RANKING GENES BY RELEVANCE TO A DISEASE

Shivani Agarwal [1*] and Shiladitya Sengupta [2,3]

[1] *Department of Electrical Engineering and Computer Science,*
[2] *Harvard-MIT Division of Health Sciences and Technology,*
Massachusetts Institute of Technology,
Cambridge, MA 02139, USA
[*] *Email: shivani@mit.edu*

[3] *Department of Medicine,*
Brigham and Women's Hospital, Harvard Medical School,
Boston, MA 02115, USA
Email: shiladit@mit.edu

The problem of identifying key genes that are involved in a particular disease is of fundamental importance in biology and medicine. Given the increasing availability of a variety of gene-related biological data sources, ranging from microarray expression data to protein-protein interaction data, a promising approach is to use bioinformatics methods that can analyze this data and rank genes based on potential relevance to a disease; such methods can be invaluable in helping to prioritize genes for further biological study. Recently, the problem of ranking objects has gained considerable attention in machine learning and data mining; ranking problems arise in a variety of domains ranging from document retrieval to collaborative filtering, and a variety of new learning methods have been developed that directly optimize ranking performance. Here we propose for the first time the use of such ranking methods for the problem of ranking genes. We illustrate this approach on microarray data for leukemia and colon cancer; in both cases, the ranking methods we use identify several genes that were not identified by previous methods, including some novel genes that could potentially evolve as targets for drug development. Our study suggests that ranking methods in machine learning could emerge as a powerful tool for mining gene-related data sources for the identification of genes relevant to a particular disease.

1. INTRODUCTION

One of the greatest challenges in post-genome medical research is to identify genes that are involved in a particular disease[27]. Identification of such genes not only provides a better understanding of the disease, but is also often the first step in developing treatments for it.

With the rapid growth in biological data sources containing gene-related information, including for example sequence information, microarray expression data, functional annotation data, protein-protein interaction data, and the biological and medical literature, there has been much interest in recent years in developing bioinformatics approaches that can analyze this data and help in the identification of important genes. In particular, a common goal is to rank or prioritize genes such that those relevant to the disease under study are likely to appear at the top of the ranking; the proteins corresponding to the top few genes can then be subjected to biological tests to elucidate their structural and functional properties, with a good chance that many of those tested will emerge as targets for the development of new drugs or find use as disease markers that can be helpful in diagnosis.

Over the last few years, several methods have been proposed for ranking or prioritizing genes by relevance to a disease. These methods fall into two broad classes. The first class of methods uses mostly microarray expression data; these methods focus on identifying genes that are differentially expressed in a disease, and use simple statistical measures such as the t-statistic[17, 29, 31] or related classification methods in machine learning[18, 21] to rank genes based on this property. Related to these are methods that aim to improve an initial ranking obtained from expression data by augmenting it with a network structure derived from other data sources[26, 25]. The second class of methods is more general, often making use

*Corresponding author.

of a variety of data sources; these methods start with some existing knowledge of 'training' genes already known to be related to the disease under study, and directly or indirectly rank the remaining genes based on their similarity to these training genes[14, 2, 10, 9].

Here we take a different approach, inspired by recent developments on ranking methods in machine learning. In particular, the problem of ranking objects has recently gained much attention in machine learning, data mining, and statistics, both due to its widespread applications in information retrieval and related areas, and due to the fact that it is a mathematically distinct problem from the classical learning problems of classification and regression[12, 19, 20, 15, 13, 4, 28, 11, 5]. We show here that the gene ranking problem is naturally formulated as a particular form of ranking problem termed the bipartite ranking problem[15, 4]. This allows us to exploit existing knowledge of both 'positive' training genes that are known to be related to the disease under study, and 'negative' training genes that are known to be unrelated; learning methods that directly optimize ranking performance are then used to rank the remaining genes such that genes relevant to the disease are likely to be ranked higher than those that are not relevant.

The overall approach of using ranking methods in machine learning is highly flexible and can be used in conjunction with multiple data sources. We illustrate the approach on microarray expression data for leukemia and colon cancer. Even using only microarray data, in both cases, the ranking returned by our approach identifies some novel genes that were not identified by previous methods, and that can potentially evolve as targets for drug development.

The rest of the paper is organized as follows: Section 2 describes in greater detail previous work on the gene ranking problem; in Section 3, we describe our formulation of the problem as a bipartite ranking problem in machine learning. This is followed by our experimental results in Section 4. We conclude with a discussion in Section 5.

2. PREVIOUS WORK

As discussed briefly above, several methods have been proposed for ranking or prioritizing genes based on relevance to a disease. These methods fall into two broad classes: methods in the first class use mostly microarray expression data, and rank genes based on the extent to which they are differentially expressed in the disease; methods in the second class often use a variety of data sources, and rank genes based on their similarity to a set of 'training' genes already known to be related to the disease. Here we discuss these methods in greater detail.

