Using Correlation-Based Feature Selection for a Diverse Collection of Bioinformatics Datasets

Randall Wald, Taghi M. Khoshgoftaar, and Amri Napolitano
Florida Atlantic University
Email: {rwald1, khoshgof}@fau.edu, amrifau@gmail.com

***Abstract*—The large number of genes found in most gene microarray datasets demands the use of feature selection techniques to alleviate this problem of high-dimensionality. However, the computational cost of filter-based subset evaluation techniques such as Correlation-Based Feature Selection (CFS) has generally limited the use of these techniques to smaller datasets, or at least smaller collections of gene microarray datasets. No previous work has applied CFS to a large and diverse range of bioinformatics datasets. To address this deficit, we employ nine different microarray datasets exhibiting a wide range of characteristics in terms of dataset balance (fraction of instances found in the minority class) and dataset difficulty of learning (overall difficulty of building effective classification models on raw, pre-feature-selection datasets). We also use five classification learners to discover how these perform in conjunction with CFS, along with five performance metrics to give a broad perspective on our results. The results find that CFS can be used to help build effective models, in particular when used with the 5-Nearest Neighbors learner on data that is Easy or Moderate (in terms of difficulty-of-learning) or Balanced (in terms of class distribution). For other types of data, the optimal learner varies, although in most cases the Logistic Regression learner works worst in conjunction with CFS.**

***Keywords*-Correlation-Based Feature Selection; Bioinformatics; Balance; Difficulty of Learning**

I. Introduction

Gene microarrays (gene chips) are an important technology in the domain of bioinformatics. These gene chips allow researchers to quickly identify the expression levels of thousands of genes within a given tissue sample, rather than performing analysis on each gene separately. With this technology, researchers can find paired populations (such as individuals with or without a given form of cancer, or patients who responded or didn't respond to a particular cancer treatment) and begin to understand the underlying genetic factors which distinguish these populations (classes). However, the very scale of data available from these experiments can hinder analysis, as it is difficult to directly examine the levels of thousands of genes. Thus, feature selection techniques borrowed from the domain of data mining and machine learning become necessary.

Feature selection is the general challenge of reducing the number of features (independent attributes) in a dataset. As high-dimensionality (the problem of having too many features per instance) is found across many application domains, not just bioinformatics, it has been studied at length, and three categories of feature selection have been developed. Filter-based feature ranking techniques apply a statistical filter to each feature individually, assigning scores to the features. These scores are then used to rank the features, and practitioners can select those above a given threshold. Filter-based subset evaluation, on the other hand, applies filters to complete gene subsets, evaluating how closely these genes are related not just to the class, but also to one another. This strategy of considering whole subsets is also used by wrapper-based subset evaluation, but the wrapper-based method does not use statistical filters to perform its calculations, instead relying on classification models to directly determine which feature subsets can be used to build the most effective model.

Each of these techniques has its advantages and disadvantages. Filter-based ranking is much more computationally efficient than the other two, but this comes at the price of not recognizing redundant features (those which contain information already present in the feature subset); it is only capable of discarding irrelevant features (those which lack any connection to the class whatsoever). Both of the subset-based techniques have the opposite problems (able to detect both irrelevant and redundant features, at a computational cost), and between these two, wrapper-based approaches are even more time intensive but hold the potential to find the features most important to the specific problem of building classification models [20]. Nonetheless, in application domains where finding the relevant features may be the end goal unto itself (such as gene microarray studies which seek to find genes relevant to cancer detection and patient response prediction), finding an effective and non-redundant subset with filter-based subset selection bears further study.

One filter-based subset selection technique in particular, Correlation-Based Feature Selection (CFS) [10], has been popular in other domains but little-used in bioinformatics. Moreover, those studies which have considered this technique have used small collections of datasets, and have not examined two other problems which can affect a wide range of gene microarray datasets: class imbalance and difficulty of learning. Class imbalance occurs when the fraction of instances in each class are not equal; for example, in a binary dataset (with only two classes), this occurs if the minority dataset has far fewer instances than the majority class. As some studies can consider rare cancers or drugs which only a small fraction of the population responds to, it is important to understand which models perform best in these trying circumstances. In addition, some datasets are simply very challenging to build models from, even when comparing raw models without enhancements

978-1-4799-7502-0/14 $31.00 © 2014 IEEE
DOI 10.1109/BIBE.2014.63

such as feature selection. These difficult datasets may be noisy or may simply have a very weak class boundary, but some problem domains are prone to such datasets, and again it is important to consider which approaches and techniques are most effective here. Therefore, we designed this work to showcase the effectiveness of CFS in an appropriate manner.

