**Biological Data:** [**https://www.geeksforgeeks.org/data-mining-time-series-symbolic-and-biological-sequences-data/**](https://www.geeksforgeeks.org/data-mining-time-series-symbolic-and-biological-sequences-data/)

They are made of DNA and protein sequences. They are very long and complicated but have some hidden meaning. These types of data are used for the sequence of nucleotides or amino acids. These analyses are used for aligning, indexes, analyze biological sequence and play a crucial role in bioinformatics and modern biology. Substitution trees are used to find the probabilities of amino acids and probabilities of intersections. BLAST-Basic Local Alignment Search Tool is the most effective tool for biological sequence.

* **Introduction Sequence data:**

Data mining refers to extracting or mining knowledge from large amounts of data. In other words, [Data mining](https://www.geeksforgeeks.org/data-mining/) is the science, art, and technology of discovering large and complex bodies of data in order to discover useful patterns. Theoreticians and practitioners are continually seeking improved techniques to make the process more efficient, cost-effective, and accurate.

<https://www.ncbi.nlm.nih.gov/books/NBK550342/>

“When comparing two biological sequences, an alignment is generated to view differences between the sequences at each position.”

Sequence analysis is incomplete without discussing next-generation sequencing (NGS) data. Deep sequencing is highly important due to its ability to address an increasingly diverse range of biological problems such as the ones encountered in therapeutics. A complete NGS workflow to generate a consensus sequence and haplotypes is discussed.

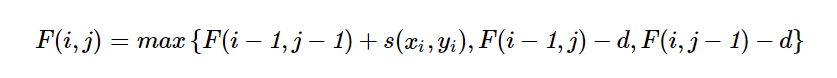
* PAIRWISE ALIGNMENT AND DYNAMIC PROGRAMMING

Pairwise alignment involves comparing two sequences against each other and finding the best possible alignment between them. The process involves scoring at each position for match, mismatch, and indels. Since matches are preferred over deletions, matches are normally assigned the highest scores, and lowest for insertions. Similarity between two sequences is inversely proportional to the number of mismatches and indels in their alignment. Although the scoring for alignment can be as simple as +1 for match, 0 for mismatch, and −2 for insertion, different scoring models have been developed based on the statistically relevant frequencies of one amino acid changing into another.

**Related Work:**

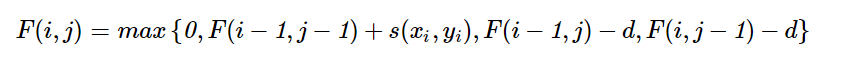
* **Traditional approaches**

**Needleman–Wunsch algorithm** Initially developed by Needleman and Wunsch in 1970, the algorithm is based on dynamic programming and allows for global or end-to-end alignment of two sequences (8). The algorithm involves three main steps, namely initialization, calculation, and trace back. A matrix of dimensions i, j is initialized, where i and j are the length of two sequences under comparison. In the second step, F(i, j ) highest score for each comparison at each position is calculated,



where “*s(xi, yi)*” is the match/mismatch score and “*d*” is the penalty for deletion.

**Smith–Waterman** algorithm Initially proposed by Smith and Waterman in 1981, the algorithm allows for local sequence alignment and is like the Needleman–Wunsch algorithm (10). Local sequence alignment can be used in situations where it is required to align smaller subsequences of two sequences. In the biological context, such a situation may arise while searching for a domain or motif within larger sequences. The algorithm comprises of the same steps as Needleman–Wunsch; however, there are two main differences. Computation of max score also includes an option of 0:



Assignment of “0” as max score corresponds to starting a new alignment. It allows for alignments to end anywhere within the matrix. The trace back therefore starts from the highest value of *F(i, j)* in the matrix and ends where it encounters 0.

* **Problems with alignment methods based on dynamic programming**

The local alignment methods based on dynamic programming are quite accurate and guarantee to find an optimally scored alignment, they are slow and not practical for sequence alignments against databases with millions of sequences. The time complexity of dynamic programming algorithms is O(mn), that is, the product of sequence lengths.

