ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Role of Machine Learning in Bioinformatics: A Survey

Ankur Singh Bist [*1], Babeesh Kumar [2]

[*1] Computer Engineering, Govind Ballabh Pant, University of Agriculture and Technology, Pantnagar, India

[2] Computer Engineering, Indain School of Mines, Dhanbad, India

ankur1990bist@gmail.com

Abstract

The size and complexity of biological data is increasing day by day. It is big challenge to deal with this growing data. In computer science there are lot of methods to deal with this problem but the best one is required for better analysis. Machine learning techniques are used in bioinformatics to deal with this problem.

Keywords: Neural Network, Machine Learning, Bioinformatics, Recognition.

Introduction

Bioinformatics is an interdisciplinary technical field that designs methods for storing, retrieving, organizing and analyzing biological data. A major task in bioinformatics is to design software tools to produce useful knowledge from biological data. Bioinformatics is a different science from biological computation, the latter being a computer science and computer engineering subfield using bioengineering and biology to create biological computers, whereas bioinformatics utilizes computers to better understand biology. Bioinformatics is similar to computational biology and has similar aims to it but differs on scale: whereas bioinformatics works with basic biological data, i.e. it works on the small scale giving attention to details, computational biology is a subfield of computer science which designs large theoretical models of biological systems desiring to enhance our understanding of them from an conceptual point of view, presently as mathematical biology does with mathematical models.

Bioinformatics utilizes different fields of computer science, mathematics and engineering to process biological data. Complex equipment are used to interpret in biological data at a much quicker pace than before and used in decoding the code of life. Databases and information systems are used to store and put in order biological data. Analyzing biological data may involve algorithms in artificial intelligence, soft computing, data mining, image processing, and simulation. The algorithms in turn depend on theoretical fundamentals such as discrete mathematics, control theory, system theory, information theory, and statistics. Commonly used software tools and technologies in the field like C, C++, Python, R, SQL, CUDA, MATLAB, and spreadsheet applications are used to obtain the objectives. Machine learning techniques are widely used for the analysis of biological data. Some of the widely used techniques used in bioinformatics like neural network, decision tree, hidden markov model etc. are discussed.

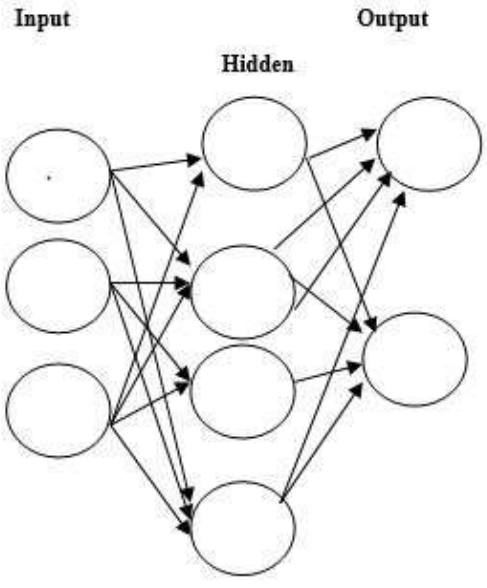


Figure1. Neural Network

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

Neural networks are similar to biological neural networks in that functions are performed collectively and in parallel by the units it is very important point because the biological neurons perform the function in parallel artificial neural network are not work accurately like biological neuron, rather than there being a clear description of task in smaller groups to which various units are assigned. The term "neural network" usually refers to models employed in statistics, cognitive psychology and artificial intelligence. Neural network models which emulate the central nervous system are part of theoretical neuroscience and computational neuroscience. Neural network models in artificial intelligence are usually referred to as artificial neural networks (ANNs).

A common use of the phrase ANN model really means the definition of a *class* of such functions where members of the class are obtained by varying parameters.