To better understand both of these concerns in the context of feature selection, we employ CFS along with nine datasets which exhibit a wide range of difficulty and balance. The present work is the first such study to use CFS on such a large and diverse collection of datasets. Our experiments also include five classification learners (5-Nearest Neighbor, Logistic Regression, Multi-Layer Perceptron, Naïve Bayes, and Support Vector Machines), along with five performance metrics (Area Under the Receiver Operating Characteristic (ROC) Curve, Area Under the Precision-Recall Curve, Arithmetic Mean of True Positive Rate and True Negative Rate, Geometric Mean of True Positive Rate and True Negative Rate, and Overall Accuracy) to study the role of class imbalance on performance measurement. Our results show that overall, and on the Easy, Moderate, and Balanced datasets, the best learner is generally 5-Nearest Neighbor. However, with the remaining categories of dataset (Hard, Slightly Imbalanced, and Imbalanced), the results are less clear, with all the learners other than Logistic Regression (which is almost always the worst technique) giving the best results for some choice of dataset group and performance metric. These variations show that studying the use of CFS on a variety of datasets gives important insight into how the nature of the data can affect the performance of classification on that data.

The remainder of this paper will be organized as follows: Section II will review related work on the use of CFS in the context of bioinformatics. Section III will detail the CFS method itself along with the learner and performance metrics used in this paper. Section IV will discuss the datasets used in this paper, with explanation of how they vary in terms of balance and difficulty of learning. In Section V, we will present our results and discussion of our results. Finally, in Section VI we will conclude and present our suggestions for future work.

II. Related Work

Although feature selection is widely recognized as being an essential element of bioinformatics (and in particular gene microarray analysis) [8], the majority of work has focused on filter-based feature ranking [13]. However, some surveys, such as Sayes et al. [20], do specifically address all types of feature selection, including filter-based feature subset evaluation. One particularly-popular form of filter-based subset evaluation is Correlation-Based Feature Selection (CFS), proposed by Mark Hall [10]. This approach is designed to judge the quality of a feature subset by balancing the individual correlations between each constituent feature and the class against the overall correlation among the features of the subset. CFS has become a popular choice for filter-based subset selection across a number of application domains [9].

Despite the complexity of CFS and the very large number of features typically found in the bioinformatics application domain, a handful of papers have applied this technique to gene microarray datasets. Yeoh et al. [29] examined the gene expression profiles of pediatric acute lymphoblastic leukemia patients to understand the different subtypes of leukemia exhibited by the patients, as well as to learn whether these expression profiles could be predictive of relapse following treatment. To build their models, CFS was used so long as it chose 20 or fewer features, with this limit imposed to reduce the computational complexity (in the event CFS wanted to find more features, it was halted and a feature ranker was used instead). These models proved effective at predicting both cancer subtype and patient response. Liu et al. [16] consider two datasets, the first being the same as Yeoh et al.'s work (although only the cancer subtypes were considered by Liu et al.), and the other dataset containing proteomic data from ovarian cancer patients. Their goal was to compare six feature selection approaches, including CFS. For the leukemia data (which was again a multi-class subtype prediction problem), CFS proved to help build the best classification models, and it was also the best for selecting the most important protein markers. Yu and Liu proposed a new algorithm for feature selection [30], [31] based on ranking the features and then (starting from the highest-ranked feature) eliminating lower-ranked features if they are more correlated with other features than with the class value, and in both papers they include CFS both among their inspirations and among those techniques they compare with their proposed approach. In their first paper [31], the techniques are compared using four benchmark cancer-identification gene expression datasets, while their second paper [30] considers both synthetic and benchmark data (using 10 datasets for the latter, although fewer than half are from the domain of bioinformatics). Finally, Wang et al. [27] contrast filter-based feature ranking, CFS, and wrapper-based subset evaluation on a different acute leukemia dataset (not the one used by Yeoh et al. and Liu et al.) as well as a second dataset representing different subtypes of diffuse large B-cell lymphoma. They note that one gene is consistently selected by all methods for the first dataset, and that this gene shows very significant differences in expression level between their two types of acute leukemia: however, less consensus was found on the second dataset. Unlike these prior works, the present work exclusively focuses on high-dimensional bioinformatics datasets, and considers a much larger and more diverse collection with such datasets (considering nine datasets which vary both in balance level and pre-feature-selection difficulty of learning).

Class imbalance is a prevalent problem within bioinformatics datasets [3]. The reason is that the class of interest usually has very few samples while the other class is much larger, with ratios of 1:100 or even 1:1000. This issue has been studied by Van Hulse et al. [23], who compared the correlations among nine rankers on five imbalanced datasets and a number of data sampling approaches (algorithms to improve the balance of datasets). Another example is Ramaswamy et al. [19] who

performed a study on a dataset whose class of interest made up only 16% of the total dataset. Abu Shanab et al. [1] examined the effects of noisy imbalanced bioinformatics datasets which have been resampled on the stability of feature rankers. The presence of class imbalance has the potential to affect the classification performance of classifiers applied toward the imbalanced datasets.

