BIOINFORMATICS ORIGINAL PAPER Vol. 27 no. 15 2011, pages 2054–2061
doi:10.1093/bioinformatics/btr353

Sequence analysis

Advance Access publication June 17, 2011

A tree-based approach for motif discovery and sequence classification

Rui Yan[1,2,*], Paul C. Boutros[3,*] and Igor Jurisica[1,2,4,*]

[1]Department of Computer Science, University of Toronto, Toronto, Canada M5S 3G4, [2]Ontario Cancer Institute and the Campbell Family Institute for Cancer Research, Princess Margaret Hospital/University Health Network, Toronto, Canada M5G 2L7, [3]Ontario Institute for Cancer Research, Toronto, Canada M5S 0A3 and [4]Department of Medical Biophysics, University of Toronto, Toronto, Canada M5S 1A8

Associate Editor: Martin Bishop

ABSTRACT

Motivation: Pattern discovery algorithms are widely used for the analysis of DNA and protein sequences. Most algorithms have been designed to find overrepresented motifs in sparse datasets of long sequences, and ignore most positional information. We introduce an algorithm optimized to exploit spatial information in sparse-but-populous datasets.

Results: Our algorithm Tree-based Weighted-Position Pattern Discovery and Classification (T-WPPDC) supports both unsupervised pattern discovery and supervised sequence classification. It identifies positionally enriched patterns using the Kullback–Leibler distance between foreground and background sequences at each position. This spatial information is used to discover positionally important patterns. T-WPPDC then uses a scoring function to discriminate different biological classes. We validated T-WPPDC on an important biological problem: prediction of single nucleotide polymorphisms (SNPs) from flanking sequence. We evaluated 672 separate experiments on 120 datasets derived from multiple species. T-WPPDC outperformed other pattern discovery methods and was comparable to the supervised machine learning algorithms. The algorithm is computationally efficient and largely insensitive to dataset size. It allows arbitrary parameterization and is embarrassingly parallelizable.

Conclusions: T-WPPDC is a minimally parameterized algorithm for both pattern discovery and sequence classification that directly incorporates positional information. We use it to confirm the predictability of SNPs from flanking sequence, and show that positional information is a key to this biological problem.

Contacts: ruiyan@cs.toronto.edu; paul.boutros@oicr.on.ca; juris@ai.toronto.edu

Availability: The algorithm, code and data are available at: http://www.cs.utoronto.ca/~juris/data/TWPPDC

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on January 7, 2011; revised on May 31, 2011; accepted on June 4, 2011

1 INTRODUCTION

Pattern discovery algorithms have been widely used in bioinformatics to analyze recurrent groups of symbols, such as DNA and protein sequences. Existing methods can be divided into two classes based on their underlying approach: probabilistic versus deterministic. Probabilistic methods maximize the likelihood between a motif pattern model and a background pattern model. Gibbs sampling and its variants are the prototypical probabilistic methods (Bailey and Elkan, 1995; Lawrence *et al.*, 1993). Some deterministic methods construct candidate patterns from a given pattern length and alphabet size [e.g. Oligo-Analysis/Dyad-Analysis (van Helden *et al.*, 2000) and YMF (Sinha and Tompa, 2003)], some enumerate possible patterns from given sequences [e.g. MOPAC (Ganesh *et al.,* 2003)], while others use tree structures [e.g. Weeder (Pevesi *et al.,* 2004)] or other mathematical approaches [e.g. Projection (Buhler and Tompa, 2002)]. Despite this methodological diversity, there are three issues not well-addressed by the existing algorithms.

First, positional variability in sequences is typically ignored. Most widely used pattern discovery methods focus on finding high-frequency sequences independent of their location, primarily representing patterns with position-specific scoring matrices (Sinha and Tompa, 2003) or Markov models (Thijs *et al.,* 2002). However, these techniques assume equal occurrence probabilities at all positions within a sequence. This assumption does not hold for many biological datasets (Birney *et al.,* 2007). A recent method Amadeus uses a localization score to estimate whether the occurrences of the motif tend to cluster at specific distance from the transcription start site (Linhart *et al.,* 2008); however, they focused on the specific distance or a range of sequences, not all the positions of each sequences. Second, it is unclear how to handle the numerous, low-information content motifs that occur in biological datasets. The presence of such motifs is a major reason for the low accuracy of current pattern discovery methods (Linhart *et al.,* 2008; Tompa *et al.,* 2005). Third, current algorithms focus on analyzing sparse datasets comprising a small number of long sequences. The advent of next-generation genome sequencing has led to much populous, short-sequence data.

To address these issues, we developed a new pattern discovery algorithm called Tree-based Weighted-Position Pattern Discovery and Classification (T-WPPDC). T-WPPDC first applies Kullback–Leibler distance between foreground and background sequences to determine a weight for each sequence position. It next integrates this spatial data to discover positionally important patterns. Such patterns are used to classify sequences. Moreover, the tree structure used by T-WPPDC allows handling of different

*To whom correspondence should be addressed.

pattern lengths, sequence lengths and alphabet sizes. In addition, the algorithm is embarrassingly parallel. Another tree structure method, Weeder (Pevesi *et al.,* 2004), applies a suffix tree to spell the sequences. Our method differs from Weeder since trees in T-WPPDC do not only hold the possible candidate of motifs, but also the positional weights. Furthermore, our method is used for classification.

We tested T-WPPDC on an important biological problem: predicting single nucleotide polymorphisms (SNPs). SNPs are single base pair variations in the genome, and are likely the most common form of genetic variation. On average, 1 out of every 1000 bp may be SNPs (Schafer and Hawkins, 1997). The function of most SNPs remains unclear; especially those not associated with changes in protein sequence. Genome-wide linkage analyses have implicated a large number of SNPs in disorders ranging from Crohn's disease (Vilani *et al.,* 2009) to cancer (Houlston *et al.,* 2008) and to quantitative traits such as height (Suzuki *et al.,* 2009). Many groups are attempting to predict the functional effects of individual SNPs (Li *et al.,* 2010; Ribas *et al.,* 2006), and this problem has grown in importance with the advent of cheap genome-sequencing (Hudson *et al.,* 2010).

