Artificial Intelligence and Bioinformatics

Jacques Nicolas

▶ **To cite this version:**

Jacques Nicolas. Artificial Intelligence and Bioinformatics. 2018. hal-01850570v1

HAL Id: hal-01850570

https://inria.hal.science/hal-01850570v1

Preprint submitted on 27 Jul 2018 (v1), last revised 18 Dec 2020 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Artificial Intelligence and Bioinformatics

Jacques Nicolas

Abstract The chapter shines a light on the strong links shared by Artificial intelligence and Bioinformatics since many years. Bioinformatics offers many NP-hard problems that are challenging for Artificial intelligence and we introduce a selection of them to illustrate the vitality of the field and provide a gentle introduction for people interested in its research questions. Together with the framing of questions, we point to several achievements and progresses made in the literature with the hope it can help the bioinformatician, bioanalyst or biologist to have access to state of the art methods.

1 Introduction

The links between Artificial intelligence and Bioinformatics are long-standing and strong. J. Lederberg, a professor of genetics from Stanford interested in exobiology, pointed out in the early 1960's the need for large biological equipments to be assisted by sophisticated programs [Hundley et al., 1963]. At a time when personal computers did not exist and the amount of data was measured in Kbytes, J. Lederberg shared working relationships with E. Feigenbaum and J. McCarthy and expressed visionary ideas in this small NASA report:

> In the experimental sciences ... applications will involve searching through data accumulations like spectra and other physical properties, the experimenter forming generalizing hypothesis and using the computer to test them against the file.
> Experienced insight into the role of the human and machine components of these systems should be invaluable in the further development of artificial intelligence, the programming of machines to simulate as far as possible those human cognitive processes that we begin to understand.These generalizations of human intellectual capability, and their delegation to

Jacques Nicolas
Univ. Rennes, Inria, CNRS, IRISA, Rennes, France,
e-mail: jacques.nicolas@inria.fr

> machines, continue to refine the strategic role of the human element and to give it increasing leverage in the solution of complex problems.

At the same time, Artificial intelligence was looking at complex, real-world problems to stimulate this science and get some measurable impact of its outcomes. The DENDRAL project, an expert system recognizing organic compounds from mass spectrometry data, was one of the first outcomes of these efforts [Lindsay et al., 1993]. It introduced task-specific knowledge about a problem field as a source of heuristics, and paved the way for proteomics studies. One of the first success twenty years later was Protean, a system that aimed to interpret NMR data to determine protein three-dimensional structures [Hayes-Roth et al., 1986]. It also initiated the development of realistic Machine Learning applications with Meta-Dendral, which was a system that learnt the rules necessary for Dendral from (spectra, structure) pairs [Feigenbaum and Buchanan, 1993]. Note that surprisingly enough, the assistance for the interpretation of mass spectra remains in high demand in biology, whether for studying small objects like peptides, glycans or various metabolites as well as for the study of larger molecular complexes. This mutual interest has not decreased over time and it is not an exaggeration to say that the contributions or the need for Artificial Intelligence may be found everywhere in bioinformatics.

1.1 A Major Application Field for Artificial Intelligence

Artificial Intelligence takes a close interest in the development of bioinformatics for at least two reasons. The first one dates back to the very beginning of Artificial Intelligence, when living organisms were used as a source of inspiration for the design of computational methods that were more robust to the treatment of real data, including the tolerance of uncertain and imprecise data, auto-adaptation and learning skills. Most of these methods form the basis of what is called soft computing: neural networks, evolutionary algorithms such as genetic algorithms or evolution strategies, immunological computation and various optimization techniques such as swarm or ant colony optimization, and uncertainty logics such as fuzzy set or possibility theory. The second one dates back to the beginning of the century, when Biology became a data-intensive field of science, with the emergence of numerous instruments scanning life at the molecular level.

In fact, Biology is not just a science of data, it is also a science of knowledge. From this point of view, there is a clear gradient from Physics to Chemistry then Biology in the complexity of structures and systems. First of all, living organisms store information in a discrete way in complex molecules and contain many symbolic machines to decipher the corresponding code (e.g. polymerases, ribosomes, and spliceosome). But life may also be characterized by the presence of many context-dependent causal influences related to biological functions [Walker and Davies, 2013]. A cell state is determined not only by its genetic code but also by its history (cells have a memory) and its environment, which strongly impact the

expression of genes: at any time in its life, only a small fraction of molecules in a cell will react. Biology is certainly the science *par excellence* of relations and dependencies, in the sense that it studies many different types of objects related by a number of different interactions appearing at different levels in a hierarchical structure of organization. The famous paradigm coined by R. Kowalski, "Algorithm = Logic+ Control" [Kowalski, 1979] could be rephrased as "Life = Logic+ Control" to emphasize the importance of these two components in all forms of life (although the control component does not simply define the computing strategy in this case but can have a feedback effect on the logic itself). It is thus a perfect application field of Artificial Intelligence for studies in knowledge representation, reasoning and machine learning. Moreover, Bioinformatics tackles numerous NP-hard problems that are interesting challenges for Artificial Intelligence.

Roughly speaking, the four main research tracks in Bioinformatics relate to statistics, algorithms on words ("stringology"), modeling (data bases, knowledge representation and system modeling) and combinatorial and optimization problems. The first one addresses issues that originate from the treatment of noisy observation data or from global effects of populations of variable elements like in population genetics. Although these questions are of interest to Artificial Intelligence, the contributions in this domain clearly derive from established methods in statistics. The second track led to many developments since sequences are the main source of data. Even expression data that measure the quantity of certain compounds are provided now in the form of sets of sequences. In this domain, the size of data – often many gigabytes - makes it essential to find linear analysis algorithms. Indexation techniques play a major role in achieving this goal. The last two tracks fully concern Artificial Intelligence techniques and still offer many interesting research challenges.

There were two possible ways to conduct a review on Artificial Intelligence and Bioinformatics: making a review, for each AI subfield, of existing or possible applications in bioinformatics, or making a review of problems in bioinformatics that are or would be interesting to AI people. The first choice would have been significantly imbalanced since Machine Learning and optimization in general occur in an overwhelming majority of papers [Baldi and Brunak, 2001; Bhaskar et al., 2006; Mitra et al., 2008; Zhang and Rajapakse, 2009; Inza et al., 2010]. Machine learning generally faces hard problems in Bioinformatics cumulating small sample size, individual variability, high dimensionality and complex relations between data. For this reason and probably also because biology is a science that is used to cross-check various hints to draw robust conclusions, ensemble learning (e.g. bagging, boosting or random forests, see Chapter 11 of Volume 1) has been particularly studied in this context [Yang et al., 2010]. It has been applied to various problems and showed interesting results each time, either for gene or protein expression data [Liu et al., 2010; Piao et al., 2012; Wu et al., 2003; Gertheiss and Tutz, 2009], the prediction of regulatory elements [Gordon et al., 2005; Wang et al., 2009; Lihu and Holban, 2015] and a large range of other applications such as the analysis of interaction data [Nagi et al., 2017], protein structure prediction [Bouziane et al., 2015] or automatic function annotation [Schietgat et al., 2010; Galiez et al., 2016; Yang et al., 2016a;

Smitha and Reddy, 2016].

As usual, applications are a driving force for methodological developments and interesting extensions of the ensemble approach have been proposed. In particular, the inclusion of feature selection in this framework has been identified as an important issue, since it is subject to instability [Okun and Priisalu, 2007; Awada et al., 2012; Pes et al., 2017]. Many studies are available on this subject, ranging from a bootstrap strategy sampling and ranking data before feature extraction [Yang et al., 2011; Wald et al., 2012] to the integration of several testing methods or selection methods [Sheela and Rangarajan, 2017; Brahim and Limam, 2017]. From the point of view of ensemble prediction techniques, Support Vector Machines (see Chapter 11 of Volume 1) have been often used, either during feature selection [Jong et al., 2004] or with different kernels [Guan et al., 2008] or combined with bagging for instance [Gordon et al., 2005]. In general, Random Forests seem more adapted to high-dimension data, a frequently occurring case in Bioinformatics [Diaz-Uriarte, 2007; Lertampaiporn et al., 2014; Pashaei et al., 2017] and to providing explanations to the predictions, an increasing concern in recent literature. Bagging and boosting seem more adapted to black box predictors and small sample size and can be usefully supplemented by transfer learning techniques [Mei and Zhu, 2014].

This chapter aims to give a more precise view of the major current or emerging bioinformatics challenges that are of interest to Artificial Intelligence in order to encourage further works on this topic. The abundance of papers in the field prevents an exhaustive presentation of all the various topics addressed. We have made a selection that we hope is representative of the variety of themes, trying each time to give the basics necessary to understand a specific problem, a formalization of this problem and a few achievements and progresses in relation to this problem. People interested in further reading could consult general reviews like [Hassanien et al., 2008, 2013] or articles that are oriented towards a particular subfield of AI such as agents [Keedwell and Narayanan, 2005; Merelli et al., 2007] or knowledge discovery [Holzinger et al., 2014]. For the field of Bioinformatics itself many introductory texts exist, see for instance [Singh, 2015; Ramsden, 2004]. Before reviewing the research work and getting to the heart of the subject, we begin with a brief introduction to the different types of data processed in bioinformatics.

1.2 Bioinformatics: Analyzing Life Data at the Molecular Level

Living cells are made of a large variety of molecules of different types performing different functions. Apart from water, the main constituents are biopolymers, i.e., molecules forming long chains of elementary components linked by covalent bonds, which can be considered at an abstract level as texts on fixed alphabets. There are four main classes of biopolymers: DNA, RNA, proteins and glycans.

The best-known kind of macromolecule is made of DNA and holds genetic information. DNA is made of four components (bases) represented by four letters (A, T, G, and C) that match together to form in most cases a double strand (A with T and C with G being the canonical matches). The DNA text is highly structured and includes regions coding for genes and others coding for the regulation of these genes. This structure differs depending on the absence of a nucleus in cells (bacteria and archaea) or the containment of DNA in a nucleus (eukaryotes). The order of magnitude of genome lengths is 10^6 bp (letters) for bacterial genomes and 10^9 bp for eukaryotes. Basically an organism's DNA is the same in all its cells but DNA is not a long quiet river: it can be modified by so-called epigenetic factors like DNA methylation or histone modifications. It is also well known that DNA can acquire mutations that cause genetic differences (polymorphism) between individuals. The most common are single nucleotide point mutations called SNPs (one SNP every 1000 bases for human individuals). However, mutations can occur throughout life on any cell's DNA and are then transmitted to the daughter cell lineage. All these transformations can potentially favor the appearance of diseases like cancer. An ancient but still topical challenge is the classification of species and the reconstruction from their DNA of their evolution scheme from ancestral species (see section 6). A main challenge on DNA is the annotation problem, which consists in discovering the functional content encoded by the nucleotide sequences: where are the genes and other genomic units and can we relate them to known elements ? For simple organisms it is routinely achieved by laboratories, using off-the-shelf software pipelines, but it remain a complex task for higher eukaryotic organisms (see section 3). A more recent challenge is the representation and management of all variations that occur between individuals or even individual cells, a challenge that is also central to RNA.

The second macromolecule, RNA, also made of four nucleotides (A, U, G and C), has been particularly studied in the last fifteen years and shown a key role in the regulation processes. RNA is thought to have been a precursor molecule to life on earth [Robertson and Joyce, 2012]. RNA is a biochemical mediator having both the capacity to store information as DNA and to act as a biochemical catalyst like proteins. The primitive quasi-life forms of viruses are made of RNA. It is in most cases a single-strand folded onto itself, giving rise to various structures in space related to their function. A number of RNA types exist (a typical eukaryotic cell can contain ten of thousands of RNA species) that are built from DNA through a process called transcription. In a given cell type, a specific part of the DNA is expressed in RNA, generally with multiple copies. One of the most important RNA species is messenger RNA (mRNA), expressed and matured from regions of the DNA called (protein) genes and acting as an intermediate for the synthesis of proteins (through a process called translation). It is estimated that in the human genome 85% of DNA is transcribed in RNA and 3% of DNA encodes protein-coding genes. To give an order of magnitude of cell complexity, approximately 350,000 mRNA molecules are present in a single mammalian cell, made up of approximately 12,000 variants with a typical length of 2 kb. The abundance of each variant can range from 10,000 to a few copies. Some mRNAs comprise 3Other types of RNA are used for gene

expression regulation, translation or RNA processing. The RNA level of expression of thousands of genes in a sample of cells can be measured simultaneously (a process called gene expression profiling) for different conditions and at different time points. A major challenge of RNA analysis is the determination of differentially expressed genes across experimental conditions, development stages, or healthy/pathological conditions [Han et al., 2015]. Beyond expression of individual genes, advanced studied are looking for gene sets [Huang et al., 2008] (see section 2) or even gene networks, either to represent co-expression networks or differentially expressed biological pathways [Khatri et al., 2012] (see section 5). Another challenge on RNA is the search of expressed gene variants that are variations from the same DNA coding region that produce alternative transcripts (alternative splicing) depending on the environment (see section 3).

The falling cost of sequencing technology for these two types of nucleic acid chains is having a marked impact on the biological research community. Essentially, sequencing projects have generally become small-scale affairs that are now carried out by individual laboratories.

Proteins, the third type of macromolecule are the main actors of the cell, performing a large range of functions ranging from cell structure (e.g. collagen) to metabolic reaction catalysis (enzymes), transport, cell signaling, and DNA replication. Proteins control the continuous component of life dynamics: essentially all biochemical reactions have continuous rates depending on protein expression. The building blocks of proteins are amino acids (AA, 20 amino acids in the genetic code) that are bonded together by peptide bonds. When two amino acids combine to form a peptide, water is removed and what remains of each amino acid is called a (amino-acid) residue. The order of magnitude of protein lengths is 10^3 AA. Proteins fold into 3D-structures called tertiary structures that depend on their amino acid sequence and determine their function. A protein tertiary structure is slightly flexible and may adopt several conformations but it may be described at a more abstract level in terms of simpler secondary structure elements (mainly alpha helices and beta sheets, the rest being generally classified as random coils) or, on the contrary, be assembled into dimeric structures or large protein complexes that function as a single entity. We know that the structure of proteins is crucial to understanding their role. Unlike nucleic polymers, even the sequence of proteins is hard to obtain since proteins are often transformed by post-translation modifications (e.g. the most common modification, glycosylation is estimated to occur in greater than 70% of the eukaryotic proteins, see the next paragraph on polysaccharides). Protein structures can only be obtained through timely and costly experiments and a long-standing challenge in bioinformatics is to predict this structure from the sequence (see section 4). Another particularly active field of research because of its importance for human health is the development of new drugs (see section 7).

A more recent topic from the point of view of bioinformatics relates to nonribosomal peptides (NRP), which are small chains of amino-acids (size less than 50 AA) that are essentially produced by microorganisms, using specialized large (complexes

of) peptide-synthetase enzymes instead of ribosomes. Their characteristic is to use a large alphabet (more than 300 variations of amino-acids have been identified, compared to the 20 AA in aminoacids) and have a linear, branching and/or cyclical structure. These natural peptides have many biological functions (antibiotics, immunosuppressants, antifungals, toxins) that are of high interest for medicine and biotechnology. In fact, there is also a largely unexplored space of ribosomal peptides that share many of the structural and enzymatic features of NRPs, paving the way for a discipline in its own right called peptidomics [Dallas et al., 2015].

The last type of biopolymer are polysaccharides, which are carbohydrate structures made of long chains of monosaccharide elements often called single sugars (mainly 4 types but many hundreds structural isomers and variants have been identified), typically joined with covalent glycosidic bonds. Polysaccharides are often a reserve of energy for organisms (e.g. starch may be converted into glucose) or can have a structural role (e.g. chitin used in composite materials forming the exoskeleton of insects or the shells of crustaceans and mollusks, or cellulose used as a dietary fiber). They are crucial to the functioning of multi-cellular organisms and are implied in protein folding, cell-cell interaction, immune response (antigen-antibody interaction) and epithelium protection, serving as mediators (or carriers) of many information transmission processes. Their structure can be linear or branched. Polysaccharides are particularly important in biology when associated with other elements and are referred to as glycans in this context. At an abstract level, glycans can be modeled by ordered labeled trees. The main glycans are found associated with proteins (glycoproteins or proteoglycans) and lipids (glycolipids). The modification of some protein amino acids after their translation by glycans is called glycosylation. The identification of glycan structures (sequencing) remains a complex experimental process, as is the identification of glycosylated sites in a protein. Unlike other biopolymers, progress in glycobiology is recent [Frank and Schloissnig, 2010]. Due to the importance of glycans in practical applications (drug targets, biofuels, alternatives to petroleum-based polymers), there is, however, no doubt that this field will experience major developments in the coming years.

When the entire set (or a large set) of elements of a certain type is available through high-throughput experiments, they are called omics data. For the four types described, the corresponding terms are genome, transcriptome, proteome and glycome. One could add another high-throughput source of information, the bibliome [Grivell, 2002; Sarkar, 2015], since millions of documents are published in Biology and can be considered as raw data out of reach of manual treatment (see next section).

Many biopolymers work by interacting with other biopolymers. For instance the regulation of gene expression is ensured by many mechanisms involving RNA (e.g., microRNA are small non-coding RNA molecules that interact with mRNA to regulate gene expression after transcription) or proteins (e.g. histones are proteins interacting with DNA in a structure called chromatine that organizes the very long DNA

chain of a chromosome into structural units). Proteins can act on DNA by attaching to a specific sequence of DNA adjacent to a gene that they regulate (transcription factor) or by generating complementary DNA (cDNA) from an RNA template (reverse transcriptase, a mechanism used by retroviruses). Protein kinases modify other proteins and are essential for the transmission of external signals within cells. All these interactions between numerous compounds occur within networks at different levels (in a compartment inside the cell, in the cell or in the extracellular matrix surrounding the cells,in a tissue or in a microbiota) and at different time scales and a great deal of research addresses this subject (see section 2 and 5).

2 Data and Knowledge Management

In molecular biology, the flow of data from multiple high-throughput observation devices combined with the results of (semi-)automated analyses of these observations is collected in many databases (the journal Nucleic Acids Research, which publishes each year a catalogue of such databases in a special issue, lists for instance 152 databases in 2017, 54 of which are new. [Galperin et al., 2017]). The difficulties of analyzing observations to obtain functional knowledge about the organisms studied are not discussed here. The complex process of functional annotation of data is indeed more a matter of software engineering than artificial intelligence. An exception is the use of multiple agents to ease this task [Jiang and Ramachandran, 2010]. Predictions made by machine learning methods are treated in other sections and the next section for instance deals with the annotation of genes.

This section is about data and knowledge management. The maintenance of all these databases quickly became a nightmare due to the rapid increase in data with different levels of quality, which requires frequent releases, changes in database schema and an updating process that grows generally quadratically with the number of organisms. Moreover, the question of integration of all these sources of data is a central concern for biology. It is managed through the creation of ontologies, the representation of graphs linking heterogeneous data and, to a small extent, through automated reasoning [Blake and Bult, 2006]. The stake here is to support and leverage the transition from a world of isolated islands of expertise knowledge to a world of inter-related domains.

In terms of data integration, the requirements are:

- Identify entities (unambiguously);
- Describe them (i.e., their properties and their relations with other entities) and ensure each element of a description is itself identifiable;
- Combine descriptions from multiple places (typically different aspects of the same entity);
- Support semantically reach querying and reasoning on descriptions.

Over the last decade, Semantic Web technologies such as RDF, SPARQL and OWL from the W3C have provided the infrastructure supporting the linked Open Data initiative [Bizer et al., 2009]. This has played a key role for integrating bioinformatics data [Cannata et al., 2008; Post et al., 2007; Bellazzi, 2014].

2.1 Information Extraction in Biological Literature

Despite strong and growing efforts to normalize data formats and collect observations in databases, a large amount of information relevant to biology research is still recorded as free text in journal articles and in comment fields of databases [Hirschman et al., 2002]. Since interactions are ubiquitous in biology, retrieving interactions between identified components is a very important task. This assumes that different naming variants of a same component have first been recognized and it is far from easy due to early lax practices: use of common names (e.g. *bag, can, cat, or, six, top* are all gene or protein names), use of synonyms (e.g. *can* as well as *n214* are aliases for *nucleoporin 214*), ambiguous names (e.g. the string *cat* may refer to enzyme *Chloramphenicol acetyltransferase* in bacteria, or a gene for the *catalase* enzyme in Human), or terms used in other biological contexts (e.g., *shotgun*, which can refer both to *DE-cadherin* gene or to a technique used for sequencing long DNA strands) [Chen et al., 2005].

An interesting recent development concerns the rapprochement between two communities, namely expert curators who read each publication to extract the relevant information in a standardized computationally tractable format and specialists in text-mining applied to biologically relevant problems. Manual curation is a very complex task that will likely need human experts over the long term, but it does not scale with the growth of biomedical literature (estimated in 2016 at more than 3 publications per minute for the main database of citations for biomedical literature Pubmed, which comprises more than 27 million citations). In 2007, the team working on RegulonDB, a carefully curated database on transcriptional regulation in E. coli, showed that 45% of the knowledge content could be extracted using rule-based natural language processing and that it allows for an additional 0.15% new rules to be proposed, of which a quarter was subsequently shown to be accurate [Rodríguez-Penagos et al., 2007]. The BioGRID database, which systematically curates the biomedical literature for genetic and protein interaction data has been involved in a coordinated effort with the BioCreative (Critical Assessment of Information Extraction in Biology) initiative to produce the BioC-BioGRID corpus, which contains a test collection of 120 full text articles with both biological entity annotations (gene and species) and molecular interaction annotations (protein and genetic interactions) [Islamaj Doğan et al., 2017]. This collaborative work allowed for guidelines for building text-mining systems that assist curation of biological databases to be proposed and fosters further research in this area. We extracted from the previous paper a list of four tasks in full text analysis that are particularly relevant in this context:

The curator assistance problem Given the full text of a research paper in biology, solve the following tasks:

- Recognition of evidence described in the current article vs information taken from other articles;
- Recognition of evidence which is supported by experimental data vs hypothetical or vague statements;
- Distinction between statements describing the layout vs statements describing the results of an experiment;
- Recognition of negative statements.