Perhaps the root of the trouble with class imbalance is in how the classification algorithms are designed. A majority of classification algorithms assume that the classes involved will have an equal presence in the dataset [12]. Lusa et al. [17] found that there is a bias towards the majority class when working with imbalanced data. They also stated that unless we take steps to combat class imbalance, the performance of the analysis will suffer. This phenomenon may be due to two different factors: feature selection with certain classifiers focuses on accuracy and so will focus on the majority class, and due to the small number of instances the outliers are given more weight and can distort the data [2]. Recommendations for combating some of these issues include applying data sampling methods [24] which is a common method for dealing with class imbalance, either adding instances to the minority class (oversampling) or removing instances from the majority class (undersampling).

On the other hand, relatively little work has considered the problem of difficulty of learning: that is, how some datasets contain noise and imprecise class boundaries and how these can have effects beyond simply reducing classification performance. Three works recently published by our research group consider this problem in more detail, however. We considered the problem of using gene microarray datasets to predict whether patients will show therapeutic response to different cancer treatments, and discussed how this problem is inherently very challenging (as all patients will be suffering from the same cancer, and they will only vary from one another insofar as some respond to treatment while others do not) [25]. We discussed how when attempting to compare an unrelated factor (such as degree of class imbalance), it is important to observe whether each level of the factor being considered has an equivalent level of difficulty of learning, and how this has the potential to give misleading results if different levels of difficulty of learning are mistaken to be different levels of the factor of interest [6]. And we also explored how difficulty of learning, a concept inherently tied to the classification performance of various datasets, can also impair the stability of feature selection techniques without regard to their eventual classification performance, demonstrating that this concept is broader than classification alone [26]. Overall, for these reasons we felt it was important to consider our work in this paper in the context of difficulty of learning.

III. Methods

A total of five learners were used along with the correlation-based feature selection (CFS) technique in this case study. CFS itself is presented in Section III-A, while the learners are presented individually in Section III-B, and our performance metrics and evaluation procedure are discussed in Section III-C.

A. Correlation-Based Feature Selection

For our experiments, Correlation-Based Feature Selection (CFS) [10] was used to choose the best feature subsets for each dataset. This technique employs the Pearson correlation coefficient [14], a correlation metric designed to balance the need to have the features correlate with the class and the need to have the features not correlate with one another. The Pearson correlation coefficient is found with the following formula:

$$M_S = \frac{k\overline{r_{cf}}}{\sqrt{k + k(k-1)\overline{r_{ff}}}}$$

In this formula, M_S is the merit of the current set of features, k is the number of features, $\overline{r_{cf}}$ is the mean of the correlations between each feature and the class, and $\overline{r_{ff}}$ is the mean of the pairwise correlations between every two features. In both cases, correlations are calculated using symmetric uncertainty [28], an information-theoretic measure of how changes in one feature affect the uncertainty of the other, and which compensates for inherent entropy in either feature. As desired, the numerator increases when the set of features is particularly good at classifying the data, while the denominator increases when the set has a great deal of self-correlation, which implies redundancy. Therefore, the larger the M_S the better the subset.

In order to use this metric for feature selection, we need an approach to search through the realm of possible feature subsets in order to find the one which scores the highest. As the exhaustive approach is computationally infeasible (requiring that $O(2^n)$ subsets be evaluated for n features), we chose a more efficient strategy: Greedy Stepwise. This starts with an empty "working" feature set and progressively adds features, one at a time, until a stopping criterion is reached. At each step, we consider all feature subsets which include the current "working" feature subset and exactly one feature not presently in that set. We find the quality of each of these subsets (using the CFS metric), and then choose which of these gives the best performance to be the new "working" subset. This process is iterated until none of the new subsets improve performance, or until the "working" subset has at least 100 features (our stopping criterion). The final "working" subset is then given as the procedure's output. We chose 100 features as that is a appropriate feature subset size [7].

B. Learners

Five learners were chosen for our analysis: 5-Nearest Neighbor (5-NN), Logistic Regression (LR), Multi-Layer Perceptron (MLP), Naïve Bayes (NB), and Support Vector Machines (SVM). These were all chosen to give a wide range of classification models which do not include embedded feature selection. All models were built using the WEKA machine learning toolkit [11], using default parameters unless otherwise specified. Note that any changes to default parameter values were applied when experimentation showed an overall

Dataset Difficulty	Name	# Minority Instances	Total # of Instances	% Minority Instances	# of Attributes	Average AUC
Easy	**BCancer 50k** [5]	200	400	50.00%	54614	0.85636
	Lung [22]	64	203	31.53%	12601	0.86851
	ALL [22]	79	327	24.16%	12559	0.84748
Moderate	**Prostate** [22]	59	136	43.38%	12601	0.78225
	Colon [23]	22	62	35.48%	2001	0.79413
	Brain Tumor [22]	23	90	25.56%	27680	0.72096
Hard	**DLBCL NIH** [22]	102	240	42.50%	7400	0.58527
	Chanrion 2008 [4]	52	155	33.55%	22657	0.67207
	Pawitan 2005 [18]	40	159	25.16%	12066	0.61082

TABLE I: Details of the Datasets

improvement of the classification performance [24]. Due to space limitations, we only give a brief outline of these techniques; for further information, we direct readers to Witten and Frank [28].

