Bioinformatics Project

DD2404 - Bioinformatics
Reducing noise in protein multialignments

Christopher Herron - cherron@kth.se - 1993.09.20-1979
David Olanders - davola@kth.se - 1993.06.04-8613
André Zachrisson - andrez@kth.se - 1994.05.21-8257
January 3, 2019

Examiner: Lars Arvestad

Abstract

Due to noise, it is not always possible to infer a correct alignment when working with protein multi-alignments. Therefore it is of interest to evaluate if noise reduction is worthwhile, by comparing the symmetric distance of an unfilterd vs noise reduced tree to a reference tree. In this experiment noise is defined as columns with more than 50% indels, at least 50% of the amino acids in the column are unique or columns where no amino acid appears more than twice. The noise reduction is done with *Python* and built in modules such as *subprocess*, to create trees with *FastPhylo* and *dendropy* to calculate the symmetric distance. The statistics indicate that there is a marginally lower mean distance from the reduced tree for both symmetric and asymmetric test cases. The biggest difference is found where the average amount of mutations per site in the sequences is the highest, however the difference is at most 7%. Based on the method, definitions and test data in this experiment it is not worth using noise reduction.

Contents

1 Introduction

Protein multialignments often have elements of noise after applying the alignment process. The homologous proteins can possess areas that evolved at such a rapid pace that it's not possible to infer a correct alignment, or possess areas that is simply not inherited and because of this should not be aligned. Because of this it can be interesting to eliminate these types of areas so that subsequential analysis, and especially phylogeny inference, doesn't consider those columns. A column from a multialignment process is, in this report, considered noisy and thereby removed if at least one of the following three conditions are fulfilled:

- There are more than 50% indels in the column
- At least 50% of the amino acids in the column are unique
- No amino acid appears more than twice in the column.

This report aims at implementing a method for noise reduction and after this evaluate its impact on phylogeny inference. From this it will be possible to conclude if it's worthwhile to use noise reduction on a multialignment process. The implementation of methods for testing noise reduction was successfully achieved.

2 Result and Discussion

With the implementation of a noise reduction method, calculations of distances between the given reference tree and the unfiltered and noise reduced alignment were computed for the symmetric and asymmetric data. The distances that was of interest was the minimum, maximum and mean distance since these distances give a sufficient overview of the relation between the unfiltered and noise reduced multialignments. The minimum, maximum and mean distances are presented in Table 1.

Table 1: Distance comparison to reference tree of a multialigned process (Unfiltered) and a noise reduced multialigned process (Noise reduced).

Test File	**Min Distance**		**Max Distance**		**Mean Distance**	
	Unfiltered	**Noise reduced**	**Unfiltered**	**Noise reduced**	**Unfiltered**	**Noise reduced**
$asymmetric_{0.5}$	0	0	18	18	7.63	7.54
$asymmetric_{1.0}$	2	2	18	18	9.43	9.39
$asymmetric_{2.0}$	4	4	20	22	12.34	11.69
$symmetric_{0.5}$	0	0	10	12	4.16	3.97
$symmetric_{1.0}$	0	0	10	10	5.01	4.88
$symmetric_{2.0}$	0	0	16	16	6.96	6.53

The statistics in Table 1 showed that the minimum and maximum distance was not affected if noise reduction was applied, with the exception of the maximum distance for $asymmetric_{2.0}$ and $symmetric_{0.5}$ where these were two units higher in the noise reduced alignments. However, when examining the mean distance, a small difference between the unfiltered and the noise reduced was detected. This difference varies for the different input data, *symmetric* or *asymmetric*, but is the biggest in the cases where the average amount of mutations per site in the sequences is highest, i.e $symmetric_{2.0}$ and $asymmetric_{2.0}$. However, this is considered as low since the biggest difference is only 7%, in the $symmetric_{2.0}$ data case. This thereby implicate that noise reduction was not worthwhile. To further analyze the implication, the distance difference between the unfiltered and noise reduced alignments to the reference tree for the asymmetric and symmetric data were computed, see Figure 1 and Figure 2.