In the initial attempts to improve the speed for sequence comparisons, heuristic algorithms like BLAST ([11](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), BLAT ([12](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), and FASTA ([13](https://www.ncbi.nlm.nih.gov/books/NBK550342/), [14](https://www.ncbi.nlm.nih.gov/books/NBK550342/)) were created. Further advancements in the efficiency of similarity search algorithms came with algorithms like LSCluster ([15](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), Usearch ([16](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), Vsearch ([17](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), Diamond ([18](https://www.ncbi.nlm.nih.gov/books/NBK550342/)) and Ghostx ([19](https://www.ncbi.nlm.nih.gov/books/NBK550342/)). In general, these algorithms search for exact matches and extend the alignment from those matches trying to estimate the optimal scoring alignment.

**Basic Local Alignment Search Tool**, initially developed by Altschul and colleagues ([11](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), is based on the idea that the best scoring sequence alignment would contain the highest number of identical matches or highly scoring sub-alignments. The algorithm carries out the following steps: (i) reduce the query sequence into small subsequences called seeds, (ii) search these seeds across the database for exact matches, and (iii) extend the exact matches into an un-gapped alignment until a maximal scoring extension is reached. The use of seeds to first search for exact matches greatly increases the whole searching process and the un-gapped alignment misses only a small set of significant matches. The accuracy and sensitivity of BLAST made it amongst the most widely used search algorithm in the biological world. A variant of BLAST named Position-Specific-Iterative BLAST (PSI-BLAST) extends the basic BLAST algorithm ([20](https://www.ncbi.nlm.nih.gov/books/NBK550342/)). PSI-BLAST performs multiple iterations of BLAST and uses the hits found in one iteration as a query for the next iteration. Although slower due to sheer amount of calculations required, PSI-BLAST is considered a reliable tool to find distant homology relationships. Although BLAST and PSI-BLAST are extensively used, recently developed methods offer results with higher accuracy and sensitivity.

**Hidden Markov models (HMM)** have been used efficiently for numerous applications to understand and explore biological data. One such example is HMM–HMM-based lightning fast sequence search (HHblits) introduced in 2012 (21). The tool can be used as an alternative for BLAST and PSI-BLAST and is 50 to 100 times more sensitive. The high sensitivity of the tool can be attributed to the algorithm which relies on comparing the HMM representations of the sequences. Although profile–profile or HMM–HMM alignments are very slow to compute, the prefilter in HHblits reduces the required alignments from millions to thousands, thus giving it a considerable speed advantage. HHblits represents each sequence in the database as a profile HMM. Prefiltering reduces the number of HMM comparisons for similarity search by selecting only those target sequences where the largest un-gapped alignment exists, and a Smith–Waterman based alignment reveals a significant E-value

* **MACHINE LEARNING AND SEQUENCE ANALYSIS**

Biological data provide amongst the perfect use cases of machine learning and artificial intelligence algorithms. This is the reason that researchers in the field of bioinformatics and computational biology have used statistical analysis and inference since the very beginning. **Techniques like maximum likelihood (22) and neighbor joining (23) have been used for comparative genomics. Naïve Bayes and Markov chains have been extensively used for sequence analysis**. **Logistic regressions, support vector machines, and random forests have been used in numerous applications ranging from prediction of protein sequence or structural elements to classification of proteins into different structural and functional classes**. With the development of deep neural networks, we observe an increase in the use of the algorithms like long short-term memory (LSTM) (24) and convolutional neural networks (CNN or ConvNet) (25) to predict the different features and behavior of proteins, for example, protein contact prediction and prediction of post-translational modifications.

Machine learning methods are broadly divided into two types, supervised and un-supervised learning. Based on the inherent features of the data, if it is not labeled and cannot be assigned to any type, then classification is done using unsupervised learning. For instance, the classification of proteins into different groups is done based on their sequence similarity to each other. K-means clustering algorithm (26) and Markov clustering (27) can be used in unsupervised classification. On the other hand, if the data are labeled into different sets, this information can be used to train the computer by showing it positive and negative examples. Once the training is complete, the accuracy of training can be tested using similar data not used in the training dataset. Any classification technique following training and testing procedures using labeled data is termed supervised machine learning. Examples for this type of learning include SVM, HMM, random forest, and CNN.