5-NN is a lazy instance-based learner which does not build a model per se but which uses the training data directly to make predictions about the test data. In particular, to classify a given instance, it finds the five nearest neighbors from the training set for that instance, and then has these vote (using weight by 1/distance) on the proper class value. LR is a simple regression model which uses the logistic function to normalize the probability between 0 and 1. MLP is a multi-layer perceptron-based learner, with a single hidden layer contain 3 nodes and 10% of the data held back for validation of when to stop the backpropogation-based learning process. NB is a Bayesian learner which uses Bayes's Theorem to find the posterior probability of the class values given the observed feature values. Although NB makes the naïve assumption that all feature values are statistically independent, it has been show to give good performance even when this assumption is not true [15]. SVM is a Support Vector Machine designed to find the maximal margin hyperplane separating the classes, with a complexity constant of 5.0 and the `buildLogisticModels` parameter set to `true`. The default linear kernel was used.

C. Performance Measurement and Cross-Validation

The presence of imbalanced data highlights the importance of choosing an appropriate performance metric, in order to ensure that minority-class instances (also known as positive instances) do not all end up misclassified [21]. In addition, the choice of metric can affect the outcome, and it is not always clear which result is most meaningful. For these reasons, we use five different performance metrics: Area Under the Receiver Operating Characteristic Curve (AUC), Area Under the Precision-Recall Curve (PRC), Best Arithmetic Mean (AMean), Best Geometric Mean (GMean), and Default Accuracy (Acc). AUC builds a graph of the True Positive Rate vs. True Negative Rate as the classifier decision threshold is varied, and then uses the area under this graph as the performance across all decision thresholds. PRC is similar, only the curve graphs Precision vs. Recall. Both GMean and AMean consider the mean value of True Positive Rate and True Negative Rate, considering different threshold values and choosing the value which optimizes the metric; they differ in whether they find the arithmetic or geometric means, respectively. Finally, Acc is simply the fraction of instances which are classified correctly, regardless of their class membership; although this metric is generally inappropriate for use on imbalanced data, we present it for completeness.

Cross-validation was used for building and testing the classification models. This process begins by dividing the data into N equal-size subsets (folds), and then models are built (trained) on $N-1$ of these and tested on the Nth fold, called the hold-out fold. This process is repeated N times, so that each fold is used as the hold-out fold exactly once. For building our models, we let $N = 5$, giving five-fold cross validation. In order to avoid the risk of a bad split, the entire cross-validation process was repeated a total of four times. The five performance metrics were calculated by collecting the results across all test folds and creating a single value for each run of cross-validation; thus, for each dataset we have four values for each combination of learner and performance metric. These values were then averaged together (across the appropriate datasets) for our results tables. Note that the feature selection process is performed on all twenty datasets generated by each instance of the four runs of five-fold cross-validation.

IV. Case Study

In this case study, we consider nine datasets across the domains of bioinformatics and patient response prediction. A summary of these datasets is found in Table I. All datasets are gene microarray datasets. That is, the features represent the expression levels of various genes, as measured by how strongly different gene probes react to the mRNA produced from a tissue sample, and the class values (all of which are binary) come from whether a patient has cancer, what type of cancer a patient has, or whether the patient responded well to a particular cancer treatment. In particular, the Chanrion 2008 and Pawitan 2005 datasets come from the domain of patient response prediction, while all other datasets pertain to cancer detection or identification. Due to space limitations, we cannot elaborate further on the details of each individual dataset; for further information, please refer to the citations found within the table.

The last column, Average AUC, refers to the classification performance on these datasets when building models without

feature selection. This is used to show that some of these datasets are notable for being difficult to model (such that models do not perform well), while others are particularly easy. One of the goals of our experiments was to determine how different dataset characteristics affect the optimal choices for CFS feature selection, and this "difficulty of learning" value (also known as "dataset difficulty") was chosen as an important characteristic to study [26]. The values in the table were calculated using a set of six different classification models: 5-NN, MLP, NB, SVM, and two versions of a C4.5 decision tree (C4.5 D and C4.5 N). Descriptions of the learners other than C4.5-D and C4.5-N are found in Section III-B. C4.5 D is the C4.5 decision tree classifier with the default parameter values. C4.5 N is the same classifier but with Laplace smoothing enabled and pruning disabled. All of these learners are available using the WEKA Data Mining toolkit [11], and all default values were used unless otherwise specified. Note that the results from these classifiers (without feature selection) were used only to determine the difficulty of the datasets and have no further bearing on the rest of the experiment.

