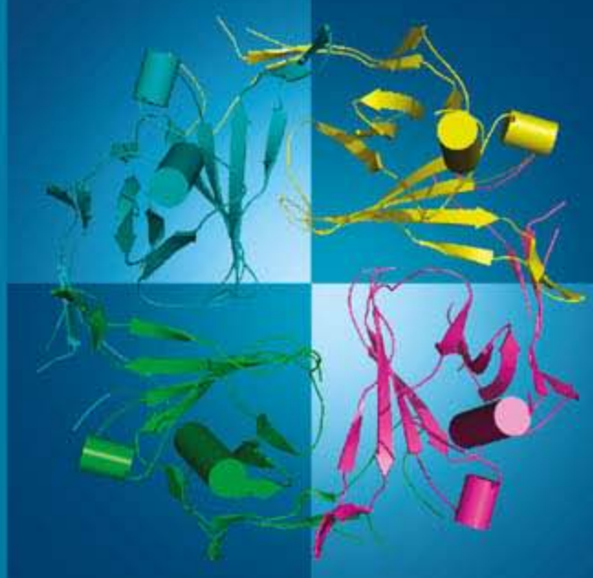
Venkatarajan S. Mathura
Pandjassarame Kangueane



Bioinformatics

A Concept-Based Introduction

Springer

BIOINFORMATICS: A CONCEPT-BASED INTRODUCTION

BIOINFORMATICS: A CONCEPT-BASED INTRODUCTION

Venkatarajan Subramanian Mathura, Ph.D
Bioinformatics Scientist, Roskamp Institute
Sarasota, Florida, USA

and

Pandjassarame Kangueane, Ph.D
Managing Director, Biomed-Informatics
Pondicherry, India

Springer

Venkatarajan S. Mathura
Roskamp Institute
2040 Whitfield Avenue
Sarasota, FL 34243

Pandjassarame Kangueane
Biomed-Informatics
17A, Main Road
Irulan Chandai Annex
Pondicherry 607 402, India

ISBN: 978-0-387-84869-3 e-ISBN: 978-0-387-84870-9
DOI: 10.1007/978-0-387-84870-9

Library of Congress Control Number: 2008932359

Cover illustration: Ribbon diagram of the Cytoplasmic Domain of GProtein-Gated Inward Rectifier Potassium Channel Ki r3.2 (PDB:2e4f) using rendering program pyMOL.

Printed on acid-free paper

springer.com

This book is dedicated to our families

Preface

Scientific disciplines evolve and mature into different areas of specialization to accommodate new knowledge and methods that are being developed by the research community. The last decade has seen a dramatic change in most fields and the information technology has revolutionized several fields. Bioinformatics is the perfect marriage between computer science and advanced biology. Complex biological processes, macromolecular components and their functional interplay define the basis of living cells. Biological experiments that aim to reveal the complexity of cellular systems and biomolecular functions produce huge volumes of data or information that needs to be efficiently handled for tangible results. Exponential increase in genome sequences, protein sequences, protein interactions and biological networks/pathways information has created a demand for efficient information handling. This led to the birth of the field of Bioinformatics that aims to handle biological information using computational methods and algorithms. Bioinformatics is evolving into a mature field with an ever-increasing participation from the scientific community. The past five years have seen a rapid increase in the number of scientific journals in this field. It is impossible to include all the topics of Bioinformatics in a book and still cater to the needs of newcomers attracted to this field. This is an introductory book that provides a balance between computational methods and biological information. Instead of delving in depth, for each topic we provide a broad but necessary content that will benefit readers with different levels of expertise.

Venkatarajan S. Mathura and Pandjassarame Kangueane
Editors

Acknowledgments

We thank Mr. Joseph Burns and Ms. Marcia Kidston of Springer for their help and encouragement. We also thank Ms. Archchana Alagiriswammy for the help with formatting the book. We extend our sincere thanks to all the chapter authors for their contributions and dedication to complete this book.

Venkatarajan S. Mathura and Pandjassarame Kangueane
Editors

Contents

Contributing Authors

Ghania Ait-Ghezala	Roskamp Institute, Sarasota, Florida
Clement Chung	University of Toronto, Toronto, Canada
Andrew Emili	University of Toronto, Toronto, Canada
Pankaj Gupta	Roskamp Institute, Sarasota, Florida
Pingzhao Hu	University of Toronto, Toronto, Canada
Hui Jiang	York University, Toronto, Canada
Deepak N. Kolippakkam	Harvard University, Boston, Massachusets
Venkatarajan S. Mathura	Roskamp Institute, Sarasota, Florida
Michael Mullan	Roskamp Institute, Sarasota, Florida
Pandjassarame Kangueane	Biomedical Informatics, Pondicherry, India
Daniel Paris	Roskamp Institute, Sarasota, Florida
Nikunj Patel	Roskamp Institute, Sarasota, Florida
Fai Poon	Roskamp Institute, Sarasota, Florida
Amita N. Quadros	Roskamp Institute, Sarasota, Florida
Senthilkumar Radhakrishnan	Cal Tech, Pasadena, California
Arun Ramani	University of Texas, Austin, Texas
Meena Sakharkar	Nanyang Technological University, Singapore
Claude-Henry Volmar	Roskamp Institute, Sarasota, Florida

Yiting Yu Nanyang Technological University, Singapore

Michael Q. Zhang CSHL, Cold Spring Harbor, New York

Chaolin Zhang SUNY, Stony Brook, New York

Cui Zhanhua Nanyang Technological University, Singapore

Bing Zhao Nanyang Technological University, Singapore

Hongyi Zhou State University of New York, Buffalo, New York

Yaoqi Zhou State University of New York, Buffalo, New York

Chapter 1

Introduction to Biological Systems

Claude-Henry Volmar, Nikunj Patel, Amita N. Quadros, Daniel Paris, Venkatarajan S. Mathura, and Michael Mullan

Roskamp Institute, 2040 Whitfield Avenue, Sarasota, Florida 34243, USA

Abstract: Living organisms are composed of macromolecules like DNA, RNA, proteins, and carbohydrates that dictate various processes. This chapter provides a glimpse of biological macromolecules and their interplay resulting in biological process or pathways.

Key words: Proteins, Amino acids, DNA, RNA, Cell signaling, Biological systems

1. Molecules of Life

Biochemical molecules such as deoxy ribo nucleic acid (DNA), ribo nucleic acid (RNA), proteins, carbohydrates, and lipids are fundamental for cellular organization and their complex interplay with each other dictates various aspects of living things. They enable a systematic execution of numerous biological processes in a defined manner to maintain life at the cellular level (Kitano, 2002; Noble, 2002). The genetic materials (DNA and RNA) are tightly regulated in organisms. At any given moment, organisms have to deal with different pressures (internal or external) by controlling various biochemical molecules thus maintaining a balance or in other terms, homeostasis. The proper function of biochemical molecules is crucial to the survival of any given organism. Since mutations and other modifications caused by selective pressures are sometimes irreparable, organisms often have to adapt in order to survive and pass on their genes to the next generation. Organisms, therefore, evolve. The mechanisms involved in such a difficult task as maintaining the basic life of an organism are very complex. Regulation at the molecular level is essential for the maintenance

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_1,

of life at the cellular level. Problems at the molecular level often result in physiological alterations that in turn affect homeostasis of the whole organism.

The life of an organism is mapped in its genome, a long sequence of nucleic acids that consists of the entire set of chromosomes of the organism. Genes are a stretch of nucleic acids, which represent a functional aspect of the genome. Each gene codes for a limited set of proteins. The same genes may be found in very distant animals such as a cow and a jellyfish, but their regulation (control of the activity of those genes) may be different and appears to be of utmost importance. Cellular processes such as apoptosis or programmed cell death are encoded within the genome of individual organisms in a complex manner. With the recent sequencing of the human genome, mankind has for the first time the opportunity to attempt to understand the involvement of the genes in sequence of events involved in the development of an organism (ontogenesis) as well as in the etiology of various diseases. RNA is the product of the transcription of DNA and is then translated into polypeptide, which folds into a functional form called protein. Any mutation in DNA, if not repaired by the various polymerases, may result in the transcription of faulty RNA resulting in a wrong protein being translated lacking its original activity. This may cause major problems such as protein aggregation and misfolded proteins, which are not degradable and result in fatal diseases. Naturally occurring single nucleotide polymorphism (SNP) among human population may influence gene function and expression in individuals. Functional variants or genetic changes like SNPs that alter amino acids in proteins, gene expression, and gene splicing are of great interest. The first step of regulation is trying to fix problems at the DNA levels. The next step is to mend at the RNA level through gene splicing and then at the protein level via proteosome/ubiquitin pathways. In eukaryotes, higher-level organism, DNA is transcribed to RNA (Pre-mRNA) that consists of introns and exons. The exons possess the codes that will be translated into proteins whereas the introns are eventually cut out through gene splicing. The resulting RNA is referred to as messenger RNA (mRNA). This messenger RNA may or may not get translated into peptide that folds into a functional protein.

2. Nucleic Acids: DNA Versus RNA

Nucleic acids are made of long chains of nucletotides that consist of nitrogenous base, a sugar moiety, and phosphodieseter connections. Deoxyribonucleic acid (DNA) is basically a sequence of nucleic acids that

exists as a double helix. It consists of the nitrogenous bases **adenine (A), thymine (T), cytosine (C), and guanine (G)** (Watson and Crick, 1953). Adenine and guanine are purines whereas Thymine and Cytosine are pyrimidines (Figure 1.1). In the DNA double helix, purines always pair with pyrimidines by weak hydrogen bonds. This pairing is based on the Watson and Crick complementation of A-T and G-C. This pairing results in a double helix with a constant diameter of 20 Angstrom (Å), with a complete helical turn every 34 Å, and consists of 10 bases per turn. Each branch of the DNA double helix consists of a stretch of nucleotides (nitrogenous bases attached to a sugar phosphate backbone). The two branches are then held together via hydrogen bonds between purines and pyrimidines that are on opposite sides.

This knowledge was crucial in understanding the process of heredity. DNA has a semi-conservative replication (Meselson and Stahl, 1958). The double helix opens up (in a fork-like fashion) and each strand serves as a parental template for replication of the DNA. The replication occurs from 5' to 3' by DNA polymerase. Each daughter strand ends up being the

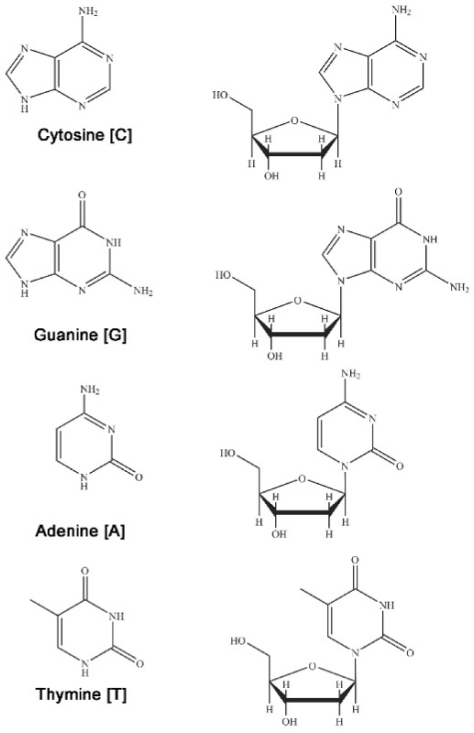


Figure 1.1 Nitrogenous bases and nucleotides in DNA.

complement of a parental strand. Subsequently, each replicated DNA fragment has one parental strand and one daughter strand, hence the term semi-conservative. The genetic make-up of an individual is termed genotype. Most of the DNA sequences among individuals are conserved but genetic variation in 0.1% of DNA influences disease risk, metabolic activity, and drug response. It is important to map occurrence of variation in the human genome, which can help to identify allelic polymorphisms that result in disease. Computational techniques that can rapidly compare entire genome and genes will help to identify polymorphism among population. Comparative genomics is a field in which DNA sequences across several genomes are compared to understand evolutionary aspects of biological processes.

RNA consists of the nitrogenous bases **adenine (A), uracil (U), cytosine (C), and guanine (G)** and can fold into a complex tertiary structure with hair-pin bends that have unpaired bases. Recurring RNA structural motifs have been observed and attributed to biological function. Some of the conformationally recurring motifs include GNRA-like tetraloop, S1, S2, kink turns. Comparative Algorithm to Discover Recurring Elements of Structure (COMPADRES) is an automated approach to identify such recurrent motifs (Wadley and Pyle, 2004). Some of these motifs may contact residues in proteins that are essential for biological function, for example, a pi-turn motif is found on RNA that interacts with ribosomal protein L2.

3. Understanding Proteins: Sequence–Structure–Function

Understanding a protein involves understanding its sequence, structure, and function. Primary sequence of a protein can be represented by 20 unique alphabets. Individual properties and standard residue codes for each of these amino acids can be obtained from the following website: http://www.imb-jena.de/IMAGE_AA.html#Properties. Studies have shown that amino acids can be exchanged with each other without compromising changes in the structure (Azarya-Sprinzak et al., 1997; Benner et al., 1994; Gonnet et al., 1992; Johnson and Overington, 1993; Jones et al., 1992; Naor et al., 1996). Such exchanges are possible because amino acids share similar physico-chemical properties, and changes within similar groups are tolerated (Taylor, 1986). The degree of substitution at a particular residue position depends on the functional role and the environmental location of the residue in the folded form of the protein (Azarya-Sprinzak et al., 1997). Due to this, a number of slightly different sequences may adopt similar structure (divergent evolution) and function. If sequence, structure, and function of a

set of related proteins are already known then inference rules can be derived. These rules can be applied to classify a new sequence for which no structure or function is known. Such inference rules can be a set of conserved residues like sequence motifs or structural motifs that is present in all the members in a related set of proteins (Falquet et al., 2002; Guruprasad and Shivaprasad, 2000; Hofmann et al., 1999; Hutchinson and Thornton, 1996). The effective means of understanding sequence information coming out of genomic projects will require assigning structure and function. Protein sequences that have evolved from a common parent share similar structure and function. If the parent protein structure is known then one can apply comparative modeling techniques to obtain the geometric information for the unknown protein. Hence, relating protein sequences to their structural parent or to a known fold using computational techniques will be critical to handle biological information effectively.

4. Biological Systems, Signals, and Pathways

Many genes are regulated at a given time inside a cell. Regulatory proteins switch these genes on or off based on internal or external cue. A complex network of proteins, small organic molecules, and ions facilitates this regulatory process. For a cell to receive stimuli from the surrounding environment and to devise appropriate responses, signaling pathways are essential. Biological systems have evolved with robust dynamic response to a wide variety of stimuli. Wide positive and negative feedback networks orchestrate such a complex level function. If one considers the multitude of factors which are capable of eliciting cellular responses, it is not surprising that cellular signaling pathways are likely to be extremely complex and diverse. Proteins present in the extracellular environment can come from different sources. Those that are secreted by cells surrounding the "recipient" cell are known as paracrine signals, those that are released by organs distant from the recipient cell are known as endocrine signals, and those released by the cell itself are known as autocrine signals. The proteins in the extracellular milieu can be divided into three broad categories, based upon their effects in the cell: (a) those causing an immediate change in cellular metabolism, (b) those eliciting changes in gene transcription, and (c) those causing fluctuations in electrical conductivity across the plasma membrane. One key aspect of protein binding to the cell surface is specificity. Since the particular molecules binding to the cell surface are intended to elicit specific responses, they must be very selective in the pathways that they initiate. At the same time, it is equally important to

consider interactions between different cellular pathways, as the cell must respond collectively to a variety of stimuli at any one time.

Protein signaling pathways are extremely broad and encompass many different signal transduction pathways. For example the Notch signaling pathway is critical in developmental processes co-ordinated by signal transducer proteins and transcriptional activators, leading to changes at the gene transcription level. The Notch pathway is activated upon contact with neighboring cells expressing Notch ligands. In humans, the ligands that are capable of activating notch are Delta and Serrate. These ligands are membrane bound; therefore, close cell proximity is required for activation of Notch pathways. In some ways, Notch signaling can be considered a "classical" pathway; the binding of Notch ligand to Notch receptor ultimately results in the translocation of the Notch intracellular domain to the nucleus and effects upon gene transcription. Notch ligands are single-pass transmembrane proteins, which contain multiple epidermal growth factors like repeats in the extracellular domain. There are several such signaling pathways that are responsible for widely observed biological processes. A large catalog of such signaling pathways is available at BioCarta pathway listing (http://www.biocarta.com/). Metabolic systems, immune response systems, protein transport, cell cycle and development are some of robust processes that are fundamental to complex biological systems facilitated by macromolecular interactions occurring at different compartments or organelles in the cell.

One of the crucial events during evolution that was responsible for the formation of a cell was the development of an outer membrane. With further evolution and selection, the cells of the present day all have a plasma membrane mainly comprised of phospholipids. Classification of cells as prokaryotes and eucaryotes is based on the absence or presence of a functional nucleus that contains DNA. Most cells have a plasma membrane and other organelles such as golgi apparatus, endoplasmic reticulum, nucleus, mitochondria. The first organisms on earth were unicellular such as bacteria and protozoa. So the question that arises is what led to the evolution of multicellular organisms. With our current knowledge of biology, we can explain the origin and importance of cell–cell interactions. Cell–cell interactions are crucial and are part of every aspect of the cell in eukaryotes. These interactions were responsible for the evolution of multicellular organisms. When we define cell–cell interaction it means communication of cells for division, differentiation, reproduction, migration, apoptosis, contact inhibition, etc. There are over 200 types of cells in the human body broadly classified on the basis of the tissue they are present in, namely epithelia, connective tissue, nervous tissue, and muscle. Cooperation among cellular processes is required for the induction of an antibody response in B cells as

well as for the sensitization of T cells. In addition, the action of activated T cells on target cells is cellular interaction. Macrophages are an essential participant in some of these interactions. The body's ability to replace dead cells and repair damage is by two distinct processes namely regeneration of injured tissue by parenchymal cells and replacement by connective tissue. The mechanism in both of these processes involves cell growth and differentiation as well as cell–matrix interactions. Several proteins control the timing of the events in the cell cycle, which is tightly regulated to ensure that cells divide only when necessary. The loss of this regulation is the hallmark of cancer, which is also due to loss of control in contact inhibition. Major control switches of the cell cycle are cyclin-dependent kinases. Each cyclin-dependent kinase forms a complex with a particular cyclin, a protein that binds and activates the cyclin-dependent kinase. The kinase part of the complex is an enzyme that adds a phosphate to various proteins required for progression of a cell through the cycle. These added phosphates alter the structure of the protein and can activate or inactivate the protein, depending on its function. There are specific cyclin-dependent kinase/cyclin complexes at the entry points into the G1, S, and M phases of the cell cycle, as well as additional factors that help prepare the cell to enter S phase and M phase. Normal mammalian cells show contact inhibition; that is, they respond to contact with other cells by ceasing cell division. Therefore, cells can divide to fill in a gap, but they stop dividing as soon as there are enough cells to fill the gap. This characteristic is lost in cancer cells, which continue to grow after they touch other cells, causing a large mass of cells to form.

5. Technological Advances and Their Benefits to Biology

Technological advances have helped in elucidating the sequence and structure of macromolecules. In 1977, Gilbert and Sanger developed a DNA sequencing method, enzymatic chain termination procedure, and reported the complete genome sequence of bacteriophage fX174 (Sanger et al., 1977). In 1986, Leroy Hood and his co-workers designed the first semi-automated DNA sequencer using Sanger's chain termination method. Commercial manufacture of DNA sequences by Applied Biosystems® and its later improvements enabled rapid sequencing capacity. Scale-up of automated high-throughput DNA sequencing has enabled rapid accumulation of DNA sequence information. In 1990, human genome and in parallel other genome projects that aimed to sequence the genomic information in organisms completely were planned (Cantor, 1990; Watson, 1990). Owing to the importance of information from scientific projects that determine sequence and structure of biological macromolecules, submission of biological

sequence and structure information into central databanks has become mandatory. National Center for Biological Information maintains genetic sequence database GenBank. As of August 2005 it contained approximately 47 million sequences (Benson et al., 2005). This excludes many of the on-going genomic sequences that are yet to be submitted. The discovery of double-helical structure of DNA by Watson and Crick in 1953 using X-ray diffraction patterns led to the understanding of its function and replication mechanism (Watson et al., 1953). This revolutionized biology and basically created the field of molecular biology. Breakthrough in protein X-ray crystallography came with the solution of the phase problem by Perutz and his co-workers in 1954 by applying isomorphous replacement technique (Green et al., 1954). In 1960, John Kendrew and his co-workers solved the first X-ray structure for myoglobin at 6 Å resolution (Kendrew et al., 1958). NMR was later applied to obtain the solution structures of biomolecules (Wagner and Wuthrich, 1979). Protein structures are deposited in the Protein Data Base (PDB) and currently it hosts 32,045 (as of November 2005) protein structures (http://www.rcsb.org) representing thousand unique folds, and 8625 sequences share 50% identity (Berman et al., 2000). Genomic sequencing has resulted in copious amounts of sequence data. DNA sequence alone will not be useful and understanding the complete picture requires annotation of these sequences. Using sampling techniques and extrapolation from the EST experiments, mRNA from known genes, and cross-species gene density comparison, it has been predicted that human genome consists of at least 25,000 genes (Fields et al., 1994; Liang et al., 2000; Roest Crollius et al., 2000; Smaglik, 2000). The information content in genome sequence alone has limited application. Given a DNA sequence gene prediction and gene modeling (Reese et al., 2000) can help in understanding the protein that it codes for in terms of its primary sequence. With the expansion of sequence information it becomes essential to understand the structure, function, interaction, and regulation of proteins in order to understand cellular processes. Microarrays and proteomic technologies enable large-scale study of transcriptome and the proteome, respectively. In this century, technology development has enabled scientists to have necessary tools to study complex biological systems and process. Such studies will shed more light into the complex process underlying every living organism.

6. The Role of Bioinformatics in Big Picture

In the above sections, we provided a glimpse of biological organization and complexity. Scientific community is still unraveling many features of life

using advanced technologies. In recent years, biological science has seen two fields emerge, genomics and proteomics. These fields are rapidly advancing, and bioinformatics provides the tools to analyze and interpret the vast amount of data that is surging. Handling and analyzing biological information is the subject of computational biology and bioinformatics. Computational tasks faced in biology can be broadly divided into *selection* and *classification* problems. *Classification* involves assigning a member to a set or subset that has some defined properties. For example, given a DNA sequence, the *classification algorithms* attempt to address whether the protein it codes for is a tyrosine kinase. On the other hand, *selection* algorithms are involved in data mining for example, identifying a DNA repair-related protein or a serine protease from genomic sequence. To deal with these types of problems first of all we need data sets that are accumulated and curated by experts. Next we need algorithms and specific software tools that can provide necessary search, score, and analyze biological information. Chapter 2 defines some of the widely used algorithms. In order to design tools, a bioinformaticist is expected to write effective and deliverable programs. We describe in Chapter 3 scripting languages and database programs that can be used to design simple programs or prototypes that can be delivered over the web. Chapter 4 describes some of the widely available biological information collection in public databases. Chapters 5, 6, and 7 provide details of biological sequence and structure analysis with emphasis on proteins. Chapters 8 and 9 briefly give overview of genomics and proteomics field. With vast amount of information available in journals, biological text mining, and semantic ontologies for organizing this information is gaining importance. Chapter 10 provides text-mining methods. Chapters 11, 12, 13, and 14 describe practical applications and biological problems that can be studied using bioinformatics tools.

7. Exercises

1. Molecular circuits and electronic circuits have feedback controls. Identify biological signaling pathways and electronic circuits that have positive- and negative-feedback. Predict what will happen if such feedback regulations are perturbed.
2. Identify five different signaling pathways and discuss their importance.
3. What are viruses? Name few viruses that infect bacteria. How do viruses replicate? Describe some perturbations caused on a biological system due to virus infection.

4. Proteins or peptides can function as hormones, enzymes, and transporters. Identify examples for each of them. Compare and contrast glycoproteins and proteoglycans.
5. Describe how proteins are translated and sorted to different organelles in eukaryotes. Compare and contrast this with protein synthesis in prokaryotes..
6. Orthologs are genes that have evolved across organisms that generally have conserved function. Paralogs are genes that have evolved within organism due to duplication events. Identify orthologs and paralogs in globin gene.

References

Azarya-Sprinzak, E., Naor, D., Wolfson, H. J., and Nussinov, R. (1997). Interchanges of spatially neighbouring residues in structurally conserved environments, Protein Eng *10*, 1109–22.

Benner, S. A., Cohen, M. A., and Gonnet, G. H. (1994). Amino acid substitution during functionally constrained divergent evolution of protein sequences, Protein Eng *7*, 1323–32.

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., and Wheeler, D. L. (2005). GenBank, Nucleic Acids Res *33*, D34–8.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. (2000). The Protein Data Bank, Nucleic Acids Res *28*, 235–42.

Cantor, C. R. (1990). Orchestrating the Human Genome Project, Science *248*, 49–51.

Falquet, L., Pagni, M., Bucher, P., Hulo, N., Sigrist, C. J., Hofmann, K., and Bairoch, A. (2002). The PROSITE database, its status in 2002, Nucleic Acids Res *30*, 235–8.

Fields, C., Adams, M. D., White, O., and Venter, J. C. (1994). How many genes in the human genome? Nat Genet *7*, 345–6.

Gonnet, G. H., Cohen, M. A., and Benner, S. A. (1992). Exhaustive matching of the entire protein sequence database, Science *256*, 1443–5.

Green, D. W., Ingram, V. M., and Perutz, M. F. (1954). The structure of haemoglobin V. Imidazole-methaemoglobin: a further check of the signs, Proc Roy Soc *A225*, 287–307.

Guruprasad, K., and Shivaprasad, M. (2000). Database of structural motifs in proteins (DSMP), Bioinformatics *16*, 373–375.

Hofmann, K., Bucher, P., Falquet, L., and Bairoch, A. (1999). The PROSITE database, its status in 1999, Nucleic Acids Res *27*, 215–9.

Hutchinson, E. G., and Thornton, J. M. (1996). PROMOTIF--a program to identify and analyze structural motifs in proteins, Protein Sci *5*, 212–20.

Johnson, M. S., and Overington, J. P. (1993). A structural basis for sequence comparisons. An evaluation of scoring methodologies, J Mol Biol *233*, 716–38.

Jones, D. T., Taylor, W. R., and Thornton, J. M. (1992). The rapid generation of mutation data matrices from protein sequences, Comput Appl Biosci *8*, 275–82.

Kendrew, J. C., Bodo, G., Dintzis, H. M., Parrish, R. G., Wyckoff, H., and Phillips, D. C. (1958). A three-dimensional model of the myoglobin molecule obtained by x-ray analysis, Nature *181*, 662–6.

Kitano, H. (2002). Systems biology: a brief overview, Science *295*, 1662–4.

Liang, F., Holt, I., Pertea, G., Karamycheva, S., Salzberg, S. L., and Quackenbush, J. (2000). Gene index analysis of the human genome estimates approximately 120,000 genes, Nat Genet *25*, 239–40.

Meselson, M., and Stahl, F. W. (1958). The Replication Of Dna In Escherichia Coli, Proc Natl Acad Sci U S A *44*, 671–82.

Naor, D., Fischer, D., Jernigan, R. L., Wolfson, H. J., and Nussinov, R. (1996). Amino acid pair interchanges at spatially conserved locations, J Mol Biol *256*, 924–38.

Noble, D. (2002). Modeling the heart--from genes to cells to the whole organ, Science *295*, 1678–82.

Reese, M. G., Kulp, D., Tammana, H., and Haussler, D. (2000). Genie--gene finding in Drosophila melanogaster, Genome Res *10*, 529–38.

Roest Crollius, H., Jaillon, O., Bernot, A., Dasilva, C., Bouneau, L., Fischer, C., Fizames, C., Wincker, P., Brottier, P., Quetier, F., et al. (2000). Estimate of human gene number provided by genome-wide analysis using Tetraodon nigroviridis DNA sequence, Nat Genet *25*, 235–8.

Sanger, F., Nicklen, S., and Coulson, A. R. (1977). DNA sequencing with chain-terminating inhibitors, Proc Natl Acad Sci U S A *74*, 5463–7.

Smaglik, P. (2000). Researchers take a gamble on the human genome, Nature *405*, 264.

Taylor, W. R. (1986). The classification of amino acid conservation, J Theor Biol *119*, 205–18.

Wadley, L. M., and Pyle, A. M. (2004). The identification of novel RNA structural motifs using COMPADRES: an automated approach to structural discovery, Nucleic Acids Res *32*, 6650–9.

Wagner, G., and Wuthrich, K. (1979). Structural interpretation of the amide proton exchange in the basic pancreatic trypsin inhibitor and related proteins, J Mol Biol *134*, 75–94.

Watson, J. D. (1990). The human genome project: past, present, and future, Science *248*, 44–9.

Watson, J. D., and Crick, F. H. (1953). Molecular structure of nucleic acids; a structure for deoxyribose nucleic acid, Nature *171*, 737–8.

Chapter 2

Computer Programming Fundamentals and Concepts

Deepak N. Kolippakkam[1], Pankaj Gupta[2], and Venkatarajan S. Mathura[2]

[1]*Harvard University, Boston, USA*
[2]*Roskamp Institute, 2040 Whitfield Avenue, Sarasota, Florida, USA*

Abstract: This chapter introduces programming methods and languages that help beginners to learn essentials of computer programming. Two interpreted languages: Perl and PHP are introduced with examples. A popular open source database, MySQL is also included. Additional web-pointers are provided.

Key words: Perl, PHP, MySQL, RDBMS

1. Purpose

These days most modern curricula include computer programming concepts and fundamentals. Not to warrant this assumption about all readers, this chapter is included as a way to pick up techniques for those who haven't done programming. We introduce two interpreted scripting languages that are widely used in developing bioinformatics applications and MySQL™ database. Scripting languages are powerful yet easy to learn. We outline two scripting languages: Perl and PHP. Pointers are provided to set up a web-server on a Linux platform. After reading and working through all the exercises in this chapter, a reader will be able to roll out their database-driven bioinformatics web application!

2. Learning Objective

- Data types, operators, and routines in Perl or PHP
- Simple scripts to say 'Hello World!'

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_2,

- Database design with MySQL™
- A simple database-driven web application

3. Perl Programming

Perl stands for Practical Extraction and Report Language developed by Larry Wall. It is a powerful language that can be written in a lucid style. The concepts of the language and the syntax are very easy to learn and do not require prior programming experience to start. As a Bioinformatician, you will be expected to be a programmer who can get things done quickly and effectively. Nevertheless, Perl can be used to achieve complex tasks and build an entire application. There are many bioinformatics tools that have been written in Perl and it is a widely used prototyping language to try out your rough sketch or proof of principle. In short, it is a language that will get your job done quickly and effectively. Perl has rich pattern matching and regular expressions that enable text processing effectively, making it attractive for developing bioinformatics applications. It is essential to learn core of the language that includes variables, operators, control structures, functions, and subroutines.

3.1 Variables

In Perl, data-structures include scalar, array, and hash. Scalars are represented by a '$' prefix. Scalars can store numeric or string literals. For example,

```
$pH=7.5;
$Sequence="ACCTCCAGAA";
$moltype='DNA';
```

String literals are enclosed in either single or double quotes. Strings literals that have double quotes are interpreted. Numerical literals can be integer or a float. Subtype conversion takes place automatically depending on the context or the operator. For example,

```
$resPos='5';
$newPos=$resPos+12;
```

$newPos contains value 17. The string value is automatically converted into a numerical type before addition. Unlike in C or C++, where one needs to specify the variable types and conversions, Perl provides an easy way to define scalar variables. Arrays contain several scalars and can be multidimensional. Arrays are prefixed with the symbol '@'. For example,

```
@protein_symbols=("BCL2","APE","DAF");
```

Thus, the variable protein_symbols contain a list of three protein names indexed by a number 0, 1, and 2. One can retrieve individual values stored in an array by using the scalar form of the array.

```
print $protein_symbols[0];
```

will print 'BCL2'. The index starts from 0; hence, the first array element stored is defined as the scalar with index 0. The last index of a one-dimensional array can be obtained using special symbol '$#'. For example,

```
print $#protein_symbols;
```

will print 2. In case of arrays, the index is a numerical key. Perl provides a sophisticated data structure called hash that can index a value based on any key. Hashes are defined using symbol '%' before the variable name.

```
%proteinhash=(
               "BCL2"=>"B cell lymphoma 2",
               "APE"=>"AP endonuclease",
               "DAF"=>"Decay accelerating factor"
                );
```

Individual values can be retrieved by defining a scalar as:

```
print $proteinhash{'APE'};
```

This will print 'AP endonuclease'. Special functions are available for manipulating both arrays and hashes. In order to obtain all the keys in a hash and store it in an array, one can use the keyword *keys*. For example,

```
@proteinsymbols=keys %proteinhash;
```

The above code creates a new array `@proteinsymbols` using the keys from the hash. Alternatively, if one uses the keyword '*values*', actual values can be extracted from the hash.

3.2 Operators

Standard mathematical operators like addition (+), subtraction (–), multiplication (*), modulus (%), and exponentiation (**) are available. If a + or . operator is used in the context of two strings, then it performs concatenation. Assignment operator = is used to assign a value for variables. If an arithmetic or string concatenation operator is present before the assignment operator, the left side value is operated with the right side value and a new left side value is computed. For example,

```
$a="hello";
$a.=" world";
print $a;
```

This will print 'hello world'. Auto increment or decrement can be performed by prefixing or suffixing of a variable with `++` or `--`. Logical operators include And (`&&`), Or (`||`), Not (`!`). Comparison operators are equal (`==` for numeric, `eq` for string), less than equal (numeric <=, `le` for string), not equal

(numeric !=, `ne` for string), greater than (numeric >, `gt` for string), less than (numeric <, `lt` for string), or comparison (numeric <=>, `cmp` for string).

3.3 Control Structures

Perl provides various control structures like `if..elsif`, `unless`, `while`, `until`, `foreach`, and `for`. These commands provide conditional structure, looping, or cycles. There are breaking out commands like `next`, `last`, and `exit` which, if executed, breaks out of the loop or the program. The `if` structure executes if a condition is satisfied. Multiple `if` can be combined using `elsif`. For example,

```
if($molecultype eq "DNA"){
    print "It is a DNA\n";
}elsif($molecultype eq "RNA"){
    print "It is an RNA\n";
}
```

If a block of code is required to be executed while a condition is false, then the choice should be `unless` command. `While` and `until` statements can be applied where looping over a condition is required. The `for` statement can be used for looping some structure for a specified number of times. The structure of a `for` statement includes a starting value, an exit condition, and an increment operator. For example, to print 1 to 100:

```
for($i=1;$i<101;$i++){
   print $i,"\n";
}
```

If you have an array and would like to loop through the array values, `foreach` statement should be used.

```
#!/usr/bin/perl
#define a hash
%proteinhash=(
               "BCL2"=>"B cell lymphoma 2",
                  "APE"=>"AP endonuclease",
               "DAF"=>"Decay accelerating factor"
               );
#loop through an array of keys present in the hash
foreach(keys %proteinhash){

#push command is used to push a value into an array
#{$_} represent current element while looping an array
  push(@newproteinsymbols,$_);
  push(@newproteindescp, $proteinhash{$_});
}

for($j=0;$j<=$#newproteinsymbols;$j++){
  if($newproteinsymbols[$j] eq "BCL2"){
     print $newproteinsymbols[$j],"\n";
  }
}
```

The above code will print 'BCL2'.

3.4 Regular Expressions

Perl is a powerful program for regular expression matching. It has UNIX style regular expression and pattern matching. Pattern matching in Perl can be learned best by practising and trying to construct complex expressions. Matching operator evaluates the presence of a pattern or word in a given line and can be used in conditional statements. For example,

```
$present_line=~ m/BIOINFO/;
```

The above evaluates whether the word 'BIOINFO' occurs at any location in a sentence (including partial matches). If you like to find the occurrence in the start then:

```
$present_line=~ m/^BIOINFO/;
```

or to find a word at the end of a sentence

```
$present_line=~ m/BIOINFORMATICIAN$/;
```

The above will find the occurrence of a complete word (in this case BIOINFORMATICIAN at the end of the line). If you would like to identify poly-glutamine repeat of exactly 20, one can use special repetitive operators. For example, `{n}` that follows 'Q' in the pattern operation will match repetitive 'Q's exactly `'n'` times.

```
$present_line="KLQVQQQQQQMMEF";
if($present_line=~ m/Q{6}/){
   print "Found a trivial poly Q match";
}
```

Other repetitive operators include '`*`' (none or more), '`.`' (any), `{n,m}` atleast `n` occurrences, and not more than `m`. Escape characters like '`\n`' for new line, '`\t`' for tab are available. Additionally, '`\w`' represents a word, '`\s`' space character, and '`\d`' for digit character. Grouping of patterns is possible using `()` or `[]` operators. Patterns that are present inside `()` will be matched and made available in a special variable `$1`, `$2`, and so forth depending on the occurrence of `()` in the pattern. The `[]` type brackets can be used for '`or`' operators. For example, `[ATE]` means residues `A`, `T`, or `E` should occur at a position and `(ATE|ALK)` means the seqlet `ATE` or `ALK` should be present. Let's write a motif-identifying program that detects phosphorylation site of Caesin II kinase. The pattern of phosphorylated site as documented in PROSITE (PDOC00006) should have an acidic residue (either Asp or Glu), which must be present within three residues from the C-terminal of the phosphate acceptor site. The acceptor site is Ser or Thr. The motif can be represented in regular expression as `[ST].{2}[DE]`. This expression means that the first position (acceptor site) should have either `S` or `T`, followed by exactly any two residues, and finally acidic residues `D` or `E`.

```
$seq="AKKLVFLSDLEMMMMQQQPR";
if($seq= ~m/[ST].{2}[DE]/){
   print "Possible Caesin II Kinase site is found";
}
```

3.5 File Handling

Files in Perl can be opened for reading or writing using the command 'open' and a handler keyword. For example, to open a file 'input.txt' available in the directory in which the program is being executed:

```
open(INP,"input.txt");
```

This command will pass the input as file handle INP which can be read and stored in an array:

```
@data=<INP>;
#To close the filehandle
close(INP);
```

and to open a file for writing one should use the symbol '>'. For example,

```
open(OUT,">output.txt");
print OUT "hello world!\n";
close(OUT);
```

3.6 Subroutines and Functions

Subroutines can be written in Perl by passing variables or reference to variables. Each subroutine starts with the word subroutine followed by a unique name. The return command at the end of the subroutine passes back the values to main program. The variables passed into subroutine can be accessed using a special variable @ and local variables that are initialized within subroutine can be defined using 'my'.

```
($add_val,$prod_val)=operatemynum(1,7);
print $add_val," ",$prod_val,"\n";
subroutine operatemynum {
my ($firstnum, $secondnum, $sumofnum, $prodofnum);
$firstnum=@_[0];
$secondnum=@_[1];
$sumofnum=$firstnum+$secondnum;
$prodofnum=$firstnum*$secondnum;
return ($sumofnum, $prodofnum);
}
```

The subroutine operatemynum returns a sum and product of two numbers. Arrays and hash can be passed as a reference into the subroutine. For example,

```
operateonarrayofnum(\@arrayofnum1,\@arrayofnum2);
```

To dereference the values one should prefix the variable with '$'. For example, to access the value of the first array in firstnums:

```
$value=$$firstnums[0];
```

Similarly, arrays created inside the subroutine can be passed to main program as a reference.

So far, a glimpse of Perl language is covered here. Readers who are not familiar with Perl can obtain more information at Comprehensive Perl Archive Network website (http://www.cpan.org). Advanced topics like Perl modules, classes, and other object-oriented subjects are left to the users to learn further.

4. PHP Programming

PHP (recursive acronym for PHP: Hypertext Preprocessor) is an open-source scripting language used (but not limited to) web development and generating HTML content (http://www.php.net). The idea of PHP was originally conceived by Robert Lerdorf, but has since been subjected to various changes. The current version of PHP is 5.2.1 (March, 2007). The main feature of PHP is that, it can be embedded into existing HTML documents and can turn a static page into a dynamic data-driven web application. In addition, PHP has a rich array of functions for pattern matching, database connectivity, graphics, image manipulation, XML, etc. PHP can be used in three different ways:

- *Server-side scripting*: Creating dynamic web content including forms, XML documents, graphics, Flash animations, PDF files, etc.
- *Command-line scripting*: Similar to Perl or the Unix shell, PHP can be used to perform system administration tasks, backups, parsing, etc.
- *Client-side GUI applications*: Using the PHP-GTK module, cross platform GUI applications can be created.

In the following section, we detail about:

4.1. PHP Language syntax and data types
4.2. Creating Web Interfaces
4.3. Accessing data from an RDBMS – MySQL
4.4. Creating a full fledged data-driven web application using PHP and MySQL

The examples in this text would be specific to a web application. Command line scripting examples would be specified otherwise.

4.1 Language Syntax and Data Types

Since PHP can be embedded into HTML contents, the PHP interpreter needs to know which of the lines of code are PHP and which are not. In general,

anything which is enclosed within <?php and ?> tags is assumed to be PHP content. Of course, the syntax of the PHP language would be checked at run time. Consider the following piece of code:

```
<center>HTML code begins<br></center>
<?php
  echo "PHP code<br>";
  echo "Why should all programs begin with Hello World<br>";
  // This is a comment
  /* This is also a comment */
?>
<center>HTML code ends<br></center>
```

This code instructs the PHP interpreter to consider the code between the PHP tags as PHP code. The result would be:

```
HTML code begins
PHP code
Why should all programs begin with Hello World
HTML code ends
```

Now that we have seen our first PHP script in action, let us proceed to discuss some of the data types available in PHP. Akin to all normal programming languages, basic data types include Boolean, integers, floating point numbers, strings, arrays, and objects. PHP also has a data type called 'resource' which can hold a reference to an external resource such as a database connection, ftp/ldap operation, etc. To quickly go through these data types, consider the following code snippets

Boolean: Variables that hold either 'True' or 'False' values. Mainly used for testing truth values of certain conditions.

```
$a_flag = True;
if ($a_flag)
{  // do something; }
```

Integer and floats: Integers are signed/un-signed whole numbers (can be in accord with the decimal, octal, or the hexadecimal number system). Octal numbers need to be preceded with a 0 and hex numbers with a 0X. There are various functions for type conversions to and from integers.

```
$a = 10; // un-signed integer
$b = -23; // signed integer
$o = 012; // octal number
$h = 0X123; // hexadecimal number
```

Floats can be used in the same way as integers. Data types need not be declared explicitly (as in C or Java).

```
$complicated_float_number = 1.3; // float example
```

Strings: Strings are a sequence of characters (unlimited in size). A variable can be specified as a string in two main ways: single-quoted or double-quoted strings.

```
$string1 = 'example string';
echo '$string1'; //
$string2 = "box of";
echo "I got a {$string2} chocolates";
// double quotes allow string parsing with variables
echo "I got a {$string2} papers";
echo "where are the \"islets of langerhans\"?\n";
// outputs: where are the "islets of langerhans"? and a line feed.
//Note that double quotes within double quotes need to be escaped.
```

Similar to PHP code being embedded into HTML, the opposite is also possible: *HTML code embedded within PHP*. This is a special kind of a string specification called 'heredoc'.

```
<?php
   // some PHP code
print<<<HTML
   <hr><br>
   <table align = center border = 1>
   <tr><td>1</td></tr>
   <tr><td>2</td></tr>
   </table>
   <br>
HTML;
   // some PHP code
?>
```

Note that '`print`' can be used in all occasions as '`echo`' but not vice versa.

Arrays: Perhaps the most commonly used data structure in programming is the array. An array holds a list of values, which can be identified by an index (or position). An array can either have just values or keys associated with values.

```
$array1 = array('a','e','i',o','u'); // creates an array of vowels
$array1[0] = 'a'; // creates the same array, although laboriously
$array1[1] = 'e';
$array1[2] = 'i';
$array1[3] = 'o';
$array1[4] = 'u';
print_r($array1); //prints the contents "index =>value" format.
$array2 = array('Japan' => 'Tokyo', 'Italy' => 'Rome');
// the above array is an associative array, similar to a hash.
foreach ($array2 as $country => $capital){
     echo "The capital of $country is $capital", "<br>";
}// would print each country in the array and its capital.
for ($i=0;$i<count($array1);$i++){
     echo "$array1[$i],"<br>";
} // would print vowels.
```

Accessing individual array members is possible by specifying the index of that value. Functions are available to sort, merge, reverse, search, etc. on arrays.

Objects: PHP also supports object-oriented programming. For using the object data type, first a class has to be created. Once that is done, any number of objects can be made using the 'new' keyword.

```
class sqs
{
     var $name ='';
     function name ($name1 = NULL)
     {
          $this->name = $name1;
          return $this->name;
     }
} // once the class has been defined, objects can be created.
$n1 = new sqs;
$n1->name('ABCD');
echo "Welcome, $n1->name";
```

Resources: Communicating to an external resource can be done by using the resource data type. For instance, in our text, we would be providing examples where a database connection is needed and data is transferred back and forth. Depending on what kind of a database is being used, the connection method and the parameters to be passed would differ. For example,

```
$db=mysql_connect("localhost","username","password");
mysql_select_db("sampledb",$db);
```

This code creates a connection to a MySQL database using the given user name and password. It specifically establishes a connection to the database named 'sampledb'. On successful connection, the user would be able to access individual tables, make joins, etc. Hence,

```
$result = mysql_query("select PROT_NAME from pdbase where PROT_ID =
'23228'",$db);
```

can be executed. We would discuss about database connectivity in more detail in 4.3.

4.2 Creating Web Interfaces

One of the main uses of PHP is its ability to create dynamic web content. PHP can be used to access form values, upload files, send cookies, set/unset sessions, etc. Consider the following simple HTML form:

```
<html>
<head><title>Simple form1</title></head>
<body>
<center><h4>Enter Information</h4></center>
<form action = 'sform1.php' method='POST'>
<table align = center border = 1>
<tr><td>Enter your name:</td><td><input type = text name =
NAME_FIELD size = 10/></td></tr>
<tr><td>Select favorite color</td><td><select name = FAV_COLOR>
     <option value = red selected>Red</option>
```

```
      <option value = green>Green</option>
      <option value = blue>Blue</option></select>
      </td>
</tr>
</table><br>
<center><input type = 'submit name = SUBMIT1 value =
Submit></center>
</form>
</body>
</html>
```

This code produces a form with a text field, a drop down menu, and a submit button. When the user enters information and clicks on submit, this form would look for a PHP script named 'sform1.php' and passes the form values using the POST method. But not yet! We have not created the 'receiving' PHP script for this form. PHP can access form parameters (or values) using either the POST method or the GET method. When the form is submitted using the GET method, the form values are encoded in the URL. This is the same as the query string, which one sees in the URL (address bar) after a search result has been completed. In contrast, the POST method leaves the URL clean and passes the information in the body of the HTTP request. But the important difference of these two methods is that the GET method is *idempotent* and the POST is not. In simple terms, the GET method should be used in cases where the *response* page is not going to change (since the user can *bookmark* a search result URL along with the form values), and the POST method should be used when the content in the response pages changes over time – like our dynamic data-driven web applications.

So, in the above example, there are three form parameters being passed: the name field, color field, and the submit event. Considering the submit event as a form parameter may be counter-intuitive, but makes sense when there are several submit buttons in the same page (and they occur *all* the time!). Thus, a PHP script needs to be written which would receive the POST values and do something with it. The following is the sform1.php script.

```
<?php
if (true == isset($_POST['SUBMIT1']))
{
        $name_f = ''; $color_f = '';
        $name_f = $_POST['NAME_FIELD'];
        $color_f = $_POST['FAV_COLOR'];

        if (strlen($name_f) > 0)
        {
                echo "<b>$name_f</b> selected <font color =
                  $color_f>$color_f</font><br>";
        }
        else
        {
          echo "<b>Anonymous</b>
               selected <font color = $color_f>$color_f</font><br>";
        }
 }
?>
```

The code is enclosed within the PHP tags. First, we need to make sure that the user clicked the SUBMIT1 button. This check is needed if there were multiple forms in our HTML code, but even so, it is good practice to make this check. $name_f and $color_f would store future values of the NAME_FIELD and FAV_COLOR. Since we use the POST method, we need to receive the form values using the $_POST. If the user had entered a User1 as the name and selected green color, 'User1 selected green color' would be displayed on the browser. If there was no name specified, 'Anonymous' is used instead. This is just to demonstrate that one could perform virtually any kind of checks and conditions here. This kind of error-checking is inevitable, since most of the time the PHP script continues to take the data from the form and proceeds to populate a database table. Also note that in this case, the nameless HTML code was saved as a separate entity, as was the sform1.php file. However, one can also combine both of them into one file (in fact that is the common way) and execute it as a PHP file. Notice that we liberally used HTML tags inside our PHP code. In this text, we have studied about PHP, a popular programming language, and its utility in building web applications. This is only an introduction and a quick start guide to PHP for building data-driven web applications. If one is more interested, www.php.net and www.mysql.com have detailed function listings and examples.