The OntoGene system is a state-of-the-art text mining system for the detection of entities and relationships from various items such as genes, proteins, chemicals but also drugs and diseases, implemented as web services for more flexibility. It provides standard text pre-processing tasks including identification of sections, sentence splitting, tokenization lemmatization and stemming, and can optionally perform syntactic analysis using a dependency parser (see Chapter 4 of this volume for more information on natural language analysis). It includes a module for entity recognition, using a rule-based approach, and disambiguation, using HMM-based learning from noisy data. This last point is crucial since it is hard to obtain a specialized dictionary in every domain. OntoGene has been applied to RegulonDB, a manually curated resource about the regulation of gene expression in E.coli [Rinaldi et al., 2017].

Going beyond the state of the art will imply also taking into account the variety of forms in which knowledge is available in papers. As was mentioned in [Rodríguez-Penagos et al., 2007], texts are not always sufficient and tables and figures and their captions contain what human curators need to generate relevant information. The recognition of evidence in particular can be greatly enhanced with such data. In this respect, the development of efficient and robust figure extraction methods [Clark and Divvala, 2016], able to scale to large databases of documents such as semantic scholar is certainly good news when it comes to fostering these studies.

2.2 Biological Ontologies

As Stevens et al. noted [Stevens et al., 2000],

> Much of biology works by applying prior knowledge [...] to an unknown entity, rather than the application of a set of axioms that will elicit knowledge. In addition, the complex biological data stored in bioinformatics databases often require the addition of knowledge to specify and constrain the values held in that database.

The knowledge we are focusing on is mostly symbolic. It should typically support comparison, generalization, association and deduction [Bechhofer et al., 2013].

Such knowledge is typically represented in ontologies which Bard defines as [Bard and Rhee, 2004] "formal representations of areas of knowledge in which the essential terms are combined with structuring rules that describe the relationship between the terms". In the early 1960s, the National Library of Medicine proposed a controlled vocabulary, Medical Subject Headings (MeSH), for the purpose of indexing journal articles and books in the life sciences and facilitate searching. It consists of hierarchically organized sets of terms that permits searching at various levels of specificity. the MEDLINE/PubMed article database and the NLM's catalog of book holdings. Available since a few years in RDF format, it currently contains over 28,000 descriptors accessible via 90,000 entry terms. Since then, a large number of ontologies have been developed in biology. A repository like Bioportal [Whetzel et al., 2011] was referencing 685 ontologies and 95M direct annotations in 2017. Moreover, a huge amount of data annotated by ontologies are now available via public SPARQL endpoints like the EBI RDF platform [Jupp et al., 2014], which is built on the OpenLink triple store technology and allows a programmatic access to these data.

The Gene Ontology (GO) is probably the best example of a significant development in biological ontologies [Ashburner et al., 2000], the number of papers citing this article (more than 20000) being a good indication of its importance. GO is both a standard terminology for describing the function of genes and gene products and a corpus of evidence-based GO annotations for gene products. In 2016, it contained more than 40,000 terms and 90,000 relations, over 600,000 annotations with an experimental evidence extracted from 140,000 papers and is also regularly revised by a dedicated team and requests by the scientific community. GO consists of three independent ontologies in the form of directed acyclic graphs (DAG): "molecular function" describing activities (e.g. catalytic or binding activities) at the molecular level, "biological process" giving programs accomplished by these activities, and "cellular component" where the function occurs. Each concept includes a term (recommended name), an identifier, definition (explanation and references) and synonyms. The DAG is essentially a tree with a few children having several parents. The relationship of child to parent can be either "is_a", "part_of", "regulates" or "occurs_in".

It is associated with GO tools such as browsing, SQL querying, and the Term Enrichment Service to find terms that are significantly more present in a set of product genes than by chance. It is also supported by external tools such as Blast2GO [Götz et al., 2008], dedicated to high-throughput functional annotation of genomic sequences. For the annotation of a new sequence, Blast2GO looks for homologous sequences in sequence databases with the comparison tool Blast, transfers the annotation of these homologous sequences and applies various rules to enhance the final annotations given relationships between the three subontologies and other databases on protein domains and pathways and using natural language text mining to simplify or structure the annotations. Since the management of ontologies has a strong technological component and uses a lot of engineering work, there are of course generic tools that are applied to GO as for instance ontologyX [Greene et al., 2017], a pack-

age for integrating ontologies in the R environment. As a data provider of growing importance, GO is not used solely for annotation purpose but as a source of features for prediction purpose. For instance, the issue of predicting the subcellular location of a protein has been addressed by machine learning methods including GO terms as discriminant features of protein locations [Mei, 2012; Li et al., 2012; Wan et al., 2014]. The idea in [Mei, 2012] is to retrieve proteins homologous to the target sequence by looking for matches against the InterPro protein signature databases (InterproScan), to extract their GO terms in the three subontologies, then to learn a kernel for a SVM to transfer the appropriate GO terms on the target protein and use them for predicting the location. The interest of using GO or other ontologies through transfer learning to enhance predictions has been confirmed in other studies such as the prediction of the associations between human genes and phenotypes based on human protein–protein interaction network [Petegrosso et al., 2017].

In recent years, the tendency has in fact been to transform the ontology into a true knowledge base by integrating other sources (for instance the database of molecular interactions IntAct), by adding axioms and biological reasoning abilities, and by working on more elaborated representations as for instance for the description of biochemical pathways [The GO Consortium, 2017]. There is thus a strong opportunity for the IA community to transfer and test some tools in this domain. The Web Ontology Language (OWL) is used in advanced versions of GO that include "has_part" and "occurs_in" relations and propose a fully axiomatized content giving access to cross ontology relationships. These other Open Biological Ontologies come from OBO, a consortium with an editorial committee that ensures coordinated development in biological and medical sciences of ontologies. They are designed to be interoperable and logically well-formed and to incorporate accurate representations of biological reality [Smith et al., 2007]. Thus, GO includes, for instance, links to (small) Chemical Entities of Biological Interest (Chebi [Hastings et al., 2013]) and a multi-species anatomy ontology (Uberon, [Mungall et al., 2012]). From the point of view of the integration of different ontologies, a central problem is that of alignment, where one tries to match the entities with each other [Shvaiko and Euzenat, 2013]. From the point of view of knowledge representation, a form of negation has been introduced in GO: when a gene product is expected to have a certain activity but it is known from experiments that it is not the case, it is emphasized with a Not qualifier [The GO Consortium, 2017]. Although there are currently very few negative annotations in GO, this is a clear advance with respect to reasoning. The full exploitation of biological knowledge expressed in OWL needs highly efficient reasoners on the underlying descriptive logics. A good example of such a framework is Aber-OWL [Hoehndorf et al., 2015], which uses the ELK reasoner. The main fulltext index of the scientific literature in Biology, PubMed, is built on top of Aber-OWL, and give an ontology-based access to more than 27M citations for biomedical literature from MEDLINE, life science journals, and online books.

The field of biological ontologies provides a large scale experimental field for researches at the crossroads of Semantic Web and Knowledge Bases. It is led by

a dynamic community that proposes many research issues and we will just point at two of them to conclude this section. *Tracking the inconsistencies* in such complex knowledge bases subject to many types of updates (new terms, obsoletion, new name for a term, term merge, etc.) is a research task of high importance. In [Chen et al., 2007] authors propose an ontology-based framework to detect the inconsistency in biological databases. This task is approached as a quality control problem in [Ghisalberti et al., 2010]. In [Park et al., 2011], as many as 27 databases are used to check GO and, besides syntactic errors, semantic inconsistencies are checked concerning redundancy and use of species-specific definitions. A more global Belief Revision approach is proposed in [Deagustini et al., 2016], where consolidations operators are built satisfying a fixed set of properties, based on kernel contraction and restoration and performed by the application of incision functions that select formulas to delete (conflicts). Although developed forDatalog$^{\pm}$ ontologies, these operators can be applied to Description Logics and it seems to be an interesting research direction for bio-ontologies.

As coined in [Matentzoglu et al., 2017], building ontologies using OWL is a difficult a, error-prone process and these errors have to be made explicit to authors. A general technique that seems to give good results in limited experiments is to improve the understanding of correct and efficient authoring actions by providing entailment set changes. In his perspectives, the GO consortium has also announced moving towards the *introduction of relations* between annotations for the function description of gene products in the context of a larger biological process. The new model, called LEGO for "Logical Extension of the Gene Ontology" is a neat progress towards the study of causality in biological networks. The LEGO formalism will define how different traditional GO annotations can be combined into a larger 'model' of gene and system function. Preliminary studies have been presented in [Huntley et al., 2014]. The idea is to associate to standard annotations (a pair single gene product *GP*-single GO term) a relational extension of the form $Relation(Entity)$ depending on the gene product, where *Relation* concern either a chemical (molecular relation) or any other entity like a cell type (contextual relation). This extension is interpreted internally as a relation $Relation(GP, Entity)$.

3 Gene and Non-Coding RNA Prediction

Gene prediction is a task that occurs at the beginning of a new genome annotation, just after completing DNA sequencing. It refers to the process of identifying the regions of genomic DNA that encode protein-coding genes or RNA genes. The prediction of protein genes of prokaryotic genomes (bacteria and archaea) is relatively easy since they usually appear without interruption in the sequence, as a single block named open reading frame (ORF). A remaining difficulty is the prediction of ribosomal frameshift events, a particular mechanism present in all organisms including viruses, which causes a shift by one or two nucleotides when translating the mRNA

code, thereby changing the ORF and the protein code. It is generally treated through HMM prediction [Kislyuk et al., 2009; Antonov and Borodovsky, 2010], but the difficulty to obtain experimentally validated frameshifts has reduced opportunities for model learning. The recently developed ribosome profiling technique (Ribo-Seq) that allows precise mapping of the locations of translating ribosomes on mRNAs should boost interest in research on this topic [Michel et al., 2012].

For protein of eukaryotic genomes, genes are made of a mosaic of blocks and an important subtask is the search of possible isoforms called splicing variations made of the combination of specific coding parts called exons, which are built by a special editing process during or just after transcription (transformation of the DNA code in a RNA molecule). Alternative splicing occurs for instance in half of the human genes, largely increasing the diversity of proteins and their specificity in different tissues. A gene is in this case the set of all the exons appearing in these isoforms. Splicing uses faint signals that are not completely understood and the prediction of alternative splicing variants remains a primary challenge in gene prediction. Moreover, part of the remaining sequence (called introns) can be used in some rare variants. In fact, the definition of a gene has evolved significantly with discoveries and it is likely that much remains to be discovered in this area and requires new developments in bioinformatics. Examples of recent advances in gene knowledge include chimeric mRNAs that are produced by joining exons from two or more different gene loci [Lai et al., 2015].

Historically, the development of gene prediction algorithms was based solely on the DNA sequence since the technology was still limited with respect to RNA sequencing. In order to have access to transcribed sequences, people were generating Expressed Sequence Tag (EST), short sequences of DNA synthesized from mRNA by special enzymes making the reverse transformation. EST are still in use for genetic studies of populations (simple sequence repeat (SSR) markers are ideal for this purpose). Since then, the RNA-Seq technology has given far more efficient access to transcribed sequences, introducing a small revolution for gene discovery, and other hints, such as mass spectrometry data, are also available to help finding protein-coding genes. Another evolution is due to the accumulation of genome sequences that enables the transfer of knowledge about these genomes to building new ones. We briefly review these aspects, starting with the analysis of DNA sequences. A review of practical tools is available in [Hoff and Stanke, 2015].

Main gene finders, such as GeneMark, GeneID, GlimmerHMM, AUGUSTUS, and SNAP, have been developed to recognize specific features in the DNA sequences (translation start site, sequence composition, splice site patterns, etc.) whose cumulated presence is signaling the presence of genes. They also frequently include the prediction of functional elements closely associated to genes, such as regulatory regions. In most cases HMMs form the basis for modeling these patterns, with tools having pre-built HMM models for several model species and that can tune the parameters of these probabilistic state models to a new genome by training them on a user-provided subset of known genes. The recent tendency is to include more com-

plete learning capacities in integrated frameworks: see for instance WebAugustus [Hoff and Stanke, 2015] and SnowyOWL [Reid et al., 2014] for Augustus and the work of Lomsadze et al. for Genmark [Lomsadze et al., 2005, 2014](Genmark-ET and ES).

The technology used to acquire RNA sequences has made major improvements over recent years, firstly through high-throughput sequencing of short reads (e.g. sequences of length 150), then much longer sequences (e.g. of length 1500). Despite these improvements, the state-of-the-art of transcript-based approaches for gene recognition is still unsatisfying, and there is surely room for advanced AI techniques in the analysis of these new data. This is mainly a combinatorial issue of assembling full transcripts from the fragments due to the diversity of situations (a giant jigsaw puzzle), the difficulty in quantifying the expression levels (solving a system of linear Diophantine equations) and the difficulty in recognizing and discarding long untranslated regions that may contain fragments close to protein-like coding sequences. The integration of RNA-seq data in the training of gene finders has been proposed in [Lomsadze et al., 2014; Hoff et al., 2015]. In [Pertea et al., 2015], transcript reconstruction and quantification are solved simultaneously, the quantification question being stated as a maximal flow issue on an alternative splice graph of overlapping fragments that authors solve with a specific breadth first search algorithm.

In the domain of health, individual variations (mutations) that occur in genes are known to be a major factor of diseases directly or indirectly, and the increasing access to individual whole-genome sequences could be the vector of a deep change in care strategy generally referred to as “precision medicine” [Aronson and Rehm, 2015]. As mentioned in the introduction to this section, splicing is an essential feature of eukaryotes. It is estimated that at least 90% of human genes have splicing variants. Some mutations directly impact exons and their translation into viable proteins, but missplicing due to mutations in introns is also an important source of human diseases. It is possible in some cases to align the RNA sequences on a reference genome and this strategy is often chosen in the case of the human genome and does not use artificial intelligence. The state-of-the-art in this domain has reached a high level of sensitivity [Medina et al., 2016], using the combination of an indexing method (Burrows-Wheeler Transform) in order to obtain a first set of high-quality mapped reads and a constrained dynamic programming search (Smith-Waterman) for resolving more difficult splicing variants. However, the problem remains difficult in the absence of a reference genome or with highly altered variants (tumour tissue sampling).

The prediction of splice sites (frontiers between introns and exons) include donor (exon/intron boundary) and acceptor (intron/exon boundary) splice site recognition and the recognition of a particular structure called branchpoint element. It generally occurs on a window of 100-150 nucleotides sliding on the sequence. Many machine learning methods have been used for the donor and acceptor prediction tasks, in-

cluding Neural Networks, Decision Trees, HMM, and Support Vector Machines, or a combination of them. For instance, [Wei et al., 2013] trains a first order Markov model to generate sequence features on the conditional probability of presence of each aminoacid within the site and outside the site. These and other features (e.g. trinucleotide compositions) are first subject to a feature selection step, then used to build a SVM model from a training set. It has been slightly refined by Pashaei et al. in a series of papers showing the difficulty to choose methods that have simultaneously a high accuracy, use a few parameters and are efficient[Pashaei et al., 2016a,b, 2017; Pashaei and Aydin, 2017].They first introduced a boosting algorithm (AdaBoostM1 with C4.5 for the weak learner), then introduced second order Markov models, then replaced the SVM and boosting method by a bagging algorithm (Random Forest classifier) and came back to Adaboost with another feature encoding scheme.

At a higher level, alternative splicing leads to multiple transcripts from a single gene, using different pairs of donor/acceptor splice sites. These splicing variants can be recovered by using EST data or RNA sequences [Pirola et al., 2012]. The basis for the representation of all variants is a *splicing graph*, i.e. a DAG whose vertices are blocks (maximal sequences of exons fragments that always appear together in all variants) and edges correspond to adjacency in at least one variant [Lacroix et al., 2008]. From a set of sequencing reads, it is possible to obtain a weighted graph, i.e.the number of reads(abundance) supporting each edge. Then the *transcriptome reconstruction problem* consists to find for each possible path (and thus each variant) in the splice graph an abundance compatible with these data and the possible errors in sequencing. It is an integer linear programming issue that has not always a solution but that seems to be solvable in practice if the list of all variants is given [Lacroix et al., 2008]. A variant of this graph has been proposed in [Beretta et al., 2014] where it is the vertices rather than the edges that are weighted, each weight being the size of the sequence associated to a block. Authors are looking for a minimal-weight splicing graph having the same set of k-mers (string of length k) than the set of reads.

4 Protein Structure Prediction and Computational Protein Design

The issue of predicting the structure of a protein from the mere observation of its sequence is one of the oldest challenges in bioinformatics and still warrants much study. Given a protein sequence and some fixed environmental conditions, the folding in space of this sequence is a deterministic process that leads to a specific conformation, up to small vibrations. The best source of protein structures, PDB (Protein Data Bank), contained 42,000 distinct proteins with known 3D structures in 2017. In contrast, the main source of protein sequences, UniProtKB, contained more than 500,000 reviewed entries and almost 100 M automatically annotated se-

quences. This gives an idea of the gap between experimental capacities and the needed knowledge of structure annotations.

The structure prediction research community is very dynamic both because it is a fundamental research issue relating to understanding the basic building blocks of life from a structural and functional point of view and also because it has, since 1994, been centered around a very challenging competition, a kind of "Olympic Games" known as CASP (Critical Assessment of protein Structure Prediction, http://predictioncenter.org). Targets for structure prediction are either structures about to be solved by X-ray crystallography or NMR spectroscopy and that will be available in the Protein Data Bank at the end of the competition. A hundred teams are participating in about ten prediction categories and results are published in the journal Proteins [Moult et al., 2018].

Since the prediction of 3D structures is a very complex issue, a number of subproblems have been defined. At the lowest level, one can look for secondary structures, where each position in the sequence is assigned a conformational state between a small number of possibilities (generally 3 classes, alpha helix, beta strand or coil). Other predictions can relate to physical measures such as solvent accessibility or localization information such as transmembrane regions or protein subcellular localization. It can also relate to particular bonds like disulfide bonds or hydrophobic interactions. Finally, a very useful intermediary level of representation for protein structures is made by *contact maps*, which points to close positions within the structure (distance less than a given threshold).

4.1 Secondary Structure Prediction, a Benchmark Model for Structural Bioinformatics

Secondary structure prediction (SSP) is the first step to understanding the 3D structure of a protein and is essential in the kinetics of protein folding. It is certainly one of the bioinformatics issues on which the highest number of machine learning methods has been tested, due to an early effort to standardize secondary structure states and accuracy measures and propose benchmark data sets. In [Yang et al., 2016b], 266 methods were counted between 1973 and 2016, a research effort that underscores the importance of bioinformatics problems for machine learning development. Currently, most methods are used to predict several structural parameters and, particularly, solvent accessibility.

Secondary structures are local regular conformations spontaneously formed as an intermediate during protein folding. They are defined by specific hydrogen bonding arrangements between the amino and carboxyl groups in the backbone of the molecule. The prediction is generally made on 3 classes (alpha helices, beta sheets and random coils, see section 1) and achieved on a window of about 15 aminoacids sliding on the protein sequence. The current tendency is to work on 8 classes. The interest of ensemble learning has been pointed out very early on this problem [Guer-

meur et al., 1999], for instance by using Multivariate Linear Regression to combine the scores of multiple classifiers.

Generally, a two-step procedure is used: one for a raw prediction of the class, and one for smoothing this prediction over the sequence, taking into account correlations between several positions. Decisive progress then came from the introduction of more domain knowledge into the prediction process. First, instead of only considering the protein sequences, the fact that all proteins are related through evolution has been taken into account by looking for known proteins similar to the one studied [Rost and Sander, 1993]. It not only served to improve the prediction score but also to define a position-specific reliability index pointing at regions of the proteins with high confidence predictions. Once similar proteins have been collected, a multiple alignment of their sequences provides information about the importance of each position that can be used in the classification procedure, for instance in the form of a profile HMM [Eddy, 1998; Cuff and Barton, 2000]. Together with an ensemble strategy, it has given birth to three important prediction servers based on neural networks, PSIPRED [McGuffin et al., 2000], Distill [Baú et al., 2006], and JPred [Drozdetskiy et al., 2015].

The current state of the art introduces another source of knowledge, the PDB, which has sufficiently grown to propose a large set of non-redundant protein chains (5800) with a known 3D structure and thus a known secondary structure. A simple strategy has thus been proposed in [Magnan and Baldi, 2014], where the basic prediction uses a bidirectional recursive neural network and the sequence regions with sufficient similarity with some of these chains are assigned with their majority secondary structure.

As stated in this paper, the problem seems virtually solved with such a strategy, but it is not the end of the story. Using existing structures as templates does not convey any information on the logics of folding, although it must exist since the number of possible structures seems very small as compared to the variety of sequences and the size of the conformation space. Any progress in prediction methods for SSP may benefit other more difficult prediction problems on protein structure, but this requires going beyond case-based reasoning to discover the folding logic. If no, it seems likely that the complete 3D prediction problem will remain hard to solve. This is why people continue to measure accurately the contribution of prediction methods without the input of structural information and even without the use of homology with other sequences.

