiments.

Thus, it is important to examine the behavior of data integration methods from the perspective of diverse biological functions. We focus our analysis on Bayesian networks, considering both generative and discriminative learning frameworks. Bayesian networks provide an interpretable framework for examining machine learning across a variety of biological functions, experimental data types, and network parameters. We thus investigate the characteristics of Bayesian data integration by breaking performance down with respect to specific biological processes drawn from the Gene Ontology[16]. We further decompose the network's behavior by examining its dependence on each of its heterogeneous data sources, and we examine the effect of network structure by comparing a multilayer network structure to a single-layer naive Bayesian classifier. Finally, both of these network structures can be parameterized with expert estimates, with probabilities learned generatively

[*] Corresponding author.

through expectation maximization (EM)[17], or with a discriminative model learned using Extended Logistic Regression (ELR)[18].

Varying these parameters allows us to evaluate the predictive power of expert estimation, generative learning, and discriminative learning for each configuration of the Bayesian model. This results in a detailed comparison of functional predictions for individual ontology terms, network parameters, and data sources. All of these facets of functional prediction are nonspecific to Bayesian networks, allowing our conclusions to generalize to other heterogeneous data integration techniques.

2. RESULTS

We evaluated per-function Bayesian network performance over four variables: overall network structure, experimental data types, conditional probability sources (expert estimated, generatively learned, or discriminatively learned), and stability of learned parameters over varying initial conditions. The system described in Troyanskaya et al[15] acted as a basis, providing a predefined multilevel network structure and fixed conditional probabilities estimated by a consensus of domain experts. We integrated a variety of data sources heterogeneous both in their experimental origin (i.e. physical binding versus pathway membership) and in their computational behavior (i.e. discrete versus continuous data).

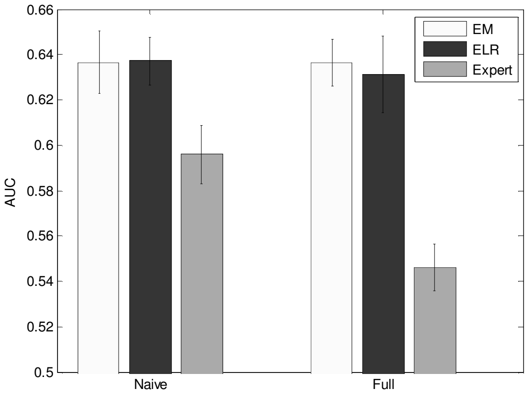

Fig. 1. Areas under sensitivity/specificity curves over all data for the six primary network configurations: three parameter estimation methods (expert estimation in blue, generative EM in beige, and discriminative ELR in red) and two network structures (naive and full). Evaluations are over a five-fold cross validation. Both generative and discriminative learning show a substantial improvement over expert estimation, particularly in the full network.

In all cases, evaluation was performed against a gold standard of functional relationships derived from *S. cerevisiae* GO annotations (experiments with the MIPS functional hierarchy[19] generated similar results). Our overall results appear in Figure 1, which shows that both generative and discriminative learning improve upon expert estimated probabilities (particularly in the full network). However, a global evaluation such as this cannot reveal how well each model predicts interactions within specific biological processes. For example, will these predictions be helpful to a biologist interested in DNA replication, or is the learned performance due to improvements in other functional areas?

To address questions such as this, we examined individual areas under receiver operating characteristics (ROC) curves (AUCs) for each term in a subset of the Gene Ontology (see Methods). This made it possible to monitor performance within individual functional categories as network parameters varied. Figure 2 displays the results, and it is important to note that performance varies far more across functional categories than it does across network parameters, learning techniques, or data sets. This means that for any aggregate, cross-functional evaluation to remain biologically relevant, it is necessary to keep in mind that it may represent average behavior based on strong performance in only a few functional areas. For example, without functional analysis such as that in Figure 2, it would be difficult to determine that even the most accurate predictions included in Figure 1 are often inapplicable to RNA processing terms (purple cluster, Figure 2). Conversely, we might be more inclined to trust predictions for uncharacterized genes paired with known genes annotated to metabolic terms (red cluster, Figure 2).

Interestingly, network structure proved to have little effect on learned networks, while it greatly impacted the performance of expert estimated parameters. Experiments were performed on two network structures, a slight modification of that proposed in Troyanskaya et al[15] and a naive Bayesian simplification of this model (Figure 5). Hidden nodes

