Fusion of Feature Selection Methods for Pairwise Scoring SVM

Man-Wai Mak [a] and Sun-Yuan Kung [b]

[a] *Dept. of Electronic and Information Engineering*
The Hong Kong Polytechnic University, Hong Kong SAR

[b] *Dept. of Electrical Engineering*
Princeton University, USA

Abstract

It has been recently discovered that stacking the pairwise comparison scores between an unknown patterns and a set of known patterns can result in feature vectors with nice discriminative properties for classification. However, such technique can be hampered by the curse of dimensionality because the vectors size is equal to the training set size. To overcome this problem, this paper investigates various filter and wrapper feature selection techniques for reducing the feature dimension of pairwise scoring matrices and argues that these two types of selection techniques are complementary to each other. A fusion technique is then proposed to combine the ranking criteria of filter and wrapper methods at algorithmic level. Evaluations on a subcellular localization benchmark demonstrate that feature sets selected by the fusion methods outperform those selected by the individual methods alone.

Key words: Feature selection, filter, wrapper, curse of dimensionality, protein sequences, subcellular localization, kernel methods, support vector machines.

1 Introduction

In computational biology, the subcellular location and structural family of a protein provide important information about its biochemical functions. However, experimental analysis of proteins is time-consuming and cannot be performed on genome-wide scales. Therefore, a reliable and efficient method is essential for automating the prediction of proteins' subcellular locations and

Email address: enmwmak@polyu.edu.hk (Man-Wai Mak).
URL: http://www.eie.polyu.edu.hk/∼mwmak/mypage.htm (Man-Wai Mak).

the classification of protein sequences into functional and structural families. One of the successful techniques is to compare the unknown sequences against some known sequences. The idea is based on the notion that similarity (homology) in sequences, to a certain extent, also means closeness in function and structure.

The comparison of two sequences are often hampered by the fact that the two sequences often have different lengths whether or not they belong to the same family. To overcome this problem, pairwise comparison between a sequence and a set of known sequences has been a popular scheme for creating fixed-size feature vectors from variable-length sequences [1–3]. Although this pairwise approach can usually create feature vectors with good discriminative properties, it also has its own limitation. The main problem is that the feature dimension is the same as the number of training patterns. This leads to the curse of dimensionality, because the training set size could be very large. In fact, for the applications addressed in this paper, they are in the range of several thousands. High dimensionality in feature spaces increases the computational cost in both the learning phase and prediction phase. In the prediction phase, the more features used the more the computation required and the lower the retrieval speed. Fortunately, the prediction time is often linearly proportional to the number of features selected. Unfortunately, in the learning phase, the computational demand may grow exponentially with the number of features. Because a large number of sequences are being added to sequence databases in a daily basis, it is imperative to reduce the complexity of the pairwise scoring approach.

An obvious solution to the curse of dimensionality problem is to reduce the feature size and yet retaining the most important information critical for the classification of the training patterns. Research in protein homology detection has found that just over 10% of proteins' profiles contribute 90% of the total score for positive training sequences [4], suggesting that some features are more relevant to the classification task than the others. The feature size can be reduced by either finding principle subspace or weeding out those less significant features. The latter approach is known as feature selection and is the focus of this paper.

Feature selection often depends on a joint consideration of two conflicting aspects: computational cost and achievable performance. The best choice usually represents an optimal tradeoff between these factors. The challenge lies in how to reach a useful dimension reduction while conceding minimum sacrifice on accuracy or other desired performance.

Feature selection methods can be divided into two categories: filter and wrapper. In the filter method, feature selection and classifier design are separated in that a subset of features is firstly selected and then classifiers are trained

based on the selected features. For example, the discriminative power of features are ranked according to their Fisher discriminant ratio [5] such that features with high ranks are retained. The wrapper approach [6–9], on the other hand, uses classification accuracies or criteria derived from the classifier to rank the discriminative power of all (or part) of the possible feature subsets and select the subset that produces the best performance. Therefore, the selected features are bound to the type of classifier that was used in the feature selection process.

Both filter and wrapper methods have their own merits and limitations. For example, although filter methods are simple and fast, the selected features may be highly correlated because the feature set may contain many highly discriminative features but with almost identical characteristics. Moreover, most ranking criteria do not take the combined effect of features into account. This paper proposes fusing the filter and wrapper methods at the algorithmic level to leverage the advantages of both methods. Evaluations on a subcellular localization benchmark demonstrate that the proposed method outperforms the state-of-the-art filter and wrapper methods, particularly when the number of allowed features is very small.

The paper is organized as follows. Section 2 outlines the profile alignment algorithms and the pairwise scoring kernels for protein subcellular localization. Section 3 describes two important feature selection strategies—filter and wrapper—and proposes a fusion technique to combine both strategies. The fusion method is then compared with some existing selection methods through a subcellular localization task in Section 4, and conclusions are drawn in Section 5.

2 Pairwise Scoring Kernels for Subcellular Localization

Denote $\mathcal{D} = \{S^{(1)}, \ldots, S^{(T)}\}$ as a training set containing T protein sequences. To efficiently produce the profiles of a protein sequence (called query sequence), the sequence is used as a seed to search and align homologous sequences from protein databases such as Swissprot [10] using the PSI-BLAST program [11]. Let us denote the operation of PSI-BLAST search given the query sequence $S^{(i)}$ as

$$\phi^{(i)} \equiv \phi(S^{(i)}) : S^{(i)} \longrightarrow \{\mathbf{P}^{(i)}, \mathbf{Q}^{(i)}\},$$

where $\mathbf{P}^{(i)}$ and $\mathbf{Q}^{(i)}$ are the position-specific scoring matrix (PSSM) and position-specific frequency matrix (PSFM) of $S^{(i)}$, respectively. The homolog information pertaining to the aligned sequences is represented by these two matrices (also called profiles), which have 20 rows and L columns, where L

is the number of amino acids in the query sequence. Because these matrices contain rich information about the remote homolog of the query sequence, highly discriminative features can be derived from them for the prediction of subcellular locations and protein functions.

