Kjetil Klepper

Integrated approaches for motif discovery in genomic regions

Thesis for the degree of Philosophiae Doctor

Trondheim, May 2013

Norwegian University of Science and Technology
Faculty of Medicine
Department of Cancer Research and Molecular Medicine

NTNU – Trondheim
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the degree of Philosophiae Doctor

Faculty of Medicine
Department of Cancer Research and Molecular Medicine

ISBN 978-82-471-4390-2 (printed ver.)
ISBN 978-82-471-4391-9 (electronic ver.)
ISSN 1503-8181

Doctoral theses at NTNU, 2013:143

Printed by NTNU-trykk

NORGES TEKNISK-NATURVITENSKAPELIGE UNIVERSITET
DET MEDISINSKE FAKULTET

Integrerte metoder for motivoppdaging i genomiske regioner

Proteiner utgjør en stor gruppe makromolekyler som er essensielle for alt liv. De fungerer blant annet som byggeklosser i celler eller som enzymer som katalyserer kjemiske reaksjoner. Nye proteiner lages kontinuerlig i levende celler, og oppskriftene på alle de forskjellige proteinene som en organisme trenger er beskrevet i *gener* som er kodet i organismens DNA. Prosessen med å lage et nytt protein består av to trinn. I det første trinnet (*transkripsjon*) lages en kopi av gensekvensen i RNA, og denne kopien blir så benyttet i det neste trinnet (*translasjon*) som en mal for å sette sammen kjeden av aminosyrer som danner det ferdige proteinet. Ettersom forskjellige typer proteiner brukes i ulike sammenhenger og i ulike celle-typer, må prosessen med å lage nye proteiner være nøye regulert. Det første trinnet er hovedsakelig regulert av *transkripsjonsfaktorer* som gjenkjenner og binder seg til bestemte sekvensmønstre i DNA, gjerne lokalisert i områder som ligger foran genene. Transkripsjonsfaktorer hjelper til med å rekruttere transkripsjonsmaskineriet til starten av genet og starte transkripsjonsprosessen. Men noen faktorer spiller en motsatt rolle og kan nedregulere gener ved å forhindre at transkripsjon finner sted. Ulike typer av transkripsjons-faktorer regulerer forskjellige gener, og for å få et fullstendig bilde av hvordan genene reguleres er det viktig å finne ut av hva slags sekvensmønster hver transkripsjons-faktor gjenkjenner (bindingsmotivet) og hvor de binder i genomet (bindingssetene).

Å avdekke bindingsmotiver og bindingsseter ved hjelp av eksperimentelle metoder kan være både tidkrevende og dyrt, og det har derfor vært stor interesse for å utvikle dataprogrammer som kan predikere dette automatisk ut fra DNA-sekvenser. Hundrevis av dataprogrammer har blitt utviklet til dette formålet, og de kan grovt deles inn i to klasser: *motivskanning*-verktøy benytter seg av modeller av allerede kjente bindingsmotiver for å finne nye potensielle bindingsseter i sekvensene. Såkalte *de novo* motivoppdagingsmetoder er derimot i stand til å finne nye bindingsmotiver og bindingsseter ved å søke etter overrepresenterte sekvensmønstre i sett av DNA-sekvenser som er antatt å inneholde bindingsseter for samme transkripsjonsfaktor.

Uavhengige evalueringer av slike motivoppdagingsverktøy (inkludert en evaluering vi selv har foretatt og beskrevet i den første artikkelen i avhandlingen) har dessverre vist at disse metodene ikke alltid fungerer så bra som man kunne håpe på. En viktig grunn til dette er at de fleste metodene bare baserer seg på informasjonen som ligger i selve DNA-sekvensen, men det er mange andre forhold som også kan påvirke om en transkripsjonsfaktor faktisk er i stand til å binde til DNAet og utføre sin regulatoriske oppgave.

Nye metoder som har kommet de siste årene har vist at det er mulig å oppnå bedre resultater ved å ta hensyn til andre typer av informasjon i tillegg til DNA-sekvensen, som for eksempel informasjon om hvilke områder av genomet som er konservert sammenlignet med beslektede organismer, hvilke områder som har en åpen kromatinstruktur i forskjellige celle-typer og hvilke transkripsjonsfaktorer som er kjent å samarbeide med hverandre. Vi har derfor utviklet et nytt motivoppdagings-verktøy (beskrevet i de to siste artiklene) som kan brukes til å integrere mange ulike typer informasjon i denne prosessen på en generell og fleksibel måte.

Kandidat: Kjetil Klepper
Institutt: Institutt for kreftforskning og molekylærmedisin
Veileder: Finn Drabløs
Finansieringskilder: Forskningsrådets program for funksjonell genomforskning (FUGE) og ELIXIR.no

Ovennevnte avhandling er funnet verdig til å forsvares offentlig for graden PhD i medisinsk teknologi.
Disputas finner sted i Auditoriet, Medisinsk Teknisk Forskningssenter, tirsdag 21. mai 2013, kl. 12:15.

Abstract

Recipes for all the proteins that are needed by an organism are described in its genes which are encoded in DNA. In order to create a new protein, a copy of the DNA recipe must first be transcribed into RNA and this transcript is subsequently translated into protein. Since different proteins are used in different cell-types and at different times, the process of creating new proteins must be tightly regulated. The first step in this process is mainly regulated by *transcription factors* which bind to specific sequence patterns in the DNA and help recruit the transcriptional apparatus to the start of the gene and initiate transcription. An important step in elucidating the gene regulatory networks of an organism is thus to determine which sequence pattern each transcription factor binds to (the *binding motif*) and also the sites where they bind.

Although such motifs and binding sites are best determined experimentally, computational tools for motif discovery seem to offer a convenient, fast and cost-effective alternative to experimental methods. Hundreds of software programs have therefore been developed for this purpose. These tools can broadly be divided into two classes: *motif scanning* tools rely on predefined models of binding motifs and search sequences for matches to these motifs in order to identify potential binding sites. *De novo* motif discovery methods, on the other hand, aim to find new motifs and binding sites without such prior knowledge by looking for overrepresented patterns in sequences believed to be regulated by common factors.

However, independent assessment studies of computational motif discovery tools have shown that the performance of these methods is limited, especially with respect to predicting functionally active binding sites in real genomic sequences. One reason for this is that most of these tools only base their predictions on information in the DNA sequence itself, but many other aspects besides the presence of a binding motif can influence whether a transcription factor will actually be able to bind and exert its regulatory function, including for instance the local chromatin conformation around the binding site or the presence of cooperative factors binding nearby.

More recent approaches have demonstrated that binding site predictions can be improved by also considering additional information related to e.g. phylogenetic conservation, nucleosome occupancy, DNase hypersensitive sites, epigenetic features, gene expression and transcription factor interactions. To this end we have developed a new software workbench which is able to integrate additional information from a variety of sources into the motif discovery process in a coherent and flexible way.

Preface

This is truly an exciting time to be a bioinformatician!

When I first started my PhD (admittedly far too many years ago), I could barely envision the possibilities that would lie ahead. Back then there wasn't really much annotation data available besides the genome sequences themselves, and when I came up with the idea for the PriorsEditor tool (which would eventually develop into MotifLab), I could only hope that enough extra data would become available in the future so that the tool would actually prove useful. To my great excitement, more and more data tracks were published as time progressed, and by the time I was finally finished writing the software, it could already be used for numerous applications.

There are many people that deserve to be acknowledged as my PhD now comes to an end. I want to thank my colleagues at NTNU, and especially the co-authors of my first paper, for scientific collaboration and social company. I also want to thank all the people that came to me for assistance with bioinformatics analyses during the years I worked at the national bioinformatics help desk. Their needs identified shortcomings in my software which helped to improve it considerably.

Most of all I want to thank my supervisor Finn Drabløs. Not the least for providing me the opportunity to do a PhD in the first place, but more importantly for not giving up on me when the first few years proved somewhat fruitless, but rather entrusting me with more time so that I was eventually able to complete a PhD project that I can feel proud of.

Kjetil Klepper

The work for this PhD project was carried out at the Department of Cancer Research and Molecular Medicine at NTNU under the supervision of Professor Finn Drabløs. The project was funded by The National Programme for Research in Functional Genomics in Norway (FUGE) and the Norwegian infrastructure for bioinformatics ELIXIR.no, both in The Research Council of Norway (NFR).

List of papers

Paper I) Klepper K, Sandve GK, Abul O, Johansen J and Drabløs F: **Assessment of composite motif discovery methods**. *BMC Bioinformatics* 2008, **9**:123

Paper II) Klepper K and Drabløs F: **PriorsEditor: a tool for the creation and use of positional priors in motif discovery**. *Bioinformatics* 2010, **26**(17):2195-7

Paper III) Klepper K and Drabløs F: **MotifLab: a tools and data integration workbench for motif discovery and regulatory sequence analysis**. *BMC Bioinformatics* 2013, **14**:9

Abbreviations

5meC	5-methylcytosine
ASP	average site performance
ATP	adenosine triphosphate
bp	base pair
CAGE	cap analysis of gene expression
CC	correlation coefficient
CDS	coding DNA sequence
ChIP	chromatin immunoprecipitation
CRM	*cis*-regulatory module
DNA	deoxyribonucleic acid
DNase HS	DNase hypersensitive site
DNMT	DNA methyltransferase
EM	expectation maximization
EMSA	electrophoretic mobility shift assay
FN	false negative
FP	false positive
GO	gene ontology
GTP	guanosine triphosphate
HAT	histone acetyltransferace
HDAC	histone deacetylase
HMM	hidden Markov model
HMT	histone methyltransferase
IC	information content
IUPAC	International Union of Pure and Applied Chemistry
mRNA	messenger RNA
miRNA	micro RNA
PC	performance coefficient
PCM	position count matrix

PCR	polymerase chain reaction
PET	paired-end tags
PFM	position frequency matrix
PIC	transcription pre-initiation complex
Pol II	(RNA) polymerase II
PPV	positive predictive value
PWM	position weight matrix
RNA	ribonucleic acid
SELEX	systematic evolution of ligands by exponential enrichment
Sn	sensitivity
SNP	single nucleotide polymorphism
Sp	specificity
TF	transcription factor
TFBS	transcription factor binding site
TN	true negative
TP	true positive
TSS	transcription start site
UTR	untranslated region

Contents

Genes and gene regulation

Proteins and genes

Proteins are a diverse class of macromolecules that are essential for all life. Some proteins have a structural function and are used as building blocks in the body, for instance *keratin* which is the key component in skin, hair and nails. Others are involved in communication and cell signalling, such as the hormone *insulin*. A very important group of proteins are those that act as *enzymes* to catalyse chemical reactions. Hardly any process occurs in a cell without enzymes being involved in some way. Enzymes break down molecules and construct new ones, they move around chemical groups from one molecule to another to change their functional properties, and they extract energy from nutrients. Each enzyme has its own specific function that it executes in a controlled way. Complex organisms have tens of thousands of different types of proteins, and new proteins are synthesized continuously in all living cells.

Although proteins vary significantly in size, shape and complexity, they are all built up in the same basic way. A protein is essentially just a string of **amino acids** that are linked together in a long chain, and this chain folds up into a functional three-dimensional structure. There are 20 standard types of proteinogenic amino acids, and the order in which they are incorporated into the chain decides the final shape and functional properties of the protein. A protein can consist of anywhere between a few dozen to several thousand amino acids. The synthesis of new proteins are, of course, also carried out by enzymes, but in order to produce all the different kinds of proteins correctly, the organism must have access to some sort of descriptions on how to construct them. The recipes for all the proteins that are needed by an organism can be found in its **genes** which are encoded in **DNA**.

DNA (deoxyribonucleic acid) is another macromolecule which is fundamental to all living beings. Similar to proteins, DNA consists of a long chain of smaller molecules connected together, but in DNA these molecules are **nucleotides** rather than amino acids. A single nucleotide is made up of a **nucleobase** (or simply *base*) connected to a pentose sugar (in DNA this sugar is *deoxyribose*) which in turn is connected to a phosphate group at the 5' carbon atom. Four different types of nucleobases are found in DNA: *adenine*, *cytosine*, *guanine* and *thymine*, abbreviated A, C, G and T respectively (Figure 1). The phosphate groups enable linking between the nucleotides by connecting to the 3' carbon atom in one sugar molecule and the 5' carbon in the next. This ordered linking arrangement gives the DNA chain a sense of direction. The chain of repeated sugar-phosphate groups is called the “backbone” and is similar throughout the

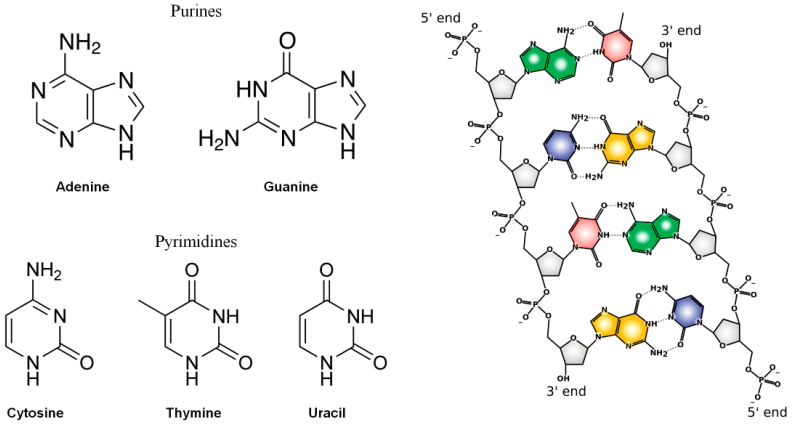


Figure 1: Left: Diagrams of the four nucleobases found in DNA plus uracil which is found in RNA. **Right:** Adenines (in green) on one strand form base pairs with thymines (red) on the opposite strand and cytosines (blue) pair up with guanines (yellow). The sugar-phosphate backbones of the two strands are shown in grey. (Images adapted from Wikipedia).

molecule, so the actual information carried by the DNA is encoded in the specific arrangement of the bases connected to this backbone. By chaining nucleotides together, a single DNA molecule can grow to contain several hundred millions of nucleotides. Since DNA is always synthesized by attaching new nucleotides to the 3' carbon of the previous nucleotide, it is customary to label the 3'-end as the "tail end" of the DNA molecule and read DNA sequences from the 5'-end towards the 3'-end, e.g. 5'-CATCGTCTGTTCAGA-3'. The 5'-end is also often called the *upstream* end and the 3'-end is called the *downstream* end.

Unlike proteins, the DNA chain is not normally folded up into a complex three-dimensional structure determined by its nucleotide sequence. Rather, a single chain of DNA, called a *strand*, is linked to another strand of DNA to form *double-stranded DNA*. In this structure, every base from one strand is paired up with a base on the other strand in a way reminiscent of a zipper. Adenines are always paired with thymines and cytosines are paired with guanines, and so these pairs are called *complementary base pairs*. Because of this complementarity, the nucleotide sequence of one strand can always be deduced from the sequence on the other strand. The backbones of the two strands run in opposite directions of each other, and one strand is called the *direct strand* while the opposite strand is called the *reverse strand*. In addition, these two strands twist around a central axis with a full revolution once every ~10.4 base pairs, so the final shape of the DNA molecule is that of a double helix (Figure 2)[1].

Figure 2: The two complementary DNA strands twist around a central axis to form a double helix. The backbones of the two strands are closer together on one side of the helix than the other, and this gives rise to two grooves of different sizes where the *major groove* is about twice as wide as the *minor groove*.

The connections between cytosine and guanine bases are made up of three hydrogen bonds, and these connections are somewhat stronger than the connections between adenines and thymines which only consist of two hydrogen bonds. Even so, since hydrogen bonds are among the weakest chemical bonds, the two strands of a DNA molecule can easily be separated and rejoined by enzymes that need to access the genetic information.

The way that DNA molecules encode recipes for proteins is called the **genetic code**. According to this code, each of the 20 different amino acids is represented by a nucleotide triplet called a **codon**. A sequence of such triplet codons in the DNA can thus encode the sequence of amino acids that make up a protein. For example, the codon "ATG" corresponds to the amino acid *methionine* and the triplet "AAG" corresponds to *lysine*, so the DNA sequence "ATGAAGAAG" would encode for an amino acid chain consisting of one methionine followed by two lysines. Since there are four different nucleotides there are $4^3 = 64$ possible triplet codons, which is much more than what is required to cover all amino acids. The genetic code therefore allows for some redundancy, and most amino acids can be represented by several different codons. *Cysteine*, for instance, can be encoded by both "TGT" and "TGC", and other amino acids have three, four or even six codons. The three codons "TAA", "TAG" and "TGA" do not correspond to any amino acids, and these so-called "stop codons" are used to signal the end of the protein recipe.

The portion of a DNA molecule which encodes a protein is called a **gene** (although one gene can potentially encode several related proteins and some genes encode information used to create other things than proteins). Since DNA molecules can be rather long, they can contain several genes at different locations, and genes can reside on both the direct and reverse strand. In many organisms, the genetic information is split across several DNA molecules, each of which is called a **chromosome**, and the total genetic information in all chromosomes taken together makes up an organism's **genome**. The *human genome*, for example, consists of 23 pairs of chromosomes, with one copy of each chromosome inherited from each parent. The *haploid* human genome (counting only one copy of each chromosome) spans more than 3 billion base pairs and contains an estimated number of 20,000–25,000 genes [2]. Only a very small percentage of the human genome consists of protein-coding genes, however. The rest – which is called *non-coding DNA* – serve other purposes and can, for instance, be involved in gene regulation or be important for chromosome structure and integrity.

The process of creating a protein based on a gene is called "expressing the gene" and it involves two steps. In the first step, called **transcription**, a copy of the gene is made in **RNA**. RNA (ribonucleic acid) is similar to DNA except that it contains a different type of sugar in its backbone (*ribose* rather than *deoxyribose*), it uses a nucleotide called *uracil* instead of *thymine*, and the transcribed copy is single-stranded. The RNA molecule is used as a messenger to pass the protein recipe over to complex molecular machines called **ribosomes** which use the RNA as a template to synthesize proteins. This second step is known as **translation**.

In eukaryotes, the newly transcribed *messenger RNA* (mRNA) molecule is subjected to additional processing before it is transported out of the cell nucleus to the ribosomes which reside in the cytosol. First, the 5' end of the mRNA strand is capped with a GTP nucleotide (a guanosine with three phosphate groups) which is connected at its 5' carbon rather than the usual 3' connection. At the other end of the mRNA strand, a long chain of adenines, called a *poly-A tail*, is attached. These modifications protect the mRNA molecule from being degraded by other enzymes and they are also important for regulating the export of the mRNA out of the nucleus and for proper translation of the mRNA by the ribosomes. In addition, eukaryotic genes may contain segments that are not part of the protein recipes. These segments, known as **introns**, are cut out of the mRNA molecule, and the remaining parts, called **exons**, are spliced together to form the *mature messenger RNA*. Sometimes, one or more of these exons can also be cut from the mRNA in a process known as "alternative splicing". As a result, the same gene can give rise to several different protein recipes. A gene can also have multiple transcription start sites (TSS). These alternative TSSs can be located relatively close to each other and just result in a slight variation at the start of the transcript or they could be located far apart and potentially initiate transcription at different exons [3].

When a single cell divides, it copies its entire DNA so that the two resulting daughter cells each receive an identical copy of the organism's entire genome. Consequently, all cells in an organism contain the same genetic information, and the difference between cells of different tissues (muscle cells, skin cells, brain cells, blood cells etc.) lies not in which genes they contain but in which genes the cells express. For example, all cells in the human body have the gene for *insulin*, but this gene is only expressed in a special type of cells in the pancreas. Other genes are only expressed at certain times during the life of an organism, for instance at the blastula stage of early embryonic development [4]. Even single-celled organisms need to adapt their gene expression levels in response to changes in their environment. Because proper gene expression is so crucial for the development and functioning of all organisms, it needs to be tightly regulated. Gene regulation can occur at both the transcriptional and translational stage; however, in this thesis I will focus on transcriptional regulation.

Gene transcription

Gene transcription is performed by a complex containing an enzyme called *RNA polymerase*. Prokaryotes have only one type of RNA polymerase whereas eukaryotes have several which are used for different kinds of genes. The polymerase used to transcribe protein-coding genes is called Polymerase II (or just Pol II or RNAP II). The transcription process itself can be divided into three main stages: *initiation*, *elongation* and *termination*.

In the initiation stage, the RNA polymerase binds to the DNA molecule immediately upstream of the transcription start site of the gene. Once bound, the RNA polymerase unwinds a short stretch of the DNA (about 10–20 base pairs) and separates the two strands to create a small "transcription bubble". The RNA polymerase then proceeds by synthesizing a short strand of RNA nucleotides which pairs with one of the DNA strands. The DNA strand that is complementary to the RNA strand is called the *template strand* (or *antisense strand*) whereas the unpaired DNA strand that has the same base sequence as the resulting RNA strand is called the *coding strand* (or *sense strand*).

After the initiation stage, transcription can either be prematurely aborted (resulting in just a tiny RNA fragment) or proceed to the elongation stage. In this second stage, the RNA polymerase travels along the template strand in a 3' to 5' direction, unwinding and separating the DNA strands as it passes. New nucleotides that are complementary to the template strand are incorporated into the growing RNA strand at its 3' end. Only a short segment at this end of the RNA strand is paired with the template strand at any time, and as the RNA polymerase moves downstream along the gene, the RNA strand peels off in the other direction and the DNA behind the polymerase is rewound to form a double helix again.

When the full gene has been transcribed, the RNA polymerase encounters a termination signal in the DNA sequence which causes the newly synthesized RNA strand to be released and the RNA polymerase to disassociate from the DNA molecule, thus ending transcription.

The placement and orientation of the RNA polymerase on the DNA molecule determines where transcription starts and in which direction it proceeds, so it is important that the enzyme is placed correctly during the initiation stage. The minimal DNA region that is required to bind the RNA polymerase is called the **core promoter**. In prokaryotes, the core promoter commonly consists of two sequence elements located at approximately 10 bp and 35 bp upstream of the transcription start site [5, 6]. The -10 element is also called the "Pribnow box" and has the sequence motif "TATAAT" whereas the -35 element has the motif "TTGACA" (on the coding strand). These sequence elements are recognized and bound by a *sigma factor* which forms a complex with the RNA polymerase and facilitates proper positioning. In eukaryotes the situation is much more complicated, and several different proteins are required to guide the polymerase to the promoter and perform other functions necessary for transcription initiation. These proteins, which for Pol II include TFIIA, TFIIB, TFIID, TFIIE, TFIIF, and TFIIH, are called **general transcription factors** and they combine with the polymerase to form the **transcription preinitiation complex (PIC)**. The architecture of the core promoter is also much more diverse in eukaryotes compared to prokaryotes. The first eukaryotic core promoter element that was identified was the "TATA-box" which is usually located about 25 to 30 bp upstream of the TSS. The TATA-box has the consensus sequence "TATAAA" and is recognized by the *TATA-binding protein* (TBP) which is a subunit of TFIID. Only a small percentage of eukaryotic genes have TATA-boxes in their core promoters, however. Other common core promoter elements include the *Initiator element* (Inr) with sequence motif "YYANWYY[1]", the *downstream promoter element* (DPE) with motif "RGWYV", the *TFIIB recognition element* (BRE) with motif "SSRCGCC" and the *motif ten element* (MTE) with motif "CSARCSSAACGS" [7-9]. These elements can usually be found at specific locations relative to the TSS, and the promoter of any given gene may contain all, some or none of these motifs. The composition of the core promoter and sequence variation within promoter elements can influence the affinity of the polymerase complex towards the promoter and hence the rate by which the polymerase binds and initiates transcription. Even so, the basal transcription complex by itself is normally only capable of driving low levels of transcription.

[1] The sequence motifs are notated with IUPAC symbols for degenerate bases as described in Table 1 on page 32.

Transcription factors

In both prokaryotes and eukaryotes, additional DNA-binding proteins called **regulatory** or **sequence-specific transcription factors** (TFs) are required to increase the rate of transcription beyond basal levels or to repress transcription altogether. Every gene is usually under regulation by several such transcription factors which can be active at different times or in different cell types. For example, some TFs are activated by external signals or changes in the environment (such as the presence of a specific hormone or nutrient or a change in body temperature) and allows cells to express genes that are needed in response to such conditions [10, 11]. A single type of transcription factor can potentially also regulate many different genes, which implies that there is a many-to-many relationship between genes and transcription factors.

Transcription factors can act as either *activators* or *repressors*. Activators increase the probability of transcription by assisting in the recruitment of the preinitiation complex to the core promoter or by helping the RNA polymerase escape the promoter and proceed with elongation [12, 13]. Repressors, on the other hand, can hinder transcription by blocking the binding of the preinitiation complex or interfering with other activators that are required for transcription [14]. Some estimates claim that more than 8–10% of the genes in the human genome could potentially encode for transcription factors [15, 16], which would imply that at least a few thousand different TFs are involved in gene regulation in higher organisms. Since transcription factors are themselves proteins which are encoded by genes, they are also responsible for regulating the expression of other TFs and sometimes even their own expression. Hence, transcription factors are fundamental components of complex regulatory networks that allow for precise positive and negative control over the spatial and temporal expression of genes.

Large proteins are typically composed of several independent structural *domains* which serve distinct functions. Transcription factors commonly have a *DNA-binding domain* (also called "*cis*-acting domain") and an *activation/repression domain* (or "*trans*-acting domain). Many TFs also have a *dimerization domain* which allows it to form homo- or heterodimer complexes with other factors. Such dimerization might be necessary for some TFs in order to obtain a functional structure. The activation domain allows the TF to interact with other factors and to influence the basal transcription complex, often via binding to additional *co-activators* (or *co-repressors*) such as the *mediator* complex [17]. The DNA-binding domain, on the other hand, allows the TF to bind to particular locations on the DNA molecule. Most transcription factors recognize specific DNA sequence patterns that they bind to, and such patterns are called "binding motifs". A binding motif is usually rather short, about 6–12 bp [18]. For example, the transcription factor c-Myc binds to the motif "CACGTG" (which is a rather common binding motif also known as the "E-box") and the factor NF-AT binds to "GGAAA".

Most TFs recognize their motifs in a DNA sequence by inserting their binding domain into the major groove of the double helix. Here, each distinct sequence pattern of base pairs display a unique signature of methyl groups, hydrogen atoms and hydrogen donors and acceptors which can connect with the right transcription factor through electrostatic and Van der Waals forces [19]. Not all bases within the binding motif need to be in contact with the TF, however, and for some positions the interaction between the bases and the TF might be weaker than others. Hence, some sequence variation is often allowed in the binding motif as long as the overall interaction is strong enough to support binding, and transcription factors can normally bind to several slightly different sequence patterns, albeit with varying affinity [20].

Although there are several thousand different TFs, most of them rely on a small number of common mechanisms in order to bind to the DNA. This allows transcription factors to be grouped into families and subclasses thereof based on similarities in their DNA-binding domains [16, 21].

Some of the major transcription factor families are (Figure 3):

- **Helix-turn-helix (HTH)**

  The helix-turn-helix binding domain consists of two alpha-helices at approximately right angles to each other joined by a short strand of amino acids. One of the helices (the “recognition helix”) is inserted into the major groove of the DNA where it makes contact with the nucleotide sequence while the other helix helps to position the recognition helix and stabilize the interaction. Notable subclasses of this family include the *homeodomain*, which in addition to the regular HTH contains a third helix, and the *winged-HTH* which also consists of three helices but have additional beta-sheets as well (“wings”).

- **Basic leucine zippers (b-ZIP)**

  B-ZIP factors are long alpha-helices where the C-terminals contain an amphipathic *leucine zipper* dimerization domain consisting of heptad repeats of amino acids with a leucine residue at every seventh position. The structure of the alpha-helix is such that these hydrophobic leucines all end up on the same side whereas the other side of the helix contains hydrophilic residues. The hydrophobic leucine region allows two B-ZIP factors to “zip together” as homo- or hetero-dimers in the shape of a Y. The loose N-terminal ends contain basic regions which bind to the DNA molecule in the major groove. Homodimers bind to a motif which consists of two half-sites arranged as a palindrome, whereas heterodimers can bind any combination of half-sites.

- **Basic helix-loop-helix (bHLH)**

  The basic helix-loop-helix domain is structurally similar to the basic leucine zipper, except that each monomer consists of not one, but two alpha helices which are connected by an unstructured loop region. The loop provides more flexibility with respect to DNA binding. The C-terminal alpha helix contains a zipper-like dimerization region which allows for formation of homo- or heterodimers when binding to other bHLH-factors, and the N-terminal alpha helix contains a basic region which can interact with the DNA when inserted into the major groove. Transcription factors in the bHLH-family typically bind to the previously mentioned E-box motif, although there are exceptions.

- **Zinc fingers**

  The family of transcription factors labelled as zinc fingers is the most diverse of those mentioned here, but a common feature is that their binding domain is made up of a short stretch of amino acids which is folded into a compact structure stabilized by coordination with a zinc ion (Zn^{2+}). The most common subclass is the Cys_2His_2 zinc finger which consists of a two-stranded antiparallel beta sheet and an alpha helix. A zinc ion makes contact with the side chains of two cysteine residues in the beta sheet and two histidine residues in the helix. As per usual, the alpha helix facilitates motif recognition by binding to the major groove in the DNA. Each zinc finger typically binds to a 3 bp sequence motif, and transcription factors can contain multiple tandem zinc fingers which interact with successive 3 bp groups in the DNA sequence in order to increase the length and specificity of the binding motif and provide stronger interaction with the DNA. Another important subclass is called Cys_4 because the zinc ion makes contact with four cysteine residues. Whereas Cys_2His_2 transcription factors usually contain three or more zinc fingers and bind as monomers, Cys_4 factors generally only contain two fingers but bind as homo- or heterodimers. The binding motif of homodimeric Cys_4 factors are made up of two inverted repeats.

The majority of TFs bind to the DNA by inserting one or more alpha helices into the major groove, but there are also factors that make contact in other ways, for instance via beta-strand or loop regions instead of helices or by interacting with the minor groove or sugar-phosphate backbone [22, 23].

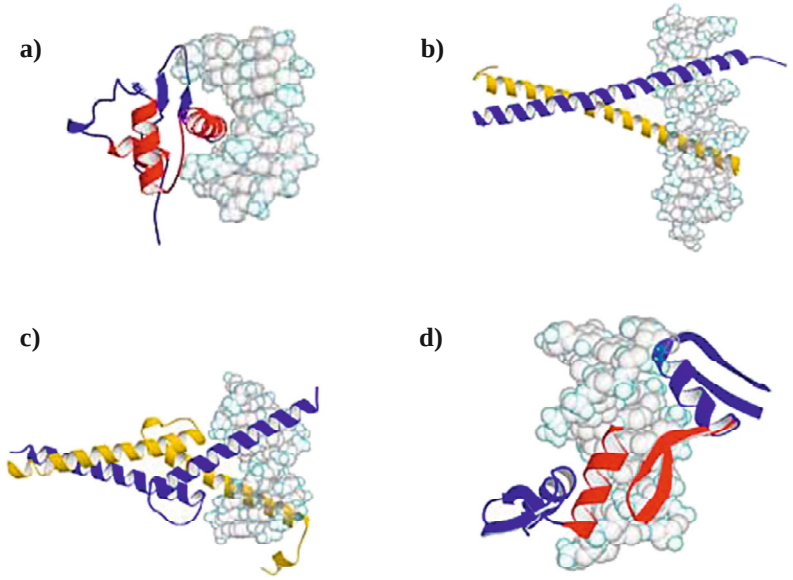


Figure 3: Illustration of transcription factors from four major families binding to DNA. **a)** Helix-turn-helix. **b)** Basic leucine zipper. **c)** Basic helix-loop-helix. **d)** Zinc finger.

Images from: Luscombe NM, Austin SE, Berman HM and Thornton JM (2000) "An overview of the structures of protein-DNA complexes", *Genome Biology* 1(1). Published by BioMed Central. Used with permission.

The occurrence of a binding motif in a DNA sequence where a TF binds is called a *binding site* (TFBS) or sometimes also a *response element*. Since many of the TFs that regulate a gene need to come in direct or indirect contact with the transcription complex, their binding sites are often located in the vicinity of the gene's core promoter, for instance in the region immediately upstream of the core promoter – called the *proximal promoter* – or in the 5'UTR downstream of the transcription start site (Figure 4). However, binding sites can also reside within *enhancer* and *silencer* regions (binding activators and repressors respectively) that might be located thousands of bases away from the TSS of the genes they regulate in either direction, and even within other genes or on different chromosomes [24]. The TFs binding to such distal regulatory elements can be brought in contact with the transcription complex at the promoter through "DNA looping" [25]. Because the DNA double helix is flexible and can be bent back on itself, regions that are seemingly far apart in the DNA sequence can actually be located close to each other in three-dimensional space (Figure 5). Genes that are located next to each other in the genome need not have the same expression profile, however, so it is crucial that enhancers only interact with their target genes and not with other genes

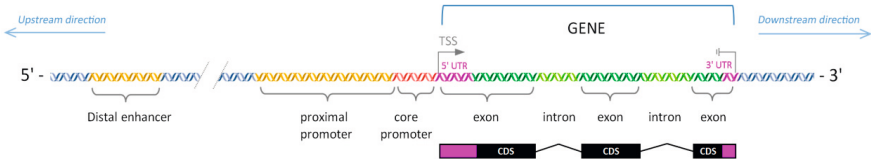


Figure 4: Illustration of a segment of DNA containing a gene and its upstream regulatory elements. This gene has two introns that do not contain protein coding information, and these will be spliced out of the transcript to form the mature mRNA which only consists of the three exon regions illustrated with boxes. However, the actual *coding DNA sequence* (CDS) shown in black starts in the middle of the first exon, so the beginning of the transcript contains a 5' *untranslated region* (5'UTR) shown in purple. Also, the stop codon which marks the end of the protein recipe is located in the middle of the last exon, so the transcript contains another untranslated region at the 3' end (3'UTR).

that might be located nearby. To prevent enhancer regions from influencing the wrong genes, regulatory elements known as *insulators* can block the interaction between an enhancer and a promoter when it occurs somewhere between them. In vertebrates this blocking is mediated primarily by the CTCF-factor which binds to the insulator element, whereas invertebrates have several factors that can act at insulators [26-28].

Transcription factors do not usually operate alone but work in combination with other TFs and co-factors in order to achieve the required regulatory control. This combinatorial approach increases the number of possible configurations and allows a comparatively small number of TFs to control a large number of genes under various circumstances. Groups of binding motifs for co-operating TFs that appear together in the DNA sequence are called “composite motifs” or *cis*-regulatory modules (CRM). Two different and complementary models have been proposed to explain how such modules might function. For CRMs adhering to the “enhanceosome model”, the relative position and orientation of all the binding sites are strictly defined, so that when the target factors bind to their respective sites, they will be arranged in such a way that they can easily form a higher-order protein complex. All the target factors thus have to bind simultaneously for this complex to form, and if just a single factor is missing, the resulting complex might not be functional. The composition of “billboard model” CRMs is more flexible, and only a subset of the binding sites in the module might be occupied at any given time. The regulatory apparatus will read the “messages” conveyed by the bound transcription factors and interpret this information in a context-dependent manner [29, 30].

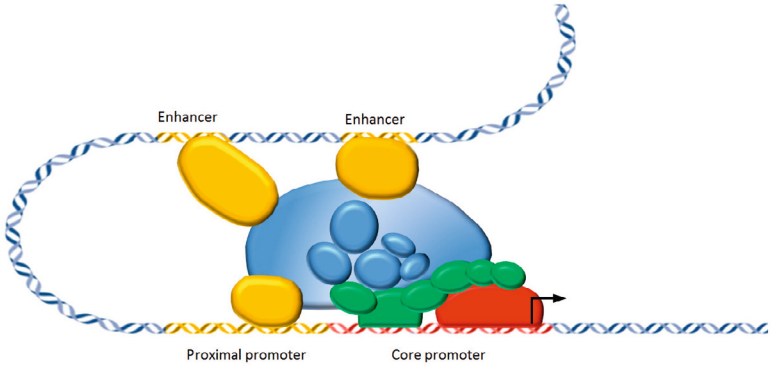


Figure 5: Illustration of the transcription preinitiation complex assembling at the promoter of a gene. The RNA polymerase is shown in red, general transcription factors in green, sequence-specific transcription factors binding to the proximal promoter and distal enhancers in yellow and co-factors in blue.

CRMs with similar configurations can appear within regulatory regions of different genes and thereby coordinate regulation of genes that should be expressed together [31, 32]. A single gene can also be regulated by many CRMs which function in different contexts. A classic example of this is the even-skipped (*eve*) gene in *Drosophila melanogaster* which is expressed in seven distinct stripes along the anterior-posterior axis of the early embryo. The spatial expression of this gene is controlled by five CRMs, each one responsible for driving expression in one or two stripes [33].

Regulatory regions such as promoters and enhancers are responsible for integrating signals conveyed by all the bound transcription factors and translate this information into appropriate regulatory responses. Complex behaviour will naturally require more regulatory DNA, and it has been shown that genes which must react to more signals tend to have longer promoter regions than genes which respond to fewer signals [34]. Probably for the same reason, genomes of complex multicellular organisms seem to overall contain much more regulatory DNA (and also a higher proportion of genes encoding for regulatory proteins) than genomes of simpler organisms, even though the number of genes itself might not be substantially greater [35].

Chromatin organisation and epigenetic regulation

The total length of all the chromosomes within a single cell in eukaryotic organisms can amount to several meters, but all this DNA has to fit inside the cell nucleus which is just a few micrometres wide. An efficient and dynamic packaging strategy is thus needed in order to condense the DNA but still allow the transcriptional machinery easy access to genes that need to be expressed.

The packing of the genome is organised in a hierarchical fashion with each level attaining a more dense structure of the DNA. This is accomplished through the use of proteins which function as a sort of scaffold that the DNA attaches to. The DNA together with all the bound proteins (those involved in packing but also transcription factors and others) are collectively referred to as **chromatin**. Two forms of chromatin have traditionally been recognized on the basis of staining patterns visible under a microscope. The loosely packed *euchromatin* comprises regions of the genome that are actively used by the cell, whereas regions that are not needed are packed into much denser *heterochromatin* [36]. More recent research has suggested that there could be as many as five principal types of chromatin, and these may again be further divided into additional subtypes [37].

At the most fundamental level of chromatin organisation, ~147 bp long stretches of the DNA double helix are wrapped 1.67 turns around globular octamer proteins consisting of the core histone proteins H2A, H2B, H3 and H4 (two of each) into complexes called **nucleosomes**. This arrangement is repeated along the DNA molecule like “beads on a string” with adjacent nucleosomes separated by 10–80 bp of unbound *linker DNA*. This conformation results in about 5- to 10-fold compaction of the DNA (Figure 6). Further compaction (about 50-fold) is facilitated by a fifth histone protein, linker histone H1, which binds to the outside of the nucleosome where the DNA enters and exits, and this is involved in linking together nucleosomes into a denser 30 nm wide fibre. Higher organisational levels can condense the DNA around thousand-fold during interphase and as much as ten-thousand-fold during mitosis [38].

Some of the five histone protein types mentioned above exist in different variants that can alter the physical properties of the nucleosomes, and these are utilized at specific times or at particular places in the genome. For example, the histone H2A has a variant named H2A.Z which makes the nucleosome core bind the DNA more tightly than usual, and this variant is frequently incorporated into nucleosomes located near the 5'-end of both active and inactive genes [39, 40]. On the other hand, variant H3.3 of histone H3 is incorporated throughout transcribed genes and also within regulatory sequences [41].

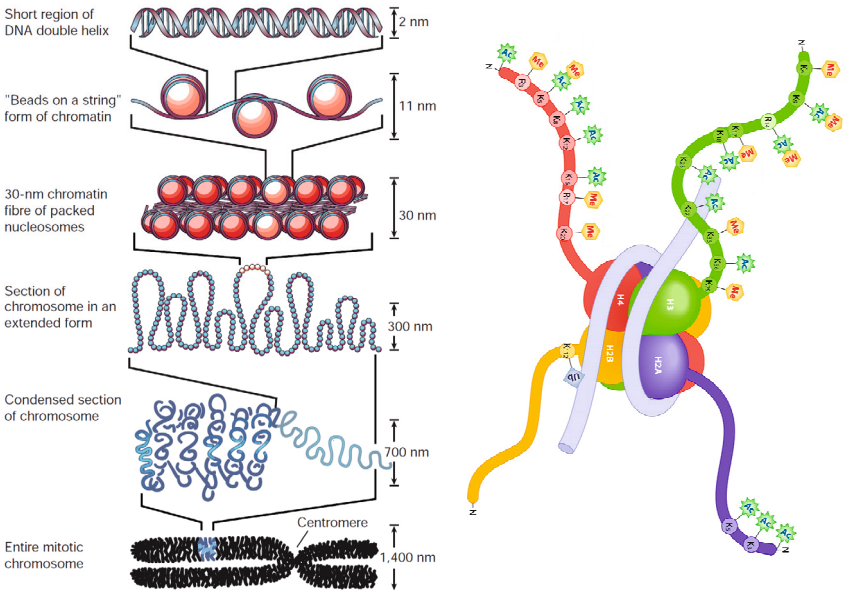


Figure 6: Left: The DNA double helix is wrapped around globular histone octamers to form nucleosomes which facilitate further condensation of the chromatin fiber. **Right:** Close-up view of a nucleosome with the DNA shown in grey and histones with protruding N-terminal tails in colour. Residues that can be modified by methylation or acetylation are indicated.

Image on the left reprinted by permission from Macmillan Publishers Ltd: Nature; G. Felsenfeld & M. Groudine (2003) "Controlling the double helix", *Nature*, 421(6921):448-53. Copyright 2003.
Image on the right is from L. Cui & J. Miao (2010) "Chromatin-mediated epigenetic regulation in the malaria parasite *Plasmodium falciparum*", *Eukaryotic Cell*, 9(8):1138-49. Published by American Society for Microbiology. Used with permission.

In addition to exchanging histone variants, the properties of nucleosomes can also be altered through covalent *histone modifications*. Histone proteins, especially H3 and H4, have unstructured N-terminal tails (and also C-terminal tail for H2A) that protrude from the nucleosomes, and these can undergo post-translational modification at specific residues, such as *acetylation* of lysines, mono- or di- *methylation* of lysines and arginines and tri-*methylation* of lysines, *phosphorylation* of serines and threonines, and also *ubiquitination*, *SUMOylation*, *citrullination*, and *ADP-ribosylation* [42]. These modifications can be introduced by some enzymes, like histone methyltransferases (HTM) and histone acetyltransferases (HAT), and removed by others, such as histone demethylases and histone deacetylases (HDAC).

Hypoacetylation of histones has traditionally been associated with the more densely packed heterochromatin whereas hyperacetylation has been associated with more open chromatin conformations. The situation is not always so simple, however, since acetylation can be associated with both active and repressed portions of the DNA [43]. The same is true for methylation. For example, the modifications H3K4me3 (tri-methylation of lysine (K) residue 4 in histone H3), H3K9me1 and H3K27me1 are commonly found in promoter regions of active genes whereas H3K9me2 and H3K27me3 are linked to gene repression [44, 45]. It has been hypothesized that such covalent modifications form a “histone code” which is read and interpreted by various enzymes that specifically recognize and bind to these modified histones. This includes, among others, ATP-dependent chromatin remodelling complexes that can move around nucleosomes, evict them or incorporate new nucleosomes [42, 46].

Distinct histone marks and combinations of marks are frequently found in relation to genomic elements that serve specific functions [41, 47-49]. The H3K4me3 mark, as mentioned before, is abundant within promoter regions of active genes, but mono-methylation of the same residue (H3K4me1) is more common in enhancer regions [50]. Active regulatory regions are also marked by acetylations at H3K27 and H3K9. Gene bodies of transcribed genes, on the other hand, are marked by H3K36me3 and H4K20me1, whereas genes repressed by Polycomb group proteins are marked by H3K27me3. This repressive mark can also be found together with the active mark H3K4me3 in so-called “bivalent promoters” which are associated with genes that are currently silent but are poised for rapid activation at a later time [51]. Histone modifications can therefore be important in determining how the different parts of a genome should be interpreted, and simply by examining the “chromatin state” [52] of a region we can obtain clues to the role of the underlying DNA sequence (Figure 7).

The organisation of DNA into chromatin constitutes an additional level of gene regulation which controls access to the DNA sequence itself. Before the basal transcription complex can bind to the core promoter of a gene and initiate transcription, the chromatin conformation in the region must be relaxed and any nucleosomes blocking the core promoter must be evicted. The presence of nucleosomes and condensed chromatin can also prevent binding by regular transcription factors. However, some “pioneer factors” are able to bind to condensed regions, and they can open up the chromatin for subsequent binding by other factors as well, either by disrupting the local configuration of nucleosomes themselves or by recruiting histone acetyltransferases or other remodelling complexes [53, 54]. Conversely, some repressing TFs can recruit histone deacetylases to remove acetylations which leads to compaction of the chromatin [14].

Also nucleotides in the DNA sequence itself can be covalently modified. The most common DNA modification in vertebrates is methylation of the fifth carbon in cytosines (5meC). In humans, this mark is introduced by DNA methyltransferases (DNMT) which specifically target cytosines that are followed by a guanine in the DNA sequence ("CpG-sites"). Because methylated cytosines often spontaneously deaminate into thymines and these are not always correctly repaired by the DNA-repair apparatus, such

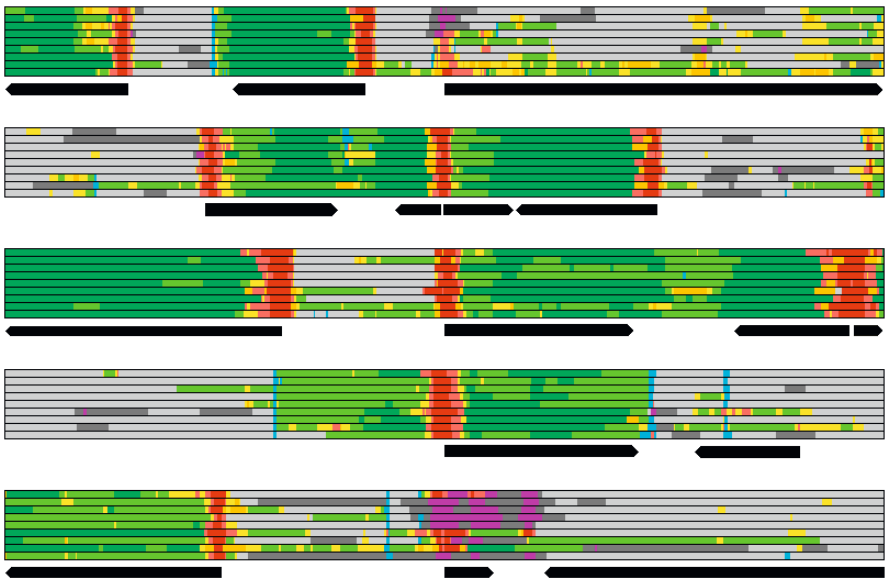


Figure 7: This figure shows the "chromatin states" of nine human cell-types in five 100 Kbp regions. The states were derived by combining information about eight histone modification marks plus data on CTCF-binding.

Colour codes: Red=active promoter, pink=weak promoter, purple=inactive or poised promoter, orange=strong enhancer, yellow=weak enhancer, green=transcribed region, light green=weakly transcribed, blue=insulator, dark grey=polycomb-repressed, light grey=heterochromatin or repressed/copy number variation.

Transcribed genes annotated in the Ensembl database are indicated in black. Based on the chromatin state alone it is possible to predict the start of gene regions and the direction of transcription. Note, however, that some of the promoters appear to be bi-directional and control transcription of genes extending outwards in both directions. In the fourth region, for instance, it appears that there could be a gene extending to the left of the active promoter region (although no such transcript is currently annotated in Ensembl). Notice also how the insulator elements in blue can delimit chromatin domains with different histone modification profiles.

This epigenetic map can give us information on the activity of genes in different cells. For example, in the top region, the third gene appears to be suppressed (packed in heterochromatin) in the first 3–4 cell-types but is transcribed (at least weakly) in the remaining ones. Similar tendencies can be seen for the middle gene in the bottom region which is only active in 2–3 cell-types but is marked as "poised" in the rest.

CpG-sites are much less frequent than other dinucleotides [55]. The majority of the remaining CpGs in humans are methylated [56], except within so-called “CpG-islands”, which are regions with high GC-content and a surprisingly high number of CpG-sites [57, 58]. CpG-islands are often found in promoter regions of genes, and methylation in these promoter regions is linked to gene repression [59]. Some repressive factors, like MeCP2, will only bind to methylated binding sites [60] whereas other factors might not be able to bind to their target sites if these are methylated [61]. Methylation of CpG-sites can also lead to gene activation, for instance in a reported case where a methylation within an insulator element would prohibit binding of the CTCF-factor and thereby allow an enhancer to activate transcription of the nearby *Igf2* gene [62].

The various histone variants, histone modifications and DNA modifications together constitute what is called the “epigenetic code”. Whereas the DNA sequence itself is always the same in all the cells of an organism, the epigenetic information on top of the genome can vary between cells of different types. This additional level of regulation is thus very important in cellular differentiation, since it can control which genes and regulatory elements that are accessible for use by the different cell types [63, 64].

