DNA sequence analysis : hypercomplex number representation, statistical properties of CIS-regulatory region

Li, Ya Jing

2008

Li, Y. J. (2008). DNA sequence analysis : hypercomplex number representation, statistical properties of CIS-regulatory region. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/20672>

<https://doi.org/10.32657/10356/20672>

DNA Sequence Analysis: Hypercomplex Number Representation, Statistical Properties of Cis-regulatory Region

Li Yajing

School of Mechanical and Aerospace Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2008

Acknowledgements

I would like to express my deepest gratitude and appreciation to my supervisor, Professor Shu Jian-Jun, for his guidance, encouragement and patient assistance, without whom this work may have not been completed.

I also would like to thank Miss Ouw Lishan, for her valuable suggestions.

Dedication

Dedication

To my parents, for all their sacrifices, to Taosheng, for supporting and pushing me along, and to my kids, for the courage they give me.

Table of Contents

Acknowledgements	i
Dedication	ii
Table of Contents	iii
Summary	viii
List of Figures	x
List of Tables	xv
Nomenclatures	xviii
Chapter 1 Introduction	
1.1 Introduction	1
1.1.1 DNA	2
1.1.2 Gene and protein	4
1.1.3 Regulation of gene expression	7
1.2 Cis-regulatory modules and transcription factor binding sites	10
1.2.1 Cis-regulatory modules	10
1.2.2 Transcription factor binding sites and motifs	13
1.3 Computational approach for identifying CRMs	17
1.3.1 Computational techniques based on known TFBS	17
1.3.2 Computational techniques based on homology	18
1.3.3 Computational techniques based on known CRMs	19
1.4 Objectives and outline of this thesis	20
Chapter 2 Pairwise alignment of DNA with hypercomplex number representation	
2.1 Introduction	21

Table of Contents

2.2 Hypercomplex number	23
2.3 Hypercomplex number representations of DNA nucleotide base code	24
2.4 Dot matrix with hypercomplex number representation of DNA	27
2.4.1 Effect of truncation value on the dot matrix analysis	30
2.4.2 Effect of window size and stringency on the dot matrix analysis	32
2.5 Scoring matrix for hypercomplex number representation	35
2.5.1 Global alignment using hypercomplex number representation	37
2.5.2 Local alignment using hypercomplex number representation	41
2.6 Discussion and conclusion	44
 Chapter 3 Cross-correlation of DNA with hypercomplex number representation	
3.1 Introduction	46
3.2 Hypercomplex number representation of DNA motif	48
3.3 Hypercomplex cross-correlation	51
3.3.1 Pairwise DNA sequence alignment with hypercomplex cross-correlation	51
3.3.2 Motif scanning with hypercomplex cross-correlation	54
3.3.3 Implementation the hypercomplex cross-correlation by hypercomplex Fourier transform	55
3.4 Results and discussion	56
3.4.1 Pairwise alignment of DNA with hypercomplex cross-correlation	56

Table of Contents

3.4.2 Motif scanning with hypercomplex cross-correlation	62
3.4.3 Time complexity for motif scanning with hypercomplex cross-correlation	69
3.5 Conclusion	70
 Chapter 4 Statistical properties of regulatory DNA sequences: fat-tail test	
4.1 Introduction	71
4.2 Materials and methods	73
4.2.1 Training data sets	73
4.2.2 Fat-tail test	74
4.2.3 Fat-tail kurtosis coefficient $F-K_r$ and $F-KK$	77
4.3 Results	83
4.3.1 Distribution of similar-word set occurrence for CRMs	83
4.3.2 Distribution of similar-word set occurrence for exons	89
4.3.3 Distribution of similar-word set occurrence for non-coding, presumed non-regulatory DNA (NCNRs)	94
4.3.4 Fat-tail test results for three training data sets	99
4.3.5 Comparison with fluffy-tail test	112
4.3.6 Time complexity	116
4.4 Discussion	119
4.4.1 Comparison between $F-K_r$ and $F-KK$	119
4.4.2 Codon usage bias and fat-tail kurtosis coefficient	119
4.4.3 Application to the REDfly test set	120

Table of Contents

4.4.4 Correlation between the distribution of similar-word set occurrence and experimentally verified TFBS	120
4.4.5 Tandem repeat region and fat-tail test	121
4.4.6 Remarks	126
4.5 Conclusion	128
 Chapter 5 Statistical properties of regulatory DNA sequences: thin-tail test	
5.1 Introduction	129
5.2 Materials and methods	130
5.2.1 Thin-tail test	130
5.2.2 <i>Z-score</i> – statistical evaluation of overrepresentation for similar-word set	134
5.2.3 Thin-tail kurtosis coefficient $F-K_r$ and $F-KK$	136
5.3 Results	138
5.3.1 Distribution of similar-word set <i>Z-score</i> for CRMs	138
5.3.2 Distribution of similar-word set <i>Z-score</i> for exons	144
5.3.3 Distribution of similar-word set <i>Z-score</i> for non-coding, presumed non-regulatory DNA (NCNRs)	149
5.3.4 Thin-tail test results for three training data sets	154
5.3.5 Comparison with fat-tail test	161
5.3.6 Comparison with fluffy-tail test	163
5.3.7 Time complexity	165
5.4 Discussion	168
5.4.1 Comparison between $T-K_r$ and $T-KK$	168

Table of Contents

5.4.2 Application to the REDfly test set	168
5.4.3 Remarks	169
5.5 Conclusion	169
 Chapter 6 Conclusions	
6.1 Hypercomplex number representation of DNA	170
6.2 Statistical properties of cis-regulatory region	170
 Publications	172
 References	173
 Appendix A Hypercomplex cross-correlation calculation	190
 Appendix B Results on fat-tail test	197
 Appendix C Results on thin-tail test	217

Summary

Automated analysis of genomic data is a crucial problem in post-sequencing genome era. In this thesis, we focus on exploring computational method to identify cis-regulatory regions automatically.

A hypercomplex number representation of DNA is proposed to facilitate comparing DNA sequences with fuzzy composition. With the hypercomplex number representation, the conventional sequence analysis methods, such as, dot matrix analysis, dynamic programming, and cross-correlation method have been extended and improved to align DNA sequences with fuzzy composition. Hypercomplex dot matrix analysis can provide more control over the degree of alignment desired. A new scoring model has been proposed to accommodate the hypercomplex number representation of DNA and integrated with dynamic programming alignment method. By using hypercomplex cross-correlation, the match and mismatch information between two aligned DNA sequences are separately stored in resultant real part and imaginary parts respectively. The mismatch information is very useful to refine consensus sequence based motif scanning.

Summary

Statistical properties of cis-regulatory modules (CRMs) are explored by estimating the distribution of similar-word set occurrence. It is observed that cis-regulatory modules tend to have fat-tail distribution for similar-word set occurrence. Thus, fat-tail test with two kurtosis coefficients $F-K_r$ and $F-KK$ has been proposed to distinguish CRMs from non-CRMs. The performance of the proposed method on the experimentally verified *Drosophila melanogaster* regulatory regions shows that the method allows us to predict whether a sequence may be regulatory in a relatively sensitive and efficient way.

List of Figures

Figure 1.1	The DNA double helix	3
Figure 1.2	From gene to protein	6
Figure 1.3	Transcriptional regulation	9
Figure 1.4	Schematic of regulation of <i>Drosophila melanogaster</i> pair-rule gene eve-skipped (eve) 2	12
Figure 2.1	Dot matrix of example sequences with truncation values of (a) 0.3 and (b) 0.5	31
Figure 2.2	Dot matrix of example sequences with a sliding window size of 3 and stringency of 1	33
Figure 2.3	Dot matrix of example sequences with a sliding window size of 5 and stringency of 3	34
Figure 2.4	Derivation option for the $F(i, j)$ value	39
Figure 2.5	Global dynamic programming matrix for hypercomplex number representation of DNA sequences	40
Figure 2.6	Local dynamic programming matrix for hypercomplex number representation of DNA sequences	43
Figure 3.1	Real part of hypercomplex correlation of (a) HIVMN against HIVRF and (b) HIVMN against SIVMM142	57
Figure 3.2	Real part of hypercomplex cross-correlation of human TGFA sequence against mouse TGFA sequence	59

List of figures

Figure 3.3	Real part of hypercomplex cross-correlation of human TGFA sequence with 1 bp deletion in the center position against mouse TGFA sequence	60
Figure 3.4	Real part of hypercomplex cross-correlation of (a) human TGFA sequence with 70 bp deletion in the center position against mouse TGFA sequence	61
Figure 3.5	Real part of hypercomplex cross-correlation of H4/g gene against consensus sequence 'TATAWAWR'. (Hypercomplex representation of consensus sequence is vector A in Table 3.1)	65
Figure 3.6	Partial sum plots for H.sapiens H4/g gene compared with consensus sequence 'TATAWAWR', showing the match and mismatch information from (a) position 185 and (b) position 381	66
Figure 3.7	Real part of hypercomplex cross-correlation of H. sapiens H4/g gene against frequency based consensus sequence. (Hypercomplex number representation is vector B in Table 3.1)	67
Figure 3.8	Real part of hypercomplex cross-correlation of H.sapiens H4/g gene against hypercomplex position weight vector (HPWV) C shown in Table 3.1.	68
Figure 4.1	Flow chart of fat-tail test	81
Figure 4.2	95% confidence interval for the kurtosis value of normal distribution with sample size of 649	82
Figure 4.3	Histogram of similar-word set occurrence O for the hb cis-regulatory module	85

List of figures

Figure 4.4	Histogram of similar-word set occurrence O for the hb cis-regulatory module, after shuffling	86
Figure 4.5	Cumulative histograms of similar-word set Occurrence O for the hb cis-regulatory module and its shuffled version	87
Figure 4.6	Histograms of similar-word set Occurrence O for the hb cis-regulatory module and its shuffled versions	88
Figure 4.7	Histogram of similar-word set Occurrence O for the exon CG8229	90
Figure 4.8	Histogram of similar-word set Occurrence O for the exon CG8229, after shuffling	91
Figure 4.9	Cumulative histograms of similar-word set Occurrence O for the exon CG8229 and its shuffled version	92
Figure 4.10	Histograms of similar-word set Occurrence O for the exon CG8229 and its shuffled versions	93
Figure 4.11	Histogram of similar-word set Occurrence O for the example NCNR sequence that randomly selected from <i>Drosophila melanogaster</i> genome	95
Figure 4.12	Histogram of similar-word set Occurrence O for the example NCNR sequence, after shuffling	96
Figure 4.13	Cumulative histograms of similar-word set Occurrence O for the example NCNR sequence and its shuffled version	97
Figure 4.14	Histograms of similar-word set Occurrence O for the example NCNR sequence and its shuffled versions	98

List of figures

Figure 4.15	Histograms for regulatory, coding sequences classified by $F-K_{50}$	103
Figure 4.16	Histograms for regulatory, NCNR sequences classified by $F-K_{50}$	104
Figure 4.17	Box-plot of the fat-tail kurtosis coefficient $F-K_{50}$ (Y-axis) for the three functional regions	105
Figure 4.18	Histograms for regulatory, coding sequences classified by $F-KK$	106
Figure 4.19	Histograms for regulatory, NCNR sequences classified by $F-KK$	107
Figure 4.20	Box-plot of the fat-tail kurtosis coefficient $F-KK$ (Y-axis) for the three functional regions	108
Figure 4.21	Histograms for regulatory, coding sequences classified by $F-M_{50}$	109
Figure 4.22	Histograms for regulatory, NCNR sequences classified by $F-M_{50}$	110
Figure 4.23	Box-plot of the fat-tail kurtosis coefficient $F-M_{50}$ (Y-axis) for the three functional regions	111
Figure 5.1	Flow chart of thin-tail test	133
Figure 5.2	Histogram of similar-word set Z -score for the hb cis-regulatory module	140
Figure 5.3	Histogram of similar-word set Z -score for the hb cis-regulatory module, after shuffling	141
Figure 5.4	Cumulative histograms of similar-word set Z -score for the hb cis-regulatory module and its shuffled version	142
Figure 5.5	Histograms of similar-word set Z -score for the hb cis-regulatory module and its shuffled versions	143
Figure 5.6	Histogram of similar-word set Z -score for the exon CG12653	145
Figure 5.7	Histogram of similar-word set Z -score for the exon CG12653, after shuffling	146
Figure 5.8	Cumulative histograms of similar-word set Z -score for the exon CG12653 and its shuffled version	147

List of figures

Figure 5.9	Histograms of similar-word set <i>Z-score</i> for the exon CG12653 and its shuffled versions	148
Figure 5.10	Histogram of similar-word set <i>Z-score</i> for an example NCNR sequence that randomly selected from <i>Drosophila melanogaster</i>	150
Figure 5.11	Histogram of similar-word set <i>Z-score</i> for the example NCNR sequence, after shuffling	151
Figure 5.12	Cumulative histograms of similar-word set <i>Z-score</i> for the example NCNR sequence and its shuffled version	152
Figure 5.13	Histograms of similar-word set <i>Z-score</i> for the example NCNR sequence and its shuffled versions	153
Figure 5.14	Histograms for regulatory, coding sequences classified by $T-K_{50}$	157
Figure 5.15	Histograms for regulatory, NCNR sequences classified by $T-K_{50}$	158
Figure 5.16	Histograms for regulatory, coding sequences classified by $T-KK$	159
Figure 5.17	Histograms for regulatory and NCNR sequences classified by $T-KK$	160

List of Tables

Table 1.1	The IUPAC (international union of pure and applied chemistry) code for representing degenerate nucleotide consensus sequence	15
Table 1.2	TATA-box base frequency and position weight matrix for Eukaryotic RNA Polymerase II promoters	16
Table 2.1	The IUPAC codes and their hypercomplex number representations	26
Table 2.2	Value of dot product of hypercomplex number representation per aligned base pair of example sequences	29
Table 2.3	New scoring matrix derived from dot product of hypercomplex number representation of DNA bases	36
Table 3.1	Base frequency, consensus sequence, position weight matrix of TATA-box for Eukaryotic RNA Polymerase II promoters (Bucher, 1990), and their hypercomplex number representation.	50
Table 3.2	Hypercomplex product results between hypercomplex represented DNA base codes	53
Table 4.1	60 cis-regulatory modules of <i>Drosophila melanogaster</i>	80
Table 4.2	Classification of the 180 sequences by fat-tail test with kurtosis coefficient (a) $F-K_{50}$, (b) $F-KK$ and (c) $F-M_{50}$	102
Table 4.3	Evaluation of fat-tail kurtosis coefficient $F-K_{50}$, fat-tail kurtosis coefficient $F-KK$, and fluffiness coefficient F_{50} on distinguishing CRMs from exons	114

List of tables

Table 4.4	Evaluation of fat-tail kurtosis coefficient $F-K_{50}$, fat-tail kurtosis coefficient $F-KK$, and fluffy-tail coefficient F_{50} , on distinguishing CRMs from NCNRs	115
Table 4.5	Sensitivity of $F-K_r$ to choice of r , the number of randomization, for the hb cis-regulatory region. (the value of kurtosis statistic of original sequence $k_{\text{original}} = 2.70$)	117
Table 4.6	Run-time for calculating fat-tail test coefficient $F-K_{50}$, $F-KK$, and F_{50} .	118
Table 4.7	Classification of the 180 sequences by fat-tail test with coefficient $F-K_{50}$, $F-KK$, and F_{50} before repeat-masking.	123
Table 4.8	Evaluation of fat-tail kurtosis coefficient $F-K_{50}$, fat-tail kurtosis coefficient $F-KK$, and fluffiness coefficient F_{50} on distinguishing CRMs from exons, before and after repeat-masking.	124
Table 4.9	Evaluation of fat-tail kurtosis coefficient $F-K_{50}$, fat-tail kurtosis coefficient $F-KK$, and fluffy-tail coefficient F_{50} , on distinguishing CRMs from NCNRs, before and after repeat-masking	125
Table 5.1	Classification of the 180 sequences by thin-tail test with kurtosis coefficient $T-K_{50}$ and $T-KK$	156
Table 5.2	Comparison between thin-tail test and fat-tail test on distinguishing CRMs from NCNRs	162
Table 5.3	Comparison between thin-tail test and fluffy-tail test on distinguishing CRMs from NCNRs	164

List of tables

Table 5.4	Sensitivity of $T-K_r$ to choice of r , the number of randomization, for the hb regulatory sequence	166
Table 5.5	Computing time for calculating thin-tail test kurtosis coefficient T - K_r , $T-KK$ and fat-tail test kurtosis coefficient $F-K_r$, $F-KK$	167

Nomenclatures

σ_r	Standard deviation of the kurtosis of r shuffled sequences
σ_o	Standard deviation of the similar-word set occurrence O for a given sequence
$\sigma_{\max,r}$	Standard deviation of the number of occurrences of the similar-word set with the highest incidence for r shuffled sequences.
σ_Z	Standard deviation of the similar-word set Z -score Z for a given sequence
3D	Three-dimension
$\mathbf{B}_j(\mathbf{H})$	j -neighborhood of a word $\mathbf{H}$
CRMs	Cis-regulatory modules
95% C.I.	95% confidence interval for the kurtosis of normal distribution
$E, E(O(\mathbf{K}_S))$	Expected number of occurrences of a set of words in a random sequence
F_r	Fluffy tail coefficient calculated by shuffling original sequence r times
$F\text{-}K_r$	Fat-tail kurtosis coefficient calculated by shuffling original sequence r times
$F\text{-}K_{50}$	Fat-tail kurtosis coefficient corresponding to $r = 50$
$F\text{-}KK$	Fat-tail kurtosis coefficient
$F\text{-}M_{50}$	Fat-tail coefficient corresponding to $r = 50$
HPFV	Hypercomplex position frequency vector
HPWV	Hypercomplex position weight vector
j	Base number of mismatch
$\mathbf{K}_S$	Similar-word set corresponding to seed word S
k	Kurtosis of data

Nomenclatures

k_{original}	Kurtosis of similar-word set distribution for original sequence
$\bar{k}_r$	Mean of the kurtosis of r shuffled sequences
M	Total Number of seed words
m	Length of seed word
N	Length of DNA sequence
NCNRs	Non-coding, presumed non-regulatory regions
$O(H)$	Number of occurrences of a word H
$O, O(K_S)$	Number of occurrences of a set of word K_S
O_i	Number of occurrences of the i th similar-word set
$\bar{O}$	Mean of the similar-word set occurrence O
$O_{\text{max, original}}$	Number of occurrences of the similar-word set with highest incidence for original sequence
$\bar{O}_{\text{max}, r}$	Mean of the number of occurrences of the similar-word set with the highest incidence for r shuffled sequences.
P	Length of motif
PFM	Position frequency matrix
PWM	Position weight matrix
r	Number of randomization
S	Seed word
sek	Standard error of kurtosis
SN	Sensitivity
SP	Specificity

Nomenclatures

TF	Transcription factor
TFBS	Transcription factor binding site
$V, V(O(K_S))$	Variance of the number of occurrences of a set of words in a random sequence
Z	Z score for a set of words

Chapter 1

Introduction

1.1 Introduction

The great progress of biotechnology makes the deoxyribonucleic acid (DNA) sequencing more efficiently. With the first draft of the human genome, a sequence of nearly 3 billion nucleotides, has already been put together (Venter, et al., 2001), huge amount of DNA sequences of various organisms have been successfully sequenced with higher accuracy (Adams, et al., 2000; Myers, et al., 2000). These achievements have brought to the fore an entirely different and more imposing challenge – that of identifying significant features in the sequence, and interpreting their biological roles in the information processing machinery of the cell. While the richness and complexity of knowledge in molecular biology makes humans best equipped to meet this challenge, the sheer magnitude of data that is available makes a compelling case for much of the expertise to be transferred to computational method that can automate the data processing. This thesis addresses a specific problem in this emergent area of automated analysis of genomic data.

Before we can state the addressed problem clearly, we need to familiarize ourselves with fundamental facts and ideas of molecular biology. An excellent exposition of the subject can be found in (Lewin, 2000).

1.1.1 DNA

DNA is a double-helical molecule that carries all genetic information of a cell. Each strand of the DNA double-helix is a sequence of four distinct types of units, called “nucleotides”, jointed together by sugar-phosphate linkages between consecutive nucleotides. The four distinct nucleotides, or “bases”, are adenine, cytosine, guanine, and thymine, and are popularly denoted by their initial letters (A, C, G, T) respectively. The bases have a strong tendency to form hydrogen bonds with other bases in a specific way: G pairs with C, and A pairs with T. G and C are said to be “complementary” to each other, and so are A and T. Each base in one strand of the DNA molecule is paired with its complementary base in the other strand. Such pairs of hydrogen bonded bases are often called “base pairs”, and the DNA can thus be looked upon as a sequence of base pairs (Figure 1.1). DNA sequences are often represented by the sequence of bases in one of the strands, since this completely determines the base sequence in the other strand. There is chemical asymmetry between the two ends of any strand, thereby providing directionality to the sequence. The sequence is written from the so-called 5' end of the strand to the 3' end. In its natural state, DNA exists in a highly coiled structure called the “chromatin”, but during most of its important functions, such as replication or transcription, it is processed locally as linear sequence. The fact that for certain purposes we can treat DNA as a linear sequence make it especially amenable to analysis by computational algorithms, since it is now simply a string over the alphabet $\Sigma = \{A, T, G, C\}$.

Figure 1.1 The DNA double helix. A group of sugar, phosphate and the attached bases is called a DNA nucleotide. G and C bind each other; A and T bind each other.

(<http://www.accessexcellence.org/RC/VL/GG/dna2.html>)

1.1.2 Genes and proteins

As mentioned above, DNA is the carrier of genetic information. However, all of its sequence need not carry such information. The important information is stored in several loci that are interspersed throughout the sequence. The most important kind of information is stored in small segments called genes. A gene is a relatively short (hundreds to several thousand bp) fragment of DNA that serves as the template for generating other kinds of important molecules. Gene controls a hereditary characteristic of organism. Each gene encodes the information needed to manufacture a specific molecule, usually proteins, and sometimes other molecules like rRNA and tRNA.

A protein is a polymer consisting of many linked, smaller components called amino acids. There are 20 possible types of amino acids in protein and, just as the case in DNA single strand; they are connected with strong bonds, one after the other, forming a long one-dimensional chain (backbone) of a specific directionality. Therefore, as in DNA, a character string mathematically represents each protein. Protein molecule, also called a “polypeptide”, tends to fold into complex three-dimensional (3-D) structure that is necessary for its function. Thousands of types of proteins are present in the different cells of an organism, each protein with a distinct amino-acid sequence and 3-D structure. They are responsible for carrying out nearly all of the essential functions in the living cell.

From genes to protein

The mechanism whereby a specific protein is synthesized from its genes is fairly well understood, at least at a high level. Figure 1.2 illustrates this process, which takes place in multiple stages. In many genes, the DNA sequences coding for proteins, or “exons”, may be interrupted by stretches of non-coding DNA, called “introns”. In the first step of synthesis of protein, called “transcription”, the DNA that includes all the exons and introns of gene is first transcribed into complementary RNA copy called “nuclear RNA” (or nRNA), then, introns are subtracted from nRNA to construct a molecule called mRNA. mRNA is an RNA (ribonucleic acid) molecule, which, like DNA, is a polymer of nucleotide, except that it is single stranded, uses ribose instead of deoxyribose as the sugar in the backbone, and uses the nucleotide uracil (U) instead of thymine. The mRNA is the messenger that carries information from the DNA, where the genes reside, to a structure called the ribosome, where the protein is produced.

The ribosome is the site for the next major step of protein synthesis, called “translation”. Here the mRNA sequence is processed linearly, three nucleotides at a time, to produce a chain of amino acids. Each successive triplet of nucleotides in mRNA, called a codon, is translated into a specific amino acid, according to the universal genetic code. More than one codon may translate to the same amino acid. The translation of a codon into specific amino acid is mediated by molecule called tRNA, which is capable of recognizing a specific codon as well as recruiting a specific amino acid. The chain of amino acids formed as a result of translation is a polypeptide that folds to a specific 3-D structure giving the protein its final form.

Figure 1.2 From gene to protein: the process by which a protein is formed from its gene template. Exon of gene is transcribed to form mRNA. The mRNA then transfer to the ribosome, where translation occurs. A polypeptide chain is formed by translating of mRNA, using amino acids as raw materials and tRNA as mediators. (<http://www.accessexcellence.org/RC/VL/GG/gene2.html>)

1.1.3 Regulation of gene expression

The synthesis of a protein from its gene template is called gene expression. Not all genes need to be expressed at all times or in all cells. Usually, gene expression needs to be regulated both spatially and temporally. Regulation of gene expression has multistage: transcriptional regulation, RNA processing regulation, RNA transport regulation, translation regulation, and protein activity regulation. In this thesis, we focus on transcriptional regulation.

Transcriptional regulation

For the mRNA transcript to be produced from gene template, the RNA polymerase complex first need to be recruited near the 5' end of the gene, at a position called the transcription start site (TSS) (Figure 1.3). In some cases, a protein molecule (transcription factor) binds to the DNA near the TSS, and then interacts with the RNA polymerase complex; either inducing or inhibiting the latter's DNA-binding capacity, thus to either promote or suppress gene expression. The sequence elements within the DNA template that act along with this transcription factor to stimulate the initiation of transcription are termed promoters. Promoter sequences, which are located close to and upstream of the gene (around 500bp from TSS), promote the ability of RNA polymerase to recognize the nucleotide at which initiation begins. Addition sequence elements are present within genes that act in *cis* by binding with transcription factor to enhance polymerase activity even further. These elements, which may be located far upstream or downstream of the target gene, are termed enhancer or *cis*-regulatory

Chapter 1 Introduction

modules (CRMs) (Figure 1.3). They often bind with enhancer-binding protein to dictate the rate of transcription.

The regulated gene itself may code another transcription factor, which in turn regulates another gene. In general, there may be a network of such regulatory interactions involving genes and their products. This is often called a “regulatory network”. A detailed study of this topic is presented in (Davidson, 2001).

The sequencing of complete genomes from multiple organisms has revealed that most differences in organism complexity are due to elements of gene regulation that reside in inter-genetic regions. Both within and between species, transcription factor binding sites (TFBS) and the proteins that recognize them govern the activity of cellular pathways that mediate adaptive responses and survival (Schroeder, et al., 2004).

Thus identification of regulatory region is crucial step in the study of gene regulation and variety of organism. Promoters are located immediately upstream of a gene's TSS and often contain a variety of sequence signals such as TATA box, CCAAT box, and different TFBS. These characteristics have been used in approaches for finding promoter (Fickett and Hatzigeorgiou, 1997; Pedersen, et al., 1999; Qiu, 2003). In contrast, enhancers do not share these signals and operate in a manner that is relatively independent of orientation or distance from their target gene. Because of the lack of common signals and because the search for enhancer cannot be limited to the few hundred base pairs upstream of TSS, finding enhancer is a more difficult problem.

Figure 1.3 Transcriptional regulation: A transcription factor binds upstream of a gene and activates the RNA polymerase, thereby promoting transcription initiation. Regulatory regions of higher eukaryotes can be subdivided into proximal regulatory units – promoters- which are located close to and upstream of the gene (around 500 bp from TSS), and distal transcription regulatory units called enhancers or cis-regulatory modules (CRMs). CRMs may be located far upstream or downstream of the target gene (Lewin, 2000).

1.2 Cis-regulatory modules, transcription factor binding sites

For the rest of this thesis, we shall focus on the problem of identification of cis-regulatory modules (CRMs).

1.2.1 Cis-regulatory modules

There is not a commonly accepted definition of a cis-regulatory module, owing to their inherent variability in structure. We give a working definition of cis-regulatory module as follows:

- A “module” has a cluster of *binding sites* that mediate the action of several transcription factors, to control a target gene’s expression. Typically, 3-6 transcription factors are involved in regulating a module.
- Modules are typically 200-2000 bp long.
- One or many occurrences of binding sites for each of transcription factors.
- Multiple copies of binding sites of the same type is called *homotypic* binding site cluster; multiple copies of binding sites of different types is called *heterotypic* binding site cluster.

Careful genetic and biochemical dissection of numerous genes involved in *Drosophila* development suggests some general principles for how cis-regulatory regions are organized. For example, the cis-regulatory region of the pair-rule gene *even-skipped* (*eve*) is organized into a series of discrete sequence regions of roughly 500 bp in length, each of which controls a distinct component of *eve*’s expression pattern (Goto, et al., 1989; Sackerson, et al., 1999; Small, et al., 1996; Stanojevic, et al., 1991). This

Chapter 1 Introduction

modular organization of cis-regulatory regions is observed in many developmental genes in *Drosophila*, and in other organisms (Davidson, 2001). In general, several transcription factors bind to each of these cis-regulatory modules (CRMs), and there are often multiple binding sites for each of these factors. Presumably, multiple bound transcription factors act combinatorially to confer specific transcriptional activity. For example, Figure 1.4 shows the regulation of *Drosophila melanogaster* pair-rule gene *eve*-skipped (*eve*) 2 (Arnosti, et al., 1996; Small, et al., 1996). Expression patterns of the transcription factors important for stripe 2 regulation (Bicoid (Bcd), Hunchback (Hb), Giant (Gt), and Kruppel (Kr)) are shown schematically in Figure 1.4 (A). Stripe 2 expression is activated by Bicoid and Hunchback in a broad anterior domain. This activation is repressed by Giant protein anteriorly and Kruppel protein posteriorly to set the borders of the stripe (shown in Figure 1.4 (B)). All four genetically defined regulators (transcription factors) of stripe 2 expression bind in vitro to multiple sites in a 480 bp enhancer that is necessary and sufficient for directing expression at the position of *eve* stripe 2. The relative positions of binding sites for Bcd (B1-B5), Hb (H), Kr (K) and Gt (G1-G3) are shown in Figure 1.4(C).

Figure 1.4 Schematic of regulation of *Drosophila melanogaster* pair-rule gene eve-skipped (*eve*) 2, an example of heterotypic binding site cluster. (A) Expression patterns of the regulatory factors important for *eve* stripe 2 regulations. (B) Stripe 2 expression is activated by Bcd and Hb in a broad anterior domain. This activation is repressed by Gt protein anteriorly and Kr protein posteriorly to set the borders of the stripe. (C) The relative positions of binding sites for Bcd (B1-B5), Hb (H), Kr (K) and Gt (G1-G3) in the enhancer region.