Given the profiles of two sequences $S^{(i)}$ and $S^{(j)}$, we can apply the Smith-Waterman algorithm [12] and its affine gap extension [13] to align $\mathbf{P}^{(i)}$, $\mathbf{Q}^{(i)}$, $\mathbf{P}^{(j)}$, and $\mathbf{Q}^{(j)}$ to obtain the normalized profile-alignment score $\zeta(\phi^{(i)}, \phi^{(j)})$, as detailed in the appendix. The scores $\{\zeta(\phi^{(i)}, \phi^{(j)})\}_{i,j=1}^{T}$ constitute a symmetric matrix $\mathbf{Z}$ whose columns can be considered as T-dimensional vectors:

$$\boldsymbol{\zeta}^{(j)} = [\zeta(\phi^{(1)}, \phi^{(j)}) \quad \dots \quad \zeta(\phi^{(T)}, \phi^{(j)})]^{\mathsf{T}} \quad j = 1, \dots, T. \tag{1}$$

This means that there are T feature vectors with dimension equal to the training set size. The T T-dimensional column vectors can be used to train M one-vs-rest SVMs for an M-class protein prediction problem:

$$f_m(S) = \sum_{j \in \mathcal{S}_m} y_{m,j} \alpha_{m,j} K(\phi(S), \phi(S^{(j)})) + b_m \qquad m = 1, \dots, M, \tag{2}$$

where S is an unknown sequence, $y_{m,j} \in \{+1, -1\}$, $\mathcal{S}_m$ contains the indexes of support vectors, $\alpha_{m,j}$ are Lagrange multipliers, and

$$K(\phi(S)), \phi(S^{(j)}) = g(\boldsymbol{\zeta}, \boldsymbol{\zeta}^{(j)})$$

is a kernel function. When $K(\cdot)$ is a linear kernel, we have

$$g(\boldsymbol{\zeta}, \boldsymbol{\zeta}^{(j)}) =< \boldsymbol{\zeta}, \boldsymbol{\zeta}^{(j)} >= \sum_{t=1}^{T} \zeta(\phi(S^{(t)}), \phi(S)) \zeta(\phi(S^{(t)}), \phi(S^{(j)})). \tag{3}$$

Alternatively, $K(\cdot)$ can be a non-linear kernel such as RBF:

$$g(\boldsymbol{\zeta}, \boldsymbol{\zeta}^{(j)}) = g(\|\boldsymbol{\zeta} - \boldsymbol{\zeta}^{(j)}\|) = \exp\left\{ -\frac{1}{\sigma^2} \sum_{t=1}^{T} \left[\zeta(\phi(S^{(t)}), \phi(S)) - \zeta(\phi(S^{(t)}), \phi(S^{(j)})) \right]^2 \right\}. \tag{4}$$

During prediction, the class of an unknown sequence S can be obtained by

$$y(S) = \arg \max_{m=1}^{M} f_m(S).$$

3 Feature Selection Methods

Feature selection methods can be divided into two main categories: filter and wrapper. These categories differ in their timing in determining the classifiers.

In other words, the main difference lies in whether or not they are classifier dependent.

(1) **Filter**: The selection method/criterion is self-determined, i.e., it is independent of the classifier.
(2) **Wrapper**: The classification method is predetermined.

While the filter method is computationally efficient, it fails to consider the critical inter-feature redundancy. The wrapper approach, on the other hand, can fully rectify this problem, although its closed-loop process is very computational demanding.

It is natural to combine/consider both methods in order to reach an optimal strategy. To this end, two more comprehensive and advanced approaches have been proposed:

(1) **Fusion Approach**: The criterion functions of and the features obtained from the filter and wrapper methods can be combined via a fusion scheme.
(2) **Embedded Approach**: The classifier is jointly determined with the selection scheme [14,15]; in other words, the two components are interdependent.

This section will focus on the filter and wrapper methods. A fusion scheme that combines these two methods at the algorithmic level will also be discussed.

3.1 Filter Methods

In the filter method, feature selection and classifier design are separated in that a subset of features is firstly selected and then classifiers are trained based on the selected features. From the structural dependence perspective, while the classifier has to depend on the selection strategy, the features are selected with no regard to what classifier will be adopted.

Several ranking criteria for filter methods have been proposed in the literature. They include (1) correlation coefficients [5], (2) Fisher discriminant ratio [16], (3) symmetric divergence [17], (4) information theoretic criteria [18], and (5) feature weighting (ReliefF) [19]. The first three consider the features individually in the selection process, whereas the last two take the interaction of features into account.

3.1.1 Correlation Coefficients

In this approach, the discriminative power of features is evaluated independently based on their correlation coefficients (CC) [5]:

$$\mathrm{CC}(j) = \frac{\mu_j^+ - \mu_j^-}{\sigma_j^+ + \sigma_j^-}, \tag{5}$$

where μ_j^+, μ_j^-, σ_j^+, and σ_j^- represent the class-conditional means and standard derivations of the j-th feature, respectively. Furey et al. [20] proposed to use $|\mathrm{CC}(j)|$ as ranking criterion. Features with high $\mathrm{CC}(j)$ or $|\mathrm{CC}(j)|$ are selected for classification. The method is intuitive appealing because high CC means that the corresponding features produce maximum separation between the positive and negative classes.