Regulation of different classes of genes

Genes in higher eukaryotes can be divided into three major classes based on their expression profiles across cell-types and developmental stages, and it has been suggested that the genes within each class share similar fundamental modes of regulation, at least to some extent [65].

Type I) Tissue-specific genes

Tissue-specific genes are genes that are only expressed in a single tissue, or at most a few tissues, in terminally differentiated adult cells. The core promoters of these genes generally have low GC-content and often have a TATA-box and an Inr-element which constrain the placement of the transcriptional complex. The transcription start sites for these genes thus tend to be sharply defined. The position of the TSS is usually occupied by a nucleosome which has to be moved out of the way before transcription can commence, and the nucleosomes in the region are disorderly positioned. The promoters of active genes are marked by H3K27ac, H3K4me2 and H3K4me3 (but the latter is only present downstream of the TSS). Tissue-specific genes tend to be predominantly regulated by CRMs in the proximal promoter region.

Type II) Housekeeping genes

Housekeeping genes encode for proteins that are required for normal functioning and maintenance in all cells, such as those involved in DNA replication and repair and in cellular metabolism. These genes are thus ubiquitously expressed across all cell types. Promoters of housekeeping genes are characterized by an open chromatin conformation, and the region around the TSS is devoid of nucleosomes, which means that they can readily be bound by the transcription complex. In fact, even for inactive genes the polymerase might already be recruited to the core promoter, just waiting for the appropriate signal to initiate transcription [66]. The nucleosomes on either side of the nucleosome-free region in the promoter have a very ordered configuration and they are marked by H3K4me3, H3K4me2 and H3K27ac. The genes are typically also under regulation be a few nearby enhancers which are marked by H3K4me1. The promoters have high GC-content and are characterized by a single short CpG-island which overlaps the TSS (but only in vertebrates as invertebrates do not have CpG-islands). They do not contain TATA-boxes but are rather associated with weaker motifs with more flexible positioning. The placement of the polymerase in the promoter is therefore not strictly defined, and these genes tend to exhibit a rather broad distribution of TSSs.

Type III) Developmentally regulated genes

These genes are involved in multicellular development and differentiation, and their expression is precisely coordinated across different cells in a tissue or anatomical structure. This is the most diverse class when it comes to regulation, and the genes are usually regulated by multiple long-range enhancers in addition to the promoter. The promoters have high GC-content and many long CpG-islands (in vertebrates) which often extend into the gene body itself, and the whole gene body is also marked by H3K4me3 (unlike housekeeping genes which only have this mark in the promoter region). Developmentally regulated genes that are not needed tend to be silenced by Polycomb group proteins which deposit H3K27me3 marks throughout the gene and promoter regions. There are also many “poised” genes which have bivalent promoters displaying both activating and repressing histone marks. The genes in this class have a sharper TSS-distribution than the housekeeping genes, and the promoters tend to have an Inr-element which also often occurs in combination with a DPE.

Post-transcriptional regulation

Proper gene regulation is crucial for controlling the amount of active proteins of different types in the cell. Although this thesis will only focus on gene regulation at the transcriptional level, it is important to keep in mind that regulation can also occur at later stages. For example, genes can be regulated post-transcriptionally by *microRNA* (miRNA) [67, 68]. MicroRNAs are short RNA molecules that are transcribed from the genome in the same way as mRNA, but they are not translated into proteins. Rather, miRNAs can suppress target mRNAs by binding to complementary sites in these transcripts. Depending on the level of complementarity between the miRNA and the binding site, the target mRNA will either be cleaved directly or translation of the mRNA will be repressed. These miRNAs can thus regulate the amount of transcribed mRNA that will be available for translation into proteins.

Even after gene translation is completed, the resulting protein products might have to be *post-translationally modified* in order to be fully functional. This will often involve the addition of e.g. phosphate groups to the protein which can change the conformation of the protein and convert it from an inactive to an active state [69]. Proteins may also have to form complexes with other proteins in order to be useful, and it would then be possible to regulate the formation and activity of the whole complex simply by regulating just one of its components [70]. Finally, proteins that are no longer needed can be tagged for destruction and degraded [71].

Experimental detection of motifs and binding sites

Determining the binding motifs of transcription factors and identifying the sites where the factors bind in the genome is an important step towards unravelling the complex gene regulatory networks of an organism. This chapter summarizes some of the most popular methods for experimental characterization of transcription factor binding.

Electrophoretic mobility shift assay (EMSA)

Electrophoretic mobility shift assay (also called *gel shift assay* or *band shift assay*) is perhaps the simplest method for detecting protein-DNA interactions [72]. It is based on *gel electrophoresis*, which is a technique that can be used to separate molecules such as DNA, RNA and proteins by size or electric charge. The method can analyse short fragments of DNA ranging in length from a few dozen to a few hundred bases. The DNA fragment is first amplified to create a large number of identical copies. One portion of this is kept as a control while the rest is allowed to interact with proteins that could potentially bind to the DNA. To perform the assay, a mixture containing agarose or polyacrylamide is prepared and poured into a box where it solidifies into a gel. The control DNA is poured into a small well at one end of the gel and the DNA-protein mixture is poured into another. Next, an electric current is applied to the gel to create an electric field. The negative pole is located close to the wells while the positive pole is one the other side of the box. Since DNA molecules are negatively charged due to the phosphates in the backbone, they will start to travel through the gel towards the positive pole. Bigger molecules will encounter more resistance in the gel and will therefore travel slower than smaller molecules. After a period of time, the current is turned off and the location of the DNA in the two lanes is determined. Molecules that have travelled at the same speed will end up in the same location and form bands in the gel. If the DNA has been tagged with a radioactive marker such as P-32, these bands can be visualized by exposing the gel to an X-ray sensitive film (or fluorescent tags can be used instead). If the DNA was not bound by the proteins, the DNA fragments in both lanes will have travelled the same distance and the two resulting bands will be aligned. However, if the proteins did bind to the DNA, the large DNA-protein complexes will have travelled shorter than the unbound DNA and the two bands will be shifted relative to each other as illustrated in Figure 8. A variation of the

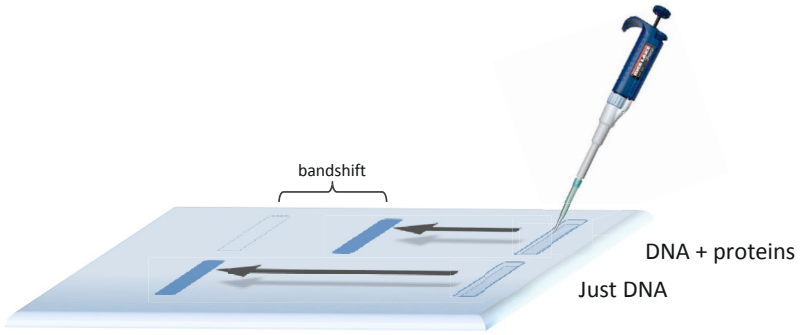


Figure 8: EMSA assay. Tagged DNA fragments mixed with proteins are deposited in one well while a control mixture with only tagged DNA fragments is deposited in a second well. When a current is applied to the gel, the DNA molecules will begin to travel towards the positive pole which is located on the far side. If the protein in question has bound to the DNA, the fragments in that lane will have travelled a shorter distance than the fragments in the control lane.

basic method, called *supershift assay*, induces a more prominent band shift by attaching an antibody to the protein to create a larger complex which moves even slower [73].

The EMSA method can show that a DNA fragment has been bound by a protein, but it is not able to pinpoint the exact location of the binding site within a long DNA sequence. However, it is possible to narrow down the region by running several experiments with increasingly smaller DNA fragments or by selectively mutating positions in the DNA sequence and see if this disrupts protein binding.

DNase footprinting

DNase footprinting [74] is somewhat similar to EMSA in that it also relies on gel electrophoresis and can be used to analyse protein binding to DNA fragments up to a few hundred base pairs in length. But unlike EMSA, this method is able determine the exact location and base sequence of the binding site within a larger DNA fragment. DNase footprinting is based on a method for sequencing DNA proposed by Maxam and Gilbert [75] which proceeds as follows: First, the DNA fragment to be analysed is amplified and the copies are labelled with a P-32 radioactive tag (or fluorescent tag) at the 5' end. The DNA is then divided into four batches, and each batch is treated with a different set of chemicals which cause the DNA fragments to degrade and break up into smaller pieces. The various chemicals used will introduce breaks after a specific type of nucleotide in each batch (A, C, G and T respectively). The chemical reactions are only

allowed to continue for a brief period of time before they are stopped, so the fragments will just be broken into two pieces on average. The exact locations were these breaks will happen will be arbitrary and vary from fragment to fragment, but they will always occur after a known base type depending on the batch. If all goes well, the batches will contain tagged fragments of all possible lengths ranging from one single nucleotide to the full length of the original DNA fragment. Next, each batch is loaded into a well in a gel and the fragments are separated by electrophoresis. This will give rise to several bands in each lane, corresponding to the number of times the nucleotide appears in the sequence. As always, shorter DNA fragments will travel longer, and this makes it easy to deduce the DNA sequence from the resulting band patterns as shown in Figure 9.

For the footprinting assay, an additional batch of 5'-tagged DNA fragments is prepared and mixed with proteins that could potentially bind to the DNA. This mixture is treated with the endonuclease enzyme DNase I which under the right conditions will cleave DNA at any position (albeit with some bias). Again, this enzymatic reaction is stopped after a short time so that each fragment is cleaved just once on average. The crux of the method is that the DNase enzyme is unable to cleave the DNA at positions which are bound by a protein. Thus, if protein-binding has occurred, the batch will contain 5'-tagged fragments of all sizes except for fragments that end within the protein binding site. Any bound proteins are removed from the DNA before the fragments are separated by gel electrophoresis in an additional lane alongside the four sequencing lanes described earlier (all the batches should be run simultaneously on the gel). If no proteins bound to the DNA, the lane for the DNA fragments treated with DNase will contain bands wherever there are bands in any of the other sequencing lanes. But if binding did take place, the lane will contain a small stretch devoid of bands – called a *footprint* – corresponding to the location of the bound protein (see bottom lane in Figure 9).

The footprinting method is able to simultaneously detect several binding sites within the analysed DNA sequence (corresponding to multiple footprints), but if different types of proteins are tested at the same time it is not possible to determine which protein bound to which site.

Methods based on chromatin immunoprecipitation

EMSA and DNase footprinting are relatively simple and do not require a lot of resources, but they are somewhat limited in that they can only be used to analyse small isolated DNA fragments. In recent years, methods based on *chromatin immunoprecipitation* (ChIP) have gained a lot of popularity since they can be applied to detect protein binding *in vivo* across whole genomes in a single experiment [76-78]. Various approaches have been proposed, but the initial steps are mostly the same for all the methods (Figure 10).

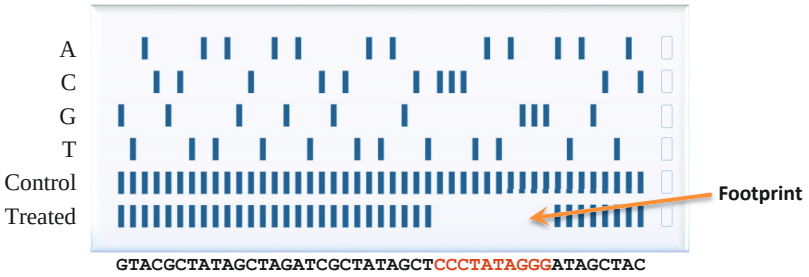


Figure 9: DNAse footprinting assay. The DNA is deposited in wells on the right side and the DNA fragments travel towards the positive pole on left side. The top four lanes are sequencing lanes where the DNA fragments have been cleaved after a specific base. Going from left to right, the first of the bands is found in the G-lane, and this means that the sequence starts with a "G". The next band is in the T-lane and corresponds to fragments with the sequence "GT". The third band is found in the A-lane and corresponds to "GTA" fragments, etc. Hence, by examining the location of the bands from left to right we are able to deduce the specific nucleotide sequence as shown beneath the figure. The two lanes at the bottom contain fragments cleaved by DNase. The first of these is a control lane with fragments cleaved at every position. The last lane contains DNA fragments that have been bound by proteins, and this lane clearly shows that a segment corresponding to the sequence "CCCTATAGGG" has been protected from cleavage.

First, DNA and proteins are allowed to interact either *in vivo* or *in vitro*, and any proteins that might have bound to the DNA are cross-linked with formaldehyde. This introduces covalent bonds between the DNA and the proteins and ensures that the proteins do not fall off during subsequent steps in the analysis. The DNA, which could be in the form of whole chromosomes, is cut into smaller fragments of about 200–1000 bp using endonucleases or through *sonication* (employing sound waves to forcefully vibrate the DNA until it breaks). Anti-bodies that are specific to proteins of interest are then used to precipitate the DNA-protein complexes. The anti-bodies are pre-attached to either magnetic beads, which allow the complexes to be fished out using a magnet, or to beads of agarose or sepharose which allow the complexes to be separated from the rest of the solution by centrifugation. Any unbound DNA is washed away before the formaldehyde cross-links are reversed through heating. The bound proteins are removed by enzymatic digestion and the recovered DNA is purified. This results in a pool of DNA which is enriched for fragments bound by the transcription factor.

The final step is to determine the genomic location of these DNA fragments, and this can be done in several ways. In the approach known as **ChIP-chip** [79] the fragments

are separated into single-stranded DNA which are tagged with a fluorescent dye and hybridized to a microarray. A microarray is basically just a small glass plate with thousands of tiny spots, each containing single-stranded DNA probes with known sequence composition. For instance, in a “genome-wide tiling array” each spot corresponds to a small part of a genome (typically 25–100 bp), and all the spots together cover the whole genomic sequence (or at least the interesting parts). The fluorescent-labelled fragments from the ChIP-sample bind to complementary probes on the microarray, and by measuring the light emitted from each spot, the amount of bound DNA can be determined. Often, a control sample containing total fragmented DNA (not enriched by ChIP) is tagged with a different colour dye and hybridized together with the ChIP-sample on the same microarray in order to correct for experimental bias.

New high-throughput DNA sequencing techniques have made it possible to sequence all the bound DNA fragments directly and map them to a genome to determine their locations. This approach is called **ChIP-seq** [44]. Although the DNA fragments can be a few hundred bp long, typically only the first ~25 bp of each fragment is sequenced since this is usually enough to uniquely determine its genomic location. If a histogram showing the number of sequencing reads mapping to each genomic position is created, regions bound by a TF will exhibit a characteristic peak (or rather two peaks slightly offset from each other corresponding to reads mapping to the direct and reverse strands). Many software tools have been designed to identify such peaks corresponding to potential TFBS [80].

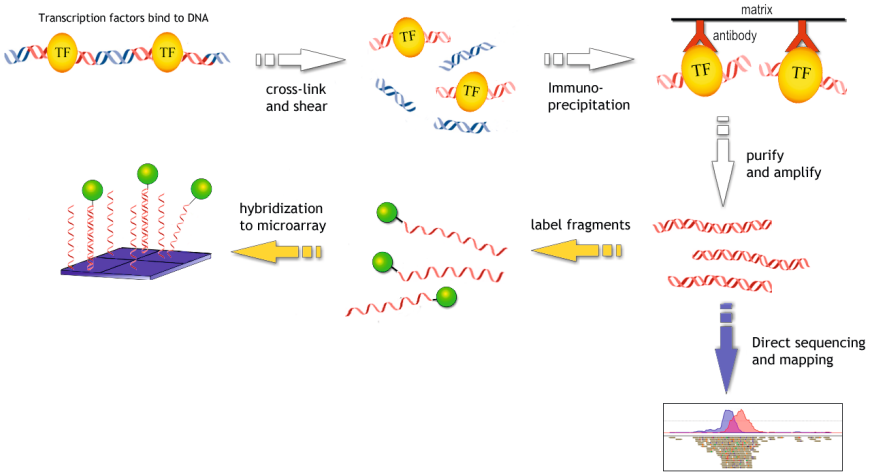


Figure 10: Steps involved in methods based on ChIP. After the TF-bound DNA fragments have been extracted by immunoprecipitation, they can either be labelled and hybridized to a microarray (ChIP-chip) or sequenced directly (ChIP-seq).
(Image adapted from Wikipedia)

Because the DNA fragments retrieved by chromatin immunoprecipitation are relatively long, the traditional ChIP-chip and ChIP-seq methods do not have very high resolution and are typically only able to identify the location of a TFBS to within a few hundred bases. However, a newly proposed extension of the ChIP-seq method called **ChIP-exo** claim to be able to pinpoint the binding sites with single-base precision [81]. After the binding and fragmentation steps have been carried out as usual, the DNA is treated with *lambda exonuclease* which digests the fragments in a strand-specific 5'-to-3' direction. The exonuclease will eat away at the 5'-ends until it reaches a fixed distance from the bound protein while leaving the 3'-ends intact. When the sequence reads from these fragments are mapped to the genome, all the reads will align and result in a sharp, narrow peak corresponding to the exact location of the binding site. This is in contrast to the standard ChIP-seq method where different fragments for the same TFBS can have different 5'-ends, thus resulting in a much broader peak.

One limitation of ChIP-based methods is that specific antibodies must be available for the transcription factors one wishes to analyse, and this will not always be the case. Other problems include false positive predictions caused by *unspecific binding* (because a protein happened to be close to some parts of the DNA when it was cross-linked) or *indirect binding* where the TF of interest did not bind the DNA directly but rather bound to another protein which in turn bound to the DNA. The bioinformatics steps of these methods are not without challenges either. Especially the peak-calling process for ChIP-seq can be far from trivial [82, 83].

ChIP-based methods are not only used to locate binding sites for transcription factors but are also popular for determining genomic distributions of other DNA-binding proteins, such as nucleosomes that exhibit particular histone modifications [44, 49].

DamID

DNA adenine methyltransferase identification (DamID) is another method that can be used to determine genome-wide *in vivo* binding of transcription factors or other DNA-binding proteins in eukaryotic cells [84, 85]. The method requires that the TF to be studied is first fused to an enzyme from *E. coli* called Dam. Dam is a methyltransferase which attaches a methyl group to the N6 nitrogen of adenines within the specific sequence pattern G<u>A</u>TC. When the TF–Dam fusion protein binds to a motif recognized by the transcription factor, the tethered Dam will start to methylate any GATC patterns found in the vicinity of the binding site up to a few thousand bases away. Since natural methylation of adenines is rare in eukaryotes, the presence of a large number of such modified nucleotides within a region can be taken as an indication of a nearby TFBS. To identify the methylated regions, the DNA is treated with the restriction endonuclease *DpnI* which cleaves DNA only at methylated GATC sites. Specially designed adapter

sequences are ligated to the ends of the resulting DNA fragments before the DNA is treated with yet another restriction endonuclease (DpnII) which cleaves DNA at non-methylated GATCs. This ensures that only fragments associated with methylated GATCs have adapters. A PCR step is performed to selectively amplify these methylated fragments using primers that are complementary to the adapter sequences. The genomic locations of these fragments can then be determined by hybridization to microarrays. It should be noted that the transcription factor in the TF–Dam fusion protein does not have to be bound to a TFBS for the Dam enzyme to work. Some non-specific methylation by diffuse TF–Dams will always occur, and it is important to correct for these background methylation levels to avoid too many false positive predictions. The same experiment should therefore be repeated using pure Dam rather than fusion proteins as a control, and the fragments from the two experiments should be tagged with different coloured fluorescent dyes and hybridized to the same microarray. This allows the ratio between TF-targeted methylation and background levels to be determined.

One advantage of the DamID method is that no antibody for the TF is needed, so it can be used to analyse any TF that can be fused with Dam. Also, the method does not require a large number of cells, and introduction of the Dam enzyme does not noticeably interfere with normal cell functioning. The resolution achieved with DamID is somewhat limited, however, and the location of a binding site can only be determined to within a few thousand base pairs, which is much worse than ChIP-seq.

SELEX

The invention of the method now popularly known as SELEX (systematic evolution of ligands by exponential enrichment) or *in vitro selection* is usually attributed to two independent research groups that both published papers based on similar principles in August 1990 [86, 87] (although related ideas had been proposed earlier [88]). The SELEX process is inspired by natural evolution whereby a pool of initially random DNA oligonucleotides are “evolved” through generations and the “fittest” individuals in each generation (i.e. the DNA sequences that bind the TF strongest) are selected and may produce offspring (identical or near-identical copies) for the next generation.

The full assay proceeds as follows:

1) An initial library of random DNA oligonucleotides is created, typically consisting of about 10^{13} to 10^{15} sequences with a size of 20–100 bp.

2) The DNA pool is incubated together with transcription factors of a selected type. Some of the DNA sequences will hopefully bind the TFs, but probably only weakly in the first rounds.

3) The bound complexes are separated from the unbound complexes ("selection step"). This is usually done by *affinity chromatography*.

4) The bound DNA is subsequently released and the sequences are amplified by PCR to create a new DNA pool for the next iteration ("reproduction step").

5) Steps 2–4 are repeated as necessary until the DNA pool has converged to relatively few unique sequences that have the highest affinity and specificity for the target TF.

6) After the final iteration cycle, a representative number of DNA oligos are sequenced to determine the binding motif for the TF. The proportion of each unique sequence in this sample will reflect its binding affinity for the TF.

The selection criteria in step 3 can be made more stringent with each iteration to retain DNA sequences with progressively higher binding affinity to the target TF. This can be done by reducing the concentration of the TF or by changing the binding and washing conditions (buffer composition, volume, incubation time, etc.). The initial random DNA pool will probably not contain all possible sequence motifs, and the sequences that have the highest affinity to the TF might not even be present in the initial pool. So, to increase the chance of finding the optimal binding sequences, some variation can be introduced to the sequences in the reproduction step, either by modification or mutagenesis (for instance by using error-prone PCR for amplification).

Since the SELEX method relies on artificial, randomly generated DNA sequences, it cannot be used to discover actual binding sites in genomic DNA. On the other hand, it is a good method for inferring the binding motif for a transcription factor and determining its preference towards different sequences.

Protein-binding microarrays

Protein-binding microarrays [89-91] are microarrays where the probes consist of double-stranded DNA sequences. When transcription factors are added to the microarray, they will bind to probes containing their binding motifs. By tagging the TFs with fluorescent dyes, the amount of TFs bound to each spot on the microarray can be measured. The probes can contain a range of artificial sequences (typically 10 bp in each spot) which together cover all possible sequence combinations and allows the binding affinity of the TF to any sequence to be determined [20, 92]. Alternatively, genome-wide tiling-arrays can be used to detect possible binding sites across whole

genomes in a single experiment [93]. Since this method is purely *in vitro*, it is not affected by chromatin conditions and can find sites that are difficult to detect with condition-dependent ChIP-based methods because they are rarely used *in vivo*.

Computational detection of motifs and binding sites

Performing the necessary experiments in a wet lab is the only way to know for sure that a transcription factor has bound to DNA, to determine the exact location and sequence of the binding site, to identify the nature of the bound TF (if it is not already known), and to assess the regulatory impact of the binding event. However, such experiments can be both expensive and time-consuming, especially in earlier times when they had to be conducted manually, one binding site at a time. As a result there has been great interest in using computers to analyse DNA sequences, and this has led to a wave of motif discovery programs aimed at predicting motifs and binding sites computationally.

Recent years have seen a rapid growth in throughput for experimental methods, and it is now possible to analyse binding events for a TF in complete genomes in a matter of days. Although large-scale experimental projects, such as ENCODE [94], aspire to identify all binding sites in the human genome by systematically mapping sites for a large number of transcription factors across several cell lines, it will still take some time before this work is completed. And besides, these high-throughput experimental methods produce such huge amounts of data that it is absolutely necessary to use computers to do the analysis anyway. With the advent of next-generation sequencing, genomes for new organisms are being sequenced at an unprecedented rate, and these genomes are predominantly annotated with the use of bioinformatics tools in automated pipelines. It is therefore safe to say that computational motif discovery is more important and relevant now than ever before.

Computational methods for motif discovery can be divided into two main classes:

1) Methods that predict possible binding sites in a DNA sequence based on prior knowledge of the motif that a TF binds to (this approach is called *motif scanning*)

2) Methods that do not have such knowledge and must simultaneously discover both the motif itself and the corresponding binding sites (this is often referred to as *de novo* or *ab initio* motif discovery).

Motif representation

Whether binding sites are detected experimentally of predicted computationally, computer programs need some way to represent the binding motif with a model which can be used to distinguish sequence patterns that a transcription factor might bind from those it will not bind. An ideal motif model should be both sensitive and precise. That is, the model should cover as many actual binding sequences as possible, while at the same time not match too many spurious sequences. One possible model would be to just use a list of known unique binding sequences directly. This model would perhaps seem ideal because it would match all known binding patterns and none other. However, a major drawback with such a model is that it would not be able to generalize to other unseen sequences that could also bind the TF, so the actual sensitivity of the model can be less than perceived. This is especially true if the model was created from a limited number of TFBSs which might not fully reflect the sequence variation of the actual binding motif. Many other ways of modelling binding motifs have been proposed, with the most popular models being *consensus sequences* and *matrices*.

Consensus sequence

The simplest and most compact motif model is the "consensus sequence" whereby a set of binding sequences is represented with a single sequence pattern. This consensus sequence can be constructed in several ways. One could, for instance, use the binding sequence which occurs most often in the set as the consensus or create an altogether new sequence based on the most frequent nucleotide in each position (note that such a sequence might not be present in the original set). Since this model is just a single sequence, it is by nature very strict and it might not even cover all the binding sites that went into creating it. To allow the model to generalize to other resembling binding sequences, one can decide that the model should implicitly include all sequences that deviate from the consensus in a fixed number of positions, usually a single position for short motifs and two or more for longer motifs. When a consensus model is used in this way, it is often referred to as a "mismatch model" [95]. A mismatch model does not usually impose any constraints on which positions are allowed to vary, and this can make the model overly promiscuous. A motif of length *N* with *m* mismatches will cover $\sum_0^m \binom{N}{m} 3^m$ sequence variants, so a motif of length 8, for example, will match 25 sequence variants if one mismatch is tolerated and as many as 277 sequence variants when allowing up to two mismatches.

Although most transcription factors usually tolerate some sequence variation in their binding sites, the nature and location of the variation can have great impact on the binding ability of a factor. Some nucleotides within a binding site might not actually be in physical contact with the TF at all, so it does not really matter what types of

nucleotides occur in these positions. For other positions the TF might have absolute demands regarding the nucleotide type, and any deviation from the preferred type might abolish binding altogether. Finally, there could be positions where one of several different nucleotide types are acceptable to the TF, but the strength of the binding may vary depending on the particular nucleotide.

A more descriptive way to model binding motifs is with *degenerate consensus sequences* where each position can contain either a specific nucleotide or one of two or more alternative nucleotides. The International Union of Pure and Applied Chemistry (IUPAC) has defined a standard set of symbols to refer to ambiguous nucleotides [96], and these are listed in Table 1. Using the IUPAC symbols, the two binding sequences “TGACTA” and “TGCGTA”, which differ in the middle two positions, can be represented by the degenerate consensus “TGMSTA” (where “M” means “either A or C” and “S” means “either C or G”). This motif would also generalize to match two additional sequences, “TGAGTA” and “TGCCTA”, which may or may not be able to bind the same TF.

A straightforward way to derive an IUPAC consensus sequence would be to represent each position with the most specific symbol covering all nucleotides encountered in that position across the binding sites. Although this would result in a sensitive model, it might not be very precise and neither does it reflect well the relative frequencies of nucleotides in each position. For example, if 9 out of 10 binding sites have a “T” in one position but the last sequence has a “C”, this variability would then be modelled by the double-degenerate symbol “Y”. However, this symbol would intuitively suggest that the two nucleotides are equally likely to appear in this position, which is clearly not the case. As an alternative approach, Cavener [97] proposed assigning unambiguous symbols (A,C,G,T) to positions where a single nucleotide occurs in more than half of the sequences and more than twice as often as the second most frequent nucleotide. If no single nucleotide satisfied this condition, then a double-degenerate symbol (Y,R,M,K,S,W) could be used if the combined frequency of the two most frequent nucleotides exceeded 75%. If neither of these rules applied, the position would be assigned the symbol “N”. These rules have later been extended with an additional rule dictating the use of triple-degenerate symbols (B,D,H,V) in cases where three different nucleotides appear in a position and the criteria for a more specific symbol are not met. Several other strategies for assigning consensus symbols have been suggested as well [98].

Degenerate consensus sequences can alternatively be represented as *regular expressions*. For example, the IUPAC consensus sequence “TGAYCV” can be written as “TGA[CT]C[ACG]” using POSIX notation where the brackets group together alternative nucleotides for a position. Regular expressions have additional advantages as well, since they can easily model optional *insertions* and *variable gaps* in motifs.

Symbol	Represents	Interpretation	Complement
A	A	Adenine	T
C	C	Cytosine	G
G	G	Guanine	C
T	T	Thymine	A
R	A + G	Purine	Y
Y	C + T	Pyrimidine	R
M	A + C	Amino	K
K	G + T	Ketone	M
W	A + T	Weak	W
S	C + G	Strong	S
B	C + G + T	Not A	V
D	A + G + T	Not C	H
H	A + C + T	Not G	D
V	A + C + G	Not T	B
N	A + C + G + T	Any	N

Table 1: IUPAC nucleotide symbols

Variable gaps are a common feature of TFs that bind as dimers. For these TFs, each monomer part binds to one half-site and the full binding motif consists of two such half-sites which, depending on the structural flexibility of the dimer, might be separated by a fixed or variable number of nucleotides. For example, the regular expression "TACN{2,4}GTA" describes a binding motif consisting of two palindromic "TAC" half-sites separated by a variable gap between 2 and 4 bp long.

Matrix model

Whereas the consensus sequence model only provides information about the most frequent base (or bases) in each position, the matrix model holds information regarding the relative preference for each of the four base types. For a motif of length m this information is stored in a $(4 \times m)$ numerical matrix where each of the four rows corresponds to one of the base types and the columns correspond to positions in the motif.

Stormo *et al.* [99] used a perceptron learning algorithm to determine the matrix values that would allow the model to optimally discriminate between a set of positive sequences (containing binding sites) and a set of negative sequences (without binding sites). However, when a set of binding sites are known, a more straightforward way of assigning the values is normally employed. The most basic matrix model is the

position count matrix (PCM) where the value for a base b in position i in the motif is simply set to the number of times that base appears in position i among all the known binding sites. If we normalize a PCM by dividing the base count values with the number of binding sites, we get a *position frequency matrix* (PFM) instead. Staden [100] further log-transformed these frequency values to arrive at a *position weight matrix* (PWM)[1]. If the probability p_b of observing the base b at any position in a genome is known, we can create an alternative weight matrix which takes these background probabilities into account by using the log-ratio of the frequency of the base from the binding sites $f_{b,i}$ compared to the background frequency: $\log\left(\frac{f_{b,i}}{p_b}\right)$. Assuming that each position in the motif contribute independently and additively to the binding energy of the TF (and assuming also that the frequencies obtained from the set of known TFBS are not biased), it can be shown that these log-ratio weights are optimal in the sense that they maximize the probability of binding to all TFBS [101, 102].

Given a PFM we can calculate the *information content* [103] of position i in the motif with the following equation:

$$\mathrm{IC}(i) = 2 + \sum_{b \in A,C,G,T} f_{b,i} \log_2 f_{b,i}$$

Information content (IC) is inversely related to the concept of *entropy*, which is a measure of the uncertainty associated with a random variable [104]. In fact, the formula for entropy is just the negative of the second term in the equation above. Information content and entropy are often measured in number of *bits*, where one bit is the amount of information required to distinguish between two equally likely outcomes. If we know from the PFM that "T" is the only base that can appear in position i within a binding site, then there is no uncertainty and the entropy for that position is 0 bits (and the IC is 2 bits). If we know that both "A" and "T" are equally likely to appear, the entropy is 1.0 because we are missing one bit of information in order to uniquely specify the actual base. If all four bases have equal probability, the entropy would reach its maximum of 2.0 (and IC would be 0). Thus, the more conserved a position is the more information it contains and the entropy (uncertainty) will be lower. Conversely, positions that allow more variation have more uncertainty and hence less information content. By summing up the IC for each position we get the total information content of the motif.

[1] To avoid taking the logarithm of zero, a small value called a *pseudocount* is usually added to all entries in the frequency matrix as described on page 38.

To account for skewed background probabilities, a generalization of the previous equation, called *relative entropy* or Kullback-Leibler divergence [105], can be used instead:

$$\mathrm{IC}(i) = \sum_{b \in A,C,G,T} f_{b,i} \log_2 \frac{f_{b,i}}{p_b}$$

This equation is exactly the same as the previous one for the special case when all bases have equal background probabilities ($p_b = 0.25$ for all b's). The Kullback-Leibler divergence is equivalent to a log-likelihood ratio measuring the degree of disagreement between the base frequencies from the motif and the background frequencies.

The information content of a motif is directly related to the number of times the motif is expected to occur within a random sequence. This expected frequency is given by the formula: 2^{-IC}. So, for example, a non-degenerate 6 bp motif (which has an IC of 6×2 = 12) is expected to occur once in every $2^{12} = 4^6 = 4096$ bp. It should be noted, however, that the matrix model itself does not actually discriminate between matching and non-matching sequences. Rather, the matrix can be used to calculate a match score towards a sequence which reflects the free energy of binding to the sequence relative to a random background [106].

Schneider and Stephens introduced an intuitive graphical way of representing binding motifs which they called "sequence logos" [107]. In a sequence logo, each position is drawn as a stack of base letters on top of each other. The letters are sorted according to base frequency, with the most frequent base for the position appearing on top. The height of each base letter in the stack is proportional to its relative frequency, so if e.g. the base "A" occurs twice as often as "T" in a position, the height of the "A" letter will be twice the height of "T". Finally, the whole stack is scaled according to the IC-content for that position. Positions that have high conservation among the binding sites will thus have taller stacks than positions which allow for greater variation. An example of a sequence logo is shown in Figure 11e.

Higher-order models

A potential limitation with the traditional consensus and matrix models is that they assume that all the positions within the binding motif are independent, which implies that the probability of observing a specific base in a certain position is not influenced by which bases occur in other positions. Although this seems to be a reasonable assumption in most cases (see e.g. references in [102]) it is not valid in general [102, 108, 109], and several higher-order models have therefore been proposed to capture more complex relationships within motifs.

a)

```
1: TGACGTCA
2: TGACCTCG
3: TGACGCCA
4: TGCGCTCA
5: TGACGTCG
6: AGAGGTCA
7: TGATGTCG
8: AGAGGTCA
```

AGAGGTCA	Consensus based on the most frequent binding sequence
TGACGTCA	Consensus based on the most frequent base in each position
TGASGTCR	IUPAC consensus sequence

e)

b)

2	0	7	0	0	0	0	5
0	0	1	4	2	1	8	0
0	8	8	2	5	0	0	3
6	0	8	1	0	7	0	0

A
C
G
T

c)

0.250	0.000	0.875	0.000	0.000	0.000	0.000	0.625
0.000	0.000	0.125	0.500	0.250	0.125	1.000	0.000
0.000	1.000	0.000	0.375	0.750	0.000	0.000	0.375
0.750	0.000	0.000	0.125	0.000	0.875	0.000	0.000

d)

0.000	-3.258	1.224	-3.258	-3.258	-3.258	-3.258	0.892
-3.258	-3.258	-0.655	0.673	0.000	-0.655	1.357	-3.258
-3.258	1.357	-3.258	0.392	1.072	-3.258	-3.258	0.392
1.072	-3.258	-3.258	-0.655	-3.258	1.224	-3.258	-3.258

Figure 11: Different motif representations. **a)** Eight binding sites have been aligned, and shown beneath are three alternative consensus models for the binding motif. **b)** The four rows of this position count matrix (PCM) show respectively the number of A, C, G and T's in each of the positions. **c)** The position frequency matrix (PFM) was created by normalizing the PCM based on the number of binding sites. **d)** A log-transformed weight matrix. Positive values indicate that the base have a higher frequency in the binding sites compared to a background. **e)** Sequence logo representation of the motif.

The strongest dependence relationships often seem to be between adjacent positions in motifs [110], and one simple representation that is able to model such relationships is a *dinucleotide matrix* with 16 rows and $(m-1)$ columns where the value in each cell reflects the probability of observing a specific pair of nucleotides starting in a position [111]. The dinucleotide matrix can be viewed as an example of an *inhomogeneous first-order Markov model*. In a Markov model representation, the probability of observing a

specific base in a position will be dependent on the contents of the N positions that precedes it (for a model of order N). With *homogeneous* Markov models, the probabilities are not dependent on the specific location of the position in question, but with *inhomogeneous* models the probabilities can vary from position to position. For example, in a homogeneous model the probability that a "G" will be followed by a "C" might be 30% irrespective of the location of these bases, while for an inhomogeneous model a "G" in the second position of a binding motif can have a 40% probability of being followed by a "C" in the third position whereas a "G" in the seventh position can have a 25% probability of being followed by a "C" in the eight position. If some positions in the motif are independent while others are dependent on preceding positions, a *variable-order* Markov model can be used. Dependencies that involve positions located further apart from each other can be modelled using e.g. *Bayesian networks* [112, 113] or other variations on Markov models such as *permuted Markov models* [114].

Sharon *et al.* [115] proposed modelling motifs in a very general way as a set of features where each feature would be associated with a weight reflecting the importance of that feature for the TF–DNA interaction. The overall strength of a potential binding site could then be calculated by summing up the weighted contributions of all features that were present in the site. A feature can be as simple as saying that "there should be a 'C' in position 3 (with a weight of 0.6)", or they can capture dependencies between positions by saying e.g. "there should be a 'C' in position 3 and a 'G' in position 8 (with a weight of 0.2)". Features can thus be used to model the same single or combined nucleotide probabilities as the previously mentioned motif models, but features can also represent more general global properties of a motif, for example that "the motif should be palindromic" or that "the total GC-content should be greater than 60%".

Even though higher-order motif models have been shown to improve the precision of binding site discovery, they have not gained much popularity. One reason for this is that higher-order models require more training data to fit all their parameters, and this was a problem before the advent of high-throughput experimental methods when the number of known binding sites for each transcription factor was still limited. Higher-order models also have a tendency to *overfit* the training data, which is especially problematic if the data is noisy. Thus, the simpler models based on consensus sequences and $(4 \times m)$ matrices are still the most widely used ways of representing motifs. On the other hand, homogeneous higher-order Markov models have been very popular for representing *background distributions* that model the probabilities of observing specific bases and combinations of bases in the sequence *outside* of binding sites.

Motif scanning

Over the years, binding motifs for a large number of transcription factors have been experimentally determined and published (see e.g. [116]), and motif models have been collected into publicly available databases such as TRANSFAC [117], JASPAR [118], ScerTF [119], SCPD [120], YEASTRACT [121], RegulonDB [122] and PRODORIC [123] to name just a few. Using such existing motif models, it is relatively easy (at least in theory) to match these against sequences in order to predict new potential binding sites, and a host of motif scanning tools have been developed for this purpose.

The motif scanning process itself is relatively straightforward: given a sequence of length L and a motif of length m, the motif is aligned with the sequence at every position from 1 to $(L - m + 1)$ and a match score for each position is calculated. Sites starting at positions that score above or equal to some decided cutoff threshold are considered good matches to the model and thus potential binding sites for the corresponding transcription factor (Figure 12). Matches to the opposite strand can be considered by reverse complementing either the model or the sequence.

For consensus sequence models, the match score could simply be the number of positions in the motif that match the aligned positions in the sequence. The cutoff threshold would then typically be set to m (corresponding to 100% match) or "$m - e$" if e mismatches are tolerated. An alternative scoring scheme could be used for IUPAC models whereby matches to degenerate positions in the motif are assigned lower scores. For example, matches to A, C, G and T could receive a normal score of 1.0 for that position, matches to double-degenerate symbols (Y,R,K,M,S,W) could receive a score of 0.5, matches to triple-degenerate symbols e.g. 0.33, and matches to the "N" symbol would not result in additional points since this will always be matched anyway [124].

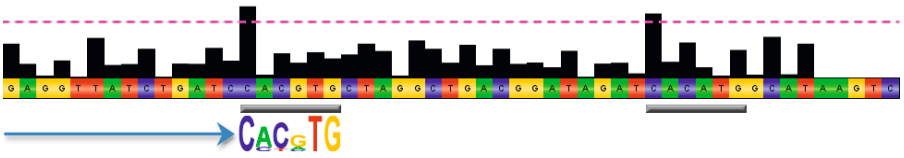


Figure 12: Motif scanning. The match score to the motif "CACRTG" at each position is illustrated with bars above the sequence. Positions that score higher than the selected threshold indicated by the pink dashed line are considered matches. This sequence contains two such matching sites, the first one with binding sequence "CACGTG" and the second with binding sequence "CACATG".

To calculate a match score for a matrix model, one would combine the matrix values for the bases that match the sequence in each position. In other words, if the sequence contains the base b at aligned position i relative to the motif start, one would use the value $f_{b,i}$ from the matrix for that position. Harr *et al.* [125] proposed two different ways to combine the values across all positions: either summing up the values and dividing by the motif length (arithmetic mean score) or multiplying all the values together. Since they used matrices that had been normalized by dividing the values with the highest value in each column, both of these approaches would lead to a maximum score of 1.0 for the best matching sequence. If the matrix is a regular PFM, the frequency values can be interpreted as probabilities of observing the different bases in different positions according to the motif model, and multiplying the values for the matching bases would then result in a score which corresponds to the joint probability that the sequence matches the motif at the aligned position. Multiplying across values from a PFM is equivalent to summing across the log-transformed values of a PWM. If a background-corrected log-ratio PWM is used, a positive sum score would imply that the sequence segment under consideration is more likely to be a match to the motif than a match to the background (and vice versa for negative scores).

If a PFM has a value of zero for a matching base, this will always lead to a total score of zero if the values are multiplied (or equivalently a value of $-\infty$ if log-transformed frequencies are added up). The aligned sequence segment will then never be considered a potential match to the motif, even if all the other positions in the site receive the highest possible scores. Although this could be a biologically valid conclusion (since the presence of the wrong base in a position might block TF binding), such a result is usually considered to be too strict. If the motif model is based on a limited number of TFBS, the reason for the zero entry in the model could simply be that none of the candidate sites happened to contain that particular base in that position, not because it is incompatible with TF binding. So, to correct for potential small sample bias and allow all sequences a chance of being considered a match, a small value called a *pseudocount* is usually added to each value in the matrix to ensure that none of the entries are zero [126].

The most important consideration when performing motif scanning is choosing a sensible cutoff threshold for the match score, since this will directly affect the sensitivity and precision of the search. Selecting a high threshold will lead to fewer matches, but this could potentially mean that we miss out on some true binding sites. On the other hand, setting the threshold too low could result in a lot of false positive predictions.

Many motif scanning programs convert the raw match score to a relative match score using the formula: $score = \frac{(score_{raw} - score_{min})}{(score_{max} - score_{min})}$, where $score_{max}$ and $score_{min}$ are

the highest and lowest obtainable scores according to the model. An intuitive choice for a cutoff threshold would then be to select some high relative value such as e.g. 85% or 90% to ensure that predicted sites have a high degree of similarity to the motif model. Although this is a fairly common, such thresholds will always be somewhat arbitrary. Staden chose a different approach where he calculated the match score towards all the binding sites that went into creating a matrix and then used the lowest of these values as the threshold to ensure that all known binding sequences would be classified as matches [100]. This would assign individual cutoff thresholds to different matrices, which could potentially be more beneficial than using the same threshold for all. However, if there is much variation in the binding sites the threshold would have to be set rather low, and this would lead to many false predictions. Some have attempted to derive thresholds that could in some sense be considered optimal for discriminating between binding sites and the background [119, 127, 128]. For example, the motif scanning program MATCH [129] comes with a choice of three different pre-calculated thresholds for each matrix in the TRANSFAC database: a sensitive threshold to minimise the false negative rate (minFN), a strict threshold to minimise the false positive rate (minFP) and a threshold to minimise the sum of both of these errors (minSUM). To decide the "minFN" threshold they generated potential binding sequences by sampling from the PFM and chose a cutoff value that would recognize at least 90% of these sequences. For the "minFP" threshold they scanned sequences taken from the second exons of genes (which are not expected to contain functional TFBS) and chose the lowest cutoff value for which no matches were found. Another feature of the MATCH program is that it operates with two separate thresholds to fine-tune the selection of sites. First, a match score is calculated for the "core" of the motif, which is defined as the five consecutive bases within the motif that have the highest combined IC-content. Only if the core region scores above a required "core threshold" does the program proceed to calculate a score for the entire motif and compares this to a second "matrix threshold".

In addition to calculating a raw match score, some motif scanning programs can also assess the statistical significance of this score by estimating how likely it would be that a match with such a high score could occur in the sequence simply by chance [130-132]. These programs will usually allow the cutoff threshold to be applied to the p-value rather than the raw match score. Some methods can even make use of higher-order background models to improve discrimination between binding motifs and background sequence [133, 134].

De novo motif discovery

Searching for binding sites without any prior knowledge of the binding motif can be likened to searching for the proverbial needle in the haystack – except that in this

scenario the needle itself would also be made of hay. That is to say, since both the binding sites and the rest of the DNA sequence are made up of the same four nucleotides, there is really not much that distinguishes a binding motif from the surrounding background. Nevertheless, the task is not completely impossible and many different strategies have been proposed to solve the problem.

Nearly all *de novo* motif discovery methods are based on the same basic principle: if a set of sequences are expected to bind the same transcription factor, they should all contain the same binding motif. It should therefore be possible to identify this motif by searching for a short pattern that is common to all of these sequences (see example in Figure 13). The first motif discovery program to operate in this way was created by Korn *et al*. in 1977 [135], and this program has since been followed by several hundred other methods [95].

The set of input sequences can be selected in several ways. One popular choice is to analyse promoter sequences from *co-expressed genes*, i.e. genes that have similar expression patterns. The motivation behind this is that genes that are expressed in the same way might also be regulated in the same way by the same transcription factors. Another possibility is to consider *orthologous* promoter sequences, which are promoter

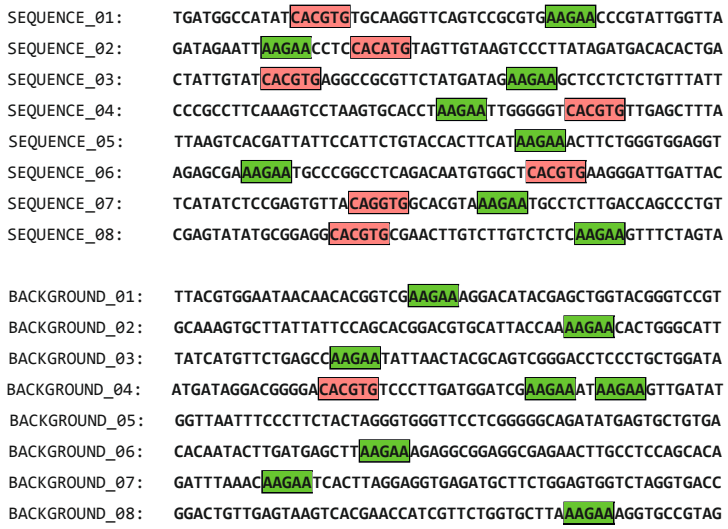


Figure 13: *de novo* motif discovery. The pattern "CACGTG" (with some variation) shown in red is found in 7 out of the 8 target sequences and the "AAGAA" pattern in green is found in all of them. These two patterns are therefore good candidates for being binding motifs for common regulators. However, the "AAGAA" motif is not unique to the target dataset but is also frequent in a background set based on randomly selected sequences.

sequences for the same gene from different species. This assumes, of course, that the gene is regulated by the same factors in all of these species. One final option is to use sequences that have been shown to bind the same transcription factor through experimental methods such as ChIP, protein-binding microarrays or EMSA.

Although simple enough in principle, there are some immediate concerns with this basic approach. First of all, the binding patterns may not be identical in all the input sequences, so a motif discovery program should take this into account and allow for some variation in the motif. Second, some of the input sequences might have been wrongfully assumed to be bound by the transcription factor, when in fact they are regulated by other means and do not contain the target binding motif at all. Forcing the motif discovery program to predict a binding site in every sequence can thus introduce unnecessary noise in the motif model. It is therefore common among methods to allow some sequences to lack a binding site, but require for instance that at least k of the input sequences should contain one (the number k is called *quorum*). The biggest concern, however, is that since the DNA alphabet is so small and the patterns considered are short, the probability of observing any such pattern within a longer sequence is relatively high. For example, in a 1000 bp long uniformly random DNA sequence, every 5 bp sequence pattern is expected to occur about once on average simply by chance, and a 6 bp pattern with one allowed mismatch would occur between four and five times. A set of sequences is therefore likely to contain many similarities that are not related to the target binding motif. To further complicate matters, genomes are not random and they can contain a lot of *repeat elements*, such as *transposons* and *satellite repeats* [136]. Repeat elements are sequence patterns that occur many times throughout a genome, and they can easily be mistaken for a common transcription factor binding motif if they appear often in the input sequences [137]. Because of this, it is often recommended to mask out such repeat regions in the sequences to avoid misleading the motif discovery program [138]. Another way to alleviate the problem with common sequence patterns is to compare the occurrence frequency of each pattern in the input sequences to the number of times they occur in a separate set of negative sequences that are not expected to contain the target binding motif. The true motif should presumably be present in most of the input sequences but few or none of the negative sequences [139-142]. Instead of an explicit negative dataset, one could also use a Markov background model from which the expected frequency of each sequence pattern can be calculated [143]. Patterns that are statistically over-represented in the input sequences compared to their expected frequency would then be good candidates for the actual target motif. Different statistics can be employed to assess the level of significance of over-representation, such as *z-score* [144-147], *binomial test* [148, 149] or *Fisher's exact test* [142, 150, 151].