As discussed, one major goal of our experiments was to discover the influence of dataset characteristics on classification performance in the context of filter-based feature selection. In particular, we chose to focus on two different characteristics: dataset difficulty and balance level. Dataset difficulty was defined based on the average AUC performance (as outlined above), and balance level was specified by considering the percentage of instances found in the minority class (as all of our datasets are binary, there is only one minority class). To facilitate our experiments, we created three levels of each of these factors. For difficulty of learning, the datasets were divided into Easy (Average AUC $\geq$ 0.8), Moderate (Average AUC $<$ 0.8 and $\geq$ 0.7), and Hard (Average AUC $<$ 0.7). For balance level, we divided the datasets into Balanced (% Minority $>$ 40%), Slightly Imbalanced (% Minority $\leq$ 40% and $\geq$ 26%), and Imbalanced (% Minority $<$ 26%). In Table I, we separated the Easy, Moderate, and Hard datasets into different physical groupings, and sort by balance level within each grouping; in addition, all Balanced datasets have their names printed in **bold**, and all Imbalanced datasets have their names printed in *italics*. Note that there is exactly one dataset for each combination of balance level and difficulty of learning.

V. RESULTS

Our results for building classification models following feature selection are presented in Tables II through IV. Each table presents the classification results in terms of the five performance metrics (as specified by the column) when using the learner given by the row. All datasets first had CFS applied to reduce their features, as discussed in III-A. Table II includes the results averaged across all nine datasets, while Tables III and IV break the datasets into three groups based on the levels of dataset difficulty and balance, respectively. For each group of datasets (the whole table for Table II, or each of the three blocks within Tables III and IV), the best and worst learner for each performance metric are printed in **bold** and *italics*, respectively.

Learner	Performance Metric				
	AUC	PRC	AMean	GMean	Acc
5-NN	0.84814	**0.80492**	**83.48115**	**82.99964**	**0.83088**
LR	*0.79405*	*0.72390*	*77.63286*	*77.24868*	*0.77465*
MLP	0.84924	0.80183	82.99260	82.76716	0.82916
NB	0.83981	0.76556	82.03809	81.81581	0.80161
SVM	**0.85192**	0.79097	83.05434	82.62981	0.81472

TABLE II: Results Across All Datasets

Difficulty Level	Learner	Performance Metric				
		AUC	PRC	AMean	GMean	Acc
Easy	5-NN	**0.98063**	0.96839	**96.97758**	**96.96472**	**0.96245**
	LR	*0.96583*	0.95016	*94.11773*	*94.09769*	*0.93474*
	MLP	0.98008	**0.97226**	96.54759	96.53708	0.96083
	NB	0.97575	*0.94098*	96.48057	96.47755	0.95911
	SVM	0.97388	0.94424	95.75900	95.75078	0.93709
Moderate	5-NN	0.87070	**0.83280**	**86.76235**	**86.51248**	**0.84761**
	LR	*0.79066*	*0.66780*	*76.26144*	*75.92181*	*0.75130*
	MLP	0.87619	0.81727	85.17662	85.01985	0.84711
	NB	0.85555	0.75677	81.69778	81.26672	0.76406
	SVM	**0.88422**	0.81892	85.50385	85.38954	0.84532
Hard	5-NN	0.69309	0.61358	66.70353	65.52171	**0.68258**
	LR	*0.62565*	*0.55375*	*62.51941*	*61.72653*	*0.63791*
	MLP	0.69146	**0.61595**	67.25360	66.74453	0.67953
	NB	0.68814	0.59895	**67.93592**	**67.70315**	0.68166
	SVM	**0.69765**	0.60976	67.90017	66.74911	0.66174

TABLE III: Results by Difficulty Level

In Table II, we see the results across all nine datasets. From this, it is apparent that 5-NN gives the best classification performance for most of the performance metrics, but SVM outperforms it when considering the AUC metric. This demonstrates that the choice of metric can have an impact on which model appears to perform best. On the other hand, LR performs worst across all five performance metrics by a sizable margin, suggesting that particularly bad models will always give bad results.

Table III contains the results broken down by difficulty level. Here, we see a wider range of results, with all learners other than LR being the best choice for some combination of difficulty level and performance metric. For the Easy and Moderate datasets, we find that like the overall results, 5-NN is usually the best performer, although only the Moderate datasets match the overall results for all performance metrics. For the Easy datasets, 5-NN is the best performer in terms of AUC, but MLP is best in terms of PRC. The Hard datasets show a great variety of results, with SVM and MLP being best in terms of AUC and PRC respectively (somewhat similar to the Moderate and Easy results, respectively), and NB being best in terms of AMean and GMean. Only the Acc metric shows 5-NN to be the best learner. This change suggests that while 5-NN may work well on the less-challenging datasets, for the most difficult problems it is not an appropriate learner. The range of results found within the Hard datasets also serves to underscore that with especially challenging datasets, minor changes such as using a different performance metric can have a larger impact on the relative performance of different learners.