A similar ranking criterion is based on the t-statistics [21]

$$z_j = \frac{\mu_j^+ - \mu_j^-}{\sqrt{\frac{(s_j^+)^2}{N^+} + \frac{(s_j^-)^2}{N^-}}}, \tag{6}$$

where

$$(s_j^+)^2 = \frac{\sum_{k \in \mathcal{C}^+} (x_{kj} - \mu_j^+)^2}{N^+ - 1} \quad \text{and} \quad (s_j^-)^2 = \frac{\sum_{k \in \mathcal{C}^-} (x_{kj} - \mu_j^-)^2}{N^- - 1} \tag{7}$$

where N^+ and N^- are the numbers of training samples in the positive class $\mathcal{C}^+$ and negative class $\mathcal{C}^-$, respectively.

3.1.2 Fisher Discriminant Ratio

A slight variant of correlation coefficients is the Fisher discriminant ratio (FDR) [16]:

$$\mathrm{FDR}(j) = \frac{\left(\mu_j^+ - \mu_j^-\right)^2}{(\sigma_j^+)^2 + (\sigma_j^-)^2}. \tag{8}$$

Note that FDR and CC are very similar; the only difference is that FDR ignores the sign of the difference in means, whereas CC takes the sign into consideration. This property of CC turns out to be important for feature selection, as demonstrated in our experimental results in Section 4.

3.1.3 Symmetric Divergence

By assuming the distributions of the positive and negative classes as Gaussians, Mak and Kung [17] proposed ranking the features according to the

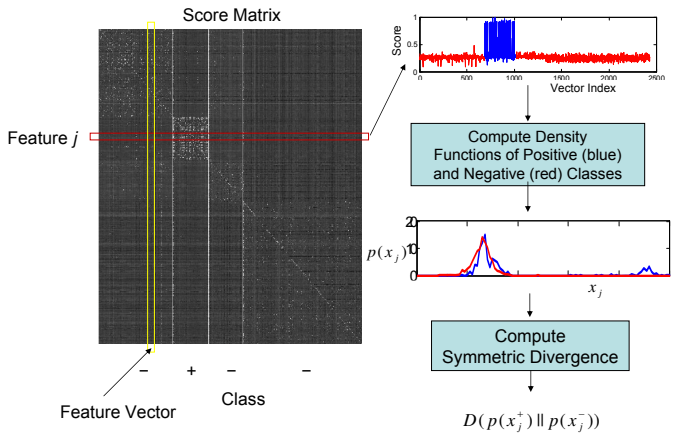


Fig. 1. Computation of symmetric divergence (Eq. 9) in a subcellular localization task.

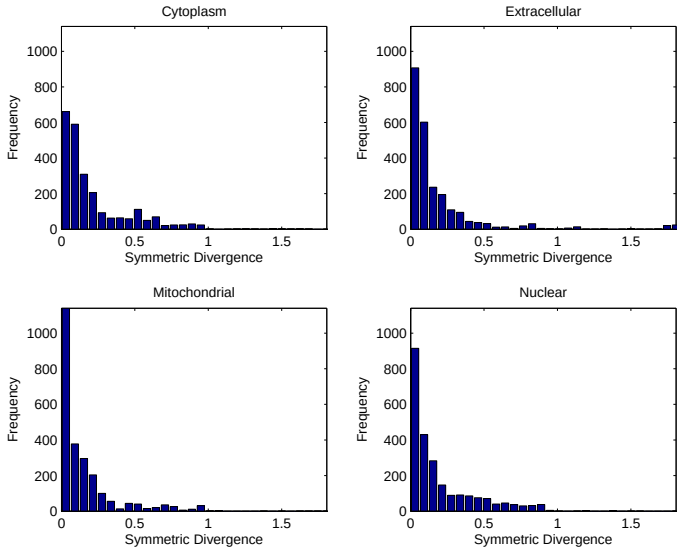


Fig. 2. Histograms of symmetric divergences in a subcellular localization task.

symmetric divergence between the positive and negative class distributions:

$$
\begin{aligned}
D\left(p(x_j^+)\|p(x_j^-)\right) &= E\left\{\log\frac{p(x_j^+)}{p(x_j^-)}\middle|\mathcal{C}^+\right\} + E\left\{\log\frac{p(x_j^-)}{p(x_j^+)}\middle|\mathcal{C}^-\right\} \\
&= \frac{1}{2}\left(\frac{(\sigma_j^+)^2}{(\sigma_j^-)^2} + \frac{(\sigma_j^-)^2}{(\sigma_j^+)^2}\right) - 1 + \frac{1}{2}\left(\frac{\left(\mu_j^+ - \mu_j^-\right)^2}{(\sigma_j^+)^2 + (\sigma_j^-)^2}\right) \qquad (9)
\end{aligned}
$$

where $p(x_j^+)$ and $p(x_j^-)$ represent the density functions of the positive and negative classes, respectively. Figure 1 illustrates the procedure of computing the symmetric divergence of feature j and Figure 2 shows the histograms of the symmetric divergences. Apparently, only a small fraction of the features have large symmetric divergences, which means that only a few feature are relevant for classification.

3.1.4 Limitations of Filter Methods

Selecting features based on the filter methods suffers from several serious drawbacks. First, these methods require users to set a cutoff point in the ranking scores under which features are deemed to be irrelevant for classification. However, the optimal number of features (i.e., the best cutoff point) is usually unknown. (To overcome this limitation, we have proposed the vector-index-adaptive SVM [9] in which the optimal number of features is determined automatically by the SVM training algorithm.) Second, the selected features may be highly correlated because the feature set may contain many highly discriminative features but with almost identical characteristics. This means that some of these features can be removed without affecting the classification accuracy. Third, the ranking criteria do not take the combined effect of features into account. For example, a low-ranked feature may become very useful for classification when combined with another low-ranked feature. To overcome these limitations, researchers have proposed using the performance of classifiers to guide the feature selection progress, which is to be discussed next.