The next challenge to consider is how one should proceed to identify the best candidate pattern(s). One possibility is to enumerate all 4^m patterns of a given length m and count

how many times each of these patterns appear in the input sequences compared to their expected frequencies. Closely resembling matches to each pattern could also be considered to account for variability in the binding motif. Alternatively, one can enumerate patterns containing IUPAC degenerate symbols [143]. This kind of search can be done exhaustively as long as the motif length is small, and the optimal solution is then guaranteed to be found. However, since the number of patterns to go through grows exponentially with the motif size, this approach is infeasible for longer motifs.

Improvements in speed can be achieved by not enumerating all possible patterns but only those that are actually present in the input sequences (with possible mismatches) [152], by using efficient data structures (such as suffix-trees) to organize the data [146, 153, 154], or by pruning parts of the search-space that are not likely to contain promising solutions [155, 156].

Another popular approach is to search through different binding site alignments. One binding site is selected from each sequence, and the chosen sites are compared to see how similar they are, using either information content or log-likelihood ratio as a measure of similarity. If we want to analyse N sequences, each of length L, to find a binding motif of size m, the total number of possible site alignments would be $(L - m + 1)^N$, and this is only considering one strand. An exhaustive search through all of these alignments would normally be out of the question (finding the optimal solution is actually NP-hard [157, 158]), so some heuristic search procedure is usually employed. For instance, the method CONSENSUS [159] starts off by making an exhaustive search through all pairwise site alignments in the first two sequences only. An initial set of $(L - m + 1)$ matrices, built from each site in the first sequence combined with its best match in the second sequence, is used to iteratively search the remaining sequences in a greedy fashion. For each one of the initial matrices, the best matching site is found in the third sequence, and the matrix is updated by incorporating this new site. The process is repeated with each new sequence until all sequences have been covered. In the end, the matrices with the highest information content represent the best site alignments and are thus the best candidates for the binding motif. In order for this method to work well, the target motif should have strong matches in the first few sequences so that the search is not misled. To increase the chance of that happening, the method should preferably be run several times with different ordering of the sequences, and this is done by default in the most recent version of CONSENSUS.

Expectation-Maximization (EM) is a search heuristic that was first used for motif discovery by Lawrence and Reilly [160] but has since been widely adopted by other methods, including MEME [161], Improbizer [162] and PhyME [163]. Unlike the greedy CONSENSUS method, which incorporates new sites one sequence at a time, EM begins with an initial (possibly random) selection of candidate sites from all sequences (one from each). These sites are used to build a matrix model for the motif

and a separate model for the background is derived from the frequencies of bases in sequence positions that are not covered by these sites. If we postulate that an observed sequence contains a binding motif at a known position, it is easy to calculate the probability that this sequence could have been generated by the models. Using Bayes' rule, we can also calculate the likelihood that each position in the observed sequence could contain the motif. It should be noted that the position with the highest such likelihood need not correspond to the site that actually went into creating the motif model. An updated model is then created based on all possible sites, but each site is weighted by the likelihood that a site would start in that position (and similarly for the background). The process is repeated several times until the model converges. Even though the initial motif model will probably have very low information content, the model will improve with each iteration because sites that better match the model will be given more weight when the model is updated.

One problem with EM is that it is a *deterministic hill climbing* algorithm which can get stuck in local maxima in the search space and fail to find the globally optimal solution. It is therefore advisable to run the algorithm several times with varying initial configurations in order to explore more of the search space. An alternative approach would be to use a *stochastic hill climbing* heuristic instead, and this is done by the Gibbs sampler method [164]. Like EM, the Gibbs sampler starts by selecting one candidate site from each input sequence to create and initial matrix model and then updates this model over several iterations. In each iteration, one of the sequences is chosen (in a fixed order or at random) and a matrix model is created based on the candidate sites from all the other sequences. The matrix, together with a background model, are used to calculate a match score for each position in the sequence that was held out, and these scores are converted into a probability distribution from which a new candidate site for the sequence is sampled. Thus, the Gibbs sampler exchanges the current candidate site from the sequence with a new site which is chosen at random, but sites that represent better matches to the current model are more likely to be selected. Unlike EM, this stochastic approach does not necessarily converge, so the algorithm is commonly stopped after a fixed number of iterations or when a solution has been found that is deemed to be good enough. Gibbs sampling is one of the most popular search heuristics for motif discovery and is used by methods such as AlignACE [165], ANN-Spec [166], BioProspector [167], MotifSampler [168], PRIORITY [169] and SeSiMCMC [170].

Lots of other heuristic search procedures have been applied to the motif discovery problem as well, including *simulated annealing* [137, 171], *ant colony optimization* [172], *particle swarm optimization* [173, 174], use of *genetic algorithms* to evolve site alignments [175-178] or to evolve the motif model directly [179-181]. Some methods treat the motif discovery problem in terms of graph-theory where each k-mer word in the input sequences is represented as a node in a graph and nodes that correspond to

similar words are connected with edges. The problem of discovering common patterns in a set of sequences is then equivalent to finding cliques or high-density subgraphs [182-184].

Another approach to motif discovery, which is radically different from the ones previously mentioned, does not rely on sequences containing common motifs at all, but rather attempts to infer the binding motif for a transcription factor directly based on the known three-dimensional protein-structure of the TF and the contacts it makes with the DNA molecule when it binds [185, 186].

Module discovery

Many motif discovery programs are able to return multiple motif predictions for the same dataset. Enumerative methods, for example, can return the N most significant motifs encountered, and alignment-based methods can perform motif discovery first once to identify the most significant motif in the dataset and then mask out the binding sites for this motif and repeat the process in order to find the second most significant motif, etc. Nevertheless, the motifs returned by these methods are discovered independently of each other, and their binding sites might not even be located in the same sequences.

However, there are also several *module discovery* programs available that specifically aim to discover combinations of motifs for transcription factors that might work cooperatively to regulate genes ("composite motifs") [187, 188]. Some of these module discovery methods extend the basic strategy used by *de novo* motif discovery programs of searching for patterns that occur in several sequences, but instead of looking for binding sites for a single transcription factor they look for groups of co-occurring sites for multiple factors [189-193]. Other methods identify potential modules by looking for dense clusters of binding sites located within relatively short sequence regions. The program MSCAN [194], for example, slides a window of user-defined width across the sequence and estimates the combined statistical significance of all predicted TFBS that fall within the window. If the calculated p-value is below some preselected threshold, the binding sites in the window are considered to constitute a regulatory module. Another popular approach is to model the sequences with a (hierarchical) mixture model or hidden Markov model with two top-level states representing modules and background respectively. The task is then to find the chain of state-transitions that is most likely to generate the input sequence. In the process, each position in the sequence will be tagged as either belonging to the module state or not [195-198].

The distinction between *scanning* (searching for modules with known composition) and *de novo* discovery is less clear for module discovery than for single motif discovery.

On one hand there are methods that can search sequences for occurrences of strictly defined modules, for example a binding site for the transcription factor AP-1 followed by a binding site for Ets within 10 to 15 bp. At the other end of the spectrum are methods that can identify new modules given no other input than a set of sequences. In between these extremes we find methods that rely on varying amounts of prior information. For example, many module discovery methods require a library of single motif models (usually a collection of PWMs) to be provided which they can use to identify candidate binding sites for single transcription factors. These methods are called “motif aware” [188]. If such methods are given a very small motif collection as input, the composition of the target module will basically be pre-determined, and the methods are then reduced to mere module scanning methods. However, if they are provided with a large motif collection they will also have to discover which of these candidate motifs are involved in modules and which are not. Methods that do not rely on predefined motif libraries are called “motif blind”, but some of these might still require some supervision to guide them, for instance in the form of a training set of sequences containing known module instances [199].

Ensemble methods

No single motif discovery method is perfect and all methods can make mistakes, either by failing to identify a true motif or by falsely predicting a pattern to be a motif. Different methods will frequently return differing predictions when run on the same dataset. Even the same method can give varying results if it is run multiple times with different parameter settings (or even with identical settings if the method is based on a stochastic algorithm). It can therefore be instructive to try out several methods and compare their predictions to see if they are in agreement. As long as the methods do not all make the same mistakes, it should also be possible to take the results from an ensemble of methods and combine them into potentially more reliable predictions based on consensus.

One way to combine results from multiple tools is with a simple *voting* scheme: if at least N of the methods in the ensemble predict the same motif or binding site, that prediction is considered to be reliable. On the other hand, if too few methods predict the motif, it is regarded as a false prediction. Alternatively, one can just say that the motif that got the highest number of votes is the best candidate. It is also possible to use a slightly more advanced approach where each method’s vote is assigned a weight based on our confidence in the method. Variations on the voting scheme are employed by e.g. CEA [200], EMD [201] and MotifVoter [202].

The voting approach is based on the assumption that motifs which are found by a small minority of methods are more likely to represent spurious predictions. However, these

could of course also be true motifs that are missed by the majority, especially if the methods in question have different focus. Some ensemble methods deliberately employ a diverse set of motif discovery tools to increase the sensitivity of the search. For example, the SCOPE tool uses an ensemble of three methods that each covers a distinct part of the search space: the first method (BEAM) searches for non-degenerate motifs, the second method (PRISM) searches for short degenerate motifs, and the third method (SPACER) searches for long, highly degenerate motifs with gaps [203].

Evaluating the performance of computational methods

A very important question is of course: how well do these computational motif discovery methods work in practice? To what extent can we rely on the predictions that these programs make?

To evaluate the performance of a motif discovery program, we can test it on a dataset where we already know the correct answer. The predictions output by the program can then be compared with the answer to see if they are in agreement. Test datasets can consist of real genomic sequences where the locations of binding sites have been determined by experimental procedures, or they can be artificial datasets where a selected target motif has been planted somewhere in real or randomly generated background sequences.

One big problem with using real data is that genomic sequences tend to be under-annotated, which means that the sequences are likely to contain additional binding sites that we are currently unaware of. If a motif discovery program finds and reports these sites instead of the ones we know, it will be unfairly penalised because its predictions will then (incorrectly) be regarded as false.

With artificial datasets we can have full control over all the parameters, such as the size and degeneracy of the motif, the number and locations of planted sites, the length and composition of the background and the number of sequences in the dataset. By adjusting these parameters it is possible to tune the signal-to-noise ratio and influence the difficultness of the dataset [204]. Artificial datasets are used in the classic "(l, d)-motif challenge problem" where a fixed binding pattern of length l is planted in a number of synthetic sequences, but in each sequence this pattern is mutated in exactly d randomly selected positions to introduce variability [182]. Such datasets are very useful to assess the pattern-finding capabilities of motif discovery methods and see how much noise they will tolerate before they fail to identify the target motif. However, target sites that are constructed in this way will not necessarily reflect the true variability of a normal TF binding motif, so the problem is not directly equivalent to finding real motifs in real sequences. Another limitation with artificial datasets is that

any subtle signals in the flanking sequence around a binding site that could potentially influence its functionality will inevitably be absent. Real datasets where the genomic locations of the sequences are known also have the advantage that motif discovery programs can potentially utilize additional annotation data or information related to these sequences if they are able, such as e.g. the distance to the nearest gene start, the conservation level of each position in the sequence or the presence of DNase hypersensitive sites, nucleosomes and other epigenetic marks.

A compromise between real and artificial datasets is to use “semi-artificial” datasets where *real* binding sites are flanked by random background sequences. This will ensure that the target motif has the characteristics that would be expected of a real motif and that no other binding sites are present to confuse motif discovery programs. And if the genomic locations of the target sites are known, it will still be possible to utilize the additional information mentioned above.

Some datasets gain popularity and are frequently used when testing programs. The most famous benchmark compilation for motif discovery assessment is probably the one published by Tompa *et al.* [205]. They created 52 datasets (plus 4 negative control sets) based on real binding sites retrieved from the TRANSFAC database. Each dataset contained binding sites for one specific transcription factor and each one was issued in three different versions: a “real” version consisting of the actual promoter sequence containing the binding site, a semi-artificial version called “Markov” where the flanking sequence around the binding sites was replaced by a new sequence randomly generated according to a third-order Markov model, and a “generic” version where the binding site was planted in a different randomly chosen promoter sequence from the same genome. Swapping the backgrounds in the “generic” datasets meant that the sequences in each dataset were less likely to contain common binding sites for other transcription factors besides the target factor.

Sandve *et al.* [206] used machine learning to analyse these datasets in order to estimate an upper bound on the performance that could reasonably be expected by motif discovery programs. They found that in many cases the datasets suggested by Tompa were too hard, and it would actually be impossible to discriminate the binding sites from the background using traditional motif discovery procedures. They therefore proposed a new improved benchmark suite with 50 datasets (also based on data from TRANSFAC) where they made sure that it would be at least theoretically possible for programs to identify the target sites. In addition they made a second harder benchmark consisting of 25 cases that could not be “solved” using the standard motif models (mismatch model, IUPAC and matrix model) but would require motif discovery programs to employ more advanced representation models.

To complement the benchmark by Tompa *et al.*, which was based on eukaryotic binding sites, Hu *et al.* [200] created a prokaryotic benchmark suite with binding sites for *Escherichia coli* taken from the RegulonDB database. This benchmark consists of two collections of datasets. In the first collection (with 62 datasets) the sequences are full intergenic regions with a single known target binding site in each. In the second collection (70 datasets) the binding sites are surrounded by a fixed number of flanking bases ranging from 20 to 800 bp on either side.

Quest *et al.* published a tool called the "Motif Tool Assessment Platform" (MTAP) that can be used to automatically create benchmark datasets with selected properties and evaluate motif discovery methods on these datasets [207]. The binding sites used for the datasets come from many different databases.

Module discovery programs have often been tested on two datasets published by Wasserman and Fickett [31] and Krivan and Wasserman [32] which contain regulatory modules driving tissue-specific gene expression in muscle and liver cells respectively. However, the modules in these datasets are heterogeneous so they are not ideal for testing programs that require the same set of motifs to appear in all sequences.

Another frequently used dataset for module discovery consists of modules implicated to be involved in the anterior-posterior segmentation of the *Drosophila* blastoderm [208]. Ivan *et al.* [209] used these modules along with additional data from the REDfly database [210] to create 33 benchmark datasets in a fashion similar to the "generic" datasets introduced by Tompa. The annotated modules were taken out of their original context and placed inside other sequences from the non-coding part of the *Drosophila* genome. Each new background sequence, which was confined to be ten times the size of the planted module, was required to have a GC-content similar to the native context of the module. A similar approach was employed in a benchmark study by Su *et al.* [211].

Evaluating binding site predictions

The strictest form of evaluation is to compare the binding site predictions made by a program to the known correct answer at the *nucleotide level.* If a program correctly predicts that a nucleotide is part of a binding site, this is called a *true positive* prediction (TP). A nucleotide correctly predicted as not being part of a binding site is a *true negative* (TN). If a nucleotide is predicted as lying within a binding site when in reality it is not, this is called a *false positive* prediction (FP) or Type I error. A missed binding site position, on the other hand, is called a *false negative* (FN) or Type II error. The number of TP, TN, FP and FN predictions serves as basis for several common performance measures which are listed in Table 2.

Measure	Formula
Sensitivity	$TP/(TP+FN)$
Specificity	$TN/(TN+FP)$
Positive predictive value	$TP/(TP+FP)$
Performance coefficient	$TP/(TP+FP+FN)$
Average site performance	$\frac{1}{2}\times\left(\frac{TP}{(TP+FN)}+\frac{TP}{(TP+FP)}\right)$
F-measure	$2TP/(2TP+FP+FN)$
Accuracy	$(TP+TN)/(TP+TN+FP+FN)$
Correlation coefficient	$\frac{(TP\times TN)-(FP\times FN)}{(TP+FN)(TN+FP)(TP+FP)(TN+FN)}$

Table 2: Common performance measures

Sensitivity (Sn), also known as *recall*, is a measure of the fraction of binding site positions that have been correctly predicted by a program. It is found by dividing the true positive predictions (TP) by the total number of true sites (the ones correctly predicted (TP) and the ones missed (FN)). A motif discovery method that displays high sensitivity is less likely to miss out on true binding sites (commit type II errors). This measure should never be used as the sole indicator of a program's performance, however, since it is possible for a method to obtain a perfect sensitivity score simply by predicting all nucleotides as lying within binding sites.

Sensitivity is thus commonly considered in conjunction with other performance measures, such as for instance *specificity* (Sp). Specificity is analogous to sensitivity except that it applies to non-binding site nucleotides. It is the number of nucleotides correctly predicted as background divided by the total number of true background nucleotides. However, the usefulness of the specificity measure is somewhat limited in relation to motif discovery since the size of the background is usually significantly larger than the binding sites in common benchmark datasets. As long as a method does not make too many or too large binding site predictions, the TN-score will dominate the expression and the specificity score will tend to always be near perfect.

A more informative measure then is the *positive predictive value* (PPV), also called *precision*, which is the proportion of binding site predictions made by a program that actually correspond to real binding sites. Methods that score high according to PPV are less likely to make spurious binding site predictions (commit type I errors).

While sensitivity is the fraction of correctly predicted binding sites in relation to all true sites and PPV is the fraction of correctly predicted sites in relation to all predicted sites, the *performance coefficient* (PC), also known as *phi-score* or *Jaccard index*, captures aspects of both of these measures. PC is the number of correctly predicted sites divided by the union of true and predicted sites. A more straightforward combination of sensitivity and PPV is the *average site performance* measure (ASP), which is simply the arithmetic mean of Sn and PPV. The related *F-measure* is the *harmonic* mean of Sn and PPV. The F-measure is equal to ASP when the values for Sn and PPV are identical, but if Sn and PPV are different (e.g if one is high but the other is low) then the score for the F-measure will be closer to the smallest of these values.

Each of the statistical performance measures mentioned so far are based on only two or three out of the four parameters TP, TN, FP and FN. They are thus aimed at measuring particular aspects of a programs performance, and it is often possible to design a motif discovery program so that it optimizes one or more of these metrics at the expense of others. For instance, programs can generally obtain high sensitivity scores by making numerous binding site predictions (which could be detrimental to the PPV score if many of these are false) or obtain high specificity scores by being more conservative and make fewer predictions (which could give low sensitivity scores).

One measure that combines all four parameters is *accuracy*. Accuracy is simply the fraction of total nucleotides that are correctly classified, either as binding site or background. Although this measure provides a more overall view, the result can still be biased if the number of binding site nucleotides in a sequence is skewed compared to the number of background nucleotides, which is commonly the case for motif discovery benchmark datasets. So, like specificity, the accuracy measure has somewhat limited use since it has a tendency to give an exaggerated impression of performance.

Another measure that combines all four variables but also accounts for differences in the number of binding site and background nucleotides is the *correlation coefficient* (CC). CC is a measure of the overall agreement between the predicted and the true instances. A CC score of 1.0 means that the predictions made by a program is in perfect agreement with the true binding sites, while a score of -1.0 would imply the complete opposite (that a program has predicted all true binding sites as background and all the background as binding sites). A CC score close to zero implies that there is no statistical correlation between the predictions made by a program and the location of true binding sites. Such a score would be expected if the predictions were based purely on random guessing.

Nucleotide level evaluation can sometimes be considered too strict, since it can severely penalize predictions that are even the slightest bit off target even though the predictions themselves mostly overlap with true sites. For example, if a dataset contains binding

sites for a motif of length 12 bp that has a highly conserved core of 6 bp but more variable flanks and a motif discovery program only predicts this middle core as the motif, the sensitivity can never reach higher than 0.5, even if the locations of all the sites were essentially correctly predicted.

An alternative to *nucleotide level* scoring is *site level* scoring where one does not consider whether each individual nucleotide is predicted correctly or not, but rather whether the location of a predicted site overlaps to some degree with a true binding site. A predicted TFBS is considered a *true positive* at the site level if it overlaps a true TFBS with more than e.g. 25%. A prediction which does not overlap a true site (or has too little overlap) is considered a *false positive* prediction, while a true binding site that is not sufficiently overlapped by a prediction is a *false negative*. It is not intuitive to define what should constitute a *true negative* prediction at the site level. Hence, measures that are based on TN, such as specificity, accuracy and correlation coefficient, are normally undefined in this case.

Evaluating motif predictions

Another way to evaluate a motif discovery program is to compare the predicted motifs directly to the target motifs to see how similar they are [212]. An advantage with this approach is that it can also be used in cases where we do not know the exact locations of the binding sites within the sequences but we know the binding motif of the target TF, such as for datasets compiled from sequences shown to be bound by the same TF through ChIP-seq experiments or similar means.

To assess the similarity of the predicted and target motifs, the two motifs are aligned and compared position by position to calculate a total similarity score. A simple way to align the motifs is just to slide one motif along the other and select the relative offset which results in the best match, but more advanced algorithms that also allow for gaps in the alignment, such as Needleman-Wunsch [213] or Smith-Waterman [214], can also be used. Both orientations should be considered when determining the optimal alignment.

If the motifs are represented by consensus sequences, it is rather straightforward to see how many of the positions that match up, but if the motifs are represented by matrices we have to compare these column by column. Several metrics have been suggested for pairwise column comparison, including Euclidean distance, Pearson's correlation, Pearson's chi-squared test, average log-likelihood ratio, (symmetric) Kullback-Leibler divergence and Fisher-Irwin exact test. There are also many published tools available specifically for comparing motifs, such as STAMP [215], MATLIGN [216], TOMTOM [217] and T-Reg Comparator [218].

Problems and limitations with traditional sequence-based approaches

The field of computational motif discovery has traditionally focused mainly on the algorithmic and statistical challenges related to finding significant patterns in sequences. Since the search-space tends to grow exponentially for longer motifs or larger sequence sets, clever search heuristics must be employed to render the problem tractable. Also, good statistical models for representing motifs and background sequences are necessary to avoid problems with low-complexity regions and other spurious similarities. These are, of course, important issues to deal with and also very interesting areas to research in their own right. However, with respect to discovering functional binding sites, a purely sequence-based method has fundamental limits, and there are very good reasons for rather approaching the problem from a more biological point of view and try to incorporate more knowledge from this domain.

For example, assuming a set of co-expressed genes are indeed all regulated by the same TFs binding to the promoters (which is rarely the case), the next issue which must be resolved before one can proceed with motif discovery is to select which sequence regions to analyse, keeping in mind that binding sites can potentially be located both upstream and downstream of the transcription start site. For organisms with small genomes, such as bacteria and yeast, this is not that big a problem. Since their intergenic regions tend to be short, it is feasible to analyse the full upstream sequence extending all the way up to the end of the preceding gene. However, for humans and other organisms with larger genomes the closest upstream gene can be located tens of thousands of bases away, and it is difficult to determine the actual size of the promoter from the sequence alone. To ensure that the target sites are indeed included in the sequences, it is not uncommon to analyse rather arbitrarily large regions, but this will of course make the search harder for motif discovery programs. To further complicate matters, the genes in question could have alternative transcription start sites that might be located far apart, and it is by no means certain that the considered start sites are in fact the ones that are used in the given context [3]. These issues can potentially be resolved by considering additional biologically relevant data. For instance, information about chromatin accessibility and epigenetic marks can provide clues to the size and location of regions with potential regulatory roles, and data from CAGE-experiments can indicate the particular transcription start site which might be used under a specific condition [219].

A fundamental problem with sequence-based approaches, however, is that the mere presence of a potential binding motif in a sequence does not necessarily imply that it represents a functional binding site *in vivo* [124]. This is particularly important to keep in mind when performing motif scanning with known motif models, since even the best possible match to a model does not guarantee that the site will actually be bound by the TF. In fact, according to the “futility theorem” put forth by Wasserman and Sandelin, the vast majority of motif hits produced by scanning procedures are likely to represent

false predictions [220]. This might in some cases be attributed to bad or incomplete motif models that tend to produce and excessive number of hits. The popular motif database TRANSFAC, for example, contain many half-site motif models that only cover a part of the full motif that is actually required for binding. One of these motifs, a half-site for heat shock factor (HSF) with consensus sequence “aGAAn”, has an information content of only 5.24, which implies that this motif is expected to occur by chance once in every 38 bp in a random sequence. However, the full binding motif for the heat shock factor (which binds as a trimer upon activation) is actually composed of three such sites located in tandem (where each sub-site can have any orientation). Even if a motif model covers the full binding motif for a single factor, this factor could be dependent on forming complexes with other factors nearby in order to bind in a stable way. On the other hand, transcription factors can also be blocked from binding as a result of steric hindrance caused by other non-interacting factors that bind too close to their own target sites. The activity at a particular binding motif instance in the DNA will thus depend on the context, and while a binding motif can be bound by a TF in one place, this might not be the case for other locations containing the motif, even if the sequence pattern itself is exactly the same. This context-dependency will, of course, also involve the chromatin conformation, since this dictates the general accessibility of different regions in the DNA. Last but not least, no matter how accessible a binding site is, it cannot be bound by its associated factor unless that factor is actually expressed in the cell and is present in an activated state which makes it suitable for binding. And even if the TF is able to bind, this single event in itself might not necessarily be sufficient to influence the target gene. In fact, in a genome-wide study of TF binding in yeast, the authors estimated that only about half of the observed binding events actually resulted in a change in gene expression [221]. Hence, relying on additional information related to the state of the cells under investigation is necessary for determining whether a particular instance of a binding motif in the DNA sequence might actually correspond to an active and functional binding site under given conditions.

Utilizing additional information

As bioinformaticians have come to realize the inherent limitations of merely considering the primary DNA sequence when searching for functional motifs, more and more methods have been published which aim to incorporate other types of information into the motif discovery process as well.

Many motif discovery tools rely on information about *evolutionary conservation* as indicative evidence of functional elements. This is based on the knowledge that species evolve through the accumulation of base mutations and other alterations in their genomes. Some of these mutations can be beneficial for the affected individual while

others can be detrimental. Still others might be neutral and have no noticeable effect on the phenotype at all. Over time, beneficial and neutral mutations are allowed to spread throughout the population whereas detrimental mutations will be eradicated as a result of natural selection. Since most mutations that occur within functional genomic regions (including regulatory sites) are likely to be disruptive, these have less chance of being propagated to future generations compared to mutations outside of functional elements. Thus, if we use a global alignment program to align orthologous promoter sequences from two or more related species and then measure the cross-species variation in each position, we would expect functional binding sites to stand out as blocks of highly conserved positions ("phylogenetic footprints") separated by background sequence segments with higher variation (Figure 14) [222]. In order for this phylogenetic footprinting approach to work satisfactory, the species considered should be closely enough related to contain the same TFBSs but still divergent enough so that the surrounding sequence has had time to evolve. A similar approach called "phylogenetic shadowing" can be used for species that are very closely related provided that sequences from enough genomes are available to expose the evolutionary constraints on different positions [223].

It can sometimes happen that a binding site disappears from one location but its regulatory function is taken over by an equivalent site nearby [224]. If the position of a binding site has changed between species, it will be difficult to detect it through global sequence alignment procedures. However, such sites can still be found by local alignment methods which are normally used for motif discovery. To further highlight the important functional similarities between the sequences (as opposed to similarities just resulting from genomes being evolutionary close), Blanchette and Tompa included information about the phylogenetic relationship between the sequences as input to their motif discovery program FootPrinter [225]. Several other methods have also been developed which exploit phylogenetic conservation to identify binding sites (see e.g.

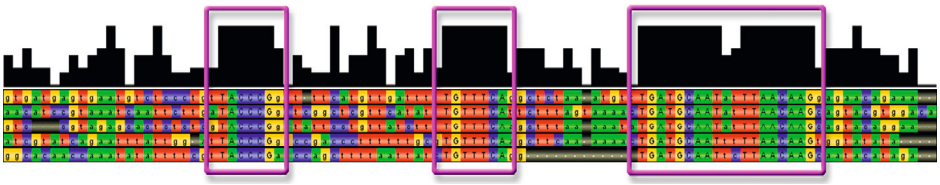


Figure 14: Phylogenetic footprinting. The figure shows an alignment of orthologous promoter sequences from five related yeast species. The conservation level of each position is indicated with black bars above the sequences. The highly conserved "footprint" regions marked with purple boxes are likely to represent functional binding sites.

references in [226, 227]). Conservation can be a very useful source of additional information when searching for motifs, since conserved sites in promoter regions are very likely to correspond to functional binding sites. However, this type of information will of course not be able to identify sites that are species-specific and therefore not present in other organisms.

Chromatin conformation is an important determinant of transcription factor binding, and it has been demonstrated that nucleosome occupancy is lower at functional TFBS compared to non-functional TFBS with the same motifs [228]. Narlikar *et al.* incorporated information about nucleosome occupancy when searching for novel motifs with their Gibbs-sampling based program PRIORITY [229]. Standard Gibbs-sampling methods work by calculating a posterior score for each sequence position based on their match to the current motif and background models, and these models are updated iteratively. The PRIORITY method modulates these scores by taking into account a *prior probability* that each individual position could contain an active binding site. Such position-specific priors, or *positional priors*, can be used to represent many kinds of sequence-related information. In addition to creating positional priors based on nucleosome occupancy, the authors of PRIORITY have also tried out priors based on binding motif features for TFs of different structural classes [169], information about DNA duplex stability [230] and phylogenetic conservation [231].

Because of the simplicity and general applicability of positional priors, similar functionality has been incorporated into many other motif discovery methods as well. Bailey *et al.* added support for positional priors in a recent version of their popular MEME tool and demonstrated its utility with a conservation-based prior [232]. Carvalho and Oliveira extended their combinatorial motif discovery algorithm RISOTTO by post-processing its output with a greedy procedure utilizing positional priors. The new method was named GRISOTTO [233]. Tang *et al.* tested their BayesMD method with positional priors based on conservation and local sequence complexity [234]. Qi *et al.* used priors based on ChIP-chip data to improve the spatial resolution of genome-wide TFBS predictions in yeast with their Joint Binding Deconvolution (JBD) method [235]. The ChIPMunk method developed by Kulakovskiy *et al.* was designed to use ChIP-seq peak profiles as positional priors [236].

The use of positional priors is not limited to *de novo* motif discovery; they can also be employed in conjunction with motif scanning methods. Lähdesmäki *et al.* combined PWM matching with additional evidence in the form of positional priors in their ProbTF tool [237]. They also proposed ways to combine evidence from multiple data sources – including conservation, regulatory potential [238, 239], nucleosome positioning, CpG-islands, ChIP-chip binding data and DNase hypersensitive sites – into a unified probabilistic framework. Information about DNase hypersensitive sites (DNase HS) has traditionally been considered somewhat of a “gold standard” for

identifying potential regulatory regions [240]. As previously mentioned in the chapter on experimental methods, DNase I is an enzyme that will cleave DNA relatively indiscriminately as long as it can access the DNA, but the presence of transcription factors and other bound proteins (including nucleosomes) will prohibit cleavage. Hypersensitivity to DNase cleavage is therefore a hallmark of open and accessible chromatin. The creators of the MEME suite updated their FIMO motif scanning tool [132] to make use of positional priors and tested it with priors based on both DNase HS and the four histone modification marks H3K4me1, H3K4me3, H3K9ac and H3K27ac [241]. Incorporating any of these features lead to improvements over standard PWM scanning, but the greatest gain was achieved with a combination of both DNase HS and either of the two histone marks H3K4me3 and H3K27ac.

Ramsey *et al.* trained linear classifiers based on PWM match scores plus one or two additional features. They found that information about "acetylation valleys" (regions with low acetylation in between regions enriched in acetylation) was particularly predictive of functional binding sites [242]. Such valleys are likely to represent nucleosome-depleted regions in open chromatin areas.

Pique-Regi *et al.* integrated information about cell- or tissue-specific experimental data (DNase HS and histone modifications) with general genomic information related to conservation and distance to nearest TSS in their CENTIPEDE algorithm. This data was combined with motif discovery based on 756 known and 49 novel motifs to produce a genome-wide map of TFBS in human lymphoblastoid cell lines [243].

The most ambitious effort to date with respect to integrating sequence-related features is probably the General Binding Preference (GBP) predictor by Ernst *et al.* [244]. They used logistic regression to train a classifier to discriminate between transcription factor binding sites and background sequence based on a linear combination of 29 features related to conservation, melting temperature, GC-content, CpG-islands, repeat regions, gene regions (including exons, introns, CDS, 5'UTR and 3'UTR), distance to TSS, DNase HS, histone methylations, H2A.Z, CTCF-binding and RNA polymerase II binding. Using this classifier they generated GBP scores for every position in the entire human genome and found that GBP was a good indicator of TFBSs, either by itself, or better yet, in combination with known motif models.

Module discovery programs and other methods predicting general regulatory regions can, of course, also benefit from incorporating additional information. One of the first module discovery methods to consider extra information was Stubb, which allowed orthologous sequences to be included as input to highlight conserved binding sites [198]. More recently, information about chromatin and epigenetics has been employed by tools such as CIS [245], Chromia [246], Combinatorial CRM decoder (CCD) [247] and i-cisTarget [248].

Even if a TF binds to a motif in the DNA, this binding event by itself might not have any regulatory effect, and genome-wide motif scanning will always produce many non-functional hits. However, motifs that are common to promoter sequences of genes shown to have similar differential expression are more likely to play an active role in this regulation. Most motif discovery programs only consider gene expression information in an implicit qualitative sense, but some methods include quantitative data about the expression level of each individual gene. Methods like REDUCE [249], Motif Regressor [250] and MARA [251] correlate these gene expression values with the presence of discovered binding sites to infer which motifs or combination of motifs are most likely to produce the observed behaviour. The MARA method can also incorporate additional evidence for each TFBS in its computational model, including the binding sites' conservation and position relative to the TSS.

Aim of study

Determining the location of transcription factor binding sites in the genome is an important step in elucidating the gene regulatory networks of an organism, and computational tools for motif discovery can offer a convenient, fast and cost-effective alternative to experimental methods. Although hundreds of motif discovery programs have been published, independent assessment studies have shown that the performance of such methods on datasets based on real genomic sequences is limited. This has previously been demonstrated by benchmarking of single motif discovery methods and has here been shown also for the discovery of composite motifs (paper I).

One reason for the limited performance is that most of these tools only base their predictions on information in the DNA sequence itself, but many other factors besides the presence of a binding motif will influence whether a transcription factor will actually be able to bind and exert its function. More recent tools have demonstrated that incorporating additional information, related to e.g. epigenetics, into the motif discovery process can improve the ability of computational methods to predict functional binding sites. As newly developed high-throughput experimental methods are now generating genome-wide data on various genomic features at an unprecedented rate, many of these could be relevant to consider with respect to motif prediction.

The aim of this project has hence been twofold:

1) To identify sources of data that can potentially be utilized to improve computational motif discovery (described in this thesis)

2) To make it easier for researchers to take advantage of such data by developing new software tools that can integrate various forms of information into the motif discovery process in a coherent and flexible way (paper II and III).

Integrating information to improve motif discovery

The need for improved motif discovery approaches

Bioinformaticians developing motif discovery tools usually report on favourable performances in their own publications. However, independent assessment studies, such as the seminal benchmark paper by Tompa *et al.* [205], tend to give less optimistic reviews of these tools, especially when analysing genomic sequences from higher organisms. Although the low performance in these independent studies can partly be attributed to the design of the datasets used and the stringency of the evaluation process, it is clear that there is still room for improvement with respect to computational motif discovery. In a benchmark study conducted by our own research group, we assembled datasets in such a way that it would be at least theoretically possible, from a machine learning perspective, to discriminate the target binding motifs from the background sequence. Nevertheless, the measured performances of two popular motif discovery methods on these benchmark datasets were still very low [206].

Later on, we conducted a companion assessment study of composite motif discovery methods which also demonstrated room for improvement (**paper I**). Most of the tools included in that study relied on a first step to scan the sequences with a provided motif collection to find a set of candidate binding sites, and then they proceeded to search through these candidates in order to identify potential modules. If the methods were only given the target motifs as input, most of them performed relatively well (at least on some datasets). However, when we diluted the motif collections by adding increasing numbers of decoy motifs, the performance dropped considerably for most methods since the target signal would then be drowned in noise (see Figures 5 and 6 in paper I). These results are more encouraging if we look at the situation in reverse, however. That is, if we start out with a noisy or hard dataset, it should be possible to increase the chance of identifying the correct motifs if we are able to reduce the noise and narrow down the search space. This can be done by incorporating additional knowledge into the motif discovery process. Since both of our benchmark suites are based on real genomic sequences rather than ones which are synthetically created, they are suitable for evaluating methods that are able to integrate information related to genomic features such as conservation level or epigenetics.

The motif discovery pipeline

Any strategy which aims to improve motif discovery by integration of additional information is closely linked to the structure of the motif discovery process itself. This includes both the discovery pipeline, as well as the data that are used in the pipeline. The process of discovering motifs and binding sites computationally can be broadly divided into four steps:

1. **Selecting which sequences to analyse**

   For *de novo* motif discovery this will usually involve selecting sequences that have a high likelihood of being regulated by the same transcription factors, such as segments bound by the same TF according to ChIP experiments, orthologous regulatory sequences from related species, genes for proteins that are involved in the same processes, genes that are co-expressed (and hence potentially co-regulated) or even a combination of the aforementioned. An issue to keep in mind when analysing co-expressed genes is that genes can have several alternative promoters, and the canonical start sites which are annotated in gene databases might not be the ones which were used for the transcripts expressed in a particular experiment. One must also choose the exact sequence region to consider for the analysis; i.e. how many bases to include upstream or downstream of the TSS. Longer sequences will be more likely to contain the target motifs, but they will also introduce more noise and make it harder for the motif discovery method. Finally, some methods depend on a second negative dataset to use for comparison and sequences to include in this set might also have to be decided on.

2. **Pre-processing**

   After the selected sequences have been obtained, they might require some pre-processing before commencing with motif discovery, for example to clean them up by masking low-complexity regions or other repeats. If additional types of information are to be incorporated into the motif discovery process, this data might also have to be processed at this stage.

3. **Motif discovery**

   When the sequences and additional data are ready, they can be passed on to a motif discovery or motif scanning program for analysis. Multiple programs can optionally be run on the same data or the same program can be run several times with different parameter settings to broaden the scope of the search.

4. **Post-processing and validation**

   In the final step of a typical pipeline, the results obtained in the motif discovery step are post-processed and validated to remove potentially false predictions. This could e.g. involve assessing the statistical significance of the predictions,

comparison of all predicted motifs to remove duplicates or to cluster similar predictions together, or comparing predictions to libraries of known transcription factor motifs.

The four steps are not always strictly separated or executed in a distinct order. Most motif discovery programs perform some form of assessment or validation of their own predictions before reporting the results, and some also do their own pre-processing. The pre- and post-processing steps can optionally be left out, and it is also possible to go back and forth to repeat steps in the pipeline. For example, some methods predict multiple motifs by cycling through the pre-processing and motif discovery steps; after a first motif is predicted, the binding sites for this motif are masked from the sequences to avoid predicting the same motif over again in a second motif discovery step. The "iterated clustering" approach by Abul *et al.* repeatedly cycles through all four of these steps [252]; based on an initial run-through of the first three steps, post-processing is performed to evaluate the predicted motifs and cluster the sequences into more coherent groups based on which motifs they contain. The motif discovery process is then repeated separately on each cluster before all predicted motifs are compared once again and the sequences re-assigned to new clusters.

Information levels in data for motif discovery

Information about sequences in a dataset can be represented at several levels:

1. **Nucleotide level**

   This level represents information about a single position in a sequence, such as the specific nucleotide at that position or the conservation level of that position across different species.

2. **Site level**

   Information related to continuous stretches of nucleotides within a sequence. For instance that a specific region in the sequence encodes a gene or binds a transcription factor.

3. **Inter-site level**

   Information about how sites within a sequence relate to each other. For example that transcription factors binding to two adjacent sites can interact with each other.

4. **Sequence level**

   Information related to a single sequence in the dataset. For example that a sequence is the promoter of a gene, that this gene has a certain expression level in a given tissue or that the protein encoded by that gene has a specific function.

5. **Inter-sequence level**

   Information about how two or more sequences in the dataset relate to each other. For example that two sequences are actually orthologous promoters from two different species or that one sequence contains a distal enhancer which is brought in contact with a promoter represented by a second sequence.

6. **Dataset level**

   Information about the dataset as a whole, for instance that all the sequences are related to genes which show similar expression patterns, are related to the same pathway or are believed to contain binding motifs for the same transcription factors.

Additional data can in principle be used in all steps of the motif discovery pipeline, while contributing information into this process at multiple levels. Therefore many different data sources are relevant as additional data.

Useful data sources

The remainder of this chapter summarizes various forms of information that can potentially be utilized to improve motif discovery. Some of these are features which are able to pinpoint almost exactly the location of a TFBS. Others are indicative of larger regions that could play a role in regulation and they can therefore be used to narrow down the sequence search space. Most of the entries listed have already been proven useful by others, while a few might be considered somewhat speculative. Some of the features, like SNP variation, are not necessarily predictive of regulatory regions by themselves, but since they can influence whether a particular TF will be able to bind to a site or not, it may be relevant to at least take them into consideration when analysing sequences.

Phylogenetic conservation

Since genomic regions that have been evolutionary conserved across large phylogenetic distances are likely to represent important elements, conservation within regulatory regions can be taken as a potential indicator of functional TFBS [253]. Conserved regions can be identified through global alignment of orthologous sequences or by

searching for common motifs in such sequences (local alignment). However, information on nucleotide resolution conservation levels for many organisms is also available as pre-computed tracks made from global alignments of genomes from several species [254]. Such tracks can e.g. be used as positional priors to guide motif discovery programs towards finding conserved sites or it can be used in a post-processing step to filter predicted TFBS that are not conserved.

Nucleosome occupancy

The presence of nucleosomes can obstruct DNA and prevent transcription factors from accessing their target sites. Because of this, there seems to be some correlation between nucleosome positions and location of TFBS [255]. It is believed that the positioning of nucleosomes is, at least to some degree, facilitated by a "nucleosome positioning signal" in the DNA sequence itself consisting of specific patterns of periodically repeating dinucleotides. Many attempts have therefore been made at predicting the locations of nucleosomes computationally based on this signal [228, 256, 257]. However, the specific placement of nucleosomes is also influenced by many other factors and can vary between cell-types [258]. It has been shown, for example, that the presence of the insulator protein and transcription factor CTCF can serve as an anchor for positioning nearby nucleosomes [259]. In addition to computational predictions, data on experimentally mapped nucleosomes is also available for some organisms and cell-types (e.g. [260, 261]).

Histone modifications and chromatin state

The role of all the possible histone modifications and their interplay is not yet fully understood, but at least some modifications have been linked to accessible chromatin and active regulatory regions (e.g. H3K4me1/2/3). Histone modifications can thus potentially be used to determine the location and activity of regulatory regions in different cell-types (Figure 7), and the absence of any modification marks at all can perhaps be interpreted as a sign of nucleosome-free regions that could be readily bound by transcription factors [44].

DNA modifications and CpG-islands

DNA modifications involve covalent attachment of additional chemical groups to the standard nucleotides. The most common such modification (at least in vertebrates) is the addition of a methyl group to the fifth carbon in cytosine, but other modifications are also found to various extents in different organisms, including 5-hydroxy-

methylcytosine, N^6-methyladenosine and 7-methylguanosine. These modifications do not alter the genetic information contained in the sequence itself or the ability of nucleotides to form Watson–Crick base pairs, but they play an important role in the epigenetic regulation of genes, including genomic imprinting and X chromosome inactivation [262]. Methylation of cytosines within a binding site can block some TFs from binding, whereas other TFs will only bind to methylated sites [60, 61]. Tissue-specific genome-wide tracks of DNA methylation are now becoming available thanks to experimental techniques such as bisulfite sequencing [263], Methyl-Seq [264] and MeDIP-chip/seq [265]. In humans, most CpG-sites are methylated except within so-called CpG-islands which are commonly associated with promoter regions of especially house-keeping genes [266]. Consequently, many TSS prediction tools rely on CpG-islands as a feature to identify potential promoter regions [267, 268].

Physical properties of the DNA double helix

The genomic information is encoded in the specific sequence of the four nucleotides in the DNA, and one of the benefits of the DNA double helix as an information carrier is that the molecule itself remains relatively stable regardless of the sequence contents. Hence, any base mutations introduced in the sequence will not affect the overall integrity of the DNA molecule. However, specific sequence patterns can influence the local shape and physical properties of the double helix by altering the angle, distance and forces between adjacent and opposite bases [269]. This will determine properties such as melting temperature, duplex free energy, bending stiffness, etc. which could in turn influence the ability of transcription factors to bind [230, 244]. It has been demonstrated, for example, that AT-rich stretches devoid of TpA-dinucleotides will result in a local narrowing of the minor groove and produce an elevated negative electrostatic potential which could serve as an unspecific binding motif for transcription factors containing positively charged arginines in their binding domain [270]. Physical properties can also be exploited for general promoter prediction [271, 272].

Repeat regions

Repeat regions are abundant in genomes and they come in two main types: *transposable elements* (transposons) and *tandem repeats*. The latter consists of short sequence patterns that are repeated several times directly after each other. The length of the patterns range from 2–6 bp in *short tandem repeats* (microsatellites) up to 60 bp for so-called *minisatellites*. It is believed that such repeats are introduced as a result of slippage or other errors arising during DNA replication. Transposable elements, on the other hand, are sequence segments that are able to move from one place in the genome to another, either by cutting themselves out and reinserting themselves elsewhere or by

first producing an RNA copy which is subsequently reverse transcribed into DNA and inserted in a new place. This second mechanisms will result in additional copies of the transposon and increase the size of the host genome. In humans, for example, a ~300 bp long transposable element called *Alu* exists in more than a million copies and single-handedly comprises about 10% of the genome [273]. All in all, approximately half of the human genome consists of repeats [136]. Repeated elements are also the main reason why some seemingly simple organisms have disproportionately large genomes. Many transposons are believed to be of viral origin, and since new genomic insertions tend to be disruptive to the host, such elements are usually silenced, for instance by packing them in dense heterochromatin.

For a long time, repeat regions were considered to be mostly "junk DNA", and they are often masked from the sequences before performing motif discovery since such repeats could introduce spurious but statistically significant similarities between the sequences which can overshadow the true motifs. However, there are numerous examples demonstrating that some repeat regions do in fact play a role in gene regulation and that transposable elements containing TF binding motifs could be an important mechanism for propagating regulatory modules and for bringing different genes under coordinated control by the same factors [274-276]. It is believed, for example, that transposable elements are responsible for large parts of the c-Myc and p53 regulatory networks [277, 278]. Hence, if one has verified that a transcription factor has bound to a site located within in a certain repeat element in one place, there is a chance that this factor could also be functional in other regions containing the same repeat type.

DNase hypersensitive sites

Hypersensitivity to cleavage by DNase or micrococcal nuclease is a hallmark of open and accessible chromatin, and hypersensitive sites can serve as an indication of active regulatory elements like promoters and enhancers under given conditions. Although detection of DNase HS sites traditionally involved a lot of work, recent high-throughput experimental methods have enabled efficient mapping of hypersensitive sites at genome-wide levels [279, 280]. Regions defined as DNase HS sites typically range from about 100 bp up to a few thousand bp and are characterized by high levels of cleavage throughout the region. To identify individual TFBS one has to analyse the region further with other experimental or computational methods. One such method, called "digital genomic footprinting" aims to find TFBS at single nucleotide resolution by looking for short protected footprints in the cleavage pattern within a DNase HS region [281]. DNase HS sites are condition-specific, but can to some degree generalize to other conditions as well, especially for genes that are expressed in many cell-types [282].

Another method to detect regions with an open chromatin conformation is called FAIRE (formaldehyde-assisted isolation of regulatory elements) [283]. This technique works somewhat similarly to ChIP in that formaldehyde is used to crosslink proteins to DNA. The crosslinked chromatin is sheared into smaller fragments and the protein-bound DNA fragments are separated from unbound DNA using phenol-chloroform extraction. Since histones are the most abundant proteins in chromatin, the organic phase of the separated mixture will be dominated by nucleosomes whereas nucleosome-free DNA will be enriched in the aqueous phase. Open chromatin regions identified by FAIRE show a substantial overlap with DNase HS regions. However, compared to DNase HS, FAIRE seems to be more sensitive to detecting distal regulatory elements and less sensitive to promoter regions [284].

ChIP-seq/chip/PET/exo

ChIP-seq and related techniques that can map protein-DNA binding events on a genome-wide scale were described earlier in the chapter on experimental methods. These techniques provide evidence that a specific transcription factor has bound to a region of the DNA (although the binding itself could have been indirect via another protein). The regions returned by most of these methods tend to be in the order of a few hundred to a few thousand bases, but computational methods can narrow down the regions further to identify the individual binding sites. ChIP data can provide a starting point for selecting sets of sequences to analyse with *de novo* motif discovery methods, and the raw data signal itself can be used as positional priors. ChIP data can also be considered as a form of validation for binding sites for the target factor discovered by motif scanning. Although ChIP data is TF-specific, regions identified by one ChIP experiment can be taken as an indication that there could be binding sites for additional factors nearby as well.