* **Hidden Markov Models HMM**

is a statistical method that can be used to predict the probability of occurrence for a future event

HMMs provide the foundations for a range of complex models that can be used for multiple sequence alignment, profile searches or detection of sequence elements.

In order to understand the HMMs and their use in biological data, consider the example of binding site recognition on a DNA sequence. There is an observable sequence of nucleotides which in the right order hides underneath a binding site. We can observe the nucleotide sequence, but the presence or absence of a binding site remains hidden to us.

HMMs are particularly suited for such problems because they use observed frequencies to calculate emission and transition probabilities to decipher the hidden states. An HMM involves two types of probabilities, transition and emission probabilities. The probability of moving from one state to another is called the transition probability. The probability to observe a variable within a state is called emission or output probability.

Figure 2 shows a schematic HMM with basic architecture and elements. HMMs have been used not only to create sequence profiles but also to create probabilistic model representation of protein clusters. Pfam is an example database that clusters proteins based on their functional elements and represents them with HMM.

**The downside to HMMs** is that they assume a future event depends only on the event that happened immediately before and not in the distant past.

This **creates a limitation to use** standard HMMs in complex cases **where sequence elements influence each other** that may be close in the three-dimensional space **but sequentially lie far from each other**.

Outside of the biological world, one such example is autocompleted or word suggestions. The words appearing in suggestion are directly dependent on the word that appeared immediately before the present suggestion.

* **Neural networks**

Artificial neural network is another classification technique with numerous applications in computational biology. Neuron is the basic unit of an artificial neural network. Each neuron can have multiple input connections with weights assigned to each of them. The output value from the neurons is calculated according to its activation function. A neural network may consist of multiple layers, with each layer containing multiple neuron

This approach uses labelled data and follows the main steps listed below:

* Dataset: Divide the data into training sets and testing set (mostly 70–30% split or 60–40% split, respectively).
* Training: Use the training data to traverse over the neuron and estimate the output.
* Iterate: Based on the difference between the actual and estimated output, calculate the error and adjust the weights accordingly. Repeat step 2.
* Testing: After multiple iterations between step 2 and 3, the model is trained and can be tested. Use the test set (unseen data for model) to compute the output. As the actual label is known, the accuracy and sensitivity can be calculated based on the correct (true positives or true negatives) and incorrect classifications (false positives or false negative).
* Validation: The training- and test-set splits are randomized, and new sets are created from the existing dataset. This new test-train split is then used again iterating oversteps 2–4. The idea is to create a model independent for generalized datasets. Depending on situations, there can be multiple iterations for this step and hence referred to k-fold cross validation.
* **limitations and new scope**

The performance of machine learning in general and neural networks in particular depends highly on the quality of the data.

A high-quality data would have low noise/junk while having a high homogeneity.

Noise in biological data can refer to ambiguous sequence elements or incorrect labels.

A high homogeneity results in an equal distribution of diversity in data across different data splits.

Assuring the good quality of data before model training is a very important and time-consuming step for data scientists.

If the training dataset is not a homogenous representative of the population, it can lead to a biased classification in the models.

A bias model can show promising results for the testing dataset but fails in the actual world.

This happens because the model is trained to classify only those types of cases that it observed during the training, and a bias sample resulted in a skewed perception of the real-world scenario.

The quality of classification from neural networks also depends highly on the training iterations and size of datasets.

While the ability for high-powered computation has greatly increased in the last decade, coupled with biological big data, neural networks can be used to train accurate classifiers.

Neural networks have now evolved into their more complex form called “Dense Networks” or “Deep learning.” These networks (e.g., LSTM) comprise numerous neurons and high number of hidden layers between the input and output layers (hence deep network). Although the depth of a network results in a better-quality model, they are difficult to train due to the requirement of high computing power.

* **NEXT-GENERATION SEQUENCING**

The last three decades have seen a continuous evolution of sequencing technologies.

Starting from traditional Sanger sequencing to whole genome shot gun sequencing by Craig Venter and later next-generation sequencing (NGS) (4). The latest amongst these is the “Nanopore,” highly compact and efficient sequencing that connects to a computer via USB; it is easily transportable and fits on a small desktop.

The technology that initially required thousands of dollars per nucleotide is much cheaper now.

