Using Evolutionary Computation to Improve SVM Classification

Uday Kamath *Member, IEEE*, Amarda Shehu *Member, IEEE* and Kenneth De Jong, *Member, IEEE*

***Abstract*— Support vector machines (SVMs) are now one of the most popular machine learning techniques for solving difficult classification problems. Their effectiveness depends on two critical design decisions: 1) mapping a decision problem into an n-dimensional feature space, and 2) choosing a kernel function that maps the n-dimensional feature space into a higher dimensional and more effective classification space. The choice of kernel functions is generally limited to a small set of well-studied candidates. However, the choice of a feature set is much more open-ended without much design guidance. In fact, many SVMs are designed with standard generic feature space mappings embedded *a priori*. In this paper we describe a procedure for using an evolutionary algorithm to design more compact non-standard feature mappings that, for a fixed kernel function, significantly improves the classification accuracy of the constructed SVM.**

I. Introduction

Support vector machines (SVMs) are now one of the most popular machine learning techniques and for good reason: they have been shown to exhibit some of the best accuracy on difficult classification problems. The most impressive results have been with binary classification problems which are the focus of this paper.

The effectiveness of an SVM depends on two critical design decisions: 1) mapping a decision problem into an n-dimensional feature space, and 2) choosing a kernel function that maps the n-dimensional feature space into a higher dimensional and more effective classification space. The choice of kernel functions is generally limited to a small set of well-studied candidates (polynomials, radial basis functions, etc.). However, when the problem involves learning to classify items such as documents, images, DNA sequences, etc., the appropriate choice of a feature set is much more open-ended and without much design guidance. A standard approach is to design an SVM around a "kernel" that internally combines both a generic feature space mapping with a standard kernel function. For example, "vector space kernels" used for document classification use word frequency vectors to represent documents. In DNA sequence analysis, a standard approach is to use "spectrum kernels" that map strings into a frequency vector of a "spectrum" of feature patterns.

Although these generic feature space mappings have broad applicability and often result in acceptable classification accuracy, Intuitively, one feels that a more carefully designed and more compact set of features would improve SVM classification performance. We explore that possibility in this paper. We define and evaluate a procedure for using an evolutionary algorithm to design a compact set of features that, for a fixed kernel function, significantly improves the classification accuracy of the constructed SVM on two difficult binary classification problems drawn from the field of bioinformatics.

Uday Kamath is with Norkom Technologies in Reston, Virginia, USA 20191 (email: kamathuday@gmail.com).

Amarda Shehu is with the Department of Computer Science, George Mason University, Fairfax, Virginia, USA 22030 (email: amarda@cs.gmu.edu).

Kenneth De Jong is with the Department of Computer Science, George Mason University, Fairfax, Virginia, USA 22030 (email: kdejong@gmu.edu).

II. Evolving SVM Feature Sets

SVMs have a solid theoretical foundation in statistical learning theory [30], [31]. They continue to be used successfully for a wide variety of binary classification problems including problems that involve non-vector data objects such as text, graphs, and strings. They do so by mapping of the objects to be classified into an n-dimensional feature space. Then, given labeled training data in feature vector form, an SVM classifier transforms the training data into an internal vector space, typically of higher dimensionality. The transformation is chosen so that the positive and negative examples are linearly separable by a hyperplane in the higher-dimensional space. Then, subsequent prediction of the label of an unlabeled object involves determining on which side of the hyperplane the (transformed) point lies.

The success of this approach depends on both the choice of the feature space used and the internal transformation (the "kernel function") used. Well-known kernel functions include polynomials, radial bases, gaussian radial bases, and sigmoid functions [1]. The particular choice of a kernel is problem-specific and generally determined experimentally.

The focus of this research is on improving the feature space used for a given kernel function by first expanding the candidate set of features and then finding a compact subset of features that improves classification performance. There are a variety of algorithmic approaches one might take to do feature subset selection (see, for example, [34]). In this paper we explore the use of evolutionary algorithms for this task. This is not a new idea. There are a number of studies that have used of genetic algorithms to find good performing feature subsets of (see, for example, [29], [33], or [3]). The standard approach has been to represent feature subsets as fixed-length binary strings of length L, the cardinality of the total set of features. These binary strings represent feature subsets by having a 1 in each bit position associated with the included features. Standard crossover and mutation operators are used to explore the space of all possible subsets by generating offspring from parent strings exhibiting higher fitness.

The challenge is how to assign fitness to feature subsets. Ideally, a "wrapper" approach is used [17] in which a feature subset is giving to the underlying machine learning process

978-1-4244-8126-2/10/$26.00 ©2010 IEEE

which in turn uses the given features to construct a classifier and then evaluates its accuracy using a sound empirical methodology (e.g., k-fold validation). However, for large feature sets and large data sets, the cost of such precise fitness evaluation is computationally infeasible. A more practical approach is evolve feature subsets using a simpler fitness heuristic and then submit the best subsets found to the more rigorous evaluation via the underlying machine learning process.

This "filter" approach [17] is the one used in this paper. Candidate feature sets are evolved using a heuristic fitness function and the best found are passed to a standard machine learning procedure to construct an SVM classifier and then evaluate its accuracy.