Consider first the problem of ranking genes using microarray expression data. Such data can be represented as an expression matrix $\mathbf{X} = [x_{ik}] \in \mathbb{R}^{N \times d}$, where N is the number of genes whose expression levels are measured, d is the number of biological samples (*e.g.*, tissue samples from different patients, or different experimental conditions), and x_{ik} is the expression level of gene i in sample k. We shall use $\mathbf{x}_i \in \mathbb{R}^d$ to denote the expression vector of gene i across the d samples, and $\widetilde{\mathbf{x}}_k \in \mathbb{R}^N$ to denote the expression vector of sample k across the N genes. The problem of finding a ranking of genes can then be viewed as a problem of finding a ranking function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ that assigns a real-valued score $f(\mathbf{x}_i)$ to gene i; gene i is ranked higher than gene j if $f(\mathbf{x}_i) > f(\mathbf{x}_j)$.

The first class of methods is designed for the above problem and assumes that the biological samples are of two different types, so that with each sample k, there is an associated label $y_k \in \{-1, 1\}$ denoting its type (*e.g.*, disease or normal, or one of two different forms of a disease). The function f is then chosen to rank genes based on their ability to distinguish between the two types of samples. For example, in several studies[17, 29, 31], the score $f(\mathbf{x}_i)$ assigned to gene i is taken to be some measure of correlation between the gene's expression vector $\mathbf{x}_i$ and the label vector $\mathbf{y}$, such as the t-statistic; high scores then correspond to genes that have markedly different expression levels across the two types of samples. In some other studies[18, 21], classification methods in machine learning are used in the following indirect manner: the sample vectors $\widetilde{\mathbf{x}}_k$ together with their binary labels y_k are provided as training examples to a classification algorithm that learns a linear classifier $h : \mathbb{R}^N \rightarrow \{-1, 1\}$ of the form $h(\widetilde{\mathbf{x}}) = \text{sign}(\widetilde{\mathbf{w}} \cdot \widetilde{\mathbf{x}} + \theta)$, where $\widetilde{\mathbf{w}} \in \mathbb{R}^N, \theta \in \mathbb{R}$; the goal in learning the classifier is actually to classify accurately new biological samples, but an implicit

ranking over the genes is obtained by viewing each weight $\widetilde{w}_i$ in the classifier as a measure of the contribution of gene i, and taking $f(\mathbf{x}_i) = |\widetilde{w}_i|$.

Related to the above methods are methods that aim to improve an initial ranking obtained from expression data by augmenting it with a network structure derived from other data sources. For example, the GeneRank algorithm of Morrison et al.[26], which is based on the PageRank algorithm used by the Google search engine, starts with an initial ranking of genes based on differential expression scores, which could be derived using any of the above methods, and then improves this ranking by propagating scores across genes in the network; in this case, the network was derived from Gene Ontology (GO) annotation data[30]. Similarly, the algorithm of Ma et al.[25] improves an initial ranking of genes obtained from expression data using a Markov random field approach that effectively constrains the ranking to assign similar scores to genes that are closely connected in the network; in this case, the network was derived from protein-protein interaction data.

The second class of methods focuses on ranking genes by their similarity to a set of 'training' genes already known to be related to the disease, and can often use a variety of data sources. For example, in a study by Freudenberg and Propping[14], diseases known to be caused by certain genes are first grouped into clusters according to their phenotypic similarity. Given a target disease, all disease clusters similar to the target disease are identified, again based on phenotypic similarity; the genes known to cause the diseases in these clusters are then viewed as 'training' examples for the target disease. The remaining genes are then ranked based on their similarity to these genes; in this case, similarity between genes was measured using GO annotations. In a more recent study by Aerts et al.[2], multiple data sources are used, including microarray expression data, protein-protein interaction data, and the biomedical literature. For each data source, the similarity of each gene to a set of 'training' genes known to be related to the target disease is computed using a data-specific similarity measure; this results in a distinct ranking for each data source. These rankings are then combined into a single overall ranking using order statistics. A number of other recent studies[10, 9] have used protein-protein interaction data together with network-based methods in order to rank genes based on an initial set of 'training' genes related to the disease of interest.

There are also some methods that rank or prioritize genes based on their overall likelihood of being involved in some disease in general (without reference to a specific disease)[23, 1]; we do not discuss these methods here since our goal is to identify genes involved in a particular disease.

Below we present an alternative approach to ranking genes that is based on recent developments on ranking methods in machine learning. In particular, we show that the gene ranking problem can be formulated naturally as a particular form of ranking problem known as the bipartite ranking problem[15, 4]; this allows us to exploit existing knowledge of both 'positive' training genes that are known to be related to the disease under study, and 'negative' training genes that are known to be unrelated, and to automatically learn from these training examples a ranking over the remaining genes that tends to rank relevant genes higher than irrelevant ones.