In contrast, less research has focused on the causes of SNPs: why do they occur where they do? Zhao studied two SNPs databases such as Celera's CgsSNP and RefSNP and found that natural selection influences patterns of genome variations (Zhao *et al.,* 2003). Another study described a map with 1.42 million SNPs and showed that SNPs have been historically passed on across generations (Sachidanandam *et al.,* 2001). However, recent study has discovered the effect sequence position (Zhang and Zhao, 2004) and our previous work also characterized sequence-based determinants of SNPs (Yan *et al.,* 2007). Surprisingly, we were able to predict SNPs from flanking sequences alone using machine-learning methods. Pattern discovery methods failed at this task, with near-chance prediction accuracies (Yan *et al.,* 2007).

To further examine this question, we tested T-WPPDC on the DNA sequences flanking known SNPs. Information is unevenly distributed in these sequences, with many motifs showing positional bias. Moreover, some biological factors such as natural selection, non-uniform experimentation and stochasticity interfere with sequence-based SNP analysis. T-WPPDC is able to select informative motifs despite these factors and shows superior performance relative to existing pattern discovery methods.

2 MATERIALS AND METHODS

2.1 Datasets

We evaluated SNP datasets from different species: (i) human chromosomes 21 and 22 and (ii) human and mouse chromosome X (which have a high degree of synteny). Both SNP datasets were generated as described previously (Yan *et al.,* 2007), with minor modifications. Repeat-masked human genome sequence (build hg18), and mouse genome sequences (mm8) and SNP annotations were downloaded from the University of California, Santa Cruz (UCSC) genome browser database (Karolchik *et al.,* 2003) and dbSNP build126 for human and mouse. The SNP sequence is combined with equal length of 5′ and 3′ flanking sequence from chromosomes 21 and 22 for human, chromosome X for human and mouse, parsed by a Perl script (v5.8.8). We create a negative control sequence for each SNP by randomly selecting a base of the same reference allele from the same chromosome. To avoid SNPs under strong selection, we applied four criteria to both positive and negative sequences: first, SNPs within 5000 bp of any known exon (RefSeq annotation of February 8, 2007) were excluded. Second, SNPs with discordant UCSC and dbSNP allelic annotations were excluded. Third, non-SNP polymorphisms (e.g. indels) and SNPs with unknown strandedness were excluded. Fourth, SNPs with ambiguous or repeat-masked bases in their 5′ or 3′ flanking regions were excluded. These criteria removed two-thirds of SNPs. For each of the 12 SNP alleles, our dataset contains equal numbers of true positives (TP) and true negatives (TN). It is possible that some randomly selected negative controls are novel polymorphisms but this contamination is predicted to be under 0.1% (Yan *et al.,* 2007). For method evaluation, we divided each allelic dataset into equal-sized training and testing groups. Combined, there are 120 datasets are used, 72 datasets for chromosome 21 and 22 with symmetrical flanking sequences of 50, 100 and 150 bp (12 alleles × training/testing × 3 flanking lengths) and 48 datasets for chromosome X (12 alleles × training/testing × 2 species), which leads to 672 experiments (72×4 pattern lengths $+\,48\times2$ cross-species $\times\,4$ pattern lengths). Our final chromosomes 21 and 22 human datasets contain 50 bp sequences (average number of sequences ± SD: 5435 ± 4114), 100 bp sequences (4939 ± 3750) and 150 bp sequences (4518 ± 3434). Our final chromosome X datasets contain 50 bp sequences with 4188 ± 2703 for human and 9461 ± 6465 for mouse (shown in Supplementary Table S1).

2.2 Tree-based weighted position pattern discovery and classification (T-WPPDC)

T-WPPDC is designed to combine positional information and a novel scoring system to identify maximally predictive patterns. It uses a tree structure to greatly reduce algorithmic complexity. First, *Tree Construction* builds trees from training sequences. Each node of the tree holds a value representing the likelihood of a given pattern at a given sequence position. Second, T-WPPDC calculates the *Kullback–Leibler* $(K-L)$ *Distance* (Kullback, 1987; Kullback and Leibler, 1951) for each tree to measure positional differences. Third, *Leaf Selection* introduces a new scoring system that identifies high information content patterns to handle the low-signal motifs. Fourth, T-WPPDC scans through each test sequence and performs *Sequence Classification*.

Here are the notations used in this section: T-WPPDC requires two sets of sequences, foreground and background. To simplify the description, we use class A as foreground and B as background in this section. Letters N and L represent sequence number and length, respectively. We study DNA sequences, therefore, the alphabet is $M\in\{\text{A,C,G,T}\}$ and number of letters in the alphabet M is 4. As with most pattern discovery methods, the length of a pattern P, called K, must be fixed prior to analysis. We use l to describe a particular sequence, and lowercase letters, i and j represent a position within a particular sequence.

2.2.1 Tree construction T-WPPDC first constructs trees for the two sequence classes A and B. We scan all patterns with a K-width window from [0, L-1] in all sequences and use patterns at the i-th position construct tree-$A(i)$ and tree-$B(i)$. Therefore, for the training class A sequences, we built Forest-A of $(L-K+1)$ A-trees.

Tree Construction Rules:

- A node of a tree i has M children.
- A child has exactly one parent.
- Each child corresponds to an element of M.
- A node holds an alphabet, score V, and a pointer to its child.
- The depth of tree is the pattern length K.
- There are $L-K+1$ trees in each Forest.
- Tree-size depends on M (alphabet size) and K (pattern length).