The effectiveness of this evolutionary approach is illustrated using two difficult binary classification problems from the field of bioinformatics. Both involve classification of DNA sequences represented as variable-length strings drawn from the familiar {A,C,G,T} alphabet.

The most recent successful SVM applications to these kind of problems employ a "spectrum kernels" that use simple generic spectrum feature spaces. Spectrum features are all finite-length sequences that can be generated from an alphabet α [19], [2]. For a fixed k, a k-spectrum is the set of $d = |\alpha|^k$ features that correspond to all strings of length k (k-mers). A DNA sequence data string S is mapped into this d-dimensional feature space by simply recording the normalized occurrence of each k-mer in S. The normalization is with respect to the total number of times the k-mers appear in the sequence.

For instance, the sequence ACGT contains three 2-mers AC, CG, and GT ($\Sigma = \{A, C, G, T\}$.) The $k = 2$ spectrum representation of this sequence is the 16-element vector with 0.33 entries corresponding to 2-mers above and 0 entries corresponding to all others.

The sense one gets is that this is a rather generic and all-inclusive method of choosing discriminating features for an SVM. The motivation for this research is the hypothesis that SVM classification can be enhanced by using a smaller and more selective set of features (motifs) evolved by an appropriately designed evolutionary algorithm.

For a given motif length k one could adopt the standard binary string subset representation used by other genetic-algorithm-based approaches in order to evolve an effective subset of k-mers from the full spectrum feature set. However, as noted above, the cardinality d of the full spectrum feature set is $(|\alpha| = 4)^k$. Other studies in the literature suggest that motifs of at least length $k = 6$ are required for accurate classification, resulting in binary string representations of lengths longer than 4000. Since we wanted to explore motifs with $k > 6$ and motifs expressed from a larger alphabet, the standard binary string subset representation was infeasible. Rather, we chose to evolve populations of motifs directly, and use the best found as our SVM feature set.

1) Motif Representation: The individuals (motifs) in our EA are variable-length strings generated from the IUPAC code for DNA sequences [4] detailed in Table I. This code contains characters that represent more than one nucleotide, allow motifs in which specific positions are not constrained to specific nucleotides but rather to a group of nucleotides with shared properties. In this way, the motifs offered by the EA after its final iteration may reveal interesting insight into what sequence features are important for accurate classification.

We chose to vary the length of our motifs from 6-mers to 12-mers. This reflects the lack of *a priori* information on the length of optimal motifs and the findings in [22] suggest that motifs longer than 5-mers are needed to achieve more than 80% classification accuracy with an SVM.

TABLE I. IUPAC code is adapted from [4].

Symbol	Meaning	Description Origin
G	G	**G**uanine
A	A	**A**denine
T	T	**Thymine**
C	C	**C**ytosine
R	G or A	pu**R**ine
Y	T or C	p**Y**rimidine
M	A or C	a**M**ino
K	G or T	**K**etone
S	G or C	**S**trong interaction
W	A or T	**W**eak interaction
H	A or C or T	**H** follows G in alphabet
B	G or T or C	**B** follows A in alphabet
V	G or C or A	**V** follows U in alphabet
D	G or A or T	**D** follows C in alphabet
N	G or A or T or C	a**N**y

2) Offspring Generation: Given a current population of individuals (motifs), the EA creates offspring using two basic operators: mutation and crossover. The EA used in this research gives equal probability to each of the operators.

If the mutation operator is chosen to create a new offspring, any of the individuals in the current parent population have equal probability of being selected. Once an individual is chosen for mutation, any of its symbols has equal probability of being mutated into any of the other symbols of the IUPAC code shown in Table I.

Any pair of individuals has equal probability of selection for crossover. Once two individuals are chosen as parents, their genetic material is combined to produce an offspring. While our EA implementation allows a variety of crossover operators [6], only one-point crossover was used for the experiments in this paper.

3) Fitness Function: The fitness function evaluates individuals in a population in order to provide selection pressure towards improvement. While the true value of an individual motif is measured in terms of its effectiveness in the context of SVM-based classification, constructing an SVM and evaluating its classification accuracy each time an offspring needs to be evaluated is impractical from a computational cost perspective. Therefore, we employed a simpler fitness function that approximates the way in which features in an SVM are employed in the kernel function. In essence, the fitness function rewards individuals that are over-represented in positive examples and under-represented in negative examples.

Specifically, the fitness function calculates the absolute difference between the percentage of positive examples and the percentage of negative examples that contain an individual (motif). In this way, a motif that is found in all positive examples and no negative examples will have the highest score of 100, whereas a motif found in equal amounts in both will have the lowest score of 0.

A. EA Population Dynamics

We search this space of candidate motifs using a $(\mu + \lambda)$-style EA. In the experiments reported in this paper, we used $\mu = 100$ parents and $\lambda = 25$ offspring, beginning with 100 randomly generated motifs. In each generation, 25 offspring are produced from (uniform) randomly selected parents using both mutation and crossover operators. Truncation selection is employed to determine which 100 of the 125 will survive as the next generation of parents.

In the experiments reported in section V, an upper bound 5000 generations was used, although convergence in the top fitness scores (i.e., the hall of fame) was generally obtained within 500 generations. The hall of fame motifs were then evaluated for their effectiveness as features in SVM classification.

