BioSystems 137 (2015) 73–83

Contents lists available at ScienceDirect

BioSystems

journal homepage: www.elsevier.com/locate/biosystems

Molecular learning with DNA kernel machines

Yung-Kyun Noh [a], Daniel D. Lee [b], Kyung-Ae Yang [c], Cheongtag Kim [d], Byoung-Tak Zhang [e,*]

[a] Department of Mechanical and Aerospace Engineering, Seoul National University, Republic of Korea
[b] Department of Electrical and Systems Engineering, University of Pennsylvania, USA
[c] Department of Medicine, Columbia University, USA
[d] Department of Psychology, Seoul National University, Republic of Korea
[e] School of Computer Science and Engineering, Seoul National University, Republic of Korea

ARTICLE INFO

Article history:
Received 29 April 2015
Received in revised form 23 June 2015
Accepted 25 June 2015
Available online 7 July 2015

Keywords:
DNA computing
Machine learning
Learning *in vitro*
Kernel methods
Molecular algorithms

ABSTRACT

We present a computational learning method for bio-molecular classification. This method shows how to design biochemical operations both for learning and pattern classification. As opposed to prior work, our molecular algorithm learns generic classes considering the realization *in vitro* via a sequence of molecular biological operations on sets of DNA examples. Specifically, hybridization between DNA molecules is interpreted as computing the inner product between embedded vectors in a corresponding vector space, and our algorithm performs learning of a binary classifier in this vector space. We analyze the thermodynamic behavior of these learning algorithms, and show simulations on artificial and real datasets as well as demonstrate preliminary wet experimental results using gel electrophoresis.

1. Introduction

Molecular computation offers the potential for computing devices to be integrated seamlessly with biological systems (Benenson, 2012; Kahan et al., 2008). With molecular computation, a machine can be placed inside living tissue to quantitatively measure DNA expression and perform pattern recognition in order to identify potential diseases before any physical symptoms emerge. For this purpose, previous research devoted the effort to articulating pattern recognition algorithms to be implemented by biomolecular substrates.

However, most of the previous methods did not consider *learning* patterns from training examples. For example, a simple implementation of logic gates is proposed for neural networks using metabolic regulation (Laplante et al., 1995), without the implementation of the learning property of neural networks. In Kim et al. (2005), solving various dynamic neural network systems is proposed from RNA transcription processes with the light of similarity between two dynamical behaviors, where the realizability of the idea is reported later (Kim et al., 2006; Zhang et al., 2007), but the the molecules are designed to perform a pre-determined network. Additionally, Qian et al. (2011) proposed a systematic way of constructing feedforward and recurrent neural networks by cascading node units operated by DNA hybridization. In Mills et al. (2001) and Lim et al. (2010), a weighted sum operation on DNA molecules is designed to realize a simple perceptron algorithm. All these works have demonstrated the utility of molecular computation for solving pattern recognition problems, many of them exploiting the advantage of their inherently parallel interactions. However, these studies were confined to implementing pre-defined perceptrons and did not address the problem of *learning*. Any of these methods did not try to adapt the computational weight parameters to the pattern of training examples.

In this work, we present a molecular learning algorithm that can perform such pattern recognition *in vitro* directly on the biological molecules themselves, and provide simulations showing the state-of-the-art recognition accuracy for biological pattern recognition problems. We show how to model DNA sequences as embedded vectors in a vector space and the hybridization operation as a computation of the dot product between these vectors. DNA hybridization can then be interpreted as computing the inner product in the associated feature space via a Mercer kernel, i.e. the well-known kernel trick in machine learning (Schölkopf and Smola, 2001). Using the definition of inner product in a feature space, state-of-the-art pattern recognition algorithms can be used, which are simpler in both understanding and implementing than

* Corresponding author. Tel.: +82 2 880 1833.
E-mail address: btzhang@bi.snu.ac.kr (B.-T. Zhang).

http://dx.doi.org/10.1016/j.biosystems.2015.06.007
0303-2647/

the conventional artificial neural networks algorithm. In this work, a biomolecular computation process is modeled as a sequence of computing various kernel matrices, resulting in a well-defined learning and classification algorithm.

In our design, each computed kernel element is the pairwise similarity between two DNA molecules. The similarity is measured *in vitro* through hybridization and can be explicitly defined via the interaction energies of complementary DNA base pairs. Our algorithm manipulates populations of DNA sequences via hybridization and denaturing operations, modifying distributions of the associated vectors in the kernel feature space. After learning is performed on data examples, an unknown DNA sequence molecule can be directly classified using the learned weights in the molecular population. Our simulations with biological data show that the proposed algorithm achieves state-of-the-art performance, comparing favorably with traditional support vector machines and kernel Fisher discriminant analysis algorithms.

We also analyze our algorithm using thermodynamics and kinetics for DNA hybridization. We first obtain the thermodynamic properties for DNA hybridization based on the previous work (Kim et al., 2008; Sahu et al., 2006; SantaLucia and Hicks, 2004). DNA thermodynamics explain how the hybridization probability is determined by the change of energy and entropy as well as the temperature. Based on this work, we design the experiment schedule where the learning is performed properly. We provide a simple kinetic model explaining how the kernel matrix can be positive definite for appropriate temperature schedule. The suggested temperature schedule during hybridization is a simple cooling schedule with a constant speed from high temperature to low temperature. We can also apply small variations of the schedule to have different positive definite kernels, where these variations can be compared to the tuning parameters that control the sparsity of kernel matrix.

Specific implementation methods on DNA molecules are also proposed considering the constraint without traditional computing architectures such as semiconductor devices. We designed a small experiment with real DNA molecules and present preliminary experiments demonstrating how our proposed methods can be applied for real *in vitro* application.

The remainder of the paper is organized as follows: In Section 2, we briefly explain how kernel methods are used for DNA sequence analysis in our work, and we show how the data are embedded into a feature space of the associated kernel of hybridized molecules. Section 3 explains the molecular learning and classification algorithm as well as their geometrical interpretation in the associated feature space. In Section 4, we explain how positive definiteness of the kernel can be guaranteed, and in Section 5, we present simulation results on both synthetic and benchmark datasets. In Section 6, a discussion on the real implementation is provided, and preliminary *in vitro* experiment is presented. Finally, we conclude with a discussion in Section 7.

2. DNA kernel for molecular computation

Our definition of the kernel measures similarity between DNA sequences using similarities in biomolecular interaction. DNA molecules interact as they diffuse in solution through Brownian motion (Bennett, 1982), and can generate thermodynamic reactions by binding and unbinding complementary strands until dynamic equilibrium is reached. The resulting quantity of double-stranded sequences reflect biological similarity between DNA sequences, which can be viewed as coefficients of a novel kernel matrix here. We formally define this kernel matrix in this section, and show how it can be used for learning and classification later.

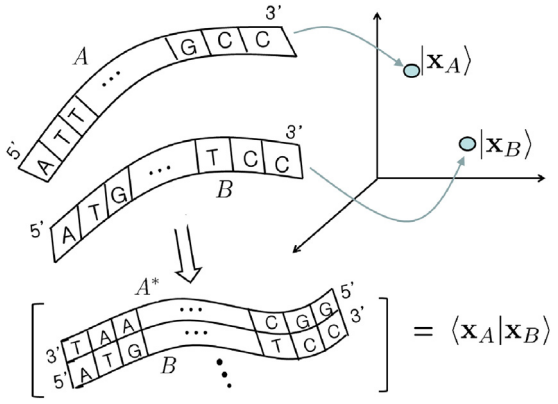

Fig. 1. Each single stranded DNA sequence is mapped into a vector space by an inner product, defined by the amount of resulting double-stranded product in the hybridization reactions.

2.1. Kernel definition using hybridization

We consider two-class training data composed of N single-stranded DNA molecules. The sequences are the strings from nucleotide alphabets $\Sigma = \{A, T, G, C\}$, and each sequence is labeled with a binary class label $y \in \{+1, -1\}$. Learning occurs on a dataset $\{|\mathbf{x}_i\rangle, y_i\}_{i=1}^N$ where $|\mathbf{x}_i\rangle$ is a vector corresponding to the ith sequence, and y_i is its label. We also consider the conjugate vector $\langle \mathbf{x}_j|$ of a sequence j, and we write a kernel element K_{ij} as the inner product of ith and jth data.

$$K_{ij} = \langle \mathbf{x}_j | \mathbf{x}_i \rangle \tag{1}$$

The embedding of each $|\mathbf{x}_i\rangle$ in a feature space is specified by the kernel matrix K, as illustrated in Fig. 1.

In our algorithm, pairwise similarity is defined via the hybridization reaction. Hybridization is the process of binding two single-stranded molecules together to make a double-stranded molecule. For DNA molecules, hybridization affinity exists between nucleotide pairs 'A' and 'T' and between 'G' and 'C', which are known to have a complementary relationship. DNA sequences are also directed, distinguished by the 5′ and 3′ ends of sequences. Hybridization occurs by matching the 3′ side of one sequence with the 5′ end of the other sequence. The complementary version of a DNA sequence can be generated by replacing nucleotides according to (A→T, T→A, G→C, and C→G), and reversing the 5′–3′ direction. For example, the complementary sequence of [5′-GCCATA-3′] is [5′-TATGGC-3′] as shown in Fig. 2. Complementary pairs are sequences having the greatest hybridization affinity.

