19

NANOPORE CHEMINFORMATICS-BASED STUDIES OF INDIVIDUAL MOLECULAR INTERACTIONS

Stephen Winters-Hilt

19.1 INTRODUCTION

Channel current-based nanopore cheminformatics provides an incredibly versatile method for transducing single-molecule events into discernable channel current blockade levels. Single biomolecules, and the ends of biopolymers such as DNA, have been examined in solution with nanometer-scale precision [1–6]. In early studies [1], it was found that complete base-pair dissociations of dsDNA to ssDNA, "melting," could be observed for sufficiently short DNA hairpins. In later works [3, 5], the nanopore detector attained angstrom resolution and was used to "read" the ends of dsDNA molecules and was operated as a chemical biosensor. In Refs. [1, 2, 4] the nanopore detector was used to observe the conformational kinetics at the termini of single DNA molecules. In Refs. [7, 8], preliminary evidence of single-molecule binding and conformational kinetics was obtained by observation of single-molecule channel blockade currents. The DNA–DNA, DNA–protein, and protein–protein binding experiments that were described were novel in that they made critical use of indirect sensing (described below), where one of the molecules in the binding

Machine Learning in Bioinformatics, Edited by Yan-Qing Zhang and Jagath C. Rajapakse

experiment is either a natural channel blockade modulator or is attached to a blockade modulator.

19.2 NANOPORE DETECTOR BACKGROUND AND METHODS

19.2.1 The Highly Stable, Nanometer-Scale, α-Hemolysin Protein Channel

The nanopore detector is based on the α-hemolysin transmembrane channel, formed by seven identical 33 kDa protein molecules secreted by *Staphylococcus aureus*. The total channel length is 10 nm and is comprised of a 5-nm transmembrane domain and a 5-nm vestibule that protrudes into the aqueous *cis* compartment [9]. The narrowest segment of the pore is a 1.5-nm-diameter aperture [9]. By comparison, a single strand of DNA is about 1.3 nm in diameter and able to translocate. Although dsDNA is too large to translocate, about 10 base pairs at one end can still be drawn into the large *cis*-side vestibule (see Fig. 19.1a). This actually permits the most sensitive experiments to date, as the ends of "captured" dsDNA molecules can be observed for extensive periods of time to resolve features [1–5].

19.2.2 The Coulter Counter

The notion of using channels as detection devices dates back to the Coulter counter [10], where pulses in channel flow were measured to count bacterial cells. Cell transport through the Coulter counter is driven by hydrostatic pressure, and interactions between the cells and the walls of the channel are ignored. Since its original formulation, channel sizes have reduced from millimeter scale to nanometer scale, and the detection mechanism has shifted from measurements of hydrostatically driven fluid flow to measurements of electrophoretically driven ion flow. Analytes observed via channel measurements are likewise reduced in scale and are now at the scale of single biomolecules such as DNA and polypeptides [1–6, 11–16]. In certain situations, intramolecular, angstrom-level features are beginning to be resolved as well [1–5].

For nanoscopic channels, interactions between channel wall and translocating biomolecules cannot, usually, be ignored. On the one hand, this complicates analysis of channel blockade signals, and on the other hand, tell-tale on-off kinetics are revealed for binding between analyte and channel, and this is what has allowed the probing of intramolecular structure on single DNA molecules [1–5].

19.2.3 Coulter Data—Blockades Typically Static

Biophysicists and medical researchers have performed measurements of ion flow through single nanopores since the 1970s [17, 18]. The use of very large (biological) pores as polymer sensors is a relatively new possibility that dates from the pioneering experiments of Bezrukov et al. [16]. Their work proved that resistive

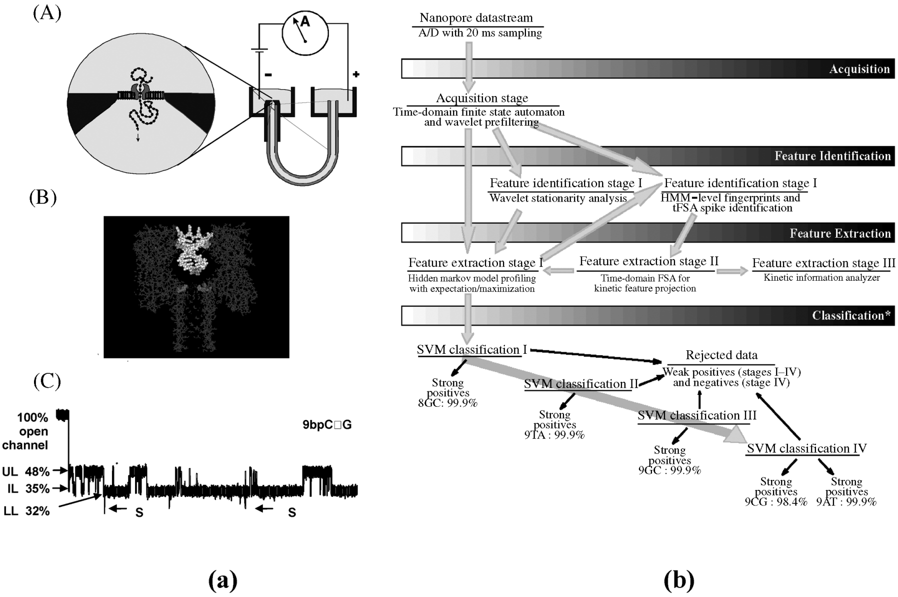
(A)
A
-
+
(B)
(C)
100% open channel
9bpC□G
UL 48%
IL 35%
LL 32%
S
S
(a)
Nanopore datastream
A/D with 20 ms sampling
Acquisition
Acquisition stage
Time-domain finite state automaton and wavelet prefiltering
Feature Identification
Feature identification stage I
Wavelet stationarity analysis
Feature identification stage I
HMM-level fingerprints and tFSA spike identification
Feature Extraction
Feature extraction stage I
Hidden markov model profiling with expectation/maximization
Feature extraction stage II
Time-domain FSA for kinetic feature projection
Feature extraction stage III
Kinetic information analyzer
Classification*
SVM classification I
Strong positives 8GC: 99.9%
SVM classification II
Strong positives 9TA : 99.9%
Rejected data
Weak positives (stages I–IV) and negatives (stage IV)
SVM classification III
Strong positives 9GC : 99.9%
SVM classification IV
Strong positives 9CG : 98.4%
Strong positives 9AT : 99.9%
(b)