1) A Parallel EA for Speciation: As described, our EA populations contain motifs of varying length. That raises a well-known issue when choosing two parents to produce an offspring via crossover: should the parents be of the same length? If so, crossover never produces offspring that have a different length than their parents. Since the specified mutation operator does not change the length either, there is no way to generate and maintain length diversity in the population. On the other hand, it is often the case in both nature and evolutionary algorithms that offspring produced by structurally dissimilar parents are inviable.

In our case, preliminary experiments indicated better results were obtained when recombining parents of equal length. Since we were interested in exploring a limited set of motif lengths (6-12), we adopted an island-model approach in which each island contains individuals of the same length (i.e., one motif species), and evolves in isolation and in parallel with other islands without migration. The best motifs found on each island were collected in a single "hall of fame" for subsequent evaluation via an SVM.

The best results, both in terms of the absolute quality of top-scoring motifs and the accuracy achieved by the SVM with these features, were obtained when employing this island model in our method. These observations are in line with recent research that shows that an island model may yield more fit chromosomes rather than crossbreeding different species [32], [6].

III. Two Difficult Bioinformatics Binary Classification Problems

The use of machine learning techniques to attack difficult bioinformatics problems continues to grow. In particular, there has been considerable interest in the application of SVM techniques to difficult binary classification problems. In this section we describe two of them which serve as our experimental testbed for our ideas on using evolutionary computation techniques to improve SVM classification accuracy.

A. DNase I Hypersensitive DNA Sites

Protein expression in cells is controlled by regulating transcription of genes into mRNA. Transcription factor proteins bind specific DNA sequences known as regulatory elements to activate or repress gene transcription. The module of regulatory elements is located a few hundred bases upstream of the gene being regulated [5], [13]. Wet-lab annotation of regulatory elements was a laborious and expensive process until the discovery of DNA sites that were hypersensitive to enzymes like DNase I. These sites, referred to as hypersensitive (HS) sites, were found to be reliable markers of regulatory elements. Their identification is currently the golden approach that promises to reveal almost all classical regulatory sequences and dramatically accelerate the functional annotation of the entire human genome [25].

The large number of discovered HS sequences have created the opportunity to develop computational methods that can learn to recognize these sequences and assist in genome-wide annotation of regulatory elements. Work in [22] shows that an SVM employing all 6-mers as features learns to recognize HS sequences with accuracy more than 80%.

HS sequences are believed to contain complex signals that facilitate recognition by specific DNA-binding proteins interacting cooperatively over short distances of 150-250 base pairs [8], [28]. While it is not known *a priori* whether recognizing such signals is computationally tractable, a reasonable hypothesis is that short motif-like sequence features can capture differences between HS and non-HS sequences. In the absence of any *a priori* knowledge of motifs, the SVM in [22] considers all subsequences of fixed length as potential motifs. However, even if features in known HS sequences are captured by some subsequences, the noise from the rest of the subsequences may adversely impact the performance of a classifier. Moreover, a brute-force approach sheds little insight into which motifs may be employed by HS sequences to interact with DNA-binding proteins.

This paper pursues the hypothesis that motifs, short subsequences of finite length that capture and maximize differences between HS and non-HS sequences, can be designed and employed as meaningful features for an SVM-based classifier. The EA presented in this paper searches the space of possible motifs for those that maximize differences between known HS and non-HS sequences. A systematic analysis compares the effectiveness of replacing all k-mers with the EA-obtained motifs in the context of an SVM-based classification. The analysis in section V shows that a small number of meaningful motifs can replace the exhaustive list of all k-mers, affording both better accuracy and insight into the actual biological signals employed by HS sequences to recognize DNA-binding proteins.

B. DNA Splice Sites

The second problem we considered involves the recognition of DNA splice sites, which mark boundaries between coding and non-coding regions in eukaryotic DNA sequences. Annotation of DNA splice sites is key to determining which coding regions, also known as exons, are concatenated together into mRNA. Transcription of DNA sequences into mRNA in eukaryotic organisms occurs only after enzymes splice away non-coding regions, also known as introns, from the precursor (pre-mRNA sequence) to leave only exon regions for transcription into mRNA. Since a protein-encoding gene is defined as a series of exon regions, annotation of splice sites is fundamental to the gene-finding problem [9].

A distinction is made between acceptor and donor splice sites. While an acceptor splice site marks the start of an exon, a donor splice site marks the end. The AG dinucleotide is a consensus sequence among (canonical) acceptor splice sites, and the GT dinucleotide is a consensus sequence among (canonical) donor splice sites. There are few non-canonical splice sites that do not contain the AG or GT consensus dinucleotides. AG and GT dinucleotides are commonly found in non-splice site sequences and cannot be used as discriminating features. Additionally, the nucleotide composition of splice site sequences is not significantly different from the nucleotide composition of non-splice site sequences [24].

The performance of early statistical approaches that employed positional probabilities to recognize splice sites was generally poor [27], [36]. Since then, significant work in machine learning focused on identifying discriminating features for splice sites [35], [15], [14], [16]. In some of the most successful splice-site prediction work [15], [14], [16], an iterative feature generation algorithm constructs interesting features out of basic k-mers to obtain a limited subset ($\sim 5,000$) of features from an exhaustive list of k-spectrum features (with k ranging from 3 to 6).

