AUTOMATIC IDENTIFICATION OF EVOLUTIONARY AND SEQUENCE RELATIONSHIPS IN LARGE SCALE PROTEIN DATA USING COMPUTATIONAL AND GRAPH-THEORETICAL ANALYSES

A Thesis Submitted to
the Graduate School of Engineering and Sciences of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in Bioengineering

by
Tunca DOĞAN

December 2012
İZMİR

We approve the thesis of **Tunca DOĞAN**

Examining Committee Members:

Assoc. Prof. Dr. Bilge KARAÇALI
Department of Electrical and Electronics Engineering,
İzmir Institute of Technology

Prof. Dr. Meral SAKIZLI
Department of Medical Biology and Genetics,
Dokuz Eylül University

Assoc. Prof. Dr. Jens ALLMER
Department of Molecular Biology and Genetics,
İzmir Institute of Technology

Assoc. Prof. Dr. Cemal ÜN
Department of Biology, Ege University

Assist. Prof. Dr. Devrim Pesen OKVUR
Department of Molecular Biology and Genetics,
İzmir Institute of Technology

11 December 2012

Assoc. Prof. Dr. Bilge KARAÇALI
Supervisor, Department of Electrical
and Electronics Engineering,
İzmir Institute of Technology

Prof. Dr. Hüseyin BASKIN
Co-Supervisor, Department of
Microbiology, Dokuz Eylül University

Assoc. Prof. Dr. Volga BULMUŞ
Head of the Department of
Biotechnology and Bioengineering

Prof. Dr. R. Tuğrul SENGER
Dean of the Graduate School of
Engineering and Sciences

ACKNOWLEDGEMENTS

First of all, I would like to express my sincere appreciation to my advisor, Dr. Bilge Karaçalı for his guidance, support, trust and encouragement throughout my PhD study. His vast scientific knowledge and convenient analytical solutions to every problem we faced during the study have always amazed me. Not to mention the comfortable and unconfined working environment he provide for us. It was essential for an independent person like me.

I would also like to thank my co-advisor Dr. Hüseyin Baskın and committee members Dr. Meral Sakızlı and Dr. Jens Allmer their valuable contributions directed our study.

I'm also grateful to my fellow lab friends Başak Esin Köktürk and Devrim Önder both for the enjoyable scientific discussions and their friendship. Biplab was a small but quite a synergistic environment. Also additional thanks to Başak for her help and assistance that ease my working life at IYTE.

I also thank all of my friends in IYTE especially from the Electrical and Electronics Engineering Department with whom I shared my working life for more than 3 years.

I especially would like to thank my father Dr. Kasım Doğan for his support and inspiration for becoming a scientist, my mother Güzin Doğan for her support and help whenever I need, my grandmother Melahat Bıçakçı especially for the delicious meals she always prepared and my brother İlhan Günay Doğan for being an eccentric, original guy and enjoyable conservations.

And finally, I would like to express my eternal love and deepest appreciation to my beloved wife Gizem Umut Doğan. Without her love, encouragement, inspiration and all the other aspects of the life we share, I wouldn't have the strength and enthusiasm to complete my thesis study.

ABSTRACT

AUTOMATIC IDENTIFICATION OF EVOLUTIONARY AND SEQUENCE RELATIONSHIPS IN LARGE SCALE PROTEIN DATA USING COMPUTATIONAL AND GRAPH-THEORETICAL ANALYSES

In this study, computational methods are developed for the automatic identification of functional/evolutionary relationships between biomolecular sequences in large and diverse datasets. Different approaches were considered during the development and optimization of the methods. The first approach focused on the expression of gene and protein sequences in high dimensional vector spaces via non-linear embedding. This allowed statistical learning algorithms to be applied on the resulting embeddings in order to cluster and/or classify the sequences. The second approach revised the pairwise similarities between sequences following multiple sequence alignment in order to eliminate the unreliable connections due to remote homology and/or poor alignment. This is achieved by thresholding the pairwise connectivity map over 2 parameters: the inferred evolutionary distances and the number of gapless positions in each pairwise alignment. The resulting connectivity map was disjoint and consisted of clusters of similar proteins. The third and the final approach sought to associate the amino acid sequences with each other over highly conserved/shared sequence segments, as shared sequence segments imply conserved functional or structural attributes. An automated method was developed to identify these segments in large and diverse collections of amino acid sequences, using a combination of sequence alignment, residue conservation scoring and graph-theoretical approaches. The method produces a table of associations between the input sequences and the identified conserved regions that can reveal both new members to the known protein families and entirely new lines. The methods were applied to a dataset composed of 17793 human proteins sequences in order to obtain a global functional relation map. On this map, functional and evolutionary properties of human proteins could be found based on their relationships to the ones bearing functional annotations. The results revealed that conserved regions corresponded strongly to annotated structural domains. This suggests the method can also be useful in identifying novel domains on protein sequences.

ÖZET

BÜYÜK ÇAPLI PROTEİN VERİSİNDE EVRİMSEL VE DİZİNSEL İLİŞKİLERİN İŞLEMSEL VE ÇİZGE TEORİSİ ANALİZLERİ İLE OTOMATİK OLARAK BELİRLENMESİ

Bu çalışmada, yüksek oranda çeşitlilik gösteren geniş veri setlerinde bulunan biyomoleküler dizilerin evrimsel/fonksiyonel ilişkilerini otomatik şekilde tanımlayan yöntemler geliştirilmiştir. Yöntemlerin oluşturulması ve optimizasyonu sırasında farklı yaklaşımlar değerlendirilmiştir. İlk yaklaşım, doğrusal olmayan gömme tekniği kullanılarak, gen ve protein dizilerinin çok boyutlu vektör uzaylarında ifade edilmeleri olmuştur. Bu yaklaşım, sonuç olarak ortaya çıkan ifadeleri kümelemek ve/veya sınıflamak amacı ile istatistiki öğrenme algoritmalarının uygulanabilmesine olanak sağlamıştır. İkinci yaklaşım, uzak homoloji ve/veya yanlış hizalama sonucunda ortaya çıkan güvenilmez bağlantıları elemek amacı ile diziler arası ikili uzaklıkları düzeltme işlemine tabi tutmak olmuştur. Bu işlem, ikili bağlantı haritasının farklı 2 değişken üzerinden eşiklenmesi ile gerçekleştirilmiştir. Bunlar, tahmin edilen evrimsel mesafeler ve ikili hizalamalarda yer alan boşluksuz pozisyonların sayısı olmuştur. Sonuç olarak ortaya çıkan bağlantı haritası, kopuk ve benzer proteinler içeren kümelerden oluşmaktadır. Üçüncü ve son yaklaşım, paylaşılan dizi parçalarının korunmuş fonksiyonel veya yapısal özellikleri ifade etmelerinden dolayı, amino asit dizilerinin bu paylaşılan/korunmuş kısımlar üzerinden birbirleri ile ilişkilendirilmeleri olmuştur. Bu kısımların çeşitlilik içeren geniş amino asit dizi koleksiyonlarında tanımlanabilmesi amacı ile dizi hizalama, amino asit korunum puanlama ve çizge teorisi yaklaşımları kullanılarak otomatik çalışan bir yöntem geliştirilmiştir. Yöntem, çıktı olarak işleme verilen diziler ile tanımlanan korunmuş bölgelerin ilişkilendirildiği bir tablo vermektedir. Bu tablo kullanılarak hem bilinen protein ailelerinin yeni üyeleri, hem de tamamen yeni aileler ortaya çıkarılabilir. Geliştirilen yöntemler, genel bir fonksiyonel ilişki haritası elde etmek amacı ile 17793 insan protein dizisinden oluşan bir veri setine uygulanmıştır. Bu harita üzerinde, fonksiyonel açıklamalar içeren proteinler ile ilişkileri dikkate alınarak, insan proteinlerinin fonksiyonel ve evrimsel özellikleri elde edilebilir. Sonuçlar, korunmuş bölgelerin tanımlanmış yapısal fonksiyonel dizi kısımlarına denk düştüğünü göstermiştir. Buna bağlı olarak, yöntem aynı zamanda protein dizileri üzerinde yeni yapısal fonksiyonel dizi kısımlarının tanımlanmasında kullanılabilir.

TABLE OF CONTENTS

LIST OF FIGURES

Figure **Page**

LIST OF TABLES

CHAPTER 1

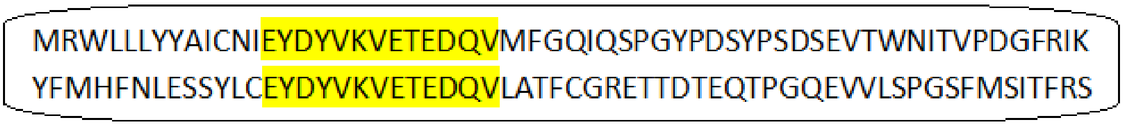
INTRODUCTION

1.1. Biomolecular Sequences and Their Analysis

Large amounts of data on the molecular attributes of living organisms are being accumulated in databases following the availability of molecular scanning tools over the last decades. Methods from different fields of science are being applied to make sense out of this huge amount of data as well. A considerable amount of this biological data consists of molecular sequences. Since they contain crucial information on molecular functions and evolutionary relationships, biomolecular sequences remain of primary interest for the researchers in the area. Most frequently studied sequences within the field are genes, that directly reflect the hereditary properties, and gene products (mostly proteins) that exhibit molecular functions.

Biomolecular sequences carry the information that make up the whole structural properties and the metabolism of an organism. During the course of evolution, each one assumes a specific task. As a result, they are highly diverse. Yet, some of them share common features. These features emerge as statistically significant similarities on different regions of the sequences. These significant similarities often indicate mutuality in the history of these sequences and/or commonality in metabolic functions. These regions remain relatively unchanged as the mutation rate acting on these segments is usually relatively low compared to the rest of the sequences: The shared regions tend to correspond to segments with molecular functions, and changes in their structure due to mutations may cause the loss of these functions, decreasing the organism’s fitness in the population. As a result, the altered sequence usually dies out due to the disadvantages brought by the mutation. The sequences remaining in the population contain the unchanged functional region, shared between these sequences.

The information regarding the shared regions is an important instrument for the discovery of the molecular processes taking place during the life of an organism. Figure 1 shows a conserved region (highlighted in yellow) between two sample amino acid sequences. Notice the similarity of amino acids in the highlighted positions as opposed to the contrast in the rest of the positions. Since most of the information regarding the functions of biomolecular sequences is located on these shared regions, this topic is a highly active area of research, aiming to identify and classify them using both experimental and computational efforts. These shared regions referred to by two widely used terms in the literature; motifs and domains.



MRWLLLYYAICNIEYDYVKVETEDQVMFGQIQSPGYPDSYPSDSEVTWNITVPDGFRIK

YFMHFNLESSYLCEYDYVKVETEDQVLATFCGRETTDTEQTPGQEVVLSPGSFMSITFRS

Figure 1. A conserved region (in yellow color) between 2 amino acid sequences.

A motif is a broad term referring to sequence segments usually composed of a few nucleotides or amino acids that possess biological significance. The term is often used for short and highly repetitive segments on sequences (D'haeseleer, 2006). Motifs have crucial roles in the metabolism such as phosphorylation, transcription termination, or acting as DNA binding sites (D'haeseleer, 2006). Identification of motifs on biomolecular sequences is an active field of study. There are many methods proposed in the literature to this purpose. Most of these methods take a set of sequences and try to extract some form of conservation, and compare it to a database of motifs. Some widely used methods are MEME Suite (Bailey *et al.*, 2009), Gibbs Motif Sampler (Thompson *et al.*, 2007) and Minimotif Miner (Rajasekaran *et al.*, 2010).

A structural domain refers to a part of a protein sequence that can function, evolve and fold independent from the rest of the protein (Phillips, 1966; Wetlaufer, 1973). Structural domains can be up to 600 amino acids in length and are usually highly specific. Each domain has a specific molecular function and a protein’s role in the metabolism is directly determined from the domains it contains. With their discovery and qualitative comprehension, studies on the functions of proteins are now performed on the domain level and vast protein domain databases are constructed. Within these databases, domains are grouped under different names, rules and regulations. Protein families (Pfam) (Finn *et al.*, 2010 and Punta *et al.*, 2012), NCBI Conserved Domain Database (Marchler-Bauer *et al.*, 2003), SCOP: Structural Classification of Proteins (Andreeva *et al.*, 2008), CATH Protein Structure Classification (Cuff *et al.*, 2011) and Simple Modular Architecture Research Tool (SMART) (Schultz *et al.*, 1998) are some examples. A representation of the sequence and the structure of the multi-domain protein pyruvate carboxylase are given in Figure 2 (a) and Figure 2 (b). Pyruvate carboxylase (PC) is a crucial enzyme for the metabolism that catalyzes the MgATP dependent and HCO_3^- carboxylation of pyruvate to form oxaloacetate (Jitrapakdee *et al.*, 2008). In Figure 2, the domains are represented by different colors, blue for Biotin Carboxylation (BC) domain, yellow for Carboxyltransferase (CT) domain, red for Biotin Carrier (BCCP) domain, and green for allosteric domain. Each of these domains contributes in different parts to the protein's function. Notice the differences in the 3-D representation of domains in the figure, revealing the structural differences between these domains.

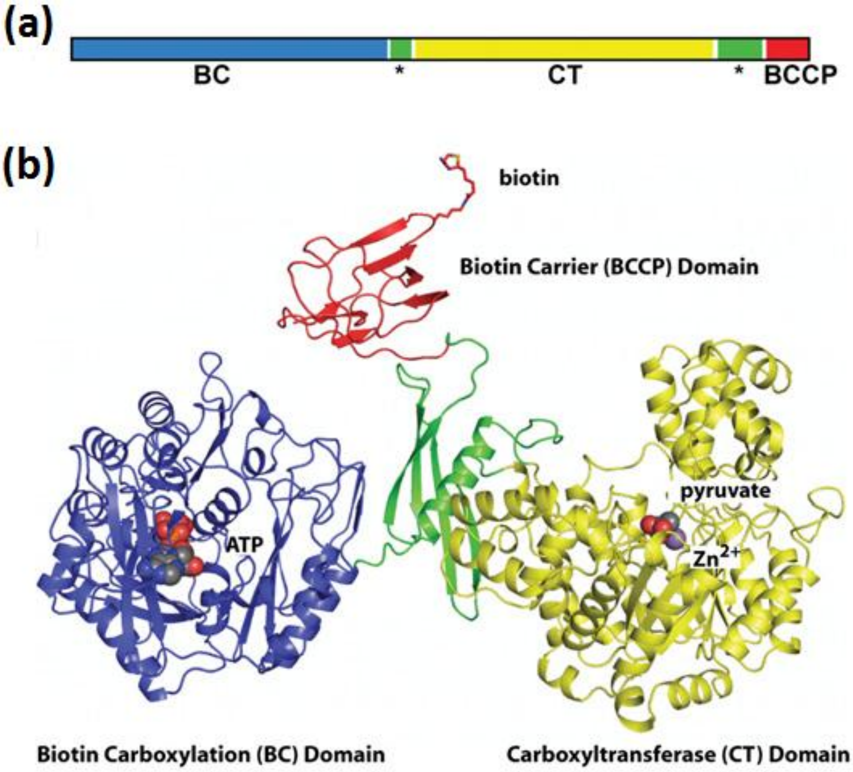

(a)
BC
*
CT
*
BCCP
(b)
biotin
Biotin Carrier (BCCP) Domain
ATP
pyruvate
Zn^{2+}
Biotin Carboxylation (BC) Domain
Carboxyltransferase (CT) Domain

Figure 2. (a) Schematic drawing of the sequence (with domains highlighted) for pyruvate carboxylase (b) The structure of the *Staphylococcus aureus* pyruvate carboxylase (blue: BC domain, yellow: CT domain, red: BCCP domain and green: allosteric domain). (Source: Jitrapakdee *et al.*, 2008)

Motifs and domains are highly conserved during the evolutionary process since changes in their structure may cause the loss of vital molecular functions. In order to extract these highly conserved/shared sequential features, biomolecular sequences should be compared to each other. However, biomolecular sequences carry a vast amount of information and due to evolution, most of them are highly diverged from each other over the course of time, and it is practically impossible to analyze them without using proper statistical methods.

The concept of sequence alignment was first applied to molecular biology decades ago to study the compositions of complex sequences in a comparative manner. Alignment methods aim to uncover shared features between sequences of interest by identifying their molecular similarities. The Needleman-Wunsch global alignment (Needleman and Wunsch, 1970) and the Smith–Waterman local alignment (Smith and Waterman, 1981) algorithms are two basic tools used primarily to that end. Many current sophisticated sequence analysis methods are based on these two pairwise alignment algorithms. Multiple sequence alignment algorithms are used for generating consensus sequences over a given collection, and they are also based on pairwise alignment. A classical multiple sequence alignment operation basically consists of 2 steps: The first one is the all-against-all pairwise alignment of the input sequences. The second step is the progressive formation of the multiple alignment by gradually introducing the sequences to the growing chain, including the gaps inserted during the pairwise alignment step. Unlike pairwise alignments, however, optimal solution is not guaranteed in the multiple sequence alignment. Clustal family tools -one of the most popular multiple sequence alignment methods- (Larkin *et al.*, 2007) are for general use to align both nucleotide and amino acid sequences, and belong to the class of progressive alignment methods. ClustalW (Larkin *et al.*, 2007) is also used for phylogenetic tree construction. MUSCLE incorporates iterations during which distance measures are refined, and results in more accurate alignments (Edgar, 2004). T-COFFEE (Notredame *et al.*, 2000), another popular multiple sequence alignment method, uses the output from Clustal and local alignments to improve weighing factors. MAFFT (Katoh and Kuma, 2002) produces alignments in reduced computation times employing the Fast Fourier Transform (Brigham, 2002).

The computational load of multiple sequence alignments is substantial, especially when the number of sequences in the datasets is high. Still results can be obtained in reasonable computation times for moderate numbers of sequences.

1.2. Evolutionary Relationships

Exploring evolutionary relationships between genes and proteins is crucial for discovering the physiological and molecular mechanisms that govern their system. This is achieved by taking account of the molecular similarities and differences between gene and protein sequences. In other words, phylogenetic methods aim to uncover the mutual history of these sequences using sequence analysis tools and statistical techniques.

The results of a phylogenetic analysis are usually displayed in the form of phylogenetic trees. These trees are branching diagrams projecting the inferred evolutionary relationships between the entities of interest. Leaf nodes at the tip of branches on these trees represent the samples (i.e. organisms or sequences). In a rooted tree, branch lengths are often correlated with time. The evolutionary time flows in the direction from internal nodes towards the leaves. The internal nodes (inside the trees) represent hypothetical common ancestors. Any two taxa bifurcating from the same internal node are assumed to be descended from the common ancestor represented by that node. An unrooted tree, however, represents the relations of leaf nodes without inference about common ancestry and time between the internal nodes.

Tree estimation methods are divided in two groups, as character based and distance based methods. A character here represents an attribute that the input samples vary upon. Maximum Parsimony is an example of the character based methods. In this method, many different possible trees are investigated in order to determine the optimum one that requires “the least number of evolutionary changes to explain the observed data” (Felsenstein, 1978). On the other hand, distance methods are based on calculating the sequence divergence between the gene or protein sequence pairs and inferring the evolutionary distances in-between via a mathematical model. Most widely used models for nucleotide sequences are the Jukes-Cantor Model (Jukes and Cantor, 1969), the Kimura’s Two Parameter Model (Kimura, 1980) and the General Time Reversible Model (Tavare, 1986). Also for amino acid sequences, the Poisson distribution (Ahrens and Dieter, 1974), the Kimura Amino Acid Substitution Model (Kimura, 1983) and the Dayhoff Model (Dayhoff *et al.*, 1978) are employed frequently to characterize the substitution structure between the different amino acids.

1.3. Problem Description

The key to understanding all aspects of life resides within the complex molecular data characterizing the living organisms. In the last decades, large quantities of molecular sequence data have been produced thanks to the technological improvements in molecular sequencing. However, a major part of this data is yet to be interpreted by the specialists. Since the sequencing of the human genome (Venter *et al.*, 2001); functions and interactions of genes and its products are being studied extensively as these are key to developing novel medical solutions to prevalent diseases and other medical complications.

Apart from expensive and laborious experimental studies, fundamentals of bioinformatics are applied to the case to construct answers to outstanding questions. Statistical approaches are used to seek significant similarities between functionally known (experimentally proven) and unknown sequences. Products of these studies are computational tools that operate mostly on nucleic acid and amino acid sequences.

Two of the outstanding issues concerning these tools are the lack of generality and standardization in the parameters. First of all, a major part of these tools are developed to process molecular data with very specific properties. This is a prevalent problem especially when the data consist of large and diverse sets. Besides, this also requires substantial preliminary information about the data, which is not possible in some cases. The second problem concerns the determination of the optimal configuration of the input parameters for a specific application. In many tools, the user is asked successive questions about the values of parameters, answers to which are not necessarily known in advance. The random selection of parameters (or the use of default ones) often concludes with poor and unreliable results. Lack of adaptive and automatic parameter selection rules also defeats the purpose of these tools, which is revealing the unknown features of the data: In most cases, the user cannot answer all of these detailed questions about the input data, and if they could, they would not have needed to use the tool in the first place.

An additional issue associated with functional clustering methods apart from parameter standardization is related to clustering performance. When applied to datasets other than the ones they are optimized for, many of these methods exhibit poor clustering performances. A contributing factor to poor clustering performance is the tendency to measure sequence similarities over their entire length. However, a biomolecular sequence is composed of segments with different properties. The regions with functional/evolutionary signatures are more significant when considering similarity searches. Evolution usually acts on these regions at lower rates compared to the rest of the sequences, as mutations in functional regions often cause the loss of associated function. However, mutations outside these regions do not affect the vital functions to that extent and the fitness of the sequences carrying these mutations can increase in the population. Consequently, the regions without functional properties are usually highly variable. When the sequences are compared over their entire length, these highly variable uninformative regions cloud the similarity that should be better focused on the comparison of functional regions, giving unreliable clustering results.

1.4. Objectives of the Study

The main objective of this study is to develop new methods for automatic extraction of sequential features from large biomolecular sequence datasets. These features correspond to functions and evolutionary relationships inferred from the similarities of the unknown sequences to the known ones. Within this objective, we also aim to eliminate the issue of parameter standardization and to construct universal methods with the ability to process large datasets composed of sequences with diverse features, such as whole proteomes.

In our study, we prefer to apply our methods on a large dataset composed of thousands of human protein sequences to obtain a global functional/evolutionary relation map on which the properties of the unknown sequences can be inferred in order to make predictions on protein function, to be used towards a general understanding and the discovery of the biological mechanism of the human proteome.

1.5. Organization of the Thesis

Here in this thesis, the efforts regarding the objectives of the study are presented under three main chapters following the Introduction in Chapter 1. Chapter 2 presents the use of vector space embedding process in the analysis of biomolecular datasets. Following the Multiple Sequence Alignment and the inference of the evolutionary distances, the sequences were expressed in high-dimensional vector spaces, and statistical methods were used to cluster and classify them. The method was modified and optimized in order to fit the biomolecular data, before it was applied to a large human protein dataset.

Chapter 3 explains the construction and subsequent thresholding of the connectivity map of the input dataset in order to eliminate the unreliable evolutionary distances. The thresholding was carried out regarding 2 features, namely, the inferred evolutionary distances and the number of gapless positions on pairwise comparisons of the alignment. The method was applied on the human protein dataset and the accuracy of the recovered connections was evaluated based on the curated functional annotations from sequence databases.

Chapter 4 details the method developed for protein function assignment by conserved region identification and association. The method employs sequence alignment, statistical grouping and conserved residue scoring; and produces a table presenting the associations between the recovered conserved regions and the input sequences along with conserved region profiles. The biological relevance of the method was assured with a clustering on gold standard datasets from the literature. The finalized method was applied to the human protein dataset and the functional relations were obtained between groups of proteins. In addition, the correspondence between the conserved regions and the curated structural domain assignments of the proteins were investigated to evaluate the performance of the method in identifying functional domains in amino acid sequences in an exhaustive and automated fashion.

CHAPTER 2

EVOLUTIONARY RELATIONSHIPS BETWEEN GENE AND PROTEIN SEQUENCES VIA NON-LINEAR EMBEDDING

A problem related to the inference of evolutionary relationships and phylogenetic trees is the calculation of evolutionary distances (mentioned in the last paragraph of Section 1.2). The evolutionary models provide useful information on the subject matter, but this information is subjected to errors that decrease the credibility of the results especially when the divergence is elevated. As shown in Equation 2.3 in Section 2.4 for Jukes-Cantor Model (Jukes and Cantor, 1969), the upper limit of the sequence distance is 0.75 where the inferred evolutionary distance becomes infinity. Moreover, as the sequence distance approaches the upper limit, the error in the resulting inference increases. This often results in phylogenetic trees with false relationships.

The aim of this part of the study is to develop a method that has the ability to infer the accurate evolutionary relationships and classify/identify gene and protein families via expressing the input sequences in high dimensional vector spaces upon which statistical learning algorithms can be applied.

With the application of vectorial expression, test samples are given unique positions in high dimensional spaces regarding their relative pairwise distances. Since complex biological sequences (our samples) are reduced to points with known coordinates, statistical methods can easily and efficiently be applied to discover their relations with each other. In a vectorial embedding procedure, number of points in the final embedding will naturally equal to the number of sequences in the dataset. Number of vector space dimensions equals to the length of the Multiple Sequence Alignment output of this dataset. In other words, each position in the alignment corresponds to a dimension where the value of sample on this dimension comes from the relative constitution of the sequences at that position. For example, if any 2 sequences have the same nucleotide on the position *x*, then these 2 sequences will have the same value at the dimension *x* in the vectorial embedding.

However, usually very high dimensional loads arise from the process since all positions (in the alignments) are taken into account leading to computational problems. At the same time, it's highly probable that some of the positions do not contribute to the evolutionary relations between the samples as much as the others (the ones experiencing high mutational rates). Apart from the heavy computational load, these positions add noise, clouding the information we seek. Reducing the vector space dimensions by only incorporating the positions with informative evolutionary signals yields better representations. This is done employing dimensionality reduction methods. Multi-Dimensional Scaling (MDS) (Kruskal and Wish, 1978) and Principle Component Analysis (PCA) (Jolliffe, 1986) are two of the popular ones in use for a long time. PCA separates correlated input variables into components orthogonal to each other. These components are ranked in the order of informativeness regarding the distribution of the input variables. C-MDS takes pairwise dissimilarity measures between test samples and positions the samples in multi-dimensional spaces regarding the minimization of a cost function. These classical methods are capable of producing successful results in relatively simple cases especially when the structure of the data is distributed on a linear plane but they usually cannot capture the nonlinear structures present in the data (Tenenbaum *et al.*, 2000). In order to capture the non-linearity in the systems, non-linear embedding methods are developed by adapting the classical techniques. Nonlinear mapping (NLM) (Sammon, 1969) method tries to capture the nonlinearity by distributing different weights to the distances inversely proportional to their values. Locally Linear Embedding (LLE) (Roweis and Saul, 2000) produces global maps with the help of local symmetries. Another novel method called Stochastic Proximity Embedding (SPE) (Agrafiotis and Xu, 2002) builds on the same geodesy as LLE but it circumvents the calculation of approximate geodesic distances between remote points and forms an arrangement that scales linearly with the number of data points used.

Stochastic Proximity Embedding has been used for uncovering the evolutionary relationships between protein sequences. In a study, SPE is applied in order to group the functionally related proteins in meaningful clusters (Farnum *et al.*, 2003). As a result, some of the proteins that share similar functions are grouped in a two dimensional space away from evolutionary distant proteins. This study revealed the potential of non-linear embedding methods in clustering gene and protein sequences.

Here we propose that, by expressing the gene or protein sequences of interest in high dimensional vector spaces via non-linear embedding using the inferred pairwise distances as the input, the errors present may be reduced significantly. Moreover, different statistical methods may be applied on these vectorial embeddings in order to cluster or classify these sequences.

ISOMAP (Isometric Feature Mapping) is a nonlinear mapping method that makes use of geodesic distances induced by neighborhood graph embedded in the classical scaling (Tenenbaum *et al.*, 2000). This provides the advantage of capturing the structure of nonlinear -curved- manifolds successfully. Input space distances are good approximations to the geodesic distances in neighboring points (Tenenbaum *et al.*, 2000). For far away points, many small jumps between these neighboring points can be added up to approximate the geodesic distance. ISOMAP method is frequently employed in geological and meteorological studies. The magnitude of the conserved neighborhood size is adjustable. Under the threshold, the method preserves all pairwise distances whereas over the margin, all distances are discarded and estimated again by the method. One indicator of the success of the embedding is the residual variance plot. It is simply a measure of the difference between the distances from the revised matrix and the ones at the final embedding. It is crucial here to clarify that this measure is not anyway related to the biological accuracy of the results in our case, it is just an indicator of the statistical success of the operation done.

Another parameter arise from the vectorial embedding process is the number of dimensions in the final embedding. It should be chosen as the lowest number that is sufficient to express the data successfully. As the number of dimensions increase, the magnitude of the residual variance decreases. Approximately, the number of dimensions at the onset of the flat region of the residual variance curve is the optimum. Also the magnitude of the residual variance at this flat region should not exceed 0.1 or else the embedding is considered to be unsuccessful.

In this part of the study, Isometric Feature Mapping (ISOMAP) algorithm is used in order to embed gene and protein sequences in high dimensional vector spaces and various classification methods are applied on these vectorial arrangements. Using this procedure, meaningful clusters -regarding functional similarities- are obtained.

2.1. Methods

The information given in this section is taken from (Tenenbaum *et al.*, 2000). The algorithm takes pairwise distances between the samples as input and assigns each of them a unique location in multi-dimensional vector spaces. When the input also consists of points in a high-dimensional space and the aim of using ISOMAP is the reduction of dimensions, Euclidean distances between the samples are computed and given to the algorithm. In the output space, Euclidean distances between the points correspond to their corrected pairwise distances. The placement of vectors in this high-dimensional space is carried out in such a way that the output embedding represents the intrinsic geometry of the data as accurately as possible.

The complete ISOMAP process consists of pre-processing step and 3 main steps. In the pre-processing step, distances $d_x(i,j)$ between all pairwise combinations i and j (from a total of N samples) are measured using Euclidean metric. The pre-processing step is not executed when the input already consist of distances. The first step of the algorithm is the construction of the neighborhood graph. The algorithm carries out the process with either one of the two different approaches. First one is connecting each point to all other points within a fixed radius named as epsilon (ϵ) and the second one is connecting each point to its K nearest neighbors. The user is required to select one of these options (ϵ-Isomap or K-Isomap) and its value at the input level. In this step, neighborhoods between the input samples are determined by defining the graph G on all input samples by connecting points with an edge i and j if $d_x(i,j)$ is less than ϵ or j is one i's K nearest neighbors. The edge length is set as $d_G(i,j) = d_x(i,j)$ in the presence of an edge, and the remaining of the pairwise connections are equalized to infinity ($d_G(i,j) = \infty$). The second step is the computation of the shortest paths. For each value of $k = 1, 2, 3, ..., N$ entries, i and j, $d_G(i,k)+d_G(k,j)$ is computed, then $d_G(i,j)$ are replaced by minimum of these 2 expressions (Equation 2.1). As a result, the final matrix contains the shortest path (graph) distances (D_G). The third and the last step is the construction of d-dimensional embedding (Y). It is basically the application of Classical Multi-Dimensional Scaling (C-MDS) to the graph distances found in the previous step.

$$D_G = \min\{d_G(i,j), d_G(i,k) + d_G(k,j)\} \tag{2.1}$$

D_G is the final graph distance matrix, d_G represents the pairwise neighborhood graph distances between any point pair, i, j and k are the points corresponding to the samples in the dataset.

Residual variance plot is employed in order to select the correct number of dimensions (d) in the output embeddings. This is the least number of dimensions that represent the geometric structure of the data accurately. True dimensionality of the data can be estimated from this plot regarding the decrease in error with increasing number of embedding dimensions. This measure is also employed in other dimensionality reduction techniques such as PCA and MDS to obtain the true dimensionality. Equation 2.2 shows the formulation of residual variance.

$$V_R = 1 - R^2\left(\widehat{D}_M, D_Y\right) \tag{2.2}$$

V_R represents the residual variance, D_Y is the matrix of Euclidean distances that the algorithm returns, D_M is the best estimate of the intrinsic manifold distances (graph distance matrix for ISOMAP) and R represents the standard linear correlation coefficient.

Figure 3 shows the true distribution of the “Swiss roll” dataset. As observed from the figure, the data is distributed upon 3 dimensions. Gray/black points represent the samples on the 3-D space. Blue dashed line is the Euclidean distance between 2 points (in black circles) in the space whereas the uninterrupted blue line represents the true geodesic distance between the same points. Notice how the line follows the non-linear (curved) structure of the distribution. The aim of employing ISOMAP here is capturing this path with a lower number of dimensions.

Figure 4 (a) shows the neighborhood graph (G) constructed on the same distribution (on 3-D space) using the K-Isomap procedure with the neighborhood size of 7. Gray lines represent edges between neighboring points and the red line represents the graph distance between the same points referred in Figure 3, in other words, the shortest path between these points in G. Figure 4 (b) shows the vector space embedding after ISOMAP process on 2 dimensions. The red line is the same graph distance shown in Figure 4 (a) and the blue line is the final approximation to the true geodesic path between the aforementioned points. Notice that, the true structural distance between these far-away points were approximated successfully by many small jumps between the neighboring points.