Figure 19.1 The α-hemolysin nanopore detector and cheminformatics architecture. (a) (A) shows a nanopore device based on the α-hemolysin channel (from Ref. 3). It has been used for analysis of single DNA molecules, such as ssDNA, shown, and dsDNA, a nine-base pair DNA hairpin is shown in (B) superimposed on the channel geometry. The channel current blockade trace for the nine-base pair DNA hairpin blockade from (B) is shown in (C). (b) The signal processing architecture that was used to classify DNA hairpins with this approach: Signal acquisition was performed using a time-domain, thresholding, finite state automaton, followed by adaptive prefiltering using a wavelet-domain finite state automaton. Hidden Markov model (HMM) processing with expectation–maximization (EM) was used for feature extraction on acquired channel blockades. Classification was then done by support vector machine (SVM) on five DNA molecules: four DNA hairpin molecules with nine base-pair stem lengths that only differed in their blunt-ended DNA termini and an eight base-pair DNA hairpin. The accuracy shown is obtained upon completing the 15th single-molecule sampling/classification (in $\sim$6 s), where SVM-based rejection on noisy signals was employed.

pulse measurements, familiar from cell counting with the Coulter counter [10], could be reduced to the molecular scale and applied to polymers in solution. A seminal study, by Kasianowicz et al. [11], then showed that individual DNA and RNA polymers could be detected via their translocation blockade of a nanoscale pore formed by α-hemolysin toxin. In such prior nanopore detection work, the data analysis problems were also of a familiar "Coulter event" form, where the event was associated with a current blockade at a certain fixed level. A more informative setting is possible with nanometer-scale channels, however, due to nonnegligible interaction between analyte and channel. In this situation, the blockading molecule will not provide a single, fixed current reduction in the channel, but will modulate the ion flow through the channel by imprinting its binding interactions and conformational kinetics on the confined channel flow environment.

19.2.4 Nanopore Detector Augmentation Using Bifunctional Molecules

The improved detector sensitivity with toggling-type auxiliary molecules opens the door to a new, highly precise means for examining the binding affinities between any two molecules (bifunctional or not), all while still in solution. The bifunctional molecules that have been studied on the nanopore detector include antibodies and aptamers that were chosen to demonstrate the specific utility of this device in drug candidate screening (see Ref. [7]). In brief, an auxiliary molecule can be rigidly/covalently bound to the molecule of interest, and then exposed to a solution containing the other molecule of interest. The transitions between different stationary phases of blockade can then be related to the bound/unbound configuration between the two molecules of interest to reveal their binding kinetics (and binding strength).

19.2.5 Detection of Short-Term Binding and Stationary Phase

There are important distinctions in how a nanopore detector can function: direct versus indirect measurement of static, stationary, dynamic (possibly modulated), or nonstationary channel blockades.

A nanopore-based detector can directly measure molecular characteristics in terms of the blockade properties of individual molecules—this is possible due to the kinetic information that is embedded in the blockade measurements, where the adsorption–desorption history of the molecule to the surrounding channel, and the configurational changes in the molecule itself directly, imprint on the ionic flow through the channel [1–6]. (*Note*: The hypothesis that the current blockade patterns are caused by adsorption–desorption, and conformational flexing, is not conclusively proven, although preliminary work on mechanism [5] and the success of the experimental approaches [1–6] add growing credence to this hypothesis.) This approach offers prospects for DNA sequencing and single nucleotide polymorphism (SNP) analysis.

The nanopore-based detector works indirectly if it uses a reporter molecule that binds to certain molecules, with subsequent distinctive blockade by the bound-molecule complex. One example of this, with the established DNA experimental protocols, is exploration of transcription factor binding sites via the different dsDNA blockade signals that occur with and without DNA binding by a hypothesized transcription factor. Similarly, a channel-captured dsDNA "gauge" that is already bound to an antibody could provide a similar blockade shift upon antigen binding to its exposed antibody. The latter description provides the general mechanism for directly observing the single-molecule antigen-binding affinities of any antibody.

19.2.6 Nanopore Observation of Conformational Kinetics

Two conformational kinetic studies have been done, one on DNA hairpins with HIV-like termini [8], and the other on antibodies (preliminary results shown in Ref. [7]). The objective of the DNA HIV-hairpin conformational study was to systematically test how DNA dinucleotide flexibility (and reactivity) could be discerned using channel current blockade information (see Ref. [8]).

19.3 CHANNEL CURRENT CHEMINFORMATICS METHODS

19.3.1 The Channel Current Cheminformatics Architecture

The signal processing architecture (Fig. 19.1b) is designed to rapidly extract useful information from noisy blockade signals using feature extraction protocols, wavelet analysis, HMMs and SVMs. For blockade signal acquisition and simple, time-domain feature extraction, a finite state automaton (FSA) approach is used [19] that is based on tuning a variety of threshold parameters. A generic HMM can be used to characterize current blockades by identifying a sequence of subblockades as a sequence of state emissions [20–22]. The parameters of the generic-HMM can then be estimated using a method called EM [23] to effect denoising. The HMM method with EM, denoted HMM/EM, is used in what follows (further background on these methods can be found in Ref. [1–6]). Classification of feature vectors obtained by the HMM for each individual blockade event is then done using SVMs, an approach that automatically provides a confidence measure on each classification.

19.3.2 The Feature Vectors for SVM Analysis

The nanopore detector is operated such that a stream of 100-ms samplings are obtained (throughput was approximately one sampling per 300 ms in Ref. [3]). Each 100 ms signal acquired by the time-domain FSA consists of a sequence of 5000 subblockade levels (with the 20 μs analog-to-digital sampling). Signal preprocessing is then used for adaptive low-pass filtering. For the data sets examined, the preprocessing is expected to permit compression on the sample sequence from 5000 to 625 samples (later HMM processing then only required construction of a dynamic programming table with 625 columns). The signal preprocessing makes use of an off-line wavelet stationarity analysis (off-line wavelet stationarity analysis, Fig. 19.1b).

With completion of preprocessing, an HMM is used to remove noise from the acquired signals and to extract features from them (feature extraction stage, Fig. 19.1b). The HMM is, initially, implemented with 50 states, corresponding to current blockades in 1% increments ranging from 20% residual current to 69% residual current. The HMM states, numbered 0–49, corresponded to the 50 different current blockade levels in the sequences that are processed. The state emission parameters of the HMM are initially set so that the state j, $0 \leq j \leq 49$ corresponding to level $L = j + 20$, can emit all possible levels, with the probability distribution over emitted levels set to a discretized Gaussian with mean L and unit variance. All transitions between states are possible and initially are equally likely. Each blockade signature is denoised by five rounds of EM training on the parameters of the HMM. After the EM iterations, 150 parameters are extracted from the HMM. The 150 feature vectors obtained from the 50-state HMM-EM/Viterbi implementation in Refs. [1–8] are the 50 dwell percentage in the different blockade levels (from the Viterbi trace-back states), the 50 variances of the emission probability distributions associated with the different states, and the 50 merged transition probabilities from the primary and secondary blockade occupation levels (fits to two-state dominant modulatory blockade signals).

