A bunch of random studies on cancer

Adrin Jalali

May 24, 2016

Contents

"All models are wrong; some models are useful."

- George Box

1 Adaptive Learning

Here we talk about adaptive and interpretable methods.

1.1 Challenges in Cancer Data

1.1.1 Cancer Heterogeneity

1.1.2 Batch Effects and Noise

1.2 RatBoost

1.2.1 Background

Over the past few decades, biology has transformed into a high throughput research field, both in terms of the number of different measurement techniques as well as the amount of variables measured by each technique (e.g., from Sanger sequencing to deep sequencing), and is more and more targeted to individual cells [99]. This has led to an unprecedented growth of biological information. Consequently, techniques that can help researchers find important insights into the data are becoming increasingly important. Predicting survival of cancer patients based on measurements from microarray experiments has been a field of great interest, but there is often very little overlap between the important genes or biomarkers identified by different studies [34]. Several reasons have been suggested to explain these findings (e.g., heterogeneity of cancer samples or insufficient sample size). Attempts have been made to incorporate additional information from other sources, such as protein-protein interaction (PPI) networks, to make the predictions more robust [25]. One of the latest approaches integrates network and expression data by introducing a network-induced classification kernel (NICK) [66]. Although this method exhibits state-of-the-art performance, the way it penalizes genes that are connected to not-predictive genes can result in selection of isolated features as important features for prediction. We observed this bias of the method towards isolated nodes on additional experiments on synthesized data as shown in Section 4.2.1. Another issue is that in PPI networks, genes or proteins, which have been known to researchers longer and are well-known, are studied more and therefore have more edges connected to them; whereas less well-known genes and proteins are in sparser areas of the network. This bias might further affect the judgment of methods like NICK that use a PPI networks as an input. Consequently, we rely on the fact that such networks exist between genes and proteins, but we do not take them as input. If there is a dependence between input features, which is the case in many biological settings, our method can benefit from this effect. Otherwise, it is reduced to a standard ensemble method. Furthermore, a central assumption underlying many methods is that all data are drawn from the same unknown underlying distribution. This may not be the case, especially for heterogeneous cancer samples, and in particular not for all measured genes.

In this work, we introduce a method that is aware of this potential bias and utilizes an estimate of the differences during the generation of the final prediction method. For this, we introduce a set of sparse classifiers based on $L1$-SVMs [16], where each set of features used by one classifier is disjoint from the selected feature set of any other classifier. Furthermore, for each feature chosen by one of the classifiers, we introduce a regression model that uses additional features and is based on Gaussian process regression. These regression models are then used to estimate how predictable the features of each classifier are for each test sample. This information can then be used to find a confidence weighting of the classifiers, i.e. up-weighting classifiers with high confidence and down-weighting classifiers with lower confidence, for each test sample. Schapire and Singer show that incorporating confidences of classifiers can improve the performance of an ensemble method [96]. However, in their setting, confidences of classifiers are estimated using the training data and are thus fixed for all test samples, whereas in our setting, we estimate confidences of individual classifiers per given test sample. Another related work includes mixture of experts, in which the model trains a set of neural networks and uses a gating network to set the weights of the networks [55]. One issue with their method is that neural networks with lower performance will not be optimized as much as networks with better performance on training data since the gate module down-weights the error propagated to them. Also training of the gating network is interconnected with the neural network experts and afftects training of those modules. Our method, in contrast, trains each module independently using all training samples, and their reliability does not affect how they are trained. Bayesian hierarchical mixtures of experts takes a more similar approach, but the method is complex, and it has a high time complexity to train the architecture of the hierarchy [14].

We show that this method exhibits state-of-the-art performance for different cancer types, with gene expression or methylation data sets as the input. Since the weighting of the classifiers is customized for each test sample, the estimated confidences can offer insights into the specific characteristics of each individual's cancer. To facilitate interpretation of the model, we then create a visualization of the important genes found through this analysis for each test sample. Additionally, we show how the important genes of the training set can be found using our learning method and cross validation.

Our idea might resemble ensemble feature selection, which involves aggregating multiple feature scores from several scoring mechanisms. These scoring mechanisms vary from being several different methods, to being the same method applied to different parts of the data such as a random cross validation scheme [95]. This idea has been studied further by other researchers and they introduced two different methods to aggregate scores from different models. They use an ensemble of support vector machines which on its own has been used to select features in a given data set in other works [51]. Although we use an ensemble of support vector machines, our goal is not to give a ranking to features of the data set, rather to find multiple parsimonious gene sets that are predictive of the outcome on their own, and use all of them in parallel to predict the outcome.

Similar to this approach, in another work, iRDA uses a different approach and can report multiple parsimonious gene sets [64]. One significant difference between iRDA and our work is that we have an embedded prediction approach using these sets, which iRDA lacks. Furthermore, gene sets are somehow ordered in iRDA according to their "strength", and within each set, redundant genes are removed. In our model redundant genes can be included in two different ways. One is within different individual learners. For example, if genes g_1 and g_2 are both strong but redundant, individual learner 1 might include g_1, and individual learner 2 might include g_2. Also, if there are more redundant or related genes in the gene pool, they will be used to estimate how reliable g_1 and g_2 are. Therefore instead of dismissing them, we exploit the fact that they exist.

Related to sorting genes and testing for significance of a reported gene set, Gene Set Enrichment Analysis (GSEA) and its modifications are a commonly used tool [101, 104]. GSEA based methods rank genes depending on how much they relate to the outcome. The choice of relationship is rather free and can vary from Pearson correlation to mutual information. Then for a given gene set, a p-value is calculated by estimating how often a random gene set appears before the given set on the list. There have been several modifications and improvements to the method [75, 31]. Although it is true that GSEA is used to assess the relevance or importance of a given set to the outcome, we need to remember that a particular gene set might consist of genes that are not necessarily important on their own, but are predictive once considered together. Our method does not consider genes individually whereas GSEA does to sort the genes in the first place. Therefore we believe GSEA based methods are not suitable to assess how well our method performs.

Analysis of NICK

Lavi, et al. modified the standard SVM formulation, which is shown in Formula 2.12, as shown in Formula 4.1 [66]. This formulation adds a penalty function to penalize weight differences if their corresponding features are connected in the given graph. The intuition is that if features are connected in the network, their weight should be somehow similar.

$$
\begin{aligned}
&\min_{\mathbf{w},w_0}\left\{\frac{1}{2}\|\mathbf{w}\|^2+\frac{1}{2}\beta\sum_{(j,k)\in E}(w_j-w_k)^2\right\}\\
&\text{s.t.:}\\
&\quad\forall i\in\{1,\cdots,n\}:(\mathbf{w}\mathbf{x}_i+w_0)y_i\geq 1
\end{aligned}
\tag{1.1}
$$

In the above formulation, E is the set of edges of the given network. They also show how to derive the dual of the above optimization problem as shown in Formula 4.2:

$$
\begin{aligned}
&\max_{\alpha}\left\{\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i^T\mathbf{L})(\mathbf{L}^T\mathbf{x}_j)\right\}\\
&\mathbf{L}\mathbf{L}^T=(\mathbf{I}+\beta\mathbf{B})^{-1}\\
&\text{s.t.:}\\
&\forall i\in\{1,\cdots,n\}:\sum_{i=1}^{n}\alpha_i y_i=0\\
&\forall i\in\{1,\cdots,n\}:\alpha_i\geq 0\\
&\text{Laplacian matrix:}\\
&\mathbf{B}=\mathbf{D}-\mathbf{A}
\end{aligned}
\tag{1.2}
$$

In Formula 4.2, $\mathbf{D}$ is a diagonal matrix having degrees of nodes on its main diagonal, $\mathbf{A}$ is the adjacency matrix, and $\mathbf{B}$ is called the graph Laplacian matrix. The benefit of the above formulation is that the input vectors can be transformed using the matrix $\mathbf{L}$, which itself comes from the Cholesky decomposition of the matrix $(\mathbf{I}+\beta\mathbf{B})^{-1}$. The parameter β in both Formulea 4.1 and 4.2 sets how much we penalize weight differences for connected vertices.

After training the model on the data using the above formulation, we can calculate back the vector $\mathbf{w}$ using Formula 4.3. We use the vector $\mathbf{w}$ to investigate which nodes and pairs of nodes are given a relatively high value compared to other features.

$$
\mathbf{w}=(\mathbf{I}+\beta\mathbf{B})^{-1}\sum_{i=1}^{n}\alpha_i y_i\mathbf{x}_i
\tag{1.3}
$$

We then use calculated feature weights in the vector **w** of both normal and modified SVM (NICK) to show which feature pairs are selected as important in the model, as shown step by step bellow. Please note that NICK transforms the data using the matrix **L**, and then solves a normal SVM on the transformed data, and therefore in the following whenever we refer to *transformed data*, that means NICK method.

1. Solve SVM problem for original and transformed data.
2. Calculate **w** for both models.
3. Compute for each pair of nodes, for each model:

$$Score(i,j) = \frac{|w_i| + |w_j|}{2} \times e^{-max(d_G(i,j),1)} \tag{1.4}$$

4. Report pairs with highest scores for both trained models.

In order to evaluate the method, we need to synthesize some data because in the real data it is not clear which features are the true discriminating features. For this purpose, we randomly generate a graph, and assign nodes to three different classes. Nodes in this graph represent genes/features in the data set. Each feature is a random variable sampled from a Gaussian distribution. If the node is independent of the target class, it gets its value from a Gaussian distribution regardless of the target class. If the feature is selected as a *signal* node, then it takes its value from two different Gaussian distributions that differ in their mean, depending on the target class of the sample. Some of these signal features are connected only to random features, and some are connected to other signal nodes. We call connected signal features a *pathway.* The generated graph and an example of selected feature nodes are shown in Figure 4.1.