3. FORMULATION AS A BIPARTITE RANKING PROBLEM

As discussed above, the problem of ranking objects has recently gained considerable attention in machine learning, data mining, and statistics, both due to its widespread applications in information retrieval and related areas, and due to the fact that ranking is a mathematically distinct problem from the classical learning problems of classification and regression[12, 19, 20, 15, 13, 4, 28, 11, 5].

In the general ranking problem in machine learning, one is given examples of order relationships among instances in some instance space $\mathcal{X}$, and the goal is to learn from these examples a ranking or ordering over $\mathcal{X}$ that ranks accurately future instances. In the most general setting of the problem, the learner is given training examples in the form of ordered pairs of instances $(x, x') \in \mathcal{X} \times \mathcal{X}$, each labeled with a ranking preference $r \in \mathbb{R}$, with the interpretation that x is to be ranked higher than x' if $r > 0$, and lower than x' if $r < 0$ ($r = 0$ indicates no ranking preference between the two instances); the penalty for mis-ordering such a pair is

proportional to $|r|$. Given a finite number of such examples $S = ((x_1, x'_1, r_1), \ldots, (x_m, x'_m, r_m))$, the goal is to learn a real-valued ranking function $f : \mathcal{X} \to \mathbb{R}$ that ranks accurately future instances; f is considered to rank an instance $x \in \mathcal{X}$ higher than an instance $x' \in \mathcal{X}$ if $f(x) > f(x')$, and lower than x' if $f(x) < f(x')$.

A particular setting of the ranking problem that has been investigated in some detail in recent years, and that will be useful for our purposes, is the *bipartite* setting[15, 4]. In the bipartite ranking problem, instances come from two categories, positive and negative; the learner is given examples of instances labeled as positive or negative, and the goal is to learn a ranking in which positive instances are ranked higher than negative ones. Such ranking problems arise, for example, in information retrieval, where one is interested in retrieving documents from some database that are relevant to a given topic; in this case, the training examples given to the learner consist of documents labeled as relevant (positive) or irrelevant (negative), and the goal is to rank the remaining documents such that relevant documents are ranked higher than irrelevant ones.

More formally, in the bipartite ranking problem, the learner is given a training sample (S_+, S_-) consisting of a sequence of 'positive' examples $S_+ = (x_1^+, \ldots, x_m^+)$ and a sequence of 'negative' examples $S_- = (x_1^-, \ldots, x_n^-)$, the x_i^+ and x_j^- being instances in some instance space $\mathcal{X}$, and the goal is to learn a real-valued ranking function $f : \mathcal{X} \to \mathbb{R}$ that ranks future positive instances higher than negative ones. The bipartite ranking problem is easily seen to be a special case of the general ranking problem described above, since a training sample $(S_+, S_-) \in \mathcal{X}^m \times \mathcal{X}^n$ in the bipartite setting can be viewed as consisting of mn examples of the form $(x_i^+, x_j^-, 1)$ for $i \in \{1, \ldots, m\}$, $j \in \{1 \ldots, n\}$, with a constant penalty for misranking any positive-negative pair of instances.

Consider now the problem of ranking genes based on relevance to a particular disease. For simplicity in this paper we focus on the problem of ranking genes using microarray expression data, although our methods can be applied using other data sources as well; indeed, as we discuss in Section 5, our methods can also be applied in conjunction with multiple data sources. Recall that we represent a microarray data set as an expression matrix $\mathbf{X} = [x_{ik}] \in \mathbb{R}^{N \times d}$, where N denotes the number of genes, d the number of biological samples, and x_{ik} the expression level of gene i in sample k. It turns out that the problem of ranking genes in this setting can be formulated naturally as a bipartite ranking problem. In this formulation, the instances to be ranked are genes, each represented by a d-dimensional expression vector; thus the instance space $\mathcal{X}$ is simply $\mathbb{R}^d$. The biologist provides a few examples $S_+ = (\mathbf{x}_1^+, \ldots, \mathbf{x}_m^+) \in (\mathbb{R}^d)^m$ of (expression vectors corresponding to) genes that are known to be relevant to the disease, and a few examples $S_- = (\mathbf{x}_1^-, \ldots, \mathbf{x}_n^-) \in (\mathbb{R}^d)^n$ of (expression vectors corresponding to) genes known to be irrelevant. Any learning algorithm for the bipartite ranking problem can then be used to automatically learn from these examples a ranking function $f : \mathbb{R}^d \to \mathbb{R}$ that tends to rank relevant genes higher than irrelevant ones.

