Bioinformatics and Learning :

Will Machines Outsmart Humans ?

Stanford Biochemistry 218 Project

ISEP 2003

Cécile Goepfer

Summary

1 Introduction

Every learning system can be considered as a black box which takes particular inputs and generates outputs. Humans and animals work this way : they receive information from their surrounding environment, analyse it and generate new information. The human brain, which enables us to treat this information, is the most powerful learning system ever known. As computing theories began to emerge, the comparison between the mind and a computer became commonly used. In the 1960s, the scientists began to work on this concept. Then mimicking the human behavior became the new challenge for computational scientists. By imitating the way the human learnt, scientists gave birth to artificial intelligence. This approach is not only used in computational sciences but also in other fields, such as biological sciences.

The application of computational methods to biological science started in the early 1980s, with the first data banks. Biologists and other scientists realised the advantages of using computers to analyse the biological data. However, the biological systems become more and more complex. The tools that used conventional algorithms are becoming unable to handle the large and rapidly expanding amount of data and thus to address real-world problems. Because machine learning approaches are efficient for data formalization, sequence and structures analysis, data integration and biological interpretation of genetic information, they tend to dethroned traditionnal methods.

The aim of this paper is to describe and compare the Human Learning theories and the Machine Learning methods applied to Bioinformatics.

2 Learning

2.1 Definition

Learning is « an adaptative process in which the tendency to perform a particular behaviour is changed by experience » (« Psychology : The Science of Behavior », N. R. Carlson, W. Buskist, G. N. Martin).

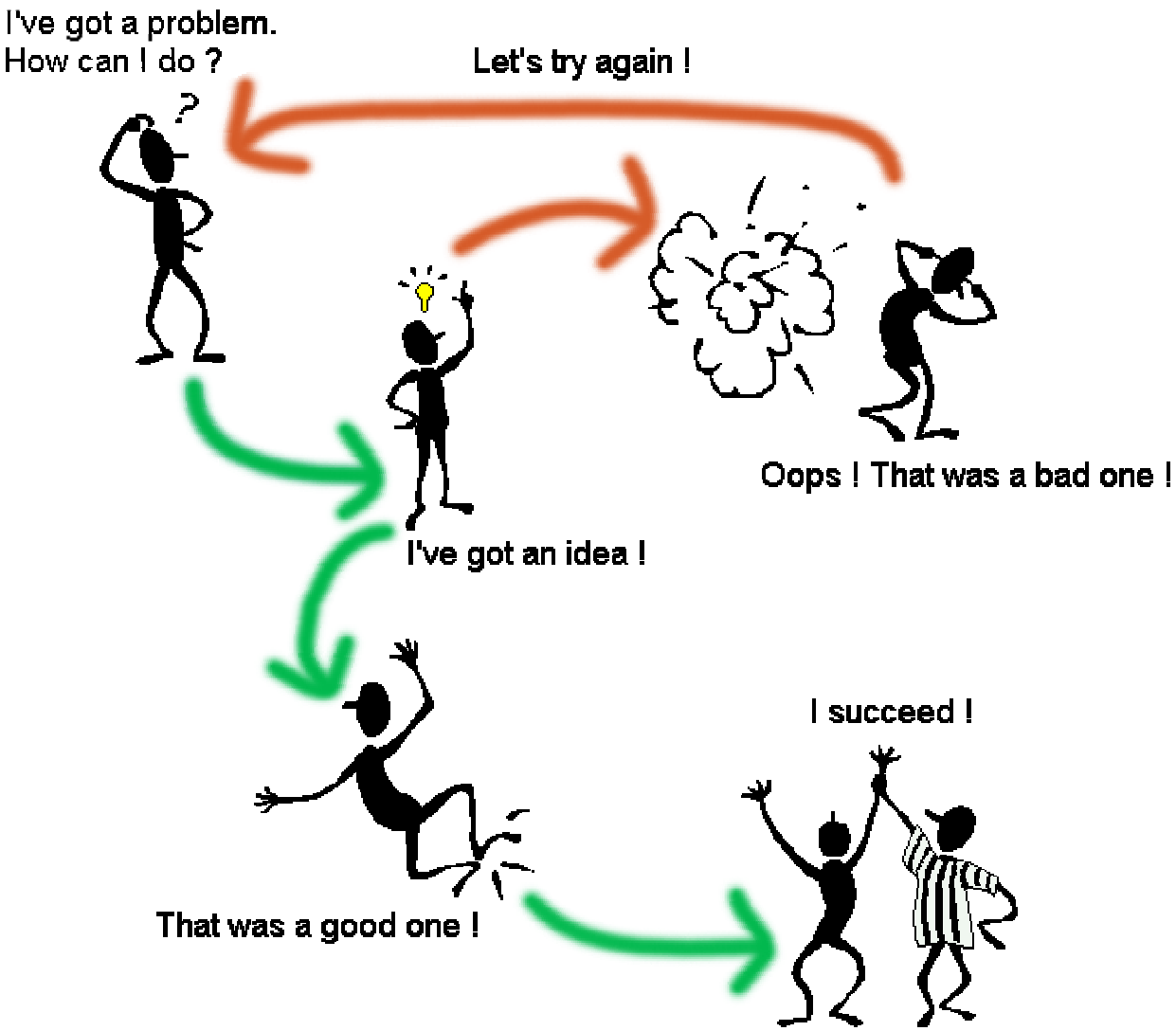
Learning is much more complex in human beings than in animals because of the emotional part. Whereas humans behave in an emotional way, animals would rather react instinctively. Nevertheless both learn by experience.

They Try,
Observe (receive a feedback),
Adapt their behavior.

2.2 Problem solving

Learning is often related to problem solving. A problem is a « state of affairs in which we have a goal but do not have a clear understanding of how it can be attained » (Holyoak, 1990).

The basic principle of problem solving can be described by the following scheme :



When dealing with a problem, we generally build a path to reach the solution by taking several actions. We find general rules (called *heuristics*) that are useful in guiding our search for this path. These rules tell us what to focus on and what strategy to take. There are two ways of reasoning : we can infer specific instance from general principles (*deductive reasoning*) or find general principles from specific facts (*inductive reasoning*).

What we generally do is classify the different data according to their characteristics as concepts. Most thinking deals with relationships and interactions among the concepts and involves the manipulation and the combination of these concepts.

In Artificial Intelligence, scientists mimic the human or animal behavior in building intelligent robots. However, in order to simplify the process they tend to ignore the emotional parts that affect human reasoning and concentrate on automatically behavior. Thus they can produce learning agents that are able to learn and develop by themselves.

2.3 Learning algorithms

"Tell me, I forget.
Show me, I remember.
Involve me, I understand."
—Chinese proverb

In order to understand how the learning process occurs in human beings and animals, scientists have developed learning algorithms. Such algorithms explain how a learner changes his behavior according to the inputs and feedback he receives from its environment. As we will see further, these algorithms are used as well in human learning as in machine learning. These algorithms can be placed into three different groups, depending on the feedback the learner receives from its environment.

2.3.1 Supervised Learning

As the names suggests, supervised learning involves learning with some supervision from an external expert. The learner is told what his response should be. He then compares his own response to the awaited response, and adjusts his internal behavior in order to give the closest answer to the targeted response. Thus he will be more likely to give the appropriate response the next time he receives the same input.

As an example, consider a student passing an exam. The exam is marked and the student notices the questions he answered incorrectly. After being shown the correct answers, he will remember them in order to succeed next time.