Using this notion of complementarity, the definition of a kernel element is straightforward. The kernel element K_{ij} is the quantity of hybridized double strands between sequence i and the complement of sequence j starting from equal amounts of of single-stranded sequences, when pairs i and j are mixed:

$$K_{ij}^{in\ vitro} = |\mathrm{dsDNA}(j^*, i)| \tag{2}$$

Here, we use j^* to represent the complementary sequence of j. With this definition, the complementary sequence is analogous to the notion of conjugacy when we consider the inner product of $|\mathbf{x}_i\rangle$ and

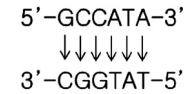

Fig. 2. One example of a complementary sequence. Nucleotides 'A' and 'T', 'G' and 'C' are exchanged, and the 5′–3′ direction is reversed.

$|\mathbf{x}_j\rangle$. Therefore, we also use the conjugate notation $\langle\mathbf{x}|$ to represent the complementary sequence.

The definition of the DNA kernel in this section, however, does not imply the positive definiteness of the kernel matrix. Reaction dynamics of hybridization are nonlinear and highly complex, and it is difficult to directly control the hybridization process. However, we will show later that if the hybridization procedure is annealed, that is starting from a high temperature and slowly cooling to low temperature, then the resulting kernel matrix will be close to a diagonally dominant positive definite matrix. With this guarantee of positive definiteness, we can design a virtual *in vitro* algorithm for learning a binary classifier from DNA molecular examples. In the following section, our *in vitro* learning algorithm is presented along with its geometrical interpretation.

3. Learning with DNA kernels

Here, we show how learning on DNA molecules can be implemented via specially designed biochemical processes. The learning algorithm yields a population of DNA molecules implicitly encoding a weight vector that can be used to classify unknown DNA examples. Our interpretation of the learning algorithm utilizes the support vector machine (SVM) framework (Schölkopf and Smola, 2001). In this context, learning has been considered as finding the closest point within the convex hulls of the class-specific training examples in the feature space (Kowalczyk, 2000). Our *in vitro* algorithm is similar, but instead learns the convex cones of the training examples, rather than the convex hulls.

We show how convex cones can be learned via populations of DNA molecules encoding kernel matrices using virtual *in vitro* hybridization processes. These kernel matrices can then be used to readout the class labels of unknown DNA examples again using only *in vitro* processes.

3.1. Learning with hybridization

We first consider the simplest algorithm, which is to classify an unknown example $|\mathbf{x}_i\rangle$ according to which of two class means is closer (Schölkopf and Smola, 2001). When the two class means are

$$\vec{c}_+ = \frac{1}{N_{+1}} \sum_{y_j=+1} |\mathbf{x}_j\rangle \quad \text{and} \quad \vec{c}_- = \frac{1}{N_{-1}} \sum_{y_j=-1} |\mathbf{x}_j\rangle, \tag{3}$$

the discriminating hyperplane is orthogonal to $\mathbf{w} = \vec{c}_{+1} - \vec{c}_{-1}$, and the label y_i of $|\mathbf{x}_i\rangle$ can be obtained using the inner product with $\mathbf{w}$:

$$\mathbf{w}^T|\mathbf{x}_i\rangle = \frac{1}{N_{+1}} \sum_{y_j=+1} \langle\mathbf{x}_j|\mathbf{x}_i\rangle + \frac{1}{N_{-1}} \sum_{y_j=-1} \langle\mathbf{x}_j|\mathbf{x}_i\rangle \tag{4}$$

$$= \frac{1}{N_{+1}} \sum_{y_j=+1} K_{ij} + \frac{1}{N_{-1}} \sum_{y_j=-1} K_{ij} \tag{5}$$

Since the amount of hybridization is proportional to the number of hybridizing molecules, we can monitor a mixture of conjugate molecules of training examples from class +1 as well as from class −1 to compute the inner product with the means. Molecules of an unknown example $|\mathbf{x}_i\rangle$ are hybridized, and the amount of double stranded hybridized product is measured. This measurement can be performed *in vitro* using hybridization-induced fluorescence, and the unknown molecular example is classified by the label with the most fluorescence calibrated by the relative amount of data N_+ and N_-.

This simple procedure shows an example of an *in vitro* classifier that can be interpreted in terms of computing kernel elements. However, this simple classifier uses only the class distribution means and does not incorporate knowledge about the full distribution of training examples. Such a classifier may not give the optimal discrimination boundaries and would also be susceptible to the presence of outliers in the training set. In the following section, we show how to design a more advanced *in vitro* process that can learn more complex discrimination boundaries using the DNA kernel matrices.

3.2. In vitro processes

We forge a two-phase process, where in the first phase discriminative information from training examples is learned, and the second phase determines the label of an unknown data example. The first *in vitro* process learns and stores discriminative information in the population of DNA molecules. In this phase, molecules of data examples and complementary sequences are hybridized, yielding two different kinds of double stranded molecules: double strands having differently labeled (hetero-labeled) molecules and double strands having identically labeled (homo-labeled) molecules. Our algorithm iteratively considers the population of hetero-labeled molecules $\langle\mathbf{x}_j|\mathbf{x}_i\rangle$ for $y_j \neq y_i$. In this population, the relative frequency of x_i molecules are modified according to the ratio between hetero-labeled molecules and all hybridized molecules, given by the quantity $\sum_{y_j \neq y_i} \langle\mathbf{x}_j|\mathbf{x}_i\rangle / \sum_{m=1}^{N} \langle\mathbf{x}_m|\mathbf{x}_i\rangle$. Each step of the learning process results in a population of hetero-labeled double-stranded molecules, which are then denatured, amplified, and used for the next hybridization reaction as shown in Fig. 3. We show that the relative distribution of data molecules $|\mathbf{x}_i\rangle$ in this population reaches a steady-state equilibrium that implicitly stores a set of weight vectors that can be used for classification.

The classification process uses the learned population from the first phase. An unknown data sequence is hybridized with molecules from this population. The class label of the DNA example is then determined by monitoring the relative amounts of hybridized molecules from each class. Similar to the discrimination algorithm which compared class means, this readout can be performed presumably via an *in vitro* fluorescence measurement.

3.3. Geometrical interpretation

The proposed learning process can be interpreted geometrically as finding a weight vector in the DNA feature space that is normal to the classification boundary. This weight vector is the difference between the two closest vectors $\mathbf{w}_{\{+1\}}$ and $\mathbf{w}_{\{-1\}}$ that are contained in the convex cones of the two classes as shown in Fig. 4. Here we prove that the proposed learning method converges to a population of DNA molecules representing this difference of the two closest vectors. The discrimination phase then classifies unknown DNA molecules by taking the inner product with this difference vector.

We represent the weight vector as $\mathbf{w} = \sum_{i=1}^{N} \alpha_i |\mathbf{x}_i\rangle$, where α_i is the relative concentration of sequence i compared to the initial population, and $|\mathbf{x}_i\rangle$ represents each datum vector in the feature space. The initial α_is are uniformly set to one, and a column vector $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_N\}^T$ is used to represent the population concentrations at a particular time.

Without loss of generality, the data are sorted with respect to their labels $y_l \in \{+1, -1\}$, $l = 1, \ldots, N$, and we separate $\boldsymbol{\alpha} \in \mathbb{R}^N$ into two vectors $\alpha_{\{c\}} \in \mathbb{R}^{N_c}$, for class $c \in \{+1, -1\}$ and $K \in \mathbb{R}^{N \times N}$ into four matrices $K_{\{c_1, c_2\}} \in \mathbb{R}^{N_{c_1}, N_{c_2}}$, $c_1, c_2 \in \{+1, -1\}$, where N_c is the number of data of class c, which satisfies $N_{c=+1} + N_{c=-1} = N$. The generation of double stranded DNA is bilinear in the number of single stranded components α_i and α_j, so that $\left|\mathrm{dsDNA}(j^*, i)\right| = \alpha_i \alpha_j K_{ij}$. The following theorem shows how the concentrations converge by

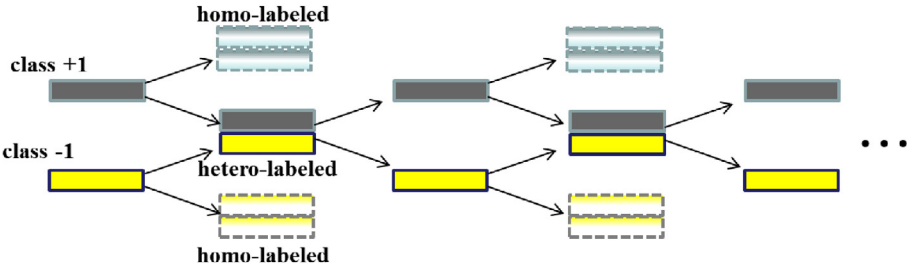

Fig. 3. Iterative update of population using hetero-labeled double-stranded molecules.

iteratively hybridizing and amplifying the hetero-labeled double stranded DNA molecules.

