Comprehensive Summaries of Uppsala Dissertations
from the Faculty of Science and Technology 999

Predicting Function of Genes and Proteins from Sequence, Structure and Expression Data

BY

TORGEIR R. HVIDSTEN

ACTA UNIVERSITATIS UPSALIENSIS
UPPSALA 2004

Dissertation presented at Uppsala University to be publicly examined in Room C8:305, BMC – Uppsala Biomedical Centre, Uppsala, Wednesday, September 22, 2004 at 14:00 for the Degree of Doctor of Philosophy.

Abstract

Hvidsten T. R.. 2004. Predicting function of genes and proteins from sequence, structure and expression data. Acta Universitatis Upsaliensis. *Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science and Technology 999*. 63 pp. Uppsala. ISBN 91-554-6014-3.

Functional genomics refers to the task of determining gene and protein function for whole genomes, and requires computational analysis of large amounts of biological data including DNA and protein sequences, protein structures and gene expressions. Machine learning methods provide a powerful tool to this end by first inducing general models from such data and already characterized genes or proteins and then by providing hypotheses on the functions of the remaining, uncharacterized cases.

This study contains four parts giving novel contributions to functional genomics through the analysis of different biological data and different aspects of biological functions. Gene Ontology played an important part in this research providing a controlled vocabulary for describing the cellular roles of genes and proteins in terms of specific molecular functions and broad biological processes.

The first part used gene expression time profiles to learn models capable of predicting the participation of genes in biological processes. The model consists of IF-THEN rules associating biological processes with minimal set of discrete changes in expression level over limited periods of time. The models were used to hypothesize new biological processes for both characterized and uncharacterized genes.

The second part investigated the combinatorial nature of gene regulation by inducing IF-THEN rules associating minimal combinations of sequence motifs common to genes with similar expression profiles. Such combinations were shown to be significantly correlated to function, and provided hypotheses on the mechanisms behind the regulation of gene expression in several biological responses.

The third part used a novel concept of local descriptors of protein structure to investigate sequence patterns governing protein structure at a local level and to predict the topological class (fold) of protein domains from sequence. Finally, the fourth part used local descriptors to represent protein structure and induced IF-THEN rule models predicting molecular function from structure.

Torgeir R. Hvidsten, Department of Information Technology, Box 337, Uppsala University, SE-75105 Uppsala, Sweden.

ISSN 1104-232X
ISBN 91-554-6014-3
urn:nbn:se:uu:diva-4490 (http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-4490)

List of papers

I. T. R. Hvidsten, A. Lægreid and J. Komorowski. Learning rule-based models of biological process from gene expression time profiles using gene ontology, *Bioinformatics* 19(9): 1116-23, 2003.

II. A. Lægreid, T. R. Hvidsten, H. Midelfart, J. Komorowski and A. K. Sandvik. Predicting Gene Ontology Biological Process From Temporal Gene Expression Patterns, *Genome Research*, 13(5): 965-979, 2003.

III. T. R. Hvidsten, B. Wilczyński, A. Kryshtafovych, J. Tiuryn, J. Komorowski and K. Fidelis. Discovering regulatory binding site modules using rule-based learning, *Submitted.*

IV. T. R. Hvidsten, A. Kryshtafovych, J. Komorowski and K. Fidelis. A novel approach to fold recognition using sequence-derived properties from sets of structurally similar local fragments of proteins, *Bioinformatics* 19 Suppl 2 (European Conference on Computational Biology): II81-II91, 2003.

V. T. R. Hvidsten, A. Kryshtafovych, J. Komorowski and K. Fidelis. Local descriptors of protein structure. Part II. Application to structure prediction. *Manuscript.*

VI. T. R. Hvidsten, A. Kryshtafovych, P. Daniluk, K. Fidelis and J. Komorowski. Predicting function from local substructures of proteins. *Submitted.*

Acknowledgements

I would like to thank my supervisor Jan Komorowski for introducing me to rough set-based rule learning, bioinformatics and science in general, for involving me in his research visions and for creating research environments that have given me many interesting scientific challenges and contacts.

Thanks to Astrid Lægreid at the Norwegian University of Science and Technology (NTNU) for help and guidance on various biological issues, and for spending countless hours interpreting my results. I am also grateful for collaboration with Herman Midelfart and Arne K. Sandvik during my two first years as a PhD student at NTNU.

Thanks also to Krzysztof Fidelis for hosting me at Lawrence Livermore National Laboratory (LLNL) on numerous occasions, and for involving me in his many ideas on structure prediction and gene regulation. During my stays at LLNL, I met several people to whom I'm grateful for help and discussions; in alphabetical order Pawel Daniluk, Michal Drabikowski, Szymon Nowakowski, Andriy Kryshtafovych, Lisa Stubbs, Ceslovas Venclovas and Bartosz Wilczynski. A special thanks to Andriy for sharing his knowledge on protein structure prediction and sights in California.

The two last years as a PhD student at the Linnaeus Centre for Bioinformatics (LCB), Uppsala University, has been a particularly rewarding time both scientifically and socially. I would like to thank all my colleagues at LCB in this regard. In particular, I would like to thank Claes Andersson, Alice Lesser, Erik Bongcam-Rudloff, Vincent Moulton and Helena Strömbergsson for valuable scientific advice, discussions and comments.

The implemented extensions to the ROSETTA system have been improved by comments and implementation-help from a number of people. Thanks to Robin Andersson, Hans-Richard Brattbakk, Sjur Huseby and Anna Johansson.

I am grateful to the Department of Computer and Information Science at the NTNU for allowing me to use their computational cluster (ClustIS: Intelligent System Cluster) even after I left the department. A special thanks to Zoran Constantinescu for helping me with various technical problems.

This work was supported in part by the Department of Computer and Information Science at NTNU, the Norwegian Research Council (grant 145609/432), The Knut and Alice Wallenberg Foundation and The Swedish Foundation for Strategic Research. The financial support is greatly appreciated.

A special thanks to Christin Grønnslett for her patience and care, and to my family for always supporting me.

Contents

CHAPTER 1 INTRODUCTION 1

1.1 Introduction to molecular biology 1

1.2 Functional genomics 6
1.2.1 DNA sequencing 7
1.2.2 Microarray technology 8
1.2.3 Crystallography and nuclear magnetic resonance 10

1.3 Bioinformatics 11
1.3.1 Databases and annotations 12
1.3.2 Machine learning 16

1.4 Aims and contributions of the study 24

CHAPTER 2 METHODS AND RESULTS 27

2.1 PAPERS I & II: Predicting biological process from gene expression time profiles 27
2.1.1 Related research: Functional gene expression analysis 28
2.1.2 Data: Expression data and Gene Ontology annotations 29
2.1.3 Method: Rough set-based rule induction and the ROSETTA system 29
2.1.4 Results 32

2.2 PAPER III: Discovering regulatory binding site modules 34
2.2.1 Related research: Combinatorial gene regulation 34
2.2.2 Data: Expression and sequence motif data 35
2.2.3 Method: Rule learning in gene regulation 35
2.2.4 Results 36

2.3 PAPERS IV & V: Fold recognition from local descriptors of protein structure 36
2.3.1 Related research: Protein structure prediction 37
2.3.2 Data: Local descriptors of protein structure 39
2.3.3 Method: Fold recognition using local descriptors of protein structure 39
2.3.4 Results 40

2.4 PAPER VI: Predicting molecular function from local descriptors of protein structure 41
2.4.1 Related research: Structural motifs 41
2.4.2 Data: Local descriptor groups and Gene Ontology annotations 42
2.4.3 Method: Rule models for learning function from structure 42
2.4.4 Results 43

CHAPTER 3 DISCUSSION AND CONCLUSIONS 45

CHAPTER 4 SUMMARY IN SWEDISH 53

REFERENCES 55

Chapter 1 Introduction

The aim of this study was to develop bioinformatics methods and tools for a number of challenges in functional genomics. Contributions include using rule learning and other machine learning methods to (a) predict the participation of genes in biological processes from gene expression time profiles, (b) to investigate the combinatorial nature of gene regulation from sequence motifs and gene expression data, (c) to recognize protein fold from protein sequence using local descriptors of protein structure and (d) to predict the molecular function of proteins from local descriptors of protein structure.

As it is our hope that this text will be read by both computer scientists and biologists, Chapter 1 will give a basic introduction to molecular biology, functional genomics and relevant computational methods in bioinformatics. With this context in place, the specific aims and contributions of this study will be stated. Chapter 2 will give a summary of the methods and results presented in detail in six self-contained research papers, while Chapter 3 will discuss some of the most important results and draw some parallels between the different contributions. Chapters 1, 2 and 3 thus provide a general framework in which the six research papers may be read and understood.

1.1 Introduction to molecular biology

Living organisms are governed by a set of inherited instructions encoded by the four letter alphabet A, G, C and T. The "letters" take physical shape in terms of four different *nucleotides* constituting the basic repeating unit of *deoxyribonucleic acid* (DNA) molecules. Each nucleotide consists of a 5-carbon sugar with a nitrogen base covalently attached[1] to carbon atom 1' and a phosphate group covalently attached to carbon atom 3' or 5'. A DNA molecule is a repeating chain of nucleotides where each phosphate group links carbon atom 3' of the sugar in one nucleotide to carbon atom 5' of the sugar in the neighboring nucleotide. There are four types of nitrogen bases determining the four different nucleotides in DNA (adenine (A), guanine (G), cytosine (C) and thymine (T)), and hence each DNA molecule represents a unique sequence of these four chemical "letters". DNA molecules are furthermore structurally organized in duplexes consisting of two helical DNA molecules coiled around a common axis forming a *double helix*. The two strands of the double helix have

[1] Covalent bonds occur when two atoms share a common pair of electrons and are the type of bindings that hold atoms together in molecules.

opposite directions for linking 3' carbon atoms to 5' carbon atoms (i.e. they are anti-parallel) and are held together by hydrogen bonds[2] between opposite bases in the two strands. An important property of the double helix is that hydrogen bonds only occur between two specific pairs of bases. A only binds to T and C only to G. This means that the two strands are *complementary* with respect to the sequence they encode, conveniently facilitating important processes such as replication and transcription (see below). In eukaryotes[3], the DNA molecules (the *genome*) are systematically packed into a number of chromosomes residing in the nuclei of each cell (in animal cells a small fraction of the DNA is located in mitochondria[4]). The actual number and content of the chromosomes varies from species to species.

The *central dogma of molecular biology* states that the genetic information hard-wired in the DNA is *transcribed* into portable *messenger ribonucleic acid* (mRNA) molecules that are subsequently *translated* into *proteins* (see Figure 1). Except for uracil (U) replacing thymine (T) in the mRNA sequence, a mRNA molecule is an exact copy of a segment of one DNA strand, and carries the information necessary to synthesize one or a small number of proteins. While the DNA may be viewed as a storage device for genetic instructions, proteins actually execute these instructions as enzymes, receptors, storage proteins, transport proteins, transcription factors, signaling molecules, hormones, etc. Exceptions are some RNAs that are not translated into proteins and that perform functions directly (tRNA, rRNA and snRNA are examples of functional RNAs that will be discussed later)

The RNA-encoding segments of the DNA are called *genes*[5]. Transcription of genes into RNAs is performed by RNA *polymerase* enzymes using one of the DNA strands as a template. The double-stranded DNA is unwound during transcription so that the strand acting as a template for the RNA synthesis can form a hybrid with the new, growing RNA. The transcribed RNA is consequently a single strand sequence complementary to the template strand

---

[2] Polar molecules may have a weak, negative charge at one region and a weak, positive charge elsewhere. Hence, when such molecules are close, the charged region of one molecule may attract the oppositely charged region of a neighboring molecule. These attractions are called hydrogen bonds.

[3] Eukaryotes refer to animals or plants consisting of cells with a membrane-enclosed nucleus and organelles. Organelles are any structure found in the viscous content of the cell (i.e. the cytoplasm).

[4] Mitochondria are large organelles responsible for most of the energy production in eukaryotic cells.

[5] In contrast, classical Mendelian genetics refer to a gene as an inheritable trait.

and identical to the DNA strand not acting as a template (except that U replaces T).

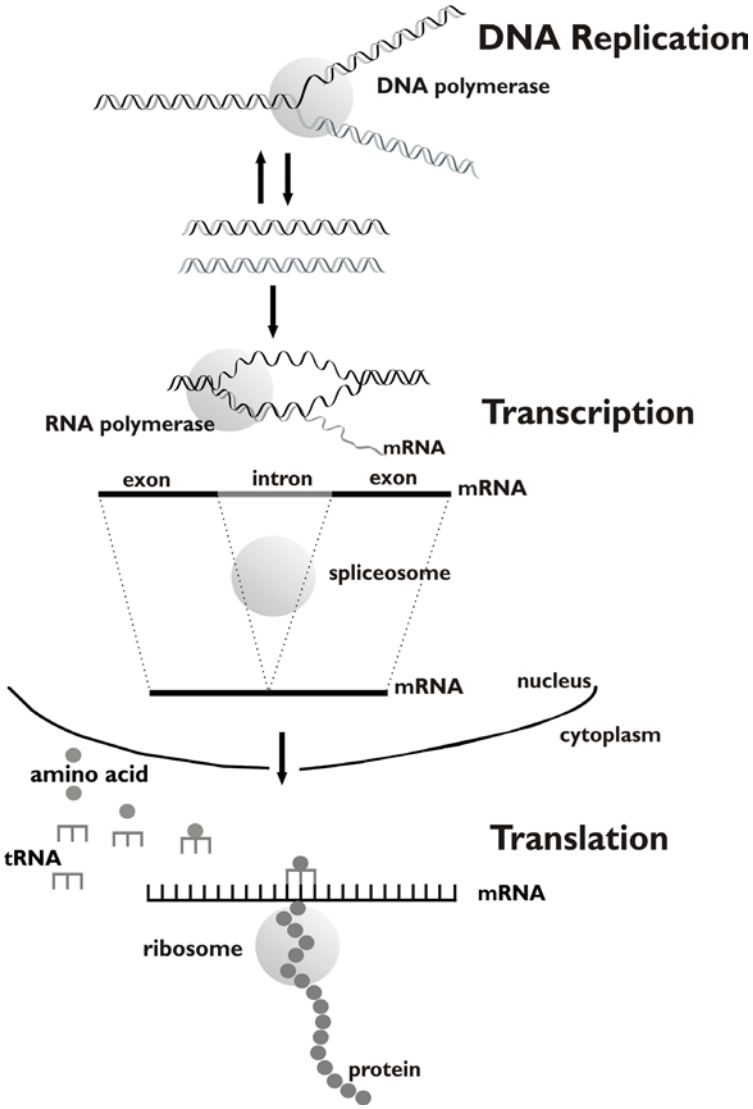


Figure 1. The central dogma of molecular biology (diagrammatic). DNA is transcribed into mRNA that is translated into protein. In addition, DNA is replicated during cell division with the help of DNA polymerase. Transcription is catalyzed by the RNA polymerase. The mRNA is processed by the spliceosome, before translated into a chain of amino acids in the ribosome. tRNA helps the translation by transporting the right amino acids to the right positions as given by the mRNA

Genes are said to be *expressed* in a cell if they are transcribed. The ability to differentially express genes in different cell types, stages of the cell cycle and under various changes in the environment constitute one important level of dynamics in cellular organisms (another level of molecular dynamics is that of proteins and their interactions with each other and other molecules). A number of factors are important for the differential expression of a particular gene in different cells, including the rate of transcription, the rate of translation and the stability of the protein. However, the most important factor is the initiation of the actual transcription. In eukaryotes, the transcription is not initialized by the RNA polymerase, but by a number of regulatory proteins called *transcription factors* that bind to the DNA and both activate and guide the polymerase. The ability of these transcription factors to selectively recognize specific short sequence elements in DNA is therefore important for the regulation of gene expression (i.e. gene regulation). Many of these regulatory elements or binding sites are in a region called the promoter located upstream of the coding sequence (upstream and downstream refer to the sequences that flank a particular gene at the 5' and 3' ends, respectively).

Most eukaryotic RNA transcripts go through a number of preprocessing steps including the removal of certain segments within the gene and the merging of the remaining segments (RNA *splicing*). This is due to the internal structure of the genes which consists of coding segments called *exons* separated by non-coding regions called *introns*. Although both segments are transcribed, the introns are later removed by a large complex (the *spliceosome*) consisting of five types of small nuclear RNAs (snRNAs) and proteins. Newer studies show that exons in complex organisms such as humans are spliced in different ways, forming different splicing variants and hence different protein products from the same gene [67].