TFBS position relative to genomic features

Many transcription factors display a clear positional preference of binding in relation to e.g. the transcription start site, and some TFs might even have different functions depending on where they bind [285, 286]. Information about the preferred binding location of a given TF can thus be used to filter out potentially spurious motif occurrences that lie outside of this area. When analysing related sequences, such as promoters from co-regulated genes, the tendency of a motif to occur at the same relative position in multiple sequences can also be considered supporting evidence that the motif could be significant for this dataset. Several motif discovery methods already consider information about the positional distribution of binding sites [287-289].

TF–TF interactions and locations of other known TFBS

Over the years, numerous binding sites from various organisms have been individually validated and published by experimental biologists. Many of these sites have also been included in central annotation databases like TRANSFAC [117] and ORegAnno [290]. Although this knowledge is valuable enough in itself, it can also be used to predict additional sites computationally. For example, if a region with a validated binding site from one species is conserved in the genome of a related species, it is likely that an equivalent functional site could also exist in that species. Furthermore, if one has already identified a single binding site, there is a high chance that there could be other functional sites nearby, since binding sites tend to occur in clusters. If, in addition, a nearby predicted TFBS binds a factor which is known to interact with the factor binding to the previously validated site, this fact could be considered supporting evidence for the predicted site [124, 221]. Many databases focusing on transcriptional regulation contain data about TF–TF interactions, and such information can also be found in more general protein-interaction databases (see e.g. [291] for a list).

CAGE data

Cap analysis of gene expression (CAGE) is a high-throughput experimental method than can determine transcription start sites on a genome-wide scale by isolating and sequencing short sequence tags originating from the 5' end of mRNA transcripts [219]. The primary use of CAGE data is to identify the particular start sites, and hence also alternative promoters, that are used by genes under various conditions. The shape of the distribution of mapped tags (sharp versus broad) can also serve as an indication of the general class the gene belongs to (tissue-specific, house-keeping, etc.) [3]. In addition, normalized tag frequencies can be used as a measure of gene expression levels. The original CAGE method worked by cleaving off the first 20 bp from the 5' end of all mRNA transcripts isolated from a cell and concatenating these into longer fragments that were sequenced. The individual 20 bp tag components could later be recovered from these sequences and mapped back to the genome. Newer sequencing technology has allowed the tags to be sequenced directly without having to construct larger concatamers [292]. Since the 5' ends of the tags correspond to the start of the transcripts, the CAGE method allows TSS usage to be determined with high sensitivity.

Gene expression

The expression levels of genes in various tissues or in response to specific treatments can be measured by many different means, including e.g. microarrays or high-throughput sequencing of RNA transcripts. Gene expression data is commonly used to

identify groups of co-expressed genes that could potentially be regulated by the same transcription factors. Most motif discovery programs do not use this information any further, but some methods explicitly incorporate quantitative gene expression levels into their computational model and relate this to the presence of potential binding motifs [249-251]. In addition to the target genes, the expression levels (and hence tissue-specificity) of the transcription factors themselves are also important to consider, since a TF is obviously not able to regulate any genes unless it is actually expressed itself in the same tissues.

Genomic variation

Although all members of the same species share the same basic genome, there are still minor but significant variations in the DNA sequences between individuals. These variations can come in the form of single base mutations (which are called *single nucleotide polymorphisms* or SNPs if they are somewhat common in the population) or as genomic insertions, deletions and duplications (e.g. *copy number variations*). Since transcription factors bind to specific DNA motifs, variation in the sequence within regulatory regions can influence the ability of TFs to bind. It is therefore prudent to keep in mind that the standard reference genomes that are normally employed for bioinformatics analyses will be slightly different from the genomes of cells used in biological experiments. This is especially important to consider when performing analyses in relation to e.g. cancer and other genetically linked disorders which are directly caused by disruptive variations which might not exist in the reference genome (see e.g. [293] for an example of a disorder caused by a SNP within a TFBS). Several public databases exist which provide information about SNPs and other common genetic variations, and some of these even have specific focus on variations affecting regulatory elements [294]. If single nucleotide variations from a large number of individuals are known, this information could potentially also be used to identify "conserved" TFBS with an approach similar to phylogenetic shadowing [295].

Gene ontology

Gene ontology (GO) is an effort to systematize knowledge about genes and gene products by describing each gene with terms from a controlled and hierarchically organized vocabulary [296]. Each GO term has a unique identifier, and many tools have been developed to identify terms that are significantly associated with a set of genes (e.g. [297, 298]). Gene ontology can for instance be used prior to *de novo* motif discovery to cluster sequences into potentially more coherent subgroups based on common terms, or it can provide information about which general class the genes belong to (house-keeping, tissue-specific, etc.) which could in turn dictate how to best

proceed with the analyses. GO can also be used to predict possible roles of TFs or as a form of validation after motifs have been discovered [299]. For example, if one has identified an overrepresented motif in promoters of genes expressed in brain tissue and the TF binding to that motif is described with the GO-term “brain development (GO:0007420)”, this is a good indication that the discovered binding sites could indeed be functional in this context.

3D chromatin structure and nuclear localisation

Bioinformaticians working on motif discovery tend to view the DNA sequence as a simple linear string, but in reality chromosomes fold into complex three-dimensional structures within the cell nucleus. Each chromosome generally occupies its own territory, but some confined movement is still possible, and it has been shown that nuclear localization affects gene activity. A prevailing hypothesis is that transcription only takes place in a few select foci called “transcription factories” and that genes loop out of their chromosomal territories and relocate to these foci upon activation [300, 301]. Such looping is also involved in bringing promoter regions in contact with long-range enhancers [302]. On the other hand, portions of the genome located in proximity to the nuclear lamina tend to be associated with inactive heterochromatin [303]. The three-dimensional structure of chromosomes and interactions between different parts of the genome can be determined with experimental methods such as *Chromosome Conformation Capture* (3C) [304] and extensions thereof [305].

MotifLab: a workbench for data integration

The original project plan to create an improved motif discovery workbench involved developing stand-alone tools for each of the four pipeline steps described in the previous chapter (i.e., selecting sequences, pre-processing, motif discovery and post-processing). These tools could then alternatively be used individually or in combination, similar to the design of other modular analyses workbenches like RSAT [306] and MotifSuite [134]

The initial effort to create a tool for the first step – a program which could group input sequences into more homogeneous clusters based on sequence similarities – was eventually abandoned as it became clear that it did not work satisfactory on real biological sequences but only on artificial data. This software tool was consequently never published.

Since *positional priors* had previously been shown to be a convenient way of representing sequence-related information [169, 229-231], further research focused on a program for the second pre-processing step which could be used to manually construct priors tracks by combining information about various features. The initial idea was just to create a web service tool where the user could assign individual weights to a fixed set of predefined features, and the final priors track would then simply be a linear combination of these features. However, this approach soon seemed too constrained, and it was rather decided on a more flexible solution whereby the user would stepwise build up a priors track by applying various operations to a set of features selected from a larger library. These operations could both be used to manipulate the contents of a single track and to combine information from multiple tracks. The application of the operations could also be limited to certain parts of the sequences by specifying conditions that had to be satisfied. To allow the program to be incorporated as a component in automated analysis pipelines, a simple protocol language was conceived in which the steps required to construct the priors track could be described. This way the program would be able to repeat these steps automatically without further user interaction. The developed software tool was eventually published as PriorsEditor (**paper II**).

Originally, PriorsEditor was only envisioned as a pre-processing tool, and the constructed priors track would have to be output to a file which could subsequently be

provided as input to a motif discovery program in the next step of the pipeline. However, as PriorsEditor was already equipped with a graphical user interface and the ability to visualize feature data tracks, the program would also be suited to visualize the locations of predicted binding sites relative to other genomic features after motif discovery. Since it could also read and write many common data formats, it seemed relatively easy to implement new functionality which would allow PriorsEditor to just pass the sequence data and positional priors track off to another motif discovery program which could be run in the background. When the motif discovery program had finished, PriorsEditor would simply read the results back again and present it visually to the user. As this would add tremendous value to the tool, the ability to perform motif discovery and also motif scanning in this way was included already in the first published version of PriorsEditor.

It was soon realized that the whole motif discovery pipeline could be incorporated into PriorsEditor by exploiting the functionality provided by operations and protocols to control the execution of individual data processing steps. The processing itself could either be performed by internal operations or in collaboration with external programs for more advanced tasks.

The three feature data types included in PriorsEditor – *DNA Sequence Datasets*, *Numeric Datasets* and *Region Datasets* – could already represent information related to the first two information levels described earlier, and these were later supplemented with additional types that could hold information about higher levels. For example, a data type to model regulatory *Modules* (composite motifs) was introduced to represent this specific type of inter-site relationship (level 3), and the *Motif* data type was extended to allow for annotation of known interaction partners. Operations were also included to support *de novo* module discovery and module scanning. Information related to individual sequences (level 4) could be represented with the new *Map* data type or more generally with *Text Variables*, whereas inter-sequence information (level 5) could to some extent be modelled with *Collections* and *Partitions* which group together related sequences.

Other types of functionality were also added to enhance PriorsEditor. One of the primary new inclusions was the ability to use machine learning to train classifier objects (*Priors Generators*) which could generate positional priors automatically based on a selected set of features. Several different *Analyses* were also introduced for the post-processing step to validate predicted motifs and binding sites or to calculate various statistics from the data. The results from these analyses could be presented as nicely formatted reports with tables and figures. Finally, a few convenient tools were added which could take advantage of the graphical user interface for interactive data exploration.

In the end, what had originally started out as a small pre-processing tool with narrow focus had grown to become a general and full-featured workbench for motif discovery. To reflect this fact, the PriorsEditor name was eventually abandoned and the program renamed as MotifLab (**paper III**).

Practical examples

A guiding principle behind MotifLab is that it should be easy to do the standard tasks related to motif discovery and scanning – such as obtaining sequences, obtaining motifs and running a motif discovery program on this data – but it should also be relatively easy to perform more advanced data processing and analyses which would normally require writing custom scripts in some general programming language.

This section includes a selection of protocol examples demonstrating how MotifLab can be used to solve a few practical problems related to motif discovery. It should be noted that it is not required for users to know the syntax of the protocol language in order to do the analyses, as the steps can be performed one by one by simply selecting the operations to execute from menus in MotifLab's graphical user interface.

Each line of a protocol contains a single command, and the first word of the command is the name of the operation to perform[1] (see Table A1 in the appendix for a list of available operations). If the results returned by the operation should be assigned to a new data object, the name of that data object is written before the command and is followed by an assignment operator in the form of an equals sign.

The protocols here are coloured according to the same default scheme which is used by MotifLab's own internal protocol editor: operations are in red, general data types are in orange whereas names of specific data objects are in blue, names of analyses and also general data formats (for input and output) are in orange, names of external programs are in green, double quoted text strings are also in green and numeric constants are coloured in pink.

The first protocol example demonstrates how motif scanning can be performed with the help of an external program. When this protocol is executed, MotifLab will ask the user to define which sequence regions to perform the analyses on (unless they are defined already). MotifLab will then obtain the DNA sequence data for these regions, load a

[1] MotifLab requires that the full command is written on a single line, but because of limitations on page width in this document, the longest commands in the protocol examples have here been divided across multiple lines. Each new command line is marked with a line number.

collection of motifs from the TRANSFAC Public database and perform motif scanning using the program SimpleScanner. The predicted binding sites (with at least 90% match to the motifs) are returned in a new track called “TFBS”.

Protocol 1: Motif scanning

```
1  DNA = new DNA Sequence Dataset(DataTrack:DNA)
2  Motifs = new Motif Collection(Collection:TRANSFAC Public)
3  TFBS = motifScanning on DNA with SimpleScanner {
       Motif Collection = Motifs,
       Threshold type = "Percentage",
       Threshold = 90
     }
```

The “new” operation in the first line will create a new DNA Sequence Dataset object based on data obtained from a track called “DNA”. Data sources for this track and several other commonly used tracks are preconfigured in MotifLab, and additional tracks can easily be added from e.g. UCSC Genome Browser [307] or DAS servers [308]. Each track can have several separate data sources for different organisms and genome builds, so even if your sequences come from different genomes, MotifLab will automatically use the correct data source for each individual sequence.

One data track which is available for many species is “Conservation”, and the next example demonstrates how this information can be used to perform a simple form of phylogenetic footprinting by filtering out predicted sites in the TFBS track that are not conserved.

Protocol 2: “Phylogenetic footprinting”

```
1  Conservation = new Numeric Dataset(DataTrack:Conservation)
2  TFBSconserved= filter TFBS where region's average Conservation < 0.2
```

Here, a *condition* is imposed on the “filter” operation in the second line (introduced by the keyword “where”), so that the operation is only applied to TFBS where the average value of the Conservation track within the TFBS region is less than some specified cutoff. Some motif models, especially from the TRANSFAC database, have a core region with high information content which is flanked by more variable positions, so only a fraction of the positions in the motif are actually under evolutionary pressure. A more advanced condition that takes this into consideration is “`where region's weighted average Conservation < 0.2`”, which will weigh the conservation value in each position with the information content of the motif in that position before taking the average.

The filter operation is one of the most useful operations, since it can be employed in the post-processing step to remove potentially false binding site predictions. The next protocol uses this operation to remove TFBS that are outside of DNase hypersensitive regions (and are thus likely to reside within more condensed chromatin).

Protocol 3: Filter TFBS predictions outside of DNase HS regions

```
1  DNaseHS = new Region Dataset(DataTrack:DNaseHS_peaks)
2  TFBS_within_DNaseHS = filter TFBS where not region overlaps DNaseHS
```

It is also possible to filter out predicted sites that are not supported by ChIP-seq data. If the ChIP-seq track contains regions for several different TFs, the challenge is to link ChIP-seq peak regions for each individual TF (which can have type names like "CEBPB" and "c-Fos") to the corresponding binding sites for the TF (which normally have types named after motif identifiers). In the protocol below, this issue is resolved by renaming the ChIP-seq regions so that their new type names contain lists of identifiers for the corresponding motifs. This is accomplished with the "type-replace" transform operation which relies on a Text Variable containing the replacements to be made in ‹KEY→VALUE› format. In the subsequent filtering step, TFBS that do not overlap with ChIP-seq peak regions *of a matching type* is removed.

Protocol 4: Filter TFBS predictions not supported by ChIP-seq peaks

```
1  ChIP_seq_peak = new Region Dataset(DataTrack:TFBS_ChIP-Seq)
2  ChIP_seq_map = new Text Variable(
       "CEBPB=>M00109 M00117",
       "c-Fos=>M00517 M00924 M00926",
       "c-Jun=>M00517",
        ...
     )
3  transform ChIP_seq_peak with type-replace(ChIP_seq_map)
4  TFBS_supported_by_ChIP_seq = filter TFBS where not region overlaps
                                      type-matching ChIP_seq_peak
```

If we know that some TFs will only be functional if they bind within a certain distance to the TSS, we can used this information to filter out predicted motif occurrences outside of the preferred locations for each individual TF.

In Protocol 5 below, the preferred binding regions will be defined with the help of two *Numeric Map*s. A Numeric Map is a data object which can associate each motif, module or sequence with an individual numeric value. Here we use one Motif Numeric Map to specify the minimum preferred distance to the TSS for each motif and a second map to specify the maximum distance. The first two lines in the protocol define the preferred

binding region of MOTIF1 as being between 40 to 90 bp upstream of the TSS whereas the preferred region of MOTIF2 is between 20 to 50 bp (other motifs are allowed to reside up to 1 Kbp away in either direction). In the third line, a new numeric track called "position" is set up where the value in each position of the track reflects its distance from the TSS.

One nice thing about the *Map* data type is that whenever a Map is used as an argument for an operation, the specific argument value will depend on the context when the operation executes. Here, for example, we have two Motif Numeric Maps as arguments to the filter operation in the fourth line. For every region this operation is applied to, the operation will first check which motif the region corresponds to and then use the value for that motif from the map. To satisfy the condition of the filter operation, the value of the position track at the start of the region (which equals the distance from the TSS) must be within the range set up by the two maps, and this range will thus depend on the specific motif in question. If we had used *Sequence Numeric Maps* as arguments instead of *Motif Numeric Maps*, the operation would have first determined which sequence it was applied to and then looked up a value for that sequence in the map. In that case, the motifs would have to be within a different region with respect to each sequence, but each region would then apply to all motifs regardless of their type.

Protocol 5: Filtering motifs outside of the TF's preferred binding location

```
1  minDistance = new Motif Numeric Map(MOTIF1=40, MOTIF2=20, -1000)
2  maxDistance = new Motif Numeric Map(MOTIF1=90, MOTIF2=50,  1000)

3  position = distance upstream from transcription start site
4  TFBS_location = filter TFBS where not region's startValue position
                                      in minDistance to maxDistance
```

Since MotifLab allows motifs to be annotated with information about known interaction partners, this information can be used to filter potentially spurious motif occurrences that do not have binding sites for any known partners nearby, as shown in Protocol 6.

Protocol 6: Filter TFBS predictions that do not have another TFBS for a known interaction partner nearby

```
1  TFBS_interacting = filter TFBS where not region's distance to
                      any interaction partner in 0 to 16
```

The other TFBS must here lie between 0 to 16 bp away. The reason for writing the condition this way rather than using the perhaps more intuitive command "filter TFBS where region's distance to closest interaction partner > 16", is to force the motifs to be non-overlapping (as overlapping sites by definition have negative distances).

We could have used Motif Numeric Maps instead of constant numbers to define the distance range, and the range would then have been determined by the motif of the target site; e.g. MOTIF1 could have required that interaction partners bind within 5 to 8 bp, whereas partners for MOTIF2 should bind within 9 to 12 bp. It is not possible to define individual distances for specific *pairs* of motifs with this approach, for instance that the distance between MOTIF1 and MOTIF2 should be within 5 to 8 bp, but if MOTIF1 partners up with MOTIF3 the distance should be between 8 to 10 bp. However, such constraints can be specified if we instead use the *Module* data type to model and search for composite motifs.

The last example of TFBS filtering demonstrates how more complex conditions can be defined by combining multiple individual conditions with Boolean operators (and/or). Such *compound conditions* can be nested to arbitrary levels by grouping conditions together with parentheses. Protocol 7 employs a compound condition to filter out binding sites that overlap with methylated CpG-sites, but only if it is known that the corresponding TF is unable to bind to methylated sites. In this particular example, a collection containing motifs for TFs known to be blocked by methylation is defined manually, but it would also be possible to include this information as a user-defined property of Motif objects and derive the collection automatically from this property.

Protocol 7: Filter TFBS that are inactive when methylated

```
1  Blocked_by_methylation = new Motif Collection(MOTIF1, MOTIF2)
2  meCpG = new Region Dataset(DataTrack:CpG_methylation_K562)

3  filter TFBS where region's type in Blocked_by_methylation
               and region overlaps meCpG
```

One issue which has to be resolved when analysing promoters regions of genes is to decide on the size of the sequence region to analyse relative to the TSS. Often, the size is chosen rather arbitrarily, but it is common to at least limit the region so that it does not overlap with any upstream genes or downstream coding regions. The following protocol will define such promoter regions by first creating a 1 bp region at the location of the TSS in each sequence and then extending these regions in both directions until they encounter either an upstream gene or a downstream CDS. If there are no annotated upstream genes or downstream CDS for a sequence, the promoter region will simply extend to the edge of the sequence.

Protocol 8: Delimiting the span of promoter regions

```
1  Genes = new Region Dataset(DataTrack:EnsemblGenes)
2  CDS   = new Region Dataset(DataTrack:CCDS)
3  Distance_to_TSS = distance upstream from transcription start site
4  Promoter = convert Distance_to_TSS to region where Distance_to_TSS=0
5  extend Promoter upstream until inside Genes,
                   downstream until inside CDS
```

A more sensible way to determine the span of the promoter regions would be to look at histone modifications in the vicinity of the TSS. The "chromatin state" track (see Figure 7) divides the whole genome into consecutive 200 bp regions and assigns each region a state based on information about different histone modifications in the region.

The example in Protocol 9 defines a primary regulatory region associated with each sequence by identifying segments of continuous 200 bp regions labelled as promoters (either *active* or *weak* but not *poised*) or enhancers (*strong* or *weak*) and then picks out one of these segments which lies closest to the TSS. The protocol first obtains chromatin state data for a specific cell type (here K562) and removes all 200 bp regions not associated with promoters/enhancers. Next, the remaining regions that lie directly next to each other are merged into larger segments, and each segment is assigned a score based on its minimum distance to the TSS. Finally, the protocol determines which such score is the smallest within each sequence and removes all segments that do not have this particular score. (The "statistic" operation in line 7 returns a Sequence Numeric Map which is used as an argument in the condition on line 8).

Protocol 9: Determining the span of promoter regions based on chromatin state

```
1  ChromatinState = new Region Dataset(DataTrack:ChromatinState_K562)
2  active_states = new Text Variable(
       "1 Active Promoter",
       "2 Weak Promoter",
       "4 Strong Enhancer",
       "5 Strong Enhancer",
       "6 Weak Enhancer",
       "7 Weak Enhancer"
   )
3  RegulatoryRegions = filter ChromatinState where not region's type
                                                 in set active_states
4  merge RegulatoryRegions closer than 1
5  Distance_to_TSS = distance upstream from transcription start site
6  set RegulatoryRegions[score] to minimum Distance_to_TSS
7  minScore = statistic "minimum score" in RegulatoryRegions
8  Promoter = filter RegulatoryRegions where region's score > minScore
```

Creating and using positional priors to guide motif discovery is one of the primary applications of MotifLab. Many data tracks, including e.g. Conservation or ChIP-seq tracks, can be used directly as positional priors since higher values in these tracks correlate well with the presence of functional binding sites (even though the tracks do not really contain probability values in a strict statistical sense).

Lines 1–4 in Protocol 10 loads a track containing the raw ChIP-seq signal data for the transcription factor GATA-1 in the K562 cell line and performs some normalization before using the track to guide the *de novo* motif discovery program ChIPMunk.

Protocol 10: Creating and using *positional priors* to guide motif discovery

```
1  ChIP_Seq = new Numeric Dataset(DataTrack:ChIPseqSignalK562bGata1)
2  normalize ChIP_Seq from range [0,sequence.max]
                       to range [0.001,sequence.max]
3  normalize ChIP_Seq sequence sum to one
4  [TFBS,Motifs] = motifDiscovery on DNA with ChIPMunk {
          Model = "Peak",
          Min motif size = 7,
          Max motif size = 25,
          Occurrences = "OOPS",
          Peaks = ChIP_Seq
     } motif-prefix="ChIPMunk"

5  TFBS_vicinity = extend TFBS by 60
6  convert TFBS_vicinity to numeric with value = 1.0
7  Conservation = new Numeric Dataset(DataTrack:Conservation)
8  increase TFBS_vicinity by Conservation where TFBS_vicinity > 0
9  set TFBS_vicinity to 0 where inside TFBS
10 normalize TFBS_vicinity from range [0,sequence.max]
                            to range [0.001,sequence.max]
11 normalize TFBS_vicinity sequence sum to one
12 mask DNA with "N" where inside TFBS
13 [TFBS_partner,Motifs_partner] = motifDiscovery on DNA with ChIPMunk {
          Model = "Peak",
          Min motif size = 7,
          Max motif size = 25,
          Occurrences = "OOPS",
          Peaks = TFBS_vicinity
     } motif-prefix="ChIPMunk"
```

The rest of the protocol (from line 5 onwards) demonstrates how MotifLab's operations can be used to manually create more elaborate priors tracks with specific search focus. Assuming that ChIPMunk was able to identify binding sites for the GATA-1 motif in the first part of the protocol, the rest of the protocol will set up a new priors track which focuses on discovering additional motifs in the vicinity of the GATA-1 sites. These motifs could potentially be bound by factors interacting with GATA-1.

The protocol first defines a “neighbourhood” region around the GATA-1 sites (in the TFBS track) by extending these TFBS by 60 bp in both directions (line 5). Next, it converts this Region Dataset into a Numeric Dataset, since positional priors must be represented by this data type. Other types of information can also be included to fine-tune the priors in these neighbourhoods. E.g. lines 7–8 adjust the priors track by assigning higher values to more conserved positions. To make sure that the motif discovery program does not simply rediscover the GATA-1 motif all over again, we set the value of the positional priors track to zero within the TFBS regions discovered earlier (and to make absolutely certain we also mask the DNA sequence itself within these sites).

The final protocol example in this chapter demonstrates how the “analyze” operation can be used to identify motifs that are significantly overrepresented in a set of sequences (see also Table A2 in the appendix for other types of analyses this operation can perform).

After performing motif scanning in the same way as described earlier (lines 1–4), the protocol loads a third-order background model based on DNA sequence composition in human promoter sequences (“EPD_human_3”) and uses this model to generate a new artificial DNA track to use as control sequences. (The “new” operation in line 6 does not actually create any new sequences; it just creates a new track for the existing sequence regions). The protocol then performs motif scanning in this artificial sequence track using the same parameter settings as before and derives a new Motif Numeric Map based on the occurrence frequency of each motif in this track.

The map object containing the expected frequencies is provided as an argument to the “count motif occurrences” analysis which counts the number of times each motif type occurs in the original TFBS track and compares these counts to the expected motif frequencies estimated from the control sequences.

The motifs that are statistically overrepresented (at an initial significance threshold of 0.05 which is subsequently Bonferroni-corrected) are extracted from the resulting Analysis object as a new Motif Collection, and the analysis results for these motifs are presented in a table which is output to a HTML-document.

Protocol 11: Identifying overrepresented motifs in a set of sequences

```
1  DNA = new DNA Sequence Dataset(DataTrack:DNA)
2  Motifs = new Motif Collection(Collection:TRANSFAC Public)
3  cutoff = new Numeric Variable(90)
4  TFBS = motifScanning on DNA with SimpleScanner {
       Motif Collection = Motifs,
       Threshold type = "Percentage",
       Threshold = cutoff
     }
5  BGmodel = new Background Model(Model:EPD_human_3)
6  DNA_control = new DNA Sequence Dataset(BGmodel)
7  TFBS_control = motifScanning on DNA_control with SimpleScanner {
       Motif Collection = Motifs,
       Threshold type = "Percentage",
       Threshold = cutoff
     }
8  ExpectedFrequencies = new Motif Numeric Map(Track:TFBS_control,
                                               Property=Frequency
                                               )
9  CountAnalysis = analyze count motif occurrences {
       Motif track = TFBS,
       Motifs = Motifs,
       Background frequencies = ExpectedFrequencies,
       Significance threshold = 0.05,
       Bonferroni correction = "All motifs"
    }
10 Overrepresented_Motifs = extract "overrepresented"
                            from CountAnalysis as Motif Collection

11 output CountAnalysis in HTML format {
       Include = Overrepresented_Motifs,
       Sort by = "p-value",
       Logos = "New images"
    }
```

The operations provided by MotifLab offer general and flexible means to process and analyse datasets, and it is often possible to solve the same problem in several different ways. The few protocol examples presented here merely scratch the surface of MotifLab's capabilities. Additional examples demonstrating more advanced analyses and applications on real biological data are included in **paper III**.

Conclusions and future work

Traditional motif discovery approaches have been limited in their ability to predict functional binding sites for transcription factors because they only rely on information in the DNA sequence itself and fail to take into consideration the biological state of the cell. However, a lot of additional data is now available which can be integrated into the motif discovery process to improve the performance, such as information about phylogenetic conservation, nucleosome occupancy, DNase hypersensitive sites, epigenetic features, gene expression and TF–TF interactions, to name just a few. To this end, a new motif discovery workbench – called MotifLab – was developed which aims to make it easier for researchers to take advantage of different types of data in combination with existing or novel motif discovery tools.

Users of MotifLab are not limited to exploiting data related to a fixed or predefined set of features; nor does the program require that users themselves obtain all the data for the sequence regions they want to analyse. Rather, data sources for feature tracks from various organisms can easily be configured so that MotifLab will be able to download the requested data automatically. Data is modelled by a few general data types that can represent information on many different levels, and this means that MotifLab should also be able to integrate new types of relevant information that might be introduced in the future. In addition, the wide range of operations provided by MotifLab allows users to manipulate the data to suit their own needs.

Many aspects of MotifLab can still be improved, however. In particular, better support for Gene Ontology has high priority, as has tools which would make it easier for users to configure their own external programs for use with MotifLab. Another important issue that warrants further focus is the ability to determine, preferably automatically, exactly which alternative promoter and distal enhancer regions that are involved in regulating a gene under specific conditions. This will require integration of several types of data, including e.g. CAGE to determine the active TSS, DNase HS or epigenetic data to determine the span of the regulatory regions and chromatin conformation data to determine which regions are interacting.

In the past few years, high-throughput experimental methods have been introduced that are able to efficiently map binding events for transcription factors on a genome-wide scale, and this has led some to question the continued relevance of computational motif discovery tools. With a couple of thousand transcription factors estimated in the human genome alone, it will still take time before binding sites for all of these are identified, however. To complicate matters, many of these factors are currently unknown or are

difficult to map with existing technologies. In addition, transcription factors can bind different sites in different cell-types or under different conditions (both normal and diseased), and it is unlikely that binding site profiles for every transcription factor will be determined experimentally for all possible conditions any time soon. Rather, there might be binding profiles available for a few conditions and binding under other conditions will have to be inferred computationally by integrating information about e.g. DNase hypersensitivity and chromatin states under these conditions. In this respect, computational tools like MotifLab will perhaps prove even more valuable in the future.

References

1. Watson JD, Crick FH: **Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid**. *Nature* 1953, **171**(4356):737-738.

2. Pertea M, Salzberg SL: **Between a chicken and a grape: estimating the number of human genes**. *Genome biology* 2010, **11**(5):206.

3. Carninci P, Sandelin A, Lenhard B, Katayama S, Shimokawa K, Ponjavic J, Semple CA, Taylor MS, Engstrom PG, Frith MC *et al*: **Genome-wide analysis of mammalian promoter architecture and evolution**. *Nature genetics* 2006, **38**(6):626-635.

4. Chapman SC, Schubert FR, Schoenwolf GC, Lumsden A: **Analysis of spatial and temporal gene expression patterns in blastula and gastrula stage chick embryos**. *Developmental biology* 2002, **245**(1):187-199.

5. Pribnow D: **Nucleotide sequence of an RNA polymerase binding site at an early T7 promoter**. *Proceedings of the National Academy of Sciences of the United States of America* 1975, **72**(3):784-788.

6. Hawley DK, McClure WR: **Compilation and analysis of Escherichia coli promoter DNA sequences**. *Nucleic acids research* 1983, **11**(8):2237-2255.

7. Smale ST, Kadonaga JT: **The RNA polymerase II core promoter**. *Annual review of biochemistry* 2003, **72**:449-479.

8. Fuda NJ, Ardehali MB, Lis JT: **Defining mechanisms that regulate RNA polymerase II transcription in vivo**. *Nature* 2009, **461**(7261):186-192.

9. Valen E, Sandelin A: **Genomic and chromatin signals underlying transcription start-site selection**. *Trends in genetics : TIG* 2011, **27**(11):475-485.

10. Robinson-Rechavi M, Escriva Garcia H, Laudet V: **The nuclear receptor superfamily**. *Journal of cell science* 2003, **116**(Pt 4):585-586.

11. Shamovsky I, Nudler E: **New insights into the mechanism of heat shock response activation**. *Cellular and molecular life sciences : CMLS* 2008, **65**(6):855-861.

12. Barberis A, Petrascheck M: **Transcription Activation in Eukaryotic Cells**. In: *eLS.* John Wiley & Sons, Ltd; 2003.

13. Brannan K, Bentley DL: **Control of Transcriptional Elongation by RNA Polymerase II: A Retrospective**. *Genetics research international* 2012, **2012**:170173.

14. Thiel G, Lietz M, Hohl M: **How mammalian transcriptional repressors work**. *European journal of biochemistry* 2004, **271**(14):2855-2862.

15. Babu MM, Luscombe NM, Aravind L, Gerstein M, Teichmann SA: **Structure and evolution of transcriptional regulatory networks**. *Current opinion in structural biology* 2004, **14**(3):283-291.

16. Stegmaier P, Kel AE, Wingender E: **Systematic DNA-binding domain classification of transcription factors**. *Genome informatics International Conference on Genome Informatics* 2004, **15**(2):276-286.

17. Taatjes DJ: **The human Mediator complex: a versatile, genome-wide regulator of transcription**. *Trends in biochemical sciences* 2010, **35**(6):315-322.

18. Sinha S, Adler AS, Field Y, Chang HY, Segal E: **Systematic functional characterization of cis-regulatory motifs in human core promoters**. *Genome research* 2008, **18**(3):477-488.

19. Fairall L, Schwabe JWR: **DNA binding by transcription factors**. In: *Transcription Factors.* Edited by Locker J. Oxford: BIOS Scientific Publishers; 2000: 65-84.

20. Badis G, Berger MF, Philippakis AA, Talukder S, Gehrke AR, Jaeger SA, Chan ET, Metzler G, Vedenko A, Chen X *et al*: **Diversity and complexity in DNA recognition by transcription factors**. *Science* 2009, **324**(5935):1720-1723.

21. Luscombe NM, Austin SE, Berman HM, Thornton JM: **An overview of the structures of protein-DNA complexes**. *Genome biology* 2000, **1**(1):REVIEWS001.

22. Bewley CA, Gronenborn AM, Clore GM: **Minor groove-binding architectural proteins: structure, function, and DNA recognition**. *Annual review of biophysics and biomolecular structure* 1998, **27**:105-131.

23. Rohs R, Jin X, West SM, Joshi R, Honig B, Mann RS: **Origins of specificity in protein-DNA recognition**. *Annual review of biochemistry* 2010, **79**:233-269.

24. Maston GA, Evans SK, Green MR: **Transcriptional regulatory elements in the human genome**. *Annual review of genomics and human genetics* 2006, **7**:29-59.

25. Saiz L, Vilar JM: **DNA looping: the consequences and its control**. *Current opinion in structural biology* 2006, **16**(3):344-350.

26. Bell AC, West AG, Felsenfeld G: **The protein CTCF is required for the enhancer blocking activity of vertebrate insulators**. *Cell* 1999, **98**(3):387-396.

27. Valenzuela L, Kamakaka RT: **Chromatin insulators**. *Annual review of genetics* 2006, **40**:107-138.

28. Wallace JA, Felsenfeld G: **We gather together: insulators and genome organization**. *Current opinion in genetics & development* 2007, **17**(5):400-407.

29. Arnosti DN, Kulkarni MM: **Transcriptional enhancers: Intelligent enhanceosomes or flexible billboards?** *Journal of cellular biochemistry* 2005, **94**(5):890-898.

30. Spitz F, Furlong EE: **Transcription factors: from enhancer binding to developmental control**. *Nature reviews genetics* 2012, **13**(9):613-626.

31. Wasserman WW, Fickett JW: **Identification of regulatory regions which confer muscle-specific gene expression**. *Journal of molecular biology* 1998, **278**(1):167-181.

32. Krivan W, Wasserman WW: **A predictive model for regulatory sequences directing liver-specific transcription**. *Genome research* 2001, **11**(9):1559-1566.

33. Jarosz DF, Taipale M, Lindquist S: **Protein homeostasis and the phenotypic manifestation of genetic diversity: principles and mechanisms**. *Annual review of genetics* 2010, **44**:189-216.

34. Kristiansson E, Thorsen M, Tamas MJ, Nerman O: **Evolutionary forces act on promoter length: identification of enriched cis-regulatory elements**. *Molecular biology and evolution* 2009, **26**(6):1299-1307.

35. Levine M, Tjian R: **Transcription regulation and animal diversity**. *Nature* 2003, **424**(6945):147-151.

36. Dillon N: **Heterochromatin structure and function**. *Biology of the cell* 2004, **96**(8):631-637.

37. Filion GJ, van Bemmel JG, Braunschweig U, Talhout W, Kind J, Ward LD, Brugman W, de Castro IJ, Kerkhoven RM, Bussemaker HJ *et al*: **Systematic protein location mapping reveals five principal chromatin types in Drosophila cells**. *Cell* 2010, **143**(2):212-224.

38. Felsenfeld G, Groudine M: **Controlling the double helix**. *Nature* 2003, **421**(6921):448-453.

39. Kumar SV, Wigge PA: **H2A.Z-containing nucleosomes mediate the thermosensory response in Arabidopsis**. *Cell* 2010, **140**(1):136-147.

40. Raisner RM, Hartley PD, Meneghini MD, Bao MZ, Liu CL, Schreiber SL, Rando OJ, Madhani HD: **Histone variant H2A.Z marks the 5' ends of both active and inactive genes in euchromatin**. *Cell* 2005, **123**(2):233-248.

41. Bell O, Tiwari VK, Thoma NH, Schubeler D: **Determinants and dynamics of genome accessibility**. *Nature reviews genetics* 2011, **12**(8):554-564.

42. Strahl BD, Allis CD: **The language of covalent histone modifications**. *Nature* 2000, **403**(6765):41-45.

43. Ucar D, Hu Q, Tan K: **Combinatorial chromatin modification patterns in the human genome revealed by subspace clustering**. *Nucleic acids research* 2011, **39**(10):4063-4075.

44. Barski A, Cuddapah S, Cui K, Roh TY, Schones DE, Wang Z, Wei G, Chepelev I, Zhao K: **High-resolution profiling of histone methylations in the human genome**. *Cell* 2007, **129**(4):823-837.

45. Rosenfeld JA, Wang Z, Schones DE, Zhao K, DeSalle R, Zhang MQ: **Determination of enriched histone modifications in non-genic portions of the human genome**. *BMC genomics* 2009, **10**:143.

46. Jenuwein T, Allis CD: **Translating the histone code**. *Science* 2001, **293**(5532):1074-1080.

47. Li B, Carey M, Workman JL: **The role of chromatin during transcription**. *Cell* 2007, **128**(4):707-719.

48. Barth TK, Imhof A: **Fast signals and slow marks: the dynamics of histone modifications**. *Trends in biochemical sciences* 2010, **35**(11):618-626.

49. Wang Z, Zang C, Rosenfeld JA, Schones DE, Barski A, Cuddapah S, Cui K, Roh TY, Peng W, Zhang MQ *et al*: **Combinatorial patterns of histone acetylations and methylations in the human genome**. *Nature genetics* 2008, **40**(7):897-903.

50. Heintzman ND, Stuart RK, Hon G, Fu Y, Ching CW, Hawkins RD, Barrera LO, Van Calcar S, Qu C, Ching KA *et al*: **Distinct and predictive chromatin signatures of transcriptional promoters and enhancers in the human genome**. *Nature genetics* 2007, **39**(3):311-318.

51. Bernstein BE, Mikkelsen TS, Xie X, Kamal M, Huebert DJ, Cuff J, Fry B, Meissner A, Wernig M, Plath K *et al*: **A bivalent chromatin structure marks key developmental genes in embryonic stem cells**. *Cell* 2006, **125**(2):315-326.

52. Ernst J, Kheradpour P, Mikkelsen TS, Shoresh N, Ward LD, Epstein CB, Zhang X, Wang L, Issner R, Coyne M *et al*: **Mapping and analysis of chromatin state dynamics in nine human cell types**. *Nature* 2011, **473**(7345):43-49.

53. Cirillo LA, Lin FR, Cuesta I, Friedman D, Jarnik M, Zaret KS: **Opening of compacted chromatin by early developmental transcription factors HNF3 (FoxA) and GATA-4**. *Molecular cell* 2002, **9**(2):279-289.

54. Zaret KS, Carroll JS: **Pioneer transcription factors: establishing competence for gene expression**. *Genes & development* 2011, **25**(21):2227-2241.

55. Bird AP: **DNA methylation and the frequency of CpG in animal DNA**. *Nucleic acids research* 1980, **8**(7):1499-1504.

56. Ehrlich M, Gama-Sosa MA, Huang LH, Midgett RM, Kuo KC, McCune RA, Gehrke C: **Amount and distribution of 5-methylcytosine in human DNA from different types of tissues of cells**. *Nucleic acids research* 1982, **10**(8):2709-2721.

57. Gardiner-Garden M, Frommer M: **CpG islands in vertebrate genomes**. *Journal of molecular biology* 1987, **196**(2):261-282.

58. Takai D, Jones PA: **Comprehensive analysis of CpG islands in human chromosomes 21 and 22**. *Proceedings of the National Academy of Sciences of the United States of America* 2002, **99**(6):3740-3745.

59. Siegfried Z, Eden S, Mendelsohn M, Feng X, Tsuberi BZ, Cedar H: **DNA methylation represses transcription in vivo**. *Nature genetics* 1999, **22**(2):203-206.

60. Nan X, Ng HH, Johnson CA, Laherty CD, Turner BM, Eisenman RN, Bird A: **Transcriptional repression by the methyl-CpG-binding protein MeCP2 involves a histone deacetylase complex**. *Nature* 1998, **393**(6683):386-389.

61. Prendergast GC, Ziff EB: **Methylation-sensitive sequence-specific DNA binding by the c-Myc basic region**. *Science* 1991, **251**(4990):186-189.

62. Bell AC, Felsenfeld G: **Methylation of a CTCF-dependent boundary controls imprinted expression of the Igf2 gene**. *Nature* 2000, **405**(6785):482-485.

63. Mikkelsen TS, Ku M, Jaffe DB, Issac B, Lieberman E, Giannoukos G, Alvarez P, Brockman W, Kim TK, Koche RP *et al*: **Genome-wide maps of chromatin state in pluripotent and lineage-committed cells**. *Nature* 2007, **448**(7153):553-560.

64. Heintzman ND, Hon GC, Hawkins RD, Kheradpour P, Stark A, Harp LF, Ye Z, Lee LK, Stuart RK, Ching CW *et al*: **Histone modifications at human enhancers reflect global cell-type-specific gene expression**. *Nature* 2009, **459**(7243):108-112.

65. Lenhard B, Sandelin A, Carninci P: **Metazoan promoters: emerging characteristics and insights into transcriptional regulation**. *Nature reviews genetics* 2012, **13**(4):233-245.

66. Muse GW, Gilchrist DA, Nechaev S, Shah R, Parker JS, Grissom SF, Zeitlinger J, Adelman K: **RNA polymerase is poised for activation across the genome**. *Nature genetics* 2007, **39**(12):1507-1511.

67. Bartel DP: **MicroRNAs: target recognition and regulatory functions**. *Cell* 2009, **136**(2):215-233.

68. Carthew RW, Sontheimer EJ: **Origins and Mechanisms of miRNAs and siRNAs**. *Cell* 2009, **136**(4):642-655.

69. Johnson LN: **The regulation of protein phosphorylation**. *Biochemical Society transactions* 2009, **37**(Pt 4):627-641.

70. de Lichtenberg U, Jensen LJ, Brunak S, Bork P: **Dynamic complex formation during the yeast cell cycle**. *Science* 2005, **307**(5710):724-727.

71. Tanaka K, Chiba T: **The proteasome: a protein-destroying machine**. *Genes to cells : devoted to molecular & cellular mechanisms* 1998, **3**(8):499-510.

72. Garner MM, Revzin A: **A gel electrophoresis method for quantifying the binding of proteins to specific DNA regions: application to components of the Escherichia coli lactose operon regulatory system**. *Nucleic acids research* 1981, **9**(13):3047-3060.

73. Kristie TM, Roizman B: **Alpha 4, the major regulatory protein of herpes simplex virus type 1, is stably and specifically associated with promoter-regulatory domains of alpha genes and of selected other viral genes**. *Proceedings of the National Academy of Sciences of the United States of America* 1986, **83**(10):3218-3222.

74. Galas DJ, Schmitz A: **DNAse footprinting: a simple method for the detection of protein-DNA binding specificity**. *Nucleic acids research* 1978, **5**(9):3157-3170.

75. Maxam AM, Gilbert W: **A new method for sequencing DNA**. *Proceedings of the National Academy of Sciences of the United States of America* 1977, **74**(2):560-564.

76. Johnson DS, Mortazavi A, Myers RM, Wold B: **Genome-wide mapping of in vivo protein-DNA interactions**. *Science* 2007, **316**(5830):1497-1502.

77. Kim TH, Abdullaev ZK, Smith AD, Ching KA, Loukinov DI, Green RD, Zhang MQ, Lobanenkov VV, Ren B: **Analysis of the vertebrate insulator protein CTCF-binding sites in the human genome**. *Cell* 2007, **128**(6):1231-1245.

78. Lee TI, Rinaldi NJ, Robert F, Odom DT, Bar-Joseph Z, Gerber GK, Hannett NM, Harbison CT, Thompson CM, Simon I *et al*: **Transcriptional regulatory networks in Saccharomyces cerevisiae**. *Science* 2002, **298**(5594):799-804.

79. Ren B, Robert F, Wyrick JJ, Aparicio O, Jennings EG, Simon I, Zeitlinger J, Schreiber J, Hannett N, Kanin E *et al*: **Genome-wide location and function of DNA binding proteins**. *Science* 2000, **290**(5500):2306-2309.

80. Wilbanks EG, Facciotti MT: **Evaluation of algorithm performance in ChIP-seq peak detection**. *PloS one* 2010, **5**(7):e11471.

81. Rhee HS, Pugh BF: **Comprehensive genome-wide protein-DNA interactions detected at single-nucleotide resolution**. *Cell* 2011, **147**(6):1408-1419.

82. Pepke S, Wold B, Mortazavi A: **Computation for ChIP-seq and RNA-seq studies**. *Nature methods* 2009, **6**(11 Suppl):S22-32.

83. Rye MB, Sætrom P, Drabløs F: **A manually curated ChIP-seq benchmark demonstrates room for improvement in current peak-finder programs**. *Nucleic acids research* 2011, **39**(4):e25.

84. van Steensel B, Henikoff S: **Identification of in vivo DNA targets of chromatin proteins using tethered dam methyltransferase**. *Nature biotechnology* 2000, **18**(4):424-428.

85. van Steensel B, Delrow J, Henikoff S: **Chromatin profiling using targeted DNA adenine methyltransferase**. *Nature genetics* 2001, **27**(3):304-308.

86. Tuerk C, Gold L: **Systematic evolution of ligands by exponential enrichment: RNA ligands to bacteriophage T4 DNA polymerase**. *Science* 1990, **249**(4968):505-510.

87. Ellington AD, Szostak JW: **In vitro selection of RNA molecules that bind specific ligands**. *Nature* 1990, **346**(6287):818-822.

88. Oliphant AR, Brandl CJ, Struhl K: **Defining the sequence specificity of DNA-binding proteins by selecting binding sites from random-sequence oligonucleotides: analysis of yeast GCN4 protein**. *Molecular and cellular biology* 1989, **9**(7):2944-2949.

89. Bulyk ML, Gentalen E, Lockhart DJ, Church GM: **Quantifying DNA-protein interactions by double-stranded DNA arrays**. *Nature biotechnology* 1999, **17**(6):573-577.

90. Bulyk ML, Huang X, Choo Y, Church GM: **Exploring the DNA-binding specificities of zinc fingers with DNA microarrays**. *Proceedings of the National Academy of Sciences of the United States of America* 2001, **98**(13):7158-7163.

91. Berger MF, Bulyk ML: **Universal protein-binding microarrays for the comprehensive characterization of the DNA-binding specificities of transcription factors**. *Nature protocols* 2009, **4**(3):393-411.

92. Warren CL, Kratochvil NC, Hauschild KE, Foister S, Brezinski ML, Dervan PB, Phillips GN, Jr., Ansari AZ: **Defining the sequence-recognition profile of DNA-binding molecules**. *Proceedings of the National Academy of Sciences of the United States of America* 2006, **103**(4):867-872.

93. Mukherjee S, Berger MF, Jona G, Wang XS, Muzzey D, Snyder M, Young RA, Bulyk ML: **Rapid analysis of the DNA-binding specificities of transcription factors with DNA microarrays**. *Nature genetics* 2004, **36**(12):1331-1339.

94. ENCODE Project Consortium: **An integrated encyclopedia of DNA elements in the human genome**. *Nature* 2012, **489**(7414):57-74.

95. Sandve GK, Drabløs F: **A survey of motif discovery methods in an integrated framework**. *Biology direct* 2006, **1**:11.

96. Cornish-Bowden A: **Nomenclature for incompletely specified bases in nucleic acid sequences: recommendations 1984**. *Nucleic acids research* 1985, **13**(9):3021-3030.

97. Cavener DR: **Comparison of the consensus sequence flanking translational start sites in Drosophila and vertebrates**. *Nucleic acids research* 1987, **15**(4):1353-1361.

98. Day WH, McMorris FR: **Critical comparison of consensus methods for molecular sequences**. *Nucleic acids research* 1992, **20**(5):1093-1099.

99. Stormo GD, Schneider TD, Gold L, Ehrenfeucht A: **Use of the 'Perceptron' algorithm to distinguish translational initiation sites in E. coli**. *Nucleic acids research* 1982, **10**(9):2997-3011.

100. Staden R: **Computer methods to locate signals in nucleic acid sequences**. *Nucleic acids research* 1984, **12**(1 Pt 2):505-519.

101. D'Haeseleer P: **What are DNA sequence motifs?** *Nature biotechnology* 2006, **24**(4):423-425.

102. Stormo GD, Fields DS: **Specificity, free energy and information content in protein-DNA interactions**. *Trends in biochemical sciences* 1998, **23**(3):109-113.

103. Schneider TD, Stormo GD, Gold L, Ehrenfeucht A: **Information content of binding sites on nucleotide sequences**. *Journal of molecular biology* 1986, **188**(3):415-431.

