

*Notes on Bioinformatics*  
*Riccardo Lo Iacono*

---

December 1, 2025

Document is WIP, typos could be found

**Contents.**

<b>1</b>	<b>Introduction to Bioinformatics</b>	<b>1</b>
1.1	A brief introduction to molecular biology . . . . .	1
<b>2</b>	<b>Pair-wise alignment</b>	<b>5</b>
2.1	Alignment and similarity measures	5
2.2	Needleman-Wunsh algorithm . . .	6
2.3	Smith-Waterman algorithm . . . .	7
<b>3</b>	<b>Multiple alignment</b>	<b>8</b>
3.1	FASTA . . . . .	8
<b>References</b>		<b>9</b>
<b>Acronyms</b>		<b>10</b>



## Section 1 – Introduction to Bioinformatics

---

### - 1 - Introduction to Bioinformatics.

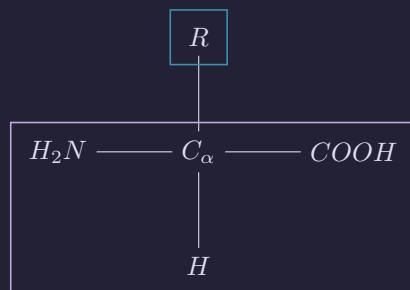
When we talk about bioinformatics, we refer to some sort of bridge between life sciences and informatics. The reason this kind of relationship is needed is due to the fact that biology, with Deoxyribonucleic Acid (DNA) sequencing, etc, is a source of big data; additionally, biology can benefit from informatics to ease some of its tasks, i.e., protein sequencing. In this notes we focus on various algorithms to manage both DNA and protein sequencing. Before doing so, in the remainder of this section we provide to the reader some notions of molecular biology; that is, we overview what aminoacids are, how these are related to proteins, the structure of DNA and RNAs and how these lead to proteins synthetis.

#### - 1.1 - A brief introduction to molecular biology.

Human bodies, as well as any other living thing, is constituted of several proteins which differ for their purpose. In this sense, we distinguish:

- **Structural proteins:** serves a building block for cells. An example of these is collagen.
- **Enzymes:** act as “catalysts” for some chemical reaction. Some reactions are so slow that, in the case enzymes did not exist, life itself would not exist.
- **Transport proteins:** carry vital substances through the body, i.e., hemoglobin.
- **Antibodies.**

Considering proteins in general, meaning without caring what's their task, these end up being a chain of smaller molecules: the *amino acids*. With reference the figure below, all amino acids share a common part



(the one boxed in ■) and differ for the corresponding side chain (a.k.a the R-group, the one boxed in ■). Though one could argue the existence of

## Section 1 – Introduction to Bioinformatics

---

exceptions, we consider the classic 20 side chains, and consequentially the 20 classic amino acids (See Appendix A for the whole list).

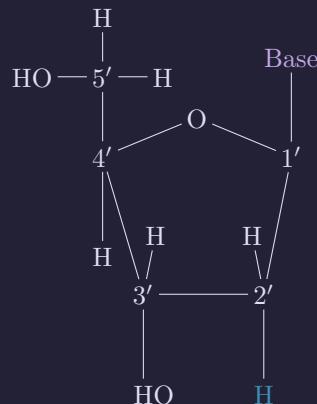
In truth proteins are not chains of amino acids, rather their residue: within a protein, amino acids are bound to each other by peptide bonds, in which the COOH of an amino acid  $A_i$  binds with the  $H_2N$  of the amino acid  $A_{i+1}$ . This bond leads to the release of a water molecule, thus neither of the amino acids has its original molecular structure.

But how do we get proteins? To answer this question we need to understand what DNA and RNA are.

### - 1.1.1 - From nucleic acids to proteins.

In nature there exist two types of nucleic acids: DNA and RNA. We present each one briefly.