One can generate a j-bp pattern by traversing the tree from the root node to the node at depth j (Supplementary Fig. S1). The score in each node at depth K, as shown $\text{LR}_A(P_K, i)$ in Formula 1, represents how likely this K-bp pattern P is to be found at position i in class A. Identical patterns in different

trees may have different score. This process is repeated identically on class B to build Forest-B with $(L-K+1)$ B-trees.

$$\mathrm{LR}_A(P_K,i)=\log\left(\frac{\text{Number of } K-\text{bp pattern } P \text{ at } i\text{-th position in sequence set } A}{\text{Total number of sequences in } A}\right) \quad (1)$$

2.2.2 Kullback–Leibler distance T-WPPDC uses *Kullback–Leibler* distance (K–L distance) to measure positional difference between class A and B. The K–L distance usually measures the distance from a true probability distribution p to a target probability distribution q (Kullback, 1987; Kullback and Leibler, 1951), but can also be viewed as the information content between two distributions. $\mathrm{KL}(A_i,B_i)$ is the distance between distributions A and B at position i $(0\le i\le L)$:

$$\mathrm{KL}(A_i,B_i)=\sum_{m=1}^{|M|}A_{i,m}\log(A_{i,m}/B_{i,m}) \quad (2)$$

where $A_{i,m}$ (or $B_{i,m}$) represents the probability of alphabet m shows at position i in class A (or B). Because K–L distance is asymmetric, $\mathrm{KL}(A_i,B_i)\neq\mathrm{KL}(B_i,A_i)$. The distance of each corresponding position i between A and B, $\mathrm{KLD}(i)$, can be defined as the sum of the $KL(A_i,B_i)$ and $KL(B_i,A_i)$. Each position in the sequence has an individual KLD, giving L distinct KLD weights. As above, we multiply the log frequency value (Formula 1) by KLD for each node at class A:

$$V_A(i,j)=\sum_{K=1}^{j}\mathrm{LR}_A(P_K,i)\times\mathrm{KLD}(i+K) \quad (3)$$

where $V_A(i,j)$ represents the node value at depth j for tree i in class A. The same process is repeated identically on each node on class B. In other words, the value at each node is the sum of its parents and its own weighted value. These weighted values are the product of the normalized log frequency ratio and the K–L distance. Intuitively, summing the parent weighted scores reflects the influence of neighboring positions. The weighted value for a node at depth j for tree i reflects the likelihood that this j-bp pattern occurs at position i. Therefore, each leaf node collects the sum of all its ancestors and every node in a tree hold values reflecting the likelihood that a specific pattern occurs at this position. This design allows handling of arbitrary pattern lengths.

2.2.3 Leaf selection As noted above, some patterns with low information simply reflect noise in the dataset and do not aid in discrimination. T-WPPDC is designed to select high signal patterns. Filtering based on signal strength trades sensitivity for increased specificity. The difference between corresponding leaf nodes from Forest-A and Forest-B (i.e. $V_A(i,j)-V_B(i,j)>\epsilon$) shows how well the pattern distinguishes A from B. If this difference is positive, then it is overrepresented in class A. The larger this difference, the more informative a pattern is. By altering ϵ we can change the sensitivity–specificity trade-off. Theoretically, $\epsilon\ge 0$ and should be set small enough to determine the optimal performance. Performance varies with different values of ϵ and T-WPPDC defines ϵ with n equally spaced points in the range $[V_A(i,j)-V_B(i,j)]$, shown in Formula 4. To optimize ϵ, one can preset n and choose the local/global optimal performance accordingly. When ϵ is zero leaf selection is disabled, meaning that all leaves are used for classification.

$$\varepsilon=\frac{|V_A(i,j)-V_B(i,j)|}{n} \quad (4)$$

2.2.4 Sequence classification As in the training procedure, a K-width window is applied to a test sequence, l. The matching pattern at each position is selected from Forest-A and Forest-B. A final score, $A(l)$, is calculated as the sum of all K-bp patterns in l, as below in Formula 5. Patterns that do not meet the leaf selection threshold, ϵ, will not contribute. This process is repeated on Forest-B to generate $B(l)$. The sequence l then is classified to the class with the larger score. The detailed workflow is shown in Figure 1.

$$A(l)=\sum_{i=0}^{L-K+1}V_A(i,K) \quad (5)$$

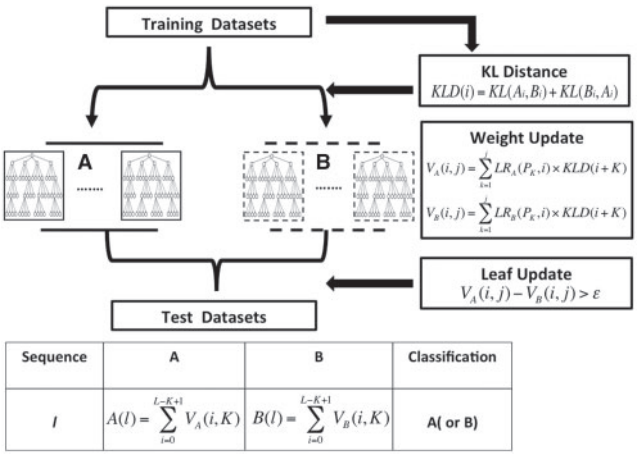

Fig. 1. Overview of T-WPPDC. T-WPPDC includes two phases: training and testing. During the training phase, it builds weighted trees from training sequences sets A and B. Each sequence set has $L-K+1$ trees (called Forest-A and Forest-B, respectively), and each tree corresponds to the patterns at a specific sequence position. Leaf selection chooses patterns that satisfy the criteria [i.e. $V_A(i,j)-V_B(i,j)>\epsilon$] for further classification. During the testing phase, each K-bp pattern from a test sequence l is evaluated for the matching patterns from trees in Forest-A. A final score $A(l)$ is calculated from Formula 5. The process is repeated to generate $B(l)$. Sequence l is then classified to a class A(or B) with a larger final score $A(l)$ [or $B(l)$].

2.3 Implementation

Assuming it takes a unit time t_1 to construct a single tree, T-WPPDC requires $(L-K+1)\times t_1$ to construct each Forest. Updating nodes with K–L distance therefore requires $(L-K+1)\times t_2$, assuming t_2 time per tree for updating. Leaf selections can be combined with sequence classification, involving matching patterns from testing sequence to the trees, which requires $\log K$ comparisons per pattern, leading to $N\times(L-K+1)\times\log K$ comparisons. Therefore, the computation complexity of T-WPPDC is a polynomial function, $\mathrm{O}(N\times(L-K)\times\log K)$ [i.e. $N\times(L-K)\times\log K+(L-K)=(L-K)\times(N\log K+1)$]. T-WPPDC loads the Forests into memory, with $(L-K+1)$ trees. Each tree size depends on the alphabet number M and the depth of trees K, $|M|^K$, giving space requirements of $\mathrm{O}(L\times|M|^K)$.

