Deriving BLOSUM-matrices for protein-coding DNA

Alexander Temper

August 23, 2023

Abstract

Sequence Alignment algorithms rely on some form of substitution matrix, which can heavily influence their results. In this work, we try to derive a substitution matrix for the genes (DNA) of a given protein family akin to the BLOSUMx matrices, which are used for amino acid sequences.

Contents

1 Introduction **1**

2 Previous work **2**

3 BLOSUM **2**

1 Introduction

Test. Sequence alignment is the task of pairing up the elements of two (biological) sequences with the aim of finding conserved regions and homologous sequences which are anticipated to have some evolutionary relation. Sequence alignment algorithms are primarily used for constructing phylogenetic trees, searching for homologous sequences with BLAST, assembling full DNA strands from short reads, creating input features for AlphaFold2 and other. Most alignment algorithms try to maximize a so called *scoring function*, which assigns to a given alignment some score. The classical methods score alignments by summing up scores assigned to each pair of letters in the given alignment. The scores for the pairs of letters can be represented by a symmetric matrix we call such matrices *scoring matrices*. Next, we formally define our vocabulary. W.l.o.g., let us restrict ourselves to pairwise alignments, i.e., alignments of two sequences.

Definition 1 *The following will be used throughout the paper:*

- *A* sequence *s is an ordered collection of letters from some alphabet of letters $\mathcal{L}$.*
- *The letter '-' is called* indel *and represents gaps in the alignment, which occur due to mutations or sequencing errors.*
- *A (pairwise)* sequence alignment *is a matrix $\mathbf{A} \in (\mathcal{L} + \{-\})^{2\times m}$, where m is the length of the alignment. We symbolize the set of all alignments of alphabet $\mathcal{L}$ of length m*
- *A scoring function $\sigma : (\mathcal{L} + \{-\})^{2\times m} \to \mathbb{Z}$ maps an alignment to its score.*

Most algorithms of concern here use a scoring function of the form $\sigma(\mathbf{A}) = \sum_{j=0}^{m} s(\mathbf{A}_{1j}, \mathbf{A}_{2j})$, where $s : (\mathcal{L} + \{-\})^2 \to \mathbf{Z}$ is a symmetric function evaluating pairs of letters, which can be represented by a matrix $\mathbf{S} \in \mathbb{Z}^{(\#\mathcal{L})+1\times(\#\mathcal{L})+1}$. Said matrix is the focus of this work.

Example 1 *Since we are talking about DNA, our alphabet of concern will be the 4 different nucleotide bases, {`A, C, G, T`}. Two exemplary DNA sequences are `ACA` and `AAGA`. One possible alignment between them is*

$$\mathbf{A} = \begin{bmatrix} \texttt{A} & \texttt{C} & - & \texttt{A} \\ \texttt{A} & \texttt{A} & \texttt{G} & \texttt{A} \end{bmatrix}$$

Indexes where the two nucleotides are equal are called matches*, where the two are not equal are called* mismatches

and where an indel is matched to a nucleotide are called gaps. *An exemplary scoring matrix might be*

$$\mathbf{S} = \begin{array}{c|ccccc} & \mathtt{A} & \mathtt{C} & \mathtt{G} & \mathtt{T} & - \\ \hline \mathtt{A} & 1 & -2 & -3 & 0 & 0 \\ \mathtt{C} & -2 & 0 & 0 & 0 & 0 \\ \mathtt{G} & -3 & 0 & 0 & 0 & -4 \\ \mathtt{T} & 0 & 0 & 0 & 0 & 0 \\ - & 0 & 0 & -4 & 0 & 0 \end{array}$$

Using the above matrix,

$$\sigma(\mathbf{A}) = s(\mathtt{A},\mathtt{A}) + s(\mathtt{C},\mathtt{A}) + s(-,\mathtt{G}) + s(\mathtt{A},\mathtt{A}) = 1 - 2 - 4 + 1 = -6.$$

2 Previous work

Most notably, the two families of classical scoring matrices are the PAM matrices and the BLOSUM matrices. However, and this is also true for most other research around sequence alignment: there is a strong emphasis on studying the alignment of proteins over studying the alignment of DNA. To the best of our knowledge, nearly all benchmarks concern *only protein alignments*, which we suspect to to be one of the reasons that there is limited research focusing on DNA alignment. There have been attempts to create gold standards for RNA alignment, however, even there, investigations suggest an overrepresentation of tRNA, thus leading to a suboptimal benchmark.