To generate our synthesized data set, for each data point, and each feature of that data point, we first assign a class to that data point, and according to the assigned class, we sample from the corresponding distribution, according to the following functions:

- Signal nodes (genes): $f(n) = \begin{cases} N(-\mu, 1) & \text{if } n \text{ is in class 1} \\ N(\mu, 1) & \text{if } n \text{ is in class 2} \end{cases}$
- Random nodes (non-signal genes): $f(n) = N(0, 1)$

We perform three experiments. Each experiment uses the same graph structure, but has a different set of signal nodes. First we put *pathway* nodes on the boundaries of the graph, then we move one *pathway* deep into the graph structure, and in the last experiment all *pathway* nodes are inside the graph structure. These three scenarios are shown in Figures 4.2, 4.3, and 4.4 accordingly.

Figure 4.2(b, c) presents node pairs with highest assigned scores as calculated in Formula 4.4, comparing normal SVM and NICK. Orange and yellow

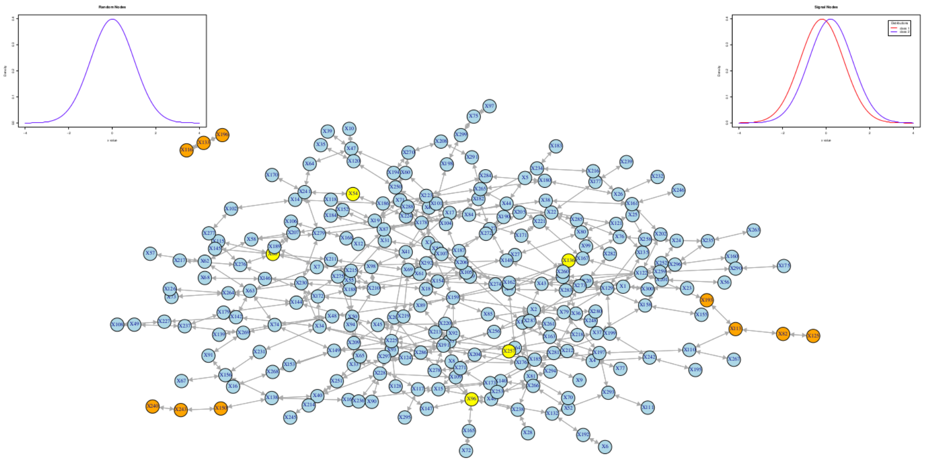


Figure 1.1: Blue: random gene, Orange: Signal node being a member of a pathway of signal nodes, Yellow: A lonely signal node

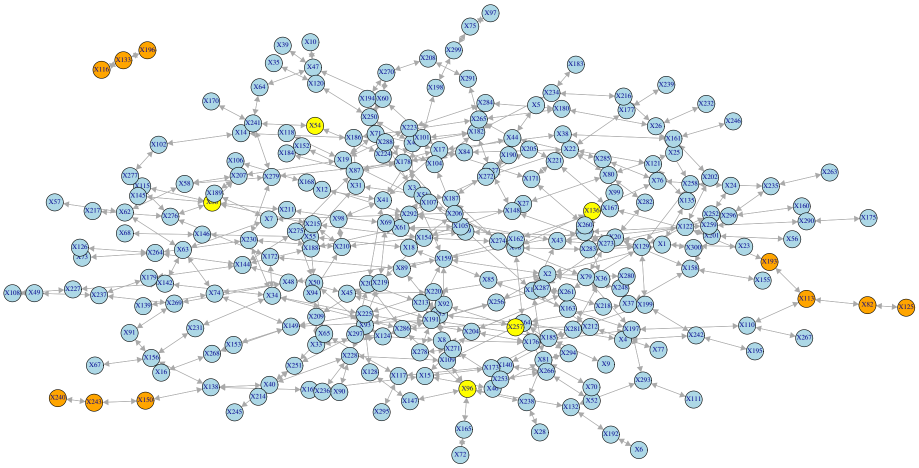
colored cells are pathway and lonely signal nodes in the graph accordingly. This experiment shows discovers signal gene pairs more effectively than a normal SVM, using the graph structure.

Figure 4.3 shows the experiment, in which one of the pathways is located inside the network. As illustrated in Figure 4.3(b, c), NICK method mostly chooses the pathway nodes located on the boundaries of the network.

Figure 4.4 illustrates the fact that non-signal features down-weight and penalize signal features when connected to them. In this example, a normal SVM detects more signal nodes than NICK.

We used the network provided in [7] for NICK to classify Van 't Veer data [111]. As expected, we realize that NICK prefers nodes outside the network to the nodes that are deep into the given network. This is shown in Figure 4.5, comparing preferred nodes in a normal SVM and NICK. The first column is gene ID, and the second is its corresponding degree in the given graph.

These experiments all together, show how such a modification in SVM optimization problem gives a bias towards genes that are not *hubs*. This is problematic considering many of those hubs in the network are partially, if not mostly, hubs due to the fact that they were discovered earlier and investigated the most. Therefore those are the most well-known genes, which in many cases happen to be biologically most relevant genes. A method such as NICK tends to penalized them because they are connected to many genes that are irrelevant to the disease in study. This experiment is our motivation to use the fact that such a biological network exists, but not to use it directly in our method.



(a) Corresponding network

Original			
X196	X196	X53	X53
X233	X233	X39	X39
X88	X88	X196	X133
X116	X116	X127	X127
X197	X197	X127	X148
X148	X148	X150	X150
X148	X273	X116	X133
X160	X160	X96	X96
X95	X95	X273	X273
X88	X115	X40	X40
X53	X8	X53	X164
X195	X195	X56	X56

(b) Discovered nodes (no NICK)

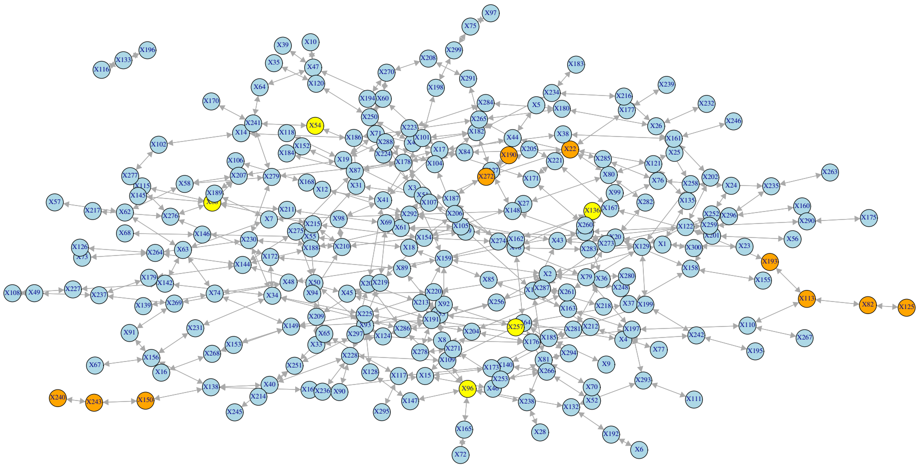
Transformed			
X196	X196	X233	X233
X196	X133	X133	X133
X133	X116	X116	X116
X95	X95	X240	X240
X39	X39	X240	X243
X59	X59	X106	X106
X243	X243	X106	X168
X114	X114	X168	X168
X243	X150	X56	X56
X39	X47	X298	X298
X150	X150	X247	X247
X125	X125	X83	X83

(c) Discovered nodes (NICK)

AUC (Original):	60.6
AUC (Transformed):	62.4
wc p-value (paired):	5.669e-09

(d) Performance measures

Figure 1.2: An easy example: here all signal pathways are on the border of the network.



(a) Corresponding network

Original			
X190	X190	X104	X104
X233	X233	X190	X272
X277	X277	X88	X88
X190	X127	X165	X165
X272	X272	X272	X22
X106	X106	X165	X96
X150	X150	X250	X250
X88	X215	X22	X22
X51	X51	X28	X28
X73	X73	X35	X35
X162	X162	X113	X113
X112	X112	X277	X102

(b) Discovered nodes (no NICK)

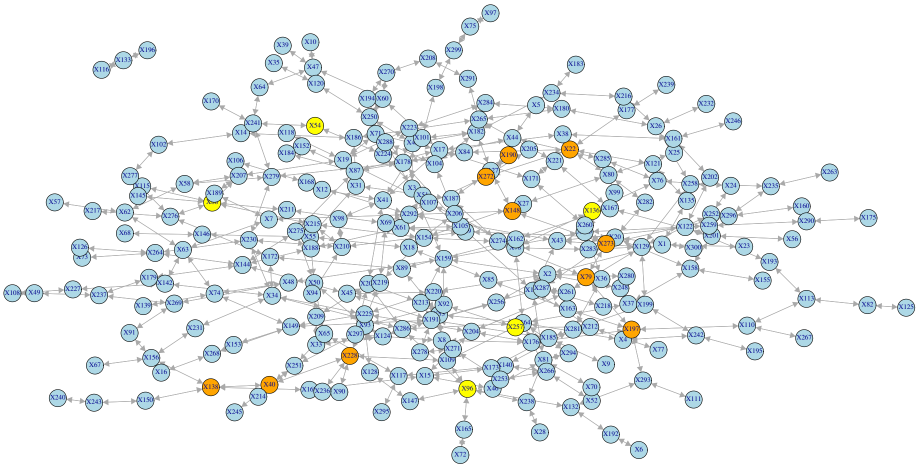
Transformed			
X233	X233	X190	X190
X112	X112	X240	X240
X190	X272	X240	X243
X86	X86	X243	X243
X243	X150	X190	X127
X150	X150	X272	X272
X246	X246	X298	X298
X106	X106	X125	X125
X35	X35	X125	X82
X247	X247	X272	X69
X272	X22	X82	X82
X100	X100	X257	X257

(c) Discovered nodes (NICK)

AUC (Original):	60.1
AUC (Transformed):	61.5
wc p-value (paired):	1.383e-06

(d) Performance measures

Figure 1.3: A medium example: here some signal pathways are on the border of the network.