(http://www.nyu.edu/fas/biology/faculty/small/smallfig7_big.html).

1.2.2 Transcription factor binding sites and motifs

A transcription factor is specialized to recognize very specific target sites in the DNA. These transcription factor binding sites (TFBS) are typically very short, between 6 and 25 bp long (Wingender, 1993). Usually the bases at all positions in the site are not equally important for binding specificity. Some positions can be occupied by any base without affecting the binding, while in other positions the presence of a particular base may be crucial for binding to occur. There may also be positions of intermediate variability where, for instance, one purine (A or G) is as good as the other, and pyrimidines (C or T) decrease binding specificity. Several biological studies have been done on specific transcription factors to determine the effect of variation in binding sites on the quality of binding. Point mutations of the binding site may be performed to measure this effect. Examination of several experimentally determined binding sites of the same transcription factor, possibly collected from different studies, can also reveal common pattern of variability. A “motif” is a characterization of the TFBS. The transcription factor has high affinity for sequences that exactly or approximately match this characterization, and relatively low affinity for sequences different from the motif.

Recurrent motifs in a collection of DNA sites are most commonly modeled by consensus sequence (also called sequence pattern or regular expression), or by position weight matrices (PWMs, also called profiles and position-specific scoring matrices, PSSMs). The consensus sequences are simply strings over 4-letter alphabets (A, T, G, C) that from the DNA. In order to capture variation in a specific

Chapter 1 Introduction

position, the degenerate IUPAC nucleic acid codes are used (Cornish-Bowden, 1985) (nomenclature for incompletely specified bases in nucleic acid sequence) (see Table 1.1). An L -long sequence motif can also be represented by a $4 \times L$ matrix with weights giving the frequency of the four DNA bases (or the logarithm) in each of L positions (see Table 1.2). The consensus and PWM representations serve as complementary approaches and have been used widely since the 1980s. The DNA consensus sequences are simpler in representation, and advantageous in terms of exact enumeration of their significance in the genome using statistical methods. The PWMs are able to capture information on the variability of a collection of DNA sites in a quantitative manner, which is not possible with the DNA consensus sequences. However, when detecting significant PWMs in DNA sequences, heuristic methods have to be used instead of exhaustive enumeration. Perhaps, the most important practical utility of these models is their application in scanning DNA sequences for new TFBS candidates.

Currently, there are two comprehensive and curated databases containing information on TFBS profiles. JASPAR (Sandelin, et al., 2004) contains a smaller set that is non-redundant, while TRANSFAC (Matys, et al., 2003) contains multiple profile models for some transcription factors.

Chapter 1 Introduction

Table 1.1 The IUPAC (international union of pure and applied chemistry) code for representing degenerate nucleotide consensus sequence (Cornish-Bowden, 1985).

Code	Meaning	Explanation
W	A or T	Weak interaction (2 H bonds)
R	A or G	PuRine
M	A or C	aMino
K	G or T	Keto
Y	C or T	pYrimidine
S	C or G	Strong interaction (3 H bonds)
D	A, G or T not C	D follows C in alphabet
H	A, C or T not G	H follows G in alphabet
V	A, C or G not T(or U)	V follows U in alphabet
B	C, G or T not A	B follows A in alphabet
N	Any base	Any base

Table 1.2 TATA-box base frequency and position weight matrix for Eukaryotic RNA Polymerase II promoters (Bucher, 1990).

Position	1	2	3	4	5	6	7	8
% A	4.1	90.5	0.8	91.0	68.9	92.5	57.1	39.8
% T	79.5	9.0	96.1	7.7	31.1	1.6	31.1	8.5
% G	4.6	0.5	0.5	1.3	0.0	5.1	11.3	40.4
% C	11.8	0.0	2.6	0.0	0.0	0.8	0.5	11.3
Consensus	T	A	T	A	W	A	W	R
W(A)	-3.05	0	-4.61	0	0	0	0	-0.01
W(T)	0	-2.28	0	-2.34	-0.52	-3.65	-0.37	-1.4
W(G)	-2.74	-4.28	-4.61	-3.77	-4.73	-2.65	-1.5	0
W(C)	-2.06	-5.22	-3.49	-5.17	-4.63	-4.12	-3.74	-1.13

1.3. Computational approaches for identifying CRMs

Ultimately, identification of regulatory region is accomplished experimentally using techniques such as empirical deletion analysis, direct binding measurements, and co-precipitation of protein-DNA complexes. However, experimental verification is expensive and time consuming. Therefore, to address the growing volumes of available genomic sequences, a number of computational algorithms that identify putative cis-regulatory modules and transcription factor binding sites using evolutionary comparisons, whole-genomic data, and known descriptions of transcription factor binding sites, has been successfully developed.

Computational techniques for predicting CRMs have been showing great promise. By the type of information they use to make the predictions, computational methods for predicting CRMs can be classified into three types: (1) based on known binding sites of regulatory proteins, (2) based on homologous sequences and (3) based on known CRMs (content based method).

1.3.1 Computational techniques based on known TFBS

This approach exploits the clustering of binding sites of known, cooperatively-acting transcription factors. Examples of methods based on binding sites of multiple transcription factors include one developed for human skeletal muscle (Wasserman and Fickett, 1998), a logistic regression analysis model for liver-specific transcription factor (Krivan and Wasserman, 2001), CIS-ANALYST (Berman, et al., 2002),

Chapter 1 Introduction

MCAST (Bailey and Noble, 2003), Ahab (Rajewsky, et al., 2002), Stubb (Sinha, et al., 2003), Cluster Buster (Frith, et al., 2003), MSCAN (Johansson, et al., 2003), and EMCMODULE (Gupta and Liu, 2005). Methods based on binding sites of single transcription factors have also been developed – SCORE (Rebeiz, et al., 2002), Fly Enhancer (Markstein, et al., 2002), and a method of searching for homotypic clusters (Lifanov, et al., 2003).

Extracting clustered recognition motifs is one of the most reliable techniques for CRMs prediction, but is limited to the recognition of similarly regulated cis-regulatory regions (Vavouri and Elgar, 2005). They cannot generally be applied to genes for which the transcription factors have not yet been studied experimentally. Thus, they cannot easily be used for a genome-wide CRM annotation since only a minor fraction of all active transcription factors and their binding sites are known.

1.3.2 Computational techniques based on homology

Methods based on homologous sequences assume that regulatory regions are under selective pressure and are therefore highly conserved in cross-genomic comparison than non function regions. These methods can be categorized by whether they search for conserved DNA by aligning homologous regions from multiple species (Blanchette and Tompa, 2002; Boffelli, et al., 2003; Sumiyama, et al., 2001), or homologous regions from two species (Grad, et al., 2004; Lenhard, et al., 2003; Loots, et al., 2002; Yuh, et al., 2002), or homologous regions from related genes in single species (Hertz and Stormo, 1999; Hughes, et al., 2000; van Helden, et al., 1998).

Performance of this method depends on the evolutionary distance between given species and on the conservational level of individual genes. This is an actively progressing area, as more and more sequenced genomes appear. However, such an approach offers little information as to the specific function of the conserved sequences. Furthermore, it is still an open question as to how many genomes are sufficient for reliable extraction of regulatory regions.

1.3.3 Computational techniques based on known CRMs

Different genome regions have different rates of evolutionary micro changes. Methods based on locations of known CRMs assume that different genome regions (introns, exon, CRMs, non-coding non-regulatory region) have different statistical properties of nucleotide composition. It is well known that one of distinct properties of CRMs is the presence of multiple copies of binding sites of the same types (binding site clusters), and that cause a biased word distribution within CRMs. This bias leaves a distinct “signature” in nucleotide composition, and methods of this type tend to detect this signature by statistical or machine-learning technique (Abnizova, et al., 2005; Chan and Kibler, 2005; Nazina and Papatsenko, 2003), thus distinguish the CRMs from non-CRMs. Moreover, such signature may be used to predict the CRMs which have not yet been experimental observed. This can serve as a valuable filtering process for experiment. References show that method based on known CRMs have been rare, and the algorithms are far from satisfactory, although some are promising.

1.4 Objectives and outline of this thesis

Computational identification of CRMs is a wide area with challenging difficulties. The objective of my research project is two fold: one is providing methods for comparing DNA sequences with fuzzy composition; another is the analysis of statistical properties that can be used to distinguish CRMs from non-CRMs. The project is focus on four main topics.

1. Pairwise alignment of DNA sequences with hypercomplex number representation.
2. Cross-correlation of DNA sequences with hypercomplex number representation.
3. Analyzing statistical properties of CRMs by estimating distribution of similar-word set.
4. Developing index to distinguish CRMs from exons and non-coding non-regulatory regions (NCNRs).

CHAPTER 2

Pairwise alignment of DNA with hypercomplex number representation

2.1 Introduction

DNA molecules are composed of four linearly linked nucleotides, adenine (A), thymine (T), guanine (G) and cytosine (C). A DNA sequence can be represented as a permutation of four characters A, T, G, and C of different lengths.

With more DNA sequences becoming available, computer programs have been developed to analyze these sequences in various ways. Dot matrix method, which is used to detect similarities between sequences, was discovered first (Mount, 2001). To compare two sequences with dot matrix, a graph is drawn with one sequence written across a page from left to right and another sequence down the page on the left-hand side. A dot is placed where the corresponding nucleotide in the two sequences is the same. The graph is then scanned for diagonals of dots, which reveal similarities. Unless the sequences are known to be very much alike, the dot matrix method is used first as this method displays any possible sequence alignments as diagonals on the matrix. The dot matrix analysis reveals the presence of insertions/deletions, and direct/inverted repeats that are rather difficult to find by other methods. The major limitation of the dot matrix analysis is that most dot matrix computer programs do not show an actual alignment.

Chapter 2 Pairwise alignment with hypercomplex representation

As the dot matrix method can not identify similarities that are interrupted, the method of sequence alignment was devised (Durbin, 1998). Sequence alignment is a procedure of comparing two sequences by searching for a series of individual characters that are in the same order. An alignment is generated, starting at the ends of the two sequences, by attempting to match all possible pairs of characters between the sequences, following a certain algorithm for matches, mismatches and gaps. This procedure generates a matrix of numbers that represent possible alignments between the sequences. The optimal alignment between the two sequences is one that gives the highest score. The dynamic programming method is guaranteed in a mathematical sense to provide the optimal alignment for a given set of user-defined variables, including the choice of scoring matrix and gap penalties. There are two types of sequence alignment: global alignment (Needleman and Wunsch, 1970) and local alignment (Smith and Waterman, 1981). In global alignment, the entire sequences are aligned from beginning to end. It is better to use global alignment for aligning sequences that are similar and have approximately the same length. In local alignment, parts of the sequences with the most matches are aligned, giving rise to a number of sub alignments in the aligned sequences. Thus local alignments are more suitable for aligning sequences that are similar only along some of their lengths, sequences that differ in length and sequences that share a conserved region or domain. These two methods of sequence alignment are sometimes used hand in hand, for more efficient sequence alignment of DNA.

Dot matrix analysis and dynamic programming reviewed above are used to compare DNA sequences with definite identity in each position. With the introduction

Chapter 2 Pairwise alignment with hypercomplex representation

of methods of rapid nucleic acid sequence determination (Alphey, L. 1997), many low quality DNA sequences, which have fuzzy composition in some positions, are needed to be analyzed. Usually, a single symbol (see Table 2.1) is used to designate a variety of possible nucleotides at a single position in low quality DNA sequences (IUPAC-IUB 1985). For example, W means the identity of a position is A or T. On the other hand, as reviewed in Chapter 1, in DNA transcription factor binding sites (TFBS), only some positions are conserved among all the sites, and the other positions have a range of variability. Thus, a transcription factor binding site is usually represented by a degenerate consensus sequence or a position weight matrix (PWM). As the number of identified TFBS is increasing rapidly, it is necessary to develop algorithms to evaluate similarity between TFBS for reducing redundant data in the TFBS database (Sandelin, et al., 2004; Wingender, et al., 2001; Zhu and Zhang, 1999), finding homologous site family.

In this chapter, with hypercomplex number representation of DNA, the conventional pairwise alignment methods will be extended to compare DNA sequences with fuzzy composition.

2.2 Hypercomplex number

The hypercomplex number (also called quaternion number), is a combination of four numbers, one real and three imaginary (Hamilton, 1866). The common notation, extended from that of the complex number is:

$$Q = r + xi + yj + zk, \quad (2.1)$$

Chapter 2 Pairwise alignment with hypercomplex representation

where r, x, y and z are real, and i, j, k are complex operators which obey the following rules:

$$i^2 = j^2 = k^2 = ijk = -1, \quad (2.2)$$

$$ij = k, jk = i, ki = j, ji = -k, kj = -i, ik = -j. \quad (2.3)$$

The conjugate of hypercomplex number $\bar{Q}$ is defined as:

$$\bar{Q} = r - xi - yj - zk. \quad (2.4)$$

To facilitate the expression of hypercomplex number product, the hypercomplex number can be regarded as a combination of one scalar and one three-component vector, thus:

$$Q = r + \vec{V}. \quad (2.5)$$

Using vector algebra notation, the hypercomplex number product can be written as:

$$[a + \vec{A}][b + \vec{B}] = ab - \vec{A} \cdot \vec{B} + a\vec{B} + b\vec{A} + \vec{A} \times \vec{B}, \quad (2.6)$$

where $[a + \vec{A}]$ and $[b + \vec{B}]$ are two hypercomplex numbers, ‘ $\cdot$ ’ and ‘ $\times$ ’ denote dot product and cross product respectively.

2.3 Hypercomplex number representations of DNA nucleotide base codes

By permutation and combination, the total number of possible mixed DNA base-nucleic acid code is $2^4 = 16$. Since there are four types of nucleotide, a four-dimensional space is essential to represent the DNA codes fully. The hypercomplex number system required here is a third-order system of the form

$$Q = P_A + P_T i + P_G j + P_C k = (P_A, P_T, P_G, P_C). \quad (2.7)$$

Chapter 2 Pairwise alignment with hypercomplex representation

To assign the values for P_A , P_T , P_G , and P_C , the probability of each DNA base appearing in the DNA base-nucleic acid codes is taken into consideration. The values of P_A , P_T , P_G , and P_C indicate the probabilities of the particular base being A, T, G, and C respectively, satisfying the basic principle:

$$P_A + P_T + P_G + P_C = 1. \quad (2.8)$$

According to the above principle, the hypercomplex number representations of the IUPAC DNA nucleotide base codes were derived and these are listed in Table 2.1. Thus a DNA sequence of length N can be represented by a hypercomplex number sequence of length N . For example, AATCAG can be represented by $(1, 1, i, k, 1, j)$; consensus sequence 'TATAWAWR' can be represented by the hypercomplex number sequence $(i, 1, i, 1, 0.5+0.5i, 1, 0.5+0.5i, 0.5+0.5j)$.

Chapter 2 Pairwise alignment with hypercomplex representation

Table 2.1 The IUPAC codes and their hypercomplex number representations.

Code	Meaning	Explanation	Hypercomplex Representation
–	No base	No base	(0, 0, 0, 0)
A	A	Adenine	(1, 0, 0, 0)
T	T	Thymine	(0, 1, 0, 0)
G	G	Guanine	(0, 0, 1, 0)
C	C	Cytosine	(0, 0, 0, 1)
W	A or T	Weak interaction 2 H bonds	($\frac{1}{2}$, $\frac{1}{2}$, 0, 0)
R	A or G	PuRine	($\frac{1}{2}$, 0, $\frac{1}{2}$, 0)
M	A or C	aMino	($\frac{1}{2}$, 0, 0, $\frac{1}{2}$)
K	G or T	Keto	(0, $\frac{1}{2}$, $\frac{1}{2}$, 0)
Y	C or T	pYrimidine	(0, $\frac{1}{2}$, 0, $\frac{1}{2}$)
S	C or G	Strong interaction 3 H bonds	(0, 0, $\frac{1}{2}$, $\frac{1}{2}$)
D	A, G or T not C	D follows C in alphabet	($\frac{1}{3}$, $\frac{1}{3}$, $\frac{1}{3}$, 0)
H	A, C or T not G	H follows G in alphabet	($\frac{1}{3}$, $\frac{1}{3}$, 0, $\frac{1}{3}$)
V	A, C or G not T(not U)	V follows U in alphabet	($\frac{1}{3}$, 0, $\frac{1}{3}$, $\frac{1}{3}$)
B	C, G or T not A	B follows A in alphabet	(0, $\frac{1}{3}$, $\frac{1}{3}$, $\frac{1}{3}$)
N	Any base	Any base	($\frac{1}{4}$, $\frac{1}{4}$, $\frac{1}{4}$, $\frac{1}{4}$)

2.4 Dot matrix with hypercomplex number representation of DNA

The implementation of the hypercomplex number representations of DNA bases codes in the dot matrix method brings forth an improvement to the conventional dot matrix method, and makes it powerful for comparing DNA sequences with fuzzy composition.

In sequence alignment by means of the conventional dot matrix method, a dot is placed where there is a match between the two sequences. In the dot matrix analysis using hypercomplex number representation of DNA bases codes, whether a dot is placed in a comparison of two DNA sequences is determined by dot product of the hypercomplex number representation of the DNA bases code and a truncation value set.

The probability of finding a match between the sequences is implied in dot product because the hypercomplex number representation assigned to each of the DNA base code is based on the probability of each nucleotide base appeared in the code. For example, in a comparison of two sequences, for an alignment between residues H and S, which have the hypercomplex number representation of $(\frac{1}{3}, \frac{1}{3}, 0, \frac{1}{3})$ and $(0, 0, \frac{1}{2}, \frac{1}{2})$, respectively, the dot product value is derived as

$$Q_H \cdot Q_S = (\frac{1}{3} \ \frac{1}{3} \ 0 \ \frac{1}{3}) \cdot (0 \ 0 \ \frac{1}{2} \ \frac{1}{2}) = 0.17 \quad (2.9)$$

This means that the match probability between the residues H and S is 0.17.

Chapter 2 Pairwise alignment with hypercomplex representation

In other words, based on the dot product value (between 0 and 1) of the hypercomplex number representation of the DNA residues in each sequence being compared, the conventional dot matrix analysis has the truncation set at the value of 1 (i.e., any value less than 1 will be truncated to 0).

Unlike the conventional dot matrix analysis, which has the fixed truncation value of 1, now a truncation value can be set for a desired stringency in finding a possible match: a higher value for higher stringency. For example, in order to prevent short diagonals from appearing too frequently and making the matrix too noisy to identify actual required aligned regions, a higher truncation value may be set so as to reduce the number of dots between the two sequences.

To illustrate the influence of various factors on the outcome of a dot matrix diagram, the following pair of sequences is selected as an example.

T G R B W B H K M W C Y
S Y A G M W D S H V R K

The dot product of the alignment between each base of the sample sequences is obtained and shown in a matrix in Table 2.2.

The varying parameters include the truncation value, the window size and the stringency (minimum requirement of number of dots to be present in the diagonal of window before a dot is placed between the alignments of the bases).

Chapter 2 Pairwise alignment with hypercomplex representation

Table 2.2 Value of dot product of hypercomplex number representation per aligned base pair of example sequences.1/3

	T	G	R	B	W	B	H	K	M	W	C	Y
S	0	0.5	0.25	0.33	0	0.33	0.17	0.25	0.25	0	0.5	0.25
Y	0.5	0	0	0.33	0.25	0.33	0.33	0.25	0.25	0.25	0.5	0.5
A	0	0	0.5	0	0.5	0	0.33	0	0.5	0.5	0	0
G	0	1	0.5	0.33	0	0.33	0	0.5	0	0	0	0
M	0	0	0.25	0.17	0.25	0.17	0.33	0	0.5	0.25	0.5	0.25
W	0.5	0	0.25	0.17	0.5	0.17	0.33	0.25	0.25	0.5	0	0.25
D	0.33	0.33	0.33	0.22	0.33	0.22	0.22	0.33	0.17	0.33	0	0.17
S	0	0.5	0.25	0.33	0	0.33	0.17	0.25	0.25	0	0.5	0.25
H	0.33	0	0.17	0.22	0.33	0.22	0.33	0.17	0.33	0.33	0.33	0.33
V	0	0.33	0.33	0.22	0.17	0.22	0.22	0.17	0.33	0.17	0.33	0.17
R	0	0.5	0.5	0.17	0.25	0.17	0.17	0.25	0.25	0.25	0	0
K	0.5	0.5	0.25	0.33	0.25	0.33	0.17	0.5	0	0.25	0	0.25

2.4.1 Effect of truncation value on the dot matrix analysis

Based on the dot product value per aligned residue of the sample sequences, the dot matrix diagram for comparing them is shown on Figure 2.1, where the truncation values are set at 0.3 and 0.5 respectively. The sequences are compared on a one to one residue basis. The dots are placed where the dot product values of the corresponding bases meet the designed truncation value.

Varying the truncation value changes the number of dots appeared on the dot matrix diagram. For the case of Figure 2.1(a), many dots are present, as the truncation value is set relatively low. The high concentration of dots on the diagram makes it deceive that there are many matched regions. However, after inserting diagonals, it is obvious that many dots are not collinear. They are only random matches all over the matrix. In addition, the number of aligned regions is also higher on Figure 2.1(a) than on Figure 2.1(b) as the truncation value indicates stringency in finding matches. With a lower truncation value on Figure 2.1(a), we are actually looking for a higher number of possible matches even with a small probability than a more certain alignment as is Figure 2.1(b) which has a higher truncation value. In actual practice, truncation value is set based on user's toleration of uncertain matches. One can try low truncation value first, such as 0.3, then increasing it to 0.5. When the new matrix of higher truncation value is compared with the previous matrix, one can get the location of more certain matching regions.

Chapter 2 Pairwise alignment with hypercomplex representation

Figure 2.1 Dot matrix of example sequences with truncation values of (a) 0.3 and (b) 0.5.

Chapter 2 Pairwise alignment with hypercomplex representation

2.4.2 Effect of window size and stringency on the dot matrix analysis

Using a truncation value of 0.3 in each window for the dot matrix analysis of the sample sequences, the influence of window size and stringency on the dot matrix diagram is investigated.

On Figure 2.2, a window size of 3 and stringency of 1 are used. Figure 2.2 depicts how a low stringency in relative terms of a small window can lead to the toleration of more partial and accidental matches. Low stringency and small windows are recommended in cases where the similarity may be weak or the genome has undergone extensive mutations.

Figure 2.3 is a case where there is a high stringency and a large window (window size of 5 and stringency of 3). Compared with Figure 2.2, less matching regions are located with this combination of parameters. The basic law of probability dictates that the chances of accidental matches are very low. This "unforgiving" arrangement between the stringency and the window size is used in cases where the hypothesis for similarity is strong, or in case where there are similar genomes mutating slowly.

In actual practice, window size of 21 and stringency of 14 are recommended for the sequence of length 1000.

Figure 2.2 Dot matrix of example sequences with a sliding window size of 3 and stringency of 1.

Chapter 2 Pairwise alignment with hypercomplex representation

Figure 2.3 Dot matrix of example sequences with a sliding window size of 5 and stringency of 3.

Chapter 2 Pairwise alignment with hypercomplex representation

2.5 Scoring matrix for hypercomplex number representation

In order to accommodate hypercomplex number representations of DNA base codes and incorporate with conventional sequence alignment method, a scoring matrix is needed to be derived.

The new scoring matrix is introduced by initially taking dot product of the DNA base hypercomplex number representation shown in Table 2.1. :

$$Q_X \cdot Q_Y = (P_A^X P_T^X P_G^X P_C^X) \cdot (P_A^Y P_T^Y P_G^Y P_C^Y) = P_A^X P_A^Y + P_T^X P_T^Y + P_G^X P_G^Y + P_C^X P_C^Y \quad (2.10)$$

The result of the dot product represents the probability of finding a match between the sequences. The new score values are then calculated using

$$s(X, Y) = (Q_X \cdot Q_Y) \times 20 - 5 \quad (2.11)$$

Where the highest aligned score is 15 and the lowest one is -5, with a gap penalty of $d = 8$ for computational efficiency. After scaling the dot product value and rounding off to the nearest integer, the new scoring matrix is shown in Table 2.3.

The conventional alignment algorithms are used together with hypercomplex number representations of DNA base codes and the new scoring matrix introduced here. A pair of DNA sequences is used throughout the rest of this chapter as a demonstration of the feasibility of using this new scoring matrix.

H	T	A	G	A	W	M	H	R	Y
T	A	W	H	C	A	M	B	H	R

Chapter 2 Pairwise alignment with hypercomplex representation

Table 2.3 New scoring matrix derived from dot product of hypercomplex number representation of DNA bases codes.

		A	T	G	C	W	R	M	K	Y	S	D	H	V	B	N	
		-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	-5	
A		-5	15	-5	-5	-5	5	5	5	-5	-5	-5	2	2	2	-5	0
T		-5	-5	15	-5	-5	5	-5	-5	5	5	-5	2	2	-5	2	0
G		-5	-5	-5	15	-5	-5	5	-5	5	-5	5	2	-5	2	2	0
C		-5	-5	-5	-5	15	-5	-5	5	-5	5	5	-5	2	2	2	0
W		-5	5	5	-5	-5	5	0	0	0	0	-5	2	2	-2	-2	0
R		-5	5	-5	5	-5	0	5	0	0	-5	0	2	-2	2	-2	0
M		-5	5	-5	-5	5	0	0	5	-5	0	0	-2	2	2	-2	0
K		-5	-5	5	5	-5	0	0	-5	5	0	0	2	-2	-2	2	0
Y		-5	-5	5	-5	5	0	-5	0	0	5	0	-2	2	-2	2	0
S		-5	-5	-5	5	5	-5	0	0	0	0	5	-2	-2	2	2	0
D		-5	2	2	2	-5	2	2	-2	2	-2	-2	2	-1	-1	-1	0
H		-5	2	2	-5	2	2	-2	2	-2	2	-2	-1	2	-1	-1	0
V		-5	2	-5	2	2	-2	2	2	-2	-2	2	-1	-1	2	-1	0
B		-5	-5	2	2	2	-2	-2	-2	2	2	2	-1	-1	-1	2	0
N		-5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Chapter 2 Pairwise alignment with hypercomplex representation

2.5.1. Global alignment using hypercomplex number representation

A matrix F indexed by i and j , one index for each sequence, is constructed, where the value of $F(i, j)$ is the score of the best alignment between the initial segment $X_1, X_2, \dots, X_i$ and the initial segment $Y_1, Y_2, \dots, Y_j$.

$F(i, j)$ are calculated with the known $F(i-1, j-1)$, $F(i-1, j)$, $F(i, j-1)$. The best score of an alignment is obtained in three ways: alignment of X_i with Y_j ; or alignment of X_i with a gap; or alignment of Y_j with a gap. The best score up to (i, j) giving the optimal alignment is the highest of these three options. Hence,

$$F(i, j) = \max \begin{cases} F(i-1, j-1) + s(X_i, Y_j) \\ F(i-1, j) - d \\ F(i, j-1) - d \end{cases} \quad (2.12)$$

The matrix of $F(i, j)$ values is built recursively by initializing $F(0, 0) = 0$, then filling the matrix from top left to bottom right using the other three values, as illustrated in Figure 2.4. For boundary conditions along the top row where $j = 0$ and the leftmost column where $i = 0$, the values of $F(i, 0)$ and $F(0, j)$ are defined as $F(i, 0) = -id$ and $F(0, j) = -jd$. As the $F(i, j)$ value is filled, a pointer is kept in each cell back to the cell from which the value is derived.

The value in the final cell of the matrix is by definition the best score for an alignment of X and Y , which is the score of the best global alignment of X to Y . A traceback is done to find this global alignment by building the alignment in reverse, starting from the final cell and following the pointers kept when building the matrix. A pair of

Chapter 2 Pairwise alignment with hypercomplex representation

symbols is added onto the front of the current alignment with each step moved in the traceback process: X_i and Y_j if the step was to $(i-1, j-1)$, or X_i and the gap character ‘-’ if the step was to $(i-1, j)$, or ‘-’ and Y_j if the step was to $(i, j-1)$. This traceback procedure finds only one alignment with the optimal score. Thus an arbitrary choice is made between the two options if the derivations at any point are equal.

Because the score is a sum over independent pieces, this algorithm is feasible. This best score up to some point in the alignment is the best score up to the point one step before, plus the increment score of the new step.

Using the new scoring matrix in Table 2.3, the global dynamic programming matrix shown in Figure 2.5 is set up using the example DNA sequence pair.

From Figure 2.5, the corresponding optimal alignment of the two sequences with a total score of 14 is obtained as follows:

H	T	A	G	A	W	M	H	R	-	Y	-
-	T	A	-	W	H	C	A	M	B	H	R

Chapter 2 Pairwise alignment with hypercomplex representation

Figure 2.4 Derivation option for the $F(i, j)$ value

Chapter 2 Pairwise alignment with hypercomplex representation

Figure 2.5 Global dynamic programming matrix for hypercomplex number representation of DNA sequences

2.5.2. Local alignment using hypercomplex number representation

Compared to the case of a global alignment, a more common situation occurs when the best alignment between subsequences of X and Y is required. This alignment most sensitively detects similarity between two highly diverged sequences that might have a common evolutionary origin along their entire length. The highest scoring alignment of such subsequences is the best local alignment.

The difference lies in the feature that for local alignment, an extra possible value for $F(i, j)$ is added such that if all other options have a value of less than 0, $F(i, j)$ take the value of 0:

$$F(i, j) = \max \begin{cases} F(i-1, j-1) + s(X_i, Y_j) \\ F(i-1, j) - d \\ F(i, j-1) - d \\ 0 \end{cases} \quad (2.13)$$

Once the value of $F(i, j)$ takes the option value of 0, a new alignment is started. The new option of 0 results in the top row and leftmost column taking the value of 0 instead of $-id$ and $-jd$ as in the case of global alignment.

In addition to the first difference, now in local alignment, an alignment could end anywhere in the matrix. The best score need not be in the bottom right corner. Instead, the traceback starts as the highest value of $F(i, j)$ over the whole matrix and ends when it reaches a cell with value 0 which corresponds to the start of the alignment.