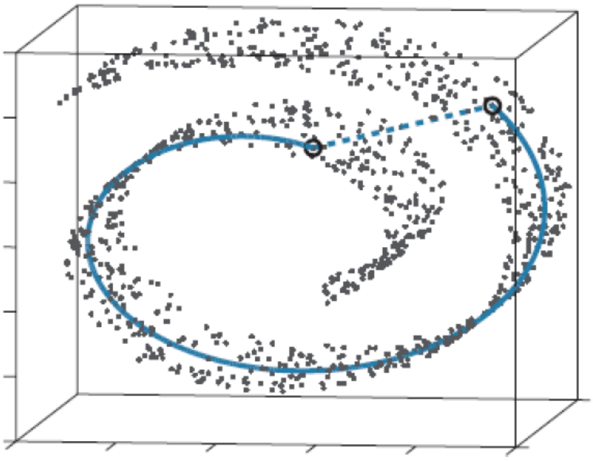


Figure 3. The distribution of the points in the “Swiss roll” dataset in 3-D space. (Source: Tenenbaum *et al.*, 2000)

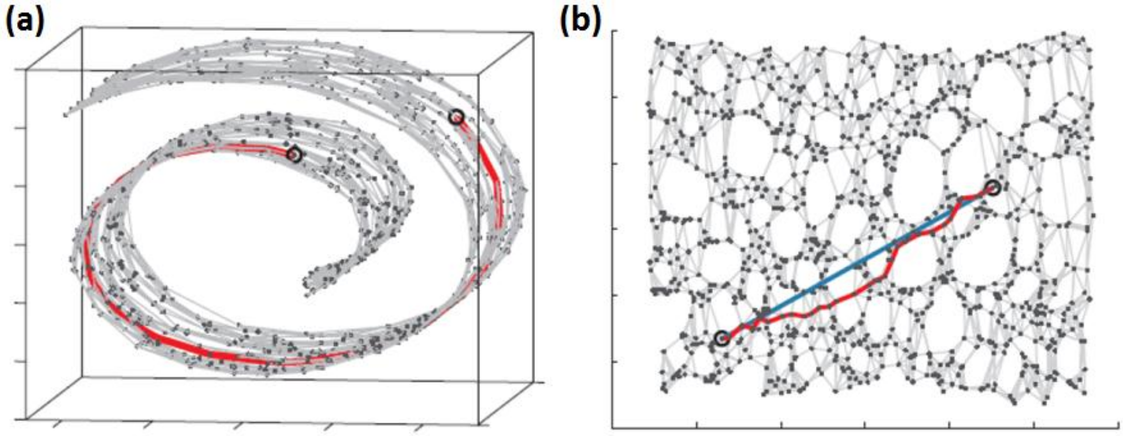


Figure 4. (a) The neighborhood graph (G) constructed on the “Swiss roll” dataset distribution (b) 2-D embedding output of ISOMAP on “Swiss roll” dataset. (Source: Tenenbaum *et al.*, 2000)

Figure 5 shows the residual variance plot of various dimensionality reduction methods for the embedding the “Swiss roll” dataset. The horizontal axis represents the number of embedding dimensions and the vertical axis represents the residual variance values. The filled black arrow at the bottom of the figure indicates the reduced number of dimensions required to accurately represent the distribution in ISOMAP process. Linear dimensionality reduction methods failed to reduce the number of dimensions whereas ISOMAP capture the non-linearity in the manifold and embed the points in 2 dimensions with nearly zero residual variance.

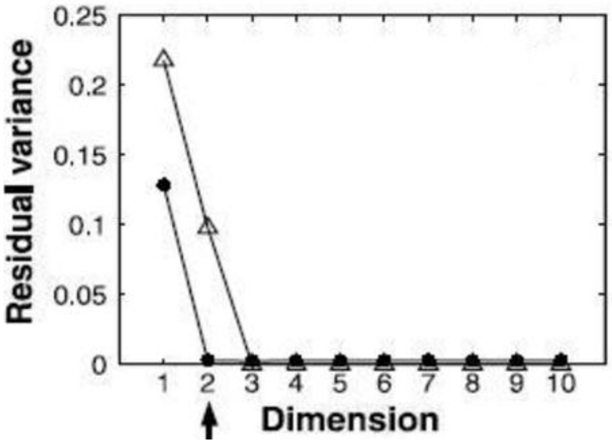


Figure 5. Residual variance plot for the embedding of "Swiss roll" dataset (open triangles: PCA and MDS, filled circles: ISOMAP). (Source: Tenenbaum *et al.*, 2000)

2.2. Embedding of MAPK1 Gene Sequences

First of all, in order to test the method on a crude basis and to get an idea about its potential in clustering gene sequences, MAPK1 gene sequences from different animals (human, wolf, cattle, chimpanzee, macaque, mouse, two different frog species, chicken and a bird) are downloaded from NCBI database. The product of MAPK1 gene is a protein acting as an integration point for multiple biochemical signals which is involved in a wide variety of cellular processes such as proliferation, differentiation, etc. MAPK1 gene is quite conserved through evolutionary timeline. The dataset contains gene fragments from the database along with full sequence genes. The reason for including the sequence fragments was to observe how the method behaves when there are un-alignable sequences in the dataset where the sequence alignment methods usually fail. The dataset contained a total of 20 sequences.

Following the multiple sequence alignment, sequence distances between sequence pairs were calculated, evolutionary distances were inferred by Jukes-Cantor model (input for ISOMAP), non-linear embedding of the sequences in a multi-dimensional vectorial space was carried out and a visual representation of evolutionary relationships in a 2-D or 3-D vector spaces were sought. In order to compare the results with a tree formed regarding the output alignment, a dendrogram was created in EMBL-EBI website using online tree formation tool following the multiple sequence alignment procedure.

Figure 6 shows the rooted dendrogram of sequences present in the dataset which is drawn via the tree formation option of EMBL-EBI Multiple Sequence Alignment tool. Different colors are used in order to ease the understanding. Yellow color is used for bird and chicken, blue color for mouse and the fragments of the mouse MAPK1 gene, red color for primates and fragments of human MAPK1 gene, green color for the 3 different frogs and grey color for the cattle and the wolf. As seen from this figure, the evolutionary relationships are quite wrong. Especially gene fragments are associated with the other sequences inaccurately. The reason behind this case is that these methods try to infer a relation between all sequence pairs and use these inferences to draw trees. As a result, when there are un-alignable sequences in the dataset, it is usually not possible to infer the phylogenetic relationships.

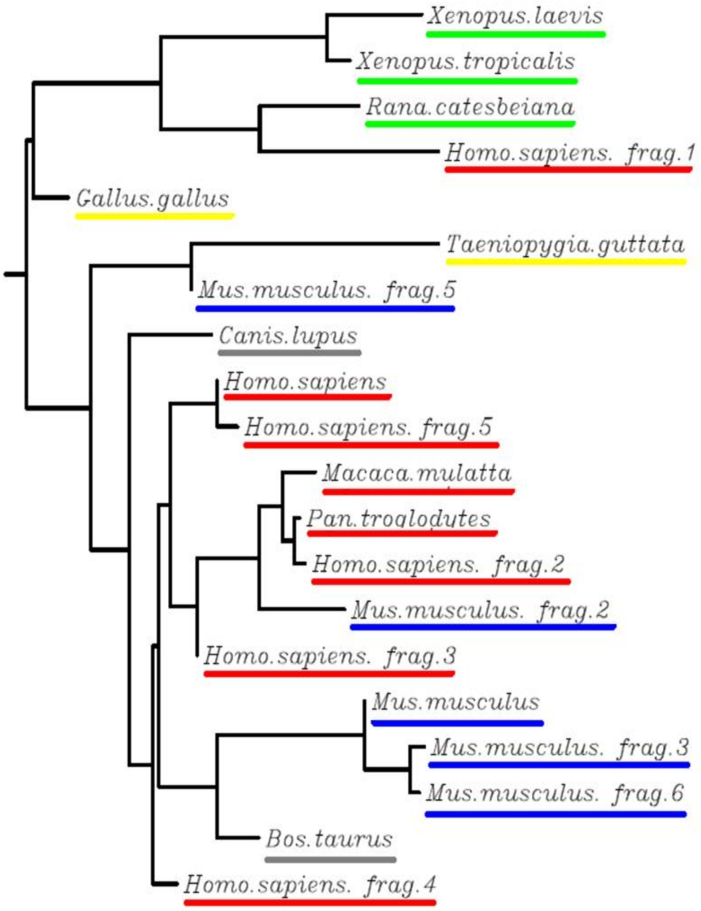


Figure 6. Rooted dendrogram drawn by online tree formation tool on EMBL-EBI website for 20 MAPK1 gene sequences (and fragments) in different animals. ("frag. #" is the abbreviation for fragment number).

Figure 7 shows the 2 dimensional vector space arrangement for the non-linear embedding of MAPK1 gene in different animals. In the figure, each colored point represents a gene sequence and the metric distance between pairs of points correspond to the inferred evolutionary distance between the gene sequences. The same coloring style is used in this representation too for the differentiation between species. As seen from this figure, sequences of each species form distinct clusters. Mouse MAPK1 gene and the fragments form one cluster within the vector space as the human MAPK1 gene (including other primates) and its fragments. This is logical since the fragments belong to MAPK1 gene also. Another important point here is that the arrangement of the sequences on the vector space has directions. These directions are thought to represent some features of the animals present on that specific direction. For example, frogs are presented on a line that extends from the center where the mouse gene and its fragments presents to the direction of negative horizontal and vertical axes. This may correspond to a feature that is shared by the frogs but not the other animals in the dataset. Similarly another direction exists for the chicken and bird and a third one for cattle and wolf. In the following steps of the study this evolution-wise directionality was examined in detail.

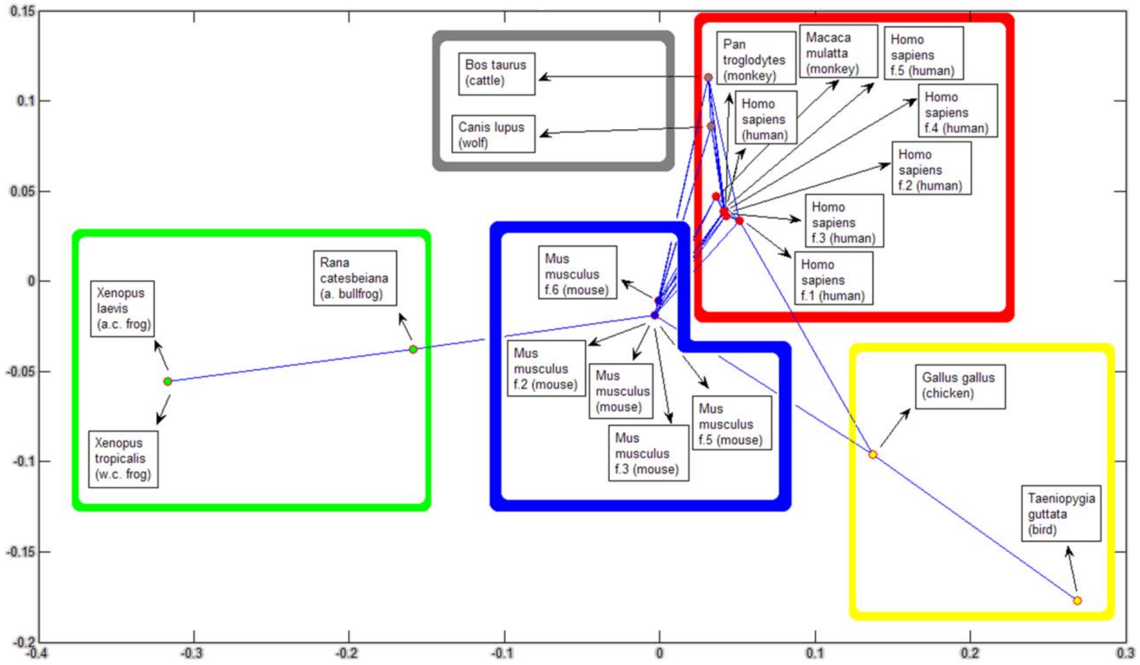


Figure 7. 2-D vector space arrangement for non-linear embedding of 20 MAPK1 gene sequences (and fragments) in different animals. (f # is the abbreviation for fragment number).

It was clearly observed when the Figure 6 and Figure 7 are compared that, even without any modification or optimization on the non-linear embedding procedure; the method produce promising results regarding uncovering the evolutionary relationships. In the next step of the study, the behavior of the algorithm on different datasets was tested.

2.3. Non-linear Embedding of Synthetic Gene Sequences

A more controlled experiment has been carried out in order to examine the non-linear embedding applied on biological sequences using computationally created synthetic gene sequences. Aim of this particular experiment was to observe visual representation of the multi-dimensional vector space arrangement of the synthetic gene sequences and to see if evolutionary directions are clearly observable on the arrangement.

Gene sequences are produced by applying random molecular substitutions at a constant average rate to the previous sequence starting from an original sequence. By this way the newly created sequence becomes the direct descendant of the previous one. A family of sequences is generated; where each one has either directly or indirectly, ancestor and descendant relations with the others. The evolutionary distances between the sequences are constant and known since the molecular substitution rate is a predetermined constant rate. Different evolutionary paths were formed by placing bifurcations at some particular locations. These paths represent the speciation event in the nature.

First evolutionary pathway ("path a") is shown in Figure 8 (a). In this path there were a total of 300 sequences, the average molecular substitution rate was 1% and there were 2 bifurcations, first one was after the 100th sequence and the second one was at the 50th sequence of the left arm (the right arm ended after the 50th sequence), after the second bifurcation the path was continued for 50 more sequences on both arms.

Second evolutionary pathway ("path b") is shown in Figure 8 (b). The total number of sequences in this path was 200 and the average molecular substitution rate was 2%. There was a direct quad-furcation from the original sequence and after that the path was continued on all 4 arms for 50 sequences. This occurrence is usually referred as the star formation in phylogenetic studies and treated as unresolved relationships of sequences which naturally should only contain bifurcations. Even though quad-furcation is not a natural formation, in order to observe the evolutionary directions and the angles between the lineages in the vector space arrangement, this simulation was thought to be suitable.

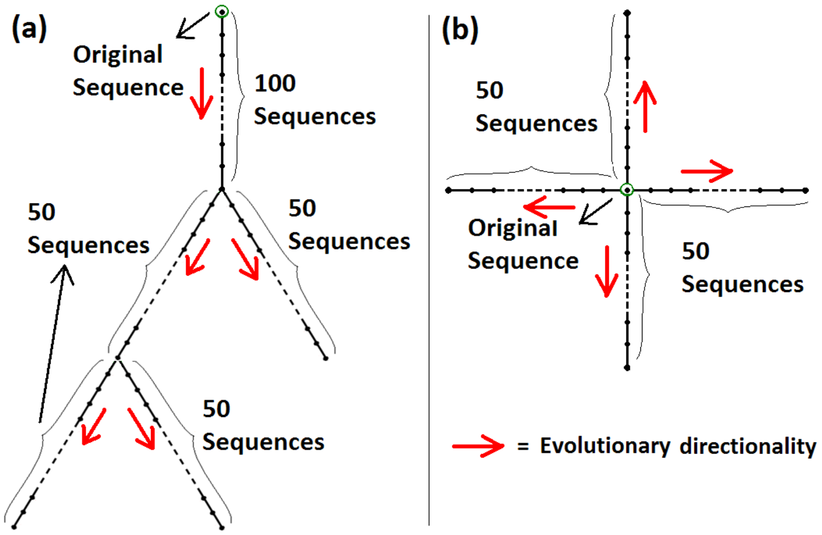


Figure 8. Tree representations of two different evolutionary pathways (“path a” and “path b”) created for evolutionary directionality analysis.

The sequences on each evolutionary path were used as 2 different datasets and non-linear embedding was performed on them. Figure **9** shows 2 (a) and 3 dimensional (b) vector space embeddings of “path a”. In the figure, each blue dot confined with a red circle (vector) represents a gene sequence. As obvious from the figure, evolutionary directions are clearly observable and each lineage has a distinct direction. At the point where the first bifurcation occurs (in the green circle both for (a) and (b)) the angles between ancestor lineage and the 2 descendant lineages are the same and equal to 120˚. At the location of the second bifurcation (dashed green line in (a) and blue circle in (b)), the angles are again 120˚ in 3 dimensional output, though, two descendant lineages overlaps and cannot be distinguished visually in the 2 dimensional output. The reason of this occurrence was the insufficiency of the number of dimensions in the 2-D arrangement. The algorithm tried to preserve the distances between the sequences as given in the input using a 2 dimensional space and this arrangement gave the pairwise metric distances closest to the input. This occurrence was not observed in 3 dimensional output because the method preserved the distances by locating the first bifurcation horizontally and the second one vertically, however, in 2 dimensions there were no suitable directions for the second bifurcation.

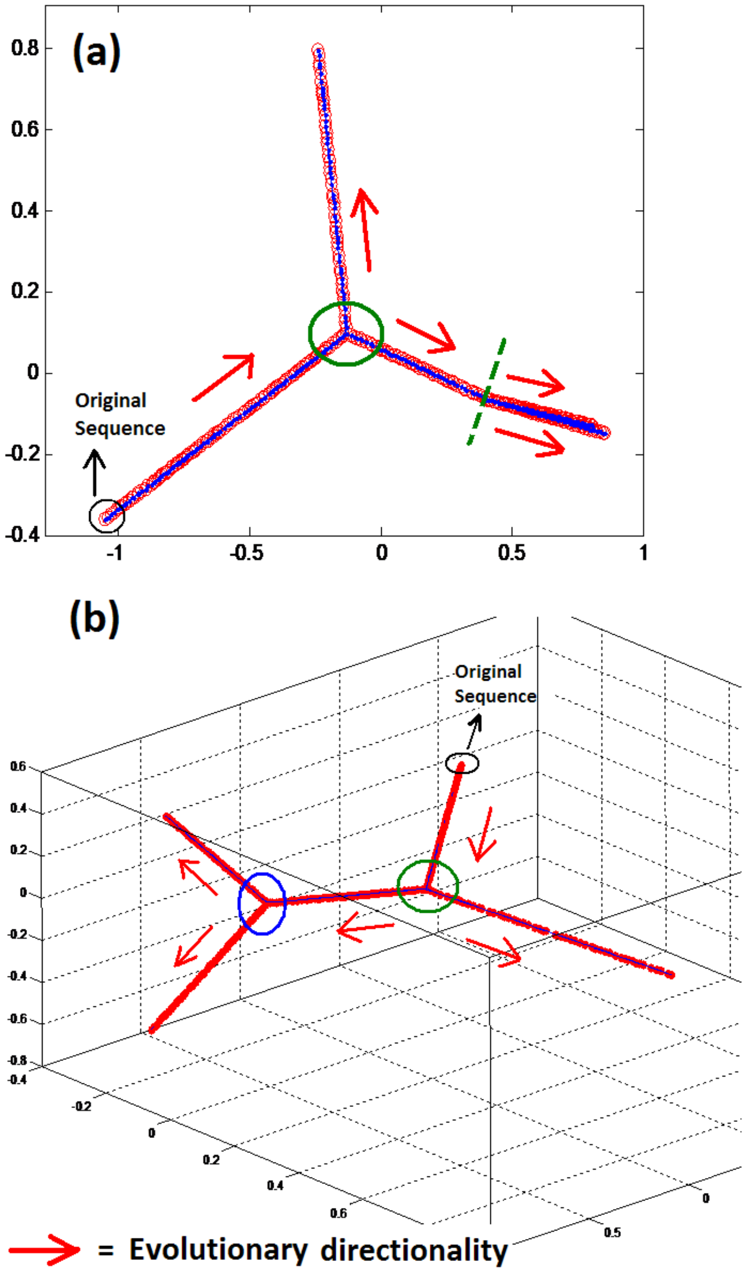


Figure 9. (a) 2-D and (b) 3-D vector space embeddings of “path a”.

Figure 10 shows 2 and 3 dimensional vector space embeddings of “path b”. The figure is colored similar to Figure 9. Similarly in this figure, the directions are clear and the same problem is valid for the 2 dimensional output. Fourth arm was placed quite close to the first one in the name of preserving all of the distances globally. In three dimensions, 4 arms extend in different directions with the same angle between each arm.

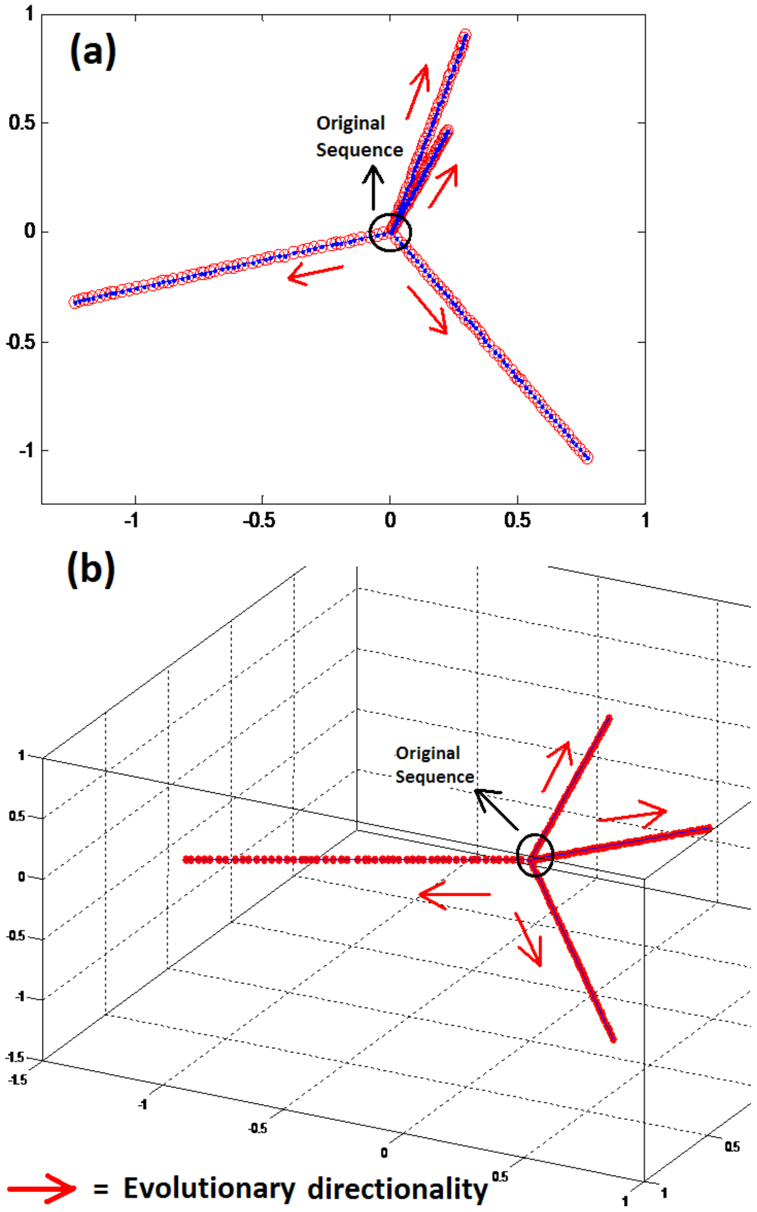


Figure 10. (a) 2-D and (b) 3-D vector space embeddings of “path b”.

At the end of this analysis, we have concluded that vector space representations of gene sequences via non-linear embedding strongly preserve the evolutionary directionality. This directionality may correspond to distinct features of the organisms, as a result, observation of these directions and the angles in-between may give clues evolutionary and/or functional relationships of sequences/organisms.

Next, we tried to measure the abilities of non-linear embedding in accurate positioning and grouping of biomolecular sequences in a quantitative manner. To this end, an experimental setup was prepared to measure the error in the inferred evolutionary relationships between the sequences before and after the embedding.

2.4. Error Analysis of Evolutionary Distances Obtained via Non-linear Embedding

The biomolecular data of the organisms lived in the past is not available presently (especially for eukaryotic organisms). As a result, evolutionary distances between molecular sequences cannot be found directly. Instead, they are approximated regarding the sequence distances and mathematical models. However, inferring evolutionary distances from sequence distances may bring along errors since the approach is mainly probabilistic. In theory, embedding gene sequences in a vector space and assuming the metric distances in-between to be equivalent to the true evolutionary distances should be valid unless the positioning of the sequences in the vector space is inaccurate. Since the method takes the evolutionary distances inferred by conventional models as input, it should reduce the amount of error already present in the input to yield accurate results. Therefore, the amount of the error before and after the non-linear embedding process should be obtained in order to find out the error reduction capacity of the method.

First of all, synthetic gene sequences are created computationally with a method similar to the previous experiment. New sequences are generated by placing random substitutions on the previous sequences. Three different paths are created and named as EP1, EP2 and EP3 (EP stands for evolutionary path). Each path consists of 501 sequences. Each sequence has the length of 1000 nucleotides and the average substitution rate is 2%. Figure 11 shows the courses of the formation of these evolutionary paths. EP1 was the simplest case, starting from an original sequence and deriving new sequences with molecular substitutions. In EP2 two gene sequences were emerged from the original sequence independent from each other (bifurcation). These two sequences were both direct descendants of the original sequence and as both were formed by transforming the original gene sequence by different random changes (with an average substitution probability of 2%). The rest of the procedure was the same as the previous one as the evolution was simulated by mutating the last gene sequence to form the new one. In this analysis, the evolution was maintained for 250 sequences in each arm giving 501 sequences in total. A more complex case was simulated in EP3. In this analysis, the evolutionary pathway was started again with a bifurcation at the beginning and continued for 100 steps (on each arm) at where bifurcations occurred on each arm again, a total of four arms were formed at this point. These four arms were continued for 75 steps more at which point the simulation ends. Again a total of 501 gene sequences were obtained at this path. These three simulations represent different evolutionary pathways of a gene in a population. Usually a natural case is much more complicated than these ones, but analyzing simpler cases may provide the ability to uncover the behavior of the method more clearly.

Second, the sequences on each path are aligned using Multiple Sequence Alignment procedure and the sequence distances were calculated. Following this procedure, evolutionary distances were inferred using the conventional Jukes-Cantor Model (Jukes and Cantor, 1969). The formulation of the model is given in Equation 2.3. Then, the embedding was processed using the inferred distances as the input. After the embedding procedure, the metric distances in the vector space were taken as the output evolutionary distances. The amounts of error in the inferred distances before and after the embedding were obtained regarding the known true evolutionary distances and compared with each other via bar graphs at different number of epoch distances (10, 10, 50, 100 and 200 epochs). Thus, the error reduction capacity of non-linear embedding is determined with the inspection of the results.

The optimum number of dimensions for the non-linear embedding of EP1 and EP2 was found to be 1 and the optimum epsilon values are decided to be 0.0255 and 0.0680 for EP1 and EP2 respectively. As for EP3, the optimum number of dimensions was 3 and the optimum epsilon value was 0.0890.

$$d = -(3/4)\log[1 - (4/3)D] \tag{2.3}$$

D represents the sequence distance and *d* is the inferred evolutionary distance.

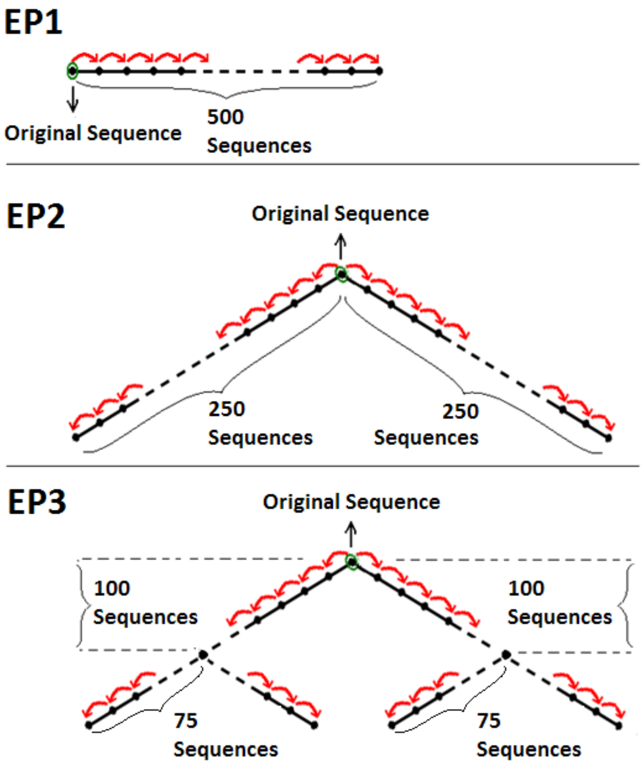


Figure 11. The courses of EP1, EP2 and EP3.

Error values for EP1, EP2 and EP3 are calculated separately. Apart from this, error analysis is not carried once for all the pairwise distances exist in a path instead, done separately for certain number of epoch distances (selections are 10, 10, 50, 100 and 200 epochs). The reason for this application is that the amount of error changes drastically with the amount of distance.

The parameter used for the examination of the error reduction capacity was the average absolute error value. It's the average of the absolute differences between the inferred evolutionary distances and the real distances. The parameter was calculated both before and after the non-linear embedding application at different epoch distances. Mathematical expression for average absolute error value calculation for Jukes-Cantor model (error before the application of non-linear embedding) is given in Equation 2.4 and the one for the calculation following the non-linear embedding process in Equation 2.5.

$$E_{jk_e} = \sum_{i=1}^{n-e} \sum_{j=i+e}^{n} \left| r_{ij} - d_{ij} \right| / m_e \qquad (2.4)$$

$$E_{v_e} = \sum_{i=1}^{n-e} \sum_{j=i+e}^{n} \left| r_{ij} - dv_{ij} \right| / m_e \qquad (2.5)$$

E_{jke} represents the average absolute error for Jukes-Cantor model for the epoch distance e whereas E_{ve} is the one for the non-linear embedding procedure; r_{ij} is real distance, d_{ij} is the distance inferred by Jukes-Cantor and dv_{ij} is the distance after the non-linear embedding between the sequences i and j, n is the total number of sequences and lastly, m_e is the total number of sequence pair combinations for epoch distance e.

Figure 12 shows average absolute error comparison for EP1, Figure 13 for EP2 and Figure 14 for EP3. In these figures, the horizontal axis represents the selected epoch distances (10, 20, 50, 100, 200) and the vertical axis represents the corresponding average absolute error values. As shown in Figure 12, Figure 13 and Figure 14; conventional Jukes-Cantor model and non-linear embedding gave similar error values up to 50 epoch distance. Jukes-Cantor model gave quite low and acceptable error values at these distances. It is obvious from the same figures that, at 100 and 200 epoch distances Jukes-Cantor Model's error increased suddenly. Besides, the non-linear embedding reduced the error significantly to a degree similar to the errors at 0 to 50 epoch distances. At high distances such as 100 to 200 epochs -which corresponds to 0.65 to 0.75 sequence distances and 1.5 to 3.0 inferred evolutionary distances- non-linear embedding is shown to have a significant error reduction capacity. At 100 epoch distance, the error value was nearly 6 and 5 times (for EP1 and EP2 respectively) higher for Jukes-Cantor model then the one for non-linear embedding (Figure 12 & Figure 13). At 200 epoch distance, this difference was 19 and 17 times (for EP1 and EP2 respectively) again higher for Jukes-Cantor model (Figure 12 & Figure 13). As represented in Figure 14, error values were significantly higher for non-linear embedding than the ones for Jukes-Cantor model for 10, 20 and 50 epoch distances. However, at 100 and 200 epoch distances, non-linear embedding produced significantly lower error values similar to the ones in the analyses for EP1 and EP2. At 100 and 200 epoch distances, the error values are nearly 2 and 21 times (respectively) higher for conventional Jukes-Cantor model (Figure 14).

At low distances (0 to 50 epoch) the amounts of error for the Jukes-Cantor model was reasonable, however, after this point, the error increased drastically. The reason of this error increment lies in the formulation of the model. Evolutionary distance is calculated directly from the sequence distance and as the sequence distance gets closer to 0.75 (which stands as the upper limit for this model) the evolutionary distance value becomes unstable. Sequence distance of 0.75 produce infinite evolutionary distance due to the presence of natural logarithm of zero in the formula.

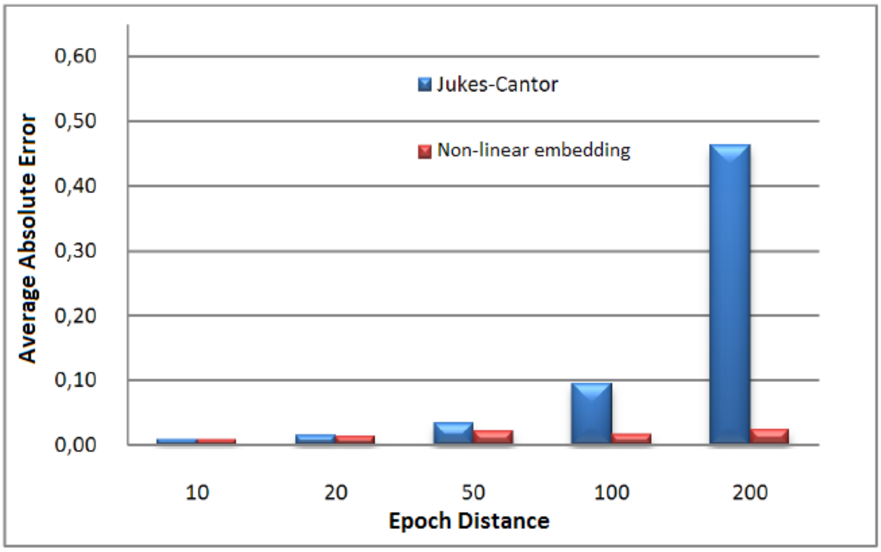


Figure 12. Average absolute error comparison at different epoch distances for EP1 (blue bars: Jukes-Cantor errors, red bars: non-linear embedding errors).

The potential of non-linear embedding in reducing the error rates at high epoch distances is attributed to the re-calculation of the distance between two far-away points by taking small jumps between proximal points located in-between and summing these small distances on the output manifold Instead of taking all of the given input distances into consideration.

In other words, the error arising from the distance between proximal points is quite low compared to the error of the distance between the far away points. While calculating high distances, the method adds short and reliable distances to each other and succeeds in keeping the error low.

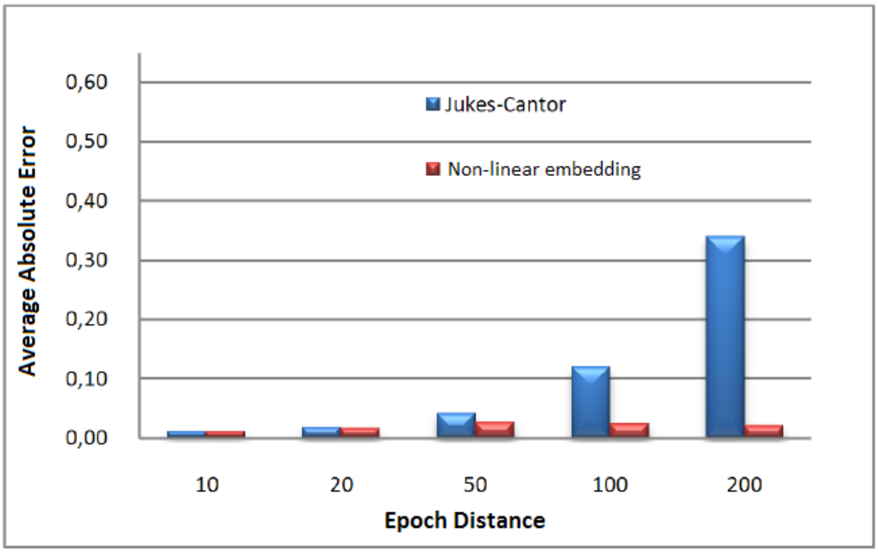


Figure 13. Average absolute error comparison at different epoch distances for EP2 (blue bars: Jukes-Cantor errors, red bars: non-linear embedding errors).

To comment on the high errors observed in the analysis of EP3 at 10, 20 and 50 epoch distances, the presence of elevated number of bifurcations in the evolutionary pathway increases the error especially at low epoch distances by increasing the complexity of the system. In order to reduce the total error globally, the algorithm sacrifices low distances and keep high distances accurate while operating with the limited number of dimensions. As a result, non-linear embedding produced reduced errors at high distances (100 epoch and higher).