To the best of our knowledge, evolutionary computation techniques have not been investigated for their ability to reveal discriminating features for the recognition of splice sites; hence, the motivation for our application of an EA to find splice site motifs that can serve as discriminating features in the context of classification. Determining motifs in acceptor and donor splice sites is important not only to improve the accuracy of recognizing splice sites, but also to find the inherent sequence-based signals employed by splice sites to recognize and interact with the spliceosome. The analysis in section V compares the effectiveness of replacing all k-mers with the EA-obtained motifs in an SVM. Similar classification accuracy, $> 80\%$, is obtained with fewer motif features than currently employed in splice-site recognition [15], [14]. Additionally, inspection of these motifs reveals that they contain interesting sequence signals that facilitate interactions with the spliceosome.

IV. Methods

To summarize, we use an evolutionary algorithm (EA) to design motifs, short subsequences of finite length, that capture and maximize differences between positive and negative examples for classification purposes. The classification performance of an SVM using these evolved motifs is compared with the performance of a baseline SVM using standard spectrum features.

Unlike work in [22] which employs linear kernel functions, our SVM classifier employs an RBF. Recent work that compares RBFs to polynomial kernel functions suggests that RBFs perform better than polynomial kernel functions [10]. While significant work is devoted to determining optimal kernel functions for specific problem domains (e.g., [23], [21], [20]), the focus in this work is on determining optimal motifs that capture important features of DNA sequences.

A. DNase I HS Data

The HS input data set employed in the fitness function of our EA and the training and cross-validation of the SVM consists of 280 experimentally-established erythroid HS sequences and 737 non-HS sequences. This data set was obtained from noble.gs.washington.edu/proj/hs. The HS sequences were identified from throughout the human genome through a novel experimental methodology that identifies HS sites employing cloning and in-vivo activity of K562 erythroid cells [25]. The non-HS sequences, collected (and distributed proportionally) from throughout the human genome, were not hypersensitive when tested in the same cell type. Both 280 HS and 737 non-HS sequences have similar average lengths of 242 nucleotides.

B. Splice Site Data

The splice site input data set is extracted from $5,057$ human RefSeq pre-mRNA sequences obtained from NCBI (www.ncbi.nlm.nih.gov/). These sequences were used to construct positive (containing splice sites) sequences, and negative (non splice-site) sequences. A splice-site sequence in the training data set consists of 162 nucleotides, 80 upstream of the NCBI-annotated donor (AG) dinucleotide and 80 downstream of the annotated acceptor (GT) dinucleotide (80+AG/GT+80). This definition closely follows that employed in machine learning literature on splice site recognition [24], [15], [14] to allow direct comparisons with related work. The resulting positive training set consists of $25,504$ donor and $25,504$ acceptor splice-site sequences. Keeping in line with work in [15], [14], negative sequences of 162 nucleotides are defined around randomly picked AG- or GT-pair locations that are not part of NCBI-annotated splice sites. The negative training set consists of $200,000$ AG-centered and $200,000$ GT-centered non-splice site sequences.

C. ROC Performance Measures

We employ receiver operating characteristic (ROC) curves to measure and compare the quality of SVM classifiers [11]. A trained SVM is used to produce a ranked list of positive instances from a set of unlabeled data. The positive instances

are ordered from the most to least confident. If one varies a threshold from the top to the bottom of the list, the rate of true and false positives change with the threshold. An ROC curve plots the true positive rate as a function of the false positive rate as the threshold changes. We define the *ROC score* of an SVM to be the area under a given ROC curve. While random ranking is expected to yield a score of ~ 0.5, the ROC score reaches 1 if the SVM correctly places all of the correct positive instances above the threshold.

D. Cross-validation

The performance of the SVM is evaluated via 10-fold cross-validation. The datasets are randomly divided into 10 subsets of equal size. The SVM is trained on 90% of the subsets and tested on the 10% held out. The ROC scores reported are the average ROC score obtained over the 10-fold validations. These mean ROC scores are used to compare the performance of an SVM using the evolved motifs as features with that of a baseline SVM using spectrum features.

E. Implementation Details

The proposed method is implemented in Java and run on an Intel Core2 Duo machine with 4GB RAM and 2.66GHz CPU. The EA implementation builds upon the Evolutionary Computation software that accompanies a textbook on the topic [6]. The method integrates genomic sequences with utilities for bioinformatics by making use of the open-source Biojava project [12]. Finally, our implementation of SVM in this work builds upon the LibSVM package [7].

V. Results

We conducted a systematic analysis that investigates whether employing the EA-obtained motifs as features improves the classification accuracy in the context of an SVM with an RBF kernel. Our baseline SVM used all 6-mers as the spectrum feature set.

We then repeated the process replacing the 6-mers with the EA-generated motifs. Recall that our EA evolved a list of candidate motifs ranked by a heuristic fitness function. Of interest then is how SVM classification performance changes as a function of the number of EA-generated motifs used as features. To get a sense of this we ran comparison experiments using the top 25, 50, 100, and 200 motifs.

We repeated this analysis for both binary classification problems. Our results are presented in the following sections.