5. Basic RDBMS and SQL

Relational Database Management Systems (RDBMS) have three kinds of statements to maintain data. They are:

5.1 Data Definition Language (DDL)

They are used to define data structures in the database. Examples are:

a. Create: Creating databases or tables
b. Drop: Deleting databases or tables
c. Alter: Change structure of a table – changing columns, renaming columns, etc.
d. Describe: Display the structure of a table
e. Use: Choose and use a particular database/table

Usage:

```
// create a database
create database employee;

// create a table
```

```
   create table project (PROJ_NAME varchar(20), PROJ_NO
int(4), PROJ_DATE date, PROJ_MANAGER varchar(45),
PROJ_STATUS blob);

   // delete a table
   drop table project;

   // delete a database
   drop database employee;

   // delete a column
   alter table project drop column PROJ_DATE;

   // add new columns
   alter table project add column PROJ_START_DATE date,
PROJ_END_DATE date

   // change the type of a particular column
   alter table project modify PROJ_NO int(5);

   // change the table name
   alter table project rename project_1;

   //change column names
   alter table project_1 change PROJ_NAME PROJECT_NAME
varchar(20);
```

5.2 Data Manipulation Language (DML)

They are used to manipulate data in the database. Examples are:

a. Insert: Add a new record (row) to a table
b. Select: Select a group of records from one or more tables with or without conditions
c. Update: Change values of a particular record based on a condition
d. Delete: Remove records based on a condition

Usage:

```
   // insert new record in a table
   insert into project values ('PROJECT 1', 20,
'2005/06/06','MANAGER123','XYZ');
   // select particular rows based on a condition
   select PROJ_NAME, PROJ_NO from project where PROJ_NO
> 20;
   // select - using foreign keys
   select a.EMP_NO, b.PROJ_NO from employee_details a,
project b where a.PROJ_NO = b.PROJ_NO group by
a.EMP_NO;
   //update a row
   update project set PROJ_MANAGER = 'TEMP_MAGR123'
where PROJ_MANAGER = 'MANAGER123';
   // delete a row
   delete from project where PROJ_NO = 20;
```

5.3 Data Control Language (DCL)

They are used to administer permission and control access of data in the database. Examples are:

a. Grant and Revoke: Grant and Revoke are used to create accounts and grant/revoke specific rights to a user

Usage:

```
// create a new user
grant all on *.* to 'username'@'hostname' identified
by 'pass123' with grant options;

// revoke rights
revoke all privileges from 'username';
```

6. Web-Pointers

CPAN (http://www.cpan.org) has the official and additional links to user contributed Perl codes and modules. ActiveState (http://www.activestate.com/Products/activeperl/) website contains binary distribution of Perl for windows.

PHP (http://www.php.net) official home page contains directions to install PHP executable and configuration under different servers. Detailed documentation is available at the site with examples.

MySQL (http://www.mysql.org) official homepage has the link for program download and complete documentation. In addition to database server and client programs, one can also find query browser interface. PHP/MySQL tutorial is available at: http://hotwired.lycos.com/webmonkey/. Quick reference cards for commands in PHP, MySQL, Linux, Perl can be obtained from http://www.digilife.be/quickreferences/quickrefs.htm.

Bio-Perl project (http://www.bioperl.org) is an open source project that develops Perl based modules, classes, and routines for bioinformatics research. Some of the modules/classes include biological sequence analysis, database access, search tools, alignment, phylogenetic tree constructions (a dendrogram representing evolution of sequences), gene structure prediction, protein structure retrieval, etc. Bio-java (http://www.biojava.org) is a project that develops bioinformatics tools and APIs in Java framework.

EMBOSS (http://emboss.sourceforge.net) is an open source bioinformatics software project that has collection of programs to facilitate computational molecular biology.

Bio-linux (http://www.biolinux.org) distributes pre-compiled linux rpms of several bioinformatics packages for installation in linux operating system.

Chapter 3

Introduction to Algorithms

Senthilkumar Radhakrishnan[1], Deepak Kolippakkam[2], and Venkatarajan S. Mathura[3]

[1]*California Institute of Technology, Pasadena, USA*
[2]*Harvard University, Boston, USA*
[3]*Roskamp Institute, 2040 Whitfield Avenue, Sarasota, FL 34243*

Abstract: Computational methods are designed to solve complex problems systematically and efficiently. Classification and selection procedures are often used in biological sequence and other data analysis. This chapter provides an introduction to different methods like clustering, hypothesis - testing, and classification methods.

Key words: Bayesian methods, Decision trees, Bayesian, Neural network, Clustering, Support Vector Machines, PCA

1. Introduction

Systematic solutions to complex problems are highly desirable and algorithms are designed to efficiently solve them in a finite time. Rigorous data-analysis and computer science problems use computer programs that implement suitable algorithm to arrive at a solution. These include recursive/iterative methods, graph search algorithms, dynamic programming, greedy algorithms, etc. A thorough discourse on computational algorithms is out of scope for an introductory book but we provide here some of the term definitions of widely used computational methods that are frequently applied in the forthcoming chapters in bioinformatics.

1.1 Classification

The process of dividing a dataset into mutually exclusive groups such that the members of each group are as "close" as possible to one another, and

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_3,

different groups are as “far” as possible from one another, where distance is measured with respect to specific variables or class labels (for predicting) is known as classification. For instance, determining whether a protein binds to DNA or not based on sequence and structural motifs, cancer type classification based on micro array expression are good examples of classification problems. Classification falls under the category of “supervised learning”. Some of the classification methods widely applied in bioinformatics is Decision trees, Support-vector machine based classifier, Bayesian Classifiers, Neural Network Classifiers, etc.

1.2 Hypothesis Testing

Hypothesis tests are statistical procedures for making rational decisions about the reality of effects. It is an inference technique where either one accepts a null hypothesis (Ho, e.g.: no difference between control and treatment group) or rejects it (that is accepting the alternative). Either one is true, but not both (mutually exclusive and exhaustive). Hypothesis testing is carried out on the observed sample data that represents characteristic population using test statistics. The test statistic quantifies the difference between normally distributed (in case of t-test) hypothetical population and the observed data, which can be used to obtain a p-value. The p-value is the probability to observe sample data assuming null hypothesis is true. A p-value of 0.01 means that the chance of observing sample data is only 1/100 while there is no effect or null is true (no effect on treatment). Since the chance of such observation is very small, we reject null hypothesis and conclude that there is a difference between control and the treatment group. Several statistics exist: t-statistic is for testing mean, F, or Chi-test for testing variance. Hypothesis testing is used in micro array data analysis, sequence analysis, etc.

1.3 Decision Tree

Decision trees are a simple approach to the problem of learning from a set of independent instances. Every node in the decision tree (except the leaf nodes) involves testing of a particular attribute. In most cases, the test at a node compares an attribute value with a constant. Sometimes, the test may be between the values of a set of attributes too. Leaf nodes give the classification labels or the probability distribution over all possible classifications. Decision trees are used in protein secondary structure prediction (Selbig et al., 1999), protein sorting signal prediction, etc.

1.4 Clustering

Clustering is applicable when there is no "class-label" to predict, but rather when the instances need to be divided into groups. The clusters formed would then reflect some mechanism in the domain, which causes the instances to bear a strong resemblance to one another (within its cluster) than they do with the remaining instances (in other clusters). Since there is nothing to "learn" from the instances, clustering is a type of "Unsupervised Learning". Clustering of genes based on expression profiles is widely used to interpret micro array data.

1.5 Principal Component Analysis

PCA is a technique that is used to simplify a dataset. For instance, one would be interested in reducing the dimensionality of a dataset. PCA is a linear transformation which chooses a new co-ordinate system for the data such that the greatest variance by any projection of the dataset comes to lie on the first axis (also called as the first principal component), the second greatest variance in the second axis, and so on. Essentially, a set of correlated variables is transformed into a set of uncorrelated variables, which are ordered by reducing the variability. The uncorrelated variables are linear combinations of the original variables and the last of these variables can be removed with minimum loss of information. PCA is also known as Karhunen-Loeve transformation.

1.6 Multidimensional Scaling

Multidimensional scaling can detect meaningful underlying dimensions, which can help explain observed similarities and dissimilarities (distances) between the investigated data points. Hence multidimensional scaling can provide a visual representation of the observed patterns in the objects under study. For instance, given a distance table between cities, a multidimensional scaling procedure can produce a map showing the relations between the cities in terms of distance. Multidimensional scaling has been applied to analyze gene correlation in micro array experiments (Taguchi and Oono, 2005), to correlate similarity matrices with physical-chemical properties, to derive descriptors for amino acids (Mathura et al., 2003), etc.

1.7 Regression Analysis

Regression is another model extraction method in which the predicted values are continuous valued function rather than discrete class labels. It is a

statistical technique, which is used to determine the parameters of a function that cause the function to best-fit a set of observations (data). For example, dependence of blood pressure Y on the age X of a person is called as regression of Y on X. Regression mostly produces an optimal solution, where the error would be kept at a minimum.

1.8 Linear Discriminant Analysis

Linear discriminant analysis (LDA) is a classification technique that makes use of a weighted sum. For each object to be classified, LDA takes a weighted sum of values of variables that determine the classification. For example, a financial institution can offer a bank loan, after determining the risk of default by the customer, taking salary, credit history, financial commitments into considerations. LDA is mainly used for two-class classifications, where the data is assigned to one of the classes based on its characteristics. In the previous example, a customer is either offered a loan or not after evaluating the above-mentioned list of criteria. Mathematically, a linear discriminant equation can be represented as follows:

$$\mathbf{Y_i = a_1X_1 + a_2X_2 + a_3X_3 + \ldots\ldots + a_nX_n + C}$$

where there are "n" number of predictor variables, "i" can take two values representing the two classes, and "C" is a constant.

One good example of application of LDA in biology is the analysis of microarray data (Hakak et al., 2001), provided the need is a simplistic classification of data into two groups. An online demonstration of this approach for gene classification can be found at http://www.biostat.harvard.edu/complab/dchip/lda.htm

1.9 Fuzzy Logic

Fuzzy Logic is a problem-solving control system methodology which incorporates a simple rule based `IF X and Y then Z` approach, rather than solving the problem by mathematical modeling. It is empirically based and mimics how a person makes decisions. The rate of error is usually very low, since the system can correct itself by using a simple feedback procedure.

Fuzzy logic most aptly mirrors the uncertain world by including a shade of "gray" whereas the binary logic of a computer by definition can understand only "black" and "white". For this reason, fuzzy logic is more appropriate to model non-discrete, continuous systems, which are plentiful in a number of fields. Apart from control systems, where fuzzy logic finds

its main application (Hayward and Davidson, 2003), it is being increasingly used in medicine and biology (Ibbini and Masadeh, 2005; Phuong and Kreinovich, 2001). Some recent examples from the literature include the use of fuzzy logic in prognosis of cancer (Seker et al., 2003), tumor marker profiling (Schneider et al., 2003a; Schneider et al., 2003b), and gene expression data analysis (Ressom et al., 2003).

1.10 Pattern Recognition

Pattern recognition is an art of identifying patterns within previously learned data (a priori) or by statistical information extracted from the patterns. A complete pattern recognition system consists of a sensor (for gathering observations – data acquisition), a feature extraction module that computes numerical or statistical information from the observed data and a classifier that does the actual job of describing the features.

The most straightforward application of pattern recognition in biology is to deduce similarities either at the level of DNA or amino acid sequences. In fact, it was shown recently that this technique performs very well for analysis of gene promoter sequences (http://promoterplot.fmi.ch/) (Di Cara et al., 2005). Other uses of pattern recognition include functional site prediction in proteins (Yang et al., 2005), gene expression analysis (Coberley et al., 2004; Szabo et al., 2002; Valafar, 2002), and protein secondary structure prediction (Oldfield, 2002).

1.11 Bayesian Statistics

Bayesian statistics use the rules of probabilities to make inferences about a parameter (Berry, 1996). Conditional probabilities are especially used to describe a phenomenon in a Bayesian model. Bayes' theorem can be represented by the following equation:

$$\mathbf{p(A|B) = p(B|A) \times p(A)/p(B)}$$

where p(A|B) is the probability of an event A occurring knowing event B, p(B|A) is the probability of event B occurring knowing A, p(A) and p(B) are the respective individual probabilities of events A and B. The main advantage of Bayesian methods lie in their robustness even with partial information and poorly determined parameters as inputs (Eddy, 2004; Shoemaker et al., 1999). Bayesian statistics have a number of applications, especially in genetic analysis tasks such as molecular evolution (Sinsheimer et al., 1996; States and Botstein, 1991), quantitative trait locus (QTL) mapping (Satagopan et al., 1996; Uimari et al., 1996), and linkage mapping (Suh et al., 2003).

1.12 Neural Networks

Artificial Neural Network (ANN) is an information processing model which imitates the biological nervous system. The Network consists of a large number of inter-connected nodes called neurons, which work together in solving a problem. The main idea behind ANNs is learning by example. Every ANN is designed specifically to solve a particular problem – such as Data classification, Pattern Recognition, Image Understanding, etc. ANNs can perform adaptive learning (ability to do tasks based on training or experience), self-organizing (creates its own representation of the information it processes), and fault tolerance. ANNs are particularly suited for solving complex problems with non-linear relationship, which are plentiful in biology. For instance, ANNs have been used for elucidating drug mechanisms (van Osdol et al., 2000; Weinstein et al., 1992), finding active antisense oligonucleotides (Giddings et al., 2002), predicting tertiary structure of proteins (Stolorz et al., 1992), etc. A recent interesting example illustrates the feasibility of using ANNs for designing genome-wide short interfering RNAs (siRNAs) (Huesken et al., 2005).

Several resources, both free and commercial are available, that aid in the development of ANNs. For example, the Stuttgart neural network simulator (SNNS at http://www-ra.informatik.uni-tuebingen.de/SNNS/) is a widely used software simulator with graphical network editing and visualization tools for developing neural networks on unix systems. Also, the mathworks neural network toolbox (http://www.mathworks.com/products/neuralnet/) for MATLAB is a commercially available set of functions for the design, implementation, visualization, and simulation of neural networks.

1.13 Hidden Markov Model

Hidden Markov models (HMMs) are normally used to find patterns that appear over a space of time. For instance, if we are interested in deducing the weather from a piece of seaweed – “soggy” seaweed implies wet weather and “dry” seaweed implies dry weather. If the seaweed is in an intermediate state, say, “damp”, we are not sure. Another consideration (basis), which can be used to deduce the current weather, would be the weather on the previous day (previous state). So in this example, we have two states – the observed state (seaweed) and the hidden state (weather). Hence by combining the previous state with the current state, we may be able to predict the next state.

HMMs have been used widely in pattern recognition in strings of indeterminate length, the prime examples of which include the DNA and protein sequences. More specifically, HMMs have been successfully applied for homology detection in proteins, trans-membrane helix

predictions (Krogh et al., 2001), gene predictions (Birney and Durbin, 2000), detection of CpG islands (Pachter and Sturmfels, 2004) and even for predicting phosphorylation sites on proteins (Senawongse et al., 2005). A number of online resources are available which use HMMs to solve important biological problems. For instance, HMMER (http://hmmer.janelia.org/) is a package that can be useful for protein sequence analysis, TMHMM (http://www.cbs.dtu.dk/services/TMHMM/) for prediction of trans-membrane helices, and HMM genie (http://www.fruitfly.org/seq_tools/genie.html) for gene prediction.

1.14 Support Vector Machines

Support vector machines (SVMs), a supervised learning method, was originally proposed by Vapnik in the late 1970s (Vapnik, 1979) and has been receiving increasing attention in the recent years (Vapnik and Chapelle, 2000). Given a set of data points that belong to two classes, an SVM attempts to find a hyperplane such that maximum possible numbers of points of the same class remain on the same side and concomitantly maximizing the distance of either class from the hyperplane. Understandably, training an SVM on a large dataset with many classes can be slow. SVMs have found a number of applications (Yang, 2004), some of which include classification of high throughput gene expression data (Brown et al., 2000), prediction of subcellular localization of proteins (Yu et al., 2006), predicting protein stability changes (Capriotti et al., 2005), enzyme family classifications, (Cai et al., 2004), etc. List of useful URLs related to data mining in bioinformatics provided in Table 3.1.

2. Exercises

1. Most of the machine learning methods defined above often results in a slightly different classification of the same data. Furthermore, one may need to optimize parameters/ weights used for setting up calculation. Sensitivity, Specificity, ROC curve, Cross-validation, Boot-strap are some of the metrics/methods one can adopt to understand performance of different classification methods. Define these and provide example of how these can be used for performance comparison.
2. You are interested in an automated text classifier that will read abstracts of publications by an individual and classify him into one of the fields “Bioinformatician”, “Biologist”, “Mathematician”, or “Computer Scientist”. To do this, you need abstracts of publications (in relevant field) that can be obtained from different journals over

the web: Journal of Biological Chemistry (http://www.jbc.org), PubMed (http://www.ncbi.nlm.nih.gov), Bioinformatics Journal (http://bioinformatics.oxfordjournals.org), Citeseer (http://citeseer.ist.psu.edu), etc. Implement a naïve Bayes text classifier that can be trained using abstracts from these fields. Use your own publication or your friends to automatically interpret your field of study. (Sample implementation in Perl is available at http://www.ddj.com/development-tools/184406064).

3. You are interested in identifying closely related amino acids in terms of their physical-chemical property. So you would like to perform a cluster analysis. Using the different properties for amino acids available at APDBase (http://www.rfdn.org/bioinfo/APDbase/index.html) and the software tool for clustering HCE (http://www.cs.umd.edu/hcil/hce/) perform analysis using different methods available in the tool. What amino acids are similar in their physical-chemical properties?

3. Useful Web-Pointers

Table 3.1 Useful URLs for data mining and statistical methods used in bioinformatics

Comments	URL
Data mining concepts and tutorials	http://www.thearling.com/text/dmwhite/dmwhite.htm
Statistical concepts, data mining techniques, machine learning methods	http://www.statsoft.com/
Hidden-Markov model introduction	http://www.comp.leeds.ac.uk/roger/HiddenMarkovModels/html_dev/main.html
General math & statistics	http://mathworld.wolfram.com
Electronic statistic book	http://www.xplore-stat.de/ebooks/ebooks.html
Open source Statistical programming package	http://lib.stat.cmu.edu/R/CRAN/
Open source numerical computation package	http://www.octave.org
Free encyclopedia with detailed links to	http://www.wikipedia.org

Comments	URL
Bioinformatics methods, statistical, and other computational methods	
Genetic algorithm tutorial	http://samizdat.mines.edu/ga_tutorial/
Data analysis methods and definitions	http://www.itl.nist.gov/div898/handbook/index.htm
Support vector machine program and theory	http://svmlight.joachims.org
Kernel Machines, theory, and links to programs	http://www.kernel-machines.org

References

Berry, D. A. (1996). Statistics: A Bayesian Perspective, Wadsworth Publishing.

Birney, E., and Durbin, R. (2000). Using GeneWise in the Drosophila annotation experiment, Genome Res *10*, 547–8.

Brown, M. P., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C. W., Furey, T. S., Ares, M., Jr., and Haussler, D. (2000). Knowledge-based analysis of microarray gene expression data by using support vector machines, Proc Natl Acad Sci U S A *97*, 262–7.

Cai, C. Z., Han, L. Y., Ji, Z. L., and Chen, Y. Z. (2004). Enzyme family classification by support vector machines, Proteins *55*, 66–76.

Capriotti, E., Fariselli, P., Calabrese, R., and Casadio, R. (2005). Predicting protein stability changes from sequences using support vector machines, Bioinformatics *21 Suppl 2*, ii54–ii58.

Coberley, C., Elashoff, M., and Mertz, L. (2004). Match/X, A gene expression pattern recognition algorithm used to identify genes which may be related to CDC2 function and cell cycle regulation, Cell Cycle *3*, 804–10.

Di Cara, A., Schmidt, K., Hemmings, B. A., and Oakeley, E. J. (2005). PromoterPlot: a graphical display of promoter similarities by pattern recognition, Nucleic Acids Res *33*, W423–6.

Eddy, S. R. (2004). What is Bayesian statistics? Nat Biotechnol *22*, 1177–8.

Giddings, M. C., Shah, A. A., Freier, S., Atkins, J. F., Gesteland, R. F., and Matveeva, O. V. (2002). Artificial neural network prediction of antisense oligodeoxynucleotide activity, Nucleic Acids Res *30*, 4295–304.

Hakak, Y., Walker, J. R., Li, C., Wong, W. H., Davis, K. L., Buxbaum, J. D., Haroutunian, V., and Fienberg, A. A. (2001). Genome-wide expression analysis reveals dysregulation of myelination-related genes in chronic schizophrenia, Proc Natl Acad Sci U S A *98*, 4746–51.

Hayward, G., and Davidson, V. (2003). Fuzzy logic applications, Analyst *128*, 1304–6.

Huesken, D., Lange, J., Mickanin, C., Weiler, J., Asselbergs, F., Warner, J., Meloon, B., Engel, S., Rosenberg, A., Cohen, D*., et al.* (2005). Design of a genome-wide siRNA library using an artificial neural network, Nat Biotechnol *23*, 995–1001.

Ibbini, M. S., and Masadeh, M. A. (2005). A fuzzy logic based closed-loop control system for blood glucose level regulation in diabetics, J Med Eng Technol *29*, 64–9.

Krogh, A., Larsson, B., von Heijne, G., and Sonnhammer, E. L. (2001). Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes, J Mol Biol *305*, 567–80.

Mathura, V. S., Schein, C. H., and Braun, W. (2003). Identifying property based sequence motifs in protein families and superfamilies: application to DNase-1 related endonucleases, Bioinformatics *19*, 1381–90.

Oldfield, T. (2002). Pattern-recognition methods to identify secondary structure within X-ray crystallographic electron-density maps, Acta Crystallogr D Biol Crystallogr *58*, 487–93.

Pachter, L., and Sturmfels, B. (2004). Parametric inference for biological sequence analysis, Proc Natl Acad Sci U S A *101*, 16138–43.

Phuong, N. H., and Kreinovich, V. (2001). Fuzzy logic and its applications in medicine, Int J Med Inform *62*, 165–73.

Ressom, H., Reynolds, R., and Varghese, R. S. (2003). Increasing the efficiency of fuzzy logic-based gene expression data analysis, Physiol Genomics *13*, 107–17.

Satagopan, J. M., Yandell, B. S., Newton, M. A., and Osborn, T. C. (1996). A bayesian approach to detect quantitative trait loci using Markov chain Monte Carlo, Genetics *144*, 805–16.

Schneider, J., Peltri, G., Bitterlich, N., Neu, K., Velcovsky, H. G., Morr, H., Katz, N., and Eigenbrodt, E. (2003a). Fuzzy logic-based tumor marker profiles including a new marker tumor M2-PK improved sensitivity to the detection of progression in lung cancer patients, Anticancer Res *23*, 899–906.

Schneider, J., Peltri, G., Bitterlich, N., Philipp, M., Velcovsky, H. G., Morr, H., Katz, N., and Eigenbrodt, E. (2003b). Fuzzy logic-based tumor marker profiles improved sensitivity of the detection of progression in small-cell lung cancer patients, Clin Exp Med *2*, 185–91.

Seker, H., Odetayo, M. O., Petrovic, D., and Naguib, R. N. (2003). A fuzzy logic based-method for prognostic decision making in breast and prostate cancers, IEEE Trans Inf Technol Biomed *7*, 114–22.

Selbig, J., Mevissen, T., and Lengauer, T. (1999). Decision tree-based formation of consensus protein secondary structure prediction, Bioinformatics *15*, 1039–46.

Senawongse, P., Dalby, A. R., and Yang, Z. R. (2005). Predicting the phosphorylation sites using hidden Markov models and machine learning methods, J Chem Inf Model *45*, 1147–52.

Shoemaker, J. S., Painter, I. S., and Weir, B. S. (1999). Bayesian statistics in genetics: a guide for the uninitiated, Trends Genet *15*, 354–8.

Sinsheimer, J. S., Lake, J. A., and Little, R. J. (1996). Bayesian hypothesis testing of four-taxon topologies using molecular sequence data, Biometrics *52*, 193–210.

States, D. J., and Botstein, D. (1991). Molecular sequence accuracy and the analysis of protein coding regions, Proc Natl Acad Sci U S A *88*, 5518–22.

Stolorz, P., Lapedes, A., and Xia, Y. (1992). Predicting protein secondary structure using neural net and statistical methods, J Mol Biol *225*, 363–77.

Suh, Y. J., Ye, K. Q., and Mendell, N. R. (2003). A method for evaluating the results of Bayesian model selection: application to linkage analyses of attributes determined by two or more genes, Hum Hered *55*, 147–52.

Szabo, A., Boucher, K., Carroll, W. L., Klebanov, L. B., Tsodikov, A. D., and Yakovlev, A. Y. (2002). Variable selection and pattern recognition with gene expression data generated by the microarray technology, Math Biosci *176*, 71–98.

Taguchi, Y. H., and Oono, Y. (2005). Relational patterns of gene expression via non-metric multidimensional scaling analysis, Bioinformatics *21*, 730–40.

Uimari, P., Thaller, G., and Hoeschele, I. (1996). The use of multiple markers in a Bayesian method for mapping quantitative trait loci, Genetics *143*, 1831–42.

Valafar, F. (2002). Pattern recognition techniques in microarray data analysis: a survey, Ann N Y Acad Sci *980*, 41–64.

van Osdol, W. W., Myers, T. G., and Weinstein, J. N. (2000). Neural network techniques for informatics of cancer drug discovery, Methods Enzymol *321*, 369–95.

Vapnik, V. (1979). Estimation of Dependences Based on Empirical Data (in Russian). Nauka Moscow.

Vapnik, V., and Chapelle, O. (2000). Bounds on error expectation for support vector machines, Neural Comput *12*, 2013–36.

Weinstein, J. N., Kohn, K. W., Grever, M. R., Viswanadhan, V. N., Rubinstein, L. V., Monks, A. P., Scudiero, D. A., Welch, L., Koutsoukos, A. D., Chiausa, A. J., and et al. (1992). Neural computing in cancer drug development: predicting mechanism of action, Science *258*, 447–51.

Yang, Z. R. (2004). Biological applications of support vector machines, Brief Bioinform *5*, 328–38.

Yang, Z. R., Wang, L., Young, N., Trudgian, D., and Chou, K. C. (2005). Pattern recognition methods for protein functional site prediction, Curr Protein Pept Sci *6*, 479–91.

Yu, X., Cao, J., Cai, Y., Shi, T., and Li, Y. (2006). Predicting rRNA-, RNA-, and DNA-binding proteins from primary structure with support vector machines, J Theor Biol *240*, 175–84.

Chapter 4

Biological Sequence Databases

Meena Sakharkar [1], Pandjassarame Kangueane [2], and Venkatarajan S. Mathura [3]

[1] *Nanyang Technological University, Singapore*
[2] *Biomedical Informatics, Pondicherry, India*
[3] *Roskamp Institute, 2040 Whitfield Avenue, Sarasota, Florida, USA*

Abstract: Biological data available today surpasses information content in several fields. It is critical to logically organize and disseminate these contents to end users. In this chapter, we learn about biological databases that serve as the gateway for researchers.

Key words: NCBI, SWISS-PROT, PDB, GenomeNet

1. Purpose

Biological portals and databases are important sources of sequence, structure, and other relevant information. Understanding contents of these databases will help in extracting knowledge relevant to your projects in an efficient way. In this chapter, we will detail some of the major biological databases and their contents.

2. Learning Objective

- Identify major biological databases and portals
- Understand their information content
- Perform structured queries and derive biological information

3. Introduction

The past few decades have witnessed a widespread application of computers for analysis and modeling of biological data. This currently has a huge

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_4,

impact on the practice of molecular biology and already gave birth to a new discipline called "Bioinformatics". A major aspect of this revolution is the storage, retrieval, and analysis of biological datasets maintained by centralized resources worldwide. Advancement in molecular biology techniques and high throughput methods has resulted in a dramatic increase in genomic and proteomic data. These include a wide range of information, such as macromolecular sequences and structures; genetic and physical genomic maps, polymorphisms; bibliographic information; molecular chemical properties, etc. Simultaneously, the rapid expansion of biomedical knowledge, reduction in computing costs, spread of internet access, and the recent emergence of high throughput structural and functional genomic technologies has led to a rapid growth of electronically available data. Submission of such data into public archives has led to numerous biological databases that can be accessed for querying and retrieving of necessary information by the scientific community. These databases store molecular information of multiple organisms and are, thus, reflections of the cellular and molecular organization of life. This chapter summarizes few such databases and their content. The Molecular Biology Database Collection by Micheal Galperin is a compendium of several databases (Galperin, 2005). All databases included in the Collection are freely available to the public. The 2005 update includes 719 databases. The databases are organized in a hierarchical classification that simplifies the process of finding the right database for any given task. The growing number of databases related to immunology, plant, and organelle research has been accommodated by separating them into three new categories. Almost all of these databases can be searched and retrieved by using high-quality bioinformatics tools based on several features for example sequence homology, map location, keyword, accession number, and other features in the records. Biological databases are constantly being updated and re-engineered to allow more powerful data query methods. A majority of these tools and databases are maintained in a highly integrated form by major organizations such as National Center for Biotechnology Information (NCBI) (Wheeler et al., 2005), European Molecular Biology Laboratory Nucleotide Sequence Database (EMBL) (Kanz et al., 2005), and DNA Data Bank of Japan (DDBJ) (Tateno et al., 2005).

Protein sequences were the first to be assembled into databases and made freely available. In the 1960s and 1970s, Margaret Dayhoff's pioneering work on protein evolution led to the distribution of the Protein Sequence Database, now well-known as the international Protein Information Resource (PIR) (George et al., 1986). Concurrently, in 1980, the first public releases of SWISS-PROT sequence database took shape (Bairoch and Boeckmann, 1991). Simultaneously, the first nucleic acid sequence

databases began to prosper in order to cope with the increasing quantities of sequence data being generated worldwide (e.g. GenBank, EMBL, and DDBJ) (Benson et al., 2005; Kanz et al., 2005; Tateno et al., 2005). The accumulated data was stored in the first genomic databases such as GenBank, EMBL, and DDBJ and novel computational methods were developed for further analysis of the collected data (e.g. sequence similarity searches, functional, and structural predictions).

Databases in general can be classified in to *primary*, *secondary,* and *composite databases*. A primary database contains information of the sequence or structure alone. Examples of these include SWISS-PROT and PIR (Wu et al., 2003) for protein sequences, GenBank, EMBL ,and DDBJ for Genome sequences and the Protein Databank PDB (Berman et al., 2000) for protein structures.

A secondary database contains derived information from the primary database. A secondary sequence database contains information like the conserved sequence, signature sequence, and active site residues of the protein families arrived by multiple sequence alignment of a set of related proteins. A secondary structure database contains entries of the PDB in an organized way. These contain entries that are classified according to their structure such as all alpha proteins, all beta proteins, etc. These also contain information on conserved secondary structure motifs of a particular protein. Some of the secondary database created and hosted by various researchers at their individual laboratories include SCOP (Lo Conte et al., 2000), developed at Cambridge University, CATH (Pearl et al., 2005), developed at University College of London, PROSITE (Hulo et al., 2004) of Swiss Institute of Bioinformatics, and eMOTIF (Huang and Brutlag, 2001) at Stanford and the conserved domain database search, CDART (Geer et al., 2002), and protein interactions at NCBI.

Several secondary databases use data from various primary databases and generate new data that may be organism specific or interest specific. These databases provide an integrated and panoramic view for the sequence of interest. A few secondary databases on molecular evolution are listed in later sections.

3.1 Genomic Sequence Databases – GenBank, EMBL, DDBJ

GenBank, (Benson et al., 2005) was built by the National Center for Biotechnology Information (NCBI), is part of the International Nucleotide Sequence Database Collaboration, along with its two partners, the DNA Data Bank of Japan (DDBJ, Mishima, Japan) and the European Molecular Biology Laboratory (EMBL) nucleotide database from the European Bioinformatics Institute (EBI, Hinxton, UK). GenBank is a comprehensive

database that contains publicly available DNA sequences for more than 165,000 named organisms, obtained primarily through submissions from individual laboratories and batch submissions from large-scale sequencing projects. Daily data exchange with the EMBL Data Library in the UK and the DNA DataBank of Japan helps to ensure worldwide coverage. GenBank is accessible through NCBI's retrieval system, Entrez, which integrates data from the major DNA and protein sequence databases along with taxonomy, genome, mapping, protein structure, and domain information, and the biomedical journal literature via PubMed (McEntyre and Lipman, 2001). BLAST provides sequence similarity searches of GenBank and other sequence databases. Nearly 45 million sequences are available and it's growing exponentially. GenBank releases new sequences every two months and can be downloaded from ftp://ftp.ncbi.nih.gov. Incremental updates are also available that can be obtained from ftp://ftp.ncbi.nih.gov/genbank/daily-nc. Each sequence is annotated that contains information about the literature reference, size, organism, sequence features, and protein translations, if available. The sequences are arranged under several divisions like ENV, EST, Genome survey sequences (GSS), Plant sequences (PLN), rodents ROD, Sequence tagged site (STS), etc. GenBank provides an interface to authors who would like to submit new sequences. BankIt and SEQUIN are two programs that can be used for submitting author-annotated sequences, which are then automatically indexed to be included in the GenBank. Entries under different divisions are indexed using symbols, for example EST sequences can be found in gbest*.seq files and GSS sequences in gbgss*.seq.

GenBank entry includes a concise description of the sequence, the scientific name, and taxonomy of the source organism, bibliographic references, and a table of features (http://www.ncbi.nlm.nih.gov/collab/FT/index.html) listing areas of biological significance, such as coding regions and their protein translations, transcription units, repeat regions, and sites of mutations or modifications. Table 4.1 lists major sequence databases and Table 4.2 lists different sequence divisions in GenBank.

3.2 Protein Sequence Databases

The most significant protein databases include the Swiss Protein Databank (SWISS-PROT) (Boeckmann et al., 2003), the translation of the DNA sequences in EMBL (TrEMBL), Protein Information Resource (PIR), the Munich Information Center for Proteins (MIPS) (Mewes et al., 2004), and the 3D structures in the Protein DataBank (PDB) (Berman et al., 2000). The rate of growth of the protein databases has been more linear compared to the DNA databases.

Table 4.1 List of URL for major biological databases

Biological database	Major components	URL
National Center for Biological Information	Pubmed, CDD, COG, OMIM, Genomes, CGAP, dbEST, dbGSS, dbMHC, dbSNP, dbSTS, GenBank, Genes, HomoloGene, MeSH, MGC, MMDB, OMSSA, OMSSA, PubCHEM, RefSeq, UNIGENE, VAST, GEO	http://www.ncbi.nlm.nih.gov http://www.ncbi.nlm.nih.gov/Sitemap/AlphaList.html
European Bioinformatics Institute	BioMart, ChEBI, EMBL-SVA, UniProt, ArrayExpress, ASD, CSA, GOA, IntAct, IntEnz, DALI, MSD, MSDchem, MSDlite, RESID	http://www.ebi.ac.uk/services
http://www.expasy.org	PROSITE, SWISS-2DPAGE, SWISS-3DIMAGE, Ashbya, ENZYME, Biolinks	http://www.expasy.org http://www.expasy.org/links.html#Proteins
http://www.ensembl.org http://www.genome.jp/	Genome database KEGG, DBGET GLYCAN, BRITE, CYORF, BSORF, LIGAND	http://www.ensembl.org http://www.genome.jp

Table 4.2 Different division under which sequences are made available in GenBank

CODE	Description
PRI	primate sequences
ROD	rodent sequences
MAM	other mammalian sequences
VRT	other vertebrate sequences
INV	invertebrate sequences
PLN	plant, fungal, and algal sequences
BCT	bacterial sequences
VRL	viral sequences
PHG	bacteriophage sequences
SYN	synthetic sequences
UNA	unannotated sequences
EST	EST sequences (expressed sequence tags)
PAT	patent sequences
STS	STS sequences (sequence tagged sites)
GSS	GSS sequences (genome survey sequences)
HTG	HTGS sequences (high throughput genomic sequences)
HTC	HTC sequences (high throughput cDNA sequences)
ENV	Environmental sampling sequences

3.2.1 SWISS-PROT

SWISS-PROT is a protein sequence knowledgebase and has direct links to specialized databases with minimal redundancy (Boeckmann et al., 2003). The data consists principally of the amino acid sequence, the protein name (description), taxonomic data, and citation information. SWISS-PROT provides cross-references to external data collections such as the underlying DNA sequence entries in the DDBJ/EMBL/GenBank nucleotide sequence databases, 2D and 3D protein structure databases, various protein domain and family characterization databases, posttranslational modification (PTM) databases, species-specific data collections, variant databases, and disease databases. SWISS-PROT is gradually being enhanced by the addition of a number of features that are specifically intended for researchers working on human genetic diseases, such as links to human gene databases: OMIM (Hamosh et al., 2005), GeneCards (Safran et al., 2002), GeneLynx (Lenhard et al., 2003), Genew (Wain et al., 2004) as well as to many gene-specific mutation databases. SWISS-PROT and TrEMBL can be obtained by anonymous FTP from the ExPASy server ftp.expasy.org and EBI server ftp.ebi.ac.uk/pub/. Further information as how to obtain weekly updates and complete data sets in various formats is available at http://www.expasy.org/sprot/download.html.

3.2.2 PIR

The Protein Information Resource (PIR) serves as an integrated public resource of functional annotation of protein data to support genomic/proteomic research and scientific discovery (George et al., 1986). It is a non-redundant, expertly annotated, fully classified, and extensively cross-referenced protein sequence database. The PIR anonymous FTP site (ftp://nbrfa.georgetown.edu/pir_databases) provides direct file transfer.

3.2.3 PDB

The PDB is the single worldwide repository for the processing and distribution of 3-D structure data of large molecules of proteins and nucleic acids (Berman et al., 2000). It is available at: http://www.pdb.org/. The PDB contains 31123 structures as of 31 May 2005.

3.3 Secondary Databases on Molecular Evolution

3.3.1 EXINT

With the accumulation of sequence data, information on the exon/intron organization of eukaryotic genes is becoming widely available. However,

the retrieval of this information, particularly on a large scale basis, is a difficult task. ExInt (Sakharkar et al., 2000) is a database of all intron-containing genes from eukaryotes present in GenBank. It collects information about the exon/intron organization of eukaryotic genes present in GenBank and organizes the data in a retrieval form available on the WWW. ExInt has been divided into four subsets: predicted entries, experimental entries, organellar, and nuclear genes. The database is available at: http://sege.ntu.edu.sg/wester/exint/.

3.3.2 MIDB

MIDB is a database containing discordant intron positions in homologous genes (Sakharkar et al., 2000). Discordant intron positions are those that are either closely located in homologous genes (within a window of 10 nucleotides) or an intron position that is present in one gene but not in any of its homologs. The MIDB database aims at systematically collecting information about mismatched introns in the genes from GenBank and organizing it into a form useful for understanding the genomics and dynamics of introns thereby helping understand the evolution of genes. MIDB allows examining of intron movements and allows mapping of intron positions from homologous proteins onto a single sequence. The database is of potential use for molecular biologists in general and for researchers who are interested in gene evolution and eukaryotic gene structure. Partial analysis of this database allowed us to identify a few putative cases of intron sliding. The database is available at http://sege.ntu.edu.sg/ wester/midb/.

3.3.3 GSEGE

Eukaryotic genes are either “intron containing” or “intronless”. Eukaryotic “intronless” genes are interesting datasets for comparative genomics and evolutionary studies. Genome SEGE is a database for “intronless” genes in completely sequenced eukaryotic genomes (Sakharkar and Kangueane, 2004). Eukaryotic “intronless” genes are extracted from nine completely sequenced genomes (four of which are unicellular and five of which are multi-cellular). The database provides information on the distribution of “intronless” genes in different genomes together with their length distributions in each genome. Additionally, the search tool provides pre-computed PROSITE motifs for each sequence in the database with appropriate hyperlinks to InterPro. A search facility is also available through the web server. GSEGE is available at http://sege.ntu.edu.sg/wester/intronless.

References

Bairoch, A. and Boeckmann, B. (1991) The SWISS-PROT protein sequence data bank. *Nucleic Acids Res* **19 Suppl**, 2247–9.

Benson, D.A., Karsch-Mizrachi, I., et al. (2005) GenBank. *Nucleic Acids Res* **33**(Database issue), D34–8.

Berman, H.M., Westbrook, J., et al. (2000) The protein data Bank. *Nucleic Acids Res* **28**(1), 235–42.

Boeckmann, B., Bairoch, A., et al. (2003) The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res* **31**(1), 365–70.

Galperin, M.Y. (2005) The Molecular Biology Database Collection: 2005 update. *Nucleic Acids Res* **33**(Database issue), D5–24.

Geer, L.Y., Domrachev, M., et al. (2002) CDART: protein homology by domain architecture. *Genome Res* **12**(10), 1619–23.

George, D.G., Barker, W.C., et al. (1986) The protein identification resource (PIR). *Nucleic Acids Res* **14**(1), 11–5.

Hamosh, A., Scott, A.F., et al. (2005) Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. *Nucleic Acids Res* **33**(Database issue), D514–7.

Huang, J.Y. and Brutlag, D.L. (2001) The EMOTIF database. *Nucleic Acids Res* **29**(1), 202–4.

Hulo, N., Sigrist, C.J., et al. (2004) Recent improvements to the PROSITE database. *Nucleic Acids Res* **32**(Database issue), D134–7.

Kanz, C., Aldebert, P., et al. (2005) The EMBL Nucleotide Sequence Database. *Nucleic Acids Res* **33**(Database issue), D29–33.

Lenhard, B., Wahlestedt, C., et al. (2003) GeneLynx mouse: integrated portal to the mouse genome. *Genome Res* **13**(6B), 1501–4.

Lo Conte, L., Ailey, B., et al. (2000) SCOP: a structural classification of proteins database. *Nucleic Acids Res* **28**(1), 257–9.

McEntyre, J. and Lipman, D. (2001) PubMed: bridging the information gap. *Cmaj* **164**(9), 1317–9.

Mewes, H.W., Amid, C., et al. (2004) MIPS: analysis and annotation of proteins from whole genomes. *Nucleic Acids Res* **32**(Database issue), D41–4.

Pearl, F., Todd, A., et al. (2005) The CATH Domain Structure Database and related resources Gene3D and DHS provide comprehensive domain family information for genome analysis. *Nucleic Acids Res* **33**(Database issue), D247–51.

Safran, M., Solomon, I., et al. (2002) GeneCards 2002: towards a complete, object-oriented, human gene compendium. *Bioinformatics* **18**(11), 1542–3.

Sakharkar, M.K. and Kangueane, P. (2004) Genome SEGE: a database for 'intronless' genes in eukaryotic genomes. *BMC Bioinformatics* **5**, 67.

Sakharkar, M., Long, M., et al. (2000) ExInt: an Exon/Intron database. *Nucleic Acids Res* **28**(1), 191–2.

Tateno, Y., Saitou, N., et al. (2005) DDBJ in collaboration with mass-sequencing teams on annotation. *Nucleic Acids Res* **33**(Database issue), D25–8.

Wain, H.M., Lush, M.J., et al. (2004) Genew: the Human Gene Nomenclature Database, 2004 updates. *Nucleic Acids Res* **32**(Database issue), D255–7.

Wheeler, D.L., Barrett, T., et al. (2005) Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res* **33**(Database issue), D39–45.

Wu, C.H., Yeh, L.S., et al. (2003) The Protein Information Resource. *Nucleic Acids Res* **31**(1), 345–7.

Chapter 5

Biological Sequence Search and Analysis

Venkatarajan S. Mathura

Roskamp Institute, 2040 Whitfield Avenue, Sarasota, Florida 34243, USA

Abstract: Protein and genomic sequence analyses helps in understanding the structure, function, and organization of cellular systems. Important features of genes include identifying promoter regions, protein-coding regions, and intron-exon boundaries. Protein sequence analysis involves identifying functional motifs and patterns. Sequence search tools help in identifying similar sequences in protein and genomic databases. Here, we will discuss bioinformatics tools that help in biological sequence searches and analyses.

Key words: BLAST, Dynamic programming, CLUSTALW, Sequence Motifs

1. Purpose

Proteins that are coded by genes, achieve complex functions in biological organisms. The DNA is composed of four different nucleotides and can be represented as a string. Similarly, proteins can be represented as a string composed of 20 amino acid alphabets. Studying protein and DNA sequences involve analyzing these strings. This chapter mostly deals with studying protein sequence features using different software tools.

2. Learning Objectives

- Sequence search algorithms and tools
- Multiple sequence alignment tools
- Sequence motif identification

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_5,

3. Introduction

3.1 Similarity Matrices and Alignment

Sequence search programs take a candidate sequence and searches a database. Such tools use a scoring function to align a query sequence globally or partially with a member in the database. Sequence alignment involves mapping corresponding positions in two sequence strings. If two strings are identical then the alphabet at every position will match the alphabet in another string. Aligning two biological sequences is not as simple as aligning two strings due to degeneracy among constituting alphabets. For example, during evolution of a protein, a particular position may be replaced by a similar amino acid (e.g. isoleucine replaces leucine), which has a different alphabet. So, one must have a quantitative measure for the chance of a particular amino acid or nucleotide to be replaced by another. Further biological sequence alignment should accommodate insertions or deletions of positions, which are likely during evolution. Thus, an alignment of two biological strings involves mapping positions that have identical or similar alphabets and accommodating gaps at positions where insertion or deletion events have occurred. Biological sequences can be of unequal lengths and their *a priori* equivalent positions are unknown. Thus, aligning two sequences is a complex task that attempts to compare every position and trace a path that has a high score.

The similarity matrix (also called substitution matrix) is constructed based on the observed exchange frequencies among amino acids. It is a 20×20 matrix with entries for pair-wise exchanges. Protein sequences that are closely evolved conserve both structure and function (Wilson et al., 2000). If two sequences evolve from a common parent then they are called *homologous sequences* or related by *homology*. Homologous sequences need not be 100% identical and can have equivalent positions occupied by a similar amino acid (Abagyan and Batalov, 1997; Rost, 1999). Core residues that are important for their structure and function are conserved. If one can identify a family of such closely related sequences, then it is possible to calculate a substitution score from the observed exchange frequencies among amino acids. Let Q_{ij} represent the frequency of exchange between amino acids *i* and *j*, and the background or natural frequency of amino acids *i* and *j* be P_i and P_j, respectively. Then, the substitution score S_{ij} is related to Q_{ij} and the background frequencies by the formula (Lipman et al., 1984):

$$S_{ij} = 1/\lambda * \ln (Q_{ij}/P_i*P_j) \qquad \text{Eq. (1)}$$

where λ represents the scaling factor and is generally set to log value of 2 (S_{ij} is expressed in bits) or to 1 (S_{ij} is expressed in nats).

3.1.1 Mutation Data Matrix or PAM Matrix

Margaret Dayhoff derived the first substitution matrix from the naturally observed frequency of residues (Dayhoff and Schwartz, 1978). She manually constructed the alignments of several protein families that have nearly identical sequences and phylogenetic trees from which she calculated the substitutions that occurred at each diverging branches. Thus, from observed mutations she scored the relative frequency in which two amino acids are exchanged. She introduced *point-accepted mutation* (PAM) as a unit of evolutionary divergence. One PAM unit is defined as 1 amino acid replacement among 100 positions in a protein sequence. Substitution matrix constructed based on such sequence sets were (PAM1 matrix) extrapolated further for higher evolutionary divergence (by multiplying lower order PAMs). Such extrapolation was possible after an assumption of a Markov model or independent mutations among residue positions. Thus, PAM250 corresponds to exchanges that can be expected among highly divergent sequences. Highly divergent sequences generally require a longer alignment to infer homology compared to closely related sequences. PAM250 is preferred to identify divergent protein sequences that have weaker similarities and generally require longer alignments. Lower order PAM matrices are preferred for aligning short regions with higher similarity. PAM matrices are derived based on extrapolation of observed frequency of exchanges using a global alignment of closely related sequence families. Although such extrapolation is valid under a model for evolutionary process, this method has a limitation as it assumes an even pressure for mutation along the entire length of a protein sequence.