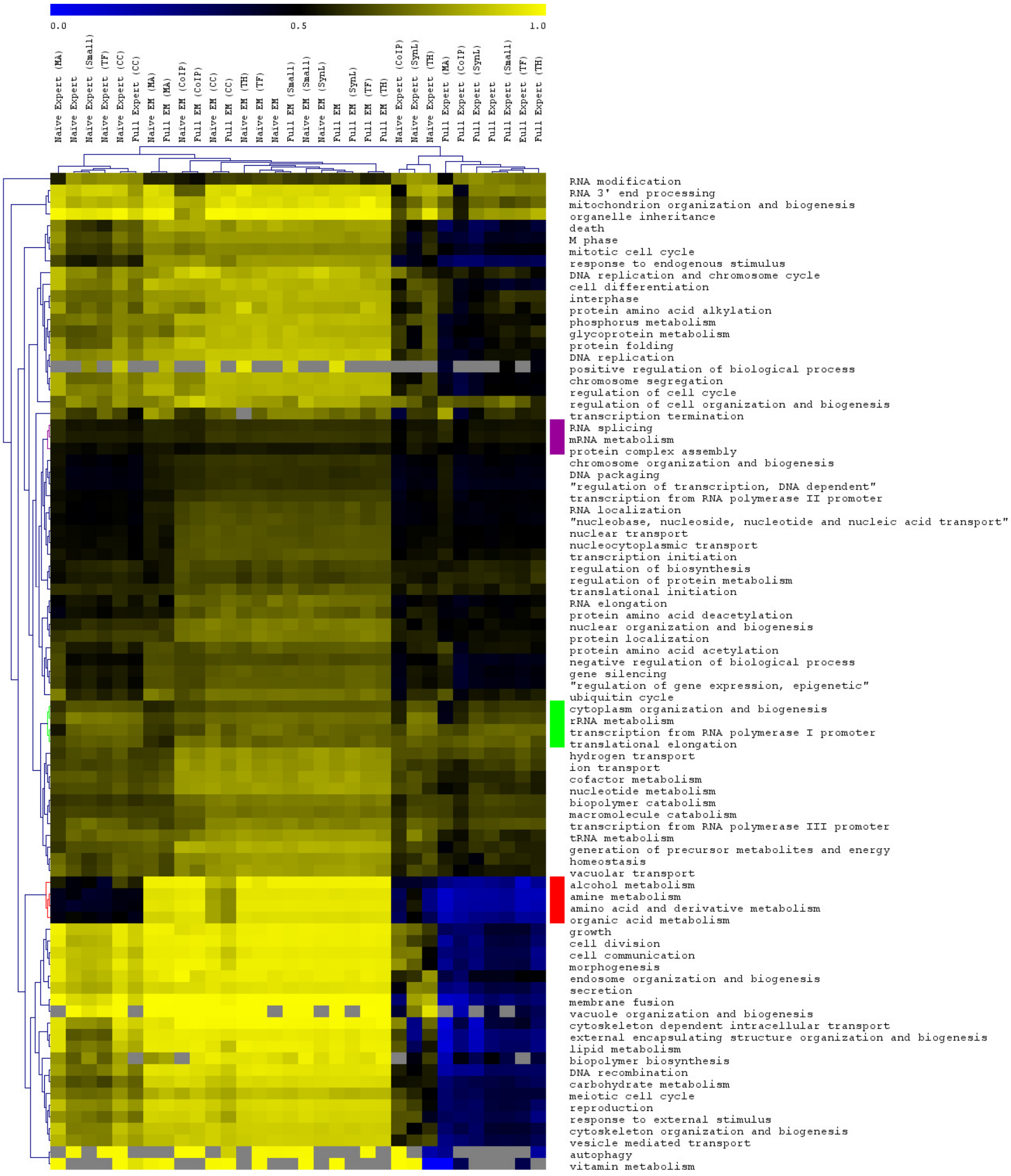

Fig. 2. A heat map of pairwise functional relationship prediction within individual Gene Ontology processes. Yellow indicates an AUC above random, blue below, and black exactly random (AUC = 0.5). Each column represents a network configuration (a combination of structure, parameter source, and data set presence), and each row represents a biological function. ELR networks perform similarly to their EM counterparts and have been omitted for clarity. Grey cells indicate network configurations for which fewer than ten gene pairs were available for evaluating a functional category. Marked clusters indicate terms that are consistently poorly predicted (purple), predicted well (green), and predicted well only by learned networks (red).