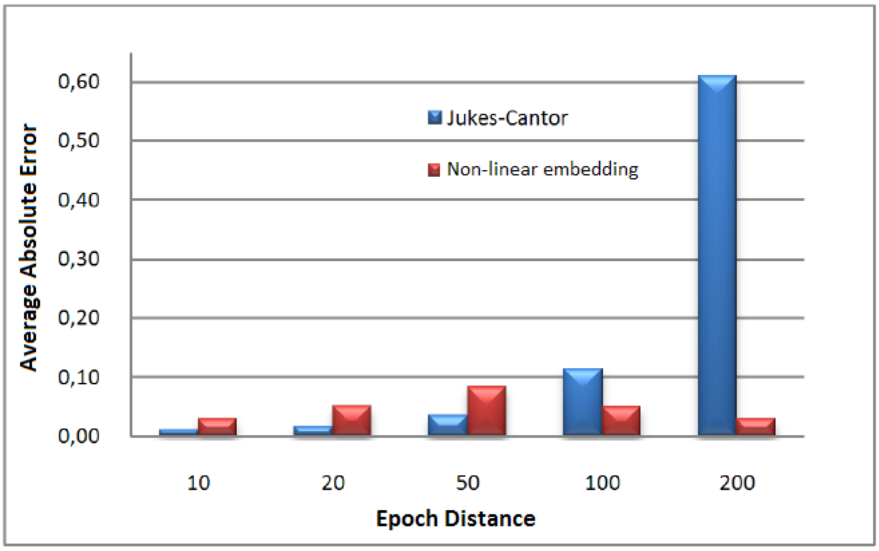


Figure 14. Average absolute error comparison at different epoch distances for EP3 (blue bars: Jukes-Cantor errors, red bars: non-linear embedding errors).

This analysis showed that non-linear embedding of gene sequences clearly reduces the amount of error present in the calculation of evolutionary distances between remote sequences. It’s concluded that with selection of correct parameters, the algorithm has the capacity to uncover the true evolutionary relationships between remote gene and protein sequences. To test this argument further on real sequences, the method was used to cluster 3 different protein families.

2.5. Clustering the Members of 3 Eukaryotic Protein Families Using Different Clustering Algorithms

The potential of non-linear embedding in discovering evolutionary relationships between gene and protein sequences was observed in the previous analyses. Here in this analysis we tested the accuracy of the method in clustering different eukaryotic protein families.

The motivation behind clustering protein sequences is that, evolutionary proximal sequences usually have similar functions and vice versa. In other words, evolutionary distances and functional similarities are strongly correlated with each other. As a result, it’s assumed that clustering biomolecular sequences regarding their evolutionary distances has a potential for revealing their functional relationships.

To test the accuracy of non-linear embedding in separating different protein families, 3 eukaryotic families with unrelated functions are selected from Prosite (Sigrist *et al.*, 2010), Swiss-Prot manually annotated protein database (Bairoch and Apweiler, 2000). For all three protein families, only function positive proteins are included in the experiment. First family was “ACTININ_1” with the accession number of “PS00019” which was described in the database as actinin-type actin-binding domain signature 1. This family had 84 function positive proteins discovered so far and members had an average of 2000 nucleotides per sequence. Second family was “MIF” with the accession number of “PS01158” and described as the macrophage migration inhibitory factor family signature with 27 proteins with an average length of 120 nucleotides per sequence. Third and the last family was “NNMT_PNMT_TEMT” with an accession number “PS01100”. This family was described as NNMT/PNMT/TEMT family of methyl transferases signature and it had 13 members and with an average length of 297 nucleotides. A total of 124 protein sequences were downloaded from

Prosite database and subjected to the multiple sequence alignment procedure using stand-alone ClustalW v2.0 package (Larkin *et al.*, 2007) with default parameters. After the alignment, the evolutionary distances were inferred using Jukes-Cantor model and are given to non-linear embedding process using 1 to 10 dimensional vector spaces. Clustering algorithms such as *k*-Nearest Neighbor (Cover and Hart, 1967), Maximum Likelihood (Duda *et al.*, 2001) and Support Vector Machines (SVM) (Cristianini and Shawe-Taylor, 2000) were applied on each embedding in order to cluster the output vectors. Then, the accuracy of the clustering was calculated regarding the true families of the proteins.

Figure 15 shows the true grouping of the protein sequences marked on the 2 dimensional non-linear embedding. It is clearly seen from this figure that, protein sequences are located in three visually distinguishable groups except one protein which is located close to the members of an unrelated family. The method can be considered successful regarding this figure but in order to show its accuracy on a statistical basis, clustering algorithms are applied on the output embeddings.

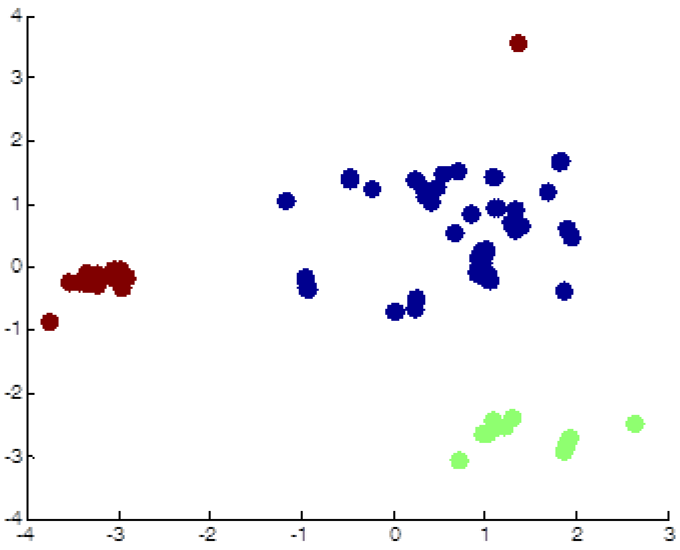


Figure 15. 2-D non-linear vector space embedding output of 3 eukaryotic protein families: “ACTININ_1” (blue), “MIF” (red) and “NNMT_PNMT_TEMT” (green).

Two different analyses were carried out regarding the *k*-Nearest Neighbor Classification algorithm. First, using leave-one-out technique and second using 50% of the points for the classifier training and the other 50% for testing. The selected number of neighbors was 1 (*k*=1) for the simplicity. The results appeared to be quite similar (data not shown) and only the results of leave-one-out strategy were selected to be shown for the rest of the section. As an alternative, Maximum Likelihood Classification method is applied to cluster the output vectors. Employing the widely used statistical decision rule Maximum Likelihood, the probabilities of each vector to be included in each and every class were obtained and the vectors were assigned to the class with the highest probability. As a second alternative, Support Vector Machines (SVM) algorithm with radial basis function kernel was used to classify the vectors. Since SVM compares just 2 groups at each application, a total of 3 applications had been performed regarding group 1 vs. 2 and 3, 2 vs. 1 and 3; and lastly, 3 vs. 1 and 2. As in the *k*-Nearest Neighbor application, analyses were made for both leave-one-out and 50% of the points for the classifier training strategies. The results for these two strategies were again quite similar (data not shown), and the leave-one-out strategy is selected to be displayed here.

Table 1 shows the performance of non-linear embedding process in separating unrelated protein families from each other by measuring the clustering accuracy (using different clustering algorithms) at different number of embedding dimensions. The goal here is to select of the minimum number of dimensions that represent the evolutionary relations accurately -in order to simplify the system as much as possible to get a visual output at 1, 2 or 3 dimensions when possible-. The two dimensional space was observed to be sufficient for *k*-Nearest Neighbor and SVM methods, whereas 6 dimensions were selected for the Maximum Likelihood method. Clustering success at selected number of dimensional spaces were over 95% for all methods. Selected numbers of dimensions and the accuracy are indicated as yellow highlights in Table 1.

The clustering results of the selected algorithms are marked on 2 dimensional non-linear embedding output and falsely grouped proteins are pointed out in black circles in Figure 16 for *k*-Nearest Neighbor, in Figure 17 for Maximum Likelihood and in Figure 18 for SVM. As seen from these figures, only one protein for *k*-Nearest Neighbor and SVM classifications, and 3 proteins in Maximum Likelihood classification, were misclassified.

Table 1. Clustering success for non-linear embedding regarding a range of embedding dimensions for (a) Nearest Neighbor (b) Maximum Likelihood and (c) Support Vector Machines methods (highest performances are highlighted in yellow color).

(a) Nearest Neighbor (leave-one-out)

	Number of embedding dimensions									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Clustering success	0,8306	0,9919	0,9919	0,9919	0,9919	0,9919	0,9919	0,9919	0,9919	0,9919

(b) Maximum Likelihood

	Number of embedding dimensions									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Clustering success	0,3145	0,8790	0,9355	0,9516	0,9597	0,9758	0,9677	0,9597	0,9435	0,9516

(c) Support Vector Machine (leave-one-out)

	Number of embedding dimensions									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Clustering success	0,4879	0,9919	0,9919	0,9758	0,9758	0,9758	0,9758	0,9758	0,9758	0,9758

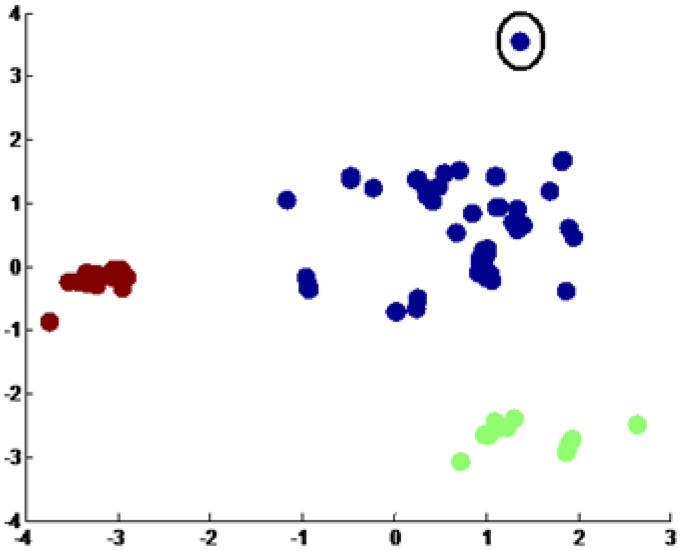


Figure 16. *k*-Nearest Neighbor classification (leave-one-out) clusters shown on 2-D non-linear embedding of 3 eukaryotic protein families. Blue: cluster 1, red: cluster 2 and green: cluster 3. Black circle: falsely classified proteins.

Table 2 shows the confusion matrices regarding the clustering of 3 protein families with the *k*-Nearest Neighbor algorithm (*k*=1) with the comparison of clustering using the initial Jukes-Cantor distances and clustering using the distances after the non-linear embedding process. The *k*-Nearest Neighbor algorithm (*k*=1) takes only the closest vector into consideration and the error rates of Jukes-Cantor distances are acceptable at low distances, as a result, leave-one-out strategy produced the same performance of classification before and after embedding on a 2-D vector space (Table 2, a and b). However, when only 10% of the points were used for the classifier training (the procedure is repeated 100 times with random selection of points for the classifier training), non-linearly embedded arrangement was slightly more accurately than the classification on Jukes-Cantor distances (Table 2, c and d). Since only a few points were selected for classifier training in this case, some of the test points were classified regarding far-away reference vectors and in the end, the high amount error in elevated Jukes-Cantor distances decreased the classification accuracy. On the other hand non-linear embedding reduced the error rates at these high epoch distances according to the previous experiment and this was reflected herein too.

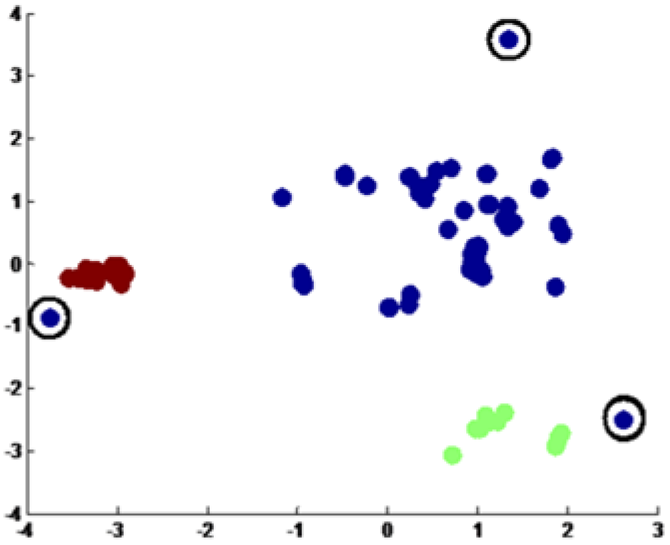


Figure 17. Maximum Likelihood Classification clusters shown on 2-D non-linear embedding of 3 eukaryotic protein families. Blue: cluster 1, red: cluster 2 and green: cluster 3. Black circle: falsely classified proteins.

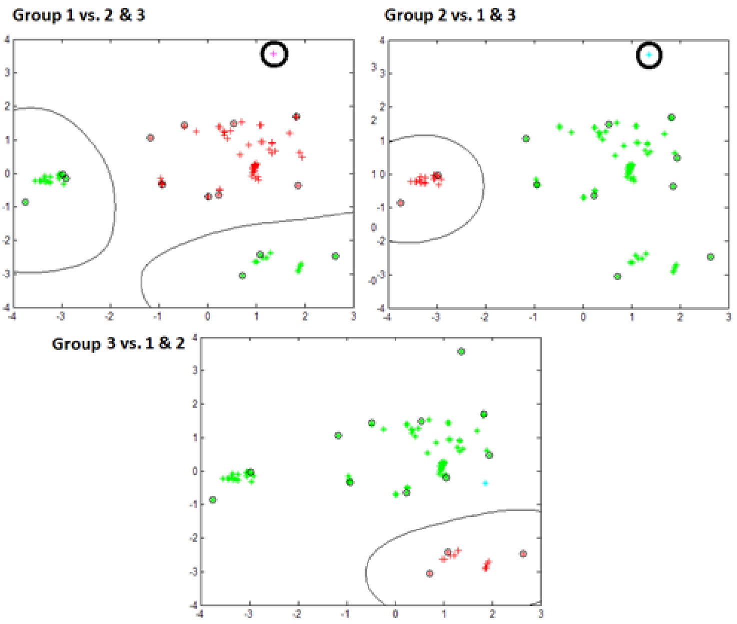


Figure 18. SVM classification (leave-one-out) clusters shown on 2-D non-linear embedding of 3 eukaryotic protein families. Blue: cluster 1, red: cluster 2 on each combination. Black circle: falsely classified proteins.

Table 2. Confusion matrices for the Nearest Neighbor Classification using (a) and (c): Jukes-Cantor distances, (b) and (d): non-linear embedding (2-D); (a) and (b): using leave-one-out method, (c) and (d): using 10% of the points for classifier training.

(a)

		True Grouping		
		1	**2**	**3**
Nearest	**1**	84	1	0
Neighbor	**2**	0	26	0
Class.	**3**	0	0	13

Clustering success: 0.9919

(b)

		True Grouping		
		1	**2**	**3**
Nearest	**1**	84	1	0
Neighbor	**2**	0	26	0
Class.	**3**	0	0	13

Clustering success: 0.9919

(c)

		True Grouping		
		1	**2**	**3**
Nearest	**1**	7436	117	364
Neighbor	**2**	140	2305	0
Class.	**3**	14	0	824

Clustering success: 0.9433

(d)

		True Grouping		
		1	**2**	**3**
Nearest	**1**	7585	117	364
Neighbor	**2**	5	2305	0
Class.	**3**	0	0	824

Clustering success: 0.9566

From Table 1, Figure 16, Figure 17 and Figure 18; it can be concluded that the non-linear embedding process accurately separated the examined protein families from each other on the 2 dimensional vectorial plane. Also to note that, this dataset was an easy case as the inter distances between protein families were quite large. Though, non-linear embedding has increased the accuracy of this separation as shown in Table 2 when only a few (10%) of the vectors were used for classifier training. These results indicate the potential of non-linear embedding in uncovering the functions of unknown proteins and genes with respect to their distances to the known ones in the multi-dimensional vectorial space. More complex (harder to classify) datasets was to be examined in order to investigate this potential further.

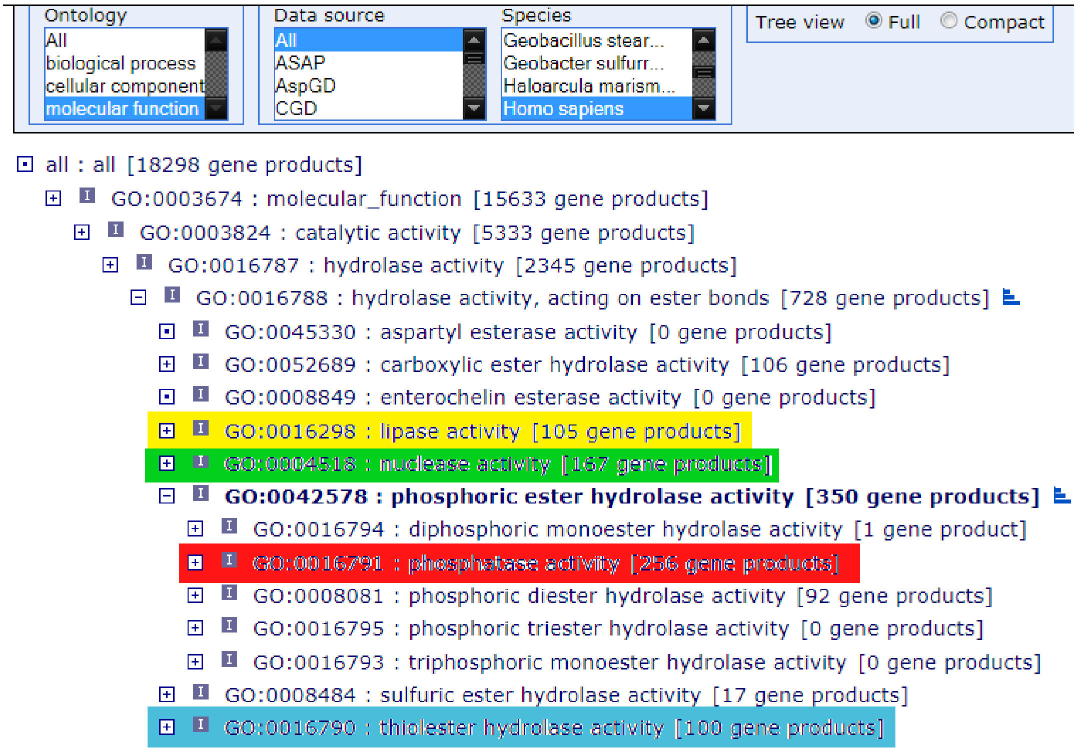
2.6. Non-Linear Embedding of Protein Families with Similar Functions from Gene Ontology Database

The potential of non-linear embedding in separating distant protein families was proved in the previous experiment. Here a new dataset consisting of protein families of similar functions has been prepared taking Gene Ontology (The Gene Ontology Consortium, 2000) associations into account.

Gene Ontology is a project aiming to standardize the gene and gene product attributes by assigning controlled vocabulary terms to each of them under three main topics: molecular function, biological process and cellular component (The Gene Ontology Consortium, 2000). Molecular function is the first one and represents the specific function of the sequence in the metabolism; biological process is the general operation during which this specific function is carried out; and cellular component is the location where this product functions. There is a hierarchical construction of these terms from broad to specific and a gene (or its product) is identified more clearly with growing number of associations. The sequences filtered through the careful inspection of GO makes them reliable samples for their functional annotations and there is a clear indication of evolutionary and functional relatedness (homology) between biomolecular sequences with shared GO terms.

AmiGO Browser (Carbon *et al.*, 2009) was used to select 625 human proteins regarding molecular functions. Proteins from four different molecular function groups namely, sequences with lipase activity, nuclease activity, thiolester hydrolase activity and phosphatase activity were collected in similar numbers from each group. Figure 19 shows the hierarchical relationships of the selected molecular functions on AmiGO browser highlighted in different colors.

First of all, the sequences were downloaded from UniProt Database (The UniProt Consortium, 2011) using the accession numbers obtained from Gene Ontology Database. After that, Multiple Sequence Alignment procedure was carried out using the stand-alone ClustalW v2.0 (Larkin *et al.*, 2007) package with default parameters. Next, the evolutionary distances were inferred as previously and lastly, the non-linear embedding procedure was processed with default parameters and regarding a neighborhood size of 7 nearest neighbors (k=7).



Ontology: All, biological process, cellular component, **molecular function**

Data source: **All**, ASAP, AspGD, CGD

Species: Geobacillus stear..., Geobacter sulfurr..., Haloarcula marism..., **Homo sapiens**

Tree view ● Full ○ Compact

```
⊟ all : all [18298 gene products]
  ⊞ GO:0003674 : molecular_function [15633 gene products]
    ⊞ GO:0003824 : catalytic activity [5333 gene products]
      ⊞ GO:0016787 : hydrolase activity [2345 gene products]
        ⊟ GO:0016788 : hydrolase activity, acting on ester bonds [728 gene products]
          ⊞ GO:0045330 : aspartyl esterase activity [0 gene products]
          ⊞ GO:0052689 : carboxylic ester hydrolase activity [106 gene products]
          ⊞ GO:0008849 : enterochelin esterase activity [0 gene products]
          ⊞ GO:0016298 : lipase activity [105 gene products]
          ⊞ GO:0004518 : nuclease activity [167 gene products]
          ⊟ GO:0042578 : phosphoric ester hydrolase activity [350 gene products]
            ⊞ GO:0016794 : diphosphoric monoester hydrolase activity [1 gene product]
            ⊞ GO:0016791 : phosphatase activity [256 gene products]
            ⊞ GO:0008081 : phosphoric diester hydrolase activity [92 gene products]
            ⊞ GO:0016795 : phosphoric triester hydrolase activity [0 gene products]
            ⊞ GO:0016793 : triphosphoric monoester hydrolase activity [0 gene products]
          ⊞ GO:0008484 : sulfuric ester hydrolase activity [17 gene products]
          ⊞ GO:0016790 : thiolester hydrolase activity [100 gene products]
```

Figure 19. The relationships of the selected molecular functions in AmiGO browser, the selected protein groups are highlighted in different colors.

Figure 20 shows the 2 dimensional vector space embedding output. Colors represent different protein families. Intertwined appearance of the protein families indicates that a successful separation between protein families could not be achieved on 2-D this time. The residual variance plot was examined in order to observe the minimum number of dimensions sufficient to represent the data accurately.

Figure 21 shows the residual variance plot for the non-linear embedding process. It is obvious that reaching the flat region (around value of 0.1) was not possible in 10 dimensions. This result was probably due to the high complexity of the dataset and the absence of the ancestral sequences that relate the proteins to each other accurately. Embedding the sequences in higher number of vector space dimensions might yield acceptable residual variance values.

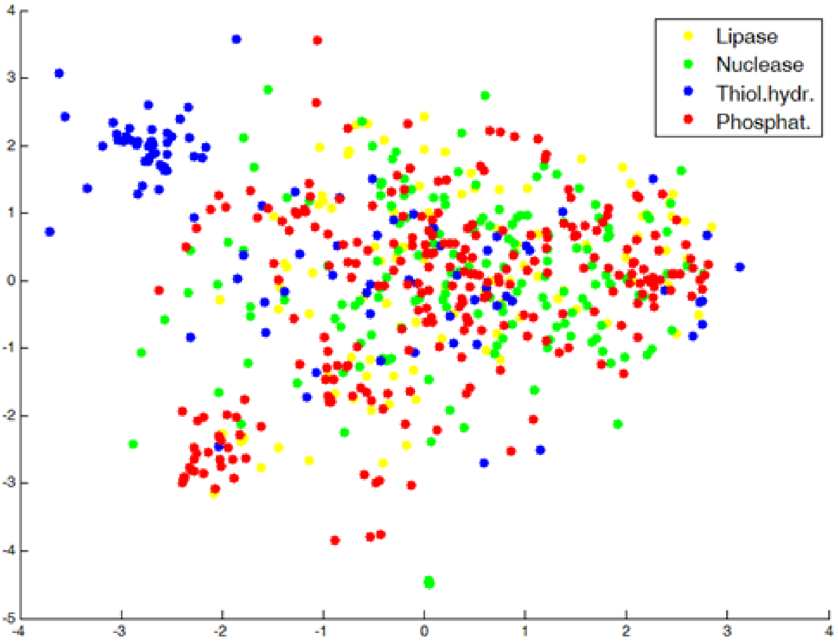


Figure 20. 2-D non-linear embedding output of 4 similar protein families (yellow: lipase, green: nuclease, blue: thiolester hydrolase and red: phosphatase activities).

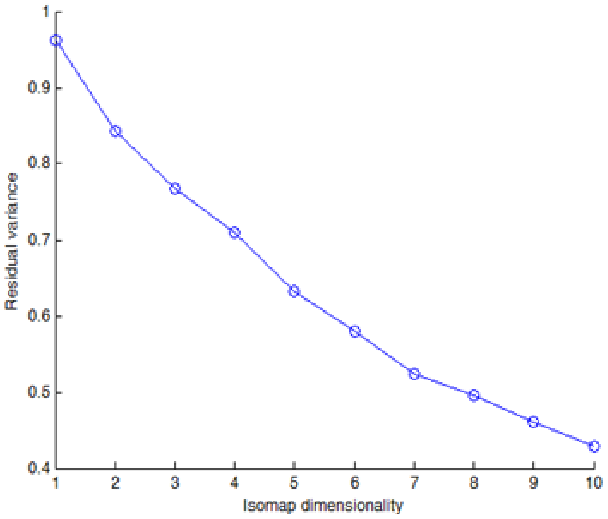


Figure 21. Residual variance plot for the non-linear embedding of 4 similar protein families.

The embedding process was repeated with 1 to 200 dimensions this time. Figure 22 shows the residual variance plot for this new batch. The variance reaches its minimum value at about 30 dimensions. Since a visual output could not be obtained from an embedding with this number of dimensions to judge on the separation accuracy, k-Nearest Neighbor Classification algorithm was applied with leave-one-out procedure to all embedding outputs from 1 to 200 dimensions. Figure 23 shows the embedding dimensions on the horizontal axis and the clustering accuracy on the vertical axis. At 2 dimensions, embedding accuracy was around 0.5 and at 20 dimensions; maximum accuracy has been reached with 0.7 and declined to 0.65 on average for the rest of the dimensions. At this point, it’s obvious that an accurate separation could not be obtained between the protein families with similar molecular functions via non-linear embedding even at elevated number of vector space dimensions.

It was concluded that, small datasets consisting of similar proteins from the same timeline -with nearly the same amount of diversifications from a common ancestor- were difficult to be classified unless ancestor sequences are included in the set. When the inter-cluster distances become similar to intra-cluster distances, stepwise revision of elevated distances cannot be processed accurately. Decreasing the neighborhood size might be a solution but this time; some of the vectors disconnects from the rest, forming components. At this point, the non-linear embedding method should be revised to be able to keep neighborhood size at low values without disconnecting the input sequences.

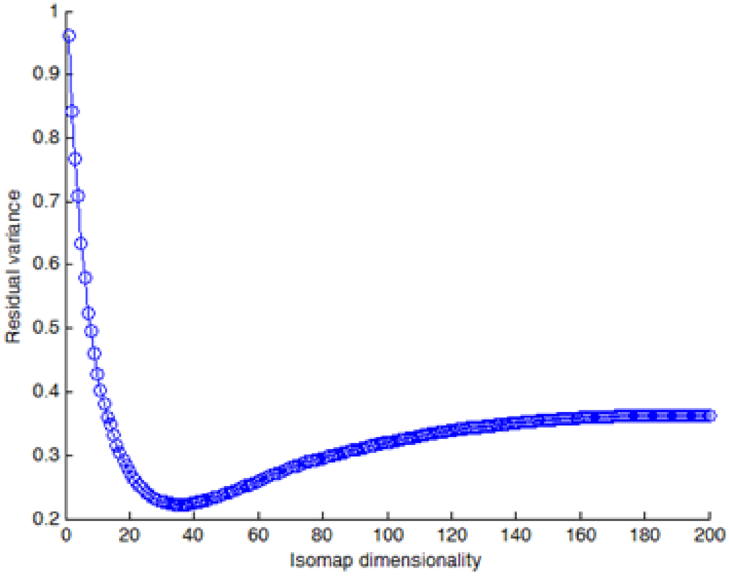


Figure 22. Residual variance plot for the non-linear embedding of 4 similar protein families from 1 to 200 dimensions.

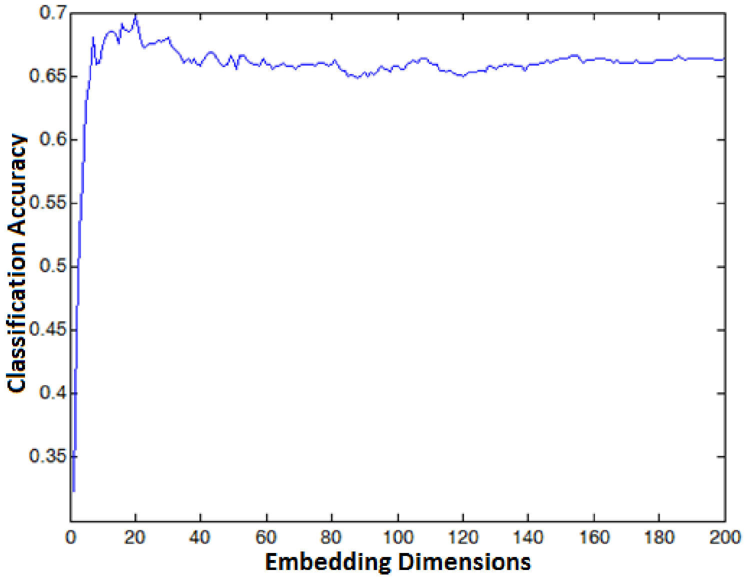


Figure 23. Embedding dimensions vs. classification accuracy for k-Nearest Neighbor Classification of non-linear embedding outputs.

2.7. Revision of the Non-Linear Embedding with MST and Testing with Synthetic Gene Sequence Dataset

In order to solve the problem of disconnection of the map at low neighborhood sizes, addition of Minimum Spanning Tree (MST) (Gallager *et al.*, 1983) connectivity to the non-linear embedding neighborhood calculation was carried out. This addition was thought to prevent the component formation while keeping the neighborhood sizes at low values thus allowing to the step-wise revision of the distances in the case of high intra-cluster and low inter-cluster distances.

A spanning tree of an undirected graph is a sub-graph that connects all the vertices together. A Minimum Spanning Tree (MST) is a spanning tree with the minimum weight. Figure 24 shows an example MST on a random distribution of points in a 2 dimensional space. MST connections were to be incorporated to the non-linear embedding process to keep all of the vectors connected at all times. Addition of MST connectivity to ISOMAP was done by changing the distance matrix revision procedure using Dijkstra's algorithm (Dijkstra, 1959). First, MST connectivity is added to the neighborhood graph so even if the neighborhood size was selected as zero, the number of connected components would still be one. After the selection of the neighborhood size, the desired connectivities are added on top of the MST graph. This new algorithm was called MST-Isomap.

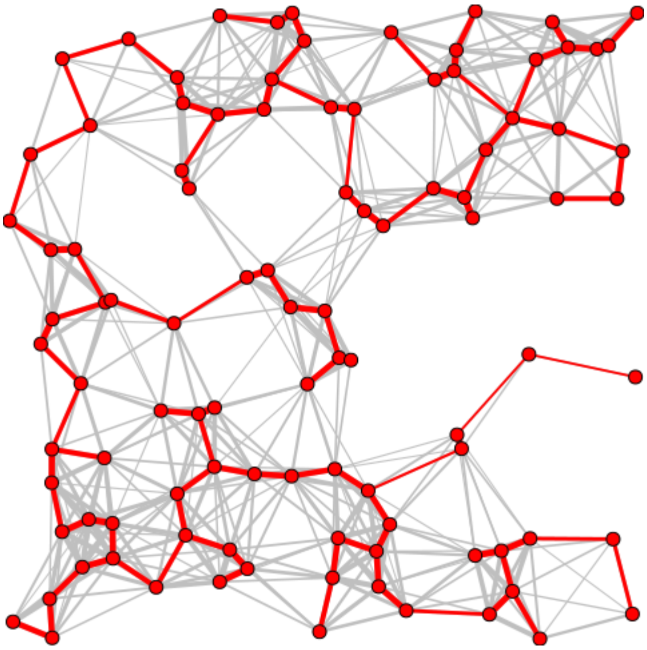


Figure 24. Minimum Spanning Tree (red colored network) of a random geometric distribution on 2 dimensions. (Source: Csardi and Nepusz, 2006)

In order to test the performance of the algorithm, a synthetic dataset of gene sequences were created by accumulating point mutations on an original sequence. A total of 162 sequences were created belonging to 4 main families. Only leaf nodes (sequences from present time) were included in the dataset and the distances between families were selected to be quite high. Figure 25 shows the tree representation of the sequences in the dataset with edge lengths. After that, evolutionary distances were inferred as previously. Normal and MST non-linear embeddings were applied on the evolutionary distance table using different neighborhood sizes and embedding dimensions. Lastly, pairwise errors were calculated and compared with each other.

Residual variance plot of the process using k-Isomap with the neighborhood size of 61 is shown in Figure 26. It is obvious from this figure, 2 dimensions were sufficient for an accurate embedding. Figure 27 shows the 2-D embedding output for the same run with colors to represent different protein families. The families are separated from each other considerably as seen from the figure; however, the change of error in different runs (with *k*-Isomap and MST-Isomap) should be measured precisely to observe the effect of adding MST connections to the process.

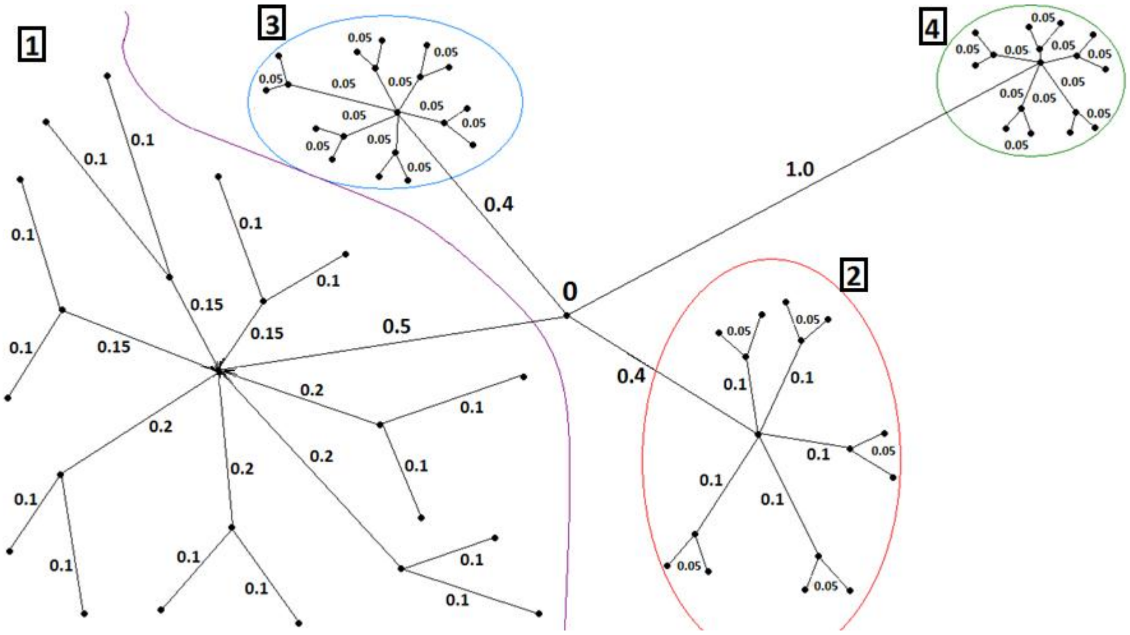


Figure 25. Tree representation of the synthetic sequence dataset, consisting of 4 protein families. Families are separated by colored curves, numbers in black squares shows the family numbers and the numbers next to edges show evolutionary distances.