19.3.3 τ-FSA Blockade Acquisition and Feature Extraction

A channel current spike detector algorithm was developed in Ref. [8] to characterize the brief, very strong blockade "spike" behavior observed for molecules that occasionally break in the region exposed to the limiting aperture's strong electrophoretic force region. (In Ref. [1–6], where nine base-pair hairpins were studied, the spike events were attributed to a fray/extension event on the terminal base pair.) Together, the formulation of HMM-EM, FSAs, and spike detector provides a robust method for analysis of channel current data. Application of these methods is described in Ref. [8] for radiation-damaged DNA signals. The spike detector software is designed to count "anomalous" spikes, that is, spike noise not attributable to the gaussian fluctuations about the mean of the dominant blockade level. Spike count plots are generated to show increasing counts as cutoff thresholds are relaxed (to where eventually any downward deflection will be counted as a spike). The plots are automatically generated and automatically fit with extrapolations of their linear phases (exponential phases occur when cutoffs begin to probe the noise band of a blockade state—typically Gaussian noise "tails"). The extrapolations provide an estimate of "true" anomalous spike counts.

19.3.4 Markov Chains

Key "short-term memory" property of a Markov chain $P(x_i|x_{i-1,} \ldots, x_1) = P(x_i|x_{i-1}) = a_{x_{i-1}x_i}$, where $a_{x_{i-1}x_i}$ are sometimes referred to as "transition probabilities," and we have $P(x) = P(x_{L,} x_{L-1}, \ldots, x_1) = P(x_1) \prod_{i=2\ldots L} a_{x_{i-1}x_i}$. If we denote C_y for the count of events y, C_{xy} for the count of simultaneous events x and y, T_y for the count of strings of length 1, and T_{xy} for the count of strings of length 2, $a_{x_{i-1}x_i} = P(x \mid y) = P(x,y)/P(y) = [C_{xy}/T_{xy}]/[C_y/T_y]$. Note that since $T_{xy} + 1 = T_y \rightarrow T_{xy} \cong T_y$ (sequential data sample property if one long training block), $a_{x_{i-1}x_i} \cong C_{xy}/C_y = C_{xy}/\sum_x C_{xy}$; so C_{xy} is complete information for determining transition probabilities.

19.3.5 Viterbi Path

The recursive algorithm for the most likely state path given an observed sequence (the Viterbi algorithm) is expressed in terms of v_{ki}, the probability of the most probable path that ends with observation $Z_i = z_i$ and state $S_i = k$. The recursive relation is $v_{ki} = \max_n\{e_{ki}a_{nk}v_{n(i-1)}\}$, where the $\max_n\{\ldots\}$ operation returns the maximum value of the argument over different values of index n, and the boundary condition on the recursion is $v_{k0} = e_{k0}p_k$. The a_{kl} are the transition probabilities $P(S_i = l \mid S_{i\text{-}1} = k)$ to go from state k to state l. The e_{kb} are the emission probabilities $P(Z_i = b \mid S_i = k)$ while in state k. The emission probabilities are the main place where the data are brought into the HMM–EM algorithm (An inversion on the emission probability is possible when the states and emissions share the same alphabet of states/quantized emissions, and it is described in the results). The Viterbi path labelings are then recursively defined by $p(S_i|S_{(i+1)} = n) = \text{argma}x_k\{v_{ki}a_{kn}\}$, where the $\text{argma}x_n\{\ldots\}$ operation returns the index n with maximum value of the argument. The evaluation of sequence probability (and its Viterbi labeling) takes the emission and transition probabilities as *a* given. Estimates on these emission and transition probabilities are usually obtained by the EM algorithm.

19.3.6 Forward and Backward Probabilities

The forward/backward probabilities are used in the HMM–EM algorithm. The probabilities occur when evaluating $p(Z_{0\ldots L-1})$ by breaking the sequence probability $p(Z_{0\ldots L-1})$ into two pieces via use of a single hidden variable treated as a Bayesian parameter: $p(Z_{0\ldots L-1}) = \Sigma_k p(Z_{0\ldots i}, s_i = k) p(Z_{i+1\ldots L-1}, s_i = k) = \Sigma_k f_{ki} b_{ki}$, where $f_{ki} = p(Z_{0\ldots i}, s_i = k)$ and $b_{ki} = p(Z_{i+1\ldots L-1}, s_i = k)$. Given stationarity, the state transition probabilities and the state probabilities at the *i*th observation satisfy the trivial relation $p_{qi} = \Sigma_k a_{kq} p_{k(i-1)}$, where $p_{qi} = p(S_i = q)$ and $p_{q0} = p(S = q)$, and the latter probabilities are the state priors. The trivial recursion relation that is implied can be thought of as an operator equation, with operation the product by a_{kq} followed by summation (contraction) on the *k* index. The operator equation can be rewritten using an implied summation convention on repeated Greek-font indices (Einstein summation convention): $p_q = a_{\beta q} p_\beta$. Transition probabilities in a similar operator role, but now taking into consideration local sequence information via the emission probabilities, are found in recursively defined expressions for the forward variables, $f_{ki} = e_{ki}(a_{\beta k} f_{\beta(i-1)})$, and backward variables, $b_{ki} = a_{k\beta} e_{\beta(i+1)} b_{\beta(i+1)}$. The recursive definitions on forward and backward variables permit efficient computation of observed sequence probabilities using dynamic programming tables. It is at this critical juncture that side information must mesh well with the states (column components in the table), that is, in a manner like the emission or transition probabilities. Length information, for example, can be incorporated via length-distribution-biased transition probabilities (as described in a new method in Ref. [24]).

19.3.7 HMM-with-Duration Channel Current Signal Analysis

The HMM-with-Duration implementation, described in Ref. [24], has been tested in terms of its performance at parsing synthetic blockade signals. The synthetic data range over an exhaustive set of possibilities for thorough testing of the HMM-with-duration. The synthetic data used in Ref. [24] were designed to have two levels, with lifetime in each level determined by a governing distribution (Poisson and Gaussian distributions with a range of mean values were considered). The results clearly demonstrate the superior performance of the HMM-with-duration over its simpler, HMM-without-duration, formulation. With the use of the EVA-projection method, this affords a robust means to obtain kinetic feature extraction. The HMM with duration is critical for accurate kinetic feature extraction, and the results in Ref. [24] suggest that this problem can be elegantly solved with a pairing of the HMM-with-duration stabilization with EVA projection.