104. Shannon CE: **A Mathematical Theory of Communication**. *Bell System Technical Journal* 1948, **27**(3):379–423.

105. Kullback S, Leibler RA: **On Information and Sufficiency**. *Annals of Mathematical Statistics* 1951, **22**(1):79-86.

106. Berg OG, von Hippel PH: **Selection of DNA binding sites by regulatory proteins. Statistical-mechanical theory and application to operators and promoters**. *Journal of molecular biology* 1987, **193**(4):723-750.

107. Schneider TD, Stephens RM: **Sequence logos: a new way to display consensus sequences**. *Nucleic acids research* 1990, **18**(20):6097-6100.

108. Bulyk ML, Johnson PL, Church GM: **Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors**. *Nucleic acids research* 2002, **30**(5):1255-1261.

109. Benos PV, Lapedes AS, Stormo GD: **Probabilistic code for DNA recognition by proteins of the EGR family**. *Journal of molecular biology* 2002, **323**(4):701-727.

110. Sadeghi M-R, Zare-Mirakabad F, Tahmasebi M, Sadeghi M: **EPWM: An Extended Position Weight Matrix for Motif Representation in Biological Sequences**. *Journal of Basic and Applied Scientific Research* 2012, **2**(7):7276-7286.

111. Gershenzon NI, Stormo GD, Ioshikhes IP: **Computational technique for improvement of the position-weight matrices for the DNA/protein binding sites**. *Nucleic acids research* 2005, **33**(7):2290-2301.

112. Barash Y, Elidan G, Friedman N, Kaplan T: **Modeling Dependencies in Protein-DNA Binding Sites**. In: *Conference on Research in Computational Molecular Biology (RECOMB): April 10-13, 2003; Berlin, Germany*. 2003: 28-37.

113. Ben-Gal I, Shani A, Gohr A, Grau J, Arviv S, Shmilovici A, Posch S, Grosse I: **Identification of transcription factor binding sites with variable-order Bayesian networks**. *Bioinformatics* 2005, **21**(11):2657-2666.

114. Zhao X, Huang H, Speed TP: **Finding short DNA motifs using permuted Markov models**. *Journal of computational biology* 2005, **12**(6):894-906.

115. Sharon E, Lubliner S, Segal E: **A feature-based approach to modeling protein-DNA interactions**. *PLoS computational biology* 2008, **4**(8):e1000154.

116. Faisst S, Meyer S: **Compilation of vertebrate-encoded transcription factors**. *Nucleic acids research* 1992, **20**(1):3-26.

117. Matys V, Kel-Margoulis OV, Fricke E, Liebich I, Land S, Barre-Dirrie A, Reuter I, Chekmenev D, Krull M, Hornischer K *et al*: **TRANSFAC and its module TRANSCompel: transcriptional gene regulation in eukaryotes**. *Nucleic acids research* 2006, **34**(Database issue):D108-110.

118. Portales-Casamar E, Thongjuea S, Kwon AT, Arenillas D, Zhao X, Valen E, Yusuf D, Lenhard B, Wasserman WW, Sandelin A: **JASPAR 2010: the greatly expanded open-access database of transcription factor binding profiles**. *Nucleic acids research* 2010, **38**(Database issue):D105-110.

119. Spivak AT, Stormo GD: **ScerTF: a comprehensive database of benchmarked position weight matrices for Saccharomyces species**. *Nucleic acids research* 2012, **40**(Database issue):D162-168.

120. Zhu J, Zhang MQ: **SCPD: a promoter database of the yeast Saccharomyces cerevisiae**. *Bioinformatics* 1999, **15**(7-8):607-611.

121. Teixeira MC, Monteiro P, Jain P, Tenreiro S, Fernandes AR, Mira NP, Alenquer M, Freitas AT, Oliveira AL, Sa-Correia I: **The YEASTRACT database: a tool for the analysis of transcription regulatory associations in Saccharomyces cerevisiae**. *Nucleic acids research* 2006, **34**(Database issue):D446-451.

122. Gama-Castro S, Salgado H, Peralta-Gil M, Santos-Zavaleta A, Muniz-Rascado L, Solano-Lira H, Jimenez-Jacinto V, Weiss V, Garcia-Sotelo JS, Lopez-Fuentes A *et al*: **RegulonDB version 7.0: transcriptional regulation of Escherichia coli K-12 integrated within genetic sensory response units (Gensor Units)**. *Nucleic acids research* 2011, **39**(Database issue):D98-105.

123. Grote A, Klein J, Retter I, Haddad I, Behling S, Bunk B, Biegler I, Yarmolinetz S, Jahn D, Munch R: **PRODORIC (release 2009): a database and tool platform for the analysis of gene regulation in prokaryotes**. *Nucleic acids research* 2009, **37**(Database issue):D61-65.

124. Zhang Y, Wu W, Cheng Y, King DC, Harris RS, Taylor J, Chiaromonte F, Hardison RC: **Primary sequence and epigenetic determinants of in vivo occupancy of genomic DNA by GATA1**. *Nucleic acids research* 2009, **37**(21):7024-7038.

125. Harr R, Haggstrom M, Gustafsson P: **Search algorithm for pattern match analysis of nucleic acid sequences**. *Nucleic acids research* 1983, **11**(9):2943-2957.

126. Nishida K, Frith MC, Nakai K: **Pseudocounts for transcription factor binding sites**. *Nucleic acids research* 2009, **37**(3):939-944.

127. Bucher P: **Weight matrix descriptions of four eukaryotic RNA polymerase II promoter elements derived from 502 unrelated promoter sequences**. *Journal of molecular biology* 1990, **212**(4):563-578.

128. Tsunoda T, Takagi T: **Estimating transcription factor bindability on DNA**. *Bioinformatics* 1999, **15**(7-8):622-630.

129. Kel AE, Gossling E, Reuter I, Cheremushkin E, Kel-Margoulis OV, Wingender E: **MATCH: A tool for searching transcription factor binding sites in DNA sequences**. *Nucleic acids research* 2003, **31**(13):3576-3579.

130. Thomas-Chollier M, Sand O, Turatsinze JV, Janky R, Defrance M, Vervisch E, Brohee S, van Helden J: **RSAT: regulatory sequence analysis tools**. *Nucleic acids research* 2008, **36**(Web Server issue):W119-127.

131. Frith MC, Fu Y, Yu L, Chen JF, Hansen U, Weng Z: **Detection of functional DNA motifs via statistical over-representation**. *Nucleic acids research* 2004, **32**(4):1372-1381.

132. Grant CE, Bailey TL, Noble WS: **FIMO: scanning for occurrences of a given motif**. *Bioinformatics* 2011, **27**(7):1017-1018.

133. Turatsinze JV, Thomas-Chollier M, Defrance M, van Helden J: **Using RSAT to scan genome sequences for transcription factor binding sites and cis-regulatory modules**. *Nature protocols* 2008, **3**(10):1578-1588.

134. Claeys M, Storms V, Sun H, Michoel T, Marchal K: **MotifSuite: workflow for probabilistic motif detection and assessment**. *Bioinformatics* 2012, **28**(14):1931-1932.

135. Korn LJ, Queen CL, Wegman MN: **Computer analysis of nucleic acid regulatory sequences**. *Proceedings of the National Academy of Sciences of the United States of America* 1977, **74**(10):4401-4405.

136. Lander ES, Linton LM, Birren B, Nusbaum C, Zody MC, Baldwin J, Devon K, Dewar K, Doyle M, FitzHugh W *et al*: **Initial sequencing and analysis of the human genome**. *Nature* 2001, **409**(6822):860-921.

137. Frith MC, Hansen U, Spouge JL, Weng Z: **Finding functional sequence elements by multiple local alignment**. *Nucleic acids research* 2004, **32**(1):189-200.

138. Bailey TL: **Discovering sequence motifs**. *Methods in molecular biology* 2008, **452**:231-251.

139. Sumazin P, Chen G, Hata N, Smith AD, Zhang T, Zhang MQ: **DWE: discriminating word enumerator**. *Bioinformatics* 2005, **21**(1):31-38.

140. Smith AD, Sumazin P, Zhang MQ: **Identifying tissue-selective transcription factor binding sites in vertebrate promoters**. *Proceedings of the National Academy of Sciences of the United States of America* 2005, **102**(5):1560-1565.

141. Redhead E, Bailey TL: **Discriminative motif discovery in DNA and protein sequences using the DEME algorithm**. *BMC bioinformatics* 2007, **8**:385.

142. Bailey TL: **DREME: motif discovery in transcription factor ChIP-seq data**. *Bioinformatics* 2011, **27**(12):1653-1659.

143. Sinha S, Tompa M: **A statistical method for finding transcription factor binding sites**. *Proceedings of the International Conference on Intelligent Systems for Molecular Biology* 2000, **8**:344-354.

144. Sinha S, Tompa M: **YMF: A program for discovery of novel transcription factor binding sites by statistical overrepresentation**. *Nucleic acids research* 2003, **31**(13):3586-3588.

145. Wang G, Yu T, Zhang W: **WordSpy: identifying transcription factor binding motifs by building a dictionary and learning a grammar**. *Nucleic acids research* 2005, **33**(Web Server issue):W412-416.

146. Ettwiller L, Paten B, Ramialison M, Birney E, Wittbrodt J: **Trawler: de novo regulatory motif discovery pipeline for chromatin immunoprecipitation**. *Nature methods* 2007, **4**(7):563-565.

147. Tompa M: **An exact method for finding short motifs in sequences, with application to the ribosome binding site problem**. *Proceedings of the International Conference on Intelligent Systems for Molecular Biology* 1999:262-271.

148. Aerts S, Van Loo P, Thijs G, Mayer H, de Martin R, Moreau Y, De Moor B: **TOUCAN 2: the all-inclusive open source workbench for regulatory sequence analysis**. *Nucleic acids research* 2005, **33**(Web Server issue):W393-396.

149. Jia H, Li J: **Finding Transcription Factor Binding Motifs for Coregulated Genes by Combining Sequence Overrepresentation with Cross-Species Conservation**. *Journal of probability and statistic* 2012, **2012**:830575.

150. Linhart C, Halperin Y, Shamir R: **Transcription factor and microRNA motif discovery: the Amadeus platform and a compendium of metazoan target sets**. *Genome research* 2008, **18**(7):1180-1189.

151. Romer KA, Kayombya GR, Fraenkel E: **WebMOTIFS: automated discovery, filtering and scoring of DNA sequence motifs using multiple programs and Bayesian approaches**. *Nucleic acids research* 2007, **35**(Web Server issue):W217-220.

152. Waterman MS, Arratia R, Galas DJ: **Pattern recognition in several sequences: consensus and alignment**. *Bulletin of mathematical biology* 1984, **46**(4):515-527.

153. Pavesi G, Mauri G, Pesole G: **An algorithm for finding signals of unknown length in DNA sequences**. *Bioinformatics* 2001, **17 Suppl 1**:S207-214.

154. Marsan L, Sagot MF: **Algorithms for extracting structured motifs using a suffix tree with an application to promoter and regulatory site consensus identification**. *Journal of computational biology* 2000, **7**(3-4):345-362.

155. Eskin E, Pevzner PA: **Finding composite regulatory patterns in DNA sequences**. *Bioinformatics* 2002, **18 Suppl 1**:S354-363.

156. Rigoutsos I, Floratos A: **Combinatorial pattern discovery in biological sequences: The TEIRESIAS algorithm**. *Bioinformatics* 1998, **14**(1):55-67.

157. Akutsu T, Arimura H, Shimozono S: **On approximation algorithms for local multiple alignment**. In: *Proceedings of the fourth annual international conference on Computational molecular biology; Tokyo, Japan*. 2000: 1-7.

158. Li M, Ma B, Wang L: **Finding Similar Regions in Many Sequences**. *Journal of Computer and System Sciences* 2002, **65**(1):73-96.

159. Stormo GD, Hartzell GW, 3rd: **Identifying protein-binding sites from unaligned DNA fragments**. *Proceedings of the National Academy of Sciences of the United States of America* 1989, **86**(4):1183-1187.

160. Lawrence CE, Reilly AA: **An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences**. *Proteins* 1990, **7**(1):41-51.

161. Bailey TL, Elkan C: **Fitting a mixture model by expectation maximization to discover motifs in biopolymers**. *Proceedings of the International Conference on Intelligent Systems for Molecular Biology* 1994, **2**:28-36.

162. Ao W, Gaudet J, Kent WJ, Muttumu S, Mango SE: **Environmentally induced foregut remodeling by PHA-4/FoxA and DAF-12/NHR**. *Science* 2004, **305**(5691):1743-1746.

163. Sinha S, Blanchette M, Tompa M: **PhyME: a probabilistic algorithm for finding motifs in sets of orthologous sequences**. *BMC bioinformatics* 2004, **5**:170.

164. Lawrence CE, Altschul SF, Boguski MS, Liu JS, Neuwald AF, Wootton JC: **Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment**. *Science* 1993, **262**(5131):208-214.

165. Roth FP, Hughes JD, Estep PW, Church GM: **Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation**. *Nature biotechnology* 1998, **16**(10):939-945.

166. Workman CT, Stormo GD: **ANN-Spec: a method for discovering transcription factor binding sites with improved specificity**. *Pacific Symposium on Biocomputing* 2000:467-478.

167. Liu X, Brutlag DL, Liu JS: **BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes**. *Pacific Symposium on Biocomputing* 2001:127-138.

168. Thijs G, Lescot M, Marchal K, Rombauts S, De Moor B, Rouze P, Moreau Y: **A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling**. *Bioinformatics* 2001, **17**(12):1113-1122.

169. Narlikar L, Gordân R, Ohler U, Hartemink AJ: **Informative priors based on transcription factor structural class improve de novo motif discovery**. *Bioinformatics* 2006, **22**(14):e384-392.

170. Favorov AV, Gelfand MS, Gerasimova AV, Ravcheev DA, Mironov AA, Makeev VJ: **A Gibbs sampler for identification of symmetrically structured, spaced DNA motifs with improved estimation of the signal length**. *Bioinformatics* 2005, **21**(10):2240-2245.

171. Valen E, Sandelin A, Winther O, Krogh A: **Discovery of regulatory elements is improved by a discriminatory approach**. *PLoS computational biology* 2009, **5**(11):e1000562.

172. Yang C-H, Liu Y-T, Chuang L-Y: **DNA Motif Discovery Based on Ant Colony Optimization and Expectation Maximization**. In: *Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS): March 16-18, 2011; Hong Kong*. 2011: 169-174.

173. Reddy US, Arock M, Reddy AV: **Planted (l,d)-motif finding using particle swarm optimization**. *IJCA Special Issue on Evolutionary Computation for Optimization Techniques* 2010, **2**:51–56.

174. Lei C, Ruan J: **A particle swarm optimization-based algorithm for finding gapped motifs**. *BioData mining* 2010, **3**:9.

175. Fogel GB, Weekes DG, Varga G, Dow ER, Harlow HB, Onyia JE, Su C: **Discovery of sequence motifs related to coexpression of genes using evolutionary computation**. *Nucleic acids research* 2004, **32**(13):3826-3835.

176. Che D, Song Y, Rasheed K: **MDGA: motif discovery using a genetic algorithm**. In: *Proceedings of the 2005 conference on Genetic and evolutionary computation (GECCO '05): June 25-29; Washington D.C., USA*: Edited by Beyer H-G. ACM 2005: 447-452.

177. Wei Z, Jensen ST: **GAME: detecting cis-regulatory elements using a genetic algorithm**. *Bioinformatics* 2006, **22**(13):1577-1584.

178. Kaya M: **MOGAMOD: Multi-objective genetic algorithm for motif discovery**. *Expert Systems with Applications* 2009, **36**(2, Part 1):1039-1047.

179. Lones MA, Tyrrell AM: **Regulatory motif discovery using a population clustering evolutionary algorithm**. *IEEE/ACM transactions on computational biology and bioinformatics / IEEE, ACM* 2007, **4**(3):403-414.

180. Luo JW, Wang T: **Motif discovery using an immune genetic algorithm**. *Journal of theoretical biology* 2010, **264**(2):319-325.

181. Liu FFM, Tsai JJP, Chen RM, Chen SN, Shih SH: **FMGA: finding motifs by genetic algorithm**. In: *Proceeding of the 4th IEEE International Symposium on BioInformatics and BioEngineering (BIBE 2004): 19-21 May 2004; Taichung, Taiwan*. 459-466.

182. Pevzner PA, Sze SH: **Combinatorial approaches to finding subtle signals in DNA sequences**. *Proceedings of the International Conference on Intelligent Systems for Molecular Biology (ISMB)* 2000, **8**:269-278.

183. Fratkin E, Naughton BT, Brutlag DL, Batzoglou S: **MotifCut: regulatory motifs finding with maximum density subgraphs**. *Bioinformatics* 2006, **22**(14):e150-157.

184. Zhang S, Li S, Niu M, Pham PT, Su Z: **MotifClick: prediction of cis-regulatory binding sites via merging cliques**. *BMC bioinformatics* 2011, **12**:238.

185. Morozov AV, Havranek JJ, Baker D, Siggia ED: **Protein-DNA binding specificity predictions with structural models**. *Nucleic acids research* 2005, **33**(18):5781-5798.

186. Alamanova D, Stegmaier P, Kel A: **Creating PWMs of transcription factors using 3D structure-based computation of protein-DNA free binding energies**. *BMC bioinformatics* 2010, **11**:225.

187. Van Loo P, Marynen P: **Computational methods for the detection of cis-regulatory modules**. *Briefings in bioinformatics* 2009, **10**(5):509-524.

188. Aerts S: **Computational strategies for the genome-wide identification of cis-regulatory elements and transcriptional targets**. *Current topics in developmental biology* 2012, **98**:121-145.

189. Aerts S, Van Loo P, Moreau Y, De Moor B: **A genetic algorithm for the detection of new cis-regulatory modules in sets of coregulated genes**. *Bioinformatics* 2004, **20**(12):1974-1976.

190. Kel A, Konovalova T, Waleev T, Cheremushkin E, Kel-Margoulis O, Wingender E: **Composite Module Analyst: a fitness-based tool for identification of transcription factor binding site combinations**. *Bioinformatics* 2006, **22**(10):1190-1197.

191. Zhou Q, Wong WH: **CisModule: de novo discovery of cis-regulatory modules by hierarchical mixture modeling**. *Proceedings of the National Academy of Sciences of the United States of America* 2004, **101**(33):12114-12119.

192. Gupta M, Liu JS: **De novo cis-regulatory module elicitation for eukaryotic genomes**. *Proceedings of the National Academy of Sciences of the United States of America* 2005, **102**(20):7079-7084.

193. Van Loo P, Aerts S, Thienpont B, De Moor B, Moreau Y, Marynen P: **ModuleMiner - improved computational detection of cis-regulatory modules: are there different modes of gene regulation in embryonic development and adult tissues?** *Genome biology* 2008, **9**(4):R66.

194. Johansson O, Alkema W, Wasserman WW, Lagergren J: **Identification of functional clusters of transcription factor binding motifs in genome sequences: the MSCAN algorithm**. *Bioinformatics* 2003, **19 Suppl 1**:i169-176.

195. Frith MC, Hansen U, Weng Z: **Detection of cis-element clusters in higher eukaryotic DNA**. *Bioinformatics* 2001, **17**(10):878-889.

196. Frith MC, Li MC, Weng Z: **Cluster-Buster: Finding dense clusters of motifs in DNA sequences**. *Nucleic acids research* 2003, **31**(13):3666-3668.

197. Bailey TL, Noble WS: **Searching for statistically significant regulatory modules**. *Bioinformatics* 2003, **19 Suppl 2**:ii16-25.

198. Sinha S, van Nimwegen E, Siggia ED: **A probabilistic method to detect regulatory modules**. *Bioinformatics* 2003, **19 Suppl 1**:i292-301.

199. Kantorovitz MR, Kazemian M, Kinston S, Miranda-Saavedra D, Zhu Q, Robinson GE, Gottgens B, Halfon MS, Sinha S: **Motif-blind, genome-wide discovery of cis-regulatory modules in Drosophila and mouse**. *Developmental cell* 2009, **17**(4):568-579.

200. Hu J, Li B, Kihara D: **Limitations and potentials of current motif discovery algorithms**. *Nucleic acids research* 2005, **33**(15):4899-4913.

201. Hu J, Yang YD, Kihara D: **EMD: an ensemble algorithm for discovering regulatory motifs in DNA sequences**. *BMC bioinformatics* 2006, **7**:342.

202. Wijaya E, Yiu SM, Son NT, Kanagasabai R, Sung WK: **MotifVoter: a novel ensemble method for fine-grained integration of generic motif finders**. *Bioinformatics* 2008, **24**(20):2288-2295.

203. Chakravarty A, Carlson JM, Khetani RS, Gross RH: **A novel ensemble learning method for de novo computational identification of DNA binding sites**. *BMC bioinformatics* 2007, **8**:249.

204. Li N, Tompa M: **Analysis of computational approaches for motif discovery**. *Algorithms for molecular biology* 2006, **1**:8.

205. Tompa M, Li N, Bailey TL, Church GM, De Moor B, Eskin E, Favorov AV, Frith MC, Fu Y, Kent WJ *et al*: **Assessing computational tools for the discovery of transcription factor binding sites**. *Nature biotechnology* 2005, **23**(1):137-144.

206. Sandve GK, Abul O, Walseng V, Drabløs F: **Improved benchmarks for computational motif discovery**. *BMC bioinformatics* 2007, **8**:193.

207. Quest D, Dempsey K, Shafiullah M, Bastola D, Ali H: **MTAP: the motif tool assessment platform**. *BMC bioinformatics* 2008, **9 Suppl 9**:S6.

208. Rajewsky N, Vergassola M, Gaul U, Siggia ED: **Computational detection of genomic cis-regulatory modules applied to body patterning in the early Drosophila embryo**. *BMC bioinformatics* 2002, **3**:30.

209. Ivan A, Halfon MS, Sinha S: **Computational discovery of cis-regulatory modules in Drosophila without prior knowledge of motifs**. *Genome biology* 2008, **9**(1):R22.

210. Gallo SM, Gerrard DT, Miner D, Simich M, Des Soye B, Bergman CM, Halfon MS: **REDfly v3.0: toward a comprehensive database of transcriptional regulatory elements in Drosophila**. *Nucleic acids research* 2011, **39**(Database issue):D118-123.

211. Su J, Teichmann SA, Down TA: **Assessing computational methods of cis-regulatory module prediction**. *PLoS computational biology* 2010, **6**(12):e1001020.

212. Klepper K, Sandve GK, Rye MB, Bolstad KH, Drabløs F: **Benchmarking of methods for motif discovery in DNA**. In: *Advances in Genomic Sequence Analysis and Pattern Discovery.* Edited by Elnitski L, Piontkivska H, Welch LR, vol. 7: World Scientific Publishing; 2010: 135-157.

213. Needleman SB, Wunsch CD: **A general method applicable to the search for similarities in the amino acid sequence of two proteins**. *Journal of molecular biology* 1970, **48**(3):443-453.

214. Smith TF, Waterman MS: **Identification of common molecular subsequences**. *Journal of molecular biology* 1981, **147**(1):195-197.

215. Mahony S, Benos PV: **STAMP: a web tool for exploring DNA-binding motif similarities**. *Nucleic acids research* 2007, **35**(Web Server issue):W253-258.

216. Kankainen M, Loytynoja A: **MATLIGN: a motif clustering, comparison and matching tool**. *BMC bioinformatics* 2007, **8**:189.

217. Gupta S, Stamatoyannopoulos JA, Bailey TL, Noble WS: **Quantifying similarity between motifs**. *Genome biology* 2007, **8**(2):R24.

218. Roepcke S, Grossmann S, Rahmann S, Vingron M: **T-Reg Comparator: an analysis tool for the comparison of position weight matrices**. *Nucleic acids research* 2005, **33**(Web Server issue):W438-441.

219. Shiraki T, Kondo S, Katayama S, Waki K, Kasukawa T, Kawaji H, Kodzius R, Watahiki A, Nakamura M, Arakawa T *et al*: **Cap analysis gene expression for high-throughput analysis of transcriptional starting point and identification of promoter usage**. *Proceedings of the National Academy of Sciences of the United States of America* 2003, **100**(26):15776-15781.

220. Wasserman WW, Sandelin A: **Applied bioinformatics for the identification of regulatory elements**. *Nature reviews genetics* 2004, **5**(4):276-287.

221. Ucar D, Beyer A, Parthasarathy S, Workman CT: **Predicting functionality of protein-DNA interactions by integrating diverse evidence**. *Bioinformatics* 2009, **25**(12):i137-144.

222. Tagle DA, Koop BF, Goodman M, Slightom JL, Hess DL, Jones RT: **Embryonic epsilon and gamma globin genes of a prosimian primate (Galago crassicaudatus). Nucleotide and amino acid sequences, developmental regulation and phylogenetic footprints**. *Journal of molecular biology* 1988, **203**(2):439-455.

223. Boffelli D, McAuliffe J, Ovcharenko D, Lewis KD, Ovcharenko I, Pachter L, Rubin EM: **Phylogenetic shadowing of primate sequences to find functional regions of the human genome**. *Science* 2003, **299**(5611):1391-1394.

224. Dermitzakis ET, Clark AG: **Evolution of transcription factor binding sites in Mammalian gene regulatory regions: conservation and turnover**. *Molecular biology and evolution* 2002, **19**(7):1114-1121.

225. Blanchette M, Tompa M: **Discovery of regulatory elements by a computational method for phylogenetic footprinting**. *Genome research* 2002, **12**(5):739-748.

226. Das MK, Dai HK: **A survey of DNA motif finding algorithms**. *BMC bioinformatics* 2007, **8 Suppl 7**:S21.

227. Nguyen TT, Androulakis IP: **Recent Advances in the Computational Discovery of Transcription Factor Binding Sites**. *Algorithms* 2009, **2**(1):582-605.

228. Segal E, Fondufe-Mittendorf Y, Chen L, Thastrom A, Field Y, Moore IK, Wang JP, Widom J: **A genomic code for nucleosome positioning**. *Nature* 2006, **442**(7104):772-778.

229. Narlikar L, Gordân R, Hartemink AJ: **A nucleosome-guided map of transcription factor binding sites in yeast**. *PLoS computational biology* 2007, **3**(11):e215.

230. Gordân R, Hartemink AJ: **Using DNA duplex stability information for transcription factor binding site discovery**. *Pacific Symposium on Biocomputing* 2008:453-464.

231. Gordân R, Narlikar L, Hartemink AJ: **Finding regulatory DNA motifs using alignment-free evolutionary conservation information**. *Nucleic acids research* 2010, **38**(6):e90.

232. Bailey TL, Boden M, Whitington T, Machanick P: **The value of position-specific priors in motif discovery using MEME**. *BMC bioinformatics* 2010, **11**:179.

233. Carvalho AM, Oliveira AL: **GRISOTTO: A greedy approach to improve combinatorial algorithms for motif discovery with prior knowledge**. *Algorithms for molecular biology : AMB* 2011, **6**:13.

234. Tang MH, Krogh A, Winther O: **BayesMD: flexible biological modeling for motif discovery**. *Journal of computational biology* 2008, **15**(10):1347-1363.

235. Qi Y, Rolfe A, MacIsaac KD, Gerber GK, Pokholok D, Zeitlinger J, Danford T, Dowell RD, Fraenkel E, Jaakkola TS *et al*: **High-resolution computational models of genome binding events**. *Nature biotechnology* 2006, **24**(8):963-970.

236. Kulakovskiy IV, Boeva VA, Favorov AV, Makeev VJ: **Deep and wide digging for binding motifs in ChIP-Seq data**. *Bioinformatics* 2010, **26**(20):2622-2623.

237. Lähdesmäki H, Rust AG, Shmulevich I: **Probabilistic inference of transcription factor binding from multiple data sources**. *PloS one* 2008, **3**(3):e1820.

238. Elnitski L, Hardison RC, Li J, Yang S, Kolbe D, Eswara P, O'Connor MJ, Schwartz S, Miller W, Chiaromonte F: **Distinguishing regulatory DNA from neutral sites**. *Genome research* 2003, **13**(1):64-72.

239. Kolbe D, Taylor J, Elnitski L, Eswara P, Li J, Miller W, Hardison R, Chiaromonte F: **Regulatory potential scores from genome-wide three-way alignments of human, mouse, and rat**. *Genome research* 2004, **14**(4):700-707.

240. Crawford GE, Holt IE, Mullikin JC, Tai D, Blakesley R, Bouffard G, Young A, Masiello C, Green ED, Wolfsberg TG *et al*: **Identifying gene regulatory elements by genome-wide recovery of DNase hypersensitive sites**. *Proceedings of the National Academy of Sciences of the United States of America* 2004, **101**(4):992-997.

241. Cuellar-Partida G, Buske FA, McLeay RC, Whitington T, Noble WS, Bailey TL: **Epigenetic priors for identifying active transcription factor binding sites**. *Bioinformatics* 2012, **28**(1):56-62.

242. Ramsey SA, Knijnenburg TA, Kennedy KA, Zak DE, Gilchrist M, Gold ES, Johnson CD, Lampano AE, Litvak V, Navarro G *et al*: **Genome-wide histone acetylation data improve prediction of mammalian transcription factor binding sites**. *Bioinformatics* 2010, **26**(17):2071-2075.

243. Pique-Regi R, Degner JF, Pai AA, Gaffney DJ, Gilad Y, Pritchard JK: **Accurate inference of transcription factor binding from DNA sequence and chromatin accessibility data**. *Genome research* 2011, **21**(3):447-455.

244. Ernst J, Plasterer HL, Simon I, Bar-Joseph Z: **Integrating multiple evidence sources to predict transcription factor binding in the human genome**. *Genome research* 2010, **20**(4):526-536.

245. Won KJ, Agarwal S, Shen L, Shoemaker R, Ren B, Wang W: **An integrated approach to identifying cis-regulatory modules in the human genome**. *PloS one* 2009, **4**(5):e5501.

246. Won KJ, Ren B, Wang W: **Genome-wide prediction of transcription factor binding sites using an integrated model**. *Genome biology* 2010, **11**(1):R7.

247. Kang K, Kim J, Chung JH, Lee D: **Decoding the genome with an integrative analysis tool: combinatorial CRM Decoder**. *Nucleic acids research* 2011, **39**(17):e116.

248. Herrmann C, Van de Sande B, Potier D, Aerts S: **i-cisTarget: an integrative genomics method for the prediction of regulatory features and cis-regulatory modules**. *Nucleic acids research* 2012, **40**(15):e114.

249. Bussemaker HJ, Li H, Siggia ED: **Regulatory element detection using correlation with expression**. *Nature genetics* 2001, **27**(2):167-171.

250. Conlon EM, Liu XS, Lieb JD, Liu JS: **Integrating regulatory motif discovery and genome-wide expression analysis**. *Proceedings of the National Academy of Sciences of the United States of America* 2003, **100**(6):3339-3344.

251. Suzuki H, Forrest AR, van Nimwegen E, Daub CO, Balwierz PJ, Irvine KM, Lassmann T, Ravasi T, Hasegawa Y, de Hoon MJ *et al*: **The transcriptional network that controls growth arrest and differentiation in a human myeloid leukemia cell line**. *Nature genetics* 2009, **41**(5):553-562.

252. Abul O, Drabløs F, Sandve GK: **A methodology for motif discovery employing iterated cluster re-assignment**. In: *Proceedings of the Computational Systems Bioinformatics Conference (CSB): August 14-18, 2006; Stanford, California, USA*. 2006: 257-268.

253. Duret L, Bucher P: **Searching for regulatory elements in human noncoding sequences**. *Current opinion in structural biology* 1997, **7**(3):399-406.

254. Siepel A, Bejerano G, Pedersen JS, Hinrichs AS, Hou M, Rosenbloom K, Clawson H, Spieth J, Hillier LW, Richards S *et al*: **Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes**. *Genome research* 2005, **15**(8):1034-1050.

255. Ioshikhes I, Trifonov EN, Zhang MQ: **Periodical distribution of transcription factor sites in promoter regions and connection with chromatin structure**. *Proceedings of the National Academy of Sciences of the United States of America* 1999, **96**(6):2891-2895.

256. Ioshikhes I, Bolshoy A, Derenshteyn K, Borodovsky M, Trifonov EN: **Nucleosome DNA sequence pattern revealed by multiple alignment of experimentally mapped sequences**. *Journal of molecular biology* 1996, **262**(2):129-139.

257. Gupta S, Dennis J, Thurman RE, Kingston R, Stamatoyannopoulos JA, Noble WS: **Predicting human nucleosome occupancy from primary sequence**. *PLoS computational biology* 2008, **4**(8):e1000134.

258. Segal E, Widom J: **What controls nucleosome positions?** *Trends in genetics : TIG* 2009, **25**(8):335-343.

259. Fu Y, Sinha M, Peterson CL, Weng Z: **The insulator binding protein CTCF positions 20 nucleosomes around its binding sites across the human genome**. *PLoS genetics* 2008, **4**(7):e1000138.

260. Brogaard K, Xi L, Wang JP, Widom J: **A map of nucleosome positions in yeast at base-pair resolution**. *Nature* 2012, **486**(7404):496-501.

261. Schones DE, Cui K, Cuddapah S, Roh TY, Barski A, Wang Z, Wei G, Zhao K: **Dynamic regulation of nucleosome positioning in the human genome**. *Cell* 2008, **132**(5):887-898.

262. Biliya S, Bulla LA, Jr.: **Genomic imprinting: the influence of differential methylation in the two sexes**. *Experimental biolology and medicine* 2010, **235**(2):139-147.

263. Frommer M, McDonald LE, Millar DS, Collis CM, Watt F, Grigg GW, Molloy PL, Paul CL: **A genomic sequencing protocol that yields a positive display of 5-methylcytosine residues in individual DNA strands**. *Proceedings of the National Academy of Sciences of the United States of America* 1992, **89**(5):1827-1831.

264. Brunner AL, Johnson DS, Kim SW, Valouev A, Reddy TE, Neff NF, Anton E, Medina C, Nguyen L, Chiao E *et al*: **Distinct DNA methylation patterns characterize differentiated human embryonic stem cells and developing human fetal liver**. *Genome research* 2009, **19**(6):1044-1056.

265. Down TA, Rakyan VK, Turner DJ, Flicek P, Li H, Kulesha E, Graf S, Johnson N, Herrero J, Tomazou EM *et al*: **A Bayesian deconvolution strategy for immunoprecipitation-based DNA methylome analysis**. *Nature biotechnology* 2008, **26**(7):779-785.

266. Saxonov S, Berg P, Brutlag DL: **A genome-wide analysis of CpG dinucleotides in the human genome distinguishes two distinct classes of promoters**. *Proceedings of the National Academy of Sciences of the United States of America* 2006, **103**(5):1412-1417.

267. Ioshikhes IP, Zhang MQ: **Large-scale human promoter mapping using CpG islands**. *Nature genetics* 2000, **26**(1):61-63.

268. Down TA, Hubbard TJ: **Computational detection and location of transcription start sites in mammalian genomic DNA**. *Genome research* 2002, **12**(3):458-461.

269. Travers AA: **The structural basis of DNA flexibility**. *Philosophical transactions Series A, Mathematical, physical, and engineering sciences* 2004, **362**(1820):1423-1438.

270. Rohs R, West SM, Sosinsky A, Liu P, Mann RS, Honig B: **The role of DNA shape in protein-DNA recognition**. *Nature* 2009, **461**(7268):1248-1253.

271. Florquin K, Saeys Y, Degroeve S, Rouze P, Van de Peer Y: **Large-scale structural analysis of the core promoter in mammalian and plant genomes**. *Nucleic acids research* 2005, **33**(13):4255-4264.

272. Abeel T, Saeys Y, Bonnet E, Rouze P, Van de Peer Y: **Generic eukaryotic core promoter prediction using structural features of DNA**. *Genome research* 2008, **18**(2):310-323.

273. Schmid CW: **Alu: structure, origin, evolution, significance and function of one-tenth of human DNA**. *Progress in nucleic acid research and molecular biology* 1996, **53**:283-319.

274. Feschotte C: **Transposable elements and the evolution of regulatory networks**. *Nature reviews genetics* 2008, **9**(5):397-405.

275. Britten RJ: **Cases of ancient mobile element DNA insertions that now affect gene regulation**. *Molecular phylogenetics and evolution* 1996, **5**(1):13-17.

276. Testori A, Caizzi L, Cutrupi S, Friard O, De Bortoli M, Cora D, Caselle M: **The role of Transposable Elements in shaping the combinatorial interaction of Transcription Factors**. *BMC genomics* 2012, **13**:400.

277. Wang J, Bowen NJ, Marino-Ramirez L, Jordan IK: **A c-Myc regulatory subnetwork from human transposable element sequences**. *Molecular bioSystems* 2009, **5**(12):1831-1839.

278. Wang T, Zeng J, Lowe CB, Sellers RG, Salama SR, Yang M, Burgess SM, Brachmann RK, Haussler D: **Species-specific endogenous retroviruses shape the transcriptional network of the human tumor suppressor protein p53**. *Proceedings of the National Academy of Sciences of the United States of America* 2007, **104**(47):18613-18618.

279. Sabo PJ, Kuehn MS, Thurman R, Johnson BE, Johnson EM, Cao H, Yu M, Rosenzweig E, Goldy J, Haydock A *et al*: **Genome-scale mapping of DNase I sensitivity in vivo using tiling DNA microarrays**. *Nature methods* 2006, **3**(7):511-518.

280. Boyle AP, Davis S, Shulha HP, Meltzer P, Margulies EH, Weng Z, Furey TS, Crawford GE: **High-resolution mapping and characterization of open chromatin across the genome**. *Cell* 2008, **132**(2):311-322.

281. Hesselberth JR, Chen X, Zhang Z, Sabo PJ, Sandstrom R, Reynolds AP, Thurman RE, Neph S, Kuehn MS, Noble WS *et al*: **Global mapping of protein-DNA interactions in vivo by digital genomic footprinting**. *Nature methods* 2009, **6**(4):283-289.

282. Crawford GE, Holt IE, Whittle J, Webb BD, Tai D, Davis S, Margulies EH, Chen Y, Bernat JA, Ginsburg D *et al*: **Genome-wide mapping of DNase hypersensitive sites using massively parallel signature sequencing (MPSS)**. *Genome research* 2006, **16**(1):123-131.

283. Giresi PG, Kim J, McDaniell RM, Iyer VR, Lieb JD: **FAIRE (Formaldehyde-Assisted Isolation of Regulatory Elements) isolates active regulatory elements from human chromatin**. *Genome research* 2007, **17**(6):877-885.

284. Song L, Zhang Z, Grasfeder LL, Boyle AP, Giresi PG, Lee BK, Sheffield NC, Graf S, Huss M, Keefe D *et al*: **Open chromatin defined by DNaseI and FAIRE identifies regulatory elements that shape cell-type identity**. *Genome research* 2011, **21**(10):1757-1767.

285. FitzGerald PC, Shlyakhtenko A, Mir AA, Vinson C: **Clustering of DNA sequences in human promoters**. *Genome research* 2004, **14**(8):1562-1574.

286. Tharakaraman K, Bodenreider O, Landsman D, Spouge JL, Marino-Ramirez L: **The biological function of some human transcription factor binding motifs varies with position relative to the transcription start site**. *Nucleic acids research* 2008, **36**(8):2777-2786.

287. Keilwagen J, Grau J, Paponov IA, Posch S, Strickert M, Grosse I: **De-novo discovery of differentially abundant transcription factor binding sites including their positional preference**. *PLoS computational biology* 2011, **7**(2):e1001070.

288. Davis IW, Benninger C, Benfey PN, Elich T: **POWRS: position-sensitive motif discovery**. *PloS one* 2012, **7**(7):e40373.

289. Tharakaraman K, Marino-Ramirez L, Sheetlin S, Landsman D, Spouge JL: **Alignments anchored on genomic landmarks can aid in the identification of regulatory elements**. *Bioinformatics* 2005, **21 Suppl 1**:i440-448.

290. Griffith OL, Montgomery SB, Bernier B, Chu B, Kasaian K, Aerts S, Mahony S, Sleumer MC, Bilenky M, Haeussler M *et al*: **ORegAnno: an open-access community-driven resource for regulatory annotation**. *Nucleic acids research* 2008, **36**(Database issue):D107-113.

291. Lehne B, Schlitt T: **Protein-protein interaction databases: keeping up with growing interactomes**. *Human genomics* 2009, **3**(3):291-297.

292. de Hoon M, Hayashizaki Y: **Deep cap analysis gene expression (CAGE): genome-wide identification of promoters, quantification of their expression, and network inference**. *BioTechniques* 2008, **44**(5):627-628, 630, 632.

293. Ludlow LB, Schick BP, Budarf ML, Driscoll DA, Zackai EH, Cohen A, Konkle BA: **Identification of a mutation in a GATA binding site of the platelet glycoprotein Ibbeta promoter resulting in the Bernard-Soulier syndrome**. *The Journal of biological chemistry* 1996, **271**(36):22076-22080.

294. Ponomarenko JV, Merkulova TI, Vasiliev GV, Levashova ZB, Orlova GV, Lavryushev SV, Fokin ON, Ponomarenko MP, Frolov AS, Sarai A: **rSNP_Guide, a database system for analysis of transcription factor binding to target sequences: application to SNPs and site-directed mutations**. *Nucleic acids research* 2001, **29**(1):312-316.

295. Mu XJ, Lu ZJ, Kong Y, Lam HY, Gerstein MB: **Analysis of genomic variation in non-coding elements using population-scale sequencing data from the 1000 Genomes Project**. *Nucleic acids research* 2011, **39**(16):7058-7076.

296. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT *et al*: **Gene ontology: tool for the unification of biology. The Gene Ontology Consortium**. *Nature genetics* 2000, **25**(1):25-29.

297. Beissbarth T, Speed TP: **GOstat: find statistically overrepresented Gene Ontologies within a group of genes**. *Bioinformatics* 2004, **20**(9):1464-1465.

298. Beisvag V, Junge FK, Bergum H, Jolsum L, Lydersen S, Gunther CC, Ramampiaro H, Langaas M, Sandvik AK, Laegreid A: **GeneTools – application for functional annotation and statistical hypothesis testing**. *BMC bioinformatics* 2006, **7**:470.

299. Boden M, Bailey TL: **Associating transcription factor-binding site motifs with target GO terms and target genes**. *Nucleic acids research* 2008, **36**(12):4108-4117.

300. Osborne CS, Chakalova L, Brown KE, Carter D, Horton A, Debrand E, Goyenechea B, Mitchell JA, Lopes S, Reik W *et al*: **Active genes dynamically colocalize to shared sites of ongoing transcription**. *Nature genetics* 2004, **36**(10):1065-1071.

301. Razin SV, Gavrilov AA, Pichugin A, Lipinski M, Iarovaia OV, Vassetzky YS: **Transcription factories in the context of the nuclear and genome organization**. *Nucleic acids research* 2011, **39**(21):9085-9092.

302. Sanyal A, Lajoie BR, Jain G, Dekker J: **The long-range interaction landscape of gene promoters**. *Nature* 2012, **489**(7414):109-113.

303. Guelen L, Pagie L, Brasset E, Meuleman W, Faza MB, Talhout W, Eussen BH, de Klein A, Wessels L, de Laat W *et al*: **Domain organization of human chromosomes revealed by mapping of nuclear lamina interactions**. *Nature* 2008, **453**(7197):948-951.

304. Dekker J, Rippe K, Dekker M, Kleckner N: **Capturing chromosome conformation**. *Science* 2002, **295**(5558):1306-1311.

305. de Wit E, de Laat W: **A decade of 3C technologies: insights into nuclear organization**. *Genes & development* 2012, **26**(1):11-24.

306. Thomas-Chollier M, Defrance M, Medina-Rivera A, Sand O, Herrmann C, Thieffry D, van Helden J: **RSAT 2011: regulatory sequence analysis tools**. *Nucleic acids research* 2011, **39**(Web Server issue):W86-91.

307. Dreszer TR, Karolchik D, Zweig AS, Hinrichs AS, Raney BJ, Kuhn RM, Meyer LR, Wong M, Sloan CA, Rosenbloom KR *et al*: **The UCSC Genome Browser database: extensions and updates 2011**. *Nucleic acids research* 2012, **40**(Database issue):D918-923.

308. Dowell RD, Jokerst RM, Day A, Eddy SR, Stein L: **The distributed annotation system**. *BMC bioinformatics* 2001, **2**:7.

Appendix

Table A1: List of operations in MotifLab

analyze	Performs one of the analyses listed in Table A2
apply	Applies sliding window functions to Numeric Datasets to smooth the data or to detect peaks, valleys and edges within the track
collate	Creates a new analysis object by collating results from multiple analyses
combine_numeric	Combines multiple Numeric Datasets into one track based on either the minimum, maximum, average, sum or product of the values from all datasets for each position
combine_regions	Combines regions from multiple Region Datasets into one track
convert	Converts a Numeric Dataset into a Region Dataset or vice versa
copy	Creates an identical copy of an existing data object
count	Counts the number of regions that overlap with a sliding window along the sequence
decrease	Subtraction operator. Decreases the value(s) of a numeric data object by a specified amount
delete	Deletes data objects
difference	Compares one data object to another of the same type and reports the differences between them
discriminate	Converts a regular positional priors track into a *discriminative prior*
distance	Returns a new Numeric Dataset where the value of each position is determined by its distance to a specified feature
divide	Division operator. Divides the value(s) of a numeric data object by a specified amount
ensemblePrediction	Performs *ensemble prediction* with a selected method to combine results from multiple motif discovery programs into potentially more reliable predictions
execute	Runs an external data processing program
extend	Extends the size of regions in a Region Dataset in one or both directions
extract	Extracts an individual value or property from a data object
filter	Removes regions that satisfy a given condition
increase	Addition operator. Increases the value(s) of a numeric data object by a specified amount
interpolate	Fills in values missing between discrete non-zero points in a Numeric Dataset

mask	Masks bases in a DNA sequence using either upper- or lowercase, a single specified letter or random bases sampled from a background model
merge	Merges regions that overlap or lie close to each other in the sequence
moduleDiscovery	Performs *de novo* module discovery with a selected method to discover new *cis*-regulatory modules in a set of sequences
moduleScanning	Scans DNA sequences for matches to a set of known modules
motifDiscovery	Performs *de novo* motif discovery with a selected method to discover new motifs and binding sites in a set of sequences
motifScanning	Scans DNA sequences for matches to a set of known motifs
multiply	Multiplication operator. Multiplies the value(s) of a numeric data object by a specified amount
new	Creates a new data object according to specifications
normalize	Rescales the values of a data object from one range to another
output	Outputs data objects to text documents in selected data formats
physical	Estimates different physical properties of the DNA double helix based on the sequence composition within a sliding window
plant	Randomly inserts sites for up to 5 selected motifs or a single module in a set of sequences. Returns the updated DNA sequence and a Region Dataset containing the implanted sites
predict	Creates a new positional priors track using a trained Priors Generator
prompt	Asks the user to provide a value for a data object via an interactive prompt
prune	Removes duplicate regions (identical or similar) from a Region Dataset
rank	Ranks data objects based on entries in Numeric Maps, numeric Analysis columns or internal numeric properties
score	Scans sequences with a single motif or collection of motifs and returns a Numeric Dataset containing the (highest) match score for each position
search	Searches DNA sequences for matches to regular expressions, motif consensus patterns or tandem repeats (direct or inverted)
set	Assignment operator. Sets the value(s) of a numeric data object to a new specified value
statistic	Calculates a statistic (such as maximum or average value in a track) for each sequence in a dataset
threshold	Assigns all entries in a data object that are equal to or above a specified threshold a new value and those below a second value
transform	Transforms each numeric value in a data object according to a selected mathematical function

Table A2: List of analyses that can be performed with the "analyze" operation

benchmark	Evaluates the performance of motif discovery programs by comparing tracks with predicted TFBS against a target track containing the correct answer
compare collections	Compares two collection objects to see if they have any entries in common
compare motif/region occurrences	Counts the number of times each motif or region type appears in one set of sequences and compares this to counts in a second set of sequences. Statistical tests can be used to assess whether some motifs/regions are more frequent in one of the sets
compare motif track to numeric track	Calculates min/max/average statistics for the numeric track across all binding sites for each motif
compare region datasets	Compares two region datasets and calculates statistics based on their overlap
count motif/module/region occurrences	Counts the number of times each motif, module or region type appears in a track. For motif tracks it is also possible to assess the statistical significance by comparing the number of occurrences to an expected frequency
evaluate prior	Evaluates the capabilities of Numeric Datasets to be used as positional priors for predicting target regions
GC-content	Calculates GC-content statistics for DNA sequences
motif collection statistics	Calculates statistics related to motif size, IC-content and GC-content for all motifs in a given collection
motif position distribution	Analyses the positional distribution of each motif in a track to see if they tend to be located in the same place in different sequences
motif regression	Performs regression analysis of motif scores against gene expression (or other sequence related values)
motif similarity	Compares a selected motif to all other motifs using various similarity metrics
numeric dataset distribution	Calculates distribution statistics for a Numeric Dataset or compares the distribution of values inside versus outside specific regions
numeric map correlation	Compares two Numeric Maps to determine if the values for corresponding entries are correlated
numeric map distribution	Calculates distribution statistics for the values in a Numeric Map
region dataset coverage	Calculates the fraction of each sequence which is covered by specific regions
single motif regression	Similar to the "motif regression" analysis but this gives more detailed results for a single motif

Paper I

BMC Bioinformatics

Research article

Open Access

Assessment of composite motif discovery methods

Kjetil Klepper*[1], Geir K Sandve[2], Osman Abul[3], Jostein Johansen[1] and Finn Drablos[1]

Address: [1]Department of Cancer Reasearch and Molecular Medicine, Norwegian University of Science and Technology, Trondheim, Norway, [2]Department of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway and [3]Department of Computer Engineering, TOBB University of Economics and Technology, Ankara, Turkey

Email: Kjetil Klepper* - kjetil.klepper@ntnu.no; Geir K Sandve - sandve@idi.ntnu.no; Osman Abul - osmanabul@etu.edu.tr; Jostein Johansen - j.johansen@ntnu.no; Finn Drablos - finn.drablos@ntnu.no

* Corresponding author

Published: 26 February 2008

BMC Bioinformatics 2008, **9**:123 doi:10.1186/1471-2105-9-123

Received: 26 July 2007
Accepted: 26 February 2008

This article is available from: http://www.biomedcentral.com/1471-2105/9/123

Abstract

Background: Computational discovery of regulatory elements is an important area of bioinformatics research and more than a hundred motif discovery methods have been published. Traditionally, most of these methods have addressed the problem of *single motif discovery* – discovering binding motifs for individual transcription factors. In higher organisms, however, transcription factors usually act in combination with nearby bound factors to induce specific regulatory behaviours. Hence, recent focus has shifted from single motifs to the discovery of sets of motifs bound by multiple cooperating transcription factors, so called *composite motifs* or *cis-regulatory modules*. Given the large number and diversity of methods available, independent assessment of methods becomes important. Although there have been several benchmark studies of single motif discovery, no similar studies have previously been conducted concerning composite motif discovery.