3.1.2 BLOSUM

Steven and Jorja Henikoff constructed BLOSUM (Blocks Substitution Matrix) set of matrices (Henikoff and Henikoff, 1992). Instead of deriving exchange frequency of amino acids based on the global alignment of a protein family, they used an un-gapped local alignment or blocks. These blocks represent conserved segments within the protein family. For an exchange of an amino acid i with another amino acid j, they calculated frequency of exchange Q_{ij} and individual background frequencies P_i and P_j. The individual scores were calculated as log odds as in Equation (1). By restricting the count of amino acid exchanges among sequences that shared 62% identical residues, they were able to derive BLOSUM62 matrix. Highly divergent sequences that share few identical sequences were modeled using

a lower identity cutoff. Thus, BLOSUM30 matrix should be used instead of BLOSUM62 when dealing with highly divergent sequences. Thus, BLOSUM series is constructed using the observed exchanges among protein sequences rather by an extrapolation technique as followed in building PAM matrices.

3.1.3 Other Substitution Matrices

Several substitution matrices were derived based on the original PAM or BLOSUM concepts. Many of them used an exhaustive set of sequence alignments and protein families to calculate the log-odds. Gonnet et al. used an exhaustive list of available sequences to derive a substitution matrix called GONNET (Gonnet et al., 1992). Alignments of highly divergent sequences are hard to construct solely based on sequence similarity. If structures of proteins are available, one can use the co-ordinates to align these sequences. Such alignments can be accurate even if the aligned sequences are highly divergent or have low homology. Substitution matrices derived using structural alignment includes Overington (Johnson and Overington, 1993), SDM by Sippl et al. (Prlic et al., 2000), Sub-structural matrix (Naor et al., 1996), etc. Amino acids that share similar physical-chemical properties often exchange with higher frequencies. Physico-chemical based substitution matrices or similarity matrices have been attempted. Such methods derive a distance metric based on several properties for individual amino acids. The individual scores are derived by inverting distances and used as a measure of similarity. Limitations of these methods include inclusion of sufficient number or properties and selecting appropriate weights. A multidimensional scaling method was applied to 237 properties that eliminated redundancy among different properties resulting in five orthogonal descriptors (Venkatarajan and Braun, 2001). Euclidean distances calculated using these descriptors could be used as a measure of dissimilarity to compare protein sequences. A large collection of substitution matrices are available at AAindex website (http://www.genome.jp/aaindex/).

3.2 Sequence Search and Pair-Wise Alignment

As mentioned previously, sequences that have evolved from a common ancestor share similar amino acids at equivalent residue positions. These positions cannot be interpreted in a straight forward manner because of the insertion and deletion of amino acids during the evolution of the daughter sequences. By pair-wise alignment, we mean arranging sequences of two proteins/genes with one-to-one correspondence of equivalent residue

positions that are evolutionarily conserved. If one considers amino acids as alphabets then alignment is equivalent to matching substrings within two sets of words. If an identity matrix is used, then identical residues aligned will have a score of one and the number of identical positions will be equivalent to a score. For example,

```
Sequence1      AAKLV--AKKL
Sequence2      AAKLVQQAKKL
```

Here we aligned two sequences that match nine equivalent positions. In general, scores based on identical matrices are trivial and are not sufficient to properly align two protein sequences. Amino acids can be exchanged without affecting its function. For example, a hydrophobic amino acid Ile can be replaced with another hydrophobic amino acid Val at a higher frequency, than by a charged residue Arg. Our previous notes on substitution matrix convey that pair-wise entries for amino acids may have positive or negative values. Positive scores mean a favorable substitution or in other words higher frequency of observed exchanges. Hence, optimal alignment using a substitution matrix becomes a complex task involving scoring several position equivalents (or amino acid pairs) and identifying the best arrangement. Such tasks can be handled by dynamic programming methods that involve integrating optimal solutions for sub problems and arriving at a solution in a more efficient manner. Sequence alignment methods implement dynamic programming that enables to divide complex sequence alignment tasks into manageable subtasks. Such divide and conquer methods enable optimal solution in an efficient runtime. A simple method to understand and visualize alignment is a dot plot. Graphically, it's a dot matrix of different shades reflecting scores (or average score over sliding windows). A continuous diagonal dot represents regions of high similarity. A dot plot can be generated using an applet available at http://www.isrec.isb-sib.ch/java/dotlet/Dotlet.html. By connecting regions of continuous dots, one can find an optimal alignment along the sequence.

3.3 Global Alignment Using Needleman-Wunsch Algorithm

Global alignment involves an arrangement of two sequences with their equivalent evolutionarily conserved residue positions along the entire length. Needleman-Wunsch developed a global alignment method using dynamic programming (Needleman and Wunsch, 1970). In this method, alignment of two sequences $X_{i=1..m}$ and $Y_{j=1..n}$ of length m and n respectively involve an initialization of matrix of size $m+1$ and $n+1$. The first row and column are

each filled with gap penalties. Each cell in the matrix has a score, which is filled using the following function:

$$M_{i,j} = \max \begin{cases} M_{i-1,\,j-1} + S_{ij} \\ M_{i,\,j-1} + G \\ M_{i-1,\,j} + G \end{cases}$$

M_{ij} is the score at a matrix cell *(i,j)* and $M_{i-1,\,j-1}$ is the score at the upper diagonal cell. S_{ij} is the score for amino acids X_i and Y_j from the substitution matrix like PAM or BLOSUM. G is a gap-opening penalty. Gaps are indicated by '-' in an alignment and represent insertion or deletion (in-del) event occurred in one of sequence. A penalty factor is added to open new gaps and this is set to –8 for protein sequence alignment using BLOSUM62 matrix. The gap open penalty depends on the type of matrix used. Penalties for extending gaps in a sequence are set to –2 and are modeled by multiplying the number of contiguous gaps found after opening by 1. Such a model for gaps is referred by affine gap penalty model. The global alignment uses a gap-opening penalty G. $M_{i,\,j-1}$ is the score of the adjacent cell present on the top of the current cell whereas $M_{i,\,j-1}$ is the score of adjacent cell present in the right side of the current cell. If a maximum value is found to be $M_{i,\,j-1}+G$ then the residue Y_j is aligned with a gap. Every time the maximum value is identified, a pointer is stored that maps the current cell to any of the three cells that gave rise to the current cell value. If all three values are equal, then all possible combinations of alignment are possible at that particular residue position. The final step is tracing the path that gave rise to the highest score starting with the bottom-left (representing C-terminal). This is recursively done at every cell and a final alignment is made. It is possible to obtain several possible alignments for two sequences, if there are cells along the trace back in the matrix contributed equally by all three adjacent cells. Let's align sequence A (RHEEIIIKVFFI) and sequence B (HHQKLVFF) using BLOSUM62 scoring matrix and a linear gap model with a gap penalty of –8 using Needleman-Wunsch algorithm. The optimal alignment is:

```
RHEEIIIKVFFI
HHQKL---VFF-
```

and the matrix is given in Figure 5.1.

		R	H	E	E	I	I	I	K	V	F	F	I
	0	-8	-16	-24	-32	-40	-48	-56	-64	-72	-80	-88	-96
H	-8	0	0	-8	-16	-24	-32	-40	-48	-56	-64	-72	-80
H	-16	-8	8	0	-8	-16	-24	-32	-40	-48	-56	-64	-72
Q	-24	-15	0	10	2	-6	-14	-22	-30	-38	-46	-54	-62
K	-32	-22	-8	2	11	3	-5	-13	-17	-25	-33	-41	-49
L	-40	-30	-16	-6	3	13	5	-3	-11	-16	-24	-32	-39
V	-48	-38	-24	-14	-5	6	16	8	0	-7	-15	-23	-29
F	-56	-46	-32	-22	-13	-2	8	16	8	0	-1	-9	-17
F	-64	-54	-40	-30	-21	-10	0	8	13	7	6	5	-3

Figure 5.1 Global alignment of two sequences (RHEEIIIKVFI, HHQKLVFF). The trace back is shown in shaded blocks.

3.4 Sequence Search Tools

Given a query sequence, we would like to know information about related sequences in protein or nucleic acid sequence databases. In order to infer homology relationship, one may need an alignment between the query sequence and a candidate sequence in the database. This alignment can be used for scoring similarity and can be used in ranking sequences based on similarity score. In order to efficiently search millions of sequences and produce results in a fast manner, several tools have been developed. Some of the tools are covered below.

3.4.1 Basic Local Alignment Search Tool (BLAST)

BLAST is a popular search tool designed by Alstchul et al. at NIH (Karlin and Altschul, 1990). It uses statistical methods to evaluate hits for their significance. In the first step, the program identifies sequences in the database that share common words of a pre-set size (k-tuple) and these matching words are extended to identify un-gapped common segments between two sequences. Segment pairs are extended only if they score higher than a pre-defined threshold, thus reducing time on trivial non-informative segments. Each of these segments (maximum in size) is scored using one of the substitution matrices as described above (BLOSUM62 is default) and the highest scoring segments or maximal scoring segment pairs (MSPs) are evaluated for their statistical significance by comparing MSPs with possible scores in a randomly generated sequence database.

An extreme value statistical distribution was used by Karlin and Altschul to model the HSPs (High Scoring Segment Pairs) during local alignment. If

m and *n* are the string sizes of two sequences then a score of *S* can be calculated over the block of local alignment (obtained from summation of corresponding substitution matrix scores over the alignment, please refer to the alignment section above). The expected number of sequences that can score better than or equal to *S* is given by

$$E = kmne^{-\lambda S} \qquad \text{Eq. (2)}$$

The expected number of HSPs that can score above or equal to a given score *S* increases with length of the sequences and exponentially decreases with the score (Lipman et al., 1984; Karlin and Altschul, 1990). If the *E-value* cutoff is set very low, then trivial hits may be reported. The default value for *E-value* is 0.001 (with BLOSUM62 matrix). The parameters, *k* and λ, are constants representing the scoring system. BLAST scores are reported as bit scores which is a normalized raw score obtained using the formula

$$S' = (\lambda S - \ln k)/\ln 2 \qquad \text{Eq. (3)}$$

The probability of finding atleast one HSP with a score equal to or greater than S is given by

$$P = 1 - e^{-E} \qquad \text{Eq. (4)}$$

A concise note on the alignment statistics and implementation of the BLAST search method is available at National Center for Biotechnology Information (http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html). The web-interface for BLAST program can be accessed at http://www.ncbi.nlm.nih.gov/BLAST/. There are many variants to BLAST that can effectively search DNA sequence or translated sequences (blastx or tblastn). As a rule of thumb, BLAST search should be used to collect closely related sequences that show high similarity. Higher order PAM (PAM250) or lower order BLOSUM (BLOSUM40) should be used if one likes to identify divergent sequences and in cases where an initial search produces no or very few hits. PSIBLAST is an efficient search tool that can detect weaker similarities using an iterative position specific scoring matrix. PSIBLAST search is similar to BLAST search during the first iteration. Once sequences with specified E-value cutoff are identified a position specific scoring matrix (PSSM) can be calculated. The PSSM is used for identifying other related sequences in the next iteration and if any sequence satisfies the statistical criteria, then it is included to modify the profile. A more elegant note on the implementation algorithm is described at the website http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-3.html.

3.4.2 FASTA

FASTA is a program that can rapidly identify shared regions in two sequences and score sequences in a database for homology (Pearson, 1998). The final output consists of a rank ordered list of sequences and alignment between sequences. Regions of high similarity among sequences are identified by segments with high frequency of conserved letters (ktup). A ktup value of 2 will look for pairs of conserved alphabets, a general value set for protein sequence search. In the initial step, a lookup table is created which is used for scoring a group of identities between two sequences. In the second step, a local alignment is constructed in the region of high density of identical alphabets using a similarity matrix. If there are nearby local alignments with scores greater than a preset cutoff, then initial regions are joined to produce an approximate alignment. An optimized score is calculated using Needleman-Wunch-Sellers algorithm. FASTA can be used for DNA and protein sequence search. A web interface to FASTA tool that can search sequence database is available at http://fasta.bioch.virginia.edu/. Major sequence databases like PDB (http://www.rcsb.org) and GENOME DB (http://fasta.genome.jp) also provide FASTA search facility.

3.5 Pair-Wise and Multiple-Sequence Alignment Tools

Given two sequences, pair-wise alignment produces an optimal alignment using a scoring matrix. On the other hand the main objective of multiple sequence alignment is to optimally align several related sequences such that evolutionarily constrained residues are aligned under the same column. Each column represents a residue position and can have gaps depending on insertion or deletion. BLAST or FASTA in principle can identify closely related sequences and do not provide information about conserved blocks of residues in a protein family. Creating a multiple sequence alignment of several protein sequences helps one to identify regions of significant conservation and in many cases to understand function. Multiple global sequence alignments are produced using heuristic methods based on progressive-alignment (ClustalW, TCOFFEE), simultaneous alignment of all sequences (MSA), or by using iterative strategies (Prrp).

3.5.1 ClustalW

ClustalW is a popular tool that uses several protein or DNA sequences as input to produce a multiple sequence alignment (Thompson et al., 1994). The algorithm implements progressive alignment in which sequences that are closely related are aligned to which more evolutionarily diverged sequences are added. The steps to produce a complete multiple alignment

can be broken into: (1) Identify closely related sequences among the input, (2) Order these sequences based on pair-wise similarity score, (3) Seed an alignment using those sequence pairs that have the highest similarity score, (4) Add sequences following a tree-order to create alignments in a progressive manner. The first step is similar to BLAST or FASTA where a fast approximate method is used to calculate pair-wise alignment scores for all sequences to create a distance matrix. Latest version of ClustalW has a choice of having a more rigorous approach of including a fully dynamic alignment and calculating scores at this step. A neighbor-joining method is used to construct a tree using distance scores (dissimilarity score). This tree is used for computing weights for sequences (this avoids bias due to higher number of identical or closely related sequences in the input) and also provides order of a sequence for progressive inclusion in the multiple alignment. The gap patterns introduced in the earlier stage of alignment are preserved during the addition of more divergent sequences subsequently. This is due to the fact that the alignment produced at the early stage using highly similar sequences has a high confidence. A web interface with help is available at EMBL website http://www.ebi.ac.uk/clustalw/. Stand alone versions of ClustalW program (and other variants of Clustal program like X-window based ClustalX) can be downloaded and installed (http://www.biolinux.org/). JalView is an applet that can be used to display multiple alignments, highlight physico-chemical properties and manually edit alignments. It can be accessed at http://www.ebi.ac.uk/jalview/.

3.5.2 TCoffee

TCoffee is a multiple alignment tool that has a progressive-alignment strategy to align multiple sequences based on a scoring matrix (Notredame et al., 2000). At every step of a progressive-alignment, information contained in the entire target sequences are considered during the alignment step, thus reducing bias suffered in the early steps of ClustalW. In TCoffee, a global and local pair-wise alignment of all sequences is created as a first step and weights for individual residue pairs are assigned based on the % identity of aligned residues. Using this library of primary alignments, an extended library of alignments is produced by considering all possible primary alignments among participating sequences that lead to produce a position specific library. A correct alignment is obtained using dynamic programming. A correct multiple alignment of sequence should represent a structural alignment among participating sequences. TCoffee was tested for accuracy of alignment using BaliBase (Thompson et al., 1999). Balibase is a collection of multiple sequence alignments constructed by structural superposition. TCoffee was found to produce alignments with 89.7% accuracy compared to ClustalW with 85.6%. A web interface for T-Coffee program can be accessed at http://www.igs.cnrs-mrs.fr/Tcoffee/

tcoffee_cgi/index.cgi. A downloadable version is available for Windows and Unix platforms from the same website. Both TCoffee and Clustal methods are included in the BioPerl package (http://www.bioperl.org).

3.5.3 JAligner

JAligner implements the Smith-Waterman algorithm for local pair-wise alignment with Gotoh's improvement for computing the weight of the gaps using the affine gap penalty model (Smith and Waterman, 1981; Gotoh, 1982). The implementation of JAligner improves the space complexity over the original algorithm from O (n^2) to O (n) by using only the last row (n) of working matrices instead of the whole two-dimensional matrices (n^2); however, the overall space complexity is O (n^2) for storing the trace back directions. As a standalone application, JAligner provides a friendly user interface (UI) – both graphical and command line. The graphical mode can be launched either online through a web browser or offline as a regular desktop application. As a Java library, JAligner provides *reusable* and *extendable* application programming interface (API). JAligner aligns sequences in FASTA or plain formats and generates alignments in FASTA and CLUSTAL formats. JAligner accepts *user-defined* scoring matrices; this is in addition to the set of already included scoring matrices (PAM and BLOSUM matrices). The source code and binaries are available for free at http://jaligner.sourceforge.net under the GNU Public License (GPL). JAligner is used as:

a. A simple *cross-platform* open source tool for biological local pair-wise sequence alignment with the affine gap penalty model.
b. A sample and educational implementation of the classic Smith-Waterman algorithm.

JAligner expects the following set of input parameters: two sequences in FASTA or plain formats, gap open, and gap extend penalties, Scoring matrix (e.g. PAM or BLOSUM), and output format (FASTA, CLUSTAL, and Pair). It generates the output that contains statistics about the generated alignment showing the percentage of the similarities, identities, gaps, and the score of the alignment and an alignment of the input sequences.

3.6 Sequence Motifs

Conserved patterns of nucleotides or residues that occur in a related set of sequences may possess specific functions. Such conserved patterns are defined motifs. In DNA sequences, transcription factors may bind to specific

nucleotide motifs. Identifying a sequence motif starts with collecting a set of sequences with common function. These sequences may belong to the same protein family or may be diverse with some common function, for example, Calcium binding or ATP binding. In order to identify motifs, one can first arrange all equivalent residue positions into a multiple alignment and look for conserved amino acid alphabet blocks. A regular expression can be used to express such conserved blocks of alphabets. Often it is impossible to identify a block of 100% conserved alphabets. This may be due to degeneracy among amino acids to substitute each other without disturbing the function. An expression or term widely used for description of protein sequence motifs follow regular expressions conventions used in UNIX. PROSITE is a widely used database of patterns or motifs with details of function and sequence that contain these patterns (Falquet et al., 2002; Gattiker et al., 2002; Sigrist et al., 2002). It currently contains (release 19.19) 1329 patterns. Commonly occurring patterns are expressed using regular expression terms with IUPAC single letter amino acid code. Positions where more than one amino acid can occur (or possibilities) are represented in a square bracket. Curly braces are used for excluding a set of amino acids at a particular position. The numbers inside a bracket represents minimum and maximum repetition of residues. The pattern (PA2_ASP, PS00119) for phospholipase A2 aspartic acid active site is given by: [LIVMA]-C-{LIVMFYWPCST}-C-D-{GS}-x(3)-{QS}-C. This pattern means sequence that contains specific aspartic acid protease active site contains any of the amino acids LIVMA in the first position with an absolutely conserved cysteine in the second position and fourth position. Sixth position can contain any amino acid other than small amino acids G or S. From seventh position, sequences that contain phospholipase A2 aspartic acid active site may have any 3 residue insertion represented by x(3). Thus, such expression patterns are used for enumeration of conserved amino acid alphabets among related sequences. Identifying such motifs is a non-trivial task, since shorter motifs may occur often and hence false positives are high. (In other words, the probability of random occurrence of a short sequence pattern is high.) Also, if the length of the pattern is extended, it may result in true negatives. PROSITE patterns can be downloded from ExPasy website http://www.expasy.ch/prosite/. ScanProsite is a search engine to scan PROSITE database for identifying patterns in a given sequence. Conservation of residues in a particular position can also be scored in terms of physical-chemical properties or by scoring the occurrence of amino acids with higher rate of substitution. Position-specific scoring matrices (PSSM) are probabilistic substitution frequencies or profiles calculated using a multiple alignment. Conserved motifs can also be represented by PSSM or profiles. Physico-chemical conservation is quantitatively expressed in motif identification tools like PCPMer and MASIA (Venkatarajan and Braun,

2001; Mathura et al., 2003). The PCP-motif implementation in MASIA (http://born.utmb.edu/masia/) converts the multiple alignments of protein alphabets into a numerical matrix using a property component vector. Highly conserved regions show relative entropy distribution of the property vectors different from a natural distribution with small standard deviation. Thus, MASIA/PCPMer can identify property motifs and provide a quantitative profile. Sequence alignment databases like BLOCKS, Pfam, PRINTS (Attwood et al., 2000) can provide input or the starting point for deriving profiles which can be subsequently used to identify sequences with related function or family. InterPro (http://www.ebi.ac.uk/interpro/) is an exhaustive collection of protein families, domains and functional sites of known proteins (Biswas et al., 2002; Kanapin et al., 2002; Mulder and Apweiler, 2002). Release 11.0 of InterPro contains 12294 entries covering 77.5% of UniProt. BLOCKS (http://blocks.fhcrc.org/blocks/) v 14.1 consists of 28,337 blocks based on InterPro entries with sequences from SWISSPROT and TrEMBL (Henikoff et al., 2000; Henikoff et al., 2000 b). Pfam (http://pfam.janelia.org) is a collection of annotation and alignment of protein sequences that belong to different families (Finn et al., 2006). It is manually annotated with details about structure/function of members and the seed alignments are done manually. This seed alignment is further used to build Hidden Markov Model profiles (using the tool HMMER http://hmmer.janelia.org) for subsequent searching in the translated protein database to identify homlogs. Pfam currently contains 8183 sequences (release 19) derived from Swissprot and TrEMBL database. Cross-references to PROSITE, PRINTS, and Pfam entries are available. InterPro provides cross-reference to PRINTS, Pfam, and PROSITE patterns. For example, the entry for Bcl-2, an anti-apoptotic protein, IPR000712 contains sequence signature cross-references: Pfam (PF00452), PROSITE pattern (PS01080, PS01258, PS01259), BLOCKS (IPB000712). There are three entries in PROSITE corresponding to three different conserved domains BH1, BH2, and BH3. The entire family of Bcl-2 sequences just has one entry in Pfam. Other useful features of InterPro entries are: Taxonomic coverage, structural links, etc. InterPro also cross-links to PRINTS database, for example, the entry for TUBBY protein in InterPro is IPR000007 with cross-reference to PRINTS (http://umber.sbs.man.ac.uk/dbbrowser/sprint/) database id PR01573. The PRINTS database is a compendium of protein motif fingerprints iteratively refined by sequence database scanning.

3.6.1 Generating Protein Sequence Motifs

In order to discover sequence motifs among a family of protein sequences, one can use tools like: GIBBS motif sampler (Thompson et al., 2003),

eMOTIF (Huang and Brutlag, 2001), PCPMer/MASIA (Mathura et al., 2003), Meta-MEME (Grundy et al., 1997; Grundy et al., 1997 b), and PRATT. GIBBS uses a Gibbs-motif sampling method that identifies conserved pattern among unaligned sequences using expectation maximization model. Highly conserved patterns occur at a higher probability in a set of sequence input. The method builds a list of conserved motif blocks using evolving data structures that track sequence segments and probability of such segments using bayesian statistics. A web-based GIBBS motif sampler program can be accessed at http://bayesweb.wadsworth.org/gibbs/gibbs.html. The web interface has three options: site-sampler, motif-sampler, and recursive sampler. Recursive sampler looks for multiple occurrences as a criterion for detecting motifs and motif sampler provides a list of motif hits. Gibbs motif sampler can take a list of upstream sequence for a list of co-regulated genes and identify transcriptional binding sites and other conserved regions. MASIA/PCPMer can be accessed at (http://born.utmb.edu/masia/) that takes CLUSTALW format multiple alignment as an input to identify different conserved motifs or blocks based on physical-chemical properties. The PCP macro implemented in MASIA program uses a five dimensional descriptor for amino acids to create a quantitative profile. Using relative entropy calculation using background frequency, the program identifies conserved regions in protein sequence. Meta-MEME is a motif discovery program that builds a Markov model for conserved regions in a set of training sequence (http://metameme.sdsc.edu). The HMM profile is built using expectation-maximization algorithm based on two component mixture model (MEME program). In mixture models, the probability density is represented as the sum weighted by fraction of total occurrence. The first component represents the probability of block of a sequence being a motif in the given list of sequence and the second component describes the background noise. MEME can be accessed at http://meme.nbcr.net/. MEME can be used as a general motif identification tool in both protein and DNA sequences. Another motif discovery tool is the eMOTIF program developed by the Brutlag group at the Stanford University. Using this tool one can create motifs from input of multiple-aligned sequences, search for motifs in a given sequence based on the motif library created using BLOCKS and PRINTS database or search a regular pattern in the protein sequence database. Pratt implements a branch-and-bound heuristics to search for patterns in a set of sequences (Jonassen et al., 1995). The output is a set of motifs with corresponding regular expressions. Pratt can be accessed at http://www.ebi.ac.uk/pratt/.

References

Abagyan, R.A. and Batalov, S. (1997) Do aligned sequences share the same fold? *J Mol Biol* **273**(1), 355–68.

Attwood, T.K., Croning, M.D., et al. (2000) PRINTS-S: the database formerly known as PRINTS. *Nucleic Acids Res* **28**(1), 225–7.

Biswas, M., O'Rourke, J.F., et al. (2002) Applications of InterPro in protein annotation and genome analysis. *Brief Bioinform* **3**(3), 285–95.

Dayhoff, M.O. and Schwartz, R.M. (1978). *A model of evolutionary change in proteins*. Washington DC, National Biomedical Research Foundation.

Falquet, L., Pagni, M., et al. (2002) The PROSITE database, its status in 2002. *Nucleic Acids Res* **30**(1), 235–8.

Finn, R.D., Mistry, J., et al. (2006) Pfam: clans, web tools and services. *Nucleic Acids Res* **34**(Database issue), D247–51.

Gattiker, A., Gasteiger, E., et al. (2002) ScanProsite: a reference implementation of a PROSITE scanning tool. *Appl Bioinformatics* **1**(2), 107–8.

Gonnet, G.H., Cohen, M.A., et al. (1992) Exhaustive matching of the entire protein sequence database. *Science* **256**(5062), 1443–5.

Gotoh, O. (1982) An improved algorithm for matching biological sequences. *J Mol Biol* **162**(3), 705–8.

Grundy, W.N., Bailey, T.L., et al. (1997) Hidden Markov model analysis of motifs in steroid dehydrogenases and their homologs. *Biochem Biophys Res Commun* **231**(3), 760–6.

Grundy, W.N., Bailey, T.L., et al. (1997 b) Meta-MEME: motif-based hidden Markov models of protein families. *Comput Appl Biosci* **13**(4), 397–406.

Henikoff, J.G., Greene, E.A., et al. (2000) Increased coverage of protein families with the blocks database servers. *Nucleic Acids Res* **28**(1), 228–30.

Henikoff, J.G., Pietrokovski, S., et al. (2000 b) Blocks-based methods for detecting protein homology. *Electrophoresis* **21**(9), 1700–6.

Henikoff, S. and Henikoff, J.G. (1992) Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci U S A* **89**(22), 10915–9.

Huang, J.Y. and Brutlag, D.L. (2001) The EMOTIF database. *Nucleic Acids Res* **29**(1), 202–4.

Johnson, M.S. and Overington, J.P. (1993) A structural basis for sequence comparisons. An evaluation of scoring methodologies. *J Mol Biol* **233**(4), 716–38.

Jonassen, I., Collins, J.F., et al. (1995) Finding flexible patterns in unaligned protein sequences. *Protein Sci* **4**(8), 1587–95.

Kanapin, A., Apweiler, R., et al. (2002) Interactive InterPro-based comparisons of proteins in whole genomes. *Bioinformatics* **18**(2), 374–5.

Karlin, S. and Altschul, S.F. (1990) Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc Natl Acad Sci USA* **87**(6), 2264–8.

Lipman, D.J., Wilbur, W.J., et al. (1984) On the statistical significance of nucleic acid similarities. *Nucleic Acids Res* **12**(1 Pt 1), 215–26.

Mathura, V.S., Schein, C.H., et al. (2003) Identifying property based sequence motifs in protein families and superfamilies: application to DNase-1 related endonucleases. *Bioinformatics* **19**(11), 1381–90.

Mulder, N.J. and Apweiler, R. (2002) Tools and resources for identifying protein families, domains and motifs. *Genome Biol* **3**(1), REVIEWS2001.

Naor, D., Fischer, D., et al. (1996) Amino acid pair interchanges at spatially conserved locations. *J Mol Biol* **256**(5), 924–38.

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**(3), 443–53.

Notredame, C., Higgins, D.G., et al. (2000) T-Coffee: A novel method for fast and accurate multiple sequence alignment. *J Mol Biol* **302**(1), 205–17.

Pearson, W.R. (1998) Empirical statistical estimates for sequence similarity searches. *J Mol Biol* **276**(1), 71–84.

Prlic, A., Domingues, F.S., et al. (2000) Structure-derived substitution matrices for alignment of distantly related sequences. *Protein Eng* **13**(8), 545–50.

Rost, B. (1999) Twilight zone of protein sequence alignments. *Protein Eng* **12**(2), 85–94.

Sigrist, C.J., Cerutti, L., et al. (2002) PROSITE: a documented database using patterns and profiles as motif descriptors. *Brief Bioinform* **3**(3), 265–74.

Smith, T.F. and Waterman, M.S. (1981) Identification of common molecular subsequences. *J Mol Biol* **147**(1), 195–7.

Thompson, J.D., Higgins, D.G., et al. (1994) CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res* **22**(22), 4673–80.

Thompson, J.D., Plewniak, F., et al. (1999) BAliBASE: a benchmark alignment database for the evaluation of multiple alignment programs. *Bioinformatics* **15**(1), 87–8.

Thompson, W., Rouchka, E.C., et al. (2003) Gibbs Recursive Sampler: finding transcription factor binding sites. *Nucleic Acids Res* **31**(13), 3580–5.

Venkatarajan, M.S. and Braun, W. (2001) New quantitative descriptors of amino acids based on multidimensional scaling of a large number of physical-chemical properties. *J Mol Model* **7**, 445–53.

Wilson, C.A., Kreychman, J., et al. (2000) Assessing annotation transfer for genomics: quantifying the relations between protein sequence, structure and function through traditional and probabilistic scores. *J Mol Biol* **297**(1), 233–49.

Chapter 6

Protein Structure Prediction

Hongyi Zhou[1], Yaoqi Zhou[1], and Venkatarajan S. Mathura[2]

[1]*Howard Hughes Medical Institute Center for Single Molecule Biophysics, Department of Physiology & Biophysics, State University of New York at Buffalo, 124 Sherman Hall, Buffalo, NY 14214*
[2]*Roskamp Institute, 2040 Whitfield Avenue, Sarasota, FL 34243*

Abstract: Genomic projects have provided large number of sequence information. In order to obtain tangible benefit of this information, structural and functional annotation of the sequences is a must. Understanding the structural basis for protein function enables rational drug design and novel interventions for various diseases. This chapter describes protein-modeling packages and services that integrate various tools to facilitate rapid large-scale modeling of proteins.

Key words: Comparative modeling, Homology, CASP, MPACK, SP[3], LiveBench, MODELLER

1. Introduction

With the completion of human genome projects (Cantor, 1990; Watson, 1990; Liang et al., 2000; Waterston et al., 2002), protein tertiary structure prediction from primary sequence is gaining tremendous importance (Finkel, 1997; Fischer and Eisenberg, 1997; Andrade et al., 1998; Sali, A 1998; Burley et al., 1999; Salamov et al., 1999). Experimental 3D structure determination methods such as NMR spectroscopy and X-ray crystallography produce currently ~250 structures per month compared to roughly half a million sequences submitted per month in the NCBI database. Tertiary structure prediction on a genomic scale is needed to understand complex biochemical functions of many proteins from a structural perspective (Andrade et al., 1997; Dandekar and Argos, 1997; Fetrow and Skolnick, 1998; Gerloff et al., 1998; Andrade et al., 1999; Koehl and Levitt, 1999; Rison et al., 2000; Norin and Sundstrom, 2002). Understanding the structural basis for protein function enables rapid progress in systems

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_6,

biology that aims at identifying functional networks of proteins at a large scale from genomics and proteomics projects (Koehl and Levitt, 1999). Rational drug design heavily relies on the structural knowledge of a protein (Hol, 1989; Verlinde et al., 1994; Gait and Karn, 1995).

Given a protein sequence, one can apply secondary structure prediction methods to assign secondary structural elements like helices, strands and coils. A higher-level structure prediction is to model the three-dimensional structure of the protein or fold. As described in the introductory chapter secondary structural elements are formed due to periodic weak hydrogen-bonds between donor and acceptor groups in protein. Such bonds are formed in a specific pattern only when specific residues are present. Thus, secondary structure of a protein can be predicted based on amino acid propensity to form helices or sheets.

Tertiary structure of proteins can be modeled using comparative modeling or *ab initio* methods (Forster, 2002). In comparative modeling, a related template is selected that provides a geometrical framework for modeling the unknown sequence. A template structure can be selected by searching (homologous sequence search) a structural database for sequences that share identical residues (at least 30%) with the target sequence.

The coverage of sequence space by proteins with the known 3D structure is currently not high enough to infer potential 3D fold from sequence similarity searches for all genomic sequences (Rost, 1997; Jaroszewski et al., 2002). Difficult targets that do not show obvious homology for sequences with structure can be modeled by selecting a structural template to which the target sequence can fit using threading algorithms or fold-recognition methods (Rost, 1997). Threading techniques employ structural profiles to infer the degree to which a query sequence would fit into a known fold. Threading can identify suitable templates even when the sequence alignment identity is as low as 15% (Twilight region) (Rost, 1997). Homology search or threading identify suitable template that can be used by comparative modeling tools to model the unknown protein. Structural genomic projects aim to increase the coverage of sequence space by determining new 3D folds (Gerloff et al., 1998; Burley et al., 1999; Koehl and Levitt, 1999; Lo Conte et al., 2002; Schonbrun et al., 2002), which will increase the reliability of templates identified by homology sequence search methods and threading techniques. In those cases where homology search or fold-recognition methods fail to provide a reliable template and where the target sequence is short (less than 150 residues), a pure physics based approach called *ab initio* method or a combination of physics and knowledge based method called new fold detection technique, can be applied (Orengo et al., 1999; Bonneau et al., 2001; Bonneau et al., 2002; Kihara et al., 2002; Srinivasan and Rose, 2002).

2. Secondary Structure Prediction

Secondary structure prediction methods rely on the frequency of observed amino acids in the secondary structural elements of proteins for which tertiary structure has been experimentally determined. Periodic occurrence of amino acids with specific physicochemical properties has been observed in different secondary structural elements. For example, in amphipathic helix (one with both hydrophobic and hydrophilic surface) periodic occurrence of hydrophobic residues occur at every 3–4 residues due to its packing nature (hydrophobic surface packs together to form a core while hydrophilic surface contacts with the external solvent). In the case of sheets, the packing of two beta-sheet forms a pleated structure that requires alternating hydrophobic and hydrophilic residues or complete stretch of hydrophobic residues. Helix breakers, like proline do not occur in the middle due to its restriction in forming hydrogen-bond. One of the earliest methods to predict secondary structure is based on Lim's stereochemical prediction rules that use the observed periodicity of hydrophobic residues at every 3–4 residue positions for helix and an alternating hydrophobic residue stretch for beta-sheet. Chou and Fasman used known tertiary structures of proteins to calculate residue propensity in helix, betasheet and coil. They used propensity values to assign secondary structure to residue stretches. GOR method uses a conditional probability to assign secondary structure using a window-based approach. In this method, residue frequencies of amino acids in a 17 residue window are used to calculate conditional probability of the mid residue (8th) for three different secondary structures. Advanced secondary structure prediction methods use neural networks to predict or assign secondary structural elements for each amino acid in a given sequence. For example, NNPredict (http://alexander.ucsf.edu /~nomi/nnpredict.html) based on a two layer feed forward neural network assign either "H" (for helix forming residues) or "E" (extended or sheet forming residues) to an input sequence. If a closely related structure (homologous) is available then the secondary structure of conserved or highly similar residues can be assigned based on structural template. PHD is a neural network based method that uses a multiple alignment of an input sequence to calculate probability of secondary structure. Among homologous proteins, secondary structure will be conserved; hence, distribution of amino acids at a residue position can be useful to assign probability of secondary structural state. PHD server uses BLAST to search for homologous sequences and creates a multiple alignment before feeding the distribution of amino acids at every position into a neural network. JPred (http://www.combio.dundee.ac.uk/jpred_v2/) is meta-service which assigns residue secondary structure based on the consensus of several independent

methods. It includes NNSSP (based on nearest neighbor environment profile in multiple alignments), PHD, linear discrimination based DSC, hydrogen-bonding propensity based PREDATOR, and combination of statistical weight based single sequence method MULPRED. It also includes a neural network based method, JNet that uses PSIBLAST and HMM profiles for a given input sequence. Several secondary structure prediction servers are listed in Table 6.1. A higher confidence can be obtained in secondary structure prediction if different methods are used before concluding assignments. Consensus based approaches like JPred-consensus and profile based JNet are found to have higher performance with the prediction accuracy (given as Q3 score, defined as the percentage of correctly assigned secondary structure states of helix, betasheet, or coil) greater than 75%. PSIPRED uses a feed forward neural network to predict secondary structure. There are four neural networks used in this method and an average of these four network are used for assignment.

Table 6.1 Secondary-structure prediction servers

Server	URL
JPRED	www.combio.dundee.ac.uk/jpred_v2/
NNSSP	www.softberry.com/berry.phtml?topic=nnssp&group=programs&subgroup=propt
PHD	cubic.bioc.columbia.edu/predictprotein/
NNPredict	alexander.ucsf.edu/~nomi/nnpredict.html
PSIPRED	http://bioinf.cs.ucl.ac.uk/psipred/

3. Comparative Modeling

Comparative modeling involves identifying a structural parent to an unknown sequence and the use of geometric constraints derived from the known structure to model the unknown sequence. A structural parent can be identified either by inferring homology between the target sequence and the structural parent or by measuring the fitness of a given sequence to all known folds. The term “homology” implies a common evolutionary origin among proteins. In an evolutionary tree, daughter proteins are evolved by gene duplication, random mutation, and selection, which create a net drift from parent or ancestral proteins (Aszodi and Taylor, 1996; Abagyan et al., 1997; Yu et al., 1998; Kitson et al., 2002; Reddy et al., 2002; Thornton, 2002). Daughter proteins, to a large extent share identical sequences with little residue variation and a common function with their immediate parent. Selection process allows residue changes only at positions that are structurally and functionally less important. Many studies have shown that amino acids can be interchanged with each other without compromising

changes in the structure (Swanson, 1984) (Henikoff and Henikoff, 1992). Such exchanges are possible among amino acids that have similar physical-chemical properties. As a result, exchanges within similar groups will be tolerated. Thus, protein structures are more resilient to changes in amino acids during evolution. The degree of substitution at a particular residue position depends on the functional role and the environmental location of the residue in the folded protein. Similar structure or geometry of a set of unrelated sequences might also have evolved independently due to convergent evolution. Threading or fold-recognition methods apply structural information in measuring a fit between a given fold and a sequence. Such methods can detect similarity even when the sequence identity is very low, that is, less than 30%. Experimental structure determination procedures like X-ray and NMR have accumulated information in the form of a databank called PDB (protein databank). Comparative modeling uses either homologous sequence search or fold-recognition methods to identify a suitable fold or structural parent.

3.1 Steps Involved in Comparative Modeling

Various steps involved in homology modeling of a protein are

1. Identify suitable template for modeling:
   - Infer homology between target sequence and template by comparing their sequences using a similarity matrix or use fold-recognition algorithms. Use loop library to identify templates for modeling loop regions.
   - Align two sequences such that structurally equivalent regions match.
2. Construct a 3D model for the sequence based on the template:
   - Extract geometrical constraints from parent.
   - Apply these geometrical constraints to the target.
   - Energy refine the crude model, use side-chain library to optimize side chain locations

3.2 Homologous Sequence Search Using Sequence Comparison Tools

Protein sequences have been deposited in databases like SWISSPROT-TREMBL (Bairoch and Apweiler, 1999), PIR (Barker et al., 1999), GenBank (Benson et al., 2002), and EMBL (Emmert et al., 1994). The oldest of search comparison tools is FASTA, which matches words or k-tuples between two sequences and obtains a similarity score based on the number of matches. BLASTP (Altschul et al., 1990) is another sequence search tool

that uses similarity matrix like BLOSUM (Henikoff and Henikoff, 1992) or PAM (Dayhoff and Schwartz, 1978) to compare a query sequence with subject sequences in sequence databases. BLAST is a heuristic based program that tries to find maximum segment pairs between the query sequence and subject sequences using a local alignment made with modified Smith-Waterman algorithm. For every pair-wise comparison BLAST produces a bit-score (by adding log-odd bit for every matching pairs during comparison) and an E-value (Expected value evaluates the probability that an observed similarity score occurred randomly). PSI-BLAST (Altschul et al., 1997) is a modified version of BLAST in which a position-specific scoring matrix is created and iteratively updated as new sequences above a certain E-value threshold are found. PSI-BLAST is more sensitive in identifying distantly related homologues compared to BLAST or FASTA (Schaffer et al., 2001). A BLAST/PSI-BLAST search may result in one or more sequence hits that have their structure previously determined experimentally. More advanced sequence based methods include profile-HMM (Martelli et al., 2002) that uses probabilistic models to model observed amino acid distribution at a particular position in related sequences and use it for identifying remote homologs. Packages like meta-MEME (Grundy et al., 1997) or PROBE (Neuwald et al., 1997) can be used for creating profile-HMMs and can be applied for searching sequences. If sequence based search methods fails to locate a suitable template, more effective methods like fold-recognition for identifying remote homologs can be used.

3.3 Identifying Remote Templates Using Fold-Recognition Methods

Protein sequence adopts a structure from a limited repertoire of folds. Sequence comparison methods do not include structural information. Fold-recognition methods try to match a given sequence with encoded information about a structure or, more specifically, a fold by determining the goodness of fit to a particular fold type. Most fold-recognition methods convert structural or environmental information of a particular fold into a profile and score a given sequence against the profile. 1D-3D profile method attempts to describe a 3D fold into 1D string that describes the environmental state (Bowie et al., 1991; Gribskov and Veretnik, 1996). The states include secondary structures (alpha, beta, or coil), solvent accessibility (buried, partial, or exposed), and residue charges (polar or apolar). Each position in the fold can fall into any one of 18 structural states. A scoring method includes a fit-function to measure goodness of fit of a given sequence against *a priori* distribution of amino acids in 18 states studied

using known folds. Fold-recognition servers like 123D+, BIOINBGU (Fischer, 2000), 3D-PSSM (Kelley et al., 2000) are profile-based methods. A threading method treats a fold as sets of interactions and any sequence that fits into a given fold must satisfy these interactions favorably. Pair wise interactions among amino acids are converted into contact potentials or interaction energies that describe suitability of interactions among all possible pairs of amino acids. GenThreader (Jones et al., 1999), and PROSPECT (Xu et al., 2001) servers use threading methods and sophisticated neural network based algorithms for selection of templates. The output of fold-recognition servers are templates that are ranked based on their scores and corresponding alignments.

3.4 Selection of the Alignment

If a target sequence shows high homology to a structural parent with a unique template the modeling procedure becomes straightforward. Partial regions or domains of a target sequence may match with different templates. Fold-recognition methods often return more than one possible template and corresponding alignment for the target sequence. A selection procedure that will identify the best template from multiple possible templates and an optimal alignment is necessary to model difficult targets. Evaluating alignment quality in the conserved regions of target sequence can be applied to select among multiple possible templates and alignments. A weighing scheme that uses simple voting to identify best template is available in a meta-server, which evaluates different alignments based on potential mean field scores computed using TITO (Labesse and Mornon, 1998). PCONS is an advanced neural network based consensus predictor that selects best template using output from different fold-recognition methods (Lundstrom et al., 2001). Both PCONS and TITO follow a jury scheme without collecting more information about the target sequence or refining alignment. In both the methods the information of sequence conservation in the target sequence family is not taken into account.

3.5 Construction of 3D Models Using Modeling Programs

Once suitable template(s) is identified and alignment is derived, the next step in modeling will be to obtain a 3D model. In general, most methods extract geometrical constraints like distance and dihedral angles using the template. The dihedral angles are defined by phi, psi, and omega angles for the backbone. Distance constraints are defined between conserved atoms in the target and the template. MODELLER uses an automated approach to comparative modeling by satisfying spatial restraints (Sali, 1998). The

objective function in MODELLER includes CHARMM (Brooks et al., 1983) force field terms and spatial restraints that are optimized in cartesian space. COMPOSER (Johnson et al., 1994) automatically constructs protein models using constraints derived from structurally conserved regions (SCRs). MPACK (Modeling Package) is an integrated protein-modeling suite designed to handle modeling of proteins effectively.

3.6 Protein Modeling Package – MPACK

Modeling Package (MPACK) currently handles both comparative and *ab initio* modeling procedures. The objective of this suite is to systematically bring different steps (or programs) under one roof in order to facilitate rapid model generation with minimal user effort and to create a biological data-flow pipeline for large scale modeling of protein sequences from genomic projects. This suite was created with the geometry extraction program EXDIS (Soman et al., 2000) that extracts distance and dihedral constraints, specified using one or more structural parents. Inside/outside, secondary structure predictions from MASIA (Zhu et al., 2000) and other knowledge based topological constraints can be used directly to convert into suitable distance constraints using TRANSLATE (Soman et al., 2000). Geometric constraints generated are directed as input into the distance geometry program DIAMOD (Mumenthaler and Braun, 1995) that optimally calculates structures by either starting from a random conformation or from approximate models produced by EXDIS. The program also has an option for generating models with disulphide bonds and appropriate switches for self-correcting distance geometry procedures when approximate constraints (like constraints from MASIA predictions) are used. The package is robust and efficient in handling situations wherein a user intends to model a target based on multiple template fragments. MPACK is a user-friendly tool that increases modeling efficiency by automating most data exchanges among different software components. Additional modules like template search and alignment procedures can be easily added and the entire modeling procedure can be completely automated.

3.7 SP^3 – A Web-Based Structure-Prediction Tool Using Known Protein Structures as Templates

3.7.1 Introduction

Template-based modeling of protein structures (comparative modeling and fold recognition) attempts to recognize structural similarity of two proteins with (comparative modeling) or without (fold recognition) significant

sequence identity. One way to detect structural similarity is to identify remote sequence homology via sequence comparison. Advances have been made from pair-wise to multiple sequence comparison, from sequence-to-sequence, sequence-to-profile to profile-to-profile comparison. Another way to detect structural similarity is via sequence-to-structure threading. More recent work attempts to optimally combine the sequence and structure information for a more accurate/sensitive fold recognition. For a recent review, see Godzik (2003) (Godzik, 2003). SP^3 (Zhou and Zhou, 2005) is a profile-based method that provides sequence to structure alignment based on the sequence as well as the structure information of templates.

3.7.2 Algorithm

The algorithm of SP^3 for a pair-wise alignment between a query sequence and a structural template is shown in Figure 6.1. The details are as follows. First, the program PSIBLAST (Altschul et al., 1997) is used to search homologous sequences of a query sequence from the NCBI non-redundant (NR) database (ftp://ftp.ncbi.nih.gov/blast/db/FASTA/nr.gz). As in PSIPRED (Jones, 1999), the NR database was filtered to remove low-complexity regions, transmembrane regions, and coiled-coil segments before being searched by PSIBLAST. This homolog search is conducted with an E-value cutoff of 0.001 and completed after three iterations. Homologous sequences found by PSIBLAST are then filtered by keeping only those sequences that have less than 98% identity with the query sequence and an E-value of less than 0.001. Filtered homologs are used to produce the

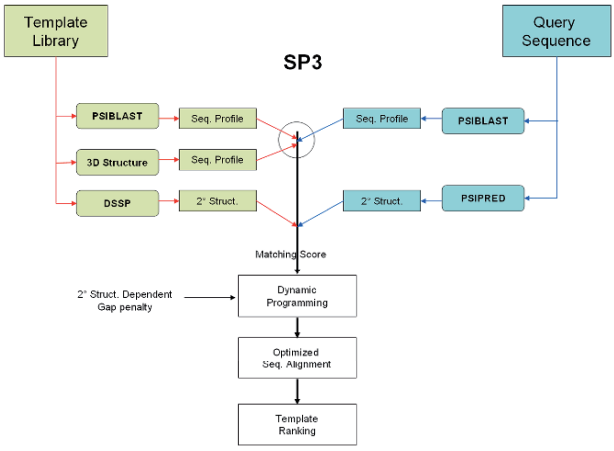


Figure 6.1 The flow chart of SP^3 that uses structural information of templates without threading. Two sequence profiles (sequence-based and structure-based) generated per template were used to match the sequence-based profile generated for the query sequence.

sequence profile that characterizes evolutionary-derived probability of a residue type at a given query sequence position. Similarly, the sequence profiles of template sequences are also obtained. Second, PSIPRED (Jones, 1999) is used to predict the secondary structure of a query sequence. The secondary structures of templates were obtained by H-bonds (DSSP-like) criteria (Kabsch and Sander, 1983). Three states (helix, strand, and coil) are used for all secondary structures.

