Available online at www.scholarsresearchlibrary.com

Scholars Research Library

Der Pharmacia Lettre, 2015, 7 (11):197-201
(http://scholarsresearchlibrary.com/archive.html)

Scholars Research Library
ISSN 0975-5071
USA CODEN: DPLEB4

Algorithmic determination of amino acid substitution group

Mukul Dev and M. Yamuna

VIT University, India

ABSTRACT

Most of the existing approaches view amino acid substitution as a pairwise phenomenon. Most methods characterizes it using substitution matrices. Some methods focus on determination of substitution groups based on the theoretical properties satisfied by the substitution groups. Be it any method algorithms on these reliable techniques are required for actual determination of the amino acid substitutions. In this paper we provide an algorithmic approach for determination of amino acid substitution group.

Key words: Amino acid, Amino acid substitution, Amino acid conservation, Amino acid properties, Substitution groups, Sequence alignment,

---

INTRODUCTION

Methods for alignment of protein sequences typically measure similarity by using a substitution matrix with scores for all possible exchanges of one amino acid with another. The most widely used matrices are based on the Dayhoff model of evolutionary rates. In [1] using a different approach, a substitution matrices from about 2000 blocks of aligned sequence segments characterizing more than 500 groups of related proteins is developed. This led to marked improvements in alignments and in searches using queries from each of the groups. In [2] a method for identifying amino acid substitution groups that are conserved empirically in aligned positions from databases of protein families is introduced. In [3] a method for identifying empirically conserved amino acid substitution groups is introduced. In contrast with existing approaches that view amino acid substitution as a pairwise phenomenon, this method identifies conserved groups of amino acids using conditional distribution matrix.

Net is a powerful programming platform with integrated visualization programming environment. C# is designed for .Net platform. C# is simple, modern, object oriented and type safe programming language e that combines the high productivity of rapid application development languages with raw power of C and C++. C# provides friendly interface, fast execution speed , high security with creating EXE files and can be run with .net framework instead of entire software.

C# could create simple client applications of Windows, XML Web services, distributed component, C/S applications, database applications, etc. Through CLR (Common Language Runtime) , the program compiled by C# will run steadily on computers with .Net Framework. Application developers normally need not be concerned with using processors or Language. Tools described herein will run on it so long as with .Net Framework [4] [5].

Proposed Algorithm

Amino acid substitution groups have long been studied because they are useful for comparing and studying group wise and consensus relationships between proteins. Various methods are developed to serve this purpose. In most cases the substitution of amino acids for one another in protein sequences has been viewed primarily as a pairwise phenomenon. In many cases, such phenomenon have proven quite useful for comparing, aligning, and exploring relationships between pairs of protein sequences. However, not all sequences are similar. So the question is when a

two protein sequences can be considered to be similar in such cases. How do we know which properties are most important for determining protein structure? Are some properties more important than others in different contexts? In [6] a theoretical determination of amino acid in such cases is provided. A matrix of amino acids using 48 properties like aromatic, aliphatic, acidic, basic, polar etc is constructed. Using this 48 x 20 matrix a 20 x 20 amino acid substitution group is determined. The table is seen in Snapshot – 1.