Supervised learning is studied in machine learning, statistical pattern recognition, and artificial neural networks.

2.3.2 Unsupervised Learning

In unsupervised learning, the learner isn't given any feedback at all. Instead, he is asked to observe the similarities and differences among the inputs he receives, and to represent them in a more efficient way to himself, as clusters or categories. It is indeed oftenly easier to divide a big task in small ones in order to treat them separately.

For example, when we try to assemble the pieces of a jigsaw, we first gather together the pieces having the same color, the same form, and so on. Then we put together the differents parts in order to build the entire jigsaw.

One concrete example is the Decrypton Project launched in 2001 in France, which aimed to build a database of all known human proteins and to search the similarities between them. In order to decrypt the genome, the scientists suggested each owner of a computer lend its machine to the science. The user had to download a small program that would run in background and make some calculations. Then by putting together the data generated by each computer (75000 webusers participated !), the scientists succeeded to compare the 500 000 proteins constituting the living world.

Other examples are the Kohonen Artificial Neural Networks (KANNs), in which the algorithms seek clusters in the data, and Principal Components Analysis (PCA).

2.3.3 Reinforcement Learning

"Everybody falls the first time"
--The Matrix

Reinforcement Learning is a third alternative, much closer to unsupervised than supervised learning. It corresponds to the 'cause and effect' method. In this method, the learner has no clue concerning the actions he should take. Thus he tries every action in order to discover the one that yields the best reward. He then receives feedback about the appropriateness of his response.

Without knowing it, we all used reinforcement learning as a child : every action a child takes is a try. His parents give him a feedback : If he makes a mistake , they rumble him so that it won't happen again, and if it is a good idea, they encourage him. The child observes its parents' reaction and adapts its behavior accordingly.

Reinforcement learning is used in several fields, such as game playing, learning in a micro-world, on-line control, and autonomous robot exploration.

3 Human Learning

3.1 Introduction

At the beginning of the century, most learning theorists still believed that children were not capable of complex mental activities. They considered newborn's mind as a blank table on which everyday life experience was gradually impressed. But soon this view was criticized. New methodologies came and enabled psychologists to prove that small children are able to build their own concepts about their surrounding environment, as life experience accumulates.

3.2 The laws of human learning

While observing human learning, learning theorists noticed some laws to which human learning obeys, that they called "laws of learning". These are listed below :

1. *Law of effect* : We are likely to repeat a behavior that formerly leaded to a success.
2. *Law of causal relationship* : To learn the relationship between an action and a result, we need to see an obvious causal relation between them.
3. *Law of causal learning* : We try to repeat the behaviors that have an obvious causal relation to the result we are waiting for.
4. *Law of causal learning* : We try to avoid the actions that have an obvious causal relation to a result we don't wish.
5. *Law of information feedback* : The outcome of an action gives some information on that action.

3.3 The main human learning theories

Several theories were suggested, trying to explain human learning. The most important are described in this section.

3.3.1 Constructivism

In constructivism learning is seen as a search for meaning. It is a philosphy of learning which states that each of us constructs his own understanding, his own mental models of the world, according to his experience. We do not just memorize and adapt someone else's understanding. Therefore learning consists simply in adjusting these mental models in order to deal with new experiences.

3.3.2 Behaviorism

The behaviorism theory was elaborated by the american psychologists J. B. Watson and B.F. Skinner. It only focuses on observable behavior, disregarding mental activities. The spirit is considered as a black box. According to this theory, mental states are patterns of behavior which don't imply mental or conscient states. These are only body movements.

Behaviorists believe that conditioning is a universal learning process. According to the behavioral pattern it produces, there are two types of conditionning :

A natural reflex responding to a stimulus is considered as *classic conditioning*. Pavlov's observation that dogs salivate while eating or even seeing food is one example.

A reinforced response to a stimulus is considered as *behavioral or operant conditioning*. It is likely to a feedback system : if a response is rewarded or reinforced, it becomes more probable in the future.

Behaviorism has been widely criticized. For it disregards mental activities, it does not account for all kind of learning. Everyday life confirms the fact that human behaviors are caused by mental activities. For example, if I believe that it's gonna rain today, I will take my umbrella.

Moreover, behaviorism is going around in circle : behaviorist analysis of mental states involves other mental states. This is particularly true if the suggested analysis implies the analysed behavior. For example, "I'm efficient because I'm competent and I'm competent because I'm efficient".

However, behaviorism reinforcement techniques can be very effective. They are in particular used successfully in human disorders such as autism and antisocial behavior.

3.3.3 Learning Styles

Each of us has its own way of learning. The learning style theory states that the quantity of learning by each individual depends more on the way the individual learns (on its own or under the supervision of someone else), than on its intelligence.

This theory is based on research demonstrating that different individuals tend to process and perceive information differently, according to their growing environment's requirements.

There are generally two types of individuals :

Concrete and abstract perceivers : Concrete perceivers get information through direct interaction with the world. Abstract perceivers are theorists : they take information in through analysis and thinking.

Active and reflective processors : Actives processors understand an experience by immediately putting it into practice, whereas reflective processors need to think about it.

3.3.4 Observational Learning

Observational learning, or *social learning theory*, consists in modifying its own behavior after observing a model. The observer will imitate the model's behavior, if he finds it attractive. The affect is called *vicarious reinforcement* if it is positive, *vicarious punishment* otherwise.

There are four different ways of learning by observation :

- *Attention* : The observer must pay attention to their surrounding environment
- *Retention* : The observer must identify the model's behavior and remember it
- *Production* : The observer must produce some kind of behavior, intellectual or physical
- *Motivation* : In general, the observer will react only if he is motivated. Reinforcement or punishment are particularly efficient in this process.

Attention and retention help the observer to acquire the model's behavior ; production and motivation help him to control his act.

What an individual observes influences his behavior but the individual's behavior also influences his environment. These influences are reciprocal. Thus the relationship between a person, the person's behavior and his environment is called *reciprocal determinism*.

3.3.5 Communities of Practice (CoPs)

In this approach, learning occurs within communities that gather together people which share the same beliefs. We are all more or less implied in such communities, at school, at work, etc. This approach is very efficient as each member of the community brings and shares his knowledge, gives and gets motivation. This kind of communities can be found in big societies, such as IBM. Chats, forums and Peer to Peer are other examples. They enable people having the same interests to share their ideas and goods.

3.3.6 Brain-based Learning

This learning theory is based on the observation of the brain. The main idea is that the brain is a parallel processor, able to perform several tasks at once. Each brain is unique.

Everybody learns, everytime. The search for meaning in each individual is innate and results from patterning and emotions. Challenge enhances learning, threat inhibits it. Learning involves as well focused attention as external perception, conscious and unconscious processes.

Brain-based learning distinguishes spatial memory and rote memory. Facts that are embedded in spatial memory are the one we understand at best.

Three techniques are associated with brain-based learning : *orchestrated immersion* (the learner is fully immersed in the learning experience) ; *relaxed alertness* (the learner is placed in a challenging environment and tries to eliminate its fears) ; *active processing* (the learner generates information by actively working on it).

3.4 The main human learning theorists

3.4.1 Jean Piaget (1896-1980)

Learning is *« the acquisition of specific behaviour for handling a particular task in a particular context »* (J. Piaget).