(a) Corresponding network

Original			
X190	X190	X101	X101
X233	X233	X190	X272
X88	X88	X297	X297
X190	X127	X93	X93
X26	X26	X138	X138
X272	X272	X272	X22
X101	X41	X123	X123
X22	X22	X101	X198
X146	X146	X228	X228
X278	X278	X72	X72
X88	X115	X96	X96
X148	X148	X112	X112

(b) Discovered nodes (no NICK)

Transformed			
X233	X233	X190	X190
X112	X112	X190	X272
X86	X86	X190	X127
X272	X272	X272	X205
X205	X205	X146	X146
X146	X68	X68	X68
X298	X298	X272	X22
X90	X90	X127	X127
X100	X100	X272	X69
X297	X297	X72	X72
X127	X148	X155	X155
X247	X247	X196	X196

(c) Discovered nodes (NICK)

AUC (Original):	60.2
AUC (Transformed):	62.5
wc p-value (paired):	8.151e-13

(d) Performance measures

Figure 1.4: A hard example: here none of the signal pathways are on the border of the network.

Original	
Node	Degree
85453	12
6605	98
56886	26
10640	16
8817	152
56894	28
5733	150
57758	8
7532	86
51	172
7566	16
3267	56
89953	4
5713	126
5193	32
5365	70
10874	132
5982	172
92140	20
332	328

(a) Discovered nodes (no NICK)

NICK	
Node	Degree
9917	0
84279	0
197370	0
51143	0
58475	0
55585	0
25949	0
54892	0
126695	0
57168	0
10456	0
148223	0
9742	0
253558	0
342527	0
10175	0
83930	0
57035	0
145482	0
57465	0

(b) Discovered nodes (NICK)

Figure 1.5: Comparison of selected nodes on Van 't Veer data [111] using NICK and a normal SVM.

1.2.2 Methods

Materials

Data Sources: In this article, our method is applied to two different data types: gene expression data and DNA methylation data, which we retrieved from The Cancer Genome Atlas (TCGA) [108]. TCGA is a joint effort of the National Cancer Institute and the National Human Genome Research Institute to advance the understanding of the molecular basis of cancer. They provide access to the different measurements from cancer samples that have been analyzed to external researchers. Samples are categorized according to diagnosed cancer from which we use the following groups:

- *Acute Myeloid Leukemia (LAML)* [110]: At the time of writing, the data set includes 200 samples. 194 samples contain methylation data and we use the part of the data measured by JHU-USC HumanMethylation450 arrays. 173 samples contain mRNA data measured by HG-U133 arrays. In this article the methylation data is referred to as TCGA-LAML. Among available characteristics of samples, “risk group” and “vital status” are chosen as target classes. These labels show the aggressiveness of the disease. In our analysis, regarding risk group, {favorable} and {intermediate/normal, poor} samples form our two group, and in the analysis of vital status, {alive} and {dead} samples form our two groups of samples.

- *Breast invasive carcinoma (BRCA)* [109]: This data set includes 993 samples with clinical data, and we use the methylation data component measured by JHU-USC HumanMethylation450 arrays. Only very few samples in this data set are indicated as having metastasized (8 samples). Hence the data are analyzed according to “tumor size”, “affected nearby lymph nodes”, “stage”, and “estrogen receptor”. Estrogen receptor was shown to be an important factor in prognosis [61], and along with other factors directly affects the decision for therapy [49, 76]. For tumor size {T1, T2} samples are one category and {T3, T4} the other category; in order to analyze affected nearby lymph nodes, {N0} is compared to {N1, N2, N3}; stage is analyzed as having {stage I, stage II} vs. {stage III} samples. Estrogen receptor status of samples is either positive or negative, and they form our two classes.

Data Preprocessing: To prepare gene expression data for analysis, microarray probes are mapped to their respective gene. If there are multiple probes for a gene, the median reported gene expression value of those probes is adopted as the gene expression for that gene.

Preparing the methylation data, we use the nearby gene for each methylation site available for each sample and each methylation site. The median beta value of methylation sites mapped to each gene is taken as the methylation value of the corresponding gene. In this process only methylation sites located on the promoter region of a gene are considered and others are discarded.

Learning a Mixture of Disjoint Classifiers

When dealing with cancer, we need to consider the fact that tumors of the same type of cancer can be very different in nature and they are usually classified as different cancer subtypes. In fact, even one single tumor can be very heterogeneous [54]. This means that the malignancies causing the cancer to happen are genetically different between subtypes, or even within subtypes, and it is possible to have multiple underlying cellular processes causing a particular cancer. Also it is important to note that the nature of our given data is such that the input features are properties measured from genes, e.g. gene expression or methylation values, and these variables are correlated and statistically dependent on each other. Our method tries to exploit these properties of the problem to infer an interpretable model with state-of-the-art performance.
Our method can be characterized by the following key parts:

Training phase:

- Fit several individual classifiers to the data, in such a way that the features of the data they use are disjoint sets.

Prediction phase:

- Calculate the prediction confidence of each individual classifier by:
  - Estimating the reliability of input features of the classifier;
  - Estimating the confidence of the output based on the decision values.
- Calculate a weighted prediction label based on the individual classifier confidences.

Properties of the Individual Classifiers: A wide variety of classifiers can be used within our framework. One requirement is that the classifier is regularized (i.e., the stronger the regularization, the less complex the model gets and consequently the less features are used). The classifier is also required to report the probability of its calculated output, or to give a decision value according to which it chooses the predicted class. We use an $L1$ regularized SVM for this purpose with a linear kernel [16]. The $L1$ regularization makes the SVM sparse, i.e. using only a few input features, and the linear kernel allows us to infer which features are used in the decision function of the SVM after it is fit to the data.

Training the Individual Classifiers: The model starts with no individual classifier and an empty set of excluded features. In each step, the excluded set of features is removed from the data, then a classifier is fit to the data. Next the features used by the most recent trained classifier are added to the excluded set. In the case of a linear kernel SVM, this is achieved by finding features with a non-zero coefficient in the model. This way the features being used by classifiers are disjoint and might represent different underlying causes of groups into which samples are to be classified.

Combining Classifiers by Estimating Confidences of Individual Predictors: Given a set of classifiers, the question is how to combine them to come up with a joint prediction value for each test sample for which we want to predict the output label. The intuition behind combining the classifiers is to put more weight on classifiers that use features whose behavior is similar to the training data. This is motivated by the fact that some parts of the test data might behave very differently to the training data, meaning that a classifier using these features should have lower performance than a classifier using features that are distributed similarly to the training data. Therefore we need to evaluate the reliability of the input features of each individual classifier. In scenarios like gene expression or methylation analysis, we usually have many input features. Furthermore, many features are correlated and statistically dependent. The idea of our new method is to build separate prediction models for each feature of each classifier. These prediction models can then be used to obtain a confidence for the feature in a given test sample. These confidences can then be combined for each classifier to give a weighting of the classifiers for the given test sample. To evaluate an observed feature f, we try to choose a few statistically dependent features, and fit a model to predict f. To find these features, first the estimated maximal information coefficient (MIC) of all other features with feature f is calculated [91]. Then, features having MIC value within the top 5% or the 5 features with highest MIC with f (if the top 5% features consist of less than 5 features), are selected as predictors of f. Given a test sample, the closer the predicted value of f is to the observed value, the more reliable it is. To quantify this, we need to not only know the predicted value of the feature, but also a confidence interval for that prediction. This can be achieved using Gaussian processes, which give the mean and variance of the posterior probability under the condition of observed values for selected features. A weighted average of these values gives us the overall reliability of the features of an individual classifier. A schematic view of the trained classifier is shown in Fig. 4.6.

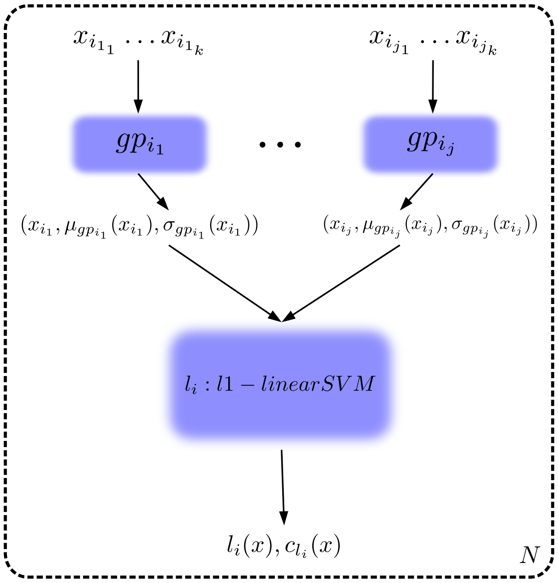

$x_{i_{1_1}} \dots x_{i_{1_k}}$
$x_{i_{j_1}} \dots x_{i_{j_k}}$
gp_{i_1}
$\dots$
gp_{i_j}
$(x_{i_1}, \mu_{gp_{i_1}}(x_{i_1}), \sigma_{gp_{i_1}}(x_{i_1}))$
$(x_{i_j}, \mu_{gp_{i_j}}(x_{i_j}), \sigma_{gp_{i_j}}(x_{i_j}))$
$l_i : l1 - linearSVM$
$l_i(x), c_{l_i}(x)$
N

Figure 1.6: Schematic view of the method

In addition to the confidence in the classifier estimated by looking at the confidences of its individual features, we also account for the confidence that the classifier has in the prediction label of the test sample. If the method supplies such a confidence value (e.g., Gaussian processes), we can directly use it. Otherwise, we estimate it using the decision value. In our setting, the linear SVM gives a decision value whose sign defines the predicted class. Using these values we estimate a confidence for each individual classifier. Several approaches exist for deriving a confidence from the decision values [67]. Whether these or other additional methods could lead to further improvements of our method, will be topic of further study.