Theorem 1. *If the selection of hetero-labeled molecules from step t to t + 1 satisfies the following equation,*

$$\alpha_i^{t+1} = \alpha_i^t \left(\frac{\sum_{y_j \neq y_i} K_{ij} \alpha_i^t \alpha_j^t}{\sum_{m=1}^{N} K_{im} \alpha_i^t \alpha_m^t} \right) = \alpha_i^t \left(\frac{\sum_{y_j \neq y_i} K_{ij} \alpha_j^t}{\sum_{m=1}^{N} K_{im} \alpha_m^t} \right) \tag{6}$$

for $i \in \{1, \ldots, N\}$, *the asymptotic distribution of* $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_N\}^T$ *optimizes the following objective function:*

$$\max_{\boldsymbol{\alpha}_{\{i\}} \in \mathbb{R}^{N_i}, i=+1,-1} \frac{\mathbf{w}_{\{-1\}}^T \mathbf{w}_{\{+1\}}}{\|\mathbf{w}_{\{-1\}}\| \|\mathbf{w}_{\{+1\}}\|} \tag{7}$$

$$s.t. \quad \mathbf{w}_{\{-1\}} = \langle \mathbf{x}_{\{-1\}} | \boldsymbol{\alpha}_{\{-1\}},$$

$$\mathbf{w}_{\{+1\}} = \langle \mathbf{x}_{\{+1\}} | \boldsymbol{\alpha}_{\{+1\}}, \quad for \quad \boldsymbol{\alpha}_{\{-1\}}, \boldsymbol{\alpha}_{\{+1\}} \geq 0.$$

Here, $|\mathbf{x}_{\{\mathbf{c}\}}\rangle$, $c \in \{+1, -1\}$ *is a* $f \times N_c$ *matrix satisfying* $K_{\{\mathbf{c_1},\mathbf{c_2}\}} = \langle \mathbf{x}_{\{\mathbf{c_1}\}} | \mathbf{x}_{\{\mathbf{c_2}\}} \rangle$ *where f is the dimension of the feature space,* $\boldsymbol{\alpha}_{\{\mathbf{c}\}}$ *is a column vector of dimension* N_c, *and the matrix* $\langle \mathbf{x}_{\{\mathbf{c}\}}|$ *is the transpose of* $|\mathbf{x}_{\{\mathbf{c}\}}\rangle$.

Proof. From Eq. (6), we can consider the update rule as equivalent to the power method of finding the eigenvector of the largest eigenvalue for the following generalized eigenvalue problem (Zwillinger, 1996):

$$\begin{pmatrix} 0 & K_{\{-1,+1\}} \\ K_{\{+1,-1\}} & 0 \end{pmatrix} \boldsymbol{\alpha} = \lambda \begin{pmatrix} K_{\{-1,-1\}} & 0 \\ 0 & K_{\{+1,+1\}} \end{pmatrix} \boldsymbol{\alpha} \tag{8}$$

Here, $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_{\{-1\}}^T \ \boldsymbol{\alpha}_{\{+1\}}^T]^T$. To show the equivalence between Eqs. (7) and (8), we first reformulate Eq. (7) as maximization of the objective function L with Lagrange multipliers $\lambda_{\{+1\}}$ and $\lambda_{\{-1\}}$:

$$\begin{aligned} L &= \mathbf{w}_{\{-1\}}^T \mathbf{w}_{\{+1\}} - \frac{1}{2}\lambda_{\{-1\}}(1 - \mathbf{w}_{\{-1\}}^T \mathbf{w}_{\{-1\}}) - \frac{1}{2}\lambda_{\{+1\}}(1 - \mathbf{w}_{\{+1\}}^T \mathbf{w}_{\{+1\}}) \\ &= \boldsymbol{\alpha}_{\{-1\}}^T \langle \mathbf{x}_{\{-1\}} | \mathbf{x}_{\{+1\}} \rangle \boldsymbol{\alpha}_{\{+1\}} - \frac{1}{2}\lambda_{\{-1\}}(1 - \boldsymbol{\alpha}_{\{-1\}}^T \langle \mathbf{x}_{\{-1\}} | \mathbf{x}_{\{-1\}} \rangle \boldsymbol{\alpha}_{\{-1\}}) \\ &\quad - \frac{1}{2}\lambda_{\{+1\}}(1 - \boldsymbol{\alpha}_{\{+1\}}^T \langle \mathbf{x}_{\{+1\}} | \mathbf{x}_{\{+1\}} \rangle \boldsymbol{\alpha}_{\{+1\}}) \\ &= \boldsymbol{\alpha}_{\{-1\}}^T K_{\{-1,+1\}} \boldsymbol{\alpha}_{\{+1\}} - \frac{1}{2}\lambda_{\{-1\}}(1 - \boldsymbol{\alpha}_{\{-1\}}^T K_{\{-1,-1\}} \boldsymbol{\alpha}_{\{-1\}}) \\ &\quad - \frac{1}{2}\lambda_{\{+1\}}(1 - \boldsymbol{\alpha}_{\{+1\}}^T K_{\{+1,+1\}} \boldsymbol{\alpha}_{\{+1\}}) \end{aligned} \tag{9}$$

The derivatives of L with respect to $\boldsymbol{\alpha}_{\{-1\}}$ and $\boldsymbol{\alpha}_{\{+1\}}$ are zero at the maximal points:

$$\left(\frac{\partial L}{\partial \boldsymbol{\alpha}_{\{-1\}}} \right)^T = K_{\{-1,+1\}} \boldsymbol{\alpha}_{\{+1\}} - \lambda_{\{-1\}} K_{\{-1,-1\}} \boldsymbol{\alpha}_{\{-1\}} = 0 \tag{10}$$

$$\left(\frac{\partial L}{\partial \boldsymbol{\alpha}_{\{+1\}}} \right)^T = K_{\{+1,-1\}} \boldsymbol{\alpha}_{\{-1\}} - \lambda_{\{+1\}} K_{\{+1,+1\}} \boldsymbol{\alpha}_{\{+1\}} = 0 \tag{11}$$

If we compare Eqs. (10) and (11) after multiplying $\boldsymbol{\alpha}_{\{-1\}}^T$ to the left of (10) and $\boldsymbol{\alpha}_{\{+1\}}^T$ to the left of (11), we see $\lambda_{\{-1\}} = \lambda_{\{+1\}}$ because we constrained $\mathbf{w}_{\{+1\}}^T \mathbf{w}_{\{+1\}} = \mathbf{w}_{\{-1\}}^T \mathbf{w}_{\{-1\}} = 1$. Eq. (8) becomes equivalent to (10) and (11), yielding the result $\lambda = \lambda_{\{-1\}} = \lambda_{\{+1\}} = \mathbf{w}_{\{-1\}}^T \mathbf{w}_{\{+1\}}$. Thus, the principal eigenvalue λ becomes the maximum value of $\mathbf{w}_{\{-1\}}^T \mathbf{w}_{\{+1\}}$, completing the proof. □

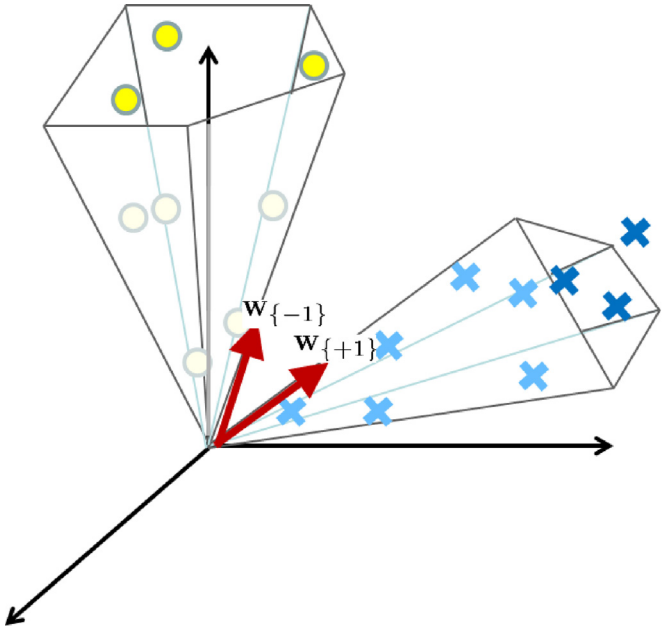

Fig. 4. Iterative updates of the hetero-labeled DNA molecules finds the two closest vectors in the feature space which are contained in convex cones of different classes.

We consider the matrix $|\mathbf{x}_{\{\mathbf{c}\}}\rangle$ in Theorem 1 as the collection of column vectors $|\mathbf{x}_i\rangle$ in class c. Then, the optimized vectors $\mathbf{w}_{\{-1\}}$ and $\mathbf{w}_{\{+1\}}$, become the closest vectors where $\mathbf{w}_{\{-1\}}$ is contained within the positive cone of class −1, and $\mathbf{w}_{\{+1\}}$ is contained within the positive cone of class +1 as shown in Fig. 4.