Swiss biologist and psychologist Jean Piaget is renowned for constructing a highly influential model of child development and learning. He eradicated the *'tabula rasa'* view of the child's mind. He affirms that the child development increases continuously from innate reflexes to mature abstract thoughts, stating that the child's development goes through the progressive socialization of an individual thought, first rebel to social adaptation, then more and more penetrated by the surrounding influences of adults. While observing children, he totally reorganizes the key concepts linked to the child development and explains them as several stages from sensorimotor stage to highly mental activities, during which the child builds cognitive structures.

These stages are :

1. *Sensorimotor stage (birth - 2 years old)* : The child needs to interact physically with his or her environment in order to understand the reality.
2. *Preoperational stage (ages 2-7)* : The child is not yet capable of abstract thoughts and still needs concrete physical interactions.
3. *Concrete operations (ages 7-11)* : The child begins to build concepts that explain his or her physical experiments.

4. *Formal operations (beginning at ages 11-15)* : The child is able to think conceptually, like an adult.

3.4.2 B.F. Skinner (1904-1990)

Contrary to Piaget, Skinner believes a child learns better with reinforcement. His theory states that if a child is gradually rewarded for each steps he does towards a specified goal, he will finally reach the right solution.

The idea of programmed learning suggested by Skinner became popular in education in the 1950's and 1960's and is still used today. This kind of program consists in breaking a specified task into small steps. The child receives immediate feedback for each progress before he can proceed to the next step and each progress is immediately rewarded. This makes the child take an active part in his learning and let him proceed at his own pace.

3.4.3 Gardner and multiple intelligences

Howard Garner's theory of multiple intelligences was first published in 1983 and encountered great success in education. Gardner doesn't believe that there is a general intelligence, but distinguishes eight different kinds of intelligence :

- *Verbal-Linguistic* : The ability to use words and language
- *Logical-Mathematical* : The ability of logical reasoning, organizing the world, counting. This is how Piaget defines the notion of "intelligence"
- *Visual-Spatial* : The ability to create mental images and to perceive the world in three dimensions
- *Body-Kinesthetic* : The ability to use his body in a wise manner, to express himself through movements, to be skillful
- *Musical-Rhythmic* : The ability to be sensible to rythm and music
- *Interpersonal* : The ability to communicate with others
- *Intrapersonal* : The ability to have a good comprehension of himself
- *Naturalist* : The capacity to identify and classify the different forms and struture (mineral, vegetal or animal) that exist in nature - this one was added by Gardner to the seven others in 1996.

3.4.4 Vygotsky and Social Cognition

The importance of active role in learning was also encouraged by Vygotsky (1968), who was deeply interested in social environment.

The social cognition model considers culture as the major influence on individual development. Indeed culture exists only in the human species and every human child grows in a cultural environment. Thus every child's development is influenced by the culture in which he evolves. First, children acquire most of their knowledge through culture. Second, the surrounding culture has an impact on the way the child think, what Vygotskians call the *tools of intellectual adaptation*. In brief, culture determines what the children think about and how they think.

During his development, the child learns through problem-solving experiments shared with an adult. Little by little the responsability for guiding the experiment transfers to the child. Language is one example. It is used by the adults to transmit culture during the first years of the child, until the child is able to use internal language to guide his own behavior. One of the best idea from Vygotsky in his influence on developmental psychology was his concept of a *zone of proximal*

development. It relates to the bandwidth between what a child can do on its own and what he cannot do without supervision.

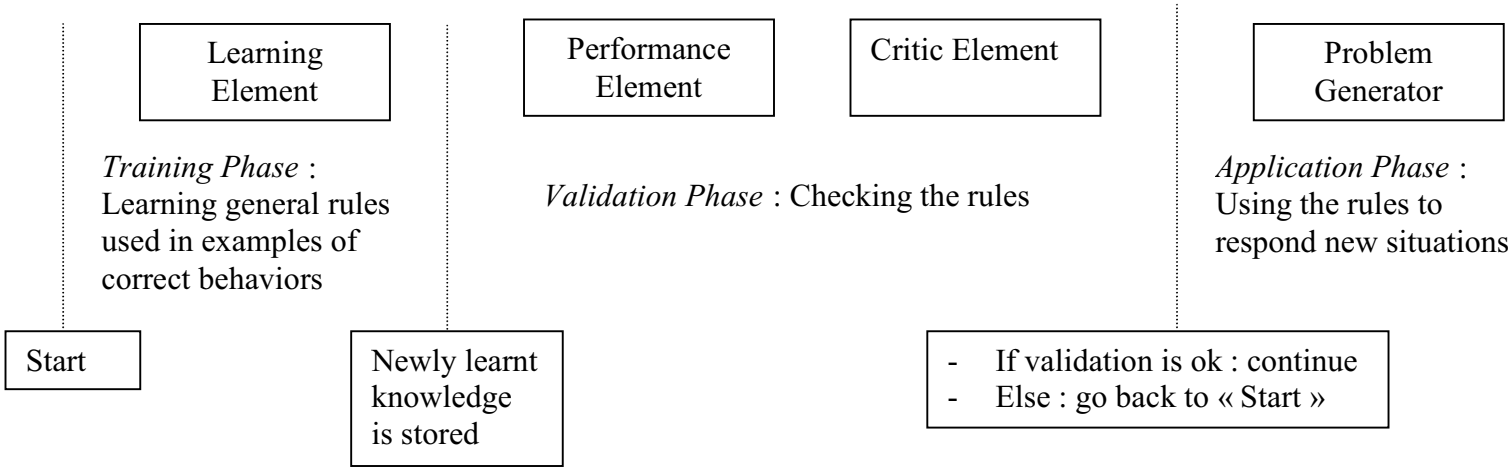
Thanks to the analyses on Human Learning, scientists were able to construct new technologies based on the human behavior which were cost-efficient and time-consuming. Machine Learning is one of them, and surely one of the most powerful.

4 Machine learning

4.1 Introduction

The main purpose of Machine Learning is to make a computer learn from experience. It enables the computer to acquire knowledge, reason with the available data to produce new knowledge, respond to new situations without supervision and adapt to the environment.

The machine learning model generally involves four elements and goes through three phases, as shown in the following scheme :



The process involves four elements, shown in the scheme :
- The *Learning Element*, which improves the process' performance
- The *Performance Element*, which decides which actions to take
- The *Critic Element*, which tells how the process is doing
- The *Probleme Generator*, which suggests actions that could bring new experience

While designing a learning system, the main problems generally encountered by experts are : to find the correct source of experience (the training data sets) ; to define the exact objective of the learning system ; and finally how to measure the performance of the system.

A good learning model is composed of an accurate and efficient agent, valuable outputs, consistency and also quite simple tasks.

4.2 The need of Machine Learning in Bioinformatics

4.2.1 Limitations of traditional methods in biological sciences

Nowadays it seems hard to imagine the world without any computer. At the time of old-fashioned computers, scientists had to collect information by hand and interpreted it into knowledge. This method was quite time-consuming and inefficient. In the 1990's, with the creation of the HUGO (Human Genome Organization) Project, the HGP (Human Genome Project) and the birth of Internet, the number of experiments exploded and it became impossible for researchers in bioinformatics to deal with such an amount of experimental data. They needed automatic systems that could produce human-comprehensible hypotheses from data.

Thus scientists turned towards computers, and created programs such as BLAST (1990) and GRAIL (1991), which permitted the automatic treatment of new data. These were among the first examples of machine learning in bioinformatics.