Tree structure design provides flexibility. It is easy to adapt to different alphabets or sequence and pattern lengths. As noted above, each tree holds patterns starting at one sequence position. The current implementation handles DNA sequences ($|M|=4$). Running time is unaffected by different alphabets, while total memory usage is a function of alphabet size.

To further reduce running time and memory requirements, T-WPPDC can be distributed to multiple nodes for parallel computing. As trees are independent from one another, each can be distributed to a separate computing node for simultaneous K–L distance calculation, leaf selection and classification. Thus, parallel memory requirements are only $\mathrm{O}(|M|^K)$, and computation complexity reduces to $\mathrm{O}(N\log K)$.

T-WPPDC was implemented in C++ with the STL-like template tree class (http://www.aei.mpg.de/~peekas/tree/). While we expect it to be platform independent, we have only tested it under RedHat Linux (v2.6.24) and compiled with GCC (v4.2.4). To evaluate computational performance in a consistent way, we ran all pattern discovery algorithms on a single system.

2.4 Related work

Recently, two other groups have described approaches to incorporate positional information into pattern discovery. The first group developed a probabilistic method that incorporates positional information into priors for Multiple Em for Motif Elicitation (MEME) analysis (PSP-MEME), which

they tested on 156 yeast ChIP-chip datasets (Bailey *et al.,* 2010). The second group developed an approach called LocalMotif that applies three scoring schemes (Narang *et al.,* 2010). Both PSP-MEME and LocalMotif require prior information on the motif, with PSP-MEME assuming OOPS (one motif per sequence) or ZOOPS models (zero or one motif per sequence) and not supporting ANR models (arbitrary number of motifs per sequence). LocalMotif requires the number of motifs to output, the number of candidates and the weights for three scoring schemes. Most of these parameters are unknown and must be estimated. Moreover, LocalMotif involves an exhaustive enumeration strategy, causing exponential complexity with pattern length, while T-WPPDC has polynomial complexity with pattern length (K), $O(N \times (L-K) \times \log K)$. Aside from these algorithmic details, T-WPPDC is the only one of these techniques that can do both motif discovery and sequence classification.

2.5 Pattern discovery analyses

To evaluate the performance of T-WPPDC, we used three publicly available pattern discovery methods: BioProspector (version. 2004) (Liu *et al.,* 2001) and Oligo (downloaded on September 21, 2007) (van Helden *et al.,* 2000) and LocalMotif (Narang *et al.,* 2010). We ran each method (including T-WPPDC) using default parameter settings. Oligo returned all possible motifs for analysis ($n = |M|^K$). We ranked the motifs by Z-score and selected the top 20 overrepresented (or underrepresented) motifs. The value 20 was selected to allow these methods to slightly exceed the number of motifs identified by BioProspector.

First, BioProspector, Oligo and LocalMotif were run separately on the foreground (true positive, TP) and background (true negative, TN) datasets. BioProspector and LocalMotif returns both consensus motifs and position weight matrices; Oligo only returns consensus motifs. Second, motifs from Oligo found identically in the TP and TN sets were excluded. Third, a Perl script (v5.8.7) was used to scan through each test sequence for motifs found only in the TP or TN training datasets. If a motif occurred exactly in the test sequence one or more times, then the sequence was predicted to have the same class (SNP or non-SNP) as the motif. Fourth, the actual and predicted classes for all test sequences were used to generate two-way tables giving the number of TPs, TNs, false-positives (FPs) and false-negatives (FNs). Fifth, overall accuracy was calculated from this two-way table in the standard way [i.e. (TP + TN)/(TP + TN + FP + FN)]. Sixth, these steps were repeated with inversion of the testing and training datasets. Seventh, position weight matrices from BioProspector and LocalMotif were analyzed by using logistic regression analysis (Wasserman and Fickett, 1998). Finally, the entire procedure was repeated separately for each pattern discovery algorithm for sequence lengths of 50, 100 and 150 bp and pattern lengths in [3,6] for Oligo and LocalMotif and [4,6] for BioProspector (BioProspector has a minimum pattern length of 4 bp). The pattern discovery methods were run on two Linux GNU 2.6.15-29 64-bit servers with identical hardware/software configuration, and compiled with GCC (v4.0.3). Logistic regression analysis was run using Weka (v3-6-2).

2.6 Machine-learning analyses

Two machine-learning algorithms were employed for comparison: Random Forests (RF) and K-nearest neighbors (KNN). These methods are well-established, minimally parameterized techniques representative of diverse fields of machine learning. RF is a non-metric classification technique (Breiman, 2001) that uses an ensemble of decision trees. KNN is a standard non-parametric classification method (Duda *et al.,* 2001). Machine-learning analyses were performed largely as described previously (Yan *et al.,* 2007). Each base of flanking sequence was expanded into four predictor variables, one for each nucleotide, and was assigned a binary value from the sequence. Machine-learning methods were implemented in the R statistical environment (v2.5.1) with RandomForests in the RandomForest package (v4.5-18) and KNN in the class package (v7.2-36). RF contained 1000 trees and K was set to 51 for KNN analysis, based on our previous analysis (Yan *et al.,* 2007).

2.7 Data visualization and statistical analyses

All plots and statistical analyses were generated in the R statistical environment (v2.11.1) using the lattice (v0.19-11) and latticeExtra (v0.6-14) packages. To evaluate the relationship between SNP position and weight w, we employed Spearman's rank-order correlation using the absolute value of the position as Spearman's correlation has minimal assumptions: (i) two variables are ordinal, interval or ratio; and (ii) there is a monotonic relationship between variables. As such, it can be used as a non-parametric test. Statistical analysis of differences in means was determined using unpaired, two-tailed Wilcoxon signed-rank test, due to its minimal assumptions: (i) the differences of two observations are assumed to be independent; and (ii) directional comparisons can be given. Both these assumptions are met in the datasets used here. To improve visualization, weights were scaled for plotting as: $w_{\text{scaled}} = w - \min(w)$. Similarly, information content values were scaled as: $\text{IC}_{\text{scaled}} = 0.3 \times [\log_2|\text{IC}| - \min(\log_2|\text{IC}|)]$ (0.3 is chosen for better visualization). These operations are monotonic and only alter the centering of the distributions. Pattern discovery methods CPU running time was loess-smoothed with a span of 0.2 (0.2 is chosen for the best smoothing fit).