More formally speaking, define X to be the set of input samples, X_s to be the input vector of sample s, y_s and $\hat{y}_s$ to be respectively the original label and predicted output of sample s, Δ to be the set of individual classifiers, l_i to be an individual classifier, Φ_{l_i} the set of input features of classifier l_i, $l_i(X_s)$ to be the label predicted by classifier l_i for sample X_s, and f to be a feature, $X_{s,f}$ to be the observed value of feature f in sample X_s, $|w_{l_i}(f)|$ to be the absolute value of the weight of feature f in the decision function of classifier l_i, and g_f to be the Gaussian process predicting feature f using feature set Φ_f. Also $\mu_{g_f(X_s)}$ and $\sigma_{g_f(X_s)}$ are the mean and standard deviation of the posterior probability given by Gaussian process g_f under the condition of observing values of features in Φ_f, and μ_{l_i} and σ_{l_i} are respectively the expected mean and standard deviation of the decision value of classifier l_i. Here F is the cumulative distribution function of a standard normal distribution.

The training phase of the model is shown in Fig. 4.7, in which, N is the number of individual learners to be included in the model, Φ_l is the union over all Φ_{l_i} and $X_{-\Phi_l}$ is the input X after discarding all features of the set Φ_l. TOP is the function which selects the maximum of the top 5 and top 5% features f' of all features ordered by MIC with feature f.
Now given a test sample X_s, the estimated confidence of a feature f is:

$$c_f(X_s) := 2 \cdot F\left(-\left|\frac{X_{s,f} - \mu_{g_f(X_s)}}{\sigma_{g_f(X_s)}}\right|\right) \tag{1.5}$$

Then the overall feature reliability or confidence of a classifier l_i is estimated as:

$$c^1_{l_i}(X_s) := \frac{\sum_{f \in \Phi_{l_i}} c_f(X_s) \cdot |w_{l_i}(f)|}{\sum_{f \in \Phi_{l_i}} |w_{l_i}(f)|} \tag{1.6}$$

Also the estimated output confidence of the classifier l_i is:

$$c^2_{l_i}(X_s) := 1 - 2 \cdot F\left(-\left|\frac{l_i(X_s) - \mu_{l_i}}{\sigma_{l_i}}\right|\right) \tag{1.7}$$

and the final confidence of the classifier l_i is then:

$$c_{l_i}(X_s) := c^1_{l_i}(X_s) \cdot c^2_{l_i}(X_s) \tag{1.8}$$

Finally, the predicted class $\hat{y}_s$ is calculated as the sign of a weighted vote among individual classifiers:

$$\hat{y}_s := \text{sign}\left(\frac{\sum_{l_i \in \Delta} c_{l_i}(X_s) \cdot l_i(X_s)}{\sum_{l_i \in \Delta} c_{l_i}(X_s)}\right) \tag{1.9}$$

Visualization of Model Predictions

The interpretation of the model can be understood on two different ways. First we assume for a given training data set, the model is trained and a new test sample is given. For the given test sample it is possible to visualize the reliability of each used feature in individual classifiers, as well as the overall confidence

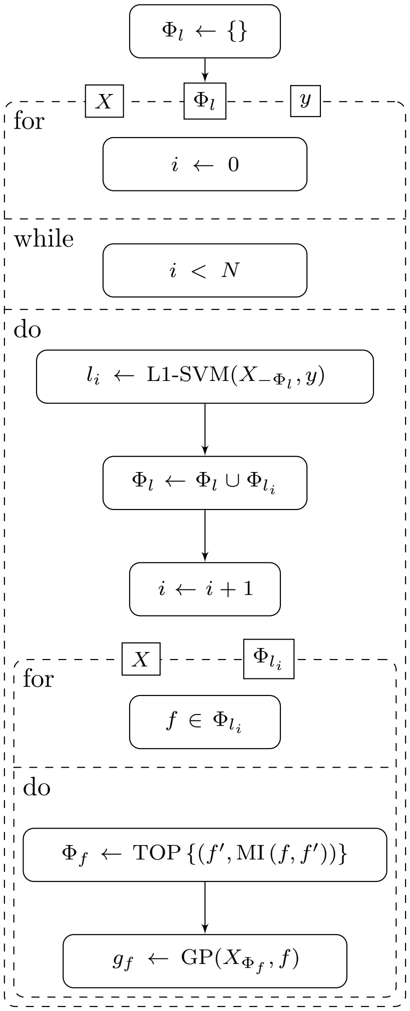

$\Phi_l \leftarrow \{\}$
X
Φ_l
y
for
$i \leftarrow 0$
while
$i < N$
do
$l_i \leftarrow$ L1-SVM$(X_{-\Phi_l}, y)$
$\Phi_l \leftarrow \Phi_l \cup \Phi_{l_i}$
$i \leftarrow i + 1$
X
Φ_{l_i}
for
$f \in \Phi_{l_i}$
do
$\Phi_f \leftarrow \mathrm{TOP}\,\{(f', \mathrm{MI}\,(f, f'))\}$
$g_f \leftarrow \mathrm{GP}(X_{\Phi_f}, f)$

Figure 1.7: UML activity diagram of the training process

of each individual classifier. Used features can be superimposed onto a PPI network as well as their reliability and the confidence of their respective individual classifier.
Gene expression and methylation level measurements from cancer samples are usually very noisy. Furthermore, cancers are usually very heterogeneous. Additionally, there might be different subgroups for each interesting group (e.g., cancer stage), for which the importance of the features also differs. To get a global picture of the important features, we therefore evaluate how often certain features are selected by the classifiers using 100 random train test partitionings with 80% of the data for training and 20% of the data for testing. To visualize high confidence relationships between features, we create a graph which has a node for every chosen feature in any of the 100 train partitions in any of the individual classifiers. The weight of an edge (s, t) is defined as the number of times the respective features have occurred together in an individual classifier. Then, all edges with low weights are discarded. In order to find a threshold to prune edges according to their weights, a Gaussian kernel density estimate is fit to the weights of the edges, and the threshold is chosen at the 90th percentile. Nodes that have an appearance frequency higher than the threshold are labeled by their gene names and edges having a higher weight than the threshold are kept in the graph.

For illustration purposes, choosing the regularization parameter is done in a way to maximize the number of genes selected with high confidence, as well as minimizing the number of genes pruned out in the process. It is important to remember that considering the results of the method under different regularization parameters is essential to make sure the selected genes possess a high confidence and are also stable regardless of sampling of the training data set.

Implementation Details

To compare the performance of our method with other methods, the implementations present in Python *scikit-learn(0.14)* package are taken. In the case of stochastic gradient boosting, the representing class is *GradientBoostingClassifier*, the number of classifiers is set to 100, and to make it sparse and prevent over-fitting, the maximum number of features for splits in trees is set to 5, and the maximum number of layers is set to 2. For AdaBoost, *AdaBoostClassifier* is used, which is an implementation of AdaBoost-SAMME [125], with weak learner set to *DecisionTreeClassifier* with maximum depth set to 2, and the number of weak classifiers set to 100. Parameters of the two boosting algorithms are chosen by a grid search on their parameter space over all the data sets and selecting the parameter sets which give a robust and stable result over all experiments.

As an SVM, ν-SVM with $\nu = 0.25$ is used, once with a linear kernel, and once with an RBF kernel; γ parameter of the RBF kernel is set to $(\text{num of features})^{-1}$. The ν parameter is set to the maximum value for which the optimization function is solvable with *libsvm* for all analyzed data sets [21]. Smaller values cause the SVM to overfit to the data and not generalize well. The Gaussian process's correlation function is a squared-exponential, and MIC is estimated using *minepy* package [6].

The PPI network used in our analysis is from the Human Protein Reference Database (HPRD) [84]. Almost all edges and relationships between proteins that are added to this database are manually extracted from literature by biologists, hence it has a lower rate of edges included in the database for which there is no evidence in the literature.

1.2.3 Results and discussion

Interpretability of Predictions

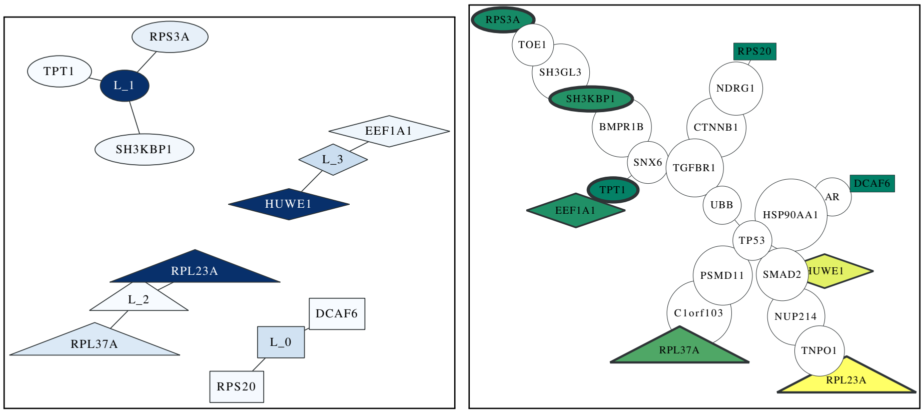
Here we present the results of running the method on the TCGA-LAML gene expression data set.