3.2 Wrapper Methods

Because the ultimate goal of feature selection is to increase classification accuracy, it is intuitive to choose a particular classification method and use its parameters or its performance on training data to guide the feature selection process (see Figure 3). Typically, this is done by selecting a subset of features and evaluating its performance on the chosen classifier, and the process is repeated until the best performing subset is obtained. Methods based on this approach are known as wrappers in the literature [22].

Most wrappers adopt a closed-loop approach in that each subset evaluation requires training a classifier, [1] making the selection process computationally expensive. On the other hand, the computational cost of the open-loop approach (such as filters) will not be as high as that of the closed-loop one, but the former may incur degradation in classification accuracy. The close-loop approach is prevailing because most applications put more emphasis on accuracy than on computational saving.

A number of wrapper approaches have been proposed for bioinformatics data mining, including the recursive feature elimination (RFE) [7] and recursive SVM [8].

[1] It is natural to use classification accuracy as the goodness measurement of feature subsets.

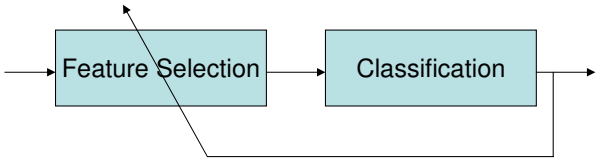


Fig. 3. The architecture of wrappers. The classifier acts as a master guiding the feature selection process.

3.2.1 SVM-RFE

Guyon et al. [7] proposed a backward elimination algorithm, namely SVM-RFE, that ranks features based on the weights of a linear SVM. The algorithm begins with using the full-feature training vectors $\mathbf{x}_i \in \Re^T$ to train a linear SVM. Features are then ranked by sorting the square of the SVM's weights $\{w_j^2\}_{j=1}^T$ in descending order, where the weight vector is given by

$$\mathbf{w} = \sum_{i \in \mathrm{SV}} \alpha_i y_i \mathbf{x}_i, \qquad \mathbf{w} \in \Re^T. \tag{10}$$

A subset of features corresponding to the end of the sorted list (i.e., those with small w_j^2) is then removed. [2] The remaining features are used to construct a new set of training vectors $\mathbf{x}_i' \in \Re^{T'}$, where $T' < T$. These vectors are then used to train another linear SVM and the process is repeated until all features have been eliminated. At the end of the iterative process, a ranked list of features is produced.

The idea of SVM-RFE can be intuitively explained by considering a two-feature case as shown in Figure 4. The figure shows two possible ways of separating the two classes of data. Boundary B (with weight vector $\mathbf{w}'$) is undesirable because of the small margin. On the other hand, Boundary A (with weight vector $\mathbf{w}$) is more desirable because of the large margin. In fact, a linear SVM will use Boundary A to classify the data. Notice that the weight vector $\mathbf{w} = [w_1 \;\; w_2]^\mathsf{T}$ in Figure 4 has the property $w_2^2 > w_1^2$, which suggests that x_2 is a more discriminative feature.

3.2.2 Recursive SVM

The recursive SVM (R-SVM) [8] resembles the operation of SVM-RFE but with one important difference. Instead of sorting the square of SVM's weights, R-SVM sorts the product of w_j and the mean sample difference of the two classes in the j-th dimension:

$$s_j = w_j(\mu_j^+ - \mu_j^-) \qquad j = 1, \ldots, T \tag{11}$$

[2] Intuitively, a small weight w_j means that the j-th axis in the feature space is irrelevant to classification and can be removed without affecting performance.

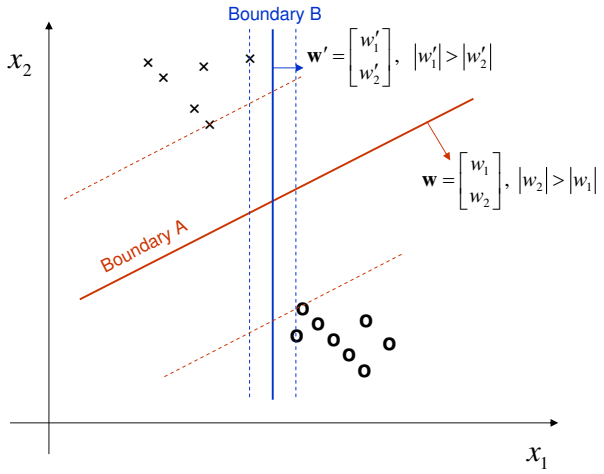


Fig. 4. Selecting one out of two features by SVM-RFE. A linear SVM will use Boundary A for classification and therefore feature x_2 will be selected by ranking the square of SVM weights $\{w_1^2, w_2^2\}$ in descending order.