4.2.2 Machine learning and Bioinformatics

One of the first application domains in machine learning was molecular biology. Machine learning approaches perform well in domains where there is a vast amount of data but little theory and this is exactly the situation in molecular biology research. Another advantage of machine learning approaches is that they can easily be adapted to a new environment. This is important in molecular biology research because new data are generated every day and it may be necessary to update the initial concept or learning hypotheses. Moreover machine learning methods are easily revisable, which is crucial for generating acquire new knowledge and make new hypothesis, which can then be interpreted by a human expert. This feedback loop between in silico and in vivo experiments improves the knowledge discovery process.

Machine learning methods cover a lot of problems in bioinformatics, such as logic programming, rule explanation, finite-state machines, data mining, functions system and problem solving systems.

4.3 Machine learning methods in Bioinformatics

4.3.1 Clustering and classification

Clustering is an unsupervised method that organises the data into classes of object having the same features. There are two main clustering algorithms : *hierarchical clustering*, in which the input data is grouped in a hierarchical way ; *k-clustering (partioning),* where each input object is assigned to exactly one group.

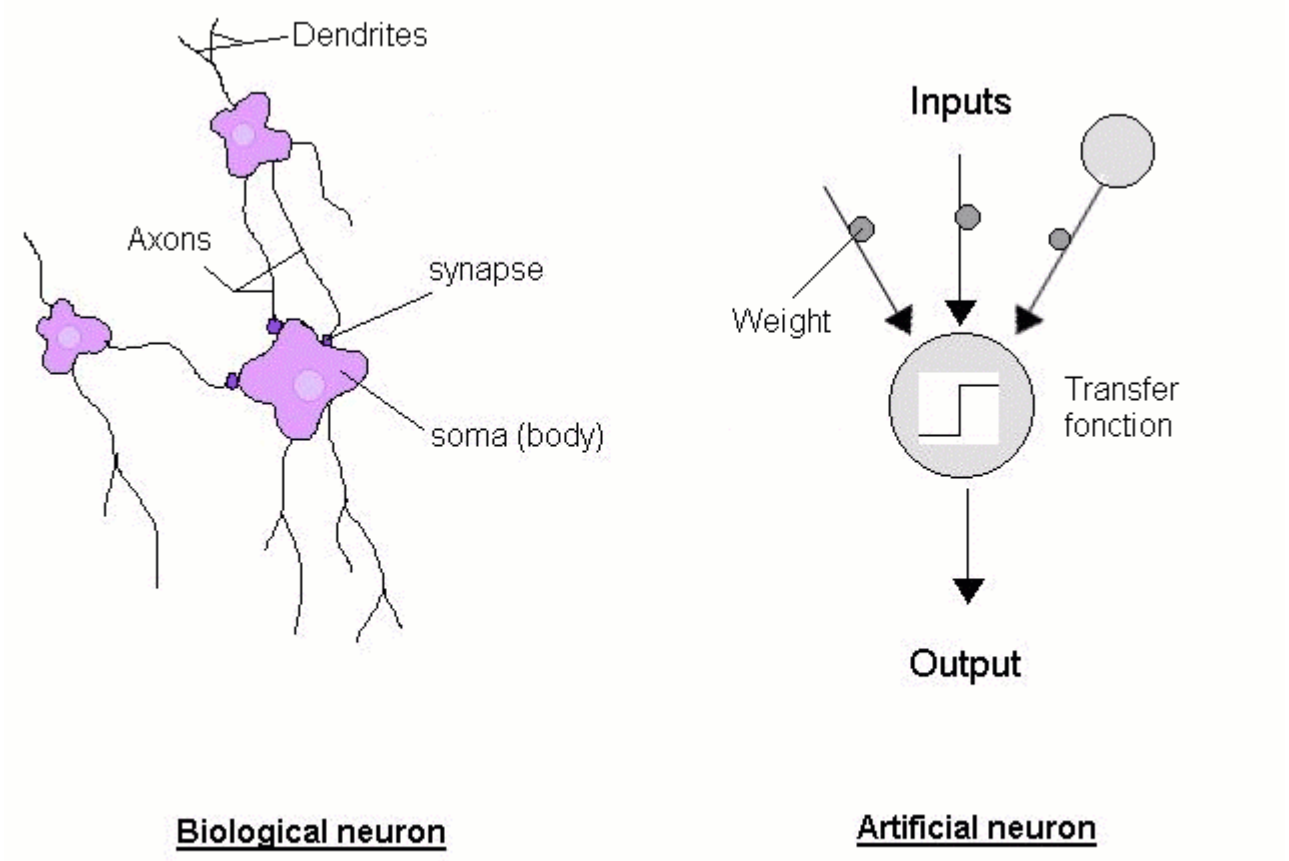
Classification is a supervised method that consists in predicting to which existing groups a given obect should belong.

Clustering and classifaction are descriptive and expressive approaches that give results which are understandable by biologists.

A lot of clustering and classification algorithms have been applied in bioinformatics, such as Support Vector Machines, Decision Trees, Hidden Markov Models, Neural Networks, UPGMA, Self Organizing Map, Principal Component Analysis, etc. We shall see some of them in the next parts.

4.3.2 Artificial Neural Networks

Artificial neural networks (ANNs) mimic the biological neural network of the human brain. The following scheme shows the comparison between a biological neuron and an artificial neuron.



Both biological and artificial neurons are single unit that generate a specific output from the information they receive and that are connected to other neurons.

In a biological neuron, the dendrites accept inputs, the soma processes the inputs, the axon then turns the processed inputs into outputs and finally the synapses transmit the outputs to the next neuron.

The artificial neuron receives a variable number of inputs from other neurons. A numerical value, the *weight*, which represents the strength of the connexion is attached to each input (it corresponds to the synapse in a biological neuron). Each neuron has an unique output, which is then transmitted to the neighboring neurons.

In ANNs, neurons correspond to *nodes*. An ANN is a graph composed of nodes with weights attached to them and linked with each other. The output of a node depends on the weight of the different inputs he receives.

In reality, an ANN is composed of elementary processors tightly connected that work in parallel. Each processor performs a particular task (presented as the « Transfer fonction » in the scheme above) and transmits the result to the next processor in the network. Learning is accomplished by adjusting these weights so that the whole network outputs the appropriate results.

Many neural networks learn using an algorithm called *backpropagation*. At the beginning, random weights are associated to its nodes. Then the network is given an input example and computes the output. Then an output error (the difference between the observed and desired output) is calculated and the weights are adjusted to decrease the error. The process is then repeated until the minimum error is found. This process is called “training”.

Most ANNs are made of several layers of nodes linking each other. Generally there are three layers in the network : the input layer, the output layer and a hidden layer in between them. Depending on the internal organization of the nodes in the network's layers, there are different types of ANNs architectures such as the feedforward architecture, the recurrent architecture and the layered architecture.

ANNs were among the first techniques applied in biological analysis, and among the most common machine learning approaches used in bioinformatics. They can indeed solve many real-world problems.