The synthesis of proteins from mRNA takes place in *ribosomes* that function as structural frameworks for translation. Ribosomes are large RNA-protein complexes consisting of a number of ribosomal RNAs (rRNAs) and proteins. The basic building blocks for proteins are *amino acids*. There are 20 amino acids, all consisting of a α-carbon atom (C_α) bound to an amino (NH_2) group, a carboxyl (COOH) group, a hydrogen (H) atom and one variable group determining the 20 different amino acids (the *side chains*). Proteins are simply linear, unbranched chains of amino acids where the amino group of one amino

acid forms a peptide bond[6] with the carboxyl group of the neighboring amino acid. The repeating chain without the variable side chains is called the main chain or the *backbone* of the protein molecule. Proteins are coded directly in the mRNA sequence in terms of successive groups of three nucleotides (*codons*). Since there are four different bases in RNA (and DNA) and three base positions in a codon, there are $4^3=64$ possible combinations for coding 20 amino acids. Hence, each amino acid is specified on average by about three different codons (the genetic code is said to be degenerate). mRNAs are translated into an amino acid chain with the help of transport RNAs (tRNAs). There is one tRNA per amino acid, capable of binding and transporting this specific amino acid. Each tRNA also includes a specific sequence (*anticodon*) that recognizes the relevant codon in the mRNA sequence so that the correct amino acid can be inserted into the growing amino acid chain.

An important principle in molecular biology is that the amino acid sequence of a protein determines its three-dimensional shape (i.e. its *structure*) and furthermore that the structure of a protein determines its function. Since the amino acid sequence is encoded in the DNA, it follows that the mechanisms of evolution (i.e. mutation and crossover) contribute almost directly in changing protein function. To accommodate different three-dimensional conformations, the 20 amino acids vary in shape, charge, hydrophobicity and reactivity. For example, the hydrophobic amino acids tend to be buried inside the protein (where they are protected from the water surrounding the protein), while the hydrophilic amino acids tend to be at the surface of the protein.

Protein structure is more complex than the double helix of DNA (see Figure 2 for an example), and may be organized into four levels. The amino acid sequence itself is referred to as the *primary structure*. When stable, the protein main chain folds into either an α *helix* (i.e. a spiral structure), a β *sheet* (i.e. a planar structure of more than one β strand) or a *coil* (i.e. a random structure) (see Figure 2a). These confirmations constitute the *secondary structure* of proteins. Furthermore, the secondary structure elements (sheets and helices) tend to form simple motifs connected by short U-shaped turns or *loops* often located at the protein surface (e.g. the common hairpin β motif consisting of two neighboring β strands joined by a loop). Several motifs form compact globular domains referred to as the *tertiary structure* of proteins. While secondary structure is stabilized by hydrogen bonds between certain side chains, tertiary structure is

[6] A peptide bond is a special chemical linage connecting amino acids into linear chains. It is formed by a condensation reaction between the amino group of one amino acid and the carboxyl group of the neighboring amino acid.

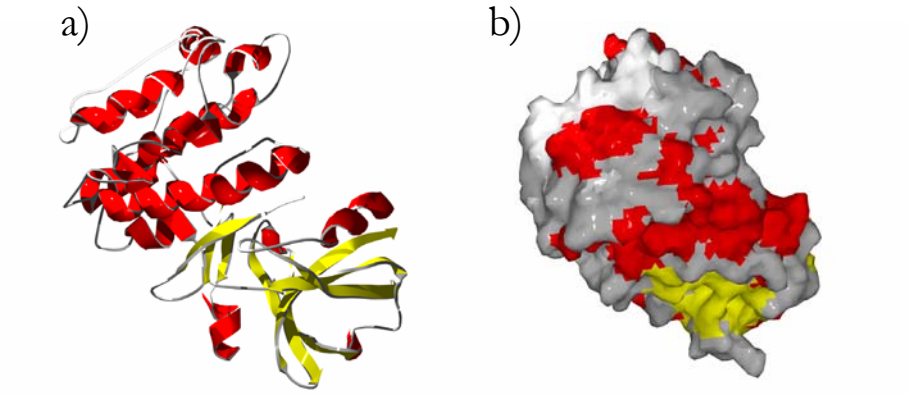


Figure 2. An example protein structure. Helices are colored red, sheets yellow and coils grey. a) shows a cartoon of the protein backbone, while b) shows the protein as a solid molecule. The pictures were generated using Swiss-PdbViewer [38].

mainly stabilized by hydrophobic interactions. Finally, some proteins consist of several amino acid chains (also called subunits) and their arrangements are referred to as the quaternary protein structure. As we have already seen with the spliceosome and the ribosome, proteins often function in large complexes involving several proteins and possibly other macromolecules.

1.2 Functional genomics

Biology has traditionally focused on classifying living systems (hierarchically) into increasingly smaller parts, and on studying these parts separately. This reductionistic research approach has culminated in molecular biology, where single molecules in terms of genes and gene products have been studied independently. This way of doing research has of course not been a result of biologists failing to realize the value of understanding the holistic molecular operation of biological systems, but rather a product of the sheer complexity of these systems and the lack of appropriate technology to probe them. With the publishing of the first complete genome sequence in 1995 (the bacteria *Haemophilus influenzae Rd* [32]), the premises have changed. A number of genome sequencing projects are now providing researchers with the basic instructions for the operation of entire organisms at an ever increasing speed (see http://www.genomesonline.org/, [10]). However, although DNA sequence data to some degree has facilitated a transition from molecular *genetics* (i.e. the study of single genes) to *genomics* (i.e. the study of all genes in a genome), genomics is more likely to complement rather than replace traditional use of genetics in understanding the detailed functioning of individual macromolecules [42]. Genomics has also undergone a change from the mapping and sequencing

of genomes to the more complex task of determining gene function (i.e. the function of the functional RNAs or proteins coded by the genes) and understanding gene regulation at a genome-wide scale. This part of genomics has been coined *functional genomics*, and has spanned a whole generation of technologies and databases to provide data and support for the statistical and computational analysis making this research possible.

Obviously, although sequence data provide us with the static map of an organism, it seems impossible to reach the goals of functional genomics using this information alone. Additional information, however, may be acquired using the opportunities that sequence data gives to identify genes and gene products and hence obtain data on the actual dynamic expression of the DNA code. One such example is technology for the genome-wide measurement of mRNA levels, providing valuable information on which genes are expressed, and thereby which gene products are active, in a potentially large range of biological contexts. Another example is methods for obtaining structural data. As stated earlier, it is a fundamental biological principle that protein sequence determines structure and that protein structure determines function. However, solving the structure of a protein is a time-consuming task and the amount of structural information therefore lags far behind the vast amount of sequence information. Structural genomics, however, promises to close this gap by combining a systematic approach to solving protein structure experimentally with computational methods for protein structure prediction. The next section will give an introduction to bioinformatics and the use of computers in aiding functional genomics. However, this section will be complemented with a short introduction to DNA sequencing, the high-throughput gene expression measurement technology of microarrays and the two most important experimental methods for solving protein structure. The two latter methods provide additional, partly sequence-derived, data for functional genomics and are of particular interest to this study.

1.2.1 DNA sequencing

There are several different techniques for determining the nucleotide sequence of DNA segments (i.e. DNA *sequencing*). In one of the most used approaches, DNA polymerase (which in the organism amongst other functions performs replication) is allowed to copy single stranded DNA segments using both altered nucleotides (*dideoxynucleotides*) and ordinary nucleotides. The alteration of the four dideoxynucleotides, corresponding to the four ordinary nucleotides, has the effect that when added by the polymerase to the growing chain, no further nucleotides can be added to the 3' end afterwards and hence the strand is terminated. Consequently, many fragments of different lengths are created, all

with a dideoxynucleotide at the 3' end. A gel solution containing the copied fragments may now be charged by a voltage so that the DNA fragments, which are slightly negative, start traveling towards the positive end of the gel. The speed at which the fragments travel depends on their length and the fragments may therefore be ordered accordingly. The four different dideoxynucleotides are labeled with four different *fluorochromes* that emit four different colors of light when absorbing radiation of specific wavelengths. The dideoxynucleotides at the 3' end may therefore be scanned with a laser and determined from the resulting image. Furthermore, the nucleotides in each position in the original DNA segment may now also be determined given fragments of all possible lengths. For example, the nucleotide in position 7 in the original segment is determined by the color of the light emitted by the dideoxynucleotide at the 3' end of fragments of length 7.

Whole genomes (or long DNA segments) may be sequenced by first dividing them into many overlapping fragments, then sequencing each of the fragments separately and finally assembling the genome sequence with the help of the overlaps. In addition to DNA, proteins may also be sequenced directly using methods such as Edman degradation.

1.2.2 Microarray technology

The complementary nature of the DNA double helix is of great importance to replication and transcription, and may also be utilized for the large-scale measurement of mRNA levels in cells. Two complementary nucleic acid molecules (i.e. strands) will combine under the right conditions to form double stranded helices. In a reaction vessel this is referred to as *hybridization*. Hence, it is possible to use identified DNA strands (*probes*) to query complex populations of unidentified, complementary strands (*targets*) by checking for hybridization. Microarrays are glass slides or wafers populated with large numbers of strands derived from identified genes. By applying a target sample of unidentified mRNA to the array, the expression level of each gene probe may be quantified from the extent of hybridization between the probes and the targets. Since one slide may contain probes from thousands of genes, one microarray experiment may determine the genome-wide expression state of a cell sample. Furthermore, systematic series of microarray experiments may reveal the specific changes in cellular gene expression associated with different physiological or pathophysiological[7] responses.

[7] Physiology is the study of life at the organism level in healthy states, while pathophysiology is the study of disease states.

The most common microarray technology is that of *DNA microarrays* [28, 93]. DNA microarrays are glass slides with DNA probes robotically printed in spots. Each spot contains probes from the same gene. The target mRNA is reverse transcribed into the more stable cDNA (*complementary* DNA) and is therefore complementary to the original mRNA. The target mRNA comes from two different samples (often called the *test sample* and the *reference sample*) and is separately labeled with the two different fluorescent dyes Cy5 and Cy3. Cy5/Cy3 are chemical groups that emit red/green light when absorbing radiation of particular wavelengths. The two target samples are in solution and are simultaneously applied to the slide. The microarray is then scanned with a laser, and the two resulting images are analyzed using image analysis software. The intensity of the red and green light from each spot is assumed to be proportional to the amount of hybridized target cDNA labeled with Cy5 and Cy3, respectively. The expression level of each gene is presented as the ratio between the intensity of the red light and the green light, and hence reflects the expression level in the test sample relative to the expression level in the reference.

The most used technology besides that of DNA microarrays is the so-called *GeneChips* manufactured by Affymetrix [33]. This technology uses photo-lithographic techniques from the semiconductor industry to synthesize *oligonucleotides* on glass wafers. These oligonucleotide probes are in general much shorter than DNA probes (20-25 bases compared to 100-2000 bases) and hence less specific to one particular gene. However, oligonucleotides are more sensitive since such short probe strands only form stable double stranded DNA with target strands that match perfectly. Hence, oligonucleotides are more versatile and may be used for example to screen for DNA variations between individuals. Unlike DNA microarrays, oligonucleotide microarrays measure the absolute mRNA level and hence only need one sample. Another advantage is that probes may be synthesized directly from sequence databases, and do not need to be produced in advance. However, the oligonucleotide microarrays are considerably more expensive to produce than DNA microarrays.

A microarray study comprises a number of steps in addition to what has been described here. Obtaining the actual mRNA measurement is preceded by the experimental design (e.g. [25]) and followed by filtering and normalization of the data (e.g. [88]) and computational data analysis (e.g. [1, 87, 99]). Only aspects of the last step will be addressed in this study.

1.2.3 Crystallography and nuclear magnetic resonance

Protein structure is physically determined by *x-ray crystallography* [107] or *nuclear magnetic resonance* (NMR) [120]. Although other methods may give different and complementary information about the structure of proteins, including the primary and quaternary structure, crystallography or NMR are needed to obtain the secondary and tertiary structure since this requires determining the arrangement of atoms within proteins.

The x-ray crystallography method depends on placing a repeating array of many identical molecules (a *crystal*) in an x-ray beam and observing the *diffraction pattern*. The x-ray beam interacts with the electrons of all atoms in the crystal. These interactions scatter x-rays in all direction and only those positively interfering with each other give rise to diffracted beams that may be seen as spots in the diffraction pattern. To calculate the positions of each atom, the amplitude, wavelength and phase of the diffracted beams are needed. The amplitude is proportional to the intensity of the spot and the wavelength is set by the x-ray source, however, the phase is lost in the diffraction pattern. The so-called *phase problem* is a major problem of crystallography and may be solved by comparing the diffraction data from the original crystal with data from crystals modified with the addition of heavy atoms. An *electron-density map* is then calculated for the repeating molecule in the crystal and interpreted as a structural model. The quality of the model mainly depends on the errors in the phases and the resolution of the diffraction pattern, which in turn depend on the crystal quality. The model is subjected to a computational process where the atoms in the model are shifted about to optimize the fit between the model and the experimental data.

NMR measures the magnetic momentum or *spin* of certain atomic nuclei. Since the spin of atoms is affected by their bonds to other atoms, this method may obtain a list of distance constraints between the atoms of the molecule. A structural model of the protein molecule may then be calculated using these constraints. The main advantage of NMR over crystallography is that the proteins are in solution and do not need to be crystallized. The problems related to obtaining good crystals are the main restriction on the rate at which crystallography produces structural models. The main disadvantage of NMR is that the method cannot currently be applied to large protein molecules and that it requires the protein to have high solubility.

1.3 Bioinformatics

The development of genomics and high-throughput experimental technologies created the need for computers to store and analyze large amounts of data. As was the case for genomics, *bioinformatics* developed from being a discipline mainly associated with sequence databases and sequence analysis to a computational science using biological data to do e.g. functional genomics. Although different definitions and views of bioinformatics exist, most researchers now use bioinformatics as a generic term for both the storage and maintenance of biological data and the use of computational data analysis methods and algorithms in functional genomics-related studies [55]. Bioinformatics thus involves a number of scientific fields including mathematics, statistics, informatics, physics, chemistry, biology and medicine. It is the definition of bioinformatics as data analysis for functional genomics that will be emphasized in this study.

One commonly used methodology in bioinformatics and functional genomics is that of *machine learning*. Machine learning addresses the problem of using computers to *learn* general concepts from observations and knowledge, and has traditionally been developed in two different schools. Statisticians develop learning methods based on the mathematical frameworks of probability theory and statistics (see e.g. [40, 52]). Computer scientists often develop methods based on models of intelligent systems (e.g. methods inspired by biology such as genetic algorithms and neural networks, or methods based on logic such as rule learning, see the section on machine learning below) [74]. The differences are primarily due to the fact that statisticians have mostly been interested in pure data analysis, while computer scientist have also been interested in building intelligent systems (e.g. robots with *artificial intelligence* [91]). However, these different views are somewhat converging, forming hybrids using elements from both statistics and computer science (e.g. *pattern recognition* [111]).

Induction refers to generalizing from observations to broad concepts and differs from *deduction* that refers to using general concepts (or theories) to infer specific hypotheses. In molecular biology, induction is particularly relevant since the general theories have not yet been worked out. For example, we know that a relationship exists between sequence and structure, but this relationship is not well understood in terms of theories that may be used to deduce good structural models for a particular protein sequence. However, we do have examples of this relationship in terms of protein structures that are experimentally solved. And machine learning methods are designed to induce models based on *examples*, partially describing the assumed underlying

functional relationship between, in this case, sequence and structure. The most common application of such models is that of prediction. However, given a model that can reliably predict protein structure from sequence (in particular for unseen proteins, i.e. proteins that were not available when the model was induced), this model obviously includes general concepts that may also be used to understand the relationship. And this understanding may in time lead to general theories. Consequently, machine learning may be used both for *predictive* and for *descriptive* purposes. In molecular biology, and in particular in functional genomics, we will se that a number of problems may be addressed using the concepts of examples and machine learning. And successful application of such methods could lead to situations where biological experiments are used to obtain information on a (representative) set of cases, models are automatically induced from these examples and finally used to fill in the missing knowledge for the remaining cases. This is the philosophy of structural genomics mentioned earlier: to solve the structure of at least one protein from each structural class (e.g. fold, see the section on databases and annotations below) experimentally and to predict the structure of the remaining proteins using sequence similarity to proteins with solved structures.

One of the major obstacles for effective use of machine learning in functional genomics has been the lack of structure in the existing biological knowledge in terms of computer readable databases and annotations. Text mining and automatic inference from free text has therefore been one major part of bioinformatics and will continue to be so (for an overview see [98]). In what follows, a short introduction will be given to relevant databases and annotation efforts. This will be followed by an introduction to the most popular machine learning methods used for utilizing these resources in functional genomics. With this in place we will be ready to state the aims and contributions of this study.