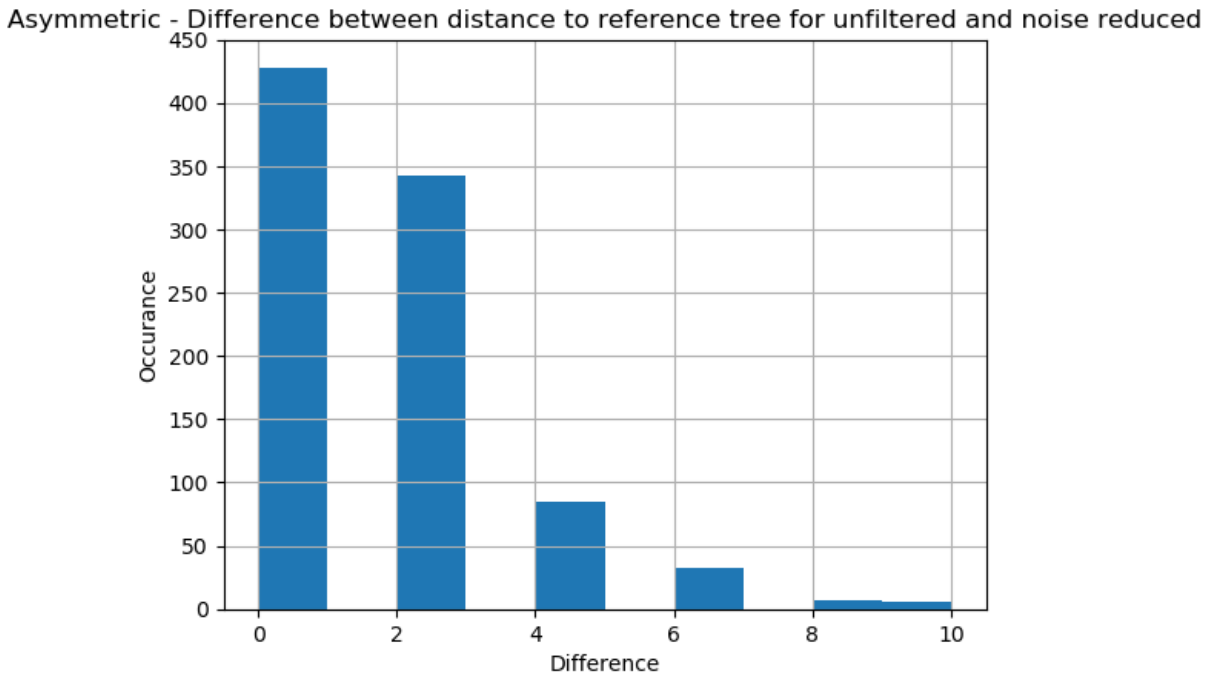

Figure 1: Histogram of the distance difference between the unfiltered and noise reduced alignments to the reference tree of the asymmetric data.

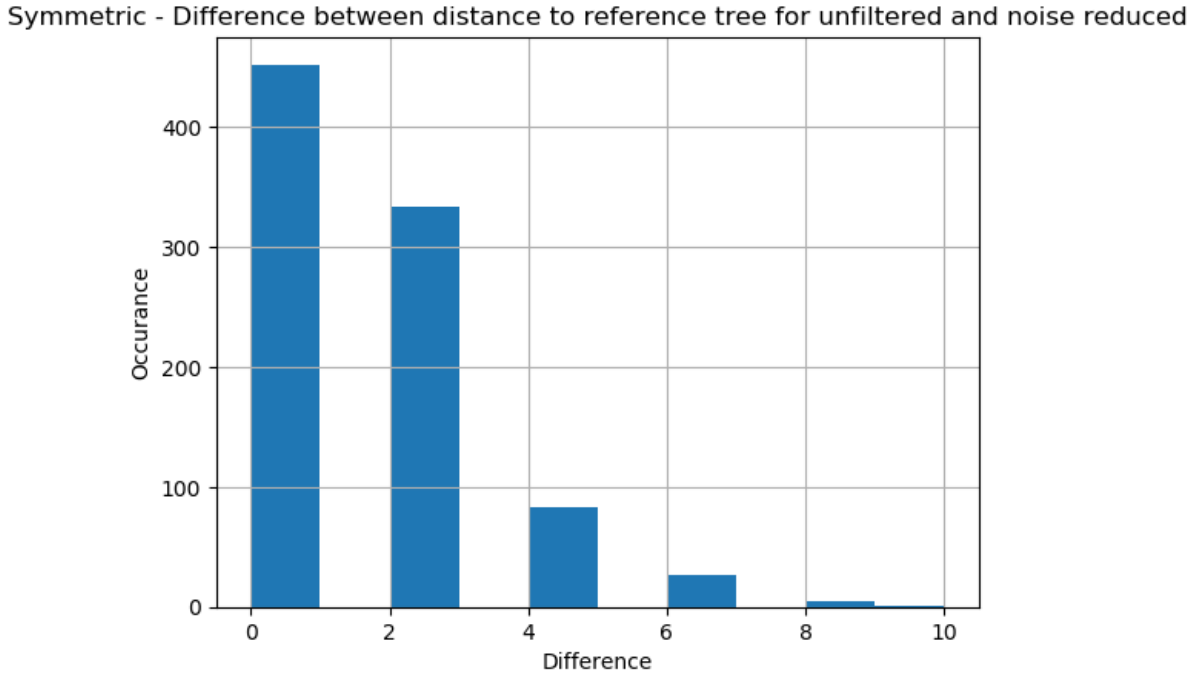

Figure 2: Histogram of the distance difference between the unfiltered and noise reduced alignments to the reference tree of the symmetric data.

Figure 1 and Figure 2 implicated that there is a low difference, 0 or 2 units, between the unfiltered and the noise reduced input compared to the reference tree for the majority of the alignments. This further implicates what was argued when examining the statistics in Table 1, that the noise reduction was not worthwhile.