Chapter 2 Pairwise alignment with hypercomplex representation

The basis for this local alignment algorithm working is that expected score for a random match must be negative; otherwise the scores for long matches between entirely unrelated sequences will be high on the basis of their lengths. As a result, the maximal scoring alignments would be global or nearly global although the algorithm is local. Similarly, there must be some score values higher than 0; if not, the algorithm cannot find any alignment at all.

Using the same pair of DNA sequences with hypercomplex number representation, the local dynamic programming algorithm is implemented to give the matrix in Figure 2.6. In the local dynamic programming matrix, it is not necessary to start the alignment at the bottom right cell. Instead, the alignment starts at the cell with the highest score so that the optimal local alignment can be found. In this case, the highest score is 38. Thus the traceback starts from there and ends when it reaches a score of 0. The optimal local alignment of this pair of example sequences has a score of 38 and is found to be

T	A	G	A	W	M	H	R	Y
T	A	—	W	H	C	A	M	B

Chapter 2 Pairwise alignment with hypercomplex representation

Figure 2.6 Local dynamic programming matrix for hypercomplex number representation of DNA sequences

Chapter 2 Pairwise alignment with hypercomplex representation

2.6 Discussion and conclusion

To fully represent the DNA base-nucleic codes in hypercomplex number, a four-dimensional space is required. The representation number assigned to each base code takes the probability of each nucleotide in the DNA base code into consideration. In real situation, it was found that the DNA sequences, especially noncoding DNA sequences, exhibit larger values of redundancy (Mantegna, et al., 1994). Thus, the probabilities for the occurrence of each nucleotide appearing in the DNA base code are not equal as shown in Table 2.1. In practice, the values assigned to P_A , P_T , P_C , and P_G are equal to the intensity of corresponding fluorescent tag respectively.

The implementation of hypercomplex number representation of DNA in the dot matrix method brings forth an improvement to the conventional method of placing a dot when there is a match between the corresponding residues of two sequences. As the hypercomplex number representation of DNA base codes is in numbers instead of alphabetical characters, the significance of probabilistic occurrence of each base in specific position is emphasized. To determine whether a dot should be placed between the aligned residues, the dot product of the hypercomplex number representations of the bases is taken and truncated. With introduction of 'value' instead of 'dots' as in the conventional dot matrix method (Mount, 2001), the truncation value can be varied. And thus, a greater control over the degree of alignment desired, besides the current control of window size and stringency, is possible. A higher truncation value corresponds to a higher stringency for longer matching regions between the sequences. With the addition of a new factor

Chapter 2 Pairwise alignment with hypercomplex representation

contributing to the outcome of the dot matrix diagram, more combinations of three parameters can be selected to appropriately produce a more accurate dot matrix analysis for the desired condition of matches.

In addition, an implied advantage of the variable truncation value using the hypercomplex representation is that the sequence may not need to be further analyzed for actual matching regions using dynamic programming. The method of imaging may be used to overlap dot matrices of a similar pair of sequences but of increasing truncation value. When the new matrix of higher truncation value is imposed on the previous matrix, a clearer picture of the location of the actual matching regions is superimposed on the screen.

To use the hypercomplex number representation of DNA sequences, a new scoring model has been derived. The new model, with the consideration of probability of each nucleotide presented in the DNA base-nucleic acid codes, use the dot product arithmetic of the residues of the sequences to be matched. The dot product value is scaled and rounded off to an integer. The dynamic programming algorithm has been applied to the sample sequence in hypercomplex number representation and the feasibility of using the hypercomplex number representation and scoring model has been verified. As most of the DNA codes consist of mixed bases, the alignments obtained for the dynamic programming algorithm is very high. This is because the algorithm can detect a possible alignment with small possibility of a match between the two sequences.

Chapter 3

Cross-correlation of DNA with hypercomplex number representation

3.1 Introduction

Cross-correlation and fast Fourier transform (FFT) have been widely used for DNA sequence matching (Felsenstein, et al., 1982; Rajasekaran, et al., 2002). Compared to conventional alignment method of Smith-Waterman (1981), and Needleman-Wunch (1970), where the computational effort is proportional to the square of sequence length ($O(N^2)$), sequence alignment using cross-correlation can be implemented as a fast ($O(N\log N)$) process. That makes it computational efficient when applied to large data sets. Furthermore, this method is robust in dealing with insertion, deletions, tandem repeats, and translocations, which are rather difficult to be detected by other popular methods for sequence alignment (Rockwood, et al., 2005).

Cross-correlation is an engineering technique for comparing two numerical signals. When being used to compare two DNA sequences, it was cooperated with two kinds of numerical representation method. One is 4-vector encoding by which a DNA sequence is decomposed into 4 independent binary indicator sequences (Felsenstein, et al., 1982; Rajasekaran, et al., 2002). The other is complex number encoding by which a DNA sequence is converted into one complex number sequence (A, T, G and C are mapped to 1, -1 , i , and $-i$, respectively) (Cheever, et al., 1991; Rockwood, et al.,

Chapter 3 Cross-correlation of DNA with hypercomplex representation

2005). With the 4-vector encoding, four cross-correlations are needed to get the total match count between a pair of DNA sequences at each alignment. The disadvantage of this method is that all the mismatch information between the two DNA sequences is missed. With the complex number encoding, only one cross-correlation is needed to get the match and mismatch information. However, the real part of the cross-correlation equals to the number of matches at each alignment of two aligned DNA sequences minus the number of complementary mismatch (alignment between A and T as well as alignment between G and C). So the real part of cross-correlation is actually an approximate measure of the similarity between the pair of DNA sequences. The peak in the real part plot, which indicates the similarity between the pair of DNA sequences, may disappear due to the offset of the complementary mismatch to the match. On the other hand complex number representation cannot encode the DNA base codes with fuzzy composition, such as IUPAC codes (Cornish-Bowden, 1985) for representation of DNA transcription factor binding sites (TFBS). This disadvantage limits the use of cross-correlation to compare DNA sequences with fuzzy composition.

Hypercomplex number representation of DNA base code proposed in chapter 2 can take the occurrence probability of each nucleotide in a DNA nucleotide base code into full account; it is very suitable for numerical encoding of DNA base code with fuzzy composition. Cooperated with development in the hypercomplex signal processing technique (Pei, et al., 2001), cross-correlation method will be improved to compare DNA sequences with fuzzy composition by using the hypercomplex number encoding of DNA sequences.

Chapter 3 Cross-correlation of DNA with hypercomplex representation

3.2 Hypercomplex number representation of DNA motif

DNA motifs are short conserved domains that are presumed to have biological functions. Often they indicate sequence-specific binding sites for proteins such as nucleases and transcription factors. In DNA binding sites, only some positions are conserved among all the sites, and the other positions have a range of variability. So a DNA binding site is usually represented by a consensus sequence or a position weight matrix (PWM). As an example, the base frequency, corresponding consensus sequence and PWM of the TATA-box for *eukaryotic RNA polymerase II promoters* are listed in Table 3.1.

The TATA-box can be represented by the following three different hypercomplex number vectors:

(1) Consensus sequence based hypercomplex number representation of TATA box:

$$A = (i, 1, i, 1, 0.5+0.5i, 1, 0.5+0.5i, 0.5+0.5j),$$

According to hypercomplex number representation of DNA base code (see Chapter 2, Table 2.1). The consensus sequence 'TATAWAWR' in Table 3.1 can be represented by hypercomplex number vector A

(2) Frequency based hypercomplex number representation of TATA box:

$$B = (0.041+0.795i+0.046j+0.118k, 0.905+0.09i+0.005j, 0.008+0.961i+0.005j+0.026k, \\ 0.91+0.077i+0.013j, \quad 0.689+0.311i, \quad 0.925+0.016i+0.051j+0.008k, \\ 0.571+0.311i+0.113j+0.005k, 0.398+0.085i+0.404j+0.113k).$$

Chapter 3 Cross-correlation of DNA with hypercomplex representation

This representation is based on every column of the base frequency matrix in Table

3.1. Each position of TATA-box is represented by a hypercomplex number ' $\% A + \% T i + \% G j + \% C k$ '.

(3) Position weight matrix (PWM) based hypercomplex number representation of TATA box:

$C = (-3.05 - 2.74j - 2.06k, -2.28i - 4.28j - 5.22k, -4.61 - 4.61j - 3.49k, -2.34i - 3.77j - 5.17k, -0.52i - 4.73j - 4.63k, -3.65i - 2.65j - 4.12k, -0.37i - 1.5j - 3.74k, -0.01 - 1.4i - 1.13k)$.

This representation is based on every column of the position weight matrix in Table

3.1. Each position of TATA-box is represented by a hypercomplex number ' $W(A) + W(T)i + W(G)j + W(C)k$ '. Then, the position weigh matrix is represented by a hypercomplex position weight vector (HPWV).

Chapter 3 Cross-correlation of DNA with hypercomplex representation

Table 3.1 Base frequency, consensus sequence, position weight matrix of TATA-box for Eukaryotic RNA Polymerase II promoters (Bucher, 1990), and their hypercomplex number representation.

Position	1	2	3	4	5	6	7	8
% A	4.1	90.5	0.8	91.0	68.9	92.5	57.1	39.8
% T	79.5	9.0	96.1	7.7	31.1	1.6	31.1	8.5
% G	4.6	0.5	0.5	1.3	0.0	5.1	11.3	40.4
% C	11.8	0.0	2.6	0.0	0.0	0.8	0.5	11.3
Consensus	T	A	T	A	W	A	W	R
W(A)	-3.05	0	-4.61	0	0	0	0	-0.01
W(T)	0	-2.28	0	-2.34	-0.52	-3.65	-0.37	-1.4
W(G)	-2.74	-4.28	-4.61	-3.77	-4.73	-2.65	-1.5	0
W(C)	-2.06	-5.22	-3.49	-5.17	-4.63	-4.12	-3.74	-1.13

(1) Consensus sequence based hypercomplex number representation of TATA box:

$$A = (i, 1, i, 1, 0.5+0.5i, 1, 0.5+0.5i, 0.5+0.5j)$$

(2) Frequency based hypercomplex number representation of TATA box:

$$B = (0.041+0.795i+0.046j+0.118k, 0.905+0.09i+0.005j, 0.008+0.961i+0.005j+0.026k, \\ 0.91+0.077i+0.013j, 0.689+0.311i, 0.925+0.016i+0.051j+0.008k, \\ 0.571+0.311i+0.113j+0.005k, 0.398+0.085i+0.404j+0.113k).$$

(3) PWM based hypercomplex number representation of TATA box:

$$C = (-3.05-2.74j-2.06k, -2.28i-4.28j-5.22k, -4.61-4.61j-3.49k, -2.34i-3.77j-5.17k, \\ -0.52i-4.73j-4.63k, -3.65i-2.65j-4.12k, -0.37i-1.5j-3.74k, -0.01-1.4i-1.13k).$$

Chapter 3 Cross-correlation of DNA with hypercomplex representation

3.3 Hypercomplex cross-correlation

Consider $\mathbf{x} = (x_0, x_1, \dots, x_{N-1})$ and $\mathbf{y} = (y_0, y_1, \dots, y_{N-1})$, two hypercomplex vectors of length N . The hypercomplex cross-correlation operation of $\mathbf{x}$ and $\mathbf{y}$ is defined to the hypercomplex vector $\mathbf{z} = (z_0, z_1, \dots, z_{N-1})$:

$$z_l = \sum_{n=0}^{N-1} [x(n+l) \bmod N] \overline{y(n)} \quad l = 0, 1, 2, \dots, N-1, \quad (3.1)$$

where $\overline{y(n)}$ is the conjugate of $y(n)$, the shift operation on $x(n)$ is carried out cyclically using modulo arithmetic for the subtraction.

3.3.1 Pairwise DNA sequence alignment with hypercomplex cross-correlation

Hypercomplex cross-correlation is an extension of cross-correlation concept to hypercomplex vector. So the process of using hypercomplex cross-correlation to compare DNA sequences is same with the one used in (Cheever, et al., 1991; Rockwood, et al., 2005), the difference is that we use hypercomplex number to encode DNA base code: 1 for A, i for T, j for G and k for C. The algorithm for pairwise DNA sequence alignment with hypercomplex cross-correlation is listed in appendix A.2.

When cross-correlation is performed between two hypercomplex represented DNA sequences, according to the hypercomplex algebra, the hypercomplex product (Equation (3.1)) results in 1 if the two aligned bases are matched, and $\pm i$ or $\pm j$ or $\pm k$ if they are mismatched (for the detailed results, see Table3.2), here $\pm i$ corresponding to

Chapter 3 Cross-correlation of DNA with hypercomplex representation

the complementary mismatch (A vs. T or G vs. C); $\pm j$ to the transitional mismatch (A vs. G or T vs. C or); and $\pm k$ to the transversion mismatch (A vs. C or G vs. T). Thus, the real part of the hypercomplex cross-correlation gives the number of matches at each alignment of the two DNA sequences, the imaginary part reflects the mismatching information: every complementary mismatch TA or CG increases i imaginary part (x) by one, due to the non-commutation of hypercomplex product, every mismatch AT or GC decreases x by one; similarly every transitional mismatch TC or GA increase j imaginary part (y) by one, every mismatch CT or AG decreases y by one; every transversion mismatch GT or CA increases k imaginary part (z) by one, every mismatch TG or AC decreases z by one. In summary, we have:

$$\begin{cases} r = n_{AA} + n_{TT} + n_{GG} + n_{CC} \\ x = n_{TA} + n_{CG} - n_{AT} - n_{GC} \\ y = n_{TC} + n_{GA} - n_{AG} - n_{CT} \\ z = n_{GT} + n_{CA} - n_{AC} - n_{TG} \end{cases} \quad (3.2)$$

where r , x , y and z denote the real and three imaginary parts of hypercomplex cross-correlation z_i ; n_{XY} denotes the number of aligned base pair XY ($X, Y \in \{A, T, G, C\}$). Thus, a peak in the real part plot of hypercomplex cross-correlation indicates similarity between two DNA sequences. Furthermore, because the real part of hypercomplex cross-correlation only reflects the match information, the peak value can be used as a measure of similarity between two DNA sequences: the larger the peak, the more similar the two DNA sequences.

Chapter 3 Cross-correlation of DNA with hypercomplex representation

Table 3.2 Hypercomplex product results between hypercomplex represented DNA base codes

	A	T	G	C
A	1	i	j	k
T	$-i$	1	k	$-j$
G	$-j$	$-k$	1	i
C	$-k$	j	$-i$	1

For matched aligned bases the hypercomplex product results in 1; for mismatched aligned bases: the hypercomplex product results in $\pm i$ if the aligned two bases are complementary mismatch (A vs. T or G vs. C); the hypercomplex product results in $\pm j$ if the aligned two bases are transitional mismatch (A vs. G or T vs. C); the hypercomplex product results in $\pm k$ if the aligned two bases are transversion mismatch (A vs. C or G vs. T).

Chapter 3 Cross-correlation of DNA with hypercomplex representation

3.3.2 Motif scanning with hypercomplex cross-correlation

When the vector y in Equation (3.1) is the hypercomplex number representation of consensus sequence, the hypercomplex cross-correlation can be used to do consensus sequence based motif scanning. If the aligned two DNA bases x and y have hypercomplex number representation $Q_x = [P_A^x + \overrightarrow{(P_T^x, P_G^x, P_C^x)}]$ and $Q_y = [P_A^y + \overrightarrow{(P_T^y, P_G^y, P_C^y)}]$, the multiplication in (3.1) is denoted as R_{xy} and calculated by:

$$\begin{aligned}
 R_{xy} = Q_x \cdot \overline{Q_y} = & P_A^x P_A^y + P_T^x P_T^y + P_G^x P_G^y + P_C^x P_C^y \\
 & + (P_T^x P_A^y + P_C^x P_G^y - P_A^x P_T^y - P_G^x P_C^y)i \\
 & + (P_T^x P_C^y + P_G^x P_A^y - P_C^x P_T^y - P_A^x P_G^y)j \\
 & + (P_G^x P_T^y + P_C^x P_A^y - P_T^x P_G^y - P_A^x P_C^y)k
 \end{aligned} \tag{3.3}$$

The real part of R_{xy} is the probability of finding a match in the alignment since the hypercomplex number representation assigned to each of the DNA nucleotide base code is based on the probability that each base appears in the base code. The imaginary parts of R_{xy} reflect the probability of finding a mismatch in the alignment: the i imaginary part is the probability of finding TA or CG pair minus the probability of finding AT or GC pair; the j imaginary part is the probability of finding TC or GA pair minus the probability of finding CT or AG pair; the k imaginary part is the probability of finding GT or CA pair minus the probability of finding TG or AC pair. Thus the real part of the hypercomplex cross-correlation z_l (see Equation (3.1)) gives a score that can be used to measure the similarity between DNA substring and the consensus sequence.

Chapter 3 Cross-correlation of DNA with hypercomplex representation

The algorithm for motif scanning with hypercomplex cross-correlation is listed in appendix A.3.

If the vector y in Equation (3.1) is changed to the hypercomplex number representation of PWM, the hypercomplex cross-correlation can be used to do profile matching (Freschi and Bogliolo, 2005; Gonnet, 2004; Gusfield, 1997). The real part of multiplication in Equation (3.1) is $P_A W_A + P_T W_T + P_G W_G + P_C W_C$. Thus, the value of real part of the hypercomplex cross-correlation z_l is equal to the profile matching score.

3.3.3 Implementation the hypercomplex cross-correlation by hypercomplex Fourier transform

The time complexity of straightforward calculation of hypercomplex cross-correlation is $O(N^2)$. However, the calculation of the hypercomplex cross-correlation can be reduced to $O(N \log N)$ by newly proposed hypercomplex Fourier Transform (HFT) (Bülow, 1999; Bülow, 2001; Pei, et al., 2001). The detail calculation of the hypercomplex cross-correlation by HFT is list in Appendix A.4.

3.4 Results and discussion

3.4.1 Pairwise alignment of DNA with hypercomplex cross-correlation

Three Human Immunodeficiency Virus (HIV) sequences are used to show the application of hypercomplex cross-correlation for DNA pairwise alignment. There are two human type 1 isolates HIVMN (HIV type 1, isolate MN, accession number M17449) and HIVRF (HIV type 1, isolate RF, accession number M17451), and one simian virus, SIVMM142 (Simian immunodeficiency virus isolate MM142m-83, accession number Y00277). Each sequence is between 9000 and 10,000 bases in length.

Figure 3.1(a) shows the real part of the hypercomplex cross-correlation of HIVMN with HIVRF. The horizontal axis is the relative position of HIVRF with respect to HIVMN, the vertical axis is the number of matches. Figure 3.1(b) is similar to (a) but shows the real part of the hypercomplex cross-correlation of HIVMN with SIVMM142.

A peak in the real part plot of the hypercomplex cross-correlation indicates similarity between two aligned sequences. The peak value can be used to measure the similarity, for example, as expected, the peak around the relative position 0, is much larger for the comparison of two different HIV isolates than for comparison of the SIV and HIV isolates. There are also two small peaks at around -9000 and +9000 in Figure 3.1(a), which corresponds to the long terminal repeat at each end of the virus.

Chapter 3 Cross-correlation of DNA with hypercomplex representation

Figure 3.1 Real part of hypercomplex cross-correlation of (a) HIVMN against HIVRF and (b) HIVMN against SIVMM142

Chapter 3 Cross-correlation of DNA with hypercomplex representation

The human TGFA sequence (accession number K03222, 867bp) and mouse TGFA sequence (accession number BC003895, 1024bp) are used to investigate the effects of insertion or deletion on the cross-correlation results. Insertion or deletion of a single nucleotide base produces a frame shift, which by the cross-correlation function will produce a split peak with a spacing of one unit as shown in Figure 3.3. And the peak amplitude will be reduced due to peak splitting. This reduction is akin to a “gap penalty” as found in conventional alignment methods. This decrease in amplitude is a useful scoring metric, with no adjustable parameters necessary for gap penalties. Insertion or deletion of larger nucleotide sizes would produce corresponding reduction in amplitude and characteristic split peak in the cross-correlation alignment as shown in Figure 3.4. This is why the cross-correlation has advantage in dealing insertion and deletion between compared sequences.

Chapter 3 Cross-correlation of DNA with hypercomplex representation

Figure 3.2 Real part of hypercomplex cross-correlation of (a) human TGFA sequence against mouse TGFA sequence; (b) the detailed central region in (a).

Chapter 3 Cross-correlation of DNA with hypercomplex representation

Figure 3.3 Real part of hypercomplex cross-correlation of (a) human TGFA sequence with 1 bp deletion in the center position against mouse TGFA sequence; (b) the detailed central region in (a).

Chapter 3 Cross-correlation of DNA with hypercomplex representation

Figure 3.4 Real part of hypercomplex cross-correlation of (a) human TGFA sequence with 70 bp deletion in the center position against mouse TGFA sequence; (b) the detailed central region in (a).

3.4.2 Motif scanning with hypercomplex cross-correlation

Another application of hypercomplex cross-correlation is to scan a query DNA sequence for the potential transcription binding sites. Here an example of using hypercomplex cross-correlation is given to locate the potential TATA box in query sequence of H.sapiens H4/g gene (Accession number: X60486.1). The base frequency, consensus sequence and PWM of the TATA-box for Eukaryotic POL II promoters and their corresponding hypercomplex number representation are listed in Table 3.1. From experiment, the TATA signal locates at position 185 as substring 'TATTTAAG'. When the conventional string match method (Prestridge, 1991) is used to scan H.sapiens H4/g gene for potential TATA box, there is not an exact match to the consensus sequence 'TATAWAWR'. However, when using cross-correlation with hypercomplex encoding, the potential TATA box can be found by setting reasonable score threshold.

The real part of hypercomplex cross-correlation of H.sapiens H4/g gene against the consensus sequence 'TATAWAWR' is shown in Figure 3.2. If the score threshold is set as 5.5, two potential TATA boxes are identified to be located at position 185 and 381 of the query DNA sequence respectively. Then the best candidate should be identified to show most similar to consensus sequence. One simply way is to line the candidate substring against the consensus sequence and then perform the base-by-base comparison. This operation can be expressed in an alignment graph. For the candidate at position 185 (candidate 1: 'TATTTAAG'), the alignment graph reads as follow:

Chapter 3 Cross-correlation of DNA with hypercomplex representation

T	A	T	T	T	A	A	G
T	A	T	A	W	A	W	R

For the candidate at position 381(candidate 2: ‘TATCTATG’), the alignment graph is:

T	A	T	C	T	A	T	G
T	A	T	A	W	A	W	R

From the alignment graphs, the two candidates match the consensus sequence at all the position except the position 4. For the candidate 1, the mismatch is a complementary mutation of ‘A to T’; for the candidate 2, the mismatch is a transversion mutation ‘A to C’. Because complementary mismatches occur more possible than the transversion mismatch in the binding sites, the candidate 1 is more similar to the consensus sequence. Thus the substring at position 185 is the best candidate.

The second approach to identify the best candidate is to use the mismatch information from the hypercomplex cross-correlation. For each peak in the Figure 3.5, the correlation shift value l , is determined, hold l fixed and plot in Figure 3.3 the partial sum function $P_l(m)$:

$$P_l(m) = \sum_{n=0}^{m-1} [x(n+1) \bmod N] \overline{y(n)} \quad m = 1, 2, \dots, N \tag{3.4}$$

The real part of the partial sum reflects the matching information and the imaginary parts reflect the mismatching information. For the real part plot (line 1), a region with slope of +1 indicates a perfect match, a horizontal region corresponds to a mismatch, a region with slope less than 1 corresponds to a fuzzy match (for example the

Chapter 3 Cross-correlation of DNA with hypercomplex representation

alignment between T and W). For the i imaginary part plot (line 2), a region with slope of +1 or -1 indicates a perfect complementary mismatch, a horizontal region indicates no complementary mismatch at that region, a region with slope less than 1 corresponds to a fuzzy mismatch. The similar meaning is for the j and k imaginary part plot (line 3 and 4). Figure 3.6 shows that the mismatch is located at the position 4 of each candidate substring: for candidate 1, the mismatch is a complementary mutation; for candidate 2, the mismatch is a transversion mutation. Thus, the candidate at position 185 is the best candidate.

Another approach to identify the best candidate is to use the hypercomplex number representation of TATA box based on the base frequency at each position (the vector B shown in Table 3.1). The real part of hypercomplex cross-correlation is shown in Figure 3.7. The candidate at the position 185 can easily be identified as the best candidate because it corresponds to maximal score of the real part.

Finally, the real part of hypercomplex cross-correlation result of H.sapiens H4/g gene against the HPWV (the vector C shown in Table 3.1) is shown in Figure 3.8. The value of real part of the hypercomplex cross-correlation is equal to the profile matching score. The substring at position 185 can easily be identified as the most possible TATA box.

Chapter 3 Cross-correlation of DNA with hypercomplex representation

Figure 3.5 Real part of hypercomplex cross-correlation of H4/g gene against consensus sequence ‘TATAWAWR’. Hypercomplex number representation of consensus sequence is vector A in Table 3.1.

Chapter 3 Cross-correlation of DNA with hypercomplex representation

Figure 3.6 Partial sum plots for H.sapiens H4/g gene compared with consensus sequence ‘TATAWAWR’, showing the match and mismatch information from (a) position 185 and (b) position 381.

Chapter 3 Cross-correlation of DNA with hypercomplex representation

Figure 3.7 Real part of hypercomplex cross-correlation of *H. sapiens* H4/g gene against frequency based consensus sequence. Hypercomplex number representation is vector B in Table 3.1.

Chapter 3 Cross-correlation of DNA with hypercomplex representation

Figure 3.8 Real part of hypercomplex cross-correlation of H.sapiens H4/g gene against hypercomplex position weight vector (HPWV) C shown in Table 3.1.

3.4.3 Time complexity for motif scanning with hypercomplex cross-correlation

The time complexity of brute-force algorithm for profile matching is $O((N-P+1)P)$ (where N is the length of the query sequence and P is the length of the PWM). The time complexity for motif scanning with hypercomplex cross-correlation is $O((N+P-1)\log(N+P-1))$. Because N is usually much larger than P , the computing time for brute-force algorithm is $O(NP)$, and the computing time for hypercomplex cross-correlation is $O(N\log N)$. When $P > \log N$, the computing time of motif scanning with hypercomplex cross-correlation is lower than traditional brute-force computation.

Chapter 3 Cross-correlation of DNA with hypercomplex representation

3.5 Conclusion

By using hypercomplex cross-correlation, the match and mismatch information between two aligned DNA sequences are separately stored in the resultant real part and imaginary parts. The real part is an exact measure of the similarity between the pair of DNA sequences at each alignment. The peak in the real part plot, which indicates the similarity between the pair of DNA sequences, is more obvious due to no offset of the complementary mismatch to the match. That makes hypercomplex cross-correlation method more powerful to detect insertion, deletion, and intersequence arrangement.

With the emphasis on the probabilistic occurrence of each DNA base on specific position by hypercomplex number representation of DNA, the hypercomplex cross-correlation might be used to scan a query DNA sequence for potential transcription binding sites. The real part of the hypercomplex cross-correlation gives a score that measures the similarity between DNA substrings and the consensus sequence. With a reasonable score threshold, this approach is able to find the binding site candidates that are undetectable by conventional string match method. When combined with partial sum plot, the hypercomplex cross-correlation is capable of not only detecting the presence of potential binding sites, but also locating the position of the best candidate from the mismatch information, making consensus sequence motif scanning as accurate as profile motif scanning.

Chapter 4

Statistical properties of regulatory DNA sequences: fat-tail test

4.1 Introduction

Identification of transcription factor binding sites and regulatory regions is a crucial step in the study of gene regulation. Regulatory regions of higher eukaryotes can be subdivided into proximal regulatory units – promoters – which are located close to 5' end of the gene, and distal transcription regulatory units called enhancers or cis-regulatory modules (CRMs). CRMs may be located far upstream or downstream of target gene, and are much more difficult to recognize because they lack proximal specific transcription signals, such as TATA box, CAAT box, transcription start site consensus, etc. Therefore, recognition of CRMs is even more difficult than recognition of promoters.

Most programs to predict CRMs, such as MCAST (Bailey and Noble, 2003) or Ahab (Rajewsky, et al., 2002) use information about known TFBSs to identify potential CRMs. Extracting clustered recognition motifs is one of the most reliable techniques for CRMs prediction, but is limited to the recognition of similarly regulated cis-regulatory regions. Thus, they cannot generally be applied to genes for which the transcription factors have not yet been studied experimentally.

Chapter 4 Fat-tail test

An alternative approach is to use information contained in the pattern of conservation between related sequences. For example the method of Grad, et al. (2004) uses patterns of sequence conservation between two species to predict CRMs with similar properties to a training set. Performance of these methods depends on the evolutionary distance between given species and the conservational level of individual genes. In addition, such an approach offers little information as to the specific function of the conserved sequences.

Another promising CRMs prediction method is based on locations of known CRMs (also known as content-based method). It is assumed that different genome regions (introns, exon, CRMs, non-coding non-regulatory region) have different statistical properties of nucleotide composition due to different rates of evolutionary micro changes. It is well known that one of distinct properties of CRMs is the presence of multiple copies of binding sites for one or several transcription factors, and that causes a biased word distribution within CRMs. This bias leaves a distinct “signature” in nucleotide composition, and content-based CRMs prediction methods tend to detect this signature by statistical or machine-learning techniques (Abnizova, et al., 2005; Chan and Kibler, 2005; Nazina and Papatsenko, 2003), thus distinguish the CRMs from non-CRMs. Content-based methods don’t need previous information about the transcription factors that regulate a target gene, thus, it might be used to predict the CRMs which have not yet been experimental observed, and serve as a valuable filtering process for experiment.

Chapter 4 Fat-tail test

The fluffy-tail test (Abnizova, et al., 2005) is one of the content-based CRMs prediction methods. It is a bootstrapping procedure to recognize statistically significant abundant similar-words in regulatory DNA. There are two problems with fluffy-tail test. The first one is fluffy-tail test looks only at the subsequence with the highest incidence in the CRMs. Therefore, fluffy-tail test may not capture the statistical features caused by heterotypic TFBS clusters in the regulatory regions. The second problem is fluffy-tail test involves large amount of calculation due to the bootstrapping procedure. This limits the use of fluffy-tail test under the situation of more and more DNA sequences need to be analyzed in the post sequencing time of genome. It is our interest to address these two problems of the fluffy-tail test and to develop more efficient and effective CRMs prediction methods.

