Development of Correlation Based Feature Selection Method by Predicting the Markov Blanket for Gene Selection Analysis

Made Adi[1] **Zhen Yun**[2] **Kwoh Chee Keong**[2]
Bioinformatics Research Centre (BIRC), Nanyang Technological University, Singapore
Email : k501773@ntu.edu.sg, pg04325488@ntu.edu.sg, asckkwoh@ntu.edu.sg

ABSTRACT: In this paper, we propose a heuristic method to select features using a Two-Phase Markov Blanket-based (TPMB) algorithm. The first phase, *filtering phase,* of TPMB algorithm works by filtering the obviously redundant features. A non-linear correlation method based on Information theory is used as a metric to measure the redundancy of a feature [1]. In second phase, *approximating phase*, the Markov Blanket (MB) of a system is estimated by employing the concept of cross entropy to identify the MB. We perform experiments on microarray data and report two popular dataset, AML-ALL [3] and colon tumor [4], in this paper. The experimental results show that the TPMB algorithm can significantly reduce the number of features while maintaining the accuracy of the classifiers.

1. INTRODUCTION

Data classification problem is described as a problem where there is a *dataset* (a collection of data instances) and a *classifier*. Each instance of data contains the same number of features which can be small or large and each data instances is labeled with a *class*. Generally, data instances are obtained from experiments. Then, the dataset is fed into the classifier which a *classification model* based on the provided data will be created. This process is called *classifier learning phase*. Eventually, when we have a data instance which we do not know the class yet, we can input unclassified data instance into the classification model and let the model decide what class this data instance belongs. Figure 1 illustrates the data classification process.

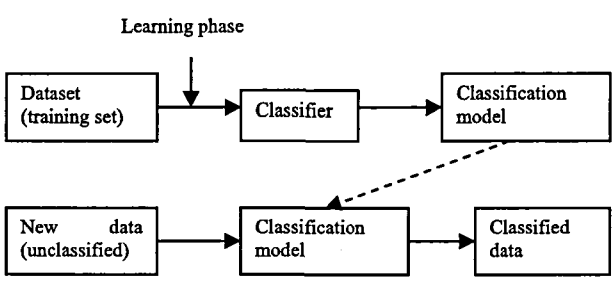

Figure 1: Data Classification Process

In most cases of data classification, a data instance from a dataset will contain a large number of features which most of them are redundant for the classifier learning phase. These redundant features will not help a classifier algorithm to build a good classification model during its training phase. On the contrary, the redundant features may deteriorate the performance of the classifier in terms of, firstly, model accuracy, as redundant data might provide spurious information during classifier's learning phase; and secondly, time taken during learning phase, as there are large number of inputs to be processed

These issues, then, can be addressed as the *curse of dimensionality* which refers to the exponential growth of hypervolume as a function of dimensionality. *Feature selection algorithm* is designed to select a relatively small number of features to represent the whole dataset and the classifier can use the *selected features* for learning process. The performance of feature selection algorithm is usually measured by the number of features it selects, accuracy of the classification model trained using selected features and time needed to do selection.

2. INFORMATION THEORY AND MARKOV BLANKET

2.1 Entropy

Information theory was first introduced by Claude E. Shannon in 1948. He introduced the concept of entropy which is a measure of uncertainty of a random variable. Given X which is a discrete random variable with alphabet X and probability mass function $p(x) = \Pr\{X = x\}, x \in \mathrm{X}$ Thus, the entropy, $H(X)$, of a discrete random variable X is defined as:

$$H(X) = -\sum_{x \in \mathrm{X}} p(x) \log p(x)$$

2.2 Conditional Entropy

When there are two random variables X and Y, we can consider them to be a single vector-valued random variable (X,Y). The *joint entropy* $H(X, Y)$ of a pair of discrete random variables (X, Y) with a joint distribution $p(x, y)$ is defined as

$$H(X,Y) = -\sum_{x \in \mathrm{X}} \sum_{y \in \mathrm{Y}} p(x,y) \log p(x,y)$$

The *conditional entropy* of a random variable is defined as, given another random variable as the expected value of the entropies of the conditional distributions, averaged over the conditioning random variable.

If distribution of $(X,Y) \sim p(x,y)$ then the *conditional entropy* $H(Y|X)$ is defined as

$$H(Y \mid X) = \sum_{x \in \mathrm{X}} p(X) H(Y \mid X = x)$$

2.3 Relative Entropy

The relative entropy is a measure of the distance between two

distributions. In statistics, it arises as an expected logarithm of the likelihood. The relative entropy $D(p||q)$ is a measure of the inefficiency of assuming that the distribution is q when the true distribution is p.