3 RESULTS

3.1 Distribution of the K–L distance

Our previous work showed that classification methods greatly outperform pattern discovery algorithms in predicting SNPs from flanking sequences (Yan *et al.,* 2007). Machine-learning methods can incorporate positional information, and so to assess the importance of this factor, we performed an information content analysis. For all SNPs that met our stringent inclusion criteria (Section 2.1), we extracted 50 bp of flanking sequence centered on the SNP. We selected an equal number of 50 bp sequences flanking non-SNP positions on the same chromosomes. For each position and for each of the 12 SNP alleles, we calculated the mutual information as:

$$\text{IC}_{\text{pos}} = \sum_b A_b \log_2 \frac{A_b}{B_b} \quad (6)$$

Here b is the nucleotide base {A,C,G,T} and A_b (B_b) is the frequency of base b at the current position from sequence A (or B). In our analysis, we set sequence set A as the TPs and B as the TNs. Figure 2 shows that positions close to the SNP (position 0) are generally more informative than those further away. However, some positions that are quite distal to the SNP itself carry information.

To incorporate this information directly into pattern discovery, we developed T-WPPDC, which uses the K–L distance (Kullback, 1987; Kullback and Leibler, 1951). K–L distance captures positional dependency in information content (Supplementary Fig. S2). For most (10/12) alleles, the distance of a position to the SNP is proportional to the K–L distance (i.e. Spearman's rank-order correlation is significant, $P < 0.05$). This holds true for flanking sequences of 50, 100 and 150 bp from human chromosomes 21, 22 and X.

As noted above, each leaf node holds a value that represents the likelihood that a specific pattern occurs at this position. As shown in Figure 3, the Foreground scores come from the leaf values from Forest-A (representing the Foreground class) and Background scores are from the leaf values of Forest-B (representing the Background class) (Fig. 3). Symmetric scores indicate little difference between

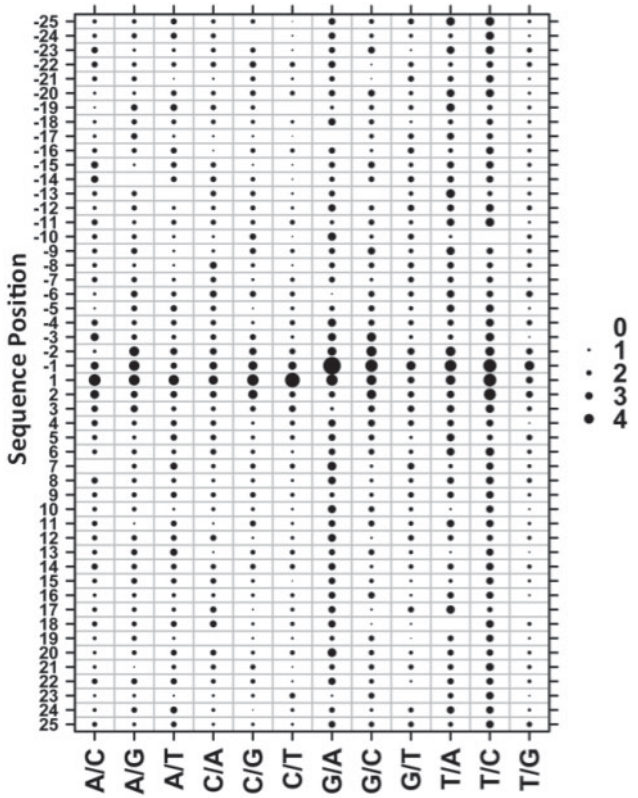

Fig. 2. Normalized SNP information content by position (sequence length: 50 bp). Columns represent the information content of flanking DNA from 25 bp 3′ to 25 bp 5′ of the SNP (which is at position 0). The size of each dot is proportional to the normalized information content (IC).

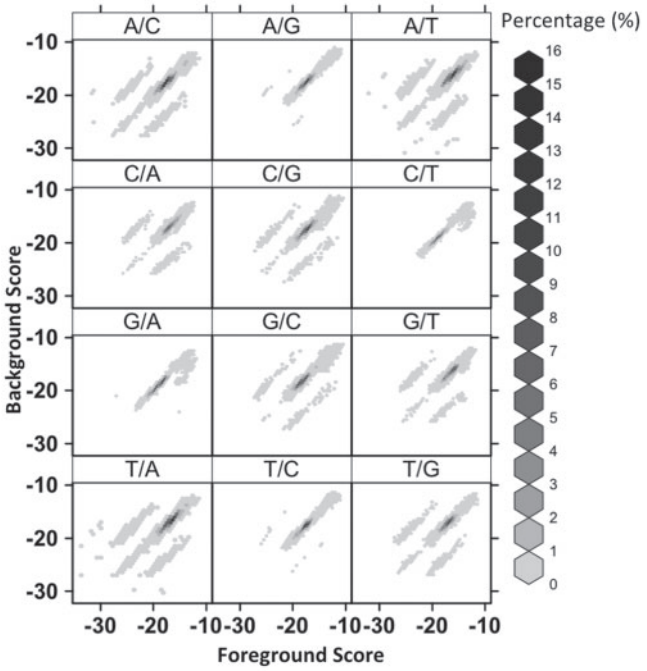

Fig. 3. SNP pattern foreground score versus background score (sequence length: 50 bp, pattern length: 4 bp, from training sequences). The *x*-axis shows the foreground scores for each candidate patterns. The *y*-axis gives the background scores for the corresponding patterns. Calculation for the score is shown in Formula 3. Shading represents the density of points in a region.

motifs from foreground and those from background. Most SNPs have symmetric score distributions except C/T and G/A SNPs (see Supplementary Figs 3–5), which shows that the C/T and G/A are more different from background. This might explain the better prediction performance of C/T and G/A alleles that is discussed below.