Visualization of Features Important for a Particular Test Sample: Having a model trained on the data, and given a test sample, it is possible to infer and visualize which individual classifier(s) is (are) influencing the prediction most. To this end, individual learners as well as the features they use are visualized as in Fig. 4.8(a). In this figure, nodes with labels starting with "*L_*" represent individual classifiers, and other nodes are labeled with their respective gene name. The color of the node shows its confidence compared to other nodes; the darker the node, the higher the confidence. In the case of a gene, it is the confidence or reliability of the feature (c_f), and in the case of an individual classifier, it is the overall estimated confidence (c_{l_i}). Edges show which classifier is using which genes in its decision function. The shape of a node represents the individual classifier they belong to.

To get a better overview of the individual features that were chosen by the classifiers for the particular test sample, we visualized the corresponding genes on a graph containing information about the PPI network in Fig. 4.8(b). We extracted the PPI information from HPRD as explained before. This way, it is possible to find over- or under-regulated pathways that might be responsible for the label (e.g., cancer stage) of the test sample. Since PPI networks can be quite dense, we removed parts of the induced network. For this purpose we computed each shortest path between all pairs of selected features. Then, the minimum spanning tree of that section was plotted, after removing branches with no selected feature.

Most of the features chosen by any of the classifiers (colored nodes) are not connected to any other chosen feature. It is known that there is in many cases a correlation between expression value of the genes whose corresponding proteins interact [56]. Therefore, a regularized model will only choose a subset of the correlated features. This explains the observation that features selected by a single model can be distant from each other on a PPI network; but if multiple disjoint sparse models are fit to the data, their selected features might happen to be close to each other on the PPI network (e.g., node TPT1 and node EEF1A1 in Fig. 4.8(b)).

It is worth noting that these plots are the result of analyzing one single given test sample. Therefore in practice, these interpretations can be used for each



(a)

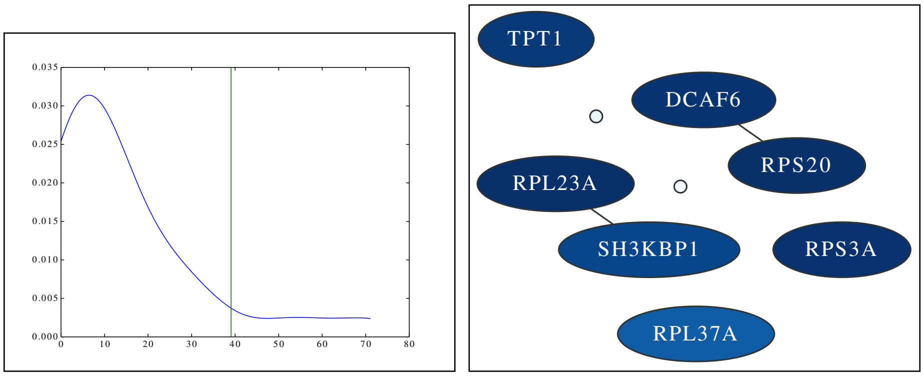
(b)

Figure 1.8: **Visualization of one model** A sample model for TCGA-LAML gene expression data **(a)** individual classifiers and their selected features; higher confidence of a node is shown by a darker color, **(b)** selected genes plotted over the PPI network; green and yellow show low and high confidence respectively, and the thickness of the border of the node shows the respective confidence of the individual classifier to which it belongs.

patient and if useful, influence the treatment that the oncologist prescribe for the patient.

Visualization of Important Global Features: As explained in Section 4.2.2, a graph is created from model structures of all 100 random training partitions, and then it is pruned to keep only high confidence nodes and edges. The density estimation of the graph edge weights and the pruned graph are plotted in Fig. 4.9 where the nodes with labels are the ones that are not pruned. The nodes in this figure that do not have any label, are the ones with frequency lower than the corresponding threshold. Among the features considered to be important were features that had previously been linked to leukemia such as SH3KBP1 [1].

What was more intriguing to see was that four out of the seven important features of the TCGA-LAML gene expression data set contained ribosomal proteins when using the risk group label, i.e. RPL37A, RPS20, RPS3A, and RPL23A. For a long time ribosomes were just considered machines that perform an unbiased translation of genes from mRNA to amino acid sequences, but this view has recently been challenged [121]. One new hypothesis is that the ribosome introduces an additional regulatory layer. Therefore, it could very well be that mutations in ribosomal proteins can lead to a misregulation of expression levels of important genes and ultimately to the development of cancer (in this case leukemia). One of the ribosomal proteins we found was RPL23A. It has been shown that loss of RPL23A can impede growth and lead to morphological abnormalities in Arabidopsis Thaliana [121]. Therefore, a mutation in RPL23A



(a)

(b)

Figure 1.9: **(a) Determine pruning threshold** Threshold is determined by finding the point after which, 90% of the area under the curve is observed from left to right. The horizontal axis shows the observed frequency or weight of the edges. **(b) Important Global Features** High confidence nodes and edges of the graph generated from the model on TCGA-LAML gene expression data. Darker color represents higher rate of being selected by a classifier.

might also have severe effects in humans. A missense mutation in RPL23A was recently found in patients having Diamond-Blackfan anemia, which is an inherited form of pure red cell aplasia (related to leukemia) [48]. Note that the model for LAML has low performance for the regularization value chosen. Nevertheless, the features shown here are also the ones with the highest confidence for models learnt with less regularization (with several other additional features). The models with less regularization show similar performance to the other methods shown in Fig. 4.10

Performance comparison

The performance of the method was compared with that of two ensemble methods, AdaBoost and stochastic gradient boosting, as well as an SVM with linear kernel, and an SVM with an RBF kernel. We also included our implementation of the NICK method [66]. We randomly partitioned the data into training and test sets with 80% of the data for training and 20% of the data for testing. To compare the performance of the different methods, Area Under the receiver operating characteristic Curve (AUC) [33] was calculated on the test set over the decision values returned by the methods on the individual samples. The process was repeated 100 times to reduce random effects. As seen in Fig. 4.10, overall performances of all methods are comparable. In some cases a single SVM works better, in some other cases ensemble algorithms give a better performance. However, in most cases an improvement in performance is observed by adding individual learners to the model, with the greatest gains due to the first few individual learners added to the model. In two cases, TCGA-LAML/Vital status and TCGA-LAML/Risk Group, our reported performance measures are significantly lower than other methods. This, however, comes from the fact that we have enforced extreme sparsity measures. The performance of the method increases and reaches the other methods' performance levels if this constraint is relaxed, as reported in supplementary 1. We enforced those sparsity measures for all models to avoid over-fitting. Optimizing the sparsity constraint via cross-validation would have been computationally expensive, which is why we preferred to be conservative. Had we optimized the sparsity constraint, we would have still been able to find the significant features while having similar performance as the other methods.

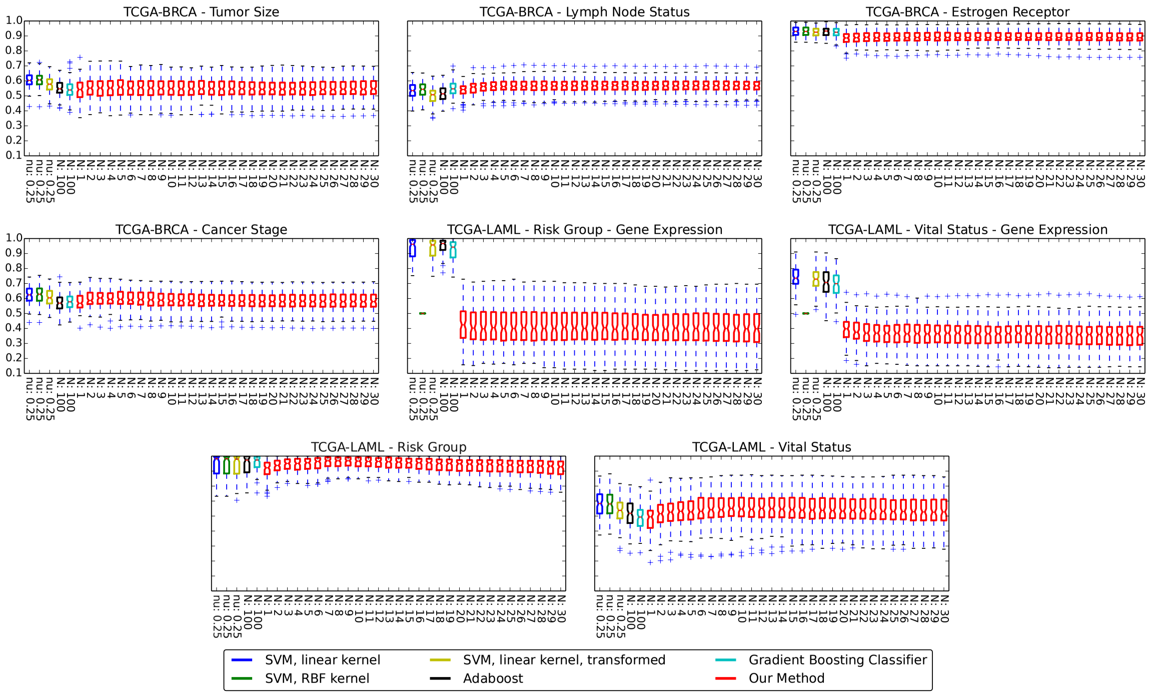


Figure 1.10: **Performance Summary (AUC)** Each box shows a 25–75% interval, as well as the median, which is shown as a horizontal line in each box.