The contribution of ensemble methods is reviewed for various classifiers in [Bouziane et al., 2015]. Standard deep learning methods do not seem to provide a significant gain with respect to other methods as is shown in [Spencer et al., 2015], which uses a belief network made of Restricted Boltzman Machines. In contrast, dedicated deep learning network architecture has been recently developed giving better results [Wang et al., 2017b, 2016c]. Studied in [Wang et al., 2016c] and available on the server RaptorX [Wang et al., 2016a], DeepCNF (Deep Convolutional Neural Fields) is a mix between a Conditional Random Field (input and output lay-

ers) and a deep Convolutional Neural Network. An in-depth analysis of the next required steps in SSP is provided in [Yang et al., 2016c]. DeepCNF makes it possible to take into account the correlations between secondary structures at different positions and to look at longer range correlations in protein sequences, which is a crucial point for SSP improvement. Indeed, beta sheets can link distant regions in proteins and it is a severe difficulty for window-based methods (the prediction error rate increases almost linearly with the number of non local contacts in proteins). In the same vein, another study has introduced the architecture LSTM-BRNN (Long Short Term Memory Bidirectional Recurrent Neural Network), and made it available on the server Spider3 with promising results. Another point is to consider meaningful features for SSP, such as intrinsically disordered (so-called chameleon sequences) or semi-disordered regions [Zhang et al., 2013], exposed parts of the protein (greater solvent accessibility in contrast with the hydrophobic core) [Faraggi et al., 2012], capping regions (boundaries of helices and sheets)[Midic et al., 2005] or proline conformation (proline is a particular aminoacid that is responsible for conformational heterogeneity) [Song et al., 2006]. The prediction can also be focused on more detailed secondary structures (8 states) or particular super-secondary structures like beta hairpins (see [Xia et al., 2010] for an SVM and ensemble-based approach), a motif of two consecutive beta strands that looks like a hairpin and are important in folding kinetics.

Our personal conclusion is that expertise and knowledge has become central to this problem, and it could be interesting to progressively integrate the relations found by a statistical approach in a reasoning framework. Early work along these lines, for instance in Inductive Logic Programming, which achieved wholly satisfactory results when it was proposed [Muggleton et al., 1992], certainly warrants new studies.

4.2 Folding in Space

The next problems after SSP in structural bioinformatics are the prediction of the *backbone structure of a protein*, which can be represented by a series of values for two angles (dihedral or torsion angles), and the prediction of contacts, typically at a distance < 8 Å. Indeed, SSP provides a rough classification into a few states that lacks precision, particularly at the interface between two secondary structures. A more relevant prediction level for the characterization or the multiple alignment of sequences on 3D structures is the level of local conformational parameters. Important atoms in the backbone of the protein are a central carbon, Cα, attached to the nitrogen N of an amine group, to the carbon C of a carboxyl group, and to a side chain specific to each aminoacid. Angles as well as distances are measured on Cα and sometimes C and N. These problems are much harder than the SSP problem and there is still room for significant improvements. A comparison of methods for predicting dihedral angles of a protein has been proposed in 2014 [Singh et al., 2014],

showing the best results for the method SPINE-X [Faraggi and Kloczkowski, 2017]. This method combines a discrete classifier and continuous real-value prediction of torsion angles trough Conditional Random Fields and is characterized by a six-step training approach that predicts successively the secondary structure, residue solvent accessibility, and torsion angles. The comparison paper also shows that part of the performance of the method is due to a simple representation trick that shifts the angles in order to have an easier separability of the trimodal distributions of angles along the three main SSP states.

Crossing results of multiple predictions of related structural parameters to enhance the prediction of one of them, possibly iteratively, is a general tendency that can be found in state-of-the-art studies [Heffernan et al., 2015; Li et al., 2017] using deep learning methods. Deep learning allows for easy integration of multiple features and, in this case, recurrent networks and introducing a prediction of the contact number gave the best result. Another interesting recent direction is to discretize (partition) the space of torsion angles into characteristic regions before prediction based on the observation (Ramachandran plot) that there are strong preferences on the possible dihedral angle pairs [Gao et al., 2017]. This idea is at the basis of *structural alphabets*[Offmann et al., 2007; Pandini et al., 2010], made of a complete set of elementary fragments extracted from the analysis of known structures and sufficient to describe all the conformations of these structures. For instance [De Brevern et al., 2000] proposed a structural alphabet of 16 elements describing the conformations of fragments of five residue length, learned in two steps using the principle of Kohonen Self-Organized Maps. Protein backbone reconstruction is achieved by a bayesian probabilistic aprroach, considering sequence windows of size 15. [Maupetit et al., 2006] proposed Sabbacc, an encoding of the protein trace in a hidden Markov model-derived structural alphabet of 155 elements describing the conformations of fragments of four residue length. Protein backbone reconstruction is achieved by a greedy algorithm assembling the alphabetical fragments in order to minimize an energy-based score.

The most detailed reduced representation of a protein structure is its *contact map*. A residue contact map is a graph that shows aminoacids with Cβ (the carbon in the side chain attached to Cα) at a distance less than 8 Å in the three-dimensional structure [Cheng and Baldi, 2007]. This information is useful for machine learning since it is invariant to rotations and translations and it helps a lot to retrieve the 3D structure [Vendruscolo et al., 1997; Adhikari et al., 2015; Pietal et al., 2015; Wang et al., 2016b]. Typically, around 5% of residue pairs are in contact in a protein and the number of contacts in a protein is only linear in its length. The Confold method [Adhikari et al., 2015] uses the iterative scheme described for torsion angle prediction to build a rough 3D model from secondary structures and contacts using a distance geometry algorithm, which then serves as an input to refine the contact and secondary structure prediction and produce a second refined model.

As stated in [Wuyun et al., 2016], most approaches in contact prediction are based on multiple sequence alignment (see section 6) and indices like mutual information between positions in the alignment [Dunn et al., 2008], since there are strong

evolutionary constraints for maintaining the protein structures. Machine learning methods may also be used to predict contacts from features extracted from multiple alignments.

A more recent tendency consists of combining the two types of methods and removing unlikely contact prediction corresponding to indirect coupling pairs obtained by transitivity or adding predictions that are likely to be missed with respect to typical patterns of contacts in proteins [Skwark et al., 2014; Jones et al., 2015]. [Skwark et al., 2014] presents PconsC2, a pipeline that uses first an ensemble method on 8 multiple alignments and 2 contact prediction methods, then corrects the prediction with deep learning through a 11x11 window providing information about the neighborhood of predicted contacts around each residue pair and a 5-layer feed-forward stack of random forest learners. The same type of approach appears in MetaPSICOV [Jones et al., 2015], using this time two stages of classic feed-forward networks, the first one learning contact pairs from 672 features extracted from multiple alignments and including an evaluation of the quality of this alignment, and the second one using from among 731 features a window 11x11 matrix of contact pairs predicted by the first stage. Physical constraints that reflect the sparsity of contacts and the presence of particular secondary structures can be added to evolutionary constraints to further reduce the search space. Interestingly enough, a method proposed in [Wang and Xu, 2013] first predicts the probability of any two residues forming a contact learning Random Forest built on about 300 features extracted from multiple alignments. In the second stage, it uses integer programming to check a set of hard and soft (that can be relaxed) constraints, trying to maximize the sum of probabilities minus the sum of penalties generated by violated soft constraints.

Among the most recent methods, a qualitative leap has been made in the study [Wang et al., 2017a] by modeling the contact map issue as a pixel-level labeling problem and using a deep learning model concatenating two deep residual neural networks. It simultaneously predicts the label of all entries in the contact matrix. Also far from the state-of-the-art, another recent work [D. Chapman et al., 2017] is interesting because it introduces a particular representation of the contact map problem as learning a logical circuit (deterministic Markov network), whose inputs are binarized features on the sequences and final outputs are a negative and a positive contact decision. The logic gates are elements of a genetic algorithm using point mutation, duplication and deletion as evolutionary operators and accuracy as fitness function. This preliminary work paves the way for more logical approaches to the problem.

Finally, the ultimate and most complex step in structural studies is the prediction of the *three-dimensional native structure of proteins*. Since the work of Anfinsen in 1973, it is regarded as a free energy minimization problem in a space of possible conformations, a hypothesis that seems to be verified with very few exceptions. This problem is too hard to be solved for most proteins, which have a high number of atoms and a corresponding space that can reach millions of degrees of freedom. In practice all methods work on so-called “coarse-grained” models, by reducing the number of considered atoms or generating pseudo-atoms abstracting some groups

of atoms [Blaszczyk et al., 2014]. One must distinguish between the number of degrees of freedom used for the representation of each aminoacid (typically from 1 to 3) and the representation itself. Often, the backbone of Cα atom positions is used as a degree of freedom and the other atoms or pseudo-atoms of the representation calculated from these positions. For instance, the CABS model uses a representation tracing Cα, Cβ, the center of mass of the side chain, and a virtual atom placed at the center of each bond between Cα, the last three values being computed from three consecutive Cα positions. Another interesting but less used possibility, the SICHO model, represents a backbone of pseudo-atoms tracing the side chain centers and calculates the Cα positions from three consecutive side chain positions. One of the finest models, Rosetta, represents proteins with 3 dihedral angles for each aminoacid, using a library of low energy backbone-dependent side-chain conformations called rotamers to constrain the set of possible side chains.

All these coarse grain models can be applied to numerous applications in structural biology. It is out of the scope of this chapter to provide a detailed account of all these applications, but we have chosen to present a specific folding abstract framework, called lattice protein folding, that allows us to rapidly test artificial intelligence methods without the need for an in-depth involvement in the physical, chemical and biochemical notions underlying this framework.

The protein folding in lattice models problem Coarse grain models have been developed in many studies within a discretized space, a grid or more generally a lattice [Mann and Backofen, 2014]. The lattice folding issue consists in finding a path of minimal score in the lattice such that (a) each residue is restricted to be placed on vertices of a lattice; b) each vertex is associated with at most one residue (self-avoiding walks); and c) a residue has to be placed in the neighborhood of the previous residue in the protein, this neighborhood being defined by a set of predetermined vectors. The cost function is generally derived from considerations on the free energy of the resulting molecule but is much simpler to compute than in continuous models. We are pointing to two types of lattice that are representative of the variety of models, the first one having been extensively studied:

The HP lattice: One of the simplest models of protein folding is the hydrophobic-hydrophilic (HP) model, which abstracts aminoacids in simply two states, hydrophobic (H, nonpolar) or hydrophilic (P, polar) and places them on a 2D square or a 3D cubic lattice. The score (energy function) is simply based on hydrophobicity: it is typically the number of nonbonding H in contact.

High-resolution lattices: Some authors argue that the main factor responsible for an observed folding are the constraints between side chains. For this reason, the SICHO model has been developed, taking the side chains as vertices and including a high number of possible interactions between these side chains [Feig et al., 2000]. The CABS lattice is the most refined

lattice that has been proposed [Koliński et al., 2004]. It has 3 interaction centers for each amino acid and a basis of 800 vectors for the Cα neighborhood.

A number of lattice models have been proposed that can represent more or less finely the natural folding of proteins, in particular with respect to secondary structures. Among the main difficulties of the search is the detection of symmetries leading to equivalent conformations [Gan et al., 2008]. Even in this simplest case, the problem is known to be NP-complete [Berger and Leighton, 1998]. Approximation algorithms exist however and folding in the cubic lattice may be achieved in linear time, for instance with an approximation ratio of $3/8$ [Newman and Ruhl, 2004]. A variety of local search methods have been tried as well as constraint methods (see Chapter 6 of Volume 2), which have proven to be very interesting in this respect [Backofen and Will, 2006; Mann et al., 2008]. A description of exact methods may be found in the review [Mann and Backofen, 2014]. Many variants of the HP model exist that enable for instance diagonals (triangular lattice), work in an hexagonal lattice [Shaw et al., 2014] or in a face-centered cubic lattice (fcc, one of the best models in this category [Pokarowski et al., 2003; Shatabda et al., 2014]). Other moderate resolution lattices exist, using a larger basis of vectors for the definition of the neighborhood. For instance, the "chess knight" model in 3D uses vectors $(\pm 2, \pm 1, 0)$. The 210 and 310 models use respectively 56 and 90 vectors.

Despite being one of the oldest models, the HP model continues to appear regularly in literature. For instance, in [Doğan and Ölmez, 2015] the problem is stated as a robot path planning problem where each amino-acid in the sequence is consecutively added to form continuous and self-avoiding amino-acid chains on the lattice. A new reinforcement learning method is applied to this planning problem where such methods are known to perform well (see Chapter 12 of Volume 1). Authors use for this purpose a compact state space representation and a distributed Q-learning algorithm (Ant-Q). In [Dubey et al., 2017], the authors propose an enhanced energy function for the square lattice model taking into account other interactions than H-H ones. A mixed integer programming formulation is presented in [Yanev et al., 2017], together with an exact algorithm and two heuristic algorithms. A swarm optimization algorithm and a Tabu search are combined in [Guo et al., 2017]. For a review of constraint programming in structural bioinformatics, see [Barahona and Krippahl, 2008]. Due to its complexity, the CABS model is closer to the continuous dynamic approaches. It can only be used in integrated environments that are predicting various structural properties like the presence of secondary structures and produce the corresponding constraints to restrain the general model [Blaszczyk et al., 2013; Gront et al., 2006].

We conclude this section by mentioning other important applications and point to some artificial intelligence techniques applied in this field.

The *homology modeling of protein structure* and the *protein threading problem* look both for the alignment of a protein sequence (the target) on a protein structure

(the template). The first method, as in the case of sequence-sequence alignments, uses an homologous protein of known structure for the template. Protein threading deals with the harder case where there is no hypothesis of the existence of a homologous protein. In this case, one relies on the fact that the number of natural foldings is very limited and that a library of core templates is available, usually consisting of a set of segments separated by fixed or variable lengths. For each template, the best alignment is obtained by optimizing an objective function scoring the compatibility between sequences and between positions in the template. The complexity of the problem depends on the chosen structural model and the amount of homologous sequences available. Finding the optimal alignment is NP-hard in the general case where there are gaps of varying length between the segments, and where the objective function includes interactions between neighboring amino acids in the structure. Many methods are based on extensions of the dynamic programming approach used for sequence alignment, adding constraints from knowledge on amino-acid preferences with respect to neighbors, mutations, solvent accessibility or secondary structures for instance. A typical program of protein threading is RaptorX [Peng and Xu, 2010], which uses a collection of regression trees to determine the scoring function of each alignment state in a Conditional Random Fields model that predict the state (match or gap) of each position in the alignment, and uses a neural network to rank the different alignments on the target and give a measure of quality for each alignment [Källberg et al., 2012].

The *protein docking problem* is a 3D matching problem: it entails finding minimal free energy conformations of a protein complex made of a protein receptor and a generally small molecule (a drug) called ligand or another protein. As in the previous problems, an efficient approach is template-based modeling, which uses the knowledge of an already known complex to guide the conformation search of the new complex [Xue et al., 2017]. For instance, case-based reasoning (see Chapter 10 of Volume 1) has been used in [Ghoorah et al., 2013] for this problem.

The *protein folding pathway prediction* problem looks at the analysis of the folding kinetics of a protein with a known 3D structure. It has been represented by roadmaps, which are graphs of conformations where each edge indicates a possible transition between conformations [Moll et al., 2008]. Roadmap-based methods were originally developed in robotics for collision-free robot motion planning and vastly extended and adapted for folding.

Finally, *computational protein design* is the process of designing new protein sequences with a fold close to a target protein structure. The ultimate goal is protein engineering, i.e., the design of new molecules adapted to target new materials or new functions, like for instance enzymes that are critical components in bioengineering and biomedical applications [Coluzza, 2017]. Rational protein design makes massive use of libraries of rotamers and can be seen as an optimization problem based on complex energy functions. In this domain, the exact solving method at the basis of many developments is a special dominance search algorithm (dead-end elimination), followed by an A* algorithm. Linear Programming, Quadratic Programming, Weighted Partial MaxSAT and Graphical Model optimization can be used to solve this problem, but a special form of Weighted Constraint Satisfaction formulation

(Cost Function Network, see Chapter 7 of Volume 2) proves to be very efficient for this task [Allouche et al., 2014; Traoré et al., 2016] .

5 Network Modelling

As emphasized in a recent editorial of a special issue of PLOS Computational Biology on biological networks [Ideker and Nussinov, 2017], networks are everywhere in biology from molecular interaction circuits or modules to ecosystems, and have become a major mode of analysis in bioinformatics studies. Networks make it possible to understand biological entities at the system level, explaining diseases or drug effects as a cumulative result of small effects of individual genes of a cell for instance. This global understanding is out of reach of scientists from the mere observation of components for relatively small systems due to non linearity/additivity of regulations or interactions and it is even harder in current developments, which are designed to model genome-scale systems. It is thus crucial to assist biologists in this task not only through the development of simulators, but also for the analysis of a network behavior in plain intelligible language, i.e., is by checking properties and finding causal explanations.

At a higher level of organization, the interactions within bacterial consortia or in host-microbial symbiosis are and will certainly be the subject of a growing number of studies. Networks are present as observed data and as an abstract graph data structure (associated with a set of dedicated methods) that represent physical structures, as well as the dynamics of living components. If one considers the kinetics of components, the time scale may vary a lot depending of the type of interaction. For instance, it can range from milliseconds to seconds between the first and last stages of a signaling cascade (activation pathway from cell surface receptors to molecules controlling a cell function such as cell division) from seconds to minutes for metabolic reactions (biochemical reactions occurring in a cell) and reach hours for regulation processes (e.g. cell mechanisms used to increase or decrease the production of specific gene products). Signaling, regulation and metabolic networks are the three main types of networks studied in cells.

The representation of networks has used many formalisms that may be characterized as discrete (qualitative) or continuous models. We will focus here on discrete modeling since artificial intelligence methods are more directly applicable in this case. The early works of R. Thomas S. Kauffman in the 1970's have have demonstrated the value of logical modeling for representing gene regulation mechanisms in cells, by seeing them as discrete dynamical systems. The gene expression levels may be abstracted using Boolean variables (e.g. active or not active), time using logical steps and the expression changes using logical functions over the set of interacting genes. This simple but powerful framework is formalized in the following definition:

Definition 1. Boolean Networks are made of a graph $G = \{g_i, i = 1,n\}$ of Boolean variables. An edge $e_{ij} \in G$ represents the fact that variable g_i is one of the inputs to variable g_j. Each node is associated with a logical formula (a transition function) giving the output value of its Boolean variable with respect to the value of its input variables. A state is a vector of values over the complete set of variables. The state transition graph (STG) is the graph of all possible transitions between states, based on an update rule. Using formulas attached to variables to compute their new value, two main update rules are used in practice: the synchronous rule, which updates all variables simultaneously, and the asynchronous rule, which updates one variable at a time.

It is possible to extend this framework to multi-valued variables in order to take into account the possibility of multiple thresholds.

Numerous studies have used this formalism on biological applications. Several reviews are available on this topic [Albert and Thakar, 2014; Abou-Jaoudé et al., 2016] and on modeling the dynamics of cellular networks [Le Novere, 2015]. Some repositories are emerging on the web ([Chelliah et al., 2015; Klarner et al., 2017]) and an informal consortium, CoLoMoTo (Consortium for Logical Models and Tools) [Abou-Jaoudé et al., 2016] is working on the definition of standards.

The extraction of networks from scientific literature has already been mentioned in section 2. It is also possible to automatically refine such an initial prior knowledge network using experimental data, such as gene expression data for gene regulation networks [Lim et al., 2016] or phosphoproteomics data for signaling networks [Videla et al., 2015]. The first paper uses a form of swarming hill climbing strategy, whereas the second one produces all possible solutions through a combinatorial approach code in Answer Set Programming (ASP). For metabolic networks, the reconstruction of the set of biochemical reactions in a newly sequenced organism can be inferred from the sequenced and annotated genomes [Karpe et al., 2011]. It works by recognizing in the new genome the enzymes catalyzing each reactions, then using the knowledge of previously known pathways from other organisms. To predict unknown or alternative pathways, it is possible to reason on atomic transfers [Boyer and Viari, 2003].

Once constructed, the metabolic network is usually too large to be analyzed directly. A general technique to reduce it is based on the description of the properties that one wishes to keep between the initial network and the reduced network and then to automatically infer the minimum possible reduced network. This was proposed in [Röhl and Bockmayr, 2017], where the authors developed a mixed-integer linear programming (MILP) approach for computing this reduction.

The analysis of networks may be achieved from a topologic, static point of view, or from a kinetics-centered, dynamic, point of view.

In metabolic networks, static analysis includes the search for two close concepts, elementary modes and minimal cut sets [Acuna et al., 2009].

Elementary flux modes (EFMs) analyze networks from a pathway-oriented perspective. They are the minimal sub-networks (with respect to set inclusion of reactions) that can function (stoichiometrically and thermodynamically feasible) in steady state. These modes can help reveal the capabilities/objectives of a cell metabolic network, that is, the matching between phenotype and genotype. The standard approach to solve this problem is to reduce it to the enumeration of extremal rays in a pointed polyhedral cone, a standard problem in computational geometry. Unfortunately, the number of EFMs can increase exponentially with the network size. An interesting question is then how to navigate into the solution space instead of enumerating the solutions. To this end, the authors of [Martin et al., 2016] allow the user to add/remove boolean constraints on the solutions that interest them. They use then a Satisfiability modulo Theory solver, CDC4, to solve both the boolean constraints on reactions occurring in the solutions and linear constraints taking into account stoichiometric data and steady state fluxes.

