Three Memorable Topics from the Bioinformatics Course

Adrian Klimaševski[1]

[1]**Vilnius University, Faculty of Mathematics and Informatics, Study Programme: Software Engineering, Year of Study: 3rd year, 1st semester**

Date: January 6, 2026
Course: Bioinformatics
Coordinator: Gediminas Alzbutas, Partn. Prof., Dr.
Substitute Coordinator: Saulius Gražulis, Prof., Dr.
Substitute Practical Classes Coordinator: Irus Grinis, Lekt.

Abstract This essay reviews 3 topics studied in the Bioinformatics course during the 2025 Autumn semester: codon and dicodon frequency analysis in viral genomes, the Needleman–Wunsch global alignment algorithm, and the BLAST sequence search tool. For each topic, the underlying problem, applied methods, and their advantages and limitations are discussed.

Contents

1 Codon and Dicodon Frequency Analysis in Viral Genomes (Lab1)

The goal is to analyze codon and dicodon usage patterns to see if *mammalian* and *bacterial* viruses can be distinguished based on their protein sequences and possibly see how they evolved. Such analyses are relevant for vaccine research.

1.1 Problem

Viral genomes often contain overlapping genes and proteins that could be encoded on both forward and reverse strands.

1.2 Method

1. For each reading frame (0, 1, 2), we searched for start codon (ATG) and stop codons (TAA, TAG, TGA). (This raised the question of why DNA contains three stop codons but only one start codon? According to [3], to reduce errors and improve efficiency)

2. Then, ORFs shorter than 100 were excluded. The remaining ones were translated into protein sequences.

3. Next, codon and dicodon frequencies were computed.

4. We calculated distances between viral genomes using Euclidean distance.

1.3 Advantages & Disadvantages

The main advantage of this approach is its simplicity and low computational cost, which makes it suitable for large viral datasets. However, it relies on simplified ORF detection rules and ignores alternative start codons, which may lead to incomplete or inaccurate protein predictions.

1.4 Euclidean Distance Calculation Function

```
def euclidean_distance(vec1, vec2):
    return math.sqrt(
        sum((vec1[k] - vec2[k]) **
            2 for k in vec1.keys())
    )
```

1.5 Results Overview

Both codon and dicodon analyses showed partial separation between *mammalian* and *bacterial* viruses. In general, dicodon clustering produced clearer group separation.

2 Needleman–Wunsch Algorithm

The goal of the Needleman–Wunsch algorithm is to compute an optimal alignment between 2 biological sequences.

2.1 Problem

Biological sequences often differ due to mutations, insertions, deletions and evolutionary divergence. The challenge is to find an alignment that maximizes similarity under a predefined scoring system.

2.2 Method

The Needleman–Wunsch algorithm constructs a scoring matrix where each cell represents the optimal alignment score. Scores are computed based on match rewards, mismatch penalties and gap penalties.
Algorithm can be separated into 3 stages:

1. Initialization of the scoring matrix.

2. Filling the matrix.

3. Backtracking to reconstruct the optimal alignment.

2.3 Advantages & Disadvantages

Needleman–Wunsch finds an optimal alignment (based on predefined penalties/rewards). However, its computational complexity $O(n^2)$ makes it impractical for large databases or very long sequences [2].

2.4 Matrix Filling

```
for I in M'First(1) + 1 .. M'Last
   (1) loop
   for J in M'First(2) + 1 .. M'
      Last(2) loop
      if Element (Seq1, I) =
         Element (Seq2, J) then
         Score_Diag := M (I-1, J-1)
            .Score + Match_Score;
      else
         Score_Diag := M (I-1, J-1)
            .Score +
            Mismatch_Penalty;
      end if;

      Max_Score := Long_Integer'Max
        (Long_Integer'Max (Score_Up
           , Score_Left),
          Score_Diag);
      M (I,J).Score := Max_Score;
   end loop;
end loop;
```

3 BLAST

The goal of BLAST is to identify similar regions between a query sequence and sequences stored in large biological databases. BLAST focuses on speed and scalability [1].

3.1 Problem Definition

Exact alignment algorithms such as Needleman–Wunsch are computationally expensive and therefore unsuitable for large-scale database searches. Also, biological sequences often share similarity only in short regions rather than across their full length. The challenge is to efficiently detect similarities/alignments.

3.2 Method

1. Identification of short exact or near-exact matches.

2. Extension of these matches in both directions to form high-scoring pairs.

3. Evaluation.

3.3 Advantages & Disadvantages

BLAST is quick and handles large databases. However, it does not guarantee finding the optimal alignment and may miss weak similarities.

Use of AI tools.
LLMs were used to assist with LaTeX formatting, minor language refinement, support in academic writing and bibliography formatting. All technical content, interpretations and code examples were written and verified by the author.

References

[1] University of California, Berkeley Library. (n.d.). *NCBI BLAST: Basic Local Alignment Search Tool*. Retrieved January 6, 2026, from https://guides.lib.berkeley.edu/ncbi/blast

[2] Kurt, F., Altilar, D. T., & Yilmazer Metin, A. (2022). Implementations of the Needleman–Wunsch algorithm for GPU architectures. In *7th Ulusal Yüksek Başarımlı Hesaplama Konferansı (BASARIM 2022)*, Istanbul Technical University. Retrieved January 6, 2026, from https://web.itu.edu.tr/yilmazerayse/docs/Furkan_BASARIM_2022.pdf

[3] Křížek, M., & Křížek, P. (2012). Why has nature invented three stop codons of DNA and only one start codon? *Journal of Theoretical Biology*, 304, 183–187. Retrieved January 6, 2026, from https://pubmed.ncbi.nlm.nih.gov/22483666/

[4] OpenAI. (2025). *ChatGPT* (GPT-5.2). Retrieved January 6, 2026, from https://chat.openai.com