Locally Optimal Parameter Settings for Pattern Discovery in Gene Expression Microarray Data

ABSTRACT

Pattern discovery on gene expression microarray data is an emerging analysis technique. It provides specific advantages over the more commonly used clustering techniques [Rigoutsos, 00]. Due to the NP-Hard nature of pattern discovery, various heuristics are employed in pattern discovery algorithms such as Teiresias. This paper explores the effects of varying parameter settings in Teiresias in terms of the patterns that are discovered in yeast cell cycle data. A metric that relates execution time to proportion of patterns discovered is developed and used to determine that Teiresias is most efficient at finding patterns when minimum-number-of-literals and window-size are set to 5 and 12, respectively. The metric and methodology are applicable to other pattern discovery algorithms and sources of data.

CONTACT

Matthew Bainbridge: Department of Computer Science, University of Saskatchewan, Saskatchewan, Canada. *mnb922@cs.usask.ca*. +1 (306) 966 2075.
Tony Kusalik: As above. *kusalik@cs.usask.ca*. +1 (306) 966 4904.

KEYWORDS: Pattern discovery, Teiresias, Gene expression analysis, Microarray data analysis, Bioinformatics, Gene clustering.

1. INTRODUCTION

Pattern discovery has been successfully applied to DNA microarray data [Califano, 00; Rigoutsos, 00] because it offers several advantages over the more commonly used clustering methods [Rigoutsos, 00]. However, pattern discovery is an NP-Hard problem [Brazma, 98] and its real-world application requires the use of heuristics. Teiresias is a pattern-finding algorithm that employs such heuristics. A disadvantage of the use of heuristics is that every important pattern may not be found.

Unfortunately, published accounts of applying Teiresias to microarray data do not provide recommendations for parameter settings nor describe how varying parameter values affect efficiency and completeness. The same is true for other pattern discovery algorithms such as SPLASH [Califano 99; 00]. This paper develops an efficiency metric for Teiresias, applies it to pattern discovery on the gene expression microarray data of Spellman et al. [Spellman, 98; Yeast, 02], and based on the results, determines the most efficient parameter settings for Teiresias.

2. BACKGROUND

A pattern is a series of two or more characters shared between multiple strings (hence forth called *streams*) that is considered *interesting* [Rigoutsos, 00]. The definition of "interesting" is application specific. A pattern is specified by a template and a list of occurrences. The template is a series of *literals* optionally interspersed with "don't care" characters (the '.' character). Each item of the list of occurrences records a stream identifier and the position (*offset*) of occurrence. The cardinality of the list is the *support* for the pattern. Patterns that occur with the same offset in all streams are considered *aligned*. If the offsets differ by no more than a finite amount, the patterns are *unaligned*. Thus all aligned patterns are also unaligned. In Figure 1, the unaligned pattern P1 occurs in streams 1 and 2 at an offset of 2 and 3, respectively. In this work, "interesting patterns" are all patterns where the offsets vary by only a small amount (e.g. abs(MAX(offset)-MIN(offset))<=2) and which contain a large number of literals.

Figure 1

```
Stream 1  AYAJKBAY
Stream 2  IUVAOPBA
P1:"A..BA"{(1,2),(2,3)}
```

Teiresias is a two-stage pattern-finding algorithm. In stage 1, it uses two parameters, L and W, to establish a minimum *density* for a given *elementary pattern*. Density is the ratio of the positions that are occupied by literals over a pattern's length [Rigoutsos, 00]. Parameter L is the minimum number of literals to appear in a window of size W. It follows that the largest *gap* that may appear in a pattern is W-L. In stage 2, elementary patterns are convoluted together to form larger patterns. In Figure 1, P1 could have been found using either L=3 W=5, which would have found "A..BA" as an elementary

pattern, or L=2 W=4 which would have found the patterns "A..B" and "BA" as elementary patterns and convoluted them together to form "A..BA".