where T is the dimension of the full feature vectors $\mathbf{x}$'s. Using the notation in Section 3.1, we may express the sum of s_j as

$$\begin{aligned} s &= \sum_{j=1}^{T} s_j = \sum_{j=1}^{T} w_j(\mu_j^+ - \mu_j^-) \\ &= \mathbf{w} \cdot (\boldsymbol{\mu}^+ - \boldsymbol{\mu}^-) \\ &= \frac{1}{N^+} \sum_{\mathbf{x} \in \mathcal{C}^+} \mathbf{w} \cdot \mathbf{x} - \frac{1}{N^-} \sum_{\mathbf{x} \in \mathcal{C}^-} \mathbf{w} \cdot \mathbf{x} \\ &= \frac{1}{N^+} \sum_{\mathbf{x} \in \mathcal{C}^+} f(\mathbf{x}) - \frac{1}{N^-} \sum_{\mathbf{x} \in \mathcal{C}^-} f(\mathbf{x}) \end{aligned} \tag{12}$$

where $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$ is the SVM output. Intuitively, maximizing s has the effect of maximizing the separation between the samples in two classes. If we sort $\{s_j\}_{j=1}^{T}$ in descending order and denote the resulting indexes as $\mathcal{I} = \{i_1, i_2, \ldots, i_T\}$, we may still obtain a score very close to s by summing s_{i_k} from $k = 1$ up to $k = i_{T'}$ where $T' < T$, i.e.,

$$\sum_{k=1}^{T'} s_{i_k} \approx \sum_{j=1}^{T} s_j = s.$$

Therefore, selecting the first T' features from the feature list $\mathcal{I}$ can maximize the separation between positive and negative samples. It was found in [8] that R-SVM is robust to noise and outliers in gene expression data.

3.3 Fusion of Filter and Wrapper

Because the filter and wrapper methods correspond to different selection criteria, these methods may be combined to enhance the relevance of the selected features. Such a prospect is further encouraged by the fact that the Fisher criterion (in the filter method) and the SVM (in the wrapper method) are complementary in nature. The difference being that the SVM places emphasis on support vectors near the decision boundary and, in contrast, the (2nd-order based) FDR is based on how separable are the positive and negative centroids when projected to that axis. In other words, the FDR depends on the distribution of all training vectors, which is more appropriate for data that follow a normal distribution. Overall, the FDR has less risk of overtraining, while SVM has a slightly better chance of delivering better accuracy.

One possible way to combine filters and wrappers is to fuse their criterion functions at algorithmic level. For example, we may combine FDR and SVM-RFE as follows:

$$\text{FDR + SVM-RFM: } s_j = \alpha \log(\mu_j^+ - \mu_j^-)^2 + (1-\alpha)\log w_j^2 \qquad (13)$$

where $0 \leq \alpha \leq 1$ controls the contribution of the selection methods. Interestingly, R-SVM can be considered as a fusion of filter and wrapper methods. This is because the ranking criterion in Eq. 11 involves both the SVM weights and difference in means. More specifically, R-SVM fuses correlation coefficients (Eq. 5 where variances in both classes are equal) with a special case of SVM-RFE (Eq. 10 where w_j instead of w_j^2 are sorted).

To compare the performance of different wrappers and to demonstrate the merit of fusing filters and wrappers, we applied these approaches to a subcellular localization task, which is to be described next.

4 Experiments and Results

4.1 Dataset and Evaluation Criterion

Reinhardt and Hubbard's eukaryotic protein dataset [23], which contains 2427 amino acid sequences, was employed to test the performance of different feature selection methods. The sequences in this dataset were extracted from SWISSPORT 33.0 and their subcellular locations (684 cytoplasm, 325 extracellular, 321 mitochondrial, and 1097 nuclear proteins) have been annotated.

We used 5-fold cross validation for performance evaluation, i.e., the original

dataset was randomly divided into 5 subsets. Each subset was singled out in turn as a testing set, and the remaining ones were merged as the training set. The process was iterated 5 times until every subset has been used for testing.

4.2 Comparing Different Filters

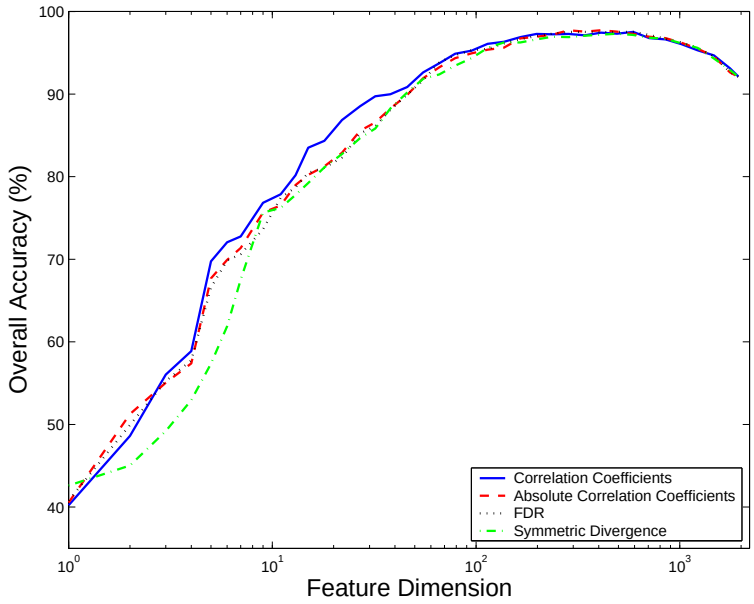
Note that ranking methods based on cross correlation, t-statistics, FDR, and symmetric divergence share a common property: Features with small variances but large difference in class means will be ranked high. However, there are also important differences. For example, Eq. 8 and Eq. 9 differ in the additional term that depends only of the ratio of variances. Therefore, features with high variance ratio between positive and negative classes will be ranked high under the symmetric divergence criterion.

To compare the capability of different filter methods, we applied cross correlation (Eq. 5), absolute cross correlation, FDR (Eq 8), and symmetric divergence (Eq. 9) to select features for classifying the subcellular locations of proteins in Reinhardt and Hubbard's dataset. We applied pairwise profile alignment to create a score matrix for classification by RBF-SVMs (see Section 2 and [3]).