Starting with DNA, as for proteins, this is a chain of simpler molecules: the nucleotides. Structurally, it presents as a double strand chain in an helix. Chemically speaking (see the figure below), DNA it's a repetition of very similar units.



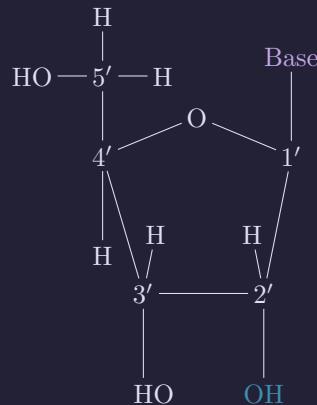
We distinguish four distinct bases: Adenine (A), Guanine (G), Cytosine (C) and Thymine (T). In DNA adenine bonds with thymine while guanine bonds with cytosine.

Each DNA strand, though bound in pair, follows an orientation: either from 5' to 3' or viceversa. In the following we use the former. This mechanism is what allows DNA replication. Essentially, since hydrogen bonds are not covalent (strong) they can be easily broken; in addition, since each nucleotide bonds only with a different one, from a single DNA strand we can get a copy of it. More precisely, given a DNA and a *primer*, that is a segment of DNA that triggers the replication, and the four nucleotides, we can synthesize a new complementary strand.

## Section 1 – Introduction to Bioinformatics

---

RNA is very similar to DNA (see figure below), and differs from it for



the following reasons:

- Structure: RNA is single-stranded;
- Thymine is replaced with Uracile (U);
- We have different types of RNA.

RNA shows its importance in DNA duplication, as explained in the next section.

### - 1.1.2 - Synthesis.

We have seen that proteins are residues of amino acids, thus, to identify each protein we can consider the amino acids in its primary structure. This is exactly the role of DNA. Precisely, each triplet of nucleotide (codon) encodes an amino acid. Let us observe that 4 nucleotides should produce 64 distinct amino acids, but as we have said, nature only provides us only 20. We can conclude that some codons identify the same acid. Additionally, some codons (UAG, UAA, UGA) do not encode any amino acid, rather they are used to signal the end of a gene.

The process that leads to protein synthesis begins with a phase known as *transcription*. In this phase, a codon AUG identifies the begin of a gene which is copied onto an RNA molecule. We obtain this way the messenger RNA, or mRNA. Now, the process just described works only for prokaryotes; for eukaryotes things are a bit more complex. Essentially, eukaryotes genes are composed of alternating parts: the *introns* and the *exons*. We care only of exons. To solve this issue, once we get the mRNA the parts corresponding to introns are removed.

Synthesis is done in a special cellular structure: the ribosome. Ribosomes are composed by proteins and a new type of RNA, the ribosomal

## **Section 1 – Introduction to Bioinformatics**

---

RNA. Here mRNA is read sequentially and a special type of RNA, the tRNA, carries the amino acid associated with the codone being read.

To summarize: within DNA we produce copies of genes, these are then transcribed via RNA and within ribosomes we synthesize the proteins.

What we just described assumes no error occurs during DNA replication. As easily understood this is rarely the case. In fact, if errors did not occur, evolution would be impossible. More in general, when dealing with DNA sequences we have to consider mutations, i.e., nucleotide X becomes nucleotide Y, and/or gaps, some nucleotide could get added/removed from the original sequence.

---

## Section 2 – Pair-wise alignment

---

### - 2 - Pair-wise alignment.

One of the most important problems in biology is that of similarity; that is, given two strings encoding either DNA or proteins sequences, find the best alignment to make them look similar. We explain the concept of similarity in the next section.

There are essentially two approaches to solve the problem, either via alignment based algorithms or via alignment-free algorithms. The former will be covered in the present section, in which we focus on pair-wise alignment, i.e., alignment of two strings; and in the one following, in which we discuss multiple alignment. We cover alignment-free algorithms in *Section ??*.