In this chapter, statistical properties of CRMs will be explored by estimating the distribution of similar-word set occurrence. Fat-tail test with kurtosis based measurement will be developed for discriminating CRMs from non-CRMs.

4.2 Materials and methods

4.2.1 Training data set

To demonstrate the statistical properties of CRMs, three training data sets are used. The positive training set is a collection of 60 experimentally verified functional *Drosophila melanogaster* regulatory regions (Table 4.1) (Nazina and Papatsenko, 2003; Papatsenko, et al., 2002). This set consists of CRMs located far from gene coding sequences and transcription start sites. It contains many binding sites (and site

clusters), best known of which are bicoid, hunchback, kruppel, knirps and caudal, the sites involved in the regulation of developmental genes (Appendix B Table B.4 and Table B.5). The two negative training sets are (1) 60 randomly picked *Drosophila* exons, and (2) 60 randomly picked *Drosophila* non-coding, presumed non-regulatory DNA sequences (NCNRs): we excluded exons and regions of length 1Kb upstream and downstream of genes, using the Ensembl Genome Browser. All sequences in the three training data sets were tandem repeat masked with the ‘Tandem Repeats Finder’ program (Benson, 1999) with parameters ‘2 3 5 80 10 50 500’ before processing.

4.2.2 Fat-tail test

One of the most striking features of CRMs is that they contain tightly clustered binding sites for one or several transcription factors, with most of sites represented multiple times (Berman, et al., 2002; Lifanov, et al., 2003). Fat-tail test was based on the assumption that the abundance of regulatory motifs within regulatory regions leaves a distinct “signature” in nucleotide composition, and that it is possible to capture this “signature” statistically. More specifically, it is hypothesized to take the form of an over-representation of similar-words (motifs). In fat-tail test, the distributions of the number of occurrences of similar-word set for input sequences are estimated.

Notation: the number of occurrences of a word H in a sequence is denoted $O(H)$. Given a set $\mathbf{K}$ of words, one denotes the number of $\mathbf{K}$ -occurrence in a sequence $O(\mathbf{K})$

$$= \sum_{H \in \mathbf{K}} O(H).$$

Definition 1 *A j -neighbor of a word H is a word F derived from H through j substitutions. The j -neighborhood is the set of its j -neighbors and it is denoted by $\mathbf{B}_j(H)$.*

For example: let $H = \text{ACCA}$. Words CCCA , ATCA and ACCT are in $\mathbf{B}_1(H)$ while ATCC is in $\mathbf{B}_2(H)$.

By this definition, for each different m -letter word S (seed word) appeared in the input sequence, a set of similar-words corresponding to S is denoted $\mathbf{K}_S = \text{SUB}_j(S)$. Fat-tail test (Figure 4.1) estimates the distribution of the number of $\mathbf{K}_S$ -occurrence $O(\mathbf{K}_S)$ for the input sequence, and then predicts the possible function of the input sequence.

The fat-tail test includes following basic stages as shown in Figure 4.1:

Step 1: Find all the seed words for the input sequence

Because the core of transcription factor binding sites (TFBS) is relatively short (around 3-5bp) (Matys, et al., 2003), 5-letter seed words ($m=5$) are considered. In the first step of fat-tail test, the input sequence is scanned to find all the different 5-letter words, allowing overlaps. The total number of the seed words is M .

Step 2: Construct the similar-word set $\mathbf{K}_S$ for each seed word S

For each 5-letter seed word S , the corresponding similar-word set $\mathbf{K}_S = \text{SUB}_j(S)$ is constructed by allowing for 1 substitution ($j = 1$). The total number of similar-word sets is M .

Step 3: Calculate the number of $\mathbf{K}_S$ -occurrence $O(\mathbf{K}_S)$

In the third step, for the input sequence, the number of $\mathbf{K}_S$ -occurrence is calculated by $O(\mathbf{K}_S) = \sum_{H \in \mathbf{K}_S} O(H)$. O instead of $O(\mathbf{K}_S)$ will be used to represent the number of $\mathbf{K}_S$ -occurrences for the input sequence later in this chapter for simplicity. Overlapping occurrences are allowed for counting similar-words.

Step 4: Count the number n of similar-word set with same O .

The number of similar-word set with same O is counted. As an example, the histogram of the distribution of the similar-word set occurrence O is plotted in Figure 4.3. In this histogram, the X axis represents the number of occurrences of similar-word set, and the Y axis shows the number of similar-word set occurring O times.

Step 5: Calculate the kurtosis (k) and maximum ($O_{\max}$) of the distribution of similar-word set occurrence.

The formula for kurtosis is

$$k = \frac{\sum_i^M (O_i - \bar{O})^4}{(M-1)\sigma_o^4} - 3 \quad (4.1)$$

Chapter 4 Fat-tail test

M is the total number of similar-word sets, O_i is the number of occurrences of the i th similar-word set. $\bar{O}$ and σ_o are the mean and standard deviation of the similar-word set occurrence O for the given sequence.

Step 6: Calculate kurtosis coefficient

Compare the distribution of similar-word set occurrence between the input sequence and the randomized shuffled versions of the input sequence by calculating fat-tail kurtosis coefficient $F-K_r$ or $F-KK$ and then predict the possible function of the input DNA sequence.

4.2.3 Fat-tail kurtosis coefficient $F-K_r$ and $F-KK$

To measure how strong the distribution of similar-word set occurrence of CRMs deviates from randomness, a fat-tail kurtosis coefficient $F-K_r$ is introduced as:

$$F - K_r = \frac{k_{original} - \bar{k}_r}{\sigma_r} \quad (4.2)$$

where $k_{original}$ is the kurtosis of the similar-word set occurrence for original sequence, $\bar{k}_r$ and σ_r are the mean and standard deviation of the kurtosis of the similar-words set occurrence for r shuffled sequences. Here a sequence is called “random” if it is obtained from original sequence by shuffling it, preserving its single nucleotide composition.

Chapter 4 Fat-tail test

$F-K_r$ can be regarded as measuring the difference between signal and noise, where the signal is taken from the original sequence, and the noise from the randomized sequences with the same composition and length. Thus, the fat-tail kurtosis coefficient $F-K_r$ is normalized for the length and base composition of the sequence, because we compare each original sequence only with respect to shuffled sequences of the same length and composition. Thus one can compare the fat-tail kurtosis coefficient $F-K_r$ for sequences with different base composition and length.

Similar to the fat-tail kurtosis coefficient $F-K_r$, another fat-tail coefficient $F-M_r$ is introduced to measure difference between original sequence and randomized shuffled sequences:

$$F - M_r = \frac{O_{\max, \text{original}} - \overline{O}_{\max, r}}{\sigma_{\max, r}} \quad (4.3)$$

where $O_{\max, \text{original}}$ is the number of occurrences of the similar-word set with highest incidence for original sequence, $\overline{O}_{\max, r}$ and $\sigma_{\max, r}$ are the mean and standard deviation of the number of occurrences of the similar-word set with the highest incidence for r shuffled sequences.

Besides the kurtosis coefficient $F-K_r$, another kurtosis based coefficient $F-KK$ is introduced as:

$$F - KK = \frac{k_{\text{original}} + 2sek}{4sek}, \quad (4.4)$$

where sek is the standard error of kurtosis calculated by:

$$sek = \sqrt{\frac{24}{M}}, \quad (4.5)$$

M is the total number of seed words.

Fat-tail kurtosis coefficient $F-KK$ is used to measure the difference between distribution of similar-word set occurrence for an input sequence and normal distribution. $[-2sek, 2sek]$ is the 95% confidence interval (C.I.) for the kurtosis of normal distribution with the sample size of M . If the value of kurtosis of distribution of similar-word set occurrence for the input sequence is outside the 95% C.I. in positive direction, the distribution for the input sequence is statistically significant fat-tail distribution. Kurtosis coefficient $F-KK$ is the difference between k_{original} and lower limit of 95% C.I. normalized by length of 95% C.I.. $F-KK$ ranges from 0 to +1 for normal distribution with 95% confidence. If the fat-tail kurtosis $F-KK > 1$ for an input sequence, the distribution of similar-word set occurrence for the input sequence is statistically significant fat-tail distribution (Figure 4.2).

Chapter 4 Fat-tail test

Table 4.1 60 cis-regulatory modules of *Drosophila melanogaster*

No.	Comment	Length	No.	Comment	Length
1	btd	1429	31	hairy_stripe6+2	1080
2	gt	1104	32	hairy_stripe6	536
3	hb_enc	1422	33	hairy_stripe5+1	1674
4	hb_P1	1160	34	hairy_stripe1	1157
5	kni	1028	35	prd_stripe1	1069
6	kr_late	3133	36	prd_stripeP	1452
7	kr_CD1	1158	37	prd_glands	1059
8	kr_CD2	1706	38	run_stripe1+7	1611
9	kr_AD2	1193	39	run_stripe5	1336
10	otd_early	1838	40	run_stripe3+7	2404
11	otd_eye	2566	41	run_late7stripe	5068
12	sal_blastodermal_early	721	42	run_late6stripe	2935
13	sal_tracheal	1087	43	en_upstream_stripe	2401
14	tll_brain_specific	1128	44	en_stripe	900
15	tll_PD	635	45	abd_A_PREiab2	534
16	tll_AD_PD	2292	46	abd_A_iab2	1745
17	eve_late1	1288	47	dll_207	3808
18	eve_stripe3+7	511	48	dll_215	929
19	eve_stripe2	730	49	dll_208	4440
20	eve_late2	1800	50	dll_ETD2b	2075
21	eve_stripe4+6	602	51	dll_ETD6	5877
22	eve_stripe1	788	52	ems_upstream	1244
23	eve_stripe5	1011	53	ubx_PBX	1550
24	ftz_upstream	1787	54	ubx_dorsal	3401
25	ftz_neurogenic	1877	55	ubx_ventral	1736
26	ftz_zebra	739	56	ubx_zebra(like)_S2	1596
27	gsb_early	516	57	ubx_PRE_polycomb	1556
28	gsb_late	1565	58	ubx_zebra(like)_S1	1796
29	hairy_stripe3+4	476	59	ubx_BXD	1593
30	hairy_stripe7	1473	60	ubx_BX	1705

Figure 4.1 Flow chart of fat-tail test. For each different m -letter word found in an input sequence, the corresponding similar-word set is constructed by allowing 1 substitution. Then the number O of occurrences is calculated for each similar-word set. In the final stage, kurtosis of distribution of similar-word set occurrence for the input sequence and its randomized shuffled versions are calculated and compared. The comparison result is represented by fat-tail kurtosis coefficient $F-K_r$ or $F-KK$.

Figure 4.2 95% confidence interval for the kurtosis value of normal distribution with sample size of 649 is (-0.38, 0.38). If kurtosis for the input sequence with same sample size is outside the 95% C.I. in positive direction, the distribution for the input sequence is statistically significant fat-tail distribution. For normal distribution, kurtosis coefficient $F-KK$ is between 0 and 1, for fat-tail distribution, $F-KK$ is larger than 1.

4.3 Results

4.3.1 Distribution of similar-word set occurrence for CRMs

For the 60 sequences in the positive training data set of CRMs, the kurtosis k and maximum $O_{\max}$ for the distribution of similar-word set occurrence were calculated. Figure 4.3 shows the distribution of similar-word set occurrence for the input sequence of hb cis-regulatory module *Drosophila melanogaster*. In this histogram, the X axis represents the number of occurrences of similar-word set, and the Y axis shows the number of similar-word set occurring O times. The total number of similar-word set is 649 for the hb cis-regulatory module. From this plot it can be seen that most similar-word sets occur 10 to 40 times, but there are outliers: some similar-word sets occur more than 60 times and the similar-word set with the highest incidence occurs 63 times. The values of kurtosis $k = 2.7$, and $O_{\max} = 63$. The value of kurtosis (2.7) is outside the 95% C.I. (-0.38, 0.38) for the kurtosis of normal distribution with the same sample size of 649 in positive direction.

If the original sequence is characterized by the presence of an unusually high number of over-represented words, its distribution of similar-word set occurrence is expected to have long right tail in comparison to a random sequence. The long right tail will make the value of kurtosis statistic is far larger than 0.

To sample a random distribution, the original sequence was shuffled 50 times. For each randomization the distribution of similar-word set occurrence was estimated.

Chapter 4 Fat-tail test

Figure 4.4 shows a typical example of the distribution of similar-word set occurrence for one of the randomly shuffled sequences of the same (hb) cis-regulatory module as in Figure 4.3. The total number of similar-word set is 749. Compared with the distribution of the original data (Figure 4.3), the randomized sequence in Figure 4.4 lacks a long right tail. The values of kurtosis $k = -0.44$, and $O_{\max} = 44$. Compared to the distribution of original sequence, the kurtosis value is significantly smaller and outside the 95% C.I. $(-0.36, 0.36)$ for the kurtosis of normal distribution with the same sample size of 749 in negative direction. Figure 4.5 shows the difference between the distribution for original sequence and its shuffled version in cumulative form. The difference between the two curves reflects the long right tail of the original data. The difference can be seen more clearly in Figure 4.6.

Figure 4.3 Histogram of similar-word set occurrence O for the hb cis-regulatory module. The distribution of similar-word set occurrence for the hb cis-regulatory module *Drosophila melanogaster* ($k=2.70$, $O_{\max}=63$). The X axis shows the number O of occurrences of similar-word set; the Y axis is for the number of word sets occurring O times. The total number of similar-word set is 649.

Figure 4.4 Histogram of similar-word set occurrence O for the hb cis-regulatory module, after shuffling. The distribution of similar-word set occurrence for one randomly shuffled version of the hb cis-regulatory module *Drosophila melanogaster*. ($k = -0.44$, $O_{\max} = 44$). The X axis shows the number O of occurrences of similar-word set; the Y axis is for the number of word sets occurring O times. The total number of similar-word set is 749.

Figure 4.5 Cumulative histograms of similar-word set Occurrence O for the hb cis-regulatory module and its shuffled version. Cumulative histograms for figures 4.3 and 4.4: solid line: original data from Figure 4.3, dotted line: randomized data from Figure 4.4. The X axis shows the number O of occurrences of similar-word set; the Y axis is for the number of word sets occurring less than O .

Figure 4.6 Histograms of similar-word set Occurrence O for the hb cis-regulatory module and its shuffled versions. (a) The distribution of the original hb cis-regulatory module (solid line); the distribution of 10 randomized sequences (dotted lines). (b) The same distribution in cumulative form. The X axis shows the number O of occurrences of similar-word set; the Y axis is for the number of word sets.

4.3.2 Distribution of similar-word set occurrence for exons

For the 60 sequences in the training data set of randomly picked *Drosophila* exons, the kurtosis k and maximum $O_{\max}$ for the distribution of similar-word set occurrence were calculated. Figure 4.7 shows the distribution of similar-word set occurrence for the sequence of exon CG8229 *Drosophila melanogaster*. The total number of similar-word set is 662. The absence of long right tail is noted in the Figure 4.7. The values of kurtosis $k = -0.28$, and $O_{\max} = 47$. The value of kurtosis is inside the 95% C.I. $(-0.38, 0.38)$ for the kurtosis of normal distribution with the same sample size of 662.

Figure 4.8 shows a typical example of the distribution of similar-word set occurrence for one of the randomly shuffled sequences of the exon CG8229. The total number of similar-word set is 668. The value of kurtosis $k = 0.06$, within the 95% C.I. $(-0.37, 0.37)$ for the kurtosis of normal distribution with the same sample size of 668. From these two plots (Figure 4.7 and 4.8) it can be seen that the distribution of similar-word set occurrence of the original sequence does not differ significantly from the distribution of its randomized version. The difference of kurtosis between the original sequence and its shuffled version is small $(-0.28 \text{ vs } 0.06)$. Both are inside the corresponding 95% C.I. for the kurtosis of normal distribution. No significance difference can be seen between the two curves in Figure 4.9 (the cumulative histograms for the original sequence and its randomized version). Similarly no significance difference can be seen in Figure 4.10 (the histograms for the original sequence and 10 randomized sequences).

Figure 4.7 Histogram of similar-word set Occurrence O for the exon CG8229. The distribution of similar-word set occurrence for the exon CG8229 *Drosophila melanogaster* ($k = -0.28$, $O_{\max} = 47$). The X axis shows the number O of occurrences of similar-word set; the Y axis is for the number of word sets occurring O times. The total number of similar-word set is 662.

Figure 4.8 Histogram of similar-word set Occurrence O for the exon CG8229, after shuffling. The distribution of similar-word set occurrence for one randomly shuffled version of the exon CG8229 *Drosophila melanogaster* ($k = 0.06$, $O_{\max} = 46$). The X axis shows the number O of occurrences of similar-word set; the Y axis is for the number of word sets occurring O times. The total number of similar-word set is 668.

Figure 4.9 Cumulative histograms of similar-word set Occurrence O for the exon CG8229 and its shuffled version. Cumulative histograms for figures 4.7 and 4.8: solid line: original data from Figure 4.7, dotted line: randomized data from Figure 4.8. The X axis shows the occurrence number O of similar-word set; the Y axis is for the number of similar-word sets.

Figure 4.10 Histograms of similar-word set Occurrence O for the exon CG8229 and its shuffled versions. (a) The distribution of the original exon sequence (solid line); the distribution of 10 randomized sequences (dotted lines). (b) The same distribution in cumulative form. The X axis shows the number O of occurrences of similar-word set; the Y axis is for the number of word sets.

4.3.3 Distribution of similar-word set for non-coding, presumed non-regulatory DNA (NCNRs)

For the 60 sequences in the training data set of randomly picked *Drosophila* NCNRs, the kurtosis k and maximum $O_{\max}$ for the distribution of similar-word set occurrence were calculated. Figure 4.11 shows the distribution of similar-words set occurrence for an example sequence of NCNR *Drosophila melanogaster*. The total number of similar-word set is 720. The presence of short right tail is noted in the Figure 4.11. The values of kurtosis $k = 0.09$, and $O_{\max} = 50$. The value of kurtosis is inside the 95% C.I. $(-0.36, 0.36)$ for the kurtosis of normal distribution with the same sample size of 720. Figure 4.12 shows a typical example of the distribution of similar-word set occurrence for one of the randomly shuffled sequences of original sequence. The total number of similar-word set is 760. The value of kurtosis is 0.25, inside the 95% C.I. $(-0.35, 0.35)$ for the kurtosis of normal distribution with the same sample size of 760. From these two plots (Figure 4.11 and 4.12) it can be seen that the distribution of similar-word set occurrence of the original sequence does not differ significantly from the distribution of its randomized version. The difference of kurtosis between the original sequence and its shuffled version is small (0.09 vs 0.25). Both are inside the 95% C.I. for the kurtosis of normal distribution. The difference between the distribution of original sequence and its shuffled version can be seen more clearly in Figure 4.14.

Figure 4.11 Histogram of similar-word set Occurrence O for an example NCNR sequence that is randomly selected from *Drosophila melanogaster* genome. The distribution of similar-word set occurrence for the example NCNR sequence of *Drosophila melanogaster* ($k = 0.09$, $O_{\max} = 50$). The X axis shows the number O of occurrences of similar-word set; the Y axis is for the number of word sets occurring O times. The total number of similar-word set is 720.

Figure 4.12 Histogram of similar-word set Occurrence O for the example NCNR sequence, after shuffling. The distribution of similar-word set occurrence for one randomly shuffled version of the example NCNR *Drosophila melanogaster* ($k = 0.25$, $O_{\max} = 48$). The X axis shows the number O of occurrences of similar-word set; the Y axis is for the number of similar-word sets occurring O times. The total number of similar-word set is 760.

Figure 4.13 Cumulative histograms of similar-word set Occurrence O for the example NCNR sequence and its shuffled version. Cumulative histograms for figures 4.11 and 4.12: solid line: original data from Figure 4.11, dotted line: randomized data from Figure 4.12. The X axis shows the number O of occurrences of similar-word set; the Y axis is for the number of similar-word sets.

Figure 4.14 Histograms of similar-word set Occurrence O for the example NCNR sequence and its shuffled versions. (a) The distribution of the original exon sequence (solid line); the distribution of 10 randomized sequences (dotted lines). (b) The same distribution in cumulative form. The X axis shows the occurrence number O of similar-word set; the Y axis is for the number of word sets with this occurrence O .

4.3.4 Fat-tail test results for three training data sets

In order to distinguish CRMs from other types of DNA, fat-tail test coefficients $F-K_r$, $F-KK$ and $F-M_r$ were calculated for the 180 sequences in our training data sets (60 sequences in each training data set). For each sequence 50 ($r = 50$) shuffled versions were generated, in calculating fat-tail test coefficient $F-K_{50}$ and $F-M_{50}$. Figure 4.15 - 4.17 shows the distribution of kurtosis coefficient $F-K_{50}$ for regulatory, coding and non-coding presumed non-regulatory DNA. One can see that CRMs tend to have larger $F-K_{50}$ than exons and NCNRs. The value $F-K_{50} = 2$ was used as a threshold to distinguish CRMs from other types of DNA. When $F-K_{50} > 2$, the input sequence was declared to have a fat-tail distribution for the similar-word set occurrence.

Table 4.2(a) lists the results of classification of the function of DNA sequences in our training data sets based on the fat-tail kurtosis coefficient $F-K_{50}$. 46 out of 60 regulatory regions (76.7%) analyzed in our positive training data set exhibit the significant fat-tail pattern ($F-K_{50} > 2$), while only 7 out of 60 exons (11.7%) in the exon data set have $F-K_{50} > 2$, and 22 out of 60 NCNRs (36.7%) in the NCNR data set have $F-K_{50} > 2$. The separation between the CRMs and NCNRs is worse than that between the CRMs and exons. However, there is a significant difference in the mean scores between the two groups (CRMs, 6.10 ± 0.85 (mean $\pm$ standard error of the mean); NCNRs 1.68 ± 0.32 ; rank sum test $P < 1.99e-07$) (Figure 4.17 column 1 and 3).

Chapter 4 Fat-tail test

Figure 4.18 - 4.20 shows the distribution of fat-tail kurtosis coefficient $F-KK$ for regulatory, coding and non-coding non-regulatory DNA in our training data sets. One can see that regulatory DNAs tend to have larger $F-KK$ than exons and NCNRs. The value $F-KK = 2$ was used as a threshold. When $F-KK > 2$, the input sequence was declared to have a fat-tail distribution for the similar-word set occurrence.

Table 4.2(b) lists the results of classification of the function of DNA sequences in our data sets based on the fat-tail kurtosis coefficient $F-KK$. 45 out of 60 CRMs (75%) in the positive training set have $F-KK > 2$, while 11 out of 60 exons (18.7%) in the exon data set have $F-KK > 2$, and 32 out of 60 sequences (53.3%) in the NCNR data set have $F-KK > 2$. Again, the separation between CRMs and NCNRs is worse than that between CRMs and exons, but the difference in the mean scores was statistically significant between CRMs and NCNRs (5.11 ± 0.59 versus 2.70 ± 0.26 ; rank sum test $P < 0.8e-03$) (Figure 4.20 column 1 and 3).

Figure 4.21 - 4.23 shows the distribution of fat-tail coefficient $F-M_{50}$ for regulatory, coding and non-coding non-regulatory DNA in our training data sets. One can see that regulatory DNAs tend to have larger $F-M_{50}$ than exons and NCNRs. The value $F-M_{50} = 2$ was used as a threshold.

Table 4.2(c) lists the results of classification of the function of DNA sequences in our data sets based on the fat-tail coefficient $F-M_{50}$. 42 out of 60 CRMs (70%) in the positive training set have $F-M_{50} > 2$, while 8 out of 60 exons (13.3%) in the exon data

set have $F\text{-}M_{50} > 2$, and 21 out of 60 sequences (35%) in the NCNR data set have $F\text{-}M_{50} > 2$. Again, the separation between CRMs and NCNRs is worse than that between CRMs and exons, but the difference in the mean scores was statistically significant between CRMs and NCNRs (6.10 ± 0.85 versus 1.67 ± 0.32 ; rank sum test $P < 5.13 \times 10^{-8}$) (Figure 4.23 column 1 and 3).

Chapter 4 Fat-tail test

Table 4.2(a) Classification of the 180 sequences by fat-tail test with kurtosis coefficient $F-K_{50}$

Functional type	$F-K_{50} > 2$	$F-K_{50} < 2$	Positive rate	Negative rate
Regulatory regions	46	14	76.7%	23.3%
Exons	7	53	11.7%	88.3%
Non-coding presumed non-regulatory regions	22	38	36.7%	63.3%

Table 4.2(b) Classification of the 180 sequences by fat-tail test with kurtosis coefficient $F-KK$

Functional type	$F-KK > 2$	$F-KK < 2$	Positive rate	Negative rate
Regulatory regions	45	15	75%	25%
Exons	11	49	18.3%	81.7%
Non-coding presumed non-regulatory regions	32	28	53.3%	46.7%

Table 4.2(c) Classification of the 180 sequences by fat-tail test with coefficient $F-M_{50}$

Functional type	$F_{50} > 2$	$F_{50} < 2$	Positive Rate	Negative rate
Regulatory regions	42	18	70%	30%
Exons	8	52	13.3%	86.7%
Non-coding presumed non-regulatory	21	39	35%	65%

Figure 4.15 Histograms for regulatory, coding sequences classified by $F-K_{50}$. (Visualizing the result in Table 4.2(a)) The X axis shows the kurtosis coefficient $F-K_{50}$; the Y axis is the number of sequences in the data set with this $F-K_{50}$.

Figure 4.16 Histograms for regulatory, NCNR sequences classified by $F-K_{50}$. (Visualizing the result in Table 4.2(a)) The X axis shows the kurtosis coefficient $F-K_{50}$; the Y axis is the number of sequences in the data set with this $F-K_{50}$.

Figure 4.17 Box-plot of the fat-tail kurtosis coefficient $F-K_{50}$ (Y-axis) for the three functional regions. q1 and q3 are the first and third quartiles of the data (box area), min median, and max are the minimum, median and maximum of the data. Rank sum test for column 1 and 3: $P < 1.99\text{e-}07$.

Figure 4.18 Histograms for regulatory, coding sequences classified by $F-KK$ (Visualizing the result in Table 4.2(b)). The X axis shows the kurtosis coefficient $F-KK$; the Y axis is the number of sequences in the data set with this $F-KK$.

Figure 4.19 Histograms for regulatory, NCNR sequences classified by $F-KK$ (Visualizing the result in Table 4.2(b)). The X axis shows the kurtosis coefficient $F-KK$; the Y axis is the number of sequences in the data set with this $F-KK$.

Figure 4.20 Box-plot of the fat-tail kurtosis coefficient $F-KK$ (Y-axis) for the three functional regions. q1 and q3 are the first and third quartiles of the data (box area), min median, and max are the minimum, median and maximum of the data. Rank sum test for column 1 and 3: $P < 0.8e-03$.

Figure 4.21 Histograms for regulatory, coding sequences classified by $F-M_{50}$ (Visualizing the result in Table 4.2(c)). The X axis shows the kurtosis coefficient $F-M_{50}$; the Y axis is the number of sequences in the data set with this $F-M_{50}$.

Figure 4.22 Histograms for regulatory, NCNR sequences classified by $F-M_{50}$ (Visualizing the result in Table 4.2(c)). The X axis shows the kurtosis coefficient $F-M_{50}$; the Y axis is the number of sequences in the data set with this $F-M_{50}$.

Figure 4.23 Box-plot of the fat-tail kurtosis coefficient $F-M_{50}$ (Y-axis) for the three functional regions. q1 and q3 are the first and third quartiles of the data (box area), min median, and max are the minimum, median and maximum of the data. Rank sum test for column 1 and 3: $P < 5.13\text{e-}08$.

4.3.5 Comparison with Fluffy-tail test

Our CRMs prediction method - fat-tail test was evaluated by comparison with fluffy-tail test (Abnizova, et al., 2005). The fluffy-tail test is a bootstrapping procedure to recognize statistically significant abundant similar-words in regulatory DNA. The fluffiness coefficient F_r was used to predict the function of input sequence. In fact, the size of maximal similar-word list (MSWL) for calculating F_r in fluffy-tail test is the number of occurrences of the similar-word set with highest incidence in our fat-tail test, so the value of F_r is equal to the value of $F-M_r$. For comparison purpose, F_r is used instead of $F-M_r$ in this section.

The CRM and exon training data sets were employed to evaluate the performance on the separation between CRMs and exons by the three indexes (1) fat-tail kurtosis coefficient $F-K_r$, (2) fat-tail kurtosis coefficient $F-KK$, and (3) fluffiness coefficient F_r . To evaluate the prediction performance, sensitivity (SN), specificity (SP) and average successful rate were calculated:

$$SN = TP/(TP+FN) \quad (4.6)$$

$$SP = TN/(TN+FP) \quad (4.7)$$

$$\text{and} \quad \text{Average successful rate} = (TP+TN)/(TP+FP+TN+FN) \quad (4.8)$$

where TP means true positive, TN means true negative, FP means false positive and FN means false negative. The most meaningful measurement is SN, which quantifies the proportion of experimental identified CRMs predicted by the method under evaluation. The results are summarized in Table 4.3. Table 4.3 shows that fat-tail test

Chapter 4 Fat-tail test

with kurtosis coefficient $F-K_{50}$ has the highest sensitivity, specificity and average successful rate, compared to the other two indexes. The second highest sensitivity is achieved by $F-KK$. Fluffy-tail test with F_{50} has the least SN but the same average successful rate with $F-KK$ due to slightly higher SP than that of $F-KK$.

CRMs and NCNR training data sets were employed to evaluate the performance of the separation between CRMs and NCNRs by the three indexes. The results are summarized in Table 4.4. Table 4.4 shows that fat-tail test with kurtosis coefficient $F-K_{50}$ has the highest sensitivity and average successful rate, compared to the other two indexes. The second highest sensitivity is achieved by $F-KK$. Fluffy-tail test with F_{50} has the least SN but the highest SP. When the performances were evaluated for discriminating between CRMs and NCNRs one problem has to be considered: some numbers of the random NCNRs are in fact currently unidentified CRMs. Thus false positive predictions are expected and high specificity may not necessarily reflect good accuracy of a test. The most meaningful measurement is the sensitivity. So fat-tail test with kurtosis coefficient $F-K_{50}$ or $F-KK$ has better separation performance than fluffy-tail test for separation between CRMs and NCNRs.