1.3.1 Databases and annotations

The Internet provides the infrastructure for accessing and sharing biological information, and has been decisive in the development of functional genomics and bioinformatics. In general, we will divide biological information into measured, unprocessed *data* such as sequences and expressions, and human-processed *knowledge* such as gene function. Data are normally stored in publicly accessible databases, while most biological knowledge is available in terms of published articles. PubMed (http://www.ncbi.nlm.nih.gov/PubMed) is the main electronic free-text database providing access to all biomedical literature

in MEDLINE[8]. However, although PubMed in principle includes all available biological knowledge, this knowledge is not easily accessible at the large scale required by functional genomics studies. A biologist may read all articles relevant to one particular gene, but the task of extracting all relevant knowledge on all characterized genes for a genome-wide study is overwhelming. Additionally, this knowledge needs to be structured in a computer readable fashion so that, for example, expression data may be automatically correlated with gene function for a large number of genes. Hence, genomic studies have pushed the formalization of biological knowledge in terms of structured vocabularies that may be used for annotating the databases. A short overview of the most important and relevant databases and annotation efforts will be given next.

The International Nucleotide Sequence Database Collaboration (INSD) consists of DNA Databank of Japan (Japan, http://www.ddbj.nig.ac.jp/, [109]), GenBank (USA, http://www.ncbi.nih.gov/Genbank/, [7]) and EMBL [9] Nucleotide Sequence Database (Europe, http://www.ebi.ac.uk/embl/, [66]). These databases store and maintain all publicly available DNA sequences according to a commonly agreed-upon standard. In addition to sequences of characterized genes, the nucleotide sequence databases include a large number of so-called expressed sequence tags (EST) [11]. ESTs are short sub-sequences of expressed DNA and are synthesized using mRNA as a template (hence the name). Many of these ESTs are not linked to any characterized genes, and are used both for gene discovery and for designing probes for microarray experiments. Since ESTs are short sub-sequences, even non-overlapping ESTs may come from the same gene. UniGene (http://www.ncbi.nlm.nih.gov/UniGene, [114]) is an experimental system attempting to bring some order to the gene/EST sequence data by automatically clustering GenBank sequences into non-redundant sets that correspond to single genes.

Swiss-Prot (http://www.ebi.ac.uk/swissprot/, [5]) is a protein sequence database that together with the TrEMBL supplement (Translated EMBL Nucleotide Sequence Data Library, [5]) contains translated protein sequences for all DNA sequences in the nucleotide sequence databases. In addition, Swiss-Prot provides extensive annotation and cross-references to other

[8] MEDLINE is the literature database maintained by the National Library of Medicine (NLM) covering the fields of medicine, nursing, dentistry, veterinary medicine, the health care system and the preclinical sciences. It contains abstracts, MeSH terms and other publication details. MeSH is a controlled hierarchical vocabulary used to index the articles.

[9] European Molecular Biology Laboratory (EMBL).

databases. Both these databases are now integrated in UniProt (Universal Protein Resource, http://www.ebi.ac.uk/uniprot/, [3]).

The Protein Data Bank (PDB, http://www.rcsb.org/pdb/, [9]) is the major database for protein structures and provides 3D coordinates for all publicly known structures. The Macromolecular Structure Database (MSD, http://www.ebi.ac.uk/msd/, [12]) includes all proteins in PDB and provide extensive annotations and cross-references to other databases such as Swiss-Prot. In addition, two major classification trees exist for protein structures. SCOP (Structural Classification of Proteins, http://scop.berkeley.edu/, [77]) classify protein domains from PDB proteins into three major levels of increasing specificity:

- Fold: Domains are classified to the same fold if their main secondary structure elements have the same relative orientation and connectivity (Protein structure topology may be defined in terms of orientation and connectivity. Orientation refers to the direction of the structural elements in space, while connectivity refers to the order of these elements along the main chain, i.e. how they are connected by the main chain).
- Superfamily: Domains are classified to the same superfamily if their sequence identity is low, but structural and functional features indicate that a common evolutionary origin is probable.
- Family: Domains classified to the same family have a clear evolutionary relationship, and normally have sequence identity greater than 30% or, in some cases where sequence identity is lower, common structural or functional features that provide definitive evidence of an evolutionary relationship.

ASTRAL (http://astral.berkeley.edu/, [21]) provides non-redundant sets of SCOP protein domains and PDB coordinates for these domains. CATH (Class, Architecture, Topology and Homologous superfamily, http://www.biochem.ucl.ac.uk/bsm/cath/, [80]) is the other major classification tree for protein domains providing a similar classification tree to that of SCOP.

Gene expression data are now also published in databases. MIAME (Minimum Information About a Microarray Experiment, [14]) is a standard specifying the information that should be published together with a microarray experiment to facilitate correct interpretation and reproducibility. A number of public databases storing gene expression data are using the MIAME standard,

including ArrayExpress (http://www.ebi.ac.uk/arrayexpress/, [15]) and GEO (Gene Expression Omnibus, http://www.ncbi.nlm.nih.gov/geo/, [29]).

The main goal of functional genomics is the genome-wide determination of gene function. Gene Ontology (GO, http://www.geneontology.org, [4]) provides an organism-independent controlled vocabulary for describing the cellular roles of genes and gene products to this end. The ontology is divided into three parts:

- Molecular function: task performed by an individual gene product.
- Biological process: broad biological goal accomplished by an ordered assembly of molecular functions.
- Cellular component: subcellular location where a gene product is active.

Each of the three parts of GO is a *directed acyclic graph* (DAG)[10] where nodes are so called GO terms describing a particular aspect of a molecular function, biological process or cellular component and edges are either is-a or part-of[11] relations connecting two nodes. GO consequently describes cellular roles at different levels of generality and offers a powerful vocabulary for annotating gene products. An *annotation* in this context is simply an association between a gene or gene product and a GO term. An annotated gene should be associated with at least one GO term from each of the three GO parts (very often biologists find that several terms from each part are needed in order to describe the role of a gene product). Obviously, annotations will reflect the knowledge biologists possess about a certain gene product and may therefore vary in terms of how general they are. The GO graph, however, describes the relationship between different GO terms and therefore provides a way of comparing the annotations of two different gene products. The GO homepage provides annotations for a number of organisms made available by different collaborating groups. The MSD database (see earlier in this section) provides GO annotations for all characterized protein structures in PDB. Finally, there exist several other controlled vocabularies for functional annotation, most

[10] A graph is defined by a finite set of nodes connected by edges. A directed acyclic graph is a graph where the edges only have one direction (often symbolized by arrows) and where there is no path (i.e. set of connected nodes) starting and ending at the same node.

[11] The is-a relationship between two terms (or nodes) means that one term (the child) is a subclass of the other term (the parent) (e.g. mitotic cell cycle is-a cell cycle). The part-of relationship means that whenever the child exists, it is as part of the parent (but not necessarily the other way around) (e.g. cell cycle is part-of cell proliferation (i.e. cell growth through cell division)).

notably the MIPS[12] functional catalogue (http://mips.gsf.de/projects/funcat, [70]).

1.3.2 Machine learning

Machine learning deals with the problem of using computers to learn general concepts from *training sets*. A training set consists of a finite number of observations labeled or annotated with class knowledge and is assumed to constitute a partial description of an underlying functional relationship between the observations and the classes. In general, the labels may be continuous values or even more complex structures. However, in this section we will deal with the so called *classification* problem in which the training observations are assumed to belong to a finite set of classes and we want to learn a model or *classifier* capable of assigning an observation to one of these classes. Moreover, we will in general assume two classes, since problems with more than two classes easily may be reduced to a set of two-class problems.

Most machine learning methods represent the observations in terms of *features*. Each observation is a set of measurements, one for each such feature, collectively constituting a *feature vector*. Each observation may alternatively be viewed as a point in the multidimensional space spanned by the features (i.e. the *feature space*). Of course, not all classification problems are easily represented in this way, and choosing the right features is a very important issue specific to each classification problem.

The machine learning methods mainly differ in how they represent the induced model. A number of different designs exist with different advantages and disadvantages. A short overview will be given in the next paragraphs, emphasizing methods that are commonly used in relevant functional genomics studies outlined in Chapter 2 (see e.g. [52, 74, 111] or specified references for further reading).

Clustering methods

Methods for discovering natural, underlying classes from a set of observations are called *clustering* or *unsupervised learning*. These methods are used when no class knowledge is available. Consequently, methods utilizing labeled training sets are called *supervised learning* reflecting the conceptual idea that a supervisor provides the labels to the learning system.

[12] Munich Information center for Protein Sequences (MIPS)

Clustering methods are divided into *iterative* methods and *hierarchical* methods. The *k-means* algorithm is the most used iterative approach. It starts with a set of *k* randomly chosen clusters of observations and iteratively (a) calculates the center of each cluster (i.e. the *centroid*), (b) assigns each observation to the cluster defined by the closest centroid and (c) returns to (a) until no more observations change clusters. The centroid of a cluster and the closeness of two observations may easily be calculated in the feature space by using e.g. the notion of distance. The *k*-means algorithm is fast and uses little memory, but depends on the initial number and configuration of clusters. A well known related method is that of *self-organizing maps*.

The most popular hierarchical clustering method is *agglomerative* hierarchical clustering. It starts with the observations as single clusters and subsequently merges the two most similar clusters until all observations reside within one big cluster. The distance between two clusters may easily be calculated as the average distance between all pairs of observations in the two clusters (*average linkage*) or the longest/shortest distance between two observations in the two clusters (*complete/single linkage*). The result of the algorithm is a tree of clusters (*dendrogram*) illuminating the similarity structures in the data set. Since the method needs to compute and store the distance between all clusters, it is much slower and uses much more memory than for example the *k*-means algorithm.

Bayes classification rule

The *Bayes classification rule* states that an observation should be assigned to the class with the highest probability given the probability distribution of feature vectors in each class. It may be proven that this rule results in an optimal *error rate* for classification (i.e. fraction of training observations classified to the wrong class). However, the true probability distribution is normally not known and hence needs to be estimated. The difficulty of estimating the distributions from the training data is why other methods exist and often perform better on real world problems.

There are two basic concepts for estimating probability distributions from data; *parametric* and *non-parametric* methods. A parametric method assumes a distribution structure (e.g. the normal distribution) and calculates its parameters from the data (e.g. average and variance for the one-dimensional normal distribution). A non-parametric method is based on constructing *histograms* from the data using for example Parzen windows or k nearest neighbor density estimation, or simulation methods such as Monte Carlo simulation or bootstrapping. In the one dimensional case, a histogram is constructed by dividing the observations into bins and using the fraction of observations from

each bin as probability estimates. In the multidimensional case, however, bins are replaced by hypercubes (e.g. *Parzen windows*). If N observations are needed from each bin to get good probability estimates in the one dimensional case, N^n observations are needed in the n-dimensional case. The dramatic increase in the number of observations needed to get good estimates is often referred to as the "*curse of dimensionality*".

Linear classifiers

Linear classifiers use a line (in two dimensions) or a hyperplane (in multiple dimensions) to separate two classes of observations in feature space. These methods generally consist of a cost function (e.g. error rate) and an optimization algorithm which iteratively changes the parameters defining the hyperplane so that the cost function is minimized over the training set.

If linear classifiers do not yield good results, the problem might be that the classes are not linearly separable. *Artificial neural networks* (ANNs) are one popular method for nonlinear problems and are based on networks of so-called *perceptrons*. A perceptron is a simple computational unit that multiplies each input value with a weight and sums up the products. In principle, the output from the perceptron is 0 if the sum is less than a particular threshold and 1 otherwise. ANNs consist of layers of perceptrons, where the output of each perceptron in one layer is connected to the input of each perceptron in the next layer. The first layer (i.e. the *input layer*) consists of the same number of perceptrons as the number of features and the last layer (i.e. the *output layer*) consists, in the case of two classes, of one perceptron. The network is trained by iteratively inputting the feature vectors to the first layer, calculating the output of each perceptron until the last perceptron, comparing the output value with the true class label and updating the weights for each perceptron by propagating the error backwards in the network (the *backpropagation algorithm*). The training stops when the network is no longer improving its classification.

Another popular method for nonlinear problems is (nonlinear) *support vector machines* (SVMs). The SVMs first map the observations in the feature space into another space using a *kernel function*. A maximally separating hyperplane is then constructed based on the observations closest to the region that separates the two classes (the *support vectors*). The performance of SVMs greatly relies on the choice of kernel function and to what degree the kernel function is able to map the original classification problem into a linearly separable one.

Context-dependent classifiers

A classifier is *context dependent* if the classification does not only depend on the feature vector of one observation, but also on the feature vectors of the other observations and on the dependencies between the classes. The task then becomes to simultaneously assign a class sequence to a sequence of observations. This corresponds to the problem of optimally aligning two sequences and therefore often occurs in DNA and amino acid sequence analysis. One of the most common approaches to this problem is to assume that the class of one observation only depends on the class of the previous observation. This model is called a (first-order) *Markov model* and may be utilized to find the optimal class sequence with a reasonable amount of computation (using e.g. dynamic programming).

k-nearest neighbor classifiers

k-nearest neighbor approaches are based on classifying observations according to the class labels of the k closest training observations in the feature space. This is probably the simplest and most intuitive approach among all supervised methods, and is therefore commonly used.

Decision trees and rule-based classifiers

Decision trees and *rule-based* classifiers work on discrete (i.e. categorical) values or by dividing the feature space into boxes (two dimensions) or hypercubes (multiple dimensions), and by combining these into complex decision surfaces (i.e. surfaces in the feature space separating the classes).

Decision trees classify observations by sorting them down a tree from the root node to the leaf nodes, where the leaf nodes actually provide the classification. Each node corresponds to a feature and redirects the observations to different child nodes depending on their values for that feature. The tree is constructed top-down by iteratively selecting the most class-separating feature as a node.

A related approach is that of learning a set of IF-THEN rules. Note that a decision tree may be represented as a set of rules by translating each path in the tree (from root to leaf) into a rule. A number of rule learners exist, and a more detailed description of rough set-based rule learning will be given in Chapter 2.

Feature selection

Feature selection refers to the problem of selecting the most important features so as to reduce their number and at the same time retaining class separability allowing classification. There are a number of reasons for doing feature selection. The obvious reason relates to reducing the computational cost of

inducing classifiers. However, more important is the fact that the number of features translates directly into the number of *classifier parameters* (e.g. the number of perceptron/weights in an artificial neural network). And there is a fundamental principle in machine learning stating that the higher the ratio between the numbers of training examples and the numbers of classifier parameters, the better the induced classifier will perform on unseen observations (e.g. more observations per dimension/feature gives better estimates of the probability distribution and hence better performance using Bayes classification rule).

There are two broad approaches to feature selection. *Filter* methods select features according to some evaluation criterion (e.g. correlation between the feature and the class knowledge) and then induce a classifier based on these features. *Wrapper* methods use the classifier itself as the evaluation criterion, and select the features that result in the best classification performance.

Feature generation/extraction refers to constructing new features based on different combinations of the old features. One example is rotating the feature space to possibly obtain better class separation (e.g. using principle component analysis).

Bootstrapping, bagging and boosting

Bootstrapping [30] is a general re-sampling method that allows statistical inference about a summary statistic (e.g. sample mean) from a data set without knowing the sample distribution. The idea is to randomly draw with replacement a large number of new data sets from the original data set and to calculate the summary statistic from each such bootstrap sample. This provides several values for the summary statistic which may be used to infer for example its variance or confidence interval.

Bagging [16] and *boosting* [92] are general methods for improving the classification performance of any supervised method. Bagging (bootstrap aggregation) uses bootstrapping to sample a large number of training sets from the original set of examples. A model is induced from each such bootstrap sample and combined (aggregated) during classification to obtain what is often a better classification performance. Boosting is a similar method in which a weight is associated with each training example. Models are iteratively induced from the training set according to these weights and used to re-classify the examples. The weights are subsequently updated to put more emphasis on incorrectly classified examples. If the applied learning method cannot utilize the weights directly, bootstrap training sets may be constructed according to the

weights (i.e. each example is drawn with a probability corresponding to the weight).

Genetic algorithms

Genetic algorithms are used to solve search problems where solutions can be coded as strings of 0's and 1's. An initial population of solutions is generated randomly and the best solutions, according to some fitness function, are iteratively chosen to breed new generations of solutions using genetic operators such as mutation and crossover. Supervised learning involves a number of search problems that may easily be approached with genetic algorithms. One example is feature selection, where each solution may be interpreted as a mask for including or excluding features.

Time complexity

The *big O* notation is used to describe the worst case running time of an algorithm as a function of its input size *n*. For example, the agglomerative hierarchical clustering algorithm using single linkage has a time complexity of $O(n^2)$ (i.e. it computes the "all-against-all" distance between observations in feature space). Hence, if 100 observations take 10 seconds to cluster, then 10000 observations (which is a typical number of genes in a microarray experiment) take 27.8 hours.