Third, a structure-based sequence profile is generated using template structure. Each template structure is divided into nine-residue fragments in the SP^3 method. Each fragment structure is compared to the fragment structures in a structural fragment library. The structures in the fragment library will be ranked according to similarity to the structure and environment (residue depth) of the template fragment. The sequences of top ranked fragments (based on structural and residue-depth alignment) are used to calculate a structure-based sequence profile for the template structure. Fourth, two sequences (query and template) are aligned with a total matching score characterized by scoring the fitness between two sequence profiles generated from multiple sequence alignment program PSI-BLAST, the fitness between the query sequence profile and structure-derived sequence profile based on the template structure, and the fitness between predicted secondary structure of the query sequence and the actual secondary structure of the template. The highly efficient local–local dynamic programming method is used to optimize the total raw score for the best alignment between the query and template sequences. In SP^3, we also used a gap penalty that depends on secondary structures (Zhou and Zhou, 2005). Finally, the matching scores between the query sequence and all templates stored in the template library are obtained. An empirical method based on raw score, normalized scores, and their Z-scores (a measure of relative score difference between one template from the rest of templates) is used for ranking the templates. The top ranked templates are used to build the structure model of the query sequence by using MODELLER (Marti-Renom et al., 2000). SP^3 is one of the most accurate and sensitive servers in structure prediction based on testing on various benchmarks (Zhou and Zhou, 2005), including LiveBench (Bujnicki, 2001). SP^3 is also among the best servers for comparative modeling targets and among the top single-method servers for all targets in the CASP 6 meeting that assessed 49 automatic web-servers (Zhou and Zhou, 2005).

3.7.3 Input and Output

The input for SP^3 is the query sequence in the FASTA format and the number of structure models to be built based on top ranked templates. The

output (in html format) contains the links to PSIBLAST output for sequence profile, PSIPRED output for the secondary structure prediction, the top 10 sequence-to-structure alignments and the structure models (in PDB format) built based on the alignments. The significance of the sequence-to-structure alignment is indicated by the Z-score for each alignment. An alignment is significant if Z-score > 6.3. The threshold was obtained from LiveBench 8 (Bujnicki et al., 2001) for predicted models with MaxSub score (Siew et al., 2000) > 0.01 when compared to their respective native structures. The output is now reported in a table format for easy understanding. Sample input and output with detailed line-to-line explanations are available online. The servers and executables are available for academic users at the "services" and "downloads" sections of http://sparks.informatics.iupui.edu/.

3.8 Modeling Servers

Web-based services for structure prediction and modeling are available. ESyPred3D (http://www.fundp.ac.be/urbm/bioinfo/esypred/) (Lambert et al., 2002) uses neural network to improve sequence alignment from several multiple alignment methods and the final model is obtained using MODELLER. The FFAS03 (Jaroszewski et al., 2005) is a fold-recognition server that uses profile-profile alignment of target sequence with sequences in a non-redundant protein sequence database (http://ffas.ljcrf.edu/ffas-cgi/cgi/ffas.pl). This profile is compared with sequences in PDB, Pfam, SCOP, etc. Initial profile is generated using PSI-BLAST. Hidden Markov based sequence alignment and modeling tool SAM-T02 produces template hits for query input (http://www.soe.ucsc.edu/compbio/HMM-apps/T02-query.html). It consists of a library of Markov models for known protein sequences and family. A query is searched across the library to identify suitable hits. Another related server is SUPERFAMILY, which uses HMM library of protein sequence family for which structure is known. SUPERFAMILY is very helpful in annotation of genome sequences and to identify homologous templates even if the identity is low (http://supfam.org/SUPERFAMILY/) (Gough et al., 2001; Gough and Chothia, 2002; Madera et al., 2004). Fold-recognition server INUB (http://inub.cse.buffalo.edu/) is based on sequence derived properties and profiles (http://inbu.cse.buffalo.edu/) (Fischer, 2000; Fischer, 2003; Sasson and Fischer, 2003). The fold-recognition server FUGUE (http://tardis.nibio.go.jp/fugue/) is based on profile library search against HOMSTRAD, a database for structural alignment of homologous sequences (de Bakker et al., 2001; Shi et al., 2001). PHYRE is a protein homology/analogy search engine, which is an improvement of the 3D-PSSM fold-recognition method (http://www.sbg.bio.ic.ac.uk/~phyre/) (Kelley et al.,

2000). mGenThreader is a fold-recognition server that uses the sequence profiles of a protein family to assign the fold (http://bioinf.cs.ucl.ac.uk/psipred/) (Jones et al., 1999).

3.9 Critical Assessment of Structure Prediction

Dr. John Moult of University of Maryland established CASP experiments in 1992 with the aim of identifying the state of the art of protein structure prediction methods, disorder region prediction, and the progress made in function assignment (Moult et al., 1995). It is a biennial experiment in which scientific groups working in the field of structure prediction participate. CASP provides an objective way of testing the performance of modeling methods and tools (Moult et al., 1999). It is a double blind experiment where both the organizers of the competition and the participants do not know the 3D structure of the protein sequence that they provide to the modeling community until the prediction period. In CASP4, 160 research groups participated and submitted 11,136 models for 43 targets. In CASP5, 188 groups participated and submitted 28,728 models for 67 targets. The recent CASP6 had 201 human experts and 65 prediction servers that predicted models for 64 targets. Nearly 41,283 models were submitted under different categories. Please visit CASP homepage at http://predictioncenter.org for more information.

3.10 Objective Testing of Modeling Tools in CASP

The comparative modeling category consists of targets for which a template can be identified by BLAST or PSI-BLAST search. The evaluation of the models includes structural alignment between the model and the experimental structure to obtain the number of residues that fit well (AL0 score). A Global Distance Test (GDT) is used to measure number of residues within a distance-cutoff after performing structural alignment of the model and the experimental structure. In CASP6, a combined score of GDT_TS (average percentage of residues that were predicted correctly within 1, 2, 4, and 8 Å) and AL0 (alignment registered correctly) were used to identify the best group. The best modeling group was Krysztof Ginalski at University of Texas. This group submitted 150 models for 64 3D targets using a consensus fold-recognition method called 3D-Jury and a profile-alignment method called Meta-BASIC. In the fold-recognition category, the targets evaluated were those for which sequence search using PSI-BLAST does not find a hit but a structural template could be identified based on structure-structure alignment in the entire PDB. Homologous fold-recognition (FA/H) category consisted of targets for which homologous templates were present. Targets

that did not have any homologous templates but have atleast a template with similar fold, were included in the analogous fold-recognition (FR/A) category. The best group in the FR/H category in CASP6 was Krysztof Ginalski and Bakers group ranked first in FR/A category. New fold category includes target with no homologous templates or similar fold in the PDB. Both FR/A targets and the six additional targets for which there were no structural templates or folds were scored under this category. Bakers group ranked the best followed by Bujnicki and Ginalski group.

References

Abagyan, R., Batalov, S., et al. (1997) Homology modeling with internal coordinate mechanics: deformation zone mapping and improvements of models via conformational search. *Proteins* Suppl 1, 29–37.

Altschul, S.F., Gish, W., et al. (1990) Basic local alignment search tool. *J Mol Biol* 215(3), 403–10.

Altschul, S.F., Madden, T.L., et al. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res* 25(17), 3389–402.

Andrade, M.A., Brown, N.P., et al. (1999) Automated genome sequence analysis and annotation. *Bioinformatics* 15(5), 391–412.

Andrade, M., Casari, G., et al. (1997) Sequence analysis of the Methanococcus jannaschii genome and the prediction of protein function. *Comput Appl Biosci* 13(4), 481–3.

Andrade, M.A., Sander, C., et al. (1998) Updated catalogue of homologues to human disease-related proteins in the yeast genome. *FEBS Lett* 426(1), 7–16.

Aszodi, A. and Taylor, W.R. (1996) Homology modelling by distance geometry. *Fold Des* 1(5), 325–34.

Bairoch, A. and Apweiler, R. (1999) The SWISS-PROT protein sequence data bank and its supplement TrEMBL in 1999. *Nucleic Acids Res* 27(1), 49–54.

Barker, W.C., Garavelli, J.S., et al. (1999) The PIR-International Protein Sequence Database. *Nucleic Acids Res* 27(1), 39–43.

Benson, D.A., Karsch-Mizrachi, I., et al. (2002) GenBank. *Nucleic Acids Res* 30(1), 17–20.

Bonneau, R., Ruczinski, I., et al. (2002) Contact order and ab initio protein structure prediction. *Protein Sci* 11(8), 1937–44.

Bonneau, R., Tsai, J., et al. (2001) Rosetta in CASP4: progress in ab initio protein structure prediction. *Proteins* Suppl 5, 119–26.

Bowie, J.U., Luthy, R., et al. (1991) A method to identify protein sequences that fold into a known three-dimensional structure. *Science* 253(5016), 164–70.

Brooks, B., Bruccoleri, R., et al. (1983) CHARMM: A Program for Macromolecular Energy, Minimization, and Molecular Dynamics Calculations. *J. Comp. Chem* 4, 187–217.

Bujnicki, J.M. (2001) Livebench-1: Large-scale automated evaluation of protein structure prediction servers. *Protein Sci* 10(352–361).

Burley, S.K., Almo, S.C., et al. (1999) Structural genomics: beyond the human genome project. *Nat Genet* 23(2), 151–7.

Cantor, C.R. (1990) Orchestrating the Human Genome Project. *Science* 248(4951), 49–51.

Dandekar, T. and Argos, P. (1997) Applying experimental data to protein fold prediction with the genetic algorithm. *Protein Eng* 10(8), 877–93.

Dayhoff, M.O. and Schwartz, R.M. (1978). *A model of evolutionary change in proteins*. Washington DC, National Biomedical Research Foundation.

de Bakker, P.I., Bateman, A., et al. (2001) HOMSTRAD: adding sequence information to structure-based alignments of homologous protein families. *Bioinformatics* 17(8), 748–9.

Emmert, D.B., Stoehr, P.J., et al. (1994) The European Bioinformatics Institute (EBI) databases. *Nucleic Acids Res* 22(17), 3445–9.

Fetrow, J.S. and Skolnick, J. (1998) Method for prediction of protein function from sequence using the sequence-to-structure-to-function paradigm with application to glutaredoxins/thioredoxins and T1 ribonucleases. *J Mol Biol* 281(5), 949–68.

Finkel, E. (1997) The Post-Genome Era: Medical Promise with Problems. *The Lancet* 349, 1228.

Fischer, D. (2000) Hybrid fold recognition: combining sequence derived properties with evolutionary information. *Pac Symp Biocomput*, 119–30.

Fischer, D. (2003) 3D-SHOTGUN: a novel, cooperative, fold-recognition meta-predictor. *Proteins* 51(3), 434–41.

Fischer, D. and Eisenberg, D. (1997) Assigning folds to the proteins encoded by the genome of Mycoplasma genitalium. *Proc Natl Acad Sci U S A* 94(22), 11929–34.

Forster, M.J. (2002) Molecular modelling in structural biology. *Micron* 33(4), 365–84.

Gait, M.J. and Karn, J. (1995) Progress in anti-HIV structure-based drug design. *Trends Biotechnol* 13(10), 430–8.

Gerloff, D.L., Joachimiak, M., et al. (1998) Structure prediction in a post-genomic environment: a secondary and tertiary structural model for the initiation factor 5A family. *Biochem Biophys Res Commun* 251(1), 173–81.

Godzik, A. (2003) Fold recognition methods. *Methods Biochem Anal* 44, 525–46.

Gough, J. and Chothia, C. (2002) SUPERFAMILY: HMMs representing all proteins of known structure. SCOP sequence searches, alignments and genome assignments. *Nucleic Acids Res* 30(1), 268–72.

Gough, J., Karplus, K., et al. (2001) Assignment of homology to genome sequences using a library of hidden Markov models that represent all proteins of known structure. *J Mol Biol* 313(4), 903–19.

Gribskov, M. and Veretnik, S. (1996) Identification of sequence pattern with profile analysis. *Methods Enzymol* 266, 198–212.

Grundy, W.N., Bailey, T.L., et al. (1997) Meta-MEME: motif-based hidden Markov models of protein families. *Comput Appl Biosci* 13(4), 397–406.

Henikoff, S. and Henikoff, J.G. (1992) Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci U S A* 89(22), 10915–9.

Hol, W.G. (1989) Protein crystallography and drug design. *Arzneimittelforschung* 39(8A), 1016–8; discussion 1019.

Jaroszewski, L., Li, W., et al. (2002) In search for more accurate alignments in the twilight zone. *Protein Sci* 11(7), 1702–13.

Jaroszewski, L., Rychlewski, L., et al. (2005) FFAS03: a server for profile--profile sequence alignments. *Nucleic Acids Res* 33(Web Server issue), W284–8.

Johnson, M.S., Srinivasan, N., et al. (1994) Knowledge-based protein modeling. *Crit Rev Biochem Mol Biol* 29(1), 1–68.

Jones, D.T. (1999) Protein secondary structure prediction based on position-specific scoring matrices. *J Mol Biol* 292(2), 195–202.

Jones, D.T., Tress, M., et al. (1999) Successful recognition of protein folds using threading methods biased by sequence similarity and predicted secondary structure. *Proteins* Suppl 3, 104–11.

Kabsch, W. and Sander, C. (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers* 22(12), 2577–637.

Kelley, L.A., MacCallum, R.M., et al. (2000) Enhanced genome annotation using structural profiles in the program 3D-PSSM. *J Mol Biol* 299(2), 499–520.

Kihara, D., Zhang, Y., et al. (2002) Ab initio protein structure prediction on a genomic scale: application to the Mycoplasma genitalium genome. *Proc Natl Acad Sci U S A* 99(9), 5993–8.

Kitson, D.H., Badretdinov, A., et al. (2002) Functional annotation of proteomic sequences based on consensus of sequence and structural analysis. *Brief Bioinform* 3(1), 32–44.

Koehl, P. and Levitt, M. (1999) A brighter future for protein structure prediction. *Nat Struct Biol* 6(2), 108–11.

Labesse, G. and Mornon, J. (1998) Tool for Incremental threading optimization (TITO) to help alignment and mTodelling of remote homologues. *Bioinformatics* 14(2), 206–11.

Lambert, C., Leonard, N., et al. (2002) ESyPred3D: Prediction of proteins 3D structures. *Bioinformatics* 18(9), 1250–6.

Liang, F., Holt, I., et al. (2000) Gene index analysis of the human genome estimates approximately 120,000 genes. *Nat Genet* 25(2), 239–40.

Lo Conte, L., Brenner, S.E., et al. (2002) SCOP database in 2002: refinements accommodate structural genomics. *Nucleic Acids Res* 30(1), 264–7.

Lundstrom, J., Rychlewski, L., et al. (2001) Pcons: a neural-network-based consensus predictor that improves fold recognition. *Protein Sci* 10(11), 2354–62.

Madera, M., Vogel, C., et al. (2004) The SUPERFAMILY database in 2004: additions and improvements. *Nucleic Acids Res* 32(Database issue), D235–9.

Martelli, P.L., Fariselli, P., et al. (2002) A sequence-profile-based HMM for predicting and discriminating beta barrel membrane proteins. *Bioinformatics* 18 Suppl 1, S46–53.

Marti-Renom, M.A., Stuart, A.C., et al. (2000) Comparative protein structure modeling of genes and genomes. *Annu Rev Biophys Biomol Struct* 29, 291–325.

Moult, J., Hubbard, T., et al. (1999) Critical assessment of methods of protein structure prediction (CASP): round III. *Proteins* Suppl 3, 2–6.

Moult, J., Pedersen, J.T., et al. (1995) A large-scale experiment to assess protein structure prediction methods. *Proteins* 23(3), ii–v.

Mumenthaler, C. and Braun, W. (1995) Predicting the helix packing of globular proteins by self-correcting distance geometry. *Protein Sci* 4(5), 863–71.

Neuwald, A.F., Liu, J.S., et al. (1997) Extracting protein alignment models from the sequence database. *Nucleic Acids Res* 25(9), 1665–77.

Norin, M. and Sundstrom, M. (2002) Structural proteomics: developments in structure-to-function predictions. *Trends Biotechnol* 20(2), 79–84.

Orengo, C.A., Bray, J.E., et al. (1999) Analysis and assessment of ab initio three-dimensional prediction, secondary structure, and contacts prediction. *Proteins* Suppl 3, 149–70.

Reddy, B.V., Li, W.W., et al. (2002) Use of conserved key amino acid positions to morph protein folds. *Biopolymers* 64(3), 139–45.

Rison, S.C., Hodgman, T.C., et al. (2000) Comparison of functional annotation schemes for genomes. *Funct Integr Genomics* 1(1), 56–69.

Rost, B. (1997) Better 1D predictions by experts with machines. *Proteins* Suppl 1, 192–7.

Salamov, A.A., Suwa, M., et al. (1999) Genome analysis: Assigning protein coding regions to three-dimensional structures. *Protein Sci* 8(4), 771–7.

Sali, A. (1998) 100,000 protein structures for the biologist. *Nat Struct Biol* 5(12), 1029–32.

Sasson, I. and Fischer, D. (2003) Modeling three-dimensional protein structures for CASP5 using the 3D-SHOTGUN meta-predictors. *Proteins* 53 Suppl 6, 389–94.

Schaffer, A.A., Aravind, L., et al. (2001) Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. *Nucleic Acids Res* 29(14), 2994–3005.

Schonbrun, J., Wedemeyer, W.J., et al. (2002) Protein structure prediction in 2002. *Curr Opin Struct Biol* 12(3), 348–54.

Shi, J., Blundell, T.L., et al. (2001) FUGUE: sequence-structure homology recognition using environment-specific substitution tables and structure-dependent gap penalties. *J Mol Biol* 310(1), 243–57.

Siew, N., Elofsson, A., et al. (2000) MaxSub: an automated measure for the assessment of protein structure prediction quality. *Bioinformatics* 16(9), 776–85.

Soman, K.V., Midoro-Horiuti, T., et al. (2000) Homology modeling and characterization of IgE binding epitopes of mountain cedar allergen Jun a 3. *Biophys J* 79(3), 1601–9.

Srinivasan, R. and Rose, G.D. (2002) Ab initio prediction of protein structure using LINUS. *Proteins* 47(4), 489–95.

Swanson, R. (1984) A vector representation for amino acid sequences. *Bull. Math. Biol* 46, 623–639.

Thornton, J. (2002) Gene family phylogenetics: tracing protein evolution on trees. *Exs*(92), 191–207.

Verlinde, C.L. and Hol, W.G. (1994) Structure-based drug design: progress, results and challenges. *Structure* 2(7), 577–87.

Waterston, R.H., Lander, E.S., et al. (2002) On the sequencing of the human genome. *Proc Natl Acad Sci U S A* 99(6), 3712–6.

Watson, J.D. (1990) The human genome project: past, present, and future. *Science* 248(4951), 44–9.

Xu, D., Crawford, O.H., et al. (2001) Application of PROSPECT in CASP4: characterizing protein structures with new folds. *Proteins* Suppl 5, 140–8.

Yu, L., White, J.V., et al. (1998) A homology identification method that combines protein sequence and structure information. *Protein Sci* 7(12), 2499–510.

Zhou, H. and Zhou, Y. (2005) Fold recognition by combining sequence profiles derived from evolution and from depth-dependent structural alignment of fragments. *Proteins* 58(2), 321–8.

Zhu, H., Schein, C.H., et al. (2000) MASIA: recognition of common patterns and properties in multiple aligned protein sequences. *Bioinformatics* 16(10), 950–1.

Chapter 7

Protein–Protein Interaction and Macromolecular Visualization

Arun Ramani[1], Venkatarajan S. Mathura[2], Cui Zhanhua[3], and Pandjassarame Kangueane[4]

[1]*University of Texas at Austin, USA*
[2]*Roskamp Institute, 2040 Whitfield Avenue, Sarasota, Florida 34243, USA*
[3]*Nanyang Technological University, Singapore*
[4]*Biomedical Informatics, Pondicherry, India*

Abstract: Large-scale prediction and understanding of protein–protein interaction is important to elucidate biological function. Protein interactions play a key role in signaling pathways, transportation, and other structural/functional roles in a cell. Several experimental and theoretical methods have been developed to predict a protein's interacting partners. Protein–protein docking is also gaining importance. A brief introduction to visualization tools for proteins is provided here.

Key words: Protein–protein interaction, Docking, Co-evolution

1. Introduction

The fundamental goal of molecular biology is to obtain a comprehensive understanding of the intricate workings of the cell, to explain the systems within the cell, their organization and interactions with one another, and the order and complexity derived from the interplay between these systems. The concerted development of experimental techniques and computational methods has provided us with a new set of tools to tackle these questions. These efforts have been fairly successful in providing insights into the inner workings of the cell. The complicated cellular milieu is composed of macromolecules and metabolites that carry out diverse functions. Macromolecular interactions, which include DNA, protein, lipid, and carbohydrates in all permutations, are fundamental to biological systems. Protein interactions play a major role in maintaining normal cell functions and physiology.

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_7,

Multi-protein complexes are emerging as important entities of biological activity inside cells that serve to create functional diversity by contextual combination of gene products and, at the same time, organize the large number of different proteins into functional units. Many a time, when studying protein complexes rather than individual proteins, the biological insight gained has been fundamental, particularly in cases in which proteins with no previous functional annotation could be placed into a functional context. The advances in technology made over the past few years now enable the study of protein complexes on a proteomic scale, and it can be anticipated that the knowledge gathered from such projects will fuel drug target discovery and validation pipelines, and that the technology will also prove valuable in the emerging field of systems biology.

Over the past several years there has been tremendous improvement in both experimental techniques for data generation using yeast two-hybrid technology (Ito et al., 2001; Uetz et al., 2000), affinity chromatography/mass spectrometry (Gavin et al., 2002), synthetic lethal assays (Tong et al., 2001; Tong et al., 2004), and computational methods for obtaining new data using genome context methods (Eisenberg et al., 2000; Mellor et al., 2002; Rzhetsky et al., 2004).

2. Experimental Methods

2.1 Yeast Two-Hybrid

The yeast two-hybrid assay is an elegant means of investigating protein–protein interactions. The yeast two-hybrid technique uses two protein domains that have specific functions: a DNA-binding domain (BD), that is capable of binding to DNA, and an activation domain (AD), that is capable of activating transcription of the DNA. In order for DNA to be transcribed, it requires a protein called a transcriptional activator (TA). This protein binds to the promoter, a region situated upstream from the gene that serves as a docking site for the transcriptional protein. Once the TA has bound to the promoter, it is then able to activate transcription via its activation domain. Hence, the activity of a TA requires both a DNA binding domain and an activation domain. If either of these domains is absent, then transcription of the gene will fail. The binding domain and the activation domain do not necessarily have to be on the same protein. In fact, a protein with a DNA binding domain can activate transcription when simply bound to another protein containing an activation domain; this principle forms the basis for the yeast two-hybrid technique.

In the two-hybrid assay, two fusion proteins are created: the protein of interest (X), which is constructed to have a DNA binding domain attached to its N-terminus, and its potential binding partner (Y), which is fused to an activation domain. If protein X interacts with protein Y, the binding of these two will form an intact and functional transcriptional activator. This newly formed transcriptional activator will then go on to transcribe a reporter gene, which is simply a gene whose protein product can be easily detected and measured. In this way, the amount of the reporter produced can be used as a measure of interaction between the protein of interest and its potential partner.

Generally, the yeast two-hybrid assay can identify novel protein–protein interactions. By using a number of different proteins as potential binding partners, it is possible to detect interactions that were previously uncharacterized. Secondly, the yeast two-hybrid assay can be used to characterize interactions already known to occur. Characterization could include determining which protein domains are responsible for the interaction, by using truncated proteins, or under what conditions interactions take place, by altering the intracellular environment.

2.2 Affinity Tagging

The purification of protein complexes has been accomplished by a multitude of different techniques ranging from classical methods such as size exclusion or ion exchange chromatography, to different varieties of affinity chromatography. The common theme of these is the use of an inherent interaction (affinity) of two biomolecules. If one of the molecules is immobilized on a solid support, the interacting molecule can be purified from cell lysate along with associated proteins. The classic co-immunoprecipitation (IP) experiment using antibodies is probably the most frequently employed method for testing whether two proteins are associated in vivo, but the method can also be successfully used for the discovery of novel interacting partners in a protein complex. Antibodies can be used in a more generic way for the isolation of protein complexes that circumvents the need for producing specific antibodies. For this purpose, bait proteins can be fused to an epitope-tag and an antibody, directed against the tag instead of the bait protein, is used for complex retrieval. As a result, many different cDNAs can be fused to the same tag in parallel and complexes retrieved using the same antibody.

2.3 Computational Methods

Computational methods for discovering specific protein interactions fall into three broad categories: (i) the identification of specific protein sequence or structural features indicative of protein interaction partners, such as sequence signatures (Sprinzak and Margalit, 2001), correlated mutations (Lockless and Ranganathan, 1999; Pazos and Valencia, 2002), and surface patches (Jones and Thornton, 1997; Lichtarge et al., 1996); (ii) the use of genomic context (Huynen et al., 2000) to identify interaction partners, exploiting information such as gene order (Dandekar et al., 1998; Overbeek et al., 1999), gene fusions (Enright et al., 1999; Marcotte et al., 1999), and phylogenetic profiles (Pellegrini et al., 1999), and (iii) the use of phylogenetic trees to account for the co-evolution of interacting proteins (Fryxell, 1996; Goh et al., 2000; Hughes and Yeager, 1999; Koretke et al., 2000; Pazos and Valencia, 2001)

Computational methods that use sequence data can be broadly classified into homology-based and non-homology-based methods. Homology-based methods refer to the inference of protein interactions and protein functions from direct comparison of protein sequences based on sequence similarity using the BLAST algorithms. There is no doubt that homology analysis remains the central methodology of genomics, that is, the one that produces the bulk of useful information. However, a group approaches in comparative genomics goes beyond sequence or structure comparison. These methods have become collectively known as genome context analysis. The notion of "context" here includes all types of associations between genes and proteins in the same or in different genomes that may point to functional interactions. If gene A is involved in function X and there is evidence that gene B functionally associates with A, then B could also be potentially involved in function X. More specifically, context in comparative genomics pertains to phylogenetic profiles of protein families, domain fusions in multidomain proteins, gene adjacency in genomes, and gene expression patterns. Indeed, genes whose products are involved in closely related functions (e.g. form different subunits of a multisubunit enzyme or participate in the same pathway) should all be either present or absent in a certain set of genomes (i.e. have similar if not identical phylogenetic patterns) and should be coordinately expressed (i.e. are expected to be encoded in the same operon or at least to have similar expression patterns). This simple logic gives us a potentially powerful way to assign genes that have no experimentally characterized homologs to particular pathways or cellular systems. Although context methods usually provide only rather general predictions, they represent a new and important development in genomics that explicitly takes advantage of the rapidly growing collection of sequenced genomes.

2.4 Co-evolution

Protein interaction specificity is vital to cell function, but the maintenance of such specificity requires that it persists even through the course of strong evolutionary change, such as the duplication and divergence of genes. Binding specificities of duplicate genes (paralogs) often diverge, such that new binding specificities are evolved. Given that such paralogous gene families abound, such as the >560 serine-threonine kinases in the human genome (Pruitt and Maglott, 2001), predicting interaction specificity can be difficult, especially when paralogs exist for both interaction partners. In these cases, the number of potential interactions grows combinatorially. This ambiguity can easily complicate the matching of ligands to specific receptors, and for such reasons, identification of ligands for orphan receptors is an important, but largely unsolved, problem (Chambers et al., 1999; Hsu et al., 2002; Saito et al., 1999). The hypothesis underlying this approach is that interacting proteins often exhibit coordinated evolution, and, therefore, tend to have similar phylogenetic trees. Goh et al. (Goh et al., 2000) demonstrated this by showing that chemokines and their receptors have very similar phylogenetic trees, as do individual domains of a single protein such as phosphoglycerate kinase. Detailed phylogenetic studies of the two-component signal transduction system (Koretke et al., 2000) show that a phylogenetic tree constructed from two-component sensor proteins has a similar structure to that from two-component regulator proteins.

2.5 Structure Based Methods

A structural perspective on protein–protein interactions (Russell et al., 2004) gives a good overview of all the methods. Protein–protein interactions occur at protein surfaces and are biophysical phenomena, governed by interface geometrical properties (interface size, planarity, sphericity, and complementarity) and chemical properties (the types of chemical groups in amino acids, hydrophobicity, electrostatic interactions, and hydrogen bonds). Towards the common goal of understanding how proteins interact, these properties have been studied using different dataset of protein structures by numerous groups (Ippolito et al., 1990; Janin and Chothia, 1990; Korn and Burnett, 1991; Stickle et al., 1992). These studies are influenced by dataset size and their characteristics. It should also be noted that these studies are based on limited dataset consisting of heterogeneous (mixture of homodimers and hetero-complexes) data.

2.5.1 Interface Size

Interface size is an important property used to describe protein interfaces and is usually characterized by interface area or the number of interface residues. The number of interface residues is shown to be linearly correlated to interface area (correlation coefficient: $r \geq 0.96$) (Bahadur et al., 2003; Chakrabarti and Janin, 2002). Jones & Thornton (Jones and Thornton, 1996) showed that homodimer interface area ranges from 368 $Å^2$ to 4746 $Å^2$ (based on a dataset of 32 dimers) and hetero-complex interface area ranges from 639 $Å^2$ to 3228 $Å^2$ (based on a dataset of 27 hetero-complexes). They also showed that, in general, interface area is often proportional to the total protein size. Dasgupta (Dasgupta et al., 1997) showed that the interface area per subunit ranges from 670 to 5540 $Å^2$ based on a sample of 23 oligomeric proteins. Lo Conte (Lo Conte et al., 1999) noted that most of the protein–protein complexes have an interface area in the range of 1200–2000 $Å^2$ and defined their interfaces as "standard size." For all the dataset (used a dataset of 75 hetero-complexes), the interface area varies from 1140 $Å^2$ to 4660 $Å^2$ with the mean value of 1940 $Å^2$. Jones (Jones et al., 2000) studied the differences between protein domain interfaces and oligomeric protein interfaces. The interface area in 46 monomeric two domain proteins ranges from 260 $Å^2$ to 3580 $Å^2$ and the interface area derived from 105 oligomeric or protein complexes ranges from 95 $Å^2$ to 2813 $Å^2$. Valdar & Thornton (Valdar and Thornton, 2001) showed that the residue conservation can help identify biologically relevant crystal contacts using a dataset of 53 families of homodimers and 65 families of monomers. The biological contact is shown to have 53.7 residues on average and this account for 25.9% of the protein surface residues. In contrast, the average non-biological contact has only 7.6 residues and covers 4.2% of the surface. Chakrabarti & Janin (Chakrabarti and Janin, 2002) studied 70 hetero-complexes and refined the identity of a typical interaction 'patch' (having an area of at least 800 $Å^2$, involving somewhat more than 20 residues and somewhat less than 100 atoms). These patches are composed of a core and a rim. Only the core residues are shown to have a composition distinct from that of the rest of the surface. The rim is interpreted as mainly isolating the core of the patch from the solvent, recalling the 'O-ring' theory (Bogan and Thorn, 1998). Single patch interface contains 47±11 residues or 23 residues per recognition site. The average interface area per subunit is 1906 $Å^2$ and the average number of interface residues is 57. Nooren & Thornton (Nooren and Thornton, 2003) analyzed the characteristics of transient protein–protein interactions using a dataset of 16 homodimers and 23 heterodimers. The interface area in homodimers is shown to range between 478 and 926 $Å^2$, which is 7–18% of the monomer surface. As compared to homodimers, the heterodimer

interface area ranges between 570 Å^2 and 2213 Å^2. Bahadur (Bahadur et al., 2003) showed that the range of interface area extends from 500 to 7000 Å^2 with the mean value of 1970 Å^2. Interfaces bury 16% of the subunit surface on average, but this fraction varies from 3 to 44% for a dataset of 122 homodimers. The average interface is shown to contain 52 residues per subunit. Although the smaller proteins obviously cannot form very large interfaces, the correlation with size is mediocre. Caffrey (Caffrey et al., 2004) used a dataset to study protein–protein interface conservation. The dataset consists of 42 chains that form homodimers, 12 chains that form heterodimers, and 10 chains that form transient complexes. The interface area is showed to range from 415 to 3568 Å^2 for heterodimers, 550–4718 Å^2 for homodimers, and 423–2361 Å^2 for transient complexes. The average number of interface residues is 44.4 (in homodimers) and 42.2 (in heterodimers). As we have seen from the above mentioned studies, interface sizes obtained can vary and this can be due to different datasets and their features, such as resolutions, size, or type these dataset.

2.5.2 Hydrogen Bonds

A hydrogen bond is a polar interaction between two electronegative atoms, one of which acts as a donor and the other as an acceptor (Jones and Thornton, 1995). The donor attracts the electron on the hydrogen with the result that the electron's orbit is more towards the donor itself. This leaves a partial positive charge on the hydrogen, which is electro-statically attracted towards the electronegative acceptor. The interaction is energetically favorable in a number of ways, including in terms of the polarization energy and the covalent energy, and particularly the electrostatic energy. Generally speaking, there are three main types of hydrogen bonds in protein structures, which are formed between main-chain and main-chain, main-chain and side-chain, side-chain and side-chain, apart from the solvent and hetero-atoms mediated hydrogen bonds (Ippolito et al., 1990; Janin and Chothia, 1990; Stickle et al., 1992). Although the energy of an average inter-molecular hydrogen bond is small, 20 KJ/mol (5 Kcal/mol) compared to 200 KJ/mol (Kcal/mol) in the case of a covalent bond, a great number of hydrogen bonds play an important role in protein–protein interactions. Studies show that there is a complementarity of hydrogen bond donor/acceptor sites (Janin and Chothia, 1990) and that having two or more intermolecular hydrogen bonds is intrinsic (Meyer et al., 1996). The numbers of inter-subunit hydrogen bonds found vary in different studies (Bahadur et al., 2003; Janin and Chothia, 1990; Jones and Thornton, 1995; Lo Conte et al., 1999; Nooren and Thornton, 2003; Xu et al., 1997) Janin & Chothia (Janin and Chothia, 1990) showed that there are 8–13 hydrogen bonds and a mean value of 10

hydrogen bonds per complex using 15 protease-inhibitor complexes, and 4 antibody-antigen complexes determined by x-ray crystallography. In protease-inhibitor complexes, there are two-thirds of hydrogen bonds involving main-chain atoms. In contrast, in antibody-antigen complexes most hydrogen bonds involve side-chain atoms. Jones & Thornton (Jones and Thornton, 1995) showed that there are 0–46 hydrogen bonds base on a dataset of 32 homodimers. On average, there are 0.88 hydrogen bonds per 100 $Å^2$ interface area with an *r* value of 0.77 between hydrogen bonds and interface area. Xu (Xu et al., 1997) showed 11 hydrogen bonds per subunit with an *r* value of 0.89 between hydrogen bonds and interface area based on the studies using a dataset of 319 protein–protein interfaces. They also showed that the estimated number of hydrogen bonds is sensitive to the geometric parameters of the bonds. The geometrical distribution of hydrogen bonds across the interfaces is non-optimal and is different from that of protein interior. The reason for this difference may be that there are more hydrophilic side-chains buried in the binding interfaces than in the protein interior. When folding, proteins are completely free to attain their optimal configurations. But when binding each other, protein molecules are already folded and have limited freedom of six degrees of translation and rotation available to achieve the most favorable configuration (Archakov et al., 2003; Xu et al., 1997). Lo Conte (Lo Conte et al., 1999) showed there are 1–34 hydrogen bonds with the mean value of 10.1 hydrogen bonds based on the study using a dataset of 75 hetero-complexes. However, there are nine hydrogen bonds in standard interfaces (defined by them with interface area range of between 1200 and 2000 $Å^2$). They also analyzed effects on the number of hydrogen bonds caused by dataset resolution. Study showed that there is one hydrogen bond per 170 $Å^2$ interface area and a *r* value of 0.84 between hydrogen bonds and interface area for 36 complexes (resolution ≤ 2.4 Å). For studies using structures with lower resolution, there were fewer hydrogen bonds and the correlation with interface area vanishes, which suggests that errors in atomic co-ordinates mask existing hydrogen bonds. Bahadur (Bahadur et al., 2003) (used a dataset of 122 homodimers) showed an average 9.0 hydrogen bonds per homodimer interface with a *r* value of 0.75 between hydrogen bonds and interface area. On average, there is one hydrogen bond per 210 $Å^2$ and the correlation is better with the polar interface area ($r = 0.83$). These findings suggest that the number of hydrogen bonds and the correlation with interface area are influenced by dataset size and their characteristics, especially the structure resolution of the dataset used.

2.5.3 Hydrophobicity

It has been demonstrated that the hydrophobic forces play an important role in protein–protein interaction (Bahadur et al., 2003; Korn and Burnett, 1991; Lijnzaad and Argos, 1997; Tsai et al., 1997; Tsai and Nussinov, 1997; Wells, 1996). The average values of contact surface hydrophobicity usually represent the mean of the hydrophobicity of the protein core and its surface (Jones and Thornton, 1996). Also, the ratio between buried hydrophobic and buried hydrophilic residues is used to measure the hydrophobic effect (Tsai and Nussinov, 1997). Studies showed that hydrophobic residues (except ALA) and the charged residue ARG are found to have an increased presence at protein–protein interfaces (Bahadur et al., 2003; Brinda et al., 2002; Dasgupta et al., 1997; Lijnzaad and Argos, 1997; Lo Conte et al., 1999; Ma et al., 2003; Zhou and Shan, 2001). Especially, TYR and TRP are found to have the highest propensity to stay at the interfaces. It has even been suggested that the binding energy of two proteins derives from the burying of hydrophobic surface areas (Chothia and Janin, 1975). Jones & Thornton (Jones and Thornton, 1996) also assumed that proteins will associate with each other by hydrophobic patches. They calculated the mean hydrophobic value for all interface residues of each complex. In all of the calculated complexes, the interface hydrophobicity is intermediate between that of the interior and the exterior. Generally speaking, the protein has a hydrophobic core in the interior and a more hydrophilic surface (Hirakawa et al., 1999). When comparing the interface hydrophobicity between homodimers and hetero-complexes, it is found that the homodimer interfaces are more hydrophobic. The difference in hydrophobicity may originate from the roles of two types of complexes (Janin et al., 1988). The hetero-complexes often function as monomers in solution and their interfaces cannot be as hydrophobic as homodimer interfaces. In contrast, the homodimers usually function in dimer forms. Hence, they have hydrophobic interfaces permanently buried within the inter-subunits. It is energetically unfavorable to have a large exposed hydrophobic area on the proteins. Tsai (Tsai and Nussinov, 1997) studied the hydrophobic effect in protein–protein interactions using a dataset of 362 protein–protein interfaces and 57 symmetry-related oligomeric interfaces. The hydrophobic effect was measured by the buried non-polar surface area or percent burial of residue types. Studies showed that the ratio between buried hydrophobic and buried hydrophilic residues is approximately 1.5. The interior of the interfaces appear to constitute a compromise between the stabilization contributed by the hydrophobic effect on the one hand and avoiding patches on the protein surfaces that are too hydrophobic on the other. The overriding conclusion is

that the hydrophobic effect plays a dominant role in protein–protein interfaces (Dill, 1990). However, it is not as strong as that observed in protein folding (Tsai et al., 1997). Nevertheless, there are exceptions, where there is no sign of a significant hydrophobic contribution at the interface (Tsai et al., 1997). Most studies analyzed the average hydrophobicity over a diverse set of protein–protein interfaces. It suffers the drawback that it blurs information on how individual interfaces are stabilized and cannot show how hydrophobic feature is distributed over the individual interfaces. Larsen (Larsen et al., 1998) studied the hydrophobic features of each protein–protein interface and highlighted them with images. Although all the interfaces are formed between two globular subunits, the hydrophobic distribution pattern over the interfaces is quite variable: (1) some interfaces show a recognizable hydrophobic core, with a single large, continuous, hydrophobic patch surrounded by a ring of inter-subunit polar interactions; (2) some interfaces do not have a single hydrophobic core; instead, they have many small hydrophobic patches which consist of 1–3 amino acids. These small hydrophobic patches are not discrete, they are linked with inter-subunit hydrogen bonds and water molecules and all patches are distributed across the interface. Moreover, the scattered hydrophobic patches do not dominate the interface character like the single large hydrophobic core; (3) there is still a small portion of the interfaces formed by extensive inter-digitation of the two subunit chains. These interfaces are highly hydrophobic similar to the hydrophobic cores inside the folded domain. They are usually associated with proteins that are quite stable and internally symmetric. From the studies on proteins folding path, they may be formed through two-state folded theory. Hydrophobic effect is important to protein–protein interactions. So, understanding it will help unveil the nature of protein–protein interactions, and help build the prediction model of protein interaction sites.

2.5.4 Amino Acid Composition at Interfaces

Many studies analyzed amino acids composition at protein–protein interfaces (Dasgupta et al., 1997; Jones and Thornton, 1996; Lijnzaad and Argos, 1997; Lo Conte et al., 1999). Different studies found that the amino acid distributions differ in different protein complexes. In small globular proteins the interface consists of 57% non-polar residues, 24% neutral polar residues, and 19% charged residues (Miller et al., 1987). While in the oligomeric proteins it consists of 65% non-polar residues, 22% neutral polar residues, and 13% charged residues, respectively (Janin et al., 1988). It indicates that the amino acid composition at the protein–protein interface is

more similar to that of the protein surface than protein interior. Jones & Thornton (Jones and Thornton, 1996) studied residue propensities at the interfaces. The results showed that the charged and polar residues, especially ARG and ASP, show an increased affinity for the interface. In addition, the hydrophobic residues MET and PRO show a slightly increased affinity for the interface. On average, the interface comprises 56% non-polar carbon-containing groups, 29% neutral polar groups, and 15% charged groups (Jones and Thornton, 1996). Lijnzaad & Argos (Lijnzaad and Argos, 1997) showed that aliphatic and aromatic residues as well as PRO are the largest contributors to interface. LEU, ILE, PHE, VAL, and PRO occur progressively more often in the larger patches, whereas the contributions of TRP and TYR roughly depend on the patch size. And hydrophobic residues were abundant in large interfaces while polar residues were more abundant in small interfaces (Glaser et al., 2001; Zhou and Shan, 2001). ALA and MET are intermediate contributors. Charged residues are shown to have less affinity to the interface, especially as the patch size grows. Dasgupta (Dasgupta et al., 1997) showed that hydrophobic interactions at oligomer interfaces favor aromatic amino acids and MET over aliphatic amino acids. However, ARG is the top residues appearing at the oligomeric interfaces. This suggests that ARG might be very helpful to inter-subunit interactions. Musafia (Musafia et al., 1995) showed that ARG is especially adaptable to the formation of multiple salt bridge interactions, which may explain its structural role at oligomer and crystal interfaces. Lo Conte (Lo Conte et al., 1999) showed that interfaces are much richer in aromatic residues HIS, TYR, PHE, and TRP than the average protein surface (21% versus 8%), and somewhat richer in aliphatic residues LEU, ILE, VAL, and MET (17% versus 11%). They are depleted in the charged residues ASP, GLU, and LYS, but not ARG (Zhou and Shan, 2001), which is the only residue that makes the largest overall contribution to interfaces (10%). Some variations are seen between different types of complexes, but the largest contribution of ARG and depletion in LYS are general conclusions. Chakrabarti & Janin (Chakrabarti and Janin, 2002) showed using a dataset of 70 protein–protein complexes that TRP and TYR have the highest propensity for the core of recognition sites, and SER and THR have a negative propensity. ARG is the most abundant core residue but also generally abundant on the protein surface. Brinda (Brinda et al., 2002) analyzed 20 homodimer interfaces using graph-spectral methods. ARG, HIS, PHE, TYR, and GLU are found to be the most preferred residues in the interface clusters. There is also a significant contribution from TRP and MET in the interface clusters when compared with the other amino acids. It is clear that the charged and aromatic residues prefer to stay interface clusters. In contrast, GLY, ALA,

VAL, and CYS are rarely found in the interface side chain clusters. Ma (Ma et al., 2003) studied structurally conserved residues between binding sites and exposed protein surfaces. The results showed that conservation of TRP on the protein surface indicates a highly potential binding site. To a less extent, conservation of PHE and MET also implies a binding site. For all three residues, there is a significant conservation in binding sites, whereas there is no conservation on the exposed surface. Bahadur (Bahadur et al., 2003) studied subunit interfaces using a dataset of 122 homodimers. LEU was shown to be the most abundant residue at the homodimer interfaces, which contributed about 10% of the interface area. Other aliphatic residues, ILE, VAL, and MET are also main contributors to interfaces. Aromatic residues, PHE, TYR, and TRP are a little more abundant at the interface than at the protein surface. Charged residues, ASP, GLU, and LYS are depleted at the interface. But ARG is the second largest contributor to homodimer interfaces after LEU. Neuvirth (Neuvirth et al., 2004) counted TYR, MET, CYS, and HIS as the most favored to be at the interface. While residues, THR, PRO, LYS, GLU, and ALA, were least commonly found at the interfaces. In this study, ARG was not found to be abundant at the interfaces. Rajamani (Rajamani et al., 2004) showed a particularity of protein interfaces which is the existence of so-called 'anchor' residues. These residues can bind to specific pockets in the target protein and create a weak intermediate state that can rapidly convert to the final complex. Molecular dynamic simulations suggest that anchor residues adopt similar conformations in the complex and in the corresponding isolated protein, although this may not be reflected by their conformations in protein crystal structures due to packing or environmental effects. Another well-characterized property of protein interfaces is the existence of 'hot-spot' residues (Halperin et al., 2004; Keskin et al., 2005; Ma et al., 2003). These residues are found to make a significant thermodynamic contribution to the complex formation in so-called 'alanine scanning' experiments. It has been shown that polar residue occur at binding interfaces and are correlated with residue conservation. The number of such residues appears to be proportional to the size of the interface. In conclusion, hydrophobic residues (except ALA) showed an increased affinity to protein–protein interfaces. Moreover, aromatic residues are a little more abundant at the interfaces than aliphatic residues. TYR and TRP are found to have the highest propensity at the interfaces. Hydrophilic and charged residues (except ARG) are found to be depleted at the interfaces. Most studies showed that ARG is among the top contributors to the interfaces.

2.5.5 Other Properties of Protein–Protein Interfaces

Besides the properties discussed above, there are also some other features characterizing protein–protein interfaces. These features include: (1) interface shape (Chakrabarti and Janin, 2002; Hurley et al., 1989; Jones and Thornton, 1996), (2) geometrical complementarity (Harpaz et al., 1994; Jones and Thornton, 1996; Lawrence and Colman, 1993; Lo Conte et al., 1999), (3) secondary structures of interface residues (Dasgupta et al., 1997; Jones and Thornton, 1996; Neuvirth et al., 2004; Tsai et al., 1997), (4) electrostatic complementarity (Janin and Chothia, 1990; Nicholls et al., 1991; Novotny and Sharp, 1992; Roberts et al., 1991), (5) water molecule effect on protein–protein interactions (Davies and Cohen, 1996; Guinto and Di Cera, 1996; Stites, 1997), and (6) conformational changes upon protein–protein association (Davies and Cohen, 1996; de Vos et al., 1992; Stanfield et al., 1993; Wilson and Stanfield, 1994).

3. Protein Structure Visualization

There are several commercial and free tools (for academic purposes) available for protein visualization. Using visual tools provides necessary perspective to understand the orientation of molecules and their arrangement in three-dimensional space. RASMOL (http://www.umass.edu/microbio/rasmol/) is a free renderer that can display proteins and organic molecules. It has a powerful scripting language and simple visual display. One can visualize proteins in wireframe, ribbons, cartoons, or space-fill mode. MOLMOL (http://www.mol.biol.ethz.ch/groups/wuthrich_group/software) is a powerful and more advanced visualization system in which a user can display several proteins at the same time. It can be used for visual docking, backbone fitting, and RMSD calculations. It can also calculate electrostatic potentials and map them on a surface representation. PyMOL (http://pymol.sourceforge.net) is python-based molecular visualization program that can be used to visualize proteins, ligands, and surfaces. Several of these visualization programs share common functionalities. In-depth understandings on how to use these tools are available at their web sites.