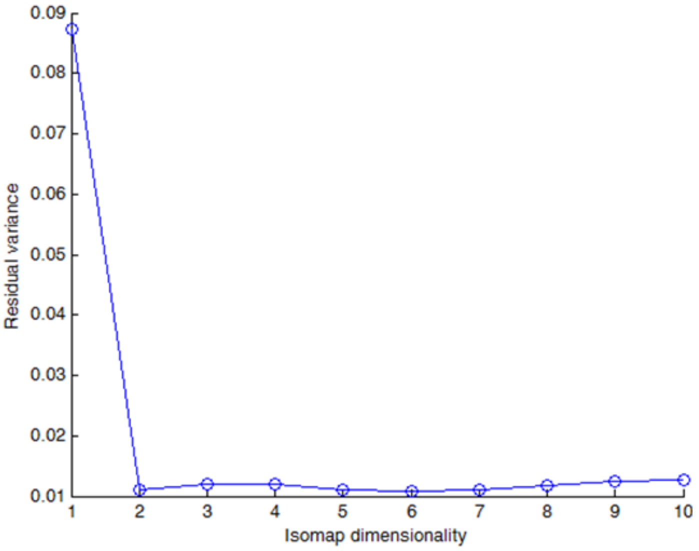


Figure 26. Residual variance plot for the non-linear embedding of the synthetic sequences dataset (ISOMAP with *k*=61).

Figure 28 shows the change of average absolute error (Equation 2.5) in pairwise distances after non-linear embedding and MST non-linear embedding processes for different number of embedding dimensions (only some of the runs are shown on the figure for the sake of visuality). It is evident that, after some degree the error falls below the error produced by the evolution model (the error on the distances before embedding) for all parameter selections. It is shown that the process with the neighborhood size of 61 performs the best and decreases the error present in the input matrix (0.375) to 0.22 at the flat region (around 50 dimensions).

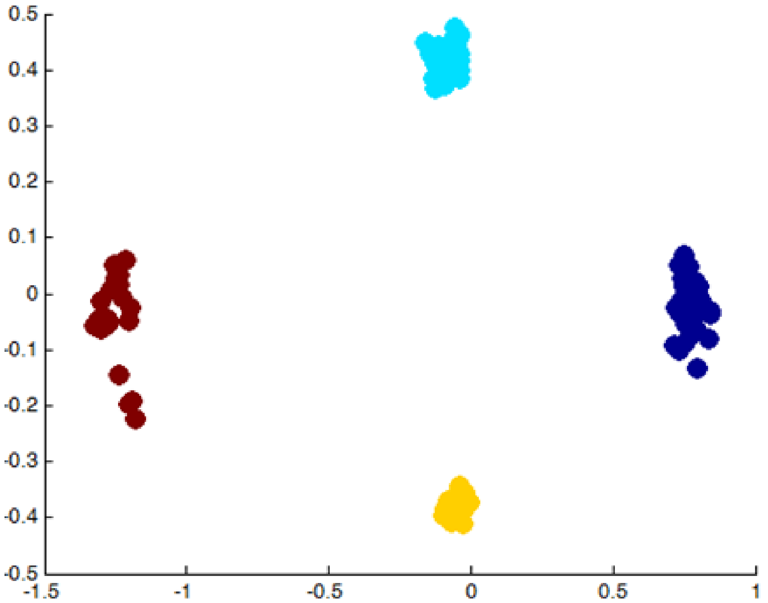


Figure 27. Non-linear embedding output of synthetic sequence dataset in 2-D, colors represent different protein families (ISOMAP with *k*=61).

Apart from it, the best performed MST non-linear embedding parameters were MST plus 45 nearest neighbors. It performed similar to Principle Component Analysis (which is calculated by setting the neighborhood size to infinity). Addition of MST to the non-linear embedding improves the results at 45 nearest neighbors but best performance was presented by non-linear embedding without MST at 61 nearest neighbors. As a result, addition of MST did not have a positive effect on decreasing the error. This was probably due to the high amount of error in some of the input distances that were conserved and carried to the output by MST.

At this point, non-linear embedding was considered to be unsuitable for the analysis of small datasets consisting of evolutionary proximate gene and protein sequences. However, it was also thought that, using large datasets such as whole proteomes might result in differently. Since evolutionary rate acting on thousands of proteins could not be the same, step-wise revision of the distance matrix might turn out to be more accurate. Consequently, we decided to go directly on the whole human proteome which was our main interest from the beginning of the study.

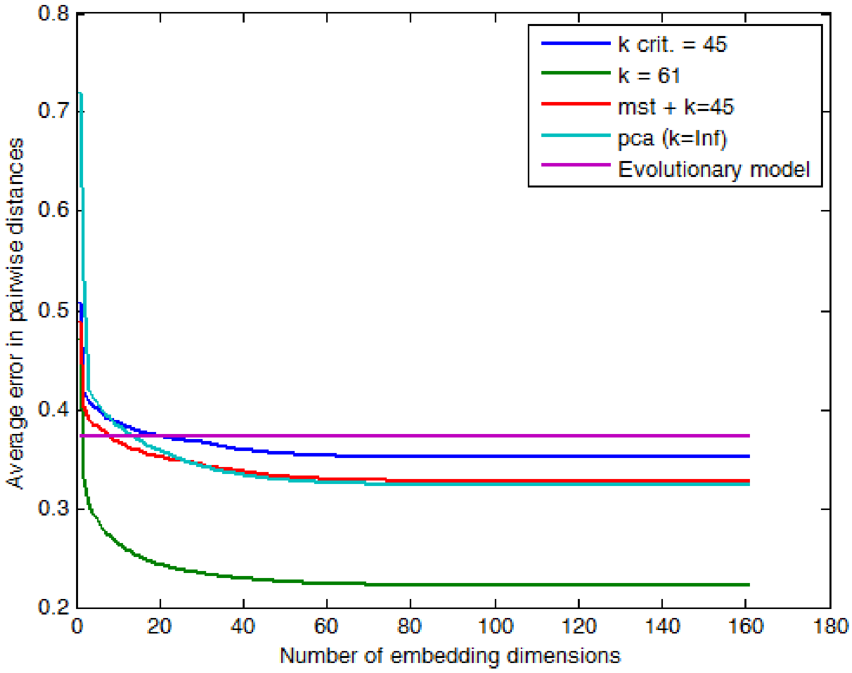


Figure 28. The change of the average absolute error in the evolutionary distances, horizontal axis represents the number of embedding dimensions and vertical axis represents the error values (blue: normal embedding (k=45), green: normal embedding (k=61), red: MST + normal embedding (k=45), cyan: normal embedding (k=infinite), purple: Jukes-Cantor model).

2.8. Evolutionary Analysis of Human Proteome with Non-linear Embedding

One of the aims of this study was to observe groups of proteins specialized for specific tasks in human body with complete analysis of the proteome. It was shown on the previous analyses that the non-linear embedding process has the potential to carry out this task accurately. Besides, statistical methods would be applied on the output high dimensional vectorial embeddings, in order to classify these groups.

Since the evolution rate was not constant and not acting exactly in the same way on all genes in the human genome (unlike the synthetic datasets in the previous analyses) stepwise formation of the distance matrix was thought to work accurately. It was expected that, in such a populated dataset, specialized groups of proteins should be separated from the bulk with perpendicular angles in the output embeddings thus making it easy to spot and separate them.

First of all, to generate the dataset, the accession numbers of human proteins with at least 1 Gene Ontology association were collected from GO website. Since Gene

Ontology project aims the standard representation and documentation of genes and its products, the proteins annotated by GO have gone through a detailed inspection and examination process, as a result their functional associations are more reliable. The dataset was formed with the download of the human protein sequences from UniProt Database (The UniProt Consortium, 2011) via the accession numbers. The dataset contained of 18011 human proteins.

As a second step, the Multiple Sequence Alignment procedure was carried out using stand-alone ClustalW v2.0 package (Larkin *et al.*, 2007) with default parameters. After that, the evolutionary distances were inferred via the built-in function of ClustalW package using Kimura Amino Acid Substitution Model (Kimura, 1983). Following an optimization on the non-linear embedding algorithm for faster process and less memory usage, the embedding of the sequences was processed and the smallest number of dimensions that give an acceptable residual variance were sought. The process was repeated a number of times to optimize the size of the neighborhood. Lastly, various operations were applied on the input and the algorithm to improve the accuracy.

First embedding was processed using 1 to 100 dimensions and a neighborhood size equals to 12 (k=12) which was the critical neighborhood size (i.e. the lowest size that yield an all connected map). Figure 29 displays the 2 dimensional (on the left) and 3 dimensional (on the right) embeddings. As expected, some of the proteins came out from the bulk like spikes with perpendicular angles to each other. The bulk (nearly 17500 proteins) remained at the intersection point of the spikes. Due to the limitation placed by the number of dimensions, only 2 perpendicular groups observed at 2-D and 3 groups at 3-D. It was also observed that the number of the spikes increased proportionately with increasing number of dimensions. The reason for the necessity of new dimensions for the formation of additional spikes was to maintain the perpendicularity between these spikes.

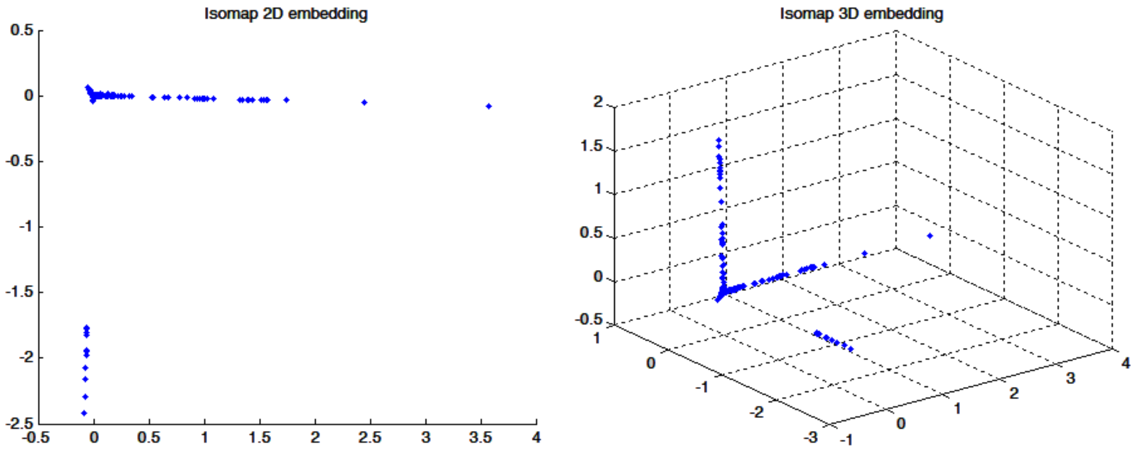


Figure 29. 2-D (left) and 3-D (right) vectorial embedding outputs on the human proteome dataset.

At this point, proteins on these spikes should be inspected to find out whether they belong to protein groups specialized for certain functions or totally random. On Figure 30, the families of the proteins in these spikes are marked on the embeddings. As expected, proteins on each spike belonged to a certain protein family. For the first 3 dimensions these groups were Dynein, Myosin and Collagen. The logic behind their perpendicularity to each other was that, functionally unrelated evolutionary differentiations are displayed as directional divergences from the bulk in different paths. Each differentiation was represented by a directional divergence on a dimension in the Euclidean space. In other words, 2 dimensions were required to observe 2 differentiations and the intersection of these ended up to be perpendicular. As a result, the required number of dimensions was equal to the protein families in the dataset. This point was roughly correlated to the minimum (at the start of the flat region) of the residual variance plot.

Figure 31 shows the residual variance plot for the embedding. The variance remained quite high even at 100 dimensions (0.78) and it was decided that the process should be repeated with a wider dimensional range in order to obtain acceptable variance values.

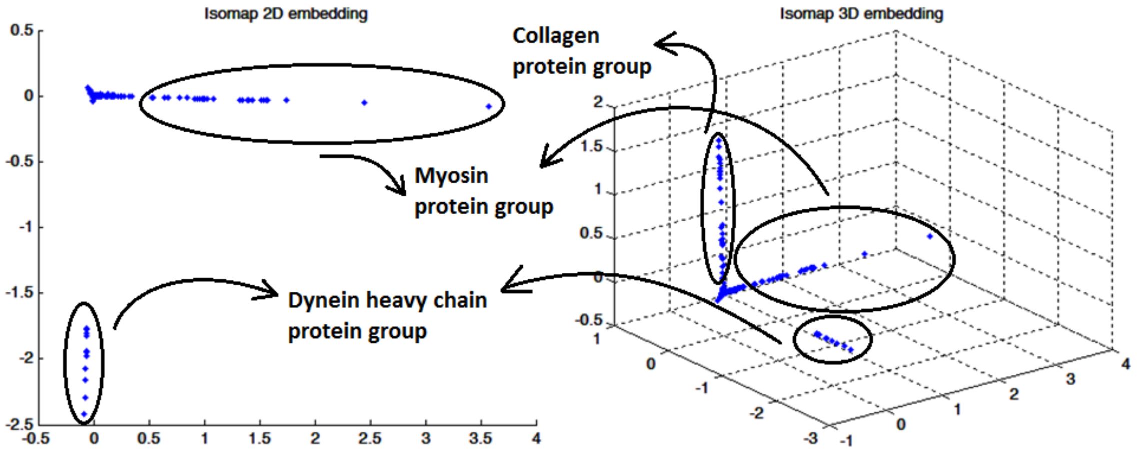


Figure 30. Major protein families marked on 2-D (left) and 3-D (right) vectorial embedding outputs on the human proteome dataset.

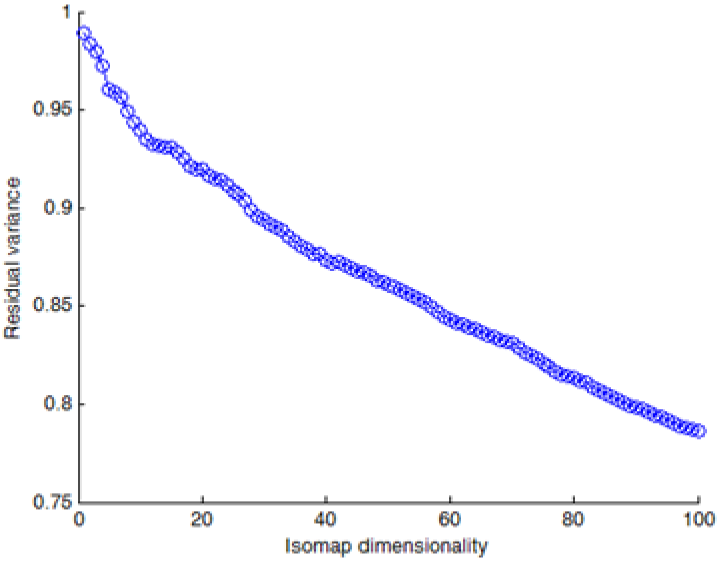


Figure 31. Residual variance plot of the non-linear embedding process on the human proteome dataset.

The embedding process was repeated using 1 to 10000 dimensions with the same parameters. Residual variance plot of this new process was shown on Figure 32. The minimum was reached around 6000 dimensions with the value of 0.1. This result roughly meant there should be nearly 6000 different protein families in the dataset. This result was considered to be incorrect since the whole dataset consisted of nearly 18000 proteins so each family should contain 3 proteins on average. The minimum variance was expected to be observed on much lower number of dimensions.

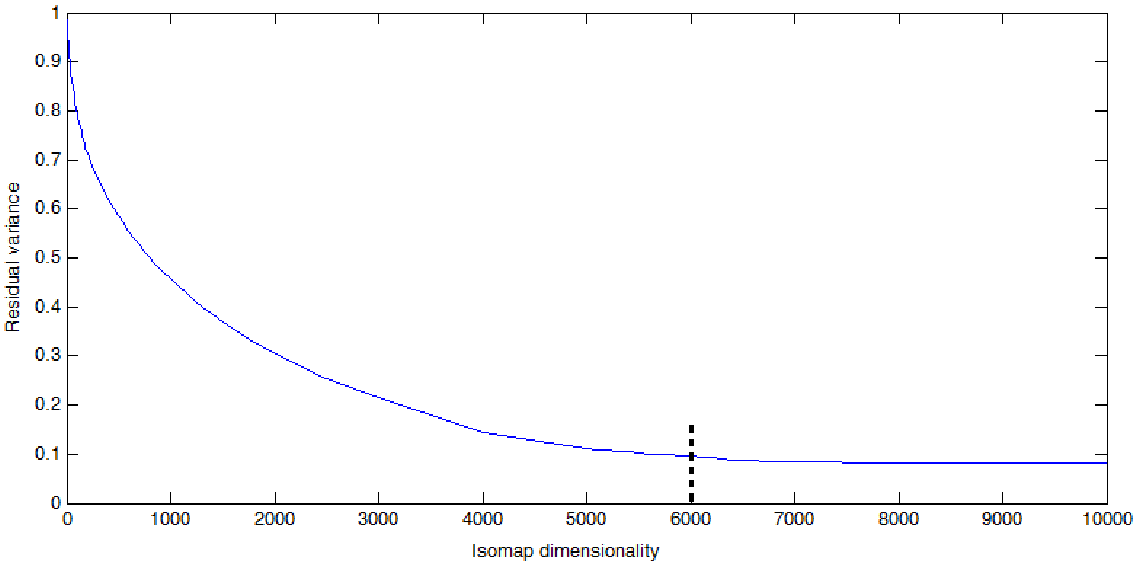


Figure 32. Residual variance plot of the second non-linear embedding process (1 to 10000 dimensions) on the human proteome dataset.

The reason of the high variance at lower dimensions might be due to the low overlap of sequences in some of the pairwise alignments (the gaps were not scored during the evolutionary distance calculation). After an inspection, it was discovered that among 162 million pairwise alignments in the Multiple Sequence Alignment process, there were many sequence couples that only 1 or 2 site were occupied by an amino acid on both sides (no gaps on both sequences) and if this had been a match, the algorithm gave zero distance between these 2 proteins. This information was not reliable and impaired the non-linear embedding distance revision step, resulting in false connectivities. Since the non-linear embedding does not need a full distance matrix to perform, these erroneous distances might be removed from the distance matrix confidentially if detected accurately.

For this purpose a new parameter was created named overlap fraction. This parameter is defined in Equation 2.6. Input pairwise distances with overlap fractions lower than a pre-defined threshold were decided to be discarded.

$$O_{f_{ij}} = \begin{cases} \dfrac{s_{ij}}{n_i}, & n_i < n_j \\ \dfrac{s_{ij}}{n_j}, & n_j \geq n_i \end{cases} \tag{2.6}$$

O_{fij} represents the overlap fraction, s_{ij} is the number of sites without gap at the pairwise comparison between the sequences i and j; n_i is the total length of sequence i and n_j is the total length of the sequence j.

According to the test run, the output graph remained connected below the threshold 0.975 so the disconnection of the map was not a problem. Using some key threshold values, distance matrix is revised and the non-linear embedding process was run. Figure 33 shows the residual variance plot for overlap fraction thresholded input distances.

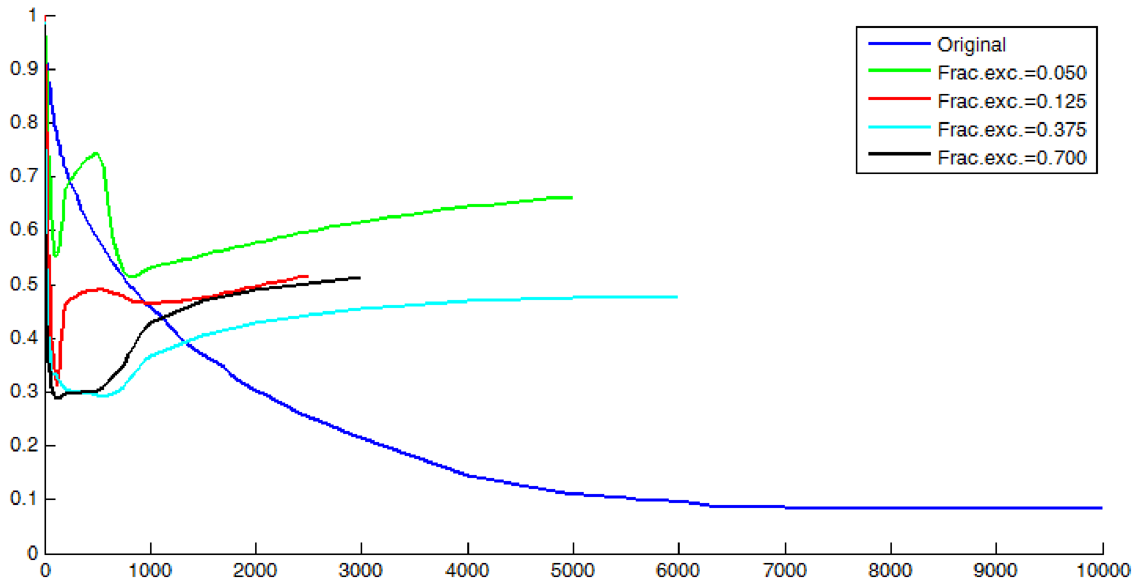


Figure 33. Residual variance plot of the non-linear embedding of human proteins with overlap fraction threshold distances; horizontal axis: the number of embedding dimensions, vertical axis: variance values (blue: the original dataset, threshold sets: green - 0.05, red - 0.125, cyan - 0.375, black - 0.7).

Excluding pairwise distances with overlap fractions under the threshold could produce low but unstable residual variance values. Variances significantly lower than the original dataset (near 0.3) were obtained by the thresholding operation around 100 dimensions; nevertheless, these variance values were not acceptable. Moreover, after 100 dimensions, variances started to increase. Due to this instability, instead of using overlap fractions, the number of overlaps (total number of sites without a gap) decided to be used directly for the thresholding operation. Input distances were thresholded with different selections and the embedding was carried out on these inputs. Figure 34 shows the residual variance plots for this operation.

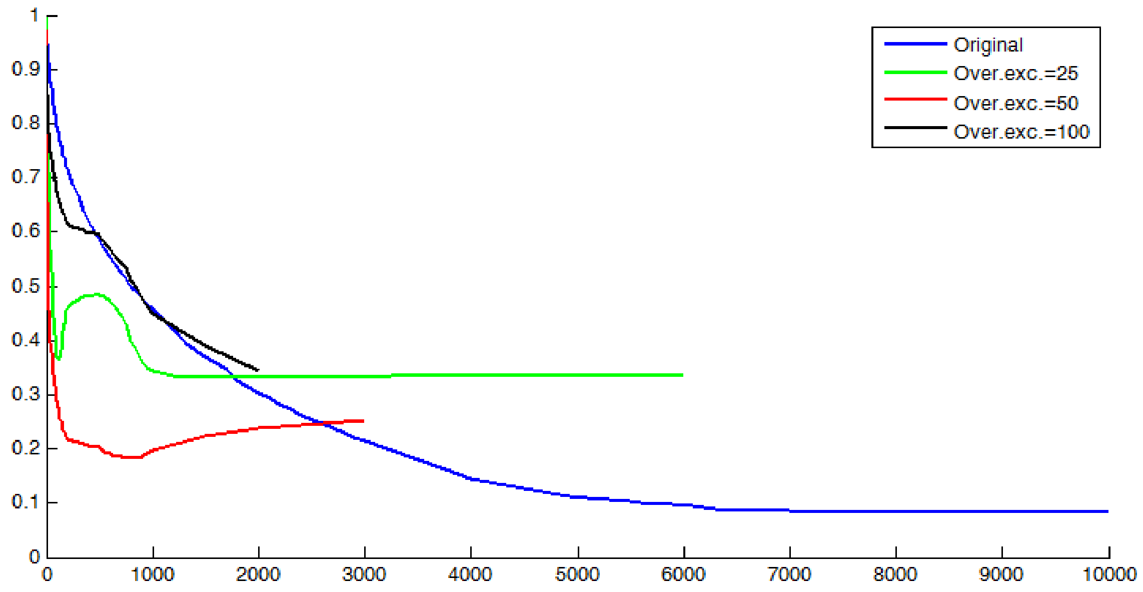


Figure 34. Residual variance plot of the non-linear embedding of the human proteins with input distances threshold directly by the number of overlaps; horizontal axis: the number of embedding dimensions, vertical axis: variance values (blue: the original dataset, threshold sets: green – 25, red - 50, black – 100 positions).

As observed from Figure 34, thresholding the distances directly by the number of overlaps, yields lower residual variance values especially at the threshold of 50 positions. This threshold yielded a residual variance lower than 0.2 around 650 dimensions. However, the output was still not acceptable as the desired variance and the stability could not be obtained in any way.

On the other hand, absence of the ancestor sequences in the datasets seriously altered the distance revision process and it’s nearly impossible to obtain an accurate separation on protein families with similar functions. A final analysis was setup in order to see the effect of presence and absence of the ancestor sequences on vector space embedding of biomolecular sequences this time with real data.

2.9. Effect of the Ancestor Sequences on Non-linear Embedding

We showed in the previous analyses with synthetic datasets that, non-linear embedding of biomolecular sequences decrease the initial error present on inferred pairwise distances significantly when the ancestor sequences were included in the dataset. However, we couldn’t test it with real data since the ancestor sequences are not available and because the real evolutionary distances between the sequences are not known. We observed unsatisfactory results with the embeddings of large datasets and with the sets containing sequences from similar protein families. We attributed the case to the absence of ancestor sequences but couldn’t observe it experimentally.

In this section, the setup and the results of an analysis to test our idea on ancestor sequences are presented. Wahlberg *et al.* studied the unresolved phylogeny of butterflies in order to make this well-emphasized species available as model organisms (Wahlberg *et al.*, 2005). They combine the molecular data (from 3 important genes) with traditional morphological characters, exploiting the synergistic effect of using different approaches at the same time to solve the clades. Bayesian inference was employed to solve the phylogeny where the inferences are based upon posterior probabilities calculated for each tree using Bayes theorem (Huelsenbeck *et al.*, 2001). MRBAYES v3.1 software was employed for the analysis in which Markov Chain Monte Carlo (MCMC) is used to solve the integrals, analytically unsolvable otherwise (Ronquist and Huelsenbeck, 2003). Many trial runs allow the determination of the correct parameters and the accuracy of the results were checked regarding the distribution of butterfly families in the tree, with a comparison of the distribution with the other methods’.

Figure 35 shows the resulting phylogenetic tree. The distribution of the members of each family to the clades appears accurate as observed from the figure. We decided to process the dataset containing the molecular data of the 57 taxa, as used in the referenced study. We subjected the sequences to Multiple Sequence Alignment and inferred the evolutionary distances with our standard procedure (as in the previous analyses). Then, the non-linear embedding procedure was run with inferred distances and the 2-D embeddings were examined (shown in Figure 36). It’s observed that a clear separation could not be obtained between the families, similar to the previous analyses. Next, we repeated the analysis this time with the inclusion of ancestor sequences inferred for each ancestral node by MRBAYES software during the careful and optimized Bayesian analysis of Wahlberg *et al.* Figure 37 shows the 2-D embedding results of this run (with the same parameters as the previous run). As seen from the figure, there is a clear separation between the families, each coming out from a shared point (root) with a spike-like formation. Descendant sequences are located near the tips of the spikes where their ancestors remain close to the root (nodes lined up in the hierarchical order). Even though the number of dimensions was not enough to handle 6 spikes -for 6 families- (plus one for outliers) with equal angles in-between, separation of families is clear enough and the evolutionary directionality is striking. The separation was much more distinct in the embedding with 6 dimensions (data not shown).

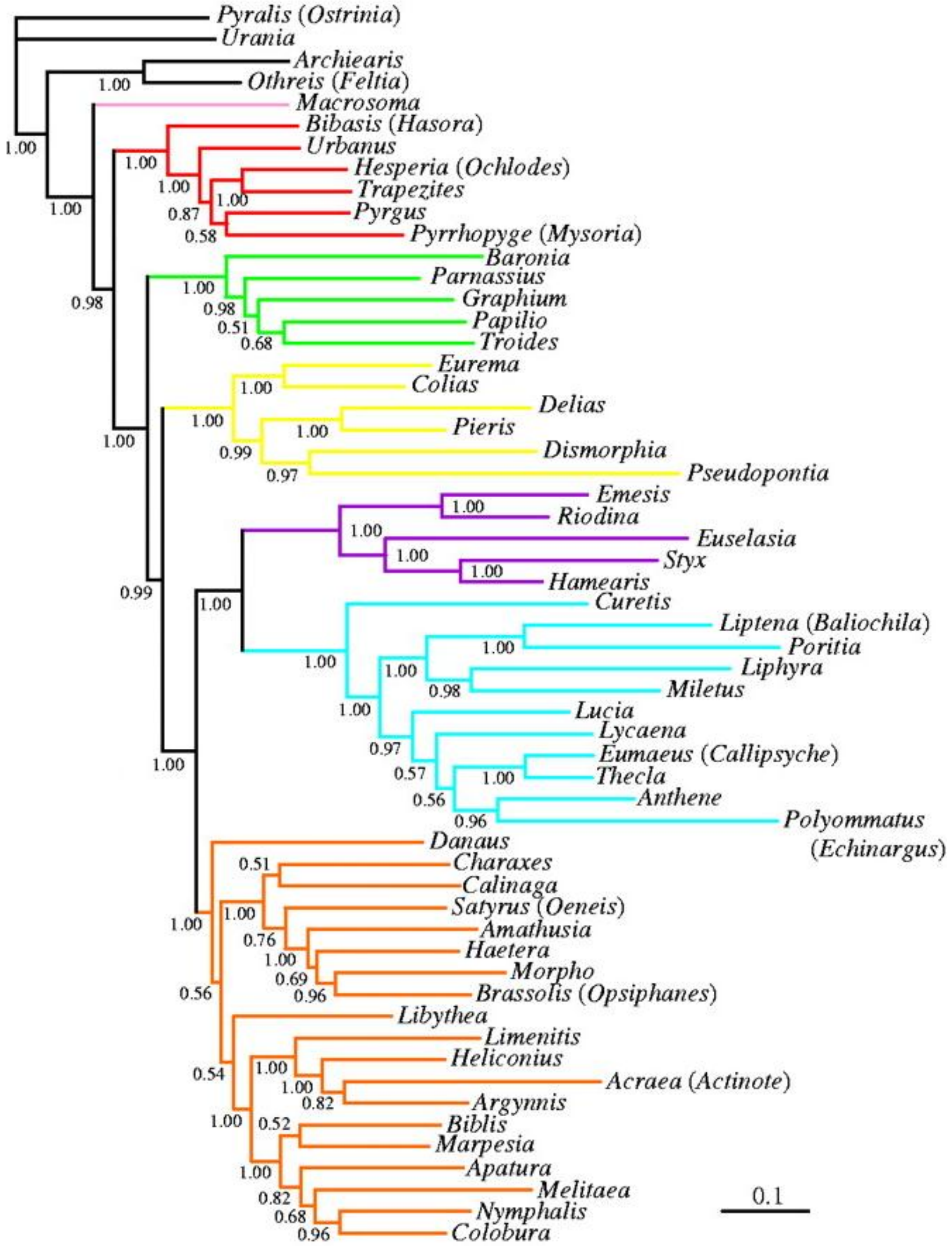


Figure 35. Phylogenetic tree for the butterfly species (57 taxa) consisting of the members from 6 different families and outliers generated regarding a combination of molecular and morphological data; colors represent families. (Source: Wahlberg *et al.*, 2005)

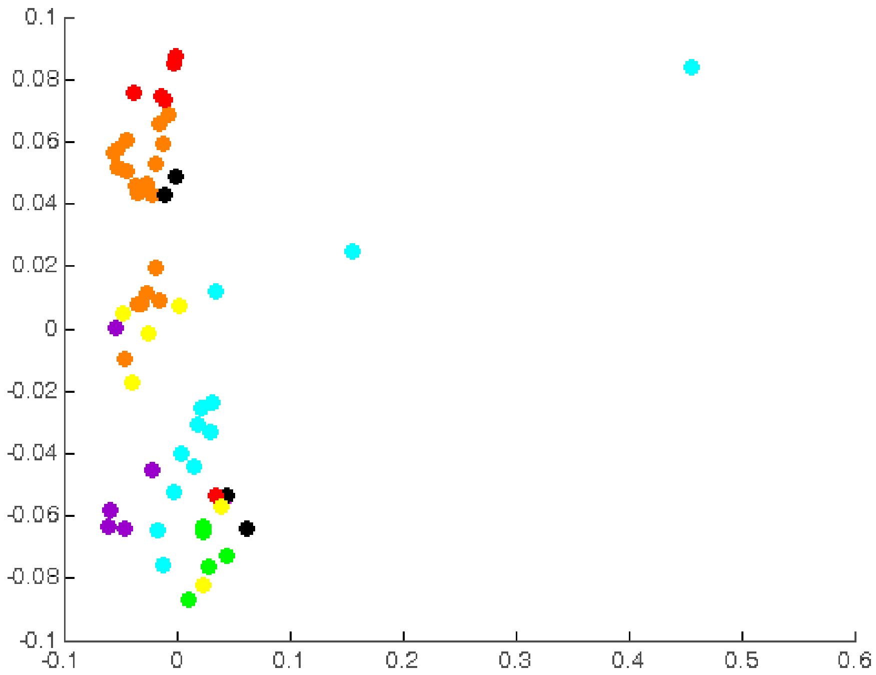


Figure 36. 2-D non-linear embedding output of the molecular data of 57 butterfly species (colors represent families as in Figure 35).

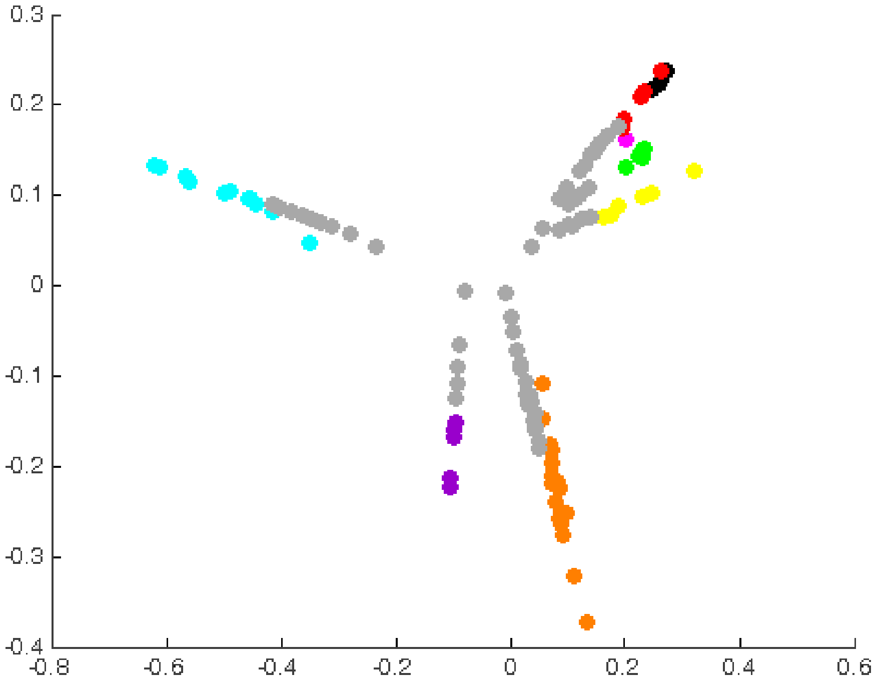


Figure 37. 2-D non-linear embedding output of the molecular data of 57 butterfly species including the generated ancestor sequences (gray color represents the ancestor sequences, other colors represent families as in Figure 35).