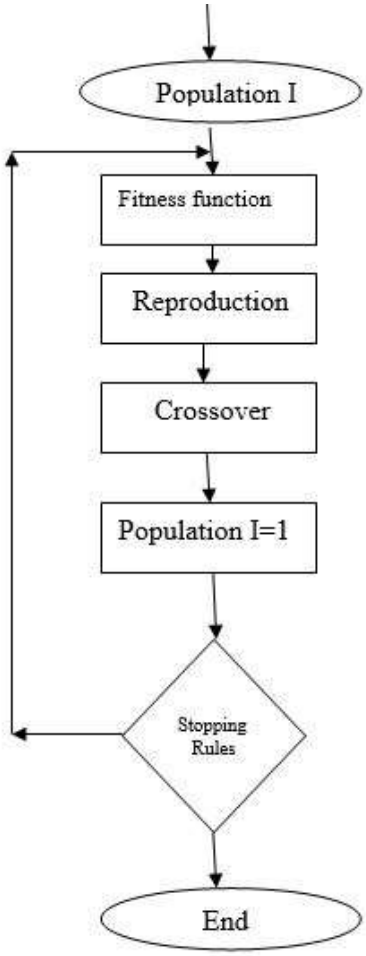


Figure2. Genetic algorithm -Optimization process

Bayesian networks

The most general task that is solved using Bayesian networks is probabilistic inference. For example, consider the water sprinkler network, and suppose it is observed that the fact that the grass is wet. There are two possible causes for this: either it is raining, or the sprinkler is on. Which is more likely? Bayes' rule can be used to compute the posterior probability of each explanation (where 0==false and 1==true) [1].

Learning Decision Trees

A decision tree is a simple representation for classifying examples. Decision tree learning is one of the most thriving techniques for supervised classification learning. A decision tree or a classification tree is a tree in which each internal (non-leaf) node is labeled with an input feature. The arcs coming from a node labeled with a feature are labeled with each of the possible values of the feature. Each leaf of the tree is labeled with a class or a probability distribution over the classes.

To classify an example, filter it down the tree, as follows. For each feature encountered in the tree, the arc corresponding to the value of the example for that feature is followed. When a leaf is reached, the classification corresponding to that leaf is returned.

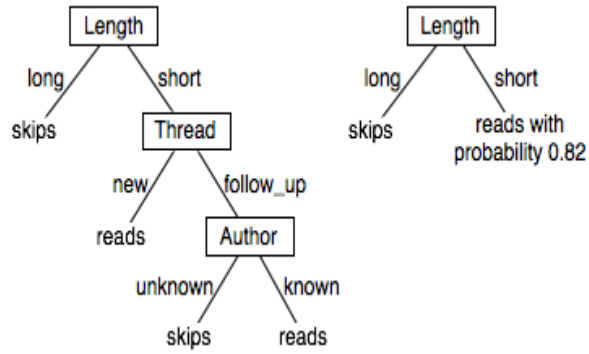


Figure 3: Two decision trees

Figure shows two possible decision trees. Each decision tree can be used to classify examples according to the user's action. To classify a new example using the tree on the left, first determine the length. If it is long, predict skips. Otherwise, check the thread. If the thread is new, predict reads. Otherwise, check the author and predict read only if the author is known. This decision tree can correctly classify all input data.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

The tree on the right makes probabilistic predictions when the length is short.

A deterministic decision tree, in which all of the leaves are classes, can be mapped into a set of rules, with each leaf of the tree corresponding to a rule. The example has the classification at the leaf if all of the conditions on the path from the root to the leaf are true.

Hidden Markov Models

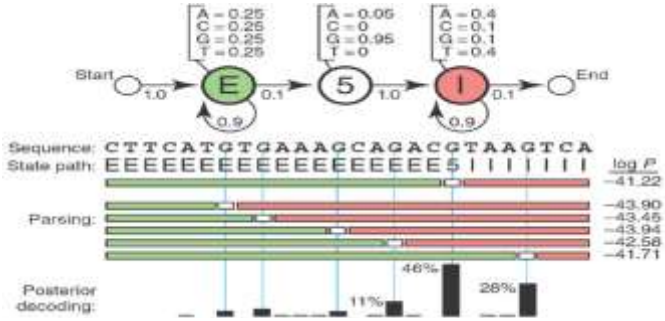


Figure 4: A toy HMM for 5' splice site recognition [2][3]

Hidden Markov models (HMMs) are a formal foundation for making probabilistic models of linear sequence 'labeling' problem. They provide a conceptual toolkit for building complex models just by drawing an intuitive picture. They are at the heart of a diverse range of programs, including gene finding, profile searches, multiple sequence alignment and regulatory site identification.