In our experiments, we used the bipartite RankBoost algorithm of Freund et al.[15], which is based on the principles of boosting[16]. An outline of the algorithm is given in Figure 1. The algorithm takes as input a training sample of the form $(S_+, S_-) \in \mathcal{X}^m \times \mathcal{X}^n$, where $\mathcal{X}$ is the instance space and $S_+ = (x_1^+, \ldots, x_m^+)$, $S_- = (x_1^-, \ldots, x_n^-)$, and produces as output a ranking function $f : \mathcal{X} \to \mathbb{R}$ that is a linear combination of some 'weak' ranking functions chosen from some base class $\mathcal{F}_{\text{base}}$. The algorithm works in rounds and maintains a distribution D_t over the set of positive-negative pairs $\left\{(x_i^+, x_j^-) : i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}\right\}$. On each round t, it chooses a weak ranking $f_t \in \mathcal{F}_{\text{base}}$ and a real number $\alpha_t \in \mathbb{R}$, and updates the distribution D_t such that instance pairs (x_i^+, x_j^-) that are mis-ranked by f_t are weighted more heavily by D_{t+1}; the extent of the update is determined by α_t (typically, $\alpha_t > 0$). The final ranking is given by a weighted combination of the weak rankings chosen in different rounds.

As discussed above, the instance space $\mathcal{X}$ in our gene ranking problem is $\mathbb{R}^d$. The base function class $\mathcal{F}_{\text{base}}$ we use contains the d coordinate projection functions $f^{(k)} : \mathbb{R}^d \to \mathbb{R}$, given by $f^{(k)}(\mathbf{x}) = x_k$ for each $k \in \{1, \ldots, d\}$. Thus on each round t, our weak learner chooses $f_t : \mathbb{R}^d \to \mathbb{R}$ to be $f_t(\mathbf{x}) = x_{k_t}$ for some $k_t \in \{1, \ldots, d\}$. In accordance with the theory

Algorithm RankBoost (Bipartite)

Input: $(S_+, S_-) \in \mathcal{X}^m \times \mathcal{X}^n$.

Initialize: $D_1(x_i^+, x_j^-) = \dfrac{1}{mn}$ for all $i \in \{1, \ldots, m\},\ j \in \{1, \ldots, n\}$.

For $t = 1, \ldots, T$:

- Train weak learner using distribution D_t; get weak ranking $f_t : \mathcal{X} \to \mathbb{R}$.
- Choose $\alpha_t \in \mathbb{R}$.
- Update: $\mathrm{D}_{t+1}(x_i^+, x_j^-) = \dfrac{1}{Z_t} D_t(x_i^+, x_j^-) \exp\left\{-\alpha_t\big(f_t(x_i^+) - f_t(x_j^-)\big)\right\}$,

  where $\mathrm{Z}_t = \displaystyle\sum_{i=1}^{m}\sum_{j=1}^{n} D_t(x_i^+, x_j^-) \exp\left\{-\alpha_t\big(f_t(x_i^+) - f_t(x_j^-)\big)\right\}$.

Output the final ranking: $f(x) = \displaystyle\sum_{t=1}^{T} \alpha_t f_t(x)$.

Fig. 1. The bipartite RankBoost algorithm of Freund et al.[15]

behind RankBoost[15], k_t is chosen as

$$k_t = \arg\min_{k \in \{1,\ldots,d\}} \left\{ \min_{\alpha \in \mathbb{R}} \sum_{i=1}^{m}\sum_{j=1}^{n} D_t(\mathbf{x}_i^+, \mathbf{x}_j^-) e^{-\alpha(x_{ik}^+ - x_{jk}^-)} \right\},$$

and α_t is then chosen as

$$\alpha_t = \arg\min_{\alpha \in \mathbb{R}} \sum_{i=1}^{m}\sum_{j=1}^{n} D_t(\mathbf{x}_i^+, \mathbf{x}_j^-) e^{-\alpha(x_{ik_t}^+ - x_{jk_t}^-)}.$$

As in the case of the AdaBoost algorithm for classification[16], with the above choice of k_t and α_t, the bipartite RankBoost algorithm can be viewed as performing coordinate descent to minimize an objective function that forms a convex upper bound on the ranking error with respect to the training sample[28]. Our final ranking function $f : \mathbb{R}^d \to \mathbb{R}$ is a linear function given by $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x}$, where $w_k = \sum_{\{t : k_t = k\}} \alpha_t$ for each $k \in \{1, \ldots, d\}$.

4. EXPERIMENTS

We evaluated our gene ranking approach on two publicly available microarray data sets: a leukemia data set[17] and a colon cancer data set[7]. Below we describe these data sets (Section 4.1), the methodology we used for selecting positive and negative training genes (Section 4.2), and our results (Section 4.3).