2.10. Concluding Remarks

These results support our claim about the necessity of ancestor sequences to yield accurate embeddings with clear functional/evolutionary separations. Note that in our case, it is mostly not possible to use a method for the construction of the ancestral nodes prior to non-linear embedding due to the obscurity of the test data. In the case of butterfly phylogeny, there was a considerable amount of information regarding both molecular and morphological properties about these well-studied species and Walberg *et al.* carefully incorporated this information to the analysis along with the optimization of the method, and obtain the phylogenetic relations at the end.

As a result of these analyses, we decided to discontinue the employment of non-linear embedding in the analysis of biomolecular sequences. Instead, a more complex thresholding operation was decided to be set to modify the input connections and then the concepts of graph theory -without the vector space embedding- were employed in order to achieve accurate functional and/or evolutionary separation of biomolecular sequences.

CHAPTER 3

2-D THRESHOLDING OF THE CONNECTIVITY MAP FOLLOWING THE MULTIPLE SEQUENCE ALIGNMENTS OF DIVERSE DATASETS

One key prerequisite to acquire a meaningful output from multiple sequence alignment procedure is to have a considerable amount of similarity between the input sequences. Multiple sequence alignment algorithms shape the alignments around these shared sequential features. If one or more of the input sequences lack this shared feature, these sequences cannot be aligned to the rest accurately in any way. The presence of non-homologous sequences sometimes misleads the propagation of the alignment and damage the output. This condition is especially reflected as errors on the phylogenetic trees drawn after the alignment. Remote sequences usually end up on irrelevant regions on the tree indicating false relations. Moreover, these sequences may lead to inaccurate branch length predictions for the whole tree. As a result only the sequences that contain a specific feature -or features- are given to the procedure. This inhibits the analysis of large datasets composed of both similar and diverse biological sequences such as whole genomes of proteomes of organisms. An exhaustive preliminary study regarding the split of the dataset into highly similar sequence groups is usually necessary and this often is handled in a guided manner using a BLAST like algorithm (Altschul, 1990) and a vast database of confirmed known sequences. Even when there are no remote sequences in the dataset, the presence of fragments of homolog sequences (frequently encountered in online databases) usually leads to the same occasion due to the obscurity of the relations between the fragments.

A connectivity map shows the pairwise relations of all possible combinations of samples in the dataset. Presence of a connection between a sequence pair indicates a significant homology in-between. On the other hand, absence of a connection indicates the lack of a significant similarity. An accurate connectivity map may yield the accurate classification or clustering of input samples. After a multiple sequence alignment operation, pairwise evolutionary distances are inferred using an evolutionary model for the phylogenetic tree construction procedure. At this point, all sequence pairs are assumed to be connected since usually very similar sequences are given to the multiple sequence alignment. As also mentioned above, if there is a remote sequence or sequences in the dataset (or fragments of sequences) this assumption leads to false homology detections, erroneous pairwise distances and finally an inaccurate phylogenetic tree inference.

Here we proposed a method to make sense out of Multiple Sequence Alignments of datasets composed of sequences from different families (including the sequence fragments) using similarity thresholding with probability distribution techniques. At the end, the sequences are split into meaningful clusters in an unsupervised way using no information other than the sequences themselves. These sequence groups (consisting of homolog proteins) then can be subjected to the multiple sequence alignment process separately to obtain accurate alignments.

This is done by first, creating a new dataset by shuffling the elements of the original set and subjecting both sets to Multiple Sequence Alignment procedure separately. Second, generating 2 dimensional histograms consisting of pairwise evolutionary distances and the number of pairwise overlapped sites (number of positions without gaps on pairwise comparisons) for the original and shuffled datasets separately. Third, drawing threshold curves on histograms using mean and standard deviation values of pairwise evolutionary distances. Fourth, calculating the probability distributions regarding the rejection of pairwise connectivities at each threshold; and decision making using a Receiver Operating Characteristics curve (Lasko *et al.*, 2005).

The method was applied on the Multiple Sequence Alignment output of a large dataset consist of 18011 human protein sequences (the same dataset from the previous analysis). The dataset contains both similar and considerably distant (up to 100% sequence divergence) proteins. At the end of the procedure, the recovered connections were compared with Gene Ontology associations (The Gene Ontology Consortium, 2000) of these proteins to observe the biological relevance of the method. Finally, the method was employed to solve a common real world task: the functional clustering of protein sequences. A gold standard dataset (Brown *et al.*, 2006) was analyzed by clustering the proteins sequences within, measuring the clustering performance and comparing it with a classical clustering operation.

The employed methods are expressed in detail in the next part of this chapter followed by the results and discussion part and a conclusion.

3.1. Methods

A flow diagram including the steps of the method is given in Figure 38.

3.1.1. Shuffled Dataset Creation

Shuffled dataset was created by shuffling the elements of each amino acid sequence from the original test dataset randomly. The shuffling operation was applied on the sequences separately so the length and amino acid composition of each sequence was preserved. The shuffled dataset contained the same number of sequences as the original dataset.

The shuffled dataset was used as a reference to represent unreliable connectivities that should be discarded. Since the elements of the sequences in this dataset were shuffled randomly, any inferred evolutionary relationships between these sequences were assumed to be emerged purely by chance.

3.1.2. Pairwise Evolutionary Distance Inference and the Calculation of Pairwise Alignment Overlaps

Right at the beginning of the procedure, we assumed that, there was a significant homology between all sequence pairs in the dataset. In other words, pairwise connectivity map was fully connected at the starting point. Most probably, some of the sequence pairs have no homology in-between, yet it was not known which ones at this point. What sought here was an indicator to measure the pairwise similarities to decide the existence or absence of significant homology. Pairwise evolutionary distance was a suitable measure to detect this similarity. Evolutionary distances close to zero, signal strong homology and as the evolutionary distances increase, homology diminishes. Since it’s usually not possible to know the real evolutionary distances between biological sequences, they are inferred from the pairwise sequence distances using substitution models -as mentioned in the previous chapters-. In this analysis, evolutionary distances were inferred using Kimura amino acid substitution model (Kimura, 1983) with the correction for multiple substitutions option.

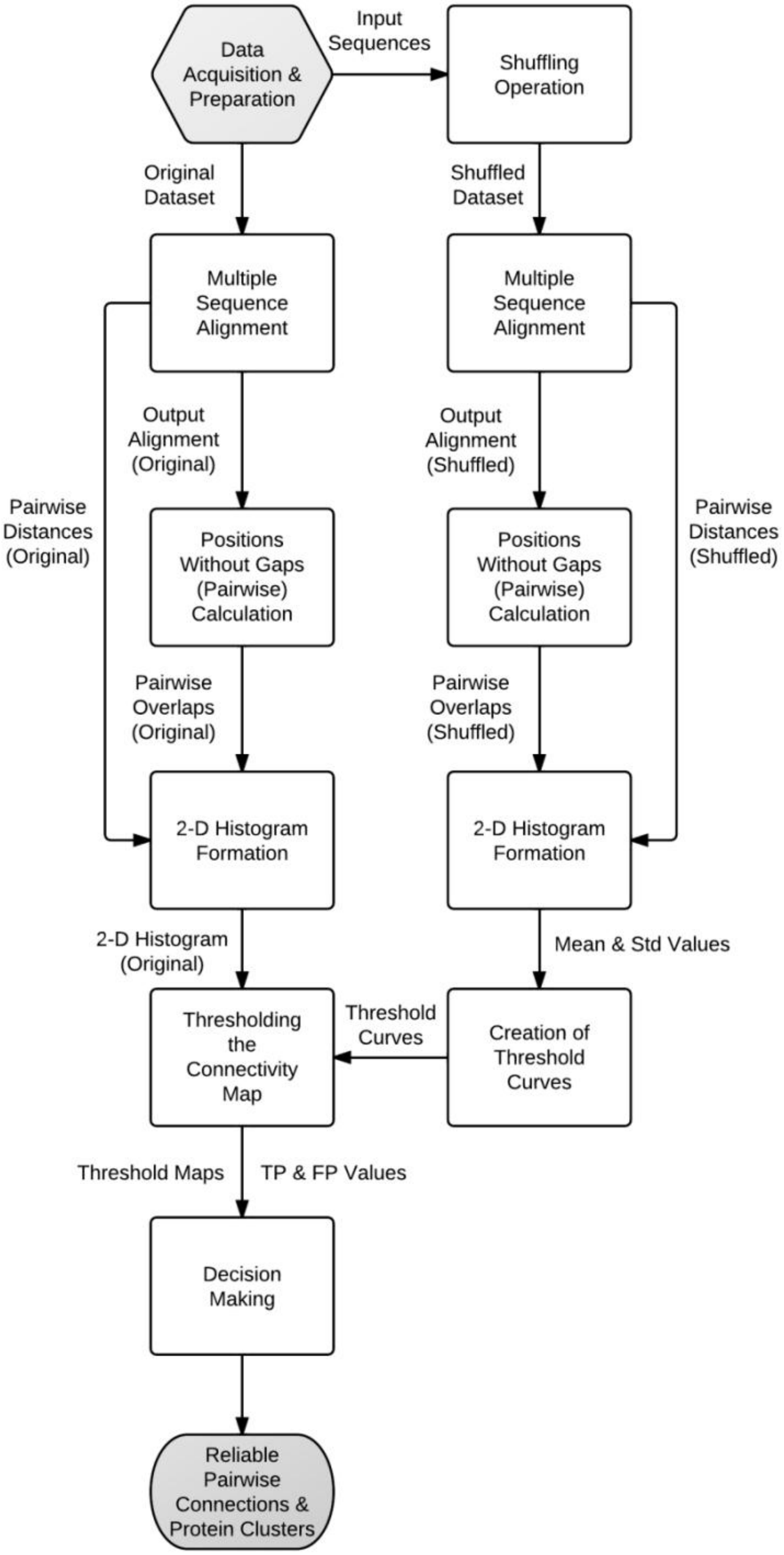
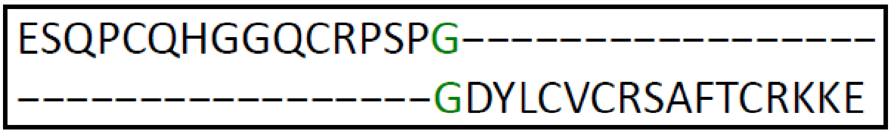

Data Acquisition & Preparation
Input Sequences
Shuffling Operation
Original Dataset
Shuffled Dataset
Multiple Sequence Alignment
Multiple Sequence Alignment
Output Alignment (Original)
Output Alignment (Shuffled)
Pairwise Distances (Original)
Pairwise Distances (Shuffled)
Positions Without Gaps (Pairwise) Calculation
Positions Without Gaps (Pairwise) Calculation
Pairwise Overlaps (Original)
Pairwise Overlaps (Shuffled)
2-D Histogram Formation
2-D Histogram Formation
2-D Histogram (Original)
Mean & Std Values
Thresholding the Connectivity Map
Threshold Curves
Creation of Threshold Curves
Threshold Maps
TP & FP Values
Decision Making
Reliable Pairwise Connections & Protein Clusters

Figure 38. Flow diagram of the thresholding connectivity map method.

In the multiple alignments of large datasets, the output alignment is usually quite lengthy. As a result, some of the sequences (especially short ones) may end up on different parts of the output alignment. It’s not possible to infer evolutionary distances of these proteins. In theory these sequences are diverged from a common ancestor so long before that the accumulated mutations makes it impossible to infer any similarity. At some other times, two distant sequences have matches (or mismatches) on a few positions (and there are gaps at the rest of the positions). After an inspection it was discovered that among all pairwise combinations in the output multiple alignments of test datasets, there were many occasions that only 1 or 2 sites were occupied by amino acid on both sequences -in other words gapless positions-. If this site gave a match, the evolutionary distance was inferred as zero between these 2 sequences since the remaining sites (including gaps) were not counted at all. However this information was not reliable as these sequences were not homologous. This case was also addressed in the previous chapter. Figure 39 shows a sample case for this phenomenon. The rows represent 2 protein sequences taken out from a test Multiple Sequence Alignment output. The position shown in green color is the only site available for calculating the evolutionary distance. Since it’s a match, the distance was calculated as zero.



```
ESQPCQHGGQCRPSPG----------------
---------------GDYLCVCRSAFTCRKKE
```

Figure 39. A sample case that leading to an unreliable evolutionary distance inference after the Multiple Sequence Alignment process.

Unreliable cases such as this one should be eliminated together with the connectivities with elevated pairwise distances. The proposed solution was eliminating the unreliable connections by thresholding the connectivity map over both pairwise evolutionary distances and the number of sites without gaps (pairwise overlaps). Similar to the pairwise evolutionary distances, the number of sites without gaps were calculated for each sequence pair in the original and the shuffled datasets (e.g. the total number of overlapped positions for the case in Figure 39 is 1).

3.1.3. 2-D Histogram Formation

A 2 dimensional histogram is a visual representation of the distribution of data just like a normal histogram. It differentiates from a normal histogram on the number of features the data is distributed upon. In a 2 dimensional histogram, the distribution of the data is shown at the intersection of two feature intervals. In the plot, the discrete intervals of the first feature are given on the horizontal axis and the ones for the second feature are located on the vertical axis. One bin is formed for each *feature 1* and *feature 2* discrete interval combination and the number of points fall between the ranges of features for that bin appears inside. For the sake of visuality 2 dimensional histograms often created as intensity graphs instead of bars.

In this study, horizontal axis of the 2 dimensional histogram represents the total number of sites without gaps for each pairwise comparison. Vertical axis represents the inferred pairwise evolutionary distances. These axes are both divided into 100 discrete intervals making 10000 bins in total. To create the intensity contrast, grayscale colormap is chosen. More populated bins are represented by a darker color and sparsely populated bins by lighter colors.

First, linearly scaled intervals were used for the colormap but this resulted in visually poor plots. Later, a logarithmic scale was preferred for the coloring intervals producing satisfying visuality. Figure 40 shows a sample 2 dimensional histogram with the intensity scale on the right side.

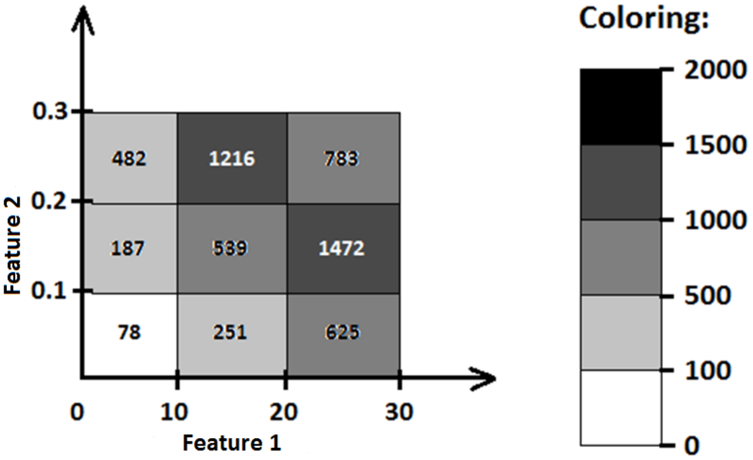


Figure 40. Sample 2-D grayscale intensity histogram (intensity scale on the right).

3.1.4. The Thresholding Operation

Standard deviation and mean values of the distribution of “pairwise distances” on each “number of sites without a gap” interval were employed in order to create the threshold curves on the 2 dimensional histograms. Equation 3.1 shows the formulation of the threshold curves.

$$\boldsymbol{T}_i = \boldsymbol{M} - \boldsymbol{S} * i \tag{3.1}$$

T_i is the i^{th} threshold curve (1 to 20), M is the mean pairwise distance –a constant value-, S is the standard deviation curve of the distribution of distances.

Standard deviation curve creation was carried out column-wise on the 2 dimensional histogram of the shuffled dataset. For each discrete “number of sites without a gap” interval, a standard deviation value was generated for the pairwise evolutionary distances. These successive values formed the standard deviation curve. Figure 41 is a representation of the standard deviation curve formation. Figure 41 (a) is a sample 2-D histogram, each red colored rectangle encloses a column of pairwise distances that a standard deviation value is calculated upon. Figure 41 (b) is the standard deviation curve drawn from the sample 2-D histogram.

Use of the standard deviation curves during the formation of the threshold curves allowed capturing the shape of the edge of the crowded portion in the 2 dimensional histogram of the shuffled dataset. This was useful for separating the meaningless/unreliable distances from the reliable distances around this edge.

Using this method, 20 different threshold curves were created that scan the area below the mean distance curve. In addition to this set, 20 new curves were created to scan the area above the mean distance curve with the formula given in Equation 3.2, making 40 curves in total.

$$\boldsymbol{To}_i = \boldsymbol{M} + \boldsymbol{S} * i \tag{3.2}$$

To_i is the i^{th} threshold curve (1 to 20) above the mean pairwise distance curve, M is the mean pairwise distance –a constant value-, S is the standard deviation curve of the distribution of distances. To avoid confusion in curve names, all of these 40 curves were named $\sigma_{1,2,3,....,40}$.

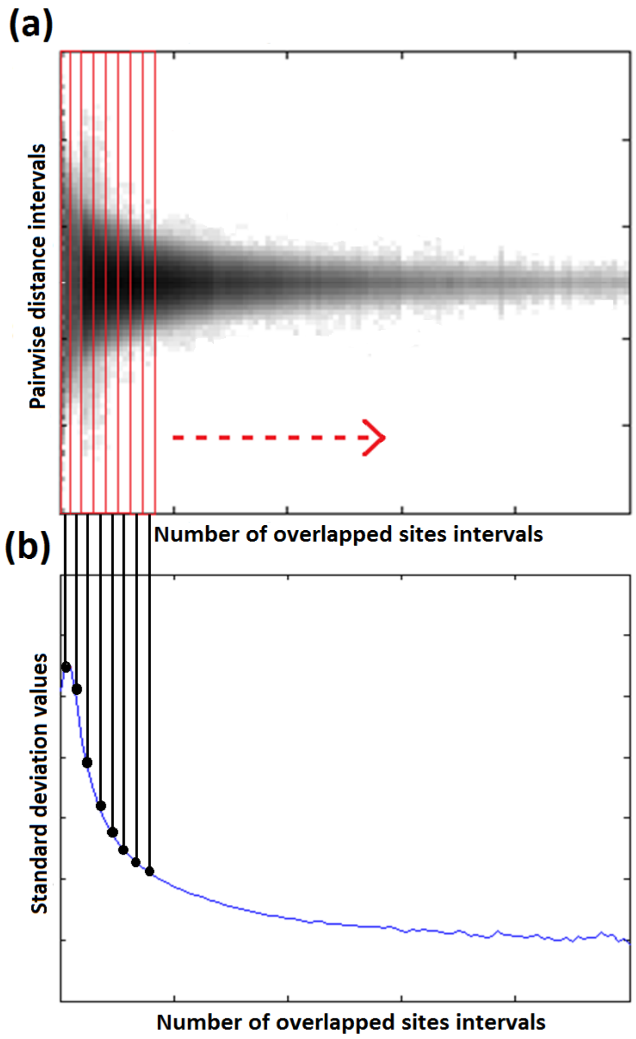


Figure 41. (a) A sample 2-D histogram, (b) the standard deviation curve drawn from the sample histogram.

3.1.5. Decision Making Step

A Receiver Operating Characteristic (ROC) curve (Lasko *et al.*, 2005) is a plot of true positive vs. false positive rates, in other words a plot of sensitivity for a binary classifier system. It is used in order to find an optimum cut-off with the desired specifications between 2 classes where the distributions significantly overlap (Lasko *et al.*, 2005). The two classes simply named as positives and negatives. The sole purpose here is to determine a point (cut-off) that the points on one side are assumed to be positives and to be negatives on the other side. This leads to the formation of 4 different groups namely true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN). True positives are the points that belong to the positives group that are also labeled as positives correctly at the decision making step. False positives truly belong to negatives group but incorrectly labeled as positives. “True negatives” is the name given to the samples in the negatives group that are also labeled correctly as negatives. And lastly, false negatives truly belong to positives group but marked incorrectly as negatives.

As the cut-off slides from left to the right, more and more points are labeled as negatives. If the final cut-off is too low, nearly all of the positives may be recovered but along with them, many points in negatives are also labeled as positives. On the other hand, if the cut-off is to be selected too high, most of the negatives may be eliminated successfully with the cost of discarding the real positives at the same time. Generally the optimum cut-off is selected at the point, the slope of the ROC (true positives rate vs. false positives rate) curve equals to one. This point corresponds to the spot where the rate of eliminating real negatives and capturing real positives are equal.

In our study, positives group corresponded to the real (from the original dataset) connections whereas negatives group corresponded to random connections (from the shuffled dataset). Motivation here was that, all connections coming from the shuffled dataset were assumed to be meaningless/unreliable; whereas, the ones from the original dataset contained both reliable and unreliable connections. In order to separate the reliable ones from the rest, a continuously increasing threshold was applied to the pairwise connections of both groups (using the previously generated curves) where the connections with the values exceeding the corresponding threshold were discarded. The presence of a pairwise connection meant, there was a significant homology between the sequence pair. Similarly, when a connection was discarded and absent as a result, it was assumed that the corresponding sequences are non-homologs. At the optimum point, most of the connections from the shuffled dataset should be discarded and the ones left from the original set were to be assumed as the reliable connections.

To this end; TP, TN, FP and FN values were calculated from the number of real and random connections discarded and remained at each threshold together with the total number of real and random connections. The ROC curve was plotted using TP and FP rates. At this point, a cut-off should be decided regarding the slope of the ROC curve. For the automatic selection of the cut-off, the point where the slope equals to 10^5 or the point where all of the random connections were eliminated (whichever comes first) was chosen.

At this point, the connectivity map became disjointed due to the removal of inter-connections. This operation forms groups of homolog sequences.

3.1.6. Calculation of the Statistical Performance Measures

Statistical measures were employed in order to evaluate the performance of the method on different tasks. These parameters consist of Recall, Precision and F-score. Recall and Precision are composed of different combinations of TP, FP and FN values. F-score incorporates both recall and precision to display the performance on a single parameter and frequently employed in clustering studies (Paccanaro *et al.*, 2006; Nepusz *et al.*, 2010; Wittkop *et al.*, 2007 and Apeltsin *et al.*, 2011). The calculation of precision, recall (sensitivity) and F-score are given in equations 4.3, 4.4 and 4.5 respectively.

3.2. Results and Discussion

3.2.1. Analysis of the Large Human Protein Dataset

Human protein sequence dataset from the previous analyses was revised and used in this analysis as well. The revision is done by updating the sequences from Gene Ontology Database. Besides, the sequences with length lower than 100 amino acids and higher than 10000 amino acids were assumed to be outliers and removed from the dataset. The final dataset consisted of nearly 17793 human protein sequences. Large datasets with sizes similar to this one usually are hard cases for techniques that rely on similarity measurements. Next, the shuffled dataset was created using randomly permuted elements of the amino acid sequences of the original dataset as explained in methods part in detail.

ClustalW2 v2.0.10 software package (Larkin *et al.*, 2007) was used for the global Multiple Sequence Alignment procedure for the original and the shuffled datasets separately with the default options. Pairwise evolutionary distances were inferred using the built-in algorithm of ClustalW2 with Kimura amino acid substitution model (Kimura, 1983) with the correction for the multiple substitutions.

By comparison of the resulted alignments for the original and the shuffled datasets, it was observed that the length of the alignment was significantly shorter -in other words less gappy- for the shuffled dataset. This result was expected beforehand. Since no meaningful alignment could be obtained from the shuffled dataset in any way, Multiple Sequence Alignment algorithm has chosen not to insert as many gaps as in the alignment of the original dataset in order to avoid gap costs. From the resulted alignments, 2-D histograms were created for the original and the shuffled datasets with the procedure described in the methods part.

The aim of thresholding the connectivity map was to eliminate the unreliable pairwise distances resulting from distant relationships or poor alignment. In a classical case with the Multiple Sequence Alignment of a few closely related proteins, this procedure would be unnecessary since the probability of getting inaccurate pairwise alignments between closely related sequences were quite low. Also with a few moderately diverged sequences, thresholding only the pairwise distance matrix to discard unreliable connectivities would be sufficient. For our case, where there were 17793 sequences that span nearly the entire functional spectrum of the human proteins discovered so far, the resulted Multiple Sequence Alignment was so large that especially some of the short amino acid sequences didn’t have any overlap on each other to calculate a pairwise distance. As also mentioned in the previous chapters, a more misleading case was where some of these sequence pairs have an overlap on just 1 or 2 residues. If there was a match on the only gapless position -since there are no mismatches-, pairwise distance between these sequences ended up as zero, even though the sequences were quite diverged from each other in their entirety. This was an extreme case but the connectivities between diverged sequences should be eliminated which still was quite frequent. To solve this problem we introduced the thresholding of the connectivity map regarding 2 different parameters. First parameter was the pairwise evolutionary distances and the second one was the overlap of the sequence pairs on the alignment output, in other words the total number of gapless sites on the pairwise comparison of the multiply aligned sequences.

2 dimensional histograms were created for the original and the shuffled datasets with the procedure described in the methods part. On these 2 dimensional histograms, clumped regions were observed and the discrepancies between the histograms of the original and the shuffled datasets were tried to be extracted.

Figure 42 represents the 2 dimensional histogram of the original dataset on the left and shuffled dataset on the right for the human protein dataset -both in log scale to increase visuality of the difference- where the horizontal axis represents the number of sites without gap intervals on pairwise comparisons and the vertical axis represents the pairwise evolutionary distance intervals. There is a visually distinct difference between the histograms around 0-2000 number of overlaps and 0-2 pairwise distances. This region on the original dataset histogram represents the reliable connections. However the region was not a clear cut as the shuffled datasets histogram also has representatives in the region. So this gray area was handled with probability distribution techniques.

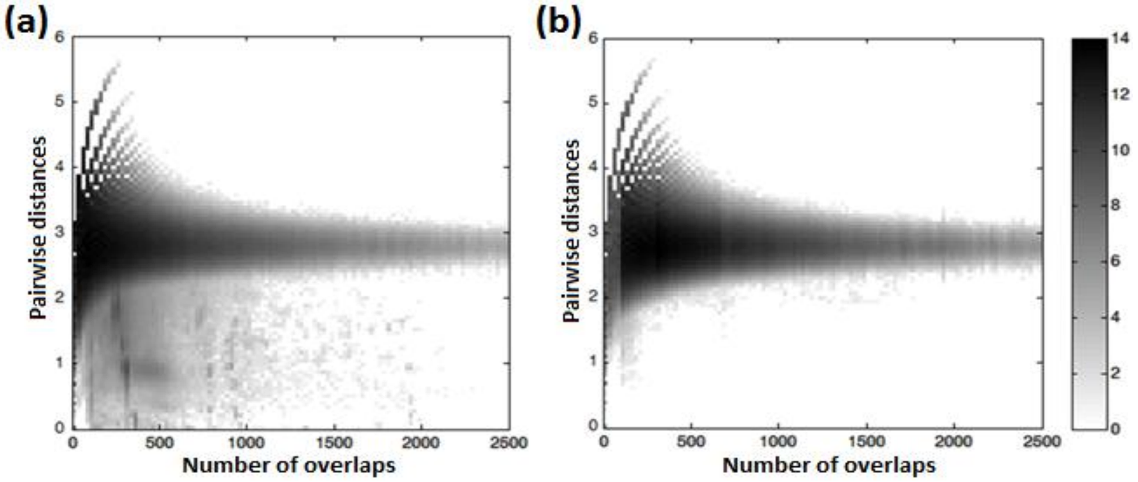


Figure 42. 2-D histograms of (a) the original and (b) the shuffled datasets in log scale formed after the Multiple Sequence Alignment of 17793 Human proteins.

Standard deviation curves were calculated as explained in methods part for the thresholding operation. In order to eliminate the noise on the curve, a normal (Gaussian) distribution model is fit on the curve (Bryc, 1995). The most suitable fit was found on the third order General Gaussian Model as shown in Equation 3.3. The standard deviation curve of the shuffled dataset in this analysis and the Gaussian Model fit are drawn together in Figure 43.

$$f(x) = a_1 * e^{-}\left[\frac{(x-b_1)}{c_1}\right]^2 + a_2 * e^{-}\left[\frac{(x-b_2)}{c_2}\right]^2 + a_3 * e^{-}\left[\frac{(x-b_3)}{c_3}\right]^2 \quad (3.3)$$

Coefficients were a_1=0.34, b_1=-37.29, c_1=139.2, a_2=0.2989, b_2=-399.1, c_2=589.9, a_3=121.4, b_3=-41610, c_3= 16360 and for the goodness of the fit, R-square was 0.9994.

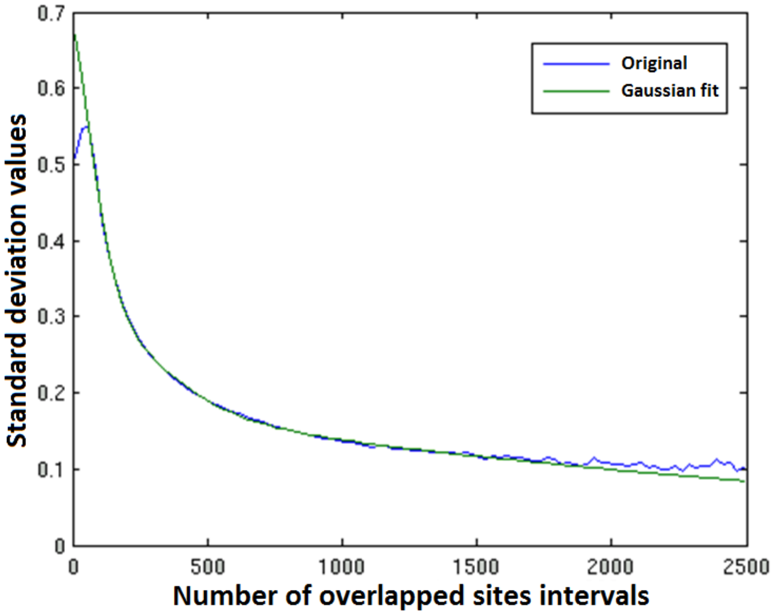


Figure 43. The standard deviation curve of the shuffled dataset (blue: the original curve green: Gaussian fit).

The threshold curves and the ROC curve are created following the procedures explained also in the methods part. Figure 44 shows the threshold curves $\sigma_{1,2,3,....,40}$ used for the creation of the ROC curve, on the 2 dimensional histogram of the shuffled dataset where the horizontal axis represents the number of sites without gap intervals on pairwise comparisons and the vertical axis represents the pairwise distance intervals as before.

In this analysis, the probability distribution of positives and negatives highly overlapped -shown in Figure 45 in log scale (thresholds vs. the rate of change in the number of connectivities left after thresholding operation in log scale)-. The ROC curve (shown in Figure 46) slope was selected to be 10^5 automatically for the cut-off. This point is shown with the black dot in the ROC curve (Figure 46). The threshold curve that yielded the selected cut-off was σ_{26}. At this cut-off 270 meaningless (≈ 0% of the total) and 213000 real (0.14% of the total) connectivities were left on the connectivity map. At this point, it appeared like most of the connections from the original dataset were eliminated however it’s crucial to mention that the connections from the original dataset were composed of false connections along with the true ones and our aim was to separate these two from each other.

2-D histogram of the original dataset with the selected threshold curve plotted over (blue colored) is shown on Figure 47. All of the pairwise connectivities that has distance and overlap values above the curve were assumed to be unreliable and discarded.

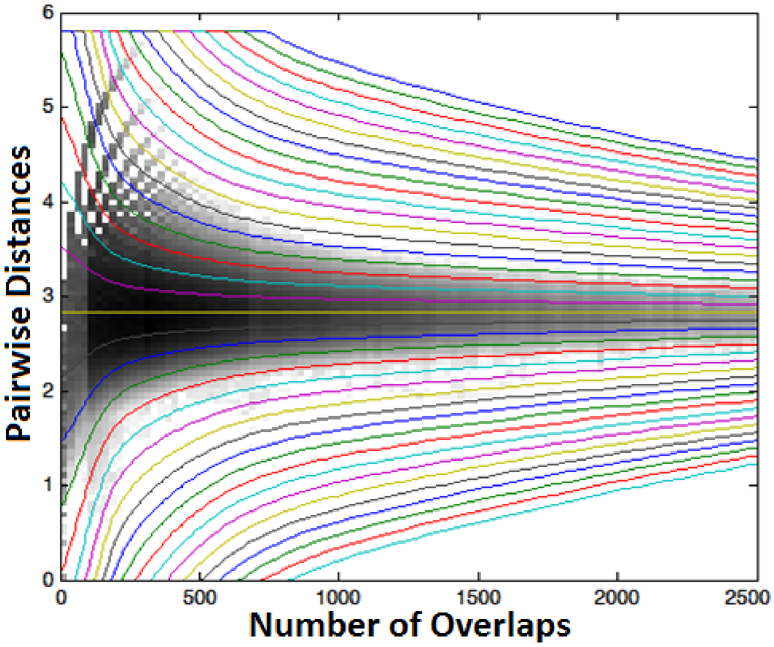


Figure 44. Threshold curves $\sigma_{1,2,3,\ldots,40}$ on the 2-D histogram of the shuffled dataset.

As expected, the threshold connectivity map became disjointed at this point due to the removal of inter-connections. It consisted of components of differing sizes and some singleton points that have no connections left to any other sequence. A component here is defined as a group of sequences that have either direct or indirect connections in-between. A manual examination over some sample components revealed that, each component was composed of similar proteins usually with significant homology. Besides, the inspection over the singleton points in the connectivity map showed that, these were the sequences that could not be aligned to any other sequence significantly in the Multiple Sequence Alignment process.