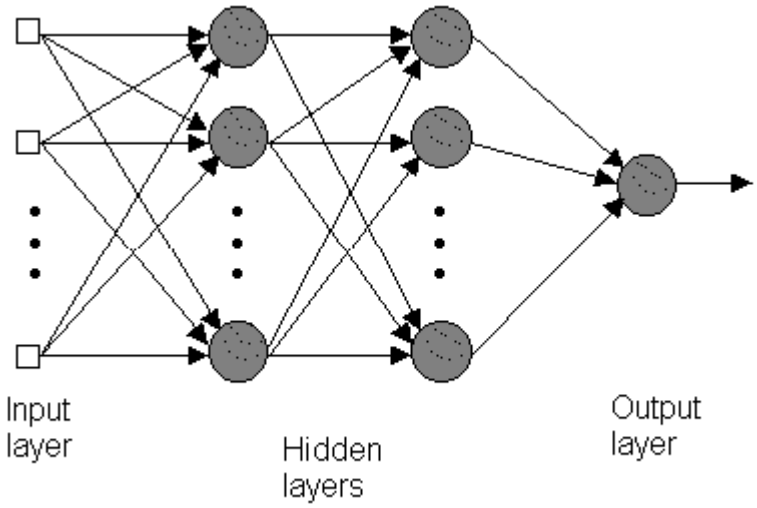
The most important caracteristic of an ANN is its ability to learn. Both supervised and unsupervised learning can be used in an ANN : In supervised learning the user tries to obtain a specified output by giving to the ANN the awaited answers. The network is able to compare its results with the awaited results and adjust the weights accordingly. The user can then compare the failures and success on the whole network. In unsupervised (or automatic) learning the network modifies the weights associated to its nodes itself.

Moreover, ANNs are very flexible : They can represent linear and non-linear relationships, compute deterministic or probabilistic outputs, have a asynchronous or synchronous dynamic, etc.

In addition, they perform well when analysing dirty data, which is useful in biology as biomolecular data are often noisy.

The main drawback of ANNs is that they are complex statistical models. Moreover it can be hard to evaluate the approaches of each nodes and in this case it may be impossible to validate the network. At last, training can be slow.

There are many different types of ANNs, but the most common is the *multilayer perceptron*. The multilayer perceptron is the most simple and the most renown artifical network. The structure is very simple : an input layer, an output layer and one or more hidden layers. Each neuron is linked to the neurons of the previous layer only. The multilayer perceptron is known as a supervised network as it requires a given output in order to learn. A graphical representation of a multilayer perceptron is shown below.

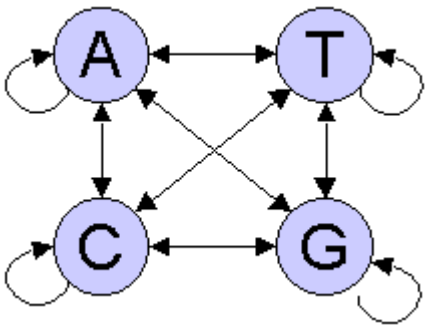


ANNs have been widely used in the protein structural and functional prediction (Hirst & Sternberg, 1992; Qian & Sejnowski, 1988; Sasagawa & Tajima, 1993; Fariselli & Casadio, 1996; Nakata, K. 1995; Macklin & Shavlik 1993) and protein classification (Wu *et. al.* 1995; Ferran & Ferrara, 1992).

4.3.3 Hidden Markov Models

Hidden Markov Models (HMMs) are general statistical models for ‘linear' problems like sequences and have been widely used in speech recognition since the early 1970s. It was introduced in the bioinformatics community in the 1990s by Haussler and Krogh.

When dealing with data where the succession of elements is important, such as in molecular biology, it is useful to have a model in which the probability of one element depends on the previous ones. For example, given a sequence of bases, it is interesting to know the probability of a base to appear in the sequence, regarding the previous bases.



Example : a first-order markov chain figuring a DNA fragment. Each base can be followed by any other base, including itself, with a specific probability.

Generally HMMs derive from the first-order Markov chain that focuses only on the sequence state. A Markov chain is a collection of states each associated with meaningful biological properties. Three categories of states can be found in an HMM : the match state, the insertion state and the deletion state. Each state emits symbols. The path from one state to another is associated to a probability, called *transition probability*. It is possible to calculate the probability that a sequence belongs to a given model by observing the transitions appearing in this sequence then by looking at the model to obtain the corresponding probabilities. The final probability is the product of the transition probabilities. The word “hidden” caracterizes the fact that the emission of a data from a state is random.

HMMs have become popular in sequence modelling and multiple alignment (Karplus *et. al.*, 1998), protein structure prediction (Asai *et. al.*, 1993; Goldman *et. al.*, 1996), phylogenetic analysis (Felsentein and Churchill, 1996; Thorne *et. al.*, 1996) and profiling. Pfam is a protein domain database constructed from HMMs and multiple sequence alignments (Sonnhammer *et. al.*, 1997). This technique is also used in the SWISS-PROT 22 database for searching other sequences that are members of the given protein family, or contain the given domain.

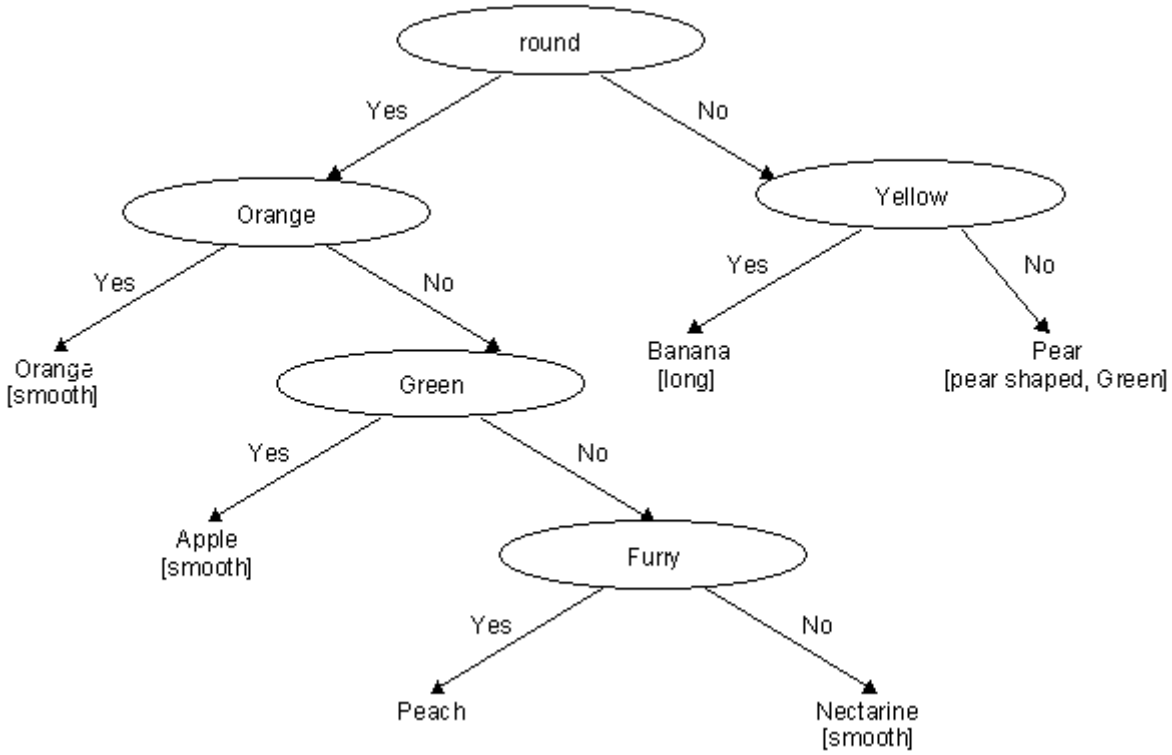
4.3.4 Decision Trees

The decision tree was developed by Quinlan. It is also known as *classification tree* or *regression tree*. The decision tree is a supervised learning technique. It is one of the widely used machine learning method because of its simplicity and practical approach. The most common algorithm used to build a decision tree is the ID3 algorithm.