Balance Level	Learner	Performance Metric				
		AUC	PRC	AMean	GMean	Acc
Balanced	5-NN	**0.82676**	**0.79739**	**81.27990**	**81.09988**	0.78366
	LR	*0.76873*	*0.72454*	*74.83803*	*74.43552*	0.73897
	MLP	0.81313	0.78115	80.01717	79.94562	0.78241
	NB	0.78380	0.74007	76.74224	76.35319	*0.70596*
	SVM	0.81571	0.78100	79.85680	79.30830	**0.78647**
Slightly Imbalanced	5-NN	0.88350	**0.88957**	86.07606	85.81697	**0.85685**
	LR	*0.80186*	*0.78334*	*77.98195*	*77.62444*	*0.77370*
	MLP	0.88001	0.87674	85.30850	85.06847	0.84811
	NB	0.88381	0.85767	85.31085	85.17344	0.84204
	SVM	**0.89231**	0.87407	**86.23430**	**86.07825**	0.83826
Imbalanced	5-NN	0.83416	0.72781	83.08750	82.08206	0.85212
	LR	*0.81154*	*0.66383*	*80.07859*	*79.68606*	*0.81127*
	MLP	**0.85459**	**0.74759**	83.65213	83.28737	**0.85695**
	NB	0.85183	0.69895	**84.06118**	**83.92079**	0.85684
	SVM	0.84773	0.71785	83.07192	82.50287	0.81941

TABLE IV: Results by Balance Level

As for the worst learners, we find that much as with the overall results, LR is almost always the worst choice regardless of the difficulty level or the performance metric; only for one combination (Easy datasets with the PRC metric) is it only second-worst.

The results broken down by balance level, as presented in Table IV, show a greater amount of diversity even than the results broken down by difficulty. This may be because Moderate datasets are still easy enough to learn that they give consistent results across different performance metrics, while Slightly Imbalanced datasets are too challenging to provide consistent results. In any event, only one balance level (the Balanced datasets) is consistent with the overall results in favoring the 5-NN learner; even so, there is a notable difference insofar as the odd performance metric out is Acc (rather than AUC or PRC), with that metric favoring the SVM learner. 5-NN and SVM also compete for the top spots in the Slightly Imbalanced datasets, with SVM winning for the AUC, AMean, and GMean metrics, and 5-NN giving the best performance only in terms of PRC and Acc. SVM's performance for the two Mean-based metrics is especially notable, because in the Hard datasets, those both gave the best performance with NB. This trait is repeated for the Imbalanced datasets, where NB is best for the AMean and GMean metrics and MLP gives the best performance for the other three. Overall, these results show that except when working with the most balanced datasets, the optimal choice of learner will depend on the choice of performance metric.

As with the earlier results, LR was almost always the worst performing learner, only being the second-worst in one case (Balanced datasets with Acc metric). It would appear that this learner gives extremely poor results not just averaged across all the datasets, but when either of the factors (difficulty or balance level) are held constant. Thus, for this data we would recommend almost any learner other than LR.

Finally, it is useful to observe how the different metrics perform relative to one another, as the levels of balance and difficulty are changed. For example, when looking at the Easy datasets, all five metrics give relatively similar results. That is, keeping in mind that the AMean and GMean measures are 100 times larger than the others due to the way the metrics are calculated, we find that values for all five metrics are generally within 0.03 of one another. However, for the Moderate and Hard datasets, the PRC metric is noticably lower than the others (often around 0.08). While these metrics measure different things, and thus it is not expected that they will give the same values, it is useful to observe that PRC is more affected by dataset difficulty than the other metrics. PRC also has related behavior for the different balance levels, showing values similar to the other metrics (within 0.04) when considering the Balanced and Slightly Imbalanced datasets, but dropping off a great deal (often more than 0.09) for the Imbalanced datasets. This shows that when working with imbalanced data, the PRC metric will generally give lower values, which should be considered when trying to evaluate how different techniques affect classification performance.

VI. Conclusion

Many bioinformatics datasets (and in particular, almost all gene microarray datasets) have a very large number of features, which requires the use of feature selection. However, little work has considered filter-based subset selection techniques such as CFS, which are able to discover small and non-redundant feature subsets. Those works which do apply CFS to bioinformatics datasets use a small collection of datasets, and do not consider important dataset properties such as class imbalance and difficulty of learning. To address this, we considered the use of CFS to alleviate the problem of high dimensionality across nine different gene microarray datasets, which contain three levels of class imbalance and three levels of difficulty of learning. We used five different learners to build our classification models, and evaluated these models using five different performance metrics. Thus, this work was designed to appropriately showcase the effectiveness of models built using CFS for feature selection.

