Feature Selection with High-Dimensional Imbalanced Data

Jason Van Hulse, Taghi M. Khoshgoftaar, Amri Napolitano, Randall Wald
jvanhulse@gmail.com, taghi@cse.fau.edu, anapoli1@fau.edu, rdwald@gmail.com
Department of Computer and Electrical Engineering and Computer Science
Florida Atlantic University, Boca Raton, Florida, USA

Abstract

Feature selection is an important topic in data mining, especially for high dimensional datasets. Filtering techniques in particular have received much attention, but detailed comparisons of their performance is lacking. This work considers three filters using classifier performance metrics and six commonly-used filters. All nine filtering techniques are compared and contrasted using five different microarray expression datasets. In addition, given that these datasets exhibit an imbalance between the number of positive and negative examples, the utilization of sampling techniques in the context of feature selection is examined.

1. Introduction

Given a dataset D with a set of attributes, independent variables, or features $\mathcal{F} = (X^1, \ldots, X^m)$, the objective of feature selection is to select a subset of features $\mathcal{F}^0 = (X^{j_1}, \ldots, X^{j_p})$ such that $p << m$ and $\mathcal{F}^0$ satisfies the particular conditions of the task at hand. For example, in a classification setting, the objective may be to extract the set of features that maximize classifier accuracy. The implicit assumption in feature selection tasks is that most of the information inherent in the dataset can be captured using only a small subset of attributes. Reducing the number of features in a dataset can have numerous benefits such as faster model training, reduced susceptability to overfitting, offsetting the pernicious effects of the curse of dimensionality, and reducing storage, memory, and processing requirements during data analysis [10]. The obvious drawback of feature selection is the possibility that a critical attribute will be omitted, thereby hurting classification performance.

Filtering techniques are among the simplest feature selection methodologies and have been the subject of numerous studies. This work presents a detailed comparison among six commonly-used filtering techniques. In addition, we propose three threshold-based filters derived from classification performance metrics. Experiments are conducted using five different microarray expression datasets of high dimensionality, and a key contribution of our work relates to learning from complex datasets by mining multiple data sources and combining the results in a unique way by measuring the Frobenius distances between correlation matrices. Our work is relevant to mining multiple data sources from another perspective as well. The integration and combination of numerous high dimensional datasets can be a very difficult effort. Feature selection can be used to substantially reduce the set of features from all datasets, greatly facilitating the data integration process.

A unique contribution of our work is that we compare the underlying attribute rankings of each technique, as opposed to building classifiers using the selected features and comparing performance metrics, such as overall accuracy, of those classifiers. By measuring the rank correlation between the attribute rankings, it is easier to discern which techniques produce similar results irrespective of the ultimate use of the data. For example, feature selection techniques A and B may result in similar accuracy when used in conjunction with classifier Z, but will the same results hold if classifiers X or Y are used? In such a situation, it is not clear if feature selection techniques A and B are truly similar for all classifiers or only for classifier Z. Further, biologists are often more interested in identifying a small set of important genes, as opposed to building a classifier from the selected genes. Two filtering techniques that produce different attribute rankings can provide additional interesting information to biologists. Therefore, to get a better understanding of the true similarities among techniques, we directly analyze the attribute rankings.

Another unique contribution of this study is related to imbalanced classes and their relation to filtering techniques. In a binary classification setting, a dataset is imbalanced if one class (the majority or negative class) outnumbers the other (minority or positive class). In an imbalanced setting, data sampling is often used to balance the class distribution prior to feature selection [22, 24]. While the primary objective of this work is to understand the relationships among the nine filtering techniques, a secondary objective is to un-

978-0-7695-3902-7/09 $26.00 © 2009 IEEE
DOI 10.1109/ICDMW.2009.35

IEEE computer society

derstand if these relationships hold after the application of different sampling techniques.

2. Related Work

The amount of data generated in the domain of molecular biology exploded, resulting in the new field of bioinformatics: the use of computer applications, in particular data mining, to analyze biological data and generate meaningful results. A variety of biological data sources are amenable to bioinformatic analysis, from protein sequences [20] and properties [14] to genetic codes [23] to mass spectroscopy results [15]. Of particular interest is the problem of analyzing microarray data.

