BIOINFTool: Bioinformatics and sequence data analysis in molecular biology using Matlab

Mai S. Mabrouk[1], Marwa Hamdy[2], Marwa Mamdouh[2], Marwa Aboelfotoh[2],Yasser M. Kadah[2]
[1]Biomedical Engineering Department, Misr University, Egypt.,[2] Biomedical Engineering Department, Cairo University, Egypt.
E-mail: msm_eng@k-space.org

Abstract **Bioinformatics is conceptualizing biology in terms of macromolecules (in the sense of physical-chemistry) and then applying "informatics" techniques (derived from disciplines such as applied mathematics, computer science, and statistics) to understand and organize the information associated with these molecules, on a large-scale. Analyses in bioinformatics predominantly focus on three types of large datasets available in molecular biology: macromolecular structures, genome sequences, and gene expression data. Techniques developed by computer scientists have enabled researchers at Celera Genomics, the Human Genome Project consortium, and other laboratories around the world to sequence the nearly 3 billion base pairs of the roughly 40,000 genes of the human genome. This feat would have been virtually impossible without computational methods. The aim of this work is to organize data in a way that allows researchers to access existing information and to submit new entries as they are produced. The information stored in these databases is essentially useless until analyzed, thus our purpose is extended much further to develop tools and resources that aid in the analysis of data. We can use these tools to analyze the data and interpret the results in a biologically meaningful manner.**

***Keywords* - bioinformatics, computational biology, DNA sequences, microarray processing, biotechnology, phylogenetic tree.**

I. INTRODUCTION

In the last few years, advances in molecular biology and the equipment available for research in this field have allowed the increasingly rapid sequencing of large numbers of the genomes of several species including the human genome which is completed in 2003. Finishing the human genome does not mean that it is 100% accurate but still some gaps present although its number is greatly reduced. One of the central goals of the effort to analyze the human genome is the identification of all genes, which are generally defined as stretches of DNA that code for particular proteins. After completing the human genome, the number of human protein-coding genes has been reduced from the expected 35,000 to only 20,000 – 25,000, a surprisingly low number of our species (Mount, D. W., 2003)

This deluge of information has necessitated the careful storage, organization and indexing of sequence information. Information science has been applied to biology to produce the field called **Bioinformatics**.

This decade was in effect the time when the field of computational biology took shape as an independent discipline, with its own problems and achievements. Efficient algorithms were developed to cope with an increasing volume of information, and their computer implementations were made available for wider scientific community. It had already become clear that computer analysis of nucleotide sequences was essential for better under standing of biology (Gingeras and Ropert, 1980). Theoretical developments in sequence analysis, for example the computation of evolutionary distances (Sellers, 1980), were followed by the development of key algorithms, such as the Smith- Waterman dynamic programming sequence alignment algorithm (Smith-Waterman, 1981 a, b) and the FASTA family of algorithms for database searching (Lipman and Pearson, 1985; Wibur and Lipman, 1983). The most pressing tasks in bioinformatics involve the analysis of sequence information. **Computational Biology** is the name given to this process, and it involves the following:

- Finding the genes in the DNA sequences of various organisms
- Developing methods to predict the structure and/or function of newly discovered proteins and structural RNA sequences.
- Clustering protein sequences into families of related sequences and the development of protein models.
- Aligning similar proteins and generating phylogenetic trees to examine evolutionary relationships.

In this paper, we developed a software package called **BIOINFTool** which is useful in exploration, interpretation and visualization of data in molecular biology. It can be used with easy graphic user interface to manipulate aligned sequences, calculate evolutionary distances, micro array analysis and inferred phylogenetic trees. BIOINFTool is written in Matlab and language and has been tested on the WINDOWS platform with Matlab version 7.0.1. The main functions implemented are: sequence

manipulation, computation of evolutionary distances derived from nucleotide, phylogenetic tree construction, sequence statistics , micro array gene expression analysis and graphic functions to visualize the results. BIOINFTool requires a single file of nucleotide or amino acid sequence in a Fasta format.

II. METHODOLOGY

Although many of the computational techniques developed by researchers in bioinformatics have been beneficial to scientists and entrepreneurs in other fields, most of these redundant discoveries represent a detour from addressing the main molecular biology challenges. The aim is to identify and describe specific information technologies in enough detail to reason from first principles when critically evaluate a glossy print advertisement, banner ad, or publication describing an innovative application of computer technology to molecular biology.

Humans are remarkable in their ability to recognize patterns by 'just looking at it'. So far, programmers have had only limited success in devising algorithms (the computer equivalent of laboratory protocols) for pattern recognition. At the same time, humans are poor at highly repetitive tasks with large quantities of data.