Algorithms that have a worst case running time of $O(n^k)$, where k is a constant, are so-called polynomial-time algorithms. Problems for which no polynomial-time algorithm has yet been discovered are said to belong to the class of *NP-complete* problems (NP stands for non-polynomial). Such problems need to be approached with approximation algorithms that find "good enough" solutions. For example, finding the optimal subset of features (which is the goal of features selection discussed earlier) is NP-complete (i.e. it requires searching trough all 2^n-1 subsets and hence has a time complexity of $O(2^n)$). Feature selection may for example be approached with the wrapper method using a genetic algorithm, or with the filter method using the correlation coefficient between each feature and the class labels. The latter approach of reducing a multi-dimensional problem into considering one dimension at a time (starting with the "best" dimension) is often referred to as a *greedy* approach.

Classifier evaluation

A classifier is best evaluated by applying it to a set of unseen observations (i.e. a *test set*). To obtain good estimates of the true classification performance it is important to use a test set that is representative for the observations that the classifier is likely to encounter in the future. In practice, it is common to divide

the available labeled observations (i.e. examples) randomly into a training set and a test set. The training set is used to induce a classifier and the test set is used for estimating the classification performance. If few observations are available, which is commonly the case, *cross validation* may get the most out of the data in terms of performance estimation. *k*-fold cross validation refers to dividing the examples into *k* equally sized subsets and using one subset for testing and the rest for training. This is done repeatedly so that each subset acts as a test set once and is part of the training set *k-1* times. If *k* equals the number of examples, this method is referred to as *leave-one-out* cross validation. To get good estimates of the classifier performance it is important that information contained in the test set is not used in the training. For example, feature selection should be done after splitting the available examples into training and test sets. Doing feature selection on all available examples implies using the class knowledge contained in future test sets to induce the classifier and hence may lead to optimistic estimates of the true classification performance.

Performance measures and ROC analysis

A number of statistics exist for measuring the performance of a classifier on a test set. *Accuracy* is simply the fraction of test observations classified to the correct class (error rate = 1-accuracy). However, accuracy may provide insufficient information when the classes contain different numbers of examples or when making one type of error is more severe than making another.

Given two classes of positive and negative observations,

- *false positives* (FP) are negative observations classified to the positive class,
- *false negatives* (FN) are positive observations classified to the negative class,
- *true positives* (TP) are correctly classified positive observations and
- *true negatives* (TN) are correctly classified negative observations.

Furthermore, *sensitivity* and *specificity* are the fractions of correctly classified positive and negative observations, respectively (i.e. TP/(TP+FN) and TN/(TN+FP)). Many classification methods do not perform classification directly, but rather output a value representing the certainty that a test observation belongs to the positive class. Hence, we are left with the problem of choosing a certainty threshold for selecting the positive class as the

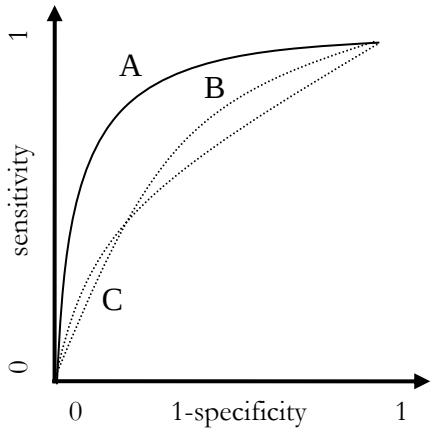


Figure 3. Example ROC curves. Clearly, classifier A performs better than both B and C. However, classifier B only performs better than C on low threshold values, while C performs better than B on high threshold values. Nonetheless, the AUC value of B is larger than that of C.

classification. The *receiver operating characteristic* (ROC) curve may be constructed by plotting sensitivity against specificity for the full range of possible threshold values (see Figure 3). A number of classification applications are associated with different costs for making a false positive classification compared to making a false negative classification. The ROC curve graphically displays the threshold-independent classification performance and provides a vehicle for controlling the number of false positives and false negatives. Increasing the threshold value reduces the number of false positives, but at the same time increases the number of false negatives. The *area under the ROC curve* (AUC, [39]) is often used to measure the threshold independent classification performance using one single number (i.e. AUC equal to 1 signifies a perfect discrimination of the positive and negative examples, while AUC equal to 0.5 signifies no discriminatory capability at all). The standard error of this measure is calculated using the Hanley-McNeil formula [39]. However, one should be aware that two ROC curves obtained using two competing classifiers may intersect and hence indicate that one classifier performs better for one range of threshold values, while the other performs better for another range of threshold values (see Figure 3). This information is of course lost when computing the AUC value.

Overfitting and classifier selection

A classifier is said to *overfit* the training set if there exists another classifier that performs worse on the training set, but better on the test set. A general principle for handling overfitting is related to the principle of *Occam's razor* which states that the simplest model fitting the data should be used. Hence, according to this principle we should for example use the artificial neural network with the fewest perceptrons classifying the training set satisfactorily. This principle also applies to choosing a classification method. One should for

example avoid using a nonlinear method on a linearly separable classification problem. This is of course related to the principle that the ratio between the number of training observations and the number of classifier parameters should be as large as possible (see the discussion in the feature selection paragraph above) More in-depth discussions on issues related to so-called *learning theory* may be found in e.g. [74, 111].

1.4 Aims and contributions of the study

At the core of functional genomics is the idea that sequence and sequence derived data may be used to automatically predict the function of uncharacterized genes and gene products at a genome-wide scale. This raises a number of interesting challenges including, but not restriced to, developing methods for:

- Predicting the cellular roles of genes from expression profiles.
- Dissecting the regulatory circuitry controlling the expression of genes.
- Predicting reliable protein structure models from sequences (i.e. the *folding problem*).
- Predicting protein function from structure.

By applying machine learning methods to training examples constructed from annotated sequence, structure or expression data, these challenges may be addressed. The resulting computational models will be constructed primarily for predictive purposes, but may also provide valuable insight into the biological mechanisms governing the modeled relationship. Without this insight, the predictions remain hypotheses to be experimentally validated by molecular genetics studies.

In particular,

> *the aims of this study were to provide novel contributions addressing the aforementioned challenges in functional genomics in terms of bioinformatics methods and tools that computationally learn from annotated sequence, structure and expression data.*

Novel contributions to this end include:

- A method, and a tool implemented in the ROSETTA system, for inducing rule models from gene expression time profiles predicting the participation of gene products in Gene Ontology biological processes.

- A template language for representing gene expression time profiles in terms of discrete changes in expression levels over different limited time periods.

- A rule based method for combining expression and sequence data to discover binding site modules shedding light on the combinatorial regulation of gene expression.
- An approach for validating the functional significance of binding site modules using Gene Ontology.

- A signal extraction method for retrieving sequence patterns governing protein structure at a local level represented by the novel concept of multi-fragment local descriptors of protein structure.
- A method for using sequence signals from local descriptors of protein structure to predict SCOP fold from sequence (i.e. fold recognition).

- A method, and a tool implemented in the ROSETTA system, for inducing rule models which predict Gene Ontology molecular function from local descriptors of protein structure.

Methods and results are presented in detail in six independent research papers summarized in the next chapter.

Other papers and short papers published during the course of this study are included in the references as [47, 60, 61, 65, 116].

Chapter 2 Methods and results

This study consists of six independent papers. These papers may be divided into four groups:

- Papers I and II describe a method for inducing rule-based models for predicting biological processes from gene expression time profiles, and provide extensive biological analysis showing the validity of the predictions.
- Paper III describes a method for using rule-based learning to discover (functional) binding site modules from gene expression and sequence motif data.
- Papers IV and V describe a method for sequence signal extraction and fold recognition using the novel concept of local descriptors of protein structure.
- Paper VI uses the concept of local descriptors of protein structure to induce rule-based models for prediction of protein function from structure.

This chapter will give an overview of the six papers. The overview will be divided into four natural parts as indicated above, and each part will provide a short overview of related research together with the methods and results from the papers. Chapter 1 should have provided the reader with the basic concepts in molecular biology and bioinformatics needed to understand the material presented here. A discussion of the main results in the papers will follow in Chapter 3.

2.1 PAPERS I & II: Predicting biological process from gene expression time profiles

Papers I and II describe a rule learning approach based on the rough set theory to model and predict the participation of gene products in biological processes from gene expression time profiles (earlier versions of this method were published as Komorowski *et al.* [60, 61] and Hvidsten *et al.* [47]). Gene Ontology describes three aspects of the cellular roles of genes including molecular function, biological process and cellular component. Molecular functions describe the tasks performed by single gene products and should

therefore be related to structural features important for their interaction with other molecules. Biological processes, however, describe ordered assemblies of several different molecular functions and should therefore only be related to the simultaneous regulation of genes participating in these processes. To this end, we developed a template language describing the discrete changes in expression over subsets of time points in an expression time profile. The idea behind this language is that the relative change in mRNA levels over limited periods of time is more important to distinguish one biological process from another than the absolute mRNA levels given by each time point. Moreover, we applied the rule induction framework to obtain IF-THEN rules associating particular *combinations* of discrete changes in expression with one or a small number of biological processes. The predictive performance of the approach was tested using cross validation on two expression time profile data sets with human genes. Paper I describes in detail the mathematical framework for inducing rule models, while paper II provides extensive biological evaluation of the predictions.

2.1.1 Related research: Functional gene expression analysis

A number of studies have used hierarchical clustering to elucidate the correlation between clusters of similarly expressed genes and functionally related genes (e.g. [19, 31, 48]). These studies indicated the possibility of using clusters of functionally related genes to predict the function of the remaining uncharacterized genes in those same clusters. However, functionally related genes are often anti-coregulated and moreover genes are often associated with several functions [97]. These aspects are not well modeled by a set of relatively broad non-overlapping expression clusters. Gasch and Eisen [34] addressed some of these problems using a fuzzy k-means algorithm which allows genes to be members of several expression clusters. Furthermore, Cho *et al.* [23] and Wu *et al.* [119] moved towards full cluster-based prediction by actually assigning confidence to how well expression clusters corresponded to MIPS functional categories. In particular, Wu *et al.* [119] used several clustering methods including hierarchical clustering, k-means and self-organizing maps to obtain a large number of relatively specific expression clusters. These clusters were functionally annotated and used for prediction.

Brown *et al.* [18] introduced a supervised method using support vector machines to predict six MIPS functional categories from expression data. This study also evaluated other supervised methods using cross validation, including Parzen windows, linear classifiers and decision trees. In addition, Brown *et al.* [18] provided experimental results suggesting that it is much easier to predict the participation of genes in biological process than to predict the exact

molecular function from gene expression data. The same data was later analyzed by Pavlidis *et al.* [84] using support vector machines, k-nearest neighbor approaches and a probabilistic model.

Midelfart *et al.* [71-73] introduced rough set-based rule classifiers to actively learn in the Gene Ontology graph, dynamically selecting biological processes with the best predictive performance. These studies used the template language introduced here to represent gene expression time profiles.

2.1.2 Data: Expression data and Gene Ontology annotations

Paper I applied the method to the fibroblast cell cycle expression data published by Cho *et al.* [23]. The study included expression time profiles for 6800 genes with measurements taken every other hour from 0 to 24 hours. We obtained 7679 Gene Ontology annotations to biological process for 3620 genes from the euGenes database (http://eugenes.org:8089/, [36]). From these annotations we extracted 27 broad biological processes which included 3043 genes with 5521 annotations.

Paper II analyzed predictions resulting from applying the method to the expression data published by Iyer *et al.* [48]. Variation in the expression levels of 497 genes were found during the first 24 hours of the serum response in serum starved human fibroblasts (12 measurement points). Gene Ontology biological processes were assigned by manually extracting information from databases such as UniGene and Swiss-Prot, and from literature. These annotations resulted in 23 broad biological processes which included 273 genes with 549 annotations.

The broad biological processes used for learning were selected to include as many annotated genes as possible with a minimal overlap and to be as specific as possible without having too few examples from each class. As mentioned earlier, Midelfart *et al.* [71-73] later introduced a method in which classes were selected iteratively in the Gene Ontology according to their learnability.

2.1.3 Method: Rough set-based rule induction and the ROSETTA system

A general introduction to the learning framework will be given first, followed by the specifics of applying this framework for learning biological processes from expression time profiles. Details may be found in paper I.

The rough set framework and the ROSETTA system

Pawlak's *rough set theory* [62, 82, 83] and *Boolean reasoning* [17] constitute a mathematical framework for inducing rules from examples. This framework is implemented in the ROSETTA system, a publicly available toolkit for data mining and knowledge discovery using rough sets (http://rosetta.lcb.uu.se, [63, 124]).

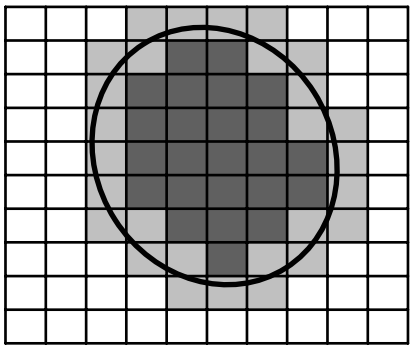


Figure 4. The rough set (the ellipse) cannot be uniquely defined by the equivalence classes (the squares), and is defined by the lower approximation (dark grey) and the upper approximation (dark plus light grey).

The *rough set theory* is a mathematical framework for analyzing tabular data. An *information system* is a table with observations (called *objects*) as rows, features (called *attributes*) as columns and discrete values as entries. The theory sees the data in terms of *equivalence classes*, i.e. sets of objects that are indiscernible (indistinguishable) with respect to the attributes. A rough set is a set of objects that cannot be uniquely represented by these equivalence classes since the set only partly overlaps with at least one of them. It may hence only be approximately described either by the equivalence classes completely contained in the set (the *lower approximation*) or the equivalence classes with at least one object in the set (the *upper approximation*) (see Figure 4).

The *decision attribute* is a unique attribute dividing the objects into *decision classes* and is provided by domain experts or a separate source of information. The information system with the decision attribute constitutes the training set or the so-called *decision system* (see Table 1 for a specific example). In particular, decision classes may be rough in which case the class knowledge itself cannot be uniquely represented using the data in the information table.

The ROSETTA system implements a *Boolean reasoning* approach to inducing *minimal decision rules* from decision systems. It constructs a *Boolean function* (i.e. a function that evaluates to true or false), called the *discernibility function*, for each object, which is true for all attribute combinations that discern this object from objects with a different decision. The function is then simplified and its minimal solutions interpreted as so-called *reducts* [106]. A reduct is hence a minimal set of attributes discerning one object from all objects with a different

decision. The reducts may be approximate (*approximate reducts*) in which case a sufficiently large fraction of objects from other decision classes is discerned [105].

IF-THEN rules are constructed by reading of the values for each attribute in the reduct (IF-part called *antecedent* or *premise*, e.g. $a_1=v_1$ AND $a_2=v_2$, where a_1 and a_2 are attributes in a reduct and v_1 and v_2 are attribute values) and associating them with one or more decision classes (THEN-part called *consequent*, e.g. $d=d_1$ OR $d=d_2$, where d is the decision attribute and d_1 and d_2 are decision classes). The THEN-part will only include one decision class unless the decision class is rough with respect to the attributes in the reduct. Rules are evaluated according to how general they are (i.e. *coverage*: the fraction of objects from the decision class in the THEN-part that also matches the IF-part) and how specific they are (i.e. *accuracy*: the fraction of objects matching the IF-part that are from the decision class of the THEN-part) (both coverage and accuracy are computed for each decision class in the THEN-part). Objects are classified by first identifying the rules with a matching IF-part and then by letting these rules cast votes to the decision classes in the corresponding THEN-parts. The number of votes cast by each rule corresponded to the *support* of the rules (i.e. the number of objects matching both the IF- and THEN-part of the rule), giving preference to rules that are general. Decision classes obtaining a fraction of votes higher than the voting thresholds from the ROC analysis are considered predictions (see section 1.3.2). If no particular costs are associated with making false positive classifications over false negative classifications, the threshold corresponding to the point on the ROC curve balancing sensitivity and specificity equally is often chosen (i.e. the point closest to (0,1) or, equivalently, the "northwestern-most" point on the ROC curve).

Finding *all* reducts is a NP-complete problem [106]. However, the ROSETTA system implements a number of approximation algorithms, including greedy algorithms [51] and genetic algorithms [118], searching for reducts. They are all based on constructing the discernibility function which has a time complexity of $O(n^2)$. The system also includes a number of other tools supporting the actual rule induction, including algorithms for discretization, rule filtering and classification, and methods for evaluating classifiers such as cross validation and ROC analysis.