An NGS pipeline comprises of two main sections:

a wet lab section involves sample preparation, amplification, and sequencing; and the second section involves a bioinformatics workflow that uses the data generated by the wet lab to derive a sequence and other information.

It is important to note that the bioinformatics section involves sequence analysis algorithms that are based on statistical and heuristic techniques to analyze and generate sequences.

This section focuses on the bioinformatics aspect of NGS since it has evolved an ecosystem of computational algorithms and pipelines around it for accurate and efficient sequencing.

NGS is a massively parallel sequencing technology, also **referred as high-throughput sequencing**, that **allows analysis of large fragments of DNA and RNA genomes with high sensitivity**, much more quickly and cheaply than the previously used Sanger sequencing methodology. In NGS, different platform technologies follow the same eight major steps.

1. *Library preparation*: The first step in NGS workflow involves preparation of high-quality and high-yield sequence library. The isolated genomic DNA or RNA is sheared into smaller fragments ranging from 150–5000 base pairs (bp) depending on the sequencing platform. The desired library can be created using either of the two fragmentation approaches, mechanical shearing or enzyme-based fragmentation ([28](https://www.ncbi.nlm.nih.gov/books/NBK550342/), [29](https://www.ncbi.nlm.nih.gov/books/NBK550342/)). Mechanical shearing methods include acoustic shearing, needle-shear, sonication, and nebulization, whereas enzyme-based methods involve transposons and restriction enzymes (endonucleases) ([30](https://www.ncbi.nlm.nih.gov/books/NBK550342/)). The small fragments known as reads have short overhangs (sticky ends) of 5’-phosphate and 3’-hydroxl groups. These ends are repaired by adenylation at 3’ ends resulting in adapter ligation that is important for amplification. During library preparation, unique barcodes can be added to the fragments facilitating multiple sequencing of various samples in the same run ([31](https://www.ncbi.nlm.nih.gov/books/NBK550342/)).
2. *Amplification*: The goal of this step is to create thousands of copies for each read. The library is loaded onto the flow-cell, and the fragments are amplified using clonal amplification methods such as emulsion PCR or bridge amplification. In emulsion PCR, the library is amplified within a tiny water droplet floating in an oil solution ([32](https://www.ncbi.nlm.nih.gov/books/NBK550342/), [33](https://www.ncbi.nlm.nih.gov/books/NBK550342/)). In bridge amplification, the single-stranded DNA from the library is hybridized to the flow-cell’s surface-bound forward and reverse oligos that are complementary to the library adapter sequences. Hybridized at one end, the singe-stranded DNA then folds over to form a bridge and binds to adapter-complementary oligos at the other end. DNA polymerase adds nucleotides to amplify DNA, and a clonal cluster is generated as the original strand is washed away leaving complementary strands of amplified DNA attached to the flow cell. ([34](https://www.ncbi.nlm.nih.gov/books/NBK550342/)).
3. *Sequencing*: The amplified individual sequences are sequenced using different platforms and sequencing technologies that include Illumina (Solexa) sequencing, Roche 454 sequencing, and Ion Torrent (Proton/PGM sequencing). Illumina (Solexa) sequencing works by simultaneously identifying DNA bases (A, T, C or G), and each base emits a unique fluorescent signal as it is added to the nucleic acid chain. Illumina sequencing involves 100–150 bp read length. Illumina has some variations that mainly differ in the amount of DNA sequenced in one run ([Table 1](https://www.ncbi.nlm.nih.gov/books/NBK550342/table/Ch4-t0001/?report=objectonly)). Roche 454 sequencing is based on pyrosequencing; a technique that detects pyrophosphate release, again using fluorescence, after nucleotides are incorporated by polymerase to a new strand of DNA. Roche 454 sequencing produces sequence reads of up to 1000 bp in length. Like Illumina, it does this by sequencing multiple reads at once by reading optical signals as bases are added. Ion Torrent (Proton / PGM sequencing) measures the direct release of H+ (protons) from the incorporation of individual bases by DNA polymerase and therefore differs from the previous two methods as it does not measure light. As in other kinds of NGS, the input DNA or RNA is fragmented, this time ~200 bp. These sequencing technologies result in raw sequencing reads (20 to 1000 bp) stored in the FASTQ format which contains both the nucleotide sequence and its corresponding quality scores. These reads can be either “single-ended” or “paired-ended.” Paired-end reads are produced when the fragment size used in the sequencing process is much longer (typically 250–500 bp long).