As the difference between L and W is increased and/or L is lowered, the number of patterns found increases exponentially. The only way to find every pattern is to set L equal to 2, W equal to the length of the stream and the support, k, to 2. However, the vast majority of the resulting patterns will be "uninteresting" (short) and the execution time will be long. The goal of tuning the values of L and W is to enrich the number of interesting patterns in the result set and to keep execution time reasonable. Unfortunately, descriptions of Teiresias do not outline how these values should be tuned.

3. METHODOLOGY

Teiresias was used to discover patterns in the test data described in Section 3.1. For each set of parameters, the execution time and number of discovered patterns of each length was recorded. Initially, "modest" values for L and W were selected (L=4, W=7). Subsequently, the value of W was increased until execution time became unacceptably long (over 24 hours), whereupon L was increased (which reduced execution time) and W was again incremented. This "inchworm-ing" process was repeated until it became clear from the metric established in Section 3.3 that the ratio of time cost to completeness was not likely to improve. From these data the most efficient values for L and W were chosen.

3.1 Data

Two subsets of publicly available yeast cell cycle data [Yeast, 02] were used. The first subset was a handpicked selection of 24 genes (CHS3, KAR4, HO, MNN1, SWE1, CLN2, TIP1, PSA1, CLN1, FKS1, KRE6, MCD1, HTB1, GOG5, SPC42, GAS1, PL30, PSD1, HTA1, RFA3, RAD27, RFA1, HHF1, HTT1) which are well studied and known to be co-regulated. In this set we expect to find many long patterns. The second subset, 24 randomly selected genes (YBL089W, YCR072C, YDR464W, YGR008C, YJR004C, YMR088C, YNL328C, YOR001W, YBR223C, YDL067C, YER087C-A, YHR139C, YML003W, YMR161W, YNR022C, YPL074W, YCR041W, YDR379W, YER164W, YJL109C, YML032C, YNL112W, YOL004W, YPR021C), acts as a control where we have no expectation of outcome. This set was used to compare results obtained from the first data set against what might occur "at random". Each of these data sets was treated in two ways: 1) no manipulation (the data is then called "non-break") or 2) external symbols were introduced in such a way as to break-up all strictly unaligned patterns ("break" data). Unaligned patterns form the majority of patterns in a result set. By breaking up these patterns Teiresias found all aligned patterns in the "break" data quickly. After pattern discovery was completed the patterns in the result set were placed into "bins" according to the number of literals in the pattern (the "bin size"). The bins used were: >60, >50 (but less than 61), >40 (but less than 51), and >30.

3.2 Evaluation metric

A metric that fairly measures the efficiency of a parameter set, Π, should take into account two things: 1) how many patterns (with a certain number of literals) is found using Π compared to how many patterns of that size exist (completeness), and 2) how long it takes to find those patterns. It is typically infeasible to compute the total number of unaligned patterns in even a small microarray data set. Fortunately, it is reasonable to assume that the proportion of interesting, unaligned patterns found using Π to all unaligned patterns in the data set is approximately equal to the ratio of interesting, aligned patterns found using Π to all aligned patterns present. This assumption is born out by additional experimentation which is not reported due to space limitations. For bin size X, define an estimate of completeness, $P_{X\Pi}$, to be (number of patterns in non-break data discovered using Π)/(total number of patterns in the data set). By the previous assumption $P_{X\Pi}$ for aligned patterns should be approximately equal to $P_{X\Pi}$ for unaligned patterns. When calculating P_XPI for aligned patterns, the denominator can be determined using "break" data. The numerator is still determined using "non-break" data. With respect to point 2, it is known that the execution time of Teiresias is linear in output [Floratos, 98]. However, interesting patterns comprise only a small fraction of the total number of patterns discovered and any interesting pattern arises from an exponential number of smaller patterns. Given a parameter set Π that discovers Z interesting patterns and another set Π' that discovers 2Z interesting patterns, we expect Π' to take an additional exponential factor of time to finish if both parameter sets Π and Π' are otherwise equally efficient. Therefore, the log of the execution times for different parameter sets (i.e. $\log(T_\Pi)$) should be

linearly related when consider-ing interesting (long) patterns. The efficiency metric for parameter set Π should give the amount of time required for each fraction of completeness achieved. Consequently, we define E(Π)X to be the ratio of run time to completeness, i.e. E(Π)X=log(TΠ)/PXΠ for pattern bin size X. The average (or weighted average) of $E(\Pi)_X$ for all bin sizes yields E(Π), the final measure of efficiency for Π. If E(Π)< E(Π') than an execution of Teiresias using Π finds more interesting patterns in a given time then an execution using Π'.