Chapter 4 Fat-tail test

Table 4.3 Evaluation of fat-tail kurtosis coefficient $F\text{-}K_{50}$, fat-tail kurtosis coefficient $F\text{-}KK$, and fluffiness coefficient F_{50} on distinguishing CRMs from exons.

	Fat-tail test		Fluffy-tail test
	$F\text{-}K_{50}$	$F\text{-}KK$	F_{50}
SN	76.7%	75%	70%
SP	88.3%	81.7%	86.7%
Averaged successful rate	82.5%	78.3%	78.3%

The three indexes were assessed by comparing their performance on the separation between CRMs and exons. NCNR were disregarded. SN = sensitivity; SP = specificity. The threshold for $F\text{-}K_{50}$, $F\text{-}KK$ and F_{50} are all set as 2. i.e. for fat-tail test, we considered input DNA sequence with $F\text{-}K_{50}>2$ as predicted CRMs, and other DNAs as predicted exons, and $F\text{-}KK >2$ as predicted CRMs, $F\text{-}KK < 2$ as predicted exons; for fluffy-tail test, we considered input DNA sequence with $F_{50} > 2$ as CRMs and other DNAs as predicted exons.

Table 4.4 Evaluation of fat-tail kurtosis coefficient $F\text{-}K_{50}$, fat-tail kurtosis coefficient $F\text{-}KK$, and fluffy-tail coefficient F_{50} , on distinguishing CRMs from NCNRs.

	Fat-tail test		Fluffy-tail test
	$F\text{-}K_{50}$	$F\text{-}KK$	F_{50}
SN	76.6%	75%	70%
SP	63.3%	46.7%	65%
Averaged successful rate	70%	60.8%	67.5%

The three indexes were assessed by comparing their performance on the separation between CRMs and NCNRs. Exons were disregarded. SN = sensitivity; SP = specificity. The threshold for $F\text{-}K_{50}$, $F\text{-}KK$ and F_{50} are all set as 2, i.e. for fat-tail test, input DNA sequences with $F\text{-}K_{50} > 2$ are declared as predicted CRMs, and other DNAs as predicted NCNRs, and $F\text{-}KK > 2$ as predicted CRMs, $F\text{-}KK < 2$ as predicted NCNRs; for fluffy-tail test, input DNA sequences with $F_{50} > 2$ are declared as CRMs and other DNAs as predicted NCNRs.

4.3.6 Time complexity

Fat-tail kurtosis coefficient $F-K_r$ is calculated by bootstrapping procedure, the value is affected by the number of randomization r . Table 4.5 shows that $F-K_r$ is somewhat unstable for different r . In order to get more reliable estimates for $F-K_r$, larger r is needed, however it will make the computational time unreasonable. For reliable result and reasonable computational time the original sequence was resample 50 times for calculating the kurtosis coefficient $F-K_r$.

Step 2 of fat-tail test takes time linear to the length of input sequence. The time complexity of calculating $F-KK$ is $O(N)$, where N is the length of input sequence. For calculating $F-K_r$ and F_r , step 2 of the fat-tail test is executed r times, thus, the time complexity is $O(Nr)$. Table 4.6 shows the run-time of fat-tail test for the input sequence of length 1000. The run-time for calculating $F-KK$ for the sequence of length 1000 is 6.42 second, it is 50 times faster than that for calculating $F-K_{50}$. All experiments were run on a 3.2 GHZ Pentium IV processor with 1G physical memory.

Table 4.5 Sensitivity of $F-K_r$ to choice of r , the number of randomization, for the hb cis-regulatory region. (the value of kurtosis for original sequence $k_{\text{original}} = 2.70$)

r	$F-K_r$	$\bar{k}_r$	σ_r
50	3.63	0.26	0.67
100	5.31	0.20	0.47
500	4.28	0.20	0.58

r is the number of randomization, $F-K_r$ is the fat-tail kurtosis coefficient , $\bar{k}_r$, σ_r are mean and standard deviation of the kurtosis of the similar-words set occurrence for r shuffled sequences.

Chapter 4 Fat-tail test

Table 4.6 Run-time for calculating fat-tail test coefficient $F-K_{50}$, $F-KK$, and F_{50} (the length of input sequence is 1000).

	Fat-tail test		Fluffy-tail test
	$F-K_{50}$	$F-KK$	F_{50}
Avg. run-time	310 sec	6.2 sec	310 sec

4.4 Discussion

One of the most striking features of CRMs is the redundancy of binding sites for a given transcription factor. The fat-tail test, proposed in this chapter, is designed to capture this feature statistically. Incorporated with the fat-tail test, kurtosis based coefficients $F-K_r$ and $F-KK$ are proposed to discriminate between CRMs and other types of DNA.

4.4.1 Comparison between $F-K_r$ and $F-KK$

Appendix B Table B.1 shows the values of $F-K_{50}$ and $F-KK$ for the 180 sequences in our training data sets. It can be seen that for most CRMs with $F-K_{50} > 2$, its corresponding $F-KK$ is larger than 2, only 3 out of 60 (5%) CRMs with $F-K_{50} > 2$, the corresponding $F-KK$ is smaller than 2. The reason for the smaller value of $F-KK$ is the distribution of similar-word set occurrence for random shuffled original sequence deviates from normal distribution due to severely bias of nucleotide composition. Thus, fat-tail kurtosis coefficient $F-KK$ is recommended to be used instead of $F-K_r$ only under the balanced nucleotide composition.

4.4.2 Codon usage bias and fat-tail kurtosis coefficient

The phenomenon of codon usage bias is well known in coding region of gene. To see if the codon usage bias has some impacts on the value of fat tail kurtosis coefficient, 60 CDS of *Drosophila melanogaster* with higher codon usage bias (Table B.2) were used to test the correlation between codon usage bias and fat tail kurtosis coefficient. Codon Adaptation Index (CAI) (Carbone, et al., 2003) was used to measure codon

usage bias. The values of CAI for these 60 CDS ranges from 0.585 to 0.813. Higher CAI means higher expression level of corresponding genes. The values of kurtosis k , the fat tail kurtosis coefficient $F-K_{50}$, and $F-KK$ for each sequence were listed in Table B.3 and shown in Figure B.1 and Figure B.2. Results show that few sequences have $F-K_{50}$, or $F-KK$ larger than 1. That means most of CDS with higher codon usage bias don't have fat tail distribution for the similar-word-set occurrence. The correlation between $F-K_{50}$, $F-KK$ and CAI are -0.2 and -0.02 respectively. These results show that the impact of codon usage bias on kurtosis coefficient of coding region is too small to make $F-K_r$ or $F-KK$ larger than the threshold of 2, and there is not obvious correlation between the kurtosis value and the expression level of gene.

4.4.3 Application to the REDfly test set

The application of fat-tail test was extended to REDfly database, which contain 628 experimentally verified CRMs from *Drosophila* (Gallo, et al., 2006). Results show that 59.5% CRMs in the database have $F-K_{50} > 2$, and 63.1% CRMs have $F-KK > 2$, passing the fat-tail test. The low pass rate may be due to the stringent threshold value. Another possible reason is that some CRMs don't contain binding sites cluster. This directs our future study: (1) check if the binding site clustering is the common feature of all CRMs; (2) optimize the threshold to get more reliable results.

4.4.4 Correlation between the distribution of similar-word set occurrence and experimentally verified TFBS

To test whether fat-tail distributions of similar-word set occurrence for CRMs are related to TFBSs, we downloaded all annotated TFBSs that are found in our training

data set of CRMs (Appendix B Table B.4 and Table B.5). The number of occurrences of TFBS related similar-word sets are listed in Appendix B Table B.6. It is noted that most TFBS related similar-word sets have incidence outside the range between the first and the third quartile. The fluffy-tail test looks only at the subsequent with the highest incidence in the CRM and thus provides a measure of homotypic TFBS clustering. Fat-tail test, on the other hand, not only considers the similar-word set with highest incidence, also considers the similar-word set whose incidence is near to the highest incidence by using the kurtosis based coefficient. Fat-tail test might be, therefore, used to measure homotypic clustering and heterotypic clustering simultaneously.

4.4.5 Tandem repeat region and fat-tail test

It is well known that along with multiple TFBS, regulatory regions often contain other statistically significant patterns such as long simple repeats (... CACACA...) or poly(N) tracts (...TTTT...). These tandem repeat regions make the separation between CRMs and NCNRs is difficult because NCNRs also contain many repetitive elements. Both fat-tail test and fluffy-tail test are sensitive to the tandem repeat elements in the given sequence. Table 4.7(a-c) list the separation results of the 180 sequences in our training data sets by fat-tail test and fluffy-tail test, before tandem repeat masking. Table 4.8 and 4.9 list the corresponding values of separation performance. In comparison with Table 4.3 and 4.4, one can see that after-repeat masking, the average successful rates for separation between CRMs and non-CRMs is

increased for both fat-tail test and fluffy-tail test. Thus, we suggest removing the repetitive elements from the given sequence before fat-tail test.

All sequences in the three training data sets were tandem repeat masked with the ‘Tandem Repeats Finder’ program (Benson, 1999) with parameters ‘2 3 5 80 10 50 500’ before processing. In order to avoid the ad hoc parameter setting for repeat-masking, special statistics accounting for word overlaps are used to solve the problem of tandem repeat elements in chapter 5.

Chapter 4 Fat-tail test

Table 4.7(a) Classification of the 180 sequences by fat-tail test with kurtosis coefficient $F\text{-}K_{50}$, before repeat-masking.

Functional type	$F\text{-}K_{50} > 2$	$F\text{-}K_{50} < 2$	Positive rate	Negative rate
Regulatory regions	44	16	73.3%	26.7%
Exons	13	47	21.6%	78.3%
Non-coding presumed non-regulatory regions	29	31	48.3%	51.7%

Table 4.7(b) Classification of the 180 sequences by fat-tail test with kurtosis coefficient $F\text{-}KK$, before repeat-masking.

Functional type	$F\text{-}KK > 2$	$F\text{-}KK < 2$	Positive rate	Negative rate
Regulatory regions	46	14	76.4%	23.3%
Exons	19	41	31.7%	68.3%
Non-coding presumed non-regulatory regions	39	21	65%	35%

Table 4.7(c) Classification of the 180 sequences by fluffy-tail test with coefficient F_{50} , before repeat-masking.

Functional type	$F_{50} > 2$	$F_{50} < 2$	Positive Rate	Negative rate
Regulatory regions	49	11	81.7%	18.3%
Exons	19	41	31.7%	68.3%
Non-coding presumed non-regulatory	31	29	51.7%	48.3%

Table 4.8 Evaluation of fat-tail kurtosis coefficient $F\text{-}K_{50}$, fat-tail kurtosis coefficient $F\text{-}KK$, and fluffiness coefficient F_{50} on distinguishing CRMs from exons, before and after repeat-masking.

	Fat-tail test				Fluffy-tail test	
	$F\text{-}K_{50}$ (before repeat- masking)	$F\text{-}K_{50}$ (after repeat- masking)	$F\text{-}KK$ (before repeat- masking)	$F\text{-}KK$ (after repeat- masking)	F_{50} (before repeat- masking)	F_{50} (after repeat- masking)
SN	73.3%	76.7%	76.6%	75%	81.7%	70%
SP	80%	88.3%	68.3%	81.7%	68.3%	86.7%
Average successful rate	76.6%	82.5%	72.5%	78.3%	72.5%	78.3%

Table 4.9 Evaluation of fat-tail kurtosis coefficient $F\text{-}K_{50}$, fat-tail kurtosis coefficient $F\text{-}KK$, and fluffy-tail coefficient F_{50} , on distinguishing CRMs from NCNRs, before and after repeat-masking.

	Fat-tail test				Fluffy-tail test	
	$F\text{-}K_{50}$ (before repeat- masking)	$F\text{-}K_{50}$ (after repeat- masking)	$F\text{-}KK$ (before repeat- masking)	$F\text{-}KK$ (after repeat- masking)	F_{50} (before repeat- masking)	F_{50} (after repeat- masking)
SN	73.3%	76.6%	76.6%	75%	81.7%	70%
SP	50%	63.3%	35%	46.7%	48.3%	65%
Average successful rate	61.6%	70%	55.8%	60.8%	65%	67.5%

4.4.6 Remarks

Transcriptional regulation is a highly complex process that differs for each individual gene and CRM. Therefore, it is difficult to find a necessary and sufficient set of rules that can be used for computational prediction of all CRMs in a genome. For fat-tail test, it is difficult to set the value of threshold to discriminate all the CRMs from non-CRMs. To solve this problem, fat-tail score could be introduced. For each input sequence, random non-coding sequences of the same length are chosen from *D. melanogaster* genome, the fat-tail coefficient is computed for each of these, and an empirical p-value computed as the fraction random sequences that scored greater than or equal to the given sequence's fat-tail coefficient. This empirical p-value is called the "fat-tail score". A sequence is classified as a CRM if its fat-tail scores (empirical p-value) is below a threshold τ .

Given a set of true CRMs, the sensitivity is simply the fraction of these classified as CRMs. The specificity of prediction is calculated by making assumptions about the fraction of randomly selected non coding sequences that are actually CRMs, as follows. Let S be an input sequence, let T_s be the event that S is classified as being a CRMs (because its fat tail test score was below threshold), let G_s be the event that S is a true CRM, and let B_s be the event that it is not a true CRM. The specificity is defined as the probability that a sequence S classified as a CRM is indeed a true CRM. That is, the specificity is given by

$$\Pr(G_s|T_s) = \frac{\Pr(T_s|G_s) \Pr(G_s)}{\Pr(T_s|G_s) \Pr(G_s) + \Pr(T_s|B_s) \Pr(B_s)}$$

Note that $\Pr(T_s|B_s) = \tau$, since the probability that a random (non-CRM) sequence has an empirical p-value below threshold τ is itself τ . $\Pr(T_s|G_s)$ is computed from the true (known) CRMs as the fraction of them that were classified as CRMs. Under the new definition for the performance measurement, we can make varying assumptions about $\Pr(G_s)$, the prior probability that a sequence S is a CRM, and set $\Pr(B_s) = 1 - \Pr(G_s)$ to obtain the specificity at threshold τ .

According to above definition, sensitivity of the prediction is based on the p-value score used as threshold for calling a sequence a CRM, while the specificity of prediction depends on the true proportion of CRMs in the genome. By introducing p-value, therefore, we have greater control over the prediction result of fat-tail test. Parameter optimization will be an important development in the future.

Annotation of genomic sequences is the potential use of fat-tail test. For prediction of unknown CRMs, the fat-tail test may be run on every window of length 1000 (with shifts of 100) in the given sequence, with motif length of 5 and up to 1 mismatch allowed; a sequence is classified as a CRM if its fat-tail score is below a threshold τ . Usually, if the computational predicted region overlaps with the experimental annotated CRMs above 50 bp, the predicted region could be thought as a hit.

4.5 Conclusion

Statistical properties of CRMs are explored by estimating the distribution of similar-word set occurrence for 180 DNA sequences from *D. melanogaster* (60 for regulatory DNA, 60 for exon and 60 for NCNRs). Results show that CRMs tend to have fat-tail distribution for similar-word set occurrence, compared to exons and NCNRs. Based on this observation, fat-tail test with two kurtosis based indexes $F-K_r$ and $F-KK$ has been proposed to discriminate CRMs from non-CRMs. The efficacy and sensitivity of the proposed methods are evaluated through detailed study of *Drosophila melanogaster* regulatory regions, and results show the proposed methods perform better than the exiting predicting methods. By using $F-K_r$, the fat-tail test is more sensitive for distinguishing CRMs from non-CRMs than fluffy-tail test; by using $F-KK$, the fat-tail test has comparable separation success rate but 50 times faster than the fluffy-tail test.

The fat-tail test is based on the assumption that CRMs contain tightly clustered binding sites for one or several transcription factors, with most of sites represented multiple times and the redundancy of binding sites within CRMs causes the bias base composition and leaves distinct “signatures” in the distribution of similar-word set occurrence. Each similar-word set in the fat-tail test corresponds to a motif, thus, the distribution of similar-word set occurrence is a surrogate for measuring the distribution of motif occurrence for an input sequence. By using the kurtosis based coefficient, fat-tail test might be used to measure homotypic clustering and heterotypic clustering simultaneously.

Chapter 5

Statistical properties of regulatory DNA sequences: thin-tail test

5.1 Introduction

Fat-tail test and fluffy-tail test can distinguish CRMs from exons effectively, but the performance of separation between CRMs and NCNRs are far from satisfactory. One of the reasons for the low performance is that both CRMs and NCNRs contain repetitive elements such as long simple repeats (... CACACA...) or poly(N) tracts (...TTTT...). These strings are biologically significant but, on the whole, are less interesting than over-represented string with more balanced AT/GC ratio. In chapter 4, these repetitive elements are removed by the ‘Tandem Repeats Finder’ program (Benson, 1999) with ad hoc parameters ‘2 3 5 80 10 50 500’ before training for fat-tail test. In order to avoid ad hoc parameter setting for repeat-masking, special statistics accounting for word overlaps is important when estimating the distribution of similar-word set. In this chapter, the statistical properties of CRMs will be explored by evaluating the overrepresentation of similar-word set. *Z-score* will be used as the measure of the overrepresentation of the similar-word set.

5.2 Materials and methods

The same training data sets used in chapter 4 were employed here to estimate the statistical properties of CRMs. The difference is that repetitive elements are retained.

5.2.1 Thin-tail test

Thin-tail test is mainly used to distinguish CRMs from NCNRs. The procedure is similar to the fat-tail test proposed in chapter 4, the difference is, instead of the number of occurrences, *Z-score* is used to measure the overrepresentation of similar-word set in thin-tail test. The thin-tail test (Figure 5.1) checks the kurtosis value of the distribution of similar-word set *Z-score* for a given sequence and then predicts the possible function. The test features special statistics accounting for word overlaps.

The thin-tail test includes following basic stages as shown in Figure 5.1:

Step 1: Find all the seed words for the input sequence

In the first step of thin-tail test, the input sequence is scanned to find all the different 5-letter words, allowing overlaps. The total number of the seed words is M .

Step 2: Construct the similar-word set K_S for each seed word S

For each 5-letter seed word S , the corresponding similar-word set K_S ($K_S = \text{SUB}_j(S)$) is constructed by allowing for 1 substitution ($j = 1$). The total number of similar-word sets is M .

Step 3: Calculate the number of K_S -occurrence $O(K_S)$

In the third step, for the given sequence, the number of K_S -occurrence is calculated by

$O(K_S) = \sum_{H \in K_S} O(H)$. Overlapping occurrences are allowed for counting similar-words.

Step4: Statistical evaluation of overrepresentation of similar-word set

For each similar-word set, calculate the expectation $E(O(K_S))$ and the variance $V(O(K_S))$ for the number of K_S -occurrences in a random sequence with the same composition as the given sequence. Then, calculate *Z-score* Z for each similar-word set according to Equation 5.2 (See sec. 5.2.2 for further detail). O instead of $O(K_S)$, E instead of $E(O(K_S))$, and V instead of $V(O(K_S))$ are used later in this chapter for simplicity.

Step 5: Count the number n of similar-word set with same Z .

The number of similar-word set with same Z is counted. As an example, the histogram of the distribution of the similar-word set *Z-score* is plotted in Figure 5.2. In this histogram, the X axis represents the value of *Z-score* of similar-word set, and the Y axis shows the number of similar-word set with this *Z-score* Z .

Step 6: Calculate the kurtosis (k) of the distribution of similar-word set *Z-score*.

The formula for kurtosis is

$$k = \frac{\sum_i^M (Z_i - \bar{Z})^4}{(M-1)\sigma_Z^4} - 3 \quad (5.1)$$

M is the total number of similar-word set, Z_i is the value of Z -score of the i th similar-word set. $\bar{Z}$ and σ_Z are the mean and standard deviation of the similar-word set Z -score Z for the given sequence.

Step 7: Calculate kurtosis coefficient

Compare the distribution of similar-word set Z -score between the given sequence and the randomized shuffled versions of given sequence by calculating thin-tail kurtosis coefficient $T-K_r$ or $T-KK$ and then predict the possible function of the given DNA sequence.

Figure 5.1 Flow chart of thin-tail test. For each different m-letter word found in the given sequence, the overrepresentation of corresponding similar-word set is statistically evaluated through *Z-score*. Then the kurtosis of *Z-score* distribution is calculated and compared with that for the shuffled versions of given sequence. The comparison result is represented by kurtosis coefficient $T-K_r$ or $T-KK$.

5.2.2 *Z-score* – statistical evaluation of overrepresentation for similar-word set

A similar-word set that occurs significantly more often than chance expectation is said to be overrepresentation. A reasonable overrepresentation measure would reflect whether the actual number of occurrences O of the similar-word set is significantly greater than the number of it counted in a random sequence of the same length and composition. We used as this measure *Z-score* of O , a standard measure of statistical significance:

$$Z = \frac{O - E}{\sqrt{V}}, \quad (5.2)$$

where E is the expected number of occurrences of the similar-word set in a random sequence, and V is its variations (Leung, et al., 1996). It was found that both the variance V and the expectation E for the number of occurrences of a set of words can be *exactly* calculated analytically in a random Bernoulli type sequence using the generation function technique (Régner, 2000).

Notation: Let $\mathbf{K}_s$ be a set of m -letter words ω , $m=|\omega|$. Let R be an N -letter random sequence over the four-letter DNA alphabet with the vector of letter probabilities $\mathbf{p}(p_A, p_T, p_G, p_C)$. Then for each given word ω , its occurrence probability will be:

$$P(\omega) = \prod_{i=1}^m p_{\omega_i}, \quad (5.3)$$

Chapter 5 Thin-tail test

where ω_i is the i th character of the word ω . For the entire similar-word set $\mathbf{K}_s$, the occurrence probability can be calculated by:

$$P(\mathbf{K}_s) = \sum_{\omega \in \mathbf{K}_s} P(\omega) \quad (5.4)$$

It was demonstrated (Régner, 2000) that the expectation of the number of occurrences of a set of words $\mathbf{K}_s$ in a random sequence of length L generated by a Bernoulli model, is:

$$E(O) = (N - m + 1) \sum_{\omega \in \mathbf{K}_s} p(\omega). \quad (5.5)$$

And the variance is

$$Var(O) = c_1 N + c_2, \quad (5.6)$$

where

$$c_1 = \sum_{\omega \in \mathbf{K}_s} p(\omega) + (1 - 2m) \left(\sum_{\omega \in \mathbf{K}_s} p(\omega) \right)^2 + 2 \sum_{\omega \in \mathbf{K}_s} p(\omega) \left(\sum_{f \in \mathbf{K}_s} A_{\omega, f} - 1 \right) \quad (5.7)$$

and

$$c_2 = O(m/N), \quad (5.8)$$

In Equation 5.7, $A_{\omega, f}$ is called associated probability between string ω and f . Its value depends on the structure of the words in the set $\mathbf{K}_s$. Given two strings H and F , the overlapping set is defined as the set of H -suffixes that are F prefixes. Associated F suffixes form the correlation set $\mathcal{A}_{H, F}$. Each word in $\mathcal{A}_{H, F}$, when concatenated to the right of H creates an occurrence of F . For example, let $H = \text{'AACAA'}$, and $F = \text{'AAAT'}$, then $\mathcal{A}_{H, F} = \{\text{'AT'}, \text{'AAT'}\}$ and $\mathcal{A}_{F, H}$ is the empty set. The autocorrelation

set $A_{H,H} = \{\varepsilon, CAA, ACAA\}$ contains the empty string ε . Associated probability $A_{H,F}$ is calculated by:

$$A_{H,F} = \sum_{\omega \in A_{H,F}} p(\omega). \quad (5.9)$$

For example, associated probability $A_{AACAA, AAAT} = 0.3^2 + 0.3^3$, in a biased Bernoulli model where $(p_A, p_T, p_G, p_C) = (0.3, 0.3, 0.2, 0.2)$.

The variance for a word set with many overlaps between its words is larger than that for a word set lacking overlaps between its words, thus makes the corresponding *Z-score* smaller. For example, the *Z-score* of similar-word set derived from simple repeat strings, such as TTTTT or AAAAA is smaller than that of the similar-word set derived from the seed word with more balanced composition.

5.2.4 Thin-tail kurtosis coefficient $T-K_r$ and $T-KK$

To measure how strong the similar-word set *Z-score* distribution of CRMs deviates from randomness, a thin-tail kurtosis coefficient $T-K_r$ was introduced as:

$$T-K_r = \frac{k_{original} - \bar{k}_r}{\sigma_r}, \quad (5.10)$$

where $k_{original}$ is the kurtosis of similar-word set *Z-score* distribution for original sequence, $\bar{k}_r$ and σ_r are the mean and standard deviation of the kurtosis of the distribution for r shuffled sequences. Here we call the sequence “random” if it is

Chapter 5 Thin-tail test

obtained from original sequence by shuffling it, preserving its single nucleotide composition.

Similar to fat-tail test, another thin-tail kurtosis coefficient $T-KK$ was introduced as:

$$T-KK = \frac{k + 2sek}{4sek}, \quad (5.11)$$

where sek is the standard error of kurtosis calculated by:

$$sek = \sqrt{\frac{24}{M}}, \quad (5.12)$$

where M is the total number of similar-word set.

Thin-tail kurtosis coefficient $T-KK$ is used to measure how strong the distribution of similar-word set Z -score for a given sequence deviates from normal distribution. $[-2sek, 2sek]$ is the 95% confidence interval (C.I.) for the kurtosis of normal distribution with the sample size of M . If the value of kurtosis of the distribution of similar-word set occurrence for the input sequence is outside the 95% C.I. in negative direction, the distribution of the input sequence is statistically significant thin-tail distribution. Kurtosis coefficient $F-KK$ is the difference between k_{original} and the lower limit of 95% confidence interval normalized by the length of 95% confidence interval. $F-KK$ ranges from 0 to +1 for normal distribution with 95% confidence. If the thin-tail kurtosis $F-KK$ is lower than 0 for a given sequence, the distribution of similar-word set occurrence for the given sequence is statistically significant thin-tail distribution.

5.3 Result

5.3.1 Distribution of similar-word set *Z-score* for CRMs

For the 60 sequences in the positive training data set of CRMs, the kurtosis k for the distribution of similar-word set *Z-score* were calculated. Figure 5.2 shows the distribution of similar-word set *Z-score* for the input sequence of hb cis-regulatory module *Drosophila melanogaster*. The total number of similar word set is 691. In this histogram, the X axis represents the *Z-score* of similar-word set, and the Y axis shows the number of similar-word sets having this *Z-score*. From this plot it can be seen that compared to normal distribution, some similar-word set have extreme positive/negative *Z-score* ($Z > 3$ or $Z < -3$). That means some similar-word sets are overrepresented or underrepresented. The value of kurtosis k is -0.3. It is inside the 95% C.I. (-0.38, +0.38) for the kurtosis of normal distribution with the same sample size of 691.

To sample a random distribution, the original sequence was shuffled 50 times. For each randomization the distribution of the similar-word set *Z-score* was estimated. Figure 5.3 shows a typical example of the distribution of similar-word set *Z-score* for one of the randomly shuffled sequences of the same (hb) cis-regulatory module as in Figure 5.2. The total number of similar-word set is 710. Compared with the distribution of the original sequence (Figure 5.2), the randomized sequence in Figure

Chapter 5 Thin-tail test

5.3 lacks overrepresented/underrepresented similar-word set ($Z > 3$ or $Z < -3$). The value of kurtosis k is -0.14. The value of kurtosis (-0.14) is inside the 95% C.I. (-0.37, +0.37) for the kurtosis of normal distribution with the same sample size of 710. Figure 5.5 (a) shows the difference between the distribution of original sequence and those of 10 randomized sequences, the distributions for the randomized sequences (dotted lines) are thinner in the centre of the figure compared to the distribution for the original sequence. Figure 5.4 shows the difference between the distribution of the original sequence and that of the randomized sequence in cumulative form.

Figure 5.2 Histogram of similar-word set *Z-score* for the hb cis-regulatory module. The distribution of similar-word set *Z-score* for the hb cis-regulatory module *Drosophila melanogaster* ($k = -0.3$). The X axis shows the *Z-score* of similar-word set; the Y axis is for the number of similar-word sets with this *Z-score*.

Figure 5.3 Histogram of similar-word set *Z-score* for the hb cis-regulatory module, after shuffling. The distribution of similar-word set *Z-score* for one randomly shuffled version of the hb cis-regulatory module *Drosophila melanogaster* ($k = -0.14$). The X axis shows the *Z-score* of similar-word set; the Y axis is for the number of similar-word sets with this *Z-score*.

Figure 5.4 Cumulative histograms of similar-word set *Z-score* for the hb cis-regulatory module and its shuffled version. Cumulative histograms for figures 5.2 and 5.3: solid line: original data from Figure 5.2, dotted line: randomized data from Figure 5.3. The X axis shows the *Z-score* of similar-word set; the Y axis is for the number of similar-word sets.

Figure 5.5 Histograms of similar-word set Z -score for the hb cis-regulatory module and its shuffled versions. (a) The distribution of the original hb enhancer sequence (solid line); the distribution of 10 randomized sequences (dotted lines). (b) The same distribution in cumulative form. The X axis shows the Z -score of similar-word set; the Y axis is for the number of similar-word sets.

5.3.2 Distribution of similar-word set *Z-score* for exons

For the 60 sequences in the training data set of randomly picked *Drosophila* exons, the kurtosis for the distribution of similar-word set *Z-score* were calculated. Figure 5.6 shows the distribution of similar-words set *Z-score* for the sequence of exon CG12653 *Drosophila melanogaster*. The total number of similar word set is 345. It is noted that the presence of long right tail in the Figure 5.6. The value of kurtosis (3.49) is outside the 95% C.I. (-0.53, 0.53) for the kurtosis of normal distribution with the same sample size of 345 in positive direction. Figure 5.7 shows a typical example of the distribution of similar-word set *Z-score* for one of the randomly shuffled sequences of the exon CG12653. The total number of similar word set is 461. The kurtosis value is 0.16. It is inside the 95% C.I. (-0.46, 0.46) for the kurtosis of normal distribution with the same sample size of 461. From these two plots (Figure 5.6 and 5.7) it can be seen that the distribution of similar-word set *Z-score* for original exon sequence differ significantly from that of their randomized versions. The difference can be seen by the two curves in Figure 5.8, and more clearly in Figure 5.9.