4.1. Data Sets

We conducted experiments on two publicly available microarray data sets. The first of these is a leukemia data set that was first used in a study by Golub et al.[17] and was subsequently made available by the authors of that study.[a] The data set contains expression levels of 7129 genes across 72 samples. The samples in this data set correspond to tissue samples obtained from different leukemia patients; of the 72 samples, 25 are from acute myeloid leukemia (AML) and 47 from acute lymphoblastic leukemia (ALL). In many studies involving this data set, the goal has been to classify samples as belonging to AML or ALL. In our case, the goal was to rank genes by relevance to leukemia.

The second data set is a colon cancer data set that was first used by Alon et al.[7] and was subsequently made available.[b] The data set contains expression levels of 2000 genes across 62 samples. The samples in this data set correspond to tissue samples obtained from patients with and without colon cancer; of the 62 samples, 40 are from tumor tissue and 22 from normal tissue. In many studies involving this data set, the goal has been to classify samples

[a] This data set is available from `http://www.broad.mit.edu/cgi-bin/cancer/datasets.cgi`.

[b] This data set is available from `http://microarray.princeton.edu/oncology/affydata/index.html`.

Table 1. Markers for AML and ALL were selected as positive training genes for the leukemia data set; markers for colon cancer were selected as positive training genes for the colon cancer data set.

Markers for AML	Myeloperoxidase
	CD13
	CD33
	HOXA9 Homeo box A9
	MYBL2
Markers for ALL	CD19
	CD10 (CALLA)
	TCL1 (T cell leukemia)
	C-myb
	Deoxyhypusine synthase
Markers for colon cancer	Phospholipase A2
	Keratin 6 isoform
	Protein-tyrosine phosphatase PTP-H1
	Transcription factor IIIA
	Viral (v-raf) oncogene homolog 1
	Dual specificity mitogen-activated protein kinase kinase 1
	Transmembrane carcinoembryonic antigen
	Oncoprotein 18
	Phosphoenolpyruvate carboxykinase
	Extracellular signal-regulated kinase 1

as tumor or normal. Again, in our case, the goal was to rank genes by relevance to colon cancer.

4.2. Selection of Training Genes

In order to rank genes using our bipartite ranking framework, we selected for each data set a small number of training genes based on existing biological knowledge.

Of the 7129 genes in the leukemia data set, we selected 10 genes as positive training examples S_+ and 157 genes as negative training examples S_-. The 10 genes selected as positive examples are all known classical markers for either AML or ALL; these are shown in Table 1.[c] Of the 157 genes selected as negative examples, 59 are internal controls available on the Affymetrix chip (indicated in the data set); the rest are genes that are involved in a variety of physiological cellular functions, including for example house-keeping genes, genes coding for ion channels and essential enzymes, hormone-associated genes, and genes involved in cellular transport and a variety of other focal, cell-specific functions.

Similarly, of the 2000 genes in the colon cancer data set, we selected 10 genes as positive training examples S_+ and 56 genes as negative training examples S_-. The 10 genes selected as positive examples are all known markers for colon cancer; these are also shown in Table 1. Of the 56 genes selected as negative examples, 8 are internal controls; the rest are again genes that are involved in a variety of physiological cellular functions as above.

4.3. Results

Using the training genes described above, a ranking over the remaining genes in each data set was learned using the bipartite RankBoost algorithm. In each case, in order to assess the quality of the ranking produced by our method, we performed an extensive validation with the biomedical literature to determine the biological relevance of the 25 top-ranked genes. Our main resource for this literature search was PubMed[d], an online indexing service for

[c] A marker for a disease is a gene whose expression levels are distinctly altered in the disease. Markers are useful in diagnosis of diseases and in monitoring of therapeutic outcomes.
[d] PubMed website: `http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed`.
[e] Entrez Gene website: `http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene`.

Table 2. Leukemia results: 25 top-ranked genes. Relevance summary: ♦ Known marker; ◊ Potential marker; ■ Known therapeutic target; □ Potential therapeutic target; × No link found. See Supplementary Material[6] for detailed analyses of the relevance of these genes. Ranks of the genes in rankings based on the t-statistic[29] and Pearson correlation[2] are shown for comparison; see text for details.