A. Performance Analysis on the DNase I HS Data

Average ROC scores obtained from a 10-fold cross-validation for each of the experimental settings are summarized in Table II. The experimental results show that introduction of the EA-obtained motifs in the features employed by an SVM classification improves the average cross-validation accuracy by at least 7% over the baseline SVM classification that uses 6-mers as features. Moreover, the accuracy improves if the number of motifs increases from 25 to 50, reaching saturation soon afterwards.

TABLE II. Average μ, standard deviation σ, minimum, and maximum ROC scores obtained from the 10-fold cross-validation shown for each experimental setting for the HS problem.

HS Experiment	$\mu \pm \sigma$	min	max
6-mers	77.27 ± 0.0	77.3	77.3
25 motifs	84.41 ± 2.7	79.1	88.1
50 motifs	85.08 ± 2.3	80.3	87.2
100 motifs	85.13 ± 2.1	80.7	87.6
200 motifs	85.51 ± 1.4	79.1	88.10

For illustration purposes, example ROC curves are shown in Figure 1 for the baseline SVM with 6-mers as spectrum features and the last experimental setting that replaces these 6-mers with 200 EA-obtained motifs. Sensitivity is measured as TP/(TP + FN), and specificity is measured as TN/(FP+TN), where TP, TN, FP, and FN respectively refer to the number of true positives, true negatives, false positives, and false negatives. The ROC scores in Table II correspond to the area under the respective ROC curves averaged over a 10-fold validation.

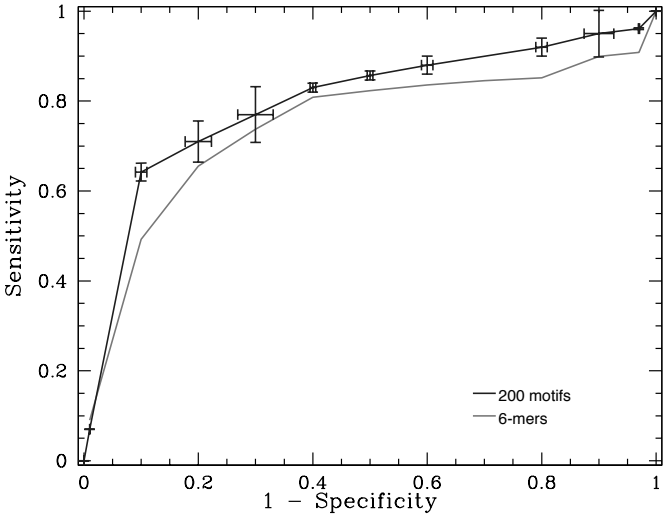


Fig. 1. Example ROC curves for the tested SVMs on the HS data. Sensitivity is plotted as a function of 1 - specificity. Error bars are calculated over 30 runs.

B. Performance Analysis on the Splice Site Data

Similarly, average ROC scores obtained from a 10-fold cross-validation for each experimental setting are summarized for splice site data in Table III. The experimental results show that introduction of the EA-obtained motifs in the features employed by an SVM classification improves the average cross-validation accuracy by at least 3% over the baseline SVM classification that uses 6-mers as features. Moreover, the accuracy improves as the number of motifs increases from 25 to 50, reaching saturation soon afterwards.

Again, for illustration purposes, example ROC curves for the Splice Data are shown in Figure 2 for the baseline SVM with 6-mers as spectrum features and the last experimental setting that replaces these 6-mers with 200 EA-obtained motifs.

TABLE III. Average μ, standard deviation σ, minimum, and maximum ROC scores obtained from the 10-fold cross-validation shown for each experimental setting for the Splice Site problem.

Splice Site Experiment	$\mu \pm \sigma$	min	max
6-mers	85.74 ± 1.2	84.1	87.9
25 motifs	88.01 ± 0.1	88.0	88.3
50 motifs	88.05 ± 0.1	88.1	89.3
100 motifs	88.10 ± 0.1	88.0	88.5
200 motifs	88.20 ± 0.2	88.1	88.9

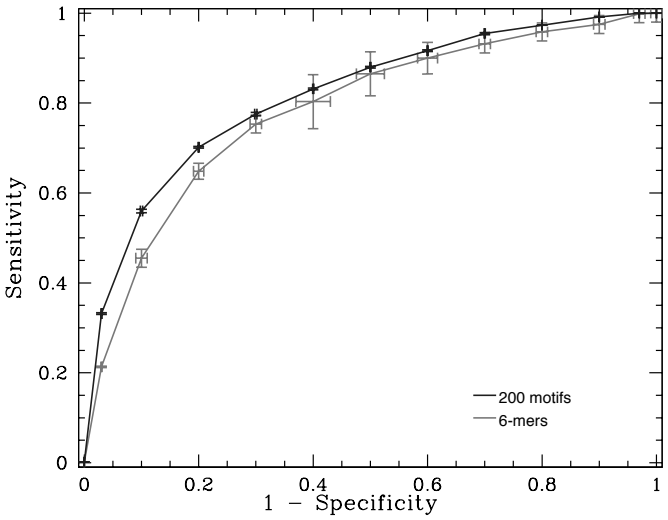


Fig. 2. Example ROC curves for the tested SVMs on the splice data. Sensitivity is plotted as a function of 1 - specificity. Error bars are calculated over 30 runs.