We find that overall, and on the Easy, Moderate, and Balanced datasets, the best learner is generally 5-NN, although with the remaining three categories (Hard, Slightly Imbalanced, and Imbalanced), there is less consensus: SVM often does well in terms of AUC, while NB is best in terms of AMean and GMean, and MLP performs well at times. LR, on the other hand, is almost always the worst learner, regardless of the type of dataset considered. Finally, we find that the PRC metric shows abnormal behavior compared to the other five metrics, with its results dropping off between the Easy and Moderate datasets, and between the Slightly Imbalanced and Imbalanced datasets, to a much greater degree than the other performance metrics.

Future work can consider an even larger collection of datasets as well as potentially incorporate additional feature selection approaches such as filter-based feature ranking and wrapper-based subset evaluation.

References

[1] A. Abu Shanab, T. M. Khoshgoftaar, R. Wald, and A. Napolitano, "Impact of noise and data sampling on stability of feature ranking

techniques for biological datasets," in *IEEE 13th International Conference on Information Reuse and Integration (IRI)*, C. Zhang, J. Joshi, E. Bertino, and B. M. Thuraisingham, Eds. IEEE, August 2012, pp. 415–422.

[2] A. Al-Shahib, R. Breitling, and D. Gilbert, "Feature selection and the class imbalance problem in predicting protein function from sequence," *Applied Bioinformatics*, vol. 4, no. 3, pp. 195–203, 2005.

[3] R. Batuwita and V. Palade, "A new performance measure for class imbalance learning. application to bioinformatics problems," in *International Conference on Machine Learning and Applications*, Dec. 2009, pp. 545–550.

[4] M. Chanrion, V. Negre, H. Fontaine, N. Salvetat, F. Bibeau, G. M. Grogan, L. Mauriac, D. Katsaros, F. Molina, C. Theillet, and J.-M. Darbon, "A gene expression signature that can predict the recurrence of tamoxifen-treated primary breast cancer," *Clinical Cancer Research*, vol. 14, no. 6, pp. 1744–1752, 2008. [Online]. Available: http://clincancerres.aacrjournals.org/content/14/6/1744.abstract

[5] D. J. Dittman, T. M. Khoshgoftaar, R. Wald, and J. Van Hulse, "Comparative analysis of DNA microarray data through the use of feature selection techniques," in *Ninth IEEE International Conference on Machine Learning and Applications (ICMLA)*, December 2010, pp. 147–152.

[6] D. J. Dittman, T. Khoshgoftaar, R. Wald, and A. Napolitano, "Gene selection stability's dependence on dataset difficulty," in *IEEE 14th International Conference on Information Reuse and Integration (IRI 2013)*, 2013, pp. 341–348.

[7] D. J. Dittman, T. M. Khoshgoftaar, R. Wald, and A. Napolitano, "Simplifying the utilization of machine learning techniques for bioinformatics," in *Machine Learning and Applications (ICMLA), 2013 12th International Conference on*, 2013, pp. 396–403.

[8] X. Feng, C. Feng, R. Guan, T. Wang, and G. Feng, "Survey of the feature selection methods for microarray analysis in bioinformatics," *Journal of Bioinformatics and Intelligent Control*, vol. 1, no. 1, pp. 40–46, 2012-06-01T00:00:00. [Online]. Available: http://www.ingentaconnect.com/content/asp/jbic/2012/00000001/00000001/art00005

[9] I. Guyon, *Feature Extraction: Foundations and Applications*, ser. Studies in Fuzziness and Soft Computing. Springer, 2006. [Online]. Available: http://books.google.com/books?id=x5hdbK8bIG0C

[10] M. Hall, "Correlation-based feature selection for machine learning," Ph.D. dissertation, The University of Waikato, Hamilton, New Zealand, April 1997.

[11] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: An update," *SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[12] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[13] C. Lazar, J. Taminau, S. Meganck, D. Steenhoff, A. Coletta, C. Molter, V. de Schaetzen, R. Duque, H. Bersini, and A. Nowe, "A survey on filter techniques for feature selection in gene expression microarray analysis," *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, vol. 9, no. 4, pp. 1106–1119, Jul. 2012. [Online]. Available: http://dx.doi.org/10.1109/TCBB.2012.33

[14] J. Lee Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/00031305.1988.10475524

[15] D. Lewis, "Naive (bayes) at forty: The independence assumption in information retrieval," in *Machine Learning: ECML-98*, ser. Lecture Notes in Computer Science, C. Nédellec and C. Rouveirol, Eds. Springer Berlin / Heidelberg, 1998, vol. 1398, pp. 4–15. [Online]. Available: http://dx.doi.org/10.1007/BFb0026666

[16] H. Liu, J. Li, and L. Wong, "A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns," *Genome Informatics*, vol. 13, pp. 51–60, 2002.