4. Databases

Protein–protein interactions and information on complexes are still being developed. Thus, there are only a few databases created based on large-scale binding experiments, literature reports, and submission from experimental

community. MINT is a database of functional interactions between biological molecules like proteins, RNA, and DNA (Zanzoni et al., 2002). Experts curate protein–protein interactions or other biological macromolecular interactions reported in the scientific literature and the information is stored in a rational database that can be queried over the internet. Presently MINT contains 4568 interactions including genetic interactions (Protein-DNA). The BIND (Bader et al., 2003) database is designed to store descriptions of interactions, molecular complexes, and pathways. This database can be queried over the web for interaction records. The interaction records are submitted by users with details of interaction sites, and sub-cellular location of the complex. The DIP database also contains details of protein–protein interactions and provides paralogous verification score based on putative interaction. The DIP database linked to SWISSPROT annotation can be queried using the DIP node id. All three databases are provided as online service and some (BIND) can be downloaded as structured file. Other useful protein interaction databases include Protein–Protein Interaction Database (PPID, http://www.anc.ed.ac.uk/mscs/PPID/) and Human Protein Reference Database (HPRD, http://www.hprd.org. Useful urls for databases and tools are listed in Table 7.1.

Table 7.1 List of useful URLs

Tool	URL
Bioverse	http://bioverse.compbio.washington.edu
In silico two hybrid	http://ecid.bioinfo.cnio.es/
InterDom	http://datam.i2r.a-star.edu.sg/interdom/
Magic	http://genome-www.stanford.edu/magic
ProtFun	http://www.cbs.dtu.dk/services/ProtFun
ProteinFunction	http://www.aber.ac.uk/compsci/Research/bio/ProteinFunction
PLEX	http://bioinformatics.icmb.utexas.edu/plex
STRING	http://www.bork.embl-heidelberg.de/STRING
Gene Neighbors	http://bioinformatics.icmb.utexas.edu/operons
WIT	http://wit.mcs.anl.gov/WIT2
InParanoid	http://inparanoid.cgb.ki.se
Clusters of Orthologs (COGs)	http://www.ncbi.nlm.nih.gov/COG
Bind	http://www.bind.ca
BRITE	http://www.genome.ad.jp/brite
Database of Interacting Proteins	http://dip.doe-mbi.ucla.edu
GRID	http://www.thebiogrid.org/
MIPS	http://mips.gsf.de/proj/yeast/CYGD/db/index.html
PIMRider	http://pim.hybrigenics.com
REACTOME	http://www.reactome.org

References

Archakov, A. I., Govorun, V. M., Dubanov, A. V., Ivanov, Y. D., Veselovsky, A. V., Lewi, P., and Janssen, P. (2003). Protein-protein interactions as a target for drugs in proteomics, Proteomics *3*, 380–91.

Bader, G. D., Betel, D., and Hogue, C. W. (2003). BIND: the Biomolecular Interaction Network Database, Nucleic Acids Res *31*, 248–50.

Bahadur, R. P., Chakrabarti, P., Rodier, F., and Janin, J. (2003). Dissecting subunit interfaces in homodimeric proteins, Proteins *53*, 708–19.

Bogan, A. A., and Thorn, K. S. (1998). Anatomy of hot spots in protein interfaces, J Mol Biol *280*, 1–9.

Brinda, K. V., Kannan, N., and Vishveshwara, S. (2002). Analysis of homodimeric protein interfaces by graph-spectral methods, Protein Eng *15*, 265–77.

Caffrey, D. R., Somaroo, S., Hughes, J. D., Mintseris, J., and Huang, E. S. (2004). Are protein-protein interfaces more conserved in sequence than the rest of the protein surface? Protein Sci *13*, 190–202.

Chakrabarti, P., and Janin, J. (2002). Dissecting protein-protein recognition sites, Proteins *47*, 334–43.

Chambers, J., Ames, R. S., Bergsma, D., Muir, A., Fitzgerald, L. R., Hervieu, G., Dytko, G. M., Foley, J. J., Martin, J., Liu, W. S., et al. (1999). Melanin-concentrating hormone is the cognate ligand for the orphan G-protein-coupled receptor SLC-1, Nature *400*, 261–5.

Chothia, C., and Janin, J. (1975). Principles of protein-protein recognition, Nature *256*, 705–8.

Dandekar, T., Snel, B., Huynen, M., and Bork, P. (1998). Conservation of gene order: a fingerprint of proteins that physically interact, Trends Biochem Sci *23*, 324–8.

Dasgupta, S., Iyer, G. H., Bryant, S. H., Lawrence, C. E., and Bell, J. A. (1997). Extent and nature of contacts between protein molecules in crystal lattices and between subunits of protein oligomers, Proteins *28*, 494–514.

Davies, D. R., and Cohen, G. H. (1996). Interactions of protein antigens with antibodies, Proc Natl Acad Sci U S A *93*, 7–12.

de Vos, A. M., Ultsch, M., and Kossiakoff, A. A. (1992). Human growth hormone and extracellular domain of its receptor: crystal structure of the complex, Science *255*, 306–12.

Dill, K. A. (1990). Dominant forces in protein folding, Biochemistry *29*, 7133–55.

Eisenberg, D., Marcotte, E. M., Xenarios, I., and Yeates, T. O. (2000). Protein function in the post-genomic era, Nature *405*, 823–6.

Enright, A. J., Iliopoulos, I., Kyrpides, N. C., and Ouzounis, C. A. (1999). Protein interaction maps for complete genomes based on gene fusion events, Nature *402*, 86–90.

Fryxell, K. J. (1996). The coevolution of gene family trees, Trends Genet *12*, 364–9.

Gavin, A. C., Bosche, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A., Schultz, J., Rick, J. M., Michon, A. M., Cruciat, C. M., et al. (2002). Functional organization of the yeast proteome by systematic analysis of protein complexes, Nature *415*, 141–7.

Glaser, F., Steinberg, D. M., Vakser, I. A., and Ben-Tal, N. (2001). Residue frequencies and pairing preferences at protein-protein interfaces, Proteins *43*, 89–102.

Goh, C. S., Bogan, A. A., Joachimiak, M., Walther, D., and Cohen, F. E. (2000). Co-evolution of proteins with their interaction partners, J Mol Biol *299*, 283–93.

Guinto, E. R., and Di Cera, E. (1996). Large heat capacity change in a protein-monovalent cation interaction, Biochemistry *35*, 8800–4.

Halperin, I., Wolfson, H., and Nussinov, R. (2004). Protein-protein interactions; coupling of structurally conserved residues and of hot spots across interfaces. Implications for docking, Structure *12*, 1027–38.

Harpaz, Y., Gerstein, M., and Chothia, C. (1994). Volume changes on protein folding, Structure *2*, 641–9.

Hirakawa, H., Muta, S., and Kuhara, S. (1999). The hydrophobic cores of proteins predicted by wavelet analysis, Bioinformatics *15*, 141–8.

Hsu, S. Y., Nakabayashi, K., Nishi, S., Kumagai, J., Kudo, M., Sherwood, O. D., and Hsueh, A. J. (2002). Activation of orphan receptors by the hormone relaxin, Science *295*, 671–4.

Hughes, A. L., and Yeager, M. (1999). Coevolution of the mammalian chemokines and their receptors, Immunogenetics *49*, 115–24.

Hurley, J. H., Thorsness, P. E., Ramalingam, V., Helmers, N. H., Koshland, D. E., Jr., and Stroud, R. M. (1989). Structure of a bacterial enzyme regulated by phosphorylation, isocitrate dehydrogenase, Proc Natl Acad Sci U S A *86*, 8635–9.

Huynen, M., Snel, B., Lathe, W., 3rd, and Bork, P. (2000). Predicting protein function by genomic context: quantitative evaluation and qualitative inferences, Genome Res *10*, 1204–10.

Ippolito, J. A., Alexander, R. S., and Christianson, D. W. (1990). Hydrogen bond stereochemistry in protein structure and function, J Mol Biol *215*, 457–71.

Ito, T., Chiba, T., Ozawa, R., Yoshida, M., Hattori, M., and Sakaki, Y. (2001). A comprehensive two-hybrid analysis to explore the yeast protein interactome, Proc Natl Acad Sci U S A *98*, 4569–74.

Janin, J., and Chothia, C. (1990). The structure of protein-protein recognition sites, J Biol Chem *265*, 16027–30.

Janin, J., Miller, S., and Chothia, C. (1988). Surface, subunit interfaces and interior of oligomeric proteins, J Mol Biol *204*, 155–64.

Jones, S., Marin, A., and Thornton, J. M. (2000). Protein domain interfaces: characterization and comparison with oligomeric protein interfaces, Protein Eng *13*, 77–82.

Jones, S., and Thornton, J. M. (1995). Protein-protein interactions: a review of protein dimer structures, Prog Biophys Mol Biol *63*, 31–65.

Jones, S., and Thornton, J. M. (1996). Principles of protein-protein interactions, Proc Natl Acad Sci U S A *93*, 13–20.

Jones, S., and Thornton, J. M. (1997). Prediction of protein-protein interaction sites using patch analysis, J Mol Biol *272*, 133–43.

Keskin, O., Ma, B., and Nussinov, R. (2005). Hot regions in protein--protein interactions: the organization and contribution of structurally conserved hot spot residues, J Mol Biol *345*, 1281–94.

Koretke, K. K., Lupas, A. N., Warren, P. V., Rosenberg, M., and Brown, J. R. (2000). Evolution of two-component signal transduction, Mol Biol Evol *17*, 1956–70.

Korn, A. P., and Burnett, R. M. (1991). Distribution and complementarity of hydropathy in multisubunit proteins, Proteins *9*, 37–55.

Larsen, T. A., Olson, A. J., and Goodsell, D. S. (1998). Morphology of protein-protein interfaces, Structure *6*, 421–7.

Lawrence, M. C., and Colman, P. M. (1993). Shape complementarity at protein/protein interfaces, J Mol Biol *234*, 946–50.

Lichtarge, O., Bourne, H. R., and Cohen, F. E. (1996). An evolutionary trace method defines binding surfaces common to protein families, J Mol Biol *257*, 342–58.

Lijnzaad, P., and Argos, P. (1997). Hydrophobic patches on protein subunit interfaces: characteristics and prediction, Proteins *28*, 333–43.

Lockless, S. W., and Ranganathan, R. (1999). Evolutionarily conserved pathways of energetic connectivity in protein families, Science *286*, 295–9.

Lo Conte, L., Chothia, C., and Janin, J. (1999). The atomic structure of protein-protein recognition sites, J Mol Biol *285*, 2177–98.

Ma, B., Elkayam, T., Wolfson, H., and Nussinov, R. (2003). Protein-protein interactions: structurally conserved residues distinguish between binding sites and exposed protein surfaces, Proc Natl Acad Sci U S A *100*, 5772–7.

Marcotte, E. M., Pellegrini, M., Ng, H. L., Rice, D. W., Yeates, T. O., and Eisenberg, D. (1999). Detecting protein function and protein-protein interactions from genome sequences, Science *285*, 751–3.

Mellor, J. C., Yanai, I., Clodfelter, K. H., Mintseris, J., and DeLisi, C. (2002). Predictome: a database of putative functional links between proteins, Nucleic Acids Res *30*, 306–9.

Meyer, M., Wilson, P., and Schomburg, D. (1996). Hydrogen bonding and molecular surface shape complementarity as a basis for protein docking, J Mol Biol *264*, 199–210.

Miller, S., Lesk, A. M., Janin, J., and Chothia, C. (1987). The accessible surface area and stability of oligomeric proteins, Nature *328*, 834–6.

Musafia, B., Buchner, V., and Arad, D. (1995). Complex salt bridges in proteins: statistical analysis of structure and function, J Mol Biol *254*, 761–70.

Neuvirth, H., Raz, R., and Schreiber, G. (2004). ProMate: a structure based prediction program to identify the location of protein-protein binding sites, J Mol Biol *338*, 181–99.

Nicholls, A., Sharp, K. A., and Honig, B. (1991). Protein folding and association: insights from the interfacial and thermodynamic properties of hydrocarbons, Proteins *11*, 281–96.

Nooren, I. M., and Thornton, J. M. (2003). Structural characterisation and functional significance of transient protein-protein interactions, J Mol Biol *325*, 991–1018.

Novotny, J., and Sharp, K. (1992). Electrostatic fields in antibodies and antibody/antigen complexes, Prog Biophys Mol Biol *58*, 203–24.

Overbeek, R., Fonstein, M., D'Souza, M., Pusch, G. D., and Maltsev, N. (1999). The use of gene clusters to infer functional coupling, Proc Natl Acad Sci U S A *96*, 2896–901.

Pazos, F., and Valencia, A. (2001). Similarity of phylogenetic trees as indicator of protein-protein interaction, Protein Eng *14*, 609–14.

Pazos, F., and Valencia, A. (2002). In silico two-hybrid system for the selection of physically interacting protein pairs, Proteins *47*, 219–27.

Pellegrini, M., Marcotte, E. M., Thompson, M. J., Eisenberg, D., and Yeates, T. O. (1999). Assigning protein functions by comparative genome analysis: protein phylogenetic profiles, Proc Natl Acad Sci U S A *96*, 4285–8.

Pruitt, K. D., and Maglott, D. R. (2001). RefSeq and LocusLink: NCBI gene-centered resources, Nucleic Acids Res *29*, 137–40.

Rajamani, D., Thiel, S., Vajda, S., and Camacho, C. J. (2004). Anchor residues in protein-protein interactions, Proc Natl Acad Sci U S A *101*, 11287–92.

Roberts, V. A., Freeman, H. C., Olson, A. J., Tainer, J. A., and Getzoff, E. D. (1991). Electrostatic orientation of the electron-transfer complex between plastocyanin and cytochrome c, J Biol Chem *266*, 13431–41.

Russell, R. B., Alber, F., Aloy, P., Davis, F. P., Korkin, D., Pichaud, M., Topf, M., and Sali, A. (2004). A structural perspective on protein-protein interactions, Curr Opin Struct Biol *14*, 313–24.

Rzhetsky, A., Iossifov, I., Koike, T., Krauthammer, M., Kra, P., Morris, M., Yu, H., Duboue, P. A., Weng, W., Wilbur, W. J., et al. (2004). GeneWays: a system for extracting, analyzing, visualizing, and integrating molecular pathway data, J Biomed Inform *37*, 43–53.

Saito, Y., Nothacker, H. P., Wang, Z., Lin, S. H., Leslie, F., and Civelli, O. (1999). Molecular characterization of the melanin-concentrating-hormone receptor, Nature *400*, 265–9.

Sprinzak, E., and Margalit, H. (2001). Correlated sequence-signatures as markers of protein-protein interaction, J Mol Biol *311*, 681–92.

Stanfield, R. L., Takimoto-Kamimura, M., Rini, J. M., Profy, A. T., and Wilson, I. A. (1993). Major antigen-induced domain rearrangements in an antibody, Structure *1*, 83–93.

Stickle, D. F., Presta, L. G., Dill, K. A., and Rose, G. D. (1992). Hydrogen bonding in globular proteins, J Mol Biol *226*, 1143–59.

Stites, W. E. (1997). Proteinminus signProtein Interactions: Interface Structure, Binding Thermodynamics, and Mutational Analysis, Chem Rev *97*, 1233–1250.

Tong, A. H., Evangelista, M., Parsons, A. B., Xu, H., Bader, G. D., Page, N., Robinson, M., Raghibizadeh, S., Hogue, C. W., Bussey, H., et al. (2001). Systematic genetic analysis with ordered arrays of yeast deletion mutants, Science *294*, 2364–8.

Tong, A. H., Lesage, G., Bader, G. D., Ding, H., Xu, H., Xin, X., Young, J., Berriz, G. F., Brost, R. L., Chang, M., et al. (2004). Global mapping of the yeast genetic interaction network, Science *303*, 808–13.

Tsai, C. J., Lin, S. L., Wolfson, H. J., and Nussinov, R. (1997). Studies of protein-protein interfaces: a statistical analysis of the hydrophobic effect, Protein Sci *6*, 53–64.

Tsai, C. J., and Nussinov, R. (1997). Hydrophobic folding units at protein-protein interfaces: implications to protein folding and to protein-protein association, Protein Sci *6*, 1426–37.

Uetz, P., Giot, L., Cagney, G., Mansfield, T. A., Judson, R. S., Knight, J. R., Lockshon, D., Narayan, V., Srinivasan, M., Pochart, P., et al. (2000). A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae, Nature *403*, 623–7.

Valdar, W. S., and Thornton, J. M. (2001). Conservation helps to identify biologically relevant crystal contacts, J Mol Biol *313*, 399–416.

Wells, J. A. (1996). Binding in the growth hormone receptor complex, Proc Natl Acad Sci U S A *93*, 1–6.

Wilson, I. A., and Stanfield, R. L. (1994). Antibody-antigen interactions: new structures and new conformational changes, Curr Opin Struct Biol *4*, 857–67.

Xu, D., Tsai, C. J., and Nussinov, R. (1997). Hydrogen bonds and salt bridges across protein-protein interfaces, Protein Eng *10*, 999–1012.

Zanzoni, A., Montecchi-Palazzi, L., Quondam, M., Ausiello, G., Helmer-Citterich, M., and Cesareni, G. (2002). MINT: a Molecular INTeraction database, FEBS Lett *513*, 135–40.

Zhou, H. X., and Shan, Y. (2001). Prediction of protein interaction sites from sequence profile and residue neighbor list, Proteins *44*, 336–43.

Chapter 8

Genes, Genomics, Microarray Methods, and Analysis

Ghania Ait-Ghezala and Venkatarajan S. Mathura

Roskamp Institute, 2040 Whitfield Avenue, Sarasota, FL 34243, USA

Abstract: Genes are defined regions of DNA that codes for proteins and forms the blueprint of living organisms. While classical molecular biology experiments attempt to characterize genes, modern large-scale analysis of gene expression techniques provides clue about regulations and control systems underlying biological process. DNA Microarray experiments attempts to capture snapshot of transcriptome. Currently high density oligonucleotide and spotted cDNA microarray are widely used for probing genomic markers. Normalized intensity values are used to identify significantly regulated genes. Biological pathways and networks that contain several regulated genes are identified further for higher-level interpretation.

Key words: Microarray, DNA chip, Oligonucleotides, Biological pathways

1. Introduction

Genomics is the study of genes, gene content, gene regulations, and transcriptome or mRNA copy numbers that characterizes biological process (McKusick, 1997). The sequence of the human genome was only the first milestone towards understanding the information coded in the DNA. The next stage of the genomic research is to drive significant knowledge from several genome projects (O'Brien et al., 1997). Even with basically all of the human genome sequence availability, the number of protein–coding genes can still only be estimated (currently 20,000–25,000). Furthermore, the specific functions of all these genes remain to be determined. The challenge is to interpret and learn how to use that information to drive a meaningful understanding of the biology of human health and disease. Much work is required to determine the function and the elements that regulate these genes throughout the genome, find variations in the DNA sequence among people (like Single Nucleotide Polymorphisms or SNPs), and determine their

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_8,

significance (Schimenti and Bucan, 1998). These variations may one day provide information about an individual's disease risk and adverse reactions to certain medications, develop and apply genome-based strategies for the early detection, diagnosis, and treatment of diseases. Transcription of DNA to RNA results in increase in the copy number of mRNA that may finally result in abundance of specific proteins by translation. Transcriptome refers to set of all available mRNA transcripts in a cell at a given time or in a biological state (like a cancer cell). Large-scale RNA expression studies use microarray experiments to identify differential regulation of genes, identify sequence variants, or nucleotide polymorphisms. Gene expression signatures can then be used to infer and further hypothesize candidate networks, pathways, or identify gene lists involved in a diseased state (Ait-Ghezala et al., 2005). Such expression signatures can also be used as diagnostic biomarker for specific disease. Recently large-scale analysis of DNA sequencing arrays has made genome scale SNP analysis possible. The first section of this chapter describes some details regarding gene characterizing methods and techniques. The second half deals with microarray data analysis.

2. Gene Identification and Characterization

2.1 Identifying Human Genes and Cloning

There are four strategies for identifying human genes:

- *Functional cloning*: information about the function of an unidentified gene is used to isolate the gene – either a gene product or a functional assay is required. This method has very limited application. Examples include genes identified for diseases like Phenyl Ketonuria (PKU) and Hemophilia type A (Clos and Choudhury , 2006).
- *Candidate gene approach*: requires sufficient information about the molecular basis of pathogenesis or the existence of a suitable animal or human model where the gene is already known to be able to make an educated guess.
- *Positional cloning*: isolation of a gene knowing only its chromosomal location, which is typically identified by linkage analysis. Construct physical and genetic map of candidate region, identify the genes within the region, and investigate each candidate gene until the disease gene is identified.
- *Positional candidate approach*: combines the positional and candidate gene approaches. Candidate region identified, usually by linkage. Genes

known to map to this region are then considered as candidates. The positional candidate approach is increasingly the method used to clone genes (Karayiorgou and Gogos, 2006). Examples include Marfan syndrome and DFNA13.

2.1.1 Physical Mapping of the Candidate Gene Interval

Once a gene has been localized to a small region of a chromosome, the task of isolating it begins. Currently this entails the construction of a physical map across the chromosomal region carrying the gene, the identification of all genes within that region and gene characterization/mutation analysis to associate a particular gene to a trait or characteristic. Physical map construction is the ultimate step in refining the region of interest and requires the construction of a "contig" overlapping cloned DNA fragments which spans the region of interest. In many cases, this will initially involve using very large clones such as Yeast Artificial Chromosome (YACs, up to two mega bases in size), Bacterial, or Phage P1 artificial chromosomes (BACs, PACs respectively, up to several hundred kilobases in size).

The two current approaches to high-resolution physical mapping are termed "top-down" (producing a macrorestriction map) and "bottom-up" (resulting in a contig map). With either strategy, the maps represent ordered sets of DNA fragments that are generated by cutting genomic DNA with restriction enzymes. The fragments are then amplified by cloning or by polymerase chain reaction (PCR) methods. Electrophoretic techniques are used to separate the fragments according to size into different bands, which can be visualized by direct DNA staining or by hybridization with DNA probes of interest.

A number of strategies can be used to reconstruct the original order of the DNA fragments in the genome. Many approaches make use of the ability of single strands of DNA and/or RNA to hybridize—to form double-stranded segments by hydrogen bonding between complementary bases. The extent of sequence positional homology between the two strands can be inferred from the length of the hybridized double-stranded segment. Physical mapping uses restriction data to determine which fragments have a specific sequence (fingerprint) in common and, therefore, overlap.

The bottom-up approach involves cutting the chromosome into small pieces, each of which is cloned, ordered, and the ordered fragments form the contiguous DNA blocks (contigs). Currently, the resulting "library" of clones varies in size from 10,000 bp to 1 Mb. An advantage of this approach is the accessibility of these stable clones to other researchers. The order of the clones constructed can be verified by FISH, which localizes cosmids to the specific regions within chromosomal bands.

Contig maps, thus, consist of a linked library of small overlapping clones representing a complete chromosomal segment. While useful for finding genes localized to a small area (under 2 Mb), contig maps are difficult to extend over large stretches of a chromosome because all regions are not easily clonable. DNA probe techniques can be used to fill in the gaps, but they are time- consuming.

Technological improvements now make possible the cloning of large DNA pieces, using artificially constructed chromosome vectors that carry human DNA fragments as large as 1 Mb. These vectors are maintained in yeast cells as artificial chromosomes (YACs). (For more explanation, see DNA Amplification.) Before YACs were developed, the largest cloning vectors (cosmids) carried inserts of only 20–40 kb. YAC methodology drastically reduces the number of clones to be ordered; many YACs span entire human genes. A more detailed map of a large YAC-insert can be produced by subcloning, a process in which fragments of the original insert are cloned into smaller-insert vectors. Because some YAC regions are unstable, large-capacity bacterial vectors (i.e. those that can accommodate large inserts) are also being developed.

2.1.2 Isolation and Analysis of Candidate Genes

The final stage of the gene discovery process, identifying causative genes within the candidate interval generally relies on knowledge of the biology of the disease being studied. Various methods including cDNA library screening, cDNA selection, CpG island identification, exon trapping, sequence analysis are used. Exon trapping is a special technique used to search for exons (protein–encoding sequences) in genomic clones. This involves construction of expression cloning vectors containing DNA sequences that are used to transfect a modified cell line. The inserted DNA will be transcribed into RNA and undergo splicing as normal. If the inserted DNA results in an abnormal-sized splice product, it may indicate that this DNA is likely to contain a coding sequence or gene.

In cDNA selection (complementary DNA), DNA is synthesized to complement the bases in a strand of mRNA. This reaction requires the action of reverse transcriptase (an RNA dependent DNA polymerase). The cDNA is representative of the exons or parts of a gene that are expressed to produce a protein in a cell as it is synthesized from mRNA that has had any introns spliced out. Accumulation of evidence that the candidate gene is involved in the disease will be based on:

- Appropriate expression pattern
- Homology to a gene implicated in an animal model of the disease or to a human gene with a similar disease phenotype
- Presence of mutations, which segregate with the disease.

Genes within the candidate interval will be prioritized for further study based on expression pattern and/or homologies to known genes. If a link between these characteristics and the biology of the disease exists, the gene becomes a stronger candidate. It is possible, however, that the disease associated gene(s) will have no homology to any known gene and, thus, no proposed function. In this case, any gene within the interval becomes a candidate for disease association. One way of assessing candidate genes is to look for differences in a gene between affected and unaffected individuals. Polymorphisms (the occurrence in a population (or among populations) of several phenotypic forms associated with alleles of a gene or homologs of one chromosome) within genes are examined to determine if they are found to preferentially associate with the affected population (TDT analysis). This approach can be used for genes with no known homologies as well as for previously identified candidate genes, and may detect "protective" or "susceptible" alleles. Polymorphisms may be identified from existing databases, but in many cases, it will be necessary to establish new polymorphisms within the candidate genes. This is done most commonly by restriction fragment length polymorphism (RFLP) or single-strand conformation polymorphism (SSCP) analysis, or by direct sequencing of DNA from affected and unaffected individuals.

Genes that emerge as strong candidates from the above analysis will be subject to mutation analysis to identify any differences, which may encode the actual etiological mutation. Having identified a susceptibility gene, a range of initial analyses can be undertaken, including gene expression analysis by northern blotting and in situ hybridization, antibody production and protein expression studies, transgenic/knockout mice, structural modeling, mutagenesis, and in vitro biochemical analyses. Such investigations are aimed at producing a range of molecular tools appropriate for probing the function of the gene product and examining its role in the generation of pathology.

2.1.3 Genome Mapping

Genomic maps serve as a scaffold for orienting sequence information. A few years ago, a researcher wanting to localize a gene, or nucleotide sequence, was forced to manually map the genomic region of interest, a time-consuming and often meticulous process. Today, thanks to new technologies and the influx of sequence data, a number of high-quality, genome-wide maps are available to the scientific community for use in their research, Human genome Database "GDB" (http://gdbwww.gdb.org/ gdbhome.html), Ensemble (http://www.ensembl.org/index.html), EMBL (http://www.ebi.ac.uk/embl/), NCBI, DDBJ.

Table 8.1 DNA or Oligonucleotide Microarray analysis softwares and links

Source Details	URL
General Links	http://ihome.cuhk.edu.hk/~b400559/array.html
CAMDA	http://www.camda.duke.edu/
BioConductor	http://www.bioconductor.org/
NetAffx	http://www.affymetrix.com/analysis/index.affx
GEO	http://www.ncbi.nlm.nih.gov/geo/
ArrayExpress	http://www.ebi.ac.uk/arrayexpress/
HuGEIndex	http://zlab.bu.edu/HugeIndex/welcome.htm
dChip	http://www.dchip.org
GENE@WORK	http://www.research.ibm.com/FunGen/FGDownloads.htm

Computerized maps make gene hunting faster, cheaper, and more practical for almost any scientist. In a nutshell, scientists would first use a genetic map to assign a gene to a relatively small area of a chromosome. They would then use a physical map to examine the region of interest close up, to determine a gene's precise location. In light of these advances, a researcher's burden has shifted from mapping a genome or genomic region of interest to navigating a vast number of web sites and databases (Table 8.1).

3. Microarray Experiments

Simultaneous measurement of several gene expressions has been made possible with the advent of photolithographic methods and robotic spotters. Advanced technologies are being used to coat glass or silica plate with synthesized DNA probes in a specific array patterns. These arrays feature regions or spots of DNA probes that measure a few micrometers for specific genes or mRNAs. A sample containing complementary strands of nucleic acids that are tagged to fluorescent label are hybridized to the microarray and this results in the detection of fluorescent intensity. The raw intensity is scaled and normalized to obtain quantitative information about the abundance of specific mRNAs or DNA. Using replicate samples from control and treatment cases one can obtain relative abundance of gene products. There are two types of microarrays based on the type or size of the probe: (1) probe cDNA spotted microarray that has probes >500 base long, (2) High density oligonucleotide arrays (e.g. GeneChip from Affymetrix Inc) with multiple short probes of 20–80 base. Microarray experiments involve (1) Sample preparation and labeling, (2) Array Hybridization (3) Readout and higher-level data-analysis. Each chip may feature probes for several genes and recently higher density arrays have been released for whole genome (e.g. Affymetrix U133 2.0 arrays for Human). Typically expression

arrays are used for measuring mRNA abundance, while sequencing or genome hybridization type arrays are used for detecting single-nucleotide polymorphism (SNPs) or other genomic instability. Expression arrays are widely used for understanding genetic regulation underlying a disease or genetic response to external stimulation from chemical or biological agents. Low-level analysis of expression array involves assigning expression or intensity index for the entire set of genes present on the array. In spotted array, this index is a ratio between two different fluorescent probe intensity (Cys 3'/Cys 5') corresponding to control and test samples. In GeneChip (Affymetrix), individual samples are hybridized to a specific oligo probe set; hence, a separate intensity index is assigned for each oligo probe. Affymetrix chip design uses a perfect match (PM) probe of 25-mer (or bases) and a mismatch (MM) probe (13th mismatch base) for a single region of a gene. Several such region or probe set are used for querying expression. The raw intensity value is the average of PM–MM over several probe pairs. The average intensity of each is set to a fixed value and a scaling factor is calculated to obtain an expression intensity index. Several software tools can assign expression index based on different statistical models which systematically corrects for backgrounds, adjusts, or normalizes expression values to facilitate comparison across different chips. Affymetrix software MAS5.0 uses average of intensity differences between Perfect and Mismatch probes within a single chip to assign expression index. Softwares like dCHIP or RMA uses probe intensity distribution across multiple or replicate chip to extract background and assign intensity index that can provide excellent sensitivity compared to MAS5.0. The open source BioConductor package developed in *R* provides implementation of different algorithms and also additional tools for low-level analysis. Microarray low-level analysis produces a statistically significant list of genes that may be regulated. Several concerns do exist in the field relating to the false-discovery rates (FDR), adequacy of sample sizes, and reproducibility of expression indexes. A false-discovery rate is the expected percentage of the final list of significant genes as measured by statistical criteria that may not be truly significant. Most FDR discovery tools use some kind of corrections for multiple testing or permutations of samples/genes labels to estimate the significance of multiple testing. Several FDR-based multiple testing procedures are widely used in the field of microarray like Benjamini and Hochberg or Storey (Storey 2002; Jung and Jang, 2006). One can use Significance Analysis of Microarray (SAM) to estimate FDR. Higher level analysis of microarray data involve gene-clustering, extraction of functional patterns, and identification of networks or pathways that are significantly altered (Tusher et al., 2001). Two-way clustering of genes and samples are generally performed using a normalized expression index across genes. The

distance metric generally used is the inverse of Pearson-correlation. Gene clustering can be performed using dCHIP, Affymetrix DataMinerTool (DMT), TreeView, GeneCluster (http://www.broad.mit.edu/cancer/software/software.html), or in BioConductor. Both supervised (e.g. K-mean clustering, Pearson correlation analysis, nearest shrunken centroids analysis, etc.) and unsupervised clustering (Hierarchical, Self-Organizing Maps) can be performed to group related gene expression across different chips. If a set of functionally related gene is statistically enriched under a node, one can infer biological significance of such functions (e.g. DNA repair enzymes). Genes are annotated and grouped according to function using standard ontologies in publicly available GO databases. Clustering based on gene ontology is applied to identify significantly altered biological functions (Please refer to Section 3.2).

3.1 Microarray Databases

Microarray data can be submitted to public repositories like Gene-Expression Omnibus (GEO) (http://www.ncbi.nlm.nih.gov/geo/) and ArrayExpress (http://www.ebi.ac.uk/arrayexpress/) that also contain other tools to perform wide list of analysis. GEO is a MIAME (Minimum Information About a Microarray Experiment)–compliant microarray data repository developed at NCBI. As of March 2007, GEO contains information about 133152 samples. It is the largest public repository of microarray data. User friendly query interface is available that can be used to retrieve raw and processed files for specific samples, platform, or species. GEO also provides information about gene profiles where a user can query a gene of interest to see differential expression pattern within an experiment set. ArrayExpress is another MIAME–complaint public database that curates microarray data. It also provides query interface for gene expression profile.

3.2 Gene Annotations, Ontology, and Pathway Databases

Once a list of genes with interesting expression profile is identified, the last step in microarray experiment is to infer the result in terms of biological context. If several members of a biological pathway or function are significantly regulated, then a hypothesis can be developed that can be tested with new experiments. A repository containing annotated genes in terms of their function, cellular location, and the pathway involved is used to identify biological process. Since common biological function can be represented by different keywords, a scoring under specific functional keyword requires declarative representation using a controlled vocabulary. Ontologies specify uniform controlled vocabulary from a specific discipline. GeneOntology

(http://www.geneontology.org/) contains a hierarchical arrangement of gene members based on controlled vocabulary to describe their attributes. It is widely used for scoring and identifying process specific regulation (e.g. Copper Ion homeostasis) from the gene annotation. A gene can be annotated under several terms. The terms are arranged hierarchically based on their relationship. The terms are further classified under different ontologies: biological process, cellular component, and molecular function. There are several ontology projects that are of interest to biomedical community (http://obo.sourceforge.net/). Pathway databases represent a defined set of gene products (proteins) that forms a network to achieve metabolism (e.g. Glycolysis), cell-signaling (e.g. VEGF pathway), or a disease (e.g. Alzheimer's pathway). A biological network is formed by a protein–protein interaction, protein–lipid, protein–RNA, or protein–DNA interactions. Commerical databases like Ingenuity (http://www.ingenuity.com) use literature/experimental interactions of proteins to create insilico networks. They also contain well-defined pathways. A list of protein or gene id can be scored to obtain significantly regulated networks or pathways. The **D**atabase for **A**nnotation, **V**isualization, and **I**ntegrated **D**iscovery (**DAVID**) at (http://david.abcc.ncifcrf.gov/home.jsp) is public tool that can be used for scoring significantly altered function or to identify biological process. Kyoto Encyclopedia of Genes and Genomes (KEGG) (http://www.genome.jp/kegg/) is a public repository of biological pathways that also contains tools like KegArray, which can be used in microarray data analysis.

References

Ait-Ghezala, G, Mathura, VS, et al. (2005) Genomic regulation after CD40 stimulation in microglia: relevance to Alzheimer's disease. *Brain Res Mol Brain Res* **140**(1–2), 73–85.

Clos, J and Choudhury, K (2006) Functional cloning as a means to identify Leishmania genes involved in drug resistance. *Mini Rev Med Chem* **6**(2), 123–9.

Jung, SH and Jang, W (2006) How accurately can we control the FDR in analyzing microarray data? *Bioinformatics* **22**, 1730–1736.

Karayiorgou, M and Gogos, JA (2006) Schizophrenia genetics: uncovering positional candidate genes. *Eur J Hum Genet* **14**(5), 512–9.

McKusick, VA (1997) Genomics: structural and functional studies of genomes. *Genomics* **45**, 244–249.

O'Brien, SJ, Weinberg, J, et al. (1997) Comparative genomics: lessons from cats. *Trends Genet.* **13**, 393–399.

Schimenti, J and Bucan, M (1998) Functional genomics in the mouse: phenotype-based mutagenesis screens. *Genome Res* **8**(7), 698–710.

Storey (2002) A direct approach to false discovery rates. *J Roy Stat Soc Ser B* **64**, 479–498.

Tusher, VG, Tibshirani, R, et al. (2001) Significance analysis of microarrays applied to the ionizing radiation response. *Proc Natl Acad Sci U S A* **98**(9), 5116–21.

Chapter 9

Introduction to Proteomics

Fai Poon and Venkatarajan S. Mathura

Roskamp Institute, 2040 Whitfield Avenue, Sarasota, Florida 34243

Abstract: Proteome is defined as the total set of proteins expressed in a given cell or biological sample at a given time. The study of proteome is termed "proteomics". Proteomics research includes the separation, identification, qualitative, quantitative, and functional characterization of the entire protein profile of a given cell, tissue, and/or organism. In this chapter, we describe the processes of commonly used proteomics techniques like gel-based separation and mass-spectrometry.

Key words: 2D-gel electrophoresis, Mass-spectrometer, Post-translational modification

1. Introduction

Proteins govern most biological processes and functions. Recently, much attention has been paid to study the entire protein sets available in a cell. While genomics attempts to reveal genes and mRNA content, proteomics studies the protein complement. Before proteomics, most study on protein is limited by the availability of the antibody (Lauderback et al., 2001). Although this one–at–a–time method is commonly used and accepted still, it is impractical to use this method to study enormous proteome. The study of proteome is termed "proteomics". Proteomics research includes the separation, identification, qualitative, quantitative, and functional characterization of the entire protein profile of a given cell, tissue, and/or organism. Studying proteome also includes the profiling of isoforms, splice variants, mutants, post-translational modifications, and protein–protein interactions (Dove, 1999; Schoneich, 2003).

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_9,

2. Sample Preparation

Proteomics research involves several steps: (1) sample preparation (2) peptide/protein separation, (3) peptide/protein identification. The samples for proteomics research usually contain high concentration of ions due to lysis buffer usage while extracting proteins from the cell. The high level of ions in the buffer can cause variation of voltage and current during isoelectric focusing (IEF), thereby preventing successful isoelectric separation of proteins of interest. This phenomenon usually manifested by horizontal smearing of the protein spot during electrophoretic separation. This can be avoided by using chemicals, such as trichloroacetic acid (TCA) to precipitate the protein and subsequently use organic solvents to wash the pellet. Commercial spin ion-removal columns (Sigma and Pierce) are also available to remove ion from samples prior to IEF. Once the protein samples are extracted and purified, it should be separated before identification step. Separation reduces the complexity of protein to be analyzed. Most widely used methods for separating proteins are 2D gel electrophoresis and liquid chromatography.

3. Two-Dimensional (2D) Gel Electrophoresis

2D gel electrophoresis is commonly used to separate a mixture of proteins into single detectable protein spots. The 2D separation of proteins on gel is usually achieved according to their isoelectric point and molecular weight. In the first dimension, proteins are separated according to their isoelectric point by IEF on an immobile pH gradient strip. The resulting strips are then treated with dithiothreitol and iodacetamide to avoid cysteine–cysteine interaction, which will decrease the resolution of the second step of separation. In the second dimension, proteins are separated according to their molecular migration rate as determined by their molecular weight. Finally, the gel is stained.

The resulting 2D map allows comparison within and between groups of samples for statistical analysis. The advantages of the 2D gel electrophoresis are its consistency and high resolution. However, some caveats of this technique are still present; for example, the insolubility of membrane proteins is still a main obstacle for 2D-electrophorsis. The ionic detergents used for solubilization of membrane proteins can interfere with the focusing process. Additionally, the mass range and the detection limits also represent technical limitations of 2D-electrophoresis method. Moreover, identification of low abundant protein in a sample is usually limited by the sensitivity of the stain used for protein detection (Soreghan et al., 2003).

3.1 Image Analysis and Statistical Analysis

The proteins that are separated on 2D gels traditionally are stained by classical methods, including Coomassie blue and silver staining. However, these detection methods remain problematic due to low sensitivity (for Commassie) or poor reproducibility and dynamic range (for silver). The recent development of fluorescent dyes, namely SYPRO™ Ruby, overcame these problems with its sensitive (1–2 ng) detection limits and linear dynamic range over three orders of magnitude (Molloy and Witzmann, 2002). The resulting 2D map is then analyzed by software designed for image analysis, which allows gel–to–gel comparison. These software usually generate a large amount of data accumulated from multiple 2D gels. Widely used 2D gel analysis software are like PDQuest (Biorad), ImageMaster 2D/Melanie (http://expasy.org/melanie), etc. Some of these software were evaluated in a recent survey (Righetti et al., 2004). These software enable investigator in matching and analysis of protein spots among differential gels and blots. The principle involved in the intensity measurement of protein spots in 2D electrophoresis is similar to those of densitometric measurement. After completion of spot matching, the normalized intensity of each protein spot from individual gels is compared between groups using statistical analysis. Although the software use raw-image patterns for automatic alignment, additional methods like neighboring spot patterns or landmark matching, automatic image warping, and hands-on or manual processing is still necessary for accurate results.

3.2 In-Gel Digestion and Mass Spectrometry

Following 2D image analysis, the protein spot of interest is excised and treated with ammonium bicarbonate and acetonitrile to remove detergents that may interfere with the protease activity step of in-gel digestion. The excised spots are then in-gel-digested with a protease (trypsin is commonly used) in an optimal buffer for its activity. The digested peptides are then easily eluted from the gel to undergo mass spectrometry analysis. In-gel digestion not only reduces the mass of a protein into small peptides ideal for mass spectrometry, but also forms a collection of proteolytically sequence-specific peptides that enables the identification of the protein.

4. Mass Spectrometry

Mass spectrometry is a technique where the mass of an ion is measured for the characterization of the molecule of interest. Mass spectrometry is

composed of a sample inlet, an ionization source, a mass analyzer, and a detector:

- Sample Inlet

  Sample inlet is where the sample is delivered to the mass spectrometry. The sample inlet could be a liquid chromatography (LC), gas chromatography, or a capillary electrophoresis instrument. In proteomics application, the most commonly used sample inlet is an LC for its solvent compatibility with the ionization source.

- Ionization source

  In mass spectrometry, the molecules have to be converted into their ionized form to be measured. There are many kinds of ionization source that are used for the ionization of different compounds. The commonly used ionization sources for proteomics applications are electrospray ionization (ESI) and matrix assisted laser disorption ionization (MALDI). ESI and MALDI are soft ionization techniques and hence there is less fragmentation of the peptides or proteins during ionization. However, peptides or proteins can be further fragmented by collision with inert gases to get sequence information. ESI is commonly used with different separation techniques including reverse phase liquid chromatography, which is used for separation of peptides based on their hydrophobicity.

- Mass Analyzer

  The ions formed in the ionization source have to be separated depending on their mass to charge values in order to be detected. The commonly used mass analyzers are ion trap, quadrupole mass filter, time–of–flight and hybrid forms. The ion trap and quadrupole mass analyzers separate ions by applying rf and dc voltages that isolate specific mass to charge value of the ion of interest while ejecting the other ions. The time of flight separates the ions by the time it takes the ions to move from the ionization source to the detector, which is inversely proportional to the mass to charge value of the ions.

4.1 Mass Spectrometry in Proteomics

Mass spectrometry facilitates rapid identification and characterization of thousands of protein in a sample. The traditional ionization fragmentation of peptides do not provide accurate peptide mass. However, development of two Nobel prize-winning ionization methods, MALDI (matrix assisted laser desorption/ionixation) and ESI (electrospray ionization), enabled the ionization of large biological macromolecules without fragmentation

required for the identification of proteins (Beavis and Chait, 1989). The peaks detected by mass spectrometer results in a mass spectrum (mass to charge *m/z* in x-axis and ion count or intensity in the y-axis) that represent the peptide ions mass, which can be used for protein identification.

In MALDI, the peptide samples are mixed to an acidic matrix and condensed on a metal plate. This plate is subjected to laser radiation. The peptides are incorporated into the crystal lattice of the matrix during the condensation process. Various compounds are used as the matrices for laser absorption. One of the widely used matrices for peptides is α-cyano-4-hyroxy cinnamic acid, which provides high sensitivity, and negligible matrix adducts formation during the laser absorption. When the high-energy laser is strikes the matrix, the peptides along with the matrix particles are vaporized. In order to ensure the sublimation of the matrix-peptides, high vacuum are generally applied during this process. The positive ions of the peptides are formed in gas phase due to the acidic nature of the matrix. The ions are then accelerated into the mass analyzer. Since MALDI is a pulsed ionization technique, it is generally couple with TOF (time-of-flight) mass analyzer.

In ESI, the peptide in the solution is sprayed at atmospheric pressure through an outlet with high electric-potential difference between the outlet and the mass spectrometer. This potential difference generates a repulsive coulombic force among like charge droplets that extends to form a cone (Taylor cone) from the outlet, causing the solution to disperse into fine droplets. The solvent continuously evaporates while the charges of the droplets remain constant. Droplet fission occurs when the coulumbic repulsion exceed the surface tension of the droplets. This process continues until nanometer sized droplets are produced to form a single peptide ion for the mass spectrometer. The charges on the single peptide are statistically distributed over the peptide. Multiply charged ions are possible during this process. ESI offer significant advantages for the analysis of peptides with large molecular weight.

5. Bioinformatics Applications for Identification

Since the peaks of the resulting mass spectrum represent the peptide ion's mass in the sample of interest, the peaks can be back correlated to the mass of the peptides produced by protease digestion from a larger protein. Databases are available for theoretical digests of all known proteins. Matching the peptide-mass data obtained from sample of interest to theoretically digested protein database can be used to successfully identify unknown proteins. This process, peptide mass fingerprint (PMF) matching, must account for several factors such as molecular weight, pI, and the

probability of a similar peptide occurrence in the whole database, for the identification of a protein. Many software search engines can perform this matching process automatically. Such engines output a probability score for each theoretically digested protein indicating the certainty of the identification. The threshold score, which indicates whether the experimental mass spectrum significantly matches the *in silico* digested protein spectrum, is calculated by mathematical algorithms specific to each search. Although false identification is possible, it can be avoided by applying molecular weight and pI of the protein spot obtained from the 2D map as additional filter. Other means of validation and confirmation are usually necessary for correct protein identifications. PMF obtained after processing raw signals can be searched using several online or standalone softwares. Most widely used search engine is MASCOT (http://www.matrixscience.com). Several protein sequence databases can be queried (MSDB, NCBI nr, SwissProt, and dbEST) by MASCOT. The search produces a list of protein hits with scores above a significant threshold. The protein score is a logarithmically scaled probability of the observed hit to be a random event in a given database (–10*log(P)). Several fixed or variable modifications can be selected during the search. Additional criteria of protein mass and pI can be specified to narrow protein hit. Apart from searching for protein hit using PMF, MASCOT can also be used for MS/MS search using datafile format from several vendors. MS-Fit is another web tool from UCSF that can search for protein hits using PMF. It is part of useful mass-spectrum search tools called ProteinProspector (http://prospector.ucsf.edu/). Using tandem mass-spectrometer one can collect MS/MS that can be used to search for peptide sequence or perform *de novo* sequencing using different ions (e.g. X, Y, Z, a, b, c). SEQUEST, orginally developed at University of Washington, is a software tool that can be used for processing/searching peptide sequences using MS/MS. To understand more about peptide identification and fragmentation pattern, please visit http://www.proteomesoftware.com/ Proteome_software_pro_ protein_id.html. GPMDB is a database of MS/MS spectra for different peptides that can be used for assignment validation. PeptideAtlas is a public database of peptides observed during tandem mass spectrometry (http://www.peptideatlas.org). Currently, it contains information about peptides from Yeast and Human. PRoteomics IDentification Database (PRIDE) is a public repository of peptide and protein identified using mass spectrometry. One can compare protein/peptide hits across experiments. For each experiment, it provides details about the experiment, sample information, peptides or protein identified, type of instrument, and its settings.

6. Conclusion

The technology to perform proteomics has improved rapidly over recent years. Many high- throughput methods and software are being developed in order to improve sensitivity and detection limit of 2D gel electrophoresis, and new algorithms to search or process mass-spectrum that will enable researchers to detect low abundant proteins. When such techniques mature, a large body of information will become available to better understand diseases and to develop biomarkers for diagnosis. Moreover, information from proteomic experiments may lead to new hypotheses that can bring an in-depth understanding of pathogenesis and develop therapy for many diseases. Collaboration among physicians, biological chemists, and software engineers will be necessary to accomplish this.

References

Beavis, RC and Chait, BT (1989) Cinnamic acid derivatives as matrices for ultraviolet laser desorption mass spectrometry of proteins. *Rapid Commun Mass Spectrom* 3 (12)(432–5).

Dove, A (1999) Proteomics: Translating Genomics into Products. *Nat Biotechnol* 17 (3) (233–6).