Much of the research on microarray analysis has focused on improving classification models used to categorize unknown samples as "healthy" or "sick". Because the data have thousands of features, feature selection is a necessity. A variety of techniques have been employed for the selection step. Some researchers only use the standard array of feature ranking and subset evaluation filters and wrappers, coupled with traditional data mining techniques; these either analyze filters alone [8] or compare filters and wrappers [11] or filters, wrappers, and principal component analysis [16]. Others employ genetic algorithms [12] or minimum redundancy [18] to find the optimal subset of genes for classification purposes. Others propose novel feature selection techniques designed for microarray analysis [17].

One common feature of these papers is that they evaluate the quality of their feature selection techniques by testing their ability to classify the data. The feature rankings themselves might undergo some ad hoc comparisons [12], but no systemic study comparing feature selection algorithms on their own has been conducted. The closest the literature comes to this approach is in evaluating feature clustering techniques [1]; these also treat genes as features while focusing specifically on finding novel genes which are especially relevant to the underlying biological problem. Nonetheless, using the feature ranking techniques described in this work without the addition of a classifier opens up a new strategy for identifying genes and proteins associated with disease states.

Chen and Wasikowski [4] propose a technique called FAST which is based on the area under a ROC curve generated by moving the decision boundary of a single feature classifier with thresholds placed using an even-bin distribution. The technique we propose in this work (Section 4) is much more general than that of Chen and Wasikowski. Their work calculates a ROC curve by discretizing the distribution, while ours does not require discretization, making it more precise and eliminating the often vexing question of how wide the bins should be. Futher, we utilize three different classifier performance metrics, and note that our technique can be extended to many others (an exploration of which we leave to future work).

3 Filter-Based Ranking

A great deal of work has been conducted on various filter-based approaches to feature selection [7, 10, 27, 21]. These employ a variety of filtering strategies, examining the relevance of each feature to the class value. Some of the more frequently-used strategies are the χ^2 statistic, information gain, gain ratio, ReliefF, and symmetric uncertainty.

3.1 χ^2 Statistic

Feature selection using the χ^2 statistic is analogous to performing a hypothesis test on the distribution of the class as it relates to the values of the feature in question. The null hypothesis is that there is no correlation; each value is as likely to have instances in any one class as any other class. Under the null hypothesis, if p of the instances have a given value and q of the instances are in a specific class, $(p \cdot q)/n$ instances have a given value and are in a specific class (n is the total number of instances in the dataset). This is because p/n instances have the value and q/n instances are in the class, and if the probabilities are independent (i.e., the null hypothesis) their joint probability is their product. Given the null hypothesis, the χ^2 statistic measures how far away the actual value is from the expected value:

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(O_{i,j} - E_{i,j})^2}{E_{i,j}}. \tag{1}$$

In this equation, r is the number of different values of the feature in question, c is the number of classes in question (in this work, $c = 2$), $O_{i,j}$ is the number of instances with value i which are in class j, and $E_{i,j}$ is the expected number of instances with value i and class j, based on $(p \cdot q)/n$. The larger this chi-squared statistic, the more unlikely it is that the distribution of values and classes are independent; that is, they are related, and the feature in question is relevant to the class.

3.2 Information Gain

A second feature evaluator is the information gain (IG) metric. This is the application of a more general technique, the measurement of informational entropy, to the problem of deciding how important a given feature is. Informational entropy, when measured using Shannon entropy, is notionally the number of bits of data it would take to encode a given piece of information [27]. The more space a piece of

information takes to encode, the more entropy it has. Intuitively, this makes sense because a random string has maximum entropy and cannot be compressed, while a highly-ordered string can be written with a brief description of the string's information.

In the context of classification, the distribution of instances among classes is the information in question. If the instances are randomly assigned among the classes, the number of bits necessary to encode this class distribution is high, because each instance would need to be enumerated. On the other hand, if all the instances are in a single class, the entropy would be lower, because the bit-string would simply say "All instances save for these few are in the first class." Therefore a function measuring entropy must increase when the class distribution gets more spread out and be able to be applied recursively to permit finding the entropy of subsets of the data. The following formula satisfies both of these requirements:

$$H(D) = -\sum_{i=1}^{l}(n_i/n)\log(n_i/n) \tag{2}$$

where dataset D has $n = \mid D \mid$ instances and n_i members in class $c_i, i = 1, \ldots, l$. The entropy of any subset $D_0 \subset D$ can be calculated using Eq. 2 by only considering instances in D_0. To find the information gain of feature X, sum the entropy for each value ($X_j, j = 1, \ldots, m$) of the feature

$$H(D|X) = \sum_{j=1}^{m}(|X_j|/n)H(D|X = X_j), \tag{3}$$

where $H(D|X = X_j)$ is the entropy calculated relative to the subset of instances that have a value of X_j for attribute X. If X is a good description of the class, each value of that feature will have little entropy in its class distribution; for each value most of the instances should be primarily in one class. The information gain of an attribute is measured by the reduction in entropy: $IG(X) = H(D) - H(D|X)$. The greater the decrease in entropy when considering attribute X individually, the more significant feature X is for prediction.