	Gene	Relevance Summary	t-Statistic Rank	Pearson Rank
1.	KIAA0220	□	6628	2461
2.	G-gamma globin	♦	3578	3567
3.	Delta-globin	♦	3663	3532
4.	Brain-expressed HHCPA78 homolog	□	6734	2390
5.	Myeloperoxidase	♦	139	6573
6.	Probable protein disulfide isomerase ER-60 precursor	□	6650	575
7.	NPM1 Nucleophosmin	♦	405	1115
8.	CD34	♦	6732	643
9.	Elongation factor-1-beta	×	4460	3413
10.	CD24	♦	81	1
11.	60S ribosomal protein L23	□	1950	73
12.	5-aminolevulinic acid synthase	□	4750	3351
13.	HLA class II histocompatibility antigen	♦	5114	298
14.	Epstein-Barr virus small RNA-associated protein	□	6388	1650
15.	HNRPA1 Heterogeneous nuclear ribonucleoprotein A1	□	4188	1791
16.	Azurocidin	♦	162	6789
17.	Red cell anion exchanger (EPB3, AE1, Band 3)	□	3853	4926
18.	Topoisomerase II beta	■	17	3
19.	HLA class I histocompatibility antigen	×	265	34
20.	Probable G protein-coupled receptor LCR1 homolog	□	30	62
21.	HLA-SB alpha gene (class II antigen)	×	6374	317
22.	Int-6	◊	3878	912
23.	Alpha-tubulin	□	5506	1367
24.	Terminal transferase	♦	6	9
25.	Glycophorin B precursor	◊	3045	5668

life sciences literature. We also performed database searches using Entrez Gene[e], an online tool that allows one to search specifically for information related to a given gene, and BLAST[f], a sequence alignment tool that allows one to search for DNA sequences similar to a given sequence (and to thus find genes that are homologous to a given gene)[8].

Table 2 lists the top 25 genes in the ranking learned for leukemia, together with a summary of their biological relevance to leukemia as could be determined from validation with the biomedical literature. Detailed analyses of the relevance of these genes can be found in the Supplementary Material[6]. Altogether, 22 of the 25 top-ranked genes have either a known relation to leukemia or, based on our validation, a potential relation to leukemia; of these, 9 are known markers, 1 is a known therapeutic target, 2 are potential markers, and 10 are potential therapeutic targets.

For comparison, Table 2 also shows the ranks of the above genes in a ranking based on (absolute values of) the t-statistic computed from the AML/ALL distinction[29], and in a ranking based on the similarity of each gene to the 10 positive training genes, computed as the Pearson correlation between the expression vector of each gene and the average expression vector of the positive training genes[2].[g] The average rank of these 25 genes in the t-statistic ranking is 3371.8, and in the Pearson ranking is 2060.8. Conversely, the top 25 genes in the t-statistic ranking have an average rank of 1634.1 in our ranking; the top 25 genes in the Pearson ranking have an average rank of 976.7. The overall rankings are also quite different from each other: the Kendall τ cor-

[f] BLAST website: `http://www.ncbi.nlm.nih.gov/BLAST/`.
[g] For a fair comparison, genes used for training were removed before calculating the t-statistic and Pearson ranks.

Table 3. Colon cancer results: 25 top-ranked genes. Relevance summary: ♦ Known marker; ◊ Potential marker; ■ Known therapeutic target; □ Potential therapeutic target; × No link found. See Supplementary Material[6] for detailed analyses of the relevance of these genes. Ranks of the genes in rankings based on the t-statistic[29] and Pearson correlation[2] are shown for comparison; see text for details.

	Gene	Relevance Summary	t-Statistic Rank	Pearson Rank
1.	26 kDa cell surface protein TAPA-1	□	858	1292
2.	Id1	□	1357	140
3.	Cleavage and polyadenylation specificity factor	×	290	1585
4.	Interferon-inducible protein 9-27	◊	90	1394
5.	Nonspecific crossreacting antigen	♦	202	1775
6.	cAMP response element regulatory protein (CREB2)	□	684	527
7.	MHC class I HLA-Bw58	×	1867	1339
8.	Translational initiation factor 2 gamma subunit	×	101	1265
9.	Splicing factor (CC1.4)	□	463	545
10.	Nucleolar protein (B23)	◊	7	1606
11.	Lactate dehydrogenase-A (LDH-A)	◊	447	670
12.	Guanine nucleotide-binding protein G(OLF)	□	707	396
13.	LI-cadherin	◊	1431	72
14.	Lysozyme	♦	128	1845
15.	Prolyl 4-hydroxylase (P4HB)	◊	358	952
16.	Eukaryotic initiation factor 4AII	□	1163	253
17.	HLA class I histocompatibility antigen	×	934	454
18.	Interferon-inducible protein 1-8D	□	308	1447
19.	Very long chain acyl-CoA dehydrogenase	×	1703	170
20.	Dipeptidase	♦	721	1886
21.	Heat shock 27 kDa protein	□	645	946
22.	Tyrosine-protein kinase receptor TIE-1 precursor	◊□	596	926
23.	Mitochondrial import receptor MOM38	×	1928	197
24.	Mitochondrial matrix protein P1 precursor	□	3	1614
25.	Eukaryotic initiation factor EIF-4A homolog	□	386	1126

relation between the ranking learned by our method (over all 6962 genes) and that based on the t-statistic is 0.0361, and between our ranking and the Pearson ranking is 0.2229. This suggests that all these approaches could be used in a complementary manner to identify new genes.