A minimal cut set (MCS) is a minimal (irreducible) set of reactions in the network whose inactivation will definitely lead to a failure in certain network functions (see the paragraph on perturbation analysis at the end of this section) [Klamt, 2006]. It helps to identify crucial, fragile parts in the network structure and to select suitable targets for repressing undesired metabolic functions.

Studying the dynamics of a given Boolean network entails three main issues on the associated state transition graph $STG = (S, T)$:

The state transition graph search problem

Attractors: Find all minimal subsets of states $A \subseteq S$ that are trap domains, i. e., such that the transition from an element of A yields an element of A. As an important particular case, stable or steady states are attractors reduced to a singleton and do not depend on the update scheme. The others are called cyclic attractors. The detection of stable or cyclic attractors is NP-hard [Akutsu et al., 2012].

Reachability: Check the absence/presence of specified trajectories in STG, i.e., such that there are paths in STG whose elements s_i belong to a specified subset S_i of S.

Perturbation analysis: Check the effect of fixing some variable values or some logical function values on the attractors and reachability properties.

Stable state atractors have been shown to correspond to identified cell differentiation states, a good example being the study of the regulatory and signaling networks associated with Th-cell subtypes differentiation [Naldi et al., 2010]. Cyclic attractors have been shown in the cell cycle networks of Yeast or Mammals (see e.g. [Barberis et al., 2017; Traynard et al., 2016]). Attractors provide, more generally speaking, pointers to the possible steady functioning modes of the studied systems, either in normal conditions or under degraded conditions. They are also a way of

checking the quality of a model by comparing attractors with the observed behavior of the biological system.

The detection of stable or cyclic attractors is NP-hard [Akutsu et al., 2012]. A number of algorithms have been proposed to find attractors of a Boolean network, mostly using (reduced-order) binary decision diagrams. Quite naturally, a SAT-based bounded model checking method has been experimented in the case of synchronous networks [Dubrova and Teslenko, 2011]. Basically, the idea is to search by unfolding the transition relation for paths of bounded length p in the STG, to increase p if there is a path that is not a circuit and to mark the variables that are part of a circuit as attractors and exclude them from possible paths.

In practice, the knowledge of the formula associated with each gene and of the update scheme may be incomplete. The system ASP-G [Mushthofa et al., 2014] uses a higher level language for describing the network and the update scheme (with predicates such as activates, inhibited or changed) and relies on Answer Set Programming solvers to search efficiently the STG based on the previous SAT-based approach. The authors use incremental solving on the path length and, once a solution is found, the attractor is removed from the search space by adding a new constraint. In [Abdallah et al., 2017], ASP is also used for more ambitious work extended to both synchronous and asynchronous update modes and considering automata networks instead of Boolean networks, a framework that enables multivalued domains for variables and where more sophisticated update rules like nondeterministic synchronous mode can be introduced. In its current state, the program works in incremental mode (looking for paths of fixed size) and can produce several times the same attractor described by several cycles.

Another interesting approach [Klarner et al., 2015] looks at minimal and maximal trap domains represented by partial states (with free variables) instead of attractors and computes them efficiently from the set of prime implicants of the Boolean formulas. An insightful paper by K. Inoue has established a deeper relationship between Boolean networks and Logic programs [Inoue, 2011] (Logic programming is described in Chapter 4 of Volume 2). Stable sates of a network may indeed be characterized as supported models of the corresponding logic program, a correspondence that allows a trivial coding of their search.

The reachability problem is basically searching trajectories in a state transition graph of size 2^n, where n is the number of Boolean variables and is a perfect application field for model checking and temporal logics. All studies in this field formalize this graph with a finite state transition system (FSTS) that is easily mapped into a Kripke structure. This framework is very general and can be applied far beyond the case of Boolean networks. In the field of biological systems, fine simulation of their dynamics by piecewise linear differential equations can, in particular, be considered. The principle is to associate with states hyper-rectangles in the concentration space [Batt et al., 2008] and a tool like *GNA* allows us to generate and export the $FSTS$ from differential models [Batt et al., 2012]. See [Carrillo et al., 2012; Brim et al., 2013] for reviews of existing tools. Once discretized, the state space can be

explored via a logic allowing branching time (states with more than one immediate future) like CTL (Computational Tree Logic) or the μ-calculus, a practical issue being that it helps biologists to formulate their questions. This generic question in model checking has been addressed in [Monteiro et al., 2008] through the development of a specific language of patterns with place-holders that allows the use of predefined state descriptors (e.g. increases or isSteadyState) and predefined types of questions (e.g. possibility of occurrence of a pattern or if-then statements). Specialized model checkers have been developed for the study of genetic regulation networks such as Antelope [Arellano et al., 2011], which addresses the important question of not just checking but exhibiting the states with a given property, using a Hybrid CTL with state variables. The size of the state space remains a crucial parameter for model checkers and the current state of the art is still too limited to query all the known models.

The third problem on Boolean networks, perturbation analysis, has been mainly applied to Probabilistic Boolean Networks, a stochastic extension of the standard framework where the transition function is replaced by a set of possible functions with an associated probability distribution. One of the main objectives of the analysis is focused on intervening in biological cell dynamics in order to alter the gene regulatory network or the signaling network and avoid undesirable cellular states, particularly in the search for a therapeutic strategy (e.g. to counteract the development of cancerous cells). The difficulty is how to bypass the inherent living system's robustness that uses many redundant pathways, while avoiding side effects and thus looking to minimize the necessary changes. Finding an optimal control strategy leading to a desired state by changing some variable values is NP-hard [Akutsu et al., 2007]. It is also possible to act on the transition functions by altering the transition probabilities or flipping a minimum number of values in the truth table [Xiao and Dougherty, 2007]. People are often using quantitative approaches to solve this problem by finely tuning the system's kinetic behavior and the use of Artificial Intelligence techniques on this problem is less well developed.
The problem has been set within a three-valued logical framework (to represent knock-out, knock-in and no intervention operations) in [Samaga et al., 2010]. It defines *intervention problems* made of a set of pairs (G,C), where G is a goal made of desired values for some target species (e.g. genes or reactions) and C equates to environment constraints setting some other species to fixed values. The issue is then to find (subset-)minimal intervention sets (MIS), i.e., a set of values for a set of species S such that all goals G are satisfied in their context C and at least one goal is not satisfied if an element of S is removed. The authors introduce a dedicated breadth-first search algorithm and emphasize the importance of preprocessing to reduce the dimension of a practical problem by finding classes of equivalence containing interventions having the same effect on target species. An ASP encoding is proposed in [Kaminski et al., 2013] to enumerate all MIS for real-world signaling networks, showing that negation by default and recursive definition of reachability are valuable tools for searching for larger intervention sets and potentially solving the unbounded problem where the size of the intervention sets is not bounded. In

practice, the ideal network is not known and it would be interesting to look for solutions that are compatible with several alternative networks explaining the same system.

Apart from signaling networks, metabolic networks have also been studied from the point of view of control, the interventions consisting of deleting reactions and/or regulating the reaction fluxes. One of the main industrial opportunities of such control is the optimized production of some target compounds by microbial organisms, a process called metabolic engineering, which is a research axis of synthetic biology. The MIS problem is transformed into a very similar Minimal Cut Set problem (MCS) [von Kamp and Klamt, 2014] or Regulatory MCS (RegMCS) [Mahadevan et al., 2015] and Mixed Integer Linear Programming models have been reported in these publications in relation to solving MCS and RegMCS.

In contrast to the previous problem, the use of perturbations to learn the network has been the subject of many publications. Very few studies address both problems, one exception being the toolbox caspo [Videla et al., 2017], which proposes functions dedicated to each problem.

6 Understanding Evolution

The study of evolution is certainly a main topic of interest for biologists and bioinformatics has had a huge impact in this field since the advent of sequencing techniques. It is not merely of interest to evolutionary biologists and for the study of biodiversity: evolution is a fundamental mechanism that helps to solve hard problems and obtain reliable answers at the level of populations but also in respect of structural or functional biological issues. It is thus also a key issue for bioinformatics. The term Evolution refers essentially in biology to changes in the inheritable traits of populations which occur over generations. Since easy access to the molecular content of living things, evolution can be considered at a much finer level than before, and evolution can even be observed at the individual level. A particularly important recent application relates to the evolutionary process in cancer. A tumor is an evolutionary process. It starts from a single cell and evolves with an anarchic development, including somatic mutations. Technology allows us now to sample a tumor and try to retrieve the history of tumoral cells, a combinatorial problem that may be treated by a phylogenetic approach [Popic et al., 2015; Malikic et al., 2015; Caravagna et al., 2016; Schwartz and Schäffer, 2017].

The most cited papers in all research fields, as stated in news published in the journal Nature in 2014 on the top 100 papers cited since 1900, are, together with the Sangers Sequencing method and amplification methods, two papers on phylogenetics and two programs in this field. Moreover, phylogeny makes extensive use of sequence comparison and programs in this field are well represented with two versions of Blast and two versions of Clustal being cited (the multiple sequence

alignment method on ClustalW is ranked 10th). It is why we start this section with a description of multiple alignment.

6.1 Multiple Sequence Alignment

At the core of many problems involving sequences in biology (sequence assembly, functional or structural annotation, homology search and phylogeny) lies the issue of sequence alignment. Its goal is to line up the letters in several sequences in order to exhibit a maximal similarity between letters at the same position. It may concern protein, DNA or RNA sequences, which are under a selective pressure. Indeed, this problem stems from the fact that all species originate from a common ancestor: sequences are assumed to be on the leaves of some unknown common evolutionary tree and thus share common characteristics that have become blurred by various mutation and insertion/deletion events. This is the reason why multiple sequence alignment and phylogeny are closely related, although other aspects like structural or functional properties may be taken into account. An alignment of a set of sequences helps to recover the evolutionary tree of the species they come from and, conversely, a known or assumed evolutionary tree (the guide-tree) helps to recover a relevant alignment of sequences. In fact, a growing number of authors are trying to build sequence alignment and phylogeny simultaneously [Ng et al., 2017]. Note that the issue of sequence conservation modeling is not reduced to multiple alignment. If one is interested in sequence annotation or functional prediction, multiple alignment can usefully be extended to a pattern recognition problem (HMM profile) or even to grammatical inference studies (automata), working with a more expressive syntactic model of conservation [Coste, 2016]. Moreover, multiple alignment takes into account evolution events such as point mutations, insertions and deletions, but more complex events can occur in a genome such as duplications, inversions, or recombinations, and there is still a lot of work to be done to address all these sources of variability.

Formally speaking, the Multiple Sequence Alignment (MSA) issue is as follows:

Definition 2. Given a set of sequences $S = s_1, ..., s_n$ on a finite alphabet Σ, an MSA of S is a set of sequences $A = \{a_1, ..., a_n\}$ on $\Sigma \cup \{"-"\}$, where $"-"$ is a new letter representing insertion/deletion events (gaps) in the aligned sequences. Moreover, all elements of A have the same length and all a_i are equal to s_i up to the deletion of the $"-"$ characters. Given a cost or score function c on pairs of sequences in an MSA, the MSA issue looks for an MSA A optimizing the value of a function of c.

A frequent choice in this general definition is the Sum of Pairs criterion (SP) that minimizes the sum $\Sigma_{i<j}\ c(a_i, a_j)$, where c is typically defined as a sum along the alignment positions of a scoring of letters at these positions and the gaps receive a special treatment with an affine function. In this setting and its variants, MSA is an NP-complete problem and, for this reason, using a standard dynamic programming

approach provides an exact solution in practice only for a very small number of sequences, typically no more than 3. Artificial intelligence took an early interest in this problem since it can be reduced to finding a shortest path in a huge graph joining the possible alignment positions of characters in each sequence and a branch and bound algorithm can be applied [Gupta et al., 1995]. Variants of the A* algorithm (see Chapter 1 of Volume 2) have been developed by Ikeda et al. and Yoshizumi et al. both for the exact and the approximated case [Ikeda and Imai, 1999; Yoshizumi et al., 2000]. This gave rise to a number of papers on space-efficient or faster heuristics [Zhou and Hansen, 2004; Korf et al., 2005; Schroedl, 2005], or the recursive best first search MREC enabling the exact optimal alignment of up to 11 sequences [Koshino et al., 2006]. Recent works are focusing on solutions using external disk space, adapted to the best first search order, and multi-threaded computation [Hatem and Ruml, 2013; Sundfeld et al., 2017], pushing further the limits of exact methods.

A number of suboptimal approaches have been developed and are still to be developed, for instance to scale to large sets of sequences (million) or large sequences (whole genomes). A basis of almost all these approaches is the use of a progressive strategy, starting from pairwise alignments and trying to combine them in the best ordering. Following the works of Korostensky and Gonnet, it is possible to define the progressive alignment using a circular sum measure where each sequence is aligned with exactly two sequences [Korostensky and Gonnet, 1999; Gonnet et al., 2000]. Multiple alignment is reduced this way to a Traveling Salesman Problem, where the goal is to find the circular ordering of sequences minimizing the sum. Although this approach has some relevance with respect to evolution, it appears to have not been pursued apart from a small piece of work in [Abu-Srhan and Al Daoud, 2013]. The progressive strategy is often combined with an iterative strategy, where solutions are progressively refined in order to improve the alignment. Stochastic optimization is useful in this case. In [Omar et al., 2005], a genetic algorithm is used for the progressive part and a simulated annealing algorithm is used for the iterative part. Many aligners have been developed but none outperforms the other in all cases, depending on the properties of sequences (presence of domains, partial sequences, intrinsically unstructured regions, alternatively regions with known 2D/3D structure, etc.). A natural approach is then to propose meta-methods running a number of algorithms in parallel and choosing the best alignment in the different results [Muller et al., 2010]. More recently, the concept of assisted multiple alignment has emerged as an important issue for more efficient and more relevant alignments. AlexSys is an expert system in protein multiple sequence alignment that learns rules predicting for each method if it is suited or not to the sequences to be aligned [Aniba et al., 2010].

6.2 Building Phylogenetic Trees

Given a set of species (or taxonomic units), each one being usually represented by a subset of its gene sequences, molecular phylogenetic studies try to infer a phy-

logenetic tree that reflects the actual lineages of species during evolution. Multiple alignments are just one (important) source of data for building phylogenetic trees. Numerous other sources of information are used to build, compare and reconcile trees. The following definition is adapted from [Brooks et al., 2007; Erdem, 2011; Miranda et al., 2014]

Definition 3. A phylogeny is a septuple $P = V, E, F, C, D_C, v, \mathscr{L}$, where (V, E) is a graph commonly describing a rooted binary tree, F is the set of terminal nodes of the graph (leaves of the tree), C is a set of qualitative attributes, the characters, with domains D_C, v is a function giving the value of each attribute for each terminal node in F, and $\mathscr{L}$ is an optional function that provides a real length for each edge. Variants exist with unrooted trees, non binary trees, or even phylogenetic networks that take into account the possibility of a reticulate evolution due to the exchange of genes between species (horizontal transfer).

The characters may be binary and, moreover, cladistic (the values are ordered during evolution from an ancestral state to derived states). When data are made of genetic sequences, characters are positions in a multiple alignment with at least two different letters (called SNPs). These mutation positions are generally binary characters. Nodes of the tree may be considered as states of the evolution process and edges as transformations, such as mutations. Considering only its structure (without function $\mathscr{L}$), a general problem is to build a tree on a set of species that is correct with respect to a given set of characters.

The phylogeny inference problem Given a set F of taxonomic units and a triple C, D_C, v providing character values for each element in F, decide if there exists a phylogeny that fulfills one of these criteria:

- k-compatibility: at least k characters must be compatible with the tree. A character is compatible with a phylogeny tree if the set of all vertices having the same value for this character forms a subtree.
- k-parsimony: the tree may be mapped to a rectilinear Steiner tree with a size at most k. In such a tree, edges have an integer length that is positive or zero. The paths between two elements of F have a length that equals the Manhattan distance between the vectors of their characters.
- d-goodness-of-fit: Assumes the analysis of characters to be summarized with a symmetric dissimilarity matrix D between pairs of elements of F. The tree must have a goodness-of-fit at least d with respect to D. Given a phylogenetic tree with a function $\mathscr{L}$ associating a value with each of its edges, it is possible to build a symmetric matrix P giving the path length between pairs of elements of F. The goodness-of-fit is the Euclidean distance between P and D (i.e. the Frobenius norm of the difference of the two matrices).

All these problems and variants have been proved NP-complete by Day and Sankoff. Apart from the decision problems, people often look for optimization ver-

sions, for instance looking for a tree with a maximum number of compatible characters. A topological criterion may also be used to build the tree from unrooted trees produced on subsets of taxonomic units, typically quartets of species.

The whole setting may be a bit more complex since a number of assumptions have to be added to obtain realistic trees, the main one being about the evolution of characters.

In a tree, starting from a state where all characters are considered absent, a character may be gained once and for all (perfect phylogeny), gained once and then lost at most once (persistent phylogeny), gained several times but never lost (Camin-Sokal criterion), etc. Constraint modeling frameworks can adjust with great flexibility to these many different criteria. Integer Linear Programming has been successful in providing different models, depending on the type of data to be processed [Sridhar et al., 2008] (perfect phylogenies, maximum parsimony), [El-Kebir et al., 2015] (perfect phylogenies from tumor multisample sequences), [Gusfield, 2015] (persistent phylogenies), [Bonizzoni et al., 2017] (incomplete perfect phylogenies on tumoral sequences). Answer set programming has been used in [Brooks et al., 2007] (perfect phylogenies), [Kavanagh et al., 2006] (Camin-Sokal criterion) and [Wu et al., 2007] (maximum quartet consistency).

In practice, even with these assumptions, a number of equivalent solutions may exist and it is fundamental to put the experts in the decision loop. This is why in recent years algorithmics is not the sole concern of evolutionary bioinformatics. There is a growing interest in a knowledge-based approach in this field since it brings together a large community and many results that may be partially contradictory have to be integrated. The ability to build complex queries and check combinatorial properties has shed light on logical approaches. In particular, if one considers a phylogenetic tree as a transition system (by adding loops on the leaves), it is possible to apply a temporal logic to make queries on the tree properties, the state properties or a mix of both. Requeno et al. [Requeno et al., 2013] have thus developed a model checking framework where it is possible to use CTL on phylogenetic trees. The possibility to obtain not only verifications but also counter-examples if formulas are not satisfied is important in a practical interactive context where the user makes an intensive use of queries to mine the trees (e.g. checking if there are back mutations in the tree can be used to detect these mutational events). This framework has been extended to the treatment of quantitative information through the use of stochastic logics [Requeno and Colom, 2016]. This allows to introduce in queries some probabilities and an explicit time. It is of significant interest if one wants to test models of evolution and to compute maximum likelihood estimations for trees in this context.

Other logical frameworks such as Answer Set Programming have been used. For instance, the supertree construction problem, which consists of building a tree that is maximally consistent with a set of trees built on overlapping sets of species, has been encoded as an ASP model in [Koponen et al., 2015]. A web service interface API has been developed in ASP for TreeBASE, a relational database designed to manage and explore information on phylogenetic relationships [Le et al., 2012] and a toolkit has

been developed for the alignment, consistency checking and inconsistency repair of taxonomies, using various reasoning systems (first order, Answer Set and dedicated provers) [Chen et al., 2013] [Franz et al., 2015]. Note that the alignment of trees may be used to build a phylogeny with a divide and conquer approach in order to improve search efficiency. For instance, [Ford et al., 2015] describes a method splitting the set of characters into subsets for which the search for a perfect phylogeny is possible and then use the 'anchor' trees built on these subsets to constrain the search of the whole tree.

7 Drug Discovery

High throughput screening (HTS) refers to a set of techniques aiming at identifying biologically active molecules that exhibit useful properties among elements of a large database of chemical compounds. The selection of these candidate molecules together with an accurate prediction of their molecular activity is an important economic issue. In particular, such databases are used by the chemical industry and have a major value in pharmacology for developing new drugs and reduce the need for animal testing. They can contain millions of components. The key problem is to establish *structure-activity relationships* (SAR), i.e., to predict a biological activity from molecular descriptors and some knowledge on physico-chemical properties of chemicals. If we try to predict a degree of activity, we will talk about QSAR (quantitative SAR) and when activity is replaced by other physicochemical properties, we will talk about SPR (structure-property relationships). This question is at the crossroads of chemoinformatics and bioinformatics. Solutions are based on the assumption that "similar" molecules generally share a similar activity but it is far from being so simple in reality, some minor structural changes being enough to completely change the activity of a molecule. This problem is generally referred as the *activity cliff* [Cruz-Monteagudo et al., 2014; Dimova and Bajorath, 2016].

A fundamental notion in biochemistry is that of receptor/ligand interaction. A ligand is a substance that forms a complex with a biomolecule. Ligand binding to a receptor protein changes the 3D conformation and thus the functional state of this protein. When the structure of the target protein is known, the most commonly used approach is molecular docking (see Section 4) and the approach is called structure-based drug design (SBDD). There are now a number of drugs whose development was heavily influenced by SBDD, such as HIV protease inhibitors [Kitchen et al., 2004]. A recent review of work in this area is available in [Ferreira et al., 2015]. When the structure is unknown, the approach is called ligand-based drug design (LBDD) and the research described in the rest of this section mostly falls within this approach. In fact, most recent methods try to associate similarities of both ligands and receptor protein by concatenating their descriptors for learning classifiers. This computational chemogenomic approach is particularly useful for a case-based/analogical reasoning approach, although it does not although it does not

appear to have been formalized in these terms [Brown et al., 2013]. either when a target is searched for a new ligand on the basis of similarity with other ligands having known targets, or It applies either when searching for a new ligand on the basis of similarity with other ligands having known targets, or vice versa when searching for a new target on the basis of similarity with proteins having known active ligands.