Lauderback, CM, Hackett, JM, et al. (2001) The glial glutamate transporter, GLT -1, is oxidatively modified by 4-hydroxy-2-nonenal in the Alzheimer's disease brain: the role of Abetal 1 - 42. *J Neurochem* 78 (2)(413–6).

Molloy, MP and Witzmann, FA (2002) Proteomics: technologies and applications. *Brief funct Genomic Proteomic* 1 (1)(23–39).

Righetti, PG, Castagna et al. (2004) Critical survey of quantitative proeomics in two dimensional electrophoretic approaches. *J Chromatogr A* 1051 (1–2)(3–17 Review).

Schoneich, C (2003) Proteomics in gerontological Research. *Exp Gerontol* 38 (5)(473–81).

Soreghan, BA, Yang et al. (2003) High – throughput proteomic-based identification of oxidatively induced protein carbonylation in mouse brain. *Pharm Res* 20 (11)(1713–20).

Chapter 10

Biomedical Literature Mining

Chaolin Zhang[1,2] and Michael Q. Zhang[1]

[1]*Cold Spring Harbor Laboratory, 1 Bungtwon Road, Cold Spring Harbor, NY, 11724*
[2]*Department of Biomedical Engineering, State University of Newyork at Stony Brook, NY 11794*

Abstract: A hurdle of large-scale genomic studies is to incorporate existing knowledge from published literature. This is accomplished by human experts but suffers from the heavy labor and the difficulty to keep knowledge up to date. Biomedical literature mining provides a potential solution to extracting and integrating useful information from literature automatically, which can lead to new discoveries.

Key words: Literature mining, Automatic knowledge extraction, Gene interaction, Network

1. Introduction

In this post-genomic era, the focus of genomic research has been shifting from sequencing to annotating gene functions (Watson, 1990; Venter et al., 2001; Cavalli-Sforza, 2005). High-throughput experimental technologies developed in the last decade now permit us to assay the whole genome in various aspects (Lockhart et al., 1996; Emili and Cagney, 2000; Impey et al., 2004; Kim et al., 2005; Yuan et al., 2005). These progresses have promoted the study of "systems biology", which aims to deciphergener regulatory networks at the genomic scale. (Aderem, 2005; Kirschner, 2005; Liu, ET 2005). The characteristic of these genome-wide studies is the huge amount of data. New scientific discoveries depend largely on the incorporation of existing knowledge. Comprehensive understanding of scattered knowledge seemingly unrelated can shed new light to future research.

Every piece of knowledge, which marked important discoveries in the past, is presumably documented in published literature in the form of unstructured or semi-structured text. With the exponential growth of publications, however, manually tracking literature is beyond the ability of

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_10,

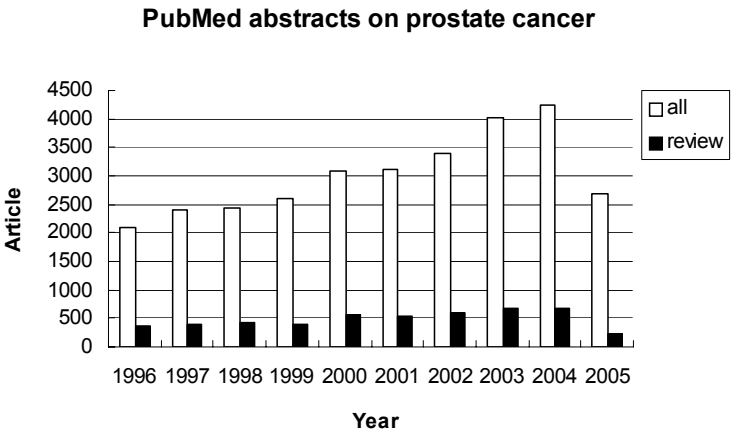


Figure 10.1 PubMed abstracts searched by keywords "prostate cancer," limiting the date of publication. The number of reviews was obtained by limiting the type of article.

any individual. For example, the most widely used biomedical literature database, NCBI's PubMed, has included over 15 million abstracts back to the 1950s. Even in a particular field like "prostate cancer", there have been more than 50,000 abstracts, including 7,000 reviews, and these numbers increase at the rate of 4000 abstracts and 600 reviews each year (Figure 10.1).

To facilitate effective explorations and reuse of existing knowledge, NCBI's GenBank associates a list of references to each sequence in the feature table. Various databases collect existing knowledge from literature and relate it to genes, for example, GeneRIF, GeneCards (Safran et al., 2002), and GeneLynx (Lenhard et al., 2001); proteins, for example, SwissProt (Boeckmann et al., 2003); molecular interactions, for example, DIP (Salwinski et al., 2004), BIND (Bader et al., 2001), KEGG (Kanehisa and Goto, 2000); and diseases, for example, OMIM (Hamosh et al., 2002). Other approaches, such as Gene Ontology (GO), create controlled vocabularies to annotate genes (Ashburner et al., 2000). However, the manual curation process is very laborious and, thus, difficult to keep up to date. Furthermore, these resources do not provide tools which can be readily used to interpret data generated by high-throughput experiments.

On the other hand, biomedical literature mining (BioLM) provides a potential solution by extracting and organizing useful information from online literature in an automatic and timely manner. In the past few years, significant efforts have been made to apply this technology to a variety of tasks, such as automatic gene/protein name recognition (Fukuda et al., 1998; Collier et al., 2000; Shen et al., 2003; Zhou et al., 2004; Shi and Campagne, 2005), abbreviation dictionaries of biomedical terms (Adar, 2004; Wren et al., 2005), reported gene locations on chromosomes(Leek, 1997), improving

homology searches (Chang et al., 2001), gene group coherence (Raychaudhuri et al., 2003), gene or protein interaction networks (Jenssen et al., 2001; Stephens et al., 2001; Hoffmann and Valencia, 2004), and the relation between genes and disease (Stephens et al., 2001; Matsunaga and Muramatsu, 2005).

In this chapter, we introduce major issues in BioLM, with an emphasis on the automatic discovery of gene functions and interactions. Interested users are encouraged to read reviews by (Hirschman et al., 2002) for technical challenges of BioLM, (Shatkay and Feldman, 2003) for connection of BioLM with general literature mining, and (Krallinger and Valencia, 2005) for online resources.

2. Literature Sources for Mining

Most journals are now available online and provide the opportunity to perform full text analysis [see e.g. ref (Wilkinson and Huberman, 2004)]. However, the abstracts of biomedical articles are more readily available from NCBI's PubMed, the largest biomedical literature database. Currently, PubMed has included over 15 million abstracts. While it is true that full text articles provide more complete information, we can expect that the most important results of an article are usually summarized in the abstracts (Schuemie et al., 2004).

Other useful sources are the semi-structured annotation information in GenBank feature tables as well as the meta-databases mentioned in the introduction. These databases usually have manually curated information from literature along with references linking genes to previous studies. A lot of tools are available to analyze the enrichment of GO terms in a list of genes (http://www.geneontology.org/GO.tools.shtml). Recently, (Rubinstein and Simon, 2005) used GeneRIF as well as PubMed abstracts to annotate microarray results. The benefit of using manual annotations lies in the accuracy. However, they do not guarantee the completeness of coverage. Therefore, a lot of literature mining systems use PubMed abstracts as literature sources.

Most tasks focus on literature of a particular subject in order to reduce computational costs and noise (Zhang and Li, 2004). The step to retrieve a collection of literature relevant to a particular field (e.g. prostate cancer) is information retrieval (IR). IR can be based on keywords (Boolean) or example. In a keywords-based method, a query is a list of terms related to the field of a study. All documents with the query terms are then returned.

In contrast, the example based method works by asking users to provide an example article of their interest. Then the system returns articles similar to the example. The main point here is to measure the similarity of two articles. This is done by representing an article with a vector, $\mathbf{w}(k)=[w_1(k),w_2(k),...,w_m(k)]^T$, where $k=1,2,...,D$ is the index of each article and each element $w_i(k)$, $i=1,2,...,D$ is the weighted occurrence of word i in article k. For example, in the Term Frequency, Inverse Document Frequency (TFIDF) method,

$$w_i(k)=TF_i(k)\bullet IDF(i),\ IDFi=\log(D/DF_i)$$

where $TF_i(k)$ is the number of occurrences of word i in article k and DF_i is the number of articles containing word i. This measure is straightforward in that uncommon words particularly enriched in an article are good representatives of the article and, thus, assigned greater weights. With this vector representation, the similarity of two articles j and k can be measured by the cosine of the angle between the two vectors:

$$S(k,l)=\mathbf{w}(k)\bullet\mathbf{w}(l)/(|\mathbf{w}(k)|\bullet|\mathbf{w}(l)|)$$

PubMed Entrez is the most widely used IR system to get interested articles. It allows users to use both Boolean and example search. Besides the web interface, it also provides a set of tools called E-Utilities (http://eutils.ncbi.nlm.nih.gov/entrez/query/static/eutils_help.html) for users to retrieve documents in a batch mode.

3. Recognition of Biological Terms

After a collection of literature is prepared, the next step is to index biological terms such as gene and protein names. In some other applications, keywords related to molecular interactions, biological process, disease, etc., are also indexed (Stephens et al., 2001; Temkin and Gilder, 2003; Rubinstein and Simon, 2005). This task is not trivial because both synonymy (multiple words having same meaning) and polysemy (words having multiple meanings) are very common for biological terms. Synonyms should be appropriately mapped to a unique identifier while ambiguities should be removed. The noise introduced in this step can greatly affect the accuracy of later steps. Since gene names and protein names are often used equivalently, the disambiguity of the two is extremely difficult and depends largely on context. Therefore, we do not distinguish them deliberately in our discussion.

3.1 Gene/Protein Name Recognition

While great success has been achieved in name entities recognition of human names and addresses from news articles (~95% in accuracy), the recognition of gene or protein names seems to be more difficult (Fukuda et al., 1998). The gene nomenclature has not been standardized. Once names and symbols of new genes were published in journal articles and got fixed, they will not be affected by later corrections. Therefore, the name space representing a gene can become quite large. A gene usually has a standard symbol, a full name, and several non-standard symbols or aliases. For example, in the dictionary of human gene names by Human Gene Nomenclature Committee (HGNC), a gene has three symbols or names in average, including one standard symbol. In some cases, the commonly used name is not necessarily the standard one (e.g. p53 vs. TP53). Obviously, it is fundamental to map synonyms of a gene to a unique identifier, such as the HGNC approved symbol.

As a practical approach, the gene name dictionaries provided by genomic databases can be readily used to match gene names in the text. A summary of commonly used dictionaries is listed in Table 10.1.

Complementary to using existing gene name dictionaries, studies of *ab initio* gene or protein name recognition have been motivated by two reasons. First, the dictionaries do not include recently discovered genes. This is common because of the fast development of molecular biology and genetics. Second, a lot of gene names or symbols have several variations. Not all of them are included in dictionaries. For example, REST (RE1-silencing transcription factor) is also called NRSF, neuron-restrictive silencer factor, neuron-restrictive silencing factor, NRSE-binding factor, etc.

Table 10.1 Databases providing gene name dictionaries

Database	Species	URL	Ref
Entrez Gene /LocusLink	Multiple species	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene	
HGNC	Human	http://www.gene.ucl.ac.uk/nomenclature/	
GDB	Human	http://www.gdb.org/	
OMIM	Human	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=omim	
SwissProt	Multiple species	http://us.expasy.org/sprot/	
SGD	Yeast	http://www.yeastgenome.org/	
Flybase	Fly	http://fbserver.gen.cam.ac.uk:7081	
Wormbase	Worm	http://www.wormbase.org/	
RGD	Rat	http://rgd.mcw.edu/	
MGD	Mouse	http://www.informatics.jax.org/	
DogMap	Dog	http://www.dogmap.ch/	

Ab initio recognition methods generally fall into two categories: rule-based or machine learning-based. Rule-based methods use a set of manually created rules which characterize patterns of gene or protein names (Fukuda et al., 1998). In contrast, machine learning approaches try to learn statistical features from protein names and their context from a training set (Collier et al., 2000; Shen et al., 2003), then use these features to calculate the probability if a new string of text contains gene or protein names. The accuracy of these approaches varies greatly from 20 to 95%, partly because of the lack of benchmark datasets used for training and cross-validation (Fukuda et al., 1998; Collier et al., 2000; Zhou et al., 2004; Cohen et al., 2005).

3.2 Removing Gene/Protein Name Ambiguities

A gene symbol can be the abbreviation of different genes. In the HGNC gene name dictionary, more than 500 symbols have ambiguities. This is even more serious when multiple species are in consideration (Chen et al., 2005). For example, the string '*CAT*' represents different genes in cow, chicken, fly, human, mouse, pig, deer, and sheep.

Baring this in mind, more rigorous recognition strategies must be employed to remove or reduce ambiguities. Simple rules can be very effective here. One can use case sensitive matches for gene symbols, remove all common words and/or terms that are too short, remove terms with abnormally high occurrences, or require the co-occurrence of short symbols and full names (Jenssen et al., 2001; Rubinstein and Simon, 2005). Algorithms for automatic gene/protein name recognition and abbreviation dictionary construction can also be used to filter false positives (Temkin and Gilder, 2003; Wilkinson and Huberman, 2004).

3.3 Collecting Other Keywords

To characterize gene functions or gene relationships, related keywords also have to be indexed. In some cases, these keywords can be submitted by users. For example, two web applications, MILANO and PubMatrix, accept both gene names and keywords from user input, and search pair-wise gene-keywords co-occurrences (Becker et al., 2003; Rubinstein and Simon, 2005). In other cases, (Stephens et al., 2001; Temkin and Gilder, 2003) collected keywords to describe gene interactions. OMIM includes terms related to diseases (Hamosh et al., 2002). Gene Ontology (GO) contains a limited vocabulary related to gene annotations (Ashburner et al., 2000), which are manually linked to genes. MeSH terms are a controlled vocabulary used to summarize PubMed abstracts. The association between these keywords and

genes or abstracts can provide rich information. They can be incorporated into the discovery of gene functional annotation, gene interactions, and other implicit relationships (see next section for more discussion).

4. Mining Biological Relationships

Since the literature covers every aspect of existing discoveries, there is almost no limit of the types of information which can be extracted. Here we focus on using literature mining to build gene interaction networks, extract gene functional annotations, and evaluate functional coherence of gene groups.

4.1 Detecting Gene Interactions by Co-occurrence

Building gene interaction networks from literature was pioneered by the work of Jenssen and Stephens et al. (Jenssen et al., 2001; Stephens et al., 2001). The basic assumption underlying these studies is that two genes cited in the same article must be related in certain aspects. The network integrating scattered relationships can provide a holistic view of the gene network.

(Jenssen et al., 2001) indexed all named human genes in all PubMed abstracts at the time of the work. They found that the co-occurrences of genes can reflect real biological relationships, which can facilitate the interpretation of microarray experiments. By comparison with a manually curated database, they estimated that 50% of gene pairs with co-occurrence have real meaningful biological relationships at a recall of 50%. PubGene is a web tool that allows users to query a gene and retrieve the sub-network, which is displayed graphically (http://www.pubgene.org/).

In contrast, (Stephens et al., 2001) selected only a set of abstracts and genes related to a particular field. The authors also used a more quantitative measure of relations, derived from TFIDF weight,

$$A_{ij}=\sum_k w_i(k)\bullet w_j(k)$$

This measure assigns more weight if two genes are always co-cited in articles and a smaller weight if genes are ubiquitous individually.

Interestingly, the gene co-occurrence network shares very similar properties with other biological and social networks, which are scale free (Jeong et al., 2000). This also suggests the validity of the gene co-occurrence network.

While it is true that a co-cited gene pair often has certain biological relations, these relationships are not necessarily explicit. Therefore, co-occurrence in abstracts alone is often insufficient to define a gene interaction for those databases annotating gene functions. For those tasks, gene interactions should be accurately and explicitly presented.

(Ding et al., 2002) compared the effect of different text units (abstract, sentences, phrases), where co-occurrence is defined, to the performance of gene interaction extraction. They found that the text unit in sentences gives the best balance between precision (85%) and recall (65%). In the study of (Stephens et al., 2001), the authors require the co-occurrence of predefined keywords with two genes in the same sentence.

Further performance improvement may be achieved by deeper lexical analysis (e.g. see ref Temkin and Gilder, 2003). However, this is generally more difficult to implement and more computationally intensive. As far as we know, there is no automatic system which can detect gene interactions with comparable performance of manual annotation. Therefore, some databases use computational programs to do a pre-screening (Zanzoni et al., 2002; Donaldson et al., 2003), before a further examination by expert annotators.

4.2 Inferring Implicit Relationships

In contrast to database annotations, which require accurate extraction of explicit relationship between genes, another direction of study is to identify implicit relationship. Although two genes do not co-occur in the same articles, they might have implicit relationship if they share certain properties, such as neighbor genes (Wren et al., 2004), and annotation terms from GO, MeSH, and OMIM (Jenssen et al., 2001) (Figure 10.2). Of course, this implicit relationship is not necessarily a molecular interaction, but can also be similarity in function, cellular localization, homology, molecular pathway, etc.

The objects served as intermediate to bridge the two genes may vary, but the underlying method is almost the same. The basic idea is to measure the overlap of objects linked with gene A and those linked with gene B. This can be done by calculating the expected overlap simply by chance, which follows a hypergeometric distribution (Ramani et al., 2005). More formally, suppose gene A links with m objects whereas gene B links with n objects, the total number of objects which can potentially link with gene A or B is N.

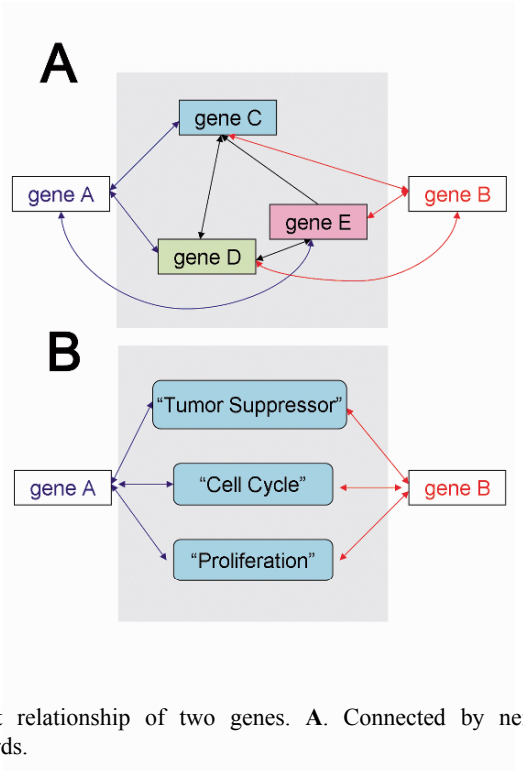


Figure 10.2 Implicit relationship of two genes. **A**. Connected by neighbor genes. **B**. Connected by keywords.

Then the probability that gene A and gene B share l or more links by chance is

$$p(s \geq l) = 1 - \sum_{s=0}^{l-1} p(s \mid m, n, N),$$

Where $$p(s \mid m, n, N) = \begin{cases} \binom{n}{k}\binom{N-n}{m-k} \Big/ \binom{N}{m} & N \geq m + n - k \\ 0 & \text{otherwise} \end{cases}.$$

measures the significance of the implicit relation. When multiple pairs are examined, multiple-test correction can be applied as described previously (Reiner et al., 2003; Storey and Tibshirani, 2003).

4.3 Identifying Sub-networks of Communities

If the literature collection of genes are not specific enough for a specific field, the resulting gene network can be very large and extremely difficult to be manually explored. Algorithms have been developed to identify

potentially important sub-networks or communities (Girvan and Newman, 2002; Wilkinson and Huberman, 2004; Palla et al., 2005).

There is no strict definition of a "community", but generally speaking, a community should have more (or heavier) links between members inside and fewer (or lighter) links with members outside. With a quantitative measure of link weights, we can partition the whole network into smaller sub-networks by selecting a threshold of link weights. All links below the threshold are removed. However, this simple approach can suffer from the intrinsic incompleteness of gene relationship and inaccuracy of the link weights derived from literatures. Further more, gene networks, as well as other biological and social networks, have recursive and overlapping communities in nature, which could not be characterized by simple network partition. Here we introduce a definition of community recently proposed by (Palla et al., 2005), which aims to uncover overlapping structures. Using clique analysis (a k-clique is a fully connected sub-network of size k), they define a community as a collection of *k*-cliques that can be reached from each other through adjacent k-cliques. Adjacency of two *k*-cliques here means that they share $k-1$ nodes. This definition was successfully applied to identify meaningful communities in co-authorship, word association and protein interaction networks. Their results also confirm that overlap is a significant and universal feature of many real world communities, including gene networks.

4.4 Evaluating Functional Coherence of Gene Group

As a last example of BioLM, a natural extension of pair-wise relationship is to evaluate whether a group of genes have coherent functions, for example, if they are related to the same disease or in the same pathway. This is particularly useful to interpret gene lists generated from clustering analysis of high-throughput experimental data or network partition. Several literature mining tools allow users to submit a gene list as well as keywords list (Becker et al., 2003; Rubinstein and Simon, 2005). The co-occurrences of genes and keywords are tabulated with genes in rows and keywords in columns. The functional coherence of the gene group is measured by the enrichment of a keyword or several related keywords. Chi-square test can be applied to test the significance of association by comparison with a random gene list (Sokal and Rohlf, 1995).

5. Acknowledgments

This work was supported by funds from NIH to MQZ.

References

Adar, E. (2004) SaRAD: a Simple and Robust Abbreviation Dictionary. *Bioinformatics* **20**(4), 527–533.

Aderem, A. (2005) Systems biology: its practice and challenges. *Cell* **121**(4), 511–3.

Ashburner, M., Ball, C.A., et al. (2000) Gene Ontology: tool for the unification of biology. *Nat Genet* **25**(1), 25–29.

Bader, G.D., Donaldson, I., et al. (2001) BIND--The Biomolecular Interaction Network Database. *Nucl. Acids. Res.* **29**(1), 242–245.

Becker, K., Hosack, D., et al. (2003) PubMatrix: a tool for multiplex literature mining. *BMC Bioinformatics* **4**(1), 61.

Boeckmann, B., Bairoch, A., et al. (2003) The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucl. Acids Res.* **31**(1), 365–370.

Cavalli-Sforza, L.L. (2005) The Human Genome Diversity Project: past, present and future. *Nat Rev Genet* **6**(4), 333–40.

Chang, J.T., Raychaudhuri, S., et al. (2001). Including biological literature improves homology search. Pac Symp Biocomput.

Chen, L., Liu, H., et al. (2005) Gene name ambiguity of eukaryotic nomenclatures. *Bioinformatics* **21**(2), 248–256.

Cohen, A., Hersh, W., et al. (2005) Using co-occurrence network structure to extract synonymous gene and protein names from MEDLINE abstracts. *BMC Bioinformatics* **6**(1), 103.

Collier, N., Nobata, C., et al. (2000). Extracting the names of genes and gene products with a hidden Markov model. Proceedings of the 18th International Conference on Computational Linguistics (COLING'2000), Saarbruck, Allemagne.

Ding, J., Berleant, D., et al. (2002). Mining MEDLINE: abstracts, sentences, or phrases? Pac Symp Biocomput.

Donaldson, I., Martin, J., et al. (2003) PreBIND and Textomy – mining the biomedical literature for protein-protein interactions using a support vector machine. *BMC Bioinformatics* **4**(1), 11.

Emili, A.Q. and Cagney, G. (2000) Large-scale functional analysis using peptide or protein arrays. *Nat Biotechnol* **18**(4), 393–7.

Fukuda, K., Tsunoda, T., et al. (1998). Torward information extraction: identifying protein names from biological papers. Proceedings of the Pacific Symposium on Biocomputing'98 (PSB'98), Hawaii.

Girvan, M. and Newman, M.E.J. (2002) Community structure in social and biological networks. *PNAS* **99**(12), 7821–7826.

Hamosh, A., Scott, A.F., et al. (2002) Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. *Nucl. Acids Res.* **30**(1), 52–55.

Hirschman, L., Park, J.C., et al. (2002) Accomplishments and challenges in literature data mining for biology. *Bioinformatics* **18**(12), 1553–1561.

Hoffmann, R. and Valencia, A. (2004) A gene network for navigating the literature. *Nat Genet* **36**(7), 664.

Impey, S., McCorkle, S.R., et al. (2004) Defining the CREB regulon: a genome-wide analysis of transcription factor regulatory regions. *Cell* **119**(7), 1041–54.

Jenssen, T.K., Laegreid, A., et al. (2001) A literature network of human genes for high-throughput analysis of gene expression. *Nat Genet* **28**(1), 21–28.

Jeong, H., Tombor, B., et al. (2000) The large-scale organization of metabolic networks. *Nature* **407**(6804), 651–654.

Kanehisa, M. and Goto, S. (2000) KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucl. Acids. Res.* **28**(1), 27–30.

Kim, T.H., Barrera, L.O., et al. (2005) A high-resolution map of active promoters in the human genome. *Nature* **436**(7052), 876–80.

Kirschner, M.W. (2005) The meaning of systems biology. *Cell* **121**(4), 503–4.

Krallinger, M. and Valencia, A. (2005) Text-mining and information-retrieval services for molecular biology. *Genome Biology* **6**(7), 224.

Leek, T.R. (1997). Information extraction using hidden Markov models. Department of Computer Science, University of California, San Diego.

Lenhard, B., Hayes, W.S., et al. (2001) GeneLynx: a gene-centric portal to the human genome. *Genome Res* **11**(12), 2151–7.

Liu, E.T. (2005) Systems biology, integrative biology, predictive biology. *Cell* **121**(4), 505–6.

Lockhart, D.J., Dong, H., et al. (1996) Expression monitoring by hybridization to high-density oligonucleotide arrays. *Nat Biotechnol* **14**(13), 1675–80.

Matsunaga, T. and Muramatsu, M.-a. (2005) Knowledge-based computational search for genes associated with the metabolic syndrome. *Bioinformatics* **21**(14), 3146–3154.

Palla, G., Derenyi, I., et al. (2005) Uncovering the overlapping community structure of complex networks in nature and society. *Nature* **435**(7043), 814–818.

Ramani, A., Bunescu, R., et al. (2005) Consolidating the set of known human protein-protein interactions in preparation for large-scale mapping of the human interactome. *Genome Biology* **6**(5), R40.

Raychaudhuri, S., Schutze, H., et al. (2003) Inclusion of textual documentation in the analysis of multidimensional data sets: Application to gene expression data. *Machine Learning* **52**(1–2), 119–145.

Reiner, A., Yekutieli, D., et al. (2003) Identifying differentially expressed genes using false discovery rate controlling procedures. *Bioinformatics* **19**(3), 368–375.

Rubinstein, R. and Simon, I. (2005) MILANO – custom annotation of microarray results using automatic literature searches. *BMC Bioinformatics* **6**(1), 12.

Safran, M., Solomon, I., et al. (2002) GeneCards 2002: towards a complete, object-oriented, human gene compendium. *Bioinformatics* **18**(11), 1542–3.

Salwinski, L., Miller, C.S., et al. (2004) The Database of Interacting Proteins: 2004 update. *Nucl. Acids Res.* **32**(90001), D449–451.

Schuemie, M.J., Weeber, M., et al. (2004) Distribution of information in biomedical abstracts and full-text publications. *Bioinformatics* **20**(16), 2597–2604.

Shatkay, H. and Feldman, R. (2003) Mining the Biomedical Literature in the Genomic Era: An Overview. *Journal of Computational Biology* **10**(6), 821–855.

Shen, D., Zhang, J., et al. (2003). Effective adaptation of hidden markov model-based named entity recognizer for biomedical domain. ACL-03 Workshop on Natural Language Processing in Biomedicine.

Shi, L. and Campagne, F. (2005) Building a protein name dictionary from full text: a machine learning term extraction approach. *BMC Bioinformatics* **6**(1), 88.

Sokal, R.R. and Rohlf, F.J. (1995). Biometry. New York, W. H. Freeman.

Stephens, M., Palakal, M., et al. (2001). Detecting gene relationships from MEDLINE abatracts. Pac Symp Biocomput.

Storey, J.D. and Tibshirani, R. (2003) Statistical significance for genomewide studies. *PNAS* **100**(16), 9440–9445.

Temkin, J.M. and Gilder, M.R. (2003) Extraction of protein interaction information from unstructured text using a context-free grammar. *Bioinformatics* **19**(16), 2046–2053.

Venter, J.C., Adams, M.D., et al. (2001) The sequence of the human genome. *Science* **291**(5507), 1304–51.

Watson, J.D. (1990) The human genome project: past, present, and future. *Science* **248**(4951), 44–9.

Wilkinson, D.M. and Huberman, B.A. (2004) A method for finding communities of related genes. *PNAS* **101**(suppl_1), 5241–5248.

Wren, J.D., Bekeredjian, R., et al. (2004) Knowledge discovery by automated identification and ranking of implicit relationships. *Bioinformatics* **20**(3), 389–398.

Wren, J.D., Chang, J.T., et al. (2005) Biomedical term mapping databases. *Nucl. Acids Res.* **33**(suppl_1), D289–293.

Yuan, G.C., Liu, Y.J., et al. (2005) Genome-scale identification of nucleosome positions in S. cerevisiae. *Science* **309**(5734), 626–30.

Zanzoni, A., Montecchi-Palazzi, L., et al. (2002) MINT: a Molecular INTeraction database. *FEBS Letters* **513**(1), 135–140.

Zhang, C. and Li, S. (2004). Modeling of neuro-endoimmune network via subject oriented literature mining. The Fourth International Conference on Bioinformatics of Genome Regulation and Structure (BGRS'2004).

Zhou, G., Zhang, J., et al. (2004) Recognizing names in biomedical texts: a machine learning approach. *Bioinformatics* **20**(7), 1178–1190.

Chapter 11

Computational Immunology: HLA-peptide Binding Prediction

Pandjassarame Kangueane[1], Bing Zhao[2], and Meena K. Sakharkar[2]

[1]*Biomedical Informatics, India*
[2]*Nanyang Technological University, Singapore*

Abstract: HLA molecules are immune proteins that play an important role in T-cell mediated immune response. They bind short 8–20 residues long peptides from antigen proteins to induce immune response. Therefore, the binding of short antigen peptides to HLA molecules is the rate limiting step in T-cell mediated immune response. Several constructs of overlapping short peptides can be designed from a given protein antigen sequence. The number of overlapping peptides is large for systematic experimental testing. Moreover, HLA molecules are highly polymorphic and more than 1500 HLA alleles are known among the human population. Thus, the binding of short peptides to HLA is combinatorial and specific. The binding can be studied using expensive and laborious competitive binding assays. Alternatively, prediction of peptide binding to HLA molecules is highly useful. Efficient prediction models enable systematic scanning of candidate peptides in an effective manner. Here, we describe some commonly used prediction models.

Key words: HLA, Polymorphism, Binding, Prediction, Epitope, Vaccine candidates

1. Background

An important goal of computational immunology is to design peptide vaccine candidates. Considerable efforts have been focused on designing T-cell epitopes. These are short antigenic peptides (8–20 residues long) capable of inducing immune response by binding to HLA molecules (Pamer and Cresswell, 1998). In this process, short peptides from antigen bind HLA molecules and the HLA-peptide complex bind T-cell receptors (TCR) in T-cells. Hundreds of naturally processed peptides of varying length are

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_11,

produced during intracellular antigen fragmentation and only a fraction bind specific HLA molecules to elicit immune response. HLA molecules are highly polymorphic and more than 1,500 HLA alleles are found among different ethic groups (Robinson et al., 2003). Due to high HLA polymorphism and peptide combinations, the binding of peptides to HLA molecules is specific and sensitive. The possible combinations of HLA-peptide complexes are extremely large. Therefore, it is of interest to predict HLA-peptide binding using mathematical models. This phenomenon is technologically exploited to design short epitopes from pathogen proteomes capable of specifically binding a maximum number of HLA molecules representing wider ethnic population. HLA-peptide binding prediction finds application in antigenic peptide selection, degeneration and discrimination during T-cell mediated immune response (Disis et al., 1996; Kawashima et al., 1998; Sarobe et al., 1998; Cooper et al., 1999; Ishioka et al., 1999; Viret and Janeway, 1999). Application of HLA-peptide prediction models in the design of vaccine candidates and immuno-therapeutics (Iwasaki and Barber, 1998; Morgan et al., 1998) is economically advantageous. Prediction is economically advantageous because HLA-peptide binding specificity is generally determined by competitive binding assay of overlapping peptides in antigen sequence. This is laborious, time consuming and expensive to identify highly specific peptides that are recognized by T-cell receptors (Sette et al., 1994; van der Burg et al., 1996). HLA genes are present at different loci in human chromosomes. The HLA class I and HLA Class II genes are known to be associated with CD8+ and CD4+ T-cells respectively, during cell mediated immune response. These two types of HLA molecules have distinctly different structural architectures. However, they both have a structurally similar peptide binding groove. Class I molecules bind peptides of length 8–10 residues and class II molecules bind peptides of length 12–20 residues. Peptide binding to class I molecules are well defined and peptide binding to class II molecules are less well defined. Therefore, prediction of HLA-peptide binding is not generally trivial. However, prediction can be performed using two different types of mathematical models. The first type uses known HLA binding peptides for deriving quantitative matrices and to train non-linear models such as ANN (artificial neutral network), HMM (hidden Markov model), and SVM (support vector machine) for prediction. This approach requires HLA allele specific peptide data and their application is also allele specific and the method is thereafter referred as **`HLA BINDING PEPTIDE BASED METHOD`**. The second type uses energy functions for building molecular models. This approach requires protein structural templates for model building and the method is thereafter referred as **`MOLECULAR STRUCTURE BASED`.** This chapter describes these two types of HLA-peptide binding methods.

2. HLA Molecules

Two classes of HLA molecules are commonly known to be associated with the immune function. Class I HLA molecules are 270 residues long arranged in a single monomer α chain (stabilized by a 99 residues long β_2-microglobulin chain). Class II HLA sequences are 360 residues long arranged in two chains (α & β) of 90–100 residues each. HLA sequences are highly polymorphic (sequence variations) among different ethnic population. The different HLA sequences can be obtained from the IMGT/HLA database (http://www.ebi.ac.uk/imgt/hla/). The HLA nomenclature committee reviews these sequences on a regular basis and represents these sequences using specific HLA allele names (e.g. HLA-A*0201). The sequence for HLA-A*0201 allele from the IMGT/HLA database is given in Figure 11.1. The formation of HLA peptide structural complex is shown for class I molecules in Figure 11.1. However, the formation of HLA-peptide complex is highly combinatorial given the number of known alleles and possible natural peptide constructs.

The HLA class I molecule consists of four distinct domains. These include the a_1, a_2, a_3, and b-2 microglobulin (b_2m). The a_1, and a_2 function as the peptide-binding domain. There is a groove formed between the alpha helices of the a_1 and a_2 domains. The floor of the groove is composed of beta sheets derived from the same domains. The size of the binding site is

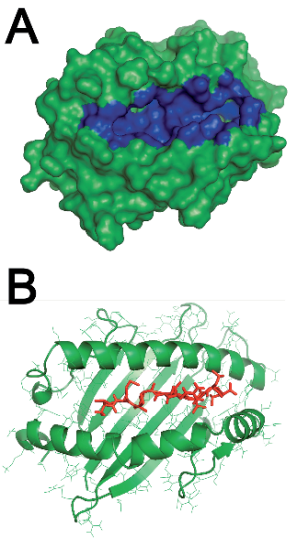


Figure 11.1 Formation of HLA-peptide (PDB: 1MHE) complex is shown. A: HLA molecule surface model, with binding region in dark shade B: HLA-peptide complex in ribbon with peptide depicted in wireframe. Figure generated using pyMOL.

approximately 25 Å long, 10 Å wide, and 5 Å deep. If this groove acted as the lock in the very common "lock and key" model of many proteins, each HLA molecule could only bind to a single selected peptide. The HLA molecule binds to peptides tightly. The groove consists of various pockets. A pocket is defined as the unit having an affinity for a corresponding peptide side chain. Some pockets have a well-shaped structure with an affinity for only one side chain. Other pockets have an affinity for a group of side chains, and sometimes the boundaries between the pockets are not clear. Hydrogen bonding between HLA class I residues and the peptide NH2 and COOH termini allow for a "peptide sequence-independent" binding for short peptides. Pockets A and F play key roles in this binding. They are located at the ends of the groove and accommodate the NH2 and COOH termini, respectively. The side chain of the first residue points upwards towards the solvent. Therefore, there is little restriction on the type of amino acids that can be accommodated by Pocket A. Pocket F consists of a hydrophobic floor and a hydrophilic entrance. Unlike the residues accommodated by Pocket A, these amino acid side chains point toward the floor of the groove. Therefore, bulky aromatic residues are restricted from Pocket F. Peptides recognized by the HLA class I molecules tend to be eight or nine amino acids long. The ends of the peptide are bound to Pockets A and F, and some anchor residues may bind to the middle of the groove. Depending on the length of the peptide, a prominent bulging from the groove will occur. Longer peptides will have a more pronounced bulging from the middle of the groove. This bulging may allow for recognition and direct interaction with the T-cell receptors. The HLA class I molecule can bind to a variety of peptides, because it binds to the region that is common among peptides—the backbone, and it ignores the varying side chains of the peptides.

3. HLA Binding Peptide Based Methods

Several models have been developed using peptide data in large databases derived from naturally bound peptides (Kubo et al., 1994; Meister et al., 1995; Rammensee et al., 1995; Rammensee et al., 1999) or synthetic peptide libraries (Parker et al., 1994; Stryhn et al., 1996). These methods depend on the amount of HLA allele specific peptide binding data. This data is available in several published papers. This data represents information on HLA binding peptides and non-binding peptides. Brusic and Harrison collected such information from published papers and developed a database called HLAPEP in 1998. This is a pioneering work of data collection on HLA binding peptides which later led to the development of prediction

models by Brusic and colleagues (Brusic et al., 1998). Following this several authors have developed similar databases for HLA binding and HLA non-binding peptides. These datasets were regularly reviewed and updated. This data is the basis for the development of a number of HLA-peptide binding prediction models listed below.

- Binding motif model (Kubo et al., 1994; D'Amaro et al., 1995; Meister et al., 1995; Rammensee et al., 1995; Rammensee et al., 1999)
- Artificial Neural Network model (Adams and Koziol , 1995; Brusic et al., 1998; Milik et al., 1998)
- Stepwise Discriminant Analysis (Mallios, 1999)
- Hidden Markov Model Based Methods (Mamitsuka, 1998; Brusic et al., 2002; Noguchi et al., 2002)
- Support Vector Machines (Donnes and Elofsson, 2002)
- Quantitative Matrix Based Methods (Parker et al., 1994)
- Positional Scanning (Udaka et al., 2000)
- Profile Motifs (Gribskov et al., 1987; Thompson et al., 1994)
- Additive Method (Free and Wilson, 1964; Doytchinova et al., 2002)

3.1 Sequence Based Prediction Models

HLA bound peptides are generally restricted at some primary positions (second or fifth and last positions) of the peptides. These residues are known as anchor residues and positions are known as anchor positions. Allele specific sequence motifs can be identified by studying the frequencies of amino acids in anchor positions. For example, a simple motif based-prediction method is based on the observation that the peptides binding to HLA A*0201 are often nonamers, and frequently have two anchor residues, a lysine in position 2 and a Valine in position 9 (Rammensee et al., 1995). Besides the anchor residues, there are also weaker preferences for specific amino acids in other positions. One method to include this information is to use a profile, in the form of a matrix, where each type of amino acid in each position is given a score. The scores can be calculated from observed amino-acid frequencies in each position or be set manually. The sum of contribution by all the residues gives predicted binding value. These methods are easy to implement, and are one of the most popular methods applied to HLAp binding prediction. One frequently used profile based prediction method is SYFPEITHI (Rammensee et al., 1999). These methods are based on the assumption that different peptide positions contribute in an additive manner to the overall binding affinity, the contribution of each peptide residue to the binding affinity is independent of other neighboring amino acid residues. The interactions between different positions are not

taken into account. Furthermore, their real predictive power is directly dependent on the amount of experimental data used to interpolate HLA binding properties. Thus, HLA alleles for which rather few (or no) experimental data are available are unsuitable for these sequence based prediction methods.

Besides, these methods do not consider information from non-binding peptides. This information can be used by machine learning methods. In the meantime, the predictive power of machine learning methods is not affected by the interactions among peptide residue positions. Prediction of MHC-peptides has been made by using machine learning approaches such as artificial neural network (ANN) (Adams and Koziol, 1995; Brusic et al., 1998; Milik et al., 1998), stepwise discriminant analysis (Mallios, 1999), hidden Markov model (HMM) (Mamitsuka, 1998; Brusic et al., 2002; Noguchi et al., 2002), and Support Vector Machine (SVM) (Donnes and Elofsson, 2002). Gulukota et al. (Gulukota et al., 1997) showed that one advantage of machine learning algorithms compared to profile methods seems to be that they have a higher specificity. This is possible due to the inclusion of non-binding data in the training. A machine learning approach extracts useful information from a large amount of data and creates a good probabilistic model. In the case of MHC-peptide prediction, a data set of known binders and known (or supposed) non-binders is used. This set is then used to build a model that discriminates between binding peptides and non-binding peptides. This model can then be used to predict whether a novel peptide binds or not. Besides these methods, there are other methods that can be considered as an extension form the sequence based methods like profile motifs (Reche et al., 2002) and additive method (Doytchinova et al., 2002), as they also consider the influence of the structure of the HLAp complexes.

3.1.1 Binding Motif-Based Methods

Binding motif-based methods determine peptide binding property by identifying general position based pattern of amino acids in favor of HLA peptide binding (Rammensee et al., 1995; Rammensee et al., 1999). The binding of a peptide to an allele is examined on the basis of occurrence of specific residues at specific position. The presence of motifs will determine whether a peptide will bind to specific allele or not. HLA bound peptides are generally restricted at two primary positions (second or fifth and last positions). These residues are known as anchor residues and positions are known as anchor positions. Based on these anchor positions (Rammensee et al., 1995), simple binding motifs have been defined for specific HLA alleles. Thus, position based patterns of recurrent amino acids in a known dataset are identified and generated. However, the compliance of a peptide sequence to

such a binding motif is neither sufficient nor necessary to ensure binding (Ruppert et al., 1993; Townsend et al., 2006). The usefulness of the motifs is further diminished due to the presence of the secondary anchor residues at the non-conserved positions (Ruppert et al., 1993). This method has been proved to be too simple, as the binding ability of a peptide to a given HLA molecule cannot be explained exclusively in terms of the presence or absence of a few anchor residues.

Obviously, it is not enough to consider only the contribution of amino acid on anchor positions to the overall binding affinity. Weaker preferences for specific amino acids in other positions should be taken into account to improve the predictive power of these methods. In some profile based methods, a score is given for each type of amino acid in each position. The scores can be calculated from observed amino acid frequencies in each position or be set manually. The sum of the scores for a given peptide is then used to make predictions.

An example of this category of methods is the method developed by Parker et al. (Parker et al., 1994), which predicts the relative binding strengths of all possible nona-peptides to the HLA class I molecule based on experimental peptide binding data. The method is based on the observation that each side-chain of the peptide contributes a certain amount to the stability of the HLA complex that is independent of the sequence of the peptide. These contributions is quantify based on the binding data from a set of 154 peptides. A table containing 180 coefficients (20 amino acids × 9 positions) is generated, each of which represents the contribution of one particular amino acid residue at a specified position within the peptide to binding to HLA. A web-based HLA peptide binding prediction service based on this work is maintained by BIMAS (BioInformatics & Molecular Analysis Section) at NIH with the URL: http://bimas.cit.nih.gov/molbio/hla_bind/, which ranks potential 8-mer, 9-mer, or 10-mer peptides for a limited set of HLA alleles according to predicted binding affinity. Another web site for predicting HLA binding peptides is SYFPEITHI (Rammensee et al., 1999) (http://www.syfpeithi.de/), which is primarily a database for HLA binding peptides and peptide motifs. It also provides motif-based HLA binding peptide prediction. The matrices in SYFPEITHI were adjusted manually, by assigning a high score (10) for frequently occurring anchor residues, a score of eight to amino acids that occur in a significant amount and a score of six to rarely occurring residues. Preferred amino acids in other positions have scores that range from one to six and amino acids regarded as unfavorable have scores ranging from –3 to –1. SYFPEITHI prediction can be done for 13 different HLA class I types.

It has been shown that profile based methods are correct in about 30% of the time, meaning that one-third of the predicted binders actually bind

(Gulukota et al., 1997). Prediction models based on binding motifs are mostly all-or-nothing algorithms with very high false negative rates. Both motifs and matrix models present only one sequence pattern in the given set of data that will bind to an HLA molecule. They cannot extract multiple sequence patterns hidden in a given set of data separately, even if each of them has sufficient binding ability.

3.1.2 Artificial Neural Network Based Methods

Artificial Neural Networks (ANNs) are complex, non-linear, and self-training systems that are able to extract and retain patterns hidden in the training data and recognize them in an input dataset (Adams and Koziol, 1995; Brusic et al., 1998; Milik et al., 1998). Neural networks are excellent at classifying non-linear data. In the case of HLA binding peptide prediction, it can utilize the information form both binders and non-binders. The training data of peptide sequences have to be properly aligned. ANN-based models have proven very effective for the prediction of class-I (A*0201, Kb) (Milik et al., 1998) and class II HLA (DRB1*0401) binding peptides (Brusic et al., 1998). Gulukota et al. (Gulukota et al., 1997) reported that the performance of the back propagation neural networks exceeds those of matrix models and motifs in discriminating peptides that bind to an HLA molecule from other peptides (Figure 11.2).

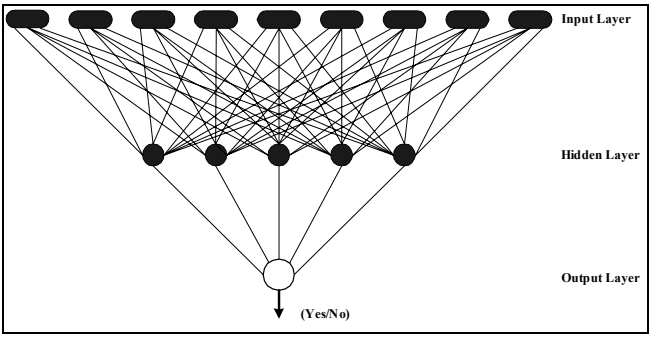


Figure 11.2 The architecture of the neural network is shown. There are three layers' of nodes connected in a defined topology, where each node has input and output connections to other nodes. In general, a neural network will receive an input pattern (e.g. an amino acid sequence whose secondary structure is to be predicted), which sets the values of the nodes on the first layer (the input layer). These values are propagated according to transfer functions (the connections) to the next layer of nodes, which propagate their values to the next layer, until the output layer is reached. The pattern of activation of the output layer is the output of the network.

The application of ANN on prediction of HLA binding peptide is based on the assumption that the binding can be influenced by the amino acids on all the positions of the peptide. An ANN has the ability to simultaneously analyze the influence of all the amino acids of the peptide and, thus, may improve binding predictions. Adams and Koziol (Adams and Koziol, 1995) applied Neural Networks to predict the binding capacity of peptides to HLA-A*0201. With a large set of binding data from 552 nonamers and 486 decamers, the neural networks achieve a predictive hit rate of 0.78 for classifying peptides as good or intermediate *versus* weak or non-binders. The neural nets also depict specific motifs for different binding capacities. ANN is theoretically applicable to all HLA class molecules, given a suitable training dataset of known binding affinities. The trained networks can then be used to perform a systematic search through all pathogen or tumor antigen protein sequences for potential cytotoxic T-lymphocyte epitopes.

The major drawback of ANN based methods is that the ANN models require large training set with known binding data. In addition, promiscuous prediction is not feasible since data for one allele cannot be extrapolated to other alleles. An interesting approach is a combination of neural networks and evolutionary algorithm (Brusic et al., 1998). An evolutionary algorithm is used to evolve a scoring matrix during preprocessing and the combined approach is shown to perform better, which achieved a correct classification percentage for both binders and non-binders in excess of 80% (Brusic et al., 1998).

3.1.3 Stepwise Discriminant Analysis

Stepwise discriminant analysis (SDS) is another data-driven algorithm. The results are data dependent and change when the data sets change. However, as the data sets grow, the sample space is better represented and the influence of individual peptides decreases. Mallios (Mallios, 1999) applied an iterative SDA on a large molecular database to derive quantitative motifs for peptide binding (Figure 11.3). From two mutually exclusive sets, stepwise discriminant analysis builds a Bayesian discriminant function that classifies each element into one of the two sets (Mallios, 1999). Specifically, an element is assigned to a set if the Bayesian posterior probability of belonging to that set exceeds the probability of belonging to the complementary set. Arguments for the function are selected from a list of potential predictor variables.