	A	R	N	D	C	E	Q	G	H	I	L	K	M	F	P	S	T	W	Y	V	
A	1.00	0.13	0.38	0.29	0.63	0.21	0.29	0.75	0.21	0.71	0.79	0.17	0.71	0.58	0.63	0.50	0.29	0.33	0.17	0.75	A
R	0.13	1.00	0.42	0.42	0.25	0.50	0.50	0.13	0.58	0.17	0.25	0.96	0.33	0.29	0.17	0.29	0.17	0.38	0.29	0.13	R
N	0.38	0.42	1.00	0.75	0.50	0.58	0.83	0.38	0.58	0.33	0.42	0.46	0.50	0.38	0.50	0.63	0.58	0.46	0.38	0.46	N
D	0.29	0.42	0.75	1.00	0.42	0.83	0.58	0.29	0.58	0.25	0.33	0.46	0.42	0.29	0.42	0.46	0.42	0.29	0.29	0.38	D
C	0.63	0.25	0.50	0.42	1.00	0.33	0.42	0.46	0.42	0.42	0.50	0.29	0.67	0.46	0.42	0.63	0.42	0.38	0.38	0.46	C
E	0.21	0.50	0.58	0.83	0.33	1.00	0.75	0.21	0.58	0.25	0.33	0.54	0.42	0.29	0.25	0.38	0.25	0.29	0.29	0.21	E
Q	0.29	0.50	0.83	0.58	0.42	0.75	1.00	0.29	0.58	0.33	0.42	0.54	0.50	0.38	0.33	0.54	0.42	0.46	0.38	0.29	Q
G	0.75	0.13	0.38	0.29	0.46	0.21	0.29	1.00	0.21	0.46	0.54	0.17	0.54	0.42	0.54	0.50	0.29	0.25	0.08	0.50	G
H	0.21	0.58	0.58	0.58	0.42	0.58	0.58	0.21	1.00	0.25	0.33	0.63	0.42	0.38	0.25	0.46	0.33	0.54	0.46	0.21	H
I	0.71	0.17	0.33	0.25	0.42	0.25	0.33	0.46	0.25	1.00	0.92	0.21	0.75	0.63	0.58	0.29	0.42	0.38	0.21	0.88	I
L	0.79	0.25	0.42	0.33	0.50	0.33	0.42	0.54	0.33	0.92	1.00	0.29	0.83	0.71	0.67	0.38	0.33	0.46	0.29	0.88	L
K	0.17	0.96	0.46	0.46	0.29	0.54	0.54	0.17	0.63	0.21	0.29	1.00	0.38	0.33	0.21	0.33	0.21	0.42	0.33	0.17	K
M	0.71	0.33	0.50	0.42	0.67	0.42	0.50	0.54	0.42	0.75	0.83	0.38	1.00	0.79	0.67	0.46	0.33	0.54	0.38	0.71	M
F	0.58	0.29	0.38	0.29	0.46	0.29	0.38	0.42	0.38	0.63	0.71	0.33	0.79	1.00	0.54	0.33	0.21	0.67	0.58	0.58	F
P	0.63	0.17	0.50	0.42	0.42	0.25	0.33	0.54	0.25	0.58	0.67	0.21	0.67	0.54	1.00	0.38	0.33	0.46	0.29	0.71	P
S	0.50	0.29	0.63	0.46	0.63	0.38	0.54	0.50	0.46	0.29	0.38	0.33	0.46	0.33	0.38	1.00	0.79	0.50	0.50	0.33	S
T	0.29	0.17	0.58	0.42	0.42	0.25	0.42	0.29	0.33	0.42	0.33	0.21	0.33	0.21	0.33	0.79	1.00	0.38	0.38	0.46	T
W	0.33	0.38	0.46	0.29	0.38	0.29	0.46	0.25	0.54	0.38	0.46	0.42	0.54	0.67	0.46	0.50	0.38	1.00	0.75	0.33	W
Y	0.17	0.29	0.38	0.29	0.38	0.29	0.38	0.08	0.46	0.21	0.29	0.33	0.38	0.58	0.29	0.50	0.38	0.75	1.00	0.17	Y
V	0.75	0.13	0.46	0.38	0.46	0.21	0.29	0.50	0.21	0.88	0.88	0.17	0.71	0.58	0.71	0.33	0.46	0.33	0.17	1.00	V
	A	R	N	D	C	E	Q	G	H	I	L	K	M	F	P	S	T	W	Y	V	

Snapshot – 1

This table is meaningful because if the score of almost 50 properties is similar then it is considered for amino acid substitution group. The grey color cells represent the amino acid substitution groups. This is a theoretical method of determining if two sequences are similar. In this paper we have developed an algorithm for determining this substitution group. This enables us to determine those sequences which match atleast by 50% of the properties.

Algorithm

Let S1 and S2 be the two amino acid sequences.

Step 1 Convert the sequences into character array stating from 0.

Step 2 If the array is of same length then we continue further, else we get a comment as Enter Equal Length Protein Sequences

Step 3 If the sequences are of same length then, we go for pairwise comparison of the sequences. If the entry in sequence 1 matches with the one in sequence 2 as per the amino acid substitution group from Snapshot – 1, we assign a value 1 else value 0.

Step 4 If all the resulting values are 1, then the sequences match each other.

Step 5 If atleast one entry is 0, then we conclude that the sequences do not match.

Illustration

Let S1: LIVMAFPG, S2: VILAMPFG

S1	L	I	V	M	A	F	P	G
S2	V	I	L	A	M	P	F	G
Binary value	1	1	1	1	1	1	1	1
Conclusion	**The sequences match each other**							

S1	L	I	V	M	A	F	P	G
S2	P	V	L	I	M	I	G	F
Binary value	1	1	1	1	1	1	1	0
Conclusion	**The sequences do not match each other**							

Implementation of the Algorithm

We have implemented the algorithm in C++. The main part of the code is seen in Snapshot – 2.