It's useful to imagine an HMM generating a sequence. When a state is visited, a residue is emitted from the state's emission probability distribution. Then which state to visit is chosen next according to the state's transition probability distribution. The model thus generates two strings of information. One is the underlying *state path* (the labels), as we transition from state to state. The other is the *observed sequence* (the DNA), each residue being emitted from one state in the state path. The state path is a Markov chain, meaning that what state is reached to next depends only on what state is the current one. Since only observed sequence is given, this underlying state path is hidden. The state path is a *hidden Markov chain*. The probability $P(S,\pi|HMM,\theta)$ that an HMM with parameters θ generates a state path π and an observed sequence S is the product of all the emission probabilities and transition probabilities used. An HMM is a *full probabilistic model*—the model parameters and the overall sequence 'scores' are all probabilities. Therefore, we can use Bayesian probability theory to manipulate these numbers in standard, powerful ways, including optimizing parameters and interpreting the significance of scores.

Case Based Reasoning

Case-based reasoning (CBR), broadly construed, is the process of solving new problems based on the solutions of similar past problems. An auto mechanic who fixes an engine by recalling another car that exhibited similar symptoms is using case-based reasoning. A lawyer who advocates a particular outcome in a trial based on legal precedents or a judge who creates case law is using case-based reasoning. So, too, an engineer copying working elements of nature (practicing biomimicry), is treating nature as a database of solutions to problems. Case-based reasoning is a prominent kind of analogy making.

It has been argued that case-based reasoning is not only a powerful method for computer reasoning, but also a pervasive behavior in everyday human problem solving; or, more radically, that all reasoning is based on past cases personally experienced. This view is related to prototype theory, which is most deeply explored in cognitive science.

Figure 5: Case Based Reasoning

Categories of Bioinformatics Tools

The size of data of molecular database is given in table:

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

Database	Data Size	URL
GenBank (July, 2001)	=12,244,000 sequences	Http://www.ncbi.nlm.nih.gov/Genbank/
Protein Information Resource, PIR (July, 2001)	≈233000 sequences	Http://pir.georgetown.edu/
Database of protein families and domains, PROSITE (April, 2001)	1474 different patterns, rules and profiles/matrices	Http://ca.expasy.org/prosite/
Protein Data Bank, PDB (July, 2001)	15531 structures	Http://www.rcsb.org/pdb/
Structural Classification of Protein, SCOP (July, 2000)	26219 Domains	Http://scop.mrc-lmb.cam.ac.uk/scop/
Protein Structure Classification - CATH (November, 2000)	25320 Domains	Http://www.biochem.ucl.ac.uk/cath/
Kyoto Encyclopedia of Genes and Genomes, KEGG (April, 2001)	150 pathways	Http://www.genome.ad.jp/kegg/

There are data-mining software that retrieves data from genomic sequence databases and also visualization tools to analyze and retrieve information from proteomic databases. These can be classified as:-

1. homology and similarity tools
2. protein functional analysis tools
3. sequence analysis tools

Bioinformatics is done with sequence search programs like BLAST, sequence analysis programs, like the EMBOSS and Staden packages, structure prediction programs like THREADER or PHD or molecular imaging/modeling programs like RasMol and WHATIF.

Homology and Similarity Tools:

Homologous sequences are sequences that are related by divergence from a common ancestor. Thus the degree of similarity between two sequences can be calculated while their homology is a case of being either true of false. This set of tools can be used to recognize similarities between original query sequences of unknown structure and function and database sequences whose structure and function have been elucidated.

Protein Function Analysis:

This group of programs allows comparing the protein sequence to the secondary (or derived) protein databases that hold information on motifs, signatures and protein domains. Highly significant hits against these different pattern databases allow approximating the biochemical function of query protein.

Structural Analysis:

This set of tool allows you to compare structures with the known structure databases. The function of a protein is more directly a consequence of its structure rather than its sequence with structural homolog tending to share functions.

Sequence Analysis:

This set of tools allows carrying out further, more detailed analysis on query sequence including evolutionary analysis, identification of mutations and compositional biases. The identification of these and other biological properties are all clues that aid the search to elucidate the specific function of sequence.

Other Bioinformatics Tools:

BLAST:

BLAST (**B**asic **L**ocal **A**lignment **S**earch **T**ool) comes under the category of homology and similarity tools. It is a set of search programs designed for the Windows platform and is used to perform fast similarity searches regardless of whether the query is for protein or DNA. Comparison of nucleotide sequences in a database can be performed. Also a protein database can be searched to find a match against the queried protein sequence. NCBI has also introduced the new queuing system to BLAST (Q BLAST) that allows users to retrieve results at their convenience and format their results multiple times with different formatting options. Depending on the type of sequences to compare, there are different programs:

- blastp matches an amino acid query sequence against a protein sequence database
- blastn matches a nucleotide query sequence against a nucleotide sequence database
- blastx matches a nucleotide query sequence translated in all reading frames against a protein sequence database

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

- tblastn matches a protein query sequence against a nucleotide sequence database dynamically translated in all reading frames
- tblastx matches the six-frame translations of a nucleotide query sequence against the six-frame translations of a nucleotide sequence database

FASTA:

FAST homology searches **a**ll sequences. An alignment program for protein sequences is created by Pearsin and Lipman in 1988. The program is one of the many heuristic algorithms proposed to speed up sequence comparison. The basic idea is to add a fast prescreen step to locate the highly matching segments between two sequences, and then extend these matching segments to local alignments using more rigorous algorithms such as Smith-Waterman [4].