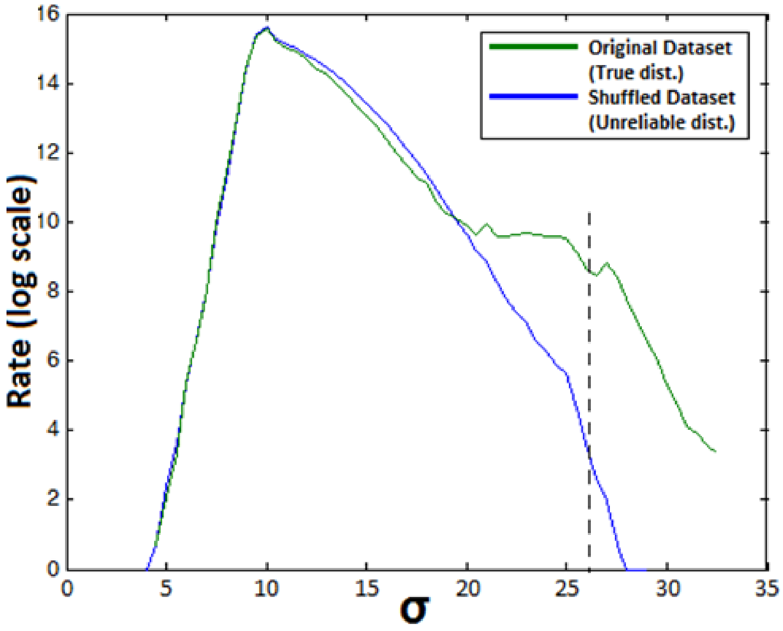


Figure 45. Rate of change curves for the number of remaining connections (in log scale). Horizontal axis represents different thresholds; vertical axis is the rate of change in the number of connectivities left after thresholding operation (in log scale). The dashed vertical line corresponds to the selected threshold.

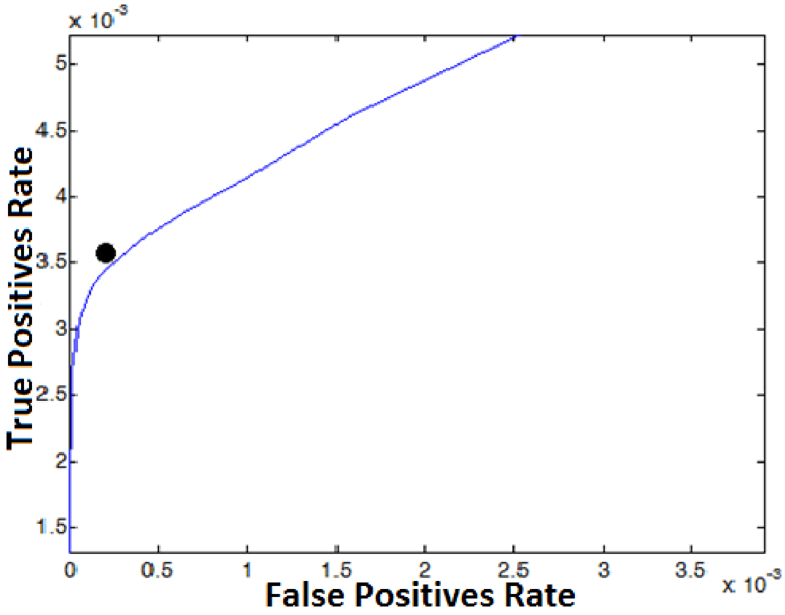


Figure 46. The ROC curve for the thresholding operation (the black dot corresponds to the TP and FP rate values at the selected threshold).

At this point in the study it was clear that, most of the inter-group distances were quite large, unreliable and dumped during the thresholding operation. After the thresholding, 445 components were formed with varying sizes. The largest component contained 476 sequences and the smallest ones contained only 2 sequences.

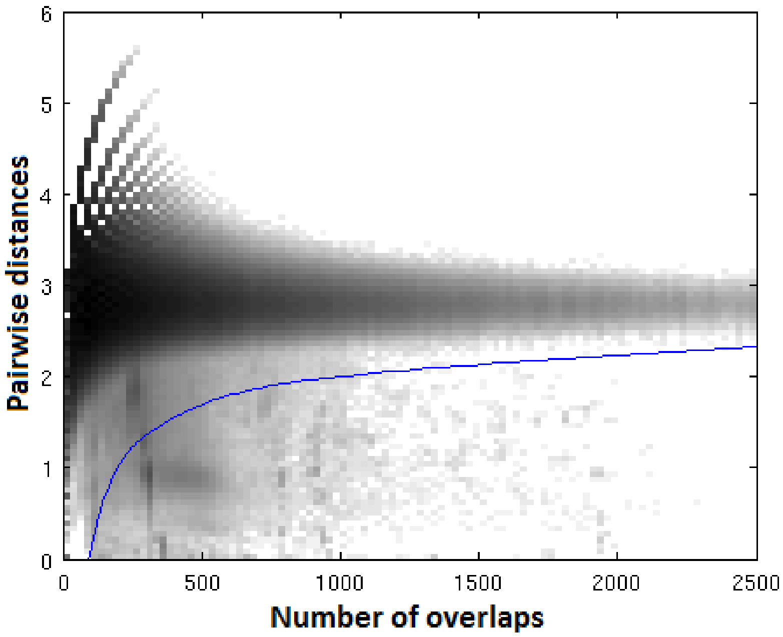


Figure 47. 2-D histogram of the original dataset with the selected threshold curve (σ_{26}) plot over.

In order to examine the biological relevance of our grouping, we tested our recovered true connections against the GO associations of the input sequences. We prepared the reference connection map by searching for shared GO terms between sequences and assuming significant homology (existence of a connection) between these sequences. Any two sequences were assumed to be connected (related) when there is at least one shared GO term in-between. By this way, connections were formed between 37.9% of all possible sequence pairs. We measured performance by counting the true and false connections found in our analysis regarding the reference connections. When we got a connection that was also present in the reference map, we counted a true positive (TP) and when we had a connection that didn't appear in the reference, it was a false positive (FP). We calculated the precision measure (positive predictive value) as given in Equation 4.3. A precision value of 1 meant all of the recovered connections were accurate. Our precision output was 0.981 whereas the same number of connections selected randomly resulted in 0.426 precision. Also to show how our method disposed meaningless connections, the same test was applied directly to the pairwise evolutionary distance (Kimura model) output of the multiple sequence alignment procedure (a classical 1-D thresholding). The distance map was threshold with the disposal of the distances greater than 2. This was a reasonable value to assume homology and also the remaining number of connections in the map appeared to be nearly the same as our result providing the fair comparison of the performances. Precision for the classical thresholding over the pairwise distances was found as 0.799. The difference was nearly 20% in favor of our method which was a considerably significant improvement.

The results supported our claim as thresholding the pairwise connectivity map over 2 dimensions (the number of positions without gaps in the pairwise comparisons of aligned sequences and inferred evolutionary distances) after the multiple sequence alignment procedure, assures the disposal of false homology detections and help make sense out of multiple alignments of large and mixed datasets. In addition, the detection of the potential multiple sequence alignment disrupters (distant sequences and homolog sequence fragments in the dataset) is provided by the proposed method.

3.2.2. Clustering of the Reference Dataset

Clustering of biomolecular sequences is an active area of research where the sequences are tried to be grouped under evolutionary and/or functional constraints in order to infer the history and functions of the unknown sequences (regarding the known ones). Over the last decade, many clustering algorithms were developed employing different statistical approaches. Some popular methods from the literature are TribeMCL (Enright *et al.*, 2002), Spectral Clustering (Paccanaro *et al.*, 2006 and Nepusz *et al.*, 2010), FORCE (Wittkop *et al.*, 2007) and TransClust (Wittkop *et al.*, 2010).

At the final step of the study, members of a standard dataset composed of 866 manually curated enzymes (in 91 families) (Brown *et al.*, 2006) were clustered and the accuracy of this application was measured (regarding the families that the sequences belong to) and compared with a classical thresholding operation incorporating only pairwise evolutionary distances. This conventional operation acting over 1 dimension takes part in most of the clustering methods -such as thresholding the BLAST (Altschul *et al.*, 1997) e-values-. This dataset was referred as a gold standard set and frequently employed in the testing of clustering algorithms in the literature (Apeltsin *et al.*, 2011; Wittkop *et al.*, 2010 and Miele *et al.*, 2012).

First of all, the sequences were obtained via online material published by Brown *et al.*, 2006. Next, the shuffled dataset was generated and both sets were subjected to the multiple sequence alignment procedure using ClustalW2 v2.0.10 software package (Larkin *et al.*, 2007) with the default options. Then, the pairwise evolutionary distances were inferred using Kimura amino acid substitution model (Kimura, 1983) with the correction for multiple substitutions option. After that, the numbers of overlapped positions on alignments were calculated, 2-D histograms were formed, and threshold and ROC curves were drawn as described in the Methods part. The cut-off was selected automatically at the point where no connections remained from the shuffled dataset. After the thresholding operation, sequences were clustered regarding the recovered pairwise connections. Since the presence of a connection between a sequence pair indicates a significant homology/similarity, these sequences appear in the same cluster. All sequences with a direct or an indirect connection in-between were grouped together. This approach is similar to the widely used graph theory method Connected Component Analysis (Diestel, 2010) that was also employed in biomolecular sequence clustering methods frequently.

Figure 48 (a) and (b) show the 2-D histograms (with the threshold curves plotted over) of the original reference dataset and its shuffled version respectively (in log scale). The true/reliable connections are visible on Figure 48 with dark color just over the baseline of the x-axis. Figure 49 shows the curves for the classical 1-D thresholding operation on the 2-D histogram of the original reference dataset. Notice the curves here are linear and parallel to x-axis since this operation does not incorporate number of overlapped positions.

Table 3 shows the Precision, Recall and F-measure values for the clustering performance of the conventional 1-D thresholding operation (first column) and the proposed method (second column) using the threshold curve selected automatically. For a fair comparison between the proposed method and the conventional thresholding operation, the average clustering performances regarding all threshold curves are given in the third and fourth columns. Best F-measures are given in bold. As seen from Table 1, the clustering performance was increased nearly 6.5% (F-measure: 0.827 to 0.882) when the proposed method was employed instead of the conventional thresholding with automatically selected threshold curve. On the other hand, the average clustering performance was increased around 7.9% (F-measure: 0.712 to 0.768) with our method.

These results indicate the effectiveness of our proposed approach in the functional clustering of amino acid sequences.

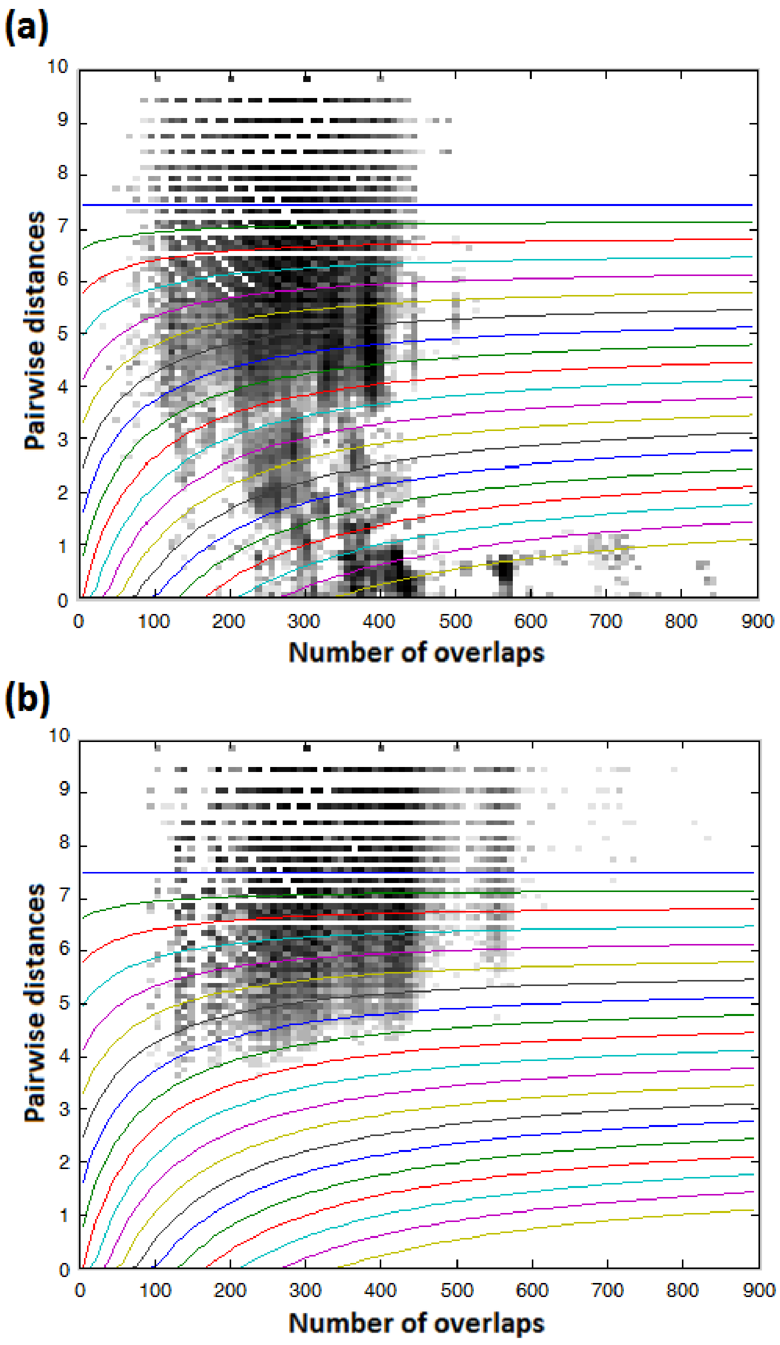


Figure 48. 2-D threshold curves on the 2-D histograms of (a) the original and (b) the shuffled gold standard dataset.

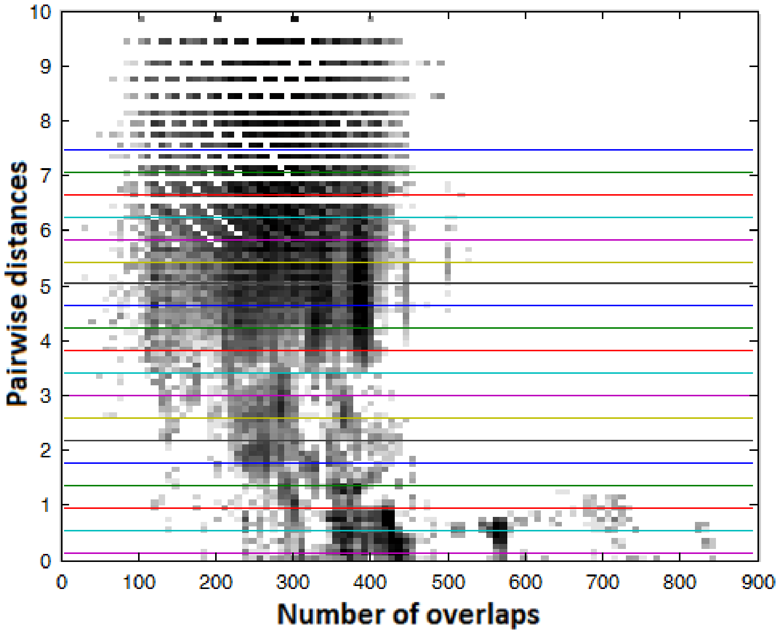


Figure 49. 1-D threshold curves on the 2-D histogram of the original gold standard dataset.

Table 3. Clustering performance measures for the standard dataset after the conventional 1-D and 2-D thresholding operations.

	At the selected curve		Average of all curves	
	1-D Threshold	**2-D Threshold**	**1-D Threshold**	**2-D Threshold**
Precision	0.711	0.794	0.700	0.723
Recall	0.990	0.991	0.892	0.935
F-measure	0.827	**0.882**	0.712	**0.768**

3.3. Concluding Remarks

As mentioned previously, Multiple Sequence Alignments of large datasets (consisting of thousands of sequences) exert enormous computational loads. The load is reflected to the user as elevated computation times (e.g. it took nearly 30 days to process 17793 human proteins). Due to this problem, parallelization of Multiple Sequence Alignment process on the computers with multiple cores is an active area of research. ClustalO package (Sievers *et al.*, 2011) is popular parallel Multiple Sequence Alignment implementation. One important problem to be solved about these algorithms is the parallelization of the progressive alignment step of the procedure. Since this step is iterative (the result of the previous operation is required to process the next one), the parallelization is nearly impossible.

At this point, in order to get rid of the high amount computational load and achieve practical computation times, we decided to employ only pairwise alignment at the initial analysis of the dataset. We planned to achieve the separation of the input sequences into biologically meaningful groups regarding their pairwise alignments and focus each group individually to infer functional relationships. This method is explained in detail with the results of its applications of different datasets (including human proteins) in the next chapter.

CHAPTER 4

AUTOMATIC IDENTIFICATION OF CONSERVED REGIONS IN LARGE DATASETS INCLUDING REMOTE PROTEIN SEQUENCES

Graph theory concepts are frequently incorporated to the similarity based sequence analysis methods. In these methods, biomolecular sequences are treated as vertices of a graph and the presence or absence of a significant statistical sequential similarity between the sequences determines the existence of a path in-between forming a connectivity map. This significant statistical sequential similarity then corresponds to conserved features shared by the two sequences.

GeneRAGE (Enright and Ouzounis, 2000) is one of the earliest methods to employ this concept in similarity based methods in an efficient way where sequence alignment and single linkage cluster-ing are combined to cluster large protein datasets in a simplistic way. TribeMCL (Enright *et al.*, 2002) is more efficient and complex method from the same research group that incorporate Markov Clustering for rapid and accurate clustering of protein sequences also to address multi-domain sequences. Apeltsin *et al.*, add edge weight distribution with automated threshold selection to initial similarity network and manage to increase the clustering performance of fast MCL to that of novel highly efficient clustering algorithms on a gold standard dataset (Apeltsin *et al.*, 2011). However, this threshold heuristics should be tested on other datasets composed of different sequences. Spectral Clustering (Paccanaro *et al.*, 2006) -and its modified fast implementation with user-interface SCPS (Nepusz *et al.*, 2010)- is an efficient and widely used algorithm for biological sequence clustering. It also is a Markov Clustering algorithm with a global approach. Actually, Spectral Clustering is quite similar to TribeMCL. The differentiation between these methods lies within the propagation of the Markov chain on the graph. A comparison in-between these algorithms, is given in (Paccanaro *et al.*, 2006).

FORCE (Wittkop *et al.*, 2007) and its user interface TransClust (Wittkop *et al.*, 2010) is a powerful method that use pairwise similarity measures and weighted cluster editing to achieve accurate clustering of large datasets. In this method the input graph is edited into a transitive graph regarding the minimization of a cost function (Wittkop *et al.*, 2007). The user is required to enter a similarity threshold at the input level. The authors have fixed a semi-automatic parameter estimation procedure (for similarity threshold) that works efficiently when a gold-standard subset exists for the dataset. On the other hand, they reported an issue about parameter standardization for the analysis of unknown sequences. Also the computation times are relatively higher than the methods mentioned beforehand.

HiFiX (Miele *et al.*, 2012) is a novel and efficient clustering method that acts over the entire length of input sequences instead of regional homogeneity. This makes the method suitable to analyze multi-domain proteins and partially prevent input parameter determination problem. It's mainly build to process very large datasets composed of thousands of families to infer phylogenetic relations. In this method, the sequences are first assigned to clusters of pre-families (optimizing sensitivity) followed by merging of clusters into families with the help of qualitative multiple sequence alignment evaluations (Miele *et al.*, 2012). The method performs as good as or better than the other novel clustering algorithms on the gold standard set and significantly better in clustering large multi-domain bacterial dataset. However, the first step of the method still contains parameters for users to decide and also the method is not suitable to detect relationships of partially alignable remote sequences (Miele *et al.*, 2012).

The Connected Component Analysis is a widely used graph theory application (Diestel, 2010), employed both as a stand-alone method and as an intermediary step in other sequence clustering methods. In an undirected graph *G*, two vertices *a* and *b* are connected if there is a path from *a* to *b*. A connected component is a connected sub-graph of *G* that contains all the vertices with a path to each other directly or undirectly. If there are paths from a to *b* and *b* to *c* (even if not *a* to *c*), all of these vertices are located in the same connected component (Diestel, 2010). Sequences in a connected component are assumed to share a significant similarity and belong to the same cluster. Yet, when used on multi-domain proteins, unrelated sequences are usually grouped into the same clusters due to the domain-chaining effect (Mohseni-Zadeh *et al.*, 2004).

The Cluster-C (Mohseni-Zadeh *et al.*, 2004) method efficiently avoids the chaining effect by incorporating maximal clique extraction on the connectivity map following a pairwise similarity search. A maximal clique (fully connected sub-component) is a subset of an undirected graph where each vertex is directly connected to every other vertex. Note that unlike connected components, a vertex (sequence) may exist in more than one maximal clique. This allows capturing a second, third or so on features located on a sequence by looking at its involvement in different maximal cliques. Figure 50 shows a representation of three maximal cliques within a connected component on a 2 dimensional undirected graph. Black dots represent vertices and red lines represent edges in-between. Large green circle shows that all points belong to a single component since there is either direct or in-direct connection between all points. Small blue circles represent 3 maximal cliques two of which share a point. The point with a yellow mark inside is the shared vertex belong both to clique 2 and 3.

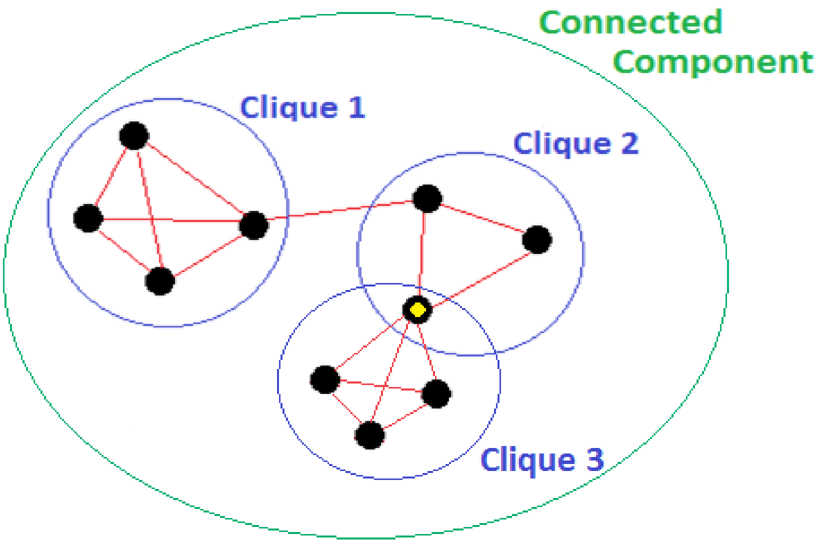


Figure 50. Representation of a connected component and 3 maximal cliques inside, on an undirected graph.

The incorporation of maximal clique finding into clustering suffers from practical problems especially on large datasets, such as clique redundancy. In theory, sequences in each maximal clique should contain at least one unique conserved feature. In practice however, the maximal cliques are redundant, with a shared region represented in more than one maximal clique. This arises from the accidental removal of pairwise connections (during the thresholding) due to highly remote homology or just poor alignment between some of the homolog sequence pairs. This inevitably decreases the accuracy of the results and burdens a heavy computational load.

Three important bottle-necks stand out in general motif discovering approaches. The first one is the treatment of multi-domain proteins. Most of these methods are optimized to process single domain sequences and the assignment of multi-domain proteins into clusters is sometimes problematic. The second issue is the standardization of the input parameters: The behaviors of these algorithms are controlled by several parameters to be provided by the user at the input level. However, in the absence of a known standard about the input sequences, selecting the correct parameters becomes nearly impossible. As a result, the accuracy of the results decline. Finally, in most of the methods, no further processing can be applied upon remote input sequences left out as singleton points after the initial similarity search. In our method, we address all three of these issues.

In this chapter, we propose a new method for automatic separation of large collections of diverse sequences into biologically relevant groups for accurate functional assignments by exposing highly conserved regions and associating them with the input sequences using statistical grouping and graph theory concepts. This is done first by grouping the sequences in connected components of significant similarities regarding their pairwise alignment e-values and then, splitting the sequences in each connected component into fully connected sub-components (maximal cliques) consisting of sequences containing a shared feature. Next, the shared/conserved regions on multiple sequence alignments of the member sequences of each maximal clique were located using a residue conservation scoring algorithm, and conserved region profiles were formed and queried on input sequences. Finally, the associations between the input sequences and the identified highly conserved key regions were presented as a table that can be used to infer relationships between the sequences (as well as between the conserved regions) and to assign functions.

We have tested our method's biological relevance by carrying out clustering on standard datasets (domain sequences) from the SCOP Database (Andreeva *et al.*, 2008) that were used previously in the literature (Nepusz *et al.*, 2010; Paccanaro *et al.*, 2006) and comparing clustering performances to the widely used clustering methods.

Finally we have applied our method on the previously analyzed human protein dataset of 17793 sequences to obtain a global functional relation map of human proteins. The dataset contained both similar and considerably distant proteins. We have evaluated the performance of our method in identifying the functional domains on the input sequences by comparing the identified conserved/shared regions and their associations with input sequences to the reference functional domain associations obtained from Protein Family (Pfam) database (Finn *et al.*, 2010 and Punta *et al.*, 2012) and NCBI Conserved Domain Database (CDD) (Marchler-Bauer *et al.*, 2011). The results revealed that, the discovered conserved regions highly correspond to structural domains present on the input proteins.

The details of the proposed large-scale conserved region discovery method - namely Protein Function Assignment by Conserved Region Identification and Association- are presented in the next section. The results of the comparative performance evaluation experiments as well as the application of the method to 17793 human proteins is provided in Results Section, followed by the discussion of the results along with the significance of the method in Discussion Section.

4.1. Methods

The flow diagram of Protein Function Assignment by Conserved Region Identification and Association is given in Figure 51. We describe each step in detail below.

4.1.1. Pairwise Sequence Alignment

A stand-alone version of SSEARCH algorithm from FASTA v36.3.5 software package (Pearson and Lipman, 1988) was used for the Smith-Waterman pairwise all-against-all sequence alignment (Smith and Waterman, 1981) with the default options. The BLAST algorithm (Altschul *et al.*, 1997) could also be used at this step to reduce the computation time should need be. After that, a square matrix was formed using the pairwise alignment e-values and threshold with the default value of 0.01. Hence, the pairwise alignment e-values that exceeded 0.01 were removed from the matrix. The default threshold value was selected after many trial tests with reference datasets (data not shown), though it is possible to set a different threshold value to suite a particular sequence dataset.

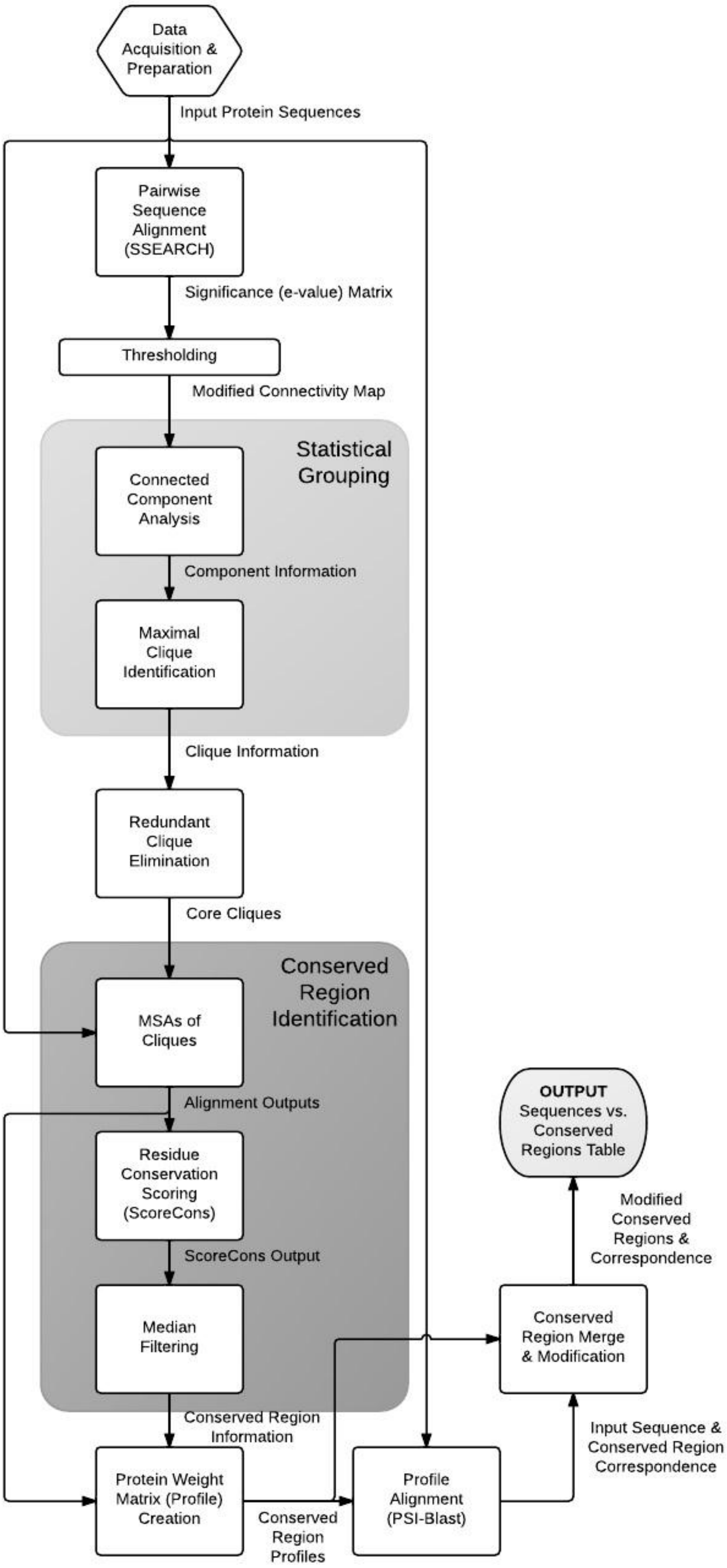

Data Acquisition & Preparation
Input Protein Sequences
Pairwise Sequence Alignment (SSEARCH)
Significance (e-value) Matrix
Thresholding
Modified Connectivity Map
Statistical Grouping
Connected Component Analysis
Component Information
Maximal Clique Identification
Clique Information
Redundant Clique Elimination
Core Cliques
Conserved Region Identification
MSAs of Cliques
Alignment Outputs
Residue Conservation Scoring (ScoreCons)
ScoreCons Output
Median Filtering
Conserved Region Information
Protein Weight Matrix (Profile) Creation
Conserved Region Profiles
Profile Alignment (PSI-Blast)
Input Sequence & Conserved Region Correspondence
Conserved Region Merge & Modification
Modified Conserved Regions & Correspondence
OUTPUT Sequences vs. Conserved Regions Table

Figure 51. Flow diagram of Protein Function Assignment by Conserved Region Identification and Association Method.

4.1.2. Statistical Grouping

Routines provided by the MATLAB® Bioinformatics Toolbox (The MathWorks Inc., 2010) were used for the Connected Component Analysis. The input sequences were grouped into components possessing a direct or an indirect connection between every sequence pair. This guaranteed that two sequences in different connected components not to have a significant similarity.

Next, a maximal clique identification procedure was applied on each connected component. In order to reduce the computational load exerted by the clique identification process, large connected components composed of more than 100 sequences were divided into random groups of 100 sequences and maximal cliques were found on each of these groups separately, using the Bron–Kerbosch algorithm (Bron and Kerbosch, 1973). This procedure, however, produced several redundant cliques that differed from each other by a few sequences, revolving around an underlying clique missing a few connections in the connectivity map.

In order to detect and eliminate the redundant cliques, Hamming distances (Hamming, 1950) between maximal cliques were computed and divided by the total number of sequences in the corresponding cliques, providing the fractional Hamming distances between all clique pairs in the component.

Hamming distance is a measure of difference between two strings of equal length. It's basically the minimum number of substitutions to change the first string into the second (Hamming, 1950). We define the fractional hamming distance between a pair of cliques as the regular Hamming distance divided by the total number of proteins in both cliques. This normalization eliminated the effects of the contrast between the clique sizes on the distance measure. Calculation of the fractional Hamming distance is given in Equation 4.1.

$$H_{f_{ab}} = \frac{\sum_{i=1}^{n} m_{ab_i}}{n_a + n_b} \tag{4.1}$$

H_{fab} is the fractional Hamming distance between cliques a and b, m_{abi} is a binary variable that represent the match or mismatch at the i^{th} position between cliques a and b (0 if there is a match and 1 if there is a mismatch), n is the total number of proteins in the test, n_1 and n_2 are the total number of proteins in the corresponding cliques.

The cliques were then clustered using a pre-defined fractional Hamming distance threshold of 0.3 and the redundant cliques were eliminated by selecting the clique with the highest number of sequences to represent each group. The default threshold was set according to our tests in which this value almost never eliminated a non-redundant clique, but separated many of the redundant ones. The remaining redundancy was removed using an additional procedure explained at Section 4.1.4.

4.1.3. Conserved Region Identification & Search Process

First, member proteins of each maximal clique were subjected to global multiple sequence alignment individually using the ClustalO package (Sievers *et al.*, 2011) with default parameters. Then, a residue conservation scoring algorithm was employed to review the multiple alignment of each maximal clique. Many different residue conservation scoring methods exist in the literature. These methods are designed to scan the multiple alignments column-wise and reveal the conservation degree of each position in terms of the stereochemical diversity, diversity of symbols based on theoretical entropy, and/or amino acid frequency (Valdar, 2002). In this work, one of the most conventional, the valdar01 scoring method (Valdar, 2002) was used, where a substitution matrix is employed to evaluate the stereochemical diversity. Consequently, each position was scored between 0 (no conservation) and 1 (full conservation). A local version of ScoreCons algorithm (Valdar, 2002) was used with default parameters to carry out the procedure.

Since the residue conservation scoring algorithm acts on each position independently, the output is inevitably noisy. In order to clearly identify the conserved regions, we have used the one-dimensional Median Filtering method (Boyle and Thomas, 1988) with order (or neighborhood size) of 50. This method was shown to preserve the edge regions in the original signal better than most of the linear de-noising/smoothing methods (Boyle and Thomas, 1988), and yield a more accurate detection, especially around the boundaries of the conserved regions. The order of the median filter was set to match the minimum size of the conserved regions: Our method was aimed at detecting the conserved regions that are longer than 20 amino acids, since nearly all of the structural domains registered on the databases fall in this area. The filter takes the median of 50 values around the filtered position; -25 to the left and 25 to the right-, and as a result, regions shorter than 25 amino acids were filtered out. Allowing 1 or 2 amino acids at both edges to be rounded off by the filter achieved the detection of conserved regions of length 20 amino acids and higher.

Another key component here is the selection of the threshold to accept the positions with an exceeding score as conserved positions. As discussed above, an uninterrupted series of conserved positions with length greater than or equal to 20 amino acids were labeled as a conserved region. Thus, the threshold score should strike a critical balance to identify only the true conserved residues without missing any.

In order to determine the threshold conservation score, reference multiple sequence alignments of different eukaryotic proteins that were employed for building NCBI-curated domain profiles were downloaded from NCBI CDD web site (Marchler-Bauer *et al.*, 2011). These regions were labeled as the locations of the domains on these alignments were known. Residue conservation scoring algorithm was applied upon them and the output was smoothed. After that, domain regions were extracted using different threshold score selections.