4. *Quality control and read filtration*: After sequencing is complete, the read data are in electronic form and can be processed to generate a whole genome or a specific gene sequence using a bioinformatics NGS pipeline. Although quality control and filtration is the fourth step in generating a full analyzable sequence, it is the first step in a bioinformatics NGS pipeline. Read filtration involves removing low confidence and erroneous reads from the dataset. The amplified raw reads pass through quality control check using FastQC ([35](https://www.ncbi.nlm.nih.gov/books/NBK550342/)) that can produce a detailed report on the number, quality, and coverage of reads. These methods mostly work on sequence analysis techniques like clustering short reads to calculate their frequency and quality scores. It is followed by read filtration, clipping of adapters and low-quality base pairs from 3’ and 5’ ends using software such as CutAdapt ([36](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), trimmomatic ([37](https://www.ncbi.nlm.nih.gov/books/NBK550342/)) and others.
5. *Alignment*: Once the read quality is acceptable, millions of raw sequence reads (single-end or paired-end) are mapped and aligned using either a reference based assembly (in which reference sequence is available) or de novo assembly (in the absence of a reference sequence). The sequence reads of variable lengths are aligned using different bioinformatics alignment tools such as BWA ([38](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), Bowtie ([39](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), and TopHat ([40](https://www.ncbi.nlm.nih.gov/books/NBK550342/)). These heuristic-based aligners allow fast sequence alignment and generate a consensus sequence from the alignment by searching the overlapping portions of the reads and merging them into longer reads in order to construct a region of interest, that is, genes or a whole genome. The main aim of this step is to generate a consensus sequence from the millions of reads. A consensus sequence shows the genetic makeup at the time of the sample collection. This step marks the completion of sequence generation for a partial or a whole genome. The following steps are important for an in-depth analysis beyond generation of only a single sequence.
6. *Variant identification*: NGS is not only time efficient but also provides the data for an in-depth sequence analysis. Variant analysis uses the reads file to determine the conserved and variable nucleotides at specific positions. As this process involves statistical calculations spanning over millions of reads, it is both a time and computationally intensive process. Bootstrap resampling of reads can be used to assess the quality of variant calling scores. The variations within the genomic sequences such as single-nucleotide polymorphisms (SNPs), single-nucleotide variants (SNV), and indels (insertions and deletions) are detected using software such as SAMtools ([41](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), Genome Analysis Toolkit (GATK) ([42](https://www.ncbi.nlm.nih.gov/books/NBK550342/)), and VarScan ([43](https://www.ncbi.nlm.nih.gov/books/NBK550342/), [44](https://www.ncbi.nlm.nih.gov/books/NBK550342/)). Both SAMtools and GATK use the Bayesian probabilistic approach to identify true variants from alignment errors, whereas VarScan uses a heuristic approach. Most NGS methods for SNV detection are designed to detect germline variations in an individual’s genome, whereas the variations that are identified within a population are referred as SNPs.
7. *Annotation*: The genetic variants detected are annotated based on the published peer-reviewed literature and public genetic variant databases.
8. *Interpretation of variants*: Lastly, medical professionals will interpret these variants and obtain the patient’s clinical history in order to establish a most accurate diagnosis. This includes examining different disease pathways and gene network analysis and identifying actual mutations causing a disease.

* **CONCLUSION**

Sequence analysis is a broad area of research with sub-domains. Alignment of sequences can reveal important information concerning the structural and functional sites within sequences. It is used to explore the evolutionary path of sequences by identifying the sequence orthologs and homologs. Sequence analysis also involves the use of machine learning techniques for classification and prediction of sequence elements. Statistical methods are used to create sequence profiles and identify other distantly related sequences with a higher precision. Advancement of sequencing technologies has resulted in a next-generation era that opened the doors to personalized medicine and haplotype/quasi-species detection. With correctly organized NGS pipelines, it is possible to analyze the effects of drugs directly at the sequence level.