For example, the top-ranking gene identified by our method, KIAA0220, would not be pulled out by either the t-statistic or the Pearson correlation. A BLAST search revealed that the protein coded for by this gene is homologous to PI3-kinase-related kinase SMG-1. The molecular function of the encoded protein is not yet known, but its homology to PI3-kinase makes it an exciting pharmacological target: the dysregulation of the PI3-kinase signaling pathway has been implicated in multiple cancer types[24], and pharmacological agents targeting this pathway are currently in clinical trials. This suggests that the protein encoded by KIAA0220 could possibly evolve as a similar target for the therapeutic management of leukemia. Indeed, we recently screened for the expression of isoforms of this gene in a human leukemia cell line, and real-time polymerase chain reaction (PCR) analysis revealed an up-regulation of the mRNA transcripts of this gene by several folds as compared with expression in normal cells. We are currently conducting further biological characterizations of the functions of this gene; these studies will be reported elsewhere.

The top 25 genes in the ranking learned for colon cancer are shown in Table 3; again, detailed analyses for all these genes can be found in the Supplementary Material[6]. In this case, 19 of the 25 top-ranked genes have a known or potential relation to colon cancer; of these, 3 are known markers, 6 are potential markers, and 11 are potential therapeutic targets (one is both a potential marker and a potential therapeutic target). Again, for comparison, we also show the ranks

of these genes in a ranking based on (absolute values of) the t-statistic computed from the tumor/normal distinction[29], and in a ranking based on the Pearson correlation between the expression vector of each gene and the average expression vector of the 10 positive training genes[2]. In this case, the average rank of these 25 genes in the t-statistic ranking is 695.1, and in the Pearson ranking is 976.9. Conversely, the top 25 genes in the t-statistic ranking have an average rank of 848.8 in our ranking; the top 25 genes in the Pearson ranking have an average rank of 758.6. As before, the overall rankings are also quite different: the Kendall τ correlation between the ranking learned by our method (over all 1934 genes) and that based on the t-statistic is 0.1114, and between our ranking and the Pearson ranking is 0.1731, suggesting again a complementary role for these approaches in identifying relevant genes.

5. DISCUSSION

We have proposed a new approach for ranking genes by relevance to a disease. Our approach makes use of recent developments on ranking methods in machine learning; specifically, we have shown that the gene ranking problem is naturally formulated as a bipartite ranking problem in machine learning. We have demonstrated our approach on microarray expression data for leukemia and colon cancer; in both cases, our ranking method has identified several genes that were not identified by previous methods. For example, the KIAA0220 gene, which was ranked highest by our method for leukemia, has shown promising results in preliminary screening experiments in human leukemia cells. This gene is especially exciting due to its homology to PI3-kinase, which has been implicated in various other types of cancer and is currently being targeted for drug development. We are currently conducting further biological characterizations of the functions of this gene.

Our approach has several advantages compared with previous gene ranking approaches. In the context of microarray data, our approach does not make any assumptions about the nature of the biological samples, unlike approaches that assume the samples come from two different classes. More generally, our approach exploits existing biological knowledge not only in the form of positive training genes known to be relevant to a disease, as is done by similarity-based ranking methods, but also in the form of negative training genes known to be irrelevant.

The overall approach of using ranking methods in machine learning is highly flexible and can be used in conjunction with a variety of data sources. For vector-valued data, one can use the RankBoost algorithm we have used in our experiments, or a variety of other learning methods. For other types of data, one can use the RankBoost algorithm with an appropriate base function class $\mathcal{F}_{\text{base}}$, or other learning methods such as kernel-based ranking methods with an appropriate kernel[19, 20, 3]. For example, for graph or network data, such as protein-protein interaction data, one can use graph ranking methods that effectively derive a kernel from the Laplacian matrix of the input graph[3]. One can also use ranking methods in machine learning in conjunction with multiple data sources: for example, by first learning a ranking from each individual data source, and then combining the rankings using methods such as those of Aerts et al.[2], or by combining kernels corresponding to different data sources[22] and directly learning a single ranking using the combined kernel.

Our study suggests that ranking methods in machine learning could emerge as a powerful tool for mining biological data sources for the identification of genes relevant to a particular disease.

Acknowledgments

We would like to thank an anonymous reviewer for pointing us to some relevant references. This work was supported in part by NSF award DMS-0732334 (to SA), and an Era of Hope Scholar Award from the Department of Defense and a grant from the Mary Kay Ash Charitable Foundation (to SS).