Note that, unlike the ROC curves shown for the HS sequence recognition problem in Figure 1, Figure 2 contains error bars even for the accuracy obtained with the baseline SVM classification employing 6-mers. Since the training data set splice site and non-splice sequences was very large (above 200K), the calculation of sensitivity and specificity was conducted on smaller samples of $10,000$ training sequences, keeping the ratio between the positive and negative examples the same as in the original training data set. This was repeated 30 times to obtain an average ROC curve with the error bars shown in Figure 2

C. Information Gain Analysis

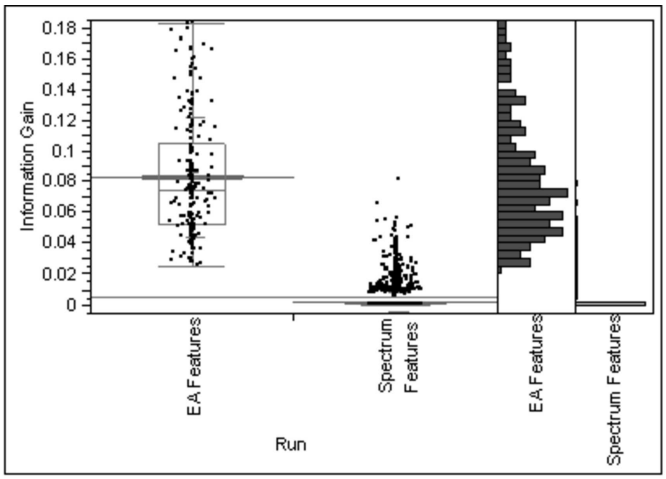
Another way of understanding the usefulness of the EA-evolved motifs is to perform a standard "information gain" analysis. Recall that the information gain associated with an attribute A associated with a set S, $Gain(S, A)$, is the difference in the entropy of the full set S and the subsets of S exhibiting the values of attribute A. More precisely,

$Gain(S, A) = Ent(S) - \sum_{v \in V(A)} \frac{S_v}{S} Ent(S_v)$

where $V(A)$ denotes the set of all possible values of attribute A, and S_v denotes the subset of S where A takes value v.

If we calculate the information gain for each of the evolved motifs and compare that with the information gain associated with each of the 6-mers, we obtain additional insight into their differences.

For the HS dataset, 3587 out of $4,096$ of the 6-mer spectrum features ($\sim 87\%$) show an information gain of zero. By contrast, the information gain for every EA-evolved motif is positive. If we sum the information gain of the entire feature set (naive Bayes assumption of independence), the cumulative information gain of the 6-mers is 8.16 while that of the evolved motifs is 16.51. Figure 3 summarizes the information gain calculations graphically using JMP software [26].



Means and Standard Deviations

Level	Nr.	Mean	StdDev	StdErrMean	Lower 95%	Upper 95%
Motifs	200	0.083	0.039	0.003	0.077	0.088
6-mers	4096	0.002	0.006	0.000	0.002	0.002

Fig. 3. Information gain values shown for 6-mers and 200 EA-obtained motifs for the HS sequence recognition problem.

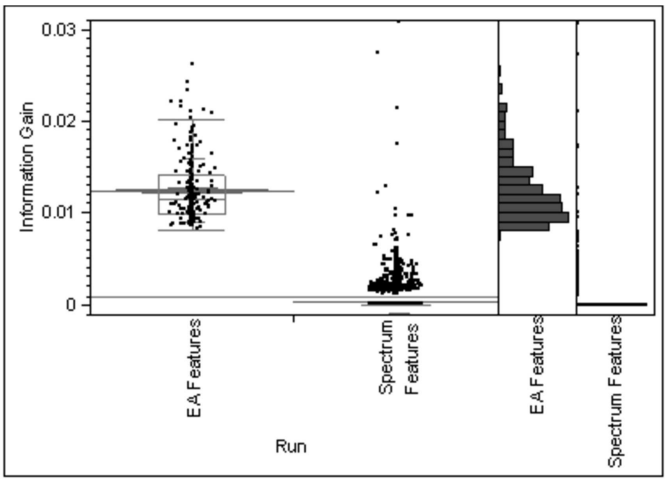
A similar analysis of the Splice Data shows that the information gain of $\sim 85\%$ of the 6-mers is zero while the EA-evolved motifs are not. The cumulative information gain of the 6-mers is 1.8 in contrast to 2.7 for the evolved motifs. Figure 4 summarizes this graphically.

The above comparison of the information gain from EA-evolved motifs versus that from 6-mers provides statistical evidence that EA-evolved motifs are more effective discrimination features than exhaustive spectrum features. The combination of a higher sum of information gains obtained with much fewer features illustrate how evolutionary computing can be employed to compress the number of features while increasing both information gain and overall classification accuracy.

D. Analysis of Motifs for Presence of Biological Signals

Additional insight into why the EA-obtained motifs afford higher SVM classification accuracy with fewer features than all k-mer spectrum features can be obtained through an analysis of these motifs for presence of any biological signals. Both HS and splice site contain complex biological signals that are used to recognize and interact with DNA-binding proteins and even RNA (in the case of splice sites).