<https://www.ncbi.nlm.nih.gov/books/NBK550342/>

<https://www.frontiersin.org/articles/10.3389/fbioe.2020.01032/full>

* **DNA Sequence Coding**

When processing the DNA sequence, it is necessary to convert the string sequence into a numerical value, so as to form a matrix input model training.

there are three methods for sequence encoding:

sequential encoding,

one-hot encoding,

and k-mer encoding (Choong and Lee, 2017).

The characteristics of the three DNA encoding methods are shown in Table 1.

The performance of sequential encoding is comparable to one-hot encoding, but the training time is significantly reduced.

One-hot encoding is widely used in deep learning methods and is very suitable for algorithms such as CNN (convolutional neural networks).

In addition, the performance of one-hot encoding is quite consistent in different data sets, but a suitable CNN is required to get good performance. Ordinal codes represented by matrices perform best in some evaluation data sets. The performance of CNN in discovering DNA motifs depends on the proper design of sequence encoding and representation.

The good performance of the ordinal coding method shows that there is still room for improvement in the single-point coding method.

* **DNA Sequence Similarity**

The main mining modes of machine learning include data characterization and differentiation, data frequent patterns, association and correlation, classification and regression of data predictive analysis, cluster analysis, and outlier analysis.

Data mining for DNA sequences is generally carried out from these aspects, and research in these areas is inseparable from similarity analysis between sequences (Pearson, 2013). It can be seen that sequence similarity is the basis of DNA sequence data mining.

**Sequence similarity** means that there are similar or identical sites between sequences.

The sequence similarity can be a quantitative value or a qualitative description.

If the degree of similarity between two sequences exceeds 30%, It is considered that the two sequences have a homologous relationship. Therefore, if the two sequences are highly similar, the two sequences are likely to have a common evolutionary ancestor. At the same time, if a sequence similar to the unknown sequence can be found from the sequences with known functions, we can further predict the function (Rogozin et al., 1996) of the unknown sequence.

**One of the main problems of DNA sequence similarity** research is to search for sequences whose similarity to a specified sequence exceeds a certain threshold.

The most commonly used method is to establish a similarity matrix (Henikoff and Henikoff, 1992) and find the best match between sequences in consideration of possible insertions, deletions, and mutations.

The study of sequence similarity is divided into global similarity research and local similarity research.

The global similarity is the similarity matching of the entire sequence, which is suitable for sequences with a high degree of similarity at the global level. The Needleman-Wunsch algorithm is a typical sequence alignment algorithm (Pearson and Lipman, 1988).

However, genes only account for about 2% of the DNA sequence, that is, only a few sequence fragments have a functional role. Although there is no similarity between sequences as a whole, there are similarities in some local areas. Therefore, it is more meaningful to study local similarity than global similarity. Typical local alignment algorithms include the Smith-Waterman algorithm based on dynamic programming algorithm and heuristic database similarity search algorithms FASTA and BLAST (basic local alignment search tool).

In a recent study, Delibas and Arslan (2020) proposed a non-aligned sequence similarity analysis method, a new method of DNA sequence similarity analysis using the similarity calculation of texture images, which is a digital image processing method.

Sequence similarity is one of the key processes of DNA sequence analysis in computational biology and bioinformatics.

**In the study of gene function analysis, protein structure prediction and sequence retrieval, similarity calculations are required**.

We select the appropriate sequence similarity analysis method and improve it according to actual application requirements and biological background. This is the basis and key of DNA sequence data mining.

* **Machine Learning Algorithm**

In the past few decades, we have witnessed the revolutionary development of biomedical research and biotechnology and the explosive growth of biomedical data.

The problem has changed from the accumulation of biomedical data to **how to mine useful knowledge from the data**.

On the one hand, the rapid development of biotechnology and biological data analysis methods has led to the emergence of a challenging new field: bioinformatics.

On the other hand, the continuous development of biological data mining technology has produced a large number of effective and well-scalable algorithms.

**How to build a bridge between the two fields** of machine learning and bioinformatics to successfully analyze biomedical data is worthy of attention and research.

In particular, we should analyze how to use data mining for effective biomedical data analysis, and outline some research questions that may stimulate the further development of powerful biological machine learning algorithms.

* **Basic Process of Data Mining**

Data mining is a discipline that combines classic statistical tools with computer science algorithms.