In the case of HLA binding motifs determination, the two mutually exclusive sets are the binding data set of subsequences and the non-binding data set of subsequences. The potential predictor variables for each case (subsequence) describe the biochemistry and position of each amino acid

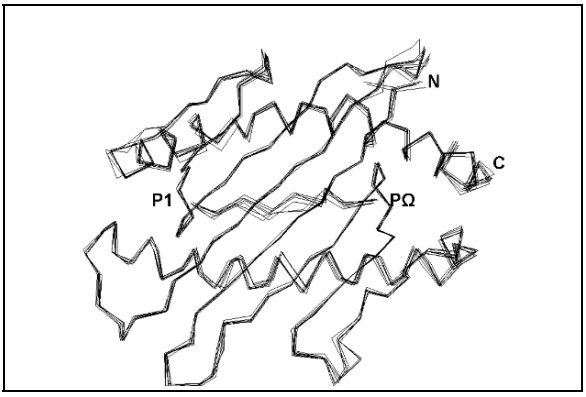


Figure 11.3 Superimposed HLAp structures. The picture shows the conserved structure of HLA class I-peptide complexes, and how peptides of different sequences bind to the same HLA molecule. The structures reveal that residues at anchor positions of the peptide fit into corresponding pockets in the HLA groove, and that the peptide backbone is hydrogen bonded to several side-chains of the MHC. A conserved form therefore binds a multitude of different peptides, which rearrange themselves within the imposed constraints. The structural information can be used as the basis for modeling studies.

residue. The decision rule for classification is based on the probability of set membership. The resultant model is quantitative and can be used to predict peptide binding. Mallios (Mallios, 1999) produced four closely related models for HLA-DR1. Each model correctly classifies >90% of the peptides in the database.

3.1.4 Hidden Markov Model Based Methods

The major problems in the Motif, Matrix, or ANN based methods is that all these methods assume that the size of peptides that bind to HLA molecules is fixed, though actually the length of peptides that bind to HLA molecules is variable. Thus, these methods cannot predict the binding ability of a peptide whose length is longer or shorter than that of the peptides used in training, and thus, available training and test data are extremely limited. Specifically, although ANN are able to learn multiple sequence patterns in a given set of data automatically, the network parameters are given only as real-valued weights attached to edges connecting nodes in the network. Consequently, the weights cannot present any understandable training results.

Mamitsuka et al. (Mamitsuka, 1998) applied supervised learning of a hidden Markov model (HMM) to overcome these shortcomings. HMMs are suitable for representing time-series sequences (strings) having flexible lengths. They used a fully connected HMM, which can automatically divide multiple sequence patterns hidden in a given set of data into separate

patterns and is able to rep resent more than one sequence pattern hidden in a set of given training data. Besides, a trained HMM can be presented as a comprehensible form, just like a sequence profile derived from multiple sequence alignment. Their HMMs were trained for HLA-A2, HLA-DR1, and HLA-DR4. Two experiments were performed. Compared with a backpropagation neural network, the average discrimination accuracy of the HMMS is approximately 2–15% better.

Brusic et al. (Brusic et al., 2002) applied HMM for the prediction of peptide binding to the HLA-A2 supertype. Combined with a representation of peptide/HLA interactions in which the specific HLA–peptide interaction by combining each amino acid of the peptide with the variable amino acids of its positional environment, their system, called MULTIPRED, showed high accuracy of peptide-binding predictions for HLA-A*0201, A*0204, and A*0205 alleles, good accuracy for A*0206 allele, and marginal accuracy for A*0203 allele. Noguchi et al. (Noguchi et al., 2002) applied HMM with successive state splitting (S-HMM) on prediction of peptides binding to a class II HLA, HLA-DRB1*0101). In the relative operating characteristic (ROC) analysis, the S-HMM prediction had values of ROC>0.85, which is better than other machine learning methods. In addition, the S-HMM may be trained on positive binding data only, and the preprocessing of training data, such as peptide alignment and the selection of binding cores, is not required. Thus, the method is simpler to implement.

3.1.5 Support Vector Machines

Support vector machines (SVM) are supervised classifiers that try to find a linear separation between different classes of points in a high-dimensional space. In a 2D space, this separator is a line; in 3D, it's a plane. In general, this separating surface is called a hyperplane. Support vector machines have two special features. First, instead of just finding any separating hyperplane, they are guaranteed to find the optimal one, or the one whose placement yields the largest separation between the two classes. The data points nearest the frontier between the two classes are called the support vectors, which refer to the coordinates of the data points. Second, although SVMs are linear classifiers, they can classify non-linearly separable sets of points by transforming the original data points into a higher dimensional space in which they can be separated by a linear surface.

Donnes and Elofsson (Donnes and Elofsson, 2002) presented a novel approach, called SVMHC, based on support vector machines to predict the binding of peptides to HLA class I molecules. Their result shows that the method seems to perform slightly better than profile based methods. SVHLA currently provides prediction for 26 HLA class I types from the HLAPEP

database or alternatively 6 HLA class I types from the higher quality SYFPEITHI database. The method is easy to apply to a large number of HLA class I types as more peptide data are available.

3.1.6 Quantitative Matrix Based Methods

Quantitative matrices provide a detailed linear model which is easy to implement (Sette et al., 1989; Parker et al., 1994; Gulukota et al., 1997). In this category of methods, the contribution to binding at each peptide position within the binding groove is quantified (Parker et al., 1994). This method involves producing a matrix in which every entry (X, Y) represents a score associated with amino acid residue X at position Y. The position-specific amino acid values reflect the structural properties of HLA alleles, therefore representing a fingerprint for HLA binding domains. Summing the scores for every residue in a given peptide yields a predicted binding score. The method has been extensively tested for HLA-A2.1 binding predictions. A recent implementation of matrix method is ProPred1 which allows prediction for 47 class-I alleles using co-efficient matrixes and empirical equations.

The matrix method enables prediction for a wide pool of peptides in a high-throughput manner unlike motif based approaches. However, a serious obstacle is in the generation of binding co-efficient matrix for each HLA allele which requires the experimental testing of hundreds of peptides. Another clear limitation of this method is that it assumes that every amino acid residue in a certain position influences binding independently of its neighbors. Matrix methods can give fast predictions on the basis of simple patterns, but they have less capacity to encode non-linear dependencies. This limitation does not yield matrix-based methods completely useless as there is certain simple generalization about amino acid preferences at specific positions influencing binding.

3.1.7 Positional Scanning

Udaka et al. (Udaka et al., 2000) analyzed the specificities of three mouse HLA class I molecules, K^b, D^b, and L^d, positional scanning using synthetic combinatorial peptide libraries. Graded concentrations of peptides were tested for HLA binding. Peptide concentrations that stabilized a half-maximal number of peptide-receptive HLA class I molecules (SD_{50}) were calculated from the mean fluorescence intensities acquired in an arithmetric scale. Correlation between MHC-binding scores and experimentally measured SD_{50} values was analyzed. By comparing the MHC-binding capacities of sublibraries with the completely random library as

reference, the impact of a given amino acid at the position was evaluated. By scanning all nine positions, the profiles of amino acid preference by K^b, D^b, and L^d were obtained. Each HLA molecule exhibited a distinct profile. The library scanning yielded a quantitative measure of the impact of every amino acid on HLA binding. The result of the analysis was used to create a scoring program to predict MHC-binding peptides in proteins.

The scoring program was then tested with a number of peptides by comparing the prediction with the experimental binding. The score and the experimental binding exhibited a linear correlation but with substantial deviations of data points. Statistically, for approximately 80% of randomly chosen peptides, MHC-binding capacity could be predicted within one log concentration of peptides for a half-maximal binding. Known cytotoxic T-lymphocyte epitope peptides could be predicted, with a few exceptions. Although the positional scanning data are only informative about an additive component of the binding energy supplied from individual amino acids on peptide, they still provide better information than anchor amino acids alone.

3.1.8 Profile Motifs

This method is an extension from the motif matrix methods (Stryhn et al., 1996). The most simple sequence patterns that are usually extracted from large numbers of existing known peptides, or from pool sequencing experiments, has been proved to be too simple. Motif matrices have been developed to overcome these limitations by accounting for the preference of every amino acid type at every position in the peptide (Rammensee et al., 1999). Coefficients in these matrices relate to the strength of the amino acid signals in a pool sequence of peptides eluted from a given MHCI molecule, or to the occurrence of an amino acid in a set of binding peptides.

However, it is well established that position specific scoring matrices (PSSM) or profiles created from a set of aligned sequences provides a better way for defining and recognizing sequence motifs (Gribskov et al., 1987). There are several methods to generate PSSM from aligned sequences, usually including distinct sequence weighting methods (Thompson et al., 1994). In all cases, profile coefficients relate to the observed frequency of every amino acid at the position column of the alignment, corrected by the expected frequency of that amino acid in the background using a reference database. Thus, in this approach the binding potential of any peptide (query) to a given HLA molecule can be obtained by comparing the query to a PSSM created from a set of aligned MHC-specific peptides.

Reche et al. (Reche et al., 2002) have derived alignments and profiles from a collection of peptides known to bind two specific class I MHC, K^d , and D^b, compatible with the structural and molecular basis of the HLAp

interaction. A search algorithm, RANKPEP, which ranks all possible peptides from a test protein using PSSM coefficients, was developed to automate the screening process. It was shown that for Kb and Db molecules, that profiles created from aligned peptides are very sensitive in identifying MHCI-restricted epitopes. The predictive power of the method was evaluated by running RANKPEP on proteins known to bear K^b- and D^b-restricted T-cell epitopes. Analysis of the results indicates that > 80% of these epitopes are among the top 2% of scoring peptides.

These profiles are guided by structural data indicating differences in binding residues involving peptides of distinct length. Prediction of peptide-HLA binding using a variety of MHC-specific PSSMs is publicly available on line (Reche et al., 2002). This method is data driven, thus, heavily depends on the quality of the binding data, while available binding data are collected from different sources with different experimental conditions and with peptide sequences of biased amino acid composition, which is a limitation of all data driven prediction method.

3.1.9 Additive Method

This method further developed the additivity concept, developed by Free and Wilson (Free and Wilson, 1964) whereby each substituent makes an additive and constant contribution to the biological activity regardless of substituent variation in the rest of the molecule. The values of the individual group contributions are calculated by multiple linear regression (MLR) analysis. The models based on the additivity concept are simple to perform and easy to interpret. Because of that they have found a wide application in molecular design over the years. However, it has been shown that the conformation of a certain amino acid side chain at a certain position strongly depends on the neighboring amino acids (Fremont et al., 1995). This means that the additivity hypothesis is not sufficient to explain the binding abilities of the peptides.

This method is developed to overcome the above limitations (Doytchinova et al., 2002), which is based on the assumption that the binding affinity of a peptide depends on the contributions from each amino acid as well as on the interactions between the adjacent and every second side-chain. In this method, the partial least squares (PLS) were implemented for the multiple linear regression analysis between the different residue interaction terms and the binding affinity. (PLS) method belongs to so-called projection methods. These methods handle data matrixes with more variables than observations very well, and the data can be both noisy and highly collinear. In this situation, conventional statistical methods such as multiple regression produce a formula that fits the training data but is

unreliable for prediction. PLS forms new variables as linear combinations of the old ones and then uses them as predictors of the biological activity.

Doytchinova et al. applied this method on class I molecule HLA-A*0201, using a training set of 420 experimental IC_{50} values (Doytchinova et al., 2002). The predictive power of the method was assessed by a "leave-one-out" cross-validation with an independent test set of 89 peptides. The mean value of the residuals between the experimental and predicted pIC_{50} values was 0.508 for this test set. The additive method for quantitative binding affinity prediction is easy and fast to use and gives a quantitative value for the binding affinity with very good predictive powers. It can also give a quantitative assessment of individual amino acid contributions at any position in the peptide. The additive method has been implemented in a program for rapid T-cell epitope search. The method is universal and can be applied to any peptide-protein interaction where binding data is known. However, due to its nature as a sequence-based method, the prediction power will eventually depends on the quality of the binding affinity data of the training set.

3.1.10 Summary

The application of any of the detailed models above is restricted to either one or few HLA alleles depending on the availability of training set. As shown in Table 11.1, these models have been tested for H-D^b (D'Amaro et al., 1995; Udaka et al., 2000), H-K^b (D'Amaro et al., 1995; Udaka et al., 2000), H-L^d (Udaka et al., 2000), HLA-A2 (Brusic et al., 1998), DR1 (Mallios, 1999), DRB1*0101 (Noguchi et al., 2002), and another 26 alleles (Donnes and Elofsson, 2002). It has been shown that their prediction accuracy varies from 90 to 100% for these datasets. It should be noted that the size of the training set is different in different cases. It is worth mentioning that the sizes of negative and positive data used in such developments are also variable and these factors play an important role in the overall estimation of their accuracy and predictive power. It will be interesting to compare the usefulness of these techniques either individually or combined and check their prediction efficiencies using the same set of training and prediction dataset.

3.2 Molecular Structure Based Predictions

The second category of prediction methods uses known three-dimensional structures of HLAp complexes. The current release of Protein Data Bank (Berman et al., 2000) contains a number of unique HLA structures. These structures enable a better understanding of the structural principles

Table 11.1 Tools for predicting MHC peptide

Tool	URL
CTLPRED	http://www.imtech.res.in/raghava/ctlpred/
PROPRED	http://www.imtech.res.in/raghava/propred1/
MAPPP	http://www.mpiib-berlin.mpg.de/MAPPP/binding.html
NHLAPRED	http://www.imtech.res.in/raghava/nhlapred/
HLABIND	http://thr.cit.nih.gov/molbio/hla_bind/
LPPEP	http://zlab.bu.edu/zhiping/lppep.html
SVMHC	http://www-bs.informatik.uni-tuebingen.de/Services/SVMHC
NetMHC	http://www.cbs.dtu.dk/services/NetMHC/
MHCPred	http://www.jenner.ac.uk/MHCPred/
MMBPRED	http://www.imtech.res.in/raghava/mmbpred/
MHCBIND	http://margalit.huji.ac.il/Teppred/mhc-bind/index.html
SYFPEITHI	http://www.syfpeithi.de/
PROPRED	http://www.imtech.res.in/raghava/propred/
EPIPREDICT	http://www.epipredict.de/Prediction/prediction.html
HLADRPRED	http://www.imtech.res.in/raghava/hladr4pred/
MHC2PRED	http://www.imtech.res.in/raghava/mhc2pred/
TED	http://www.bioinformation.net/ted/

governing peptide recognition by HLA molecules (Batalia and Collins, 1997) (Figure 11.3). HLA molecules bind peptides of diverse sequence with great affinity and long half-life. Most peptides selected by class-I molecules are 8–10 residues long and conserved amino acids bind the invariant portions of the peptides, presenting anchoring backbone atoms at positions 2 and C, N termini (Madden et al., 1992; Guo et al., 1993). Auxiliary anchors at P1 and P3 usually fine tune peptide recognition (Madden et al., 1992; Ruppert et al., 1993). Each anchoring side chain interacts with one of the six polymorphic HLA pockets (Saper et al., 1991; Guo et al., 1993), whose structural fold is conserved in evolution with physicochemical diversity for allele specificity (Falk et al., 1990). Through a set of hydrogen bonds to the main chain of the peptide, the termini of the peptide are oriented into specific pockets that are designed to accommodate the chemical nature of the peptide residues. Thus, the orientation (amino to carboxyl) of the antigenic peptide is fixed for all HLA molecules. However, this arrangement is affected by peptide length. Longer peptides may zigzag (Madden et al., 1993) or bulge (Guo et al., 1992; Collins et al., 1995) to allow peptides of greater length to maintain the relative position of the termini. In addition, longer peptides may bind and maintain original binding at the N terminal end with appropriate structural adjustments at C terminal to allow for peptide extension outside the binding groove.

The following alleles are presented, superimposed by the HLA a1 and a2 domains. HLA-A2 with a nonamer and a decamer (1hhiand 1hhh, respectively), HLA-Aw68 with a nonamer (aw68), HLA-B27 with a nonamer (b27), HLA-B35 with an octamer (1a1n), HLA-B53 with a nonamer (1a1m), HLA-B8 with an octamer (1agd), H2-Kbwith an octamer and a nonamer (2vaa and 2vab, respectively), H2-Dbwith a nonamer (1hoc), and H2-M3 with a nonamer (1mhc). The peptide C_α and C_β atoms are shown within the trace of the HLA structure. The first and last positions of the peptide are labeled as P_1, and P_Ω, respectively.

Various methods calculating binding free energy of HLAp complexes, based on different energy scoring functions have been developed. Since binding free energy of MHC-peptide complex is related to the affinity of MHC-peptide binding, binders and no-binders can be discriminated and the approach produces absolute or relative peptide binding affinity. Free energy calculation is either based on statistical pair-wise potentials tables or free energy scoring functions. A recent approach based on free energy involved threading of the peptides using known templates followed by evaluation of their binding by statistical pair wise potentials (Altuvia et al., 1995; Altuvia et al., 1997; Schueler-Furman et al., 1998; Schueler-Furman et al., 2000). However, it does not allow the direct prediction of binding affinity values unlike methods capable of calculating the absolute binding free energies from three-dimensional homology models. Although, it is difficult to develop a universal free energy function for HLAp binding, attempts have been made towards this goal (Rognan et al., 1999).

The backbone conformations of bound peptides are not generally conserved in the binding groove. The bound peptides are flexible and the middle part of the peptides usually bulges out the binding groove. This bulging part allows the backbones to take different patterns at the groove. The generic peptide structure determination methods using Monte Carlo, molecular dynamics simulations, dynamic programming, free energy mapping, or threading are suited for binding free energy calculations, but they all have difficulty in predicting the conformation of the peptide in the groove. Another method uses computational combinatorial ligand design (CCLD) (Zeng et al., 2001) for placing amino acids in specific pockets. A method based on three-dimensional quantitative structure affinity relationship (3D QSAR) of HLAp complexes (Doytchinova and Flower, 2001; Doytchinova et al., 2002) has also been developed. Most approaches, except for threading (Altuvia et al., 1995), are not generally suitable for systematic high-throughput genome scanning. However, the TEPITOPE software developed by Hammer and colleagues uses pocket profiles generated using structural data (Sturniolo et al., 1999). This method is shown effective for HLA-DR alleles.

3.2.1 MDS

The first approach based on molecular dynamics simulation (MDS) of HLAp complexes (Rognan et al., 1994) allowed a crude discrimination of binders from non-binders. The approach produced change in free energy during simulation of the HLA-B*2705 complex with six different peptides in AMBER force field. The result exhibited unexpected structure-activity relationships. Various structural and dynamical properties of the solvated protein–peptide complexes (atomic fluctuations, solvent-accessible surface areas, hydrogen bonding pattern) were found to be in qualitative agreement with the available binding data. The molecular dynamics method could be used as a complementary tool to T cell-epitope predictions, as crystal structures of HLA proteins available. This method is not suitable for high-throughput predictions due to extensive computational requirements in capturing the simulation trajectory.

3.2.2 Threading with Knowledge Based Free Energy Scoring

The knowledge based scoring method based on pair-wise contacts uses solved or modeled structure for HLAp binding calculations. The physical, chemical compatibility between peptide and HLA groove is estimated using pair-wise potential matrix. The binding score is obtained by adding all pair-wise values for residues in the pocket with the corresponding peptide residues at every position. This enables the ranking of peptides for HLAp binding (Altuvia et al., 1995; Altuvia et al., 1997; Schueler-Furman et al., 1998; Schueler-Furman et al., 2000). The structure based approach is based on two main determinants: (1) The availability of appropriate peptide structural template; (2) the choice of a pair-wise potential table.

Various knowledge based pair-wise potentials have been derived from known protein structures (Jernigan and Bahar, 1996; Jones and Thornton, 1996; Skolnick et al., 1997). A basic approximation underlying these potentials is that total “free energy” of a protein can be expressed as a sum of independent pair-wise interactions. The frequencies of residue pairs in the structures are assumed to represent the interaction preference between different types of residues. This interaction preference between two amino acids is expressed by its comparison with their affinity to a “reference state.” Various matrices have been published using distinct reference states (Skolnick et al., 1997). Miyazawa and Jernigan used solvent as reference state and developed a matrix with emphasis on hydrophobic interactions (Miyazawa and Jernigan, 1985; Miyazawa and Jernigan, 1996). Betancourt and Thirumalai (Betancourt and Thirumalai, 1999) modified the table by changing the reference state from solvent to a defined, single solvent-like

molecule. The resulting matrix represents hydrophilic interactions. Altuvia et al. used the potential tables of Miyazawa and Jernigan to rank modeled HLAp structures (Altuvia et al., 1995; Altuvia et al., 1997). However, the procedure failed to predict hydrophilic interactions (Altuvia et al., 1997). The pair-wise potential table of Betancourt and Thirumalai (Betancourt and Thirumalai, 1999) successfully selected hydrophilic interactions (Schueler-Furman et al., 2000).

3.2.3 Free Energy Scoring Function Based Methods

These methods aim is to determine ligands capable of binding from a series of candidate ligands by calculating binding free energy. They generally do not require predetermined experimental data for model development and can produce relatively accurate binding affinity. HLAp binding or non-binding data is not enough to predict whether the peptide can induce immune response. Accurate calculation of HLAp binding free energy difference using 3D- structures by simple free energy scoring functions is CPU intensive.

In recent years, a number of free energy scoring functions has been developed for different purposes and these functions are used for HLAp binding predictions. One of the recent approaches for class I HLAp binding prediction is a tailor-made free energy scoring function (FRESNO) combined with homology modeling (Rognan et al., 1999; Logean et al., 2001; Logean and Rognan, 2002). Starting from the primary sequence of the protein antigen, individual 3D structures of all possible class I MHC-peptide (8-, 9-, and 10-mers) complexes are constructed by homology modeling. The critical issue in this approach is identification of peptide templates for structure prediction. The FRESNO scoring function is then used to calculate binding free energy of HLAp interactions. The approach allows for the prediction of absolute binding affinities in a high throughput manner. An extension to this work is EpiDock (Logean and Rognan, 2002) which is (1) shown to predict potential T-cell epitopes from viral proteomes (2) used to roughly predict still unknown peptide binding motifs for novel class I HLA alleles.

3.2.4 Virtual Matrix Based Methods

Virtual matrices, like quantitative matrices, provide a detailed model in which binding of each peptide residue with HLA pockets is quantified using pocket profiles (Sturniolo et al., 1999). Virtual matrices are derived by assigning and combining pocket-specific binding properties using structural features or homology principles from known HLA structures and

extrapolating to other alleles, while quantitative matrices are obtained using peptide data with known allele specific binding data. The advantage over quantitative matrices is that the method is generic and can be applied to any given allele. One implementation of the algorithm is the software package TEPITOPE (Sturniolo et al., 1999). Its capacity has been demonstrated for 11 HLA-DR alleles. Furthermore, they have been successfully applied to predict T-cell epitopes in oncology, allergy, and autoimmune diseases (Rognan et al., 1994; Hammer et al., 1995; Gross et al., 1998; Cochlovius et al., 2000; Stassar et al., 2001).

3.2.5 CCLD

Computational combinatorial ligand design (CCLD) is a computational tool used to assist drug design, by clustering compounds in classes of drug-like and non-drug-like molecules. The method selects fragments that bind favorably to a macromolecular target of known three-dimensional structure. Optimal positions and orientations of functional groups on the surface of the macromolecule are exhaustively searched, and then sorted according to an approximated binding free energy. The CCLD method allows the fast and automatic generation of a multitude of highly diverse compounds, by connecting in a combinatorial fashion the functional groups in their minimized positions. The fragments are linked as two atoms may be either fused, or connected by a covalent bond or a small linker unit. To avoid the combinatorial explosion problem, pruning of the growing ligand is performed according to the average value of the approximated binding free energy of its fragments.

The CCLD method uses the 3D information from the crystal structure of the molecule. It can generate both sequence and structure of predicted ligands. Zeng et al. (Zeng et al., 2001) applied the method on the prediction of peptides that bind a HLA molecule with known crystal structure. Using chemical fragments as models for amino acid residues, a set of sequences for peptides predicted to bind in the HLA peptide-binding groove were produced. The probabilities for specific amino acids occurring at each position of the peptide were calculated based on these sequences. Their results show the CCLD approach is a sensitive method that can capture the important features of both sequence and structural data.

3.2.6 3D-QSAR

Three-dimensional quantitative structure-affinity relationship (3D QSAR) studies have been applied to explore the molecular interactions between HLA and peptides. They provide easily interpretable coefficient contour

maps identifying the areas of the peptides that require a particular physicochemical property to increase binding. One of the 3D QSAR methods, Comparative Molecular Similarity Indices Analysis (CoMSIA), is a very reliable method for investigating the structure-activity trends within sets of biological molecules. It has been successively applied in pharmaceutical discovery of small molecule drugs. It is a statistic approach that seeks to correlate relative differences in molecular descriptor values to a dependent property (e.g. the binding affinity). In that respect, CoMSIA is a method able to map similarities or dissimilarities between molecules. The explanatory power of CoMSIA methods is considerable, manifest not only in their ability to accurately predict binding affinities, but also in their capacity to display advantageous and disadvantageous 3D interaction potential mapped onto the structures of molecules being investigated.

CoMSIA method have been applied on peptide-HLA binding prediction with class I HLA molecule HLA-A*0201 and 200 and 66 nonamer peptides (Doytchinova and Flower, 2001; Doytchinova et al., 2002). CoMSIA uses the interaction potential around aligned sets of 3D peptide structures to describe the contributions to binding. The relationship between physico-chemical properties and the affinities of peptide binding was investigated. The X-ray structure of one nonameric viral peptide was used as a starting conformation, on which the structures of the remaining peptides were built. Five types of similarity index (steric bulk, electrostatic potential, local hydrophobicity, hydrogen-bond donor, and hydrogen-bond acceptor abilities) were calculated, using a common probe atom with 1 Å radius, charge +1, hydrophobicity +1, hydrogen-bond donor, and acceptor properties +1. Since only the combination of all fields provided a complete insight, only an all-fields model was analyzed further. The same parameters as for the additive method were used to assess the predictive power of the final model. Three types of cross validation were performed.

One difficulty of the application of the CoMSIA on peptide prediction is that peptides size is large compared to small molecules and the diversity of the physico-chemical properties associated with each position being examined. However, good agreement was found between the results generated by other techniques (Doytchinova and Flower, 2001).

CoMSIA can predict the binding affinity of a peptide with an amino acid not presented in the initial training set, but it cannot assess the contribution of each amino acid at each position and the interactions between them. Another advantage of the CoMSIA method is that it returns 3D representations of the analysis for visual investigation, which indicate where adding particular kinds of functionality to the peptides would contribute to activity, either positively or negatively. However, because the method is also

data driven, the predictivity of this method is also dependent on the quality of binding data.

3.2.7 3D-Additive Method

HLAp binding method (Zhao et al., 2003; Kangueane and Sakharkar, 2005) was recently developed using structural information gathered from class-I HLAp crystal structures**.** In this method, nine virtual pockets are defined and the binding affinity between HLA and peptide is given by the sum of residue-residue compatibility between peptide residues and corresponding virtual pockets. The quantification of the interaction between the HLAp residue pair is calculated by the application of the Q matrix, which quantified the interaction between the 20 amino acids based on 237 physico-chemical properties. The prediction method was intensively verified using ROC analysis based on large quantity of HLAp binding data. The method produces high efficiency (average 60%) with good sensitivity (50–73%) and specificity (52–58%), although the accuracy is moderate (60%). The method is simple, effective and most important applicable to all HLA allele whose sequence is clearly defined.

4. Conclusion

The HLAp binding prediction model should be suited for high throughput scanning of a pathogen proteome with high sensitivity capable of covering maximum number of HLA alleles. The method that requires very few experiments in the identification of vaccine candidates is technologically advantageous. In this chapter we discussed the merits and demerits of several tools and techniques such as BIMAS-HLA_BIND, ProPred1, SYFPEITHI, EPIMATRIX, EPIPREDICT, PREDICT, MDS, CCLD, 3D-QSAR, TEPITOPE, FRESNO, EpiDock, and virtual pockets for HLAp binding predictions. We hope that this review provides a comparison of different available methods on this subject. A list of useful tools to predict MHC binding peptides is available in Table #11.1. We conclude that the choice of the tools and their mode of development are critical to their application in immunology and users of such tools should be aware of such limitations. Other coupled parameters such as peptide processing, transport, loading, TCR repertoires, and subsequent immune elucidation factors have to be clearly modeled for appropriate application of HLAp binding prediction models in immuno-therapeutics and vaccine design. The next few years promise many such prediction tools for use in immuno-biology.

References

Adams, H.P. and Koziol, J.A. (1995) Prediction of binding to MHC class I molecules. *J Immunol Methods* **185**(2), 181–90.

Altuvia, Y., Schueler, O., et al. (1995) Ranking potential binding peptides to MHC molecules by a computational threading approach. *J Mol Biol* **249**(2), 244–50.

Altuvia, Y., Sette, A., et al. (1997) A structure-based algorithm to predict potential binding peptides to MHC molecules with hydrophobic binding pockets. *Hum Immunol* **58**(1), 1–11.

Batalia, M.A. and Collins, E.J. (1997) Peptide binding by class I and class II MHC molecules. *Biopolymers* **43**(4), 281–302.

Berman, H.M., Westbrook, J., et al. (2000) The Protein Data Bank. *Nucleic Acids Res* **28**(1), 235–42.

Betancourt, M.R. and Thirumalai, D. (1999) Pair potentials for protein folding: choice of reference states and sensitivity of predicted native states to variations in the interaction schemes. *Protein Sci* **8**(2), 361–9.

Brusic, V., Petrovsky, N., et al. (2002) Prediction of promiscuous peptides that bind HLA class I molecules. *Immunol Cell Biol* **80**(3), 280–5.

Brusic, V., Rudy, G., et al. (1998) Prediction of MHC class II-binding peptides using an evolutionary algorithm and artificial neural network. *Bioinformatics* **14**(2), 121–30.

Cochlovius, B., Stassar, M., et al. (2000) In vitro and in vivo induction of a Th cell response toward peptides of the melanoma-associated glycoprotein 100 protein selected by the TEPITOPE program. *J Immunol* **165**(8), 4731–41.

Collins, E.J., Garboczi, D.N., et al. (1995) The three-dimensional structure of a class I major histocompatibility complex molecule missing the alpha 3 domain of the heavy chain. *Proc Natl Acad Sci U S A* **92**(4), 1218–21.

Cooper, S., Erickson, A.L., et al. (1999) Analysis of a successful immune response against hepatitis C virus. *Immunity* **10**(4), 439–49.

D'Amaro, J., Houbiers, J.G., et al. (1995) A computer program for predicting possible cytotoxic T lymphocyte epitopes based on HLA class I peptide-binding motifs. *Hum Immunol* **43**(1), 13–8.

Disis, M.L., Gralow, J.R., et al. (1996) Peptide-based, but not whole protein, vaccines elicit immunity to HER-2/neu, oncogenic self-protein. *J Immunol* **156**(9), 3151–8.

Donnes, P. and Elofsson, A. (2002) Prediction of MHC class I binding peptides, using SVMHC. *BMC Bioinformatics* **3**, 25.

Doytchinova, I.A., Blythe, M.J., et al. (2002) Additive method for the prediction of protein-peptide binding affinity. Application to the MHC class I molecule HLA-A*0201. *J Proteome Res* **1**(3), 263–72.

Doytchinova, I.A. and Flower, D.R. (2001) Toward the quantitative prediction of T-cell epitopes: coMFA and coMSIA studies of peptides with affinity for the class I MHC molecule HLA-A*0201. *J Med Chem* **44**(22), 3572–81.

Falk, K., Rotzschke, O., et al. (1990) Cellular peptide composition governed by major histocompatibility complex class I molecules. *Nature* **348**(6298), 248–51.

Free, S.M., Jr. and Wilson, J.W. (1964) A Mathematical Contribution To Structure-Activity Studies. *J Med Chem* **53**, 395–9.

Fremont, D.H., Stura, E.A., et al. (1995) Crystal structure of an H-2Kb-ovalbumin peptide complex reveals the interplay of primary and secondary anchor positions in the major histocompatibility complex binding groove. *Proc Natl Acad Sci U S A* **92**(7), 2479–83.

Gribskov, M., McLachlan, A.D., et al. (1987) Profile analysis: detection of distantly related proteins. *Proc Natl Acad Sci U S A* **84**(13), 4355–8.

Gross, D.M., Forsthuber, T., et al. (1998) Identification of LFA-1 as a candidate autoantigen in treatment-resistant Lyme arthritis. *Science* **281**(5377), 703–6.

Gulukota, K., Sidney, J., et al. (1997) Two complementary methods for predicting peptides binding major histocompatibility complex molecules. *J Mol Biol* **267**(5), 1258–67.

Guo, H.C., Jardetzky, T.S., et al. (1992) Different length peptides bind to HLA-Aw68 similarly at their ends but bulge out in the middle. *Nature* **360**(6402), 364–6.

Guo, H.C., Madden, D.R., et al. (1993) Comparison of the P2 specificity pocket in three human histocompatibility antigens: HLA-A*6801, HLA-A*0201, and HLA-B*2705. *Proc Natl Acad Sci U S A* **90**(17), 8053–7.

Hammer, J., Gallazzi, F., et al. (1995) Peptide binding specificity of HLA-DR4 molecules: correlation with rheumatoid arthritis association. *J Exp Med* **181**(5), 1847–55.

Ishioka, G.Y., Fikes, J., et al. (1999) Utilization of MHC class I transgenic mice for development of minigene DNA vaccines encoding multiple HLA-restricted CTL epitopes. *J Immunol* **162**(7), 3915–25.

Iwasaki, A. and Barber, B.H. (1998) Induction by DNA immunization of a protective antitumor cytotoxic T lymphocyte response against a minimal-epitope-expressing tumor. *Cancer Immunol Immunother* **45**(5), 273–9.

Jernigan, R.L. and Bahar, I. (1996) Structure-derived potentials and protein simulations. *Curr Opin Struct Biol* **6**(2), 195–209.

Jones, D.T. and Thornton, J.M. (1996) Potential energy functions for threading. *Curr Opin Struct Biol* **6**(2), 210–6.

Kangueane, P. and Sakharkar, M.K. (2005) T-Epitope Designer: A HLA-peptide binding prediction server. *Bioinformation* **1** (1), 21–24.

Kawashima, I., Hudson, S.J., et al. (1998) The multi-epitope approach for immunotherapy for cancer: identification of several CTL epitopes from various tumor-associated antigens expressed on solid epithelial tumors. *Hum Immunol* **59**(1), 1–14.

Kubo, R.T., Sette, A., et al. (1994) Definition of specific peptide motifs for four major HLA-A alleles. *J Immunol* **152**(8), 3913–24.

Logean, A. and Rognan, D. (2002) Recovery of known T-cell epitopes by computational scanning of a viral genome. *J Comput Aided Mol Des* **16**(4), 229–43.

Logean, A., Sette, A., et al. (2001) Customized versus universal scoring functions: application to class I MHC-peptide binding free energy predictions. *Bioorg Med Chem Lett* **11**(5), 675–9.

Madden, D.R., Garboczi, D.N., et al. (1993) The antigenic identity of peptide-MHC complexes: a comparison of the conformations of five viral peptides presented by HLA-A2. *Cell* **75**(4), 693–708.

Madden, D.R., Gorga, J.C., et al. (1992) The three-dimensional structure of HLA-B27 at 2.1 A resolution suggests a general mechanism for tight peptide binding to MHC. *Cell* **70**(6), 1035–48.

Mallios, R.R. (1999) Class II MHC quantitative binding motifs derived from a large molecular database with a versatile iterative stepwise discriminant analysis meta-algorithm. *Bioinformatics* **15**(6), 432–9.

Mamitsuka, H. (1998) Predicting peptides that bind to MHC molecules using supervised learning of hidden Markov models. *Proteins* **33**(4), 460–74.

Meister, G.E., Roberts, C.G., et al. (1995) Two novel T cell epitope prediction algorithms based on MHC-binding motifs; comparison of predicted and published epitopes from Mycobacterium tuberculosis and HIV protein sequences. *Vaccine* **13**(6), 581–91.

Milik, M., Sauer, D., et al. (1998) Application of an artificial neural network to predict specific class I MHC binding peptide sequences. *Nat Biotechnol* **16**(8), 753–6.

Miyazawa, S. and Jernigan, R.L. (1985) Estimation of effective inter-residue contact energies from protein crystal structure, quasi-chemical approximation. *Macromolecules* **18**, 534.

Miyazawa, S. and Jernigan, R.L. (1996) Residue-residue potentials with a favorable contact pair term and an unfavorable high packing density term, for simulation and threading. *J Mol Biol* **256**(3), 623–44.

Morgan, D.J., Kreuwel, H.T., et al. (1998) Activation of low avidity CTL specific for a self epitope results in tumor rejection but not autoimmunity. *J Immunol* **160**(2), 643–51.

Noguchi, H., Kato, R., et al. (2002) Hidden Markov model-based prediction of antigenic peptides that interact with MHC class II molecules. *J Biosci Bioeng* **94**(3), 264–70.

Pamer, E. and Cresswell, P. (1998) Mechanisms of MHC class I--restricted antigen processing. *Annu Rev Immunol* **16**, 323–58.

Parker, K.C., Bednarek, M.A., et al. (1994) Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. *J Immunol* **152**(1), 163–75.

Rammensee, H., Bachmann, J., et al. (1999) SYFPEITHI: database for MHC ligands and peptide motifs. *Immunogenetics* **50**(3-4), 213–9.

Rammensee, H.G., Friede, T., et al. (1995) MHC ligands and peptide motifs: first listing. *Immunogenetics* **41**(4), 178–228.

Reche, P.A., Glutting, J.P., et al. (2002) Prediction of MHC class I binding peptides using profile motifs. *Hum Immunol* **63**(9), 701–9.

Robinson, J., Waller, M.J., et al. (2003) IMGT/HLA and IMGT/MHC: sequence databases for the study of the major histocompatibility complex. *Nucleic Acids Res* **31**(1), 311–4.

Rognan, D., Lauemoller, S.L., et al. (1999) Predicting binding affinities of protein ligands from three-dimensional models: application to peptide binding to class I major histocompatibility proteins. *J Med Chem* **42**(22), 4650–8.

Rognan, D., Scapozza, L., et al. (1994) Molecular dynamics simulation of MHC-peptide complexes as a tool for predicting potential T cell epitopes. *Biochemistry* **33**(38), 11476–85.

Ruppert, J., Sidney, J., et al. (1993) Prominent role of secondary anchor residues in peptide binding to HLA-A2.1 molecules. *Cell* **74**(5), 929–37.

Saper, M.A., Bjorkman, P.J., et al. (1991) Refined structure of the human histocompatibility antigen HLA-A2 at 2.6 A resolution. *J Mol Biol* **219**(2), 277–319.

Sarobe, P., Pendleton, C.D., et al. (1998) Enhanced in vitro potency and in vivo immunogenicity of a CTL epitope from hepatitis C virus core protein following amino acid replacement at secondary HLA-A2.1 binding positions. *J Clin Invest* **102**(6), 1239–48.

Schueler-Furman, O., Altuvia, Y., et al. (2000) Structure-based prediction of binding peptides to MHC class I molecules: application to a broad range of MHC alleles. *Protein Sci* **9**(9), 1838–46.

Schueler-Furman, O., Elber, R., et al. (1998) Knowledge-based structure prediction of MHC class I bound peptides: a study of 23 complexes. *Fold Des* **3**(6), 549–64.

Sette, A., Buus, S., et al. (1989) Prediction of major histocompatibility complex binding regions of protein antigens by sequence pattern analysis. *Proc Natl Acad Sci U S A* **86**(9), 3296–300.

Sette, A., Vitiello, A., et al. (1994) The relationship between class I binding affinity and immunogenicity of potential cytotoxic T cell epitopes. *J Immunol* **153**(12), 5586–92.

Skolnick, J., Jaroszewski, L., et al. (1997) Derivation and testing of pair potentials for protein folding. When is the quasichemical approximation correct? *Protein Sci* **6**(3), 676–88.

Stassar, M.J., Raddrizzani, L., et al. (2001) T-helper cell-response to MHC class II-binding peptides of the renal cell carcinoma-associated antigen RAGE-1. *Immunobiology* **203**(5), 743–55.

Stryhn, A., Pedersen, L.O., et al. (1996) Peptide binding specificity of major histocompatibility complex class I resolved into an array of apparently independent subspecificities: quantitation by peptide libraries and improved prediction of binding. *Eur J Immunol* **26**(8), 1911–8.

Sturniolo, T., Bono, E., et al. (1999) Generation of tissue-specific and promiscuous HLA ligand databases using DNA microarrays and virtual HLA class II matrices. *Nat Biotechnol* **17**(6), 555–61.

Thompson, J.D., Higgins, D.G., et al. (1994) Improved sensitivity of profile searches through the use of sequence weights and gap excision. *Comput Appl Biosci* **10**(1), 19–29.

Townsend, A.R., Rothbard, J., et al. (2006) The epitopes of influenza nucleoprotein recognized by cytotoxic T lymphocytes can be defined with short synthetic peptides. 1986. *J Immunol* **176**(9), 5141–50.

Udaka, K., Wiesmuller, K.H., et al. (2000) An automated prediction of MHC class I-binding peptides based on positional scanning with peptide libraries. *Immunogenetics* **51**(10), 816–28.

van der Burg, S.H., Visseren, M.J., et al. (1996) Immunogenicity of peptides bound to MHC class I molecules depends on the MHC-peptide complex stability. *J Immunol* **156**(9), 3308–14.

Viret, C. and Janeway, C.A., Jr. (1999) MHC and T cell development. *Rev Immunogenet* **1**(1), 91–104.

Zeng, J., Treutlein, H.R., et al. (2001) Predicting sequences and structures of MHC-binding peptides: a computational combinatorial approach. *J Comput Aided Mol Des* **15**, 573.

Zhao, B., Mathura, V.S., et al. (2003) A novel MHCp binding prediction model. *Hum Immunol* **64**(12), 1123–43.

Chapter 12

Bioinformatics Application: Eukaryotic Gene Count and Evolution

Meena K. Sakharkar[1] and Pandjassarame Kangueane[2]

[1]*Nanyang Technological University, Singapore*
[2]*Biomedical Informatics, India*

Abstract: In this chapter, we describe the comparison of gene numbers in different eukaryotic genomes. Unlike prokaryotes, eukaryotic genes are often split into exons (coding sequence segments) and introns (non-coding sequence segments). However, the number of exons and introns vary in different genes across diverse genome species. It is found that the intron number varies from 0 to >100 in different eukaryotic genes. This results in SEG (Single exon genes) and MEG (multi exon genes). Thus, SEG have 0 intron and MEG have at least one intron. Consequently, we compared the SEG and MEG fraction across different eukaryotic genomes. The comparison helped to discuss the evolutionary selection of SEG and MEG fraction in eukaryotic genomes.

Key words: Introns, Genes, Exons, Eukaryotes

1. Introduction

The number of genes in eukaryotes varies between genomes. Eukaryotic genes are broadly classified into intron (non coding sequence segment) bearing MEG (multi exon genes) and intronless SEG (single exon genes). The ExInt (Sakharkar et al., 2000) database contains MEG sequences and the SEGE (Sakharkar et al., 2002) database contains SEG sequences from GenBank (Benson et al., 2000). SEG sequences are uninterrupted by introns and are similar to prokaryotic genes in structure. Recently, SEG sequences were derived from completely sequenced eukaryotic genomes led to the development of Genome SEGE (Sakharkar et al., 2004). The differential selection of SEG and MEG in eukaryotic genomes is interesting. Hence, it is important to study the origin and evolution of SEG and MEG in eukaryotes.

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_12,

To study their evolution on a genomic scale is resource intensive, information demanding and extremely complex. We undertook the first initiative of identifying and counting SEG and MEG for nine eukaryotic genomes. Here, we compare the proportional selection of SEG and MEG in different genomes.

2. Methodology

2.1 Identification of SEG

GenBank format files in genome banks were used to create a dataset containing entries that are reservedly considered as "single exonic" genes according to the CDS FEATURE convention. By definition, we consider an entry to be putatively "single exonic" in gene structure if it contains the following description patterns in the corresponding GenBank lines.

- Contain the word "DNA" in the LOCUS line at positions 48–53 as per the new locus line format.
- Contain the pattern "CDS" in the FEATURES.

The "CDS" line in the FEATURES should contain a continuous span of bases indicated by the number of the first and the last bases in the range separated by two periods (e.g. 23..78). If symbols "<" or ">" are indicated at the end points of the range, the entry is discarded because the range is beyond specified base number in such cases. When operators such as "complement (location)" are used in the "CDS" line, the feature is read as complementary to the location indicated and, therefore, the complementary strands are read from 5' to 3'.

2.2 Identification of MEG

GenBank format files in genome banks (ftp://ftp.ncbi.nih.gov/genomes) were used to create a dataset containing entries that are reservedly considered as ''Multi exonic'' genes according to the "CDS" FEATURE convention. By definition, we consider an entry to be putatively ''Multi exonic'' in gene structure if it contains the following description patterns in the corresponding GenBank lines.

1. Contain the word “DNA” in the LOCUS line at positions 48–53 as per the new locus line format.
2. Contain the pattern “CDS” in the FEATURES.
3. The “CDS” line in the FEATURES should contain `join` followed by a continuous span of bases indicated by the number of the first and the last bases for each exon in the range separated by two periods within parenthesis (e.g. join(23..78,123..180)). The exons are separated by `,`. If symbols “<” or “>” are indicated at the end points of the range, the entry is discarded because the range is beyond specified base number in such cases. When operators such as “join(complement(location))” are used in the “CDS” line, the feature is read as complementary to the location indicated and, therefore, the complementary strands are read from 5’ to 3’.

2.3 Pseudogenes

Data processing and cleaning is an essential part of biological knowledge discovery. Hence, we eliminated all identifiable processed pseudogenes by scanning for polyadenylation signal (AATAAA) and polyadenylation tail using a modified procedure of Harrison and colleagues (Harrison et al., 2002). In this procedure, by definition, we consider a sequence to represent a pseudogene if it contains a polyadenylation tail (>15A) within 1000 nucleotides from the stop codon with a preceding polyadenylation signal.

2.4 Caveats

Genome annotation is an inherently dynamic process in which it is necessary to use many different sources of data, which are not updated in a rigorous fashion. It should also be noted that annotation is not generally uniform and consistent because various procedures are used by different groups for genome annotation. During genome annotation, a gene may have been annotated with a SEG or MEG CDS in the FEATURE for three main reasons: (1) the gene is truly SEG or MEG, (2) SEG is of retroposition origin (Fink, 1987; Brosius, 1999), (3) false positive prediction by gene finding algorithms. False positives are not removed from the current dataset due to lack of a methodology. Nevertheless, the gene finding algorithms are reasonably optimized to find SEG and MEG.

It should also be noted that our approach does not include a small fraction of eukaryotic SEG and MEG that do not follow the “CDS” feature convention. We also do not consider entries that are annotated as NA, RNA, mRNA, tRNA, rRNA, uRNA, snRNA, or snoRNA in the LOCUS line at positions 48–53, and in the UTR (un-translated regions of the genome).

2.5 Total Genes

The sum of SEG and MEG counts is considered as the total gene count in each genome for this analysis.

3. Results and Discussion

3.1 Utility of SEG and MEG Sequences to the Study of Evolution

The proportions of MEG and SEG in eukaryotes complement each other in different species. The varying proportion is related to the degree of genome complexity. The subtle interplay between their proportions might aid in efficient genome organization during evolution. A wealth of information can be obtained by comparing MEG and SEG sequences between two or more genomes to identify features conserved or diverged during evolution. Comparison of more closely related genomes can reveal similarities in gene order. Such analysis could also shed light on genome architecture and help understand why the genome is arranged the way it is and how its structure affects function. A systematic mapping between functional genes and their SEG/MEG paralogs can provide a matrix for genomic rearrangement and gene duplication. Different SEG/MEG gene sets available in the ExInt (Sakharkar et al., 2000), SEGE (Sakharkar et al., 2002), and Genome SEGE (Sakharkar et al., 2004) databases will provide an opportunity to perform many–to–many comparison between genomes. Such analysis will provide information on paralogy and orthology at a molecular level. Analysis of the datasets using non-linear probabilistic models may provide acceptable evidence for molecular evolution of SEG and MEG.