Graphic similarity comparisons use the power of the computer to present relationships between sequences in such a graphic form that enables the human researcher to discern patterns in the data. For this study, we developed a research software package that is having a simple graphical user interface that helps researchers working in this field. This package consists of four main modules, these modules are: pairwise alignment module, Statistical module, phylogenetic tree module and micro array gene expression module.

A. Pairwise sequence alignment module

Sequence alignment is the procedure of comparing two (pair-wise alignment) or more (multiple sequence alignment) sequences by searching for a series of individual characters or character patterns that are in the same order in the sequences.

In BIOINFTool, we used the dynamic programming methods which is first used for global alignment of sequences by Needleman and Wunsch (1970) and for local alignment by Smith and Waterman (1981a), provides one or more alignments of the sequences. An alignment is generated by starting at the ends of the two sequences and attempting to match all possible pairs of characters between the sequences and by following a scoring scheme for matches, mismatches, and gaps. This procedure generates a matrix of numbers that represents all possible alignments between the sequences. The highest set of sequential scores in the matrix defines an optimal alignment. The dynamic programming method is guaranteed in a mathematical sense to provide the optimal (very best or highest scoring) alignment for a given set of user-defined variables, including choice of scoring matrix and gap penalties. Fortunately, experience with the dynamic programming method has provided much help for making the best choices, and dynamic programming has become widely used There are two types of sequence alignment, global and local, **In global alignment**, an attempt is made to align the entire sequence, using as many characters as possible, up to both ends of each sequence. Sequences that are quite similar and approximately the same length are suitable candidates for global alignment. **In local alignment**, stretches of sequence with the highest density of matches are aligned, thus generating one or more islands of matches or sub alignments in the aligned sequences. Local alignments are more suitable for aligning sequences that are similar along some of their lengths but dissimilar in others, sequences that differ in length or sequences that share a conserved region or domain.

For applying a global alignment or/and local alignment and getting a score for both of them, we should have a sequence as in our package the user is asked to enter the sequence by two ways. The first way by the accession numbers of the sequence to retrieve the sequences in its ORF (open reading frames). In the second way, the user is asked to enter the sequence directly. The user also should enter the required substitution matrices upon his choice. Finally, we can get global alignment (NW) or/and local alignment (SW) with a score that determines the degree of similarity. We also introduce an option of saving these sequences in a FASTA format file giving the ability to read this file again any time. Dot plots are one of the easiest ways to look for similarity between sequences. The diagonal line shown below indicates that there may be a good alignment between the two sequences. Our package has ability to show dot plot of two sequences for example if the user enters two similar sequences, then the dot plot will be liner.

The package also introduces an option to perform a Randomization test by comparing any two DNA or amino acid sequence then obtaining an authentic score. Then, we scramble the bottom sequences 100 times obtaining 100 “randomized” scores. If the comparison is “real” we expect the authentic score to be several standard deviations above the mean of the “randomized” scores. The package can also generate a random sequence of DNA, RNA and protein from a finite alphabet.

B. Sequence statistics module.

One of the most important tasks of our package is to investigate the nucleotide content for any DNA sequence. This module uses some sequence statistics operations to determine mono-nucleotide, di-nucleotide, and tri-nucleotide content; the user can also uses this module to locate the sequence ORF. It is known that it is a difficult task to determine the

protein-coding sequence for a eukaryotic gene because introns (noncoding sections) are mixed with exons. However, prokaryotic genes generally do not have introns and mRNA sequences have the introns removed. So, Identifying the start and stop codons for translation determines the protein-coding section or open reading frame (ORF) in a sequence is important, the task that our package can do perfectly. Once the ORF for a gene or mRNA is known, the user can translate a nucleotide sequence to its corresponding amino acid sequence. Many public databases for nucleotide sequences are accessible from the Web for this purpose. In this module, users can easily extract more specific positions by using functions developed in the BIOINFTool. For each sequence, some basic statistics such as nucleotide composition and GC content can be reported**.** Other functions include the calculation of segregating sites, taking the reverse complement or translating a sequence and determining the sequence complexity**.**

