

# Reducing noise in protein multi-alignments

## Introduction

Multi-alignments are noisy. Homologous proteins can contain regions that are not inherited and should therefore not be aligned, and other regions may have evolved so fast that the correct multi-alignment is impossible to infer. In order to get rid of such problematic regions in subsequent analysis, noisy columns are removed. In this project, a column is considered noisy if

- there are indels,
- at least 50% of amino acids are unique

To validate whether this reduction is useful or not, I had to infer a tree from the given alignments and the noise-reduced alignments of an alignment set having different mutation rates. Then the reference tree of particular alignments was compared with the noise-reduced tree and the tree from the given alignment.

## Results and Discussion

In case of symmetric data, the frequency of recovery of the reference tree improves a bit but in case of asymmetric data it doesn't make any difference. Moreover, the average symmetric difference in case of both symmetric and asymmetric data improves insignificantly.

So it's not worthwhile to use multi-alignment noise reduction based on the above specified criterion.

### Frequency of reference tree being recovered:

Data Group	Before noise reduction	After noise reduction
asymmetric_0.5	1	1
asymmetric_1.0	0	0
asymmetric_2.0	0	0
symmetric_0.5	21	27
symmetric_1.0	16	18
symmetric_2.0	6	12

### Average symmetric difference:

Data Group	Before noise reduction	After noise reduction
symmetric_2.0	7.33	7.29
asymmetric_1.0	9.4	8.88
asymmetric_2.0	12.51	11.39
symmetric_0.5	4.07	4.06
symmetric_1.0	5.07	4.81
symmetric_2.0	7.06	6.28

## **Controls for correctness:**

To ensure correctness of program ; I have tried to make case distinction of possible exceptions that might occur. For example , records in multi-alignment file should have same length , reference tree should be in valid form to be parsed by dendropy for measuring difference, input alignments should be in proper format , detection of empty and/or bad files , absence of data to work with . Python exception handling is there to reduce the burden of massive coding .

## **Programming Issue/Used Tools:**

I have used phylip protdist and phylip neighbor instead of fastprot and fnj respectively.