3 Methods and Materials

The programming language *Python* was used to efficiently, through statistics, test if noise reduction had a positive effect on inferring alignments. The *Python* script was designed by firstly creating a noise reduction algorithm. This algorithm was then tested with five different corner cases:

- An empty document
- Only one sequence in the file
- There are more than 50% indels in the column
- At least 50% of the amino acids in the column are unique
- No amino acid appears more than twice in the column.

The next step was to visualize the results for further confirmation that the algorithm worked. This was done by using Seaview which visualizes multialignments. In Figure 3 an unfiltered multialignment is presented and in Figure 4 the noise reduced version is presented, both with the same test data.

Figure 3: Example unfiltered multialignment.

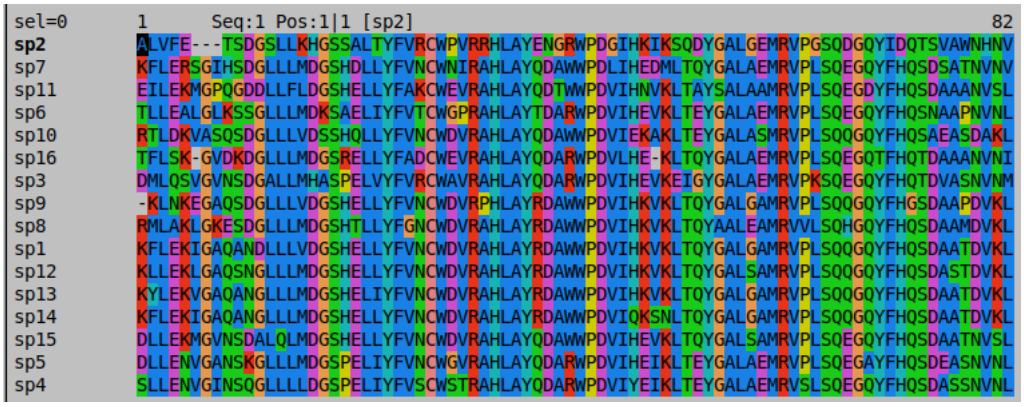

Figure 4: Example noise reduced multialignment.

The next step was to, for each gene alignment, infer a tree using the programs *fastprot* and *fnj* from the software *FastPhylo*. Since *FastPhylo* runs from the terminal, the *Python* module *subprocess* was used to run all the files by automation (note: *subprocess*es is slow). The programs *fastprot* and *fnj* both create files which are stored in a temporary directory, since there was no need to save them for this experiment.

After the unfiltered and noise reduced trees have been created, the last remaining step was to compare the trees. To compare the unfiltered and noise reduced tree with the reference tree the *Python* module *dendropy* was used. The output i.e the distance to the reference tree was saved and written in a csv file format since collecting the results from all 6 test files, which all have 300 alignments, was time consuming. The statistics was calculated in a separate *Python* script using the csv file with the distances as input.

4 Stafford Noble Discussion

File and Directory Organization
During the project, the directories and the file organization used followed the recommended structure in the literature. The structure used has five main folders, data, doc, src, bin and results. Src and bin is where the source code and the executable is stored, in data we stored all data that we used in the experiment, in results the results for each day we executed the experiment was stored in a separate folder in a chronological order and finally in doc the report is stored together with images relating to the report.

The Lab Notebook
We have used a simplified notebook written in the online texteditor Overleaf, thereby adapting an online notebook as Stafford Noble suggests. To access the notebook one needs a password and invitation to edit the document which, at least the password part, is mentioned in the Stafford Noble text. We have also added email correspondence, though not thoroughly described. The notebook gives an overview of what was done each day and what needs to be done the next time, thereby not describing what was done in detail, as Stafford Noble suggests. Notes from conversations were not added. This was done for simplicity since the three of us sat together and worked simultaneously on the project. We only keep one notebook, in contrast to three individual ones as Stafford Noble suggests for our project, since we worked together and thereby didn't need more thoroughly description of what was done. The notebook resides in the results directory, as Stafford Noble suggests.

Carrying Out a Single Experiment
Since we worked with the language Python, a script was easily stored and further developed through the version controller Github. Since our main script, that ran the experiment, took a long time we used a second script (statistics) to summarize the results through tables and plots.

Handling and Preventing Errors
During the project we always tried to write robust code. We did this by constantly checking if the output was correct based on the input. The debugger pdb was a great tool to check the temporary files and the output of functions. We did however not create output files with temporary names. This did not seem necessary as the result file was created at completing the experiment. If the results were wrong the experiment was done again and the previous result file overwritten, thus removing the chance of using incorrect results.

Command Lines versus Scripts versus Programs
In this project we have created a main script that does not need any input from the user and the result will automatically be stored in the results folder under the current date. There is also a separate script for calculating the statistics for the project. The code could be executed faster if we use c for some functionality. *The Value of Version Control*
For this project we used GitHub for version control. The branch function was never used as we never left the code "broken" before committing. All automatically generated files that were not part of the result where in a temporary folder, which means they were never committed.