Application to expression time profiles and biological process annotations

In the specific case of modeling biological processes from expression time profiles, information tables were constructed from the expression data using the template language. That is, genes were objects, time windows or

subintervals in the expression time profile were attributes and the templates matching the gene expression profiles in the particular subintervals were entries. We used three templates defining increasing, decreasing and constant expression level, and one entry value defining no match to any template. Decision classes were defined corresponding to the selected broad biological processes, and the rule induction framework was applied to the corresponding decision table using a genetic algorithm to find approximate reducts.

Table 1. An example decision table. Objects are genes, attributes are limited time periods and the decision attribute is Gene Ontology annotations to biological processes.

Genes	...	0h-4h	...	6h-10h	...	14h-18h	...	Biological process
Y07909	...	increasing	...	decreasing	...	constant	...	cell proliferation
X58377	...	increasing	...	decreasing	...	constant	...	cell proliferation
U66468	...	increasing	...	decreasing	...	constant	...	cell proliferation
X58377	...	increasing	...	decreasing	...	constant	...	cell-cell signaling
X85106	...	increasing	...	decreasing	...	constant	...	intracellular signaling cascade
Y07909	...	increasing	...	decreasing	...	constant	...	oncogenesis
...	...	...	...	...	...	...	...	...

An example from the data studied in paper I is shown in Table 1. The table illustrates a reduct of three limited time periods discerning *cell proliferation* genes from genes with other decisions. Six gene-annotation pairs are indiscernable with respect to this reduct. Note that the annotations to *cell-cell signaling* and *oncogenesis* are a result of the fact that genes X58377 and Y07909 have more than one annotation. However, the expression profile of gene X85106 with an annotation to *intracellular signaling cascade* could not be discerned from the cell proliferation genes using the template language. The resulting rule

IF 0h-4h (increasing) AND 6h-10h (decreasing)
AND 14h-18h (constant)
THEN GO (cell proliferation) OR GO (cell-cell signaling)
OR GO (intracellular signaling cascade)
OR GO (oncogenesis)

describes the limited set of biological processes (THEN-part) associated with particular expression profile constraints (IF-part, e.g. 0h-4h (increasing) means increasing expression level from 0 to 4 hours). During classification this rule will cast three votes to cell proliferation and one vote to the other three biological processes.

2.1.4 Results

Paper I provided a detailed description of the method used to analyze expression data in both papers. The cross validation results indicated the ability

of the approach to find general expression profile features that may be used to classify unseen gene profiles. Both papers obtain good cross validation AUC values for a number of biological processes (results where shown to be statistically significant using the randomization test introduced in paper VI and explained in section 2.4.3, see http://www.lcb.uu.se/~hvidsten/fibroblast/ (not published)).

The cross validation results may in general be considered estimates for the prediction quality of the uncharacterized genes using a model induced from *all* training examples. In addition to predict the biological process of uncharacterized genes, we also used a model induced from all examples to re-classify characterized genes. In paper II, false positives were used to guide a second literature search for possible missing annotations (i.e. information on biological process annotations existing in the literature, but overlooked during the initial literature search). Of the 14 genes with a false positive re-classification to *DNA metabolism*, 4 were found to actually participate in this process. Furthermore, it was revealed that 12 of the 24 false positive re-classifications to *oncogenesis* also represented missing annotations. It is not surprising that annotations in general may be incomplete either because not everything is known about these genes or because information was missed by the annotator. Our results indicated that these information holes may be filled using false positives (provided by the rule models) as a guide for conducting new literature searches.

To evaluate the predicted biological processes for uncharacterized genes, we searched for homology information that could be used to make assumptions about the biological processes of these genes. Of the 24 genes where such assumptions could be made in paper I, 11 genes had one or more classifications that matched this assumption.

The methodology is implemented as a package in the ROSETTA system with a graphical user interface providing a number of options such as specifying parameters in the template language, the cost on false positives for the ROC-analysis, the number of iterations in the cross validation and the degree of approximation in the genetic algorithm searching for reducts. The system then outputs cross validation estimates, re-classifications for the characterized genes and predictions for the uncharacterized genes. It also provides an option to run randomization tests to estimate the statistical significance of the cross validation results. The system has been used in several student projects and exercises [13, 46, 50]. The LCB Datawarehouse (https://dw.lcb.uu.se/), a platform for microarray gene expression data analysis, provides the biologists with Gene

Ontology annotations and an option to export data that may be directly analyzed using the ROSETTA system.

2.2 PAPER III: Discovering regulatory binding site modules

Paper III described a rule learning approach to discovering regulatory binding site modules. Transcription factors regulate gene expression by binding selectively to sequence sites in regulatory regions of genes. In order for a limited number of transcription factors to respond to a large number of stress conditions, transcription factors combine in different ways to facilitate a large number of different expression outcomes. It is reasonable to assume that genes regulated by the same combination of transcription factors (*co-regulated*) through the corresponding binding sites in their regulatory regions also exhibit similar expression profiles (*co-expressed*). Consequently, we may analyze the combinatorial nature of gene regulation using this assumption and genome-wide sequence and expression data. We obtained IF-THEN rules linking binding site combinations (*binding site modules*) to genes with similar expression profiles. Discovered binding site modules were evaluated using binding interaction probabilities from a genome-wide location analysis and annotations from Gene Ontology.

2.2.1 Related research: Combinatorial gene regulation

Several studies have used cluster analysis to find potential binding sites by mining the sequences of co-expressed genes for common sequence motifs (e.g. [8, 27, 34, 117]). Such methods provide the basic data for investigating the combinatorial nature of gene regulation. Pilpel *et al.* [86] provided evidence for the existence of combinatorial interaction between transcription factors by observing a significant increase in expression similarity between genes sharing one common transcription factor binding site and genes sharing a pair of binding sites. The study provided a simple, yet effective, method for proving the combinatorial nature of gene regulation in yeast. Segal *et al.* [95] used a probabilistic algorithm to obtain sets of genes that are co-regulated (gene modules) through a combination of sequence motifs. The algorithm first clusters expression data into gene modules and then selects motif combinations for each module. It then iteratively moves genes between modules to optimize the degree to which selected motifs explain the expression profiles in the modules. Segal *et al.* [94] used the same methodology to build gene module networks using gene expression data and candidate regulators such as known transcription factors or signaling proteins. Beer and Tavazoie [6] built similar networks using expression data and sequence motifs.

2.2.2 Data: Expression and sequence motif data

We tested our methodology using the binding site database established by Hughes *et al.* [44] containing information on 43 known binding sites and 313 putative motifs and their occurrences in the promoters of all genes in the yeast genome. We also used expression profiles of yeast genes under six different sets of conditions; cell cycle [22], sporulation [24], diauxic shift [27], heat and cold shock [31], pheromone [89], and DNA-damaging agents [49].

For evaluation we used yeast Gene Ontology annotations from the Gene Ontology homepage and results from a genome-wide location analysis providing probabilities for the actual binding interaction between all known transcription factors and promoters in yeast [68].

2.2.3 Method: Rule learning in gene regulation

For the gene regulation analysis in paper III, we constructed an information table with genes as rows, binding sites as columns and entries 1 or 0 depending on whether the binding site was present in the promoter region of the corresponding gene or not. For each gene we identified other genes with a similar expression profile using a threshold value for the Euclidean distance in the expression space. The corresponding expression clusters were used as decision classes. The learning framework (see section 2.1.3) was then subsequently applied to each gene to obtain rules associating a minimal binding site combination with particular expression clusters. We then removed rules that did not provide clear patterns in terms of a binding site module associated with several genes where a majority had similar expression. For example, the rule

IF RAP1 AND SWI5 AND MCM1' **THEN** similar expression

associates known binding sites RAP1, SWI5 and MCM1' with similar expression profiles in five of six expression data sets

To evaluate the discovered binding site modules, we calculated P-values for each Gene Ontology term and each transcription factor (using bindings hypothesized by the genome-wide location study). Each P-value corresponded to the probability of observing at least the observed number of genes being annotated or bound by the same Gene Ontology term/transcription factor. We then identified the fraction of significant rules from each data set, and compared this fraction to the corresponding fractions from randomly sampled sets of genes with common binding sites, similar expression or neither.

2.2.4 Results

Paper III showed that genes associated with a discovered binding site module have a significantly higher probability of being bound by the same transcription factors, as hypothesized by the genome-wide location analysis, and of being biologically related in terms of Gene Ontology annotations than genes associated with common binding sites or similar expression alone. The method thus provides specific hypotheses on co-regulation that may later be experimentally verified.

The rule learning approach differ from the approaches taken by Segal *et al.* [95], and to some degree Beer and Tavazoie [6], in that we explored a large number of relatively small, overlapping clusters rather than trying to explaining co-regulation using a set of broad non-overlapping gene clusters. We then selected instances where substantial evidence for co-regulation existed. Applying this method to expression data obtained under several different stress conditions resulted in a number of binding site modules common to several of these responses in addition to modules that seemed to be exclusive to a particular stress response. The overlaps between modules clearly showed the large extent to which relatively few transcription factors combine to facilitate a much large number of expression outcomes.

2.3 PAPERS IV & V: Fold recognition from local descriptors of protein structure

Since the experimentally derived protein structures in PDB cover only a fraction of the proteins that have been sequenced, methods for the automatic prediction of protein structure from sequence is of great general interest. However, also the limited problem of assigning sequences to already defined classes of topologically similar proteins (folds) is of great practical interest. Protein domains with similar folds often share the same molecular function [41, 75, 96] and therefore the ability to reliably recognize fold from sequence may be of great help in a number of research areas, including basic understanding of molecular biology.

a)

Descriptor	Segment 1		Segment 2		Segment 3		Segment 4		Segment 5	
1qgoa_#8	4-10	ALLVVSF	39-43	FRAFT	63-67	LQALQ	77-83	VAIQSLH	91-95	EKIVR
1qo2a_#78	74-80	EHIQIGG	47-51	IHVVD	67-71	EKLSE	97-101	--RRQIV	89-93	EKLRK
1rpxa_#73	69-75	LPLDVHL	40-44	IHVDV	58-62	LVVDS	93-97	--DIVSV	85-89	PDFIK
1nsj__#82	78-84	NAVQLHG	58-62	GVFVN	66-70	EKILD	98-104	ILVIKAV	89-93	ELCRK
1mla_1#7	3-9	QFAFVFP	87-91	MMAGH	262-266	EYMAA	270-276	EHLYEVG	283-287	GLTKR
1qfja2#108	104-110	PMILIAG	134-138	TIYWG	183-187	TAVLQ	195-201	HDIYIAG	207-211	KIARD
1efvb1#8	4-10	LRVLVAV	119-123	LVLLG	47-51	EEAVR	59-65	KEVIAVS	76-80	RTALA
1iow_1#8	4-10	KIAVLLG	38-42	YPVDP	48-52	TQLKS	56-62	QKVFIAL	70-74	GTLQG
1yaca_#57	53-59	PTILTTS	80-84	PYIAR	97-101	VKAVK	14-20	AVLLVDH	120-124	AFPAL
1ig0a2#188	184-190	ISLLALG	40-44	TLLIL	128-132	TKCVN	216-222	FKLCYMT	200-204	VHSIT

b)

c)

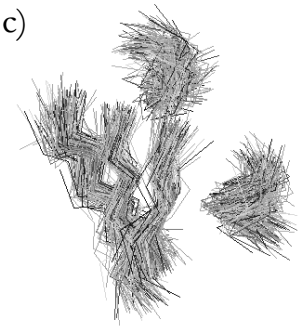


Figure 5. An example descriptor group. a) shows the sequence fragments associated with the local substructure in b) and c). b) shows the structure of the central descriptor of the group, while c) shows the structure of all similar descriptors. Descriptors are named using the syntax "domain name"#"position of central amino acid".

Papers IV and V described a method for retrieving sequence patterns (*signals*) governing protein structure at a local level and a method for using these sequence patterns to predict SCOP fold from sequence (i.e. *fold recognition*). For this purpose we used the novel concept of *local descriptors of protein structure* introduced by Kryshtafovych and Fidelis [64]. A local descriptor of protein structure is a set of short continuous backbone fragments centered in three dimensions around a particular amino acid (see Figure 5b). We used a library of popular local descriptors (called *descriptor groups*) to study sequence patterns responsible for these particular geometrical confirmations (see Figure 5ac). We will frequently refer to descriptor groups as *local substructures*.

2.3.1 Related research: Protein structure prediction

In general, protein structure prediction methods may be divided into the three broad categories of *comparative modeling*, *fold recognition*, and *ab initio prediction*. Comparative modeling and fold recognition both rely on identifying one or more potential *templates* (i.e. proteins that are structurally similar to the target) and then on building a model by transferring the structural information from

the template(s) to the target. However, the fold recognition methods attempt to detect the right templates even in cases of little or no observable sequence similarity to the target protein. The *ab initio* prediction methods forgo the necessity of identifying a template altogether (see fragment-based methods below). CASP (Critical Assessment of Techniques for Protein Structure Prediction, http://predictioncenter.llnl.gov) has shown that there is progress on fold recognition targets both for fold recognition and *ab initio* prediction methods [59, 76, 103, 113]. However, homology modeling is still much more reliable when good templates exist [112].

Position specific scoring matrices (PSSMs) are a commonly used vehicle for modeling the sequence content of multiple alignments. Each entry in such a matrix is based on the estimated probability of a particular amino acid (column) occurring in a specific position in the alignment (row). This probability may be estimated using only the observed frequency of this amino acid or also using *prior* knowledge on substitution likelihoods (from so-called *substitution matrices* or using e.g. Dirichlet mixtures [104]). Including prior knowledge is particularly relevant when few observations are available. A multiple alignment may be matched to a sequence or another multiple alignment by calculating the sum of the products of each corresponding entry. Another approach is *Hidden Markov Models* (HMMs) where the multiple alignment is modeled by a set of states (emitting an amino acid with a probability) connected by transition probabilities (i.e. the probability of moving to a new state given the current state) (see context dependent classifiers in section 1.3.2).

PSI-BLAST [2] is used to build multiple alignments between a target sequence and sequences in a database, and is a powerful tool for inferring structural, functional and evolutionary information from already characterized sequences. PSI-BLAST constructs a PSSM from sequences in the database with the best score and then iteratively constructs new PSSMs from sequences found using the PSSM from the previous iteration. PSI-BLAST uses an E-score that equals the number of matches expected by chance with a score equal to or greater than the actually obtained score.

The aim of so-called *fragment-based* methods is to predict structure for targets with little or no detectable sequence similarity to templates of known structure (fold recognition) or where templates of known structure do not exist at all (*ab initio* prediction). Even in the latter case, the structural elements needed to assemble the protein structure may exist in the structure databases.

Baker and others [20, 101, 102] used single backbone fragments to assemble structural models. In particular, Simons *et al.* [101, 102] used fragment of similar local sequence to assemble protein structure models with the help of a scoring function derived from proteins of known structure. Bystroff and Baker [20] employed an iterative approach to find sequence fragments that correlated strongly with structure. Karplus, Karchin and others [56, 57] described a method using not only single fragments but local 3D environments of secondary structure. The descriptor approach differs from all these methods in that it considers structural motifs incorporating all fragments in a structural neighborhood. Furthermore, as we will see, these structural motifs are selected without using any sequence information.

2.3.2 Data: Local descriptors of protein structure

374,558 local descriptors of protein structure with at least 3 non-overlapping backbone fragments of 5 or more residues were found in 4006 protein domains of known structure with less than 40% sequence identity to each other (ASTRAL version 1.57). A library of popular local substructures was built by first grouping structurally similar descriptors and then selecting a representative subset of 4197 descriptor groups containing at least seven descriptors (for details see Kryshtafovych and Fidelis [64]).

In both papers IV and V we used a fold-oriented version of the library described above. This library consisted of 4084 representative groups containing only descriptors from protein domains classified to the same SCOP fold. Paper V analyzed a subset of 3793 groups from the 135 SCOP folds with at least five fold-oriented descriptor groups. The test set for estimating the performance in fold recognition included all protein domains from the 135 folds in ASTRAL version 1.59 and 1.61 with less than 40% sequence identity to any protein in the training set (i.e. ASTRAL version 1.57). Paper IV analyzed 2537 groups from the 49 SCOP folds with at least five fold-oriented groups containing at least 20 descriptors. The test set included all protein domains from the 49 folds in ASTRAL version 1.59 and not in the training set.