3.2 Selection of SEG and MEG in Different Eukaryotic Genomes

Table 12.1 shows total gene, SEG, and MEG count in nine eukaryotic genomes. Data shows MEG and SEG complement each other in each other. The differences reflect inherent variations in different genome architectures and evolutionary divergences. Although, this trend is not surprising, the actual estimates are interesting in the sense that their proportions in some genomes are distinctly greater than others. Mere comparisons of these counts provide valuable insight towards genome selection. We note from Table 12.1

Table 12.1 An estimate of total gene count, SEG, and MEG count in different eukaryotic genomes is given. The SEG fraction is defined as the percentage ratio of SEG count and gene count. Pseudo = processed pseudo genes that are SEG. SEG (a) = SEG count after eliminating processed pseudo genes. SEG and MEG fraction adds up to 100 in each genome.

Genomes	Size (Mb)	Genes #	Single Exon Genes (SEG)			SEG (%)	MEG (%)
			Total	Pseudo	SEG		
E. cuniculi	2.9	2,028	1,981	0	1,981	97.7	2.3
S. cerevisiae	12.1	6,004	5,551	60	5,491	92.5	7.5
P. falciparum	23	5,544	2,471	991	1,480	44.6	55.4
S. pombe	13.8	5,213	2,585	17	2,468	49.6	50.4
C. elegans	97	24,607	654	3	651	2.7	97.3
A. thaliana	125	29,483	5,920	84	5,836	20.1	79.9
D. melanogaster	180	11,357	2,049	29	2,020	18.0	82.0
M. musculus	2500	26,771	4,218	105	4,113	15.8	84.2
H. sapiens	2900	27,675	3,408	103	3,305	12.3	87.7

that unicellular (45–98%) and multi-cellular (3–20%) genomes are distinguished by SEG proportion in them. Generally, the SEG fraction is greater in unicellular than multi-cellular genomes. This implies that unicellular genomes with very short generation times have larger fraction, while multi-cellular genomes with long generation times have smaller fraction. The Pearson correlation co-efficient (r) between SEG count and genome size is 0.2. This is much weaker than the Pearson correlation co-efficient between total gene count and genome size (r = 0.61). The r value between SEG count and gene count is 0.3. However, the r value between SEG fraction and genome size is –0.45. This suggests that SEG fraction decreases with genome size. Interestingly, the r value between SEG fraction and gene count is –0.80 (Figure 12.1). Thus, SEG fraction strongly decreases with total gene count in these genomes. In other words, genomes with high gene count contain low SEG fraction. We also found that the r value between SEG fraction and gene density (total gene count/Mb genome size) is 0.88. This relationship is strong and SEG fraction increases linearly with increase in gene density in these genomes (Figure 12.2). These patterns are very interesting and subsequent analysis is required to gain further insight into their selection and genome design. However, the bits and pieces of derived information have to be bridged together to signify the trend between SEG fraction and genome content. We hope to compare and contrast estimates from different genomes of distant phylogeny.

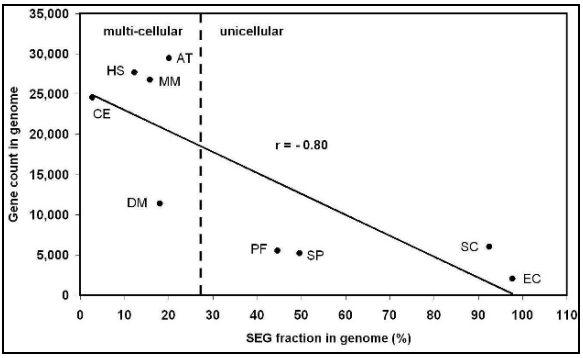


Figure 12.1 Relationship between SEG fraction and gene count is given.

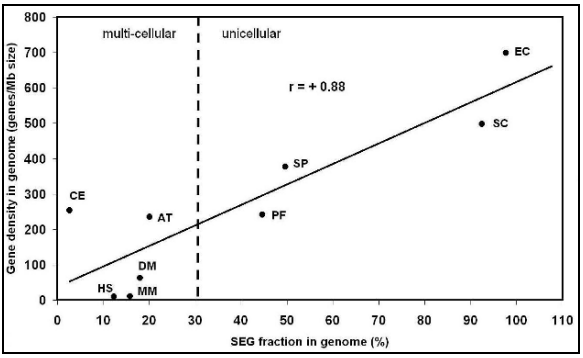


Figure 12.2 Relationship between SEG fraction and gene density is given. EC = E. cuniculi, SC = S. cerevisiae, SP = S. pombe, PF = P. falciparum, CE = C. elegans, AT = A. thaliana, DM = D. melanogaster, MM = M. musculus, HS = H. sapiens. The MEG fraction complements the SEG fraction for each genome.

3.3 Mechanism of SEG Origin

Table 12.2 shows that multi-cellular genomes contain about 12–20% SEG. This is not true for *C. elegans* and it contains only 2.7% SEG. The latest update (October, 2003) of the human genome contains 3,408 SEG sequences (about 12% of total genes). These estimates are relatively large and their mere existence in many intron-rich genomes demands further investigations. It has been suggested that a significant fraction of human SEG have been generated by retro-transposition (Brosius,1999). Therefore, the presence of SEG can be explained by the mechanism of retro-position. This occurs by homologous recombination between the genomic copy of a gene and an intronless cDNA (Fink, 1987). The later is produced by reverse transcription of the corresponding mRNA, a mechanism that produces SEG genes in eukaryotes. In an independent experiment by sequence comparison, we

Table 12.2 URLs of Intron and Exon databases

Database Name	URL
ExInt	http://sege.ntu.edu.sg/wester/exint/
SEGE	http://sege.ntu.edu.sg/wester/sege/
Genome SEGE	http://sege.ntu.edu.sg/wester/intronless/
Human SEGE	http://sege.ntu.edu.sg/wester/intronless/human

found that about 20% (366) of unique SEG (purged at 40% sequence identity) show (MEG) correspondence with at least 40% sequence identity (data not shown). This strongly supports the hypothesis that human SEG arose by retro-transposition.

The human genome team suggested that a very small fraction of total human genes (<1%) is exclusively homologous to bacterial genes (Lander et al., 2001). Therefore, we compared human SEG with 430,011 prokaryotic protein sequences derived from 135 prokaryotic genomes. About 99% of human SEG lack homology with prokaryotic sequences. This suggests that human SEG did not evolve by gene transfer from bacteria to human. Nonetheless, the absence of homology between human SEG and prokaryotic proteins supports the hypothesis that SEG probably arose by retro-position. Additional data on paralogous SEG may provide further evidence towards the possible mechanism of their origin by retro-position.

4. Conclusion

The differential selection of SEG and MEG in the genomes of higher organism is perplexing. Different eukaryotic genomes have varying proportions of SEG and MEG, and a sizeable fraction of SEG are found in many intron-rich multi-cellular genomes. We believe that these estimates will improve our understanding on the differential selection (as a process or force) of SEG and MEG in different eukaryotic genomes. The biological role of SEG and MEG in the genomes of higher organism is not completely understood. Here, we show that different eukaryotic genomes have varying SEG and MEG fraction, and a sizeable portion of SEG is found in many intron-rich multi-cellular genomes. This report provides an overview of SEG and MEG count and fraction. This shows their relationship to genome size, gene count, and gene density. It is also interesting to note that a large proportion of SEG are associated with unicellular organisms with very short generation times, while a small proportion of SEG is common in relatively complex multi-cellular organisms with long generation times. We hope that these estimates will help to probe into the biological role of SEG and MEG towards genome design.

References

Benson, D.A., Karsch-Mizrachi, I., et al. (2000) GenBank. *Nucleic Acids Res* **28**(1), 15–8.

Brosius, J. (1999) Many G-protein-coupled receptors are encoded by retrogenes. *Trends Genet* **15**(8), 304–5.

Fink, G.R. (1987) Pseudogenes in yeast? *Cell* **49**(1), 5–6.

Harrison, P.M., Hegyi, H., et al. (2002) Molecular fossils in the human genome: identification and analysis of the pseudogenes in chromosomes 21 and 22. *Genome Res* **12**(2), 272–80.

Lander, E.S., Linton, L.M., et al. (2001) Initial sequencing and analysis of the human genome. *Nature* **409**(6822), 860–921.

Sakharkar, M., Long, M., et al. (2000) ExInt: an Exon/Intron database. *Nucleic Acids Res* **28**(1), 191–2.

Sakharkar, M.K. and Kangueane, P. (2004) Genome SEGE: a database for 'intronless' genes in eukaryotic genomes. *BMC Bioinformatics* **5**, 67.

Sakharkar, M.K., Kangueane, P., et al. (2002) SEGE: A database on 'intron less/single exonic' genes from eukaryotes. *Bioinformatics* **18**(9), 1266–7.

Chapter 13

Bioinformatics Application: Predicting Protein Subcellular Localization by Applying Machine Learning

Pingzhao Hu[1,2], Clement Chung[1], Hui Jiang[2], and Andrew Emili[1]

[1]*Program in Proteomics and Bioinformatics, Banting and Best Department of Medical Research, University of Toronto, Toronto, Ontario, Canada M5G 1L6*
[2]*Department of Computer Science, York University, Toronto, Ontario, Canada M3J IP3*

Abstract: The subcellular localization of a protein is closely correlated with its function. Automatic prediction of subcellular localization based on protein sequence properties remains a challenging problem. Here, we propose a proteomic screening-based machine learning approach for interpreting differential detection of proteins in isolated organellar compartments by high-throughput mass spectrometry. The method deals with some core limitations existing in previous approaches, such as multi-compartmental ambiguity. When applied to a global-scale proteomic study, our method achieved an excellent overall accuracy of 80.5% and precision 75.1% for four major organellar compartments (cytosol, membranes, mitochondria, and nucleus). The classifiers were able to predict the subcellular localization of 2390 previously uncharacterized proteins, 1370 of which were assigned to one or more compartments with at least 80% confidence.

Key words: Subcellular localization, Multi-compartment, Proteomics, Protein expression profiling, Machine learning, Automatic prediction

1. Introduction

Determining the subcellular localization of a protein in a cell is a key to understanding its function and can facilitate biochemical experiments aimed at characterizing additional biological properties, such as purification. However, traditional experimental methods for examining subcellular localization are generally time-consuming and costly, and are currently not practical on a genome-wide scale. Given the rapidly expanding plethora of

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_13,

uncharacterized proteins identified by the many ongoing genome-sequencing projects, it is highly desirable to predict a protein's subcellular localization automatically (Lu et al., 2004). Currently, most of the automatic protein subcellular localization prediction methods fall into one of three categories (Scott et al., 2004). The first one is prediction based on amino acid composition, as originally suggested by Nakashima and Nishikawa (1994). Different machine learning algorithms have been developed that make use amino acid composition information towards this end, including neural networks (Reinhardt and Hubbard, 1998), support vector machines (SVM) (Hua and Sun, 2001), covariant discrimination (Chou and Elrod, 1998) and augmented covariant discrimination methods (Chou, 2000) as well as SVM incorporating quasi-sequence-order effects (Cai et al., 2002). The second major approach is prediction based on calculating a set of sequence-derived parameters and comparing these with a representation of a number of localization rules that have been collated from the literature. The most widely used algorithm in this category is the popular PSORT algorithm (Nakai and Kanehisa, 1992), which is a commonly-used bioinformatics tool. The key idea of this approach is to decide the presence of various sequence motifs that enable proteins to be localized to a certain compartment. Different types of prior knowledge are required for this determination, which are, actually, hard to get for uncharacterized proteins. The third category of prediction is the homology-based prediction (Mott et al., 2002; Chou and Cai, 2005; Lu et al., 2004), wherein the inferences are based on transference of knowledge from characterized to unknown homologous proteins.

One of main limitations in most of these studies is that their principle methods focus on mono-compartment prediction (that is, a protein is presumed to localize to a single organelle only). For example Lu et al. (2004) constructed a parser to extract a simple ontological representation for proteins assigned to multiple compartments, without exploiting the information encoded by multi-localizations. Similarly, while Scott et al. (2004) built a Bayesian-based tool for subcellular localization prediction which can integrate multi-source information to assign a protein to multiple compartments, essentially it does not consider or exploit the multi-compartment issue during the building of the predictors.

As an alternate to sequence- or homology-based predictions, proteomic methods based on subcellular fractionation in combination with high-throughput protein mass spectrometry have emerged as a powerful alternative experimental platform for assessing subcellular localization directly. Indeed, substantive recent technical advances now make this the preferred approach for genome-wide protein identification and quantification with high sensitivity and accuracy (Yates, 2004). Compared with previous sequence information-derived prediction methods, these newer proteomic

profiling-based screening methods are also proving to be more effective for resolving ambiguous or difficult localization problems (Schirmer et al., 2005). However, current procedures involving biochemical methods for subcellular fractionation are still far from perfect, and artifacts due to cross-contamination can create misleading results.

The present study not only is devoted to addressing the multi-compartment problem, but also provides a new strategy for automatic prediction of protein's subcellular localizations based on differential detection of proteins in isolated organellar compartments by high-throughput mass spectrometry.

2. Methods

2.1 Data Sets and Preprocessing

In this global-scale mouse proteomic study, healthy adult brain, heart, kidney, liver, lung, and embryonic placenta were excised from euthanized 6–8 week old ICR female mice. The tissues were gently disrupted and fractionated into four major subcellular compartments (cytosol, microsomes, mitochondria, and nuclei) using differential ultracentrifugation. The proteins were identified by tandem mass spectrometry followed by database searches of the acquired spectra using the multidimensional protein identification technology (MudPIT) (Yates, 2004). The procedures for processing, searching, and rigorously evaluating the proteomic expression profiles have been detailed by Kislinger and Emili (2003) and Kislinger et al. (2003). A total of 4768 proteins were confidently identified in this analysis. We estimated protein relative abundance in the respective fractions based on the ratio of the cumulative number of spectra matching to any given protein in each sample (Liu et al., 2004). The experimental variance in the recorded protein expression levels proved to be quite large, making interpretation of the data more difficult. Hence, we normalized the data to have a standardized mean of zero and variance of one across each subcellular compartment.

In order to generate a suitable supervised learning approach for predicting protein subcellular localizations, we needed to obtain a reference set of proteins with known subcellular localizations. For this, we obtained the annotations for 1558 proteins from the SWISS-PROT database (http://ca.expasy.org/sprot/). These proteins were used to construct training and testing sets. Additionally, we compiled an independent test set of 820 proteins that had been independently identified in a single highly purified organelle in a previous proteomic study (Andersen et al., 2005; Beausoleil et al., 1997; Krapfenbauer et al., 2003; Mootha et al., 2003; Nielsen et al.,

Table 13.1 Number of proteins per subcellular location

Subcellular localization	No. Seq. used in training	Number used in tests	Number of predictions
Cytosol	672	69	2390
Membranes	769	283	
Mitochondria	188	217	
Nucleus	570	251	
Total # of compartment specific proteins	1558	820	
Total # of multi-labeled proteins	641	0	

2005; Schirmer et al., 2005; Wu et al., 2003; Wu et al., 2004). All of the remaining uncharacterized proteins without labels in SWISS-PROT or not belonging to the gold-standard test set were used for prediction. Table 13.1 show a summary of the number of proteins per subcellular compartment used for training, test, and prediction.

As we can see from the table, more than one-third of training proteins are multi-labeled. Since there are no good measures to evaluate classifiers trained on multi-labeled data, we removed all the multi-labeled proteins from gold standard test data to training data so that we kept all the test data as single-labeled.

2.2 Learning Algorithm

Many learning algorithms, such as K-nearest neighbors (KNN) (Huang and Li, 2004; Cai and Chou, 2004), support vector machines (SVM) (Park and Kanehisa, 2003), and Bayesian methods (Lu et al., 2004; Scott et al., 2004), have been used for subcellular localization prediction. Dudoit et al. (2002) have reported an extensive comparison of the effectiveness of different supervised statistical learning methods for cancer classification using genomic data like gene expression profiles measured by DNA microarrays. They demonstrated that the simpler methods, such as KNN, often produce better results using a number of performance measures. In this study, we evaluated some advanced learning algorithms, such as SVM, but likewise in our hands these methods did not outperform the simple methods such as KNN (P.H.; unpublished observations). Therefore, we applied the KNN learning method to our proteomic datasets to better infer subcellular localization. KNN is a supervised non-parametric learning algorithm (Ripley, 1996; Hastie et al., 2001). Given a protein of unknown or uncertain subcellular localization, the algorithm finds the KNN in the training set based on minimum distance (in Euclidean distance, as used by Dudoit et al. (2002) of the target protein to the reference training set and assigns a subcellular localization according to a majority vote based on the subcellular localizations of these K neighbors. The optimal number of neighbors (K) used in generating the classifier is chosen by

the standard procedure of cross-validation. That is, for a given training set, the performance of the KNN for a set of K is determined by cross-validation, and the K that produces the smallest error is used. The prediction confidence (probability) is then determined based on the proportion of votes for the preferred subcellular localization.

2.3 Evaluating Performance of the Learning Algorithm

We rigorously evaluated the performance of our machine learning algorithm in two different ways. The first approach involved the standard method of 10-fold cross-validation (Mitchell, 1997). In this procedure, we randomly divided the training set associated with each subcellular compartment into 10 sub-groups (G_1 , G_2 , …, G_{10}), keeping the number of proteins in the localization class approximately the same across each training category. We then constructed 10 different classifiers (C_1, C_2, …, C_{10}), where C_i use all of the training proteins from all of the groups except G_i . Proteins in group G_i were used for testing classifier C_i . The second way, and more stringent means of assessing classifier performance was based on an independent test set, using a gold standard reference dataset to evaluate the classifiers built with all of the training data.

For each of the two methods, we used the following statistical terms (Hastie et al., 2001) to assess performance.

Accuracy: the rate of correct predictions compared to all predictions for a given subcellular localization ((TP+TN)/(TP+FN+TN+FP)).
Precision: The portion of true positive with respect all predicted positive for a given subcellular localization (TP/(TP+FP)),

Where TP, FP, TN, FN denote the total number of true positives, false positives, true negatives, and false negatives, respectively. We also defined overall classifier accuracy and precision as the average accuracy and precision calculated for each of the four subcellular localizations, together with a measure of classifier sensitivity (=TP/(TP+FN)) and specificity (=TN/(TN+FP)).

2.4 Strategy for Multi-class/Multi-label Classification

One of the main objectives of this study was to confidently assign at least one subcellular localization to each uncharacterized protein based on the properties learned from the proteomic profiles of a set of proteins with known subcellular localizations, which can be formulated to be a multi-class supervised classification problem. A common approach for dealing with multiple classes is to transform the multi-class learning problem into a set of binary classification problems, which is also known as a "one-against-

others" method (Yeang et al., 2001). For the binary classification, the proteins are associated with a specified subcellular localization are labeled as positive and all others as negative. As shown in Table 13.1, many of the proteins in our training set were annotated in several different subcellular localizations. So far, there are no effective computational procedures that can be used to treat this difficult multiplex (i.e., multi-label, multi-localization) problem (Chou and Cai, 2005). Indeed, in previous studies of proteins subcellular localization (Park and Kanehisa, 2003; Huang and Li, 2004; Cai and Chou, 2004 and Lu et al., 2004), this multiplex challenge was not directly considered. As a first step towards resolving this, we applied a method called "cross-training" (Boutell et al., 2004), which has been applied with some success as a means of rationalizing pattern recognition as applied to multi-label semantic scene classification. In our implementation of this approach, we used the multi-labeled proteins as positive examples for each of the four associated localization classes during training. For example, if a protein was annotated as both nuclear and mitochondrial, it was considered as a positive example during training of both the nuclear and mitochondrial classes, but never as a negative example of either category.

2.5 Optimal Sampling Methods for Imbalanced Data Sets

When we applied the so-called "one-against-others" method to deal with multi-label classification, another serious problem emerged in that the positive examples of a subcellular localization tend to be under-represented relative to the far larger number of proteins (negative examples) in the other compartment classes (alternate organelles). A recent compelling analysis by Weiss and Provost (2003) concluded that the natural class distribution is generally not the best distribution for learning a classifier. Indeed, the excess of negative examples in the training dataset poses several pitfalls for classical machine learning systems. These limitations include that we will build either trivial classifiers that completely ignore the minority class or classifiers with many small (specific) disjunctions that tend to overfit the training samples. Recently, some attempts have been proposed in the machine learning community to overcome imbalanced training data set during binary classification. The newer methods are primarily focused on optimizing sampling over the training examples, and involve either (i) under-sampling – reducing the negative class by randomly removing a subset of the negative examples from the training set, or (ii) over-sampling – increasing the positive class by replicating the positive examples. Unfortunately, over-sampling with replication does not always improve the effectiveness of minority (positive) class prediction. This deficiency is due to the classifier becoming very specific in the minority class decision region,

leading to over-fitting of the examples. In contrast, the under-sampling approach forces the learning algorithm to focus on different degrees of the class distribution while at the same time increasing the presence of the minority class in the training examples, which can lead to the generation of a more robust classifier. Therefore, we opted for the under-sampling method, which is also known as asymmetric bagging strategy (Tao and Tang, 2004), to deal with this data imbalance problem. The bagging strategy incorporates the benefits of both bootstrapping (i.e. repeat random sampling of the training samples) and aggregation (i.e. combine the classifiers trained on bootstrap samples). Multiple classifiers are generated by training on multiple sets of samples produced by bootstrapping. Aggregation of the generated classifiers can then be implemented by majority voting rule. Experimental and theoretical results have shown that bagging can improve the performance of a good but unstable classifier significantly (Breiman, 1996). However, directly using the bagging procedure for protein subcellular localization prediction was not appropriate since we had only a relatively small number of positive examples. Since there are far more negative samples than the positive samples, we applied the asymmetric bagging strategy, which executes bootstrapping only on the negative samples to overcome this limitation. In this way, each generated classifier will be trained on a more balanced number of positive and negative samples.

2.6 Algorithm of Asymmetric Bagging Strategy

A starting assumption is that there are a set number of classes that define a proteins possible subcellular localization. When building a training model for a given localization class, we treated the examples (proteins) belonging to that class as positive training set, S+, and associated all others as the negative training set, S-. However, the multi-labeled proteins (that is, those that were linked to more than one subcellular localization in the literature) were considered only in the positive training set. The algorithm is described in Table 13.2.

Table 13.2 Algorithm of asymmetric bagging KNN

Input: positive training set S^+, negative training set S^-, weak classifier I (KNN), integer T (number of generated classifiers based on bootstrap samples), and x is the test protein.

(1) For i=1 to T {
(2) S_i^- =bootstrap samples from S-, with $|S_i^-| = |S^+|$.
(3) C_i =I(S_i^-, S^+)
(4) }
(5) C*(x)=aggregation { C_i(x, S_i^-, S^+), $1 \le i \le T$ }. The aggregation is based on majority voting rule.

Output: classifier C*

3. Results

We derived an optimal K (the number of neighbors) based on 10-fold cross-validation using the training set for a set of values of K (K=1, 2 , …, 20). A value of K=8 produced the highest overall accuracy and precision. Table 13.3 shows the 10-fold cross validation results for training data using 8NN learning method. As can be seen, the combined overall prediction accuracy and precision of the learning approach was 80.4% and 75.4%, respectively. The precisions of the four subcellular localizations were quite similar, although the accuracies associated with each compartment were more varied. For example, the accuracy of cytosolic predictions was only 72.7%, whereas that of mitochondria was more impressive at 92.0%.

Table 13.4 shows the test results based on the independent test set of gold standard reference proteins using the same 8NN learning method. Although the overall prediction accuracy and precision was nearly identical to that achieved by 10-fold cross-validation of the training set, it seems that the classifiers trained on all training data exhibited relatively poor performance for the cytosol and membrane fractions. As shown in Table 13.1, the method generated 641 multi-labeled proteins from the training set in that these proteins were assigned to at least two subcellular localizations. The statistics shown in Table 13.5 indicate that more than half of these multi-labeled proteins are linked to cytosol and membranes. Therefore, it is very possible that the ineffective performance is due in large part to incomplete resolution of the multi-labeled training problem.

Table 13.3 10-fold cross-validation performance on training data

Subcellular location	Precision	Accuracy
Cytosol	72.9	72.7
Membranes	75.1	76.1
Mitochondria	75.6	92.0
Nucleus	78.1	80.6
Overall Performance	75.4	80.4

Table 13.4 Prediction performance on testing data

Subcellular localization	Precision (%)	Accuracy (%)
Cytosol	63.3	92.6
Microsomes	52.7	66.2
Mitochondria	91.8	82.2
Nucleus	92.7	80.9
Overall Performance	75.1	80.5

Table 13.5 Single/multi-labeled training set for each subcellualr localization

Subcellular localization	Number of training	Number of single labeled	Number of multi-labeled
Cytosol	672	232	440
Microsomes	769	357	412
Mitochondria	188	36	152
Nucleus	570	353	217

As a last measure to performance, we also explored the use of Receiver Operating Characteristics (ROC) curves to evaluate the power of different classifiers for predicting protein subcellular localization. ROC curves have been used to depict the pattern of sensitivity and specificity observed when the performance of a classifier is evaluated at different thresholds (Bradley, 1997). Since the prediction confidence (probability) from 8NN classifiers varies between zero and one, we created 100 thresholds of equal interval across the range of prediction confidence. For each of the 100 thresholds, we calculated classifier specificity, sensitivity, accuracy, and precision based on the gold standard independent test data. Overall, the classifiers of cytosol, mitochondria, and nucleus showed similar performance, whereas the membrane fraction was notably worst. On the left- most side of the ROC curves, where the highest specificity was reached, mitochondria and nucleus exhibited the best better performance. These data may again hint to problems associated with multiple labels during multi-class training.

When the accuracy and precision value calculated from the 100 thresholds defined above is plotted against the confidence (results not shown), from the curve slop, it is clear that confidence and precision increase monotonically. Higher stringency yields a sparse lowering of the accuracy due to an enhanced rate of false negatives. The simple relationship between classifier precision and confidence can be used for evaluating the precision and accuracy of new predictions.

Using the trained classifiers derived for the four fractions, we applied the methods outlined above to our complete set of proteomic data. In this manner, we were able to assign 2390 of the proteins to at least one the four compartments, many of which were previously uncharacterized with respect to their subcellular localization. Given a minimum confidence threshold equal to 80%, where the highest overall accuracy was reached for each of the four classifiers, 1332 of the proteins were predicted to be associated with one compartment, while 38 were assigned to two or more organelles. These results point to the potential of machine learning as applied to proteomics data to make significant new biological inferences. We still have 1020 proteins which can not be assigned to any of the four subcellular localizations with at least 80% confidence.

4. Discussion

In this study, we have proposed a new framework for predicting protein subcellular localization on a genome-wide scale. The framework addresses some of the key problems associated with predicting multiple organellar compartments given proteins of uncertain association. Using large-scale proteomics data as a building block, the method achieved an overall accuracy 80.5% and precision 75.1% over the four major cellular compartments. We have confidently predicted the organellar localizations of more than 1,000 orphan proteins without previously described localizations in any of the major public annotation databases.

It should be noted that there are still some limitations to our methodology. First, we have just addressed the complex multiplex issue (multi-class) in the first training step of the learning process. We chose to simplify the problem in test step into a series of binary single localization calculations. This is clearly suboptimal from both a biological and a theoretical perspective. Second, we were unable to confidently assign more than 1,000 proteins to any one organelle, suggesting that more efficient algorithms need to be developed.

Scott et al. (2004) have previously reported that the incorporation of knowledge concerning the presence or absence of protein structural domains of motifs (such as InterPro motifs), as well as sequence signal peptides, and the number of putative trans-membrane regions, into a Bayesian machine learning framework can produce good prediction performance. As noted previously, the combination of amino acid composition and sequence-order information can also produce solid performance in subcellular localization prediction (Cai et al., 2002). Since these sequence-based approaches are quite different from, and indeed are even orthogonal to, our experimentally-based approach, logically integration of these various algorithms may allow for even better prediction power.

References

Andersen, J. S., Lam, Y. W., Leung, A. K., Ong, S. E., Lyon, C. E., Lamond, A. I., and Mann, M., 2005, Nucleolar proteome dynamics, *Nature* 433:77–83.

Beausoleil, S. A., Jedrychowski, M., Schwartz, D., Elias, J. E., Villen, J., Li, J., Cohn, M. A., Bradley, A. P., 1997, The use of the area under the ROC curve in the evaluation of machine learning algorithms, *Pattern Recognit*. 30:1145–1159.

Boutell, M., Shen, X., Luo, J., and Brown, C., 2004, Learning multi-label semantic scene classification, *Pattern Recognit*, 37:1757–1771.

Breiman, L., 1996, Bagging predictor, *Mach Learn* 24:123–140.

Cai, Y. D. and Chou, K. C, 2004, Predicting subcellular localization of proteins in a hybridization space, *Bioinformatics* 20:1151–1156.

Cai, Y. D., Liu, X. J., Xu, X. B., and Chou, K. C., 2002, Support vector machines for prediction of protein subcellular location by incorporating quasi-sequence-order effect, *J. Cell. Biochem*. 84:343–348.

Chou, K. C., 2000, Prediction of protein subcellular locations by incorporating quasi-sequence-order effect, Biochem. *Biophys. Res. Commun*. 278:477–483.

Chou, K. C. and Cai, Y. D., 2005, Predicting protein localization in budding yeast, *Bioinformatics.* 21:994–950.

Chou, K. C. and Elrod, D. W., 1998, Using discriminant function for prediction of subcellular location of prokaryotic proteins, *Biochem. Biophys. Res Commun.* 252:63–68.

Dudoit, S., Fridlyand, J., and Speed T. P., 2002, Comparison of discrimination methods for the classification of tumors using gene expression data, *J Amer Stat Assoc*. 97:77–87.

Hastie, T., Tibshirani, R., and Friedman, J., 2001, The elements of statistical learning. New York: Springer.

Hua, S. and Sun, Z., 2001 Support vector machine approach for protein subcellular localization prediction, *Bioinformatics*. 17:721–728.

Huang, Y. and Li, Y., 2004, Prediction of protein subcellular localizations using fuzzy k-NN method, *Bioinformatics*. 20:21–28.

Kislinger, T., and Emili, A., 2003, Going global: protein expression profiling using shotgun mass spectrometry, *Curr Opin Mol Ther.* 5:285–293.

Kislinger, T., Rahman, K., Radulovic, D., Cox, B., Rossant, J., and Emili, A., 2003, PRISM, a Generic Large Scale Proteomic Investigation Strategy for Mammals, *Mol Cell Proteomics*. 2:96–106.

Krapfenbauer, K., Fountoulakis, M., and Lubec, G., 2003, A rat brain protein expression map including cytosolic and enriched mitochondrial and microsomal fractions, *Electrophoresis*. 24:1847–1870.

Liu, H., Sadygov, R. G., and Yates, J. R., 3rd, 2004, A model for random sampling and estimation of relative protein abundance in shotgun proteomics, *Anal Chem.* 76: 4193–4201.

Lu, Z., Szafron, D., Greiner, R., Lu, P., Wishart, D.S., Poulin, B., Anvik, J., Macdonell, C., and Eisner, R., 2004, Predicting subcellular localization of proteins using machine-learned classifiers, *Bioinformatics.* 20:547–556.

Mitchell, T.M., 1997, Machine Learning. McGraw-Hill, N.Y.

Mootha, V. K., Bunkenborg, J., Olsen, J. V., Hjerrild, M., Wisniewski, J. R., Stahl, E., Bolouri, M. S., Ray, H. N., Sihag, S., Kamal, M., et al., 2003, Integrated analysis of protein composition, tissue diversity, and gene regulation in mouse mitochondria, *Cell*. 115:629–640.

Mott, R., Schultz, J., Bork, P., and Ponting, C.P., 2002, Predicting protein cellular localization using a domain projection method, *Genome Res.* 12:1168–1174.

Nakai, K. and Kanehisa, M., 1992, A knowledge base for predicting protein localization sites in eukaryotic cells, *Genomics.* 14:897–911.

Nakashima, H. and Nishikawa, K., 1994, Discrimination of intracellular and extracellular proteins using amino acid composition and residue-pair frequencies, *J. Mol. Biol.* 238: 54–61.

Nielsen, P. A., Olsen, J. V., Podtelejnikov, A. V., Andersen, J. R., Mann, M., and Wisniewski, J. R., 2005, Proteomic mapping of brain plasma membrane proteins, *Mol Cell Proteomics*. 4:402–408.

Park, J. K. and Kanehisa, M., 2003, Prediction of protein subcellular localizations by support vector machines using compositions of amino acids and amino acid pairs, *Bioinformatics*. 19:1656–1663.

Reinhardt, A. and Hubbard, T., 1998, Using neural networks for prediction of the subcellular location of proteins, *Nucleic Acids Res.* 26:2230–2236.

Ripley, B. D., 1996, Pattern recognition and neural networks. Cambridge: Cambridge University Press.

Schirmer, E. C., Florens, L., Guan, T., Yates, J. R., 3rd, and Gerace, L., 2005, Identification of novel integral membrane proteins of the nuclear envelope with potential disease links using subtractive proteomics, *Novartis Found Symp,* 264:63–76; discussion 76–80, 227–230.

Scott, M. S., Thomas, D. Y., and Hallett, M.T., 2004, Predicting subcellular localization via protein motif co-occurrence, *Genome Res*, 14:1957–1966.

Tao, D. and Tang, X., 2004, Random sampling based SVM for relevance feedback image retrieval. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'04)*, 1063–1069.

Weiss, G.M. and Provost, F., 2003, Learning When Training Data are Costly: The Effect of Class Distribution on Tree Induction, *J Artif Intell Res*. 19:315–354.

Wu, C. C., MacCoss, M. J., Howell, K. E., and Yates, J. R., 3rd, 2003, A method for the comprehensive proteomic analysis of membrane proteins, *Nat Biotechnol.* 21:532–538.

Wu, C. C., MacCoss, M. J., Mardones, G., Finnigan, C., Mogelsvang, S., Yates, J. R., 3rd, and Howell, K. E., 2004, Organellar proteomics reveals Golgi arginine dimethylation, *Mol Biol Cell.* 15:2907–2919.

Yates, J. R., 3rd, 2004, Mass spectral analysis in proteomics, *Annu Rev Biophys Biomol Struc.* 33:297–316.

Yeang, C. H., Ramaswamy, S., Tamayo, P., Mukherjee, S., Rifkin, R. M., Angelo, M., Reich, M., Lander, E., Mesirov, J., and Golub, T., 2001, Molecular classification of multiple tumor types, *Bioinformatics*. 17 suppl., S316–S322.

Chapter 14

Bioinformatics Analysis: Gene Fusion

Meena Kishore Sakharkar[1], Yiting Yu[1], and Pandjassarame Kangueane[2]

[1]*Nanyang Technology University, Singapore*
[2]*Biomedical Informatics, India*

Abstract: Gene fusion is an important evolutionary phenomenon. Human fusion proteins consisting of two or more fusion partners of bacterial origin exhibit accreted (enhanced or novel) function. These proteins mimic operons, simulate protein subunit interfaces in bacteria, exhibit multiple functions, and show alternative splicing in humans. They are also associated with metabolites having greater connectivity in complex networks.

Key words: Gene fusion, Evolution, Domain, Fusion protein

1. Introduction

A fusion gene in one species consists of fusion partners from one or more species. The transfer of genes and bringing together of genes from two genomes into a single gene (gene fusion) has long been identified as a potentially important evolutionary phenomenon (Long, 2000). Gene fusion has been identified across various phylogenetic groups and this suggests that there exist processes other than vertical inheritance during evolution (Genereux and Logsdon, 2003). In recent years, databases have been constructed to identify fusion events across distant phylogenies. These databases contain fusion proteins between human and yeast (Truong and Ikura, 2003); human and bacteria (Yiting et al., 2004), and among bacteria (Suhre and Claverie , 2004).

An interesting relational algebra approach has been demonstrated to identify fusion proteins across different phylogenetic distances (Truong and Ikura, 2003). Yanai and colleagues used gene fusion to establish links between fusion genes and functional network of their involvement (Yanai et al., 2001). Fusion genes gain added advantage in higher organisms by

V.S. Mathura, P. Kangueane, *Bioinformatics: A Concept-Based Introduction*,
DOI 10.1007/978-0-387-84870-9_14,

coupling biochemical/signal transduction reactions through tight regulation of fusion partners, compared to individual fusion partners in lower organisms (Tsoka and Ouzounis , 2001). Thus, fusion genes produce proteins with novel or enhanced function. Gene fusion has also been used to illustrate protein subunit interactions (Marcotte et al., 1999), enhanced substrate specificity (Katzen et al., 2002), and multi-functional enzyme specificity (Berthonneau and Mirande, 2000).

The human genome contains a small fraction of genes (<1%) exclusively homologous to bacterial genes (International Human Genome Seqeuencing Consortium, 2001). Though, lateral gene transfer and differential loss of genes (Andersson et al., 2001) have been described to account for the presence of bacterial genes in the human genome, the frequencies of these occurrences remain a subject of conjecture (Salzberg et al., 2001). Two opposing forces work in palindrome: one that shuffles the genome and the other that prevents the shuffle by gene fusion. Thus, fusion genes are treated as one unit, working in synergy to achieve optimal functionality. Here, we report human fusion genes consisting of two or more fusion partners of bacterial origin. We describe examples of fusion proteins mimicking bacterial operons, simulating bacterial subunit interfaces, exhibiting multiple functions, and showing alternative splicing. They are also associated with metabolites in complex networks.

2. Identification of Fusion Proteins

We used two datasets of protein sequences for this analysis. The first dataset consists of 37,490 human proteins and the second dataset consists of 223,676 bacterial proteins from 71 completed bacterial genomes. A comparison was performed after removing homologous sequences in each dataset at 40% sequence identity cut-off (homologous proteins share a common fold at > 40% identity) using the purging program CD-HIT (Li et al., 2001). We compared the human proteins (non homologous set of 26673) with the bacterial proteins (non homologous set of 102135) using BLASTP. All matches with an E-value (expectation value) < 10^{-10} were further processed using in house Perl scripts for the identification of human fusion proteins. By definition, each fusion protein should match two or more fusion partners of bacterial origin. This procedure identified 141 human fusion proteins consisting of two or more fusion partners of prokaryotic origin. Information on these proteins is made available at http://sege.ntu.edu.sg/wester/fusion. Molecular functions were assigned for 29 of the fusion proteins using data collected from literature. They mimic operons, simulate protein subunit interfaces in bacteria, exhibit multiple functions, and show alternative splicing in humans.

2.1 Human Fusion Proteins Mimicking Bacterial Operons

Bacterial genes involved in a related pathway are arranged as operons (cluster of genes that are juxtaposed next to each other and are transcribed as one unit). This is also true in the unsegmented worm *C. elegans* that is shown to have operons (von Mering and Bork, 2002). Fusion is a way of co-regulation as efficiently as operons with two or more juxtaposed genes in a single unit. Here we describe an example of a fusion protein mimicking genes in a bacterial system. This could be a potent indicator of optimal design. The fusion protein pyrroline-5-carboxylate synthetase (P5CS) catalyzes ATP and NAD(P)H dependent conversion of L-glutamate to glutamic γ-semialdehyde (GSA) in proline biosynthesis. The P5CS protein is bi-functional with γ-glutamate-5-kinase (γ-GK) and γ-glutamyl phosphate reductase (γ-GPR) activities required for proline biosynthesis (Aral et al., 1996). N terminal γ-GK and C terminal γ-GPR match prokaryotic GK and GPR proteins, respectively. In *T. thermophilus*, these two proteins operate as one operon with GK preceding GPR (Kosuge et al., 1994). This suggests that two or more partners form fusion proteins in human.

2.2 Human Fusion Proteins Simulating Bacterial Subunit Interfaces

Some fusion proteins simulate protein subunit interfaces in bacteria. For example, the human fusion protein acetyl co-enzyme A carboxylase β simulates the dimer of propionyl co-A carboxylase α subunit and propionyl co-A carboxylase β subunit in *Mycobacterium smegmatis*. Thus, two domains in acetyl co-enzyme A carboxylase β simulate a subunit interface formed by propionyl co-A carboxylase α subunit and propionyl co-A carboxylase β subunit in *Mycobacterium smegmatis*. This suggests that fusion events select subunit interfaces by fusing two fusion partners into a single polypeptide chain. Marcotte and colleagues identified human fusion proteins succinyl Co-A transferase and δ-1-pyroline-5-carboxylate synthetase made up of fusion components that are known or predicted to interact in *E. coli* (Marcotte et al., 1999). Interestingly, our approach identified these two fusion proteins. It should also be noted that these two proteins not only simulate protein–protein interfaces in *E. coli* but also mimic operon like structures in *T. thermophilus* and *M. barkeri,* respectively.

2.3 Fusion Proteins Exhibiting Multiple Functions

Many human multi-functional proteins catalyze successive reactions in biochemical/signal transduction pathways. The reaction rate is maximally

optimized in these cases because the subsequent reaction centers (active sites) are physically placed side by side. This facilitates the easy capture of reaction intermediates from one reaction center to another as substrates (circumventing diffusion effects). Clustering of active sites for catalyzing a reaction sequence has several potential advantages: the catalytic activity can be enhanced because the local substrate concentrations are increased significantly. By sequestering reactive intermediates, their conversion by undesired chemical reactions is prevented as substrates are channeled from one catalytic site to the next (Perham, 1975). A covalently linked multifunctional protein is likely to be more stable than non-covalently formed protein subunit interfaces containing reaction (or active) centers. Thus, fusion of two or more mono-functional bacterial proteins into a single polypeptide in a higher organism is certainly under selective advantage in evolution. The fusion protein GARS-AIRS-GART exhibits multiple functions in human. Each of GARS, AIRS, and GART proteins are mono-functional and part of the *pur* operon in *B. subtilis* and *E. coli* (Ebbole and Zalkin, 1987). The GARS-AIRS is a bifunctional protein in *S. cerevisiae* and GARS-AIRS-GART is tri-functional in *Drosophila*. In human, it is found that GARS-AIRS-GART is tri-functional and is formed by the fusion of three mono-functional enzymes. Thus, human fusion proteins exhibit expanded function by physical co-existence of two or more mono-functional fusion partners.

2.4 Fusion Proteins Showing Alternative Splicing

A classic example of a fusion protein exhibiting alternative splicing is the GARS–AIRS–GART gene that produces two spliced variants, namely: (1) a tri-functional GARS–AIRS–GART; (2) a mono-functional GARS. The mono-functional GARS protein is produced by differential use of an intronic poly-adenylation signal located in the intron separating the last GARS exon from the first AIRS exon. Separate GARS and GARS–AIRS–GART mRNAs have been observed in human, mouse, chicken, and *D. melanogaster.*

3. Remarks on Fusion Proteins

Modular organization of proteins has been postulated as a widely used strategy for protein evolution. Analysis of human fusion proteins suggests that these proteins exhibit enhanced or novel functions in human compared to their fusion partners (which are physically separated) in bacteria. These fusion proteins are found to mimic operons and simulate bacterial protein subunit interfaces. They are also found to exhibit multiple functions and

alternative splicing in humans. Our findings strongly suggest that, by the acquisition of additional active domains, fusion proteins expand their substrate specificity, and evolve functional novelty.

Protein evolution is extremely efficient in generating systems that are optimally adapted in cellular environment. Optimality can be achieved by changing the topology of metabolic networks by tuning enzymatic or regulatory materials. Here, we show that metabolites like oxaloacetate, acetyl co-A, succinyl co-A, succinate, and glutamate are products of fusion proteins. These metabolites have high connectivity index, suggesting their greater degree of involvement within networks. This observation implies the association of fusion proteins with complex metabolic networks. The association between human fusion proteins and metabolites with high connectivity is intriguing. Detailed analysis of fusion proteins highlights the transition from a 'protein–protein interface' to either a 'domain–domain interface' or an operon structure (a group of genes all controlled by the same regulatory element). This evolutionary transition is interesting and it is important to systematically investigate the functional link between fusion partners and fused proteins using thermodynamics calculations. The transition may be thermodynamically favorable as fusion proteins acquire reduced entropy compared to their physically separated fusion partners. Therefore, it is envisaged that fusion proteins confer selective advantage in the evolution of regulating metabolic dynamics. This is specifically advantageous for multi-enzyme complexes as fusion proteins select kinetic advantage over fusion components by increasing connectivity with metabolites. It is also reported that fusion of components into a single polypeptide ensures stability between physically connected domain structures and active sites for a balanced stoichiometric production of intermediates in complex networks. The physical proximity of multiple active centers in the same metabolic pathways alleviates molecular diffusion and reduces side reactions in cellular environment. Our data for the six metabolic enzymes having fusion structures aligns well with these observations. This enables fusion proteins to catalyze sequential steps in a biochemical pathway because association of two active sites enhances the efficiency of two consecutive reactions. Thus, fused protein architecture illustrates an evolutionary strategy for maintaining complex stoichiometric balance. Physical connection between fused domains increases structural propensity between active centers for the regulation of material balance. Since fusion proteins help in the evolution of complex networks, even a modest addition of domains could significantly increase interactions. This strategy helps to maintain equilibrium in a dynamic network with huge nodes. Thus, large networks of molecular interactions are regulated by relatively few genes in some organisms.

The hypothesis underlying this analysis is that a fusion gene in human can indicate an association between the independent genes in bacteria, assuming that orthologous genes have parallel functions in both human and one or more bacteria. Linking genes by way of fusion events, as proposed earlier can hint at direct physical interactions between proteins or a more general functional association such as between sequential members in a metabolic pathway. One of many possible mechanisms of fusion events is lateral gene transfer and this hypothesis remains as speculation due to lack of sufficient genome data of distant evolutionary origin. The idea of gene transfer from a prokaryote to human is intriguing. However, the significant mechanical barriers, as well as constraints to natural selection, warn caveats when considering inter-kingdom gene transfer.

References

Andersson, J.O., Doolittle, W.F., et al. (2001) Genomics. Are there bugs in our genome? *Science* **292**(5523), 1848–50.

Aral, B., Schlenzig, J.S., et al. (1996) Database cloning human delta 1-pyrroline-5-carboxylate synthetase (P5CS) cDNA: a bifunctional enzyme catalyzing the first 2 steps in proline biosynthesis. *C R Acad Sci III* **319**(3), 171–8.

Berthonneau, E. and Mirande, M. (2000) A gene fusion event in the evolution of aminoacyl-tRNA synthetases. *FEBS Lett* **470**(3), 300–4.

Ebbole, D.J. and Zalkin, H. (1987) Cloning and characterization of a 12-gene cluster from Bacillus subtilis encoding nine enzymes for de novo purine nucleotide synthesis. *J Biol Chem* **262**(17), 8274–87.

Genereux, D.P. and Logsdon, J.M., Jr. (2003) Much ado about bacteria-to-vertebrate lateral gene transfer. *Trends Genet* **19**(4), 191–5.

Katzen, F., Deshmukh, M., et al. (2002) Evolutionary domain fusion expanded the substrate specificity of the transmembrane electron transporter DsbD. *Embo J* **21**(15), 3960–9.

Kosuge, T., Tabata, K., et al. (1994) Molecular cloning and sequence analysis of the proBA operon from an extremely thermophilic eubacterium Thermus thermophilus. *FEMS Microbiol Lett* **123**(1–2), 55–61.

Li, W., Jaroszewski, L., et al. (2001) Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics* **17**(3), 282–3.

Long, M. (2000) A new function evolved from gene fusion. *Genome Res* **10**(11), 1655–7.

Marcotte, E.M., Pellegrini, M., et al. (1999) Detecting protein function and protein-protein interactions from genome sequences. *Science* **285**(5428), 751–3.

Perham, R.N. (1975) Self-assembly of biological macromolecules. *Philos Trans R Soc Lond B Biol Sci* **272**(915), 123–36.

Salzberg, S.L., White, O., et al. (2001) Microbial genes in the human genome: lateral transfer or gene loss? *Science* **292**(5523), 1903–6.

Suhre, K. and Claverie, J.M. (2004) FusionDB: a database for in-depth analysis of prokaryotic gene fusion events. *Nucleic Acids Res* **32**(Database issue), D273–6.

Truong, K. and Ikura, M. (2003) Domain fusion analysis by applying relational algebra to protein sequence and domain databases. *BMC Bioinformatics* **4**, 16.

Tsoka, S. and Ouzounis, C.A. (2001) Functional versatility and molecular diversity of the metabolic map of Escherichia coli. *Genome Res* **11**(9), 1503–10.

von Mering, C. and Bork, P. (2002) Teamed up for transcription. *Nature* **417**(6891), 797–8.

Yanai, I., Derti, A., et al. (2001) Genes linked by fusion events are generally of the same functional category: a systematic analysis of 30 microbial genomes. *Proc Natl Acad Sci U SA* **98**(14), 7940–5.

Yiting, Y., Chaturvedi, I., et al. (2004) Can ends justify the means? Digging deep for human fusion genes of prokaryotic origin. *Front Biosci* **9**, 2964–71.

Index