3.2 Prediction accuracy

T-WPPDC was evaluated by selecting half of each dataset for training and half for testing/validation. The training and testing datasets were then inverted and the procedure repeated (i.e. 2-fold cross-validation). We used sequences of lengths 50, 100 and 150 bp for SNPs from human chromosomes 21, 22 and X. We also performed four species-specific tests: Hs-Hs (train and test on human sequences), Mm-Mm (train and test on mouse sequences), Hs-Mm (train on human sequences and test on mouse sequences) and Mm-Hs (train on mouse sequences and test on human sequences). We used four different pattern lengths of 3–6 bp. In total, we conducted 672 experiments: 288 experiments on chromosome 21/22 and 384 experiments on chromosome X. We focus on overall prediction accuracy [(TP + TN)/(TP + TN + FP + FN)], but report multiple metrics (Supplementary Tables S2 and S3).

T-WPPDC with leaf selection off consistently outperforms existing pattern discovery methods. First, the overall prediction accuracy is $55.5 \pm 3.6\%$ when threshold $\epsilon = 0$ (Fig. 4 for chromosome 21/22 and Supplementary Fig. S6 for chromosome X). This is a statistically significant improvement over Oligo, BioProspector and LocalMotif (Oligo: $50.0 \pm 0.8\%$, $P = 5.61 \times 10^{-81}$; BioProspector: $50.8 \pm 0.9\%$, $P = 1.46 \times 10^{-72}$; LocalMotif: $51.3 \pm 1.2\%$, $P = 1.39 \times 10^{-70}$). We are unable to detect a difference between T-WPPDC in this naïve mode and KNN, a fully supervised method ($54.9 \pm 2.9\%$, $P = 0.61$), but remains lower than that of RF ($58.9 \pm 3.8\%$, $P = 2.66 \times 10^{-4}$).

These modest prediction accuracies suggest that not all SNPs are predicable because of non-sequence factors like natural selection. To demonstrate this, we exploited the fact that not all patterns carry equivalent information. Patterns with similar frequencies in TP and TN cases are mostly noise and have little independent predictive capacity. Enabling leaf selection helps to removes these patterns. The performance varies with threshold stringency (i.e. ϵ value), so we executed T-WPPDC at 20 equally spaced point (i.e. $n = 20$ in Formula 4) (Fig. 5). We chose the last point prior to any decline as the optimum position. Leaf-selection improves median accuracy, at the expense of increased SDs. In this mode, it classifies ~40% of sequences (Supplementary Fig. S7) and outperforms KNN on the Chromosome X ($P = 2.00 \times 10^{-3}$) and RF on Chromosome 21/22 ($P = 0.05$) datasets. T-WPPDC with leaf selection also is statistically indistinguishable from performance of KNN in the Chromosome 21/22 datasets ($P = 0.24$) and RF in Chromosome X ($P = 0.69$) datasets. We note that these analyses are not precisely comparable as we did not exploit the potential of RF or KNN to prioritize sequences based on vote numbers.

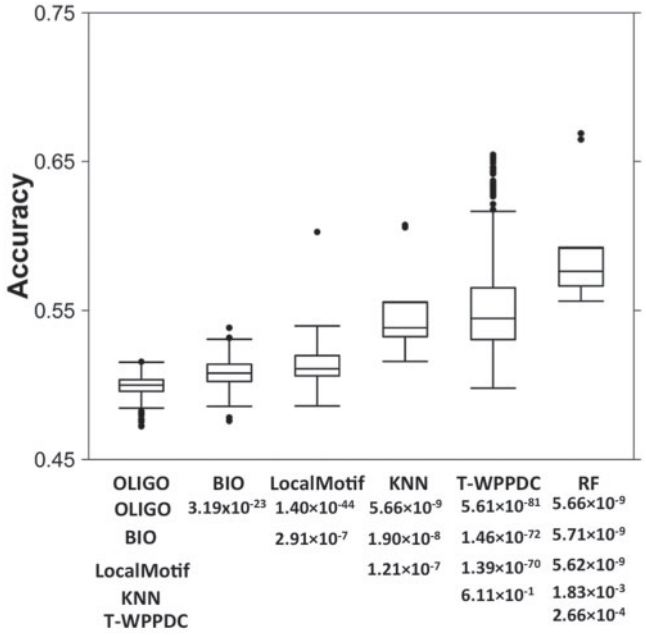

Fig. 4. Accuracy comparisons across all methods. (Data: chromosome 21/22) A boxplot of algorithms evaluated in this study (from left to right: pattern discovery method-OLIGO, BIO, LocalMotif, classification method-KNN, T-WPPDC with leaf selection off option, RF). The *y*-axis is the overall prediction accuracy [(TP + TN)/(TP + TN + FP + FN)], ranging from 0.47 to 0.67. T-WPPDC with leaf selection off includes all the patterns for prediction. A *P*-value is calculated by Wilcoxon signed-rank test, unpaired, two sided (OLIGO, Oligo; BIO, BioProspector; LocalMotif, LocalMotif).

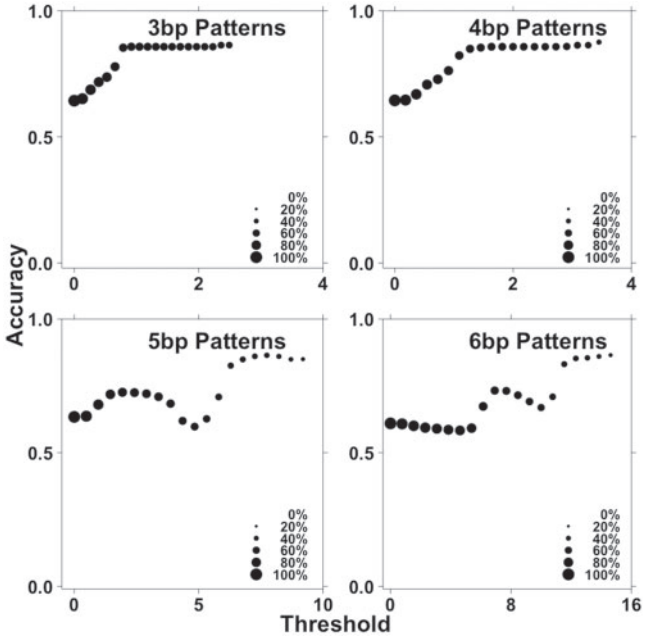

Fig. 5. Prediction accuracy (sequence length: 50 bp, SNPS: C/T, training from chromosome 21/22 training sequences and testing from chromosome 21/22 testing sequences). The *x*-axis shows the range of the threshold ϵ. The *y*-axis shows the prediction accuracy. Size for each dot represents the coverage rate, range from 0.1% to 100%, which means the numbers of sequences are identified. The point at $x = 0$ represents no leaf-selection.