Biological activity is generally dose-dependent and can be described by many parameters. In pharmacology, it is represented by two types of attributes, the activity of the target and the toxicity of the drug, which is itself described along four dimensions reflecting the life cycle of the substance and the different aspects of its transformation in the organism such as bio-availability or biodegradability (ADME: "absorption, distribution, metabolism, and excretion"). If all data and attributes can be turned into numerical or ordered values, it is possible to build mathematical functions that can predict the activities of new chemicals. Historically, the structure-activity study was based on simple models where the degree of activity was assumed to be a linear function of the measured properties (e.g. hydrophobicity) on chemical compounds. Simple statistical methods such as linear regression were typically applied for such studies. Since then, machine learning techniques have played an increasingly important role in this field [Lavecchia, 2015]. In its simplest form, the drug discovery challenge may be formalized as the following machine learning issue:

Definition 4. Given a set of graphs labeled as positive or negative instances of molecular compound fragments with a given property, build a predictor for this property enabling the classification of new compounds.

The tested property can take various forms : activity (active/nonactive or a degree), drug-likeness, ADME property (absorption, solubility or permeability, metabolic stability, etc.), toxicity.

A number of open-source or commercial rule-based systems have been developed and are still used to solve this prediction problem in various domains such has skin sensitization, hepatotoxicity, or carcinogenic compounds [Raies and Bajic, 2016]. Knowledge-based expert systems are routinely used to predict potential chemical toxicity, on the basis of qualitative evidence. For instance, Toxtree [Benigni and Bossa, 2008] operates with a a manually designed set of rules for evaluating the mutagenic/carcinogenic potential of chemicals. At a finer level, predicting xenobiotic metabolism (the way an organism degrades a chemical compound that it does not naturally produce in several metabolites) seems more challenging since a lot of false-positive can be produced[Judson, 2014].

Besides expert systems, which assume the existence of a large knowledge base, there are of course automated prediction methods. The main methods in this field are Bayesian methods and SVM. The most recent ones use deep learning. Bayesian networks (see Chapter 8 of Volume 2) have been used in [Abdo et al., 2010, 2014]. Authors propose to train a Bayesian belief network and use it in [Abdo et al., 2014] to infer the activity class of a target compound. In the network, there is a termi-

nal node for the target compound and other terminal nodes for sets of compounds known tho share some activity. The root nodes of the network represent the presence of specific fragments (substructures) in compounds. The calculation of conditional probabilities is adapted to the graph structure of chemical data and the target molecule is assigned to the most similar class based on the presence of common fragments. SVM (see Chapter 11 of Volume 1) have been used with Gaussian as well as simple linear kernels [Hinselmann et al., 2011]. More specific kernels have been designed [Vert and Jacob, 2008], in particular graph-kernels that work on labeled graphs [Mahé and Vert, 2009]. Decision trees have also been used with some success, particularly Random Forests. For instance members of Pfizer showed that RF can produce results as good as SVM for predicting the relationship between the chemical structure of a compound and its metabolic stability [Sakiyama et al., 2008]. More recently, they have proved to be interesting for toxicology prediction (Tox21 challenge, [Banerjee et al., 2016]). RF have also been used for the protein-ligand docking problem in [Ballester and Mitchell, 2010]. molecules into the target's binding site (pose identification), and predicting how strongly the docked conformation binds to the target As for protein structure prediction (see section 4), deep learning has allowed to decrease the necessity to select optimal descriptors, although to the detriment of explainability of predictions. A good review of deep learning approaches in this field is available in [Gawehn et al., 2016]. Note that the search space is huge when crossing available chemical compounds and target proteins and the incompleteness of databases is a serious concern for all these methods [Mestres et al., 2008]. For this reason, a number of authors stress the importance of *active learning* to better select the relevant part of databases or relevant experiments in order to transfer the available knowledge to new cases of protein-ligand association [Wei et al., 2015; Naik et al., 2016; Reker et al., 2016, 2017]. In [Reker et al., 2017], authors use Random Forests for the learning component and propose a 'curiosity' criterion to select incrementally the relevant interactions in the database of known interactions. The curiosity measure selects each time the interaction for which there is the least consensus among the decision trees when classifying the interaction as active or not. This strategy, which is compatible with incremental selection of the most interesting experiments to refine the knowledge base, is also proving effective for machine learning using 10 to 20 percent of the database.

Except when drug discovery is based on deep learning techniques, the preprocessing phase is itself a hard problem, since it is necessary to extract from chemical databases the structural fragments that will be used as instances in the previous problem. This leads to a data mining problem:

Definition 5. Given a set of graphs representing molecular compounds, build a set of frequently occurring subgraphs.

For a review of the multiple descriptors that have been used to represent molecular data, see [Sawada et al., 2014]. The structural and physicochemical fragments at the origin of the biological behavior of chemicals are often called *structural alerts* in the literature. One of the early AI approach for this problem is the system CASE

(and later MultiCASE) [Rannug et al., 1991], which considers structural subunits containing less than 10 connected heavy atoms and learns if they are active by measuring their occurrence probability with respect to a binomial distribution. Since then, the graph data mining approach has been prevalent [Takigawa and Mamitsuka, 2013; Sherhod et al., 2014]. The search space of frequent subgraphs is explored either with a Breadth-First strategy (Apriori approach) or with a Depth-First strategy (Pattern-Growth approach). The AGM method [Inokuchi et al., 2000] uses the Apriori approach and works on canonical forms of graph adjacency matrices. It incrementally increases the size of matrices, merging at step $k+1$ frequent matrices of size k resulting from step k. An interesting extension of this track of research proposes to integrate Apriori with the Version Space framework in order to produce the most specific and general molecular fragments corresponding to toxic compounds [De Raedt and Kramer, 2001; Helma et al., 2002]. This is in the full continuity of the founding work on the Meta-Dendral system that we mentioned in the introduction to this chapter Methods using the Pattern-Growth approach are described in [Lepailleur et al., 2013]. From frequent atoms, they build increasingly larger frequent molecules by adding new bonds.The authors highlight the interest of searching for discriminating patterns (emerging patterns, jumping patterns) by considering frequency ratios (growth rate) in addition to frequencies, and especially most specific discriminating patterns (closed or representative pruned molecular patterns).

In [Shao et al., 2015], the issue is set as mining discriminant subgraphs from graph data with multiple labels and it is shown how produced subgraphs can be applied to drug adverse effect prediction problem.

Among recent works, the notion of Pareto dominance with respect to a set of user-preference measures has proved to be of high importance for the selection of useful patterns (called skypatterns). The work described in [Ugarte et al., 2017] proposes a static method whose efficiency is based on a condensed representation of patterns and a dynamic method whose effectiveness is based on improved pruning through iterative use of the patterns produced to refine dominance constraints. It is applied to the search of toxic chemical fragments, using as preference measures frequency, growth rate and chemical aromaticity. Authors have used the DynCSP framework of dynamical constraint solving [Verfaillie and Jussien, 2005] for posting new constraints from current sky patterns and the Gecode toolkit [Schulte and Stuckey, 2008] for the implementation.

8 Glycobiology

As for proteins, understanding the biological functions of carbohydrates (glycans) and relating them to their structure remains experimentally difficult and classification, machine learning or data mining methods are needed to propose general models or predictors of these functions. Unlike proteins, the information in databases [Pérez et al., 2015; Tiemeyer et al., 2017; Kanehisa, 2017] are rooted trees with ordered children (up to 5) and not simply sequences, a characteristic that introduces

interesting challenges. Moreover, the basic units of glycans (the nodes of the tree, monosaccharides) exist in numerous derived forms (e.g. more than 100 in bacteria). The root is a specific sugar that binds to its environment (cell or protein). The edges are made of several types of sugar bonds (say a dozen). The total number of glycans is estimated to be in the order of hundreds of thousands. This renders the representation of carbohydrates a difficult problem, more difficult than the analysis of trees that appear in RNA folding secondary structures.

Ontologies have started to be developed for this field. For instance, GlycoRDF [Ranzinger et al., 2015] proposes a standard OWL ontology that gives access for a number of glycomics databases to an RDF representation of various data ranging from publications relating to glycan structures to experimental datasets. Glycomics offers a nice setting to compare different technologies for graph databases or knowledge bases. For instance, RDF and Property Graph representation have been compared in respect of the glycan substructure search issue [Alocci et al., 2015], showing a clear advantage for RDF representations.

The physical recognition of glycan structures is currently treated by tandem mass spectrometry (MS/MS) and liquid chromatography. Two approaches are possible to infer a glycan structure from its MS/MS spectrum. In the simplest case we can use a large curated database of already known structures together with their spectra and develop a matching program for the annotation of a new spectrum. Sparql queries through RDF technology can be sufficient in this case. The other approach (de novo sequencing) tries to assign structures to peaks of the spectrum without any database. This requires machine learning methods to help peak assignment. The Glyfon program [Kumozaki et al., 2015] builds from a spectrum a graph of possible monosaccharide assignments and their links to other peaks using information on mass difference between two peaks and searches the space of all assignments compatible with a realistic glycan structure using an integer programming approach with Lagrangian relaxation. The parameters of the objective function are learned through a structured SVM, a task made tricky by the availability of a training set of structure-spectrum pairs, but not the corresponding residue-peak pairs.

One of the main demands in glycomics bioinformatics tasks is the data mining of glycan structure databases to classify glycans and discriminate the classes on the basis of the structural patterns that they contain [Mamitsuka, 2011]. Genetic programming has been adapted in [Miyahara and Kuboyama, 2014] to learn glycan motifs through the use of tag tree patterns. An interesting adaptation of SVM classifiers has been proposed in [Yamanishi et al., 2007], who introduce tree kernels for glycans. In practice, a tree kernel measures the similarity between two trees by counting the number of common subtrees, possibly with the same size and/or the same depth, and a powerful restriction is to consider only subtrees that are close with respect to the sibling relation (co-rooted trees).

Since motifs are as important as the identification of classes, the choice of features has to be compatible with a feature selection method in order to extract the

high-scoring subtrees. Authors have applied this work with success to the task of predicting the blood origin of glycans among leukemic and non-leukemic blood cell types and finding a glycan motif typical of leukemic cells. Instead of using a dedicated kernel and then extracting features, some authors have tried to directly produce the relevant attributes through pattern mining in glycan structures, the presence of each frequent subtree becoming a binary attribute. A method is proposed in [Takigawa et al., 2010], which claims better results on a mixed set of existing and randomly synthesized glycans than the previous method.

Frequent subtrees are extracted using two criteria, the search of subtrees that are significantly more frequent than the tree they come from and the search of significant subtrees with respect to a Fisher test using a control dataset.

A more ambitious approach for analyzing the structure of glycans is to use formal grammars. This way, one can not only discover motifs (associated to non-terminals of the grammar) but also the hierarchical relations they share (the rules of the grammar). T. Akutsu has introduced elementary ordered tree grammars for this purpose [Akutsu, 2010], where production rules use trees with edges labeled either using terminal or non-terminal symbols and one leaf in the tree may be tagged to indicate where another tree may be attached. Grammars are restricted to Chomsky's normal form (two non-terminals on the right-hand side). The idea is then to use grammar-based compression as a criterion to find interesting structures: the problem is to find the smallest grammar that generates exactly a given tree. An integer programming model is proposed in [Zhao et al., 2010], with a small experimentation on glycan trees labeled with the types of monosaccharides and the use of Cplex as a solver. This work is extended in [Zhao et al., 2015] to take into account multiple trees, this time using glycans labeled with the glycosyl transferases that enable the linkage of monosaccharides in the construct (a small experiment on RNA secondary structures is also provided). An interesting point in this extension is that a grammar could directly reflect the construction process of the molecule.

In this respect, glycobiology offers a specific application field for pathway reconstruction techniques (see 5). Indeed, the formation of each glycan structure results from dedicated biochemical pathways using polymerization reactions catalyzed by specific enzymes. One important question is thus to associate one or several genes corresponding to these enzymes to reactions that progressively transform a glycan structure. This can be studied in bacteria thanks to knockout experiments observing the effect of discarding a gene on the produced structures and the presence/absence of genes in related strains, as it is used in [Sternberg et al., 2013]. This paper demonstrates the application of inductive logic programming (Progol) to learn the gene-rule associations. Authors emphasize the interesting fact that aside from using some background knowledge on biochemistry (pathways, decomposition of glycan) and strain serotypes, it is necessary to introduce some speculative assumptions to better score the competing hypotheses. As in other areas of Artificial intelligence, it appears that the formalization of preferences has been key to successful predictions.

It seems that there is still scope for other research on this problem, using other techniques, and to our knowledge, no grammatical inference method has been ap-

plied so far to look at a grammar generalizing a positive training set of glycan structures and possibly rejecting a negative training set. The last study also points to the interest in the development of preference reasoning and possibly preference learning to help select from the predictions a reasonable subset of hypotheses that will be the subject of experimental testing.

There are also probabilistic approaches that have proposed extensions of Hidden Markov Models for the treatment of ordered trees. Among the most interesting ones, Ordered Tree Markov Model (OTMM) considers dependencies between parents and their first child and dependencies between ordered children [Ueda et al., 2005], and profilePTSMM (Probabilistic Sibling-dependent Tree Markov Model) considers two different types of transition dependencies between parents and all their children and dependencies between ordered children, together with the introduction of match/delete and insert states as in profile HMM [Aoki-Kinoshita et al., 2006; Aoki-Kinoshita, 2015].

Another common task in glycomics is the prediction of the glycosylation state of proteins. There are four types of glycosylation, the main one being O-linked and N-linked glycosylation, then C-linked glycosylation. It is known to occur on particular sites in the protein, partially characterized by short sequence motifs. The issue is to predict the glycosylation type and the sites.

The glycosylation site prediction problem

- Given a type of glycosylation, given a set P of proteins with known sequence and glycosylation sites and some optional background knowledge providing functional features or annotations for any protein,
- Build a classifier that can predict the glycosylation sites of this type in a new protein.

Of course, it is possible to state the problem as a three- or four-class problem instead of building one classifier for each type.

As in many bioinformatics applications, the difficulty is to find a trade-off between the number of parameters of the learned models and the relative scarcity of available data. Several predictors are often combined to this end.

For example, in [Senger and Karim, 2008] a set of recurrent Elman networks are trained to predict the major presence of a certain type of glycans in N-linked glycosylation of proteins, the training data being provided by predictors of the secondary structure and the accessibility state of these proteins from their sequence. In [Chauhan et al., 2013], using the same type of information, the three major types of glycosylation are predicted with an SVM-based approach, using a Gaussian RBF kernel and a carefully selected non-redundant dataset. Authors have chosen this

approach after testing numerous methods available in the Weka machine learning toolkit (namely Random Forest, Logistic Model Trees, various types of SVM in libsvm and with Sequential Minimal Optimization SMO, Bayesian network and naive Bayes). The philosophy of the best methods is to use a maximum number of attributes, including derived attributes that result from auxiliary predictors, and to add a feature selection stage to avoid overfitting. Among state-of-the-art methods at the time of this review, GlycoMine [Li et al., 2015] makes use of a knowledge base extracted from a number of databases of protein features (Gene ontology, Kegg, Pfam, Uniprot, etc.) and uses a feature selection procedure based both on mutual information and information gain.

It is likely that many problems studied on sequences will have an extension on glycan trees. It is thus a new field of study and application of AI techniques developed for sequences to these more complex structures. For instance, algorithms have been developed recently for glycan multiple alignments [Hosoda et al., 2017] and it would be interesting to check ideas developed on sequence multiple alignment (see section 6) in this new context.

9 Conclusion

Bioinformatics is a field full of incomplete data, knowledge expertise and NP–complete problems, and is as such a playground offering many opportunities for Artificial intelligence studies. This exciting interdisciplinary field comes at a cost. The first difficulty is to cope with the rapidly advancing technology. It is not always easy to distinguish short term problems that will be rapidly obsolete thanks to the next generation technology from more fundamental issues that are created by accessing a new kind of data. Two significant trends seem, however, to be emerging in this field.

First, there is an extensive use of weighted sequence data to cover all omics observations, giving both access to their qualitative and quantitative content. A weight may be a quality score that reflects the probability that a letter at a given position in the sequence was correctly observed by the sequencer (this tends to be standardized in file formats like FastQ, which codes letters as well as quality by ASCII characters) or an abundance (read count) that reflects the degree of expression of an element of the sequence in the observed sample. Sequencing is no longer reduced to the analysis of DNA and is now applied as a high-throughput technology for the identification of chromosomal 3D structures, the observation of epigenetic factors like DNA methylation or histone modification, the analysis of coding and non-coding types of RNA, translation efficiency of proteins [MGlincy and Ingolia, 2017], and all the interactions like the interactions between DNA and proteins [Soon et al., 2013]. This unification from the technological side is good news for all researchers in Bioinformatics since it provides some stability to their results.

Secondly, access to individual observations, which started with genomes of individual organisms and culminates now in the development of single cell analysis [Baron and Yanai, 2017; Bock et al., 2016; Gawad et al., 2016; Yuan et al., 2017], is a clear breakthrough in molecular biology. Understanding the pool of variations that lead to diseases, analyzing a microbial community and the exchanges that occur between its elements and tracking the embryonic development of a pool of cells, are all attainable with the technological developments. It opens the door to personalized medicine and to the rational representation and understanding of populations at all levels. Most likely, new types of intelligent models and methods are needed and will emerge to address these new challenges.

Once the raw data have been preprocessed, what makes studies successful is a deep understanding of the peculiarities of a particular biological question. The devil is in the detail. These peculiarities are key to addressing the complexity and push forward the boundaries of feasibility, but it is also a guarantee that relevant solutions for the biologist will be proposed. Of course, it is possible to work with standard benchmarks and build on the accomplishments of predecessors. But there are plenty of opportunities to participate in biological discoveries. The only advice I could address to AI people eager to start out in this field and contribute to these discoveries is to keep their focus on a specific set of questions and to ensure that there is a biologist expert on these questions in the loop.

Besides standard biology, an alternative and much more prospective route to cooperation between computer science and biology is synthetic biology, a field interested by life engineering, trying to design living components with simplified, fully controlled behaviors that can be assembled. It is not only a question of introducing a new technology: since experiments can be better controlled, it provides key to an in-depth understanding of living systems. This could be inspiring for AI in its goal of better understanding the components of intelligence. Neurons are no more the sole interesting cells in this respect. It has been shown by T. Nakagaki for instant that even primitive systems like ciliates or slime are able to memorize and have learning capacities. Some authors start to think of BI (Bio-artificial Intelligence) after AI [Nesbeth et al., 2016] by looking at ways to implement learners with synthetic gene and protein networks.

Acknowledgements I would like to thank authors of the French version of this chapter who offered me a primary material of quality to start this English version: F. Coste, C. Nédellec, T. Schiex and J.P. Vert. Thanks also to O. Dameron and F. Coste for their proofreading of the manuscript.

References

Abdallah, E. B., Folschette, M., Roux, O., and Magnin, M. (2017). ASP-based method for the enumeration of attractors in non-deterministic synchronous

and asynchronous multi-valued networks. *Algorithms for Molecular Biology*, 12(1):20.

Abdo, A., Chen, B., Mueller, C., Salim, N., and Willett, P. (2010). Ligand-based virtual screening using bayesian networks. *Journal of chemical information and modeling*, 50(6):1012–1020.

Abdo, A., Leclère, V., Jacques, P., Salim, N., and Pupin, M. (2014). Prediction of new bioactive molecules using a bayesian belief network. *Journal of Chemical Information and Modeling*, 54(1):30–36. PMID: 24392938.

Abou-Jaoudé, W., Traynard, P., Monteiro, P. T., Saez-Rodriguez, J., Helikar, T., Thieffry, D., and Chaouiya, C. (2016). Logical modeling and dynamical analysis of cellular networks. *Frontiers in genetics*, 7.

Abu-Srhan, A. and Al Daoud, E. (2013). A hybrid algorithm using a genetic algorithm and cuckoo search algorithm to solve the traveling salesman problem and its application to multiple sequence alignment. *International Journal of Advanced Science and Technology*, 61:29–38.

Acuna, V., Chierichetti, F., Lacroix, V., Marchetti-Spaccamela, A., Sagot, M.-F., and Stougie, L. (2009). Modes and cuts in metabolic networks: Complexity and algorithms. *Biosystems*, 95(1):51–60.

Adhikari, B., Bhattacharya, D., Cao, R., and Cheng, J. (2015). CONFOLD: Residue-residue contact-guided ab initio protein folding. *Proteins: Structure, Function, and Bioinformatics*, 83(8):1436–1449.

Akutsu, T. (2010). A bisection algorithm for grammar-based compression of ordered trees. *Information Processing Letters*, 110(18-19):815–820.

Akutsu, T., Hayashida, M., Ching, W.-K., and Ng, M. K. (2007). Control of boolean networks: Hardness results and algorithms for tree structured networks. *Journal of theoretical biology*, 244(4):670–679.

Akutsu, T., Kosub, S., Melkman, A. A., and Tamura, T. (2012). Finding a periodic attractor of a Boolean network. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, 9(5):1410–1421.

Albert, R. and Thakar, J. (2014). Boolean modeling: a logic-based dynamic approach for understanding signaling and regulatory networks and for making useful predictions. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 6(5):353–369.

Allouche, D., André, I., Barbe, S., Davies, J., de Givry, S., Katsirelos, G., O'Sullivan, B., Prestwich, S., Schiex, T., and Traoré, S. (2014). Computational protein design as an optimization problem. *Artificial Intelligence*, 212:59 – 79.