3.3 Gain Ratio

The gain ratio (GR) is often used to overcome one of the important flaws inherent in information gain, specifically that information gain tends to favor attributes with a large number of distinct values. The classic example is a customer database which includes each customer's credit card number. If that number were used as a feature for feature selection, it would have a perfect information gain, because for each value, all the instances with that value are in the same class. Generally speaking, a feature may be useful when maximizing the information gain while simultaneously minimizing the number of attribute values. $IV(X)$ is defined as the intrinsic value of attribute X:

$$IV(X) = -\sum_{i=1}^{p}(|X_i|/n)\log(|X_i|/n) \tag{4}$$

where $|X_i|$ is the number of instances where attribute X takes value X_i, the number of distinct values of X is p, and n is the total number of instances in the dataset. IV adds a separate term for each value of the feature times the log of the relative number of instances in each value. The gain ratio of X is defined as the entropy of X divided by the intrinsic value of X:

$$GR(X) = \frac{IG(X)}{IV(X)} = \frac{H(D) - H(D|X)}{H(X)}. \tag{5}$$

3.4 Relief and ReliefF

Relief [13] is an instance-based feature ranking technique. ReliefF is an extension of the Relief algorithm that can handle noise and multiclass data sets, and is implemented in the Weka data mining tool [26]. When the weightByDistance (weight nearest neighbors by their distance) parameter was set as default (false), the algorithm is referred to as Relief; when the parameter was set to true, the algorithm is referred to as Relief-W.

3.5 Symmetric Uncertainty

Symmetric uncertainty (SymUn) applied to evaluating feature-class correlation has many similarities to gain ratio, though its original formulation was designed to find the redundency between two features. It is found using the following formula:

$$U(D, X) = 2 \cdot \frac{H(D) - H(D|X)}{H(D) + H(X)} \tag{6}$$

As before, D is the set of all instances, considered based on how they are divided up into classes; X is the set of all instances, considered based on how they are divided up by the feature in question; and $H(D|X)$ is the entropy of D (that is, the class) when considered within each value of the feature in question.

4 Threshold-Based Filters

Three filter-based attribute ranking techniques are proposed in this section. These procedures were developed and implemented by our research group within Weka [26]. Each

independent attribute is paired individually with the class attribute and that two attribute dataset is evaluated using different performance metrics. The three proposed techniques simply represent three different classifier performance metrics calculated in the evaluation of the hypothetical dataset containing only the single attribute currently being ranked (in addition to the class attribute). Typically, classifier performance metrics are calculated based on a set of posterior probabilities (between 0 and 1). The attribute ranking framework we propose is equivalent to normalizing the attribute values (so that they fall between 0 and 1) and treating those values as the posterior probabilities from which to calculate performance metrics.

More specifically, feature X^j is mapped to $\hat{X}^j$

$$X^j \mapsto \hat{X}^j = \frac{X^j - \min(X^j)}{\max(X^j) - \min(X^j)}. \tag{7}$$

$\hat{X}^j$ can now be thought of as a posterior probability, with an important exception: the default decision threshold of 0.5 used in standard binary classification to assign the predicted class may not be sensible. For example, given an attribute with a few very large values, almost all examples may have a transformed value below 0.5, making this threshold impractial. Therefore, we propose the use of performance measures that can be calculated at various points in the distribution of $\hat{X}^j$, either taking the maximum possible value (F-measure or geometric mean) or the area under the true positive/false positive curve (AUC).