Like a normal tree, a decision tree is composed of a root, branches and leaves. The non terminal nodes (called *decision nodes*) are associated to a question (*attribute*). Answers (*values*) are associated to the branches linking to the children of a non-terminal node. The path from the root to a node corresponds to a set of questions and answers. The items attached to a non-terminal node are those which correspond to the set of questions and answers along the path going from the root to

this node. This set of questions and answers determines a conjunction of attributes which defines one possible example.

Example of a Decision Tree build using the ID3 algorithm : Starting from a round fruit it goes trough questions such as « is the fruit orange ?» or « is the fruit furry ?» and finally reaches the right answer, that is « Peach ».



Source : « Machine Learning – Why and How explained »

Decision trees are easy to use, practical, and perform well in noisy environments. However, overfitting of the data and overlapping in the classes may occur, and decision tress can be hard to optimise.

The decision trees are used in classifying membrane protein sequences according to functional classes (Shimozono *et. al.*, 1992), protein structure prediction (Cherkauer and Shavlik, 1993 ; Selbig *et. al.*, 1999), locating protein coding genes (Salzberg, 1995).

4.3.5 Support Vector Machines

The Support Vector Machines (SVM) method is a kernel method invented by V. Vapnik. It aims at creating functions from a training data set. The functions can be classification functions or general regression functions.

Given a training data set, Support Vector Machines choose the hypothesis corresponding to the largest sphere that can be inscribed in the space of all possible hypothesis. The boundaries of this space that are tangent to the sphere define the support vectors.

SVM method is a powerful classification learning approach and is claimed to outperform most other algorithms. Although SVMs have good generalisation performance, training a large data set with SVMs can be slow.

Even though SVMs is a new learning technique, it is already popular in the bioinformatics research and has been used in several applications such as protein fold recognition (Ding and Dubchak, 2001), classification of microarray data (Furey *et. al.,* 2000) and recognition of translation initiation sites (Zien *et. al.,* 2000). They can also be applied to regression, classification, and density estimation problems.

4.3.6 Inductive Logic Programming

Inductive logic programming derives from inductive learning and logic programming. It aims to develop methods to induce hypothesis from observations and to synthesize new knowldege from experience. Given background knowledge and examples, this method will predict the simplest hypothesis. Then a set of IF-THEN rules can be created from the hypothesis.

Because inductive logic programming takes the advantages of inductive learning and logic programming, it is considered to be more powerful than classical machine learning techniques. Moreover, this technique is easily understandable by humans.

The disadvantage of inductive logic programming is the lack of probability in its rules. This is important while solving bioinformatics problems because of their high degree of uncertainty.

Inductive logic programming has been applied in several research areas in bioinformatics such as protein secondary structure prediction (Sternberg *et. al.*, 1992, Muggleton, *et. al.*, 1995), drug design (King *et. al.*, 1995), and mutagenicity prediction (Srinivasan, *et.al.*, 1995).

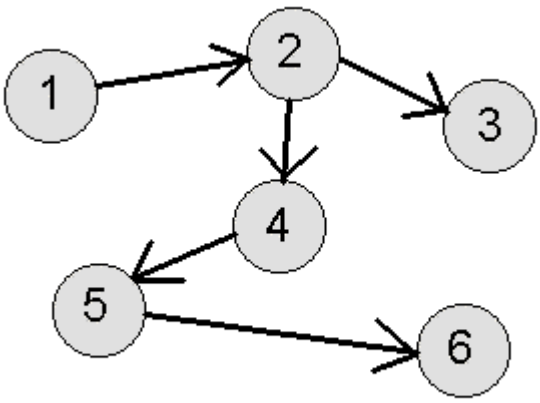
4.3.7 Knowledge Discovery in Databases

The Knowledge Discovery in Databases is the extraction of useful information from databases. It is based on the hypothesis that useful information are hidden in the database. The main idea of the method is to observe the database and find some patterns in the database that are understandable by experts. Interpreting and evaluating these patterns will lead to the discovery of new knowledge.

This technique is mainly used to identify structural regularities in proteins (Su *et. al.*, 1999), compare topology based protein structure (Gilbert *et. al.*, 1999; Gilbert *et. al.*, 2000), and in gene-expression microarray data (Zweiger, 1999).

4.3.8 Bayesian Networks

The word « bayesian » comes from the Reverend Bayes and more precisely from his famous theorem. A bayesian network is a probabilistic graphical model which permits to obtain, capitalize and make use of knowledge.

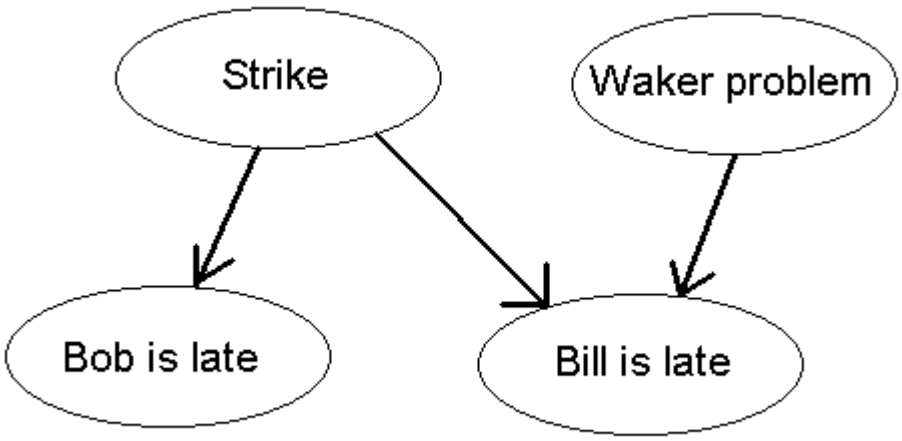


In a bayesian network, the nodes represent probabilistic variables and the links deterministic or probabilistic relationships between these variables. The graph is acyclic : it doesn't contains any loop. Given a set of conditional assertions about the variables and a set of probability associated with these variables, it is possible to produce a joint probability distribution for the variables. Then by incorporating the statistical methods into the network, the model is able to make the best decision based on the probabilities collected from the relationships between the variables.

Let's look at a simple example. Imagine we have the following set of assertions :

5. Bob and Bill go to work by using different means of transport : Bill uses his car, while Bob takes the train.
6. Bob rarely misses his train and the train is mostly on time, except on striking days. However, a strike doesn't necessary imply that Bob will be late. And a strike can also delay Bill since it can cause bottle-necks.
7. Moreover Bill is often late because he doesn't hear his waker. Thus a strike doesn't increase much the probability that he could be late.
8. In case of a strike, Bill is less likely to be late than Bob.

This set of assertions conducts to the following graph :



Because a strike may cause Bob's delay, we draw a relationship between « Strike » and « Bob is late ». In the case of Bill, either a strike or a waker problem can cause his delay but the probability of a waker problem is higher. Then if we associate weights with the different

relationships corresponding to the probability that it occurs, the network will be able to tell which one of Bob or Bill will be most likely to be late.

One of the reason why bayesian network are frequently used, besides their conviviality and their efficiency, is their versatility. Moreover, bayesian networks are able to learn and predict missing data from incomplete data sets. They perform well when combining background knowledge and data. They avoid overfitting the data when dealing with a computational complexity problem, and provide standard optimal decision making compared to other machine learning approches.

Nevertheless, target concept in Bayesian networks might not be representable.

Bayesian networks can be used to model the splice sites in a DNA sequence (Cai *et. al.*, 2000), predict protein secondary structure (Schmidler *et. al.*, 2000), find genealogical relationships between individuals, etc.