The literature for DNA scoring matrices derived from data is sparse. Hamada et al. [2] derive scoring matrices from specific organisms sequenced by specific sequencers. Further, as they claim, different genes in different organisms have significantly differing rates of mutation, which is why general-purpose DNA scoring matrices might be a bad idea. Their main focus lies on recovering and mitigating the sequencing errors and projecting the differing GC contents of different species in the resulting matrix. They evaluate their data on simulated data, which by definition makes some assumptions, thus is not too optimal. Their method performs slightly better than 2 manually created scoring matrices.

3 BLOSUM

As a successor to the PAM matrices, Henikoff and Henikoff [3] first constructed the BLOSUM matrices for protein alignments. A wonderful explanation thereof was written by Eddy [1], however, we shall briefly dive into the theoretical underpinnings of BLOSUM here as well.

Underlying BLOSUM is the equation that, given the two paired letters $a \in \mathcal{L}, b \in \mathcal{L}$,

$$s(a,b) = \lambda \log_2 \frac{P(a,b)}{P(a)P(b)}.$$

Let us disect this:

- $s(a,b)$ is the score of a and b being aligned.
- Our two hypotheses are are that
  1. a and b are related evolutionarily and ought to be aligned and
  2. a and b are aligned due to random chance.
- We are interested in the odds of the former being true over the latter and encoding this into a score.
- We can approximate this with already existing, aligned data. The original paper used the so called Blocks matrix for this.
- We can get approximate $P(a,b)$ by counting all the aligned pairs of the already existing alignment and normalizing those counts to probabilities. $P(a)$ can be computed analogously.
- For numerical reasons, we would like the score to be an integer, which is why we scale all all scores with $\lambda = 2$.

This family of matrices has become a de-facto standard for aligning amino acid sequences - however, not so much for DNA. BLASTn, i.e., BLAST for nucleotide bases, currently uses a matrix where matches are rewarded with +2 and mismatches are penalized with -3. It is this very assumed generality that has motivated this paper.

Interestingly enough, there have been mistakes in the original computation of the BLOSUM matrices, which are claimed to have been improving them quietly. This, however, might be attributed to the fact that the benchmarks today might be influenced from the BLOSUM of the past.

Experiments and results

Method

We constructed a fully automated pipeline which takes as input an identification code of a protein family on InterPro and the similarity threshold x and computes the corresponding BLOSUMx matrix. The source code can be found online, yet, here we give some detail on the implementation and issues we faced.

First, we search the NCBI protein database for the given identification code using eDirect. This yields only a subset of the desired genes, since not every protein in the database is annotated with all protein family codes. It is however, to the best of our knowledge, the fastest and most reliable way to download genes of a given protein family on InterPro. For the scope of this work, we deem this sufficient.