Analogous to the procedure for calculating rates in a classification setting with a posterior probability, the true positive (TPR), true negative (TNR), false positive (FPR), and false negative (FNR) rates can be calculated at each threshold $t \in [0, 1]$ relative to the normalized attribute $\hat{X}^j$. For example,

$$TPR(t) = \frac{\#\text{ of positive class examples with } \hat{X}^j > t}{\#\text{ of positive class examples}}. \tag{8}$$

$FNR(t)$, $TNR(t)$, and $FPR(t)$ can be defined similarly. Precision $PRE(t)$ is defined as the number of positive examples with $\hat{X}^j > t$ divided by the total number of examples with $\hat{X}^j > t$. The three attribute ranking techniques we propose, AUC, F-measure, and geometric mean, use these five accuracy rates as described below. AUC, F-measure, and geometric mean are calculated for each attribute individually, and attributes with higher values are determined to better predict the class attribute.

4.1 ROC Curves

Receiver Operating Characteristic [19], or ROC, curves graph true positive rate on the y-axis versus the false positive rate on the x-axis. The resulting curve illustrates the trade-off between detection rate and false alarm rate. The ROC curve demonstrates the performance of a classifier across the complete range of possible decision thresholds, and accordingly does not assume any particular misclassification costs or class prior probabilities. In this study, ROC curves are generated by varying the decision threshold t used to transform the normalized attribute values into a predicted class. In other words, the true positive and false positive rates are calculated as the threshold for the normalized attribute varies from 0 to 1. The area under the ROC curve (AUC) is used to provide a single numerical metric for comparing the predictive power of each attribute. This definition is different than the one used by Chen and Wasikowski [4], which consider only a small subset of the possible threshold values when calculating the true positive and false positive rates.

4.2 Geometric Mean

The geometric mean (GM) is square root of the product of the true positive rate and true negative rate. GM ranges from 0 to 1, and an attribute that is perfectly correlated to the class provides a value of 1. GM is a useful performance measure since it is inclined to maximize the true positive rate and the true negative rate while keeping them relatively balanced. Such error rates are often preferred, depending on the application domain. GM is calculated at each value of the normalized attribute range, and the maximum value of GM is used as a measure of attribute strength.

$$\text{GM} = \max_{t \in [0,1]} \sqrt{TPR(t) \times TNR(t)} \tag{9}$$

4.3 F-Measure

The F-measure (F) is derived from *recall* (or true positive rate) and *precision*. F-measure uses a tunable parameter β to indicate the relative importance of recall and precision. In other words, β can be modified to place more emphasis on either recall or precision. Typically, $\beta = 1$ is used (as is the case in our study).

$$\text{F-measure} = \max_{t \in [0,1]} \frac{(1 + \beta^2) \times TPR(t) \times PRE(t)}{\beta^2 \times TPR(t) + PRE(t)}. \tag{10}$$

Recall and precision are calculated at each point along the normalized attribute range of 0 to 1. The maximum F-measure obtained by each attribute represents how strongly that particular attribute relates to the class, according to the F-measure.

5 Data Sampling Techniques

We apply four common data sampling techniques, each of which has been shown to be effective at improving clas-

Dataset Name	Abbreviation	# Attributes	# Instances	# Positive	% Positive
ECML Pancreas	ECML	27680	90	8	8.9%
Central Nervous System	CNS	7130	60	21	35.0%
Colon	Colon	2001	62	22	35.5%
DLBCL Tumor	Tumor	7130	77	19	24.7%
Lung Cancer	Cancer	12534	181	31	17.1%

Table 1. The Microarray Expression datasets

sification performance in previous research [24]. Random undersampling, random oversampling, and SMOTE each require a parameter indicating the percentage of minority class examples after the application of the technique - in this work, we balance the class distribution exactly so that after sampling, 50% of the instances belong to the minority class. Wilson's editing does not require a user specified class distribution.

5.1 Random Resampling

The two most common data sampling techniques are *random oversampling* (ROS) and *random undersampling* (RUS). Random oversampling duplicates instances (selected randomly) of the minority class. While this does help to balance the class distribution, no new information is added to the dataset and this may lead to overfitting [6]. Also, the size of the training datasets is increased, which causes longer model training times. Random undersampling randomly discards instances from the majority class. In doing so, the class distribution can be balanced, but important information can be lost when examples are discarded at random.

5.2 SMOTE

Chawla et al. [3] proposed an intelligent oversampling method called Synthetic Minority Oversampling Technique or SMOTE. SMOTE (denoted SM in this work) adds new, artificial minority examples by interpolating between pre-existing minority instances rather than simply duplicating original examples. The newly created instances cause the minority regions of the feature-space to be fuller and more general. The technique first finds the k nearest neighbors of the minority class for each minority example (the paper recommends $k = 5$). The artificial examples are then generated in the direction of some or all of the nearest neighbors, depending on the amount of oversampling desired.