The relative entropy or *Kullback Leibler Distance* between two probability mass function $p(x)$ and $q(x)$ is defined as

$$D(p \| q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)}$$

Entropy is always non-negative and is zero if and only if $p=q$. However it is not a true distance between distributions since it is not symmetric and does not satisfy the triangle inequality. Nonetheless, it is often useful to think of relative entropy as a "distance" between distributions.

2.4 Mutual Information

We use entropy to measure how random a system is. When there are two random variables (X and Y) which are parts of random systems, the prior knowledge of one random variable will reduce the entropy of another system or keep the entropy of another system the same. The amount of entropy removed upon knowing the occurrence of another random variable is called the *mutual information.*

Mutual information, I(X;Y), is defined as the reduction in the uncertainty of X due to the knowledge of Y or vice versa as:

$$I(X;Y) = H(Y) - H(X \mid Y) = H(X) - H(Y \mid X)$$

Mutual information definition in aboved equation is actually the derivation of relative entropy between the joint distribution and the product distribution defined below:

$$\begin{aligned} I(X;Y) &= D(P(x,y) \| P(x)P(y)) \\ &= \sum_x \sum_y P(x,y) \log \frac{P(x,y)}{P(x)P(y)} \\ &= \sum_{x,y} P(x,y) \log \frac{P(x \mid y)}{P(x)} \\ &= -\sum_{x,y} P(x,y) \log P(x) + \sum_{x,y} P(x,y) \log P(x \mid y) \\ &= H(X) - H(X \mid Y) \end{aligned}$$

The relationship between mutual information and entropy can be seen on Figure 2.

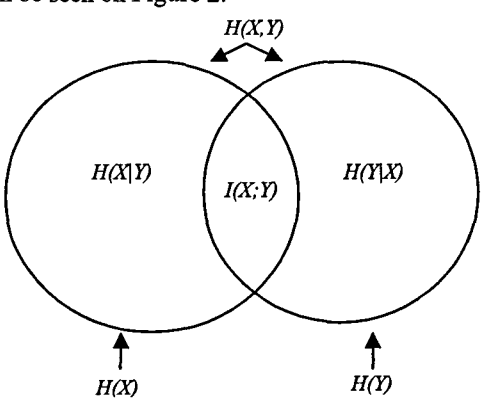

Figure 2: Relationship between Mutual Information and Entropy

2.5 Markov Blanket

Koller and Sahami [2] discussed the concept of *conditional independence* between features. Two variables (e.g, T and X) are said to be conditionally independent given some set of variables Z if, for any assignment of values, t, z, and x to the variables T, Z, and X respectively, is $P(T=t \mid X=x, Z=z) = P(T=t \mid Z=z)$. That is, X gives us no information about T beyond what is already in Z. Thus, we can calculate that $D(P(T \mid X,Z) \| P(T \mid Z)) = 0$. If this is related with mutual information concept, the mutual information between T and X given Z is:

$$\begin{aligned} I(T;X \mid Z) &= H(T \mid Z) - H(T \mid X,Z) \\ &= -\sum_{t,z} P(T \mid Z) \log P(T \mid Z) + \sum_{t,z,x} P(T \mid X,Z) \log P(T \mid X,Z) \\ &= -\sum_{t,z} P(T \mid Z) \log P(T \mid Z) + \sum_{t,z} P(T \mid Z) \log P(T \mid Z) \\ &= 0 \end{aligned}$$

The *Markov blanket* of a variable of interest T, denoted as *MB(T)*, is the minimum conditioning set that makes all other features independent for T [6]. Given this property, knowledge of only the features of the *MB(T)* is enough to determine the probability distribution of T and the values of all other features become unnecessary. Therefore, the variables in the *MB(T)* are adequate for optimal classification. Then, the Markov blanket of a variable interest T, *MB(T)* can be represented as a minimal set for which $I(X;T \mid MB(T)) = 0, \forall X \in \{V - T - MB(T)\}$.

3. TPMB ALGORITHM

The algorithm to select the minimum set of features that represents the Markov blanket of the target class is NP hard. To cope with this scalability problem, we propose the TPMB algorithm which is a heuristic algorithm to select features based on Markov blanket concept. This algorithm consists of two phases, filtering and approximating.

3.1 Filtering Phase

The scalability problem occurs because there are too many features to handle. Thus, in this phase we eliminate the features that do not posses an "obvious" relation with the target class. There are many ways to go about eliminating redundant features. One of the ways is to measure the correlation of each feature. In this chapter we are going to discuss about how we can filter redundant features by measuring the correlation between each feature and the target class.