Results: We have developed a benchmarking framework for composite motif discovery and used it to evaluate the performance of eight published module discovery tools. Benchmark datasets were constructed based on real genomic sequences containing experimentally verified regulatory modules, and the module discovery programs were asked to predict both the locations of these modules and to specify the single motifs involved. To aid the programs in their search, we provided position weight matrices corresponding to the binding motifs of the transcription factors involved. In addition, selections of decoy matrices were mixed with the genuine matrices on one dataset to test the response of programs to varying levels of noise.

Conclusion: Although some of the methods tested tended to score somewhat better than others overall, there were still large variations between individual datasets and no single method performed consistently better than the rest in all situations. The variation in performance on individual datasets also shows that the new benchmark datasets represents a suitable variety of challenges to most methods for module discovery.

Background

A key step in the process of gene regulation is the binding of transcription factors to specific *cis*-regulatory regions of the genome, usually located in the proximal promoter upstream of target genes or in distal enhancer regions [1,2]. Each transcription factor recognizes and binds to a more or less distinct nucleotide pattern – a *motif* – thereby regulating the expression of the nearby gene. Determining the location and specificity of each transcription factor binding site in the genome is thus an important prerequisite for reconstructing the gene regulatory network of an organism.

Since establishing these binding sites experimentally is a rather laborious process, much effort has been made to develop methods that can automatically discover such binding sites and motifs directly from genomic sequence data. More than a hundred methods have already been proposed [3], and new methods are published nearly every month. There is a large diversity in the algorithms and models used, and the field has not yet reached agreement on the optimal approach. Most methods search for short, statistically overrepresented patterns in a set of sequences believed to be enriched in binding sites for particular transcription factors, such as promoter sequences from coregulated genes or orthologous genes in distantly related species.

In higher organism, however, transcription factors seldom function in isolation, but act in concert with nearby bound factors in a combinatorial manner to induce specific regulatory behaviours. A set of binding motifs associated with a cooperating set of transcription factors is called a *composite motif* or *cis-regulatory module*. In recent years, the field of computational motif discovery has therefore shifted from the detection of single motifs towards the discovery of entire regulatory modules.

The diversity of approaches to module discovery is even greater than for single motif discovery, and methods vary widely in what they expect as input and what they provide as output. For instance, methods like Co-Bind [4], LOGOS [5] and CisModule [6] expect only a set of coregulated or orthologous promoter sequences as input and are able to infer both the location and the structure of modules with few prior assumptions regarding their nature. These programs infer an internal model that includes a representation of each individual transcription factor binding motif as well as constraints on the distances between them. On the other hand, programs such as LRA [7] and Hexdiff [8] demand as input a collection of already known module sites to serve as training data. The known positive sites are used along with negative sequence examples to build a model representation which can then be compared to new sequences in order to identify novel module instances. Searching for new matches to a previously defined model might be considered a special case of module discovery and is often referred to as module *scanning*. Programs that specialize in searching for modules this way without inferring the models themselves include ModuleInspector [9] and ModuleScanner [10]. The general problem of module discovery, however, usually involves inferring both a model representation of the modules and to find their locations in the sequences.

Most module discovery methods require users to supply a set of candidate single motif models in the form of IUPAC consensus strings or position weight matrices (PWM) [11]. These are used to discover putative transcription factor binding sites in the sequences, and the programs then search for significant combinations of such binding sites to report as modules.

What constitutes a significant combination varies between methods. MSCAN [12], for instance, searches for regions within sequences that have unusually high densities of binding sites, more so than would be expected from chance alone. The types of the binding motifs are irrelevant, however, and each potential module instance is analyzed independently from the rest. Other tools, like ModuleSearcher [10], Composite Module Analyst [13] and CREME [14], search for specific combinations of motifs that co-occur multiple times in regulatory regions of related genes.

With an increasing number of programs available, both for single and composite motif discovery, there is a growing need among end users for reliable and unbiased information regarding the comparative merits of different approaches. A few independent investigations have been undertaken to assess the performance of selected single motif discovery methods, for instance by Sze *et al*. [15] and Hu *et al*. [16]. The most comprehensive benchmark study to date was carried out by Tompa *et al*. and included thirteen of the most popular single motif discovery methods [17]. The authors of this study also provided a web service to enable new methods to be assessed and compared to the original methods using the same datasets.

However, in spite of the increased interest in regulatory modules, we are not aware of any similar independent benchmarking efforts that have been undertaken with respect to composite motif discovery.

Results

We have developed a framework for assessing and comparing the performance of methods for the discovery of composite motifs. Sequence sets containing real, experimentally verified modules are made available for download through our web service, and users can test programs

of their own choice on these datasets and submit the results back to the web service to get the predictions evaluated. Results are presented both as tabulated values and in graphical format, and performances of different methods can be compared. Since most module discovery tools require users to input candidate motifs, each sequence dataset is supplemented by a set of PWMs capable of detecting the binding sites involved in the modules. To test how programs respond to varying levels of noise in the PWM sets, we created extended PWM sets for one of our datasets where the genuine matrices were mixed with various decoy matrices.

Scoring predictions

We adopted a simple and general definition of a module: a *module* is a *cis*-regulatory element consisting of a collection of single binding sites for transcription factors. A module is thus characterized by only two aspects in our framework: its *location* in a sequence and its *composition*, that is, the set of transcription factor binding motifs involved. A module's location is further defined as the smallest contiguous sequence segment encompassing all the single binding sites in the module, including also the intervening bases. For our purpose, the composition of a module is represented by a set of PWM identifiers. Different modules that share the same composition are said to belong to the same *module class*. Module class definitions may also be limited by structural *constraints*. These are rules governing, among others, the strand bias, order and distances between the transcription factor binding sites of modules of the same class. Since it requires a substantial effort to determine these constraints experimentally, this kind of information is available for a very limited number of classes. Few methods also report such module constraints explicitly. Consequently, we have chosen not to consider this aspect of modules further in our framework, at least for the time being.

Module discovery programs are requested to predict both the location of modules and to identify the motifs involved by naming the proper PWMs. However, not all programs are able to perform both these tasks. The MCAST program [18], for instance, only reports the location of predicted modules, even though it uses a set of PWMs to detect single binding sites internally. On the other hand, programs that discover single motifs *de novo* without relying on pre-constructed matrices have, of course, no way of correctly naming the motifs involved. Methods like that of Perco *et al.* [19] and GCMD [20] identify modules by looking for groups of PWMs whose binding sites consistently appear together in multiple sequences, but disregard any further information about the precise position of these sites. Hence, such programs only report the composition of modules but not their location. By assessing the location and composition aspects of modules separately, our framework can equally well be used with programs that predict only one or the other.

To measure prediction accuracy of methods with respect to module location, we have used the *nucleotide-level correlation coefficient* (*nCC*). This statistic has been widely used before, among others, for coding region identification and gene structure prediction [21]. It was also adopted by Tompa *et al.* to evaluate binding site predictions in their single motif discovery benchmark study. The value of *nCC* lies in the range -1 to +1. A score of +1 indicates that a prediction is coincident with the correct answer; whereas a score of -1 means that the prediction is exactly the inverse of the correct answer. Random predictions will generally result in *nCC*-values close to zero.

$$nCC = \frac{TP \cdot TN - FN \cdot FP}{\sqrt{(TP+FN)(TN+FP)(TP+FP)(TN+FN)}}$$

Here, *TP* is the number of nucleotides in a sequence that are correctly predicted by a program as belonging to a module, while *TN* is the number of nucleotides correctly identified as background. *FN* is the number of true module nucleotides incorrectly classified as background, and *FP* is the number of background nucleotides incorrectly classified as belonging to a module.

A similar statistic, the *motif-level correlation coefficient* (*mCC*), was used to evaluate prediction accuracy with respect to module composition. The definition of *mCC* follows that of *nCC*, except that instead of counting the number of nucleotides, we count the number of single motifs (or PWMs) correctly or incorrectly classified as being part of a module or not. Hence, for *mCC*, *TP* is the number of PWMs correctly identified as constituents of the module, while *FP* is the number of PWMs incorrectly predicted as being part of a module. Note that the correlation statistics, as defined here, are only applicable when both the datasets and the predictions made by a program contain a combination of module and non-module instances, if not, the divisor will be zero and the value of the statistic will be undefined. Consequently, the *mCC*-score is only informative when the set of PWMs supplied to a module discovery program contains false positives, i.e. additional matrices besides those that are actually involved in the modules. Final scores for each dataset are obtained by summing up *TP*, *FP*, *TN* and *FN* over all sequences before calculating the correlation scores. If no module predictions are made on a set of sequences, the resulting scores for *nCC* and *mCC* are assigned a value of zero rather than being left undefined. In addition to *CC* scores, several other statistics mentioned in [17] such as *sensitivity*, *specificity*, *positive predictive value*, *performance*

coefficient (phi-score) and *average site performance* are calculated for both nucleotide- and motif-level.

Datasets

We compiled three datasets from sequences containing experimentally verified regulatory modules. The first and the last two datasets have different characteristics and were chosen to complement each other to test methods under different conditions.

Our main dataset was based on annotated composite motifs from the TRANSCompel database [22]. The modules selected for this dataset are small, each consisting of exactly two single binding sites for different transcription factors (TFs), but we specifically chose modules that had multiple similar instances in several sequences. Sequences containing modules from the same class were grouped together producing ten sequence sets named after their constituent single motifs as shown in Table 1. Each of the sequences in a set contained at least one copy of the module with the same two motifs, but the order, orientation and distance between the TFBS could vary between sequences. Separate PWM collections, with matrices for the two single motifs involved, were constructed for each of the sequence sets. All in all there were eleven distinct single TF binding motifs in our full TRANSCompel dataset, and PWMs representing these motifs were collected from the companion TRANSFAC database [22]. Since TRANSFAC often contains several different PWMs for each motif, we grouped all the matrices corresponding to a particular motif into an equivalence set, essentially treating these PWMs as if they were one and the same with respect to prediction and scoring. In addition to the TRANSFAC matrix sets, we also constructed eleven custom matrices that were specifically tailored to the particular motifs and binding sites present in the sequences (see Methods). Assessment of module discovery programs on the TRANSCompel dataset was conducted using both the TRANSFAC sets and the customized PWM sets independently. The motivation for using two different PWM sets was to test the stability of methods and examine how the specific representations used for single motifs might influence the ability of methods to find the correct modules.

The two last datasets were based on combinations of TFBS found in the regulatory regions of genes specifically expressed in liver [23] and muscle [7] cells. The modules here are usually larger compared to the TRANSCompel modules, containing up to nine binding sites for four different motifs in the liver regulatory regions and up to eight sites for five motifs in the muscle regions. PWMs for these motifs were taken from the respective publications. The composition of the modules in these two datasets is variable; modules can contain multiple binding sites for the same motifs and not all motifs are present in every module.

While most programs require candidate PWMs to be entered, this can pose a problem for users who might not always know in advance the kind of modules that should be present in a sequence or which transcription factors that might bind. It could be the case, for instance, that a researcher has only a set of promoters from a coregulated set of genes and is interested in identifying the hitherto unknown module that controls the common expression of these genes. A popular strategy then is to employ an excessive set of PWMs which, hopefully, also includes the appropriate matrices. An extreme, but not unlikely, scenario would be to use all the matrices available from a published compilation like TRANSFAC (774 matrices in release 9.4) or Jaspar [24] (123 core matrices). Although this approach will inevitably lead to lots of false positive PWM matches that might thwart the module discovery process, good module discovery tools should nonetheless be able to report the true module instances without simultaneously predicting too many spurious occurrences.

Table 1: Datasets

Sequence set	Sequences	Modules	Total size (bp)	Module size, min-max (avg)
AP1-Ets	16	17	14860	14 – 99 (27)
AP1-NFAT	8	11	6893	14 – 19 (16)
AP1-NFκB	7	8	6532	18 – 135 (53)
CEBP-NFκB	8	8	7308	44 – 118 (84)
Ebox-Ets	4	6	3489	16 – 50 (25)
Ets-AML	5	5	4053	13 – 30 (19)
IRF-NFκB	6	6	5344	23 – 71 (43)
NFκB-HMGIY	6	7	5393	10 – 32 (13)
PU1-IRF	5	5	4530	12 – 14 (13)
Sp1-Ets	7	8	5787	16 – 117 (37)
Liver	12	14	11943	26 – 176 (112)
Muscle	24	24	20427	14 – 294 (120)

A brief overview of the ten TRANSCompel sequence sets and the liver and muscle datasets used in the assessment. Further information can be found in Additional File 1.

To simulate these conditions and test methods' response to noisy PWM sets, each PWM set under the TRANSCompel dataset was issued in multiple versions with progressively more decoy matrices added to the set of true annotated motifs. Decoy matrices were randomly sampled from the complete TRANSFAC compilation after removing the matrices corresponding to the true motifs for a sequence set. Decoy sets are available at 50%, 75%, 90%, 95% and 99% levels, where the percentage number relates the amount of decoy matrices in the set. Thus, a custom PWM set at the 90% level includes 2 genuine matrices and 18 decoy matrices. The number of decoy matrices in the TRANSFAC PWM sets varies with each module class but is always higher than for the custom sets at the same percentage level. Information on the exact number of PWMs in each set is available in Additional File 1. The 99% sets include as decoys all of the matrices from TRANSFAC which do not correspond to the correct motifs. They are called "99%" for consistency, although the actual percentage of decoys ranges between 95% and 99% depending on the module class. To avert artefacts stemming from possibly biased selections of decoys, all decoy sets (except at the 99% level) consist of ten independently sampled decoy collections, and the final correlation statistics for a decoy level are calculated by averaging prediction scores made from using each collection in turn. This also means that variation due to any stochastic nature of algorithms will be averaged over ten independent runs.

Benchmark of module discovery methods

Using our assessment framework, we benchmarked eight published methods for module discovery: *CisModule* [6], *Cister* [25], *Cluster-Buster* [26], *Composite Module Analyst (CMA)* [13], *MCAST* [18], *ModuleSearcher* [10], *MSCAN* [12] and *Stubb* [27]. See Table 2 for brief descriptions of each of these methods. CisModule, CMA and ModuleSearcher process all the sequences in a dataset simultaneously and look for instances of similar modules across multiple sequences. The other methods examine the sequences individually, although Stubb considers multiple instances of similar modules within the same sequence. Except for MCAST, which does not report module composition, all the programs report both the location and composition of modules. CisModule, however, predicts modules *de novo* without relying on supplied PWM sets and so does not name the single motifs involved the way we require. Hence, motif-level scores were not calculated for MCAST and CisModule. Cluster-Buster and MCAST report the full module segments, while the rest of the methods list the positions of the PWM hits in the modules. In these cases we extracted the start position of the first reported binding site and the end position of the last binding site and used these as the boundaries of a module prediction.

We generally relied on default parameter settings for all programs. However, since choosing the proper parameter values can sometimes prove crucial for a method's performance, we decided to provide the programs with a few general clues where applicable; specifically, that the size of modules should not exceed 200 bp (300 bp in the muscle dataset) and that the modules should consist of exactly two single binding sites for different TFs in the TRANSCompel dataset but possibly up to ten binding sites for four and five different TFs on the liver and muscle sets respectively. Furthermore, binding sites could potentially overlap and the composition of the modules in liver and muscle sets should be allowed to vary between sequences.

Figures 1a and 1b show the resulting nucleotide-level correlation scores on each sequence set in the TRANSCompel dataset when methods were supplied with TRANSFAC matrices and custom matrices respectively. The scores vary widely between individual sequence sets but are generally fairly well correlated between methods, so that most methods tend to get high (or low) scores on the same sets. The notable exception is CisModule which performs poorly on all sequence sets. The correlation suggests that some sequence sets are inherently more easy (or difficult) to tackle than others. Scores for CEBP-NFκB and IRF-NFκB are the highest overall. The reasons why these sets are generally easy to predict might be that their modules are quite long and the matrices representing the single binding motifs have high information content (see Table 3 and Additional File 1). Conversely, the short size of the modules and the low information content of PWMs for AP1-NFAT would make this a hard sequence set. We also calculated combined scores for the whole TRANSCompel dataset which are shown in the inset legends of Figure 1 and graphically in Figure 2. These combined scores were obtained by summing up TP, TN, FP, FN over all sequence sets when calculating the score measures. The highest combined *nCC* scores achieved were 0.388 with the TRANSFAC matrices (MSCAN) and 0.38 with custom matrices (MCAST). The average performances across all methods were also about the same with the two PWM sets. Some methods performed quite differently depending on the PWMs, however. For instance, MCAST scored much better using custom matrices than with TRANSFAC matrices, while MSCAN and Cluster-Buster did a better with job with TRANSFAC. The rank order of methods is thus somewhat altered between the two cases. Still, some tendencies remain: CMA, Cluster-Buster, MCAST, ModuleSearcher and MSCAN occupy the top five positions in both cases, followed by Cister and Stubb and then finally CisModule which consistently scored lowest.

Figure 3 shows the results of mixing the PWM sets with an equal proportion of decoy matrices. The addition of decoy PWMs leads to a drop in score values for almost all meth-

Table 2: Description of module discovery tools

CisModule	CisModule models the structure of sequences with a two-level hierarchical mixture-model and uses a Bayesian approach with Gibbs sampling to simultaneously infer the modules, TFBSs and PWMs based on their joint posterior distribution, which is the probability of a model given the input sequence set. At the first level, sequences are viewed as a mixture of module instances and background. At the second level, modules are modelled as a mixture of motifs and inter-module background. Parameters of the model include the widths and representations (PWMs) of single motifs and parameters related to distances between modules and between TFBS within modules. From a random initialization, CisModule iteratively cycles through steps of parameter update and module-motif detection. New parameter values are sampled from their conditional posterior distributions based on the currently predicted modules and motifs, and new predictions of modules and TFBSs are then sampled based on these updated parameter values. Positions in the sequences where the marginal posterior probability of being sampled within modules was greater than 0.5 were output as module predictions.
Cister	Given a set of PWMs and parameters specifying the expected number of motifs in modules, the expected distances between motifs in modules and the expected distance between modules, *Cister* builds a Hidden Markov Model (HMM) with three basic states: *motif*, *intra-module background* and *inter-module background*. Transition probabilities between these states follow geometric distributions according to the expected values input by the user. In the motif state, one of the PWMs is chosen uniformly at random and used to decide the probabilities of outputting nucleotides. Background-state emission probabilities are estimated from a sliding window centered on the current base in the query sequence. From this HMM, the posterior probability that each base in the input sequence was generated from a module state as opposed to the inter-module state can be calculated. Predicted modules are defined to occur at local maxima of this posterior probability curve where the value is at least 0.5 and no larger value is observed within 1200 bp.
Cluster-Buster	Cluster-Buster is developed by the same group that made *Cister* and is designed to search for clusters of pre-specified motifs in nucleotide sequences. Like Cister, Cluster-Buster constructs a HMM-model based on the user-supplied PWMs, an expected distance between motifs in clusters and background distributions estimated from the input sequence over sliding windows. Log likelihood ratios are used to determine whether a sequence is more likely to be generated by a "cluster-model" or a "background-model". Cluster-Buster uses a linear time heuristic to rapidly estimate log likelihood ratios for all subsequences of the input sequence and outputs those subsequences with ratios above a specified threshold that do not overlap with other higher scoring subsequences.
Composite Module Analyst (CMA)	The promoter model in CMA is expressed as a Boolean combination of one or more *composite modules* (CM), each of which consist of a set of single, independent motifs as well as pairs of motifs that must obey certain constraints on distance and orientation. Given a candidate promoter model, the method searches for potential matches to the CMs in the sequences, and a final promoter score is calculated after the presence or absence of each CM is established. CMA employs a Genetic Algorithm to search for the promoter model which best discriminates between a set of positive (co-regulated) and a set of negative sequences. The fitness function is based on a linear combination of several properties of the distribution of the promoter scores and of the individual CM scores in the two sequence sets.
MCAST	MCAST builds a HMM-model consisting of an intra-module state, an inter-module state and motif-states based on the supplied PWMs. The score for a motif-state is called a *p*-score and is the negative logarithm of the *p*-value of a log-odds score based on the probability of a segment in the target sequence being generated either by the PWM or a fixed, user-specified zero-order Markov background model. MCAST forbids transitions into motif-states that result in *p*-scores lower than some chosen threshold. Some state transitions are associated with certain costs. For instance, entering the inter-module state from a motif-state incurs a large one-time penalty while cycling through the intra-module state incurs smaller penalties for each nucleotide emitted. The Viterbi algorithm is used to find the highest scoring path through the HMM with respect to the input sequence, classifying each position in the sequence as either belonging to a module or to the background. Potential module segments are scored according to the number of motifs in the module and the *p*-scores of these motifs and are penalized by the number of intra-module background bases. Finally, modules are ranked according to the estimated *E*-values of these scores.
ModuleSearcher	Given a list of PWM hits with match scores for putative TFBSs in a sequence set, ModuleSearcher finds the module model (set of *k* PWMs) that best fits the sequences. The score of a module model is calculated as the sum of scores over all sequences, and the score function for a single sequence is based on the best scoring set of TFBSs in the sequence that corresponds to the PWMs in the module model. To be considered a valid TFBS set the binding sites must all lie within a short window, and the user can choose to ignore TFBS sets with overlapping binding sites or penalize sets that lack sites for some PWMs. An A*-algorithm (or alternatively a Genetic Algorithm) is employed to search the space of possible subsets of *k* motifs from the full PWM library in order to find the highest scoring module model.
MSCAN	MSCAN discovers modules by evaluating the combined statistical significance of sets of potential non-overlapping TF binding sites in a sliding window along the input sequence. PWMs are compared against each position within the window to obtain match scores, and *p*-values are calculated as the probability of obtaining similar or higher scores at a specific position in a random sequence with nucleotide distribution similar to the distribution in the window. MSCAN proceeds by calculating significance scores for all combinations of up to *k* binding sites in the window and then selects the optimal combination (the one with the lowest score). A prediction is output if a final *p*-value computed from this score is less than some user-specified threshold.

Table 2: Description of module discovery tools *(Continued)*

Stubb	The HMM used by Stubb consists of motif states based on supplied PWMs and a single background state based on a *k*th-order Markov model with probability distribution estimated from a sliding window. The scoring function is the log likelihood ratio that the sequence within a limited window was more likely generated by the full model than with a HMM consisting of only the background state. Unlike the other HMM methods presented here, the transition probabilities between states in Stubb are not based on user-input expectancies, but are estimated from the sequence using the Baum-Welch algorithm. This procedure finds the set of transition probabilities that maximizes the scoring function. If Stubb finds that some motifs are highly correlated with respect to order, it can make use of *correlated transition probabilities*. This means that the probability of entering a specific motif state will dependent on which previous motif was output. Stubb can also utilize phylogenetic comparisons between sequences from multiple species to highlight potentially regulatory modules.

The table contains short descriptions of the eight methods included in the assessment. All methods except for CisModule rely on supplied PWMs and consider matches on both strands, usually with equal probability (however, Stubb can estimate strand biases for all PWMs in a preprocessing step). Not all methods are able to consider overlapping single binding sites, which do occur in a few modules.

ods. The drop is greater for the TRANSFAC PWMs, presumably because these sets contain more genuine matrices and therefore also more decoys. Contrary to expectation, some methods actually score slightly better on certain sequence sets when decoys are in use. Examples are Cister on Ets-AML and Stubb on Ebox-Ets with custom matrices. One explanation for this could be that these methods make use of decoy motifs that just happen to have a high degree of overlap with genuine modules. To examine whether the modules are predicted with the correct motifs or not, we can look at the corresponding motif-level correlation scores as shown in Figure 4. The generally high mCC scores obtained for IRF-NFκB support the notion that this is an easy sequence set, while the difficulty for most methods in selecting the correct motifs for CEBP-NFκB explains the higher drop seen in nCC for this set when decoys were added. CMA and ModuleSearcher are by far the best methods at predicting the correct composition of modules with both TRANSFAC and custom PWMs as input, although CMA does perform notably poor on two specific sequence sets. The mCC score for the third best method, Cluster-Buster, is less than half of that of ModuleSearcher.

Figures 5 and 6 show score tendencies as increasingly more decoys are added to the PWM sets. The nucleotide-level performances of CMA and ModuleSearcher are only slightly affected by the larger amounts of decoys, whereas the scores for the other methods steadily decline. At the motif-level we clearly see a division into two groups with

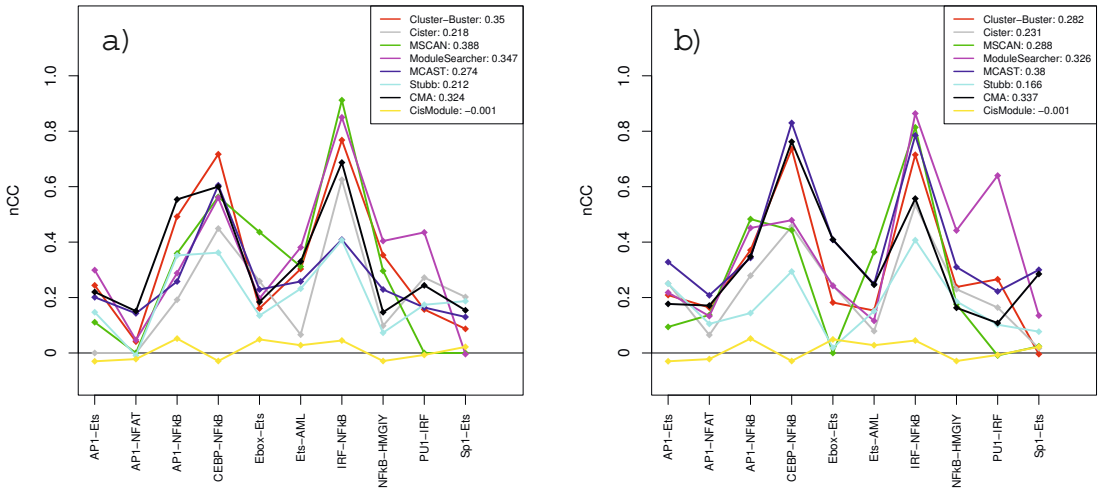


Figure 1
Nucleotide-level correlation scores on the TRANSCompel dataset. The graphs show *nCC* scores for each of the ten sequence sets in the TRANSCompel dataset when methods are supplied with TRANSFAC PWM sets (a) and custom matrices (b).

Table 3: Correlations between dataset properties and *nCC* scores

	TRANSFAC PWMs		Custom PWMs	
	Average *nCC*	Highest *nCC*	Average *nCC*	Highest *nCC*
Number of sequences	-0.23	-0.16	-0.23	-0.05
Length of shortest sequence	0.30	0.18	0.30	0.13
Average sequence length	0.40	0.33	0.42	0.43
Total sequence set length	-0.19	-0.12	-0.18	-0.02
Number of module instances	-0.38	-0.32	-0.40	-0.19
Size of smallest module	**0.61**	**0.69**	**0.67**	**0.73**
Size of largest module	0.26	0.34	0.19	0.35
Average module size	**0.60**	**0.68**	0.59	**0.70**
Module size standard deviation	0.23	0.29	0.13	0.29
IC-content (lowest)	0.46	0.45	**0.73**	0.47
IC-content (total)	**0.75**	**0.73**	**0.78**	0.54
Module/background-ratio	0.53	0.61	0.51	**0.63**

We conducted a simple correlation analysis to examine which properties of the TRANSCompel sequence sets and PWMs correlated best with the highest and average *nCC* scores obtained by the methods on these sets. "IC-content (lowest)" is the *information content* (IC) of the PWM with the lowest IC of the two involved in each sequence set. The information content of a PWM is inversely related to the amount of variability in the binding patterns from which the PWM is constructed [38]. PWMs with higher information content are more specific and match only sites with a high degree of similarity to the consensus motif. "IC-content (total)" is the sum of IC-contents for the two motifs (for TRANSFAC PWMs we used the PWM with the highest IC in each equivalence set to represent the motif). The three highest values are highlighted in each column. The properties that seem to correlate best with methods' performances are the minimum and average size of modules (in basepairs) and the total IC-content, which would imply that module discovery is harder for datasets containing short and degenerate modules.

CMA and ModuleSearcher performing significantly better than the rest. Additional graphs detailing the effects of added noise with respect to each individual sequence set and the variations due to different decoy selections can be found at our web site.

Results for the liver and muscle datasets are shown in Figures 7 and 8. For these datasets we supplied only four liver- and five muscle-PWMs respectively, and no decoy matrices were used. Since the modules in these datasets do not necessarily include binding sites for all of these motifs however, we could calculate motif-level scores by treating the PWMs for the missing motifs as false instances. All methods, except CisModule, did a better job on locating the modules in the liver dataset than in the TRANSCompel dataset. Cluster-Buster scored highest, but Stubb

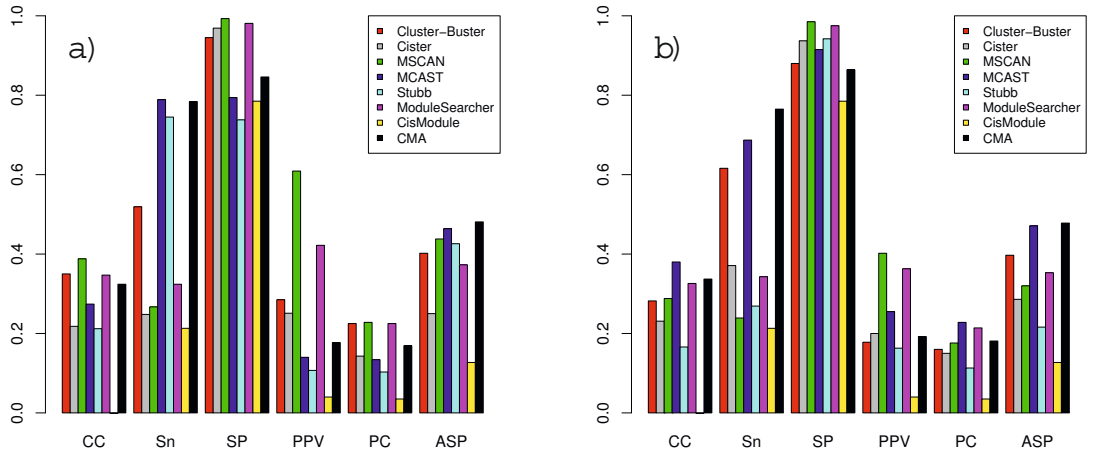


Figure 2
Combined performance scores on the full TRANSCompel dataset. Combined nucleotide-level scores obtained for different performance measures when using TRANSFAC PWM sets (a) and custom matrices (b).

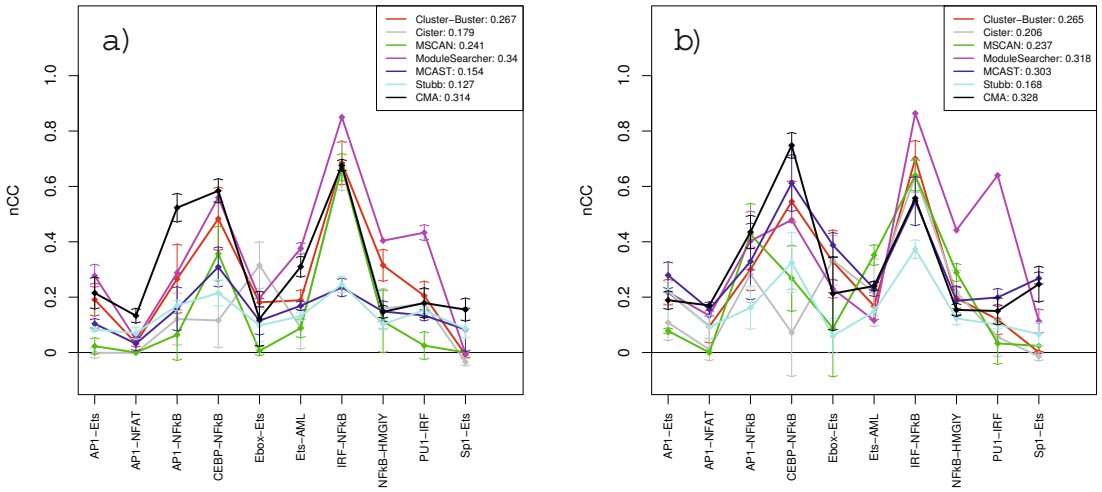


Figure 3
Nucleotide-level correlation scores with 50% noise in the PWM sets. The graphs show *nCC* scores when using TRANSFAC PWM sets (a) and custom matrices (b) with an equal proportion of decoy matrices added. Each value represents the average score over ten runs with different decoy selections.

showed the largest improvement in *nCC* score. The motif-level scores, on the other hand, were not very high, which can most likely be attributed to overprediction of motifs in the case of CMA and underprediction for MSCAN. Results on the muscle dataset display the same main tendencies as the other two datasets, but for the first time,

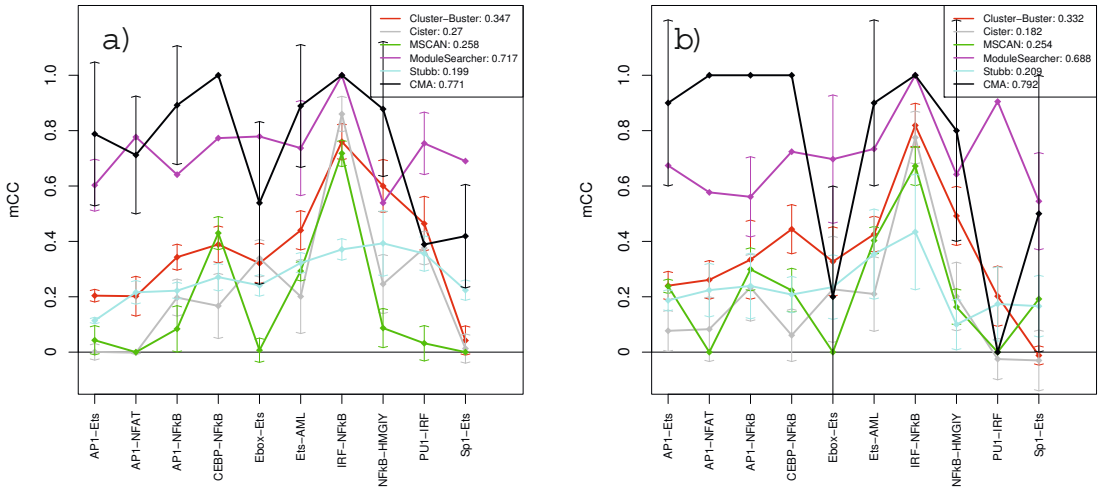


Figure 4
Motif-level correlation scores with 50% noise in the PWM sets. The graphs show *mCC* scores when using TRANSFAC PWM sets (a) and custom matrices (b) with an equal proportion of decoy matrices added. Each value represents the average score over ten runs with different decoy selections.

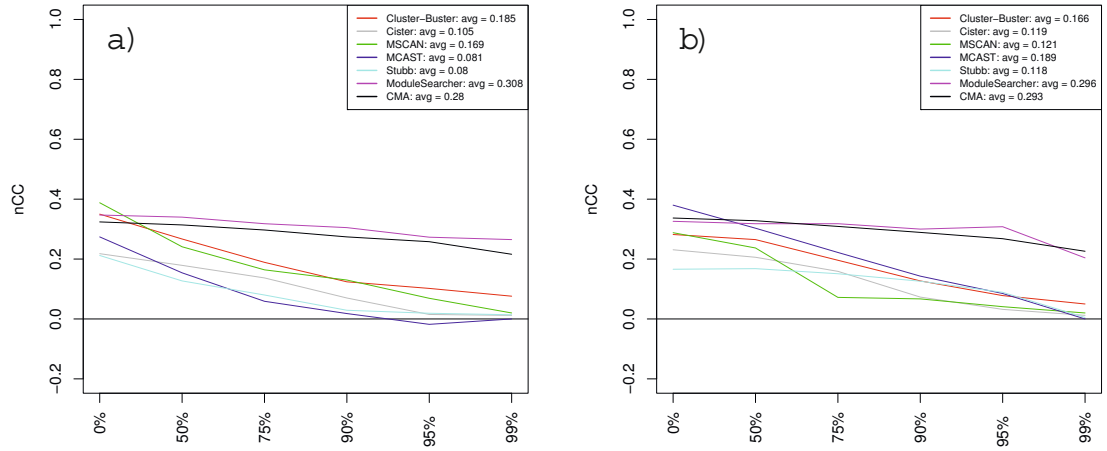


Figure 5
Nucleotide-level correlation scores at different noise levels. Plot of *nCC* scores at increasing noise levels when methods are supplied with TRANSFAC PWM sets (a) and custom matrices (b). Scores shown are averages over all sequence sets and decoy selections at each noise level. MCAST was unable to function properly with very large PWM sets and was therefore assigned a score of zero at the 99% level.

CisModule obtains an *nCC* score above zero and actually bypasses one the other methods.

Discussion

Objective benchmarking efforts are important for providing unbiased reviews of published methods and for establishing the methodological frontier with respect to bioinformatics techniques. In this study we wanted to explore benchmarking in the context of module discovery and to investigate related design issues such as dataset construction and performance evaluation.

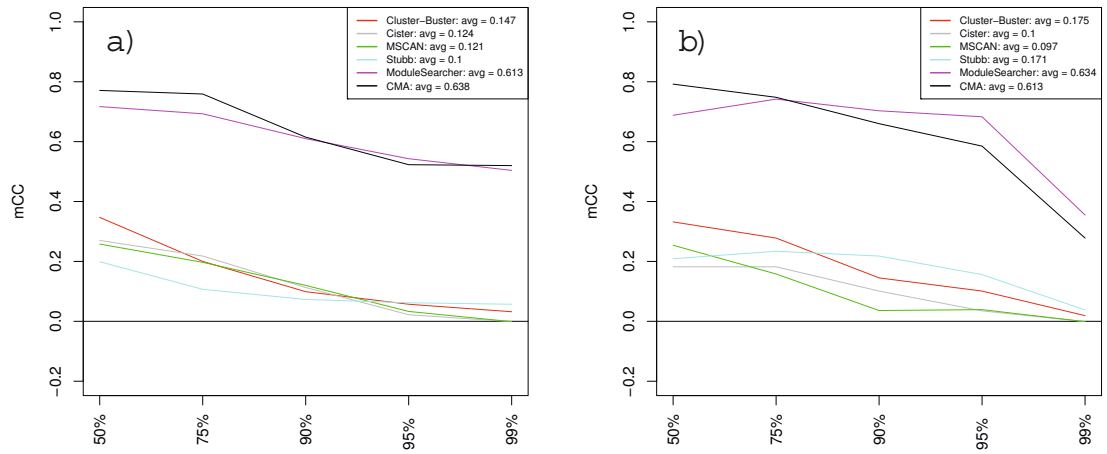


Figure 6
Motif-level correlation scores at different noise levels. Plot of *mCC* scores at increasing noise levels when methods are supplied with TRANSFAC PWM sets (a) and custom matrices (b). Scores shown are averages over all sequence sets and decoy selections at each noise level.

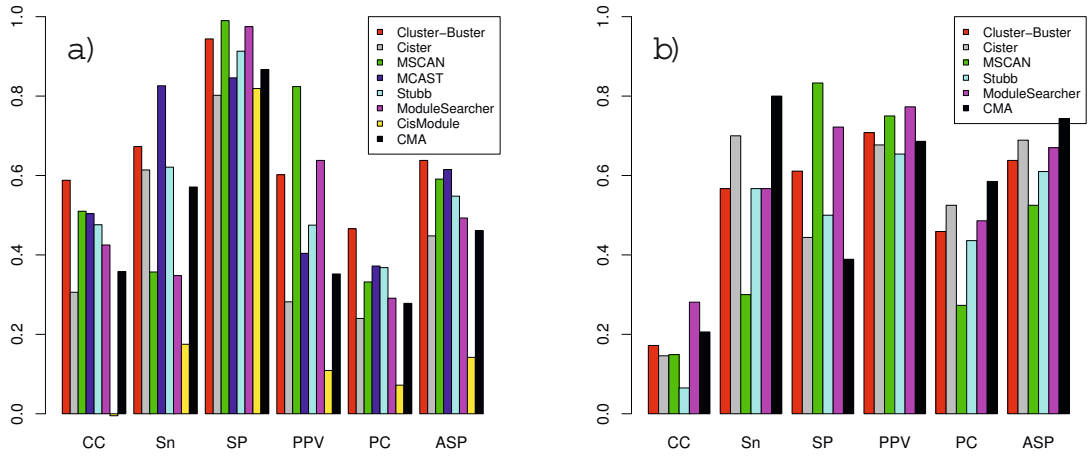


Figure 7
Performances on the liver dataset. Scores obtained on the liver dataset for different performance measures at nucleotide-level (a) and motif-level (b).

Benchmarking of tools for composite motif discovery is harder than benchmarking of single motif discovery tools, since the former methods are more diverse with respect to input requirements and the type of predictions they make. We have aimed at creating a simple and general framework that can be used with a wide range of methods. Nevertheless, we do not provide every kind of information that programs might ask for, and not all module discovery tools can be fairly assessed with our system.

To construct the benchmark datasets we relied on real genomic sequences containing experimentally verified modules, rather than creating synthetic datasets with fabricated and planted modules. The motivation for only

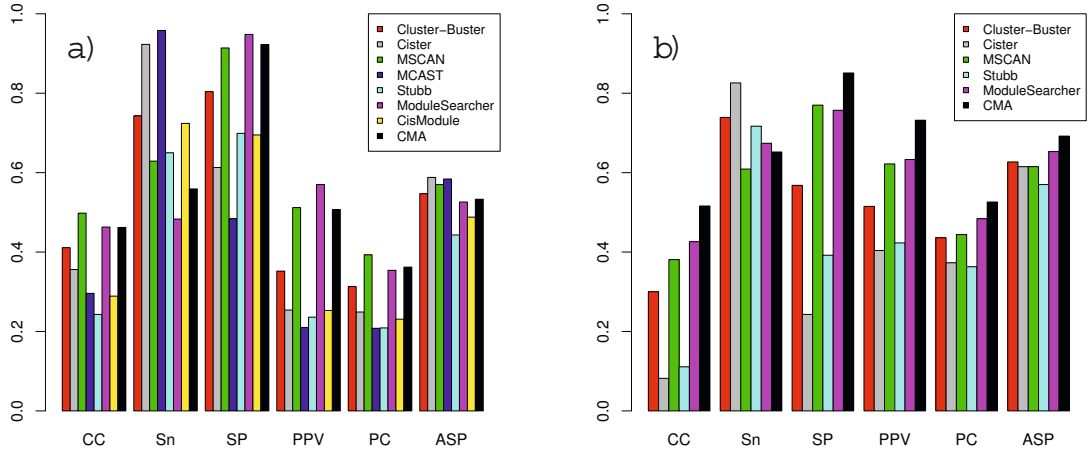


Figure 8
Performances on the muscle dataset. Scores obtained on the muscle dataset for different performance measures at nucleotide-level (a) and motif-level (b).

using real data was to avoid introducing artificial bias related to the composition and constraints of modules. Good benchmark datasets should be diverse enough to discriminate the behaviour of different methods, when possible, and provide them with a wide range of realistic challenges. For module discovery these challenges could include discovering modules with few or many single motifs, tightly clustered or widely spaced motifs and modules with highly conserved or degenerate binding sites. Ideally, benchmark datasets should also be novel to the methods tested. Currently the amount of experimental data available is too limited to achieve all of these goals. The particular dataset we have constructed based on TRANSCompel data is novel in terms of performance testing. The modules in TRANSCompel are short, however, and to include larger modules we were forced to rely on a few well-known datasets from liver and muscle regulatory regions that have been used extensively in the past for testing and possibly for designing and developing module discovery methods. Some methods might therefore be intrinsically biased to perform well on these sets. It is conspicuous, for instance, that CisModule – which was tested with muscle data in its original publication – scored comparably well to the other methods on our muscle set, yet close to zero on both the TRANSCompel and liver datasets.

We chose the *correlation coefficient* as our main statistic for evaluating and comparing module discovery methods because it captures aspects of two of the most commonly used performance measures – *sensitivity* and *specificity* – into a single score value. However, since different statistics often favour different methods, it is prudent to consider several measures to get a better comprehension of each method's qualities. The sensitivity measure (Sn), for instance, tells us to what extent a method's predictions include the true module instances. At the nucleotide level, MCAST seems overall to be the most sensitive method among those tested here, while CMA shows high sensitivity on the TRANSCompel dataset. Yet, to achieve these high sensitivity scores the methods at the same time make a lot of false positive predictions, as can be seen from the lower *positive predictive values* (PPV). MSCAN and ModuleSearcher, on the other hand, generally have the highest nucleotide-level PPV scores, which tells us that the positive predictions made by these two programs are more trustworthy than predictions made by the other programs.

PWMs from the TRANSFAC database were used to represent both the true motifs and the decoys for the TRANSCompel dataset. A potential problem when using TRANSFAC is that many of the matrices are quite similar to each other [28]. This is partly due to some TFs being represented by several PWMs, but also because different TFs might bind to similar-looking motifs. As a result, module discovery programs can be unduly penalized for selecting an incorrect PWM at the motif level, even though the predicted PWM is very similar to the target. We have tried to remedy this situation by grouping together PWMs that correspond to the same TFs and consider these as the same motif with respect to scoring. However, there might still be other matrices in the decoy sets that can match with the annotated binding sites.

Since we are using real genomic sequences, some of the predicted modules that we label as false positives can in fact represent unannotated true positives, and so the actual performance of methods might very well be better than indicated, especially at high noise levels.

It should be noted that while the annotated length of a TF motif may vary from binding site to binding site, the length of a standard PWM is fixed, and PWMs do not always match the locations of their corresponding binding sites precisely. Perfect *nCC* scores can therefore be difficult or even impossible to obtain. The *nCC* score also drops fast if a method predicts a larger module region than what is annotated, even though the target module is correctly covered by the predicted region. This can severely penalize methods that tend to predict large module regions, especially on the TRANSCompel dataset where most modules are rather short.

Some programs can utilize additional information to strengthen confidence in predictions and improve their performance. For instance, Stubb is a sensitive method and the predictions it makes usually include the correct modules, especially when using large PWM sets; yet, its *CC*-scores are generally low because it simultaneously predicts a lot of false positives. Stubb can employ a phylogenetic footprinting [29] strategy to filter out many of these false predictions, but it requires that orthologous sequences from related species are supplied along with the regular sequences. However, in order to make the tests as comparable as possible, we have not made such additional information available to the programs in our benchmark test, unless the type of information can be expected to be readily obtained for any dataset.

Caution should thus always be taken when interpreting score values, since the reported scores might not accurately reflect the optimal capabilities of the methods. Also, we have run the programs using mostly their default parameter settings. We are fully aware that adjusting the parameters can greatly affect the performance of a program, however, selecting the most appropriate parameter values be can be tricky and running methods with default settings is probably closer to typical usage.

It is inherently difficult to conduct an assessment that is fair to all methods. Even the most minute design choice can influence the outcome if it unintentionally favours some methods over others. For instance, limiting the size of input sequences will be beneficial for most module discovery tools since it improves the signal-to-noise ratio. On the other hand, using too short sequences can disadvantage methods that require substantial amounts of data in order to derive elaborate background models. The best solution, then, is to try to balance the scales by subjecting methods to several different situations with datasets exhibiting a range of characteristics. This will make it harder still to declare a winner, since it will inevitably lead to even greater variation in the results. Then again, the purpose of benchmarks tests need not be to identify a single program that can be recommended for all needs, but rather to determine how different methods behave under different conditions, thus enabling us to select the most appropriate tool to use in specific situations.