An advantage of this metric is that it requires no special knowledge of the data in question, nor does it rely on any special features of Teiresias. Thus, the above metric and methodology should be applicable to other forms of data and other pattern finding algorithms.

4. RESULTS, CONCLUSIONS AND FUTURE WORK

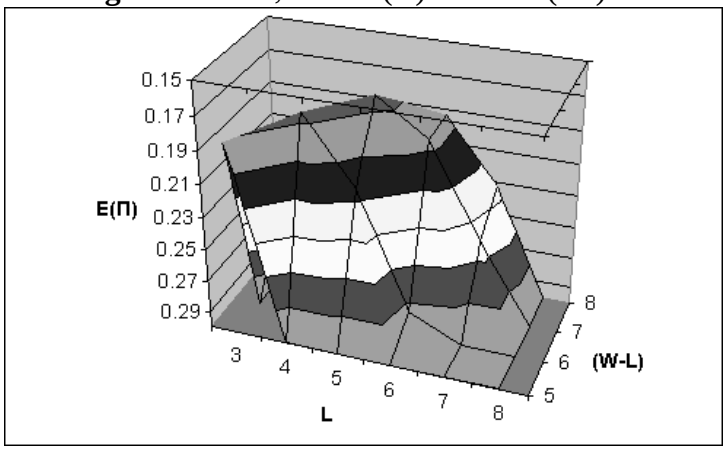
4.1 Parameters

The total number of aligned patterns, N_X, in the handpicked data set for each bin size X was found to be: $N_{>60}$ =6, $N_{>50}$=38, $N_{>40}$=231, $N_{>30}$=1628. In Table 1, for each (L,W) pair an execution time in seconds is given, as well as the number of interesting, aligned patterns found in non-break data for each bin size. From these data the efficiency of each parameter set, (L,W), can be calculated. In Figure 2, E(P) is plotted against L and (W-L). In this figure the point for L=5, (W-L)=7 corresponds to the most efficient parameter set Π* with $E(\Pi^*)_X$ = 0.1724. Further, Π* remains the most efficient parameter set even when considering a number of different weighted averages. Lastly, when this procedure was repeated on the randomly selected data, L=5, W=12 was still the most efficient parameter set. From these results we deduce that Π*=(L=5, W=12) provides the best trade-off between completeness of results and time spent finding the patterns for this type of data. The next step in this research is to test if Π* remains the most efficient parameter set for other kinds of gene expression microarray data (e.g. non-cell-cycle and non-yeast).

Table 1. Number of patterns found of each size for hand-picked, non-break data for a given (L,W) pair (where execution time < 1 day).

L	W	Time (s)	Aligned Patterns of given size			
			>60	>50	>40	>30
3	8	32607	6	35	146	769
4	10	8689	6	34	151	613
5	10	390.8	6	22	36	136
5	11	310.34	6	30	71	268
5	12	6554	6	36	147	575
6	12	146.49	6	24	43	173
6	13	331.38	6	30	88	290
6	14	19120	6	38	146	563
7	14	199.03	6	29	59	193
7	15	3465.1	6	37	108	327

Figure 2. W-L, L vs. E(Π) to find E(Π*).