A multiple sequence alignment analysis is conducted here to reveal any nucleotide patterns shared among the top-



Means and Standard Deviations

Level	Nr.	Mean	StdDev	StdErrMean	Lower 95%	Upper 95%
Motifs	200	0.012	0.003	0.000	0.012	0.013
6-mers	4096	0.000	0.001	0.000	0.000	0.000

Fig. 4. Information gain values shown for 6-mers and 200 EA-obtained motifs for the splice site sequence recognition problem.

scoring motifs employed as features for each problem. The alignment is carried out with clustalw 2.0 [18], using the IUB scoring matrix, which specifies symbol matching scores and gap penalties for sequences generated from the IUPAC code. The alignment that yields the highest additive score (data not shown) shows interesting repeated patterns shared among motifs used for the HS recognition problem and the motifs used for the splice site recognition problem.

Inspection of the alignment of HS motifs reveals that the dinucleotide CG is abundant and shared among all motifs. This result is in agreement with analysis in [22], which shows that the CG pair is abundant in HS sequences and confers high separation power to SVM. Additional patterns are present that go beyond dinucleotides, such as CGM, CGC, CGS, CGH, CGN, CSG, and even longer ones, such as CGMS, CGMSN, and CGSBN.

Inspection of the alignment of splice site motifs reveals the presence of patterns consistent with known biological signals on a typical pre-mRNA, such as branch site, pyrimidine-rich region close to acceptor splice site, splice-site consensus signals themselves, and exonic splicing enhancers. For instance, patterns like KVTYTT, which represents the nucleotide pattern G∨T ¬T T T∨C T T, or NVSGAS, whch represents the nucleotide pattern G∨T∨A∨C ¬ T G∨C GA G∨C, encapsulate biological signals like the branch site consensus or the pyrimidine tract.

The analysis of the top-scoring motifs shows that the novelty of our EA approach to feature selection is not only that the collection of features identified in this way improve prediction accuracy for HS or splice sites, but also that the selected features encapsulate biologically-interesting signals. The presence of ambiguous symbols in our motifs allows revealing beyond-nucleotide patterns that possibly encode the needed chemical properties at the interaction sites between the HS sequences and DNA-binding enzymes or the splice site sequences and the spliceosome. The motifs obtained through our EA can provide additional insight to biologists or researchers interested in detailed biological analysis of the signals that contribute to HS or splice site recognition.

VI. Conclusions

We have defined and evaluated an EA-based method to assist in creating a compact and effective set of features to be used to construct SVMs that exhibit improved performance binary classification tasks. We have illustrated the effectiveness of this approach through a series of experiments on two difficult binary classification problems taken from the field of bioinformatics.

For problems of this type, a standard SVM approach is to use a spectrum feature set based on simple motifs of fixed length. Our EA-based approach allows us to generate a smaller set of more complex and variable-length motifs which, when used as SVM features, significantly improves their classification accuracy.

The EA-based method described here is not restricted the kind of bioinformatics problems we used for evaluation purposes. The motifs evolved in these experiments are simple examples of what could easily be more complex string-matching patterns. Nor do the patterns or the objects need to be one-dimensional. Two and three dimensional image classification problems are natural candidates for this approach.

In this paper, we have focused on evolving feature sets for an SVM with a fixed, *a priori* defined kernel function. An interesting extension of this work would be to co-evolve both the kernel and the feature set.

References

[1] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In D. Haussler, editor, *5th Annual ACM Workshop on COLT*, pages 144–152. ACM Press, 1992.

[2] L. C., E. E., A. Cohen, A. Weston, and W. S. Noble. Mismatch string kernels for SVM protein classification. *Neur. Inf. Proc. Sys.*, 15(4):1441–1448, 2002.

[3] E. Cantu-Paz. Feature subset selection, class separability, and genetic algorithms. In *Proceedings of GECCO-2004*, pages 959–970. Springer Berlin / Heidelberg, 2004.

[4] I. Committee. Nomenclature committee of the international union of biochemistry (nc-iub). nomenclature for incompletely specified bases in nucleic acid sequences. recommendations 1984. 229(2):75–88, 1985.

[5] E. Davidson. *Genomic regulatory systems: Development and evolution*. Academic Press, New York, NY, 2001.

[6] K. A. De Jong. *Evolutionary computation: a unified approach*. MIT Press, Cambridge, MA, 2001.

[7] R.-E. Fan, P.-H. Chen, and C.-J. Lin. Working set selection using the second order information for training SVM. *J. Mach. Learn. Res.*, 6(1532-4435):1889–1918, 2005.

[8] G. Felsenfeld. Chromatin unfolds. *Cell*, 86(1):13–19, 1996.

[9] R. Guig0, P. Flicek, J. F. Abril, A. Reymond, J. Lagarde, F. Denoeud, S. Antonarakis, M. Ashburner, V. B. Bajic, E. Birney, R. Castelo, E. Eyras, C. Ucla, T. R. Gingeras, J. Harrow, T. Hubbard, S. E. Lewis, and M. G. Reese. EGASP: the human ENCODE genome annotation assessment project. *Genome Biol.*, 7(Suppl 1):S2.1–31, 2006.