5.3 Wilson's Editing

Wilson's editing (WE), proposed by Barandela et al. [2], modifies an older strategy (by Wilson [25]) for pruning a dataset for use with instance based learning. Using the kNN classifier with $k = 3$, WE removes all misclassified majority class examples.

6. Kendall's Tau Rank Correlation

Kendall's Tau rank correlation statistic [5] is used to measure the degree of similarity between the attribute rankings of two techniques. Suppose the rankings r_1 and r_2 are being compared. For each attribute j in the dataset there is an ordered pair $(r_1(j), r_2(j))$ where $r_1(j)$ and $r_2(j)$ are the rankings of attribute j produced by r_1 and r_2. For each pair of attributes (j_1, j_2) the rankings $(r_1(j_1), r_2(j_1))$ and $(r_1(j_2), r_2(j_2))$ are compared and given a value of $+1$ or -1 depending on whether the two rankings are concordant or discordant. Assuming that both r_1 and r_2 do not contain tied ranks, a pair of attributes (j_1, j_2) is considered *concordant* if $r_1(j_1) > r_1(j_2)$ and $r_2(j_1) > r_2(j_2)$ or $r_1(j_1) < r_1(j_2)$ and $r_2(j_1) < r_2(j_2)$. Otherwise, (j_1, j_2) are said to be *discordant*. There are a total of $\frac{n(n-1)}{2}$ pairs of attributes. If S is the sum of the scores for each pair of attributes as determined by their concordance or discordance, Kendall's Tau is calculated as $\tau = S/\frac{n(n-1)}{2}$. If all pairs are concordant, then the two rankings are in complete agreement and $\tau = 1$. If the two rankings are exactly opposite, (i.e., $\forall\ j,\ r_2(j) = n - r_1(i) + 1$), then all pairs will be discordant and $\tau = -1$. If τ is close to zero, then the correlation between the two rankings is very weak.

7. Datasets

Table 1 lists the five microarray expression datasets utilized in this work. Also included are the numbers of attributes, total instances, and positive class instances. All datasets are imbalanced relative to the class attribute. All attributes in these datasets are numeric.

8. Experimental Results

The primary objective of this work is to compare the performance of the various filter techniques. This objective is accomplished by comparing the attribute rankings produced by each filter—filter techniques that are highly correlated, based on Kendall's Tau rank correlation τ, are similar to one another.

Std/Avg	χ^2	GR	IG	Relief	Relief-W	SymUn	F	GM	AUC
χ^2		0.94652	0.97441	0.11210	0.10161	0.96820	0.18992	0.16061	0.16314
GR	0.060		0.94332	0.09698	0.08691	0.96499	0.16521	0.13797	0.14276
IG	0.028	0.069		0.11375	0.10441	0.96856	0.19468	0.16695	0.16971
Relief	0.076	0.061	0.078		0.59257	0.10717	0.26312	0.29815	0.22130
Relief-W	0.070	0.058	0.074	0.119		0.09730	0.23818	0.25224	0.19907
SymUn	0.034	0.041	0.037	0.071	0.066		0.18188	0.15321	0.15699
F	0.170	0.135	0.178	0.099	0.094	0.159		0.65904	0.60611
GM	0.138	0.108	0.148	0.070	0.078	0.129	0.193		0.66562
AUC	0.132	0.106	0.142	0.142	0.122	0.124	0.122	0.073	

Figure 1. Correlation Among Filtering Techniques Over All Five Datasets

None\WE	χ^2	GR	IG	Relief	Relief-W	SymUn	F	GM	AUC
χ^2		0.99892	0.99795	0.05557	0.05628	0.99963	0.05532	0.04802	0.05383
GR	0.99925		0.99738	0.05536	0.05613	0.99910	0.05487	0.04757	0.05338
IG	0.99919	0.99881		0.05542	0.05619	0.99778	0.05533	0.04861	0.05439
Relief	0.05158	0.05136	0.05135		0.62079	0.0555	0.30323	0.24853	0.02958
Relief-W	0.05206	0.05188	0.05191	0.62218		0.05622	0.24283	0.21428	0.06115
SymUn	0.99964	0.99942	0.99904	0.05152	0.05201		0.05515	0.04788	0.05370
F	0.05279	0.05238	0.05273	0.30289	0.23989	0.05263		0.78001	0.60473
GM	0.04502	0.04462	0.04551	0.24726	0.21077	0.04490	0.77907		0.59363
AUC	0.04991	0.04952	0.05042	0.02798	0.06154	0.04978	0.60258	0.59248	