The discriminative *in vitro* phase is then interpreted as determining whether a new test datum $|\mathbf{x}_{new}\rangle$ is closer to $\mathbf{w}_{\{-1\}}$ or $\mathbf{w}_{\{+1\}}$ in angle. The new DNA molecules are first hybridized with the learned distribution of molecules, resulting in double stranded DNA concentrations given by $\alpha_i K(new, i)$. Comparison of the amounts of hybridized molecules is equivalent to determining whether $\sum_{y_i=+1} \alpha_i K(new, i) = \mathbf{w}_{\{+1\}}^T |\mathbf{x}_{new}\rangle$ or $\sum_{y_i=-1} \alpha_i K(new, i) = \mathbf{w}_{\{-1\}}^T |\mathbf{x}_{new}\rangle$ is larger. This can be viewed as computing the inner

product of the new test example with the vectors $\mathbf{w}_{\{+1\}}$ and $\mathbf{w}_{\{-1\}}$, giving the binary class label:

$$y_{new} = \text{sign}\left\{\sum_i \alpha_i y_i K(new, i)\right\} = \text{sign}\{\mathbf{w}_{\{+1\}}^T|\mathbf{x}_{new}\rangle - \mathbf{w}_{\{-1\}}^T|\mathbf{x}_{new}\rangle\}. \quad (12)$$

The classification decision boundary is given by an hyperplane orthogonal to the difference vector $\mathbf{w}_{\{+1\}} - \mathbf{w}_{\{-1\}}$.

We have explained the learning and classification process on DNA molecules using simple *in vitro* operations. In the following section, we verify that positive definite kernel matrices can be properly formed using thermodynamic models of hybridization.

4. Hybridization kernel and positive definiteness

When hybridized results are regarded as kernel elements for classification, a positive definiteness (Schölkopf and Smola, 2001) or at least a near-positive definiteness (Haasdonk, 2005; Ong et al., 2004) of the resulting matrix should be achieved. In this section, we discuss the positive definiteness of the proposed kernel using a kinetic model of hybridization with thermodynamics incorporated. The kinetic model provides predictions for how the different kernel matrix coefficients change as a function of temperature, and we show how an appropriate positive definite kernel matrix can be formed under an annealed temperature schedule.

We consider hybridization between a primary sequence molecule $|\mathbf{x}_i\rangle$ and a complementary molecule $\langle\mathbf{x}_j|$. When $|\mathbf{x}_i\rangle$ and $\langle\mathbf{x}_j|$ hybridize to make a double strand, the process is considered to be a stochastic process of continuous hybridization and denature with transition probabilities p_h and p_d. These transition probabilities are governed by the hybridization energy and entropy changes, written as ΔE (kcal) and ΔS (kcal/K) respectively, as well as by the temperature T (K), and we can understand hybridization a Monte Carlo Markov Chain (MCMC) process with the transition probabilities (Kim et al., 2008),

$$\begin{aligned} p_h &= \begin{cases} \exp(-G/c_k T), & G \equiv \Delta E - T\Delta S \geq 0 \\ 1, & G \equiv \Delta E - T\Delta S < 0 \end{cases} \\ p_d &= \begin{cases} 1, & G \equiv \Delta E - T\Delta S \geq 0 \\ \exp(G/c_k T), & G \equiv \Delta E - T\Delta S < 0. \end{cases} \end{aligned} \quad (13)$$

When two particular single stranded molecules of the same amount hybridize, this MCMC process will make a Boltzmann distribution at equilibrium: the ratio between the concentration of double strands and the square of single strands will be proportional to $\exp\{-(\Delta E - T\Delta S)/c_k T\}$ with a Boltzmann constant $c_k = 3.2982 \times 10^4$ (kcal/K). However, in a modeling of complex hybridization with many different molecules, the final distribution does not follow the Boltzmann distribution but stays in a different equilibrium. In this situation, the kinetics of hybridization are more important for explaining the population at equilibrium.

Previously, Britten and Kohne (1968) found that the amount of hybridization is not primarily governed by the thermodynamics but follows the similar kinetics regardless of the kinds of binding, once the temperature is below a certain level. This is because the frequency of collision becomes more important than the binding probability itself, if the binding probability is high enough. In their analysis on hybridization, it is claimed that the population of single stranded molecules C changes over time t when the initial concentration of single stranded molecules is C_0:

$$\frac{C}{C_0} = \frac{1}{1 + \alpha C_0 t}. \quad (14)$$

Here, the estimation results showed that the constant α is about the same over various DNAs. Now, this equation can be approximated with small t considering that the majority of the population change occurs at the early stage when C is high:

$$C \approx C_0 - \alpha C_0^2 t. \quad (15)$$

From this equation, we can see that the amount of hybridized molecules is approximately proportional to the hybridization time and the square of the initial concentration of single stranded molecules, while the hybridization speed is not governed by the hybridization energy or entropy. Therefore, we can make a hybridization model where the amount of hybridization is proportional to the time of hybridization, when the temperature is within a regime where hybridization can occur. From Eq. (13), we assume that the regime is the temperature below $T_\theta \equiv \Delta E/\Delta S$.

Now, we consider hybridization with a cool-down schedule from a high temperature T_i to a low temperature T_f with a constant speed. Once the temperature passes the threshold temperature $T_\theta \equiv \Delta E/\Delta S$, the hybridization begins. Upon these settings, the amount of hybridized molecules can be approximated as proportional to the following terms:

$$\langle\mathbf{x}_j|\mathbf{x}_i\rangle \propto [T_\theta(i,j) - T_f]_+ - [T_\theta(i,j) - T_i]_+ \quad (16)$$

Here $[\cdot]_+ = \max[0, \cdot]$ is the rectification nonlinearity and the threshold temperature for a pair of DNA molecules $|\mathbf{x}_i\rangle$, and $\langle\mathbf{x}_j|$ is given by $\Delta E_{ij} - T_\theta(i,j)\Delta S = 0$ with binding energy ΔE_{ij} and entropy change ΔS.

Eq. (16) can be interpreted to mean that the speed of accumulation of double strands is assumed to be zero when the current temperature is higher than the threshold temperature T_θ, and the speed is constant when the temperature is lower than T_θ. In this case, the hybridized amount is proportional to the hybridizing time, and the hybridizing time is again proportional to either the difference $T_\theta - T_f$ or $T_i - T_f$, whichever is smaller, when the final temperature T_f is smaller than both T_θ and T_i. For example, when the initial temperature T_i is greater than the threshold temperature T_θ, Eq. (16) denotes $T_\theta - T_f$, while the initial temperature T_i, which is smaller than the threshold temperature T_θ, changes Eq. (16) to $T_i - T_f$. Eq. (16) also considers the situation where the final temperature T_f is not less than T_θ, which produces a zero amount. From this consideration, we obtained the amount of each double strand Eq. (16) with initial temperature T_i and final temperature T_f.

Intuitively, the annealed hybridization is guaranteed to make a positive definite kernel at the first stage when the temperature is still higher than T_θ for any non-complementary hybridization. In this case, the perfect complementary bindings will produce a diagonal matrix where the diagonal elements simply become the eigenvalues. However, we doubt whether the matrix will keep the positive definiteness after non-diagonal components start to accumulate. The next discussion will provide more evidence of how hybridization using a cooling schedule will make a positive definite matrix.

Because the kernel element is a function of the threshold temperature for binding T_θ, we can represent the kernel elements using the binding energies. For simplicity of the analysis, we use a previous work showing that the hybridization energy and entropy are simply well-described in terms of three different binding energies. These three binding energies are between A and T, or ΔE_{AT}, between C and G, or ΔE_{CG}, and the binding energy for other pairs, or ΔE_{Oth} (Kim et al., 2008). Table 1 shows the experimentally determined values for these binding energies.

Table 1
Binding energy of the individual pairs (kcal/mole base pair (MBP)).

C-G &G-C	$\Delta E_{CG} = -9.0$
A-T &T-A A-G &G-A	$\Delta E_{AT} = -7.2$
A-C &C-A T-G &G-T T-C &C-T	$\Delta E_{Oth} = -5.3$

Using these binding energies, we show that a sufficient condition for the matrix K to be positive definite. In the following theorem, we consider hybridization of DNA molecules of length l, where all sequences share a common set of m nucleotides.