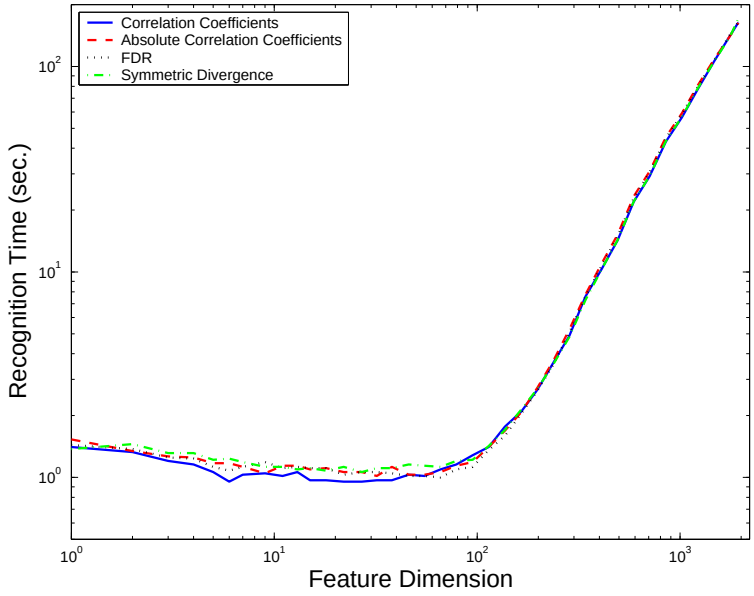
Figure 5 shows the accuracy and recognition time achieved by various filter methods when the number of selected features is progressively increases from 1 to the full size. Evidently, the accuracy increases rapidly at small feature size and becomes saturated at around 200 (the accuracy even drops slightly because of the curse of dimensionality), suggesting only one-tenth of the features are useful for classification. The results show that the performance of cross correlation (CC) is better than that of absolute cross correlation (|CC|) and FDR, suggesting that the sign in CC is important for feature selection. Results also show that for all filter methods investigated, the recognition time can be reduced by 100-fold (from 100 seconds at full feature size to 1 second at 100 features) without significant reduction in accuracy.

4.3 Fusion of Filter and Wrapper

Recall from Section 3.3 that the criteria used by filters (e.g., FDR) and wrappers (e.g., SVM-RFE) may complement each other. Here, we compared the performance of (1) filters methods, (2) wrappers methods, and (3) their fusion on the subcellular localization task. Again, the selected features were used to train four 1-vs-rest RBF-SVMs to perform the prediction. Also compared is the recognition time required by the RBF-SVMs using the features selected by these methods.



(a)



(b)

Fig. 5. (a) Accuracy and (b) recognition time based on features selected by cross correlation (CC), absolute cross correlation (|CC|), Fisher discriminant ratio (FDR), and symmetric divergence.

Figure 6 shows the performance of FDR, SVM-RFE, R-SVM, and FDR + SVM-RFE (Eq. 13). The fusion weight α in Eq. 13 was set to 0.3 in all cases. Evidently, R-SVM performs poorly while the fusion of FDR and SVM-RFE outperforms all other methods, particularly at low feature dimension. The fusion result of FDR and SVM-RFE also agrees with the notion of consistent fusion advocated in [24], i.e., fusion results are at least as good as the ones without fusion. The contrast in performance between the FDR + SVM-RFE

and R-SVM suggests that fusion strategies are instrumental in selecting the right features and that a wrong fusion strategy could be detrimental to recognition performance.

Similar to the filter case, about 100-fold reduction in recognition time can be achieved without scarifying prediction accuracy significantly. Although fast recognition may not be critical for bioinformatics because sequence classification can be done off-line, it is critically important for other applications such as biometrics where real-time recognition is required. Because pairwise scoring is a general technique for converting variable-length sequences into fix-size vectors, the feature reduction techniques proposed in this paper is certainly not limited to protein sequence classification.

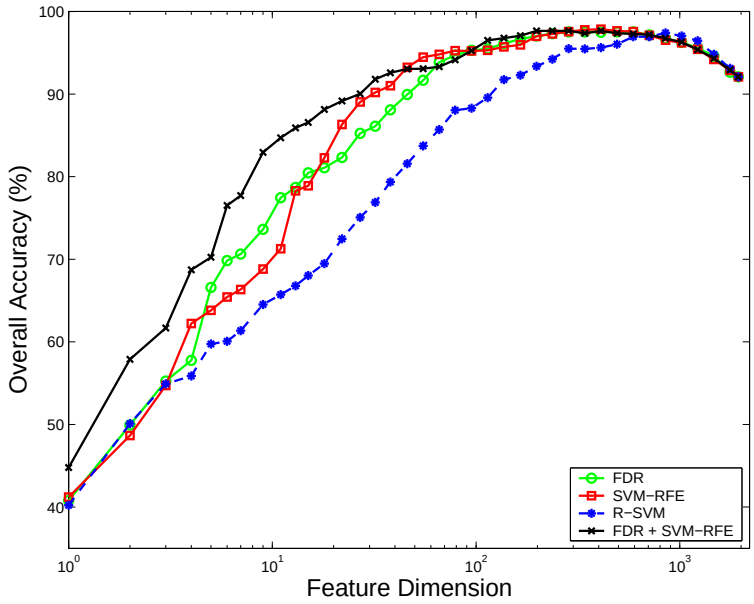
5 Concluding Remarks

We have discussed two popular types of feature selection methods—filters and wrappers—for solving the curse of dimensionality problems commonly encounter in pairwise scoring techniques. The performance of various types of filters and wrappers has been evaluated and compared under a subcellular localization benchmark. We have also proposed fusing the FDR and SVM-RFE by combining their criterion functions and found that the fusion performance is superior to either of these two methods for almost all feature dimension. While this paper has offered a promising fusion approach for feature selection, there are still more variants of fusion techniques need to be pursued. This appears to be a promising direction worth further investigation.