3.1.1 Linear Correlation

Suppose we have two variables X and Y, with means $\bar{X}$ and $\bar{X}$ respectively and standard deviations S_x and S_y respectively, linear correlation (also known as Pearson's Correlation) is computed as:

$$r = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{(n-1) S_x S_y}$$

The meaning of linear correlation (*Pearson's Correlation*) can be described in this way. Suppose that an X value is above average, and that the associated Y value was also above average. Then the product

$(X_i - \bar{X})(Yi - \bar{Y})$ would be the product of two positive numbers which would be positive. The result will also be positive if the X value and the Y value are both below average. Therefore, a positive linear correlation is evidence of a general tendency that large values of X are associated with large values of Y and small values of X are associated with small values of Y. On the other hand, a negative linear correlation is evidence of a general tendency that large values of X are associated with small values of Y and small values of X are associated with large values of Y. A correlation of 0 means there is no linear relationship between two variables. The correlation coefficient is always between -1 and +1. The closer the correlation is to +1/-1, the closer the variables to a perfect linear relationship.

3.1.2 Non Linear Correlation

Linear correlation measures may not be able to capture correlations that are not linear in nature. To overcome this shortcoming, a correlation measure based on the information-theoretical concept of entropy is adopted.

Suppose that there are two variables X with entropy $H(X)$ and Y with entropy $H(Y)$. The mutual information between two variables is denoted as $I(X,Y)$. The non-linear correlation or *symmetrical uncertainty* (SU) between two variables is defined as follows [5].

$$SU(X,Y) = 2\left[\frac{I(X;Y)}{H(X)+H(Y)}\right]$$

Information gain $I(X;Y)$ is bias towards features with more values, therefore it is normalized such that symmetrical uncertainty is within range [0,1]. Value 1 indicates that the knowledge of the value of either X or Y can completely predicts the value of the other variable and value 0 indicates that X and Y are independent [9].

3.1.3 Filtering Algorithm

Despite several benefits of choosing linear correlation as a feature goodness measure for classification, it is not safe to always assume linear correlation between features in the real world [5]. Therefore, in our approach of filtering redundant data we use non-linear approach using symmetrical uncertainty.

The filtering algorithm that we use is *Simple-Comparison (SC)* filtering (Figure 3). In this method, we measure the correlation between each feature and the target class. We also define a threshold value (ε_f) and select features which non-linear correlation to the target class is more than the threshold.

Input: $S(f_1, f_2, ..., f_N, C)$ //a training dataset
ε_f //a predefined threshold
Output: S_f //a filtered subset

1 **begin**
2 for i = 1 to N do begin
3 Calculate $SU_{i,c}$ for f_i;
4 if ($SU_{i,c} \geq \varepsilon_f$)
5 append f_i to S'_{list};
6 end for;
7 $S_f = S_{list}$;
8 **end;**

Figure 3: Simple Comparison Filtering Algorithm

3.2 Approximating Phase

After the filtering phase, the MB candidate list is further processed to approximate the Markov blanket. In approximating Markov blanket, our algorithm uses the concept of cross entropy, $\partial(P_a \| P_b) = D(P_a \| P_b)$, P_a *is the probability of target class given all features and* P_b *is the probability of target class when one feature is removed.*

Let us assume $F = \{f_1, ..., f_n\}$, $P_a = P(C|F)$ and $P_b = P(C|F - f_i)$. If $\partial(P_a \| P_b) = 0$ then we can say that C is conditionally independent of f_i and we can remove f_i from F. However, sometimes it is difficult to find such a feature that is conditionally independent to the target class given the knowledge of other features. For that reason, in our approximating algorithm (Figure 4), we introduce a predefined threshold, ε_a , such that we remove f_i if $\partial(P_a | P_b) < \varepsilon_a$.

Input: $S(f_1, f_2, ..., f_N, C)$ // training dataset
$SU_{list}(SU_{1,c}, ..., SU_{N,c})$ // correlation
ε_a // predefined threshold
Output: MB_{list} //Estimated Markov blanket

1 **begin**
2 Order S in decreasing order of $S_{i,c}$ values;
3 MB_{list} = NULL;
4 for (i=N; i>=2; i--) do begin
5 $P_a = P(C|\{S\})$;
6 $P_b = P(C|\{S\} - f_i)$;
7 If ($\partial(P_a \| P_b) < \varepsilon_a$)then
8 Remove f_i from S;
9 End if;

```
10   End for;
11   MB_list = S - C;  //training dataset minus Class
12   Return MB_list;
13  end;
```

Figure 4: Approximation Algorithm

3.3 Complexity of TPMB Algorithm

Suppose that the original dataset contains n data instances and m features. In the filtering phase of algorithm, if SC filtering algorithm is used, the filtering phase algorithmic complexity will be $O(m^2n)$. In the approximating phase, assume that the number of selected features passed from the filtering phase is p, the algorithmic complexity of approximation phase is $O(p^2n)$. It is clear that if we can reduce a large number features in first phase then we can have a lower algorithmic complexity in second phase.