Theorem 2. *For sequences of length l, having length r of variable nucleotides and length m of a common nucleotide fragment ($l = r + m$), and when common nucleotides have an average binding energy of ΔE_g, the matrix K from Eq. (16) is positive semi-definite if T_i and T_f satisfy $T_i > -(r\Delta E_{CG} + m\Delta E_g)/\Delta S$ and $T_f < -(r\Delta E_{Oth} + m\Delta E_g)/\Delta S$.*

Proof. We prove the positive semi-definiteness of matrix K by proving the positive semi-definiteness of each matrix from the two $[\cdot]_+$s in Eq. (16) separately. First, the term $T_\theta - T_f$ can be represented as

$$\frac{1}{\Delta S}\left\{-\Delta E_{ij}^r + r\Delta E_{Oth} - r\Delta E_{Oth} - m\Delta E_g - T_f\Delta S\right\}, \tag{17}$$

where ΔE_{ij}^r is the binding energy of varying sequences of length r satisfying $\Delta E_{ij} = \Delta E_{ij}^r + m\Delta E_g$. We see that the matrix $-\Delta E^r + r\Delta E_{Oth}\mathbf{1}_N\mathbf{1}_N^T$, where each element is the first two terms in (17), is positive definite. Here, $\mathbf{1}_N$ is a column vector whose N elements are all unitary. Consequently, $-\Delta E/\Delta S - T_f\mathbf{1}_N\mathbf{1}_N^T$ is positive definite when the other terms in (17), $-(r\Delta E_{Oth} + m\Delta E_g + T_f\Delta S)$ are positive, which gives the condition T_f: $T_f < -(r\Delta E_{Oth} + m\Delta E_g)/\Delta S$. Second, if $-\Delta E_{ij}/\Delta S - T_i$ in the second $[\cdot]_+$ term are all negative, then the matrix remains positive definite. This yields the additional condition $T_i > -(r\Delta E_{CG} + m\Delta E_g)/\Delta S$. □

Theorem 2 shows that positive definiteness of the kernel matrix is ensured if it is annealed from a temperature higher than $-r\Delta E_{CG} + m\Delta E_g/\Delta S$ to a temperature lower than $-r\Delta E_{Oth} + m\Delta E_g/\Delta S$. These bounds are also controlled by the average binding energy ΔE_g of a common nucleotide fragment. This value ΔE_g is always between ΔE_{CG} and ΔE_{AT}, and the presence of common nucleotides always decreases the lower bound on the initial temperature T_i and increases the upper bound on the final temperature T_f.

For example, if we consider molecular sequences of length $l = 100$, and if they have common nucleotide fragments of 50 '*A*'s and 20 '*T*'s, then $\Delta E_g = -7.79$ (kcal) according to Table 1, and $m = 70$ and $r = 30$. We also use $\Delta S = c_s l$ where $c_s = 0.023$ (kcal/K) for the entropy calculation (Kim et al., 2008). In this case, according to Theorem 2, the kernel satisfies positive definiteness if $T_i > (9 \times 30 + 7.79 \times 70)/(0.023 \times 10)(\mathrm{K}) = 81.33\,(^\circ\mathrm{C})$ and $T_f < 33.07\,(^\circ\mathrm{C})$, which is easily satified during the experimental DNA hybridization process.

More complex DNA binding models relax the assumption of independent binding energies per base pair. For example, a context-dependent binding method can be used which considers consecutive pairs of neighborhood sites (Santalucia, 1998). Even with these more complex models, analytic calculations show that the *in vitro* DNA kernel matrix will remain positive definite if the proper annealing schedule is followed.

5. Experimental results

In this section, we present several simulation experiments to show how the proposed algorithm can perform classification tasks. In the simulation, we first see that positive definiteness can be achieved by using the appropriate temperature schedule. Then we see how the kernel can change its sparseness according to different temperature schedules. Second, we simulate our learning algorithm with real biological datasets and compare the results against standard machine learning algorithms. The performance of the proposed algorithm is compared to the well-known kernel classification algorithms, SVMs and kernel Fisher discriminant analysis (kFDA), using the same kernel values.

In addition to the simulation results, we also present a biomolecular DNA experiment implementing the DNA kernel and hetero-labeled molecular selection. The experimental results are obtained using one particular set of specially designed DNA sequences.

5.1. Thermodynamic simulation

For the simulation of hybridization processes, it is common to incorporate a dynamics equation representing the rate of change in the amount of molecules (Gillespie, 1977, 2007; Kim et al., 2006). In our first experiment, we incorporate the hybridization of strands $|\mathbf{x}_i\rangle$, $i = 1, \ldots, N$ and $\langle\mathbf{x}_j|$, $j = 1, \ldots, N$ using the rate equation in Eq. (18) and check whether the annealing with cooling schedule helps generate a positive definite kernel.

$$\begin{aligned}\frac{\partial|\mathbf{x}_i|}{\partial t} &= \sum_{j=1}^{N}\{k_d(i,j)\langle\mathbf{x}_j|\mathbf{x}_i\rangle - k_h(i,j)|\mathbf{x}_i||\mathbf{x}_j|\}\\ \frac{\partial\langle\mathbf{x}_j|\mathbf{x}_i\rangle}{\partial t} &= k_h(i,j)|\mathbf{x}_i||\mathbf{x}_j| - k_d(i,j)\langle\mathbf{x}_j|\mathbf{x}_i\rangle\end{aligned} \tag{18}$$

Here, $\left|\mathbf{x}_i\right|$ and $\langle\mathbf{x}_j|\mathbf{x}_i\rangle$ represent the amount of single and double stranded molecules, respectively, where $\left|\mathbf{x}_i\right|$ is the amount of $|\mathbf{x}_i\rangle$, and $\langle\mathbf{x}_j|\mathbf{x}_i\rangle$ is the amount of double-stranded molecules of $|\mathbf{x}_i\rangle$ and $\langle\mathbf{x}_j|$. The rate constants $k_h(i,j)$ and $k_d(i,j)$ are proportional to the hybridization probability p_h and the denaturing probability p_d in Eq. (13), determined by the hybridization energy and entropy of the sequence i and the complementary sequence j, as well as the temperature.

The system from Eq. (18) is nonlinear, where the population distribution can arrive more than one stationary distribution. Because k_d and k_h are functions of the temperature, the final distribution of single and double stranded molecules is determined by the schedule of the temperature during hybridization. Here, we generated random six sequences and their complementary sequences and tested how annealing process, proposed in Section 4, helps generate a positive definite kernel. In Fig. 5, two different annealing schedules are used in Fig. 5(a) and (c), and the construction curves of one diagonal (K_{11}) element and one off-diagonal (K_{12}) element are presented. When the annealing started from a higher temperature as in Fig. 5(a), the difference between the diagonal and off-diagonal elements is much larger than the annealing in Fig. 5(c), producing diagonal dominant kernel matrices. In this case, the matrix tends to be positive definite as shown in Fig. 5(b), which can be compared with the result with non-sufficient annealing in Fig. 5(d) representing a non-positive definite matrix.

In Fig. 5(e) and (f), temperature schedule is the same as the experiments in Fig. 5(a) and (b), while the less length of common sequence is used. While the matrix is positive definite in this case, the off-diagonal elements become too sparse as in Fig. 5(e), producing a non-informative kernel matrix, where every pair of data are orthogonal in the feature space.