1.2.4 Conclusions

Machine learning has become more and more popular in many real world scenarios for making sense of large collections of facts. Differences between the data used for training the method and new data for which the label should be predicted can limit the performance of prediction methods on those data. In this work we introduced a method that estimates these potential partial biases and incorporates them into the prediction function. We applied it to gene expression and DNA methylation measurements from cancer patients. Our method has state-of-the-art performance on many different prediction tasks. Furthermore, we show how to make sense of the predictions. Visualizing the important genes can lead to new biological insights, as shown for the TCGA-LAML data set with the risk group label. Instead of mapping the genes to PPI networks, one could also think of mapping them to signaling pathways [59].

Recently, a study showed that most published signatures are not significantly more associated with cancer outcome than random signatures [114]. One of the reasons for this finding is that the data comes from slightly different underlying hidden data distributions. Since our new method estimates this bias and corrects for it by up-weighting the classifiers that have higher confidence, we expect that it should be less susceptible to such differences in the data.

In this work we designed and developed a method that besides being a predictive model, it can be used for two different purposes. It can be used as an exploratory method to reveal potential features used in future studies; and it can be used to different underlying causes of the same disease and with its interpretability help oncologists to choose the treatment accordingly.

We would like to point out that the applicability of our method is not limited to cancer outcome prediction, and it can apply to many more scenarios. The method assumes that the data has enough features to select from, and that there are related features to those selected ones that can be used to estimate their reliability. These are conditions that almost all biological data satisfy, hence the method can be applied to them.

The method also works as a skeleton whose components can be easily substituted. For example, by changing the classifier used in individual learners to a multi-class classifier, the method would work on multi-class problems. For the sake of simplicity and without loss of generality we performed the evaluations only on binary classification problems. Also, due to the structure of our model, one possible approach would be to use a method such as iRDA and use those gene sets as features of individual learners. Whether this approach leads to better results or not requires further research. Also, the combination of maximal information coefficient and Gaussian processes is not the only feasible option, and they can be replaced with other faster methods if the time complexity of the method is of any concern. Some of these alternatives are already available on the *github* repository of the method.

1.2.5 Enhancements and Parameter Setting

In order to withdraw the hassle of parameter setting from our method, we utilize a nested cross-validation scheme to automatically search for and find best parameters for a given data-set. A nested cross-validation scheme tests the method with different parameters several times, and therefore the method must be fast enough for it to be feasible. As a result, we modified and enhanced the method.

In our method, the most time consuming part is MIC calculation for output against all the features. We included this part because Gaussian Processes in the normal setting, having as many features as we have, overfit to the data, and therefore a pre-selected small subset of features is what we feed to each Gaussian Process. Another way of reducing the complexity of a GP is to use a covariance function such as squared exponential with automatic relevance determination (ARD) covariance function [90], which we decided against after observing its running time.

1.3 Raccoon

List of Tables

List of Figures

Bibliography

[1] J. Adélaïde, V. Gelsi-Boyer, J. Rocquain, N. Carbuccia, D. J. Birnbaum, P. Finetti, F. Bertucci, M. J. Mozziconacci, N. Vey, D. Birnbaum, and M. Chaffanet. Gain of CBL-interacting protein, a possible alternative to CBL mutations in myeloid malignancies. *Leukemia*, 24(8):1539–41, Aug. 2010.

[2] N. Aghaeepour, P. K. Chattopadhyay, A. Ganesan, K. O'Neill, H. Zare, A. Jalali, H. H. Hoos, M. Roederer, and R. R. Brinkman. Early immunologic correlates of HIV protection can be identified from computational analysis of complex multivariate T-cell flow cytometry assays. *Bioinformatics*, 28(7):1009–1016, 2012.

[3] N. Aghaeepour, G. Finak, H. Hoos, T. R. Mosmann, R. Brinkman, R. Gottardo, R. H. Scheuermann, et al. Critical assessment of automated flow cytometry data analysis techniques. *Nature Methods*, 10(3):228–238, 2013.

[4] N. Aghaeepour, A. Jalali, K. O'Neill, P. K. Chattopadhyay, M. Roederer, H. H. Hoos, and R. R. Brinkman. RchyOptimyx: Cellular hierarchy optimization for flow cytometry. *Cytometry Part A*, 81(12):1022–30, 2012.

[5] N. Aghaeepour, R. Nikolic, H. Hoos, and R. Brinkman. Rapid cell population identification in flow cytometry data. *Cytometry Part A*, 79(1):6–13, 2011.

[6] D. Albanese, M. Filosi, R. Visintainer, S. Riccadonna, G. Jurman, and C. Furlanello. minerva and minepy: a c engine for the mine suite and its r, python and matlab wrappers. *Bioinformatics*, 29(3):407–408, 2013.

[7] A.-L. Barabasi and Z. N. Oltvai. Network biology: understanding the cell's functional organization. *Nature reviews genetics*, 5(2):101–113, 2004.

[8] A. Bashashati, N. Johnson, A. Khodabakhshi, M. Whiteside, H. Zare, D. Scott, K. Lo, R. Gottardo, F. Brinkman, J. Connors, et al. B cells with high side scatter parameter by flow cytometry correlate with inferior survival in diffuse large b-cell lymphoma. *American Journal of Clinical Pathology*, 137(5):805–814, 2012.

[9] R. Bellman. On a routing problem. Technical report, DTIC Document, 1956.

[10] S. Bendall, G. Nolan, M. Roederer, and P. Chattopadhyay. A deep profiler's guide to cytometry. *Trends in Immunology*, 33(7):323–332, 2012.

[11] S. Bendall, E. Simonds, P. Qiu, E. Amir, P. Krutzik, R. Finck, R. Bruggner, R. Melamed, A. Trejo, O. Ornatsky, et al. Single-cell mass cytometry of differential immune and drug responses across a human hematopoietic continuum. *Science*, 332(6030):687–696, 2011.

[12] S. C. Bendall, G. P. Nolan, M. Roederer, and P. K. Chattopadhyay. A deep profiler's guide to cytometry. *Trends in Immunology*, 33(7):323–332, July 2012.

[13] A. Biancotto, P. Dagur, J. Chris Fuchs, M. Langweiler, and J. Philip McCoy Jr. OMIP-004: In-depth characterization of human T regulatory cells. *Cytometry Part A*, 81(1):360–361, 2011.

[14] C. M. Bishop and M. Svenskn. Bayesian hierarchical mixtures of experts. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pages 57–64. Morgan Kaufmann Publishers Inc., 2002.

[15] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM, 1992.

[16] P. S. Bradley and O. L. Mangasarian. Feature selection via concave minimization and support vector machines. In *ICML*, volume 98, pages 82–90, 1998.

[17] W. Cao. Molecular characterization of human plasmacytoid dendritic cells. *Journal of Clinical Immunology*, 29(3):257–264, 2009.

[18] K. Castro, J. Ward, L. Slutsker, J. Buehler, H. Jaffe, R. Berkelman, and J. Curran. Revised classification system for HIV infection and expanded surveillance case definition for AIDS among adolescents and adults. *MMWR Recomm Rep*, 41:1–19, 1992.

[19] C. Chan, F. Feng, J. Ottinger, D. Foster, M. West, and T. Kepler. Statistical mixture modeling for cell subtype identification in flow cytometry. *Cytometry Part A*, 73(8):693–701, 2008.

[20] C. Chan, L. Lin, J. Frelinger, V. Hérbert, D. Gagnon, C. Landry, R. Sékaly, J. Enzor, J. Staats, K. Weinhold, et al. Optimization of a highly standardized carboxyfluorescein succinimidyl ester flow cytometry panel and gating strategy design using discriminative information measure evaluation. *Cytometry Part A*, 77(12):1126–1136, 2010.

[21] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[22] P. Chattopadhyay and M. Roederer. Cytometry: Today's technology and tomorrow's horizons. *Methods*, 57(3):251–258, Feb 2012.

[23] P. Chattopadhyay, M. Roederer, and D. Price. OMIP-002: Phenotypic analysis of specific human CD8+ T-cells using peptide-MHC class I multimers for any of four epitopes. *Cytometry Part A*, 77(9):821–822, 2010.

[24] P. K. Chattopadhyay, C. M. Hogerkorp, and M. Roederer. A chromatic explosion: the development and future of multiparameter flow cytometry. *Immunology*, 125(4):441, 2008.

[25] H.-Y. Chuang, E. Lee, Y.-T. Liu, D. Lee, and T. Ideker. Network-based classification of breast cancer metastasis. *Molecular systems biology*, 3:140, Jan. 2007.

[26] E. Costa, C. Pedreira, S. Barrena, Q. Lecrevisse, J. Flores, S. Quijano, J. Almeida, M. del Carmen García-Macias, S. Bottcher, J. Van Dongen, et al. Automated pattern-guided principal component analysis vs expert-based immunophenotypic classification of b-cell chronic lymphoproliferative disorders: a step forward in the standardization of clinical immunophenotyping. *Leukemia*, 24(11):1927–1933, 2010.

[27] D. R. Cox. The regression analysis of binary sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 215–242, 1958.

[28] F. Craig, R. Brinkman, E. Ten, and N. Aghaeepour. Computational analysis optimizes the flow cytometric evaluation for lymphoma. *Cytometry Part B - Clinical Cytometry*, Digital preprint, 2013.

[29] T. G. Dietterich. Ensemble learning. *The handbook of brain theory and neural networks*, pages 405–408, 2002.

[30] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.

[31] J. Dopazo. Functional interpretation of microarray experiments. *Omics: a journal of integrative biology*, 10(3):398–410, 2006.

[32] B. Efron and R. J. Tibshirani. *An introduction to the bootstrap*. CRC press, 1994.

[33] J. P. Egan. Signal detection theory and ROC analysis. *Academic Press, New York*, 1975.