2.3.3 Method: Fold recognition using local descriptors of protein structure

The descriptor groups in the library consist of one central descriptor called the *seed* descriptor and a number of aligned descriptors structurally similar to this seed. In the context of groups we refer to sequence fragments as *segments* and the alignment between corresponding segments from different descriptors as *multiple segment alignments*. The multiple segment alignments provide links

between particular local substructures and several examples of sequences from which we may extract the governing sequence signals (see Figure 5 from paper IV). We based the signals on observed frequencies of amino acids or amino acid *substitution groups* (i.e. groups of amino acids known to be substituted in structurally conserved regions of proteins) in specific position of the multiple alignment, and extracted sequence profiles or position specific scoring matrices from each segment alignment. The profiles from each group were then matched with sequence profiles from multiple alignments constructed from the target protein using PSI-BLAST. Target specific descriptors were assembled from the best matching segment alignments, thereby indirectly assigning a local substructure to the target. The score of the match between the group and the target was defined as the sum of the scores of each segment alignment weighted by the secondary structure agreement. The secondary structure of the target was predicted with PSIPRED ([54], using an artificial neural network with the position specific scoring matrix from PSI-BLAST as input).

Learning from the protein domains in the training set, we obtained acceptance thresholds for each descriptor group indicating when a score between a group and a target was sufficiently high to conclude that the structure of the target in fact included the local substructure of the group. Fold recognition was carried out by matching each group to the target, keeping the ones with a matching score higher than the acceptance threshold and computing P-values for each SCOP fold. Each P-value equaled the probability of by chance assigning the same or a higher number of groups from that fold than what was actually assigned using the sequence signal.

2.3.4 Results

Paper IV described a fold recognition approach using a large number of substitution groups [121], a genetic algorithm for selecting discriminatory subsets of significant signals and boosting for extracting several signal profiles per group.

Paper V systematically showed (a) that there exists a significant sequence signal in the descriptor group library governing structure at a local level, (b) that this signal is strong enough to allow assigning local substructure to the correct domain structures, (c) that it is possible to structurally align the local substructures along the main chain of a protein and (d) that it is possible to recognize the SCOP fold of the target protein. In particular, it was shown that the descriptor approach may provide good fold recognition for a number of protein domains where the sequence identity to any training domain was so low that PSI-BLAST could not find a relation.

Paper V also evaluated the signal extraction methods used in PSI-BLAST and provided experimental results indicating that the approach taken in paper IV performs better in the descriptor library.

2.4 PAPER VI: Predicting molecular function from local descriptors of protein structure

Structural genomics projects systematically solve experimentally at least one structure from each fold, then infer structure for the remaining proteins using comparative modeling and ultimately infer function from structure [37, 122]. Also the last problem of predicting function from structure poses a number of challenges [81]. Although proteins from the same fold often are functionally related [41, 75, 96], structural similarities in the absence of any evolutionary relationship (as observed in terms of sequence similarity or particular structural features) are not always sufficient to reliably assume that two proteins perform similar functions. Nonetheless, since structure is more highly conserved than sequence and since only a few residues in the so called *functional sites* on the protein surface need to be conserved for the function to remain stable during evolution, functional prediction from structure often has great advantages over prediction from sequence.

Paper VI used the concept of local descriptors of protein structure to induce rule models for the prediction of Gene Ontology terms. Descriptor groups (as described in section 2.3) represent popular local substructures in proteins, and may directly or indirectly correspond to important functional features characteristic to proteins with similar Gene Ontology annotations. By combining the occurrence of local substructure into IF-THEN rules we obtained a powerful model of the structure-function relationship in proteins.

2.4.1 Related research: Structural motifs

Structural motifs have been used for aligning protein structures [69, 79] and may also be used to search for structural features related to functional sites. The PROSITE database (http://ca.expasy.org/prosite/, [45]) contains biologically significant sequence patterns or sites. Kasuya and Thornton [58] associated structural fragments to these patterns. Other methods, however, investigated the structure-function relationship in proteins without using any such prior knowledge. Jonassen *et al.* [53] automatically found structural motifs using sequence patterns, and found that many of them corresponded to patterns in

PROSITE. Russell [90] did not use sequence data at all and could hence identify structural motifs that were a result of convergent evolution[13].

2.4.2 Data: Local descriptor groups and Gene Ontology annotations

We used the library of 4197 descriptor groups to represent all 4006 protein domains in ASTRAL version 1.57 in terms of local substructures (see section 2.3.2. for more details). These 4006 domains corresponded to 2878 proteins, of which 1896 were annotated with 5866 molecular function annotations in the Macromolecular Structure Database (MSD, see section 1.3.1) database. We extracted 72 specific Gene Ontology molecular function classes with at least 10 proteins. Together these classes contained 1576 proteins with 2965 annotations. Corresponding classes were extracted for biological process (121 classes containing 1613 proteins with 5797 annotations) and cellular component (19 classes containing 778 proteins with 1134 annotations).

2.4.3 Method: Rule models for learning function from structure

We constructed an information table with proteins as rows, descriptor groups as columns and entries 1 or 0 depending on whether the protein structure contained the local substructure of the group or not. We then applied the learning framework (see section 2.1.3) to obtain rules modeling the relationship between structure and different Gene Ontology annotations. For example,

IF 1gsa_2#218 AND 1ra9__#62 **THEN** GO (oxidoreductase activity)

associates the occurrence of both the local substructure centered on amino acid 218 in domain 1gsa_2 and the local substructure centered on amino acid 62 in domain 1ra9__ with the Gene Ontology annotation *oxidoreductase activity, acting on the CH-NH group of donors, NAD or NADP as acceptor.*

The predictive performance of the models was evaluated using 10-fold cross validation and ROC analysis as in papers I and II. The large number of structural features (i.e. local substructures) was reduced using a supervised feature selection method. We tested the statistical significance of the cross validation AUC values by calculating P-values for each class. This was done by randomly shuffling the annotations, performing cross validation on these

[13] Two structures with similar structural features may either be a result of divergent evolution (i.e. they evolved from a common ancestor) or of convergent evolution (i.e. they evolved from different ancestors). Only in the former case may one expect to observe significant sequence similarity.

random data sets and calculating the fraction of cross validation AUC values equal to or higher than each AUC value from the original annotations.

2.4.4 Results

Paper VI showed that a large part of the descriptor groups contained a statistically significant number of annotations to at least one of the selected Gene Ontology classes, especially in the cases of molecular function and biological process. However, cross validation showed a much stronger ability to induce general rules that could predict unseen proteins for molecular function than for biological process and cellular component. This is expected, since molecular functions describe particular tasks performed by proteins, and should therefore be related to structural features important to for example interactions with other macromolecules, while biological processes describe ordered assemblies of several molecular functions. In fact, a number of molecular function classes could be predicted with high cross validation AUC (statistically significant) and hypotheses on the molecular function of uncharacterized proteins could be put forward. These hypotheses were then compared to newer, unseen annotations, and this additional test showed agreement with the cross validation estimates.

We compared the approach of using IF-THEN rules combining several local substructures and using *very simple rules* [43] consisting of only one local substructure. The simple rules resulted in almost no molecular functions obtaining statistically significant cross validation AUC values, indicating the necessity of combining local substructures to model the complex relationship between structure and function.

The methodology is implemented as a package in the ROSETTA system. The package uses many of the components from the implementation of the methodology from Papers I and II (see section 2.1.4), but includes a feature selection method instead of the template language.

Chapter 3 Discussion and Conclusions

This study includes four novel contributions to functional genomics in terms of bioinformatics methods and tools learning from annotated sequence, structure and expression data (see Figure 6). In this chapter we will discuss some of the main results and draw some parallels between these contributions, as well as providing some directions for future work.

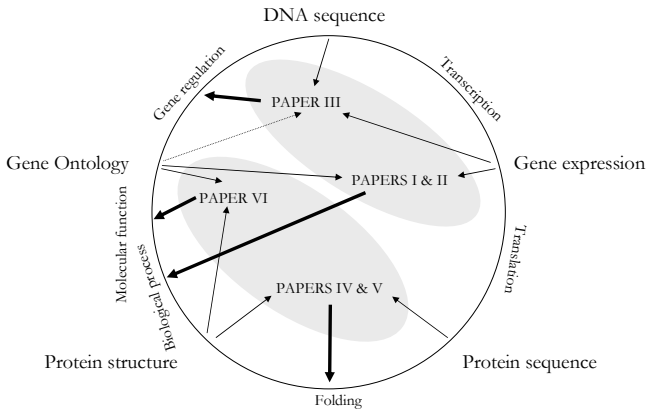


Figure 6. An overview of the six contributions (papers) in this study, indicating data sources and main biological objectives. The circle symbols the molecular biology from gene transcription, via translation and folding, to proteins participating in biological processes conducting particular molecular functions. The circle is closed by the fact that some proteins act as regulatory proteins controlling transcription. The main data and knowledge sources for functional genomics are indicated as DNA sequence, gene expression, protein sequence, protein structure and Gene Ontology annotations. The arrows show which of these data and knowledge sources the different papers use and which biological phenomena they are studying. Papers I & II and paper III are grouped since they are all contributions to the analysis of gene expression regulation. Papers IV & V and paper VI are grouped since they are all contributions to the study of protein structure and together cover the methodology needed to predict molecular function from sequence. Paper III only uses Gene Ontology for evaluation, and this is indicated with a dotted arrow.

The prediction of biological process from expression time profiles using supervised learning was inspired by the fact that related studies using clustering did not address core issues such as anti-co-regulation and several annotations per gene. Furthermore, broad expression clusters were not specific enough to address functionally co-regulated genes (as shown in paper II). The rule learning approach may be compared to clustering in that the rules specify overlapping clusters of relatively few genes. However, these clusters are specific to particular biological processes and are found using biological knowledge to guide the search. Moreover, using the template language we were able to address changes in regulation in limited time windows, resulting in a powerful and flexible approach to modeling biological process from expression. The template language also seems to handle the intrinsic noise in microarray expression data quite well. The generalizing capabilities of the rules could be objectively evaluated using cross validation and applied to provide new hypotheses on the biological processes of both previously characterized and uncharacterized genes. The well-defined framework for prediction and evaluation is a key property of supervised models, making it possible to learn from existing knowledge and apply it to new cases. However, in order for cross validation performance estimates to hold for these new, unseen cases, the training data must be representative. A possible problem of studying biological processes from expression is the fact that there are much more characterized genes among the up-regulated genes than among the down-regulated genes. This is particularly apparent in the Iyer *et al.* [48] data set studied in paper II (data not shown), and may result in optimistic performance estimates for the uncharacterized genes since these estimates were calculated from characterized genes using cross validation. We should also notice that the cross validation estimates are relative to the selected biological processes, and although these classes covered most genes, the estimates do not apply to genes participating in other process. This is also true for the classification approaches in papers IV, V and VI. However, paper V introduced P-values for each prediction and showed that these P-values quite confidently could identify wrong predictions (and furthermore, protein domains from other than the 135 predicted folds obtained in general high P-values, data not shown). Furthermore, paper VI showed how cross validation estimates could be used as a confidence measure for the prediction of uncharacterized proteins.

The assumption leading to the study of biological processes from gene expression data is that genes participating in the same processes are transcribed at the same time and that this is initiated by the same regulatory proteins. Since these regulatory proteins or transcription factors depend on recognizing particular sequence motifs (binding sites) to initialize transcription, including

sequence motif data may greatly increase the precision at which we can predict functionally co-regulated genes. In paper III we investigated the occurrence of combinations of binding sites (modules) in the promoter region of similarly expressed genes, and showed that hypotheses on functional co-regulation were much more reliable when requirements of both common sequence motifs and similar expression were applied. This clearly shows the difficulty of analyzing sequence data by itself. Sequence motifs common to several genes show the potential for co-regulation, but will produce too many false positives if used alone. However, expression data indicate the actual co-regulation, and by combining these two sources of data one may determine whether the potential (sequence data) agrees with the actual expression outcomes. Furthermore, by studying expression data under different stress conditions, one may find the specific regulatory mechanisms for that particular biological response.

Other studies have provided results showing the intuitive assumption that biological processes are more easily studied using expression data than molecular functions (i.e. tasks carried out by a single gene product) [18]. Paper III provided evidence for concluding that genes associated with the same binding sites, genes with similar expression and, in particular, genes with both the same binding sites and similar expression (i.e. discovered binding site modules) were more likely to be annotated to the same biological processes than to the same molecular functions or cellular components. However, all three parts of Gene Ontology showed statistically significant results for binding site modules. This may in part be due to strong dependencies between annotations to the different parts of Gene Ontology (see e.g. the Annotation Expander (ANNEX) tool at http://www.goat.no/, [78]). Also, preliminary results on applying the template language from papers I and II have shown better results than using clustering as in paper III, and may provide even better data to this end [115, 116]. Integrating the methodology from papers I and II and paper III is left for future research.

Paper III exemplified the advantages of combining several sources of data for studying biological processes, and in particular showed the advantage of combining data from the static sequence and the dynamic expression. Pavlidis *et al.* [85] used a support vector machine to predict 27 functional classes from expression data, and compared the result using *phylogenetic profiles* (i.e. the evolutionary history of genes) and a combinations of expression data and phylogenetic profiles. Stuart *et al.* [108] conducted a *comparative genomics* approach to predict gene function by finding genes that were co-expressed in several different organisms (humans, flies, worms and yeast). Johansson [50] has later used the method from papers I and II to predict biological processes from expression data [35] and combined this with other information sources

including phenotypic data, data on different sequence characteristics and secondary structure predictions [26].

Due to the fact that relatively few proteins have been structurally solved, genome-wide studies using structure data would often rely on first predicting structure from sequence. Papers IV and V described a structure prediction method based on the novel concept of local descriptors of protein structure. By extracting sequence signals from fragments of similar local substructure, we were able to correctly assign local substructures to protein sequences and furthermore from these substructures to infer SCOP fold. A further extension to this method would be to assemble complete, self-consistent structural models from the assigned substructures. The large part of correctly aligned substructures as shown in paper V, also for sequences with low sequence similarity to the training set, indicated that this approach is viable[14]. Paper VI showed that not only structure, but also function, may be predicted from local substructures. In principle, one could predict function from sequence by first assigning local substructures to a protein sequence and then predict function from the assigned local substructures. However, this is left for future research. Paper VI only addressed the problem of predicting function from structure in terms of representing protein structures with (structurally matched) local substructures.

Molecular functions describe the specific activities and roles of gene products involved in different biological processes occurring at different times. They should therefore be less related to similarities in expression time profiles than biological processes. On the other hand, structural features should be related to specific interacting partners and therefore be better for studying molecular functions than biological processes. Paper VI provided evidence for this in terms of much higher cross validation performance for molecular functions than for biological processes. It also indicated that the relationships between local substructures and molecular functions are of a complex nature involving combinations of structural features. As the library used in papers IV, V and VI contains popular local substructures of several (more than three) backbone fragments that are close in space, these structures are more likely to be situated in the protein core, where the proteins are densely packed, than at the surface (data in paper V also indicated that hydrophobic amino acids are more often significantly overrepresented in the library, while surface and hydrogen bond amino acids are more often significantly underrepresented). Hence, these local

[14] Research on assembling structure models from local descriptors assigned by the method in paper V is currently conducted by Michal Drabikowski in Krzysztof Fidelis´ group at Lawrence Livermore National Laboratory (LLNL).

substructures will most often not directly correspond to particular functional sites (binding sites) at the surface of the protein, but rather represent structural features indirectly related to these sites.

The different parts of this study have investigated, or provided data for investigating (seeing papers IV and V in the light of paper VI), the cellular roles of genes and gene products in terms of Gene Ontology annotations. The Gene Ontology has proven to describe many aspect of biological data including sequence data, expression data and structure data in humans (papers I and II), yeast (paper III) and other organisms (paper VI, PDB structures are taken from several different organisms). Although approaches to formalizing biological knowledge such as Gene Ontology have received criticism (e.g. Shrager [100] argues that function should only be assigned in a specific context and not as a fixed, general property, while Midelfart [71] argues that the Gene Ontology graphs are not well defined in terms of semantics), they are of vital importance as they allow genome-wide computational approaches to functional genomics. Also, Gene Ontology is an ongoing research project, constantly evolving and improving with better understanding of molecular biology.

Both Gene Ontology and SCOP are examples of human-processed knowledge. Methods that are able to incorporate and use such knowledge are often referred to as *knowledge-based* methods. Knowledge-based methods such as machine learning aim at representing the knowledge with general concepts. These concepts are general in the sense that they explain, in terms of the data, the knowledge possessed about several or all examples (e.g. "genes that are up-regulated under certain conditions (data) are involved in a specific biological process (knowledge)"). Note that the knowledge constituting the training examples are normally obtained using considerably more information that what is available to the machine learning algorithm. Hence, the goal is not to learn the skills of the biological expert (as the name "supervised learning" might suggest), but to learn how to obtain this knowledge using only the limited data available (i.e. the features used to represent the examples). For example, crystallographers use crystals of the protein to infer structure, while the goal of structure prediction is to infer structure using only sequence data.