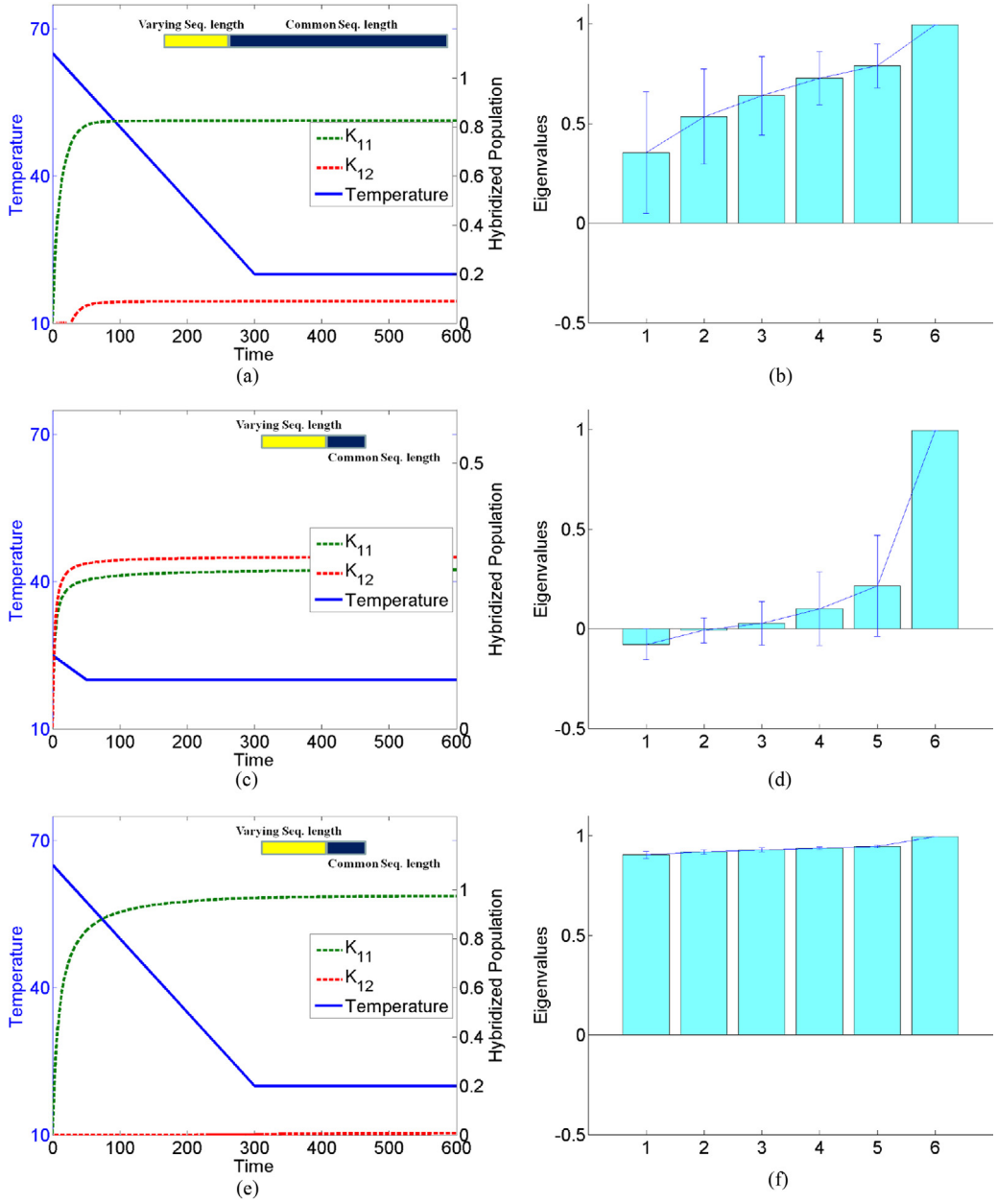

Fig. 5. Simulation of kernel values. Six random primary sequences are hybridized together with their complementary sequences using Eq. (18). The primary sequences include a common sequence where the relative length is also represented. In (a), (c), and (e), the temperature schedules and the resulting K_{11} and K_{12} are plotted over time for one example of random sequence. In (b), (d), and (f), the mean and the covariance of 6 eigenvalues of generated kernels are shown for experiments in (a), (c), and (e), respectively. In (c) and (d), annealing is not sufficient, and in (e) and (f), less common sequence is used than (a) and (b).

Once a positive definiteness can be achieved by annealing, a sparseness of the kernel can be further controlled by modifying the temperature schedule. Conventional kernels tune their sparsity using kernel parameters, such as the width parameter in a Gaussian kernel. In our proposed kernel, we can use the temperature schedule to tune the kernel as the width parameter does. For example, a sparse matrix can be generated by keeping the temperature high through hybridization. When the temperature is higher than a particular temperature, only perfectly complementary sequences can hybridize, so that the diagonal coefficients in the kernel matrix dominate.

Another method for controlling sparseness is to use common patterns in the DNA sequences as discussed with the results in Fig. 5(e) and (f). Inserting common subsequences encourage pairs of DNA sequences to hybridize more, resulting in a less sparse kernel matrix.

5.2. Classification performance

Different temperature schedules produce different kernel values. In our classification results with the generated data shown in Fig. 6, points in a two-dimensional space are labeled into two classes shown in yellow and blue color. In this space, the binding energy is given by the Euclidean distance between pairs of points. The contours represent various hybridization amounts, and change according to the annealing temperature schedules. This shows how controlling the hybridization schedule influences both the positive definiteness and sparsity of the resulting kernel matrices.

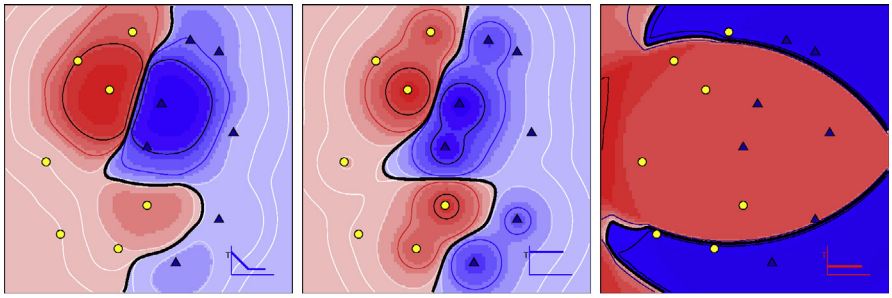

Fig. 6. Two-class data are distributed in a two dimensional space and labeled as yellow and blue. Binding energy between pairs of points are determined by the Euclidean distance and scaled so that the energy is within the range of −56 to −84(kcal). The length of binding nucleotides is set to $l = 10$ for the entropy calculation. The data are learned with different temperature schedules; test results across the data space is presented as red and blue for the yellow and blue classes respectively. Temperatures for hybridization is (a) 80 °C to 20 °C, (b) 80 °C constant, and (c) 30 °C constant. The kernel in (c) does not satisfy positive definiteness. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

With sufficient annealing as shown in Fig. 6(a), the kernel satisfies positive definiteness. In Fig. 6(c) with no annealing, the kernel does not satisfy positive definiteness, resulting in bad classification results. With high temperature hybridization in Fig. 6(b), the kernel matrix is positive definite but very diagonally dominant and sparse. In this case, the hybridization contours show that the decision surface depends more specifically on nearest neighbors as compared to the decision surface in Fig. 6(a). Such a sparse kernel matrix would be more vulnerable to noise in the training data.

Algorithm 3. Learning with *in vitro* kernels

Input: Sequences $|\mathbf{x}_i\rangle$ and complementary sequences $\langle\mathbf{x}_i|$ for all $i \in \{1, \ldots, N\}$, and constants T_f, T_i, c_s, and ϵ.

Initialize $K_{ij} = \left[-\frac{\Delta E_{ij}}{c_s l} - T_f\right]_+ - \left[-\frac{\Delta E_{ij}}{c_s l} - T_i\right]_+$, $\alpha_i^{t=0} = 1$, and $t = 0$

repeat

$\alpha_i^{t+1} = \alpha_i^t \left(\sum_{y_j \neq y_i} K_{ij}\alpha_j^t / \sum_{m=1}^{N} K_{im}\alpha_m^t\right)$

$t = t + 1$

until $\sum_i (\alpha_i^{t+1} - \alpha_i^t)^2 < \epsilon$

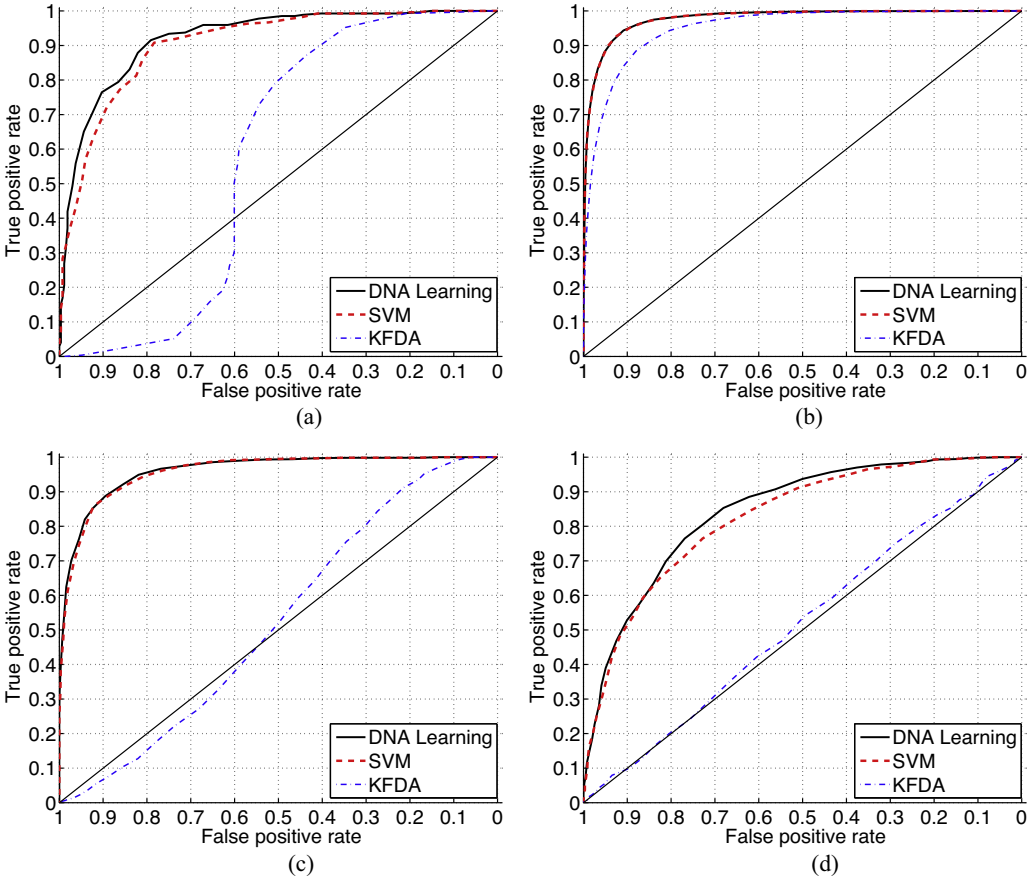

Fig. 7. ROC (Receiver Operating Characteristic) curves. The classification is performed using our DNA learning, SVMs and kernel FDA using the same DNA kernel. The FDA criterion is not in general appropriate in this data. Interestingly, the proposed DNA learning method outperforms or is similar to SVMs in all cases. The slack variable parameter in SVMs and the regularization parameter in FDA are scanned and optimized. In our DNA learning, the algorithm does not have any tunning parameters.