References

1. E. A. Adie, R. R. Adams, K. L. Evans, D. J. Porteous, and B. S. Pickard. Speeding disease gene discovery by sequence based candidate prioritization. *BMC Bioinformatics*, 6:55, 2005.
2. S. Aerts, D. Lambrechts, S. Maity, P. Van Loo, B. Coessens, F. De Smet, L.-C. Tranchevent, B. De Moor, P. Marynen, B. Hassan, P. Carmeliet, and Y. Moreau. Gene prioritization through genomic data fusion. *Nature Biotechnology*, 24(5):537–544, 2006.

3. S. Agarwal. Ranking on graph data. In *Proceedings of the 23rd International Conference on Machine Learning*, 2006.
4. S. Agarwal, T. Graepel, R. Herbrich, S. Har-Peled, and D. Roth. Generalization bounds for the area under the ROC curve. *Journal of Machine Learning Research*, 6:393–425, 2005.
5. S. Agarwal and P. Niyogi. Generalization bounds for ranking algorithms via algorithmic stability. *Journal of Machine Learning Research*, 10:441–474, 2009.
6. S. Agarwal and S. Sengupta. Ranking genes by relevance to a disease: Supplementary material. Available from http://web.mit.edu/shivani/www/Papers/2009/csb09-supplementary.pdf.
7. U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences of the USA*, 96:6745–6750, 1999.
8. S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215:403–410, 1990.
9. J. Chen, B. J. Aronow, and A. G. Jegga. Disease candidate gene identification and prioritization using protein interaction networks. *BMC Bioinformatics*, 10:73, 2009.
10. J. Y. Chen, C. Shen, and A. Y. Sivachenko. Mining Alzheimer disease relevant proteins from integrated protein interactome data. In *Pacific Symposium on Biocomputing*, volume 11, pages 367–378, 2006.
11. S. Clemencon, G. Lugosi, and N. Vayatis. Ranking and empirical minimization of U-statistics. *Annals of Statistics*, 36:844–874, 2008.
12. W. W. Cohen, R. E. Schapire, and Y. Singer. Learning to order things. *Journal of Artificial Intelligence Research*, 10:243–270, 1999.
13. K. Crammer and Y. Singer. Online ranking by projecting. *Neural Compuation*, 17(1):145–175, 2005.
14. J. Freudenberg and P. Propping. A similarity-based method for genome-wide prediciton of disease-relevant human genes. *Bioinformatics*, 18 Suppl. 2:S110–S115, 2002.
15. Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2003.
16. Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.
17. T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.
18. I. Guyon, J. Weston, S. Barnhill, and V. N. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1-3):389–422, 2002.
19. R. Herbrich, T. Graepel, and K. Obermayer. Large margin rank boundaries for ordinal regression. *Advances in Large Margin Classifiers*, pages 115–132, 2000.
20. T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 133–142, 2002.
21. B. Krishnapuram, L. Carin, and A. J. Hartemink. Joint classifier and feature optimization for cancer diagnosis using gene expression data. In *Proceedings of the 7th Annual Conference on Research in Computational Molecular Biology*, pages 167–175, 2003.
22. G. R. G. Lanckriet, N. Christianini, L. E. Ghaoui, P. Bartlett, and M. I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72, 2004.
23. N. López-Bigas and C. A. Ouzounis. Genome-wide identification of genes likely to be involved in human genetic disease. *Nucleic Acids Research*, 32(10):3108–3114, 2004.
24. J. Luo, B. D. Manning, and L. C. Cantley. Targeting the PI3K-Akt pathway in human cancer: Rationale and promise. *Cancer Cell*, 4(4):257–262, 2003.
25. X. Ma, H. Lee, L. Wang, and F. Sun. CGI: a new approach for prioritizing genes by combining gene expression and protein-protein interaction data. *Bioinformatics*, 23(2):215–221, 2007.
26. J. L. Morrison, R. Breitling, D. J. Higham, and D. R. Gilbert. GeneRank: Using search engine technology for the analysis of microarray experiments. *BMC Bioinformatics*, 6:233, 2005.
27. N. J. Risch. Searching for genetic determinants in the new millenium. *Nature*, 405:847–856, 2000.
28. C. Rudin, C. Cortes, M. Mohri, and R. E. Schapire. Margin-based ranking meets boosting in the middle. In *Proceedings of the 18th Annual Conference on Learning Theory*, 2005.
29. Y. Su, T. M. Murali, V. Pavlovic, M. Schaffer, and S. Kasif. RankGene: Identification of diagnostic genes based on expression data. *Bioinformatics*, 19(12):1578–1579, 2003.
30. The Gene Ontology Consortium. Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25:25–29, 2000.
31. Y. H. Yang, Y. Xiao, and M. R. Segal. Identifying differentially expressed genes from microarray experiments via statistic synthesis. *Bioinformatics*, 21(7):1084–1093, 2005.