Figure 2. Comparison of Correlations for None and WE, ECML Dataset

For each dataset, τ is calculated for all pairs of filters (first we consider the similarity of the filters without the use of sampling). The average and standard deviation of the correlations over all five datasets are computed, with the results presented in Figure 1. The entries in the matrix above the diagonal represent the average τ over all five datasets, while the entries below the diagonal represent the standard deviation of τ over the five datasets. For example, the attribute rankings produced by χ^2 and GR had an average rank correlation of 0.94652 with a standard deviation of 0.060. The cells in Figure 1 with dark shading represent average correlations above 0.9, while cells in light shading represent average correlations between 0.5 and 0.9. Three distinct clusters of filter techniques can be observed from the correlation matrix. There is clearly a significant relationship among SymUn, GR, IG, and χ^2, all of which have correlations well above 0.9. Relief and Relief-W exhibit moderate correlation, and the three filters based on classification performance metrics (AUC, F, and GM) also exhibit moderate correlation. The remaining pairs of filters show a less significant relationship, with τ generally much less than 0.5.

The relationships presented in Figure 1 occur when the filters are utilized on the unsampled dataset. It is interesting to consider what impact, if any, the application of sampling prior to filtering has on the correlation structure presented in Figure 1. Further, we can investigate which sampling techniques result in similar correlation structures. To perform this analysis, the Frobenius (or Hilbert-Schmidt) norm ϕ is used [9]. Let $C^{\{D_k,s_l\}}$ be the 9×9 correlation matrix for dataset D_k and sampling technique s_l - there are 25 such correlation matrices. Entry $C_{i,j}$ is the Kendall's Tau rank correlation between filters f_i and f_j, i.e., $C_{i,j} = \tau_{D_k,s_l}(f_i, f_j)$.

The distance between two correlation matrices $C^{\{D_k,s_a\}}$ and $C^{\{D_k,s_b\}}$ using the Frobenius norm ϕ is calculated as:

$$\begin{aligned}\phi_{\{D_k,s_a,s_b\}} &= ||C^{\{D_k,s_a\}} - C^{\{D_k,s_b\}}||_2 \\ &= \sqrt{\sum_{i=1}^{9}\sum_{j=1}^{9}\left(C_{i,j}^{\{D_k,s_a\}} - C_{i,j}^{\{D_k,s_b\}}\right)^2}.\end{aligned}$$

Since the correlation matrices are symmetric with 1 on the diagonal entries, the Frobenius norm simplifies to

$$\phi_{\{D_k,s_a,s_b\}} = \sqrt{2\sum_{i,j=1;i<j}^{9}\left(C_{i,j}^{\{D_k,s_a\}} - C_{i,j}^{\{D_k,s_b\}}\right)^2}.$$

Consider for example the data presented in Figure 2, where entries above the diagonal represent rank correlations between the filters after the application of WE, while the entries below the diagonal are rank correlations between filters without sampling (None) for dataset ECML. Clearly the correlation matrices for WE and None are very similar to one another. Table 2 contains the Frobenius norms of all 10 combinations of sampling techniques, with the most similar pair at the top of the table. The Frobenius norms obtained

None\ROS	χ^2	GR	IG	Relief	Relief-W	SymUn	F	GM	AUC
χ^2		0.68873	0.98037	0.30884	0.26440	0.95631	0.35520	0.61099	0.56923
GR	0.99925		0.69242	0.14336	0.10653	0.72387	0.44093	0.45780	0.57526
IG	0.99919	0.99881		0.30444	0.26021	0.95784	0.35728	0.60568	0.56667
Relief	0.05158	0.05136	0.05135		0.77650	0.29073	-0.08859	0.50128	0.21341
Relief-W	0.05206	0.05188	0.05191	0.62218		0.24687	-0.12170	0.43961	0.17068
SymUn	0.99964	0.99942	0.99904	0.05152	0.05201		0.37081	0.60183	0.57853
F	0.05279	0.05238	0.05273	0.30289	0.23989	0.05263		0.30736	0.45851
GM	0.04502	0.04462	0.04551	0.24726	0.21077	0.04490	0.77907		0.59263
AUC	0.04991	0.04952	0.05042	0.02798	0.06154	0.04978	0.60258	0.59248	