[34] L. Ein-Dor, I. Kela, G. Getz, D. Givol, and E. Domany. Outcome signature genes in breast cancer: is there a unique set? *Bioinformatics*, 21(2):171–8, Jan. 2005.

[35] M. Eller and J. Currier. OMIP-007: Phenotypic analysis of human natural killer cells. *Cytometry Part A*, 81(6):447–449, 2012.

[36] D. Eppstein. Finding the k shortest paths. *SIAM Journal on computing*, 28(2):652–673, 1998.

[37] G. Finak, A. Bashashati, R. Brinkman, and R. Gottardo. Merging mixture components for cell population identification in flow cytometry. *Advances in Bioinformatics*, v09, 2009.

[38] G. Finak, J.-M. Perez, A. Weng, and R. Gottardo. Optimizing transformations for automated, high throughput analysis of flow cytometry data. *BMC bioinformatics*, 11(1):546, 2010.

[39] R. W. Floyd. Algorithm 97: shortest path. *Communications of the ACM*, 5(6):345, 1962.

[40] K. Foulds, M. Donaldson, and M. Roederer. OMIP-005: Quality and phenotype of antigen-responsive rhesus macaque T cells. *Cytometry Part A*, 81(6):360–361, 2012.

[41] G. N. Frederickson. An optimal algorithm for selection in a min-heap. *Information and Computation*, 104(2):197–214, 1993.

[42] M. L. Fredman and R. E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the ACM (JACM)*, 34(3):596–615, 1987.

[43] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. System Sci.*, 55(1):119–139, 1997.

[44] J. H. Friedman. Stochastic gradient boosting. *Comput. Stat. Data Anal.*, 38(4):367–378, 2002.

[45] M. J. Fulwyler. Electronic separation of biological cells by volume. *Science*, 150(3698):910–911, 1965.

[46] A. Ganesan, P. K. Chattopadhyay, T. M. Brodie, J. Qin, W. Gu, J. R. Mascola, N. L. Michael, D. A. Follmann, M. Roederer, C. Decker, T. Whitman, S. Tasker, A. Weintrob, G. Wortmann, M. Zapor, M. Landrum, V. Marconi, J. Okulicz, N. Crum-Cianflone, M. Bavaro, H. Chun, R. V. Barthel, A. Johnson, B. Agan, N. Aronson, W. Bradley, G. Gandits, L. Jagodzinski, R. O'Connell, C. Eggleston, and J. Powers. Immunologic and virologic events in early HIV infection predict subsequent rate of progression. *Journal of Infectious Diseases*, 201:272–284, Jan 2010.

[47] L. Gattinoni, E. Lugli, Y. Ji, Z. Pos, C. Paulos, M. Quigley, J. Almeida, E. Gostick, Z. Yu, C. Carpenito, et al. A human memory t cell subset with stem cell-like properties. *Nature Medicine*, pages 1290–1297, 2011.

[48] H. T. Gazda, M. Preti, M. R. Sheen, M.-F. O'Donohue, A. Vlachos, and S. M. Davies et al. Frameshift mutation in p53 regulator RPL26 is associated with multiple physical abnormalities and a specific pre-ribosomal RNA processing defect in diamond-blackfan anemia. *Human mutation*, 33(7):1037–44, July 2012.

[49] A. Goldhirsch, J. H. Glick, R. D. Gelber, A. S. Coates, and H.-J. Senn. Meeting highlights: international consensus panel on the treatment of primary breast cancer. *Journal of Clinical Oncology*, 19(18):3817–3827, 2001.

[50] S. Gordon, B. Cervasi, P. Odorizzi, R. Silverman, F. Aberra, G. Ginsberg, J. Estes, M. Paiardini, I. Frank, and G. Silvestri. Disruption of intestinal CD4+ T cell homeostasis is a key marker of systemic CD4+ T cell activation in HIV-infected individuals. *The Journal of Immunology*, 185(9):5169–5179, 2010.

[51] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1-3):389–422, 2002.

[52] F. Hahne, A. H. Khodabakhshi, A. Bashashati, C.-J. Wong, R. D. Gascoyne, A. P. Weng, V. Seyfert-Margolis, K. Bourcier, A. Asare, T. Lumley, et al. Per-channel basis normalization methods for flow cytometry data. *Cytometry Part A*, 77(2):121–131, 2010.

[53] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin. The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2):83–85, 2005.

[54] G. H. Heppner. Tumor heterogeneity. *Cancer research*, 44(6):2259–2265, 1984.

[55] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[56] R. Jansen, D. Greenbaum, and M. Gerstein. Relating whole-genome expression data with protein-protein interactions. *Genome research*, 12(1):37–46, 2002.

[57] H. Jaspan, L. Liebenberg, W. Hanekom, W. Burgers, D. Coetzee, A. Williamson, F. Little, L. Myer, R. Coombs, D. Sodora, et al. Immune activation in the female genital tract during hiv infection predicts mucosal cd4 depletion and hiv shedding. *Journal of Infectious Diseases*, 204(10):1550–1556, 2011.

[58] V. M. Jiménez and A. Marzal. A lazy version of eppsteins k shortest paths algorithm. In *Experimental and Efficient Algorithms*, pages 179–191. Springer, 2003.

[59] M. Kanehisa, S. Goto, Y. Sato, M. Kawashima, M. Furumichi, and M. Tanabe. Data, information, knowledge and principle: back to metabolism in KEGG. *Nucleic acids research*, 42(Database issue):D199–205, Jan. 2014.

[60] J.-H. Kim. Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. *Computational Statistics & Data Analysis*, 53(11):3735–3745, 2009.

[61] W. A. Knight, R. B. Livingston, E. J. Gregory, and W. L. McGuire. Estrogen receptor as an independent prognostic factor for early recurrence in breast cancer. *Cancer research*, 37(12):4669–4671, 1977.

[62] A. Krug, A. Towarowski, S. Britsch, S. Rothenfusser, V. Hornung, R. Bals, T. Giese, H. Engelmann, S. Endres, A. Krieg, et al. Toll-like receptor expression reveals CpG DNA as a unique microbial stimulus for plasmacytoid dendritic cells which synergizes with CD40 ligand to induce high amounts of IL-12. *European Journal of Immunology*, 31(10):3026–3037, 2001.

[63] H.-T. KUHN. Aw (1951) nonlinear programming. In *2nd Berkeley Symposium. Berkeley, University of California Press.*

[64] H.-M. Lai, A. A. Albrecht, and K. K. Steinhöfel. irda: a new filter towards predictive, stable, and enriched candidate genes. *BMC genomics*, 16(1):1, 2015.

[65] L. Lamoreaux, R. Koup, and M. Roederer. OMIP-009: Characterization of antigen-specific human T-cells. *Cytometry Part A*, 81(5):362–363, 2012.

[66] O. Lavi, G. Dror, and R. Shamir. Network-induced classification kernels for gene expression profile analysis. *J Comput Biol.*, 19(6):694–709, June 2012.

[67] H.-T. Lin, C.-J. Lin, and R. C. Weng. A note on Platt's probabilistic outputs for support vector machines. *Machine Learning*, 68(3):267–276, Aug. 2007.

[68] K. Lo, R. Brinkman, and R. Gottardo. Automated gating of flow cytometry data via robust model-based clustering. *Cytometry Part A*, 73(4):321–332, 2008.

[69] H. T. Maecker, J. P. McCoy, and R. Nussenblatt. Standardizing immunophenotyping for the Human Immunology Project. *Nature Reviews Immunology*, 12:191–200, 2012.

[70] Y. Mahnke and M. Roederer. OMIP-001: Quality and phenotype of Ag-responsive human T-cells. *Cytometry Part A*, 77(9):819–820, 2010.

[71] T. Marafioti, J. Paterson, E. Ballabio, K. Reichard, S. Tedoldi, K. Hollowood, M. Dictor, M. Hansmann, S. Pileri, M. Dyer, et al. Novel markers of normal and neoplastic human plasmacytoid dendritic cells. *Blood*, 111(7):3778–3792, 2008.

[72] J. Mercer. Functions of positive and negative type, and their connection with the theory of integral equations. *Philosophical transactions of the royal society of London. Series A, containing papers of a mathematical or physical character*, pages 415–446, 1909.

[73] S. Mukherjee. *The emperor of all maladies: a biography of cancer.* Simon and Schuster, 2011.

[74] D. Murdoch, J. Staats, and K. Weinhold. OMIP-006: Phenotypic subset analysis of human T regulatory cells via polychromatic flow cytometry. *Cytometry Part A*, 81(4):281–283, 2012.

[75] D. Nam and S.-Y. Kim. Gene-set approach for expression pattern analysis. *Briefings in bioinformatics*, 9(3):189–197, 2008.

[76] National Institutes of Health Consensus Development Panel and others. National institutes of health consensus development conference statement: adjuvant therapy for breast cancer, november 1—3, 2000. *J Natl Cancer Inst.*, 93(13):979–989, 2001.

[77] U. Naumann, G. Luta, and M. Wand. The curvHDR method for gating flow cytometry samples. *BMC Bioinformatics*, 11(1):11–44, 2010.

[78] O. Ornatsky, D. Bandura, V. Baranov, M. Nitz, M. Winnik, and S. Tanner. Highly multiparametric analysis by mass cytometry. *Journal of Immunological Methods*, 361(6030):1–20, 2010.

[79] K. ONeill, A. Jalali, N. Aghaeepour, H. Hoos, and R. R. Brinkman. Enhanced flowtype/rchyoptimyx: a bioconductor pipeline for discovery in high-dimensional cytometry data. *Bioinformatics*, 30(9):1329–1330, 2014.

[80] M. Y. Park and T. Hastie. L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(4):659–677, 2007.