The results from our assessment of eight published module discovery tools show that the top scoring method does vary a lot between datasets. On the TRANSCompel dataset, for instance, all methods save Stubb and CisModule score better than the others on at least one sequence set. But there is also a tendency for some methods to perform consistently better or worse across several datasets. CisModule performed poorly on most sequence sets, Cister and Stubb usually scored somewhere in the middle, while CMA, ModuleSearcher, MSCAN and Cluster-Buster were often found among the top scoring methods on each set. CMA and ModuleSearcher were clearly best at identifying the correct motif types involved in the modules, and they were also the only methods capable of coping with large and noisy PWM sets. The other PWM-reliant methods appear to be more suited for detecting modules with some prior expected composition than for discovering completely new and uncharacterized modules.

There was some variation when using custom PWMs as opposed to TRANSFAC PWM sets. The average performance over all methods on the whole TRANSCompel dataset was about the same in both cases, but there were a lot of local differences between sequence sets. The most extreme example can be seen on the Ebox-Ets sequence set where MSCAN scores highest of all with TRANSFAC matrices, yet completely fails to find any true modules with custom matrices. The average deviation in scores when using either PWM set was about 0.11 and the effect could go both ways. MCAST was the only method which almost consistently scored better with one set, namely custom matrices.

Conclusion

While improvements can still be made to our systems, we have taken a first step towards developing a comprehensive testing workbench for composite motif discovery tools. The assessment system is based on two established datasets for module discovery plus a novel dataset we constructed from TRANSCompel module annotations. The performance of methods on our novel set is comparable to the previous two, demonstrating its utility as a benchmark set. Together these datasets challenge methods to discover modules with different characteristics and varying levels of difficulty.

Not surprisingly, trying to discover composite motifs *de novo* proves to be much more challenging than relying on PWMs as an aid to detect potential single binding sites. With large and noisy PWM sets, however, it becomes crucial to consider multiple instances of conserved motif combinations in order to identify true modules. In general, our study shows that there are still advances to be made in computational module discovery.

Methods

TRANSCompel dataset

Our main dataset was based on modules annotated in the TRANSCompel database [22], which is one of very few databases that contain entries for composite elements whose combinatorial binding effects have been verified through biological experiments. It comes in both a professional licensed version and a smaller public version. Our dataset was selected from TRANSCompel Professional version 9.4 which contains 421 annotated module sites from 152 different module classes. The largest modules registered in TRANSCompel are triplets (34 entries) with the remaining being pairs of binding sites (387 entries). To ensure a minimum of support for each module class, we considered only classes that had at least five annotated module sites. Unfortunately, this requirement excluded all triplets and left us with only pairs. After further discarding a few modules that were too weak to be detected with stringent PWM-thresholds, we ended up with ten sequence sets encompassing 81 module binding sites in 63 different sequences. The longest module spanned 135 bp with the average being 33 bp. The binding sequences of modules are specified in TRANSCompel by using uppercase letters to indicate bases of the constituent single motifs and lowercase letters for the intra-module background. We used the supplied references to the EMBL database [30] to obtain additional sequence bases flanking these module sites but set an upper limit of 1000 bp on the length of the sequences used. Most of the sequences were from human or mouse but also some other mammalian and a few viral sequences were included. Each sequence set was constructed around modules of one particular class made up of two single motifs

from the following set of eleven: AML, AP-1, C/EBP, E-box, Ets, IRF, HMGIY, NF-AT, NF-κB, Sp1 and PU.1. The sequence sets contained between 4 and 16 sequences and the sequences themselves ranged in length from 294 to 1000 bp (average 884 bp). All sequences contained at least one module instance, but sometimes up to three, of the designated class. Some sequences also included annotated modules of other classes. This will usually not be a problem at low noise-levels, because the other modules will only interfere if the set of PWMs supplied to a program contains decoy matrices corresponding to the motifs involved in these modules. As the noise-level approaches 99%, however, this will inevitably happen because the PWM sets then include the complete TRANSFAC collection. Since we use real genomic data, there is also always a possibility that additional unknown modules are present in the sequences. Even so, for a particular sequence set, only module sites corresponding to the designated class of that set were considered true positives.

Although the TRANSCompel database itself does not provide matrix representations for the motifs involved in modules, its companion database TRANSFAC does [22]. Unfortunately, there is not a one-to-one correspondence between transcription factors and matrices in TRANSFAC, and a single factor (or family of factors that recognize the same motif) can be represented by several different PWMs. Instead of selecting just one canonical PWM to use for each motif, we collected all matrices related to a specific motif and treated the whole set as an equivalence class. Thus, a motif can be represented by either one of the PWMs in the corresponding set, and predicted binding sites in the sequences are considered to be instances of the same motif even if the binding sites are predicted by different PWMs from the equivalence set.

As an alternative to these TRANSFAC sets, we also constructed custom PWMs for the eleven motifs involved in our module classes. For each motif we extracted the corresponding annotated binding sites plus four flanking bases on each side from our sequences and used MEME [31] to align them and infer a PWM model for the motif. Constructing matrices from the same binding sites they will later on be used to detect introduces a circularity which will probably make these sites easier to find than if the PWMs had been constructed from independent sequences. This was intentional, however. Since the purpose of our study was to assess the methods' abilities to find significant *combinations* of binding sites rather than individual sites, we wanted the individual sites to be easily detectable. To verify that the annotated single binding sites in the TRANSCompel dataset were indeed detectable by our matrices, we used an algorithm from the "TFBS" package [32] to match the PWMs against the sequences. Of the 81 single binding sites in the dataset, all but ten could be detected with an 85% relative cut-off threshold. When we lowered the cut-off to 75%, all sites could be detected. Single binding sites were considered to be detected if a predicted match to the corresponding PWM overlapped with the annotated binding site. For the TRANSFAC matrices, we regarded it as sufficient if any one of the matrices in the equivalence set made a prediction that overlapped with the annotated site.

Liver and muscle datasets

The liver dataset was based on a set of regulatory regions used as a positive training set to develop a model of liver specific regulation in the paper by Krivan and Wasserman [23]. Sequence data as well as PWM models of four TFs implied in liver specific regulation (C/EBP, HNF-1, HNF-3 and HNF-4) was downloaded from their supplementary web site [33]. After inspection of the sequence annotations, we discarded from further consideration those regulatory regions that only contained a single TFBS and also smaller annotated regions that were completely overlapped by larger regions. Furthermore, we ignored a small set of TFs that only had one binding site each in the whole dataset. This left us with regulatory regions consisting of two or more binding sites for the four TFs previously mentioned. The start position of the first TFBS and the end position of the last TFBS in each region were used as module boundaries, and the modules thus obtained varied in length from 26 to 176 bp with and average of 112 bp. Long sequences were cropped to a maximum of 1000 bp. The resulting dataset after curation consisted of 14 modules in 12 sequences with 51 binding sites for 4 different TFs. Eight of the sequences were human, two were from rat and the last two from mouse and chicken.

For the muscle dataset we selected a subset of the regulatory regions from the paper by Wasserman and Fickett [7] obtained from their web site [34]. Five motifs (Mef-2, Myf, Sp1, SRF and Tef) were reported as important in muscle regulation, and PWMs for these motifs were downloaded from the same site. We chose regions that had at least two annotated binding sites for motifs in this set and used the first and last binding site in the regions to delimit the modules. Since most of the sequences at the website were excerpts and rather short, we tried to extend them where possible by obtaining the original sequences from EMBL, though limiting the sequences to a maximum of 1000 bp as usual. The final muscle dataset used contained 24 sequences with one module in each and a total of 84 TFBS for 5 motifs. The smallest module spanned 14 bp and the longest 294 bp (average 120 bp). 10 sequences were from the mouse genome, 6 from human, 5 from rat, 2 from chicken and 1 from cow.

Further statistics on the datasets and PWMs used are summarized in Table 1 and Additional File 1.

Running the programs

Most of the methods tested could be run directly from the input sequences and a set of PWMs. Both CMA and ModuleSearcher, however, rely on separate programs to match the PWMs against the sequences in a preprocessing step. For ModuleSearcher we used the program MotifScanner since both of these methods are part of the Toucan tools suite for regulatory sequence analysis [35]. MotifScanner was run with a third order background model based on vertebrate promoter sequences, which was also available with Toucan. CMA comes bundled with Match [36] for PWM scanning. Match utilizes two different threshold values which should be individually fitted for each specific PWM. Preconstructed cut-off profiles for TRANSFAC matrices are available for different conditions, for instance to minimize either the false positive or false negative discovery rate or to minimize the sum of these two rates. As suggested in the CMA publication, we used cut-off profiles designed to minimize the false negative discovery rate. Similar cut-off profiles for the liver, muscle and custom matrices were estimated according to the procedure described for Match [36]. For each PWM we generated 50000 random oligos by sampling from the PWM distribution. The PWM was then scored against these oligos with Match, and a cut-off threshold was chosen so that 90% of the oligos obtained a match score above this threshold. Since CMA is based on a discriminative model, it also requires a set of negative sequences along with the positive dataset. As negative data we selected 1000 bp promoter segments from 50 random housekeeping genes that were part of the default negative gene set included with the method's web service [37].

Availability and requirements

The web service for assessing composite motif discovery tools, as well as all the results from our benchmark test, is available at http://tare.medisin.ntnu.no/composite.

Abbreviations

ASP, average site performance (defined as (Sn + PPV)/2); bp, base pair; FN, false negative; FP, false positive; HMM, hidden Markov model; mCC, motif-level correlation coefficient; nCC, nucleotide-level correlation coefficient; PC, performance coefficient (defined as TP/(TP + FN + FP)); PPV, positive predictive value (defined as TP/(TP + FP)); PWM, position weight matrix; Sn, sensitivity (defined as TP/(TP + FN)); Sp, specificity. (defined as TN/(TN + FP)); TF, transcription factor; TFBS, transcription factor binding site; TN, true negative; TP, true positive.

Authors' contributions

GKS and OA conceived of the study. All authors participated in the design of the study. KK and GKS assembled the datasets. JJ implemented the web service and ran all the tests together with KK. KK drafted the manuscript. FD was the project supervisor. All authors helped revise and approved the final manuscript.

Additional material

Additional File 1

Dataset statistics. *This supplementary table includes information about the datasets and modules therein, the matrices used to represent the true motifs and the number of matrices in the PWM sets at various noise levels on the TRANSCompel dataset.*

Click here for file

[http://www.biomedcentral.com/content/supplementary/1471-2105-9-123-S1.xls]

Acknowledgements

Kjetil Klepper, Jostein Johansen and Finn Drabløs were all supported by The National Programme for Research in Functional Genomics in Norway (FUGE) in The Research Council of Norway. Finn Drabløs was additionally supported by The Svanhild and Arne Must Fund for Medical Research. Osman Abul has been fully supported by an ERCIM fellowship.

References

1. Werner T: **Models for prediction and recognition of eukaryotic promoters.** *Mammalian Genome* 1999, **10(2):**168-175.
2. Wray GA, Hahn MW, Abouheif E, Balhoff JP, Pizer M, Rockman MV, Romano LA: **The Evolution of Transcriptional Regulation in Eukaryotes.** *Mol Biol Evol* 2003, **20(9):**1377-1419.
3. Sandve GK, Drabløs F: **A survey of motif discovery methods in an integrated framework.** *Biol Direct* 2006, **1:**11.
4. GuhaThakurta D, Stormo GD: **Identifying target sites for cooperatively binding factors.** *Bioinformatics* 2001, **17(7):**608-621.
5. Xing EP, Wu W, Jordan MI, Karp RM: **Logos: a modular bayesian model for de novo motif detection.** *J Bioinform Comput Biol* 2004, **2(1):**127-154.
6. Zhou Q, Wong WH: **CisModule: de novo discovery of cis-regulatory modules by hierarchical mixture modeling.** *Proc Natl Acad Sci USA* 2004, **101(33):**12114-12119.
7. Wasserman WW, Fickett JW: **Identification of regulatory regions which confer muscle-specific gene expression.** *J Mol Biol* 1998, **278(1):**167-181.
8. Chan BY, Kibler D: **Using hexamers to predict cis-regulatory motifs in Drosophila.** *BMC Bioinformatics* 2005, **6:**262.
9. Frech K, Danescu-Mayer J, Werner T: **A novel method to develop highly specific models for regulatory units detects a new LTR in GenBank which contains a functional promoter.** *J Mol Biol* 1997, **270(5):**674-687.
10. Aerts S, Van Loo P, Thijs G, Moreau Y, De Moor B: **Computational detection of cis-regulatory modules.** *Bioinformatics* 2003, **19(suppl 2):**ii5-14.
11. Stormo GD: **DNA binding sites: representation and discovery.** *Bioinformatics* 2000, **16(1):**16-23.
12. Johansson , Alkema WBL, Wasserman WW, Lagergren J: **Identification of functional clusters of transcription factor binding motifs in genome sequences: the MSCAN algorithm.** *Bioinformatics* 2003, **19(Suppl. 1):**i169-i176.
13. Kel AE, Konovalova T, Waleev T, Cheremushkin E, Kel-Margoulis OV, Wingender E: **Composite Module Analyst: a fitness-based tool for identification of transcription factor binding site combinations.** *Bioinformatics* 2006, **22(10):**1190-1197.
14. Sharan R, Ovcharenko I, Ben-Hur A, Karp RM: **CREME: a framework for identifying cis-regulatory modules in human-mouse conserved segments.** *Bioinformatics* 2003, **19(suppl 1):**i283-291.
15. Sze SH, Gelfand MS, Pevzner PA: **Finding weak motifs in DNA sequences.** In: *Proceedings of the Pacific Symposium on Biocomputing* 2002:235-246 [http://helix-web.stanford.edu/psb02/sze.pdf]. Lihue, Hawaii

16. Hu J, Li B, Kihara D: **Limitations and potentials of current motif discovery algorithms.** *Nucl Acids Res* 2005, **33(15):**4899-4913.
17. Tompa M, Li N, Bailey TL, Church GM, De Moor B, Eskin E, Favorov AV, Frith MC, Fu Y, Kent WJ, Makeev VJ, Mironov AA, Noble WS, Pavesi G, Pesole G, Régnier M, Simonis N, Sinha S, Thijs G, van Helden J, Vandenbogaert M, Weng Z, Workman C, Ye C, Zhu Z: **Assessing computational tools for the discovery of transcription factor binding sites.** *Nat Biotechnol* 2005, **23(1):**137-144.
18. Bailey TL, Noble WS: **Searching for statistically significant regulatory modules.** *Bioinformatics* 2003, **19(Suppl. 2):**ii16-ii25.
19. Perco P, Kainz A, Mayer G, Lukas A, Oberbauer R, Mayer B: **Detection of coregulation in differential gene expression profiles.** *Biosystems* 2005, **82(3):**235-247.
20. Sandve GK, Drabløs F: **Generalized composite motif discovery.** In In: *Proceedings of the 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems* Melbourne, Australia; 2005:763-769.
21. Burset M, Guigó R: **Evaluation of gene structure prediction programs.** *Genomics* 1996, **34(3):**353-367.
22. Matys V, Kel-Margoulis OV, Fricke E, Liebich I, Land S, Barre-Dirrie A, Reuter I, Chekmenev D, Krull M, Hornischer K, Voss N, Stegmaier P, Lewicki-Potapov B, Saxel H, Kel AE, Wingender E: **TRANSFAC® and its module TRANSCompel®: transcriptional gene regulation in eukaryotes.** *Nucl Acids Res* 2006, **34:**D108-D110.
23. Krivan W, Wasserman WW: **A predictive model for regulatory sequences directing liver-specific transcription.** *Genome Res* 2001, **11(9):**1559-1566.
24. Sandelin A, Alkema WBL, Engström P, Wasserman WW, Lenhard B: **JASPAR: an open-access database for eukaryotic transcription factor binding profiles.** *Nucl Acids Res* 2004, **32:**D91-D94.
25. Frith MC, Hansen U, Weng Z: **Detection of cis-element clusters in higher eukaryotic DNA.** *Bioinformatics* 2001, **17(10):**878-889.
26. Frith MC, Li MC, Weng Z: **Cluster-buster: finding dense clusters of motifs in DNA sequences.** *Nucl Acids Res* 2003, **31(13):**3666-3668.
27. Sinha S, van Nimwegen E, Siggia ED: **A probabilistic method to detect regulatory modules.** *Bioinformatics* 2003, **19(Suppl.1):**i292-i301.
28. Kielbasa SM, Gonze D, Herzel H: **Measuring similarities between transcription factor binding sites.** *BMC Bioinformatics* 2005, **6:**237.
29. Duret L, Bucher P: **Searching for regulatory elements in human noncoding sequences.** *Curr Opin Struct Biol* 1997, **7(3):**399-406.
30. Kanz C, Aldebert P, Althorpe N, Baker W, Baldwin A, Bates K, Browne P, van den Broek A, Castro M, Cochrane G, Duggan K, Eberhardt R, Faruque N, Gamble J, Diez FG, Harte N, Kulikova T, Lin Q, Lombard V, Lopez R, Mancuso R, McHale M, Nardone F, Silventoinen V, Sobhany S, Stoehr P, Tuli MA, Tzouvara K, Vaughan R, Wu D, Zhu W, Apweiler R: **The EMBL nucleotide sequence database.** *Nucl Acids Res* 2005, **33:**D29-D33.
31. Bailey TL, Elkan C: **Fitting a mixture model by expectation maximization to discover motifs in biopolymers.** In: *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology* 1994:28-36 [http://research1t.imb.uq.edu.au/bailey/t.bailey/papers/ismb94.pdf]. Stanford, California
32. Lenhard B, Wasserman WW: **TFBS: computational framework for transcription factor binding site analysis.** *Bioinformatics* 2002, **18(8):**1135-1136.
33. Krivan W, Wasserman WW: **Liver model, supplementary material.** [http://www.cisreg.ca/tjkwon].
34. Wasserman WW, Fickett JW: **Catalogue of Regulatory Elements.** [http://www.cbil.upenn.edu/MTIR/HomePage.html].
35. Aerts S, Van Loo P, Thijs G, Mayer H, de Martin R, Moreau Y, De Moor B: **TOUCAN 2: the all-inclusive open source workbench for regulatory sequence analysis.** *Nucl Acids Res* 2005, **33:**W393-W396.
36. Kel AE, Gößling E, Reuter I, Cheremushkin E, Kel-Margoulis OV, Wingender E: **MATCH: a tool for searching transcription factor binding sites in DNA sequences.** *Nucl Acids Res* 2003, **31(13):**3576-3579.
37. Waleev T, Shtokalo D, Konovalova T, Voss N, Cheremushkin E, Stegmaier P, Kel-Margoulis O, Wingender E, Kel A: **Composite Module Analyst: identification of transcription factor binding site combinations using genetic algorithm.** *Nucleic Acids Res* 2006, **34(Suppl 2):**W541-545.
38. Schneider TD, Stormo GD, Gold L, Ehrenfeucht A: **Information content of binding sites on nucleotide sequences.** *J Mol Biol* 1986, **188(3):**415-431.

Paper II

BIOINFORMATICS APPLICATIONS NOTE Vol. 26 no. 17 2010, pages 2195–2197
doi:10.1093/bioinformatics/btq357

Sequence analysis Advance Access publication July 13, 2010

PriorsEditor: a tool for the creation and use of positional priors in motif discovery

Kjetil Klepper* and Finn Drabløs

Department of Cancer Research and Molecular Medicine, Norwegian University of Science and Technology, Trondheim, Norway

Associate Editor: Dmitrij Frishman

ABSTRACT

Summary: Computational methods designed to discover transcription factor binding sites in DNA sequences often have a tendency to make a lot of false predictions. One way to improve accuracy in motif discovery is to rely on positional priors to focus the search to parts of a sequence that are considered more likely to contain functional binding sites. We present here a program called PriorsEditor that can be used to create such positional priors tracks based on a combination of several features, including phylogenetic conservation, nucleosome occupancy, histone modifications, physical properties of the DNA helix and many more.

Availability: PriorsEditor is available as a web start application and downloadable archive from http://tare.medisin.ntnu.no/priorseditor (requires Java 1.6). The web site also provides tutorials, screenshots and example protocol scripts.

Contact: kjetil.klepper@ntnu.no

Received on April 21, 2010; revised on June 17, 2010; accepted on June 30, 2010

1 INTRODUCTION

Computational discovery of transcription factor binding sites in DNA sequences is a challenging problem that has attracted a lot of research in the bioinformatics community. So far more than a hundred methods have been proposed to target this problem (Sandve and Drabløs, 2006) and the number of publications on the topic is steadily increasing.

There are two general approaches for discovering potential transcription factor binding sites with computational tools. One is to examine regulatory regions associated with a group of genes that are believed to be regulated by the same factors and search for patterns that occur in all or most of these sequences. This approach, often referred to as *de novo* motif discovery, can be used when we have no prior expectations as to what the binding motifs might look like. One concern with this approach, however, is that it might be necessary to consider rather long sequence regions to ensure that the target sites are indeed covered. Since binding motifs for transcription factors are usually short and often allow for some degeneracy, the resulting signal-to-noise ratio can be quite low, making it difficult to properly discriminate motifs from background. Another problematic issue is that DNA sequences inherently contain a lot of repeating patterns, such as tandem repeats and transposable elements, which can draw focus away from the target binding motifs when searching for similarities between sequences.

The other general motif discovery approach, called *motif scanning*, searches for sequence matches to previously defined models of binding motifs, for instance in the form of position weight matrices (PWMs; Stormo, 2000). The main drawback with motif scanning is that it tends to result in an overwhelming number of false positive predictions. According to the 'futility theorem' put forward by Wasserman and Sandelin (2004), a genome-wide scan with a typical PWM could incur in the order of 1000 false hits per functional binding site, which would make such an approach practically infeasible for accurate determination of binding sites. The problem here lies not so much in the predicted binding patterns themselves, since many of these would readily be bound by transcription factors *in vitro. In vivo*, however, most such binding sites would be non-functional, perhaps because the chromatin conformation around the sites precludes access to the DNA (Segal *et al*., 2006) or because the target factors require the cooperative binding of additional factors nearby to properly exert their regulatory function (Ravasi *et al*., 2010).

One way to improve accuracy in motif discovery is to try to narrow down the sequence search space as much as possible beforehand, for instance, by masking out portions of the sequences that resemble known repeats or considering only sequence regions that are conserved between related species (Duret and Bucher, 1997). Kolbe *et al.* (2004) introduced a measure they called 'Regulatory Potential' which combines phylogenetic conservation with distinctive hexamer frequency profiles to identify possible regulatory regions. This measure calculates a score for each position along the sequence, and regions receiving higher scores are deemed more likely to have a regulatory role. Regulatory Potential can be considered as an example of a 'positional prior' since each position is associated with an a priori probability of possessing some specific property. Positional priors can be used as an aid in motif discovery by assigning high prior values to regions that we consider more likely to contain functional binding sites and then focus the search on these regions. Besides conservation and oligonucleotide frequencies, other features that can be relevant for assigning prior values include: localized physical properties of the DNA double helix, distance from transcription start site or other binding sites, ChIP-chip and ChIP-seq data, and potentially tissue-specific epigenetic factors such as the presence of nucleosomes and associated histone modifications. Many of the aforementioned features have previously been applied and shown to improve the performance of motif discovery by themselves (see e.g. Bellora

*To whom correspondence should be addressed.

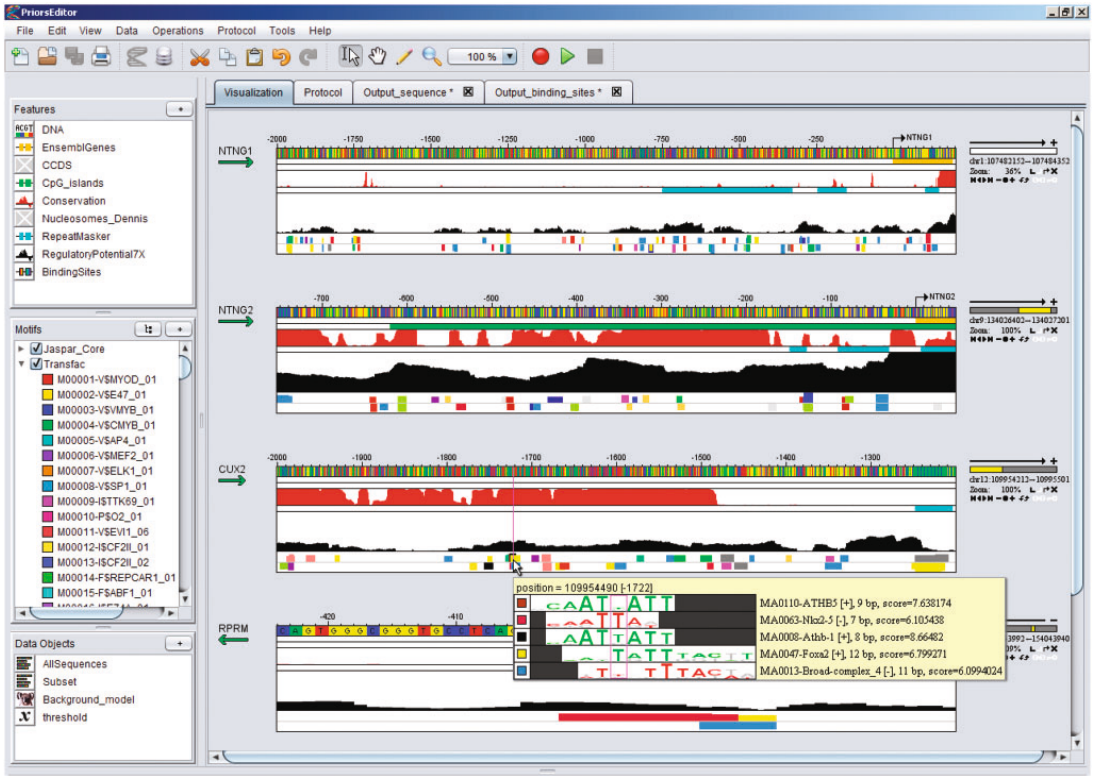


Fig. 1. The top left panel in this screenshot shows examples of some of the features that can be used as a basis to create positional priors. These features are visualized as data tracks in the main panel for a selected set of sequences. The bottom-most track contains predicted matches to TRANSFAC and JASPAR motifs in regions with non-zero RegulatoryPotential7X scores.

et al., 2007; Segal *et al*., 2006; Whitington *et al*., 2009), and it has also been demonstrated that further gain can be achieved by integrating information about multiple features (see e.g. Ernst *et al*., 2010; Lähdesmäki *et al*., 2008).

We present here a program called PriorsEditor, which allows users to easily construct positional priors tracks by combining various types of information and utilize these priors to potentially improve the motif discovery process (Fig. 1).

2 SOFTWARE DESCRIPTION

The first step in constructing a priors track with PriorsEditor is to specify the genomic coordinates for a set of sequences one wishes to analyze. Next, data for various features can be imported to annotate these genomic segments. PriorsEditor supports three types of feature data. The first type, *numeric data*, associates a numeric value with each position in the sequence and can be used to represent features such as phylogenetic conservation scores, DNA melting temperatures and nucleosome-positioning preferences. Numeric data tracks are also used to hold the final positional priors. The second feature type, *region data*, can be used to refer to continuous stretches of the DNA sequence that share some unifying properties which distinguish them from the surrounding sequence. Different regions are allowed to overlap, and regions can also be assigned values for various attributes, including type designations, score values and strand orientations. Features best represented as regions include genes, exons, repeat regions, CpG-islands and transcription factor binding sites. The last feature type, *DNA sequence data*, represents the DNA sequence itself in single-letter code. DNA sequence data can be passed on to motif discovery programs for further analysis, and it can also be used to estimate various physical properties of the DNA double helix, such as GC content, bendability and duplex-free energy. Additional feature data can be obtained from web servers such as the UCSC Genome Browser (Rhead *et al*., 2010) or be loaded from local files.

Once the data for the desired features have been loaded, the data tracks can be manipulated, compared and combined to create a priors track using a selection of available operations. These include operations to extend regions by a number of bases upstream and/or downstream, merge overlapping regions or regions within close proximity, filter out regions, normalize data tracks, smooth numeric data with sliding window functions, interpolate sparsely sampled

data, weight numeric data tracks by a constant value or position-wise by another track, combine several numeric tracks into one using either the sum or the minimum or maximum value of all the tracks at each position and several more. It is also possible to specify conditions for the operations so that they are only applied to positions or regions that satisfy the condition. For example, to design a priors track that will focus the search toward conserved regions within close proximity of other binding sites, one could start off with a phylogenetic conservation track, then load a track containing previously verified binding sites from the ORegAnno database (Griffith *et al*., 2008), extend these sites by a number of bases on either side and lower the prior values outside these extended sites.

After a priors track has been constructed, there are several ways to make use of this new data. The most straightforward way is to provide it as input to a motif discovery program that supports such additional information, for instance, PRIORITY (Narlikar *et al*., 2006) or MEME version 4.2+ (Bailey *et al*., 2010). Unfortunately, not many motif discovery programs are able to incorporate priors directly, so an alternative is to mask sequence regions that have low priors by replacing the original base letters with Xs or Ns since most motif discovery tools will simply ignore positions containing unknown bases when searching for motifs. Apart from being used to narrow down the sequence search space, priors information can also be applied to post-process results after motif discovery has been carried out, for instance, by filtering out predicted binding sites that lie in areas with low priors or adjusting the prediction scores of these sites based on the priors they overlap.

Positional priors tracks and masked sequences can be exported for use with external tools, but it is also possible to perform motif discovery from within PriorsEditor itself by using operations to launch locally installed programs. To facilitate motif scanning, PWM collections from TRANSFAC Public (Matys *et al*., 2006) and JASPAR (Portales-Casamar *et al*., 2010) have been included, and users can also import their own PWMs or define new collections based on subsets of the available PWMs.

Constructing priors tracks and performing motif discovery analyses can be tedious, especially when it involves many datasets and requires several steps to complete. If a user discovers a good combination of features to use for priors, it may be desirable to repeat the same procedure to analyze other sequence sets as well. PriorsEditor allows such repetitive tasks to be automatized through the use of protocol scripts. Protocol scripts describe a list of operations to be performed along with any specific parameter settings that apply for these operations. They can be programmed manually in a simple command language or be constructed using a 'macro recording' function which logs all operations the user carries out while in recording mode. With protocol scripts these same series of operations can be automatically applied to new sequence sets simply by the click of a button. These scripts can also be set up so that users can provide values for certain settings during the course of an execution, enabling users to select for instance a different background model or PWM threshold value to use in the new analysis.

By providing a protocol script describing the operations to be performed along with a file specifying the target sequences, it is possible to run PriorsEditor from a command-line interface instead of starting up the normal graphical interface. This allows the construction and use of positional priors to be incorporated into a batch-processing pipeline.

Funding: The National Programme for Research in Functional Genomics in Norway (FUGE) in The Research Council of Norway.

Conflict of Interest: none declared.

REFERENCES

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

Bellora,N. *et al.* (2007) Positional bias of general and tissue-specific regulatory motifs in mouse gene promoters. *BMC Genomics*, **8**, 459.

Duret,L. and Bucher,P. (1997) Searching for regulatory elements in human noncoding sequences. *Curr. Opin. Struct. Biol.*, **7**, 399–406.

Ernst,J. *et al.* (2010) Integrating multiple evidence sources to predict transcription factor binding in the human genome. *Genome Res.*, **20**, 526–536.

Griffith,O.L. *et al.* (2008) ORegAnno: an open-access community-driven resource for regulatory annotation. *Nucleic Acids Res.*, **36**, D107–D113.

Kolbe,D. *et al.* (2004) Regulatory potential scores from genome-wide three-way alignments of human, mouse, and rat. *Genome Res.*, **14**, 700–707.

Lähdesmäki,H. *et al.* (2008) Probabilistic inference of transcription factor binding from multiple data sources. *PLoS One*, **3**, e1820.

Matys,V. *et al.* (2006) TRANSFAC and its module TRANSCompel: transcriptional gene regulation in eukaryotes. *Nucleic Acids Res.*, **34**, D108–D110.

Narlikar,L. *et al.* (2006) Informative priors based on transcription factor structural class improve de novo motif discovery. *Bioinformatics*, **22**, e384–e392.

Portales-Casamar,E. *et al.* (2010) JASPAR 2010: the greatly expanded open-access database of transcription factor binding profiles. *Nucleic Acids Res.*, **38**, D105–D110.

Ravasi,T. *et al.* (2010) An atlas of combinatorial transcriptional regulation in mouse and man. *Cell*, **140**, 744–752.

Rhead,B. *et al.* (2010) The UCSC Genome Browser database: update 2010. *Nucleic Acids Res.*, **38**, D613–D619.

Sandve,G.K. and Drabløs,F. (2006) A survey of motif discovery methods in an integrated framework. *Biology Direct.*, **1**, 11.

Segal,E. *et al.* (2006) A genomic code for nucleosome positioning. *Nature*, **442**, 772–778.

Stormo,G.D. (2000) DNA binding sites: representation and discovery. *Bioinformatics*, **16**, 16–23.

Wasserman,W.W. and Sandelin,A. (2004) Applied bioinformatics for the identification of regulatory elements. *Nat. Rev. Genet.*, **5**, 276–287.

Whitington,T. *et al.* (2009) High-throughput chromatin information enables accurate tissue-specific prediction of transcription factor binding sites. *Nucleic Acids Res.*, **37**, 14–25.

Paper III

Klepper and Drabløs *BMC Bioinformatics* 2013, **14**:9
http://www.biomedcentral.com/1471-2105/14/9

BMC Bioinformatics

SOFTWARE Open Access

MotifLab: a tools and data integration workbench for motif discovery and regulatory sequence analysis

Kjetil Klepper* and Finn Drabløs*

Abstract

Background: Traditional methods for computational motif discovery often suffer from poor performance. In particular, methods that search for sequence matches to known binding motifs tend to predict many non-functional binding sites because they fail to take into consideration the biological state of the cell. In recent years, genome-wide studies have generated a lot of data that has the potential to improve our ability to identify functional motifs and binding sites, such as information about chromatin accessibility and epigenetic states in different cell types. However, it is not always trivial to make use of this data in combination with existing motif discovery tools, especially for researchers who are not skilled in bioinformatics programming.

Results: Here we present MotifLab, a general workbench for analysing regulatory sequence regions and discovering transcription factor binding sites and *cis*-regulatory modules. MotifLab supports comprehensive motif discovery and analysis by allowing users to integrate several popular motif discovery tools as well as different kinds of additional information, including phylogenetic conservation, epigenetic marks, DNase hypersensitive sites, ChIP-Seq data, positional binding preferences of transcription factors, transcription factor interactions and gene expression. MotifLab offers several data-processing operations that can be used to create, manipulate and analyse data objects, and complete analysis workflows can be constructed and automatically executed within MotifLab, including graphical presentation of the results.

Conclusions: We have developed MotifLab as a flexible workbench for motif analysis in a genomic context. The flexibility and effectiveness of this workbench has been demonstrated on selected test cases, in particular two previously published benchmark data sets for single motifs and modules, and a realistic example of genes responding to treatment with forskolin. MotifLab is freely available at http://www.motiflab.org.

Background

Computational motif discovery for transcription factor binding sites is a challenging research problem that has been studied for many years, but we are still missing approaches that can ensure generally good performance. For transcription factors with known binding motifs, scanning sequences for matches to motif models can identify potential binding sites, but the performance is often strongly degraded by a high content of false positive predictions; predicted sites that do not correspond to actual transcription factor binding events [1]. *De novo* motif discovery, i.e. discovery of potentially novel motifs from a set of DNA sequences, can work well for input sequences with high motif content, like from ChIP-Seq experiments. However, it is often less successful on more general sequence sets, based for example on regulatory regions for co-regulated genes [2].

It is a commonly used approach to not rely on predictions of just a single method, but to run several motif discovery methods on the same dataset and compare the results. The motivation is, of course, that although one individual method might be mistaken in a single case, any motif predicted by several different methods is probably more likely to be correct. Tools such as Melina [3] and Tmod [4] provide users the opportunity of running and comparing results for several methods within a unified interface, and *ensemble methods*, like EMD [5] and

* Correspondence: kjetil.klepper@ntnu.no; finn.drablos@ntnu.no
Department of Cancer Research and Molecular Medicine, Norwegian University of Science and Technology, Trondheim, Norway

MotifVoter [6], can take predictions from multiple methods as input and automatically derive a consensus. Still, the reason why motif discovery is so difficult in the first place is that binding motifs are often rather short and can vary substantially between binding sites. This makes them hard to discover with *de novo* motif discovery methods since the signal-to-noise ratio can be quite low when searching for motifs embedded in long background sequences. However, transcription factors seldom operate alone but work in concert with other transcription factors and co-factors in order to achieve the required regulatory control. Hence, groups of motifs for co-operating factors will often occur in close proximity to each other in the DNA sequence, and such "composite motifs", or *cis*-regulatory modules (CRM), can provide a stronger signal than individual motifs. Several *module discovery* methods have therefore been proposed to search for such motif groups [7].

A fundamental limitation with the traditional motif and module discovery approaches is that they only rely on information in the DNA sequence itself, but the mere presence of a binding motif does not necessarily imply that it is a functional binding site. Other conditions, such as for instance chromatin accessibility, DNA-methylation or even the distance to the transcription start site, can also influence the ability of transcription factors to bind and exert their regulatory function. Many binding sites may also function in a cell- or tissue-dependent manner, and a site which is active in one cell-type might well be inactive in others.

Recent advances in high-throughput experimental methods and large-scale genome annotation efforts, such as the ENCODE project [8], have led to an avalanche of data which is now available to researchers. ChIP-Seq data, for instance, can provide evidence that a specific transcription factor has bound to a region (albeit perhaps by indirect binding), and information about DNase hypersensitivity and epigenetic marks can indicate which regions of the DNA are generally accessible and also give clues as to their regulatory roles in different cell-types.

Newer motif/module discovery methods, including for example Chromia [9], Centipede [10], ProbTF [11], CompleteMOTIFs [12], Combinatorial CRM decoder [13] and i-cisTarget [14], try to take advantage of such additional information in order to improve their predictions. Some of these tools rely on a fixed set of features which are utilized in a predefined manner. This makes them very convenient and easy to use, but it also means that they are unable to incorporate new data unless their original creators update the underlying databases. Other methods are more general and can work with arbitrary data, but require that the users themselves obtain all the relevant data for the sequences they want to analyse and also convert this data into a format the tool can handle. This might not always be a trivial task, and it can sometimes even require that the users are skilled in programming. Hence, the threshold for making use of additional data in the analysis can often be high.

In this paper we present a tool called MotifLab which is designed to be a general workbench for analysing regulatory sequences and predicting binding sites for individual transcription factors and modules of co-operating factors. The main purpose of MotifLab is to provide a flexible framework which allows users to easily incorporate different kinds of additional information into the motif discovery process. As a motif discovery workbench it has drawn inspiration from other related tools, primarily Toucan [15], but it also shares similarities with e.g. MochiView [16], SeqVISTA [17] and RSAT [18]. MotifLab is written in Java and will run locally as a stand-alone application.

Implementation

Software description

At its core, MotifLab functions as a repository of data objects that can be manipulated and analysed using a number of available *operations*. The results can be visualized and examined interactively within the system or be output to standard text based formats (FASTA, GFF etc.) for further processing by other programs. MotifLab is not backed up by a central dedicated database server, but data can be retrieved automatically from various internet resources, such as the UCSC Genome Browser [19] or DAS servers [20], or alternatively be imported from local files. New data objects can also be derived from already existing objects or created manually from scratch.

MotifLab distinguishes between several types of data for different purposes. One of the fundamental data types in MotifLab is the *sequence,* which represents a segment of a genome, such as the promoter region associated with a specific gene. Users can create new sequence objects by specifying their chromosomal coordinates or by providing MotifLab with a list of gene identifiers and selecting a region to analyse around the genes' transcription start or end sites. The sequence objects merely function as references to genomic locations, and besides the coordinates, genome build and strand orientation of the sequences, they hold little additional information by themselves. However, sequences can be further annotated with *feature datasets*, which come in three different types: *DNA sequence datasets*, *numeric datasets* and *region datasets*. *DNA sequence datasets* contain a single base letter for each position within a sequence. Usually, they just hold the original DNA sequences for the genomic segments being investigated, but it is fully possible to have multiple DNA sequence datasets associated with the same sequences. These additional datasets can then contain masked versions of the original DNA sequence or randomly scrambled sequences to be used for statistical comparisons. *Numeric datasets*, on the other hand, have a

numeric value for each position in the sequences, and this data type can represent information such as phylogenetic conservation level, DNA stacking energy, melting temperature or basically any other signal that can vary in intensity along the sequence. The final feature data type, *region datasets*, associates each sequence with a set of regions. A *region* here refers to a subsegment of a sequence which has distinct properties that sets it apart from the rest of the sequence. Regions can represent features such as genes, CpG-islands, repeat regions or transcription factor binding sites. Different regions within the same sequence may overlap each other, and regions can also be assigned values for various attributes, including a type designation, score value and strand orientation.

MotifLab's graphical user interface offers a sophisticated sequence browser with powerful capabilities for visualizing sequences and associated feature data tracks, as shown in Figure 1. All the sequences are displayed simultaneously beneath each other in the same window so that features for different sequences can be compared visually. The browser is highly interactive and customizable, and it supports fast zooming to any scale and panning to show different parts of a sequence. The appearance of each track, including its colour, size and orientation, can be easily modified, and the order of the tracks and sequences can be rearranged or sorted according to different criteria. Individual sequences, tracks and even individual regions within region datasets can also be hidden from view to display only what the user wants to focus on at any time.

Besides sequences, another fundamental data type is the *motif*, which is used to model binding motifs for transcription factors. The binding motifs themselves are typically represented as either position weight matrices or IUPAC consensus sequences, but motif objects can be annotated with a lot of additional information as well, such as the names of different transcription factors that bind to the motif, names of organisms and tissues these factors are expressed in, references to other motifs representing known interaction partners for these factors and references to alternative models for the same motif. New motif objects are automatically added when performing motif discovery, but they can also be created manually by entering a matrix, IUPAC consensus or a set of aligned binding sequences. MotifLab includes several predefined motif models from databases such as TRANSFAC [21], JASPAR [22] and ScerTF [23].

It is often useful to be able to refer to subsets of sequences and motifs, for instance to divide a set of sequences into groups according to gene expression or to limit the search for binding motifs to transcription factors that are actually present in the cell-types being investigated. In MotifLab this can be accomplished with the help of *collection* objects. Users can create new collections by selecting data objects from a table or by supplying a list of objects to include. Collections can also be based on various statistics. For example, it is possible to create a sequence collection containing sequences with less than 40% GC-content or a motif collection with motifs that appear in at least 80% of the sequences. Somewhat related to *collections* are *partitions* which allow all data objects of a specific type to be divided into non-overlapping clusters. The *numeric map* data type associates each sequence or motif with an individual numeric value. Numeric maps can be used to hold data such as gene expression values for sequences or expected occurrence frequencies for motifs. General *text variable*s, on the other hand, can hold any kind of structured or unstructured text which will be interpreted depending on the context.

Operations and protocol scripts

MotifLab provides more than 40 data-processing operations to create, transform, combine, analyse and output data objects, including special operations to perform motif and module discovery. Some of these operations, like "output" and "copy", can be applied to any object, while others may be specific to a single type of data. The "mask" operation, for instance, can replace parts of a DNA sequence with other letters, such as X or N, or it can even replace the whole sequence with random bases sampled from a background distribution to create an entirely new artificial sequence. Numeric data objects can be transformed with arithmetic operations or other mathematical functions such as logarithms, range normalizations etc., and sliding windows can be applied to numeric features to smooth the data or to detect peaks, valleys and edges within the track. Other operations can change the size of regions by extending them in either direction or merge regions that overlap with each other. One of the simplest operations, but also one of the most useful, is the "filter" operation, since it can be employed to remove selected regions from a dataset, particularly binding sites that are suspected to represent false predictions.

Operations that target feature datasets can be limited to selected parts of sequences by specifying *conditions* that are evaluated for each individual base position or region in the dataset. These conditions can be based on the contents of the target track itself or involve information compared across several tracks. For example, a simple way to perform phylogenetic footprinting without having explicit access to orthologous sequences would be to first predict a set of binding sites in the normal way and then filter out those predictions where the average value of a conservation track within the binding sites is less than some threshold. Likewise, the process of "repeat masking", which is often performed prior to motif

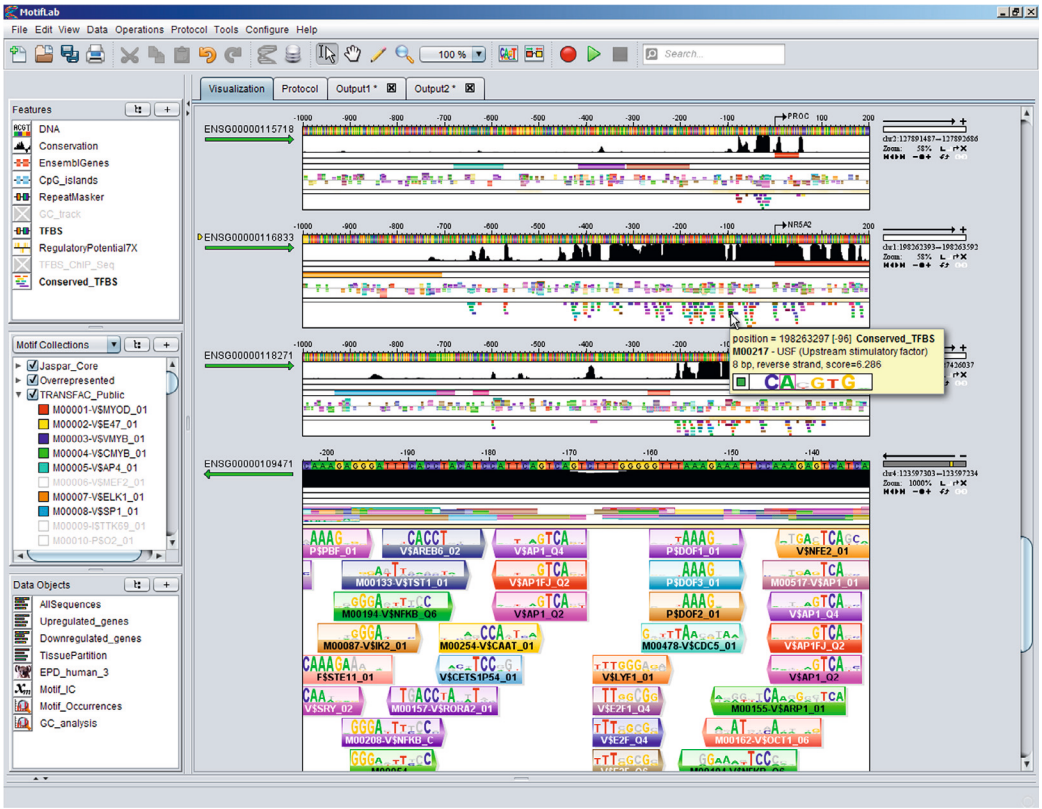


Figure 1 MotifLab's graphical user interface. The screenshot shows MotifLab's graphical user interface with three data panels to the left and with the sequence browser to the right taking up most of the screen space. The top data panel contains the feature datasets in the order they are visualized as tracks in the sequence browser, the middle data panel contains the motifs and modules, and the bottom panel contains miscellaneous data objects that to not belong in the first two panels. Features and motifs that are greyed out in the data panels are hidden from view in the sequence browser. The bottommost sequence shows a motif track in "close-up mode" which is activated at zoom-levels above 1000%. The binding sites are shown with superimposed "match logos" where the base matching the DNA sequence in that position is shown in colour and the other three bases are greyed out.

discovery, can easily be accomplished by limiting the "mask" operation to bases that lie within regions in a track containing known repeats. Conditions offer an easy way of integrating information from several features, and they can be made arbitrary complex by combining multiple individual conditions with Boolean operators.

Analysis of regulatory sequences usually involves multiple steps and requires several operations to obtain and pre-process data, discover motifs and binding sites and post-process and analyse the results. To keep track of what is being done, MotifLab provides functionality that allows users to automatically record every step they perform in a *protocol*. The protocol is written in a structured format and includes information about which operations have been executed, as well as details about their parameters, conditions and constraints. The protocol can thus serve as a form of documentation of the analysis process, but more importantly, it also makes MotifLab able to automatically apply the same workflow to other datasets as well, or to restore a previous session. Protocols can alternatively be written and edited manually, either in external text editor programs or in MotifLab's own internal protocol editor. By supplying a protocol script describing the full analysis workflow, it is possible to run MotifLab in "batch mode" from a command line without starting up the graphical user interface. This also allows MotifLab to be incorporated as a component in larger analysis pipelines.

MotifLab's graphical user interface promotes interactive data exploration, and multi-level undo/redo-functionality

provides users the opportunity to experiment with various operations and try out different parameter settings for these without having to worry about making irreversible changes to the data. Unlike some other workbench systems, MotifLab does not maintain an explicit history record which keeps track of all changes made to data and provides access to earlier states. However, when data is updated through the use of operations, the results can always be stored in a new data object under a different name rather than replacing the original object. This way, the original data can be kept intact and used for other purposes as well. It is also possible to save the entire state of MotifLab to a single "session file" to continue working on an analysis at a later time.