19.3.8 HMM-with-Duration via Cumulant Transition Probabilities

The transition probabilities for state "*s*" to remain in state "*s*," a "*ss*" transition can be computed as $\text{Prob}(ss \mid s_{\text{length}} = L) = \text{Prob}(s_{\text{length}} \geq L + 1)/\text{Prob}(s_{\text{length}} \geq L)$. The transition probabilities out of state "*s*" can have some subtleties, as shown in the following, where the states are exon (*e*), intron (*i*), and junk (*j*). In this case, the transition probabilities governing the following transitions, $(jj) \rightarrow (je)$, $(ee) \rightarrow (ej)$, $(ee) \rightarrow (ei)$, $(ii) \rightarrow (ie)$, are computed as $\text{Prob}(ei \mid e_{\text{length}} = L) = \text{Prob}(e_{\text{length}} = L)/\text{Prob}(e_{\text{length}} \geq L) \times 40/(40 + 60)$ and $\text{Prob}(ej \mid e_{\text{length}} = L) = \text{Prob}(e_{\text{length}} = L)/\text{Prob}(e_{\text{length}} \geq L) \times 60/(40 + 60)$, where the total number of (*ej*) transitions is 60 and the total number of (*ei*) transitions is 40. The pseudocode to track the critical length information, on a cellular basis in the dynamic programming table, goes as follows:

(1) Maintain separate counters for the junk, exon, and intron regions.
(2) The counters are updated as
    (a) The exon counter is set to 2 for a $(je) \rightarrow (ee)$ transition
    (b) The exon counter gets incremented by 1 for every $(ee) \rightarrow (ee)$ transition.
(3) $\text{Prob}(e_{\text{length}} \geq L + 1)$ is computed as $\text{Prob}(e_{\text{length}} \geq L + 1) = 1 - \sum_{i=1 \ldots L} \text{Prob}(e_{\text{length}} = i)$. Hence, we generate a list such that for each index "$k > 0$," the value $1 - \sum_{i=1,\ldots,k} \text{Prob}(e_{\text{length}} = i)$ is stored.

19.3.9 EVA Projection

The HMM method is based on a stationary set of emission and transition probabilities. Emission broadening, via amplification of the emission state variances, is a filtering heuristic that leads to level projection that strongly preserves transition times between major levels (see Ref. [24] for further details). This approach does not require the user to define the number of levels (classes). This is a major advantage compared to existing tools that require the user to determine the levels (classes) and perform a state projection. This allows kinetic features to be extracted with a "simple" FSA that requires minimal tuning. One important application of the HMM-with-duration method used in Ref. [24] includes kinetic feature extraction from EVA-projected channel current data (the HMM-with-duration is shown to offer a critical stabilizing capability in an example in Ref. [24]). The EVA-projected/HMMwDur processing offers a handsoff (minimal tuning) method for extracting the mean dwell times for various blockade states (the core kinetic information).

19.3.10 Support Vector Machines

SVMs are fast, easily trained discriminators [25, 26], for which strong discrimination is possible without the overfitting complications common to neural net discriminators [16]. The SVM approach also encapsulates a significant amount of model fitting and discriminatory information in the choice of kernel in the SVM, and a number of novel kernels have been developed. In application to channel current signal analysis, there is generally an abundance of experimental data available, and if not, the experimenter can usually just take more samples and make it so. In this situation, it is appropriate to seek a method good at both classifying data and evaluating a confidence in the classifications given. In this way, data that are low confidence can simply be dropped. The structural risk minimization at the heart of the SVM method's robustness also provides a strong confidence measure. For this reason, SVMs are the classification method of choice for channel current analysis as they have excellent performance at 0% data drop and as weak data are allowed to be dropped, the SVM-based approaches far exceed the performance of most other methods known.

In Ref. [3], novel information-theoretic kernels were introduced for notably better performance over standard kernels, with discrete probability distributions as part of feature vector data. The use of probability vectors, and L_1-norm feature vectors in general, turns out to be a very general formulation, wherein feature extraction makes use of signal decomposition into a complete set of separable states that can be interpreted or represented as a probability vector (or normalized collection of such, etc.). A probability vector formulation also provides a straightforward hand-off to the SVM classifiers since all feature vectors have the same length with such an approach. What this means for the SVM, however, is that geometric notions of distance are no longer the best measure for comparing feature vectors. For probability vectors (i.e., discrete distributions), the best measures of similarity are the various information-theoretic divergences: Kullback–Leibler, Renyi, and so on. By symmetrizing over the arguments of those divergences, a rich source of kernels is obtained that works well with the types of probabilistic data obtained, as shown in Refs. [3, 7, 27].

The SVM discriminators are trained by solving their Karush–Kuhn–Tucker (KKT) relations using the sequential minimal optimization (SMO) procedure [28]. A chunking [29, 30] variant of SMO is also employed to manage the large training task at each SVM node. The multiclass SVM training generally involves thousands of blockade signatures for each signal class.

19.3.11 Binary Support Vector Machines

Binary SVMs are based on a decision-hyperplane heuristic that incorporates structural risk management by attempting to impose a training-instance void, or "margin," around the decision hyperplane [25].

Feature vectors are denoted by x_{ik}, where index i labels the M feature vectors ($1 \leq i \leq M$) and index k labels the N feature vector components ($1 \leq i \leq N$). For the binary SVM, labeling of training data is done using label variable $y_i = \pm 1$ (with sign according to whether the training instance was from the positive or negative class). For hyperplane separability, elements of the training set must satisfy the following conditions: $w_\beta x_{i\beta} - b \geq +1$ for i such that $y_i = +1$, and $w_\beta x_{i\beta} - b \leq -1$ for $y_i = -1$, for some values of the coefficients $w_1, \ldots, w_N$ and b (using the convention of implied sum on repeated Greek indices). This can be written more concisely as $y_i(w_\beta x_{i\beta} - b) - 1 \geq 0$. Data points that satisfy the equality in the above are known as "support vectors" (or "active constraints").

Once training is complete, discrimination is based solely on position relative to the discriminating hyperplane: $w_\beta x_{i\beta} - b = 0$. The boundary hyperplanes on the two classes of data are separated by a distance $2/w$, known as the "margin," where

$w^2 = w_\beta w_\beta$. By increasing the margin between the separated data as much as possible, the optimal separating hyperplane is obtained. In the usual SVM formulation, the goal to maximize w^{-1} is restated as the goal to minimize w^2. The Lagrangian variational formulation then selects an optimum defined at a saddle point of $L(w,\mathrm{b};\alpha) = (w_\beta w_\beta)/2 - \alpha_\gamma y_\gamma(w_\beta x_{\gamma\beta} - b) - \alpha_0$, where $\alpha_0 = \Sigma_\gamma \alpha_\gamma$, $\alpha_\gamma \geq 0$ $(1 \leq \gamma \leq M)$. The saddle point is obtained by minimizing with respect to $\{w_1, \ldots, w_N, b\}$ and maximizing with respect to $\{\alpha_1, \ldots, \alpha_M\}$. If $y_i(w_\beta x_{i\beta}\text{-}b) - 1 \geq 0$, then maximization on α_i is achieved for $\alpha_i = 0$. If $y_i(w_\beta x_{i\beta} - b) - 1 = 0$, then there is no constraint on α_i. If $y_i(w_\beta x_{i\beta} - b) - 1 < 0$, there is a constraint violation, and $\alpha_i \rightarrow \infty$. If absolute separability is possible, the last case will eventually be eliminated for all α_i, otherwise, it is natural to limit the size of α_i by some constant upper bound, that is, $\max(\alpha_i) = C$, for all i. This is equivalent to another set of inequality constraints with $\alpha_i \leq C$. Introducing sets of Lagrange multipliers, ξ_γ and μ_γ $(1 \leq \gamma \leq M)$, to achieve this, the Lagrangian becomes