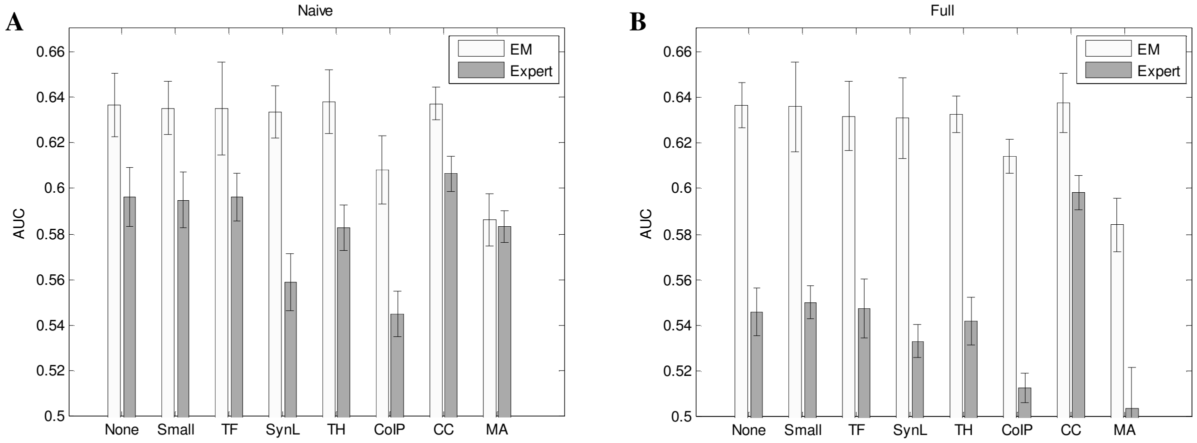

Fig. 3. A) A comparison of functional predictions with the naive network structure using EM and expert parameter estimation, removing each data set in turn. Networks with complete input were trained and evaluated using all available data sets; each other network had either one data set (cellular component (CC), coimmunoprecipitation (CoIP), transcription factor binding (TF), two-hybrid (TH), or microarrays (MA)) removed or all small data sets (biochemical association, dosage lethality, purified complexes, reconstructed complexes, and synthetic rescue) removed as a single unit. All evaluation was performed using only gene pairs with at least two data types available so as two allow evaluation with any one data set removed. AUCs are averages across five-fold cross validation. B) A comparison (as in part a) using the full network structure. Expert estimated parameters produce markedly worse performance with the full structure relative to the naive structure, and in both cases and across all data sets they are less accurate than learned parameters.

provide a way of relating similar data types and taking advantage of additional network parameters; a naive Bayesian assumption limits both the complexity and the representational power of the network[17]. The learned networks gained little from the additional parameters available in the full network, and its complexity hampered the predictive power of the expert estimated parameters (Figure 1).

Given these network configurations, it is of interest to see how much information is being contributed by different data sources. The number of pairs in the data sources varied from a few dozen to several million, and particularly in light of the potential sensitivity of learning algorithms to their training data, it is to be hoped that performance would degrade gracefully relative to the number of training examples. Figure 3 contains performance results for both network structures using expert estimated and learned network parameters. ELR and EM learning both performed essentially equivalently; they were largely unaffected by network structure, and their performance dropped off only with the removal of the largest or most informative data sets. The expert estimated probabilities proved to be much less effective when using the full network structure, but they were affected only minimally by the removal of particular data sets.

We next focused specifically on the robustness of the network's predictions in the face of variations in input data and learning characteristics. In the case of learned networks, the choice of initial probability values could conceivably influence the point to which the network converged after learning. To ensure that this was not the case, Figure 4A demonstrates the performance of the two network structures using randomized probability tables as initial parameters. Variation is small for both learning methods and network structures, justifying our use of expert estimated probabilities for initialization.

Similarly, both learned and expert estimated networks could be susceptible to fluctuations in individual probability tables or their corresponding input data sets. To investigate this possibility, we randomized the conditional probability tables for each of four nodes in the full learned network: the root "Functional Relationship" node, "Microarray Correlation," the hidden "Genetic Association" node, or the "Yeast Two-Hybrid" leaf node (see Methods). This resulted in the relative performances seen in Figure 4B. Performance degrades gracefully and roughly in proportion to the data set effects seen in Figure 3. Randomizing the "Functional Relationship" prior will not change the relative order of predictions and, as expected, leaves the performance-recall curve largely unchanged. Randomization of "Yeast Two-Hybrid" or "Microarray Correlation" have roughly the same impact as did removing the associated data sets in Figure 3.