EMBOSS:

EMBOSS
(**E**uropean **M**olecular **B**iology **O**pen **S**oftware **S**uite)
is a software-analysis package. It can work with data in a range of formats and also retrieve sequence data transparently from the Web. Extensive libraries are also provided with this package, allowing other scientists to release their software as open source. It provides a set of sequence-analysis programs, and also supports all UNIX platforms.

Clustalw:

It is a fully automated sequence alignment tool for DNA and protein
Sequences. It returns the best match over a total length of input sequences, be it a protein or a nucleic acid.

RasMol:

It is a powerful research tool to display the structure of DNA, proteins, and smaller molecules. Protein Explorer, a derivative of RasMol, is an easier to use program.

PROSPECT:

PROSPECT (PROtein Structure Prediction and Evaluation Computer ToolKit) is a protein-structure prediction system that employs a computational method called protein threading to construct a protein's 3-D model.

PatternHunter:

PatternHunter, based on Java, can identify all approximate repeats in a complete genome in a short time using little memory on a desktop computer. Its features are its advanced patented algorithm and data structures, and the java language used to create it. The Java language version of PatternHunter is just 40 KB, only 1% the size of Blast, while offering a large portion of its functionality.

COPIA:

COPIA (COnsensus Pattern Identification and Analysis) is a protein structure analysis tool for discovering motifs (conserved regions) in a family of protein sequences. Such motifs can be then used to determine membership to the family for new protein sequences, predict secondary and tertiary structure and function of proteins and study evolution history of the sequences.

Programming Languages in Bioinformatics:

JAVA-in-Bioinformatics:

Since research organizations are spread all around the globe ranging from private to academic settings, and a variety of hardware and OSs are being used, Java is emerging as a key player in bioinformatics. Physiome Sciences' computer-based biological simulation technologies and Bioinformatics Solutions' PatternHunter are two examples of the growing adoption of Java in bioinformatics.

Perl-in-Bioinformatics:

String manipulation, regular expression matching, file parsing, data format inter-conversion etc are the common text-processing tasks performed in bioinformatics. Perl excels in such tasks and is being used by many developers. Yet, there are no standard modules designed in Perl specifically for the field of bioinformatics. However, developers have designed several of their own individual modules for the purpose, which have become quite popular and are coordinated by the project named BioPerl.

Bioinformatics Projects:

BioJava:

The BioJava Project is dedicated to providing Java tools for processing biological data which includes objects for manipulating sequences, dynamic programming, file parsers, simple statistical routines, etc.

BioPerl:

The BioPerl project is an international association of developers of Perl tools for bioinformatics and provides an online resource for modules, scripts and web links for developers of Perl-based software.

BioXML:

A part of the BioPerl project, this is a resource to gather XML documentation, DTDs and XML aware tools for biology in one location.

Biocorba:

Interface objects have facilitated interoperability between bioperl and other perl packages such as Ensembl and the Annotation Workbench. However,

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

interoperability between bioperl and packages written in other languages requires additional support software. CORBA is one such framework for interlanguage support, and the biocorba project is currently implementing a CORBA interface for bioperl. With biocorba, objects written within bioperl will be able to communicate with objects written in biopython and biojava. For more information, see the biocorba project website at http://biocorba.org/. The Bioperl BioCORBA server and client bindings are available in the bioperl-corba-server and bioperl-corba-client bioperl CVS repositories respecitively, see http://cvs.bioperl.org/ for more information [4].