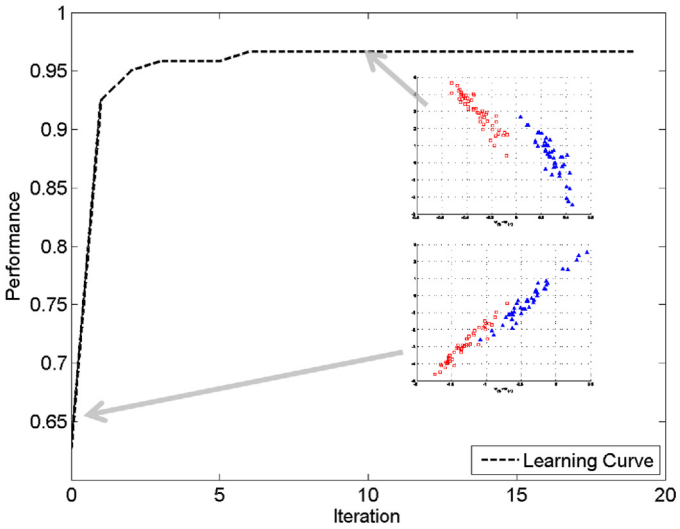

Fig. 8. Learning curve (test performance) of AML/ALL data with respect to algorithm iteration and the 2-dimensional embedding of training data on the feature space. Horizontal axis of the insets represents the vector direction $\mathbf{w}_{\{+1\}} - \mathbf{w}_{\{-1\}}$. Vertical axis is the maximum variance direction of the orthogonal space to the discriminating direction.

We next evaluate the performance of the learning algorithm, summarized in Algorithm 3, on several benchmark datasets. One dataset contains gene expression data collected from microarray experiments for discrimination of acute myelogenous leukemia (AML) and acute lymphoblastic leukemia (ALL) (Cheok et al., 2003). The microarray data are preprocessed, and 1000 genes out of 12,600 genes are used which show maximum mutual information between expression levels and labels. Each gene expression value is simplified by binary thresholding, and are encoded as 'A' and 'G' respectively for our experiment. Data are thus expressed as strings of length 1000. We used the kernel defined in Equation (16) using $T_i = 90^\circ$C and $T_f = 20^\circ$C. The results are the average of 5-fold cross-validation. At each validation, 24 different examples are reserved for testing, while the remaining 96 examples are trained.

The other dataset contains DNA sequences that need to be classified according to whether or not they contain a splice region. Three different problems are contained in this set, concerning the classification of expressed sequence tag (EST) in *C. elegans*, EST in Drosophila, and synthesized sequences, with the details of data collection and processing methods described in Rätsch et al. (2006). In general, the number of negative samples is larger than positive samples, and we used all the positive data and the same number of random negative data. For the C.elegans set, the 15,507 positive samples are divided into 15 non-overlapping subsets randomly containing 150 training and 700 testing samples. The Drosophila set of 1583 positive samples is divided into 3 distinct subsets randomly containing 100 training samples and 400 testing samples. The synthetic set has 95 positive training samples and 905 positive testing samples. Among the synthetic sets in Rätsch et al. (2006), we used the sequences where five symbols are randomly replaced.

The classification performance of the algorithm is compared with the conventional SVM and kFDA classifier. The slack variable parameter for the SVM algorithm and the regularization parameter for kFDA are optimized using cross-validation. The ROC curves (Fig. 7) shows that the classification performance of our proposed method is superior to kFDA and performs better than the SVM algorithm on the AML/ALL, Drosophila, and synthetic sequences.

We also present a sample learning curve for the AML/ALL data in Fig. 8. Within a few iterations of the learning algorithm, the test performance increases sharply indicating that the algorithm converges rapidly enough for practical implementation. The resulting embedding in feature space shows that the performance increase is indeed due to increasing the margin of the training data.

6. Biomolecular implementation

The implementation of the algorithm on actual biomolecular DNA molecules requires concatenation of various DNA operations. Such operations can include hybridization, denaturing, polymerase chain reaction (PCR, for amplification), and gel electrophoresis (for selection). In this section, we present one example of preliminary implementation using these operations of *in vitro* classification system. The experiments will show the possibility of implementation in the future, where the actual implementation needs additional complex constraints of DNA operational design.

6.1. Implementation procedure

One example of a real implementation of the proposed algorithm is presented in this section. First, we start by mixing all the oligonucleotides with data and their complementary sequences. In order to separate the homo-labeled double stranded molecules from the hetero-labeled double stranded molecules, we can use different lengths of molecules for the different classes. In Table 2, we show our example of DNA coding with six data of 21 mer, added by a common sequence of 16 mer which is divided into 8 mer on both sides, as well as a dummy sequence on the 3′ side to control the total length of the molecule. Here, three data are labeled as class +1 and the other three as class −1, and the total lengths of different classes are differentiated to be 46 mer and 54 mer respectively. The sequences are expected to be hybridized as shown in Fig. 9.

Hybridizing between these molecules will produce the homo-labeled double strands in class +1, homo-labeled strands in class −1, and hetero-labeled double strands between class +1 and class −1, all having different masses and lengths. After hybridization, we will have three different double-stranded molecules with three different masses. Now, we separate the double strands into three different bands using gel electrophoresis. The band including the hetero-labeled double strands can be collected and amplified for subsequent iterations of the learning algorithm.

In this coding example, a common sequence of 16 mer is added to all primary sequences (sequence 1, 2, and 3) and the complementary of the common sequence is added to all complementary sequences (sequence 1-C, 2-C, and 3-C) at the same location. The examples of common sequences and data sequences are presented and marked in Fig. 9 for the sequence 1 and the sequence 4-C. In our analysis, we showed that recruiting a common sequence allows the system to improve the reliability of the results from various perspectives. First, mismatch of sequences is less likely with the increase of common sequences. This property can be easily confirmed by a hybridization analyzer, such as NUPACK (Zadeh et al., 2011). However, because the cost increases with the length of sequence, the incorporation of a long common sequence is restricted in reality. Second, the common sequence helps make the matrix be positive definite as shown in Theorem 2 and Fig. 5 from the classification accuracy perspective. The dummy code of 9 mer on the 3′ side of class +1 and 17 mer of the class −1 also can be designed as a sticky end, where there is room to design for more selection operations using these dummy sequences.

After enough hybridization, hetero-labeled double strands are selected to update the population as in Theorem 1. In Eq. (6), the total number of double strands containing one particular sequence corresponds to the denominator and the number of hetero-labeled double strands among them corresponds to the nominator. Therefore, iterative hetero-labeled selection will lead the distribution having the weight in Theorem 1. After the selection, an

Table 2
Six different data are represented using twelve kinds of molecules. Each pair *i* and *i*-C represent one datum containing the data sequence and its complementary sequence.

Class	Index	Sequence
Class +1 (46mer length)	1	5′AGCAGACTTTAATGTTAATGTTATTATTACTACATCGCACGACTGA3′
	1-C	5′CGATGTAGTAATAATAACATTAACATTAAAGTCTGCTCACGACTGA3′
	2	5′AGCAGACTATGATGTTATTATTATTATTACTACATCGCACGACTGA3′
	2-C	5′CGATGTAGTAATAATAATAACATCATAGTCTGCTCACGACTGA3′
	3	5′AGCAGACTATGATGTTATTAATGTTATTACTACATCGCACGACTGA3′
	3-C	5′CGATGTAGTAATAACATTAATAACATCATAGTCTGCTCACGACTGA3′
Class −1 (54mer length)	4	5′AGCAGACTTTATTATTAATGATGTTAATGCTACATCGACAGCAAGCACGACTGA3′
	4-C	5′CGATGTAGCATTAACATCATTAATAATAAAGTCTGCTACAGCAAGCACGACTGA3′
	5	5′AGCAGACTTTAATGTTATTAATGATGTTACTACATCGACAGCAAGCACGACTGA3′
	5-C	5′CGATGTAGTAACATCATTAATAACATTAAAGTCTGCTACAGCAAGCACGACTGA3′
	6	5′AGCAGACTTTATTATTAATGATGATGATGCTACATCGACAGCAAGCACGACTGA3′
	6-C	5′CGATGTAGCATCATCATCATTAATAATAAAGTCTGCTACAGCAAGCACGACTGA3′

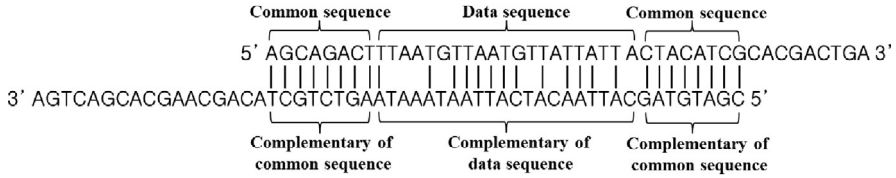

Fig. 9. Expected hybridization between molecules 1 and 4-C in Table 2.

amplification process such as PCR can be used to avoid losing the molecules. In this process, we do not have to consider a specific gain, but it is enough to keep the ratio of the population because the classification result is unaffected once the ratio is kept.