$$L(w, \mathrm{b}; \alpha, \xi, \mu) = (w_\beta w_\beta)/2 - \alpha_\gamma[y_\gamma(w_\beta x_{\gamma\beta} - b) + \xi_\gamma] + \alpha_0 + \xi_0 C - \mu_\gamma \xi_\gamma, \quad \text{where } \xi_0 = \Sigma_\gamma \xi_\gamma, \alpha_0 = \Sigma_\gamma \alpha_\gamma \quad \text{and} \quad \alpha_\gamma \geq 0 \quad \text{and} \quad \xi_\gamma \geq 0\,(1 \leq \gamma \leq M).$$

At the variational minimum on the $\{w_1, \ldots, w_N, b\}$ variables, $w_\beta = (\alpha_\gamma y_\gamma x_{\gamma\beta}$, and the Lagrangian simplifies to $L(\alpha) = \alpha_0 - (\alpha_\delta y_\delta x_{\delta\beta}(_\gamma y_\gamma x_{\gamma\beta})/2$, with $0 \leq \alpha_\gamma \leq C$ $(1 \leq \gamma \leq M)$ and $\alpha_\gamma y_\gamma = 0$, where only the variations that maximize in terms of the α_γ remain (known as the Wolfe Transformation). In this form, the computational task can be greatly simplified. By introducing an expression for the discriminating hyperplane $f_i = w_\beta x_{i\beta}$ $b = \alpha_\gamma y_\gamma x_{\gamma\beta} x_{i\beta} - b$, the variational solution for $L(\alpha)$ reduces to the following set of relations (known as the Karush–Kuhn–Tucker, or KKT, relations): (1) $\alpha_i = 0 \Leftrightarrow y_i f_i \geq 1$, (2) $0 < \alpha_i < C \Leftrightarrow y_i f_i = 1$, and (3) $\alpha_i = C \Leftrightarrow y_i\, f_i \leq 1$. When the KKT relations are satisfied for all of the α_γ (with $\alpha_\gamma y_\gamma = 0$ maintained), the solution is achieved. (The constraint $\alpha_\gamma y_\gamma = 0$ is satisfied for the initial choice of multipliers by setting the α's associated with the positive training instances to $1/N^{(+)}$ and the α's associated with the negatives to $1/N^{(-)}$, where $N^{(+)}$ is the number of positives and $N^{(-)}$ is the number of negatives.) Once the Wolfe transformation is performed, it is apparent that the training data (support vectors, in particular, KKT class (2) above) enter into the Lagrangian solely via the inner product $x_{i\beta} x_{j\beta}$. Likewise, the discriminator f_i, and KKT relations are also dependent on the data solely via the $x_{i\beta} x_{j\beta}$ inner product.

Generalizations of the SVM formulation to data-dependent inner products other than $x_{i\beta} x_{j\beta}$ are possible and are usually formulated in terms of the family of symmetric positive definite functions (reproducing kernels) satisfying Mercer's conditions [25].

19.3.12 Binary SVM Discriminator Implementation

The SVM discriminators are trained by solving their KKT relations using the SMO procedure [28]. The method described here follows the description of Ref. [28] and begins by selecting a pair of Lagrange multipliers, $\{\alpha_1, \alpha_2\}$, where at least one of the multipliers has a violation of its associated KKT relations (for simplicity, it is assumed in what follows that the multipliers selected are those associated with the first and second feature vectors: $\{x_1, x_2\}$). The SMO procedure then "freezes" variations in all but the two selected Lagrange multipliers, permitting much of the computation to be circumvented by use of analytical reductions:

$$L(\alpha_1, \alpha_2; \alpha_{\beta' \geq 3}) = \alpha_1 + \alpha_2 - (\alpha_1^2 K_{11} + \alpha_2^2 K_{22} + 2\alpha_1 \alpha_2 y_1 y_2 K_{12})/2$$
$$- \alpha_1 y_1 v_1 - \alpha_2 y_2 v_2 + \alpha_{\beta'} U_{\beta'} - (\alpha_{\beta'} \alpha_{\gamma'} y_{\beta'} y_{\gamma'} K_{\beta'_{\gamma}})/2,$$

with $\beta', \gamma' \geq 3$, and where $K_{ij} \equiv K(x_i,\ x_j)$ and $v_i \equiv \alpha_{\beta'} y_{\beta'} K_{i\beta'}$ with $\beta' \geq 3$. Due to the constraint $\alpha_\beta y_\beta = 0$, we have the relation $\alpha_1 + s\alpha_2 = -\gamma$, where $\gamma \equiv y_1 \alpha_{\beta'} y_{\beta'}$ with $\beta' \geq 3$ and $s \equiv y_1 y_2$. Substituting the constraint to eliminate references to α_1, and performing the variation on α_2: $\partial L(\alpha_2;\alpha_{\beta' \geq 3})/\partial \alpha_2 = (1 - s) + \eta \alpha_2 + s\gamma(K_{11} - K_{22}) + s y_1 v_1 - y_2 v_2$, where $\eta \equiv (2K_{12} - K_{11} + K_{22})$. Since v_i can be rewritten as $v_i = w_\beta x_{i\beta} - \alpha_1 y_1 K_{i1} - \alpha_2 y_2 K_{\mathrm{i}2}$, the variational maximum $\partial L(\alpha_2;\alpha_{\beta' \geq 3})/\partial \alpha_2 = 0$ leads to the following update rule:

$$\alpha_2^{\mathrm{new}} = \alpha_2^{\mathrm{old}} - y_2((w_\beta x_{1\beta} - y_1) - (w_\beta x_{2\beta} - y_2))/\eta.$$