[81] D. R. Parks, M. Roederer, and W. A. Moore. A new logicle display method avoids deceptive effects of logarithmic scaling for low signals and compensated data. *Cytometry Part A*, 69(6):541–551, 2006.

[82] K. Pearson. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, pages 240–242, 1895.

[83] S. Perfetto, P. Chattopadhyay, and M. Roederer. Seventeen-colour flow cytometry: unravelling the immune system. *Nature Reviews Immunology*, 4(8):648–655, 2004.

[84] S. Peri, J. D. Navarro, R. Amanchy, T. Z. Kristiansen, C. K. Jonnalagadda, V. Surendranath, V. Niranjan, B. Muthusamy, T. Gandhi, M. Gronborg, et al. Development of human protein reference database as an initial platform for approaching systems biology in humans. *Genome research*, 13(10):2363–2371, 2003.

[85] F. Preijers, E. Huys, and B. Moshaver. OMIP-010: A new 10-color monoclonal antibody panel for polychromatic immunophenotyping of small hematopoietic cell samples. *Cytometry Part A*, 81(6):453–455, 2012.

[86] S. Pyne, X. Hu, K. Wang, E. Rossin, T. Lin, L. Maier, C. Baecher-Allan, G. McLachlan, P. Tamayo, D. Hafler, et al. Automated high-dimensional flow cytometric data analysis. *Proceedings of the National Academy of Sciences*, 106(21):8519–8524, 2009.

[87] Y. Qian, Y. Liu, J. Campbell, E. Thomson, Y. M. Kong, and R. H. Scheuermann. Fcstrans: an open source software system for fcs file conversion and data transformation. *Cytometry Part A*, 81(5):353–356, 2012.

[88] Y. Qian, C. Wei, F. Eun-Hyung Lee, J. Campbell, J. Halliley, J. Lee, J. Cai, Y. Kong, E. Sadat, E. Thomson, et al. Elucidation of seventeen human peripheral blood B-cell subsets and quantification of the tetanus response using a density-based method for the automated identification of cell populations in multidimensional flow cytometry data. *Cytometry Part B: Clinical Cytometry*, 78(S1):S69–S82, 2010.

[89] P. Qiu, E. Simonds, S. Bendall, K. Gibbs Jr, R. Bruggner, M. Linderman, K. Sachs, G. Nolan, and S. Plevritis. Extracting a cellular hierarchy from high-dimensional cytometry data with spade. *Nature Biotechnology*, 29:886–891, 2011.

[90] C. E. Rasmussen. Gaussian processes for machine learning. 2006.

[91] D. N. Reshef, Y. A. Reshef, H. K. Finucane, S. R. Grossman, G. McVean, P. J. Turnbaugh, E. S. Lander, M. Mitzenmacher, and P. C. Sabeti. Detecting novel associations in large data sets. *Science*, 334(6062):1518–1524, 2011.

[92] M. Roederer. Compensation in flow cytometry. *Current Protocols in Cytometry*, pages 1–14, 2002.

[93] M. Roederer, J. Nozzi, and M. Nason. Spice: Exploration and analysis of post-cytometric complex multivariate datasets. *Cytometry Part A*, 79(2):167–174, 2011.

[94] M. Roederer and A. Tárnok. OMIPsOrchestrating multiplexity in polychromatic science. *Cytometry Part A*, 77(9):811–812, 2010.

[95] Y. Saeys, T. Abeel, and Y. Van de Peer. Robust feature selection using ensemble feature selection techniques. In *Machine learning and knowledge discovery in databases*, pages 313–325. Springer, 2008.

[96] R. E. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. *Machine learning*, 37(3):297–336, 1999.

[97] P. Schuster, N. Donhauser, K. Pritschet, M. Ries, S. Haupt, N. Kittan, K. Korn, and B. Schmidt. Co-ordinated regulation of plasmacytoid dendritic cell surface receptors upon stimulation with herpes simplex virus type 1. *Immunology*, 129(2):234–247, 2010.

[98] C. E. Shannon. The mathematical theory of communication. 1963. *MD computing: computers in medical practice*, 14(4):306–317, 1996.

[99] E. Shapiro, T. Biezuner, and S. Linnarsson. Single-cell sequencing-based technologies will revolutionize whole-organism science. *Nat Rev Genet.*, 14(9):618–30, Sept. 2013.

[100] H. M. Shapiro. *Practical flow cytometry*. John Wiley & Sons, 2005.

[101] J. Shi and M. G. Walker. Gene set enrichment analysis (gsea) for interpreting gene expression profiles. *Current Bioinformatics*, 2(2):133–137, 2007.

[102] A. Smola and V. Vapnik. Support vector regression machines. *Advances in neural information processing systems*, 9:155–161, 1997.

[103] A. J. Smola and B. Schölkopf. *Learning with kernels*. Citeseer, 1998.

[104] A. Subramanian, P. Tamayo, V. K. Mootha, S. Mukherjee, B. L. Ebert, M. A. Gillette, A. Paulovich, S. L. Pomeroy, T. R. Golub, E. S. Lander, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–15550, 2005.

[105] I. Sugár and S. Sealfon. Misty Mountain clustering: application to fast unsupervised flow cytometry gating. *BMC Bioinformatics*, 11(1):502–508, 2010.

[106] R. Suzuki and H. Shimodaira. Pvclust: an r package for assessing the uncertainty in hierarchical clustering. *Bioinformatics*, 22(12):1540–1542, 2006.

[107] M. Swiecki and M. Colonna. Unraveling the functions of plasmacytoid dendritic cells during viral infections, autoimmunity, and tolerance. *Immunological Reviews*, 234(1):142–162, 2010.

[108] The Cancer Genome Atlas Network. The Cancer Genome Atlas (TCGA). `https://tcga-data.nci.nih.gov/tcga/`, 2006.

[109] The Cancer Genome Atlas Network. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61–70, 2012.

[110] The Cancer Genome Atlas Network. Genomic and epigenomic landscapes of adult *de novo* acute myeloid leukemia. *N Engl J Med.*, 368(22):2059–2074, 2013.

[111] L. J. van 't Veer, H. Dai, M. J. Van De Vijver, Y. D. He, A. A. Hart, M. Mao, H. L. Peterse, K. van der Kooy, M. J. Marton, A. T. Witteveen, et al. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871):530–536, 2002.

[112] V. Vapnik. *The nature of statistical learning theory.* Springer Science & Business Media, 2013.

[113] V. Vapnik and A. Chervonenkis. A note on one class of perceptrons. *Automation and remote control*, 25(1), 1964.

[114] D. Venet, J. E. Dumont, and V. Detours. Most random gene expression signatures are significantly associated with breast cancer outcome. *PLoS computational biology*, 7(10):e1002240, Oct. 2011.

[115] J. P. Vial and F. Lacombe. Immunophenotyping of acute leukemia: Utility of CD45 for blast cell identification. *Methods in Cell Biology*, 64:343–358, 2001.

[116] F. Villanova, P. D. Meglio, M. Inokumad, N. Aghaeepour, E. Perucha, J. Mollon, L. Nomura, M. Hernandez-Fuentes, A. Copeh, T. Prevosti, S. Heck, V. Maino, G. Lord, R. R. Brinkman, , and F. O. Nestle. Integration of lyoplate based flow cytometry and computational analysis for standardized immunological biomarker discovery. *PLoS ONE*, 8(7), 2013.

[117] S. H. Walker and D. B. Duncan. Estimation of the probability of an event as a function of several independent variables. *Biometrika*, 54(1-2):167–179, 1967.

[118] C. Wei, J. Jung, and I. Sanz. OMIP-003: Phenotypic analysis of human memory B cells. *Cytometry Part A*, 79(11):894–896, 2011.

[119] A. Weintrob, A. Fieberg, B. Agan, A. Ganesan, N. Crum-Cianflone, V. Marconi, M. Roediger, S. Fraser, S. Wegner, and G. Wortmann. Increasing age at HIV seroconversion from 18 to 40 years is associated with favorable virologic and immunologic responses to HAART. *JAIDS Journal of Acquired Immune Deficiency Syndromes*, 49(1):40–47, 2008.

[120] A. K. White, M. VanInsberghe, O. I. Petriv, M. Hamidi, D. Sikorski, M. A. Marra, J. Piret, S. Aparicio, and C. L. Hansen. High-throughput microfluidic single-cell RT-qPCR. *Proceedings of the National Academy of Sciences*, 108(34), Aug. 2011.

[121] S. Xue and M. Barna. Specialized ribosomes: a new frontier in gene regulation and organismal biology. *Nat Rev Mol Cell Biol.*, 13(6):355–69, June 2012.

[122] J. Y. Yen. Finding the k shortest loopless paths in a network. *management Science*, 17(11):712–716, 1971.

[123] H. Zare, A. Bashashati, R. Kridel, N. Aghaeepour, G. Haffari, J. Connors, R. Gascoyne, A. Gupta, R. Brinkman, and A. Weng. Automated analysis of multidimensional flow cytometry data improves diagnostic accuracy between mantle cell lymphoma and small lymphocytic lymphoma. *American Journal of Clinical Pathology*, 137(1):75–85, 2012.

[124] H. Zare, P. Shooshtari, A. Gupta, and R. Brinkman. Data reduction for spectral clustering to analyze high throughput flow cytometry data. *BMC Bioinformatics*, 11(1):403–413, 2010.

[125] J. Zhu, H. Zou, S. Rosset, and T. Hastie. Multi-class AdaBoost. *Statistics and its Interface*, 2(3), 2009.

[126] C. Zuleger and M. Albertini. OMIP-008: Measurement of Th1 and Th2 cytokine polyfunctionality of human T cells. *Cytometry Part A*, 81(6):450–452, 2012.