Figure 3. Comparison of Correlations for None and ROS, ECML Dataset

Sampling Combination	ECML	CNS	Colon	Tumor	Cancer	Mean	Std. Dev.
None-WE	0.024	0.398	0.659	0.181	0.000	0.252	0.277
ROS-SM	0.820	0.261	0.226	0.530	1.363	0.640	0.470
None-RUS	0.981	0.376	0.513	0.677	0.844	0.678	0.244
RUS-WE	0.980	0.262	1.036	0.647	0.844	0.754	0.313
SM-WE	2.721	0.547	0.614	1.436	1.329	1.330	0.876
ROS-WE	2.795	0.657	0.703	1.577	1.223	1.391	0.873
None-SM	2.737	0.624	0.950	1.545	1.329	1.437	0.808
None-ROS	2.812	0.608	0.965	1.662	1.223	1.454	0.851
RUS-SM	2.659	0.509	1.142	1.588	1.391	1.458	0.785
RUS-ROS	2.753	0.579	1.112	1.742	1.357	1.509	0.814

Table 2. Frobenius norm distances for different sampling techniques

for each dataset separately and the average and standard deviation over all five datasets are provided. The correlation matrices obtained by None and WE were the most similar, with a mean distance of 0.252. Further on dataset ECML, the Frobenius norm distance is very small (0.024), which corresponds well with the data presented in Figure 2.

Figure 3 displays the Kendall's Tau rank correlations for the filters for dataset ECML for random oversampling above the diagonal and None below the diagonal. From Table 2, this combination of sampling techniques obtains a Frobenious norm distance of 2.812 for dataset ECML, the largest for this particular dataset. Averaged over all five datasets, ROS and None exhibit significant differences. Clearly from Figure 3, the correlation structures between filters are markedly different for ROS and None. For example, the correlation between χ^2 and GR is only 0.68873 after the application of random undersampling, while without the use of sampling, χ^2 and GR are highly correlated.

From Table 2, None and WE clearly result in very similar correlation structures among the nine filters with an average Frobenius norm distance of 0.252. In addition, this combination is similar for all five datasets and exhibits a relatively small standard deviation. As oversampling techniques, ROS and SM both result in similar correlation structures for the various filters, while on the other hand, RUS-SM and RUS-ROS obtain the two highest Frobenius norm distances. Generally speaking, there appear to be three clusters of sampling techniques in Table 2: None-WE in a unique cluster are the two most similar sampling techniques; ROS-SM, None-RUS, and RUS-WE are clustered together with average Frobenius norm distances between 0.6 and 0.8; and the remaining six combinations are clustered together with average Frobenius norm distances above 1.

9. Conclusions

Filtering is a common methodology for feature selection. This work has presented detailed experiments using nine different filtering algorithms for feature selection given high-dimensional microarray expression data. Generally speaking, the performance of the χ^2, gain ratio, information gain, and symmetric uncertainty are all highly correlated to one another. ReliefF and ReliefF-W showed moderate correlation—a priori, it might have been expected that this correlation would be higher since the techniques are very similar, but the instance weighting methodology appears to make a significant difference in the feature ranking results. Finally the three performance metric-based techniques AUC, GM, and F are also moderately correlated. The fact that the performance-metric based filters are not highly correlated with the other six, more commonly used filters such as χ^2 implies that these new filters are providing potentially interesting information not captured by the other techniques. Future work should consider performance-metric based filters in more detail and investigate whether hybrid filters might be useful to consider.

Another conclusion of this work is that the correlation structure can change dramatically if sampling techniques

are used to first balance the data. This interesting observation regarding the interaction between sampling techniques and filters suggests that more work should be conducted to gain a further understanding of this relationship. Future work should also consider the construction of classification models from the data run through feature selection algorithms to compare performance on learning tasks. Additional performance metrics can be utilized in our proposed threshold-based filtering algorithms. Finally, we recommend further work using additional datasets from the biological domain as well as datasets from other application domains.