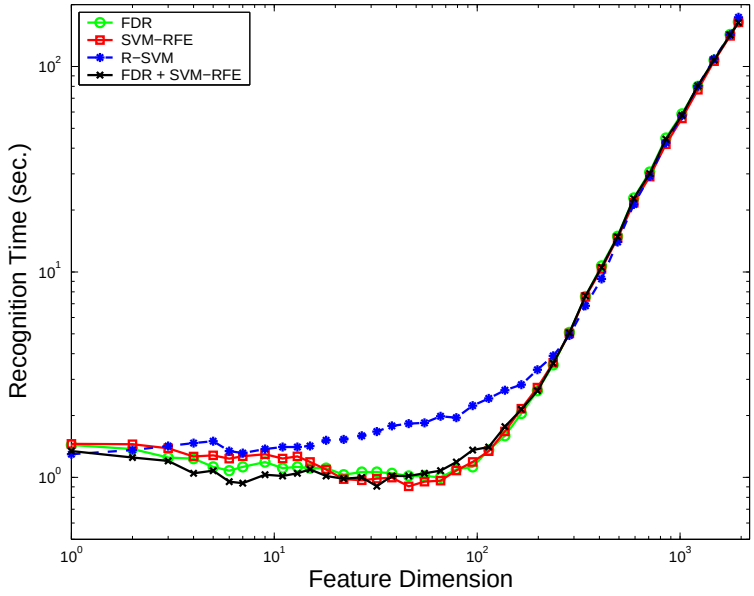
Appendix

The procedure of local pairwise profile alignment is as follows.

(1) Define partial matrices $\hat{\mathbf{P}}_u^{(i)} = [\mathbf{p}_1^{(i)} \dots \mathbf{p}_u^{(i)}]$ and $\hat{\mathbf{Q}}_v^{(j)} = [\mathbf{q}_1^{(j)} \dots \mathbf{q}_v^{(j)}]$. Construct an $(n_i + 1) \times (n_j + 1)$ matrix M whose (u, v)-th element $M(u, v)$ for $u = 1, \dots, n_i$ and $v = 1, \dots, n_j$ represents the score of an optimal profile alignment between $\hat{\mathbf{P}}_u^{(i)}$ and $\hat{\mathbf{Q}}_v^{(j)}$ and between $\hat{\mathbf{P}}_v^{(j)}$ and $\hat{\mathbf{Q}}_u^{(i)}$, given that the alignment ends with $\mathbf{p}_u^{(i)}$ aligned to $\mathbf{q}_v^{(j)}$ and $\mathbf{p}_v^{(j)}$ aligned to $\mathbf{q}_u^{(i)}$. Construct an $n_i \times n_j$ matrix I whose (u, v)-th element represents the score of an optimal alignment, given that the alignment ends with $\mathbf{p}_u^{(i)}$ or $\mathbf{q}_u^{(i)}$ aligned to a gap. Similarly, construct an $n_i \times n_j$ matrix J whose (u, v)-th element represents the score of an optimal alignment given that the alignment ends with $\mathbf{p}_v^{(j)}$ or $\mathbf{q}_v^{(j)}$ aligned to a gap.



(a)



(b)

Fig. 6. (a) Accuracy and (b) recognition time achieved by FDR, SVM-RFE, R-SVM, and the fusion of FDR and SVM-RFE (Eq. 13 with $\alpha = 0.3$) in the subcellular localization task.

(2) Initialize an accumulative score matrix M:

$$M(0,0) = 0$$
$$M(u,0) = -g_{\text{open}} - (u-1)g_{\text{ext}}$$
$$M(0,v) = -g_{\text{open}} - (v-1)g_{\text{ext}}$$

where $u = 1, 2, \ldots, n_i$, $v = 1, 2, \ldots, n_j$, and g_{open} and g_{ext} are two

user-defined parameters representing the gap opening penalty and gap extension penalty, respectively. [3]

(3) Calculate $M(u, v)$ recursively as follows:

$$M(u,v) = \max \begin{cases} 0 \\ M(u-1, v-1) + \varepsilon(S_u^{(i)}, S_v^{(j)}) \\ I(u-1, v-1) + \varepsilon(S_u^{(i)}, S_v^{(j)}) \\ J(u-1, v-1) + \varepsilon(S_u^{(i)}, S_v^{(j)}) \end{cases}$$

where

$$I(u,v) = \max \begin{cases} 0 \\ M(u-1, v) - g_{\text{open}} \\ I(u-1, v) - g_{\text{ext}} \end{cases}$$

$$J(u,v) = \max \begin{cases} 0 \\ M(u, v-1) - g_{\text{open}} \\ J(u, v-1) - g_{\text{ext}} \end{cases}$$

(4) Obtain the similarity score of the local pairwise profile alignment of $S^{(i)}$ and $S^{(j)}$ as follows:

$$\rho(S^{(i)}, S^{(j)}) = \max\{M(\hat{u}_i, \hat{v}_j), I(\hat{u}_i, \hat{v}_j), J(\hat{u}_i, \hat{v}_j)\}.$$

where $(\hat{u}_i, \hat{v}_j)$ is the position in M corresponding to the maximum local alignment score, i.e.,

$$(\hat{u}_i, \hat{v}_j) = \arg \max_{1 \le u \le n_i;\ 1 \le v \le n_j} \{M(u,v), I(u,v), J(u,v)\}.$$

(5) Repeat Steps 2 to 4 for every pair of sequences.

Acknowledgement

This work was in part supported by The Research Grant Council of the Hong Kong SAR (Project Nos. PolyU 5230/05E and A-PH18.

[3] In this work, the open gap (g_{open}) and extension gap penalties (g_{ext}) were set to 11 and 1, respectively.