Histograms consisting of the residue conservation scores of domain and non-domain regions on original scores -(a) and (b)- and on smoothed/filtered scores -(c) and (d)- are drawn in Figure 52. As obvious from this figure, higher scores are heavily accumulated in domain regions. There were some low scored residues in domain regions in the original score curves but after the filtering operation the number of the low scored positions in domain regions were decreased by nearly 30%. To determine the threshold conservation score, a receiver operating characteristic (ROC) curve (Lasko *et al.*, 2005) was drawn using reference labels of all positions and calculating true positives rate (sensitivity) and false positives rate (fall-out) at different threshold selections. This curve is shown in Figure 53. The optimum point for the threshold at the knee formation is marked as a black dot on the ROC curve. The threshold score value equals to 0.2 at this point. The positions that have a conservation score over 0.2 were assumed to be the conserved positions and a region formed by an uninterrupted series of conserved positions with a size of at least 20 amino acids was accepted as a conserved/shared region. A suitable threshold was found at a level of 0.2 at the end of the threshold determination procedure.

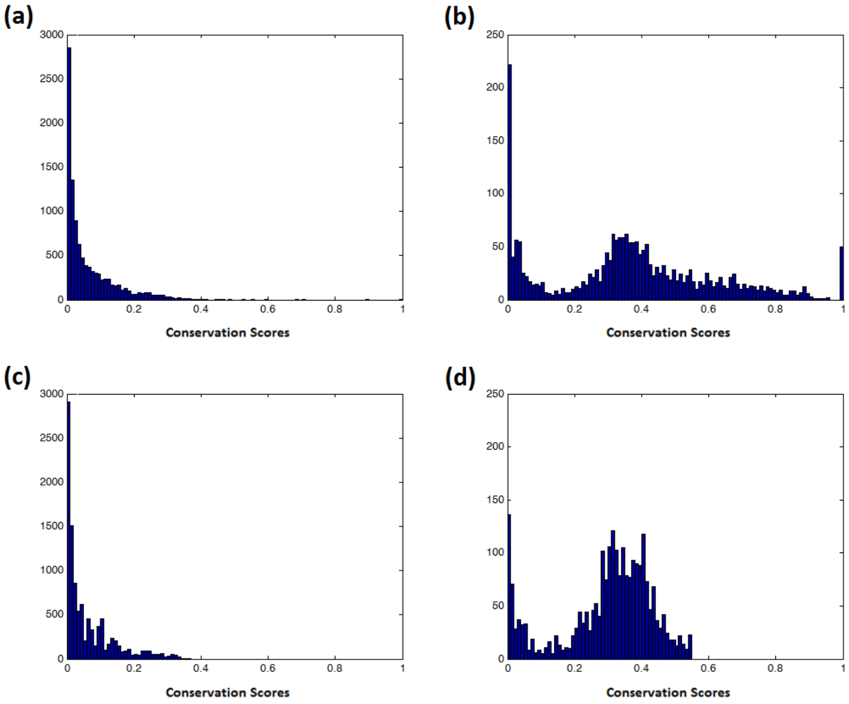


Figure 52. Residue conservation scoring histograms of curated multiple sequence alignments of different eukaryotic proteins, the original outputs: (a) the residues outside NCBI-curated functional domains, (b) the residues inside NCBI-curated functional domains. (c) and (d): the same histograms respectively after the smoothing operation.

Profiles consisting of the frequency of amino acids as well as the gaps were created for all conserved/shared regions using the multiple sequence alignment and the conserved region identification results. These profiles were then aligned to all sequences in the dataset using a local version of Position Specific Iterative Blast (PSI-blast) algorithm (Altschul *et al.*, 1997) using the default parameters. PSI-blast takes a query sequence, searches through a database, forms a profile (a PSSM) with the query and the significant hits, and searches the database again, this time querying the profile to include more remote hits. This procedure then repeats iteratively until convergence. As a result remote homologs are retrieved that might be missed with a normal blast search (Altschul *et al.*, 1997). The queries in our case are the previously generated conserved region profiles. We here carried out the PSI-blast using the "querying an intermediate PSSM" option of the algorithm. In order to include only highly significant hits, we have used a threshold of 10^{-5} over the e-values and only 1 iteration of the algorithm.

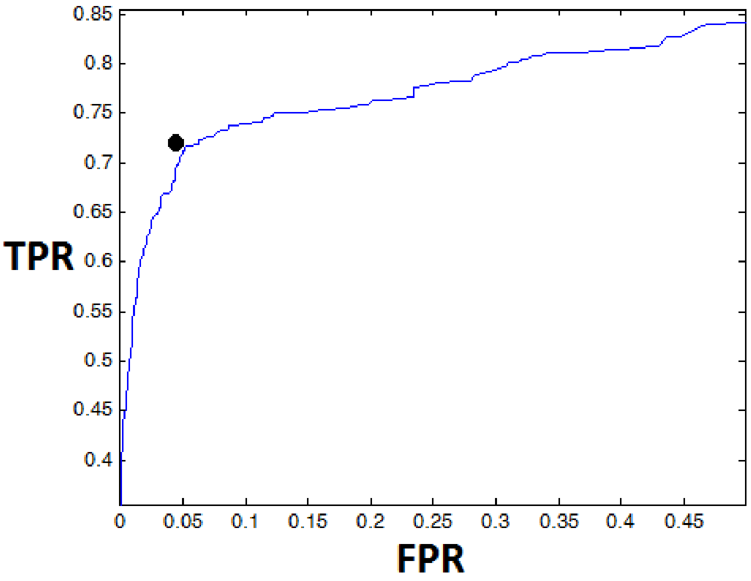


Figure 53. ROC curve for the binary classification of residues of reference multiple sequence alignments as domain or non-domain regions for the determination of threshold score in residue conservation scoring process. Black dot indicates the TPR and FPR values at the selected threshold.

Figure 54 shows the complete conserved region identification process. On the top, multiple sequence alignment output of the members of a sample clique found at the application of our method on human protein dataset. Each row represents a different sequence. Red regions represent a shared functional domain on these proteins given by both NCBI CDD and Pfam searches, black regions are the remaining filled positions and the gray ones are the gaps. The middle plot shows the residue conservation scoring output of the same alignment. Notice the elevated conservation scores correspond to the functional domain region. To eliminate the noise, median filtering was applied and the output and shown at the plot below. The positions with scores higher than the conservation threshold (0.2) form the conserved region. Also notice nearly perfect correspondence between our conserved region and the reference functional domain.

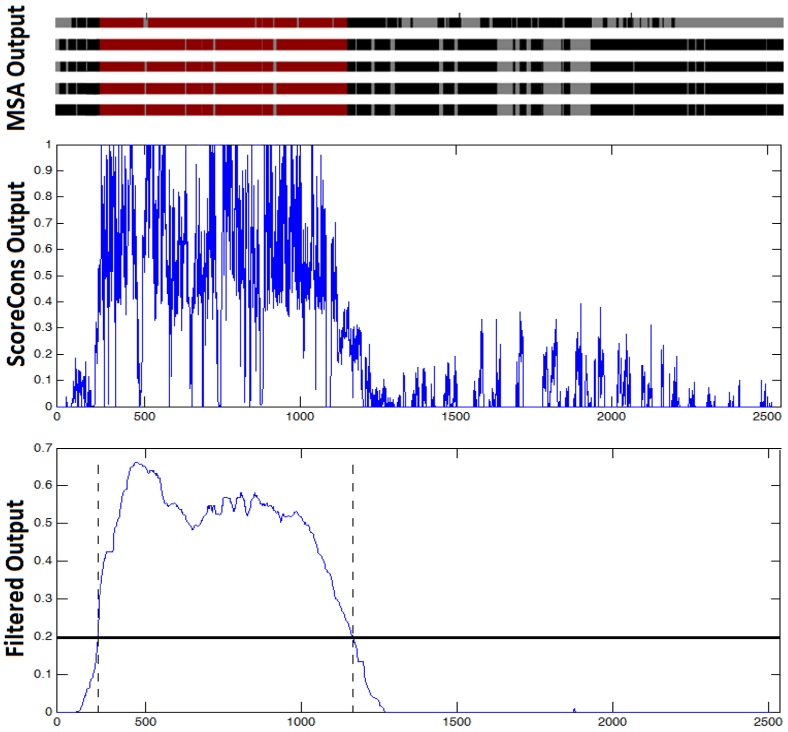


Figure 54. Representation of the conserved region identification procedure. Top: Multiple sequence alignment output of members of a sample clique, each row represents a sequence, colors - black: filled positions, red: domain regions, gray: gaps. Middle: Residue conservation scoring process (ScoreCons) output (applied on the same alignment). Down: Smoothed output with median filtering, horizontal black line: threshold score to assume conservation, vertical dashed line: signifies the borders of the recovered conserved region (MSA: Multiple sequence alignment).

4.1.4. Conserved Region Merge and Modification Step

This operation was applied on the collection of conserved regions identified above to remove the redundant conserved regions coming from the redundant maximal cliques. To this end, non-gapped consensus sequences of conserved region profiles were generated and aligned to each other in an all-against-all manner using a Smith-Waterman pairwise local alignment procedure (Smith and Waterman, 1981) provided by the SSEARCH algorithm from FASTA v36.3.5 software package (Pearson and Lipman, 1988) with the default options. The e-value threshold was set to 0.01. Among the significantly aligned regions, the ones with a higher number of associated sequences were then selected and the rest were removed from the results.

Finally, a table was generated that represented the identified conserved regions along its columns and the input sequences along its rows and, with zeros and ones filling the table indicating the associations of the conserved regions with input sequences. This table and the profiles of the identified conserved regions constituted the main outputs of the method. Table 4 shows a sample output association table, the associated conserved region and sequences are highlighted with blue color on their corresponding cell.

Table 4. A sample conserved region vs. input sequences association table (output).

	CR 1	**CR 2**	**CR 3**	**CR 4**	**CR 5**	**CR 6**
Sequence 1	1	0	0	0	0	0
Sequence 2	1	0	0	0	0	0
Sequence 3	1	0	0	0	0	0
Sequence 4	1	0	1	0	1	1
Sequence 5	1	0	1	0	1	1
Sequence 6	0	1	0	0	0	0
Sequence 7	0	1	0	0	0	0
Sequence 8	0	1	0	0	0	0
Sequence 9	0	1	0	0	0	0
Sequence 10	1	1	1	0	0	0
Sequence 11	1	1	1	0	0	0
Sequence 12	0	0	1	0	0	0
Sequence 13	0	0	1	0	0	0
Sequence 14	0	0	1	1	1	0
Sequence 15	0	0	1	1	1	0

4.1.5. Optional Final Clustering Procedure

We offer an optional fast clustering process at the end of the method with Connected Component Analysis using conserved region correspondence information as the input similarity matrix. Conserved region correspondences of sample proteins were compared with each other to obtain pairwise similarities (between 0 and 1) in a manner similar to fractional Hamming distances. Calculation of this similarity measure is given in Equation 4.2. Pairwise similarities regarding conserved region correspondences are then threshold with a pre-defined value. This value was selected as 0.3 at the parameter determination analyses. Conserved region correspondence similarities lower than 0.3 were labeled and the associated connectivities were removed from the connectivity map. After that, the threshold connectivity map was given to Connected Component Analysis procedure as the input.

$$S_{f_{xy}} = \frac{2 * \sum_{i=1}^{n} R_{xy_i}}{\sum_{i=1}^{n} K_{xy_i} + 2 * \sum_{i=1}^{n} R_{xy_i}} \tag{4.2}$$

S_{fxy} is the fractional conserved region correspondence similarity measure between sequences *x* and *y*, R_{xyi} is a binary variable that represent the match or mismatch at the i^{th} conserved region between sequences *x* and *y* (1 if there is a match and 0 if there is a mismatch), K_{xyi} is also a binary variable (a contrast of *R* variable) that represent the match or mismatch at the i^{th} conserved region between sequences *x* and *y* (0 if there is a match and 1 if there is a mismatch) and n is the total number of conserved regions.

This process gives highly accurate clustering results during the test runs when the input contains single domain proteins (or just domain sequences). On the other hand, conserved region vs. input sequence correspondence table provides a sufficient source for the associations of multi-domain proteins.

4.1.6. Calculation of the Statistical Parameters

F-score measures performance by incorporating both precision and recall (sensitivity), and displays it in a single number (Paccanaro *et al.*, 2006; Nepusz *et al.*, 2010). Since the cluster corresponding to a gold standard superfamily was not known, precision and recall were calculated for all cluster and superfamily combinations. Then the combinations that maximize the combined F-score were selected (without conflictions). The calculations of precision, recall (sensitivity) and the combined F-score are shown in equations 4.3, 4.4 and 4.5 respectively.

$$P_{ij} = \frac{TP_{ij}}{TP_{ij} + FP_{ij}} \tag{4.3}$$

$$R_{ij} = \frac{TP_{ij}}{TP_{ij} + FN_{ij}} \tag{4.4}$$

P_{ij} and R_{ij} are the precision and recall values respectively for superfamily i and cluster j. TP_{ij} is the number of proteins both present in superfamily i and cluster j. FP_{ij} is the number of proteins present in cluster j but not in superfamily i. FN_{ij} is the number of proteins present in superfamily i but not in cluster j.

$$F = \sum_{i} n_i \max_{j} \frac{2P_{ij}R_{ij}}{P_{ij} + R_{ij}} \tag{4.5}$$

F is the combined F-score, i represents superfamilies and j represents clusters, n_i is the number of proteins in superfamily i, P_{ij} and R_{ij} are the precision and recall values respectively (explained above).

4.1.7. Performance Test for the Proposed Method in the Identification of Reference Domain Hits in Human Protein Dataset

At the stage of determining the reference domain hits in the test sequences, a total of 674 sequences in Pfam-A test and 171 sequences in NCBI CDD test were found to contain more than 6 significant domain hits. Due to the high number of hits these sequences were accepted as outliers and removed from the performance tests. The rest of the analyses were carried out for the proteins with 6 domain hits or less on different regions of the sequences -not counting multi-hits on a particular region-.

Performance test steps for our method are given in Figure 55. In order to, to generate the reference domain association set, first, our dataset was first queried in batch-CDD search procedure in NCBI CDD web site with default parameters using NCBI curated domain profiles as the database and an e-value cut off of 0.05. Second, a standalone version of HMMER v3.0 algorithm (Finn *et al.*, 2011) was used for querying the dataset through Pfam-A profile Hidden Markov Model database with the default options. This way, we have discovered the confirmed functional domains on the test sequences separately for Pfam-A and NCBI curated domain databases.

Next, profile alignments between the conserved regions against NCBI CDD domains and the conserved regions against Pfam domains were carried out. Consensus sequences of our conserved region profiles were generated in order to search against the pre-formatted functional domain database of NCBI CDD and profile HMM's of Pfam. Each conserved region profile consensus sequence and domain profile were aligned to each other and significant matches were sought using local Reverse Position Specific Blast (Rps-blast) algorithm (Marchler-Bauer, 2002) for NCBI curated domains and HMMER v3.0 for Pfam-A domains with default parameters in both cases. Rps-blast is a blast type algorithm used to search a query sequence against a database of profiles in order to discover significant matches (Marchler-Bauer, 2002). A significant alignment between a conserved region profile and a reference domain indicated a high chance that these two represented the same functional domain. In some cases there were more than one significant hit. In these cases, the most significant hit -the lowest E-value- was accepted as the pair of the corresponding conserved region. In some other cases there were no significant hits to the query profile, these regions were not paired with any reference domains.

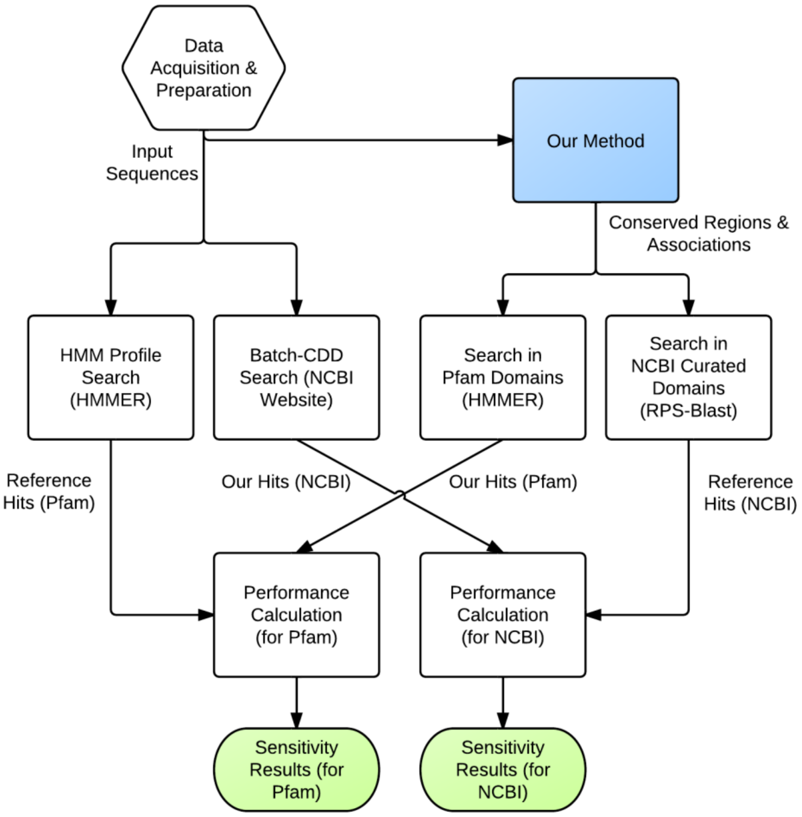

Data Acquisition & Preparation
Our Method
Input Sequences
Conserved Regions & Associations
HMM Profile Search (HMMER)
Batch-CDD Search (NCBI Website)
Search in Pfam Domains (HMMER)
Search in NCBI Curated Domains (RPS-Blast)
Reference Hits (Pfam)
Our Hits (NCBI)
Our Hits (Pfam)
Reference Hits (NCBI)
Performance Calculation (for Pfam)
Performance Calculation (for NCBI)
Sensitivity Results (for Pfam)
Sensitivity Results (for NCBI)

Figure 55. Flow diagram of the performance test for our method in identifying reference domains in human proteins.

Finally, performances were measured using sensitivity values. The domain hits to the input proteins found by our method (through conserved region to domain matching) were compared to the reference domain hits (found in the first step). True positive (TP) and false negative (FN) values were calculated. A true positive hit was obtained when the same domain was found both by our method and the reference domain search on a protein. When our method failed to find a domain present in reference search, this was counted as a false negative. This procedure was repeated for all proteins in the dataset and taken to the average to display a global performance of our method in the identification of functional domains in human proteins.

4.2. Results

First, to discover our methods success in functionally separating amino acid sequences, reference datasets from SCOP Domain Database (Andreeva *et al.*, 2008) were clustered and the performance was measured and compared with the conventional methods.

Second, the method was applied on the previously mentioned large human protein sequence dataset to obtain a global functional relation map. The accuracy of the identified relations was evaluated with respect to the domain assignments in Pfam (Finn *et al.*, 2010 and Punta *et al.*, 2012) and NCBI CDD (Marchler-Bauer *et al.*, 2011) databases as reference.

4.2.1. Clustering with Reference Datasets

We tested the performance of our method in clustering amino acid sequences using gold standard reference datasets used in previous studies in the literature. Five different datasets from SCOP 1.75 Database (Andreeva *et al.*, 2008) previously analyzed by (Nepusz *et al.*, 2010) and (Paccanaro *et al.*, 2006) to test their widely used method Spectral Clustering were taken exactly as they appeared in the referenced studies. Four of these datasets were generated by manually curating domain sequences from different superfamilies in the SCOP 1.75 Database and composed of 550 to 670 sequences each, located in 5 to 6 superfamilies. The fourth dataset was composed of the members from 8 superfamilies and was regarded as a more difficult case for clustering algorithms (Nepusz *et al.*, 2010). The fifth dataset was composed of all domain sequences in SCOP 1.75 Database refined further by removing the sequences with pairwise identity values greater than 95% (ASTRAL-95) via ASTRAL Database (Chandonia *et al.*, 2004), and by removing the members of the superfamilies with less than five domains (Nepusz *et al.*, 2010). This final dataset called SCOP≥5 contained 14309 sequences from 632 superfamilies and represented one of the most difficult cases for sequence clustering methods (Nepusz *et al.*, 2010). We have applied our method to these datasets with the default parameters without any specific parameter tuning. Our method's optional final clustering was obtained by incorporating a fast clustering process at the end by a Connected Component Analysis using the correspondence between the conserved regions and the sequences as the input similarity matrix.

Rules about the comparisons of the results with the gold standard to assure the fair assessment of the methods were used as given in (Nepusz *et al.*, 2010). Clustering performance was calculated via the combined F-scores, defined as the combination of precision and recall with equal contributions (Paccanaro *et al.*, 2006). The combined F-score was calculated as in (Nepusz *et al.*, 2010) and shown in detail in Section 4.1.6.

The clustering performances of previous methods given in (Nepusz *et al.*, 2010) are shown in Table 5 with the addition of our method in the last column. The method listed as CCA represents the Connected Component Analysis that is also used as an intermediate step in our method. The others, TribeMCL and Spectral Clustering were described in the Introduction Section.

Table 5. Clustering performance results on gold standard datasets from SCOP Database.

	Number of sequences	**F-scores**			
		CCA	**TribeMCL**	**SCPS**	**Our method**
Dataset 1	669	0.530	0.630	0.844	**0.866**
Dataset 2	587	0.681	0.772	**0.905**	0.884
Dataset 3	567	0.588	0.625	0.893	**0.906**
Dataset 4	654	0.497	0.573	0.685	**0.740**
Dataset 5	14309	0.530	0.576	0.607	**0.641**

CCA: Connected Component Analysis, SCPS: Spectral Clustering.

On the first 3 datasets representing relatively easy clustering instances, our method's performance was comparable to Spectral Clustering, the top performing algorithm from the literature. On the fourth and fifth datasets, our method outperforms all the alternatives, albeit slightly. This demonstrates the effectiveness of our approach based on statistical grouping over detected conserved regions.

To supplement these results, we have carried out an additional test to verify that the increased performance was not due to the usage of Smith-Waterman pairwise alignment in the first step instead of fast Blast algorithm as used in Spectral Clustering, especially on the complex datasets. To this end, Blast pairwise alignment results for datasets 1, 2, 3 and 4 were directly taken from (Nepusz *et al.*, 2010), and its e-values were used as input to our method. The results were similar to those obtained before: F-scores of 0.894, 0.864, 0.904 and 0.724 were achieved for datasets 1, 2, 3 and 4 respectively, indicating that our method's better performance is not due to the use of an optimal pairwise alignment algorithm. In addition, even though these datasets only contained domain sequences from SCOP database, our method extracted the most conserved core regions (occasionally the whole domain sequences were identified as conserved regions). As a result, remote sequences were clustered more accurately, owing to the correspondence between conserved regions and the input samples.

4.2.2. Functional Mapping of Human Proteins and Automatic Domain Identification

Next, we have applied our method to the previously analyzed large human protein sequence data. The details about the preparation and specifications of the dataset were given in Section 3.2.1.

Following the initial pairwise local alignment and connectivity map thresholding, we have identified 3592 connected components of varying sizes along with 2442 singleton components. Within these, 6537 maximal cliques were identified following the elimination of the redundant cliques. After the remaining intermediate steps, 4753 conserved regions were identified and presented in a table showing the association of each conserved region with all input sequences. Figure 56 shows the histogram of the all-against-all pairwise alignment e-values. The vertical black line represents the threshold e-value. Only the values between 0 and 0.1 are shown on the figure.

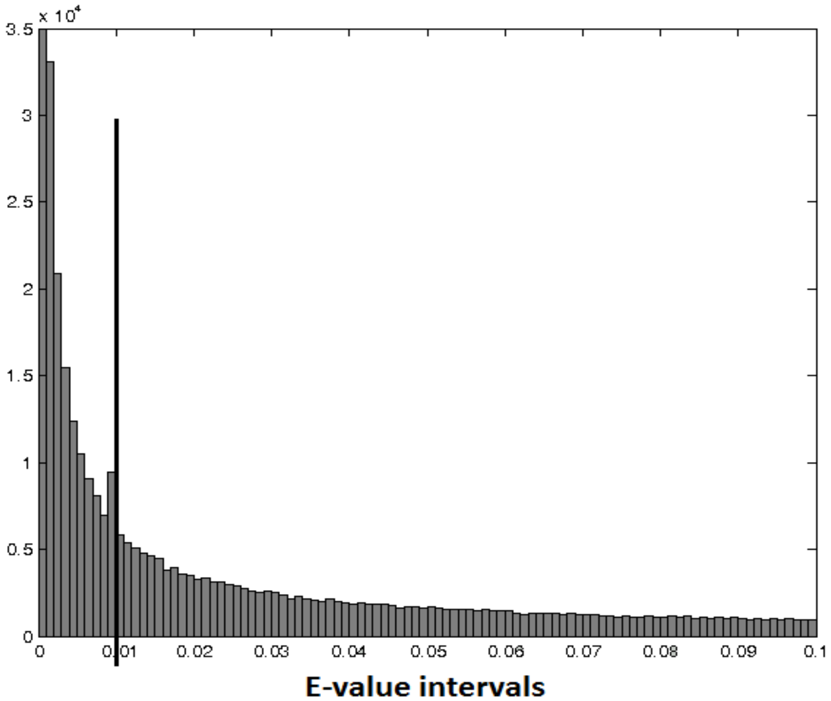


Figure 56. E-value histogram of all-against-all pairwise alignment.

The statistics about the conserved region associations with 17793 human protein sequences at the end of the analysis are given in Table 6. As observed from the table, 3531 sequences got no associations with any conserved regions, in other words no information could be recovered regarding these sequences in our analysis. An inspection over these sequences revealed that most of these were remote sequences, non-homologous to every other sequence in the set.

Table 6. The statistics of the numbers of conserved regions associated with test sequences (human protein dataset).

Number of Conserved Regions:	**Number of Sequences:**
No hits:	3531
1 hit:	4195
2 hits:	2037
3 hits:	1557
>3 hits:	6473

Out of 17793 human protein sequences.

In order to validate the results on human proteins, we have evaluated the correspondence between the conserved regions identified above and the Pfam-A domains in Protein Families Database 26.0 (Finn *et al.*, 2010 and Punta *et al.*, 2012) as well as the NCBI curated (cd) domains in Conserved Domain Database v3.07 (Marchler-Bauer *et al.*, 2011). Specifically, we have first queried all sequences in a domain search on the databases above and obtained reference domain assignments. Second, we paired conserved regions with reference domains by querying the conserved region profiles on these databases and identifying the most significant domain hit for the corresponding conserved region. A conserved region was not paired with any domains when no hits with an e-value lower than 0.01 was obtained from the search. Third, we compared the reference domain assignments on the sequences with the assignments we have recovered through sequence to conserved region and conserved region to domain associations. A more detailed explanation of this performance test is provided in Section 4.1.7.

Figure 57 shows the histograms of domain (all NCBI CDD manually curated domains) and conserved region sizes in (a) and (b) respectively. As observed from the figure, the distributions were similar, although the number domains with sizes between 100 and 700 amino acids were higher than the number of conserved regions recovered during the human protein test at the same interval. On the other hand, the number of conserved regions with lower sizes (between 20 to 100 amino acids) was higher than the domains.

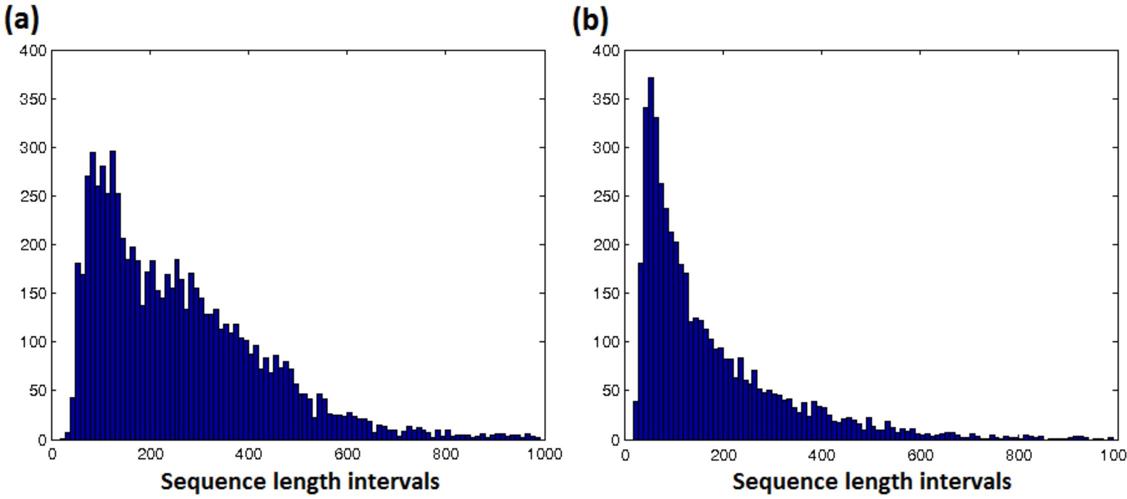


Figure 57. Length Histograms for (a) NCBI CDD curated domains (b) conserved regions recovered after the human protein test.

HMM profile search identified a total of 24197 reference Pfam-A hits on our dataset, whereas Batch-CDD search in NCBI CDD web site identified 16526 reference hits in total. The statistics on the number of reference hits are given in Table 7. Notice the high number of sequences without any domain assignments. Nearly 46% of the test sequences got no curated functional associations in these vast domain databases. Table 8 shows the statistics about domain assignments of the human protein dataset at the end of our analysis through the associations of conserved regions and NCBI CDD curated and Pfam-A domains (as explained in Section 4.1.7 and Figure 55). The number of sequences with elevated number of hits was increased with our analysis (the last 2 rows on Table 7 and Table 8).

Table 7. The statistics of reference domain hits on human protein sequences.

Number of domains:	**Number of sequences**	
	Pfam-A	**NCBI curated**
No hits:	7975	8281
1 domain:	5348	6417
2 domains:	2247	1768
3 domains:	838	628
>3 domains:	1385	699

Out of 17793 human protein sequences.

Table 8. The statistics of the domain hits on human protein sequences by our analysis through associations between conserved regions and the domains in reference databases.

Number of domains:	**Number of sequences**	
	through Pfam-A	**through NCBI**
No hits:	8322	8394
1 domain:	3720	3222
2 domains:	1476	1577
3 domains:	1184	1406
>3 domains:	3091	3194

Out of 17793 human protein sequences.

Another calculated statistical measure is the number of domain assignments of the proteins with no reference domain assignments in the human protein dataset (the number of these proteins are shown in the first row in Table 7) at the end of our analysis through the associations of conserved regions and NCBI CDD curated and Pfam-A domains (Table 9). The first and the second columns show the number of domain assignments (through conserved region to Pfam-A domain associations) and the number of direct conserved region hits respectively on the sequences with zero reference domain associations on Pfam-A database. At the third and the fourth columns of Table 9 the same statistics are given for the sequences with zero reference domain associations on NCBI CDD curated domain database. As observed from Table 9 columns 1 and 3, the number of proteins with no domain hits remained almost the same with respect to the reference hits (Table 7) regarding both databases. There were many conserved region associations to these proteins (Table 9, columns 2 and 4); however, these could not be paired with any reference domains since these regions were derived from sequences with no domain assignments.

Table 9. The statistics of the domain assignments and conserved regions associations by the proposed method on the proteins with zero reference domain assignments.

Number of hits:	**Number of sequences with 0 reference domains in:**			
	Pfam-A		**NCBI CDD**	
	through ref. dtb.	**direct CR hits**	**through ref. dtb.**	**direct CR hits**
No hits:	6877	2408	7156	2582
1 hit:	737	2145	742	2267
2 hits:	128	878	193	952
3 hits:	118	484	83	506
>3 hits:	115	2060	107	1974

Out of 7975 sequences for Pfam-A and 8281 sequences for NCBI CDD columns.

Table 10 shows the performance measures as sensitivity values in identifying reference functional domains in human protein sequences. The figure was structured with respect to the number of curated domains on each test sequence regarding the reference databases. Sensitivity (recall) value of 0.744 regarding Pfam and 0.776 for NCBI CDD may be considered quite satisfactory. Nearly 76% of the domains on the reference databases were accurately recovered by our method, with more than 77% and 61% of the reference domains on Pfam and NCBI CDD respectively being from multi-domain proteins. Notice the highest performance with the single domain proteins, and a small but gradually decreasing performance with the increasing number of domains on sequences.

Table 10. Performance of the proposed method in identifying reference functional domains in the sequences of human protein dataset.

Number of domains	Sensitivity values	
in reference database:	Pfam ref.	NCBI CDD ref.
Single domain:	0.809	0.919
Up to 2 domains:	0.779	0.883
Up to 3 domains:	0.765	0.862
Total:	0.744	0.776

(Sensitivity: TP / (TP + FN), TP: true positives, FN: false negatives)

A certain percentage of the recovered conserved regions (on average) were significantly aligned (and paired) with the reference curated structural domains and the performance of the proposed method in identifying the domains on test sequences were calculated regarding these conserved regions. However, a significant portion of the conserved regions could not be aligned with the documented domains on online databases. Table 11 shows the statistics about the number of conserved regions significantly aligned (and paired) with reference curated domains on Pfam-A and NCBI CDD curated domain databases. The first column shows the number for the paired conserved regions, whereas the second column shows the ones that could not be paired (the original conserved regions), out of the 4753 recovered regions. 51% and 41% of the conserved regions were original (did not have a correspondence on the reference domain databases) regarding Pfam-A and NCBI CDD curated domain databases respectively. At least some of these original conserved regions may correspond to new functional domains un-identified so far. Though it’s not possible to verify any of these without detailed studies focusing on each sequence (also with experimental results usually). Table 12 shows the number of input sequences containing at least one original conserved region hit and the total number of the original conserved region hits on these protein sequences. These are the sequences containing potential new domain assignments by the proposed method.

Table 11. The statistics of the conserved region pairings with the reference domains.

	Number of conserved regions:	
	match	no-match
Pfam-A	2324	2429
NCBI CDD	2795	1958

Out of 4753 conserved regions.

Table 12. The number of proteins with potential new domain assignments and the total number of new conserved region hits on these proteins.

	Number of proteins:	**Total number of hits:**
Pfam-A	7381	34678
NCBI CDD	6134	30141

With 2429 different conserved regions for Pfam-A and 1958 regions for NCBI CDD databases.

In order to observe if these original conserved regions correspond to the automatically generated (not manually curated) low significance domain entries in Pfam database, we queried the conserved region profiles against a database containing both Pfam-A and Pfam-B entries (we have done a similar operation using only the manually curated Pfam-A domains previously). Pfam-B entries were generated to supplement the Pfam database for the sequences where there are no Pfam-A associations (Finn *et al.*, 2010). Pfam-B was generated automatically using the ADDA database (Heger and Holm, 2003). Table 13 shows the information about the conserved region and reference domain pairings (regarding significant pairwise alignments) using only Pfam-A (the previous analysis) and both Pfam-A and Pfam-B databases. As observed from the table, only 27% of the original conserved regions correspond to Pfam-B domains, meaning most of these conserved regions (nearly 73%) were indeed original.

Table 13. The statistics of the conserved region and reference domain pairings with different Pfam database types.