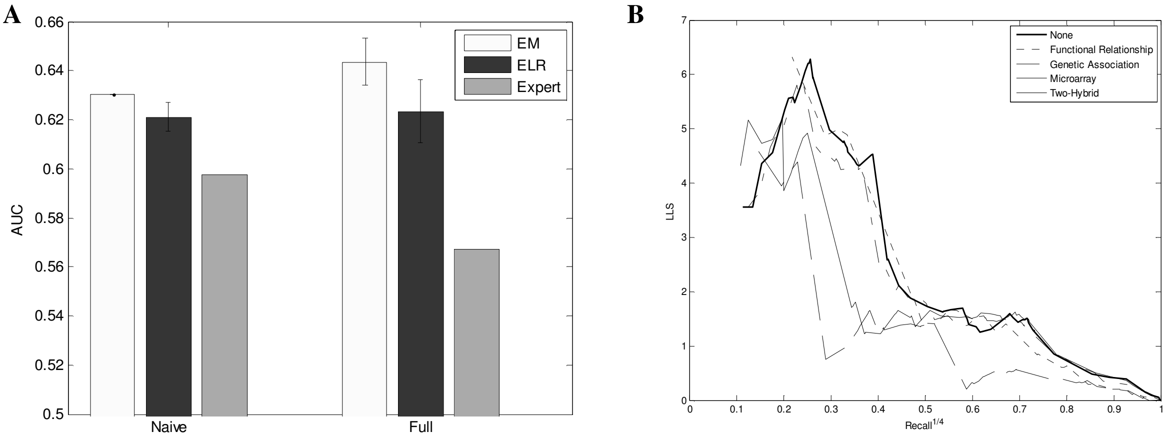

Fig. 4. A) Variation in convergence of network learned from randomly initialized probability tables. Five randomizations were performed to generate the given means and standard deviations; variation is zero for the naive EM network since expectation maximization reduces to deterministic maximum likelihood in this case. Random initialization has little impact on learned network performance. B) A comparison of the full Bayesian network learned using EM to four versions with randomized parameters for the "Functional Relationship", "Microarray Correlation", "Genetic Association", and "Yeast Two-Hybrid" nodes. For each randomized network, one conditional probability table was set to random values after learning and before evaluation. Results using ELR learning are similar (data not shown). Recall has been scaled to emphasize the high precision area of biological interest, and performance is shown using the log-likelihood score $LLS=\log_2(TP\cdot N/FP/P)$ for P total positive pairs, N total negative pairs, and TP and FP the number of true and false positives at a particular sensitivity threshold.

However, the randomization of the hidden "Genetic Association" node had little impact, again indicating the minimal benefit of the full network over a naive Bayesian structure. Thus, the network is fairly resistant to errors in input data or in the corresponding probability tables.

3. DISCUSSION

We investigated the behavior of Bayesian data integration while varying a set of parameters relevant to both the computational and biological aspects of the task. Using two network structures (naive and full), three parameter estimation methods (expert estimation, generative EM learning, and discriminative ELR learning), and several data sets, we demonstrate that learning consistently improves the predictive power of the network. More importantly, decomposing this performance into individual functional categories shows that the improvement afforded by learning varies by function, as does the network's general accuracy.

3.1. Per-function behavior

As mentioned above, it is critical that we consider performance results in the context of individual biological functions. While it is unsurprising that particular data types would have functional biases, the fact that even highly varied and heterogeneous data sources provide little predictive power under some circumstances is rarely taken into account. At least two conclusions can be drawn from the functional analysis in Figure 2. In many cases, underrepresented functional categories are areas in which good high-throughput data is certainly available; one might expect, for example, that yeast two-hybrid experiments would provide information on protein complex assembly[20]. For functional categories such as this, Figure 2 informs us that such signals can be rapidly lost in noise from experimental conditions where otherwise related genes do not function in tandem. Such categories may be better predicted directly by individually trained classifiers such as support vector machines. For other functions, though, data may not be available at all; autophagy might indicate an area in which further laboratory experimentation would substantially improve prediction performance.

There are several of these functional categories for which the best performance remains close to random given any data or network parameters. Most of these are regulatory functions (regulation of biosynthesis, regulation of protein metabolism, regulation of transcription, etc.) with several nucleic acid processing terms interspersed (RNA splicing, mRNA metabolism, DNA packaging, etc.). These terms have no strong size bias, with numbers of annotated genes ranging from tens to several hundred. Aside from issues introduced by data sparsity, it is possible that these larger terms represent more tenuous and less easily detectable functional relationships and are thus more difficult to predict. This bias may also be due to the sparsity of current high-throughput data in certain functional areas; for example, post-transcriptional modification cannot be directly detected by any of the data sets included in this study. However, even the most unreliably predicted functions remain well above random performance, with only two terms having AUCs below 0.6 in the naive EM network.

A functional perspective on performance also allows us to discover terms that are in some sense easier or harder to predict. For example, Figure 2 contains a metabolism cluster (red) that is only predicted well by learned networks; expert networks using either structure produce near or below random performance. Conversely, several groups of functional terms appear to be easy to predict or, equivalently, difficult to improve upon past a certain baseline. A cluster of processes (green) including transcription from RNA polymerase I, translational elongation, rRNA metabolism, and cytoplasm organization all fall into this category. They are predicted with reasonable accuracy by every network configuration, and learning improves their accuracies only minimally. Gene pairs in these terms tend to be supported by multiple data types (many of the most confident pairs are supported by microarray correlation, coimmunoprecipitation, and/or cellular component) and by high microarray correlations. When sufficient data is available, it is unsurprising that results become less dependent upon particular integration techniques.

3.2. Diverse data sources

As shown in Figure 3, we found that several factors influence the relationship between experimental data types and prediction performance. Most data sets (especially small ones containing fewer than 5000 gene pairs) have a negligible impact on the learned networks, and the performance of the naive expert networks remains unchanged even after the removal of some larger data sets. Removing microarrays (by far the largest data source) greatly reduces the accuracy of all four learned networks and the full expert network, but the naive expert network is only minimally affected in this case. This can be interpreted as the full expert network "trusting" the relatively noisy microarray data too much, while the naive network balances it more fairly against other data types; given the large fraction of training data microarrays account for in the learned networks, it is unsurprising that their performance suffers as well.