This discipline aims to mine knowledge from large amounts of data for scientific, computational, or industrial use.

Diagram

Description automatically generated

1. Data cleaning. Because of the increasing amount of heterogeneous data, data sets often have missing data and inconsistent data. Low data quality will have a serious negative impact on the information extraction process. Therefore, deleting incomplete, or inconsistent data is the first step in data mining.

2. Data integration. If the source of the data to be studied is different, it must be aggregated consistently.

3. Data selection. Accurately select relevant data based on the research content.

4. Data conversion. Transform or merge data into a form suitable for mining and integrate new attributes or functions useful for the data mining process.

5. Data mining. Select the appropriate model according to the problem and make subsequent improvements.

6. Mode evaluation. After acquiring knowledge from the data, select appropriate indicators to evaluate the model.

The main task of the data mining step is to correctly select one or a combination of these steps and find an effective and reliable method to solve the given problem. In recent years, machine learning has been widely used in bioinformatics analysis.

Each step of data mining is developed independently of other steps, and each step has a large number of machine learning algorithms.

* **Association Rule Mining Algorithm**

As one of the most important branches of data mining, association rule mining can identify the associations and frequent patterns of a set of items in a given database.

It consists of two sub-problems:

(1) Set the minimum support threshold and use the minimum support Find frequent itemsets from the database

(2) Use minimum confidence to find association rules that satisfy specified constraints on frequent itemsets.

Association rule mining not only plays an important role in business data analysis but has also been successful in many other fields, such as virtual shopping basket analysis and medical data analysis.

**The Apriori algorithm** is a typical association rule-based mining algorithm, which has applications in sequence pattern mining and protein structure prediction.

Many machine learning algorithms in data mining are derived based on Apriori (Zhang et al., 2014).

The basic method of association rule mining is through the use of Some metrics are used to analyze the strong associations in the database.

The most commonly used measurement methods are minimum support and minimum confidence. The Apriori algorithm uses a guided method to mine association rules between data items in the database.

* **Classification Algorithm**

Classification is one of the most studied tasks in machine learning. The principle of classification is based on the predicted attribute to predict the class of the target attribute specified by the user.

In genomics, the key issues are genome classification and sequence annotation.

In the mining of biological sequences, widely used algorithms include fuzzy sets, neural networks, genetic algorithms, and rough sets. There are also many general classification models, such as naive Bayesian networks, decision trees, neural networks, and rule learning using evolutionary algorithms.

* **Clustering Algorithm**

The clustering algorithm in machine learning can cluster together sequences with some same characteristics and explore the effective information of unknown sequences from known functions and structures.

Therefore, the clustering of biological sequences is of great significance to the research of bioinformatics.

**The difference** from the classification is that clustering does not implement a set category. Each cluster has its own common characteristics. **The purpose of** cluster analysis is to divide the data with common characteristics into one category, then use other methods to analyze the data.

In recent years, with the development of artificial intelligence, the clustering algorithm has become a popular research direction in the field of machine learning. To improve the processing capacity of large scale data, domestic and foreign scholars have conducted more in-depth research on clustering algorithms.

Several excellent clustering algorithms have emerged: there are mainly clustering algorithms based on granularity, clustering algorithms based on uncertainty, clustering algorithms based on entropy, clustering integration algorithms, etc. Besides the above-mentioned ones, there are a large number of algorithms. Each algorithm has its characteristics, an algorithm cannot be applied in all situations. Understand the advantages and disadvantages of each algorithm could help us better use and research.

* **DNA Sequence Alignment**

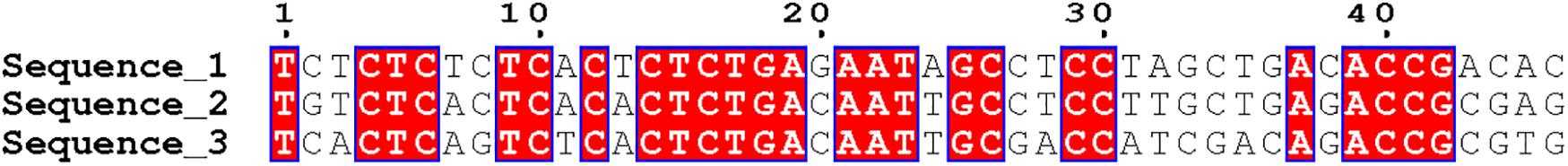
Sequence alignment can be divided into double sequence alignment and multi-sequence alignment.