Alocci, D., Mariethoz, J., Horlacher, O., Bolleman, J. T., Campbell, M. P., and Lisacek, F. (2015). Property graph vs rdf triple store:A comparison on glycan substructure search. *PLOS ONE*, 10(12):1–17.

Aniba, M. R., Poch, O., Marchler-Bauer, A., and Thompson, J. D. (2010). Alexsys: a knowledge-based expert system for multiple sequence alignment construction and analysis. *Nucleic Acids Research*, 38(19):6338.

Antonov, I. and Borodovsky, M. (2010). Genetack: frameshift identification in protein-coding sequences by the viterbi algorithm. *Journal of bioinformatics and computational biology*, 8(03):535–551.

Aoki-Kinoshita, K. F. (2015). *Analyzing Glycan-Binding Patterns with the ProfileP-STMM Tool*, pages 193–202. Springer New York, New York, NY.

Aoki-Kinoshita, K. F., Ueda, N., Mamitsuka, H., and Kanehisa, M. (2006). ProfilePSTMM: capturing tree-structure motifs in carbohydrate sugar chains. *Bioinformatics*, 22(14):e25–e34.

Arellano, G., Argil, J., Azpeitia, E., Benítez, M., Carrillo, M., Góngora, P., Rosenblueth, D. A., and Alvarez-Buylla, E. R. (2011). "Antelope": a hybrid-logic model checker for branching-time Boolean GRN analysis. *BMC bioinformatics*, 12(1):490.

Aronson, S. J. and Rehm, H. L. (2015). Building the foundation for genomics in precision medicine. *Nature*, 526(7573):336–342.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., et al. (2000). Gene Ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29.

Awada, W., Khoshgoftaar, T. M., Dittman, D., Wald, R., and Napolitano, A. (2012). A review of the stability of feature selection techniques for bioinformatics data. In *Information Reuse and Integration (IRI), 2012 IEEE 13th International Conference on*, pages 356–363. IEEE.

Backofen, R. and Will, S. (2006). A constraint-based approach to fast and exact structure prediction in three-dimensional protein models. *Constraints*, 11(1):5–30.

Baldi, P. and Brunak, S. (2001). *Bioinformatics: the Machine Learning Approach*. MIT press.

Ballester, P. J. and Mitchell, J. B. O. (2010). A machine learning approach to predicting protein–ligand binding affinity with applications to molecular docking. *Bioinformatics*, 26(9):1169–1175.

Banerjee, P., Siramshetty, V. B., Drwal, M. N., and Preissner, R. (2016). Computational methods for prediction of in vitro effects of new chemical structures. *Journal of cheminformatics*, 8(1):51.

Barahona, P. and Krippahl, L. (2008). Constraint programming in structural bioinformatics. *Constraints*, 13(1):3–20.

Barberis, M., Todd, R. G., and van der Zee, L. (2017). Advances and challenges in logical modeling of cell cycle regulation: perspective for multi-scale, integrative yeast cell models. *FEMS yeast research*, 17(1).

Bard, J. B. and Rhee, S. Y. (2004). Ontologies in biology: design, applications and future challenges. *Nature Reviews Genetics*, 5(3):213.

Baron, M. and Yanai, I. (2017). New skin for the old rna-seq ceremony: the age of single-cell multi-omics. *Genome Biology*, 18(1):159.

Batt, G., Besson, B., Ciron, P.-E., de Jong, H., Dumas, E., Geiselmann, J., Monte, R., Monteiro, P. T., Page, M., Rechenmann, F., and Ropers, D. (2012). *Genetic Network Analyzer: A Tool for the Qualitative Modeling and Simulation of Bacterial Regulatory Networks*, pages 439–462. Springer New York, New York, NY.

Batt, G., De Jong, H., Page, M., and Geiselmann, J. (2008). Symbolic reachability analysis of genetic regulatory networks using discrete abstractions. *Automatica*, 44(4):982–989.

Baú, D., Martin, A. J., Mooney, C., Vullo, A., Walsh, I., and Pollastri, G. (2006). Distill: a suite of web servers for the prediction of one-, two- and three-dimensional structural features of proteins. *BMC Bioinformatics*, 7(1):402.

Bechhofer, S., Buchan, I., De Roure, D., Missier, P., Ainsworth, J., Bhagat, J., Couch, P., Cruickshank, D., Delderfield, M., Dunlop, I., et al. (2013). Why linked data is not enough for scientists. *Future Generation Computer Systems*, 29(2):599–611.

Bellazzi, R. (2014). Big data and biomedical informatics: a challenging opportunity. *Yearbook of medical informatics*, 9(1):8.

Benigni, R. and Bossa, C. (2008). Structure alerts for carcinogenicity, and the Salmonella assay system: A novel insight through the chemical relational databases technology. *Mutation Research/Reviews in Mutation Research*, 659(3):248 – 261.

Beretta, S., Bonizzoni, P., Vedova, G. D., Pirola, Y., and Rizzi, R. (2014). Modeling alternative splicing variants from rna-seq data with isoform graphs. *Journal of Computational Biology*, 21(1):16–40.

Berger, B. and Leighton, T. (1998). Protein folding in the hydrophobic-hydrophilic (HP) model is NP-complete. *Journal of Computational Biology*, 5(1):27–40.

Bhaskar, H., Hoyle, D. C., and Singh, S. (2006). Machine learning in bioinformatics: A brief survey and recommendations for practitioners. *Computers in Biology and Medicine*, 36(10):1104 – 1125. Intelligent Technologies in Medicine and BioinformaticsIntelligent Technologies in Medicine and Bioinformatics.

Bizer, C., Heath, T., and Berners-Lee, T. (2009). Linked data-the story so far. *International journal on semantic web and information systems*, 5(3):1–22.

Blake, J. A. and Bult, C. J. (2006). Beyond the data deluge: Data integration and bio-ontologies. *Journal of Biomedical Informatics*, 39(3):314 – 320. Biomedical Ontologies.

Blaszczyk, M., Gront, D., Kmiecik, S., Ziolkowska, K., Panek, M., and Kolinski, A. (2014). *Coarse-Grained Protein Models in Structure Prediction*, pages 25–53. Springer Berlin Heidelberg, Berlin, Heidelberg.

Blaszczyk, M., Jamroz, M., Kmiecik, S., and Kolinski, A. (2013). CABS-fold: server for the de novo and consensus-based prediction of protein structure. *Nucleic acids research*, 41(W1):W406–W411.

Bock, C., Farlik, M., and C. Sheffield, N. (2016). Multi-Omics of Single Cells: Strategies and Applications. *Trends in Biotechnology*, 34.

Bonizzoni, P., Ciccolella, S., Della Vedova, G., and Soto, M. (2017). Beyond Perfect Phylogeny: Multisample phylogeny reconstruction via ILP. In *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology,and Health Informatics*, ACM-BCB '17, pages 1–10, New York, NY, USA. ACM.

Bouziane, H., Messabih, B., and Chouarfia, A. (2015). Effect of simple ensemble methods on protein secondary structure prediction. *Soft Computing*, 19(6):1663–1678.

Boyer, F. and Viari, A. (2003). Ab initio reconstruction of metabolic pathways. *Bioinformatics*, 19(suppl_2):ii26–ii34.

Brahim, A. B. and Limam, M. (2017). Ensemble feature selection for high dimensional data: a new method and a comparative study. *Advances in Data Analysis and Classification*, pages 1–16.

Brim, L., Češka, M., and Šafránek, D. (2013). Model checking of biological systems. In *Formal Methods for Dynamical Systems*, pages 63–112. Springer.

Brooks, D. R., Erdem, E., Erdoğan, S. T., Minett, J. W., and Ringe, D. (2007). Inferring phylogenetic trees using answer set programming. *Journal of Automated Reasoning*, 39(4):471–511.

Brown, J. B., Niijima, S., and Okuno, Y. (2013). Compound Protein Interaction Prediction Within Chemogenomics: Theoretical Concepts, Practical Usage, and Future Directions. *Molecular Informatics*, 32(11-12):906–921.

Cannata, N., Schröder, M., Marangoni, R., and Romano, P. (2008). A semantic Web for bioinformatics: goals, tools, systems, applications. *BMC Bioinformatics*, 9(4):S1.

Caravagna, G., Graudenzi, A., Ramazzotti, D., Sanz-Pamplona, R., De Sano, L., Mauri, G., Moreno, V., Antoniotti, M., and Mishra, B. (2016). Algorithmic methods to infer the evolutionary trajectories in cancer progression. *Proceedings of the National Academy of Sciences*, 113(28):E4025–E4034.

Carrillo, M., Góngora, P. A., and Rosenblueth, D. A. (2012). An overview of existing modeling tools making use of model checking in the analysis of biochemical networks. *Frontiers in plant science*, 3.

Chauhan, J. S., Rao, A., and Raghava, G. P. S. (2013). In silico Platform for Prediction of N-, O- and C-Glycosites in Eukaryotic Protein Sequences. *PLOS ONE*, 8(6):1–10.

Chelliah, V., Juty, N., Ajmera, I., Ali, R., Dumousseau, M., Glont, M., Hucka, M., Jalowicki, G., Keating, S., Knight-Schrijver, V., Lloret-Villas, A., Natarajan, K. N., Pettit, J.-B., Rodriguez, N., Schubert, M., Wimalaratne, S. M., Zhao, Y., Hermjakob, H., Le Novère, N., and Laibe, C. (2015). BioModels: ten-year anniversary. *Nucleic Acids Research*, 43(D1):D542–D548.

Chen, L., Liu, H., and Friedman, C. (2005). Gene name ambiguity of eukaryotic nomenclatures. *Bioinformatics*, 21(2):248–256.

Chen, M., Yu, S., Franz, N., Bowers, S., and Ludäscher, B. (2013). Euler/X:A toolkit for logic-based taxonomy integration. Technical Report 1306, 22nd International Workshop on Functional and (Constraint) Logic Programming, Technische Berichte des Instituts fur Informatik der Christian-Albrechts-Universitat zu Kiel.

Chen, Q., Chen, Y.-P. P., and Zhang, C. (2007). Detecting inconsistency in biological molecular databases using ontologies. *Data Mining and Knowledge Discovery*, 15:275–296.

Cheng, J. and Baldi, P. (2007). Improved residue contact prediction using support vector machines and a large feature set. *BMC bioinformatics*, 8(1):113.

Clark, C. and Divvala, S. (2016). Pdffigures 2.0: Mining figures from research papers. In *Digital Libraries (JCDL), 2016 IEEE/ACM Joint Conference on*, pages 143–152. IEEE.

Coluzza, I. (2017). Computational protein design: a review. *Journal of Physics: Condensed Matter*, 29(14):143001.

Coste, F. (2016). *Learning the Language of Biological Sequences*, pages 215–247. Springer Berlin Heidelberg, Berlin, Heidelberg.

Cruz-Monteagudo, M., Medina-Franco, J. L., Perez-Castillo, Y., Nicolotti, O., Cordeiro, M. N. D., and Borges, F. (2014). Activity cliffs in drug discovery: Dr jekyll or mr hyde? *Drug Discovery Today*, 19(8):1069–1080.

Cuff, J. A. and Barton, G. J. (2000). Application of multiple sequence alignment profiles to improve protein secondary structure prediction. *Proteins: Structure, Function, and Bioinformatics*, 40(3):502–511.

D. Chapman, S., Adami, C., O. Wilke, C., and B KC, D. (2017). The evolution of logic circuits for the purpose of protein contact map prediction. *PeerJ*, 5:e3139.

Dallas, D. C., Guerrero, A., Parker, E. A., Robinson, R. C., Gan, J., German, J. B., Barile, D., and Lebrilla, C. B. (2015). Current peptidomics: Applications, purification, identification, quantification, and functional analysis. *PROTEOMICS*, 15(5-6):1026–1038.

De Brevern, A., Etchebest, C., and Hazout, S. (2000). Bayesian probabilistic approach for predicting backbone structures in terms of protein blocks. *Proteins: Structure, Function, and Bioinformatics*, 41(3):271–287.

De Raedt, L. and Kramer, S. (2001). The Levelwise Version Space Algorithm and Its Application to Molecular Fragment Finding. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'01, pages 853–859, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Deagustini, C. A. D., Martinez, M. V., Falappa, M. A., and Simari, G. R. (2016). Datalog+-ontology consolidation. *Journal of Artificial Intelligence Research*, 56:613–656.

Diaz-Uriarte, R. (2007). GeneSrF and varSelRF: a web-based tool and R package for gene selection and classification using random forest. *BMC bioinformatics*, 8(1):328.

Dimova, D. and Bajorath, J. (2016). Advances in activity cliff research. *Molecular Informatics*, 35(5):181–191.

Doğan, B. and Ölmez, T. (2015). A novel state space representation for the solution of 2D-HP protein folding problem using reinforcement learning methods. *Applied Soft Computing*, 26:213–223.

Drozdetskiy, A., Cole, C., Procter, J., and Barton, G. J. (2015). Jpred4: a protein secondary structure prediction server. *Nucleic acids research*, 43(W1):W389–W394.

Dubey, S. P., Kini, N. G., Balaji, S., and Kumar, M. S. (2017). Protein structure prediction on 2d square hp lattice with revised fitness function. *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1732–1736.

Dubrova, E. and Teslenko, M. (2011). A SAT-based algorithm for finding attractors in synchronous Boolean networks. *IEEE/ACM transactions on computational biology and bioinformatics*, 8(5):1393–1399.

Dunn, S., Wahl, L., and Gloor, G. (2008). Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 24(3):333–340.

Eddy, S. R. (1998). Profile hidden markov models. *Bioinformatics (Oxford, England)*, 14(9):755–763.

El-Kebir, M., Oesper, L., Acheson-Field, H., and Raphael, B. J. (2015). Reconstruction of clonal trees and tumor composition from multi-sample sequencing data. *Bioinformatics*, 31(12):i62–i70.

Erdem, E. (2011). *Applications of Answer Set Programming in Phylogenetic Systematics*, pages 415–431. Springer Berlin Heidelberg, Berlin, Heidelberg.

Faraggi, E. and Kloczkowski, A. (2017). *Accurate Prediction of One-Dimensional Protein Structure Features Using SPINE-X*, pages 45–53. Springer New York, New York, NY.

Faraggi, E., Zhang, T., Yang, Y., Kurgan, L., and Zhou, Y. (2012). SPINE X: Improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. *Journal of Computational Chemistry*, 33(3):259–267.

Feig, M., Rotkiewicz, P., Kolinski, A., Skolnick, J., and Brooks, C. L. (2000). Accurate reconstruction of all-atom protein representations from side-chain-based low-resolution models. *Proteins: Structure, Function, and Bioinformatics*, 41(1):86–97.

Feigenbaum, E. A. and Buchanan, B. G. (1993). Dendral and meta-dendral: roots of knowledge systems and expert system applications. *Artificial Intelligence*, 59(1):233 – 240.

Ferreira, L. G., dos Santos, R. N., Oliva, G., and Andricopulo, A. D. (2015). Molecular docking and structure-based drug design strategies. *Molecules*, 20(7):13384–13421.

Ford, E., St. John, K., and Wheeler, W. C. (2015). Towards improving searches for optimal phylogenies. *Systematic Biology*, 64(1):56–65.

Frank, M. and Schloissnig, S. (2010). Bioinformatics and molecular modeling in glycobiology. *Cellular and molecular life sciences*, 67(16):2749–2772.

Franz, N. M., Chen, M., Yu, S., Kianmajd, P., Bowers, S., and Ludäscher, B. (2015). Reasoning over taxonomic change: ExploringAlignments for the perelleschus use case. *PLOS ONE*, 10(2):1–34.

Galiez, C., Magnan, C. N., Coste, F., and Baldi, P. (2016). VIRALpro: a tool to identify viral capsid and tail sequences. *Bioinformatics*, 32(9):1405–1407.

Galperin, M. Y., Fern'andez-Su'arez, X. M., and Rigden, D. J. (2017). The 24th annual nucleic acids research database issue: a look back and upcoming changes. *Nucleic Acids Research*, 45(D1):D1.

Gan, X., Kapsokalivas, L., Albrecht, A. A., and Steinhöfel, K. (2008). A symmetry-free subspace for ab initio protein folding simulations. In *Bioinformatics Research and Development*, pages 128–139. Springer.

Gao, Y., Wang, S., Deng, M., and Xu, J. (2017). Real-value and confidence prediction of protein backbone dihedral angles through a hybrid method of clustering and deep learning. *bioRxiv*.

Gawad, C., Koh, W., and Quake, S. R. (2016). Single-cell genome sequencing: current state of the science. *Nature reviews. Genetics*, 17(3):175.

Gawehn, E., Hiss, J. A., and Schneider, G. (2016). Deep learning in drug discovery. *Molecular informatics*, 35(1):3–14.

Gertheiss, J. and Tutz, G. (2009). Supervised feature selection in mass spectrometry-based proteomic profiling by blockwise boosting. *Bioinformatics*, 25(8):1076–1077.

Ghisalberti, G., Masseroli, M., and Tettamanti, L. (2010). Quality controls in integrative approaches to detect errors and inconsistencies in biological databases. *Journal of integrative bioinformatics*, 7(3):52–64.

Ghoorah, A. W., Devignes, M.-D., Smaïl-Tabbone, M., and Ritchie, D. W. (2013). Protein docking using case-based reasoning. *Proteins: Structure, Function, and Bioinformatics*, 81(12):2150–2158.

Gonnet, G. H., Korostensky, C., and Benner, S. (2000). Evaluation measures of multiple sequence alignments. *Journal of Computational Biology*, 7(1-2):261–276.

Gordon, J. J., Towsey, M. W., Hogan, J. M., Mathews, S. A., and Timms, P. (2005). Improved prediction of bacterial transcription start sites. *Bioinformatics*, 22(2):142–148.

Greene, D., Richardson, S., and Turro, E. (2017). ontologyx: a suite of r packages for working with ontological data. *Bioinformatics*, 33(7):1104–1106.

Grivell, L. (2002). Mining the bibliome: searching for a needle in a haystack? *EMBO reports*, 3(3):200–203.

Gront, D., Kmiecik, S., Koliński, A., Meinke, J. H., Zimmermann, M. T., Mohanty, S., and Hansmann, U. H. E. (2006). High throughput method for protein structure prediction. In *NIC Workshop 2006: From Computational Biophysics to System Biology*, volume 34, pages 79–82, Julich. John von Neumann Inst. for Computing.

Guan, Y., Myers, C. L., Hess, D. C., Barutcuoglu, Z., Caudy, A. A., and Troyanskaya, O. G. (2008). Predicting gene function in a hierarchical context with an ensemble of classifiers. *Genome biology*, 9(S1):S3.

Guermeur, Y., Geourjon, C., Gallinari, P., and Deléage, G. (1999). Improved performance in protein secondary structure prediction by inhomogeneous score combination. *Bioinformatics*, 15(5):413–421.

Guo, Y., Tao, F., Wu, Z., and Wang, Y. (2017). Hybrid method to solve HP model on 3D lattice and to probe protein stability upon amino acid mutations. *BMC Systems Biology*, 11(4):93.

Gupta, S. K., Kececioglu, J. D., and Schäffer, A. A. (1995). Improving the practical space and time efficiency of the shortest-paths approach to sum-of-pairs multiple sequence alignment. *Journal of Computational Biology*, 2(3):459–472.

Gusfield, D. (2015). Persistent Phylogeny: A Galled-tree and Integer Linear Programming Approach. In *Proceedings of the 6th ACM Conference on Bioinformatics, Computational Biology and Health Informatics*, BCB '15, pages 443–451, New York, NY, USA. ACM.

Götz, S., Garc'ia-G'omez, J. M., Terol, J., Williams, T. D., Nagaraj, S. H., Nueda, M. J., Robles, M., Tal'on, M., Dopazo, J., and Conesa, A. (2008). High-

throughput functional annotation and data mining with the blast2go suite. *Nucleic Acids Research*, 36(10):3420–3435.

Han, Y., Gao, S., Muegge, K., Zhang, W., and Zhou, B. (2015). Advanced applications of rna sequencing and challenges. *Bioinformatics and Biology Insights*, 9s1:BBI.S28991.

Hassanien, A. E., Al-Shammari, E. T., and Ghali, N. I. (2013). Computational intelligence techniques in bioinformatics. *Computational biology and chemistry*, 47:37–47.

Hassanien, A.-E., Milanova, M. G., Smolinski, T. G., and Abraham, A. (2008). *Computational Intelligence in Solving Bioinformatics Problems: Reviews, Perspectives, and Challenges*, pages 3–47. Springer Berlin Heidelberg, Berlin, Heidelberg.

Hastings, J., de Matos, P., Dekker, A., Ennis, M., Harsha, B., Kale, N., Muthukrishnan, V., Owen, G., Turner, S., Williams, M., and Steinbeck, C. (2013). The chebi reference database and ontology for biologically relevant chemistry: enhancements for 2013. *Nucleic Acids Research*, 41(D1):D456–D463.

Hatem, M. and Ruml, W. (2013). External Memory Best-First Search for Multiple SequenceAlignment. In *27th AAAI conference on Artificial Intelligence*.

Hayes-Roth, B., Buchanan, B. G., Lichtarge, O., Hewitt, M., Altman, R. B., Brinkley, J. F., Cornelius, C., Duncan, B. S., and Jardetzky, O. (1986). PROTEAN: Deriving protein structure from constraints. In *AAAI*, pages 904–909.

Heffernan, R., Paliwal, K., Lyons, J., Dehzangi, A., Sharma, A., Wang, J., Sattar, A., Yang, Y., and Zhou, Y. (2015). Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning. *Scientific reports*, 5:11476.