4.2 Patterns

At the outset of this research it was clear that long patterns would be found in the hand-picked data set of related genes. However, what was not clear was if the randomly selected set of genes would have any significant patterns between them. Although not the focus of this paper, 2 patterns found in the randomly selected data set prove to be very interesting and are reported here. Of the 5 patterns of length 50 or more discovered, 4 were aligned and 1 was unaligned. In Figure 3, we see the input streams for YNL112w (DBP2) and YOR001w (RRP6) and the 50-literal, aligned pattern they form. In Figure 4, we see the input streams for YDL067c (COX9) and YOL004w (SIN3) and the 50-literal, unaligned pattern that they form (note: an extra character "X" has been added to the SIN3 stream for clarity). Although no direct link between RRP6 (a 3'-5' exoribonuclease) and DBP2 (an RNA helicase) could be found in the literature, RRP6 and other members of DBP family (DBP3,4,6,7,8,9,10) are well know to be involved in

ribosome synthesis in yeast [Venema, 99]. Further, DBP2 and RRP6 cluster together very tightly using multiple correlation techniques (centred and uncentred correlation and centered and uncentred absolute correlation) and the hierarchical clustering scheme found in [Eisen, 98]. Although confirmation of a link between DBP2 and RRP6 would have to be confirmed in a wet-lab, this evidence seems to support a relationship between the two genes.

The second pattern that was examined was found displaced in time between COX9 and SIN3. The length of this pattern implies some significant relationship has been discovered. Again, however, the relationship between the two genes can only be confirmed by a wet-lab. That said it is important to point out that this relationship would likely not be found by currently used correlation and hierarchical clustering methods of [Eisen, 98]. In fact, when these techniques are applied to this data, COX9 and SIN3 appear, unsurprisingly, very far apart in the resulting tree. This seems to confirm that these displaced patterns not only exist but are not readily found using the currently most common techniques. Further, it seems likely that any existing unaligned patterns in microarray data have gone unnoticed and thus available data sets represent a rich source of hereto untapped information and are ripe to be mined using this technique.

Figure 3

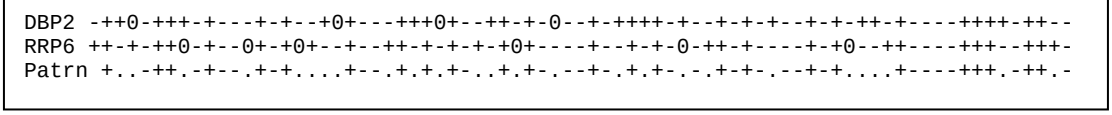
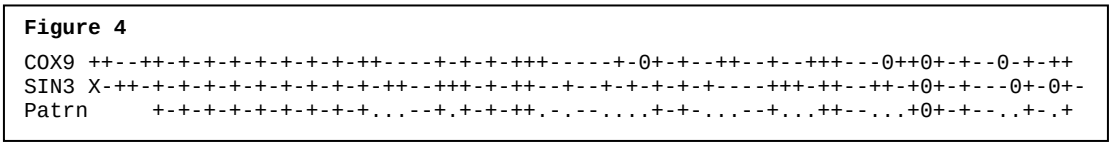


Figure 4



5. REFERENCES

Brazma A, et al., (1998). Approaches to the Automatic Discovery of Patterns in Biosequences. Journal of Computational Biology, 5(2), 279-305.

Califano, A. (1999). SPLASH: Structural Pattern Localization and Analysis by Sequential Histograms. Bioinformatics 16: 341-357.

Califano, A., et al. (2000). Analysis of gene expression microarrays for phenotype classification. ISMB'00.

Eisen M., et al. (1998). Cluster Analysis and Display of Genome-Wide Expression Patterns. Proc Natl Acad Sci USA 95, 14863-8.

Floratos, A. and I. Rigoutsos. (April 1998). On The Time Complexity Of The Teiresias Algorithm. IBM Research Report, RC 21161.

Rigoutsos, I., Floratos A., et al. (July 2000). The Emergence of Pattern Discovery Techniques in Computational Biology. Metabolic Engineering. 2(3):159-177.

Spellman PT, Sherlock G, Zhang MQ. (1998). Comprehensive Identification of Cell Cycle-Regulated Genes of the Yeast Saccharomyces Cerevisiae by Microarray Hybridization. Mol Biol Cell 9, 3273-97.

Venema, J., Tollervey, D. (1999) Ribosome Synthesis in Saccharomyces cerevisiae. Annu. Rev. Genet. 33:261-311

Yeast Cell Cycle Analysis Project. (April 11th, 2002). http://genome-www.stanford.edu/cellcycle/data/rawdata/combined.txt.