	Number of conserved regions:	
	match	**no-match**
Pfam-A	2324	2429
Pfam-A & B	2986	1767

Out of 4753 conserved regions.

4.3. Discussion

In this part of the study, we proposed a computational method to identify functional relations between protein sequences in large and diverse datasets over evolutionary conserved regions. The experimental results showed that these conserved regions highly correspond to the structural domains. Identification of these regions was achieved in a completely unsupervised manner using only sequence data subjected to sequence alignment, residue conservation scoring and graph theory concepts. First, the method was applied on gold standard datasets and the functional clustering performance was measured and compared with the conventional methods. The results indicated highly accurate clustering. Second, we used the proposed method to process a large dataset composed of 17793 human protein sequences to obtain a global functional relation map. At the end, we obtained a table representing the correspondence of the test proteins with the recovered conserved key regions. Functional relations of the proteins are clearly observed through the connections over these regions. We also measured the correspondence of our conserved regions with the manually curated functional domain assignments of the test proteins (on Pfam and NCBI CDD databases). The results showed that most of the structural domains were identified even on multi-domain human proteins.

As evidenced in the experiment results, the proposed method achieved a high performance in clustering gold standard datasets and in the automatic identification of the documented functional signatures (domain regions) on the human protein dataset. The relationships of the input sequences are reflected clearly on the output table showing the associations between the input sequences and the conserved regions. The user may perform additional procedures on this table such as clustering or the observation of the mutual aspects of a specific sequence with the others. Consequently, evolutionary and/or functional shared features can be extracted from large biological sequence datasets containing highly diverge sequences.

As is well known, grouping amino acid sequences using a linkage method such as a Connected Component Analysis imposes a domain chaining problem: A given sequence pair within a component may not necessarily share a significant sequential similarity, but appear in the same component due to the chain effect where they may both possess similarity to a third sequence over different regions (Mohseni-Zadeh *et al.*, 2004; Joseph and Durand, 2009). As a result, being in the same component does not stipulate a shared feature between all sequences in the component, though appearing in different components guarantees the absence of any significant shared features. All shared features however, are to be discovered within each component. In this work, the detection of the fully connected sub-components (maximal cliques) was employed to discover this mutuality. All sequences residing in the same maximal clique were thus guaranteed to share at least one significant sequential regional similarity, on top of any additional features shared between a smaller number of sequences in the clique.

At this point, component formation procedure may appear to be dispensable since maximal clique search discovers the feature sharing information but it is important to note that maximal clique finding is an NP-hard (non-deterministic polynomial-time hard) problem and it may take days to process a relatively small dataset of 500 sequences with an average computational power. Pre-processing with connected component identification ensures the separation of sequence clusters with no inter cluster relationships. In our method, to further reduce the computational load associated with clique identification in large components, groups of 100 random sequences (in the corresponding component) were subjected to maximal clique finding separately. While this imposed an additional amount of redundancy in the identified maximal cliques, it was resolved later at the conserved region merge and modification step.

Highly conserved regions often correspond to zones with functional signatures on amino acid sequences. Thus, the conserved regions found by our method should capture the known domains in the sequences. In the results on human proteins, conserved regions did indeed contained functional signatures, with high correspondence with the average sensitivity of 0.76 -including the multi-domain proteins- (Table 10). It’s highly probable that the performance gap between the analysis with NCBI CDD and Pfam databases was based on the differences in the domain assignments.

Note that a positive predictive rate (precision) was not calculated here since it could be deceptive. Normally, a false positive (FP) hit should be counted when we obtain a hit that doesn’t exist in the reference set. In our case these hits were ambiguous. As shown in Table 11, nearly half of the recovered conserved regions (as the result of the human protein set test) did not correspond to the domains on online databases (these were named as the original conserved regions). And as mentioned previously, the reliable Pfam-A and NCBI curated domains were accepted as our reference, thus we suggested that some of the conserved region hits might correspond to the documented low significance hits (and some might be short random fragments found only by chance). After a second conserved region to structural domain pairing this time using both Pfam-A and Pfam-B domains as the database, it was observed that only 27% of the original conserved regions corresponded to the domains in the low quality Pfam-B database (Table 13). As a result, the rest of the original conserved regions (or at least some of them) may be potential new functional signatures that have not been discovered and/or documented so far. The only way to verify this may be detailed studies directed to each sequence individually (also including experimental work for some cases).

Comparative revaluation results showed that the proposed method performed better with single domain proteins, and the performance decreases as the proteins with higher number of domains included. Inspection of the results revealed that most of the false negative hits belonged to the consecutively located domains on multi-domain proteins. However, when the variety of domain distribution on sequences was sufficiently large, these domains were identified accurately. For an explanation of this statement, supposing there are two consecutive domains on a *Sequence X* called *domain 1* and *domain 2*. If there are some sequences in the dataset that contain *domain 1* but not *domain 2* and similarly some sequences that contain *domain 2* but not *domain 1*, it is highly probable that our method will detect a maximal clique consist of the sequences with *domain 1* and another clique consist of the sequences with *domain 2*. The profiles for *domain 1* and *domain 2* will be recovered using these cliques. When the domain profiles are searched through all of the sequences in the dataset, *Sequence X* most probably will give significant hits for both *domain 1* and *domain 2* profiles and therefore two consecutive domains will be identified on the sequence. A sample case from our tests is shown in Figure 58 and Figure 59. In Figure 58 15 amino acid sequences are shown (rows) with the structural domains within highlighted in different colors (*domain 1*, *domain 2* and *domain 3*) and *Sequence X* (a multi-domain protein with 3 consecutively located domains) at the last row. In this set, apart from *Sequence X*, 13 sequences are single domain proteins and 2 were multi-domain proteins (with 2 domains on each). Figure 59 shows the clusters of sequences after the statistical grouping process. From the multiple sequence alignment of the members of *Clique 1*, a conserved region (inside red circle in the figure) was identified. Similarly, 2 more conserved regions were identified; one from *Clique 2* (inside blue circle) and one from *Clique 3* (inside green circle). Notice that *Sequence X* was found in all cliques and during the conserved region search process, all of these conserved regions were identified on *Sequence X*. Therefore, 3 conserved regions were mapped to *Sequence X* with nearly 100% overlap with the 3 domains present in this sequence. Note that our method aims to divide sequence datasets in small groups of just one shared feature instead of trying to extract several features within a large group. As a result, it can be expected to perform better when a high variety of domain combinations are present on the sequences in the dataset. This can be achieved during the analysis of large datasets consisting of both similar and diverged sequences. This makes our method a suitable candidate to analyze shared features on whole proteomes.

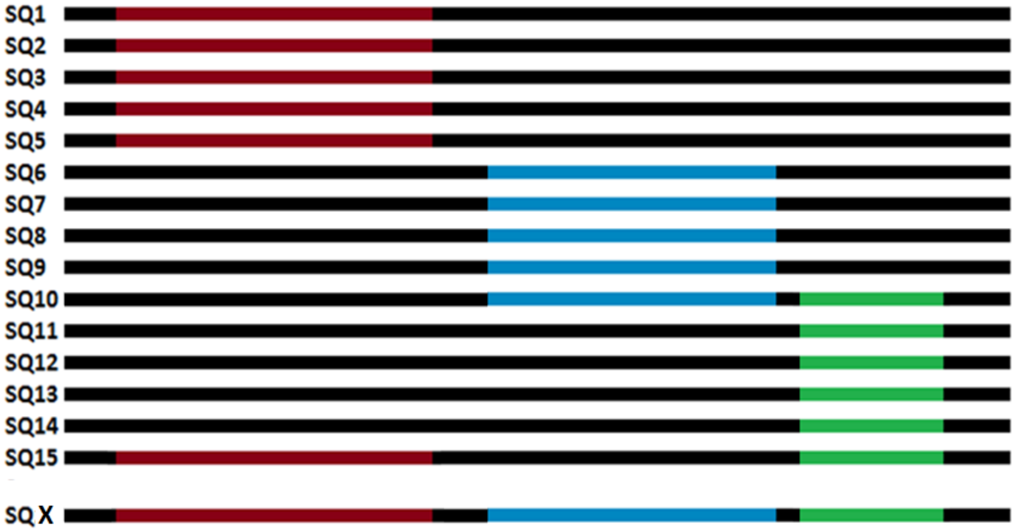


Figure 58. Representation of 16 sample protein sequences with their domains highlighted in different colors.

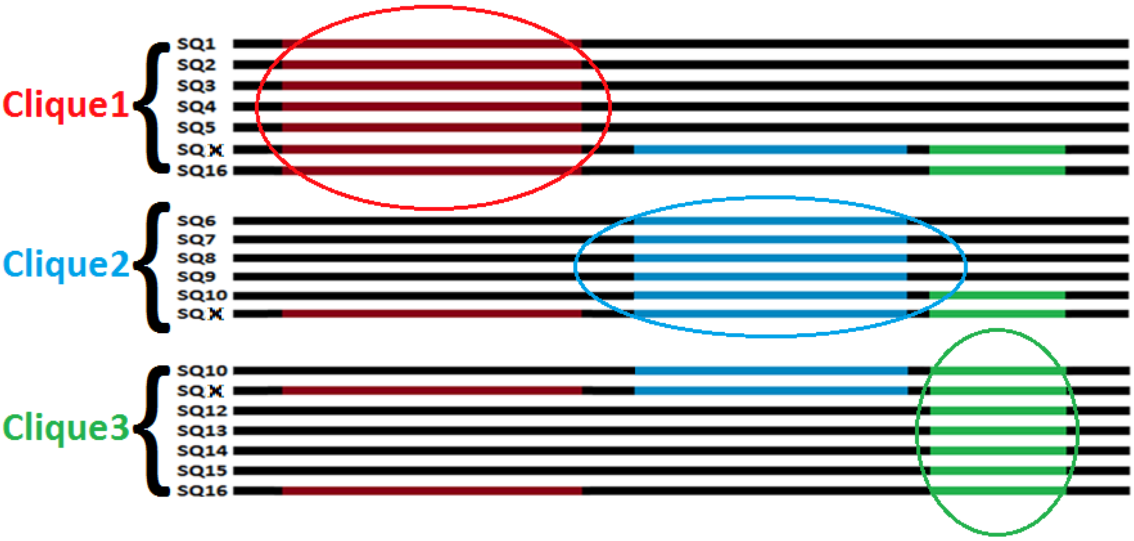


Figure 59. Representation of the sample protein sequences in different cliques after the statistical grouping process (circles correspond to the identified conserved regions).

On another note, there is nothing much to do about the input sequences that do not align with any other sequences in similarity based sequence analysis methods. Generally, they are left out of the results. In our case, we have incorporated the singleton sequences into the analysis by searching in conserved region profiles through all of the dataset. Owing to the remote homology recognition ability of profile alignments, features hidden inside these sequences were identified more clearly. This way, the reference structural domains were identified for some of the singleton sequences we had after the initial all-against-all pairwise alignment.

Finally, we have carried out a manual analysis on the results of the human protein test for possible new functional assignments. Firstly, we noted that the 1009th conserved region largely overlapped with the proteins annotated with the term: GO:0008270 – zinc ion binding where 508 out of 513 proteins associated with the conserved region contained this GO term. We have examined 1 (UniProt identifier: ‘Q14145’ and name: KEAP1 - with evidence at protein level) of the 5 proteins with no relation to GO:0008270 in our cluster. *Kelch-like ECH-associated protein 1* (KEAP1) takes part in the suppression of the transcriptional activity of NFE2L2/NRF2 protein by targeting it for ubiquitination and degradation by the proteasome (Zhang *et al.*, 2004). This protein has only 1 GO molecular function association (GO:0005515 – protein binding) and it has no direct ancestor-child relation to GO:0008270. They only joined at a high level on the hierarchical GO tree on GO:0005488 (binding). As a result, there appears to be no indication that this protein has a zinc ion binding function association in GO. Due to the high correlation between GO:0008270 term and conserved region 1009, our results predict that this protein may have a zinc ion binding function. To test the validity of this prediction, the sequence was searched through Pfam database. 3 types of structural domains were identified on the sequence, one of which is “BTB/POZ domain” (PF00651), found frequently in zinc finger proteins (Bardwell and Treisman, 1994). This finding supports our prediction since zinc ion binding function is naturally associated with zinc finger proteins. In order to take another look at the case, we have analyzed our conserved regions associated with the protein. More than 1 zinc finger domain types are paired with these conserved regions with high confidence during the performance test of our method in identifying reference domains in human dataset. One of them is *Zinc finger, C2H2 type* (PF00096) on Pfam. This domain is also one of the mapped associations to GO:0008270 (in Gene Ontology) though not to GO:0005515, providing an additional support for this prediction.

Note that in this analysis, we started from GO associations and identified a documented functional assignment (on Pfam) to a protein with our method that was not present in the GO database. Similar analyses can be made on the other proteins in the dataset using our results to identify also undocumented functional assignments.

On a final note, the correspondence table generated by our method provides the associations between the proteins and the conserved regions. As such, it allows inferring protein families as these sharing the same set of conserved regions. However, it can also be viewed as documenting the relationship between the conserved regions over the proteins that possess them simultaneously. This suggests a duality in the analysis of protein sequences: Just as the family of proteins associated with similar functional or structural attributes, one can also consider families of conserved regions that followed through the process of molecular evolution together. As a future work, this duality can be explored and exploited to aid a parallel analysis of the evolution of whole proteomes. Another potential future study would be the inspection of the correlation between the protein-protein interactions and the associations of these interacted proteins over the conserved regions. Since these regions correspond to highly conserved sequence segments with functional signatures, the interactions between the proteins may be occurring over these regions. A high correlation may indicate the potential use of the proposed method in protein interaction studies.

The details of the proposed method (Protein function assignment by conserved region identification and association) along with the applications on test sets and the results were prepared as a manuscript and submitted to a peer-reviewed bioinformatics journal. It is also published as a MATLAB® implementation freely available online for academic use together with the datasets and the results figuring in this chapter (including the global functional relation map of 17793 human protein sequences) at http://biplab.eee.iyte.edu.tr/en/projects/conregidase/.

CHAPTER 5

CONCLUSION

In this dissertation, we developed computational methods for automatic identification of sequence and evolutionary relationships in large datasets composed of diverse biomolecular sequences. These relationships usually indicate similar functional properties and/or common evolutionary histories of the sequences in consideration, used to identify unknown sequences.

To this end, we employed sequence alignment and graph theory concepts combined with complementary approaches. Input sequences were treated as vertices in an undirected graph, and the edges between two vertices signified the existence of a distinct similarity between the corresponding sequences. The lengths of the edges were associated with the degree of this similarity, as an edge with a shorter length indicated a closer relationship, such as an elevated homology. After the development of each method, we applied them to a large dataset composed of nearly 18000 human protein sequences in order to obtain a global functional relation map. In this way, the evolutionary and functional relationships between human proteins could be identified on a wider perspective, helping to discover the functions and histories of unknown sequences. The methods employed and for each approach were presented in a separate chapter of this dissertation along with the results obtained from the experiments.

The first approach focused on embedding the sequences in high dimensional spaces as vectors using non-linear embedding. The methods and the results of its applications to different cases were given in detail in Chapter 2. Among the different vector space embedding algorithms in the literature, Isometric Feature Mapping (ISOMAP) was used to express the sequences as vectors in vector spaces with previously unknown number of dimensions. Pairwise evolutionary distances were inferred using an evolution model following a multiple sequence alignment. The method produced a distribution of points that conserved the local neighborhood structure between similar sequences. The suitable number of dimensions was decided upon regarding a residual variance curve, as the smallest dimension that had the ability to express the similarity structure between the sequences accurately. Experiments on synthetic sequences revealed that the method had a significant error reduction capacity on the input pairwise evolutionary distances. Another major observation on the results was the presence of a directionality in the vector space organization: Groups of homolog sequences were placed in tight clusters, with spike-like formations extending outwards from the origin in the final vector space distribution. Homologue groups were positioned at perpendicular angles to each other in the resulting embedding. It was also possible to exploit these formations with statistical methods in order to identify functional separations between the input sequences.

After various optimizations and modifications, we applied the method on a large human protein set. The same functional directionality was also observed, but the necessary number of dimensions was much higher than expected. Furthermore it was observed that the existence of false homology detections, formed during the multiple sequence alignment and the following pairwise distance inference, impaired the non-linear embedding process. In order to eliminate the false homologies, number of gapless positions on pairwise comparisons of the output alignment, the positions available for the pairwise distance computation using an evolutionary model, were calculated and thresholded using various cut-off values. The pairwise distances, computed using a total number of gapless sites smaller than the cut-off were assumed to be unreliable and discarded before the non-linear embedding. Following more tests and modifications, it was concluded that the accuracy of the method dropped significantly when the ancestor sequences were not included in the input sequence collection. The non-linear embedding procedure refined the high distances, the ones with large amounts of error by forming geodesic paths using small and reliable distances, and these paths went through the common ancestors of the sequences in the dataset. Since no information about the actual ancestor sequences is available today, the formation of the geodesic paths and the refinement of the evolutionary distances between remote sequences could not be carried out accurately by the non-linear embedding process. At this point in the study, we moved on to a different approach than the vector space embedding strategy.

The second approach was the thresholding of the connectivity map over 2 parameters following the multiple sequence alignment procedure in order to detect false homologies, and separate the input sequences into relevant clusters of significant similarities. The proposed method and its applications on different cases are explained in detail in Chapter 3. The first parameter captured the inferred pairwise evolutionary distances, and the second one measured the number of gapless sites at each pairwise comparison of the resulted alignment, in other words, number of positions available for pairwise distance calculation. In order to discriminate between the true/reliable similarities and false/unreliable ones, a random dataset was generated by shuffling the elements of each sequence in the original dataset. The random/shuffled dataset was then subjected to the multiple sequence alignment separately and the values of the 2 parameters were calculated as described. Two-dimensional histograms were generated for the original and the shuffled datasets with respect to the two parameters. The portion of the histogram of the original dataset corresponding to the reliable connections as different from the one for the shuffled dataset was identified using probability distribution comparisons. Numerous threshold curves were generated using the standard deviation and mean values of the pairwise distance distributions for different values of the number of gapless positions from the two-dimensional histogram of the shuffled dataset. ROC curves were drawn using the number of connectivities discarded and kept on the original and the shuffled connectivity maps at each threshold. A cut-off was selected automatically at the point where the slope of the ROC curve equaled 10^5, or at the point where there were no connectivities left for the shuffled dataset. Connections with pairwise distance and number of gapless sites values above the threshold curve were assumed to be unreliable and discarded. As a result, the connectivity map contains disjoint clusters of homolog sequences.

The procedure was then applied to a large human protein dataset of sequences, resulting in 445 disjoint components in the constructed connectivity map. We tested the accuracy of the recovered connectivities using the GO associations of the proteins in the dataset. A reference connectivity map was produced by assuming a connection between any two sequences if there was at least one shared GO term. The precision of the proposed method was measured at 0.981. We compared this performance with a conventional thresholding operation over the pairwise distances as is frequently done in clustering studies. This time, the precision measure was 0.799, resulting in a nearly 20% drop in performance. We also employed the method for the functional clustering of a highly cited standard protein dataset from the literature. The method was run to cluster 866 enzymes in 91 families, and achieved an F-measure of 0.882. This performance was again compared with the performance of a classical thresholding operation employing only pairwise distances. This time, an F-measure of 0.827 was achieved at the same automatically selected threshold. For a fair comparison between the methods, average clustering performances including all threshold curves were calculated and found as

0.768 for the proposed method and 0.712 for the conventional thresholding operation. These results indicate the effectiveness of the proposed method in detecting false homologies between the input sequences, and its accuracy in functional separation of amino acid sequences following a multiple sequence alignment procedure. However, multiple sequence alignments on very large datasets required impractically high computation times. For this reason, we took a different approach for the identification of functional and evolutionary relationships on very large datasets.

The third and final approach concerned the identification of the sequence relationships through highly conserved regions, detailed in Chapter 4. Shared regions between biomolecular sequences usually indicate functional similarities and/or a common history; as a result, the identification of these segments and their associations with the input sequences reveals the relationships between those sequences. An all-against-all pairwise sequence alignment was employed as the first step of the method instead of the time consuming multiple sequence alignment. After the thresholding of the connectivity map over pairwise alignment E-values, Connected Component Analysis was carried out on the disjoint undirected graph followed by maximal clique identification on the members of each connected component. Each maximal clique was composed of sequences with at least one shared region. Next, a residue conservation scoring method was employed to scan the multiple sequence alignment of the member sequences of each clique and locate the highly conserved segments. Shared regions were identified following a smoothing operation on the output of the residue conservation scoring procedure. Profiles of the detected conserved/shared regions were generated and aligned to the input sequences using pairwise profile alignment to obtain all the associations across the dataset. The method produces a binary table with rows indexing the input sequences and columns indexing the recovered conserved regions, and ones indicating an association between the corresponding conserved region and sequence pairs. A second output of the method is the conserved region profiles generated for the alignment step. In addition, a clustering scheme is also supplied for clustering the input sequences based on their shared conserved regions. The input sequence - conserved region association table details the relationships between all sequences through the conserved regions.

In order to observe the biological relevance of the results obtained by the proposed method, we have manually curated and clustered domain sequences from SCOP Database. The clustering performance was compared with alternative biomolecular sequence clustering methods from the literature. The proposed method was generally comparable to the top performing method from the literature. However, on hard cases composed of very large and/or diverse sequences, the proposed method outperformed all the other methods. This result was attributed to focusing on the regions with functional and/or evolutionary signals during the calculation of similarities instead of treating the sequences globally.

Next, the proposed method was applied to a large human protein dataset and a global functional relation map was obtained. One of the things we expected during the development of the method was a significant correspondence of the recovered conserved regions with the structural domains in the associated sequences, since the domains contain functional signatures and they are highly conserved during the course of evolution. The method could then be used to identify structural domains in amino acid sequences. In order to test the idea, the human protein dataset was queried against manually curated structural domain profiles in well-established online domain databases (Pfam and NCBI CDD). The results of these queries were used as reference to test the proposed method. Next, the conserved region profiles were aligned to the domain profiles in these databases seeking highly significant alignments to pair conserved regions with reference domains. Finally, the domain associations of input sequences were identified using sequence to conserved region and conserved region to reference domain correspondences. These assignments were compared to the reference assignments to compute performance measures. The proposed method was able to identify the reference domains in the human protein sequences with 0.744 and 0.776 sensitivity measures on Pfam and NCBI CDD databases, respectively. It’s also important here to note that more than 77% and 61% of the reference domain hits on Pfam and NCBI CDD databases respectively were from multi-domain proteins, representing much harder cases compared to single-domain sequences. With nearly 76% of all domains identified in the human protein dataset, it was concluded that the proposed method can also be employed to discover structural domains on large amino acid sequence sets, such as whole proteomes. Furthermore, the conserved regions that could not be aligned with any reference domains during the last analysis may correspond to novel, undiscovered domains. Similarly, the associations of the input sequences over these regions may be new, undocumented relationships. It would be interesting to inspect these cases with specialists focused on these particular sequences as future work.

In summary, in this dissertation, a computational method was developed successfully for the automatic identification of sequence relationships, especially in large and diverse sets. The method was applied to a large human protein dataset, producing a global functional relation map detailing the associations between human proteins through shared conserved regions. This map remains to be explored in future studies to elucidate the organization of the molecular machinery in cells from a functional, structural as well as evolutionary perspective.

REFERENCES

Agrafiotis D.K. and Xu H. (2002) A self-organizing principle for learning nonlinear manifolds. *Proc. Natl. Acad. Sci. U.S.A.*, 99, 15869- 15872.

Ahrens J.H. and Dieter U. (1974). Computer Methods for Sampling from Gamma, Beta, Poisson and Binomial Distributions. *Computing,* 12, 223–246.

Altschul S.F., *et al.* (1990) Basic local alignment search tool. *J. Mol. Biol.*, 215, 403-410.

Altschul S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, 25, 3389-3402.

Andreeva A. *et al.* (2008) Data growth and its impact on the SCOP database: new developments. *Nucleic Acids Res.*, 36, 419-425.

Apeltsin L. *et al.* (2011) Improving the quality of protein similarity network clustering algorithms using the network edge weight distribution. *Bioinformatics*, 27, 326-33.

Bailey T.L. *et al.* (2009) MEME SUITE: tools for motif discovery and searching, *Nucleic Acids Res.*, 37, 202-208.

Bairoch A. and Apweiler R. (2000) The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. *Nucleic Acids Res.*, 28, 1.

Bardwell V.J. and Treisman R. (1994) The POZ domain: a conserved protein-protein interaction motif. *Genes Dev.*, 8, 1664-1677.

Boyle R. and Thomas R. (1988) Computer Vision: A First Course. Blackwell Scientific Publications, UK.

Brigham E.O. (2002) The Fast Fourier Transform. Prentice-Hall, New York.

Bron C. and Kerbosch J. (1973) Algorithm 457: finding all cliques of an undirected graph. *Commun. ACM*, 16, 575–577.

Brown S.D. *et al.* (2006) A gold standard set of mechanistically diverse enzyme superfamilies. *Genome Biology*, 7, 8.

Bryc W. (1995) The normal distribution: characterizations with applications. Springer-Verlag, Heidelberg, Germany.

Carbon S. *et al.* (2009) AmiGO Hub, Web Presence Working Group. AmiGO: online access to ontology and annotation data. *Bioinformatics,* 25, 288-9.

Chandonia J.M. *et al.* (2004) The ASTRAL Compendium in 2004. *Nucleic Acids Res.*, 32, 189-92.

Cover T.M. and Hart P.E. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory,* 13, 21–27.

Cristianini N. and Shawe-Taylor J. (2000) An Introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press, UK.

Csardi G. and Nepusz T. (2006) The igraph software package for complex network research. *InterJournal,* Vol. Complex Systems.

Cuff A.L. *et al.* (2011) Extending CATH: increasing coverage of the protein structure universe and linking structure with function. *Nucleic Acids Res.*, 39, 420–6.

Dayhoff M.O. *et al.* (1978). A model of evolutionary change in proteins. *Atlas of Protein Sequence and Structure,* 5, 345–352.

D'haeseleer P. (2006). What are DNA sequence motifs. *Nature Biotechnology,* 24, 423-425.

Diestel R. (2010) Graph Theory. Springer-Verlag, Heidelberg, Germany.

Dijkstra E.W. (1959) A note on two problems in connection with graphs. *Numerische Mathematik* 1, 269–271

Duda R.O., Hart P.E. and Stork D.G. (2001) Pattern Classification. John Wiley & Sons, New York.

Edgar R.C. (2004). MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.* 32, 1792–97.

Enright A.J. and Ouzounis C.A. (2000) GeneRAGE: a robust algorithm for sequence clustering and domain detection. *Bioinformatics,* 16, 451–457.

Enright A.J. *et al.* (2002) An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res.*, 30, 1575–1584.

Farnum M.A., Xu H. and Agrafiotis D.K. (2003) Exploring the nonlinear geometry of protein homology. *Protein Science*, 12, 1604-1612.

Felsenstein J. (1978). Cases in which parsimony and compatibility methods will be positively misleading. *Systematic Zoology*, 27, 401–410.

Finn R.D. *et al.* (2011) HMMER web server: interactive sequence similarity searching. *Nucleic Acids Res.*, 39, 29-37.

Finn R.D. *et al.* (2010) The Pfam protein families database. *Nucleic Acids Res.*, Database Issue 38, 211-222.

Gallager R.G., Humblet P.A. and Spira P.M. (1983) A distributed algorithm for minimum-weight spanning trees. *ACM Transactions on Programming Languages and Systems,* 5, 66–77.

Hamming R.W. (1950) Error detecting and error correcting codes. *Bell System Technical Journal*, 29, 147–160.

Heger A. and Holm L. (2003) Exhaustive enumeration of protein domain families. *J Mol Biol.,* 328, 749-67.

Huelsenbeck J.P. *et al.* (2001) Bayesian inference of phylogeny and its impact on evolutionary biology. *Science* 294, 2310–2314.

Hunter S. *et al.* (2012) InterPro in 2011: new developments in the family and domain prediction database. *Nucleic Acids Res.*, 40, 306-312.

Jitrapakdee S., *et al.* (2008) Structure, Mechanism and Regulation of Pyruvate Carboxylase. *Biochemical Journal*, 413, 369-387.

Jolliffe I.T. (1986) Principal Component Analysis. Springer-Verlag, New York.

Joseph J.M. and Durand D. (2009) Family classification without domain chaining. *Bioinformatics*, 25, 45-53.

Jukes T.H. and Cantor C.R. (1969) Evolution of Protein Molecules. Pp. 21-123 in H. N. Munro, ed. Mammalian protein metabolism. Academic Press, New York.

Katoh M. and Kuma M. (2002) MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Res.*, 30, 3059-3066.

Kimura M. (1980) Simple method for estimating evolutionary rate of base substitution through comparative studies of nucleotide sequences. *Journal of Molecular Evolution*, 16, 111-120.

Kimura M. (1983) The neutral theory of molecular evolution. Cambridge University Press, UK.

Kruskal J.B. and Wish M. (1978) Multidimensional Scaling. Sage Publications, Beverly Hills and London.

Larkin M.A. *et al.* (2007) ClustalW and ClustalX version 2. *Bioinformatics*, 23, 2947-2948.

Lasko T.A. *et al.* (2005) The use of receiver operating characteristic curves in biomedical informatics. *Journal of Biomedical Informatics*, 38, 404–415.

Marchler-Bauer A. *et al.* (2002) CDD: a database of conserved domain alignments with links to domain three-dimensional structure. *Nucleic Acids Res.*, 30, 281-3.

Marchler-Bauer A. *et al.* (2003) CDD: a curated Entrez database of conserved domain alignments. *Nucleic Acids Res.*, 31, 383-7.

Marchler-Bauer A. *et al.* (2011) CDD: a Conserved Domain Database for the functional annotation of proteins. *Nucleic Acids Res.*, 39, 225-9.

Miele V. *et al.* (2012) High-quality sequence clustering guided by network topology and multiple alignment likelihood. *Bioinformatics*, 28, 1078-85.

Mohseni-Zadeh S. *et al.* (2004) Cluster-C, an algorithm for the large-scale clustering of protein sequences based on the extraction of maximal cliques. *Computational Biology and Chemistry*, 28, 211-218.

Needleman S.B. and Wunsch C.D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, 48, 443–53.

Nepusz T. *et al.* (2010) SCPS: a fast implementation of a spectral method for detecting protein families on a genome-wide scale. *BMC Bioinformatics*, 11, 120.

Notredame C., Higgins D.G. and Heringa J. (2000) T-Coffee: A novel method for fast and accurate multiple sequence alignment. *J Mol Biol*, 302, 205–217.

Paccanaro A. *et al.* (2006) Spectral clustering of protein sequences. *Nucleic Acids Res.*, 34, 1571-1580.

Pearson W.R. and Lipman D.J. (1988) Improved tools for biological sequence comparison. *PNAS*, 85, 2444-2448.

Phillips D.C. (1966) The three-dimensional structure of an enzyme molecule. *Scientific American*, 215, 78–90.

Punta M. *et al.* (2012) The Pfam protein families database. *Nucleic Acids Res.*, 40, 290-301.

Rajasekaran S. *et al.* (2010) A computational tool for identifying minimotifs in protein-protein interactions and improving the accuracy of minimotif predictions. *Proteins: Structure, Function, and Bioinformatics*, 79, 153-164.

Ronquist F. and Huelsenbeck J.P. (2003) MRBAYES 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics* 19, 1572–1574.

Roweis S. and Saul L. (2000) Nonlinear dimensionality reduction by LLE. *Science*, 290, 2323-2326.

Sammon J.W. (1969) A nonlinear mapping for data structure analysis. *IEEE Trans. Computers*, 18, 401-409.

Schultz J. *et al.* (1998) SMART, a simple modular architecture research tool: Identification of signaling domains. *PNAS*, 95, 5857-5864.

Sievers F. *et al.* (2011) Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. *Mol. Syst. Biol.*, 7.

Sigrist C.J.A. *et al.* (2010) PROSITE, a protein domain database for functional characterization and annotation. *Nucleic Acids Res.*, 38 (Database issue), 161-6.

Smith T.F. and Waterman M.S. (1981) Identification of Common Molecular Subsequences. *J Mol Biol*, 147, 195–197.

Tavare S. (1986) Some Probabilistic and Statistical Problems in the Analysis of DNA Sequences. *Lectures on Mathematics in the Life Sciences*, 17, 57-86.

Tenenbaum J.B., de Silva V. and Langford J.C. (2000) A global geometric framework for nonlinear dimensionality reduction. *Science*, 290, 2319-2323.

The Gene Ontology Consortium (2000) Gene ontology: tool for the unification of biology. *Nat. Genet.*, 25, 25-9.

The MathWorks Inc. (2010) MATLAB version 7.10.0. Natick, Massachusetts, USA.

The UniProt Consortium (2011) Ongoing and future developments at the Universal Protein Resource. *Nucleic Acids Res.*, 39, 214-219.

Thompson W.A. *et al.* (2007) The Gibbs Centroid Sampler. *Nucleic Acids Res.*, 35 (Web Server issue), 232-7.

Valdar W.S. (2002) Scoring residue conservation. *Proteins: Structure, Function, and Genetics*, 48, 227–241.

Venter J.C. *et al.* (2001) The sequence of the human genome, *Science*, 291, 1304-1351.

Wahlberg N. *et al.* (2005) Synergistic effects of combining morphological and molecular data in resolving the phylogeny of butterflies and skippers. *Proc. R. Soc. B.*, 272, 1577-1586.

Wetlaufer D.B. (1973) Nucleation, rapid folding, and globular intrachain regions in proteins. *PNAS*, 70, 697–701.

Wittkop T. *et al.* (2007) Large scale clustering of protein sequences with FORCE -A layout based heuristic for weighted cluster editing. *BMC Bioinformatics*, 8, 396.

Wittkop T. *et al.* (2010) Partitioning biological data with transitivity clustering. *Nature Methods*, 7, 419-20.

Zhang D.D. *et al.* (2004) Keap1 is a redox-regulated substrate adaptor protein for a Cul3-dependent ubiquitin ligase complex. *Mol. Cell. Biol.*, 24, 10941-10953.

VITA

Tunca Doğan was born in Ankara, Turkey, in January 12th, 1982. He attended the Food Engineering Department in Middle East Technical University; Ankara, Turkey and acquired his B.S. degree in June 2005. Later the same year, he attended M.Sc. program at the same department and graduated in February 2008. Since then, he has been a Ph.D. student at Bioengineering Doctoral Program, Biotechnology and Bioengineering Graduate Program, Izmir Institute of Technology, Izmir, Turkey.

He worked as a research assistant at Food Engineering Department in Middle East Technical University; Ankara, Turkey, between the years 2005 and 2008. Later in 2008 he became a research assistant at Biotechnology and Bioengineering Graduate Program, Izmir Institute of Technology, Izmir, Turkey. From 2009 to now on, he has been working at the same position in Electrical and Electronics Engineering Department in Izmir Institute of Technology, Izmir, Turkey.

His current fields of research are bioinformatics, computational biology, statistical learning, biomedical information analysis, biomolecular sequence analysis and protein function analysis.