4. RESULTS

We conducted our experiments using the microarray gene expression datasets AMLALL from Golub et al [3] and colon tumor from Alon et al [4]. The AMLALL dataset contains 38 samples and 7129 genes while colon tumor dataset 62 samples and 2000 genes.

We perform 10-fold Cross Validation test on the original dataset. After that, we run TPMB algorithm with different combination of ε_f and ε_a. Next, we list down the highest ranked genes of AMLALL and Leukemia dataset based on TPMB algorithm as shown in table 1 and table 2.

AML-ALL Features	Gene Accession No	Description
4847	X95735_at	Zyxin
4499	X70297_at	CHRNA7 Cholinergic receptor, nicotinic, alpha polypeptide 7
2233	M77142_at	NUCLEOLYSIN TIA-1
1926	M31166_at	PTX3 Pentaxin-related gene, rapidly induced by IL-1 beta

Table 1: Top ranked features of AMLALL dataset which are selected by TPMB algorithm

Colon Tumor Features	Gene No.	Seq	Gene Description
1671	M26383	gene	Human monocyte-derived neutrophil-activating
		gene	Human monocyte-derived
1293	H23544	3' UTR	GTP-BINDING NUCLEAR PROTEIN RAN (Homo sapiens)
493	R87126	3' UTR	MYOSIN HEAVY CHAIN, NONMUSCLE (Gallus gallus)
513	M22382	gene	MITOCHONDRIAL MATRIX PROTEIN P1 PRECURSOR (HUMAN)

Table 2: Top ranked features of Colon Tumor dataset selected by TPMB algorithm

As the classifiers, in our experiment we used C4.5 and Least Square Support Vector Machine [7]. Table 3 shows the accuracies of the classifiers trained using dataset with full number of features and with selected number of features presented in Table 1 and Table 2.

As shown in table 3, TPMB algorithm is able to reduce the number of features significantly for both AMLALL and Colon Tumor dataset. In most cases feature reduction leads to higher accuracy except for AMLALL dataset, when LSSVM is used, there is a decrease in accuracy. However, the accuracy is still high and the decrease is not significant. Compared to the ability of TPMB algorithm to reduce the number of features, the gain out-weights the slight drop in performance.

Data	ORIGINAL			TPMB ALGORITHM		
	#features	LSSVM with linear kernel	C4.5	#features	LSSVM with linear kernel	C4.5
AMLALL	7129	96.67	84.21	4	93.33	89.47
Colon Tumor	2000	78	82.26	4	80	82.26

Table 3: Classifiers accuracy when trained with full set features and with features selected by TPMB algorithm full number of features and with selected number of features

5. CONLUSIONS

From the results of the TPMB algorithm, top ranked features have been selected. The ratios between selected and original features are very significant, i.e. 4:7129 and 4:2000. It has been shown that classifiers trained using the selected features can produce rather good prediction accuracy.

The combination of the thresholds affects timing performance and the number of selected features. The perfect combination that gives the best performance can be found by multiple experiments. Further improvements can be done to TPMB algorithm especially about how to adjust

the thresholds in first and second phases theoretically than heuristically.

REFERENCES

[1] L. Yu and H. Liu, Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution, In Proceedings of the Twentieth International Conference on Machine Learning, pp. 856-823, 2003.

[2] D. Koller and M. Sahami, Toward Optimum Feature Selection, presented at In Proceedings of the 13th International Conference on Machine Learning, 1996.

[3] T. Golub, D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. Mesirov, H. Coller, M. Loh, J. Downing, M. Caligiuri, C. Bloomfield, and E. S. Lander, Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring, Science pp. 286(5439):531-537, 1999.

[4] U.Alon, N.Barkai, S. Ybarra, D. Mack, D.A.Notterman, K. Gish, and A. J. Levine, Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, Proceedings of National Academy of Science, vol. 96, pp. 6745-6750, 1999.

[5] L. Yu and H. Liu, Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution, In Proceedings of the Twentieth International Conference on Machine Learning, pp. 856-823, 2003.

[6] C. F. Aliferis, I. Tsamardinos, and A. Statnikov, HITON: A Novel Markov Blanket Algorithm for Optimal Variable Selection, Proceedings of the 2003 American Medical Informatics Association (AMIA) Annual Symposium, pp. 21–25, 2003.

[7] Y. Zhao, Efficient Model and Feature Selection for SVM in Biomedical Data Analysis, in School of Computer Engineering. Singapore: Nanyang Technological University, 2004, pp. 99.