Once α_2^{new} is obtained, the constraint $\alpha_2^{\mathrm{new}} \leq C$ must be reverified in conjunction with the $\alpha_\beta y_\beta = 0$ constraint. If the $L(\alpha_2;\alpha_{\beta' \geq 3})$ maximization leads to a α_2^{new} that grows too large, the new α_2 must be "clipped" to the maximum value satisfying the constraints. For example, if $y_1 \neq y_2$, then increases in α_2 are matched by increases in α_1. So, depending on whether α_2 or α_1 is nearer its maximum of C, we have $\max(\alpha_2) = \mathrm{argmin}\ \{\alpha_2 + (C - \alpha_2);\ \alpha_2 + (C - \alpha_1)\}$. Similar arguments provide the following boundary conditions: (1) if $s = -1$, $\max(\alpha_2) = \mathrm{argmin}\{\alpha_2;\ C + \alpha_2 - \alpha_1\}$ and $\min(\alpha_2) = \mathrm{argmax}\ \{0; \alpha_2 - \alpha_1\}$, and (2) if $s = +1$, $\max(\alpha_2) = \mathrm{argmin}\{C; \alpha_2 + \alpha_1\}$ and $\min(\alpha_2) = \mathrm{argmax}\ \{0;\ \alpha_2 + \alpha_1 - C\}$. In terms of the new $\alpha_2^{\mathrm{new,\ clipped}}$, clipped as indicated above if necessary, the new α_1 becomes

$$\alpha_1^{\mathrm{new}} = \alpha_1^{\mathrm{old}} + s(\alpha_2^{\mathrm{old}} - a_2^{\mathrm{new,clipped}}),$$

where $s \equiv y_1 y_2$ as before. After the new α_1 and α_2 values are obtained, there still remains the task of obtaining the new b value. If the new α_1 is not "clipped," then the update must satisfy the nonboundary KKT relation: $y_1 f(x_1) = 1$, that is, $f^{\mathrm{new}}(x_1) - y_1 = 0$. By relating f^{new} to f^{old}, the following update on b is obtained:

$$b^{\mathrm{new1}} = b - (f^{\mathrm{new}}(x_1) - y_1) - y_1(\alpha_1^{\mathrm{new}} - \alpha_1^{\mathrm{old}})K_{11} - y_2(a_2^{\mathrm{new,clipped}} - \alpha_2^{\mathrm{old}})K_{12}.$$

If α_1 is clipped, but α_2 is not, the above argument holds for the α_2 multiplier, and the new b is

$$b^{\mathrm{new2}} = b - (f^{\mathrm{new}}(x_2) - y_2) - y_2(\alpha_2^{\mathrm{new}} - \alpha_2^{\mathrm{old}})K_{22} - y_1(\alpha_1^{\mathrm{new,clipped}} - \alpha_1^{\mathrm{old}})K_{12}.$$

If both α_1 and α_2 values are clipped, then any of the b values between b^{new1} and b^{new2} is acceptable and following the SMO convention, the new b is chosen to be

$$b^{\text{new}} = (b^{\text{new1}} + b^{\text{new2}})/2.$$

19.3.13 SVM Kernel/Algorithm Variants

The SVM Kernels that are used are based on "regularized" distances or divergences as those used in Refs. [3, 7, 27], where regularization is achieved by exponentiating the negative of a distance-measure squared ($d^2(x,y)$) or a symmetrized divergence measure ($D(x,y)$), the former if using a geometric heuristic for comparison of feature vectors and the latter if using a distributional heuristic. For the Gaussian Kernel, $d^2(x,y) = \Sigma_k(x_k - y_k)^2$; for the Absdiff Kernel, $d^2(x,y) = (\Sigma_k|x_k - y_k|)^{1/2}$; and for the symmetrized relative entropy kernel, $D(x,y) = D(x \,||\, y) + D(y \,||\, x)$, where $D(x \,||\, y)$ is the standard relative entropy.

19.3.14 SVM-External Clustering

As with the multiclass SVM discriminator implementations, the strong performance of the binary SVM enables SVM-external as well as SVM-internal approaches to clustering. The external-SVM clustering algorithm introduced in Ref. [27] clusters data vectors with no *a priori* knowledge of each vector's class. The algorithm works by first running a binary SVM against a data set, with each vector in the set randomly labeled, until the SVM converges. To obtain convergence, an acceptable number of KKT violators must be found. This is done through running the SVM on the randomly labeled data with different numbers of allowed violators until the number of violators allowed is near the lower bound of violators needed for the SVM to converge on the particular data set. Choice of an appropriate kernel and an acceptable sigma value will also affect convergence. After the initial convergence is achieved, the sensitivity plus specificity will be low, likely near 1. The algorithm now improves this result by iteratively relabeling the worst misclassified vectors that have confidence factor values beyond some threshold, followed by rerunning the SVM on the newly relabeled data set. This continues until no more progress can be made. Progress is determined by an increasing value of sensitivity plus specificity, hopefully nearly reaching 2. This method provides a way to cluster data sets without prior knowledge of the data's clustering characteristics, or the number of clusters.

19.4 RECENT ARCHITECTURAL REFINEMENTS

19.4.1 Data Inversion

A new form of "inverted" data injection is possible when the states and quantized emission values share the same alphabet. This new form may not have any clear

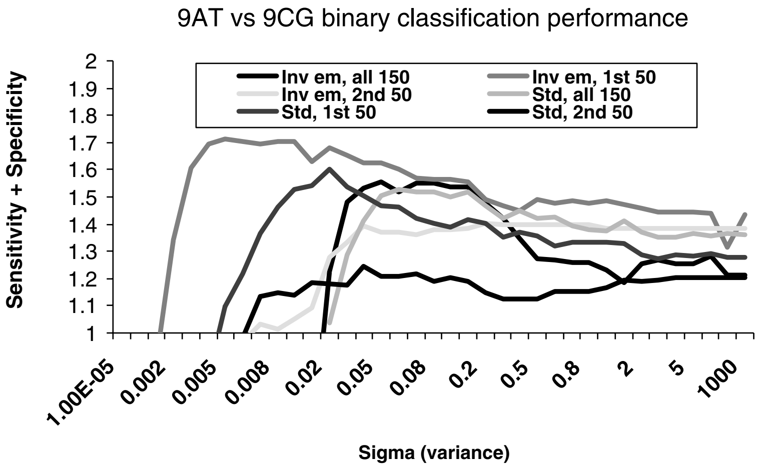

Figure 19.2 The binary classification performance using features extracted with HMM data inversion versus HMM standard. Blockade data were extracted from channel measurements of 9AT and 9CG hairpins (both hairpins with nine base-pair stems), and the data extraction involved either standard (std) emission data representations or inverted (inv) emission data, and was based on feature sets of the full 150 features, or the first 50, with the Viterbi-path level dwell time percentages, or the second 50, the emission variances (much weaker features as expected). The inverted data offer consistently better discriminatory performance by the SVM classifier.

probabilistic interpretation (use of “time-reversed” conditional probabilities or “absorption” instead of emission perhaps) but can be clearly defined in terms of the core data injection that occurs via the forward/backward variables, with emissions conditional probabilities taken with reversed conditional probabilities. Results shown in Fig. 19.2 are part of an extensive study that consistently shows approximately 5% improvement in accuracy (sensitivity + specificity) with the aforementioned data inversion (upon SVM classification), and this holds true over wide ranges of kernel parameters and collections of feature sets in all cases.