[17] L. Lusa and R. Blagus, "The class-imbalance problem for high-dimensional class prediction," in *Proceedings of the Eleventh International Conference on Machine Learning and Applications (ICMLA): Class Imbalances on Past, Present, and Future Workshop*. ICMLA, 2012, pp. 123–126.

[18] Y. Pawitan, J. Bjohle, L. Amler, A.-L. Borg, S. Egyhazi, P. Hall, X. Han, L. Holmberg, F. Huang, S. Klaar, E. Liu, L. Miller, H. Nordgren, A. Ploner, K. Sandelin, P. Shaw, J. Smeds, L. Skoog, S. Wedren, and J. Bergh, "Gene expression profiling spares early breast cancer patients from adjuvant therapy: derived and validated in two population-based cohorts," *Breast Cancer Research*, vol. 7, no. 6, pp. R953–R964, 2005. [Online]. Available: http://breast-cancer-research.com/content/7/6/R953

[19] S. Ramaswamy, K. N. Ross, E. S. Lander, and T. R. Golub, "A molecular signature of metastasis in primary solid tumors," *Nature Genetics*, vol. 33, pp. 49–54, 2003.

[20] Y. Saeys, I. n. Inza, and P. Larranaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007. [Online]. Available: http://bioinformatics.oxfordjournals.org/cgi/content/abstract/23/19/2507

[21] N. Seliya, T. M. Khoshgoftaar, and J. Van Hulse, "A study on the relationships of classifier performance metrics," in *21st International Conference on Tools with Artificial Intelligence*, November 2009, pp. 59–66.

[22] J. Van Hulse, T. M. Khoshgoftaar, and A. Napolitano, "A comparative evaluation of feature ranking methods for high dimensional bioinformatics data," in *2011 IEEE International Conference on Information Reuse and Integration (IRI)*, August 2011, pp. 315–320.

[23] J. Van Hulse, T. M. Khoshgoftaar, A. Napolitano, and R. Wald, "Feature selection with high-dimensional imbalanced data," in *IEEE International Conference on Data Mining Workshops, 2009. ICDMW '09.*, Y. Saygin, J. X. Yu, H. Kargupta, W. Wang, S. Ranka, P. S. Yu, and X. Wu, Eds. IEEE Computer Society, December 2009, pp. 507–514.

[24] J. Van Hulse, T. M. Khoshgoftaar, and A. Napolitano, "Experimental perspectives on learning from imbalanced data," in *Proceedings of the 24th international conference on Machine learning*, ser. ICML '07. New York, NY, USA: ACM, 2007, pp. 935–942. [Online]. Available: http://doi.acm.org/10.1145/1273496.1273614

[25] R. Wald and T. M. Khoshgoftaar, "Patient response datasets: Challenges and opportunities," in *IEEE 14th International Conference on Information Reuse and Integration (IRI 2013)*, 2013, pp. 254–261.

[26] R. Wald, T. M. Khoshgoftaar, A. Fazelpour, and D. J. Dittman, "Hidden dependencies between class imbalance and difficulty of learning for bioinformatics datasets," in *IEEE 14th International Conference on Information Reuse and Integration (IRI 2013)*, 2013, pp. 232–238.

[27] Y. Wang, I. V. Tetko, M. A. Hall, E. Frank, A. Facius, K. F. X. Mayer, and H. W. Mewes, "Gene selection from microarray data for cancer classification–a machine learning approach," *Computational Biology and Chemistry*, vol. 29, no. 1, pp. 37–46, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/B73G2-4F92463-1/2/55dd1384ae9cb8b7c2909abc8afba4f8

[28] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical machine learning tools and techniques*, 3rd ed. Burlington, MA: Morgan Kaufmann, January 2011.

[29] E.-J. Yeoh, M. E. Ross, S. A. Shurtleff, W. Williams, D. Patel, R. Mahfouz, F. G. Behm, S. C. Raimondi, M. V. Relling, A. Patel, C. Cheng, D. Campana, D. Wilkins, X. Zhou, J. Li, H. Liu, C.-H. Pui, W. E. Evans, C. Naeve, L. Wong, and J. R. Downing, "Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling," *Cancer Cell*, vol. 1, no. 2, pp. 133 – 143, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1535610802000326

[30] L. Yu and H. Liu, "Efficient feature selection via analysis of relevance and redundancy," *J. Mach. Learn. Res.*, vol. 5, pp. 1205–1224, Dec. 2004. [Online]. Available: http://dl.acm.org/citation.cfm?id=1005332.1044700

[31] ——, "Redundancy based feature selection for microarray data," in *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM, 2004, pp. 737–742.