Motif discovery

Discovering motifs and searching for transcription factor binding sites within sequences are some of the primary functions of MotifLab. However, MotifLab is not actually capable of performing motif discovery by itself but relies on external programs installed on the user's computer to accomplish such tasks. This makes MotifLab flexible with respect to local software preferences or novel tools. In order for MotifLab to communicate with external programs, they must conform to standard data formats for input and output and their interfaces must be described in XML-based configuration files. MotifLab already supports several popular motif discovery tools, including AlignACE [24], BioProspector [25], MDscan [26], MEME [27], MotifSampler [28] and Weeder [29], and more tools will continuously be added (visit the MotifLab web site for a complete and updated list). Many of the supported programs have also been gathered in a central repository so they can be downloaded and installed from within MotifLab.

MotifLab has separate operations for performing *motif scanning*, where external programs are provided with a collection of predefined motifs and should return a region dataset containing predicted binding sites for these motifs, and *de novo motif discovery*, where the programs should discover both the binding sites and the motifs themselves. In addition, a third operation offers support for *ensemble methods* which can take predictions from other methods as input and combine these into potentially more reliable predictions.

Tracks with predicted binding sites are called *motif tracks*, and they have a special status in MotifLab because of the connection between the binding site regions and the motif objects associated with these sites. This enables the sequence browser to visualize binding sites with motif logos superimposed on the regions (as can be seen in the bottom sequence in Figure 1), and clicking on a binding site will bring up additional information about the motif.

Using positional priors to guide motif discovery

Motif discovery is a challenging problem since it involves searching for short and often degenerate patterns embedded in potentially long sequences. However, some parts of the sequences are more likely to contain functional binding sites than others, such as regions where the chromatin has an open conformation or sites that have been conserved throughout evolution. Some motif discovery programs allow users to limit the search space by masking out parts of sequences and thereby excluding them completely from further consideration. However, this approach might be considered too strict, since the excluded regions could, in fact, contain functional binding sites that will inevitably be destroyed by the masking procedure. A more flexible alternative is to construct a *positional priors* track wherein each sequence position is assigned a score or probability value reflecting a prior belief that the position could be part of a binding site. Such a track can be used to guide motif discovery programs by biasing the search towards regions with higher probability of containing true sites. Many types of information can be represented using positional priors, for instance phylogenetic conservation [30], nucleosome occupancy [31], properties of the DNA-helix [32] and epigenetic marks [33], and information from many different sources can be combined into a single priors track [34]. Positional priors are currently only supported directly by a few motif discovery and scanning programs, including PRIORITY [35], MEME [36], FIMO [37], ChIPMunk [38] and GRISOTTO [39], but they can also be used indirectly in combination with other programs, for instance by employing positional priors to filter out likely false predictions in a post-processing step.

Although tracks related to e.g. conservation, DNase hypersensitivity and ChIP-Seq experiments do not actually contain probability values in a strict statistical sense, such tracks can often be used directly as positional priors (or after minimal processing) since higher values in these tracks correlate well with occurrences of functional binding sites. For other types of features the relationship might not be so direct, and more advanced processing will be required to generate positional priors based on such features. MotifLab is an extension of an earlier program called PriorsEditor [40], which was developed specifically for creating and using positional priors tracks for motif discovery. Many of the operations provided by MotifLab are therefore related to transforming and combining features to facilitate manual construction of positional priors tracks, for instance to make weighted combinations of several tracks. Creating positional priors tracks manually can be beneficial if you want to utilize specific biological knowledge or want to set up a track with clearly defined focus. For example, if you have a set of known binding sites for a single transcription factor and want to look for potential interaction partners for this factor, you can

create a track which focuses the search to the vicinity of these sites, possibly adjusting the track further, for instance by assigning increased weight to conserved regions.

Compared to PriorsEditor, MotifLab offers several new functions to work with positional priors, including an operation to convert a regular priors track into a *discriminative prior* (as described in [31]) and analyses to evaluate the potential merit of priors tracks. The most important new addition, however, is the introduction of "Priors Generators" that can be used to generate positional priors automatically based on information from various features. A Priors Generator is basically just a machine learning classifier that can be trained to predict whether or not a position in a sequence would be expected to lie within a transcription factor binding site depending on the values of relevant features at that position. MotifLab provides a simple "wizard" to guide users through the steps required to configure a new Priors Generator, such as selecting the target and input features, setting up a training dataset and finally training the classifier and saving the result. Once a Priors Generator has been created, it can be used to generate positional priors for any sequence as long as the required input features are available. Although Priors Generators were introduced primarily for the prediction of transcription factor binding sites, they can just as well be trained to predict other region-based features in the same manner, provided that a reasonable correlation between the target feature and the input features can be expected.

Module discovery

Co-occurrence of motifs in modules represents a higher level of *cis*-regulatory organization that can be exploited to improve motif prediction, as binding sites for interacting factors which appear in close proximity to each other are less likely to represent spurious motif occurrences. MotifLab allows motifs to be annotated with information about known interaction partners, and one way to utilize this information is simply to filter out predicted binding sites that do not have sites for potential partners within some given distance.

Regulatory modules can also be modelled explicitly in MotifLab with their own data type analogous to single motifs. A *module* is made up of multiple constituent motifs along with optional constraints on their order, their orientations relative to each other and the distances between them. Because public motif databases often contain several alternative motif models for the same transcription factors, MotifLab permits each constituent motif in a module to be represented by collections of motifs in order to achieve greater sensitivity when performing module scanning.

As for single motif discovery, MotifLab provides separate operations to scan sequences for matches to predefined modules and to search sets of sequences to identify groups of motifs that might represent novel modules. Again, both of these operations rely on external module discovery programs to do the actual work.

Statistical analyses

The *analyze* operation is a versatile operation that can be employed to perform a number of different statistical analyses ranging from simple data comparisons to more elaborate analyses like motif overrepresentation studies. It will often be used to produce the final reports for an analysis session, but it is also useful for providing rapid answers to simple questions that might arise when working with datasets, such as "what is the GC-content of these DNA sequences", "do these two collections share a significant overlap", "is property X correlated with property Y" or "is the value of this numeric track higher within some regions than outside".

The results from the analyses can be output either as HTML-documents, with nicely formatted tables and images, or in a "raw text" format suitable for parsing by other programs. Individual results can also be extracted from analysis objects and turned into other types of objects for use elsewhere. For example, if you have performed an analysis to determine the number of times each motif occurs in a sequence set ("count motif occurrences"), you can extract these counts as a *numeric map*, or you can make a *motif collection* containing the motifs that were significantly overrepresented in the sequences and use this in another analysis.

Some analyses, like the previously mentioned "count motif occurrences", will generate individual results for each motif, module or sequence, and these results are presented in interactive tables that are linked to the corresponding data items. This makes it possible to e.g. highlight entries in the tables that are members of different collection objects, or to highlight corresponding elements in the sequence browser based on selections made in the tables. For example, when examining the results from a motif overrepresentation analysis, users can select the top most significant motifs from the table and then choose "Show only these motifs" from a context menu to visualize only the binding sites for these motifs in the sequence browser. The tables are therefore not merely static presentations of the results, but can be used as a starting point for further exploration of the data. If the tables contain motifs, the motif logos will always be included in a separate column. This is very useful, since rather than just listing numerous motif identifiers or names of transcription factors the user may or may not be familiar with, the logos enable users to immediately identify properties of the corresponding motifs and see similarities between them.

Results from multiple analyses can be collated into "meta-analyses" by extracting selected columns from

individual analyses and combining them into larger tables. Information from different types of analyses can be combined in this way to produce more comprehensive reports, or results from the same analysis run multiple times with different parameter settings can be juxtaposed to assess the impact of varying these parameters.

Interactive tools

In addition to the data manipulation and analysis capabilities provided by operations, MotifLab also includes a few tools aimed at interactive exploration of data. Unlike operations, these tools cannot be controlled by protocol scripts, and they are only available through the graphical user interface. Many of the tools are intended to aid visual inspection of motif tracks, for instance by highlighting binding sites with selected properties in the sequence browser.

The **Motif Browser** and **Module Browser** are two convenient tools for managing your motif and module libraries. These browsers will show an overview of all motifs or modules currently known to the system. The entries are displayed in a table with three columns containing the name of the motif/module, a graphical logo representation, and a third property that can be chosen by the user (see Figure 2a). A filter box enables users to search for entries with specific properties, for instance motifs associated with a given transcription factor, motifs for factors expressed in specific organisms or tissues, motifs containing a given consensus sequence, or modules containing a specific constituent motif. The search filter can also be coupled to the sequence browser so that only binding sites for motifs or modules matching the selected filtering criteria will be shown in the tracks.

The **Motif Score Filter** tool is basically just a slider bar which is used to dynamically adjust a cut-off threshold. Any binding site region whose score-property falls below the selected threshold will be hidden from view in the sequence browser. This tool can thus be used to highlight sites with increasingly higher scores. Besides the standard score-property, other values associated with binding sites can be used for filtering as well, for instance the average score of a numeric data track within the binding site.

As previously mentioned, MotifLab allows motifs to be annotated with known interaction partners, and this information can be utilized by the **Interactions Viewer** to visualize potential interaction networks directly within motif tracks. When a user clicks on a binding site region, any binding sites within a chosen distance that are associated with known interaction partners of the target motif will be highlighted (Figure 2b). The network can also be expanded to show several levels of interactions in different colours. This tool is especially useful if you already have a verified binding site that can be used as a starting point to implicate additional predictions that might be likely to represent functional binding sites.

Finally, the **Positional Distribution Viewer** will draw a histogram based on the locations of all currently visible regions in a selected track across all sequences (Figure 2a, bottom). The histogram will be dynamically updated in response to events that change the visibility of regions, making it very useful in conjunction with other tools such as the Motif Browser or Motif Score Filter.

Results

This section presents three examples of practical applications using MotifLab, which also illustrate some benefits of incorporating additional information when analysing regulatory sequences. Complete protocol scripts for these examples are available from the MotifLab web site.

Example 1: Improving motif discovery with automatically generated positional priors

We have previously published a suite of benchmark datasets for single motif discovery (based on binding sites annotated in TRANSFAC) where we made sure that it would be at least theoretically possible to discriminate the target motifs from the background sequence. Nevertheless, when we tested the performance of the motif discovery program MEME on these benchmark sets, the results were not particularly encouraging [41]. In this example we use an updated version of the datasets (see Additional file 1) to demonstrate how information about various sequence-related features can be integrated into a positional priors track and used to guide MEME towards the target motifs. The features chosen as a basis for the priors track were: conservation, conserved peaks, DNase hypersensitive sites, general regions bound by transcription factors according to ChIP-Seq data, CpG-islands, gene regions, coding regions, repeat regions and regions with histone marks H3K4me1 and H3K4me3. Since not all organisms are annotated with these features at the present time, we restricted the benchmark datasets to only consist of sequences from human and mouse genomes. The updated benchmark suite comprised 22 datasets, each containing binding motifs for a particular transcription factor and consisting of at least five sequences. We used a cross-validation approach where a Priors Generator based on a neural network classifier was trained on 21 of the 22 datasets and then used to generate a positional priors track for the dataset that was held out. The priors tracks were provided as input to MEME along with the DNA sequences, and MEME was instructed to identify a single motif with size between 8 and 16 bp in each dataset. For comparison we also ran MEME with a uniform priors track (effectively the same as using no priors) and a priors track based solely on conservation.

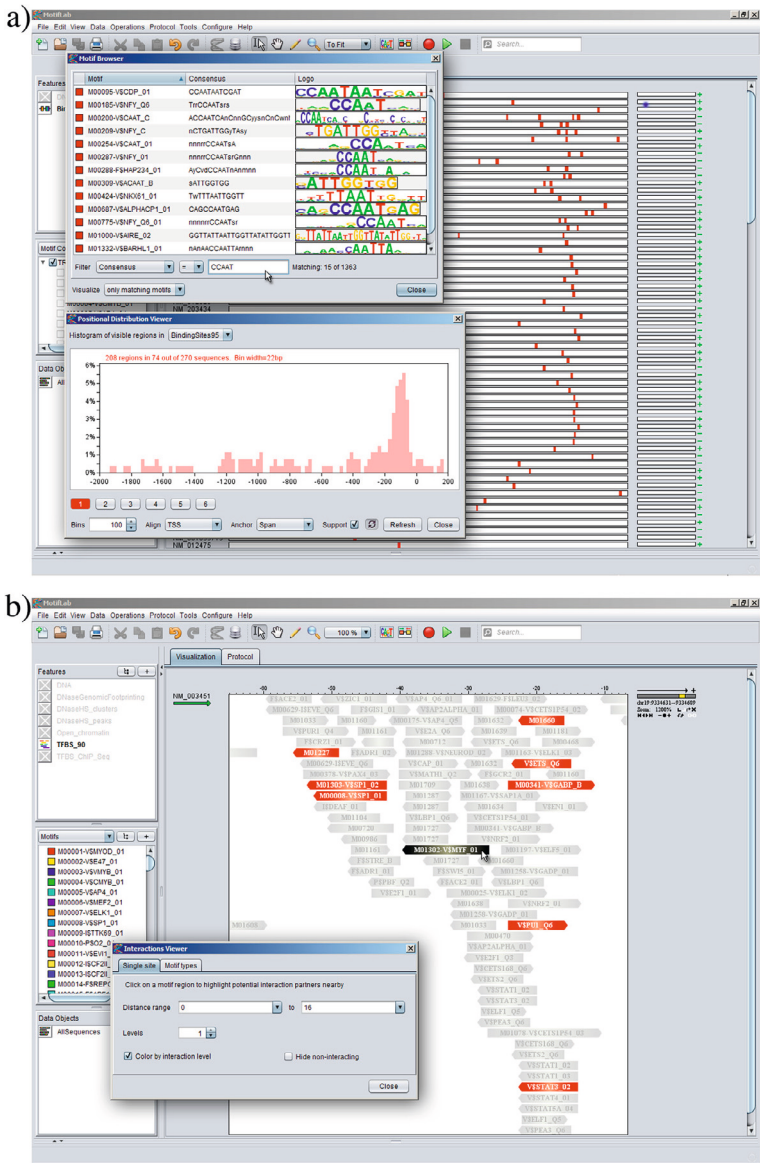


Figure 2 Examples of interactive tools. a) The **Motif Browser** tool (top dialog box) has here been used to search for TRANSFAC motifs containing the consensus sequence "CCAAT" (both orientations). The corresponding binding site predictions for the 15 motifs matching this criterion are shown as red boxes in the sequence browser partly visible in the background. The **Positional Distribution Viewer** tool (bottom dialog box) shows a histogram of the locations of these binding sites, and the prominent peak indicates that the majority of the sites are located within 200 bp upstream of the TSS. **b**) The **Interactions Viewer** tool. A part of a motif track is shown in close-up mode at 1200% scale (binding sites are displayed here without motif logos). The black binding site in the middle is the target site selected by the user and the red sites on either side have been highlighted by MotifLab as binding sites for transcription factors that are known to interact with the target factor(s) from other locations. All other binding sites are greyed out.

The results, combined over all datasets, are shown in Figure 3. Detailed results for individual datasets are provided in Additional file 1. As can be seen from the bar chart, the performance of MEME when not relying on any additional information was rather low, with an average CC score of 0.06. However, the performance increased about 3- to 4-fold for most metrics when the automatically generated positional priors were used to guide the search. Many of the target binding sites in the benchmark were located in conserved regions, and conservation was the most informative single feature with respect to binding site prediction. Conservation also contributed most to the specificity of the combined priors, while the other features primarily helped to elevate the basal prior probability slightly within some broader parts of the sequences. Figure 4 illustrates the ability of the individual features to discriminate between binding sites and the background sequence.

Even when positional priors were used, the results were far from perfect. There are several reasons for this: 1) for about half of the datasets MEME failed to predict the correct target motif as its top candidate, 2) in some datasets where MEME did identify the correct motif, alternative binding sites for the motif were selected instead of the annotated targets in a few sequences, and 3) even if MEME predicted the basic motif and binding sites correctly it did not always predict the correct size of the motif, which could have significant impact on the nucleotide-level statistics. A closer look at the predicted sites and motifs revealed that MEME found the target motif (or a resembling one) in 3 out of the 22 datasets with the uniform priors. This number increased to 8 when conservation was used to guide the search, and with the auto-generated priors MEME found the target motif in about 9 to 11 datasets (depending on the particular Priors Generator used).

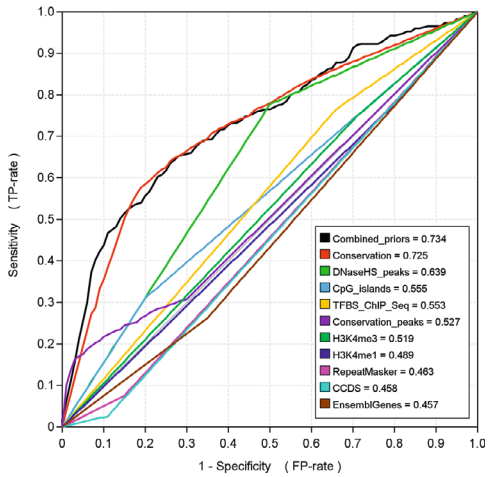


Figure 4 Predictive capabilities of individual features. These ROC-curves illustrate the ability of both the auto-generated combined priors and the 10 individual features the combined priors were based on to discriminate between sequence positions that are part of binding sites or part of the background sequence. The numbers in the legend box are the *area under the curve* (AUC) values for each feature.

Example 2: Module discovery

In a second benchmark study we evaluated the performance of eight published module discovery methods on a novel benchmark suite [7]. The suite consisted of 10 datasets with pairs of motifs appearing together in

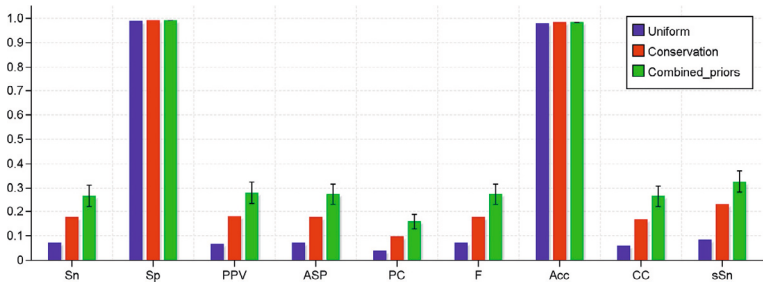


Figure 3 Results from example 1 – Single motif discovery benchmark. The figure shows the performance of MEME on the single motif discovery benchmark when guided respectively by a uniform positional priors track, a priors track based only on conservation, and a combined priors track made by automatically integrating information from several features with the use of a Priors Generator. The statistics were calculated by combining all sequences from the 22 datasets into one large dataset and measuring the overlap between the predicted binding sites and the target sites. The first eight statistics are *nucleotide*-level statistics whereas the last statistic is the *site*-level sensitivity (number of predicted sites overlapping with at least 25% of a target site). Due to the stochastic nature of the algorithm used to train the Priors Generator, the combined priors track could vary slightly depending on the training. We therefore trained 20 different Priors Generators and ran MEME with priors tracks generated by each of them. The bars show the average scores with standard deviations over the 20 runs.

multiple sequences and two additional datasets with larger heterogeneous modules involved in regulation in liver and muscle tissue respectively. Most of the methods we tested relied on a first step to scan sequences with a provided motif collection to find a set of candidate binding sites, and then they proceeded to search through these candidates in order to identify potential modules. The results showed, not surprisingly, that the task of identifying the target modules became harder as more candidate motifs were considered. In this example we demonstrate how the performance of a module discovery tool can be improved by utilizing additional information to reduce the number of candidate sites in a pre-processing step.

To generate the candidate datasets we first scanned the benchmark sequences with 1363 motifs from TRANSFAC PRO using a rather sensitive threshold (80% match) to ensure that all the target binding sites were recovered. Then we filtered the predicted sites according to various criteria to produce different candidate sets. As filtering criteria we used increasing levels of average conservation within the sites (more than 0%, 10%, 30% and 60%) or required that each site should be located nearby a site for a known interaction partner (within 10 or 20 bp). For the "liver" and "muscle" datasets we also filtered sites for transcription factors that were not known to be expressed in the respective tissues. In addition we tried several combinations of these criteria. The remaining binding site predictions for each dataset were provided as input to the module discovery tool ModuleSearcher [42].

Results for two of the datasets are shown in Figure 5, and the remaining results are included in Additional file 1. For all 12 datasets there was some form of additional information that would lead to better performance when used to filter candidate sites. However, in some cases, there were filtering criteria that would actually result in lower performance. This was especially true for the datasets "CEBP-NFkB" and "IRF-NFkB" (see Additional file 1: Figures S2d and S2g). These two datasets were the easiest in the original benchmark, and ModuleSearcher did a good job of discovering the target modules even without filtering the candidate sites. However, since only about half of the binding sites comprising these modules were conserved, using a strict conservation criterion made it impossible to correctly discover the target modules.

As an additional control we also tried to filter the candidate datasets completely at random to verify that any increase in performance was not simply due to a general reduction in the number of candidate sites. Filtering sites at random would in fact lead to better results in many cases, most notably for datasets where the baseline performance was poor. This is perhaps not so surprising, since the vast majority of the candidate sites would be considered to be false positives anyway according to the benchmark datasets. However, the increase in performance was usually not as great as when more sensible filtering criteria were employed.

Example 3: Identifying TFs regulating genes after forskolin treatment

Forskolin is a diterpene which is known to raise the level of cAMP (a second messenger) within cells [43], and this will in turn trigger many different responses, including activation of various transcription factors. HEK293 cells were treated with forskolin and the effect on gene expression was measured at different time points using microarray technology. Genes that were significantly differentially expressed compared to untreated cells were identified and sorted according to their peak time point. Of the 860 genes in total whose transcript levels were changed by the forskolin treatment, 270 had a peak differential expression after 2 hours (108 upregulated and 162 downregulated). We obtained promoter sequences for these 270 genes spanning 2000 bp upstream to 200 bp downstream of the transcription start site and performed motif scanning with 931 vertebrate motifs from TRANSFAC PRO.

The standard procedure for identifying transcription factors that might be involved in regulating a set of genes is to identify motifs that are significantly overrepresented in the dataset relative to a realistic background frequency. To estimate an expected frequency of each motif, we used a 3rd-order background model based on human promoter sequences to create a set of artificial control sequences and performed motif scanning in those sequences using the same parameter settings as before. We then derived the frequencies of the motifs from these control sequences and stored the results in a *numeric map*. Finally, we counted the number of times each motif occurred in the target dataset and used the expected motif frequencies to calculate p-values for overrepresentation with a binomial test. 113 motifs were found to be overrepresented at a significance threshold of 0.05 (Bonferroni corrected to 5.37×10^{-5} by dividing with the number of motifs considered). Many of the motifs with lowest p-values were GC-rich, which might stem from the fact that the sequences in the target dataset had a slightly higher GC-content than the control sequences used for comparison. The transcription factor CREB, which is a well-known cAMP-responsive factor but does not have a particularly GC-rich motif, was only ranked as number 57 according to p-value.

An alternative to overrepresentation is to look for motifs whose sites share similar properties across several sequences, for instance motifs that tend to appear at the same distance from the transcription start site, or motifs that are consistently conserved in many sequences. We therefore ran two additional analyses where we first

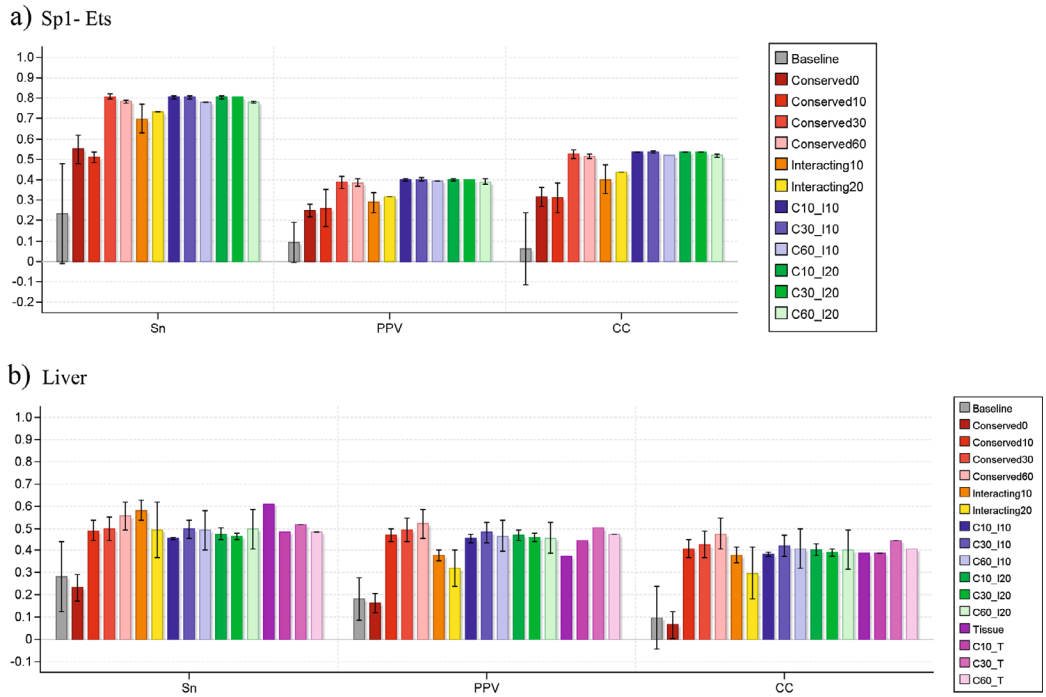


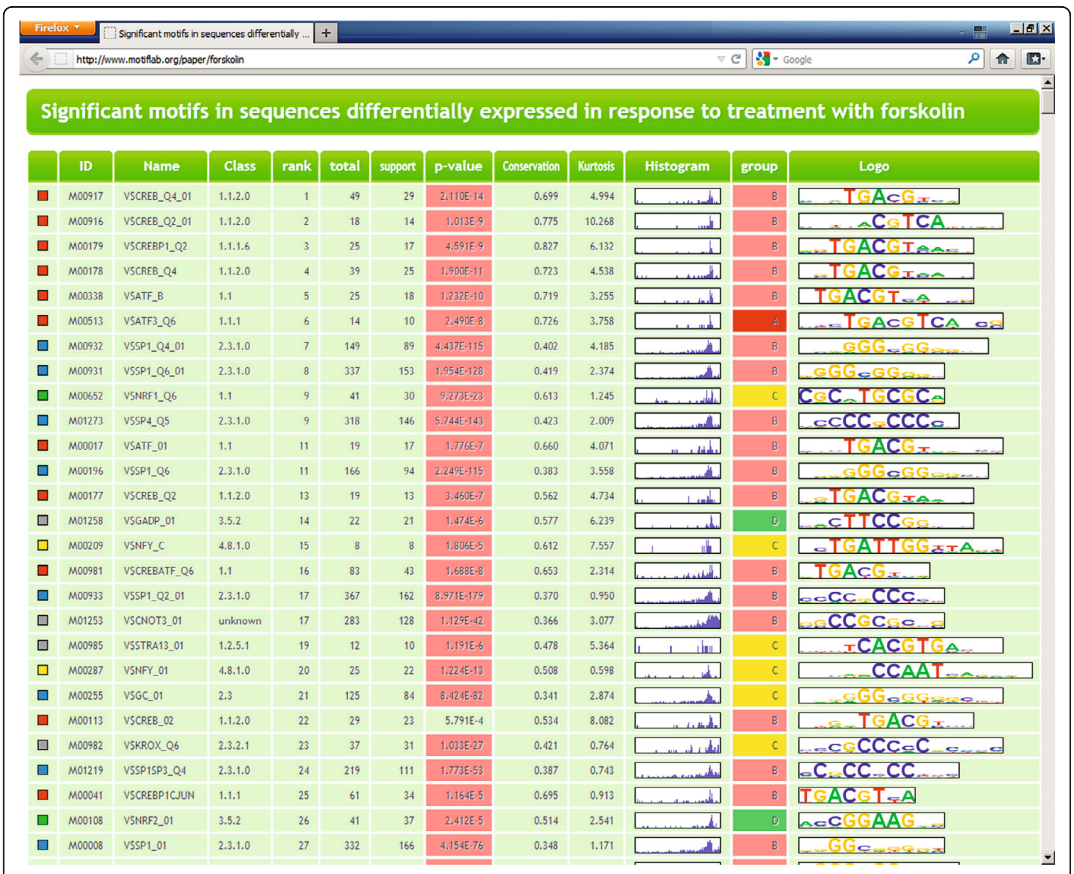
Figure 5 Results from example 2 – Module discovery benchmark. This figure shows the nucleotide-level performance of ModuleSearcher on two of the datasets from the module discovery benchmark. Since ModuleSearcher is based on a non-deterministic algorithm, we ran it 10 times on each dataset. The bars show the average scores with standard deviations. The "baseline" scores reflect the performance when no pre-processing was performed to filter candidate binding sites, and the other scores are for different filtering criteria and combinations thereof. "C10_I10" means that the sites were filtered according to both the "Conservation10" and "Interacting10" criteria etc. **a**) The "Sp1-Ets" dataset was one of the hardest in the original benchmark, but filtering sites based on either conservation or potential interacting sites nearby significantly improves the performance of ModuleSearcher on this dataset. **b**) For the "liver" dataset we also filtered binding sites for motifs that were not known to be expressed in liver ("Tissue") and combined this criterion with different requirements on conservation level ("C10_T" etc.).

calculated the average conservation level for each motif across all its binding sites and then analysed the positional distribution of the sites, using *kurtosis* as a simple measure of clustering.

Not surprisingly, the motifs that scored highest on average conservation were those that only occurred once or twice and their binding sites just happened to lie within conserved regions. These motifs are not interesting for the dataset as a whole, however, since they are at best involved in regulating only a few genes. The most interesting motifs would be those that score high on conservation and kurtosis but still occur often enough to have a significant overrepresentation p-value, so we combined these three properties into a single measure using rank sum.

According to this combined measure, CREB (along with the related factor ATF which binds to the same motif) was ranked on top, followed by the ubiquitous factor Sp1 which binds to the GC-box. Another significant transcription factor found was NF-Y which binds to the CCAAT-box. This motif scored particularly high on kurtosis, and it is known that functional binding sites for NF-Y tend to be located between 60 to 100 bp upstream of the TSS [44]. NF-Y is also known to cooperate with Sp1 to regulate some genes in response to cAMP [45,46]. The two factors NRF-1 and NRF-2 (nuclear respiratory factors) bind to different motifs, but both are ranked high and both have previously been implicated together with CREB in responses to raised levels of cAMP [47]. Interestingly, many of the sites for these two factors coincided with narrow peaks in the conservation track whose size matched exactly the width of the motifs. The fact that these sites were conserved while the flanking sequence around the sites was not is a strong indication that the sites might be functional.

Figure 6 shows the top ranking motifs from this analysis. The full table is available on-line at the MotifLab web site.



	ID	Name	Class	rank	total	support	p-value	Conservation	Kurtosis	Histogram	group	Logo
	M00917	VSCREB_Q4_01	1.1.2.0	1	49	29	2.110E-14	0.699	4.994		B	
	M00916	VSCREB_Q2_01	1.1.2.0	2	18	14	1.013E-9	0.775	10.268		B	
	M00179	VSCREBP1_Q2	1.1.1.6	3	25	17	4.591E-9	0.827	6.132		B	
	M00178	VSCREB_Q4	1.1.2.0	4	39	25	1.900E-11	0.723	4.538		B	
	M00338	VSATF_B	1.1	5	25	18	1.232E-10	0.719	3.255		B	
	M00513	VSATF3_Q6	1.1.1	6	14	10	2.490E-8	0.726	3.758		A	
	M00932	VSSP1_Q4_01	2.3.1.0	7	149	89	4.437E-115	0.402	4.185		B	
	M00931	VSSP1_Q6_01	2.3.1.0	8	337	153	1.954E-128	0.419	2.374		B	
	M00652	VSNRF1_Q6	1.1	9	41	30	9.273E-23	0.613	1.245		C	
	M01273	VSSP4_Q5	2.3.1.0	9	318	146	5.744E-143	0.423	2.009		B	
	M00017	VSATF_01	1.1	11	19	17	1.776E-7	0.660	4.071		B	
	M00196	VSSP1_Q6	2.3.1.0	11	166	94	2.249E-115	0.383	3.558		B	
	M00177	VSCREB_Q2	1.1.2.0	13	19	13	3.460E-7	0.562	4.734		B	
	M01258	VSGADP_01	3.5.2	14	22	21	1.474E-6	0.577	6.239		D	
	M00209	VSNFY_C	4.8.1.0	15	8	8	1.806E-5	0.612	7.557		C	
	M00981	VSCREBATF_Q6	1.1	16	83	43	1.688E-8	0.653	2.314		B	
	M00933	VSSP1_Q2_01	2.3.1.0	17	367	162	8.971E-179	0.370	0.950		B	
	M01253	VSCNOT3_01	unknown	17	283	128	1.129E-42	0.366	3.077		B	
	M00985	VSSTRA13_01	1.2.5.1	19	12	10	1.191E-6	0.478	5.364		C	
	M00287	VSNFY_01	4.8.1.0	20	25	22	1.224E-13	0.508	0.598		C	
	M00255	VSGC_01	2.3	21	125	84	8.424E-82	0.341	2.874		C	
	M00113	VSCREB_02	1.1.2.0	22	29	23	5.791E-4	0.534	8.082		B	
	M00982	VSKROX_Q6	2.3.2.1	23	37	31	1.033E-27	0.421	0.764		C	
	M01219	VSSP1SP3_Q4	2.3.1.0	24	219	111	1.773E-53	0.387	0.743		B	
	M00041	VSCREBP1CJUN	1.1.1	25	61	34	1.164E-5	0.695	0.913		B	
	M00108	VSNRF2_01	3.5.2	26	41	37	2.412E-5	0.514	2.541		D	
	M00008	VSSP1_01	2.3.1.0	27	332	166	4.154E-76	0.348	1.171		B	

Figure 6 Results from example 3 – Genes responding to forskolin treatment. Results from the forskolin-analysis output in HTML format. The table is a combination of results from four different analyses performed in MotifLab. The "total", "support" and "p-value" columns are from an analysis that counts the number of times each motif occurs in the sequences and estimates the significance of overrepresentation (significant p-values are highlighted in red). The "conservation" column is the average score taken from an analysis that compares the binding sites for each motif to a selected numeric feature (here conservation). The "kurtosis" and "histogram" columns are from an analysis of the positional distribution of the binding sites for each motif. The "group" column is from an analysis that compares the number of binding sites for each motif within sequences from two different groups to see if some motifs are overrepresented in one group compared to the other. Here we compared the group of upregulated genes to the downregulated genes. Motifs in the "A" and "B" groups (in red) were significantly overrepresented in the upregulated sequences whereas motifs in the "D" group (in green) were overrepresented in the downregulated sequences. Motifs in the "C" group (yellow) occurred at approximately the same rates in both groups. The table is sorted according to the combined ranks of p-values (ascending), conservation (descending) and kurtosis (descending). Note that almost all top ranking motifs are preferentially located within a narrow region upstream of the TSS, as indicated by the sharp peaks in the histograms around this position. Motif types are colour coded in the left-most column (CREB/ATF motifs with boxes in red, SP1 in blue, NF-Y in yellow, nuclear respiratory factors in green, others in grey).

Discussion

The examples given above, as well as previous publications by other groups, have shown that making use of additional information might boost the performance of motif and module discovery methods and help steer them towards regulatory elements that are more likely to be functional in a given context. However, relying on the "wrong" data, or even using data in the wrong context, can sometimes also have adverse effects. For example, filtering predicted sites based on phylogenetic conservation can lead to a higher proportion of true sites among the remaining predictions, but this will inevitably also remove any functional sites that are species-specific, and therefore not conserved. Even "gold standard" data, like DNase hypersensitive sites, should be used with some caution, especially when applied across different cell-types and conditions. To help users decide on which types of data might be useful to consider, MotifLab

includes several analyses to evaluate the merit of different types of information and to benchmark the performance of motif and module discovery methods. In fact, all the performance evaluations in the previous examples were performed within MotifLab, and the bar chart figures and ROC-curves included in this paper were produced directly from the analyses using the "output" operation.

Although many recent motif discovery tools can make use of additional data, they are often limited in what kind of data they can use and what they do with it, typically using information about known repeats to mask sequences or conservation to filter predicted binding sites. MotifLab allows users to incorporate many different types of data and use it in any way they like. No kind of information is treated as special compared to others by MotifLab, and information is represented with a few general data types. This means that it should be easy to also incorporate new kinds of data that might be available in the future.

The ability of MotifLab to process data in arbitrary ways using operations also sets this tool apart from most other motif discovery workbenches. The program has been designed so that users with some background in the field of regulatory sequence analysis should be able to rapidly learn how to perform standard tasks such as obtaining promoter sequences, annotating them with feature data and performing motif discovery or scanning. But it should also be relatively easy to perform more sophisticated pre- and post-processing tasks which would otherwise often require writing custom scripts. For the use case examples described in this paper, all the data processing steps involved in the analyses were performed within MotifLab itself.

MotifLab keeps all data objects in memory at all times rather than relying on external storage solutions. In addition, all operations are performed locally so most processing tasks will execute relatively fast. Visualization in the sequence browser is also very fast and responsive since the system does not have to wait for individual data segments to load from a server. This means that the tool has not been designed primarily to handle extremely large datasets (e.g. full genomes), although it is possible to apply it for genome-wide binding site predictions if sufficient memory is available. However, MotifLab is ideal for in-depth analysis of small to moderate datasets ranging from a single sequence to a few hundred (or even a few thousand) sequences, such as promoter sequences from groups of co-expressed genes. It is also very well suited for interactive, visual exploration of datasets and for rapid hypothesis testing.

Conclusions

Although vast amounts of genomic annotation data are now available to researchers who study transcriptional regulation, it is not necessarily trivial to make good use of this data for people who are not skilled in bioinformatics programming. The MotifLab workbench presented in this paper was designed to make it simple for users to obtain relevant data for sequences they want to study and to use this information in combination with existing motif discovery tools in many different ways. The utility and versatility of MotifLab was demonstrated through three practical analysis cases.

Availability and requirements

Project name: MotifLab
Project home page: http://www.motiflab.org
Operating system(s): MotifLab itself is OS-independent, but some external tools used by MotifLab for motif discovery etc. might only be available for some operating systems.
Programming language: Java 1.6
Other requirements: None
License: None
Any restrictions to use by non-academics: None

Additional file

Additional file 1: Supplementary methods and additional results. This supplementary file contains detailed descriptions of the procedure to generate and analyse the datasets used in examples 1 and 2, as well as results for individual datasets from those examples.

Abbreviations

Acc: Accuracy; ASP: Average site performance; AUC: Area under the curve; bp: Base pair; CC: Matthews correlation coefficient; F: F-measure; FP: False positive; PC: Performance coefficient; PPV: Positive predictive value; ROC: Receiver operating characteristic; Sn: Sensitivity (nucleotide level); sSn: Sensitivity (site level); Sp: Specificity; TF: Transcription factor; TP: True positive; TSS: Transcription start site.

Competing interests

The authors declare that they have no competing interests.

Author's contributions

KK designed and implemented the MotifLab software and drafted the manuscript. FD supervised the project. Both authors participated in the design and analysis of the examples presented in the paper, and both authors revised and approved the final manuscript.

Acknowledgements

Kjetil Klepper and Finn Drabløs were supported by The National Programme for Research in Functional Genomics in Norway (FUGE) and the Norwegian infrastructure for bioinformatics ELIXIR.no, both in The Research Council of Norway. The forskolin response data were provided by Kristine Misund, Liv Thommesen and Astrid Lægreid.

Received: 4 November 2012 Accepted: 10 January 2013
Published: 16 January 2013

References

1. Wasserman WW, Sandelin A: **Applied bioinformatics for the identification of regulatory elements.** *Nat Rev Genet* 2004, **5**:276–287.
2. Tompa M, Li N, Bailey TL, Church GM, De Moor B, Eskin E, Favorov AV, Frith MC, Fu Y, Kent WJ, *et al*: **Assessing computational tools for the discovery of transcription factor binding sites.** *Nat Biotechnol* 2005, **23**:137–144.

3. Okumura T, Makiguchi H, Makita Y, Yamashita R, Nakai K: **Melina II: a web tool for comparisons among several predictive algorithms to find potential motifs from promoter regions.** *Nucleic Acids Res* 2007, **35:**W227–W231.
4. Sun H, Yuan Y, Wu Y, Liu H, Liu JS, Xie H: **Tmod: toolbox of motif discovery.** *Bioinformatics* 2010, **26:**405–407.
5. Hu J, Yang YD, Kihara D: **EMD: an ensemble algorithm for discovering regulatory motifs in DNA sequences.** *BMC Bioinforma* 2006, **7:**342.
6. Wijaya E, Yiu SM, Son NT, Kanagasabai R, Sung WK: **MotifVoter: a novel ensemble method for fine-grained integration of generic motif finders.** *Bioinformatics* 2008, **24:**2288–2295.
7. Klepper K, Sandve GK, Abul O, Johansen J, Drablos F: **Assessment of composite motif discovery methods.** *BMC Bioinforma* 2008, **9:**123.
8. ENCODE Project Consortium: **An integrated encyclopedia of DNA elements in the human genome.** *Nature* 2012, **489:**57–74.
9. Won KJ, Ren B, Wang W: **Genome-wide prediction of transcription factor binding sites using an integrated model.** *Genome Biol* 2010, **11:**R7.
10. Pique-Regi R, Degner JF, Pai AA, Gaffney DJ, Gilad Y, Pritchard JK: **Accurate inference of transcription factor binding from DNA sequence and chromatin accessibility data.** *Genome Res* 2011, **21:**447–455.
11. Lahdesmaki H, Rust AG, Shmulevich I: **Probabilistic inference of transcription factor binding from multiple data sources.** *PLoS One* 2008, **3:**e1820.
12. Kuttippurathu L, Hsing M, Liu Y, Schmidt B, Maskell DL, Lee K, He A, Pu WT, Kong SW: **CompleteMOTIFs: DNA motif discovery platform for transcription factor binding experiments.** *Bioinformatics* 2011, **27:**715–717.
13. Kang K, Kim J, Chung JH, Lee D: **Decoding the genome with an integrative analysis tool: combinatorial CRM Decoder.** *Nucleic Acids Res* 2011, **39:**e116.
14. Herrmann C, Van de Sande B, Potier D, Aerts S: **i-cisTarget: an integrative genomics method for the prediction of regulatory features and cis-regulatory modules.** *Nucleic Acids Res* 2012, **40:**e114.
15. Aerts S, Van Loo P, Thijs G, Mayer H, de Martin R, Moreau Y, De Moor B: **TOUCAN 2: the all-inclusive open source workbench for regulatory sequence analysis.** *Nucleic Acids Res* 2005, **33:**W393–W396.
16. Homann OR, Johnson AD: **MochiView: versatile software for genome browsing and DNA motif analysis.** *BMC Biol* 2010, **8:**49.
17. Hu Z, Frith M, Niu T, Weng Z: **SeqVISTA: a graphical tool for sequence feature visualization and comparison.** *BMC Bioinforma* 2003, **4:**1.
18. Thomas-Chollier M, Defrance M, Medina-Rivera A, Sand O, Herrmann C, Thieffry D, van Helden J: **RSAT 2011: regulatory sequence analysis tools.** *Nucleic Acids Res* 2011, **39:**W86–W91.
19. Dreszer TR, Karolchik D, Zweig AS, Hinrichs AS, Raney BJ, Kuhn RM, Meyer LR, Wong M, Sloan CA, Rosenbloom KR, *et al*: **The UCSC Genome Browser database: extensions and updates 2011.** *Nucleic Acids Res* 2012, **40:**D918–D923.
20. Dowell RD, Jokerst RM, Day A, Eddy SR, Stein L: **The distributed annotation system.** *BMC Bioinforma* 2001, **2:**7.
21. Matys V, Kel-Margoulis OV, Fricke E, Liebich I, Land S, Barre-Dirrie A, Reuter I, Chekmenev D, Krull M, Hornischer K, *et al*: **TRANSFAC and its module TRANSCompel: transcriptional gene regulation in eukaryotes.** *Nucleic Acids Res* 2006, **34:**D108–D110.
22. Portales-Casamar E, Thongjuea S, Kwon AT, Arenillas D, Zhao X, Valen E, Yusuf D, Lenhard B, Wasserman WW, Sandelin A: **JASPAR 2010: the greatly expanded open-access database of transcription factor binding profiles.** *Nucleic Acids Res* 2010, **38:**D105–D110.
23. Spivak AT, Stormo GD: **ScerTF: a comprehensive database of benchmarked position weight matrices for Saccharomyces species.** *Nucleic Acids Res* 2012, **40:**D162–D168.
24. Hughes JD, Estep PW, Tavazoie S, Church GM: **Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae.** *J Mol Biol* 2000, **296:**1205–1214.
25. Liu X, Brutlag DL, Liu JS: **BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes.** *Pac Symp Biocomput* 2001, **6:**127–138.
26. Liu XS, Brutlag DL, Liu JS: **An algorithm for finding protein-DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments.** *Nat Biotechnol* 2002, **20:**835–839.
27. Bailey TL, Elkan C: **Fitting a mixture model by expectation maximization to discover motifs in biopolymers.** *Proc Int Conf Intell Syst Mol Biol* 1994, **2:**28–36.
28. Thijs G, Lescot M, Marchal K, Rombauts S, De Moor B, Rouze P, Moreau Y: **A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling.** *Bioinformatics* 2001, **17:**1113–1122.
29. Pavesi G, Mauri G, Pesole G: **An algorithm for finding signals of unknown length in DNA sequences.** *Bioinformatics* 2001, **17**(Suppl 1):S207–S214.
30. Gordan R, Narlikar L, Hartemink AJ: **Finding regulatory DNA motifs using alignment-free evolutionary conservation information.** *Nucleic Acids Res* 2010, **38:**e90.
31. Narlikar L, Gordan R, Hartemink AJ: **A nucleosome-guided map of transcription factor binding sites in yeast.** *PLoS Comput Biol* 2007, **3:**e215.
32. Gordan R, Hartemink AJ: **Using DNA duplex stability information for transcription factor binding site discovery.** *Pac Symp Biocomput* 2008, **13:**453–464.
33. Cuellar-Partida G, Buske FA, McLeay RC, Whitington T, Noble WS, Bailey TL: **Epigenetic priors for identifying active transcription factor binding sites.** *Bioinformatics* 2012, **28:**56–62.
34. Ernst J, Plasterer HL, Simon I, Bar-Joseph Z: **Integrating multiple evidence sources to predict transcription factor binding in the human genome.** *Genome Res* 2010, **20:**526–536.
35. Narlikar L, Gordan R, Ohler U, Hartemink AJ: **Informative priors based on transcription factor structural class improve de novo motif discovery.** *Bioinformatics* 2006, **22:**e384–e392.
36. Bailey TL, Boden M, Whitington T, Machanick P: **The value of position-specific priors in motif discovery using MEME.** *BMC Bioinforma* 2010, **11:**179.
37. Grant CE, Bailey TL, Noble WS: **FIMO: scanning for occurrences of a given motif.** *Bioinformatics* 2011, **27:**1017–1018.
38. Kulakovskiy IV, Boeva VA, Favorov AV, Makeev VJ: **Deep and wide digging for binding motifs in ChIP-Seq data.** *Bioinformatics* 2010, **26:**2622–2623.
39. Carvalho AM, Oliveira AL: **GRISOTTO: A greedy approach to improve combinatorial algorithms for motif discovery with prior knowledge.** *Algorithms Mol Biol* 2011, **6:**13.
40. Klepper K, Drablos F: **PriorsEditor: a tool for the creation and use of positional priors in motif discovery.** *Bioinformatics* 2010, **26:**2195–2197.
41. Sandve GK, Abul O, Walseng V, Drablos F: **Improved benchmarks for computational motif discovery.** *BMC Bioinforma* 2007, **8:**193.
42. Aerts S, Van Loo P, Thijs G, Moreau Y, De Moor B: **Computational detection of cis-regulatory modules.** *Bioinformatics* 2003, **19**(Suppl 2):ii5–ii14.
43. Seamon KB, Padgett W, Daly JW: **Forskolin: unique diterpene activator of adenylate cyclase in membranes and in intact cells.** *Proc Natl Acad Sci USA* 1981, **78:**3363–3367.
44. Mantovani R: **The molecular biology of the CCAAT-binding factor NF-Y.** *Gene* 1999, **239:**15–27.
45. Zhong ZD, Hammani K, Bae WS, DeClerck YA: **NF-Y and Sp1 cooperate for the transcriptional activation and cAMP response of human tissue inhibitor of metalloproteinases-2.** *J Biol Chem* 2000, **275:**18602–18610.
46. Cote F, Schussler N, Boularand S, Peirotes A, Thevenot E, Mallet J, Vodjdani G: **Involvement of NF-Y and Sp1 in basal and cAMP-stimulated transcriptional activation of the tryptophan hydroxylase (TPH) gene in the pineal gland.** *J Neurochem* 2002, **81:**673–685.
47. De Rasmo D, Signorile A, Papa F, Roca E, Papa S: **cAMP/Ca2+ response element-binding protein plays a central role in the biogenesis of respiratory chain proteins in mammalian cells.** *IUBMB Life* 2010, **62:**447–452.

doi:10.1186/1471-2105-14-9
Cite this article as: Klepper and Drabløs: MotifLab: a tools and data integration workbench for motif discovery and regulatory sequence analysis. *BMC Bioinformatics* 2013 **14**:9.