SVM performance on the same train/test data splits, but with 2600 uncompressed component feature vectors instead of 150 component feature vectors, offered similar performance after drop optimization. SVM performance with Adaboost on the 2600 components (taken as naive Bayes stubs), with selection for the top 150 “experts,” demonstrates a significant robustness to what the SVM can “learn” in the presence of noise (some of the 2600 components have richer information, but even more are noise contributors). This also validates the effectiveness with which the 150-parameter compression was able to describe the two-state dominant blockade data found for the nine-base-pair hairpin and other types of “toggler” blockades.

19.4.2 Automated Feature Selection Using AdaBoost

Two new methods are being pursued for automated feature selection/feature compression. This is particularly important for handling the transition probabilities

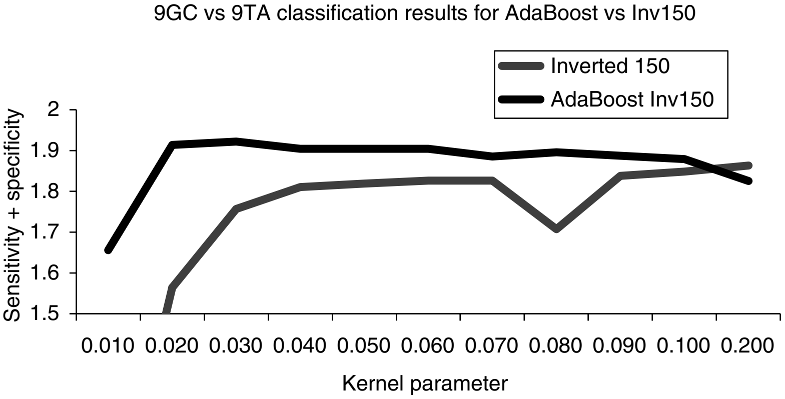

Figure 19.3 Adaboost feature selection strengthens the SVM performance of the Inverted HMM feature extraction set. Classification improvement with Adaboost taking the best 50 from the inverted emission 150-feature set. 95% accuracy is possible for discriminating 9GC from 9TA hairpins with no data dropped with use of Adaboost, and without Adaboosting, the accuracy is approximately 91%.

obtained by the HMM (if it has 50 states, it has 50^2 transition probabilities). The first builds on the transition probability compression in other ways optimized for the signals observed, and the second uses boosting (AdaBoost) over the individual emission and transition probabilities (which are used to provide a pool of weak, naïve Bayes, classifiers) to select the best features, and then use those features when passing feature vectors to the SVM classifiers, among other things. It is found, however, that boosting from the set of 150 features worked better than that from the 2600 naive Bayes, and boosting from the 50 features in the first group worked best (see Fig. 19.3). (This result is also consistent with the PCA filtering in Ref. [31], mostly reducing the 150-feature set to the first 50 features.)

19.4.3 The Machine Learning Software Interface Project

Web-accessible machine-learning tools have been developed for general pattern recognition tasks, with specific application to channel current analysis, kinetic analysis, and computational genomics. The core machine learning tools are primarily based on SVM algorithms, HMM algorithms, and FSAs. The group Web site at **http://logos.cs.uno.edu/~nano/** provides interfaces to (1) several binary SVM variants (with novel kernel selections and heuristics), (2) a multiclass (internal) SVM, (3) an SVM-based clustering tool, (4) an FSA-based nanopore spike detector, (5) an HMM channel current feature extraction tool, and (6) a kinetic feature extraction tool. The Web site is designed using HTML and CGI scripts that are executed to process the data sent when a form filled in by the user is received at the web server—results are then e-mailed to the address indicated by the user.

19.5 DISCUSSION

19.5.1 Individual Reaction Histories—Single Molecule Kinetics

Channel current-based kinetic feature extraction not only appears to be practical but are also the next key step in the study of individual reaction histories. In essence, binding strength (K_d) between molecules in solution, and conformational state transitions, can be determined via channel blockade observations corresponding to lifetimes on the different states. The ergodic hypothesis, that time averages can replace ensemble averages, can now be explored in this context as well. Nanopore detection promises to be a very precise method for evaluating binding strengths and observing single-molecule conformational changes. Adaptive software techniques to manage the complex data analysis are needed, and they are in growing demand. Recent advances have been made in channel current cheminformatics to address these issues, including new developments in distributed and unsupervised learning processes.

19.5.2 Deciphering the Transcriptome and Transcription Factor-Based Drug Discovery

The examination of transcription factor binding to target transcription factor binding site (TF/TFBS interactions) affords the possibility to understand, quantitatively, much of the transcriptome. This same information, coupled with new interaction information upon introduction of synthetic TFs (possible medicines), provides a very powerful, directed approach to drug discovery.

19.5.3 A New Window into Understanding Antibody Function

Upon binding to antigen, a series of events are initiated by the interaction of the antibody carboxy-terminal region with serum proteins and cellular receptors. Biological effects resulting from the carboxy-terminal interactions include activation of the complement cascade, binding of immune complexes by carboxy-terminal receptors on various cells, and the induction of inflammation. Nanopore detection provides a new way to study the binding/conformational histories of individual antibodies. Many critical questions regarding antibody function are still unresolved, questions that can be approached in a new way with the nanopore detector. The different antibody binding strengths to target antigen, for example, can be ranked according to the observed lifetimes of their bound states. Questions of great interest include, Are allosteric changes transmitted through the molecule upon antigen binding? Can effector function activation be observed and used to accelerate drug discovery efforts?

19.5.4 Hybrid Clustering and Scan Clustering for Indirect-Interaction Kinetic Information

An exciting area of machine learning research is being brought to bear on the kinetic signal decomposition of channel currents. The external-SVM approach described in the background and Ref. [27] offers to provide one of the most powerful, unsupervised methods for clustering. Part of the strength of the method is that it is nonparametric. Part of the weakness is in obtaining an initial clustering. To improve on this, efforts are underway to graft the external SVM onto an initial clustering using bisect-K-means (that is seeded by principle direction divisive partitioning [32–34], or principle component analysis [35], when random seeding does poorly. External-SVM clustering, along the lines of Ref. [27], may allow precise cluster regrowth by its ability to operate on a shifting support vector structure as direct label operations (binary “flipping”) are performed.