#### - 2.1 - Alignment and similarity measures.

Sequence alignment is the process through which we search for patterns within the given sequences. The process can be either applied locally or globally, depending on the needs. We discuss the Needleman-Wunsh algorithm for the global alignment case and the Smith-Waterman algorithm for the local case, (see [1, 2] for major details).

Before we proceed further, let us consider the naïve approach; that is, we just take in consideration mismatches and matches.

Considering two sequences, essentially, we consider all the possible alignments and choose the one with the highest number of matches.

---

#### Example

---

Let  $s_1 = ACCAC$  and  $s_2 = ACGGCC$ . Consider all possible alignments:

$ACCAC$	$ACCGAC$
$ACGGCC$	$ACGGCC$
$ACCAC$	$ACCGAC$
$ACGGCC$	$ACGGCC$
$ACCAC$	$ACCGAC$
$ACGGCC$	$ACGGCC$
$ACCAC$	$ACCGAC$
$ACGGCC$	$ACGGCC$

It's easy to observe that the best one is the top-right one.

---

This approach has a severe issue: it does not take in account gaps, which, as we have said, are extremely frequent in nature. Therefore,

## Section 2 – Pair-wise alignment

---

though easy to implement, we should consider something different.

The necessity to consider gaps lead to a question: how do we define “the best alignment“ when gaps are involved? To solve the problem we define the so called *similarity measure*; i.e., a measure that takes in account both matches/mismatches and gaps. The one we use is the following

$$S_{AB} = \sum_{i=1}^L s(a_i, b_i) - \sum_{k=1}^{NG} \delta + \gamma[l(k) - 1]$$

where  $\delta$  is said to be the *opening penalty*,  $\gamma \leq \delta$  is called *extention penalty* and  $l(k)$  is the length of the  $k$ -th gap.

Let us note that the concept of match/mismatch holds just for DNA and RNA sequences; in the context of amino acids, and therefore proteins, a different criterion is needed. Throuhgout this notes we use the *amino acid interconvertibility* criterion. In this case we make use of two special matrices (more correctly, substitution matrices) which tells us the value to assign to any given pair of amino acid. The ones we consider are PAM-1 and BLOSUM-62.

### - 2.2 - Needleman-Wunsh algorithm.

Proposed in 1970, the Needleman-Wunsh algorithm allows one to compute the best global alignment in  $\mathcal{O}(n^2)$  time (we assume the two sequences to have approximately the same length). The version we discuss is a slight variant of the algorithm proposed in [1].

Let  $s_1$  and  $s_2$  be the sequences to align, also let  $n, m$  be the respective lengths. The algorithm proceeds as follow:

1. Create an  $n \times m$  matrix  $M$ : each entry represents the intersection of the  $i$ -th symbol of  $s_1$  with the  $j$ -th symbol of  $s_2$ .
2. Set the initial value of each entry of the matrix accordingly to the type of sequences in analysis:
  - nucleotides: use the match/mismatch score; or
  - amino acids: use the value provided by the chosen substitution matrix.
3. Update the values of each entry as follow:

$$M(i, j) = \max \begin{cases} M(i-1, j-1) + M(i, j), \\ M(i-1, j) - \delta, \\ M(i, j-1) - \delta. \end{cases}$$

If any of the index is negative, that entry has value equal to zero.

4. Within the last row, or column, find the cell with the maximum score/s; from it track the best path/s proceeding backwards.

## Section 2 – Pair-wise alignment

---

### - 2.3 - Smith-Waterman algorithm.

Proposed in [2], the Smith-Waterman algorithm allows one to find the best local alignment in  $\mathcal{O}(n^2)$  time (we assume the two sequences to have approximately the same length).