Figure 5.6 Histogram of similar-word set *Z-score* for the exon CG12653. The distribution of similar-word set *Z-score* for the exon CG12653 *Drosophila melanogaster* (kurtosis = 3.49). The X axis shows the *Z-score* of similar-word set; the Y axis is for the number of word sets this *Z-score*.

Figure 5.7 Histogram of similar-word set *Z-score* for the exon CG12653, after shuffling. The distribution of similar-word set *Z-score* for one randomly shuffled version of the exon CG12653 *Drosophila melanogaster* (kurtosis = 0.16). The X axis shows the *Z-score* of similar-word set; the Y axis is for the number of word sets this *Z-score*.

Figure 5.8 Cumulative histograms of similar-word set *Z-score* for the exon CG12653 and its shuffled version. Cumulative histograms for figures 5.6 and 5.7: solid line: original data from Figure 5.6, dotted line: randomized data from Figure 5.7. The X axis shows the *Z-score* of similar-word set; the Y axis is for the number of similar-word sets.

Figure 5.9 Histograms of similar-word set Z -score for the exon CG12653 and its shuffled versions. (a) The distribution of the original exon sequence (solid line); the distribution of 10 randomized sequences (dotted lines). (b) The same distribution in cumulative form. The X axis shows the Z -score of similar-word set; the Y axis is for the number of similar-word sets.

5.3.3 Distribution of similar-word set *Z-score* for non-coding, presumed non-regulatory DNA (NCNRs)

For the 60 sequences in the training data set of randomly picked *Drosophila* NCNRs, the kurtosis for the distribution of similar-word set *Z-score* were calculated. Figure 5.10 shows the distribution of similar-word set *Z-score* for an example sequence of NCNR *Drosophila melanogaster*. The total number of similar-word set is 730. The presence of short right tail is noted in the Figure 4.11. The value of kurtosis ($k = 0.54$) is inside the 95% C.I. $(-0.36, 0.36)$ for the kurtosis of normal distribution with the same sample size of 730. Figure 5.11 shows a typical example of the distribution of similar-word set *Z-score* for one of the randomly shuffled sequences of the example NCNR sequence. The total number of similar-word set is 791. The value of kurtosis ($k=0.15$) is inside the 95% C.I. $(-0.35, 0.35)$ for the kurtosis of normal distribution with the same sample size of 791. The difference between the distributions of original sequence and its shuffled version can be seen by the two curves in Figure 5.12, and more clearly in Figure 5.13. The difference between the distribution of original sequence and shuffled sequence is smaller than that of the exons and larger than that of the CRMs.

Figure 5.10 Histogram of similar-word set *Z-score* for an example NCNR sequence that is randomly selected from *Drosophila melanogaster*. The distribution of similar-word set *Z-score* for the example NCNR sequence *Drosophila melanogaster* ($k = 0.54$). The X axis shows the *Z-score* of similar-word set; the Y axis is for the number of similar-word sets having this *Z-score*.

Figure 5.11 Histogram of similar-word set Z -score for the example NCNR sequence, after shuffling. The distribution of similar-word set Z -score for one randomly shuffled version of the example NCNR sequence *Drosophila melanogaster* (kurtosis $k = 0.15$). The X axis shows the Z -score of similar-word set; the Y axis is for the number of similar-word sets having this Z -score.

Figure 5.12 Cumulative histograms of similar-word set Z -score for the example NCNR sequence and its shuffled version. Cumulative histograms for figures 5.10 and 5.11: solid line: original data from Figure 5.10, dotted line: randomized data from Figure 5.11. The X axis shows the Z -score of similar-word set; the Y axis is for the number of similar-word sets.

Figure 5.13 Histograms of similar-word set Z -score for the example NCNR sequence and its shuffled versions. (a) The distribution of the original exon sequence (solid line); the distribution of 10 randomized sequences (dotted lines). (b) The same distribution in cumulative form. The X axis shows the Z -score of similar-word set; the Y axis is for the number of similar-word sets.

5.3.4Thin-tail test results for the three training data sets

In order to distinguish the CRMs from other types of DNA, thin-tail kurtosis coefficients $T-K_r$ and $T-KK$ were calculated for the 180 sequences in our training data sets (60 sequences in each training data set). For each sequence 50 ($r = 50$) shuffled versions were generated for calculating thin-tail kurtosis coefficient $T-K_{50}$. Figure 5.14 and 5.15 shows the distribution of thin-tail kurtosis coefficient $T-K_{50}$ for regulatory, coding and non-coding non-regulatory DNA. One can see that regulatory DNAs (CRMs) tend to have smaller $T-K_{50}$ than exons and NCNRs. The value $T-K_{50} = 0$ was used as a threshold to distinguish CRMs from other types of DNA. When $T-K_{50} < 0$, we declare the input sequence has a thin-tail distribution for the similar-word set Z -score.

Table 5.1(a) lists the results of classification of the function of DNA sequences in our training data sets based on the thin-tail kurtosis coefficient $T-K_{50}$, 44 out of 60 CRMs in our positive training dataset have $T-K_{50} < 0$, while 24 out of 60 (40%) NCNRs in the NCNR data set have $T-K_{50} < 0$, and 26 out of 60 (43.3%) exons in the exon data set have $T-K_{50} < 0$.

Figure 5.16 and 5.17 shows the distribution of thin-tail kurtosis coefficient $T-KK$ for regulatory, coding and non-coding non-regulatory DNA. One can see that regulatory

DNAs (CRMs) tend to have smaller $T-KK$ than exons and NCNRs. The value $T-KK = 0.6$ was used as a threshold. In our case, when $T-KK < 0.6$, the input sequence was declared to have a thin-tail distribution for the similar-word set Z -score.

Table 5.1(b) lists the results of classification of the function of DNA sequences in our training data sets based on the thin-tail kurtosis coefficient $T-KK$. 43 out of 60 (71.7%) CRMs in the positive training data set have $T-KK < 0.6$, while 25 out of 60 (41.6%) NCNRs in the NCNR data set have $T-KK < 0.6$, and 26 out of 60 (43.3%) exons in the exon data set have $T-KK < 0.6$.

Table 5.1(a) Classification of the 180 sequences by thin-tail test with kurtosis coefficient $T-K_{50}$

Functional type	$T-K_{50} < 0$	$T-K_{50} > 0$	Positive Rate	Negative rate
Regulatory regions	44	16	73.3%	26.7%
Exons	26	34	43.3%	56.7%
Non-coding presumed non-regulatory regions	24	36	40%	60%

Table 5.1(b) Classification of the 180 sequences by thin-tail test with kurtosis coefficient $T-KK$

Functional type	$T-KK < 0.6$	$T-KK > 0.6$	Positive Rate	Negative rate
Regulatory regions	43	17	71.7%	28.3%
Exons	26	34	43.3%	56.7%
Non-coding presumed non-regulatory regions	25	35	41.6%	58.4%

Figure 5.14 Histograms for regulatory, coding sequences classified by $T-K_{50}$. (Visualizing the results in Table 5.1(a)) The X axis shows the thin-tail kurtosis coefficient $T-K_{50}$; the Y axis is the number of sequences in the data set with this $T-K_{50}$.

Figure 5.15 Histograms for regulatory, NCNR sequences classified by $T-K_{50}$.

(Visualizing the results in Table 5.1(a)) The X axis shows the thin-tail kurtosis coefficient $T-K_{50}$; the Y axis is the number of sequences in the data set with this $T-K_{50}$.

Figure 5.16 Histograms for regulatory, coding sequences classified by $T-KK$. (Visualizing the result in Table 5.1(b)) The X axis shows the thin-tail kurtosis coefficient $T-KK$; the Y axis is the number of sequences in the data set with this $T-KK$.

Figure 5.17 Histograms for regulatory and NCNR sequences classified by $T-KK$.

(Visualizing the result in Table 5.1(b)) The X axis shows the thin-tail kurtosis coefficient $T-KK$; the Y axis is the number of sequences in the data set with this $T-KK$.

5.3.5 Comparison with fat-tail test

CRMs and NCNR training data sets were employed to evaluate the performance of separation between CRMs and NCNRs by thin-tail test and fat-tail test. The results are summarized in Table 5.2(a) and (b). Results show that the thin-tail test has higher average successful rate (thin-tail kurtosis coefficient $T-K_{50}$: 66.6% and $T-KK$: 65%), compared to the fat-tail test without repeat-masking (fat-tail kurtosis coefficient $F-K_{50}$: 61.6% and $F-KK$: 55.8%). Especially, the thin-tail test can more efficiently identify NCNRs (SP = 60% for $T-K_{50}$ and SP = 58.3% for $T-KK$) than the fat-tail test without repeat-masking (SP = 50% for $F-K_{50}$ and SP = 35% for $F-KK$).

Compared with the fat-tail test with repeat-masking (Table 5.2 (b)), the thin-tail test with kurtosis coefficient $T-K_{50}$ and $T-KK$ has higher average success rate than the fat-tail test with $F-KK$, but lower average success rate than the fat-tail test with $F-K_{50}$.

Chapter 5 Thin-tail test

Table 5.2(a) Comparison between thin-tail test and the fat-tail test without repeat-
masking on distinguishing CRMs from NCNRs.

	Thin-tail test		Fat-tail test (before repeat-masking)	
	$T-K_{50}$	$T-KK$	$F-K_{50}$	$F-KK$
SN	73.3%	71.7%	73.3%	76.7%
SP	60%	58.3%	50%	35%
Averaged successful rate	66.6%	65%	61.6%	55.8%

Table 5.2(b) Comparison between thin-tail test and the fat-tail test with repeat-
masking on distinguishing CRMs from NCNRs.

	Thin-tail test		Fat-tail test (after repeat-masking)	
	$T-K_{50}$	$T-KK$	$F-K_{50}$	$F-KK$
SN	73.3%	71.7%	76.6%	75%
SP	60%	58.3%	63.3%	46.7%
Averaged successful rate	66.6%	65%	70%	60.8%

5.3.6 Comparison with fluffy-tail test

CRMs and NCNR training data sets were employed to evaluate the performance of separation between CRMs and NCNRs by thin-tail test and fluffy-tail test. Results are summarized in Table 5.3(a) and (b). Results show that the thin-tail test has higher specificity but lower sensitivity than the fluffy-tail test without repeat-masking. The thin-tail test with kurtosis coefficient $T-K_{50}$ has higher average successful rate (66.6%) than the fluffy-tail test without repeat-masking (65%); the thin-tail test with kurtosis coefficient $T-KK$ has same average successful rate (65%) as the fluffy-tail test without repeat-masking.

Compared with the fluffy-tail test with repeat-masking (Table 5.3 (b)), the thin-tail test with kurtosis coefficient $T-K_{50}$ and $T-KK$ has lower average success rate than the fluffy-tail test with repeat-masking. However, the thin-tail test has higher sensitivity but lower specificity than the fluffy-tail test with repeat-masking.

Table 5.3 (a) Comparison between thin-tail test and fluffy-tail test (before repeat-
masking) on distinguishing CRMs from NCNRs

	Thin-tail test		Fluffy tail test (before repeat-masking)
	$T\text{-}K_{50}$	$T\text{-}KK$	F_{50}
SN	73.3%	71.7%	81.7%
SP	60%	58.3%	48.3%
Averaged successful rate	66.6%	65%	65%

Table 5.3 (b) Comparison between thin-tail test and fluffy-tail test (after repeat-
masking) on distinguishing CRMs from NCNRs.

	Thin-tail test		Fluffy tail test (after repeat-masking)
	$T\text{-}K_{50}$	$T\text{-}KK$	F_{50}
SN	73.3%	71.7%	70%
SP	60%	58.3%	65%
Averaged successful rate	66.6%	65%	67.5%

5.3.7 Time complexity

Similar to fat-tail coefficient $F-K_r$, thin-tail coefficient $T-K_r$ is calculated by bootstrapping procedure, the value is affected by the number of randomization r . Table 5.4 shows that $F-K_r$ is somewhat unstable for different r . In order to get more reliable estimates for $F-K_r$, larger r is need, however it will make the computational time unreasonable. For reliable result and reasonable computational time the original sequence was resample 50 times in calculating the kurtosis coefficient $T-K_r$.

Table 5.5 shows the run-time of the thin-tail test for an input sequence of length 1000. The run-time for calculating $T-KK$ is 54 second, it is 50 times faster than that for calculating $T-K_{50}$ and 9 times slower than calculating $F-KK$, and 6 times faster than calculating $F-K_{50}$ for the same input sequence. All experiments were run on a 3.2 GHZ Pentium IV processor with 1G physical memory.

Table 5.4 Sensitivity of $T-K_r$ to choice of r , the number of randomization, for the hb regulatory sequence

r	$T-K_r$	$\bar{k}$	σ_r
30	-1.50	0.06	0.24
50	-1.31	0.07	0.28
100	-1.69	0.12	0.25

r is the number of randomization, $T-K_r$ is the thin-tail kurtosis coefficient , $\bar{k}$ and σ_r are the mean and standard deviation of kurtosis of similar-word set Z -score for r random shuffled versions of original sequence.

Table 5.5 Computing time for calculating thin-tail test kurtosis coefficient $T-K_r$, $T-KK$ and fat-tail test kurtosis coefficient $F-K_r$, $F-KK$. (the length of input DNA sequence is 1000).

	Thin-tail test		Fat-tail test	
	$T-K_{50}$	$T-KK$	$F-K_{50}$	$F-KK$
Avg. run-time	2700 sec	54 sec	310 sec	6.2 sec

5.4 Discussion

In this chapter, the statistical properties of CRMs were investigated by estimating the distribution pattern of similar-word set *Z-score*. The thin-tail test with kurtosis coefficient $T-K_r$ and $T-KK$ has been proposed to discriminate CRMs from NCNRs. Results show that CRMs tend to have smaller kurtosis coefficient $T-K_r$ or $T-KK$ compared to NCNRs.

5.4.1 Comparison between $T-K_r$ and $T-KK$

In Appendix C, Table C.1 lists the values of $T-K_{50}$ and $T-KK$ for the 180 sequences in our training data sets. For most CRMs with $T-K_{50} < 0$, its corresponding $T-KK$ is smaller than 0.6, only 3 out of 60 (5%) CRMs with $T-K_{50} < 0$, the corresponding $T-KK$ is larger than 0.6. The reason for the larger value of $T-KK$ is the distribution of similar-word set *Z-score* for random shuffled original sequence derives from normal distribution due to severely bias of nucleotide composition. Thus, thin-tail kurtosis coefficient $T-KK$ is recommended to be used instead of $T-K_r$, only under the balance nucleotide composition.

5.4.2 Application to REDfly test set

The application of the thin-tail test was extended to REDfly database. Results show that 70.8% CRMs have $T-K_{50} < 0$, and 72.5% CRMs have $T-KK < 0.6$, passing the thin-tail test.

5.4.4 Remarks

Similar to fat-tail test, it is difficult to set the value of threshold to discriminate all the CRMs from NCNRs. To solve this problem, thin-tail score could be introduced. For each input sequence, random non-coding sequences of the same length are chosen from *D. melanogaster* genome, the thin-tail coefficient is computed for each of these, and an empirical p-value computed as the fraction random sequences that scored smaller than or equal to the given sequence's thin-tail coefficient. This empirical p-value is called the "thin-tail score". A sequence is classified as a CRM if its thin-tail score (empirical p-value) is below a threshold τ .

5.5 Conclusion

Statistical properties of CRMs are explored by estimating the distribution of similar-word set *Z-score*. Results show that the distribution of similar-word set *Z-score* for CRMs tends to be thin-tail distribution compared to exons and NCNRs. Based on this observation, we have proposed thin-tail test with two kurtosis based indexes: $T-K_r$ and $T-KK$. By using $T-K_r$ and $T-KK$, the thin-tail test has higher specificity for separation between CRMs and NCNRs than fat-tail test and fluffy tail test (Abnizova, et al., 2005) without repeat-masking.

Chapter 6

Conclusions

6.1 Hypercomplex number representation of DNA

A hypercomplex number representation of DNA is proposed to facilitate comparing DNA sequences with fuzzy composition. With the hypercomplex number representation, the conventional sequence analysis methods, such as, dot matrix analysis, dynamic programming, and cross-correlation method have been extended and improved to align DNA sequences with fuzzy composition. Hypercomplex dot matrix analysis can provide more control over the degree of alignment desired. A new scoring model has been proposed to accommodate the hypercomplex number representation of DNA and integrated with dynamic programming alignment method. By using hypercomplex cross-correlation, the match and mismatch information between two aligned DNA sequences are separately stored in resultant real part and imaginary parts respectively. The mismatch information is very useful to refine consensus sequence based motif scanning.

6.2 Statistical properties of cis-regulatory region

The statistical properties of cis-regulatory modules are explored in the thesis by estimating the distribution of similar-word set. It is observed that cis-regulatory

Conclusions

modules tend to have fat-tail distribution for similar-word set occurrence. Based on this observation, fat-tail test with two kurtosis based coefficients $F-K_r$ and $F-KK$ has been proposed to distinguish CRMs from non-CRMs. The performance of the proposed method on the experimentally verified *Drosophila melanogaster* regulatory regions shows that the method allows us to predict whether a sequence may be regulatory in a relatively sensitive and efficient way.

Publications

1. Li, Y., Shu, J.-J. 2006. Cross-correlation of DNA sequences using Hypercomplex number encoding. International Conference on Biomedical and Pharmaceutical Engineering. Singapore.

References

- Abnizova, I., te Bockhorst, R., Walter, K. and Gilks, W.R. (2005) Some statistical properties of regulatory DNA sequences, and their use in predicting regulatory regions in the Drosophila genome: the fluffy-tail test, *BMC Bioinformatics*, **6**, 109.
- Adams, M.D., Celniker, S.E., Holt, R.A., Evans, C.A., Gocayne, J.D., Amanatides, P.G., Scherer, S.E., Li, P.W., Hoskins, R.A., Galle, R.F., et al. (2000) The genome sequence of *Drosophila melanogaster*, *Science*, **287**, 2185-2195.
- Alphey, Luke. (1997) DNA sequencing: from experimental methods to bioinformatics. United Kingdom: BIOS Scientific Publisher Ltd.
- Anastassiou, D. (2000) Frequency-domain analysis of biomolecular sequences, *Bioinformatics*, **16**, 1073-1081.
- Arnosti, D.N., Barolo, S., Levine, M. and Small, S. (1996) The eve stripe 2 enhancer employs multiple modes of transcriptional synergy, *Development*, **122**, 205-214.
- Arnosti, D.N., Gray, S., Barolo, S., Zhou, J. and Levine, M. (1996) The gap protein knirps mediates both quenching and direct repression in the *Drosophila* embryo, *Embo J*, **15**, 3659-3666.
- Bailey, T.L. and Noble, W.S. (2003) Searching for statistically significant regulatory modules, *Bioinformatics*, **19**, ii16-25.
- Benson, G. (1999) Tandem repeats finder: a program to analyze DNA sequences, *Nucleic Acids Rs*, **27**, 573-580

References

- Berman, B.P., Nibu, Y., Pfeiffer, B.D., Tomancak, P., Celniker, S.E., Levine, M., Rubin, G.M. and Eisen, M.B. (2002) Exploiting transcription factor binding site clustering to identify cis-regulatory modules involved in pattern formation in the *Drosophila* genome, *Proc Natl Acad Sci U S A*, **99**, 757-762.
- Bertuccioli, C., Fasano, L., Jun, S., Wang, S., Sheng, G. and Desplan, C. (1996) In vivo requirement for the paired domain and homeodomain of the paired segmentation gene product, *Development*, **122**, 2673-2685.
- Blanchette, M. and Tompa, M. (2002) Discovery of regulatory elements by a computational method for phylogenetic footprinting, *Genome Res*, **12**, 739-748.
- Boffelli, D., McAuliffe, J., Ovcharenko, D., Lewis, K.D., Ovcharenko, I., Pachter, L. and Rubin, E.M. (2003) Phylogenetic shadowing of primate sequences to find functional regions of the human genome, *Science*, **299**, 1391-1394.
- Bouchard, M., St-Amand, J. and Cote, S. (2000) Combinatorial activity of pair-rule proteins on the *Drosophila* gooseberry early enhancer, *Dev Biol*, **222**, 135-146.
- Bucher, P. (1990) Weight matrix descriptions of four eukaryotic RNA polymerase II promoter elements derived from 502 unrelated promoter sequences, *J Mol Biol*, **212**, 563-578.
- Bülow, T. (1999) Hypercomplex spectral Signal Representations for the Processing and Analysis of Images. University of Kiel, Germany.

References

- Bülow, T., Sommer, G. (2001) Hypercomplex signals - A novel extension of the analytic signal to the multidimensional case, *IEEE Transactions on Signal Processing*, **49**, 2844-2852.
- Butler, B.A., Soong, J. and Gergen, J.P. (1992) The *Drosophila* segmentation gene *runt* has an extended cis-regulatory region that is required for vital expression at other stages of development, *Mech Dev*, **39**, 17-28.
- Capovilla, M., Eldon, E.D. and Pirrotta, V. (1992) The giant gene of *Drosophila* encodes a b-ZIP DNA-binding protein that regulates the expression of other segmentation gap genes, *Development*, **114**, 99-112.
- Carbone, A., Zinovyev, A., and Képès, F. (2003) Codon adaptation index as a measure of dominating codon bias. *Bioinformatics* 19: 2005-2015.
- Cartharius, K., Frech, K., Grote, K., Klocke, B., Haltmeier, M., Klingenhoff, A., Frisch, M., Bayerlein, M. and Werner, T. (2005) MatInspector and beyond: promoter analysis based on transcription factor binding sites, *Bioinformatics*, **21**, 2933-2942.
- Chan, B.Y. and Kibler, D. (2005) Using hexamers to predict cis-regulatory motifs in *Drosophila*, *BMC Bioinformatics*, **6**, 262.
- Cheever, E.A., Overton, G.C. and Searls, D.B. (1991) Fast Fourier transform-based correlation of DNA sequences using complex plane encoding, *Comput Appl Biosci*, **7**, 143-154.

References

- Cornish-Bowden, A. (1985) Nomenclature for incompletely specified bases in nucleic acid sequences: recommendations 1984, *Nucleic Acids Res*, **13**, 3021-3030.
- Davidson, E. (2001) *Genomic Regulatory Systems: Development and Evolution*. Academic Press, San Diego.
- Davis, G.K., Jaramillo, C.A. and Patel, N.H. (2001) Pax group III genes and the evolution of insect pair-rule patterning, *Development*, **128**, 3445-3458.
- de Celis, J.F., Barrio, R. and Kafatos, F.C. (1999) Regulation of the spalt/spalt-related gene complex and its function during sensory organ development in the *Drosophila* thorax, *Development*, **126**, 2653-2662.
- Durbin, R., Eddy, S. R., Krogh, A. and Mitchison, G. (1998) *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge.
- Epstein, J., Cai, J., Glaser, T., Jepeal, L. and Maas, R. (1994) Identification of a Pax paired domain recognition sequence and evidence for DNA-dependent conformational changes, *J Biol Chem*, **269**, 8355-8361.
- Felsenstein, J., Sawyer, S. and Kochin, R. (1982) An Efficient Method for Matching Nucleic-Acid Sequences, *Nucleic Acids Research*, **10**, 133-139.
- Fickett, J.W. and Hatzigeorgiou, A.G. (1997) Eukaryotic promoter recognition, *Genome Res*, **7**, 861-878.

References

- Finkelstein, R. and Perrimon, N. (1990) The orthodenticle gene is regulated by bicoid and torso and specifies Drosophila head development, *Nature*, **346**, 485-488.
- Freschi, V. and Bogliolo, A. (2005) Using sequence compression to speedup probabilistic profile matching, *Bioinformatics*, **21**, 2225-2229.
- Frith, M.C., Li, M.C. and Weng, Z. (2003) Cluster-Buster: Finding dense clusters of motifs in DNA sequences, *Nucleic Acids Res*, **31**, 3666-3668.
- Fujioka, M., Emi-Sarker, Y., Yusibova, G.L., Goto, T. and Jaynes, J.B. (1999) Analysis of an even-skipped rescue transgene reveals both composite and discrete neuronal and early blastoderm enhancers, and multi-stripe positioning by gap gene repressor gradients, *Development*, **126**, 2527-2538.
- Gallo, S.M., Li, L., Hu, Z. and Halfon, M.S. (2006) REDfly: a Regulatory Element Database for Drosophila, *Bioinformatics*, **22**, 381-383.
- Gibert, J.M., Joannin, N., Blin, M., Rigolot, C., Mouchel-Vielh, E., Queinnec, E. and Deutsch, J.S. (2002) Heterospecific transgenesis in Drosophila suggests that engrailed.a is regulated by POU proteins in the crustacean Sacculina carcini, *Dev Genes Evol*, **212**, 19-29.
- Gonnet, G.H. (2004) Some string matching problems from bioinformatics which still need better solutions *J. of Discrete Algorithms* **2**, 3-15

References

- Goto, T., Macdonald, P. and Maniatis, T. (1989) Early and late periodic patterns of even skipped expression are controlled by distinct regulatory elements that respond to different spatial cues, *Cell*, **57**, 413-422.
- Grad, Y.H., Roth, F.P., Halfon, M.S. and Church, G.M. (2004) Prediction of similarly acting cis-regulatory modules by subsequence profiling and comparative genomics in *Drosophila melanogaster* and *D.pseudoobscura*, *Bioinformatics*, **20**, 2738-2750.
- Gupta, M. and Liu, J.S. (2005) De novo cis-regulatory module elicitation for eukaryotic genomes, *Proc Natl Acad Sci U S A*, **102**, 7079-7084.
- Gusfield, D. (1997) *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology* Cambridge University Press, Cambridge.
- Guss, K.A., Nelson, C.E., Hudson, A., Kraus, M.E. and Carroll, S.B. (2001) Control of a genetic regulatory network by a selector gene, *Science*, **292**, 1164-1167.
- Hader, T., La Rosee, A., Ziebold, U., Busch, M., Taubert, H., Jackle, H. and Rivera-Pomar, R. (1998) Activation of posterior pair-rule stripe expression in response to maternal caudal and zygotic knirps activities, *Mech Dev*, **71**, 177-186.
- Hamilton, W.R. (1866) *Element of quaternion*. Longmans, Green and Co., London.
- Han, W., Yu, Y., Su, K., Kohanski, R.A. and Pick, L. (1998) A binding site for multiple transcriptional activators in the fushi tarazu proximal enhancer is essential for gene expression in vivo, *Mol Cell Biol*, **18**, 3384-3394.

References

- Hannenhalli, S. and Levy, S. (2002) Predicting transcription factor synergism, *Nucleic Acids Res*, **30**, 4278-4284.
- Hertz, G.Z. and Stormo, G.D. (1999) Identifying DNA and protein patterns with statistically significant alignments of multiple sequences, *Bioinformatics*, **15**, 563-577.
- Hoch, M., Gerwin, N., Taubert, H. and Jackle, H. (1992) Competition for overlapping sites in the regulatory region of the Drosophila gene Kruppel, *Science*, **256**, 94-97.
- Hughes, J.D., Estep, P.W., Tavazoie, S. and Church, G.M. (2000) Computational identification of cis-regulatory elements associated with groups of functionally related genes in *Saccharomyces cerevisiae*, *J Mol Biol*, **296**, 1205-1214.
- IUPAC-IUB (1985) Nomenclature for incompletely specified bases in nucleic acid sequences (recommendations 1984), *Biochem. J.* **229**, 281-286.
- Irvine, K.D., Botas, J., Jha, S., Mann, R.S. and Hogness, D.S. (1993) Negative autoregulation by Ultrabithorax controls the level and pattern of its expression, *Development*, **117**, 387-399.
- Johansson, O., Alkema, W., Wasserman, W.W. and Lagergren, J. (2003) Identification of functional clusters of transcription factor binding motifs in genome sequences: the MSCAN algorithm, *Bioinformatics*, **19 Suppl 1**, i169-176.
- Jones, B. and McGinnis, W. (1993) The regulation of empty spiracles by Abdominal-B mediates an abdominal segment identity function, *Genes Dev*, **7**, 229-240.

References

- Kania, M.A., Bonner, A.S., Duffy, J.B. and Gergen, J.P. (1990) The *Drosophila* segmentation gene *runt* encodes a novel nuclear regulatory protein that is also expressed in the developing nervous system, *Genes Dev*, **4**, 1701-1713.
- Karch, F., Bender, W. and Weiffenbach, B. (1990) *abdA* expression in *Drosophila* embryos, *Genes Dev*, **4**, 1573-1587.
- Kielbasa, S.M., Gonze, D. and Herzel, H. (2005) Measuring similarities between transcription factor binding sites, *BMC Bioinformatics*, **6**, 237.
- Klingler, M., Soong, J., Butler, B. and Gergen, J.P. (1996) Disperse versus compact elements for the regulation of *runt* stripes in *Drosophila*, *Dev Biol*, **177**, 73-84.
- Krivan, W. and Wasserman, W.W. (2001) A predictive model for regulatory sequences directing liver-specific transcription, *Genome Res*, **11**, 1559-1566.
- La Rosee-Borggreve, A., Hader, T., Wainwright, D., Sauer, F. and Jackle, H. (1999) hairy stripe 7 element mediates activation and repression in response to different domains and levels of Kruppel in the *Drosophila* embryo, *Mech Dev*, **89**, 133-140.
- Laney, J.D. and Biggin, M.D. (1997) Zeste-mediated activation by an enhancer is independent of cooperative DNA binding in vivo, *Proc Natl Acad Sci U S A*, **94**, 3602-3604.
- Langeland, J.A. and Carroll, S.B. (1993) Conservation of regulatory elements controlling hairy pair-rule stripe formation, *Development*, **117**, 585-596.