REFERENCES

1. Winters-Hilt, S. Single-molecule Biochemical Analysis Using Channel Current Cheminformatics. *UPoN 2005: Fourth International Conference on Unsolved Problems of Noise and Fluctuations in Physics, Biology, and High Technology, June 6–10, 2005. AIP Conference Proceeding*, 800: 337–342, 2005.
2. Winters-Hilt, S. and Akeson, M. Nanopore cheminformatics. *DNA and Cell Biology*, 23 (10): 675–683, 2004.
3. Winters-Hilt, S., Vercoutere, W., DeGuzman, V. S., Deamer, D. W., Akeson, M., and Haussler, D. Highly accurate classification of Watson–Crick basepairs on termini of single DNA molecules. *Biophysical Journal*, 84: 967–976, 2003.
4. Winters-Hilt, S. Highly accurate real-time classification of channel-captured DNA termini. *Third International Conference on Unsolved Problems of Noise and Fluctuations in Physics, Biology, and High Technology*, 2003, pp. 355–368.
5. Vercoutere, W., Winters-Hilt, S., DeGuzman, V. S., Deamer, D., Ridino, S., Rogers, J. T., Olsen, H. E., Marziali, A., and Akeson, M. Discrimination among individual Watson–Crick base-pairs at the termini of single DNA hairpin molecules. *Nucleic Acids Research*, 31: 1311–1318, 2003.
6. Vercoutere, W., Winters-Hilt, S., Olsen, H., Deamer, D. W., Haussler, D., and Akeson, M. Rapid discrimination among individual DNA hairpin molecules at single-nucleotide resolution using an ion channel. *Nature Biotechnology*, 19(3): 248–252, 2001.
7. Winters-Hilt, S. Nanopore detector based analysis of single-molecule conformational kinetics and binding interactions. *BMC Bioinformatics*, 7 Suppl 2: S21, 2006.
8. Winters-Hilt, S., Landry, M., Akeson, M., Tanase, M., Amin, I., Coombs, A., Morales, E., Millet, J., Baribault, C., and Sendamangalam, S. Cheminformatics methods for novel nanopore analysis of HIV DNA termini. *BMC Bioinformatics*, 7 Suppl 2: S22, 2006.
9. Song, L. Hobaugh, M. R. Shustak, C. Cheley, S. Bayley, H., and Gouaux, J. E. Structure of staphylococcal alpha-hemolysin, a heptameric transmembrane pore. *Science*, 274(5294): 1859–1866, 1996.
10. Coulter, W. H. High speed automatic blood cell counter and cell size analyzer. *Proceedings of the National Electronics Conference*, 12, 1034–1042, 1957.
11. Kasianowicz, J. J., Brandin, E., Branton, D., and Deamer, D. W. Characterization of individual polynucleotide molecules using a membrane channel. *Proceedings of the National Academy of Sciences of United States of America*, 93(24): 13770–13773, 1996.

12. Akeson, M., Branton, D., Kasianowicz, J. J., Brandin, E., and Deamer, D. W. Microsecond time-scale discrimination among polycytidylic acid, polyadenylic acid, and polyuridylic acid as homopolymers or as segments within single RNA molecules. *Biophysical Journal*, 77(6): 3227–3233, 1999.
13. Meller, A., Nivon, L., Brandin, E., Golovchenko, J., and Branton, D. Rapid nanopore discrimination between single polynucleotide molecules. *Proceedings of the National Academy of Sciences of United States of America* 97(3): 1079–1084, 2000.
14. Meller, A., Nivon, L., and Branton, D. Voltage-driven DNA translocations through a nanopore. *Physical Review Letters*, 86(15): 3435–3438, 2001.
15. Bezrukov, S. M. Ion channels as molecular coulter counters to probe metabolite transport. *The Journal of Membrane Biology*, 174: 1–13, 2000.
16. Bezrukov, S. M., Vodyanoy, I., and Parsegian, V. A. Counting polymers moving through a single ion channel. *Nature* 370(6457): 279–281, 1994.
17. Sakmann, B., Neher, E., *Single-Channel Recording*. Plenum Press, 1995.
18. Ashcroft, F. *Ion Channels and Disease*. Academic Press, 2000.
19. Cormen, T. H., Leiserson, C. E., and Rivest, R. L. *Introduction to Algorithms*. MIT-Press, Cambridge, 1989.
20. Chung, S.-H., Moore, J. B., Xia, L., Premkumar, L. S., and Gage, P. W. Characterization of single channel currents using digital signal processing techniques based on Hidden Markov Models. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 329: 265–285, 1990.
21. Chung, S.-H. and Gage, P. W. Signal processing techniques for channel current analysis based on hidden Markov models. In: P. M. Conn, ed., *Methods in Enzymology; Ion Channels, Part B*, Academic Press, Inc., San Diego, 1998, pp. 420–437.
22. Colquhoun, D. Sigworth, F. J. Fitting and statistical analysis of single-channel products. In: B., Sakmann and E., Neher, *Single-Channel Recording*, 2nd edition, Plenum Publishing Corp., New York, 1995, pp. 483–587.
23. Durbin, R. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge, UK, 1998.
24. Winters-Hilt, S. Hidden Markov Model variants and their application. *BMC Bioinformatics*, 7 Suppl 2: S14, 2006.
25. Vapnik, V. N. *The Nature of Statistical Learning Theory*, 2nd edition, Springer-Verlag, New York, 1998.
26. Burges, C. J. C. A tutorial on support vector machines for pattern recognition. *Data Mining Knowledge Discovery* 2: 121–167, 1998.
27. Winters-Hilt, S., Yelundur, A., McChesney, C., and Landry, M. Support vector machine implementations for classification & clustering. *BMC Bioinformatics*, 7 Suppl 2: S4, 2006.
28. Platt, J. C. Fast training of support vector machines using sequential minimal optimization. In: B. Scholkopf, C. J. C., Burges, and A. J. Smola, *Advances in Kernel Methods—Support Vector Learning*, MIT Press, Cambridge, 1998, Chapter 12.
29. Osuna, E., Freund, R., and Girosi, F. An improved training algorithm for support vector machines. In: J. Principe, L. Gile, N. Morgan, and E. Wilson, *Neural Networks for Signal Processing VII*, IEEE, New York, 1997, 276–285.

30. Joachims, T. Making large-scale SVM learning practical. In: B. Scholkopf, C. J. C. Burges, and A. J. Smola, Eds, *Advances in Kernel Methods—Support Vector Learning*, MIT Press, Cambridge, 1998, Chapter 11.
31. Iqbal, R., Landry, M., and Winters-Hilt, S. DNA molecule classification using feature primitives. *BMC Bioinformatics* 7 Suppl 2: S15, 2006.
32. Hand, D., Mannila, H., and Smyth, P. *Principles of Data Mining*. The MIT press Cambridge, MA, 2001.
33. O'Connel, M. J. Search program for significant variables. *Computer Physics Communications,* 8: 49–55, 1974.
34. Wall, M. E., Rechtsteiner, A., and Rocha, L. M. Singular value decomposition and principle component analysis. In: D. P., Berrar, W., Dubitsky, M., Granzow, Eds, *A Practical Approach to Microarray Data Analysis*, Kluwer, Norwell, MA, 2003, pp. 91–109.
35. Boley, D. L. Principle direction divisive partitioning. *Data Mining and Knowledge Discovery*, 2(4): 325–344, 1998.