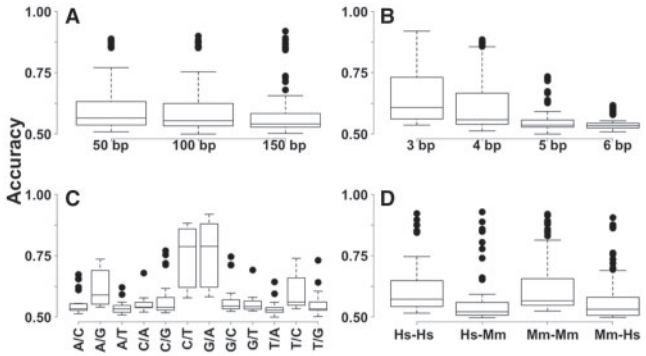

Fig. 6. Prediction accuracy impact from species, pattern length, SNPs and datatype (leaf selection on). (**A**–**C**) Shows the performance impact from different sequence length, pattern length and SNPs from chromosome 21/22. (**D**) Shows the performance impact from different species from chromosome X: Hs-Mm: intraspecies tests, Hs-Mm/Mm-Hs: cross-species tests.

To determine if prediction accuracy is a function of dataset characteristics, we analyzed sequence length, pattern length, SNP allele and species (Fig. 6). Shorter sequence (Fig. 6A) and pattern lengths (Fig. 6B) perform better than longer ones. For example, patterns with 3 bp perform on average 12% better than those with 6 bp long. More datasets should be examined in the future to explicitly test this observation. In all datasets, the C/T and G/A alleles have the best prediction accuracy (15–16% better than the overall median performance, $P = 8.68 \times 10^{-9}$ for C/T and $P = 1.98 \times 10^{-9}$ for G/A, Fig. 6C), reflecting the powerful (and known) influence of the bases immediately adjacent to the SNP for these alleles. We also noticed that interspecies performance is significantly better than the cross-species test ($P = 2.77 \times 10^{-11}$ for Hs species test, mean accuracy for Hs-Hs: 61.0%, Hs-Mm: 58.6%; and $P = 4.82 \times 10^{-9}$ for Mm species test, mean accuracy for Mm-Mm: 62.4%, Mm-Hs: 56.8%, Fig. 6D), which suggests that patterns are partially conserved across species.

3.3 CPU performance

TWPPDC is an efficient algorithm with $O(N \times L \times \log K)$ running time. Of our 672 experiments, 95% finished within 2 min on a Linux GNU 2.6.15-29-AMD64 server with Intel® Xeon® x5355 2.66 GHc CPUs and 16 GB RAM. T-WPPDC runs, on average, 39.7% faster than Oligo ($P = 0.01$) and 46.5% faster than BioProspector ($P = 1.33 \times 10^{-43}$). Figure 7 demonstrates that T-WPPDC is able to provide a lower bound of CPU running time (solid line), suggesting it is less sensitive to data size.

4 CONCLUSIONS AND FUTURE WORK

4.1 Conclusions

We have developed a new pattern discovery algorithm, T-WPPDC, which identifies positional biases between two classes of sequences. We optimized search processes with tree structures, which dynamically improve the computation complexity and allow superior parallelization. By incorporating the pattern selection, T-WPPDC is able to reach the optimal classification performance.

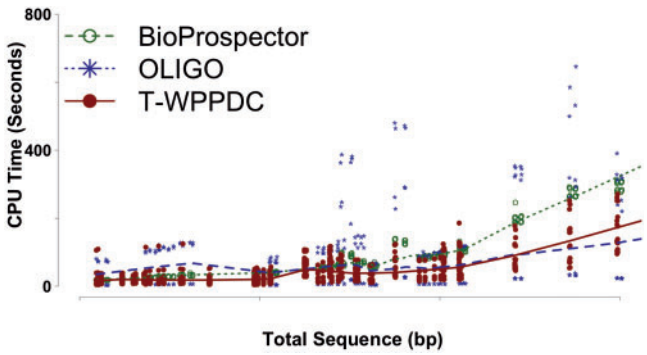

Fig. 7. CPU analysis versus total sequence (in base pair). CPU time (in seconds) as a function of the total amount of sequence in each dataset (in base pair). Each algorithm is plotted in different shapes, representing individual datasets and loess-derived lines of best-fits are plotted with a span of 0.2.

It was designed to perform well for large datasets and especially those with relatively short position size. Such datasets are increasingly common, given the high resolution of next-generation sequencing.

Our previous work demonstrated the possibility of SNP prediction from flanking DNA (Yan *et al.,* 2007), but indicated that the presence/absence of specific short patterns was not predictive. Here, we demonstrate that integrating positional and frequency data improves SNP prediction accuracy, superior to the best pattern-discovery algorithms and comparable to the best machine learning methods tested. Compared to the existing methods, T-WPPDC requires no prior information of motifs and are capable of performing both discovery and classification efficiently $[O(N \times L \times \log K)]$.

Our work also suggested that not all SNP sequences are predicable. Some are more dominated by the nature selection and random chance, while others can be predicated by short patterns. For example, C/T and G/A SNPs can reach 79% median prediction accuracy while T/A SNP can only receive 53% median accuracy. Furthermore, T-WPPDC is an efficient and flexible method that can be adapted to other sequences and parallel computing systems.