4.3.9 Genetic Algorithms

This approach was developed by Holland in 1975 and is based on biological evolution theory. Genetic algorithms mimic species evolution. First a population of individuals (that is the candidates solutions to the problem) is randomly created. The individuals are encoded as bit strings (made of "0" and "1") called *genotypes*. During each iteration, the population undergoes mutation (changes in a string) and recombination (crossover between two strings) processes to adapt the new environment. By favouring the survival of the most valuable individuals, hybrids appear that are fitter to the problem than their parents (e.g., better solutions to the problem). Thus the initial population gives birth to successive generations, that are closer and closer to the right solution. The process is repeated until the ideal solution to the problem is found.

The main structure of a genetic algorithm is the following :

1. Initialize time
2. Create an initial population (at random)
3. Evaluate how each individual adapts
4. While there is no satisfying solution and time is under time limit, do :
   - 4.1. Increment time
   - 4.2. Select parents
   - 4.3. Determine newborns' genes by recombination of parental genes
   - 4.4. Submit the population to crossover, mutation or inversion
   - 4.5. Evaluate how each individual adapts
   - 4.6. Select the survivals
5. Done

Genetic algorithms are simple to implement and are able to solve difficult high-dimensional problems. Moreover they can handle many different problems. That's why they are popular among bioinformatics researchers.

The only disadvantage of genetic algorithm is that its dynamics can be hard to understand during the process of evolution. Moreover, finding the solution can be slow and the speed of the process depends on the quality of the application.

In bioinformatics, genetic algorithms have been widely used in DNA fragment assembly (Parsons *et. al.*, 1995, Cedeno, & Vemuri, 1993; Fickett, J., & Cinkosky, M, 1993) and in multiple molecular sequence alignment (Zhang & Wong, 1997).

4.4 Machine Learning in Bioinformatics : Successes and challenges

As we will see in this part, Machine Learning methods have been succesfully applied in Bioinformatics but still need to improve.

4.4.1 Successful applications of Machine Learning methods in Bioinformatics

4.4.1.1 Identifying genes

One of the main challenge in the recent years was the sequencing of genomes, particularly the human genome. This consists in identifying the thousands of genes within each genome. Many projects (such as the Human Genome Project) and programs were then created in order to take up this huge challenge. A program based on machine learning called Glimmer is able to find almost every genes in a genome without any human intervention.

4.4.1.2 Predicting drug activity

While designing drugs with a particular biological activity or in order to understand the mechanisms governing the activity of known drugs, scientists need to know the relationships between the chemical structure of the drug and its activity. These relationships are most of the time impossible to analyse manually because of the 3D shapes involved. Inductive Logic Programming, because it directly focuses on the 2D and 3D structures of the drugs in addition to their physico-chemical properties, is renown for successfully discovering such relationships.

4.4.1.3 Gene expression analysis in cancer dignosis

Classification of patient samples is crucial in cancer diagnosis and treatment because the treatment of a cancer depends on its type. It has been suggested that the analysis of gene expression implied in cancer can greatly improve cancer classification and diagnosis. Neural trees and Bayesian networks have shown good results in gene expression analysis. Neural trees can automatically select appropriate genes during the learning process. Bayesian networks are able to capture probabilistic relationships among gene expressions that are understandable by humans. Thus both these techniques are used in gene expression analysis.

4.4.2 Challenges of Machine Learning in Bioinformatics

4.4.2.1 Learning in ‘dirty’ biological databases

The arriving of internet let the possibility to freely depose and exchange data to the public domain database. The control of errors and quality of those data is hard to do and can lead to redundancy. Some of those data are ‘dirty’ biological data, caused by experimental errors, wrong interpretation by biologists, human error during the annotation process, or non standardised techniques used in experiment. Most bioinformatics researchers have used the data without checking its relevancy.

Thus machine learning techniques must be robust to the data, especially when learning in a ‘dirty’ environment such as biological databases. They must also adapt their learning algorithm to avoid overfitting the data. Additionally the learning and process times must be short to handle the daily update of biological databases.

It is necessary to work along with biologists while interpreting and analysing biological data, since the databases need frequent updates in order to maintain the data quality. Thus the use of

machine learning methods to learn in biological databases has become one of the challenges for bioinformatics research.

4.4.2.2 Generative versus discriminative

Most of the hypotheses in biological research are based on experimental data, which have a high degree of uncertainty. Thus biological research has become a statistics dependant domain. A bioinformatics analysis performed without the use of probability theories will have a low degree of confidence.

In the previous section, I presented several machine learning methods, from generative methods (e.g. Hidden Markov Models, Bayesian Networks) to discriminative methods (e.g. Artificial Neural Networks, Genetic Algorithms). The type of machine learning being used depends on the learning goals of the application. It is necessary to choose the right technique in order to improve the quality of the hypotheses and thus to provide valuable knowledge.

4.4.2.3 Approximation versus explanation

Bioinformatics researchers have always wondered if learning methods should better provide information in every step or just the final result when solving a problem. Indeed some machine learning methods (Neural Networks, Genetic Algorithms, Hidden Markov Model) only outputs the result without any explanation concerning the learning process. That's why these methods are often called 'black box' algorithms. Although they are claimed to perform better than inductive methods, their approaches are hard to understand. Classical techniques such as decision trees give an explanation that is understandable by humans and that help to accomplish the task. The problem is still in abeyance.

4.4.2.4 Single versus multiple methods

Most of the techniques applied in bioinformatics use more than one machine learning methods. Although the use of several methods is more efficient than a single approach, combining all of them can be difficult. Moreover such an approach lacks coherence, because the learning processes and outputs of a method can differ from others.

However, one multiple approach has succeeded : the Hidden Neural Networks, which is a combination of Hidden Markov Models and Neural Networks. Nevertheless the development of multiple methods is still a challenge to take up in bioinformatics.

4.4.2.5 Too many algorithms ?

The arriving of machine learning methods in bioinformatics really improved the analysis of biological data. However, in order to fit the different data sets, many different algorithms have been created. And in fact there are too many. Bioinformatics researchers are now facing the difficulty to choose the algorithm which will fit at best their data sets.

As Baldi and Brunak state in their book « Bioinformatics : The Machine Learning Approach » : *"As a result, the need for computer / statistical / machine learning techniques is today stronger rather than weaker."*

5 Conclusion

Men have succeeded in building prodigious machines and algorithms in order to automate the manipulation and interpretation of data, thus making it fast, inexpensive and efficient. It was compulsory as regarding the huge amount of data. This shows the human intelligence and the capacity of people to create and adapt themselve to a perpetual changing environment.

However, we can wonder if we are not going too far. Although we are able to produce smart machines it seems that we are not learning faster or better. Because we are lazy by nature, we invent smart machines that do the work for us. One danger of this situation is that we become less skilled and more machine-dependant. As machines develop, they gradually replace our skills. Nowadays, it has become more important to know how to use an algorithm than to learn how to find it. Why indeed memorize data or formulas when machines can do it for us ? Will the day come when we don't need to learn anymore and totally rely on machines ? Recent years have shown that everything is possible : informatics, nanotechnology, artificial intelligence, neuroscience, genetics and bioinformatics, etc. Everything is going faster than we think. Moreover, machines have many advantages compared to humans : they are cheap, they are never ill, sad or depressed, they are fast and steady. Will machines overtaking us ?