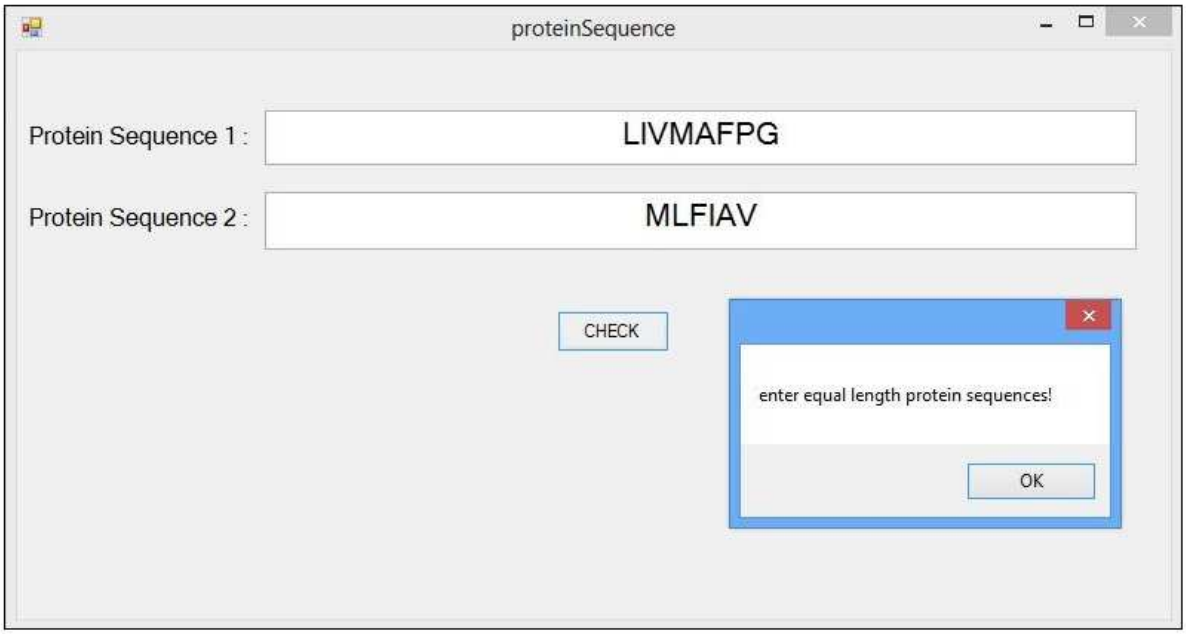
```
Algorithm : Equivalance_Protein_Sequences
Input : 2 Protein Sequences
Output : Equivalency Determination of input Sequences

step 1: Enter Seq1
step 2: Enter Seq2
step 3: if Seq1 is null or Seq2 is null
                then exit
        end if
step 4: if Seq1.lenght() equals Seq2.lenght()
step 5:         for i = 0 to Seq1.lenght() do
                        if Seq1[i] equals Seq2[i]
                                then flag ++;

                        else do
                                for j = 0 to 20 do
                                        if Seq1[i] equals Normalized Matrix Seq2[j]
                                                flag++
                        end if
        end if
step 6: if flag.length() equals seq1.length()
                print Equivalent Sequence
        else
                print Not Equivalent
        end if
```