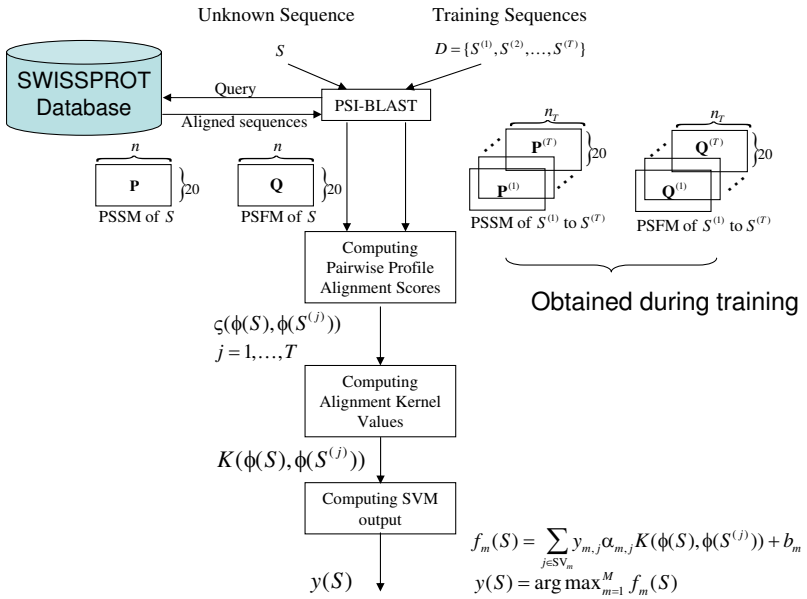


Fig. 7. Prediction of subcellular locations of proteins by pairwise profile alignment SVMs.

References

[1] L. Liao, W. S. Noble, Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships, J. Comput. Biol. 10 (6) (2003) 857–868.

[2] J. K. Kim, G. P. S. Raghava, S. Y. Bang, S. Choi, Prediction of subcellular localization of proteins using pairwise sequence alignment and support vector machine, Pattern Recog. Lett. 27 (9) (2006) 996–1001.

[3] J. Guo, M. W. Mak, S. Y. Kung, Eukaryotic protein subcellular localization based on local pairwise profile alignment SVM, in: 2006 IEEE International Workshop on Machine Learning for Signal Processing (MLSP'06), 2006, pp. 391–396.

[4] R. Kuang, E. Ie, K. Wang, K. Wang, M. Siddiqi, Y. Freund, C. Leslie, Profile-based string kernels for remote homology detection and motif extraction, J. Bioinfom. Comput. Biol. 3 (2005) 527–550.

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

[6] M. Xiong, W. Li, J. Zhao, L. Jin, E. Boerwinkle, Feature (gene) selection in gene expression-based tumor classification, Molecular Genetics and Metabolism 73 (3) (2001) 239–247.

[7] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines, Machine Learning 46 (2002) 389422.

[8] X. G. Zhang, X. Lu, Q. Shi, X. Q. Xu, H. C. E. Leung, L. N. Harris, J. D. Iglehart, A. Miron, J. S. Liu, W. H. Wong, Recursive SVM feature selection and sample classification for mass-spectrometry and microarray data, BMC Bioinformatics 7 (197).

[9] S. Y. Kung, M. W. Mak, Feature selection for pairwise scoring kernels with applications to protein subcellular localization, in: ICASSP'07, 2007, (Submitted).

[10] http://www.expasy.org/sprot.

[11] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, D. J. Lipman, Gapped BLAST and PSI-BLAST: A new generation of protein database search programs, Nucleic Acids Res. 25 (1997) 3389–3402.

[12] T. F. Smith, M. S. Waterman, Comparison of biosequences, Adv. Appl. Math. 2 (1981) 482–489.

[13] O. Gotoh, An improved algorithm for matching biological sequences, J. Mol. Biol. 162 (1982) 705–708.

[14] J. Weston, A. Elisseeff, B. Scholkopf, M. Tipping, Use of the zero-norm with linear models and kernel methods, Journal of Machine Learning Research 3 (2003) 14391461.

[15] B. Krishnapuram, L. Carin, A. Hartemink, Gene expression analysis: Joint feature selection and classifier design, in: B. Scholkopf, K. Tsuda, J. Vert (Eds.), Kernel Methods in Computational Biology, MIT Press, 2004, Ch. 14, pp. 299–317.

[16] P. Pavlidis, J. Weston, J. Cai, W. N. Grundy, Gene functional classification from heterogeneous data, in: Int. Conf. on Computational Biology, Pittsburgh, PA, 2001, pp. 249–255.

[17] M. W. Mak, S. Y. Kung, A solution to the curse of dimensionality problem in pairwise scoring techniques, in: Int. Conf. on Neural Information Processing, 2006, pp. 314–323.

[18] H. C. Peng, F. H. Long, 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) (2005) 1226–1238.

[19] K. Kira, L. Rendell, A practical approach to feature selection, in: International Conference on Machine Learning, Aberdeen, 1992, pp. 368–377.

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

[21] W. Pan, A comparative review of statistical methods for discovering differentially expressed genes in replicated microarray experiments, Bioinformatics 18 (2002) 546–554.

[22] R. Kohavi, G. H. John, Wrappers for feature selection, Artificial Intelligence 97 (1-2) (1997) 273–324.

[23] A. Reinhardt, T. Hubbard, Using neural networks for prediction of the subcellular location of proteins, Nucleic Acids Res. 26 (1998) 2230–2236.

[24] S. Y. Kung, M. W. Mak, Machine learning for multi-modality genomic signal processing, IEEE Signal Processing Magazine 23 (3) (2006) 117–121.