Ensembl:

Ensembl is an ambitious automated-genome-annotation project at EBI. Much of Ensembl's code is based on bioperl, and Ensembl developers, in turn, have contributed significant pieces of code to bioperl. In particular, the bioperl code for automated sequence annotation has been largely contributed by Ensembl developers. Describing Ensembl and its capabilities is far beyond the scope of this tutorial the Ensembl website at http://www.ensembl.org/.

bioperl-db:

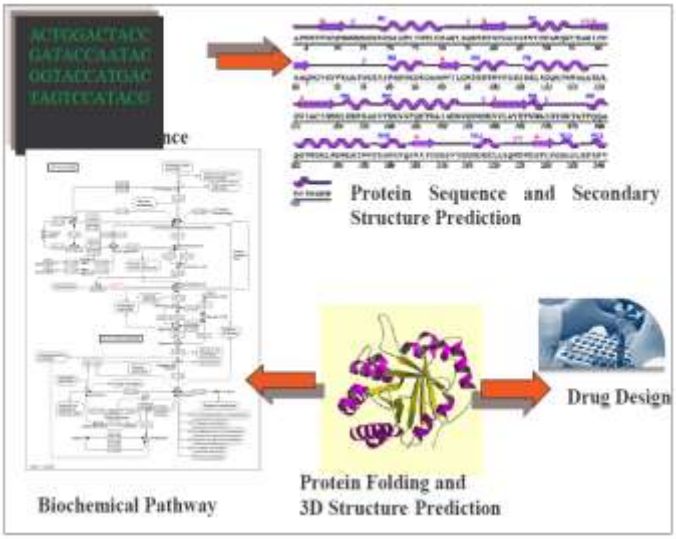
Bioperl-db is a relatively new project intended to transfer some of Ensembl's capability of integrating bioperl syntax with a standalone Mysql database (http://www.mysql.com) to the bioperl code-base. It is worth mentioning that most of the bioperl objects mentioned above map directly to tables in the bioperl-db schema. Therefore object data such as sequences, their features, and annotations can be easily loaded into the databases, as in $loader->store($newid,$seqobj) Similarly one can query the database in a variety of ways and retrieve arrays of Seq objects.

Biopython-and-biojava:

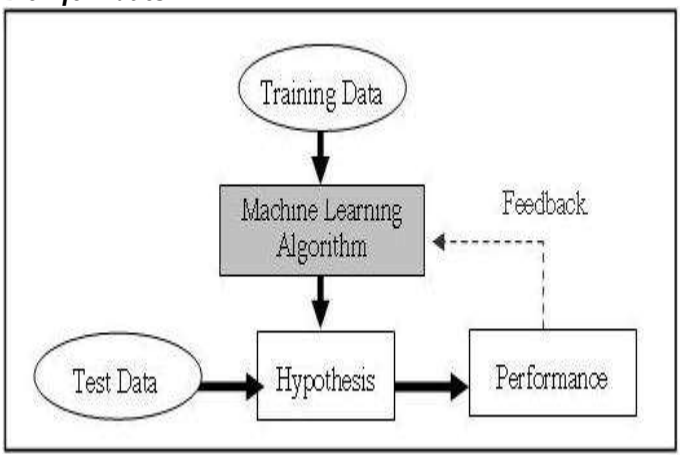
Biopython and biojava are open source projects with very similar goals to bioperl. However their code is implemented in python and java, respectively. With the development of interface objects and biocorba, it is possible to write java or python objects which can be accessed by a bioperl script, or to call bioperl objects from java or python code. Since biopython and biojava are more recent projects than bioperl, most effort to date has been to port bioperl functionality to biopython and biojava rather than the other way around. However, in the future, some bioinformatics tasks may prove to be more effectively implemented in java or python in which case being able to call them from within bioperl will become more important. For more information, visit the biojava http://biojava.org/ and biopython http://biopython.org/ websites [4].

Challenges in Bioinformatics

The bioinformatics challenges are shown in figure:-



Challenges of Machine learning techniques in Bioinformatics



There are various challenges of Machine Learning techniques in bioinformatics:-

1. Learning in dirty biological database
2. Generative Vs Discriminative
3. Approximation Vs Explanation

Single Vs Multiple Methods

Conclusion

This paper explains in detail about role of data mining techniques in bioinformatics. The attempt has been made to set the basic platform for the development of new approaches for bioinformatics. Existing techniques are playing a vital role in

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

Bioinformatics but the detailed analysis still requires more efforts.This study will be helpful for researchers working in concerned field.

References

1. *http://www.cs.ubc.ca/~murphyk/Bayes/bnintro.html*
2. *What is a hidden Markov model? Sean R Eddy*
3. *http://www.nature.com/nbt/journal/v22/n10/full/nbt1004-1315*
4. *http://bioinformaticsweb.net*