References

- Lenhard, B., Sandelin, A., Mendoza, L., Engstrom, P., Jareborg, N. and Wasserman, W.W. (2003) Identification of conserved regulatory elements by comparative genome analysis, *J Biol*, **2**, 13.
- Leung, M.Y., Marsh, G.M. and Speed, T.P. (1996) Over- and underrepresentation of short DNA words in herpesvirus genomes, *J Comput Biol*, **3**, 345-360.
- Lewin, B. (2000) *Gene VII*. Orford University Press.
- Lewis, E.B., Knafels, J.D., Mathog, D.R. and Celniker, S.E. (1995) Sequence analysis of the cis-regulatory regions of the bithorax complex of *Drosophila*, *Proc Natl Acad Sci U S A*, **92**, 8403-8407.
- Liaw, G.J., Rudolph, K.M., Huang, J.D., Dubnicoff, T., Courey, A.J. and Lengyel, J.A. (1995) The torso response element binds GAGA and NTF-1/Elf-1, and regulates tailless by relief of repression, *Genes Dev*, **9**, 3163-3176.
- Lifanov, A.P., Makeev, V.J., Nazina, A.G. and Papatsenko, D.A. (2003) Homotypic regulatory clusters in *Drosophila*, *Genome Res*, **13**, 579-588.
- Loots, G.G., Ovcharenko, I., Pachter, L., Dubchak, I. and Rubin, E.M. (2002) rVista for comparative sequence-based discovery of functional transcription factor binding sites, *Genome Res*, **12**, 832-839.
- Ludwig, M.Z., Bergman, C., Patel, N.H. and Kreitman, M. (2000) Evidence for stabilizing selection in a eukaryotic enhancer element, *Nature*, **403**, 564-567.

References

- Mantegna, R. N., Buldyrev, S.V., Goldberger, A.L., Havlin, S., Peng, C.-K., Simons, M. and Stanley, H.E. (1994) Linguistic Features of Noncoding DNA Sequences, *Phys. Rev. Lett.* **73**, 3169-3172.
- Markstein, M., Markstein, P., Markstein, V. and Levine, M.S. (2002) Genome-wide analysis of clustered Dorsal binding sites identifies putative target genes in the *Drosophila* embryo, *Proc Natl Acad Sci U S A*, **99**, 763-768.
- Martin, C.H., Mayeda, C.A., Davis, C.A., Ericsson, C.L., Knafels, J.D., Mathog, D.R., Celniker, S.E., Lewis, E.B. and Palazzolo, M.J. (1995) Complete sequence of the bithorax complex of *Drosophila*, *Proc Natl Acad Sci U S A*, **92**, 8398-8402.
- Matys, V., Fricke, E., Geffers, R., Gossling, E., Haubrock, M., Hehl, R., Hornischer, K., Karas, D., Kel, A.E., Kel-Margoulis, O.V., Kloos, D.U., Land, S., Lewicki-Potapov, B., Michael, H., Munch, R., Reuter, I., Rotert, S., Saxel, H., Scheer, M., Thiele, S. and Wingender, E. (2003) TRANSFAC: transcriptional regulation, from patterns to profiles, *Nucleic Acids Res*, **31**, 374-378.
- Miskiewicz, P., Morrissey, D., Lan, Y., Raj, L., Kessler, S., Fujioka, M., Goto, T. and Weir, M. (1996) Both the paired domain and homeodomain are required for in vivo function of *Drosophila* Paired, *Development*, **122**, 2709-2718.
- Morasso, M.I., Markova, N.G. and Sargent, T.D. (1996) Regulation of epidermal differentiation by a Distal-less homeodomain gene, *J Cell Biol*, **135**, 1879-1887.

References

- Mount, D.W. (2001) *Bioinformatics: Sequence and Genome Analysis*. Cold Spring Harbour Laboratory Press, New York.
- Myers, E.W., Sutton, G.G., Delcher, A.L., Dew, I.M., Fasulo, D.P., Flanigan, M.J., Kravitz, S.A., Mobarry, C.M., Reinert, K.H., Remington, K.A., Anson, E.L., Bolanos, R.A., Chou, H.H., Jordan, C.M., Halpern, A.L., Lonardi, S., Beasley, E.M., Brandon, R.C., Chen, L., Dunn, P.J., Lai, Z., Liang, Y., Nusskern, D.R., Zhan, M., Zhang, Q., Zheng, X., Rubin, G.M., Adams, M.D. and Venter, J.C. (2000) A whole-genome assembly of *Drosophila*, *Science*, **287**, 2196-2204.
- Nasiadka, A., Grill, A. and Krause, H.M. (2000) Mechanisms regulating target gene selection by the homeodomain-containing protein Fushi tarazu, *Development*, **127**, 2965-2976.
- Nazina, A.G. and Papatsenko, D.A. (2003) Statistical extraction of *Drosophila* cis-regulatory modules using exhaustive assessment of local word frequency, *BMC Bioinformatics*, **4**, 65.
- Needleman, S.B. and Wunsch, C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins, *J Mol Biol*, **48**, 443-453.
- O'Hara, E., Cohen, B., Cohen, S.M. and McGinnis, W. (1993) Distal-less is a downstream gene of Deformed required for ventral maxillary identity, *Development*, **117**, 847-856.

References

- Orihara, M., Hosono, C., Kojima, T. and Saigo, K. (1999) Identification of engrailed promoter elements essential for interactions with a stripe enhancer in *Drosophila* embryos, *Genes Cells*, **4**, 205-218.
- Pankratz, M.J., Busch, M., Hoch, M., Seifert, E. and Jackle, H. (1992) Spatial control of the gap gene knirps in the *Drosophila* embryo by posterior morphogen system, *Science*, **255**, 986-989.
- Papatsenko, D.A., Makeev, V.J., Lifanov, A.P., Regnier, M., Nazina, A.G. and Desplan, C. (2002) Extraction of Functional Binding Sites from Unique Regulatory Regions: The *Drosophila* Early Developmental Enhancers, *Genome Res.*, **12**, 470-481.
- Pedersen, A.G., Baldi, P., Chauvin, Y. and Brunak, S. (1999) The biology of eukaryotic promoter prediction--a review, *Comput Chem*, **23**, 191-207.
- Pei, S.C., Ding, J.J. and Chang, J.H. (2001) Efficient implementation of quaternion Fourier transform, convolution, and correlation by 2-D complex FFT, *Ieee Transactions on Signal Processing*, **49**, 2783-2797.
- Peng, C.K., Buldyrev, S.V., Goldberger, A.L., Havlin, S., Sciortino, F., Simons, M. and Stanley, H.E. (1992) Long-range correlations in nucleotide sequences, *Nature*, **356**, 168-170.
- Prestridge, D.S. (1991) SIGNAL SCAN: a computer program that scans DNA sequences for eukaryotic transcriptional elements, *Comput Appl Biosci*, **7**, 203-206.

References

- Qian, S., Capovilla, M. and Pirrotta, V. (1993) Molecular mechanisms of pattern formation by the BRE enhancer of the Ubx gene, *Embo J*, **12**, 3865-3877.
- Qiu, P. (2003) Recent advances in computational promoter analysis in understanding the transcriptional regulatory network, *Biochem Biophys Res Commun*, **309**, 495-501.
- Rajasekaran, S., Jin, X. and Spouge, J.L. (2002) The efficient computation of position-specific match scores with the fast fourier transform, *J Comput Biol*, **9**, 23-33.
- Rajewsky, N., Vergassola, M., Gaul, U. and Siggia, E.D. (2002) Computational detection of genomic cis-regulatory modules applied to body patterning in the early *Drosophila* embryo, *BMC Bioinformatics*, **3**, 30.
- Rebeiz, M., Reeves, N.L. and Posakony, J.W. (2002) SCORE: a computational approach to the identification of cis-regulatory modules and target genes in whole-genome sequence data. Site clustering over random expectation, *Proc Natl Acad Sci U S A*, **99**, 9888-9893.
- Rgnier, M. (2000) A unified approach to word occurrence probabilities *Discrete Appl. Math.*
- Rgnier, M., Lifanov, A., and Makeev, A. (2000) Three variations on word counting. *German Conference on Bioinformatics*. Heidelberg., 75–82.
- Rivera-Pomar, R., Lu, X., Perrimon, N., Taubert, H. and Jackle, H. (1995) Activation of posterior gap gene expression in the *Drosophila* blastoderm, *Nature*, **376**, 253-256.

References

- Rockwood, A.L., Crockett, D.K., Oliphant, J.R. and Elenitoba-Johnson, K.S. (2005) Sequence alignment by cross-correlation, *J Biomol Tech*, **16**, 453-458.
- Rudolph, K.M., Liaw, G.J., Daniel, A., Green, P., Courey, A.J., Hartenstein, V. and Lengyel, J.A. (1997) Complex regulatory region mediating tailless expression in early embryonic patterning and brain development, *Development*, **124**, 4297-4308.
- Sackerson, C., Fujioka, M. and Goto, T. (1999) The even-skipped locus is contained in a 16-kb chromatin domain, *Dev Biol*, **211**, 39-52.
- Sandelin, A., Alkema, W., Engstrom, P., Wasserman, W.W. and Lenhard, B. (2004) JASPAR: an open-access database for eukaryotic transcription factor binding profiles, *Nucleic Acids Res*, **32**, D91-94.
- Sandelin, A. and Wasserman, W.W. (2004) Constrained binding site diversity within families of transcription factors enhances pattern discovery bioinformatics, *J Mol Biol*, **338**, 207-215.
- Schock, F., Reischl, J., Wimmer, E., Taubert, H., Purnell, B.A. and Jackle, H. (2000) Phenotypic suppression of empty spiracles is prevented by buttonhead, *Nature*, **405**, 351-354.
- Schones, D.E., Sumazin, P. and Zhang, M.Q. (2005) Similarity of position frequency matrices for transcription factor binding sites, *Bioinformatics*, **21**, 307-313.

References

- Schroeder, M.D., Pearce, M., Fak, J., Fan, H., Unnerstall, U., Emberly, E., Rajewsky, N., Siggia, E.D. and Gaul, U. (2004) Transcriptional control in the segmentation gene network of *Drosophila*, *PLoS Biol*, **2**, E271.
- Shimell, M.J., Peterson, A.J., Burr, J., Simon, J.A. and O'Connor, M.B. (2000) Functional analysis of repressor binding sites in the *iab-2* regulatory region of the abdominal-A homeotic gene, *Dev Biol*, **218**, 38-52.
- Sinha, S., van Nimwegen, E. and Siggia, E.D. (2003) A probabilistic method to detect regulatory modules, *Bioinformatics*, **19 Suppl 1**, i292-301.
- Small, S., Blair, A. and Levine, M. (1996) Regulation of two pair-rule stripes by a single enhancer in the *Drosophila* embryo, *Dev Biol*, **175**, 314-324.
- Smith, T.F.a.W., M.S. (1981) Comparison of bio-sequences, *Adv. Appl. Math.*, **2**, 482-489.
- Stanojevic, D., Small, S. and Levine, M. (1991) Regulation of a segmentation stripe by overlapping activators and repressors in the *Drosophila* embryo, *Science*, **254**, 1385-1387.
- Sumiyama, K., Kim, C.B. and Ruddle, F.H. (2001) An efficient cis-element discovery method using multiple sequence comparisons based on evolutionary relationships, *Genomics*, **71**, 260-262.

References

- TenHarmsel, A., Austin, R.J., Savenelli, N. and Biggin, M.D. (1993) Cooperative binding at a distance by even-skipped protein correlates with repression and suggests a mechanism of silencing, *Mol Cell Biol*, **13**, 2742-2752.
- Tsai, C. and Gergen, P. (1995) Pair-rule expression of the *Drosophila fushi tarazu* gene: a nuclear receptor response element mediates the opposing regulatory effects of runt and hairy, *Development*, **121**, 453-462.
- Vachon, G., Cohen, B., Pfeifle, C., McGuffin, M.E., Botas, J. and Cohen, S.M. (1992) Homeotic genes of the Bithorax complex repress limb development in the abdomen of the *Drosophila* embryo through the target gene *Distal-less*, *Cell*, **71**, 437-450.
- van Helden, J., Andre, B. and Collado-Vides, J. (1998) Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies, *J Mol Biol*, **281**, 827-842.
- Vavouri, T. and Elgar, G. (2005) Prediction of cis-regulatory elements using binding site matrices -- the successes, the failures and the reasons for both, *Current Opinion in Genetics & Development*, **15**, 395-402.
- Venter, J.C., Adams, M.D., Myers, E.W., Li, P.W., Mural, R.J., Sutton, G.G., Smith, H.O., Yandell, M., Evans, C.A., Holt, R.A., et al. (2001) The sequence of the human genome, *Science*, **291**, 1304-1351.
- Wasserman, W.W. and Fickett, J.W. (1998) Identification of regulatory regions which confer muscle-specific gene expression, *J Mol Biol*, **278**, 167-181.

References

- Wimmer, E.A., Carleton, A., Harjes, P., Turner, T. and Desplan, C. (2000) Bicoid-independent formation of thoracic segments in *Drosophila*, *Science*, **287**, 2476-2479.
- Wingender, E., Chen, X., Fricke, E., Geffers, R., Hehl, R., Liebich, I., Krull, M., Matys, V., Michael, H., Ohnhauser, R., Pruss, M., Schacherer, F., Thiele, S. and Urbach, S. (2001) The TRANSFAC system on gene expression regulation, *Nucleic Acids Res*, **29**, 281-283.
- Wingender, E. (1993) Gene regulation in Eukaryotes, VCH Weinheim.
- Xue, L. and Noll, M. (2002) Dual role of the Pax gene paired in accessory gland development of *Drosophila*, *Development*, **129**, 339-346.
- Yuh, C.H., Brown, C.T., Livi, C.B., Rowen, L., Clarke, P.J. and Davidson, E.H. (2002) Patchy interspecific sequence similarities efficiently identify positive cis-regulatory elements in the sea urchin, *Dev Biol*, **246**, 148-161.
- Zhu, J. and Zhang, M.Q. (1999) SCPD: a promoter database of the yeast *Saccharomyces cerevisiae*, *Bioinformatics*, **15**, 607-611.

Appendix A

Hypercomplex cross-correlation calculation

A.1 Definition of hypercomplex cross-correlation

Consider $\mathbf{x} = (x_0, x_1, \dots, x_{N-1})$ and $\mathbf{y} = (y_0, y_1, \dots, y_{N-1})$, two hypercomplex vectors of length N . The hypercomplex cross-correlation operation of $\mathbf{x}$ and $\mathbf{y}$ is defined to the hypercomplex vector $\mathbf{z} = (z_0, z_1, \dots, z_{N-1})$:

$$z_l = \sum_{n=0}^{N-1} [x(n+l) \bmod N] \overline{y(n)} \quad l = 0, 1, 2, \dots, N-1,$$

(A.1)

where $\overline{y(n)}$ is the conjugate of $y(n)$, the shift operation on $x(n)$ is carried out cyclically using modulo arithmetic for the subtraction.

A.2 Algorithm for pairwise DNA sequence alignment with hypercomplex cross-correlation

The algorithm for pairwise DNA sequence alignment with hypercomplex cross-correlation is as follows, and the schematic graph is shown in Figure A.1.

Appendix A

Algorithm 1: Given two DNA sequences X and Y of length N and P respectively, calculate pairwise alignment between X and Y with hypercomplex cross-correlation:

Step 1. Transform the two DNA sequences X and Y into hypercomplex vectors

$$\mathbf{x} = (x_0, x_1, \dots, x_{N-1}) \text{ and } \mathbf{y} = (y_0, y_1, \dots, y_{P-1}).$$

Step 2. Pad $\mathbf{x}$ on the left with $P-1$ 0's and $\mathbf{y}$ on the right with $N-1$ 0's to form two vectors $\mathbf{x}' = (0, 0, \dots, x_0, x_1, \dots, x_{N-1})$ and $\mathbf{y}' = (y_0, y_1, \dots, y_{P-1}, 0, 0, \dots, 0)$ of length $N + P - 1$.

Step 3. Calculate the cyclic cross-correlation $\mathbf{z}$ between $\mathbf{x}'$ and $\mathbf{y}'$ by equation (A.1).

Step 4. Plot the real part $\mathbf{r} = (r_0, r_1, \dots, r_{N+P-1})$ of $\mathbf{z}$ against the shift l .

Step 5. Change the horizontal coordinate from shift l to relative position m by setting $m = l - (p-1)$. A peak in the real part of hypercomplex cross-correlation indicates similarity between two DNA sequences. The relative position m corresponding to the maximum of $\mathbf{r}$ is the position that corresponds to the maximal similarity.

Appendix A

Figure A.1 Sketch for pairwise DNA sequence alignment by hypercomplex cross-correlation. (a): X and Y are two DNA sequences of length N and P respectively; m is the relative position between X and Y. It is set as 0 when X_0 are aligned with Y_0 ; the real part of hypercomplex cross-correlation is actually equal to the match count of each diagonal in the graph of (a). (b): x and y are hypercomplex representation of DNA sequence X and Y. Before doing hypercomplex cross-correlation, x is padded on the left with $P-1$ 0's and y on the right with $N-1$ 0's to form two vectors $\mathbf{x}' = (0, 0, \dots, x_0, x_1, \dots, x_{N-1})$ and $\mathbf{y}' = (y_0, y_1, \dots, y_{P-1}, 0, 0, \dots, 0)$ of length $N+P-1$.

A.3 Algorithm for consensus sequence based motif scanning with hypercomplex cross-correlation

The algorithm for DNA motif scanning with hypercomplex cross-correlation is as follows:

Algorithm 2: Given a DNA sequence X of length N and a consensus sequence Y of length P , scan DNA sequence X to identify all the similar segment of X to the consensus sequence Y .

Step 1. Transform the two DNA sequences X and Y into hypercomplex vectors

$$\mathbf{x} = (x_0, x_1, \dots, x_{N-1}) \text{ and } \mathbf{y} = (y_0, y_1, \dots, y_{P-1}).$$

Step 2. Pad $\mathbf{x}$ on the left with $P-1$ 0's and $\mathbf{y}$ on the right with $N-1$ 0's to form two

$$\text{vectors } \mathbf{x}' = (0, 0, \dots, x_0, x_1, \dots, x_{N-1}) \text{ and } \mathbf{y}' = (y_0, y_1, \dots, y_{P-1}, 0, 0, \dots, 0) \text{ of length}$$

$$N+P-1.$$

Step 3. Calculate the cyclic cross correlation $\mathbf{z}$ between $\mathbf{x}'$ and $\mathbf{y}'$ by equation (A.1).

Step 4. Plot the real part $\mathbf{r} = (r_{P-1}, r_{P+1}, \dots, r_N)$ of the $\mathbf{z}$ against shift l .

Step 5. Adjust the horizontal coordinate from shift l to position m of DNA sequence X

by setting $m = l - (p-1)$.

The position corresponding to the maximum of r is the start position of the potential binding site.

A.4 Implementation hypercomplex cross-correlation by hypercomplex Fourier transform

The hypercomplex cross-correlation can be implemented by direct evaluation of the summation in Equation (A.1), which requires running time $O(N^2)$. The cost of performing evaluation of cross-correlation can be reduced to $O(N \log N)$ by the hypercomplex Fourier transforms.

A.4.1. Polar Form of hypercomplex number

A hypercomplex number can be represented in the polar form by generalizing the Euler's formula for complex numbers. That is:

$$Q = |Q|e^{\mu\theta} = |Q|(\cos \theta + \mu \sin \theta), \quad (\text{A.2})$$

where

$$\cos \theta = \frac{r}{|Q|}, \quad \sin \theta = \frac{\sqrt{x^2 + y^2 + z^2}}{|Q|}. \quad (\text{A.3})$$

Then μ can be expressed as

$$\mu = \frac{x}{\sqrt{x^2 + y^2 + z^2}}i + \frac{y}{\sqrt{x^2 + y^2 + z^2}}j + \frac{z}{\sqrt{x^2 + y^2 + z^2}}, \quad (\text{A.4})$$

$$\mu^2 = -1,$$

μ is a unit pure hypercomplex number, and it is referred to the eigen-axis. Note that μ represents the direction in the 3-D space of imaginary part of hypercomplex number and θ is referred to the eigen-angle.

A.4.2. Definition of hypercomplex Fourier Transform (HFT)

The discrete hypercomplex Fourier Transform was generalized from discrete complex Fourier Transform. Due to the non-commutativity of hypercomplex number multiplication, there are three types definition of hypercomplex Fourier transform (Pei S.-C. et al 2001), here we use the right-side form HFT defined as:

$$X(\kappa) = HFT\{x(n)\} = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) e^{-\mu_1 \frac{2\pi}{N} \kappa n}. \quad (\text{A.5})$$

Its inverse discrete hypercomplex Fourier Transform is given by

$$x(n) = IHFT\{X(\kappa)\} = \frac{1}{\sqrt{N}} \sum_{\kappa=0}^{N-1} X(\kappa) e^{\mu_1 \frac{2\pi}{N} \kappa n}, \quad (\text{A.6})$$

where the vector μ_1 is called the axis of the transform, and it is an arbitrary unit pure hypercomplex number. For computing convenience, we choose vector i as the transform axis here after.

A.4.3 Implementation of hypercomplex Fourier Transform by Fourier Transform

Assume the input hypercomplex series $\mathbf{x}(n)$:

$$x(n) = x_r(n) + x_i(n) \cdot i + x_j(n) \cdot j + x_k(n) \cdot k$$

and a given transform axis is i , the hypercomplex Fourier Transform $X(\kappa)$ is calculated as:

Appendix A

$$X(\kappa) = HFT\{x(n)\} = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x(n) e^{-i\frac{2\pi}{N}\kappa n} = FT\{x_a(n)\} + IFT\{x_b(n)\} \cdot j, \quad (\text{A.7})$$

where $x_a(n) = x_r(n) + x_i(n) \cdot i$, $x_b(n) = x_j(n) + x_k(n) \cdot i$ and $x(n) = x_a(n) + x_b(n) \cdot j$.

A.4.4. Implementation of hypercomplex cross-correlation by hypercomplex Fourier Transform

Assume two hypercomplex vector $\mathbf{x}(n)$ and $\mathbf{y}(n)$, hypercomplex cross correlation can be calculated by the following equation (Pei S.-C. et al 2001):

$$z = IHFT\{X(\kappa)\overline{Y_a(\kappa)} - X(-\kappa) \cdot jY_b(-\kappa)\} \quad (\text{A.8})$$

where $X(\kappa) = HFT\{x(n)\}$, $Y_a(\kappa) = HFT\{y_a(n)\}$, $Y_b(-\kappa) = HFT\{y_b(-n)\}$

$X(-\kappa) = HFT\{x(-n)\}$, $y_a(n) = y_r(n) + y_i(n) \cdot i$, $y_b(n) = y_j(n) - y_k(n) \cdot i$, and

$y(n) = y_a(n) + j \cdot y_b(n)$.

By using hypercomplex Fourier transform, the computational complexity of algorithm 1 and 2 is reduced to $O(N \log N)$.

Appendix B

Results on fat-tail test

Appendix B

TableB.1 Fat-tail test kurtosis coefficient $F\text{-}K_{50}$ and $F\text{-}KK$ for the 180 sequences in training data sets of chapter 4.

No.	CRMs		EXONs		NCNRs	
	$F\text{-}K_{50}$	$F\text{-}KK$	$F\text{-}K_{50}$	$F\text{-}KK$	$F\text{-}K_{50}$	$F\text{-}KK$
1	0.073053	1.5044	-1.2036	1.2499	6.2935	4.4553
2	29.236	14.93	0.37819	1.9922	4.441	7.3357
3	4.4959	4.059	-0.57746	0.83137	1.8912	2.2381
4	2.6489	3.9071	2.55	2.2599	-0.30047	0.47316
5	10.732	10.693	0.048882	1.833	4.2864	2.8823
6	7.2916	5.6874	-0.05085	1.0365	7.4927	4.0776
7	6.3139	5.4326	-0.51554	1.2237	4.0492	3.8331
8	4.7446	3.6087	-1.2253	0.057493	4.0504	2.7836
9	6.1223	5.748	0.98007	4.2541	0.38603	3.6237
10	4.1096	3.4828	-0.68511	0.13735	-0.43329	0.37557
11	8.1653	9.8968	0.19948	0.74552	0.99074	1.6891
12	2.7945	2.2421	-0.41245	1.377	3.8207	5.1832
13	3.1391	4.1766	-0.33014	0.87454	2.9681	2.3981
14	8.4548	5.5626	1.8996	1.6564	3.2972	3.521
15	4.8521	2.957	-0.35969	1.0803	-0.26578	0.46889
16	2.8262	3.1983	0.96777	1.6203	2.4161	2.5726
17	-0.61681	0.54082	0.12746	1.3744	1.6045	1.477
18	7.2534	13.157	0.34791	0.67671	2.1822	3.2306
19	2.5022	1.5881	6.0791	6.1417	5.0203	4.0438
20	-0.45418	0.37714	2.4674	2.4121	2.2484	3.2645
21	7.2424	8.6258	2.4341	4.764	-0.67658	0.33748
22	-0.19367	0.52668	1.0096	0.58901	1.1356	0.99152
23	0.24016	0.60066	2.962	4.4579	-1.0957	0.31076
24	1.6276	2.7433	0.8665	1.7352	-0.8695	0.18192
25	1.9705	2.9364	-1.214	-0.02678	4.692	2.8188
26	0.19624	0.9167	1.7452	3.5419	1.2067	0.98145
27	-0.78411	0.18109	0.21779	2.571	1.7408	1.4452
28	3.0157	1.9092	0.42443	2.797	2.0178	7.3272

Appendix B

29	5.1807	4.5471	-0.78773	-0.00699	-0.61834	0.18008
30	18.451	12.84	1.0942	1.7562	0.43944	1.5319
31	4.2462	3.1695	-0.83117	0.1913	7.9317	7.3883
32	4.5357	3.9446	-0.26145	0.63653	-0.32207	0.32507
33	18.551	24.924	-0.44874	0.50767	1.5788	1.1804
34	5.6384	3.6774	1.0435	1.0439	3.1748	2.963
35	2.466	2.2078	-0.22165	0.49856	6.1677	6.7729
36	3.1757	3.6309	1.4631	1.1323	-0.25074	0.62347
37	11.002	6.148	-1.3357	0.088919	0.9966	2.5224
38	7.1659	4.2672	-0.45094	0.4357	8.7081	5.6112
39	3.3005	3.1021	3.4918	1.5366	3.3014	6.4561
40	19.175	13.696	1.1181	0.99777	1.587	3.0116
41	31.301	12.469	-0.57912	0.44533	-0.8162	2.6213
42	8.2375	6.4099	-0.73351	0.19945	-0.09142	2.5741
43	2.991	2.372	-0.60742	0.27183	0.159	5.5501
44	-0.89755	0.17032	-0.00088	1.4491	0.51382	4.1941
45	0.39992	0.63012	-0.84978	0.018977	-0.67004	2.7327
46	19.489	8.6011	0.89271	1.8161	0.3946	4.2546
47	2.2239	1.1069	-0.24809	0.54821	2.9458	6.9858
48	1.594	1.5946	-0.67901	-0.06596	4.4931	3.789
49	1.1977	1.0306	-0.90538	-0.06811	-0.45604	1.5668
50	4.4935	2.7218	1.3566	1.1028	-0.05187	1.8668
51	1.1096	1.1397	7.2767	6.6351	1.1095	1.5911
52	5.6883	8.6743	0.10594	0.61768	-0.36207	1.889
53	7.408	4.5675	-0.73594	2.0151	-0.86202	1.3253
54	8.3875	6.7035	-0.96675	0.071557	0.036962	1.7637
55	4.057	2.8412	-0.87522	0.057587	-0.07297	1.0003
56	8.0234	8.1831	-0.22347	1.2613	-0.51969	1.2674
57	8.5664	10.955	-0.726	0.15128	-0.90127	1.0161
58	6.942	5.4259	-0.27507	0.47411	-1.0842	0.82963
59	3.7068	3.7668	-0.14915	0.49918	-0.43166	0.69451
60	10.142	9.8075	-1.0351	0.16759	-0.11446	1.8668

Appendix B

Table B.2 60 CDS of *Drosophila melanogaster*

No.	Gene name	Length	CAI	GC	GC3	No.	Gene	Length	CAI	GC	GC3
1	CG18111	354	0.799	0.603	0.895	31	CG7592	450	0.709	0.580	0.867
2	Lsp1beta	2370	0.813	0.555	0.932	32	CG1475	354	0.636	0.612	0.835
3	CG2852	363	0.734	0.605	0.854	33	CG1883	585	0.709	0.586	0.862
4	CG6783	393	0.717	0.585	0.873	34	CG11151	348	0.704	0.583	0.888
5	Cd1c2	270	0.771	0.559	0.889	35	CG3751	396	0.699	0.601	0.864
6	Eno	1302	0.773	0.610	0.876	36	CG11086	492	0.696	0.638	0.909
7	CG15829	249	0.776	0.558	0.880	37	CG7998	1011	0.696	0.608	0.822
8	oho23B	252	0.773	0.595	0.845	38	RpL18A	534	0.698	0.586	0.876
9	Tpi	744	0.607	0.609	0.828	39	CG5548	354	0.692	0.638	0.941
10	Chd64	528	0.708	0.510	0.905	40	CG4111	366	0.692	0.575	0.887
11	CG5804	249	0.764	0.574	0.831	41	Hsp60	1722	0.675	0.612	0.807
12	CG18594	531	0.771	0.606	0.876	42	Act5C	1131	0.668	0.560	0.790
13	Dbi	261	0.76	0.605	0.874	43	RpS6	654	0.682	0.604	0.794
14	Argk	1071	0.495	0.578	0.709	44	Jafrac1	585	0.675	0.591	0.856
15	RpS18	459	0.744	0.577	0.824	45	CG8332	447	0.664	0.568	0.799
16	Mp20	555	0.742	0.587	0.822	46	RpL36	348	0.662	0.583	0.879
17	CG6846	450	0.743	0.618	0.893	47	RpS17	396	0.669	0.576	0.833
18	CG14746	558	0.733	0.642	0.914	48	RpP2	342	0.659	0.629	0.798
19	CG14028	240	0.735	0.607	0.833	49	GlyP	2535	0.653	0.573	0.833
20	Ef1gamma	1293	0.731	0.560	0.894	50	CG3731	1149	0.618	0.607	0.813
21	Ald	1086	0.735	0.621	0.818	51	CG8628	255	0.656	0.565	0.835
22	CG2852	618	0.734	0.605	0.854	52	CG8756	1596	0.633	0.608	0.844
23	CG2297	432	0.735	0.602	0.882	53	CG3861	1395	0.585	0.601	0.807
24	RpL3	1212	0.712	0.589	0.825	54	CG5468	867	0.657	0.628	0.737
25	CG11522	789	0.729	0.606	0.893	55	CG8579	816	0.647	0.591	0.816
26	CG18730	1482	0.721	0.628	0.885	56	CG15093	975	0.641	0.599	0.831
27	Amy-d	1482	0.719	0.630	0.889	57	CG6295	1017	0.64	0.588	0.776
28	CG8577	558	0.711	0.636	0.898	58	CG6467	816	0.652	0.621	0.809
29	blw	1659	0.711	0.606	0.787	59	CG1153	867	0.652	0.634	0.820
30	CG18111	354	0.799	0.628	0.872	60	CG9354	507	0.653	0.647	0.828

Appendix B

CAI is Codon Adaptation Index (CAI). It is calculated by algorithm published by Carbone et.al. (2003). CAI ranges from 0 to 1, higher value of CAI means high expression level of genes.