In the discriminative phase, when we use the fluorescence intensity of two different binded fluorophores to measure the relative amount of hybridized strands, we should make sure that the distance is fixed between the fluorophore and the quencher within a hybridized molecule. Otherwise, the intensity is affected by the distance, and the intensity could not reliably measure the number of double strands. In our method, we considered only the specific hybridization as in Fig. 9 and tried to disallow any hybridization other than at a pre-determined position by adopting a complementary common sequence throughout all sequences. By allowing only the pre-defined hybridization configurations, we can know in advance the relative fluorescence intensities of different classes when they have the same number of hybridizations, and we can accordingly calibrate the reading of fluorescence intensity of each class.

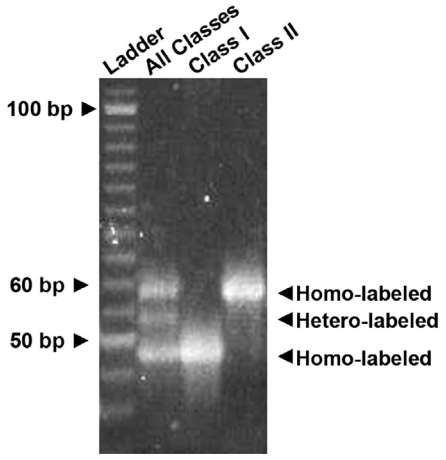

Fig. 10. Gel electrophoresis results from mixtures in Table 2. This figure shows the results of three experiments; the first one (second lane) shows the result of all sequence mixtures, and the second and third experiments (third and fourth lanes) are the results of class +1 molecules and class −1 molecules, respectively.

6.2. Hybridization implementation

In Fig. 10, we show the gel electrophoresis result with the hybridized molecules using all sequences presented in Table 2. Here, all species of the oligonucleotide were mixed with the same concentration in a 1.5 mL microcentrifuge tube (final 60 μM each under 0.5 M NaCl salt conditions). After heating the mixture at 95 °C for 3 minutes, we gradually cooled the temperature down in a heat block, after which the tube was incubated overnight with constant rotation (60 rpm) in a 37 °C incubator. Five microliter products were mixed with 1 μl of 6× loading buffer, and the mixture was loaded into a Spreadex EL300 gel (Elchrom Scientific, Switzerland). Electrophoresis was performed at 120V for 3 h in a SEA 2000 electrophoresis apparatus (Elchrom Scientific). Temperature of the running buffer (TAE) was kept constant at 20 °C. The gel was stained with SYBR Gold, and visualized using Bio-Rad gel doc 2000 (Bio-Rad, USA).

We also estimated the quantity of molecules in the three bands. The average and standard deviation for 15 trials are presented in Table 3. As expected, when we consider the resulting 2×2 matrix representing the amount of double strands of four possible class combinations of the primary and complementary strands, the matrix for this experiment is positive definite as a necessary condition for the positive definiteness of the kernel.

Another expectation is the linear relationship of the amount with respect to the binding energy. In Section 4, we explained that the distribution of the differently hybridized molecules will not follow the Boltzmann distribution with many molecules. According to our kinetic model, the amount of molecules will be proportional to the energy at first, and the distribution will stay in one nearby local attractor distribution at equilibrium. Here, the equilibrium distribution can be any attractor making all equations on the right hand side in Eq. (18) be zero. Therefore, the final distribution is

Table 3
Hybridized amount (IDV).

Homo (class +1)	209.2 ± 13.8
Hetero	200.5 ± 15.1
Homo (class −1)	206.4 ± 15.0

close to linear rather than exponential which is the case when the population follows a Boltzmann distribution.

7. Conclusions

We introduced a biomolecular algorithm for learning with DNA moleculesInterpreted within the context of kernel machines, the resulting algorithm can be viewed as learning a weight vector associated with the convex cones of the data in the associated feature space. Simulations show that the proposed molecular learning algorithm is competitive with state-of-the-art machine learning algorithms.

We also showed the possibility of exploiting the algorithm to real *in vitro* implementation. In particular, the thermodynamic simulation of constructing kernel matrix shows that a simple cooling schedule of hybridization guarantees the positive definiteness of the resulting matrix. Given the promising nature of these results, we hope that experimental groups will consider additional experiments building upon our proposed molecular algorithm and analysis. Future work needs to consider how to optimize yield in such biomolecular processes. We hope that with such work, learning and classification will be implemented seamlessly in a living cell.

Acknowledgements

This work was partly supported by AFOSR (FA2386-12-1-4087, FA9550-15-1-0002), NRF (NRF-2010-0017734-Videome), SRFC of Samsung Electronics (SRFC-IT1401-12), DAPA, ADD, and BK21Plus.

References

Benenson, Y., 2012. Biomolecular computing systems: principles, progress and potential. Nat. Rev. Genet. 13, 455–468.
Bennett, C.-H., 1982. The thermodynamics of computation – a review. Int. J. Theor. Phys. 21, 905–940.
Britten, R.J., Kohne, D.E., 1968. Repeated sequences in DNA. Science 161, 529–540.
Cheok, M.-H., Yang, W., Pui, C.-H., Downing, J.-R., Cheng, C., Naeve, C.-W., Relling, M.-V., Evans, W.-E., 2003. Treatment-specific changes in gene expression discriminate in vivo drug response in human leukemia cells. Nat. Genet. 34, 85–90.
Gillespie, D., 1977. Exact stochastic simulation of coupled chemical reactions. J. Phys. Chem. 81, 2340–2361.
Gillespie, D., 2007. Stochastic simulation of chemical kinetics. Ann. Rev. Phys. Chem. 58, 35–55.
Haasdonk, B., 2005. Feature space interpretation of SVMs with indefinite kernels. IEEE Trans. Pattern Anal. Mach. Intell. 27, 482–492.
Kahan, M., Gil, B., Adar, R., Shapiro, E., 2008. Towards molecular computers that operate in a biological environment. Phys. D: Nonlin. Phenom. 9, 1165–1172.
Kim, J., Hopfield, J., Winfree, E., 2005. Neural network computation by *in vitro* transcriptional circuits. Adv. Neural Inf. Process. Syst. 17.
Kim, J., White, K.S., Winfree, E., 2006. Construction of an *in vitro* bistable circuit from synthetic transcriptional switches. Mol. Syst. Biol. 2, 1–12.
Kim, J.-S., Lee, J.-W., Noh, Y.-K., Park, J.-Y., Lee, D.-Y., Yang, K.-A., Chai, Y.-G., Kim, J.-C., Zhang, B.-T., 2008. An evolutionary Monte Carlo algorithm for predicting DNA hybridization. BioSystems 91, 69–75.
Kowalczyk, A., 2000. Maximal margin perceptron. In: Advances in Large Margin Classifiers. MIT Press, Cambridge, MA, USA, pp. 75–113.
Laplante, J.-P., Pemberton, M., Hjelmfelt, A., Ross, J., 1995. Experiments on pattern recognition by chemical kinetics. J. Phys. Chem. 99, 10063–10065.
Lim, H.-W., Lee, S.-H., Yang, K.-A., Lee, J.-Y., Yoo, S.-I., Park, T.-H., Zhang, B.-T., 2010. *In vitro* molecular pattern classification via DNA-based weighted-sum operation. BioSystems 100, 1–7.
Mills Jr., A.P., Turberfield, M., Turberfield, A.J., Yurke, B., Platzman, P.M., 2001. Experimental aspects of DNA neural network computing. Soft Comput. 5, 10–18.
Ong, C., Mary, X., Canu, S., Smola, A., 2004. Learning with non-positive kernels. Int. Conf. Mach. Learn., 639–646.
Qian, L., Winfree, E., Bruk, J., 2011. Neural network computation with DNA strand displacement cascades. Nature 475, 368–372.
Rätsch, G., Sonnenburg, S., Schäfer, C., 2006. Learning interpretable SVMs for biological sequence classification. BMC Bioinf. 7 (Suppl. 1), S9.
Sahu, S., Wang, B., Reif, J.-H., 2006. A framework for modeling DNA based molecular systems. In: Preliminary Proceedings of the Twelfth International Meeting on DNA Computing (DNA 12), pp. 250–265.
Santalucia, J., 1998. A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics. Proc. Natl. Acad. Sci. U. S. A. 95, 1460–1465.
SantaLucia Jr., J., Hicks, D., 2004. The thermodynamics of DNA structural motifs. Ann. Rev. Biophys. Biomol. Struct. 33, 415–440.
Schölkopf, B., Smola, A.J., 2001. Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge, MA, USA.
Zadeh, J.N., Steenberg, C.D., Bois, J.S., Wolfe, B.R., Pierce, M.B., Khan, A.R., Dirks, R.M., Pierce, N.A., 2011. NUPACK: analysis and design of nucleic acid systems. J. Comput. Chem. 32, 170–173.
Zhang, D., Turberfield, A., Yurke, B., Winfree, E., 2007. Engineering entropy-driven reactions and networks catalyzed by DNA. Science 318, 1121–1125.
Zwillinger, D., 1996. CRC Standard Mathematical Tables and Formulae. CRC Press.