We also observed that removing moderately sized data sets tends to have an appropriately moderate effect on the overall precision and recall, but more varied results can be seen in a comparison of functional categories (Figure 2). Losses of coimmunoprecipitation, two-hybrid, or synthetic lethality data degrade the naive expert network's performance equally over many functional categories. Removing cellular component data actually improves both the naive and full expert networks, the latter substantially. This improvement is most visible in the group of terms for which learning is particularly beneficial (cell division, DNA recombination, biopolymer biosynthesis, lipid metabolism, and so forth). An inspection of the learned conditional probabilities in any of the network variants reveals that a positive cellular component signal decreases the posterior probability of functional relationship, which is likely indicative of the different focuses of the component and process ontologies within GO.

3.3. Parameter estimation

In general, using expectation maximization or logistic regression to learn conditional probability tables for functional prediction has several clear benefits over expert estimation. In terms of overall prediction accuracy, both precision and recall are significantly enhanced in learned networks, particularly for the full network structure. Predictions are also made much more continuously; expert-populated network predictions tend to cluster in a few tight groups (data not shown). This effectively limits the usefulness of these Bayesian networks as continuous probability estimators and restricts them to a few discrete quanta, a problem not encountered in learned networks.

When comparing ELR to EM learning, Figure 3 indicates that ELR is generally more sensitive to the removal of training data than EM. Particularly in the naive case, when expectation maximization reduces to a simple maximum likelihood estimate, ELR unsurprisingly requires more training data and processing power. The benefits of ELR for our task are seen mainly in its increased consistency, especially in the high precision/low recall area of biological interest.

Figure 4 demonstrates this best, with ELR showing a lower standard deviation over random convergences and producing slightly better results at low recall. This behavior comes at a cost of interpretability, though, since the network parameters learned during ELR are no longer directly interpretable as reliabilities of individual data sets.

It is interesting to note that Bayesian data integration in general is fairly robust to errors in the conditional probability tables. Neither the suboptimal naive expert estimates nor the "worst-case" errors introduced by randomizing the tables (Figure 4B) reduce performance significantly. These randomizations in particular indicate that performance degrades gracefully; in particular, errors in a data set's probability table are generally less harmful than complete removal of the data set (Figure 3). As was already indicated by the similarity between full and naive network performances, modifying hidden nodes has little impact on learned prediction accuracy. However, the full network parameters are clearly more difficult for experts to estimate, a known property of Bayesian probabilities[17].

When examining performance over individual functional categories in Figure 1b, the AUCs for almost all processes are either improved or left unchanged by EM or ELR relative to naive expert probability estimates. In particular, there are specific functional categories for which high-throughput data appears to perform particularly well. These include mainly metabolism terms (amino acid and derivative, alcohol, amine, organic acid, carbohydrate, etc.), many of which are related to a general stress response and are thus represented well in microarray data. A number of other terms are improved less dramatically, consisting mainly of nuclear transport and nucleotide processing categories. From the data set removal results, it appears that this is a general improvement not due to a specific data type. The only term significantly damaged by expectation maximization is RNA modification, which is slightly enriched relative to the prior for related pairs with a shared cellular component.

4. CONCLUSION

The process of collecting, analyzing, and integrating high-throughput biological data has always required a balance between automation and expert knowledge. Bayesian learning provides a natural way to incorporate prior knowledge in the context of formal probabilistic methodology, giving domain experts ample opportunity to intervene with, manipulate, and visualize results, while leaving the work of relationship discovery up to computational methods. Such tools can take advantage of biological accessibility while simultaneously scaling up to larger, heterogeneous data sets and providing fine grained information regarding individual gene pairs and specific functional categories. This applies not only to Bayesian networks, but to any sufficiently sensitive, flexible, and accessible machine learning technique.

Moreover, regardless of the machine learning technique being examined, it is necessary to evaluate the accuracy of functional predictions in the context of individual biological areas. Overall performance improvements may arise from gains in only a few functional categories (such as microarray data's strong ability to predict ribosomal functions). If a computational method is to be used to steer the direction of future laboratory experiments, care must be taken to ensure that it performs adequately in the biological areas relevant to those experiments. Similarly, if predictions are to be made across the entire genome, it is important not to exclude functional terms due to signal loss or lack of data.