In this study we have discussed several different methods for representing the relationship between observations and knowledge in terms of general concepts (IF-THEN rules, artificial neural networks, hyperplanes, etc.). However, maybe even more important than choosing the right learning formalism is choosing the right features to represent the observations. One should remember than not only are the observations in the training set mere examples taken from a larger, possibly infinite, number of potential observations, the examples themselves

are projections of the real world into one-dimensional feature vectors. One may even discuss whether relatively well defined features such as genes and proteins are a natural choice, considering for example splicing variants and protein complexes. We have seen several examples in this study of different problem-specific solutions to choosing or constructing features and feature values. Papers I and II used discrete changes in mRNA levels over time windows rather than the original measurements of the relative mRNA level at specific time points. Paper III used the occurrence of sequence motifs in the promoter region of genes. Papers IV, V and VI used the containment of local substructures to represent protein structure. Consequently, poorly performing models may not only be due to the learning algorithm or even the truth in the assumption of a relationship between data and knowledge. It may also be due to the way we choose to represent the world (i.e. which features we use). Secondary structure prediction from protein sequence into helices, sheets and coils are for example predicted with 70% to 80% accuracy. However, it has been suggested that this may be close to the upper performance-limit for secondary structure prediction given that these methods only consider relatively short sequence windows for learning (i.e. secondary structure may be a result of neighboring sequence fragments that are close in space, but far away along the sequence [110]).

As discussed earlier, given that the induced model is able to generalize to unseen observations (i.e. is able to correctly predict unseen observations), it must also have been successful in describing some general concepts important for the relationship between the data and the knowledge. However, if the model itself is difficult to interpret, it will remain a vehicle for prediction only. It is therefore meaningful to discern between the predictive and descriptive quality of a model. Most methods represent the model as a complex function of the features (e.g. a separating hyperplane in feature space or an artificial neural network). However, since features often are selected by humans and are interpretable problem-specific entities (e.g. genes, proteins or sequence motifs), loosing sight of the features may imply loosing sight of the only basis for interpretation. Methods that represent models in terms of legible combinations of these features may therefore have a greater chance of also being descriptive. The rule learning approach used in several papers in this study induces readable models in terms of IF-THEN rules such that the relationship between data and knowledge may be interpreted directly. Unfortunately, the number of rules in such models often limits their interpretability. And although the number of rules may be reduced dramatically using rule filtering methods [123], there is a tendency that a large number of redundant rules are more robust and perform better for prediction purposes (data not shown). Paper III, however, is one

example were rules were filtered and interpreted directly as potential regulatory binding site modules.

As both the complexity of the models and the performance estimates indicate, predictions in this study have to be considered hypotheses and cannot be accepted as new knowledge without further biological experiments. Hence, we must admit to be far away from the goal of doing automatic functional genomics in terms of learning from existing knowledge and predicting the remaining cases using the induced models. However, we are in a situation were we may use predictions to guide biological experiments in terms of limiting the number of possible hypotheses biologists need to validate. Also, as the number of examples in terms of annotations or solved protein structures grows, these methods will provide better hypotheses. And with time, simple hypotheses put forward by machine learning methods may lead to better understanding and ultimately general theories that will be trusted without verification by time-consuming biological experiments.

Chapter 4 Summary in Swedish

Förutsägelse av geners och proteiners funktion från sekvens-, struktur- och uttrycksdata.

Funktionell genomik kan beskrivas som uppgiften att bestämma gen- och proteinfunktion för hela genom. För att utföra detta krävs dataanalys av stora mängder biologiska data från DNA- och proteinsekvenser, proteinstrukturer och genuttryck. Kraftfulla metoder erhålls från maskininlärning där generella modeller byggs upp från existerande kunskap om gener eller proteiner. Denna kunskap används sedan för att skapa hypoteser om okända geners och proteiners funktion.

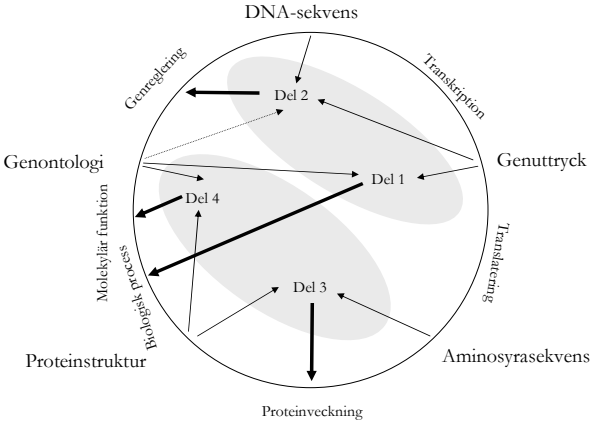
Denna avhandling består av fyra delar som bidrar till ny kunskap inom ämnet funktionell genomik där analys av olika slags biologiska data ger nya insikter om biologisk funktion. Genontologi är en kontrollerad vokabulär för att beskriva geners och proteiners cellulära roll och spelar därför genomgående en viktig roll.

I den första delen används data från tidsprofiler från genuttrycksstudier. Från detta data byggs modeller som kan förutsäga hur gener deltar i biologiska processer. Modellen består av regler som har formen ”om A så B”. Dessa regler associerar biologiska processer med diskreta förändringar i genuttrycksnivå över en begränsad tidsrymd. Modellerna används för att skapa hypoteser för hur såväl karakteriserade som okarakteriserade gener deltar i biologiska processer.

Den andra delen undersöker den kombinatoriska regleringen av genuttryck genom att inducera ”om A så B” regler som associerar minsta möjliga kombinationer av sekvensmotiv för gener med liknande uttrycksprofil. Sådana kombinationer visade sig vara signifikant korrelerade med funktion. Detta gör det möjligt att formulera hypoteser för den bakomliggande mekanismen för reglering av genuttryck av flera biologiska processer.

I den tredje delen beskrivs en ny metod där lokala deskriptorer av proteinstrukturer används för att undersöka hur mönster i aminosyrasekvens påverkar den lokala proteinstrukturen. Dessa deskriptorer används även för att från sekvensdata förutsäga topologisk klass (veckning) av proteindomäner. I den fjärde och sista delen används lokala deskriptorer för att inducera

regelmodeller av formen ”om A så B” som förutsäger molekylär funktion från struktur.



Bilden ovan ger en översikt av de fyra delarna i avhandlingen. Cirkeln symboliserar flödet av molekylärbiologiska processer som sker i cellen. Gentranskription följs av translation och veckning av proteiner som slutligen deltar i biologiska processer som styr molekylära funktioner. Cirkeln är sluten på grund av det faktum att vissa proteiner fungerar som reglerande proteiner av bland annat gentranskription. Data och kunskapskällor för funktionell genomik är DNA-sekvens, genuttryck, proteinsekvens, proteinstruktur och Genontologin. Pilarna visar hur data och kunskapskällor har används för de olika delarna i avhandlingen och vilka biologiska fenomen som har studerats. Den första och andra delen utgör en grupp eftersom de behandlar analys av reglering av genuttryck. Den tredje och fjärde delen utgör en andra grupp eftersom de bidrar till ny kunskap om proteiners funktion och metoder för att förutsäga molekylär funktion från sekvens.

References

1. Altman, R. B. and Raychaudhuri, S. Whole-genome expression analysis: challenges beyond clustering. *Curr Opin Struct Biol.* 11(3): 340-7, 2001.
2. Altschul, S. F., Madden, T. L., Schaffer, A. A., Zhang, J., Zhang, Z., Miller, W. and Lipman, D. J. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.* 25(17): 3389-402, 1997.
3. Apweiler, R., Bairoch, A., Wu, C. H., Barker, W. C., Boeckmann, B., Ferro, S., Gasteiger, E., Huang, H., Lopez, R., Magrane, M.*, et al.* UniProt: the Universal Protein knowledgebase. *Nucleic Acids Res.* 32 Database issue: D115-9, 2004.
4. Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T.*, et al.* Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet.* 25(1): 25-9, 2000.
5. Bairoch, A. and Apweiler, R. The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. *Nucleic Acids Res.* 28(1): 45-8, 2000.
6. Beer, M. A. and Tavazoie, S. Predicting gene expression from sequence. *Cell.* 117(2): 185-98, 2004.
7. Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J. and Wheeler, D. L. GenBank: update. *Nucleic Acids Res.* 32 Database issue: D23-6, 2004.
8. Berman, B. P., Nibu, Y., Pfeiffer, B. D., Tomancak, P., Celniker, S. E., Levine, M., Rubin, G. M. and Eisen, M. B. Exploiting transcription factor binding site clustering to identify cis-regulatory modules involved in pattern formation in the Drosophila genome. *Proc Natl Acad Sci U S A.* 99(2): 757-62, 2002.
9. Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N. and Bourne, P. E. The Protein Data Bank. *Nucleic Acids Res.* 28(1): 235-42, 2000.
10. Bernal, A., Ear, U. and Kyrpides, N. Genomes OnLine Database (GOLD): a monitor of genome projects world-wide. *Nucleic Acids Res.* 29(1): 126-7, 2001.
11. Boguski, M. S., Lowe, T. M. and Tolstoshev, C. M. dbEST--database for "expressed sequence tags". *Nat Genet.* 4(4): 332-3, 1993.
12. Boutselakis, H., Dimitropoulos, D., Fillon, J., Golovin, A., Henrick, K., Hussain, A., Ionides, J., John, M., Keller, P. A., Krissinel, E.*, et al.* E-MSD: the European Bioinformatics Institute Macromolecular Structure Database. *Nucleic Acids Res.* 31(1): 458-62, 2003.
13. Brattbakk, H.-R., Lægreid, A., Komorowski, J., Langaas, M., Huwiler, A., Huseby, S., Hvidsten, T. R. and Johansen, B. Ceramide Induces

Cell Proliferation, Cell Cycle Control, Lipid Metabolism in Mesangial Cells. *In preparation*, 2004.

14. Brazma, A., Hingamp, P., Quackenbush, J., Sherlock, G., Spellman, P., Stoeckert, C., Aach, J., Ansorge, W., Ball, C. A., Causton, H. C.*, et al.* Minimum information about a microarray experiment (MIAME)-toward standards for microarray data. *Nat Genet.* 29(4): 365-71, 2001.
15. Brazma, A., Parkinson, H., Sarkans, U., Shojatalab, M., Vilo, J., Abeygunawardena, N., Holloway, E., Kapushesky, M., Kemmeren, P., Lara, G. G.*, et al.* ArrayExpress--a public repository for microarray gene expression data at the EBI. *Nucleic Acids Res.* 31(1): 68-71, 2003.
16. Breiman, L. Bagging predictors. *Machine learning.* 24: 123-140, 1996.
17. Brown, F. M. *Boolean reasoning : the logic of Boolean equations.* Kluwer Academic Publishers, Boston, 1990.
18. Brown, M. P., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C. W., Furey, T. S., Ares, M., Jr. and Haussler, D. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc Natl Acad Sci U S A.* 97(1): 262-7, 2000.
19. Brown, P. O. and Botstein, D. Exploring the new world of the genome with DNA microarrays. *Nat Genet.* 21(1 Suppl): 33-7, 1999.
20. Bystroff, C. and Baker, D. Prediction of local structure in proteins using a library of sequence-structure motifs. *J Mol Biol.* 281(3): 565-77, 1998.
21. Chandonia, J. M., Hon, G., Walker, N. S., Lo Conte, L., Koehl, P., Levitt, M. and Brenner, S. E. The ASTRAL Compendium in 2004. *Nucleic Acids Res.* 32 Database issue: D189-92, 2004.
22. Cho, R. J., Campbell, M. J., Winzeler, E. A., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T. G., Gabrielian, A. E., Landsman, D., Lockhart, D. J.*, et al.* A genome-wide transcriptional analysis of the mitotic cell cycle. *Mol Cell.* 2(1): 65-73, 1998.
23. Cho, R. J., Huang, M., Campbell, M. J., Dong, H., Steinmetz, L., Sapinoso, L., Hampton, G., Elledge, S. J., Davis, R. W. and Lockhart, D. J. Transcriptional regulation and function during the human cell cycle. *Nat Genet.* 27(1): 48-54, 2001.
24. Chu, S., DeRisi, J., Eisen, M., Mulholland, J., Botstein, D., Brown, P. O. and Herskowitz, I. The transcriptional program of sporulation in budding yeast. *Science.* 282(5389): 699-705, 1998.
25. Churchill, G. A. Fundamentals of experimental design for cDNA microarrays. *Nat Genet.* 32 Suppl: 490-5, 2002.
26. Clare, A. and King, R. D. Predicting gene function in Saccharomyces cerevisiae. *Bioinformatics.* 19 Suppl 2: II42-II49, 2003.
27. DeRisi, J. L., Iyer, V. R. and Brown, P. O. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science.* 278(5338): 680-6, 1997.

28. Duggan, D. J., Bittner, M., Chen, Y., Meltzer, P. and Trent, J. M. Expression profiling using cDNA microarrays. *Nat Genet.* 21(1 Suppl): 10-4, 1999.
29. Edgar, R., Domrachev, M. and Lash, A. E. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Res.* 30(1): 207-10, 2002.
30. Efron, B. and Tibshirani, R. J. *An introduction to the Bootstrap.* Chapman & Hall, London, 1993.
31. Eisen, M. B., Spellman, P. T., Brown, P. O. and Botstein, D. Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci U S A.* 95(25): 14863-8, 1998.
32. Fleischmann, R. D., Adams, M. D., White, O., Clayton, R. A., Kirkness, E. F., Kerlavage, A. R., Bult, C. J., Tomb, J. F., Dougherty, B. A., Merrick, J. M*., et al.* Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. *Science.* 269(5223): 496-512, 1995.
33. Fodor, S. P., Read, J. L., Pirrung, M. C., Stryer, L., Lu, A. T. and Solas, D. Light-directed, spatially addressable parallel chemical synthesis. *Science.* 251(4995): 767-73, 1991.
34. Gasch, A. P. and Eisen, M. B. Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering. *Genome Biol.* 3(11): 0059.1-0059.22, 2002.
35. Gasch, A. P., Spellman, P. T., Kao, C. M., Carmel-Harel, O., Eisen, M. B., Storz, G., Botstein, D. and Brown, P. O. Genomic expression programs in the response of yeast cells to environmental changes. *Mol Biol Cell.* 11(12): 4241-57, 2000.
36. Gilbert, D. G. euGenes: a eukaryote genome information system. *Nucleic Acids Res.* 30(1): 145-8, 2002.
37. Goldsmith-Fischman, S. and Honig, B. Structural genomics: computational methods for structure analysis. *Protein Sci.* 12(9): 1813-21, 2003.
38. Guex, N. and Peitsch, M. C. SWISS-MODEL and the Swiss-PdbViewer: an environment for comparative protein modeling. *Electrophoresis.* 18(15): 2714-23, 1997.
39. Hanley, J. A. and McNeil, B. J. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology.* 143: 29-36, 1982.
40. Hastie, T., Tibshirani, R. J. and Friedman, J. *The Elements of Statistical Learning.* Springer, New York, 2001.
41. Hegyi, H. and Gerstein, M. The relationship between protein structure and function: a comprehensive survey with application to the yeast genome. *J Mol Biol.* 288(1): 147-64, 1999.
42. Hieter, P. and Boguski, M. Functional genomics: it's all how you read it. *Science.* 278(5338): 601-2, 1997.
43. Holte, R. C. Very simple classification rules perform well on most commonly used datasets. *Machine learning.* 11(1): 63-91, 1993.