Fortunately it seems that we still have time for that : machines still need men to improve indeed. Although they have the power, the speed and the stability, they are not yet able to think on their own. A supervisor must give inputs, check the process, eventually repare bugs and interpret the results. The time when a machine replaces a brain is still far from today. Well, let's hope…

6 Bibliography

Books :

« Psychology : The Science of Behavior », N. R. Carlson, W. Buskist, G. N. Martin.

Websites :

General Documentation :
- http://dmoz.org/Computers/Artificial_Intelligence/Machine_Learning
- http://dmoz.org/Computers/Artificial_Intelligence/Neural_Networks
- http://dmoz.org/Computers/Artificial_Intelligence/Genetic_Programming
- http://dmoz.org/Computers/Artificial_Intelligence/Belief_Networks
- http://crystal.biochem.queensu.ca/forsdyke/bioinfor.htm#Bioinformatics%20
- http://ihome.cuhk.edu.hk/~b400559/bioinformatics.html
- http://www.cnbc.cmu.edu/Research
- http://www.erc.caltech.edu/Research/proj_professors/learning.shtml
- http://dmoz.org/Computers/Artificial_Intelligence/Neural_Networks/People
- http://www.aic.nrl.navy.mil/~aha/research/machine-learning.html
- http://www.dsi.unifi.it/ai4bio/lecture-notes.html
- http://www.infobiogen.fr/liens/serv/tutorbioinfo.html
- http://www.infobiogen.fr/services/deambulum/fr/bioinfo_hist.html

Reinforcement Learning :
- http://www.cse.unsw.edu.au/~s2229705/rl/introduction.html
- http://www.cs.indiana.edu/~gasser/Salsa/rl.html
- http://www-anw.cs.umass.edu/~rich/book/1/node2.html

Cognition :
- http://www.rtis.com/nat/user/jfullerton/school/psyc345/program.htm
- http://www.scism.sbu.ac.uk/inmandw/review/ml/review/rev5964.html
- http://www.scism.sbu.ac.uk/inmandw/review/ml/review/rev7061.html
- http://www.scism.sbu.ac.uk/inmandw/review/ml/review/rev6756.html
- http://www.scism.sbu.ac.uk/inmandw/review/ml/review/rev6164.html
- http://www.scism.sbu.ac.uk/inmandw/review/ml/review/rev5815.html
- http://www.scism.sbu.ac.uk/inmandw/review/ml/review/rev4894.html
- http://www.scism.sbu.ac.uk/inmandw/review/ml/review/rev6542.html
- http://www.scism.sbu.ac.uk/inmandw/review/ml/review/rev8330.html
- http://www.scism.sbu.ac.uk/inmandw/review/ml/review/rev7497.html

Human Learning :
- http://www.erc.caltech.edu/Research/reports/barajas-full.html
- http://www.funderstanding.com/
- http://www.nap.edu/html/howpeople1/
- http://www.ericit.org/digests/EDO-IR-2002-05.shtml
- http://mieux.apprendre.free.fr/intel_multiples.html
- http://www.co-i-l.com/coil/knowledge-garden/cop/index.shtml

Human – Machines Interactions :
- http://www.princeton.edu/~alaink/PsyOrf322S02/PsyOrf322S02Lec2.htm
- http://www.fathom.com/feature/122414

- http://www.princeton.edu/~alaink/PsyOrf322S02/PsyOrf322S02L7PJL.htm

Machine learning :
- http://www.bioss.sari.ac.uk/student/newphddh.html
- http://www-csli.stanford.edu/icml2k/craft.html
- http://www.aic.nrl.navy.mil/~aha/research/machine-learning.html
- http://www.cs.iastate.edu/~honavar/Courses/cs673/machine-learning-courses.html
- http://www.soi.city.ac.uk/~eu790/presentations/MachineLearning
- http://www.brc.dcs.gla.ac.uk/~actan/presentations/myPresentation
- http://www.everything2.com/index.pl?node=machine%20learning
- http://osiris.sunderland.ac.uk/cbowww/AI/TEXTS/ML2/sect2.htm
- http://osiris.sunderland.ac.uk/cbowww/AI/TEXTS/ML2/sect3.htm
- http://osiris.sunderland.ac.uk/cbowww/AI/TEXTS/ML2/sect4.htm
- http://osiris.sunderland.ac.uk/cbowww/AI/TEXTS/ML2/sect5.htm
- http://www.infj.ulst.ac.uk/~cbbg23/papers/camda00_01.pdf
- http://cbit.snu.ac.kr/tutorial-2002/ppt/CBIT_ML_Tutorial.pdf
- http://www.brc.dcs.gla.ac.uk/~actan/papers/machine.pdf
- http://www.brc.dcs.gla.ac.uk/~actan/papers/preprintAPBC2003.pdf

Hidden Markov Models :
- http://www.cse.ucsc.edu/research/compbio/html_format_papers/hughkrogh96/cabios.html
- http://www.cs.mcgill.ca/~kaleigh/work/hmm/hmm_paper.html
- http://www.esil.univ-mrs.fr/~dgaut/Cours/hmm.html
- http://www.sfb363.uni-halle.de/kurs/hmm.html
- http://www.sfb363.uni-halle.de/kurs/hmm.html

Neural Networks :
- http://www.dsi.unifi.it/neural
- http://www.dsi.unifi.it/neural/portals.html
  http://dir.yahoo.com/Science/Engineering/Electrical_Engineering/Neural_Networks
- http://www.neuralmachines.com/axon/communication.html
- http://www.gisdevelopment.net/aars/acrs/1998/ps1/ps1012pf.htm
- http://hem.hj.se/~de96klda/NeuralNetworks.htm
- http://qbab.aber.ac.uk/roy/koho/kohonen.htm
- http://www.neuralmachines.com/axon/axon.html
- http://www.neuralmachines.com/axon/csm.html
- http://www.tradingsolutions.com/webhelp/concepts/concepts_neural.html
- ftp://ftp.sas.com/pub/neural/FAQ.html
- http://avalon.epm.ornl.gov/~touzetc/Book/Bouquin.htm#1.1
- www.barth.netliberte.org/ia/nn.html
- http://www.nd.com/welcome/whatisnn.htm

Bayesian Networks :
- http://www.cs.berkeley.edu/~murphyk/Bayes/bayes.html
- http://www-math.univ-mlv.fr/users/bouissou/siteRB/default.htm
- http://www.aaai.org (“ Bayesian Networks without Tears”, *Eugene Charniak*)

Genetic Programming :
- http://www.genetic-programming.org
- http://www.genetic-programming.com/gpanimatedtutorial.html
- http://www.pmsi.fr/gainit.htm

Knowledge Database Discovery :

- http://www.kdcentral.com/Resources/
- http://www.acm.org/crossroads/xrds5-2/kdd.html

Support Vector Machines, Kernel Machines :
- http://www.cs.rhul.ac.uk/~thore/research_pages/kernel_machines.htm
- http://www.kernel-machines.org
- http://research.microsoft.com/users/jplatt/svm.html

Bioinformatics :
- http://crystal.biochem.queensu.ca/forsdyke/bioinfor.htm
- http://inflamedgene.org/Bioinformatics/body_bioinformatics.html
- http://www.infobiogen.fr/services/deambulum/fr/bioinfo_hist.html

Artificial Intelligence :
- http://sampletalk.8m.com/
- http://www.compapp.dcu.ie/~tonyv/Textbook/history.html
- http://www.cs.nott.ac.uk/~gxk/courses/g5aiai/002history/history.htm