In this study, we found that Bayesian learning can be a robust method for prediction of functional relationships from heterogeneous data, but care must be taken in selecting an appropriate training method. While expert estimation provided good overall results, machine learning was able to improve both precision and recall over a wide variety of functional categories, particularly those to which high-throughput data tends to be sensitive. Both the generative expectation maximization and discriminative ELR methods consistently surpassed the expert models' predictions, particularly for the full network structure modeling hidden relationships between experimental data types. As the field expands, it is vital to adapt learning methods such as these to new high-throughput data sources, and it is equally vital to produce high-throughput data with sufficient coverage and functional diversity to realize the potential of computational methods.

It is clearly necessary to examine performance within specific functional categories to reveal many of these differences, and such evaluation is an important aspect of any functional predictor. Individual data sources come with functional biases, and integrating

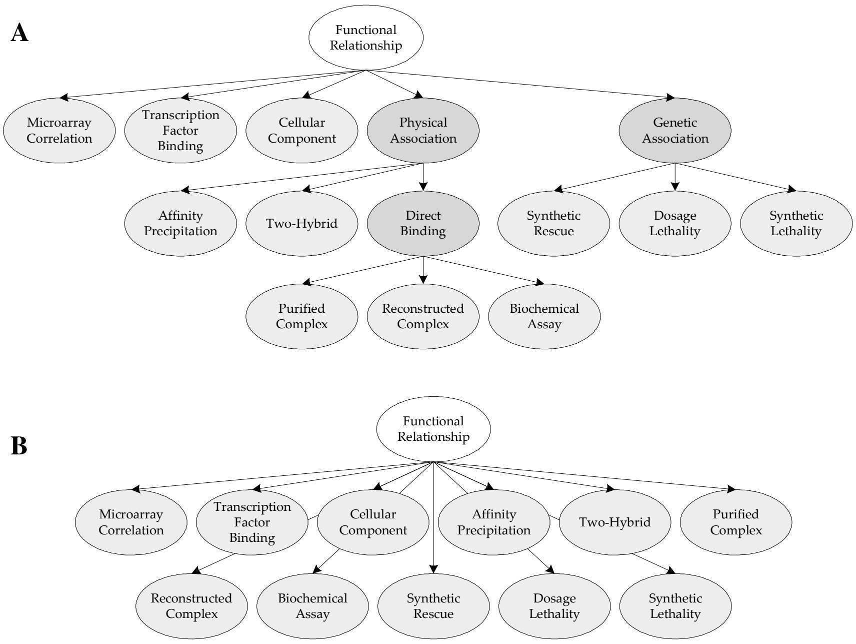

Fig. 5. A) Full network structure. Hidden nodes are shown in dark gray, data set nodes in light gray, and the output node in white. B) Naive network structure; shading is as in part A.

them without proper care can exacerbate these deficiencies; adding more data will almost always improve overall genomic performance, but this may come at the cost of drowning out specific functional categories. At the most difficult end of the scale, there still exist biological areas insufficiently covered by all high-throughput data, even in an organism as well-studied as *S. cerevisiae*. It is in areas such as these that a carefully managed integration of computational and biological knowledge can yield the most substantial returns.

5. METHODS

5.1. Bayesian network algorithms

Bayesian network inference was performed using the Lauritzen algorithm[21] for inference and expectation maximization[17] or extended logistic regression[18] for parameter learning. Expectation maximization was, in all cases, allowed to converge for five iterations; ELR ran for 1000 iterations. Some additional experiments with continuous Bayesian networks were run using the junction tree inference algorithm[22]. Performance was found to be generally below that of continuous networks (data not shown), which is perhaps unsurprising given the unsuitability of the linear Gaussian assumption. The University of Pittsburgh Decision System Laboratory's SMILE library and GENIE modeling environment[23] were used for manipulation of discrete networks, and the Intel PNL library[24] was used with continuous networks.

5.2. Bayesian network implementation

The full and naive network structures are shown in Figure 5. The former was constructed by simplifying the Troyanskaya et al[15] network's complex microarray inputs to a single correlation node and removing the trivially small "Unlinked Noncomplementation" node. Preliminary experiments showed that this made a negligible difference in performance relative to the original expert network (data not shown). The latter was constructed by removing each hidden node (those representing neither inputs to nor outputs from the predictor) while maintaining the expert estimated conditional probability tables of the remaining nodes.

Of the heterogeneous data sources, most represent positive binary genetic interactions in which a "true" result indicates that two genes interact and a "false" result indicates that they do not interact or that no data is present. Biochemical assays, coimmunoprecipitation, synthetic and dosage interactions, protein complexes (all drawn from the GRID[25] and BIND[26] databases), and transcription factor modules[27] all fall into this category, as the cellular component data (from the Gene Ontology). This allowed each of these data types to be presented to the Bayesian network as a single boolean variable per gene pair. Microarray coexpression data were collected from a variety of sources[28-38].