44. Hughes, J. D., Estep, P. W., Tavazoie, S. and Church, G. M. Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae. *J Mol Biol*. 296(5): 1205-14, 2000.
45. Hulo, N., Sigrist, C. J., Le Saux, V., Langendijk-Genevaux, P. S., Bordoli, L., Gattiker, A., De Castro, E., Bucher, P. and Bairoch, A. Recent improvements to the PROSITE database. *Nucleic Acids Res*. 32 Database issue: D134-7, 2004.
46. Huseby, S., Lægreid, A., Komorowski, J., Langaas, M., Brattbakk, H.-R., Hvidsten, T. R. and Johansen, B. Exploring Platelet-Activating factor gene expression patterns in human keratinocytes. *In preparation*, 2004.
47. Hvidsten, T. R., Komorowski, J., Sandvik, A. K. and Lægreid, A. Predicting gene function from gene expressions and ontologies. In *Pacific Symposium on Biocomputing* edited by Altman, R. B., Dunker, A. K., Hunter, L., Lauderdale, K. and Klein, T. E., pp. 299-310. World Scientific, 2001.
48. Iyer, V. R., Eisen, M. B., Ross, D. T., Schuler, G., Moore, T., Lee, J. C., Trent, J. M., Staudt, L. M., Hudson, J., Jr., Boguski, M. S*., et al.* The transcriptional program in the response of human fibroblasts to serum. *Science*. 283(5398): 83-7, 1999.
49. Jelinsky, S. A., Estep, P., Church, G. M. and Samson, L. D. Regulatory networks revealed by transcriptional profiling of damaged Saccharomyces cerevisiae cells: Rpn4 links base excision repair with proteasomes. *Mol Cell Biol*. 20(21): 8157-67, 2000.
50. Johansson, A. *A Rough Set approach to functional classfication of genes*. Department of physics and measurement technology, Linköping University, 2004.
51. Johnson, D. S. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences*. 9: 256-278, 1974.
52. Johnson, R. A. and Wichern, D. W. *Applied multivariate statistical analysis*. Prentice Hall, Upper Saddle River, N.J., 2002.
53. Jonassen, I., Eidhammer, I. and Taylor, W. R. Discovery of local packing motifs in protein structures. *Proteins*. 34(2): 206-19, 1999.
54. Jones, D. T. Protein secondary structure prediction based on position-specific scoring matrices. *J Mol Biol*. 292(2): 195-202, 1999.
55. Kanehisa, M. and Bork, P. Bioinformatics in the post-sequence era. *Nat Genet*. 33 Suppl: 305-10, 2003.
56. Karchin, R., Cline, M., Mandel-Gutfreund, Y. and Karplus, K. Hidden Markov models that use predicted local structure for fold recognition: alphabets of backbone geometry. *Proteins*. 51(4): 504-14, 2003.
57. Karplus, K., Karchin, R., Draper, J., Casper, J., Mandel-Gutfreund, Y., Diekhans, M. and Hughey, R. Combining local-structure, fold-recognition, and new fold methods for protein structure prediction. *Proteins*. 53 Suppl 6: 491-6, 2003.

58. Kasuya, A. and Thornton, J. M. Three-dimensional structure analysis of PROSITE patterns. *J Mol Biol.* 286(5): 1673-91, 1999.
59. Kinch, L. N., Wrabl, J. O., Krishna, S. S., Majumdar, I., Sadreyev, R. I., Qi, Y., Pei, J., Cheng, H. and Grishin, N. V. CASP5 assessment of fold recognition target predictions. *Proteins.* 53 Suppl 6: 395-409, 2003.
60. Komorowski, J., Hvidsten, T. R., Jenssen, T. K., Tjeldvoll, D., Hovig, E., Lægreid, A. and Sandvik, A. K. [New knowledge derived from measurement of gene expression with the DNA microarray method]. *Tidsskr Nor Laegeforen.* 121(10): 1229-32, 2001.
61. Komorowski, J., Hvidsten, T. R., Jenssen, T.-K., Tjeldvoll, D., Hovig, E., Sandvik, A. K. and Lægreid, A. Towards Knowledge Discovery from cDNA Microarray Gene Expression Data. In *Principles of Data Mining and Knowledge Discovery* edited by Zighed, D. A., Komorowski, J. and Zytkow, J., pp. 470-475, Lecture Notes in Artifical Intelligence 1910. Springer, 2000.
62. Komorowski, J., Pawlak, Z., Polkowski, L. and Skowron, A. Rough sets: A tutorial. In *Rough Fuzzy Hybridization: A New Trend in Decicion-Making* edited by Pal, S. K. and Skowron, A., p. 3-98. Springer, 1999.
63. Komorowski, J., Øhrn, A. and Skowron, A. The ROSETTA Rough Set Software System. In *Handbook of Data Mining and Knowledge Discovery* edited by Klösgen, W. and Zytkow, J., p. 554-559. Oxford University Press, 2002.
64. Kryshtafovych, A. and Fidelis, K. Local descriptors of protein structure. Part I. General approach and classification of local 3D regions in proteins. *In preparation*, 2004.
65. Kryshtafovych, A., Hvidsten, T. R., Komorowski, J. and Fidelis, K. Fold Recognition Using Sequence Fingerprints of Protein Local Substructures. In *IEEE Computer Society Bioinformatics Conference*, pp. 517-518. IEEE Computer Society, 2003.
66. Kulikova, T., Aldebert, P., Althorpe, N., Baker, W., Bates, K., Browne, P., van den Broek, A., Cochrane, G., Duggan, K., Eberhardt, R.*, et al.* The EMBL Nucleotide Sequence Database. *Nucleic Acids Res.* 32 Database issue: D27-30, 2004.
67. Lander, E. S., Linton, L. M., Birren, B., Nusbaum, C., Zody, M. C., Baldwin, J., Devon, K., Dewar, K., Doyle, M., FitzHugh, W.*, et al.* Initial sequencing and analysis of the human genome. *Nature.* 409(6822): 860-921, 2001.
68. Lee, T. I., Rinaldi, N. J., Robert, F., Odom, D. T., Bar-Joseph, Z., Gerber, G. K., Hannett, N. M., Harbison, C. T., Thompson, C. M., Simon, I.*, et al.* Transcriptional regulatory networks in Saccharomyces cerevisiae. *Science.* 298(5594): 799-804, 2002.
69. Leibowitz, N., Fligelman, Z. Y., Nussinov, R. and Wolfson, H. J. Automated multiple structure alignment and detection of a common substructural motif. *Proteins.* 43(3): 235-45, 2001.
70. Mewes, H. W., Frishman, D., Guldener, U., Mannhaupt, G., Mayer, K., Mokrejs, M., Morgenstern, B., Munsterkotter, M., Rudd, S. and Weil, B.

MIPS: a database for genomes and protein sequences. *Nucleic Acids Res.* 30(1): 31-4, 2002.

71. Midelfart, H. *Knowledge Discovery from cDNA Microarrays and a priori Knowledge.* Norwegian University of Science and Technology, Trondheim, 2003.
72. Midelfart, H. and Komorowski, J. A rough set approach to learning in a directed acyclic graph. In *Proceedings of the 3rd International Conference on Rough Sets and Current Trends in Computing* edited by Peters, J. F., Skowron, A. and Zhon, N., pp. 144-155, Lecture Notes in Artificial Intelligence 2475. Springer, 2002.
73. Midelfart, H., Lægreid, A. and Komorowski, J. Classification of gene expression data in an ontology. In *Proceedings of the 2nd International Symposium on Medical Data Analysis* edited by Crespo, J., Maojo, V. and Martin, F., pp. 186-194, Lecture Notes in Computer Science 2199. Springer, 2001.
74. Mitchell, T. M. *Machine Learning.* McGraw-Hill, New York, 1997.
75. Moult, J. and Melamud, E. From fold to function. *Curr Opin Struct Biol.* 10(3): 384-9, 2000.
76. Murzin, A. G. Structure classification-based assessment of CASP3 predictions for the fold recognition targets. *Proteins.* Suppl 3: 88-103, 1999.
77. Murzin, A. G., Brenner, S. E., Hubbard, T. and Chothia, C. SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J Mol Biol.* 247(4): 536-40, 1995.
78. Myre, S. *Development of a method for annotation in Gene Ontology and its use on gene products from microarray studies of BON- and AR42J-cells.* Norwegian University of Science and Technology, Trondheim, 2003.
79. Nakayama, S. and Willett, P. A sphere-based descriptor for matching protein structures. *J Mol Model (Online).* 8(6): 199-207, 2002.
80. Orengo, C. A., Michie, A. D., Jones, S., Jones, D. T., Swindells, M. B. and Thornton, J. M. CATH--a hierarchic classification of protein domain structures. *Structure.* 5(8): 1093-108, 1997.
81. Orengo, C. A., Todd, A. E. and Thornton, J. M. From protein structure to function. *Curr Opin Struct Biol.* 9(3): 374-82, 1999.
82. Pawlak, Z. Rough Sets. *International Journal of Information and Computer Science.* 11(5): 341-356, 1982.
83. Pawlak, Z. Rough sets: theoretical aspects of reasoning about data. In *Theory and decision library. Series D, System theory, knowledge engineering, and problem solving,* pp. 229. Kluwer Academic Publishers, 1991.
84. Pavlidis, P., Tang, C. and Noble, W. S. Classification of genes using probabilistic models of microarray expression profiles. In *Workshop on Data Mining in Bioinformatics, 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 2001.

85. Pavlidis, P., Weston, J., Cai, J. and Noble, W. S. Learning gene functional classifications from multiple data types. *J Comput Biol.* 9(2): 401-11, 2002.
86. Pilpel, Y., Sudarsanam, P. and Church, G. M. Identifying regulatory networks by combinatorial analysis of promoter elements. *Nat Genet.* 29(2): 153-9, 2001.
87. Quackenbush, J. Computational analysis of microarray data. *Nat Rev Genet.* 2(6): 418-27, 2001.
88. Quackenbush, J. Microarray data normalization and transformation. *Nat Genet.* 32 Suppl: 496-501, 2002.
89. Roberts, C. J., Nelson, B., Marton, M. J., Stoughton, R., Meyer, M. R., Bennett, H. A., He, Y. D., Dai, H., Walker, W. L., Hughes, T. R., *et al.* Signaling and circuitry of multiple MAPK pathways revealed by a matrix of global gene expression profiles. *Science.* 287(5454): 873-80, 2000.
90. Russell, R. B. Detection of protein three-dimensional side-chain patterns: new examples of convergent evolution. *J Mol Biol.* 279(5): 1211-27, 1998.
91. Russell, S. and Norvig, P. *Artificial Intelligence.* Prentice-Hall, New Jersey, 1995.
92. Schapire, R. E. The strength of weak learnability. *Machine learning.* 5: 197-227, 1990.
93. Schena, M., Shalon, D., Davis, R. W. and Brown, P. O. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science.* 270(5235): 467-70, 1995.
94. Segal, E., Shapira, M., Regev, A., Pe'er, D., Botstein, D., Koller, D. and Friedman, N. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat Genet.* 34(2): 166-76, 2003.
95. Segal, E., Yelensky, R. and Koller, D. Genome-wide discovery of transcriptional modules from DNA sequence and gene expression. *Bioinformatics.* 19 Suppl 1: I273-I282, 2003.
96. Shakhnovich, B. E., Dokholyan, N. V., DeLisi, C. and Shakhnovich, E. I. Functional fingerprints of folds: evidence for correlated structure-function evolution. *J Mol Biol.* 326(1): 1-9, 2003.
97. Shatkay, H., Edwards, S., Wilbur, W. J. and Boguski, M. Genes, themes and microarrays: using information retrieval for large-scale gene analysis. *Proc Int Conf Intell Syst Mol Biol.* 8: 317-28, 2000.
98. Shatkay, H. and Feldman, R. Mining the biomedical literature in the genomic era: an overview. *J Comput Biol.* 10(6): 821-55, 2003.
99. Sherlock, G. Analysis of large-scale gene expression data. *Curr Opin Immunol.* 12(2): 201-5, 2000.
100. Shrager, J. The fiction of function. *Bioinformatics.* 19(15): 1934-6, 2003.
101. Simons, K. T., Kooperberg, C., Huang, E. and Baker, D. Assembly of protein tertiary structures from fragments with similar local sequences

using simulated annealing and Bayesian scoring functions. *J Mol Biol.* 268(1): 209-25, 1997.

102. Simons, K. T., Ruczinski, I., Kooperberg, C., Fox, B. A., Bystroff, C. and Baker, D. Improved recognition of native-like protein structures using a combination of sequence-dependent and sequence-independent features of proteins. *Proteins.* 34(1): 82-95, 1999.
103. Sippl, M. J., Lackner, P., Domingues, F. S., Prlic, A., Malik, R., Andreeva, A. and Wiederstein, M. Assessment of the CASP4 fold recognition category. *Proteins.* Suppl 5: 55-67, 2001.
104. Sjolander, K., Karplus, K., Brown, M., Hughey, R., Krogh, A., Mian, I. S. and Haussler, D. Dirichlet mixtures: a method for improved detection of weak but significant protein sequence homology. *Comput Appl Biosci.* 12(4): 327-45, 1996.
105. Skowron, A. and Nguyen, H. S. Boolean reasoning scheme with some applications in data mining. In *Third European Symposium on Principles and Practice of Knowledge Discovery in Databases* edited by Zytkow, J. M. and Rauch, J., pp. 107-115, Lecture Notes in Artificial Intelligence. Springer-Verlag, 1999.
106. Skowron, A. and Rauszer, C. The discernibility matrices and functions in information systems. In *Intelligent Decision Support: Handbook of Applications and Advances in Rough Sets Theory* edited by Slowinski, R., p. 331-362. Kluwer Academic Publishers, 1992.
107. Smyth, M. S. and Martin, J. H. x ray crystallography. *Mol Pathol.* 53(1): 8-14, 2000.
108. Stuart, J. M., Segal, E., Koller, D. and Kim, S. K. A gene-coexpression network for global discovery of conserved genetic modules. *Science.* 302(5643): 249-55, 2003.
109. Tateno, Y., Imanishi, T., Miyazaki, S., Fukami-Kobayashi, K., Saitou, N., Sugawara, H. and Gojobori, T. DNA Data Bank of Japan (DDBJ) for genome scale research in life science. *Nucleic Acids Res.* 30(1): 27-30, 2002.
110. Taylor, W. R. and Thornton, J. M. Recognition of super-secondary structure in proteins. *J Mol Biol.* 173(4): 487-512, 1984.
111. Theodoridis, S. and Koutroumbas, K. *Pattern recognition.* Academic Press, Amsterdam ; Boston, 2003.
112. Tramontano, A. and Morea, V. Assessment of homology-based predictions in CASP5. *Proteins.* 53 Suppl 6: 352-68, 2003.
113. Venclovas, C., Zemla, A., Fidelis, K. and Moult, J. Assessment of progress over the CASP experiments. *Proteins.* 53 Suppl 6: 585-95, 2003.
114. Wheeler, D. L., Church, D. M., Edgar, R., Federhen, S., Helmberg, W., Madden, T. L., Pontius, J. U., Schuler, G. D., Schriml, L. M., Sequeira, E.*, et al.* Database resources of the National Center for Biotechnology Information: update. *Nucleic Acids Res.* 32 Database issue: D35-40, 2004.

115. Wilczynski, B., Hvidsten, T. R., Kryshtafovych, A., Komorowski, J., Tiuryn, J. and Fidelis, K. Using Local Gene Expression Similarities to Discover Regulatory Binding Site Modules. *In preparation*, 2004.
116. Wilczynski, B., Hvidsten, T. R., Kryshtafovych, A., Stubbs, L., Komorowski, J. and Fidelis, K. A rule-based framework for gene regulation pathways discovery. In *IEEE Computer Society Bioinformatics Conference*, pp. 435-436. IEEE Computer Society, 2003.
117. Vilo, J., Brazma, A., Jonassen, I., Robinson, A. and Ukkonen, E. Mining for putative regulatory elements in the yeast genome using gene expression data. *Proc Int Conf Intell Syst Mol Biol.* 8: 384-94, 2000.
118. Vinterbo, S. and Øhrn, A. Minimal approximate hitting sets and rule templates. *International Journal of Approximate Reasoning.* 25: 123-143, 2000.
119. Wu, L. F., Hughes, T. R., Davierwala, A. P., Robinson, M. D., Stoughton, R. and Altschuler, S. J. Large-scale prediction of Saccharomyces cerevisiae gene function using overlapping transcriptional clusters. *Nat Genet.* 31(3): 255-65, 2002.
120. Wuthrich, K. Determination of three-dimensional protein structures in solution by nuclear magnetic resonance: an overview. *Methods Enzymol.* 177: 125-31, 1989.
121. Yu, K. Theoretical determination of amino acid substitution groups based on qualitative physicochemical properties. *http://cmgm.stanford.edu/biochem218/Projects%202001/Yu.pdf*, 2001.
122. Zhang, C. and Kim, S. H. Overview of structural genomics: from structure to function. *Curr Opin Chem Biol.* 7(1): 28-32, 2003.
123. Ågotnes, T., Komorowski, J. and Løken, T. Taming large rule models in rough set approaches. In *Principles of Data Mining and Knowledge Discovery* edited by Zytkow, J. and Rauch, J., pp. 193-203, Lecture Notes in Artificial Intelligence 1704. Springer, 1999.
124. Øhrn, A. *Discernibility and Rough Sets in Medicine: Tools and Applications.* Norwegian University of Science and Technology, Trondheim, 1999.

Acta Universitatis Upsaliensis

Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science and Technology

Editor: The Dean of the Faculty of Science and Technology

A doctoral dissertation from the Faculty of Science and Technology, Uppsala University, is usually a summary of a number of papers. A few copies of the complete dissertation are kept at major Swedish research libraries, while the summary alone is distributed internationally through the series *Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science and Technology*. (Prior to October, 1993, the series was published under the title "Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science".)

Distribution:
Uppsala University Library
Box 510, SE-751 20 Uppsala, Sweden
www.uu.se, acta@ub.uu.se

ISSN 1104-232X
ISBN 91-554-6014-3