Helma, C., Kramer, S., and Luc, D. R. (2002). The Molecular Feature Miner MOLFEA. In *In Proceedings of the Beilstein Workshop 2002: Molecular Informatics: Confronting Complexity; Beilstein Institut*, pages 79–93.

Hinselmann, G., Rosenbaum, L., Jahn, A., Fechner, N., Ostermann, C., and Zell, A. (2011). Large-scale learning of structure- activity relationships using a linear support vector machine and problem-specific metrics. *Journal of chemical information and modeling*, 51(2):203–213.

Hirschman, L., Park, J. C., Tsujii, J., Wong, L., and Wu, C. H. (2002). Accomplishments and challenges in literature data mining for biology. *Bioinformatics*, 18(12):1553–1561.

Hoehndorf, R., Slater, L., Schofield, P. N., and Gkoutos, G. V. (2015). Aber-owl: a framework for ontology-based data access in biology. *BMC Bioinformatics*, 16(1):26.

Hoff, K. and Stanke, M. (2015). Current methods for automated annotation of protein-coding genes. *Current Opinion in Insect Science*, 7(Supplement C):8 – 14. Insect genomics * Development and regulation.

Hoff, K. J., Lange, S., Lomsadze, A., Borodovsky, M., and Stanke, M. (2015). Braker1: unsupervised rna-seq-based genome annotation with genemark-et and augustus. *Bioinformatics*, 32(5):767–769.

Holzinger, A., Dehmer, M., and Jurisica, I. (2014). Knowledge discovery and interactive data mining in bioinformatics-state-of-the-art, future challenges and research directions. *BMC bioinformatics*, 15(6):I1.

Hosoda, M., Akune, Y., and Aoki-Kinoshita, K. F. (2017). Development and application of an algorithm to compute weighted multiple glycan alignments. *Bioinformatics*, 33(9):1317–1323.

Huang, D. W., Sherman, B. T., and Lempicki, R. A. (2008). Systematic and integrative analysis of large gene lists using david bioinformatics resources. *Nature protocols*, 4(1):44.

Hundley, L., Lederberg, J., and Levinthal, E. (1963). Multivator- a biochemical laboratory for martian experiments. Technical Report NSG-81-60, NASA.

Huntley, R. P., Harris, M. A., Alam-Faruque, Y., Blake, J. A., Carbon, S., Dietze, H., Dimmer, E. C., Foulger, R. E., Hill, D. P., Khodiyar, V. K., Lock, A., Lomax, J., Lovering, R. C., Mutowo-Meullenet, P., Sawford, T., Van Auken, K., Wood, V., and Mungall, C. J. (2014). A method for increasing expressivity of gene ontology annotations using a compositional approach. *BMC Bioinformatics*, 15(1):155.

Ideker, T. and Nussinov, R. (2017). Network approaches and applications in biology. *PLOS Computational Biology*, 13(10):1–3.

Ikeda, T. and Imai, H. (1999). Enhanced A* algorithms for multiple alignments: optimal alignments for several sequences and k-opt approximate alignments for large cases. *Theoretical Computer Science*, 210(2):341 – 374.

Inokuchi, A., Washio, T., and Motoda, H. (2000). An Apriori-Based Algorithm for Mining Frequent Substructures from Graph Data. In Zighed, D. A., Komorowski, J., and Żytkow, J., editors, *Principles of Data Mining and Knowledge Discovery*, pages 13–23, Berlin, Heidelberg. Springer Berlin Heidelberg.

Inoue, K. (2011). Logic programming for Boolean networks. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, IJCAI'11, pages 924–930. AAAI Press.

Inza, I., Calvo, B., Armañanzas, R., Bengoetxea, E., Larrañaga, P., and Lozano, J. A. (2010). *Machine Learning: An Indispensable Tool in Bioinformatics*, pages 25–48. Humana Press, Totowa, NJ.

Islamaj Doğan, R., Kim, S., Chatr-aryamontri, A., Chang, C. S., Oughtred, R., Rust, J., Wilbur, W. J., Comeau, D. C., Dolinski, K., and Tyers, M. (2017). The biocbiogrid corpus: full text articles annotated for curation of protein–protein and genetic interactions. *Database*, 2017(1):baw147.

Jiang, S.-Y. and Ramachandran, S. (2010). Assigning biological functions to rice genes by genome annotation, expression analysis and mutagenesis. *Biotechnology Letters*, 32(12):1753–1763.

Jones, D. T., Singh, T., Kosciolek, T., and Tetchner, S. (2015). Metapsicov: combining coevolution methods for accurate prediction of contacts and long range hydrogen bonding in proteins. *Bioinformatics*, 31(7):999–1006.

Jong, K., Marchiori, E., Sebag, M., and Van Der Vaart, A. (2004). Feature selection in proteomic pattern data with support vector machines. In *Computational Intelligence in Bioinformatics and Computational Biology, 2004. CIBCB'04. Proceedings of the 2004 IEEE Symposium on*, pages 41–48. IEEE.

Judson, P. N. (2014). *Knowledge-Based Approaches for Predicting the Sites and Products of Metabolism*, chapter 12, pages 293–318. Wiley-Blackwell.

Jupp, S., Malone, J., Bolleman, J., Brandizi, M., Davies, M., Garcia, L., Gaulton, A., Gehant, S., Laibe, C., Redaschi, N., Wimalaratne, S. M., Martin, M., Le Novère, N., Parkinson, H., Birney, E., and Jenkinson, A. M. (2014). The ebi rdf platform: linked open data for the life sciences. *Bioinformatics*, 30(9):1338–1339.

Källberg, M., Wang, H., Wang, S., Peng, J., Wang, Z., Lu, H., and Xu, J. (2012). Template-based protein structure modeling using the raptorx web server. *Nature protocols*, 7(8):1511–1522.

Kaminski, R., Schaub, T., Siegel, A., and Videla, S. (2013). Minimal intervention strategies in logical signaling networks with ASP. *Theory and Practice of Logic Programming*, 13(4-5):675–690.

Kanehisa, M. (2017). Kegg glycan. In *A Practical Guide to Using Glycomics Databases*, pages 177–193. Springer.

Karpe, P. D., Latendresse, M., and Caspi, R. (2011). The pathway tools pathway prediction algorithm. *Standards in genomic sciences*, 5(3):424.

Kavanagh, J., Mitchell, D., Ternovska, E., Maňuch, J., Zhao, X., and Gupta, A. (2006). *Constructing Camin-Sokal Phylogenies Via Answer Set Programming*, pages 452–466. Springer Berlin Heidelberg, Berlin, Heidelberg.

Keedwell, E. and Narayanan, A. (2005). *Intelligent bioinformatics: The application of artificial intelligence techniques to bioinformatics problems*. John Wiley & Sons.

Khatri, P., Sirota, M., and Butte, A. J. (2012). Ten years of pathway analysis: Current approaches and outstanding challenges. *PLOS Computational Biology*, 8(2):1–10.

Kislyuk, A., Lomsadze, A., Lapidus, A. L., and Borodovsky, M. (2009). Frameshift detection in prokaryotic genomic sequences. *International journal of bioinformatics research and applications*, 5(4):458–477.

Kitchen, D. B., Decornez, H., Furr, J. R., and Bajorath, J. (2004). Docking and scoring in virtual screening for drug discovery: methods and applications. *Nature reviews Drug discovery*, 3(11):935.

Klamt, S. (2006). Generalized concept of minimal cut sets in biochemical networks. *Biosystems*, 83(2-3):233–247.

Klarner, H., Bockmayr, A., and Siebert, H. (2015). Computing maximal and minimal trap spaces of Boolean networks. *Natural Computing*, 14(4):535–544.

Klarner, H., Streck, A., and Siebert, H. (2017). Pyboolnet: a python package for the generation, analysis and visualization of boolean networks. *Bioinformatics*, 33(5):770–772.

Koliński, A. et al. (2004). Protein modeling and structure prediction with a reduced representation. *Acta Biochimica Polonica*, 51.

Koponen, L., Oikarinen, E., Janhunen, T., and Säilä, L. (2015). Optimizing phylogenetic supertrees using answer set programming. *Theory and Practice of Logic Programming*, 15(4-5):604–619.

Korf, R. E., Zhang, W., Thayer, I., and Hohwald, H. (2005). Frontier search. *J. ACM*, 52(5):715–748.

Korostensky, C. and Gonnet, G. (1999). Near optimal multiple sequence alignments using a traveling salesman problem approach. In *in SPIRE/CRIWG*, pages 105–114.

Koshino, M., Murata, H., Shirayama, M., and Kimura, H. (2006). Applying the various optimal solution search methods to multiple sequence alignments and performance evaluation. *Systems and Computers in Japan*, 37(11):1–10.

Kowalski, R. (1979). Algorithm= logic+ control. *Communications of the ACM*, 22(7):424–436.

Kumozaki, S., Sato, K., and Sakakibara, Y. (2015). A machine learning based approach to de novo sequencing of glycans from tandem mass spectrometry spectrum. *IEEE/ACM transactions on computational biology and bioinformatics*, 12(6):1267–1274.

Lacroix, V., Sammeth, M., Guigo, R., and Bergeron, A. (2008). Exact transcriptome reconstruction from short sequence reads. In *International Workshop on Algorithms in Bioinformatics*, pages 50–63. Springer.

Lai, J., An, J., Seim, I., Walpole, C., Hoffman, A., Moya, L., Srinivasan, S., Perry-Keene, J. L., Wang, C., Lehman, M. L., et al. (2015). Fusion transcript loci share many genomic features with non-fusion loci. *BMC genomics*, 16(1):1021.

Lavecchia, A. (2015). Machine-learning approaches in drug discovery: methods and applications. *Drug discovery today*, 20(3):318–331.

Le, T., Nguyen, H., Pontelli, E., and Son, T. C. (2012). ASP at Work: An ASP Implementation of PhyloWS. In Dovier, A. and Costa, V. S., editors, *Technical Communications of the 28th International Conference on Logic Programming (ICLP'12)*, volume 17 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 359–369, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Le Novere, N. (2015). Quantitative and logic modelling of gene and molecular networks. *Nature Reviews. Genetics*, 16(3):146.

Lepailleur, A., Poezevara, G., and Bureau, R. (2013). Automated detection of structural alerts (chemical fragments) in (eco)toxicology. *Computational and Structural Biotechnology Journal*, 5(6):e201302013.

Lertampaiporn, S., Thammarongtham, C., Nukoolkit, C., Kaewkamnerdpong, B., and Ruengjitchatchawalya, M. (2014). Identification of non-coding RNAs with a new composite feature in the Hybrid Random Forest Ensemble algorithm. *Nucleic Acids Research*, 42(11):e93.

Li, F., Li, C., Wang, M., Webb, G. I., Zhang, Y., Whisstock, J. C., and Song, J. (2015). Glycomine: a machine learning-based approach for predicting n-, c-and o-linked glycosylation in the human proteome. *Bioinformatics*, 31(9):1411–1419.

Li, H., Hou, J., Adhikari, B., Lyu, Q., and Cheng, J. (2017). Deep learning methods for protein torsion angle prediction. *BMC bioinformatics*, 18(1):417.

Li, L., Zhang, Y., Zou, L., Li, C., Yu, B., Zheng, X., and Zhou, Y. (2012). An ensemble classifier for eukaryotic protein subcellular location prediction using gene ontology categories and amino acid hydrophobicity. *PLOS ONE*, 7(1):1–12.

Lihu, A. and Holban, Ş. (2015). A review of ensemble methods for de novo motif discovery in ChIP-Seq data. *Briefings in bioinformatics*, 16(6):964–973.

Lim, C. Y., Wang, H., Woodhouse, S., Piterman, N., Wernisch, L., Fisher, J., and Göttgens, B. (2016). Btr: training asynchronous Boolean models using single-cell expression data. *BMC Bioinformatics*, 17(1):355.

Lindsay, R. K., Buchanan, B. G., Feigenbaum, E. A., and Lederberg, J. (1993). Dendral: A case study of the first expert system for scientific hypothesis formation. *Artificial Intelligence*, 61(2):209 – 261.

Liu, H., Liu, L., and Zhang, H. (2010). Ensemble gene selection for cancer classification. *Pattern Recognition*, 43(8):2763 – 2772.

Lomsadze, A., Burns, P. D., and Borodovsky, M. (2014). Integration of mapped rna-seq reads into automatic training of eukaryotic gene finding algorithm. *Nucleic Acids Research*, 42(15):e119.

Lomsadze, A., Ter-Hovhannisyan, V., Chernoff, Y. O., and Borodovsky, M. (2005). Gene identification in novel eukaryotic genomes by self-training algorithm. *Nucleic Acids Research*, 33(20):6494–6506.

Magnan, C. N. and Baldi, P. (2014). SSpro/ACCpro 5: almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning and structural similarity. *Bioinformatics*, 30(18):2592–2597.

Mahadevan, R., von Kamp, A., and Klamt, S. (2015). Genome-scale strain designs based on regulatory minimal cut sets. *Bioinformatics*, 31(17):2844–2851.

Mahé, P. and Vert, J.-P. (2009). Graph kernels based on tree patterns for molecules. *Machine Learning*, 75(1):3–35.

Malikic, S., McPherson, A. W., Donmez, N., and Sahinalp, C. S. (2015). Clonality inference in multiple tumor samples using phylogeny. *Bioinformatics*, 31(9):1349–1356.

Mamitsuka, H. (2011). Glycoinformatics: Data Mining-basedApproaches. *CHIMIA International Journal for Chemistry*, 65(1):10–13.

Mann, M. and Backofen, R. (2014). Exact methods for lattice protein models. *Bio-Algorithms and Med-Systems*, 10(4):213–225.

Mann, M., Will, S., and Backofen, R. (2008). CPSP-tools–exact and complete algorithms for high-throughput 3d lattice protein studies. *BMC bioinformatics*, 9(1):230.

Martin, M., Dague, P., Pérès, S., and Simon, L. (2016). Minimality of Metabolic Flux Modes under Boolean Regulation Constraints. In *12th International Workshop on Constraint-Based Methods for Bioinformatics WCB'16*, Toulouse, France.

Matentzoglu, N., Vigo, M., Jay, C., and Stevens, R. (2017). Inference inspector: Improving the verification of ontology authoring actions. *Web Semantics: Science, Services and Agents on the World Wide Web*.

Maupetit, J., Gautier, R., and Tufféry, P. (2006). SABBAC: online structural alphabet-based protein backbone reconstruction from alpha-carbon trace. *Nucleic Acids Research*, 34(suppl_2):W147–W151.

McGuffin, L. J., Bryson, K., and Jones, D. T. (2000). The psipred protein structure prediction server. *Bioinformatics*, 16(4):404–405.

Medina, I., T'arraga, J., Mart'inez, H., Barrachina, S., Castillo, M. I., Paschall, J., Salavert-Torres, J., Blanquer-Espert, I., Hern'andez-Garc'ia, V., Quintana-Ort'i,

E. S., and Dopazo, J. (2016). Highly sensitive and ultrafast read mapping for rna-seq analysis. *DNA Research*, 23(2):93–100.

Mei, S. (2012). Multi-label Multi-Kernel Transfer Learning for Human Protein Subcellular Localization. *PLOS ONE*, 7(6):1–12.

Mei, S. and Zhu, H. (2014). AdaBoost based multi-instance transfer learning for predicting proteome-wide interactions between salmonella and human proteins. *PLOS ONE*, 9(10):1–12.

Merelli, E., Armano, G., Cannata, N., Corradini, F., d'Inverno, M., Doms, A., Lord, P., Martin, A., Milanesi, L., Möller, S., Schroeder, M., and Luck, M. (2007). Agents in bioinformatics, computational and systems biology. *Briefings in Bioinformatics*, 8(1):45.

Mestres, J., Gregori-Puigjane, E., Valverde, S., and Sole, R. V. (2008). Data completeness—the Achilles heel of drug-target networks. *Nature biotechnology*, 26(9):983.

MGlincy, N. J. and Ingolia, N. T. (2017). Transcriptome-wide measurement of translation by ribosome profiling. *Methods*, 126:112 – 129. Post-transcriptional Regulation of Gene Expression.

Michel, A. M., Choudhury, K. R., Firth, A. E., Ingolia, N. T., Atkins, J. F., and Baranov, P. V. (2012). Observation of dually decoded regions of the human genome using ribosome profiling data. *Genome Research*, 22(11):2219–2229.

Midic, U., Dunker, A. K., and Obradovic, Z. (2005). Improving protein secondary-structure prediction by predicting ends of secondary-structure segments. In *Computational Intelligence in Bioinformatics and Computational Biology, 2005. CIBCB'05. Proceedings of the 2005 IEEE Symposium on*, pages 1–8. IEEE.

Miranda, M., Lynce, I., and Manquinho, V. (2014). Inferring phylogenetic trees using pseudo-Boolean optimization. *AI Commun.*, 27(3):229–243.

Mitra, S., Datta, S., Perkins, T., and Michailidis, G. (2008). *Introduction to machine learning and bioinformatics*. CRC Press.

Miyahara, T. and Kuboyama, T. (2014). Learning of glycan motifs using genetic programming and various fitness functions. *Journal of Advanced Computational Intelligence and Intelligent Informatics*, 18(3):401–408.

Moll, M., Schwarz, D., and Kavraki, L. E. (2008). Roadmap methods for protein folding. *Protein Structure Prediction*, pages 219–239.

Monteiro, P. T., Ropers, D., Mateescu, R., Freitas, A. T., and de Jong, H. (2008). Temporal logic patterns for querying dynamic models of cellular interaction networks. *Bioinformatics*, 24(16):i227–i233.

Moult, J., Fidelis, K., Kryshtafovych, A., Schwede, T., and Tramontano, A. (2018). Critical assessment of methods of protein structure prediction (CASP)—round xii. *Proteins: Structure, Function, and Bioinformatics*, 86:7–15.

Muggleton, S., King, R. D., and Stenberg, M. J. (1992). Protein secondary structure prediction using logic-based machine learning. *Protein Engineering, Design and Selection*, 5(7):647–657.

Muller, J., Creevey, C. J., Thompson, J. D., Arendt, D., and Bork, P. (2010). Aqua: automated quality improvement for multiple sequence alignments. *Bioinformatics*, 26(2):263–265.

Mungall, C. J., Torniai, C., Gkoutos, G. V., Lewis, S. E., and Haendel, M. A. (2012). Uberon, an integrative multi-species anatomy ontology. *Genome Biology*, 13(1):R5.

Mushthofa, M., Torres, G., Van de Peer, Y., Marchal, K., and De Cock, M. (2014). ASP-G: an ASP-based method for finding attractors in genetic regulatory networks. *Bioinformatics*, 30(21):3086–3092.

Nagi, S., Bhattacharyya, D. K., and Kalita, J. K. (2017). Complex detection from ppi data using ensemble method. *Network Modeling Analysis in Health Informatics and Bioinformatics*, 6(1):3.

Naik, A. W., Kangas, J. D., Sullivan, D. P., and Murphy, R. F. (2016). Active machine learning-driven experimentation to determine compound effects on protein patterns. *eLife*, 5:e10047.

Naldi, A., Carneiro, J., Chaouiya, C., and Thieffry, D. (2010). Diversity and plasticity of th cell types predicted from regulatory network modelling. *PLOS Computational Biology*, 6(9):1–16.

Nesbeth, D. N., Zaikin, A., Saka, Y., Romano, M. C., Giuraniuc, C. V., Kanakov, O., and Laptyeva, T. (2016). Synthetic biology routes to bio-artificial intelligence. *Essays In Biochemistry*, 60(4):381–391.

Newman, A. and Ruhl, M. (2004). Combinatorial problems on strings with applications to protein folding. In Farach-Colton, M., editor, *LATIN 2004: Theoretical Informatics*, pages 369–378, Berlin, Heidelberg. Springer Berlin Heidelberg.

Ng, C.-T., Li, C., and Fan, X. (2017). A Fast Algorithm for Reconstructing Multiple Sequence Alignment and Phylogeny Simultaneously. *Current Bioinformatics*, 12:329–348.

Offmann, B., Tyagi, M., and de Brevern, A. G. (2007). Local protein structures. *Current Bioinformatics*, 2(3):165–202.

Okun, O. and Priisalu, H. (2007). Random forest for gene expression based cancer classification: overlooked issues. *Pattern Recognition and Image Analysis*, pages 483–490.

Omar, M., Salam, R., Abdullah, R., and Rashid, N. (2005). Multiple sequence alignment using optimization algorithms. *International Journal of Computational Intelligence*, 1(2):81–89.

Pandini, A., Fornili, A., and Kleinjung, J. (2010). Structural alphabets derived from attractors in conformational space. *Bmc Bioinformatics*, 11(1):97.

Park, Y. R., Kim, J., Lee, H. W., Yoon, Y. J., and Kim, J. H. (2011). Gochase-ii: correcting semantic inconsistencies from gene ontology-based annotations for gene products. *BMC Bioinformatics*, 12(1):S40.

Pashaei, E. and Aydin, N. (2017). Frequency difference based dna encoding methods in human splice site recognition. In *Computer Science and Engineering (UBMK), 2017 International Conference on*, pages 586–591. IEEE.

Pashaei, E., Ozen, M., and Aydin, N. (2016a). Splice sites prediction of human genome using adaboost. In *Biomedical and Health Informatics (BHI), 2016 IEEE-EMBS International Conference on*, pages 300–303. IEEE.

Pashaei, E., Ozen, M., and Aydin, N. (2017). Splice site identification in human genome using random forest. *Health and Technology*, 7(1):141–152.

Pashaei, E., Yilmaz, A., and Aydin, N. (2016b). A combined svm and markov model approach for splice site identification. In *Computer and Knowledge Engineering (ICCKE), 2016 6th International Conference on*, pages 200–204. IEEE.