5.3. Data preparation

Each of the data sources was represented as a binary input with a true value indicating cooccurrence of a gene pair in the data set. For each data set, missing gene pairs were represented by false values. The microarray data described above was preprocessed by concatenating the approximately 350 conditions into a single expression vector for each gene. Genes with more than 70% missing data were removed, after which any remaining missing values were imputed using KNNImpute[39] with k=10. Pairwise relationships were calculated by computing the centered Pearson correlations for all gene pairs within individual data sets and normalizing these to the range [0, 1]. The overall correlation was then taken to be the average of these values and subsequently quantized into five bins representing values less than 0.5, 0.5-0.75, 0.75-0.8, 0.8-0.9, and greater than 0.9 for input into the Bayesian networks.

For robustness testing, network terms were chosen for randomization in such a way as to cover a variety of data set types and sizes. Randomizing the "Functional Relationship" node demonstrates the evaluation's dependence only on gene pair rank (and not on exactly probability estimation), and "Genetic Association" shows the relatively small benefit provided by the full network's hidden nodes. "Microarray Correlation" provided a large, continuous data set, and "Two-Hybrid" represented one that was smaller and discretized.

5.4. Gold standard generation

Gene Ontology terms representing positive functional relationships were selected at a 5% gene count cutoff, corresponding to GO biological process terms to which at most 321 of the 6438 *S. cerevisiae* genes were annotated[40]. Any two genes coannotated to such a term or its descendants were considered to be functionally related. Similarly, terms to which at least 15% of the genome (965 genes) was annotated represented a negative threshold; any genes coannotated to such a term and not to any more specific term were considered to be functionally unrelated. Gene pairs coannotated to intermediate terms were excluded from the gold standard and thus from the evaluation. This process resulted in a set of 720458 related gene pairs and 10566822 unrelated pairs. A 5% positive and 10% negative term cutoff tested with the MIPS hierarchy performed similarly (data not shown).

5.5. Testing and cross validation

For all networks, evaluation was performed as an average of 5-fold cross validation (approximately 953 genes per fold). Additional cross validation was performed by varying random seeds as shown in Figure 4B; each training and evaluation cycle, regardless of conditional probability table seeding, utilized five different gene sets.

Overall LLS/recall curves were generated from probabilities drawn from the topmost "Functional Relationship" network node after Bayesian inference, again averaged over 5-fold cross validation. To calculate per-functional category performance, gene pairs were considered relevant to a category if they were both annotated to the category (and thus related) or if they were unrelated and one gene was annotated to the category. Negative pairs in which neither gene was annotated to a functional term below the 5% cutoff were evaluated with every functional category. All AUCs were calculated analytically using the Wilcoxon Rank Sum formula[41]. The resulting data were converted into heat maps using the TIGR MeV[42] software.

Acknowledgments

We thank everyone in the Troyanskaya lab for their help and insightful discussions. This research was partially supported by NIH grant R01 GM071966 and NSF grant IIS-0513552 to Olga G. Troyanskaya. Dr. Troyanskaya is an Alfred P. Sloan Research Fellow.

References

1. Detours, V., et al. Integration and cross-validation of high-throughput gene expression data: comparing heterogeneous data sets. *FEBS Letters* 2003. **546**(1): 98-102.
2. Joyce, A.R. and B.O. Palsson. The model organism as a system: integrating 'omics' data sets. *Nature Reviews: Molecular Cell Biology* 2006. **7**(3): 198-210.