[10] T. Habib, C. Zhang, J. Y. Yang, M. Q. Yang, and Y. Deng. Supervised learning method for the prediction of subcellular localization of proteins using amino acid and amino acid pair composition. *BMC Genom.*, 9(Suppl1):S1–S16, 2008.

[11] J. A. Hanley and B. J. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143:29–36, 1982.

[12] R. C. Holland, T. A. Down, M. Pocock, A. Prlic, D. Huen, K. James, S. Foisy, A. Draeger, A. Yates, M. Heuer, and M. J. Schreiber. BioJava: an open-source framework for bioinformatics. *Bioinformatics*, 24(18):2096–2097, 2008.

[13] M. L. Howard and E. Davidson. cis-Regulatory control circuits in development. *Dev. Biol.*, 271(1):109–118, 2004.

[14] R. Islamaj-Dogan, L. Getoor, and W. J. Wilbur. A feature generation algorithm with applications to biological sequence classification. In H. Liu and H. Motoda, editors, *Computational Methods of Feature Selection*. Springer, Berling, Heiderlberg, 2007.

[15] R. Islamaj-Dogan, L. Getoor, W. J. Wilbur, and S. M. Mount. Features generated for computational splice-site prediction correspond to functional elements. 8:410–416, 2007.

[16] R. Islamaj-Dogan, L. Getoor, W. J. Wilbur, and S. M. Mount. SplicePort - an interactive splice-site analysis tool. *Nucl. Acids Res.*, 35:W285–291, 2007.

[17] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97:273–324, 1997.

[18] M. A. Larkin, G. Blackshields, N. P. Brown, R. Chenna, P. A. McGettigan, H. McWilliam, F. Valentin, I. M. Wallace, A. Wilm, R. Lopez, J. D. Thompson, T. J. Gibson, and D. G. Higgins. Clustalw and clustalx version 2. *Bioinformatics*, 23(21):2947–2948, 2007.

[19] C. Leslie, E. Eskin, and W. S. Noble. The spectrum kernel: a string kernel for svm protein classification, 2002.

[20] M.-Z. Lu, C. L. P. Chen, and J.-B. Huo. Optimization of combined kernel function for svm by particle swarm optimization. In *IEEE Intl Conf on Machine Learning and Cybernetics*, volume 2, pages 1160–1166, Baoding, China, 2009.

[21] M.-Z. Lu, C. L. P. Chen, J.-B. Huo, and X. Wang. Optimization of combined kernel function for svm based on large margin learning theory. In *IEEE Intl Conf on Systems, Man and Cybernetics*, pages 353–358, Singapore, 2008.

[22] W. S. Noble, S. Kuehn, R. Thurman, M. Yu, and J. A. Stamatoyannopoulos. Predicting the in vivo signature of human gene regulatory sequences. *Bioinformatics*, 21(Suppl 1):i338–i343, 2005.

[23] S.-Y. Ohn, H.-N. Nguyen, D. S. Kim, and J. S. Park. Determining optimal decision model for support vector machine by genetic algorithm. In J. Zhang, J.-H. He, and F. Y., editors, *Lecture Notes in Computer Science: Computational and Information Science*, volume 3314, pages 895–902. Springer, 2005.

[24] M. Pertea, X. Lin, and S. L. Salzberg. Genesplicer: a new computational method for splice site prediction. *Nucl. Acids Res.*, 29(5):1185–1190, 2001.

[25] P. J. Sabo, R. Humbert, M. Hawrylycz, J. C. Wallace, M. O. Dorschner, M. McArthur, and J. A. Stamatoyannopoulos. Genome-wide identification of DNase I hypersensitive sites using active chromatin sequence libraries. *Proc. Natl. Acad. Sci. USA*, 101(13):4537–4542, 2004.

[26] SAS Institute Inc. JMP Version 7, 1998-2007.

[27] R. Staden. Computer methods to locate signals in nucleic acid sequences. *Nucl. Acids Res.*, 12(1):505–519, 1984.

[28] J. A. Stamatoyannopoulos, A. Goodwin, T. Joyce, and C. H. Lowrey. NF-E2 and GATA binding motifs are required for the formation of DNase I hypersensitive site 4 of the human beta-globin locus control region. *EMBO J.*, 14(1):106–116, 1995.

[29] H. Vafaie and K. D. Jong. Genetic algorithms as a tool for feature selection in machine learning. In *Proceedings of Conference on Tools for AI 1992*, pages 200–204. Society Press, 1992.

[30] V. N. Vapnik. *The nature of statistical learning theory*. Springer, New York, NY, 1995.

[31] V. N. Vapnik. *Statistical learning theory*. Wiley & Sons, New York, NY, 1998.

[32] K. Vertanen. Genetic adventures in parallel: Towards a good island model under PVM, 1998.

[33] J. Yang and V. Honavar. Feature subset selection using a genetic algorithm. *Intelligent Systems*, 13(2):44–49, 1998.

[34] J. Yang and S. Olafsson. Near-optimal feature selection for large databases. *Journal of the Operations Research Society*, 60:1045–1055, 2009.

[35] G. Yeo and C. Burge. Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. *J. Comp. Biol.*, 11(2-3):377–394, 2003.

[36] M. Q. Zhang and T. G. Marr. A weight array method for splicing signal analysis. *Comput. Appl. Biosci.*, 9(5):499–509, 1993.