Peng, J. and Xu, J. (2010). Low-homology protein threading. *Bioinformatics*, 26(12):i294–i300.

Pérez, S., Sarkar, A., Rivet, A., Breton, C., and Imberty, A. (2015). Glyco3d: a portal for structural glycosciences. In *Glycoinformatics*, pages 241–258. Springer.

Pertea, M., Pertea, G. M., Antonescu, C. M., Chang, T.-C., Mendell, J. T., and Salzberg, S. L. (2015). Stringtie enables improved reconstruction of a transcriptome from rna-seq reads. *Nature biotechnology*, 33(3):290–295.

Pes, B., Dessì, N., and Angioni, M. (2017). Exploiting the ensemble paradigm for stable feature selection: A case study on high-dimensional genomic data. *Information Fusion*, 35(Supplement C):132 – 147.

Petegrosso, R., Park, S., Hwang, T. H., and Kuang, R. (2017). Transfer learning across ontologies for phenome–genome association prediction. *Bioinformatics*, 33(4):529–536.

Piao, Y., Piao, M., Park, K., and Ryu, K. H. (2012). An ensemble correlation-based gene selection algorithm for cancer classification with gene expression data. *Bioinformatics*, 28(24):3306–3315.

Pietal, M. J., Bujnicki, J. M., and Kozlowski, L. P. (2015). GDFuzz3D: a method for protein 3d structure reconstruction from contact maps, based on a non-euclidean distance function. *Bioinformatics*, 31(21):3499–3505.

Pirola, Y., Rizzi, R., Picardi, E., Pesole, G., Della Vedova, G., and Bonizzoni, P. (2012). Pintron: a fast method for detecting the gene structure due to alternative splicing via maximal pairings of a pattern and a text. *BMC Bioinformatics*, 13(5):S2.

Pokarowski, P., Kolinski, A., and Skolnick, J. (2003). A minimal physically realistic protein-like lattice model: designing an energy landscape that ensures all-or-none folding to a unique native state. *Biophysical journal*, 84(3):1518–1526.

Popic, V., Salari, R., Hajirasouliha, I., Kashef-Haghighi, D., West, R. B., and Batzoglou, S. (2015). Fast and scalable inference of multi-sample cancer lineages. *Genome biology*, 16(1):91.

Post, L. J. G., Roos, M., Marshall, M. S., van Driel, R., and Breit, T. M. (2007). A semantic web approach applied to integrative bioinformatics experimentation: a biological use case with genomics data. *Bioinformatics*, 23(22):3080–3087.

Raies, A. B. and Bajic, V. B. (2016). In silico toxicology: computational methods for the prediction of chemical toxicity. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 6(2):147–172.

Ramsden, J. J. (2004). *Bioinformatics: An Introduction*, volume 21 of *Computational Biology*. Springer.

Rannug, U., Sjögren, M., Rannug, A., Gillner, M., Toftgård, R., Gustafsson, J.-Å., Rosenkranz, H., and Klopman, G. (1991). Use of artificial intelligence in structure—affinity correlations of 2,3,7,8-tetrachlorodibenzo-p-dioxin (TCDD) receptor ligands. *Carcinogenesis*, 12(11):2007–2015.

Ranzinger, R., Aoki-Kinoshita, K. F., Campbell, M. P., Kawano, S., Lütteke, T., Okuda, S., Shinmachi, D., Shikanai, T., Sawaki, H., Toukach, P., Matsubara, M., Yamada, I., and Narimatsu, H. (2015). GlycoRDF: an ontology to standardize glycomics data in RDF. *Bioinformatics*, 31(6):919–925.

Reid, I., O'Toole, N., Zabaneh, O., Nourzadeh, R., Dahdouli, M., Abdellateef, M., Gordon, P. M., Soh, J., Butler, G., Sensen, C. W., and Tsang, A. (2014). Snowyowl: accurate prediction of fungal genes by using rna-seq and homology information to select among ab initio models. *BMC Bioinformatics*, 15(1):229.

Reker, D., Schneider, P., and Schneider, G. (2016). Multi-objective active machine learning rapidly improves structure-activity models and reveals new protein-protein interaction inhibitors. *Chem. Sci.*, 7:3919–3927.

Reker, D., Schneider, P., Schneider, G., and Brown, J. (2017). Active learning for computational chemogenomics. *Future Medicinal Chemistry*, 9(4):381–402.

Requeno, J. I. and Colom, J. M. (2016). Evaluation of properties over phylogenetic trees using stochastic logics. *BMC bioinformatics*, 17(1):235.

Requeno, J. I., de Miguel Casado, G., Blanco, R., and Colom, J. M. (2013). Temporal logics for phylogenetic analysis via model checking. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 10(4):1058–1070.

Rinaldi, F., Lithgow, O., Gama-Castro, S., Solano, H., López-Fuentes, A., Muñiz Rascado, L. J., Ishida-Gutiérrez, C., Méndez-Cruz, C.-F., and Collado-Vides, J. (2017). Strategies towards digital and semi-automated curation in RegulonDB. *Database*, 2017:bax012.

Robertson, M. P. and Joyce, G. F. (2012). The origins of the rna world. *Cold Spring Harbor perspectives in biology*, 4(5):a003608.

Rodríguez-Penagos, C., Salgado, H., Martínez-Flores, I., and Collado-Vides, J. (2007). Automatic reconstruction of a bacterial regulatory network using natural language processing. *BMC Bioinformatics*, 8(1):293.

Röhl, A. and Bockmayr, A. (2017). A mixed-integer linear programming approach to the reduction of genome-scale metabolic networks. *BMC Bioinformatics*, 18(1):2.

Rost, B. and Sander, C. (1993). Prediction of protein secondary structure at better than 70% accuracy. *Journal of molecular biology*, 232(2):584–599.

Sakiyama, Y., Yuki, H., Moriya, T., Hattori, K., Suzuki, M., Shimada, K., and Honma, T. (2008). Predicting human liver microsomal stability with machine learning techniques. *Journal of Molecular Graphics and Modelling*, 26(6):907 – 915.

Samaga, R., Kamp, A. V., and Klamt, S. (2010). Computing combinatorial intervention strategies and failure modes in signaling networks. *Journal of Computational Biology*, 17(1):39–53.

Sarkar, I. N. (2015). Mining the bibliome. In *Translational Informatics*, pages 75–96. Springer.

Sawada, R., Kotera, M., and Yamanishi, Y. (2014). Benchmarking a wide range of chemical descriptors for drug-target interaction prediction using a chemogenomic approach. *Molecular Informatics*, 33(11-12):719–731.

Schietgat, L., Vens, C., Struyf, J., Blockeel, H., Kocev, D., and Džeroski, S. (2010). Predicting gene function using hierarchical multi-label decision tree ensembles. *BMC bioinformatics*, 11(1):2.

Schroedl, S. (2005). An improved search algorithm for optimal multiple-sequence alignment. *Journal of Artificial Intelligence Research*, 23:587–623.

Schulte, C. and Stuckey, P. J. (2008). Efficient Constraint Propagation Engines. *Transactions on Programming Languages and Systems*, 31(1):2:1–2:43.

Schwartz, R. and Schäffer, A. A. (2017). The evolution of tumour phylogenetics: principles and practice. *Nature Reviews Genetics*, 18(4):213.

Senger, R. S. and Karim, M. N. (2008). Prediction of n-linked glycan branching patterns using artificial neural networks. *Mathematical biosciences*, 211(1):89–104.

Shao, Z., Hirayama, Y., Yamanishi, Y., and Saigo, H. (2015). Mining discriminative patterns from graph data with multiple labels and its application to quantitative structure–activity relationship (qsar) models. *Journal of Chemical Information and Modeling*, 55(12):2519–2527. PMID: 26549421.

Shatabda, S., Newton, M. A. H., and Sattar, A. (2014). Constraint-based evolutionary local search for protein structures with secondary motifs. In Pham, D.-N. and Park, S.-B., editors, *PRICAI 2014: Trends in Artificial Intelligence*, pages 333–344. Springer International Publishing.

Shaw, D. L., Islam, A. S., Rahman, M. S., and Hasan, M. (2014). Protein folding in HP model on hexagonal lattices with diagonals. *BMC bioinformatics*, 15(2):S7.

Sheela, T. and Rangarajan, L. (2017). Combination of feature selection methods for the effective classification of microarray gene expression data. In Santosh, K., Hangarge, M., Bevilacqua, V., and Negi, A., editors, *Recent Trends in Image Processing and Pattern Recognition: First International Conference, RTIP2R 2016, Bidar, India, December 16–17, 2016, Revised Selected Papers*, pages 137–145. Springer Singapore.

Sherhod, R., Judson, P. N., Hanser, T., Vessey, J. D., Webb, S. J., and Gillet, V. J. (2014). Emerging Pattern Mining To Aid Toxicological Knowledge Discovery. *Journal of Chemical Information and Modeling*, 54(7):1864–1879.

Shvaiko, P. and Euzenat, J. (2013). Ontology matching: state of the art and future challenges. *IEEE Transactions on knowledge and data engineering*, 25(1):158–176.

Singh, G. B. (2015). Introduction to bioinformatics. In *Fundamentals of Bioinformatics and Computational Biology*, pages 3–10. Springer.

Singh, H., Singh, S., and Raghava, G. P. S. (2014). Evaluation of protein dihedral angle prediction methods. *PLOS ONE*, 9(8):1–9.

Skwark, M. J., Raimondi, D., Michel, M., and Elofsson, A. (2014). Improved contact predictions using the recognition of protein like contact patterns. *PLOS Computational Biology*, 10(11):1–14.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., et al. (2007). The obo foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, 25(11):1251–1255.

Smitha, S. K. N. and Reddy, S. N. (2016). Amyloid motif prediction using ensemble approach. *Current Bioinformatics*, 11(3):357–365.

Song, J., Burrage, K., Yuan, Z., and Huber, T. (2006). Prediction of cis/trans isomerization in proteins using psi-blast profiles and secondary structure information. *BMC Bioinformatics*, 7(1):124.

Soon, W. W., Hariharan, M., and Snyder, M. P. (2013). High-throughput sequencing for biology and medicine. *Molecular systems biology*, 9(1):640.

Spencer, M., Eickholt, J., and Cheng, J. (2015). A deep learning network approach to ab initio protein secondary structure prediction. *IEEE/ACM transactions on computational biology and bioinformatics*, 12(1):103–112.

Sridhar, S., Lam, F., Blelloch, G. E., Ravi, R., and Schwartz, R. (2008). Mixed integer linear programming for maximum-parsimony phylogeny inference. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 5(3):323–331.

Sternberg, M. J., Tamaddoni-Nezhad, A., Lesk, V. I., Kay, E., Hitchen, P. G., Cootes, A., van Alphen, L. B., Lamoureux, M. P., Jarrell, H. C., Rawlings, C. J., Soo, E. C., Szymanski, C. M., Dell, A., Wren, B. W., and Muggleton, S. H. (2013). Gene function hypotheses for the campylobacter jejuni glycome generated by a logic-basedApproach. *Journal of Molecular Biology*, 425(1):186 – 197.

Stevens, R., Goble, C. A., and Bechhofer, S. (2000). Ontology-based knowledge representation for bioinformatics. *Briefings in Bioinformatics*, 1(4):398–414.

Sundfeld, D., Razzolini, C., Teodoro, G., Boukerche, A., and de Melo, A. C. M. A. (2017). Pa-star: A disk-assisted parallel a-star strategy with locality-sensitive hash for multiple sequence alignment. *Journal of Parallel and Distributed Computing*.

Takigawa, I., Hashimoto, K., Shiga, M., Kanehisa, M., and Mamitsuka, H. (2010). Mining patterns from glycan structures. In *Proceedings of the International Beilstein Symposium on Glyco-Bioinformatics*, pages 13–24.

Takigawa, I. and Mamitsuka, H. (2013). Graph mining: procedure, application to drug discovery and recent advances. *Drug Discovery Today*, 18(1):50 – 57.

The GO Consortium (2017). Expansion of the gene ontology knowledgebase and resources. *Nucleic Acids Research*, 45(D1):D331–D338.

Tiemeyer, M., Aoki, K., Paulson, J., Cummings, R. D., York, W. S., Karlsson, N. G., Lisacek, F., Packer, N. H., Campbell, M. P., Aoki, N. P., et al. (2017). GlyTouCan: an accessible glycan structure repository. *Glycobiology*, 27(10):915–919.

Traoré, S., Roberts, K. E., Allouche, D., Donald, B. R., André, I., Schiex, T., and Barbe, S. (2016). Fast search algorithms for computational protein design. *Journal of Computational Chemistry*, 37(12):1048–1058.

Traynard, P., Fauré, A., Fages, F., and Thieffry, D. (2016). Logical model specification aided by model-checking techniques: application to the mammalian cell cycle regulation. *Bioinformatics*, 32(17):i772–i780.

Ueda, N., Aoki-Kinoshita, K. F., Yamaguchi, A., Akutsu, T., and Mamitsuka, H. (2005). A probabilistic model for mining labeled ordered trees: capturing patterns in carbohydrate sugar chains. *IEEE Transactions on Knowledge and Data Engineering*, 17(8):1051–1064.

Ugarte, W., Boizumault, P., Crémilleux, B., Lepailleur, A., Loudni, S., Plantevit, M., Raïssi, C., and Soulet, A. (2017). Skypattern mining: From pattern condensed representations to dynamic constraint satisfaction problems. *Artificial Intelligence*, 244:48 – 69. Combining Constraint Solving with Mining and Learning.

Vendruscolo, M., Kussell, E., and Domany, E. (1997). Recovery of protein structure from contact maps. *Folding and Design*, 2(5):295 – 306.

Verfaillie, G. and Jussien, N. (2005). Constraint solving in uncertain and dynamic environments: A survey. *Constraints*, 10(3):253–281.

Vert, J.-P. and Jacob, L. (2008). Machine learning for in silico virtual screening and chemical genomics: new strategies. *Combinatorial chemistry & high throughput screening*, 11(8):677–685.

Videla, S., Guziolowski, C., Eduati, F., Thiele, S., Gebser, M., Nicolas, J., Saez-Rodriguez, J., Schaub, T., and Siegel, A. (2015). Learning Boolean logic models of signaling networks with ASP. *Theoretical Computer Science*, 599(Supplement C):79 – 101. Advances in Computational Methods in Systems Biology.

Videla, S., Saez-Rodriguez, J., Guziolowski, C., and Siegel, A. (2017). caspo: a toolbox for automated reasoning on the response of logical signaling networks families. *Bioinformatics*, 33(6):947–950.

von Kamp, A. and Klamt, S. (2014). Enumeration of smallest intervention strategies in genome-scale metabolic networks. *PLoS computational biology*, 10(1):e1003378.

Wald, R., Khoshgoftaar, T. M., Dittman, D., Awada, W., and Napolitano, A. (2012). An extensive comparison of feature ranking aggregation techniques in bioinformatics. In *Information Reuse and Integration (IRI), 2012 IEEE 13th International Conference on*, pages 377–384. IEEE.

Walker, S. I. and Davies, P. C. (2013). The algorithmic origins of life. *Journal of the Royal Society Interface*, 10(79):20120869.

Wan, S., Mak, M.-W., and Kung, S.-Y. (2014). HybridGO-Loc: Mining hybrid features on gene ontology for predicting subcellular localization of multi-location proteins. *PLOS ONE*, 9(3):1–12.

Wang, S., Li, W., Liu, S., and Xu, J. (2016a). RaptorX-Property: a web server for protein structure property prediction. *Nucleic Acids Research*, 44(W1):W430–W435.

Wang, S., Li, W., Zhang, R., Liu, S., and Xu, J. (2016b). CoinFold: a web server for protein contact prediction and contact-assisted protein folding. *Nucleic Acids Research*, 44(W1):W361–W366.

Wang, S., Peng, J., Ma, J., and Xu, J. (2016c). Protein secondary structure prediction using deep convolutional neural fields. *Scientific reports*, 6(18962).

Wang, S., Sun, S., Li, Z., Zhang, R., and Xu, J. (2017a). Accurate de novo prediction of protein contact map by ultra-deep learning model. *PLOS Computational Biology*, 13(1):1–34.

Wang, X., Xuan, Z., Zhao, X., Li, Y., and Zhang, M. Q. (2009). High-resolution human core-promoter prediction with CoreBoost_HM. *Genome research*, 19(2):266–275.

Wang, Y., Mao, H., and Yi, Z. (2017b). Protein secondary structure prediction by using deep learning method. *Knowledge-Based Systems*, 118:115–123.

Wang, Z. and Xu, J. (2013). Predicting protein contact map using evolutionary and physical constraints by integer programming. *Bioinformatics*, 29(13):i266–i273.

Wei, D., Zhang, H., Wei, Y., and Jiang, Q. (2013). A novel splice site prediction method using support vector machine. *Journal of Computational Information Systems*, 9(20):8053–8060.

Wei, K., Iyer, R., and Bilmes, J. (2015). Submodularity in Data Subset Selection and Active Learning. In Bach, F. and Blei, D., editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1954–1963, Lille, France. PMLR.

Whetzel, P. L., Noy, N. F., Shah, N. H., Alexander, P. R., Nyulas, C., Tudorache, T., and Musen, M. A. (2011). Bioportal: enhanced functionality via new web services from the national center for biomedical ontology to access and use ontologies in software applications. *Nucleic Acids Research*, 39(suppl_2):W541–W545.

Wu, B., Abbott, T., Fishman, D., McMurray, W., Mor, G., Stone, K., Ward, D., Williams, K., and Zhao, H. (2003). Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data. *Bioinformatics*, 19(13):1636–1643.

Wu, G., You, J.-H., and Lin, G. (2007). Quartet-based phylogeny reconstruction with answer set programming. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4(1).

Wuyun, Q., Zheng, W., Peng, Z., and Yang, J. (2016). A large-scale comparative assessment of methods for residue–residue contact prediction. *Briefings in Bioinformatics*, page bbw106.

Xia, J.-F., Wu, M., You, Z.-H., Zhao, X.-M., and Li, X.-L. (2010). Prediction of β-hairpins in proteins using physicochemical properties and structure information. *Protein and Peptide Letters*, 17(9):1123–1128.

Xiao, Y. and Dougherty, E. R. (2007). The impact of function perturbations in Boolean networks. *Bioinformatics*, 23(10):1265–1273.

Xue, L. C., Rodrigues, J. a. P., Dobbs, D., Honavar, V., and Bonvin, A. M. (2017). Template-based protein–protein docking exploiting pairwise interfacial residue restraints. *Briefings in Bioinformatics*, 18(3):458–466.

Yamanishi, Y., Bach, F., and Vert, J.-P. (2007). Glycan classification with tree kernels. *Bioinformatics*, 23(10):1211–1216.

Yanev, N., Traykov, M., Milanov, P., and Yurukov, B. (2017). Protein Folding Prediction in a Cubic Lattice in Hydrophobic-Polar Model. *Journal of Computational Biology*, 24(5):412–421.

Yang, P., Ho, J. W., Yang, Y. H., and Zhou, B. B. (2011). Gene-gene interaction filtering with ensemble of filters. *BMC bioinformatics*, 12(1):S10.

Yang, P., Humphrey, S. J., James, D. E., Yang, Y. H., and Jothi, R. (2016a). Positive-unlabeled ensemble learning for kinase substrate prediction from dynamic phosphoproteomics data. *Bioinformatics*, 32(2):252–259.

Yang, P., Hwa Yang, Y., B Zhou, B., and Y Zomaya, A. (2010). A review of ensemble methods in bioinformatics. *Current Bioinformatics*, 5(4):296–308.

Yang, Y., Gao, J., Wang, J., Heffernan, R., Hanson, J., Paliwal, K., and Zhou, Y. (2016b). Sixty-five years of the long march in protein secondary structure prediction: the final stretch? *Briefings in Bioinformatics*, page bbw129.

Yang, Y., Gao, J., Wang, J., Heffernan, R., Hanson, J., Paliwal, K., and Zhou, Y. (2016c). Sixty-five years of the long march in protein secondary structure prediction: the final stretch? *Briefings in Bioinformatics*, page bbw129.

Yoshizumi, T., Miura, T., and Ishida, T. (2000). A* with partial expansion for large branching factor problems. In *AAAI/IAAI*, pages 923–929.

Yuan, G.-C., Cai, L., Elowitz, M., Enver, T., Fan, G., Guo, G., Irizarry, R., Kharchenko, P., Kim, J., Orkin, S., Quackenbush, J., Saadatpour, A., Schroeder, T., Shivdasani, R., and Tirosh, I. (2017). Challenges and emerging directions in single-cell analysis. *Genome Biology*, 18(1):84.

Zhang, T., Faraggi, E., Li, Z., and Zhou, Y. (2013). Intrinsically semi-disordered state and its role in induced folding and protein aggregation. *Cell Biochemistry and Biophysics*, 67(3):1193–1205.

Zhang, Y. and Rajapakse, J. C. (2009). *Machine learning in bioinformatics*, volume 4. John Wiley & Sons.

Zhao, Y., Hayashida, M., and Akutsu, T. (2010). Integer programming-based method for grammar-based tree compression and its application to pattern extraction of glycan tree structures. *BMC bioinformatics*, 11(11):S4.

Zhao, Y., Hayashida, M., Cao, Y., Hwang, J., and Akutsu, T. (2015). Grammar-based compression approach to extraction of common rules among multiple trees of glycans and rnas. *BMC bioinformatics*, 16(1):128.

Zhou, R. and Hansen, E. A. (2004). Space-efficient memory-based heuristics. In *AAAI*, pages 677–682.