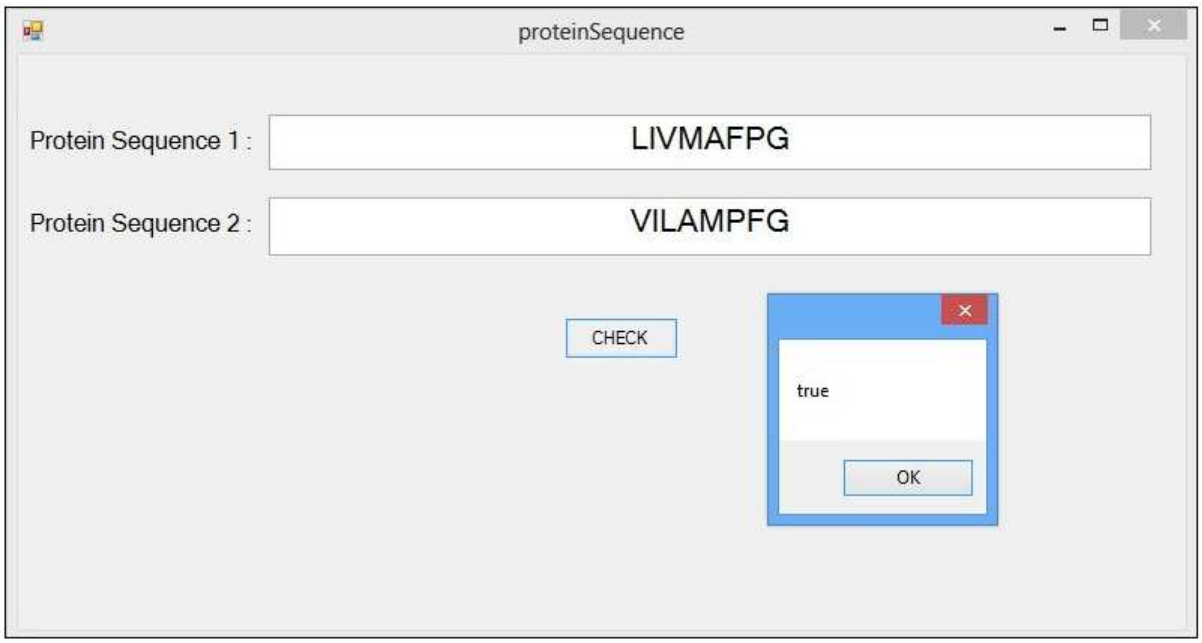
Snapshot – 2

We know that if the sequences do not match each other, then we can continue only if we enter sequence of same length. Snapshot – 3 provides a sample output when the sequences do not match each other.

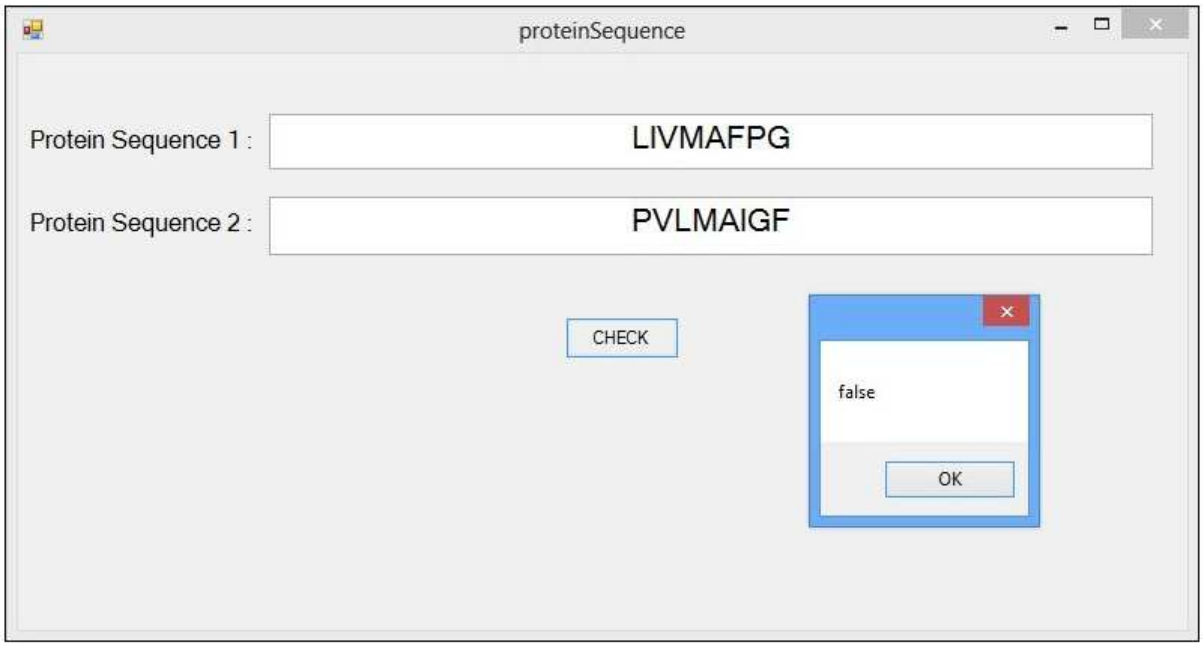


Snapshot – 3

A sample output for the sequences discussed in the illustration is seen in Snapshot – 4 and Snapshot – 5.



Snapshot – 4



Snapshot – 5

CONCLUSION

Amino acid sequence alignment is important as it helps in mutation determination. It is important to determine this to analyze genetic disorders. This method uses simple linear search method only and hence can be used for initial verification if the sequences match each other. The maximum length of string. net frameworks can accommodate is 2147483647 using super computers. A normal 4gb ram system can accommodate around 10^9 characters. Hence this proposed method can definitely be used for initial sequence verifications.

REFERENCES

[1] S Henikoff and J G Henikoff, Amino acid substitution matrices from protein blocks, Proc Natl Acad Sci U S A. **1992** Nov 15; 89(22): 10915–10919.
[2] file:///C:/Documents%20and%20Settings/Dell/My%20Documents/Downloads/00b4951 953190e7d73000000.pdf

[3] Wu TD[1], Brutlag DL., Discovering empirically conserved amino acid substitution groups in databases of protein families., Proceedings, International Conference on Intelligent Systems on Molecular Biology. **1996**;4:230-40**.**
[4] https://msdn.microsoft.com/en-IN/library/z1zx9t92.aspx
[5] https://msdn.microsoft.com/en-us/library/ms973898.aspx.
[6] biochem.stanford.edu/biochem218/Projects%202001/Yu.pdf.