Multi-sequence alignment is an extension of double sequence alignment. As the number of sequence alignments increases, the difficulty of alignment is also greater.

At present, the research of biological sequence alignment is very mature, and a large number of sequence alignment tools have appeared, such as CLUSTAL, TCOFFEE, and MUSCLE.

We selected three DNA sequences of equal length and used CLUSTAL software for sequence comparison.

The local visualization of the comparison results is shown in Figure



The red area indicates the part of the three sequences that are completely matched. The number of completely matched bases in the figure is 25. The number of bases in the sequence fragment is 46. The sequence similarity reaches 54.35%, and it can be considered that the three sequences have local similarities

* **DNA Sequence Classification**

Classification is an important mining task in machine learning. Its purpose is to learn a classification model from the training sample set **to predict the category** of unknown new samples.

The classification of biological sequences as a special data type is a popular problem in data mining.

It is a difficult problem, due to the non-numerical attributes of the biological sequence elements, the sequence relationship between the sequence elements, and the different sequence lengths of different events, etc. **Sequence classification is to predict the type of DNA sequence based on the similarity of its structure or function,** **and then predict the sequence function and the relationship between other sequences and assist in the identification of genes in DNA molecules.**

**traditional approaches within classification**

1)

Levy and Stormo (1997) proposed to use circular graphs (DAWGs) to classify DNA sequences. Müller and Koonin (2003) proposed to use vector space to classify DNA sequences. Ranawana and Palade (2005) proposed a multi-classifier system for identifying E. coli promoter sequences in DNA sequences. He Uses four different coding methods to encode the sequence and then uses the coding sequence to train four different neural networks. The classification results of the four individual neural networks were then combined through an aggregation function, which used a variation of the logarithmic opinion pool method. Experiments show that when the same data is provided to the neural network with different encoding methods, it can provide slightly different results that can be provided.

At the same time, when the results of more classifiers with the same input data are integrated into a multi-classifier, the results we can obtain are better than the single performance of the neural network.

However, the main disadvantage of the neural network design is that it is **difficult to obtain the optimal parameters of the neural network.** This will involve the deployment of the neural network and the optimization of the encoding method used.

2)

Ma et al. (2001) proposed a DNA sequence classification based on the combination of the expectation-maximization algorithm and a neural network and applied the algorithm to identify the DNA sequence classification of E. coli promoters. Ma Q uses an improved expectation-maximization algorithm to locate the −35 and −10 binding sites in the E. coli promoter sequence. It is no longer assumed that the lengths of the spacers between the binding sites and between the binding sites and the transcription start site are evenly distributed. Instead, he derives the probability distribution of these lengths. According to the information contained in each E. coli promoter sequence, he selects features and uses orthogonal coding methods to represent these features. Finally, these features are input into the neural network for promoter recognition. This method obtained good performance on different data sets.

3)

Zaki et al. (2010) proposed a variable-order hidden Markov model with the continuous state: VOGUE. VOGUE uses a variable sequence mining method to extract frequent patterns with different lengths and spacings between elements, and then he constructs a variable sequence hidden Markov model. Compared with traditional HMM, VOGUE has higher classification accuracy. However, the frequency statistical characteristics of the sub-sequences in the sequence are not considered, which affects the generalization ability of the model.

<https://www.frontiersin.org/articles/10.3389/fbioe.2020.01032/full#B25>

“entropy is a measure of the uncertainty associated with a probabilistic experiment.” To generate a probabilistic experiment, we use a known method in bioinformatics, the k-mer

In this method, each sequence is mapped in the frequency of neighbouring bases k, generating statistical information”

Python Package

* 1. SeqFeaL
  2. <https://github.com/Darisia/SASCRiP>
  3. <https://academic.oup.com/bioinformatics/article/34/14/2499/4924718>
  4. <http://www.alok-ai-lab.com/materials/PyFeat_Bioinformatics.pdf>