References

[1] W.-H. Au, K. C. C. Chan, A. K. C. Wong, and Y. Wang. Attribute clustering for grouping, selection, and classification of gene expression data. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, 2(2):83–101, 2005.

[2] R. Barandela, R. M. Valdovinos, J. S. Sanchez, and F. J. Ferri. The imbalanced training sample problem: Under or over sampling? *In Joint IAPR International Workshops on Structural, Syntactic, and Statistical Pattern Recognition (SSPR/SPR'04), Lecture Notes in Computer Science 3138*, (806-814), 2004.

[3] N. V. Chawla, L. O. Hall, K. W. Bowyer, and W. P. Kegelmeyer. SMOTE: Synthetic minority oversampling technique. *Journal of Artificial Intelligence Research*, (16):321–357, 2002.

[4] X.-w. Chen and M. Wasikowski. Fast: a roc-based feature selection metric for small samples and imbalanced data classification problems. In *KDD '08: Proc. 14th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining*, pages 124–132, New York, NY, USA, 2008. ACM.

[5] W. J. Conover. *Practical Nonparametric Statistics*. Wiley, 2nd edition, 1971.

[6] C. Drummond and R. C. Holte. C4.5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on Learning from Imbalanced Data Sets II, International Conference on Machine Learning*, 2003.

[7] G. Forman. An extensive empirical study of feature selection metrics for text classification. *J. Mach. Learn. Res.*, 3:1289–1305, 2003.

[8] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000.

[9] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 3rd edition, 1996.

[10] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182, 2003.

[11] I. Inza, P. Larraaga, R. Blanco, and A. J. Cerrolaza. Filter versus wrapper gene selection approaches in dna microarray domains. *Artificial Intelligence in Medicine*, 31(2):91 – 103, 2004. Data Mining in Genomics and Proteomics.

[12] T. Jirapech-Umpai and S. Aitken. Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes. *BMC Bioinformatics*, 6(1):148, 2005.

[13] K. Kira and L. A. Rendell. The feature selection problem: Traditional methods and a new solution. In *AAAI '92: Proc. 10th Nat'l Conf. on Artificial Intelligence*, number 10, pages 129–134. John Wiley & Sons, Ltd., July 1992.

[14] B. J. Lee, H. G. Lee, J. Y. Lee, and K. H. Ryu. Classification of enzyme function from protein sequence based on feature representation. pages 741–747, Oct. 2007.

[15] L. Li, H. Tang, Z. Wu, J. Gong, M. Gruidl, J. Zou, M. Tockman, and R. A. Clark. Data mining techniques for cancer detection using serum proteomic profiling. *Artificial Intelligence in Medicine*, 32(2):71 – 83, 2004.

[16] F. Model. Feature selection for dna methylation based cancer classification. *Bioinformatics*, 17:157–164(8), June 2001.

[17] S. D. Peddada, E. K. Lobenhofer, L. Li, C. A. Afshari, C. R. Weinberg, and D. M. Umbach. Gene selection and clustering for time-course and dose-response microarray experiments using order-restricted inference. *Bioinformatics*, 19(7):834–841, 2003.

[18] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.

[19] F. Provost and T. Fawcett. Robust classification for imprecise environments. *Machine Learning*, 42:203–231, 2001.

[20] P. Radivojac, N. V. Chawla, A. K. Dunker, and Z. Obradovic. Classification and knowledge discovery in protein databases. *Journal of Biomedical Informatics*, 37(4):224–239, 2004. Biomedical Machine Learning.

[21] Y. Saeys, I. Inza, and P. Larranaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517, 2007.

[22] C. Seiffert, T. M. Khoshgoftaar, and J. Van Hulse. Hybrid sampling for imbalanced data. *Journal of Integrated Computer-Aided Engineering*, 16(3):193–210, 2009.

[23] Y. Sun, M. Robinson, R. Adams, R. te Boekhorst, A. Rust, and N. Davey. Using feature selection filtering methods for binding site predictions. volume 1, pages 566–571, July 2006.

[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 (ICML 2007)*, pages 935–942, Corvallis, OR, USA, June 2007.

[25] D. Wilson. Asymptotic properties of nearest neighbor rules using edited data sets. *IEEE Trans. on Systems, Man and Cybernetics*, (2):408–421, 1972.

[26] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, California, 2nd edition, 2005.

[27] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In *ICML '97: Proc. 14th Int'l Conf. Machine Learning*, pages 412–420, 1997.