3. Yu, J. and F. Fotouhi. Computational approaches for predicting protein-protein interactions: a survey. *J. Medical Systems* 2006. **30**(1): 39-44.
4. Chapman, A., C. Yu, and H.V. Jagadish. Effective integration of protein data through better data modeling. *Omics* 2003. **7**(1): 101-2.
5. Lacroix, Z. Biological data integration: wrapping data and tools. *IEEE Transactions on Information Technology in Biomedicine* 2002. **6**(2): 123-8.
6. Venkatesh, T.V. and H.B. Harlow. Integromics: challenges in data integration. *Genome Biology* 2002. **3**(8): REPORTS4027.
7. Clare, A. and R.D. King. Predicting gene function in Saccharomyces cerevisiae. *Bioinformatics* 2003. **19 Suppl 2**: II42-II49.
8. Zhang, L.V., et al. Predicting co-complexed protein pairs using genomic and proteomic data integration. *BMC Bioinformatics* 2004. **5**: 38.
9. Lanckriet, G.R., et al. Kernel-based data fusion and its application to protein function prediction in yeast. *Pacific Symposium on Biocomputing* 2004: 300-11.
10. Hwang, D., et al. A data integration methodology for systems biology. *PNAS* 2005. **102**(48): 17296-301.
11. Karaoz, U., et al. Whole-genome annotation by using evidence integration in functional-linkage networks. *PNAS* 2004. **101**(9): 2888-93.
12. Lee, I., et al. A probabilistic functional network of yeast genes. *Science* 2004. **306**(5701): 1555-8.
13. Nabieva, E., et al. Whole-proteome prediction of protein function via graph-theoretic analysis of interaction maps. *Bioinformatics* 2005. **21 Suppl 1**: i302-i310.
14. Lu, L.J., et al. Assessing the limits of genomic data integration for predicting protein networks. *Genome Research* 2005. **15**(7): 945-53.
15. Troyanskaya, O.G., et al. A Bayesian framework for combining heterogeneous data sources for gene function prediction (in Saccharomyces cerevisiae). *PNAS* 2003. **100**(14): 8348-53.
16. Ashburner, M., et al. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nature Genetics* 2000. **25**(1): 25-9.
17. Neapolitan, R. *Learning Bayesian Networks*. 2004, Chicago, Illinois: Prentice Hall.
18. Greiner, R., et al. Structural Extension to Logistic Regression: Discriminative Parameter Learning of Belief Net Classifiers. *Machine Learning Journal* 2005. **59**(3): 297-322.
19. Ruepp, A., et al. The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. *Nucleic Acids Research* 2004. **32**(18): 5539-45.
20. Ito, T., et al. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *PNAS* 2001. **98**(8): 4569-74.
21. Lauritzen, S. and D. Spiegelhalter. Local Computations with Probabilities on Graphical Structures and their Application to Expert Systems. *J. Royal Statistical Society* 1988. **50**(2).
22. Jensen, F. *An Introduction to Bayesian Networks*. 1996: Springer.
23. Druzdzel, M. SMILE: Structural Modeling, Inference, and Learning Engine and GeNIe: A development environment for graphical decision-theoretic models. *Proceedings of the Sixteenth National Conference on Artificial Intelligence* 1999: 902-903.
24. Eruhimov, V., K. Murphy, and G. Bradski, *Intel's open-source probabilistic networks library*. 2003: http://www.intel.com/technology/computing/pnl/index.htm.
25. Breitkreutz, B.J., C. Stark, and M. Tyers. The GRID: the General Repository for Interaction Datasets. *Genome Biology* 2003. **4**(3): R23.
26. Bader, G.D., et al. BIND--The Biomolecular Interaction Network Database. *Nucleic Acids Research* 2001. **29**(1): 242-5.
27. Fujibuchi, W., J.S. Anderson, and D. Landsman. PROSPECT improves cis-acting regulatory element prediction by integrating expression profile data with consensus pattern searches. *Nucleic Acids Research* 2001. **29**(19): 3988-96.
28. Chu, S., et al. The transcriptional program of sporulation in budding yeast. *Science* 1998. **282**(5389): 699-705.
29. DeRisi, J.L., V.R. Iyer, and P.O. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science* 1997. **278**(5338): 680-6.
30. Gasch, A.P., et al. Genomic expression responses to DNA-damaging agents and the regulatory role of the yeast ATR homolog Mec1p. *Molecular Biology of the Cell* 2001. **12**(10): 2987-3003.
31. Gasch, A.P., et al. Genomic expression programs in the response of yeast cells to environmental changes. *Molecular Biology of the Cell* 2000. **11**(12): 4241-57.
32. Hughes, T.R., et al. Widespread aneuploidy revealed by DNA microarray expression profiling. *Nature Genetics* 2000. **25**(3): 333-7.
33. Ogawa, N., J. DeRisi, and P.O. Brown. New components of a system for phosphate accumulation

and polyphosphate metabolism in Saccharomyces cerevisiae revealed by genomic expression analysis. *Molecular Biology of the Cell* 2000. **11**(12): 4309-21.

34. Shakoury-Elizeh, M., et al. Transcriptional remodeling in response to iron deprivation in Saccharomyces cerevisiae. *Molecular Biology of the Cell* 2004. **15**(3): 1233-43.
35. Spellman, P.T., et al. Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. *Molecular Biology of the Cell* 1998. **9**(12): 3273-97.
36. Sudarsanam, P., et al. Whole-genome expression analysis of snf/swi mutants of Saccharomyces cerevisiae. *PNAS* 2000. **97**(7): 3364-9.
37. Yoshimoto, H., et al. Genome-wide analysis of gene expression regulated by the calcineurin/Crz1p signaling pathway in Saccharomyces cerevisiae. *J. Biological Chemistry* 2002. **277**(34): 31079-88.
38. Zhu, G., et al. Two yeast forkhead genes regulate the cell cycle and pseudohyphal growth. *Nature* 2000. **406**(6791): 90-4.
39. Troyanskaya, O., et al. Missing value estimation methods for DNA microarrays. *Bioinformatics* 2001. **17**(6): 520-5.
40. Hong, E., et al., *Saccharomyces Genome Database*. 2005, http://www.yeastgenome.org/.
41. Lehmann, E. *Nonparametrics: Statistical Methods Based on Ranks*. 1975: McGraw-Hill.
42. Saeed, A.I., et al. TM4: a free, open-source system for microarray data management and analysis. *Biotechniques* 2003. **34**(2): 374-8.