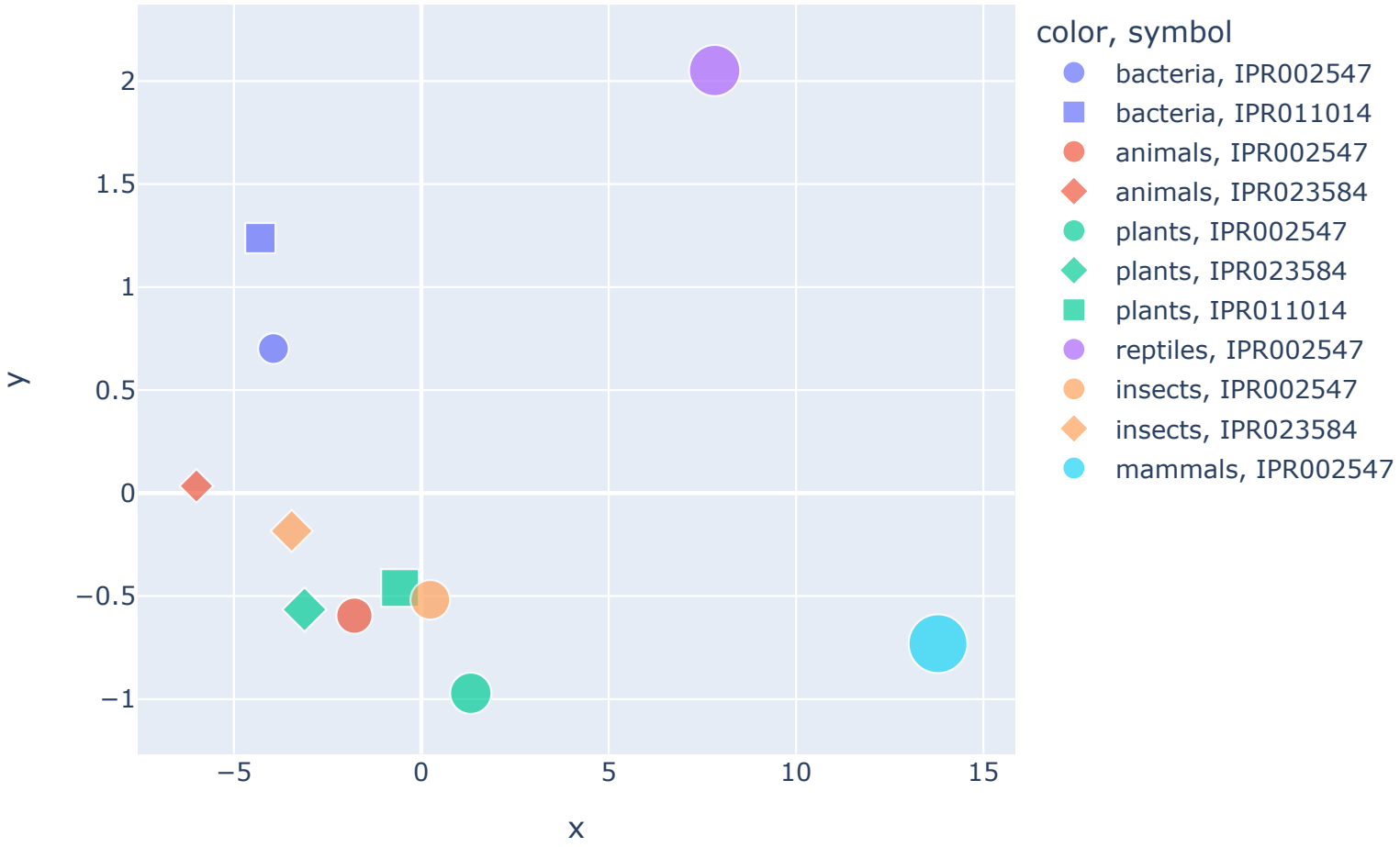
Next, the downloaded genes are preprocessed. Genes which contain elements besides `ACGT` are removed. Then, the genes are being sorted into bins, depending on their length, the rationale thereof being that most multiple sequence alignments assume that the sequences are of similar length. Further, bins which have too few sequences in them are also being removed.

Unfortunately, the PROTOMAT system originally used by Heinikoff is not to be found online any longer. Thus, we create our own blocks, which is clearly a major difference to the construction of the orgininal BLOSUM matrices.

NOTE: BLAST should provide Web interface for Matrix. To do so, each bin is getting aligned by kalign (Use different software?) with its default settings. This results in an MSA for each bin.

Thereafter, each MSA is looked at and columns which contain less than a certain percentage of gaps are selected. Then, from the 'dense' columns, we find blocks of adjacent dense columns. These will be the blocks for the computation of the BLOSUMx matrix. This is a somewhat

substantial difference: the blocks in the original paper are **gapless**, whereas the blocks in this paper **contain some gaps**.



References

[1] Sean R Eddy. "Where Did the BLOSUM62 Alignment Score Matrix Come From?" In: *Nature Biotechnology* 22.8 (Aug. 2004), pp. 1035–1036. ISSN: 1087-0156, 1546-1696. DOI: 10.1038/nbt0804-1035. (Visited on 08/05/2023).

[2] Michiaki Hamada et al. "Training Alignment Parameters for Arbitrary Sequencers with LAST-TRAIN". In: *Bioinformatics* 33.6 (Mar. 2017), pp. 926–928. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btw742. (Visited on 07/14/2023).

[3] S Henikoff and J G Henikoff. "Amino Acid Substitution Matrices from Protein Blocks." In: *Proceedings of the National Academy of Sciences* 89.22 (Nov. 1992), pp. 10915–10919. DOI: 10.1073/pnas.89.22.10915. (Visited on 08/05/2023).