GC is the percentage of nucleotide G and C appearing in the sequence.

GC3 is the percentage of nucleotide G and C appearing in the third position of codon.

Appendix B

Table B.3 Kurtosis k , Kurtosis coefficient, $F-K_{50}$ and $F-KK$ for the 60 CDS of *Drosophila melanogaster* listed in table B.2

No.	Gene name	k	$F-KK$	$F-K_{50}$	No.	Gene	k	$F-KK$	$F-K_{50}$
1	CG18111	-0.573	0.002	-1.195	31	CG7592	-0.1703	0.345	-0.561
2	Lsp1beta	0.327	0.928	-0.356	32	CG1475	-0.1443	0.352	-0.836
3	CG2852	-0.466	0.031	-1.097	33	CG1883	-0.568	-0.076	-1.140
4	CG6783	0.769	1.124	0.195	34	CG11151	0.0687	0.555	0.0371
5	Cd1c2	0.557	0.910	0.468	35	CG3751	0.506	0.935	-0.156
6	Eno	-0.183	0.284	-0.832	36	CG11086	-0.330	0.194	-1.114
7	CG15829	0.195	0.633	0.152	37	CG7998	-0.553	-0.134	-1.428
8	oho23B	-0.034	0.475	-0.162	38	RpL18A	-0.271	0.237	-0.888
9	Tpi	0.158	0.687	-0.154	39	CG5548	0.165	0.635	-0.281
10	Chd64	-0.414	0.101	-0.853	40	CG4111	0.434	0.848	0.541
11	CG5804	-0.022	0.485	-0.369	41	Hsp60	-0.413	-0.015	-2.000
12	CG18594	-0.410	0.113	-1.092	42	Act5C	-0.151	0.317	-1.015
13	Dbi	-0.985	-0.204	-1.414	43	RpS6	-0.019	0.481	-0.655
14	Argk	0.425	1.056	-0.100	44	Jafrac1	-0.207	0.287	-0.908
15	RpS18	0.202	0.684	0.194	45	CG8332	0.886	1.301	1.265
16	Mp20	0.134	0.632	-0.061	46	RpL36	-0.017	0.485	-0.298
17	CG6846	-0.312	0.226	-0.607	47	RpS17	0.257	0.721	-0.289
18	CG14746	-0.582	-0.058	-1.346	48	RpP2	-0.015	0.488	-0.878
19	CG14028	-0.089	0.437	-0.997	49	GlyP	-0.486	-0.179	-1.173
20	Ef1gamma	-0.227	0.231	-0.809	50	CG3731	-0.278	0.144	-1.156
21	Ald	-0.026	0.472	-0.843	51	CG8628	-0.291	0.297	-0.944
22	CG2852	-0.467	0.031	-1.445	52	CG8756	-0.038	0.450	-0.973
23	CG2297	-0.809	-0.203	-1.687	53	CG3861	0.005	0.507	-0.708
24	RpL3	0.222	0.761	-0.203	54	CG5468	-0.031	0.468	-0.892
25	CG11522	-0.407	0.075	-0.989	55	CG8579	0.616	1.188	-0.127
26	CG18730	0.142	0.676	-1.267	56	CG15093	-0.380	0.070	-1.573
27	Amy-d	0.131	0.663	-1.315	57	CG6295	-0.025	0.470	-0.709
28	CG8577	-0.585	-0.066	-1.381	58	CG6467	0.354	0.890	-0.761
29	blw	-0.308	0.107	-1.288	59	CG1153	0.170	0.690	-0.810
30	CG18111	-0.122	0.408	-0.763	60	CG9354	-0.351	0.190	-0.719

Figure B.1 Histograms of kurtosis coefficient $F-KK$ for the 60 CDS shown in Table B.2 (Visualizing the result in Table B.3). The X axis shows the kurtosis coefficient $F-KK$; the Y axis is the number of sequences in the data set with this $F-KK$.

Figure B.2 Histograms for the kurtosis coefficient $F-K_{50}$ for the 60 CDS shown in Table B.2 (Visualizing the result in Table B.3). The X axis shows the kurtosis coefficient $F-K_{50}$; the Y axis is the number of sequences in the data set with this $F-K_{50}$.

Appendix B

Table B.4 Regulators (transcription factors) for each cis-regulatory modules of *Drosophila melanogaster* in the CRMs training data set of chapter 4

No.	Comment	Regulators (TF)	Reference
1	btd	Bcd, Hb, Tll, Dll	(Schock, et al., 2000)
2	gt	Hb, Kr, Cad	(Rivera-Pomar, et al., 1995)
3	hb_enc	Hb, Tll	(Wimmer, et al., 2000)
4	hb_P1	–	
5	kni	Bcd, Hb, Kr, Cad	(Pankratz, et al., 1992)
6	kr_late	–	(Capovilla, et al., 1992) (Hoch, et al., 1992)
7	kr_CD1	Bcd, Hb, Gt, Kni, Tll	
8	kr_CD2	Gt	
9	kr_AD2	–	
10	otd_early	Bcd	(Finkelstein and Perrimon, 1990)
11	otd_eye	–	
12	sal_blastodermal_early	Bcd, Hb, Kr, Cad, Hkb	(Guss, et al., 2001)
13	sal_tracheal	–	(de Celis, et al., 1999)
14	tll_brain_specific	–	(Rudolph, et al., 1997) (Liaw, et al., 1995)
15	tll_PD	Bcd, Cad	
16	tll_AD_PD	Bcd	
17	eve_late1	Prd (PD), Prd (HD)	(Arnosti, et al., 1996; Fujioka, et al., 1999; Ludwig, et al., 2000; Miskiewicz, et al., 1996; Stanojevic, et al., 1991)
18	eve_stripe3+7	Hb, Kni	
19	eve_stripe2	Bcd, Hb, Kr, Gt	
20	eve_late2	Eve	
21	eve_stripe4+6	Hb, Kni	
22	eve_stripe1	Bcd, Hb, Kr	
23	eve_stripe5	Bcd, Hb	
24	ftz_upstream	Ftz, Ttk, Ftz-F1	(Han, et al., 1998; Nasiadka, et al., 2000; Tsai and Gergen, 1995)
25	ftz_neurogenic	–	
26	ftz_zebra	Cad, Ttk, Ftz-F1	
27	gsb_early	Eve, Prd (HD)	(Bouchard, et al., 2000; Davis, et al., 2001)
28	gsb_late	Wg, Opa	
29	hairy_stripe3+4	–	(Hader, et al., 1998; La Rosee-

Appendix B

30	hairy_stripe7	Kr, Hb, Kni, Cad, Tll	
31	hairy_stripe6+2	Kr, Kni, Cad	
32	hairy_stripe6	—	
33	hairy_stripe5+1	Kr, Gt, Kni	
34	hairy_stripe1	—	
35	prd_stripe1	Eve, Gt	(Bertuccioli, et al., 1996;
36	prd_stripeP	Ftz, Eve	Epstein, et al., 1994; Xue and
37	prd_glands	—	Noll, 2002)
38	run_stripe1+7	—	(Butler, et al., 1992; Kania, et
39	run_stripe5	Kr, Gt	al., 1990; Klingler, et al., 1996)
40	run_stripe3+7	Hb, Kni	
41	run_late7stripe	Eve, Ftz, Hairy, Run	
42	run_late6stripe	—	
43	en_upstream_stripe	—	(Gibert, et al., 2002; Orihara, et
44	en_stripe	Cad, Ftz, Ftz-F1, Wg	al., 1999)
45	Abd_A_PREiab2	Pho	(Karch, et al., 1990; Shimell, et
46	Abd_A_iab2	Hb, Kr, Gt, Eve, Pho	al., 2000)
47	dll_207	—	(Morasso, et al., 1996; O'Hara,
48	dll_215	—	et al., 1993; Vachon, et al.,
49	dll_208	—	1992)
50	dll_ETD2b	—	
51	dll_ETD6	—	
52	ems_upstream	Abd-B, Dfd	(Jones and McGinnis, 1993)
53	ubx_PBX	Hb, Ftz, En, Twi	(Irvine, et al., 1993; Laney and
54	ubx_dorsal	—	Biggin, 1997; Lewis, et al.,
55	ubx_ventral	—	1995; Martin, et al., 1995; Qian,
56	ubx_zebra(like)_S2	Hb	et al., 1993; TenHarmsel, et al.,
57	ubx_PRE_polycomb	—	1993)
58	ubx_zebra(like)_S1	Hb	
59	ubx_BXD	Hb, Eve, En, Zeste, Abd-B	
60	ubx_BX	Hb, Ftz, En, Twi, Tll	

Appendix B

Bcd - Bicoid, Cad - Caudal, Dfd - Deformed, En - Engrailed, Eve - Eve-skipped, Ftz - FTZ, Gt - Giant, Hairy - Hairy, Hb - Hunchback, Kni - Knirps, Kr - Kruppel, Hkb - Huckebein, PHO - Pleiohomeotic, Prd (HD) - paired HD, Prd (PD) - paired PD, Tll - Tailless, Ttk - Tramtrack, Twi - Twist

Table B.5 Motif and corresponding PFM for the main TFBS of the CRMs in the CRMs training data set (corresponding TF see Table B.4)

Bcd	Cad	Dfd	En	Eve	Ftz	Gt	Hairy	Hb
CCTAATCCC	TTTTTATGGTCG	TGATTAATGA	AATTATGATTG	ATTTGCTGCG	TGTAATTGCT	CATTATTACTCTATTTT	ACACGGCGG	TTTTTTTTG
TCTAATCCC	CTTTTATGGCCC	AGATTAATTA	TTTAATGGCGG	ATTTGCTGCT	CCTAATTGCTG	CATTATTATGTGTTTTT	GCACGCCCC	TTTTTTTTG
GCTAATCCC	TTTTTATGTCTT	TGATTAATTA	GCTAATGAGCG	GTTTGGGTT	GTTAATTGAC	GATTATTAGTCAATTGC	GCACGCCCC	ATTTTTTG
ACTAATCCC	TTTTTGATGGCGG	AGATTAATGA	GCTAATTTGSC	GTGTGCCCCG	TTTAAATTGAC	CTTTATTGCAGCATCTT	ATTTTTTG	ATTTTTTG
CGTAATCCC	TTTTTTAGGGAAC	TGATTAATGG	GTTCCAGAGCC	AATTGGGTTG	GCTAATTGGC	TTTTATTGCAGCATCTT	TCTCGCGAC	ATTTTTTG
TATAATCCC	CGTTTATGGTTC	TGATTAATGA		AATGGCCCCCT	TTTAAATGGCC	CATTATGCGTTTGTGTG		GTTTTTTG
CCTAATCCA	ACTTTATGTGAC			CGGTGCTGAG	TTTGAATTGCC	AATTTTACGTAATATT		GTTTTTTG
GGTAATCCG	GATTTATGACAC			CGGTGCTAAC	TCGAATTGTC	GATTCCTGCGTCATAAA		GTTTTTTG
CGTAATCCT	TCTTTATGATTA			TCGCGCTGAA	GTTAAGTGCT	TAATTTTACGTAACATT		ATTTTTTG
CATAATCCA	ACTTTTGGGCCA				GTTAAGTGCC			ATTTTTTG
ATTAATCCG	TTTTTGTTGGCCA				ATTAATTCCCT			GATTTTTTG
GGTAATCCT					TCTAATTAGC			TTACTTTTG
GTTAATCCG					TTTAAATTTGT			CTTCTTTTG
TTTAAATCCG					CATAATTTTT			TTTTTAATG
AATAATCCA					GGTAATATAG			TTTTTAATG
CTTAATCCG					GCTAATAAAA			TTTTTAATG
CTTAATCCT					GCTAATGAGC			TTTTTAATG
TTAAATCCC					GTTAATGATC			TTTTTAATG
CCAAATCCA					GGTAAAAGAT			TTTTTAATG
CTAAATCCG					TCATCATGTGT			TTTTTAATG
ACAAATCCT					TTTTTATGACC			TTTTTAATG
GCAAATCCG					TTTTTATGTGT			TTTTTAATG
GCAAATCCG								TTTTTTAAG
CGTGATCCT								TTTTTTAAG
GATAAGCCG								TTTTTTAAG
TTTAAACCCT								TTTTTTAAG
CTTAACCCCT								TTTTTTAAG
GCTAAGCTG								TTTTTTAAG
GCTAAGCTC								TTTTTTAAG
AATAATCTC								TTTTTTAAG
TATAATCGC								TTTTTTAAG
CTTAATCAC								TTTTTTAAG
GTTAATCTC								TTTTTTAAG
GCTAATCCG								TTTTTTAAG

Kni	Kr	Pho	Prd (HD)	Prd (PD)	Tll	Ttk	Twi
GGAGGTCGTAA TATGATCGTCT GAGGATCGTAA TGtGAACTTAC TGGGAACAAGA TAGGAACTTAA GAGAAACCTAA TAGAAAAGTAG TAGGAAAGTAG TGGGAACGCGG GGGGATCGCTG TGAGAACCCGA ACTGAACGGTC CTGGAACAATA ACTGAACATAA TAAGGACATCT	AAACCCCTTT AAACCCCTTT GAACCCCTTT TTACCCCTTT TCACCCCTTT TCACCCATTT CCATCCCTTT AATCCCTTC CACTCCCTTT CACTCCCTTC AAATCCCTTT TAATCCGTT TAATCCGTG CAACACTTT TAACCCGTT CAACCCGGT TAACCCAAT TAACCTCAGC TAATCTCGC TAATCGCAC TAACCCCTAA	TAGCCATGTTGGA CAGCCATCTCGCT TAGCCATTTGCGG CGGCCATTTGCGGG	TGGCTAATTGC GTCATAATTGC GTGTTAATGGT CTGTCAATTTC GTTATAAATTTA TGAGTAATTTG GCGGTAATTTA	GTCACGGTT GACACGGTT CTCACCGTG GACACGCCA ATCACTGAC TTTACGCAT TTCACGCTT	TAAAGTAGAA TTAAAGTAAAA AAAAGTTAAT TTAAAGTTAAA AGAAAGTCAAA GAAAAGTCAAT ATAAGTCAAG ACCAGTCAAT CAAGGTCAAA CGAGGTCATA CAAGGTCAGG AGAACTGACT AATATTGACG AAGATTAAAA TCAATTAAAA ACAAATTAAAA AAAATTAATA AAAATTTAAT AAAACTGCAC AAGAATCCTA AATAATCCAG ATAAATCCAA	TGCAAGGACA TGCCAGGACC GCCAAGGACA TCCAAGGACG GTCAAGGATG CGGCAGGACA GCACAGGACC	ACAGCGTGTG AGAGCCGATG ACAGCCCTTTG GTTGCCCTTTT ACAGCCGTCA TCATGTGTTG GCACACATTT

Appendix B

Bcd

A	5	6	6	33	34	0	0	1	4
T	6	11	28	0	0	29	0	4	7
G	11	5	0	1	0	3	0	2	11
C	12	12	0	0	0	2	34	27	12

Cad

A	2	1	0	0	0	9	0	0	2	1	3	3
T	6	6	11	11	9	2	9	0	2	3	3	1
G	1	1	0	0	2	0	2	11	7	1	1	2
C	2	3	0	0	0	0	0	0	0	6	4	5

Dfd

A	2	0	7	0	0	7	7	0	0	6
T	5	0	0	7	7	0	0	7	2	0
G	0	7	0	0	0	0	0	0	5	1
C	0	0	0	0	0	0	0	0	0	0

En

A	1	1	0	3	4	1	0	3	0	0	0
T	1	2	5	1	0	4	1	1	1	1	0
G	3	0	0	0	0	0	4	1	3	1	3
C	0	2	0	1	1	0	0	0	1	3	2

Eve

A	4	2	0	0	0	0	0	1	3	1
T	1	4	5	7	0	0	5	0	2	3
G	2	2	4	1	9	0	2	6	0	4
C	2	1	0	1	0	9	2	2	4	1

Ftz

A	1	1	0	19	22	1	3	5	5	1
T	10	11	20	2	0	19	13	5	4	8
G	9	3	2	0	0	2	6	11	6	2
C	2	7	0	1	0	0	0	1	7	11

Appendix B

Gt

A	1	7	1	0	6	0	0	5	0	2	0	3	7	0	3	1	1
T	2	2	8	9	2	9	8	0	1	3	5	2	1	8	3	7	6
G	2	0	0	0	0	0	1	3	2	4	2	1	1	0	1	1	1
C	4	0	0	0	1	0	0	1	6	0	2	3	0	1	2	0	1

Hairy

A	1	0	3	0	0	0	0	1	0
T	1	0	1	0	0	0	0	0	0
G	2	0	0	0	4	0	2	0	2
C	0	4	0	4	0	4	2	3	2

Hb

A	4	1	1	0	2	17	7	0
T	26	33	34	33	33	15	23	0
G	4	0	0	0	0	0	0	35
C	1	1	0	2	0	3	5	0

Kni

A	2	7	3	2	14	12	2	3	3	8	8
T	9	1	4	0	0	4	0	3	9	3	2
G	4	6	9	14	2	0	0	7	1	3	4
C	1	2	0	0	0	0	14	3	3	2	2

Kr

A	4	17	18	0	1	0	3	3	1
T	11	1	1	8	1	1	12	15	14
G	1	0	0	0	0	1	4	3	1
C	5	3	2	13	19	19	2	0	5

Pho

A	0	3	0	0	0	4	0	0	0	0	0	1
T	2	0	0	0	0	0	4	2	2	1	0	1
G	0	1	4	0	0	0	0	1	2	1	3	2
C	2	0	0	4	4	0	0	1	0	2	1	0

Appendix B

Prd(HD)

A	0	0	1	2	0	7	7	0	0	0	2
T	2	4	1	2	6	0	0	7	6	4	1
G	4	2	4	2	0	0	0	0	1	3	1
C	1	1	1	1	1	0	0	0	0	0	3

Prd(PD)

A	1	2	0	7	0	0	0	2	1
T	2	5	1	0	0	1	0	4	4
G	3	0	0	0	0	5	4	0	1
C	1	0	6	0	7	1	3	1	1

Tll

A	14	12	17	19	3	0	6	17	16	12
T	4	4	2	0	6	22	3	0	3	5
G	1	3	2	3	11	0	3	1	1	4
C	3	3	1	0	2	0	10	4	2	1

Ttk

A	0	0	1	4	7	0	0	7	0	3
T	3	1	0	0	0	0	0	0	1	0
G	3	3	1	0	0	7	7	0	0	2
C	1	3	5	3	0	0	0	0	6	2

Twi

A	4	0	6	0	1	0	1	1	0	1
T	1	1	1	1	0	1	3	5	6	2
G	2	1	0	5	1	1	3	1	0	4
C	0	5	0	1	5	5	0	0	1	0

Table B.6 Correlation between the distribution of similar-word set occurrence and experimentally verified TFBS for the 60 CRMs (Table 4.1)

No.	Comment	The number of occurrences of TFBS realated similar-word set	Q1-Q3	$O_{\max}$	Outside %
1	Btd	6,16,19,19,23,23,25,28,32,36,39, 43	15-26	57	50
2	Gt	15,15,21, 22, 23, 23, 28, 30, 48, 50, 73	14-21	73	81
3	hb_enc	20,36,43,45,46,61	16-26	63	83
4	hb_P1	—			
5	kni	9,12,16,16,17,29, 33,45,52,90	12-21	90	60
6	kr_late	—			
7	kr_CD1	13,16,16,17,18,18,21,21,22,25,25,28,29,30,30,31,34,34,34,36,40,43,53, 56,72,88	13-26	88	57
8	kr_CD2	20,22,26,27,41,42,47,51,52, 55,58,67	17-35	107	67
9	kr_AD2	—			
10	otd_early	35, 40	24-35	69	100
11	otd_eye	—			
12	sal_blastodermal_early	8,12,12,13,14,16	10-15	36	17
13	sal_tracheal	—			
14	tll_brain_specific	—			
15	tll_PD	10,13,16,19,30	8-15	37	60
16	tll_AD_PD	49,58	27-43	94	100
17	eve_late1	16,27	15-23	42	100
18	eve_stripe3+7	4,6, 7, 8,18,33,35	7-12	60	71
19	eve_stripe2	6, 11,11,11,12,12,12,15,15,15,17, 17,17,19, 29	9-14	29	53

20	eve_late2	28,28,29,31,31,38	21-32	52	17
21	eve_stripe4+6	9,9,10,29,36,47	8-14	47	50
22	eve_stripe1	11,12,16,19,19,22,22,24	11-17	31	62
23	eve_stripe5	15,21,22,28,31	14-21	38	60
24	ftz_upstream	21,23,32	22-35	76	33
25	ftz_neurogenic	—			
26	ftz_zebra	11,12,12,13,13,15,15,19	10-16	29	22
27	gsb_early	9,10,10,11,11,12,14	7-11	18	62.5
28	gsb_late	—			
29	hairy_stripe3+4	—			
30	hairy_stripe7	14,17,19,20, 20,20,20,23,23, 28,29,32,32,37, 54, 66,76,112	16-26	112	50
31	hairy_stripe6+2	5,13,13,15,16,17,19,20,23,24,28,36,47	11-19	47	50
32	hairy_stripe6	—			
33	hairy_stripe5+1	17,19,19,21,21,23,24,25, 27,29,29,33,33,34,41,46, 67	21-31	121	81
34	hairy_stripe1	—			
35	prd_stripe1	13,15,15,16,22,23,23,25,28,28,29	13-18	39	70
36	prd_stripeP	19,25,27,31,35,35	15-27	74	66
37	prd_glands	—			
38	run_stripe1+7	—			
39	run_stripe5	8,8,11,12,14,15,18,25, 25,26,29,30	13-22	55	45

40	run_stripe3+7	12,21,25,32,35,38,51,57,88,104,163	27-43	163	73
41	run_late7stripe	61,63,71,79,80,82,102,105,113,124,133	56-76	219	72
42	run_late6stripe	–			
43	en_upstream_stripe	–			
44	en_stripe	11,15,15,18,22,22,22,23,25	11-17	27	78
45	Abd_A_PREiab2	5,8,11,12,14	7-13	27	20
46	Abd_A_iab2	14,18,20,22,22,22,23,24,26,28,30,30, 31,32,32,32,33,33,34, 34,38,40,40,40,42,42,43,43,50,50,51,54,54,74	22-34	103	64
47	dll_207	–			
48	dll_215	–			
49	dll_208	–			
50	dll_ETD2b	–			
51	dll_ETD6	–			
52	ems_upstream	21,23,24,25,29,30	15-25	96	50
53	ubx_PBX	13,14,15,16,20,21,30,31,40,50,50,51,54,54	18-33	98	76
54	ubx_dorsal	–			
55	ubx_ventral	–			
56	ubx_zebra(like)_S2	30,60,64,98	17-34	134	75
57	ubx_PRE_polycomb	–			
58	ubx_zebra(like)_S1	26,51,60,83	18-29	83	75
59	ubx_BXD	14,17,24,28,28,29,32,33,44,45,45,50,63,68,83	19-32	83	73
60	ubx_BX	9,21,22,22,23,24,27,29,30,31,33,35,38,38,42,43,49,55,56,62, 63,75,75	20-33	116	61

Appendix B

Q1: the first quartile for the number of occurrences of similar-word set.

Q3: the third quartile for the number of occurrences of similar-word set.

$O_{\max}$: the number of occurrences of the similar-word set with highest incidence.

Outside %: the percentage of TFBS related similar-word sets whose incidence is outside Q1-Q3.

Appendix C

Results on thin tail test

Appendix C

TableC.1 Thin-tail test kurtosis coefficient $T\text{-}K_{50}$ and $T\text{-}KK$ for the 180 sequences in the training data sets of chapter 5 ($m = 5, j = 1$)

No.	CRM _s		EXON _s		NCNR _s	
	$T\text{-}K_{50}$	$T\text{-}KK$	$T\text{-}K_{50}$	$T\text{-}KK$	$T\text{-}K_{50}$	$T\text{-}KK$
1	-1.3203	0.19216	10.655	3.8101	-0.50115	0.44309
2	3.369	1.8462	2.2684	1.1864	-1.1106	0.35271
3	-2.2295	0.1007	11.354	4.2031	2.0056	1.0365
4	-2.1637	-0.04905	9.5325	3.2179	-0.61001	0.4007
5	1.6593	1.1794	3.0814	1.6638	-1.5476	0.22282
6	-0.43977	0.39129	-0.36308	0.46597	-0.65385	0.41754
7	-0.18345	0.59323	1.5638	1.0649	0.51088	0.8007
8	0.47979	0.74118	-1.8578	0.13775	0.92454	0.93538
9	0.19103	0.67507	1.2283	1.1204	0.85437	0.80158
10	-0.0006	0.54889	0.087149	0.64726	-1.0797	0.29234
11	-0.00145	0.60684	-0.75494	0.3885	0.34588	0.71696
12	-1.55	0.21456	-0.26344	0.54226	-2.2368	0.012228
13	-1.0813	0.25887	1.0187	0.90122	1.6243	1.0912
14	-1.5424	0.12833	19.384	6.2218	-1.1516	0.32061
15	-1.8281	0.010817	-0.17474	0.55292	0.72216	0.88176
16	-0.92697	0.31214	5.7998	2.4349	0.22608	0.67622
17	-0.02693	0.58183	5.2743	2.3197	-0.13996	0.5985
18	-1.0187	0.16106	0.016932	0.68099	-1.1233	0.31311
19	-1.4799	0.15126	0.75407	0.8282	-1.8358	0.069113
20	0.21626	0.66055	0.34976	0.77236	-0.54821	0.39289
21	-1.6419	0.05406	2.8062	1.5052	1.5837	1.1899
22	1.3964	1.1791	2.3767	1.3036	-1.1836	0.21475
23	-1.3819	0.16798	-0.70975	0.40819	0.092908	0.59899
24	0.04558	0.59603	7.8802	3.0308	1.3618	0.96171
25	-0.29724	0.57431	-1.0974	0.28428	-1.5428	0.16666
26	-1.0945	0.31087	-1.7549	0.096642	0.59513	0.88103
27	-0.81441	0.31963	0.60601	0.84269	-1.7545	-0.02311
28	-1.4065	0.18678	0.14897	0.6769	-1.1817	0.13497
29	-0.10518	0.52126	0.58462	0.67737	2.3757	1.3055

Appendix C

30	2.5931	1.4922	-1.8033	0.024514	-0.8469	0.32255
31	-0.54852	0.44992	-0.29819	0.48797	0.73451	0.91304
32	0.6449	0.92395	0.71883	0.78083	-0.68368	0.37707
33	0.8217	0.81973	-0.3066	0.48893	-0.99204	0.33488
34	-0.17211	0.51334	3.7116	1.4954	-1.0847	0.21963
35	-0.65419	0.37413	-0.21644	0.50692	0.21423	0.76064
36	-0.03453	0.60677	2.1694	1.2122	2.7516	1.2583
37	6.6944	2.2883	-1.6035	0.10295	2.521	1.3027
38	-1.6416	0.13829	-0.28916	0.51151	-0.69138	0.36605
39	2.3055	1.0794	2.2576	1.3517	0.30389	0.67904
40	1.648	1.1299	0.99338	0.91853	0.94266	1.0684
41	0.55242	0.74426	-0.35265	0.51132	0.73791	1.0391
42	-2.1182	0.025711	-0.90189	0.30871	1.5568	0.98592
43	-1.8135	0.12071	1.0791	0.84739	1.9964	1.4281
44	-0.38789	0.54759	2.7786	1.5115	-1.4211	0.27562
45	-1.6418	0.094186	-2.012	-0.05245	1.4786	1.2498
46	-1.5983	0.14924	-1.1265	0.33463	1.4958	0.92491
47	-0.66023	0.34538	0.55754	0.68867	-0.62629	0.39695
48	0.16877	0.64466	-0.31035	0.55667	-0.93687	0.42747
49	-0.85107	0.33426	-1.1395	0.26553	0.68837	1.0423
50	0.23592	0.68385	0.057376	0.61263	1.306	1.124
51	-1.0218	0.31225	-0.9866	0.36693	0.25015	0.69374
52	-0.35602	0.43602	0.63933	0.77892	0.57155	0.68985
53	-2.4878	-0.11109	-1.0161	0.31	0.84507	1.082
54	-1.5055	0.1183	-1.0172	0.31515	0.4933	0.81488
55	-0.11399	0.54381	-1.5127	0.13733	0.49204	0.80246
56	-1.4999	0.15602	1.8016	1.253	1.2706	0.97758
57	-1.6961	0.13197	-1.0849	0.26725	0.79668	0.91714
58	-2.2416	-0.01078	0.27436	0.73422	0.7763	0.97501
59	-1.3333	0.21056	1.2372	1.0555	0.051971	0.6551
60	-1.1864	0.26133	-0.55813	0.43862	1.4497	1.124