Let  $s_1$  and  $s_2$  be the sequences to align, and let  $n, m$  be their lengths. We proceed as follow:

1. Create an  $n \times m$  matrix: each entry represents the intersection of the  $i$ -th symbol of  $s_1$  with the  $j$ -th symbol of  $s_2$ .
2. Set the initial value of each entry of the chosen substitution matrix; either PAM or BLOSUM.
3. Update the value of each entry using the following relation:

$$M(i, j) = \max \begin{cases} 0, \\ M(i-1, j-1) + M(i, j), \\ M(i-1, j) - \delta \text{ or } \gamma, \\ M(i, j-1) - \delta \text{ or } \gamma. \end{cases}$$

If  $M(k, l)$  does not exist, i.e., either  $k$  or  $l$  are negative, assume that entry to be zero.

4. Within the whole matrix, find the maximum<sup>1</sup> score/s and from it track back the best path/s.

In the above relation, the choice between  $\delta$  and  $\gamma$  depends on the gap; i.e., if the gap just opened use  $\delta$ , use  $\gamma$  otherwise.

See that if we limit the search of the maximum score to the last row and column, we get the Needleman-Wunsh algorithm discussed previously; that is, the Needleman-Wunsh algorithm we described is a particular case of the Smith-Waterman algorithm.

---

<sup>1</sup>Alternatively, one could set a threshold and consider the path starting at each entry above said threshold.

---

## - 3 - Multiple alignment.

Multiple alignment is somewhat a generalization of the pair-wise alignment; in fact what it does is align three or more sequences and give the best possible alignment. Here the use of the word best is misleading, as we are about to explain.

Consider the following: can we use dynamic programming to compute the best alignment, as done in the case of two sequences? Assume we want to align three sequences. If we proceed as done for the Needleman-Wunsh algorithm, we must consider the  $n^3$  matrix these form. Also, how shall we compute the scores?

The acute reader might have seen how the problem is isomorphic to the problem of computing the best path in an n-dimension Manhattan Cube.

Since the dimensions of the matrix increase with the number of sequences, a dynamic programming based approach should be avoided, as these require  $\mathcal{O}(n^k)$  time, with  $k$  the number of sequences.

**Remark.** Despite the complexity, dynamic programming based algorithms exist, but are only used in the case of few sequences with a relatively short length.

Hence, unless extreme precision is needed, (e.g. medical analysis), heuristic algorithms suffice our needs. We use the remainder of this section to discuss several of such algorithms.

### - 3.1 - FASTA.

**ToDo.** Explain the algorithm

---

## References.

- [1] Saul B. Needleman and Christian D. Wunsch. “A general method applicable to the search for similarities in the amino acid sequence of two proteins”. In: *Journal of Molecular Biology* 48.3 (1970), pp. 443–453. DOI: [10.1016/0022-2836\(70\)90057-4](https://doi.org/10.1016/0022-2836(70)90057-4).
- [2] T.F. Smith and M.S. Waterman. “Identification of common molecular subsequences”. In: *Journal of Molecular Biology* 147.1 (1981), pp. 195–197. DOI: [10.1016/0022-2836\(81\)90087-5](https://doi.org/10.1016/0022-2836(81)90087-5).

---

## **Acronyms.**

**A** Adenine. 2

**C** Cytosine. 2

**DNA** Deoxyribonucleic Acid. 1

**G** Guanine. 2

**T** Thymine. 2

**U** Uracile. 3

---

## Appendix A.

Amino Acid	3-letter	1-letter
Alanine	Ala	A
Arginine	Arg	R
Asparagine	Asn	N
Aspartic acid	Asp	D
Cysteine	Cys	C
Glutamine	Gln	Q
Glutamic acid	Glu	E
Glycine	Gly	G
Histidine	His	H
Isoleucine	Ile	I
Leucine	Leu	L
Lysine	Lys	K
Methionine	Met	M
Phenylalanine	Phe	F
Proline	Pro	P
Serine	Ser	S
Threonine	Thr	T
Tryptophan	Trp	W
Tyrosine	Tyr	Y
Valine	Val	V