4.2 Future work

T-WPPDC is a flexible and effective algorithm. Current design supports variable pattern length, sequence length and alphabets. Future expansions can focus on three main areas. First, the algorithm is embarrassingly parallel, so a parallel implementation would be advantageous. Second, the binary classifier can be extended to many groups. Third, T-WPPDC currently assumes exact positional conservation and many overlook patterns with high information content but weak or fuzzy positional conservation. Future work will expand T-WPPDC to search for positional conservation in moving windows and/or to use fuzzy motif matching. Finally, it is possible that T-WPPDC provides a way of assessing the importance of positional information in specific datasets.

ACKNOWLEDGEMENTS

The authors thank Dr Kevin Brown for system administration and Dr Mehrdad Shamsi for helpful suggestions.

Funding: Natural Sciences and Engineering Council (to R.Y.); IBM Center for Advanced Studies (to R.Y.); Canadian Institutes of Health Research (to P.C.B.; MOP-57903); Canada Foundation for Innovation (grants #12301 and #203383 to I.J.); Canada Research Chair Program (to I.J.); Ontario Institute for Cancer Research (to P.C.B. through funding provided by the Government of Ontario); Ontario Ministry of Health and Long Term Care (MOHLTC, in part). The views expressed do not necessarily reflect those of the MOHLTC.

Conflict of Interest: none declared.

REFERENCES

Bailey,T.L (2010) The value of position-specific priors in motif discovery using MEME. *BMC Bioinformatics*, **11**,179.

Bailey,T.L. and Elkan,C. (1995) The value of prior knowledge in discovering motifs with MEME. In *Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology*. AAAI Press, Menlo Park, CA, pp. 21–29.

Birney, E. *et al*. (2007) Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project. *Nature* **447**, 799–816.

Breiman,L. (2001) Random forests. *Mach. Learn.,* **45**, 5–32.

Buhler,J. and Tompa,M. (2002) Finding motifs using random projections. *J. Comput. Biol.*, **9**, 225–242.

Duda,R. *et al.* (2001) *Pattern Classification*, 2nd edn. John Wiley and Sons, Inc., New York.

Ganesh,R. *et al.* (2003) MOPAC: motif binding by preprocessing and agglomerative clustering from microarrays. *Pac. Symp. Biocomput.*, **8**, 41–52.

Houlston,R.S. (2008) Meta-analysis of genome-wide association data identifies four new susceptibility loci for colorectal cancer. *Nat. Genet.*, **40**, 1426–1435.

Hudson,T.J. *et al.* (2010) International network of cancer genome projects. *Nature*, **464**, 993–998.

Karolchik,D. *et al.* (2003) The UCSC genome browser database. *Nucleic Acids Res.*, **31**, 51–54.

Kullback,S (1987) Letter to the editor: the Kullback-Leibler distance. *Am. Stat.*, **41**, 340–341.

Kullback,S and Leibler,R.A. (1951) On information and sufficiency. *Ann. Math. Stat.*, **22** , 79–86.

Lawrence,C.E. *et al.* (1993) Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science*, **262**, 208–214.

Li,K. *et al.* (2010) Association between the RAGE G82S polymorphism and Alzheimer's disease. *J. Neural Transm.*, **117**, 97–104.

Linhart,C. *et al.* (2008) Transcription factor and microRNA motif discovery: the Amadeus platform and a compendium of metazoan target sets. *Genome Res.*, **18**, 1180–1189.

Liu,X. *et al.* (2001) BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. *Pac. Symp. Biocomput.*, 127–138.

Narang,V. *et al.* (2010) Localized motif discovery in gene regulatory sequences. *Bioinformatics*, **26**, 1152–1159.

Pevesi,G. *et al.* (2004) Weeder WEB: discovery of transcription factor binding sites in a set of sequences from co-regulated genes. *Nucleic Acids Res.*, **32**, W199–W203.

Ribas,G. *et al.* (2006) Evaluating HapMap SNP data transferability in a large-scale genotyping project involving 175 cancer-associated genes. *Hum. Genet.*, **118**, 669–679.

Sachidanandam,R. *et al.* (2001) A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms. *Nature*, **409**, 928–933.

Schafer,J.A. and Hawkins,J.R. (1997) DNA variation and the future of human genetics. *Nat. Biotechnol.*, **16**, 33–39.

Sinha,S. and Tompa,M. (2003) YMF: a program for discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic Acids Res.*, **31**, 3586–3588.

Suzuki,H. *et al.* (2009) The transcriptional network that controls growth arrest and differentiation in a human myeloid leukemia cell line. *Nat. Genet.*, **41,** 553–562.

Thijs,G. *et al*. (2002) A Gibbs sampling method to detect overrepresented motifs in the upstream regions of coexpressed genes. *J. Comput. Biol.*, **9**, 447–464.

Tompa,M. *et al*. (2005) Assessing computational tools for the discovery of transcription factor binding sites. *Nat. Biotechnol.*, **23**, 137–144.

Van Helden,J. *et al.* (2000) Discovering regulatory elements in non-coding sequences by analysis of spaced dyads. *Nucleic Acids Res.*, **28**, 1808–1818.

Vilani,A.C. *et al*. (2009) Common variants in the NLRP3 region contribute to Crohn's disease susceptibility. *Nat. Genet.*, **41**, 71–76.

Wasserman,W.W. and Fickett,J.W. (1998) Identification of regulatory regions which confer muscle-specific gene expression. *J. Mol. Biol.*, **278,** 167–181.

Yan,R. *et al.* (2007) Comparison of machine learning and pattern discovery algorithms for the prediction of human single nucleotide polymorphisms. In *IEEE International Conference on Granular Computing (GRC 2007)*. San Jose, CA, pp. 452–457.

Zhao,Z.M. *et al.* (2003) Investigating single nucleotide polymorphism (SNP) density in the human genome and its implications for molecular evolution. *Gene*, **312**, 207–213.

Zhang,F.K. and Zhao,Z.M. (2004) The influence of neighboring-nucleotide composition on single nucleotide polymorphisms (SNPs) in the mouse genome and its comparison with human SNPs, *Genomics*, **84,** 785–795.