C. Phylogenetic tree module.

Phylogenetics is the study of evolutionary relationships. Phylogenetic analysis is the means of *inferring* or estimating these relationships. The evolutionary history inferred from phylogenetic analysis is usually depicted as branching, treelike diagrams that represent an estimated pedigree of the inherited relationships among molecules (‘‘gene trees’’), organisms, or both. Phylogenetics is sometimes called cladistics because the word ‘‘clade,’’ a set of descendants from a single ancestor, is derived from the Greek word for branch. However, cladistics is a particular method of hypothesizing about evolutionary relationships.The resulting relationships from cladistic analysis are most commonly represented by a phylogenetic tree.A phylogenetic analysis of a family of related nucleic acid or protein sequences is a determination of how the family might have been derived during evolution. The evolutionary relationships among the sequences are depicted by placing the sequences as outer branches on a tree. The branching relationships on the inner part of the tree then reflect the degree to which different sequences are related. **In BIOINFOtool we can construct a phylogenetic tree by** using sequence data which is helpful when you are trying to visualize the evolutionary relationships between species. The sequences can be multiply aligned or a set of nonaligned sequences, users can select a method for calculating pairwise distances between sequences, then select a method for calculating the hierarchical clustering distances used to build a tree. After locating the GenBank accession codes for the sequences that a user is interested in studying, our package can create a phylogenetic tree with the data. In **BIOINFOtool** we can create a phylogenetic tree by two methods: Firstly, we can enter species in the range of 10 species and their accession numbers and the program will create a phylogenetic tree with this data. Secondly, we can create a phylogenetic tree by writing the name of the file containing the data such as pf00002.tree; the **BIOINFOtool** will create a phylogenetic tree After we create a phylogenetic tree by any previous methods, we can explore a tree by some process such as listing the members of a tree (leaves), we can select certain branches and leaves by name from a phytree object and the **BIOINFOtool** will select them by coloring them in red color ,we can remove branch nodes one by one after selected them, and you can remove potential outliers in the tree. Users can find the closest species to selected specie in a tree and List their names. Also, users can extract a subtree from the whole tree by two ways: By removing unwanted leaves by choosing a certain distance, by entering at least 2 species and the program will extract their subtree. It is important to mention that the new version of BIOINFTool will enable users to enter unlimited accession numbers by applying some dynamic methods.

D. Micro array gene expression data analysis module.

Capturing and storage of microarray data is not an end in itself. The amounts of data from even a single microarray experiment are so large, that software tools have to be used to make any sense out of it. Clustering and class prediction are typical methods currently used in gene expression data analysis. The raw data that are produced from microarray experiments are the hybridised microarray images. To obtain information about gene expression levels, these images should be analysed, each spot on the array identified, its intensity measured and compared to the background. This is called image quantitation. Image quantiation is done by image analysis software. To obtain the final gene expression matrix from spot quantiations, all the quantities related to some gene (either on the same array or on arrays measuring the same conditions in repeated experiments) have to be combined and the entire matrix has to be scaled to make different arrays comparable [15]. Conceptually, a gene expression database can be regarded as consisting of three parts: the gene expression data matrix, gene annotation and sample annotation.

In the BIOINFTool, users First must have the file of micro array which contain the image result from the experiment, our package takes the file name to read its data, then make some processing on them like: to get the gene names of certain region, enter the range of genes. We depend on intensity of color in micro array chip to determine the level of gene expression in each spots so, we toke the median of pixel intensity in a region surrounding the spot to decrease error. In this work, users can get the plot of median of foreground red channel, median of foreground green channel, median background red channel and median background green channel.

III. RESULTS

In BININFTool, after reading a sequence, user can use the sequence statistics functions to analyze sequences form its open reading frames.

User can count the nucleotides in any sequence and also in its reverse complement as shown in figure (1).

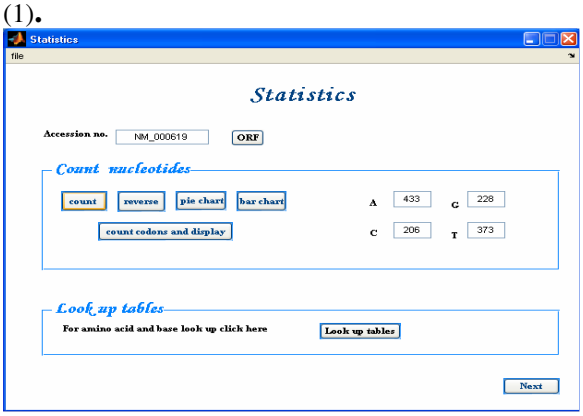

Fig 1; Count the nucleotides

Trinucleotides (codon) code for an amino acid, and there are 64 possible codons in a nucleotide sequence. Knowing the percent of codons in the sequence can be helpful when you are comparing with tables for expected codon usage.

In this work, user can determine 64 possible codons in nucleotide sequence as shown in figure (2).

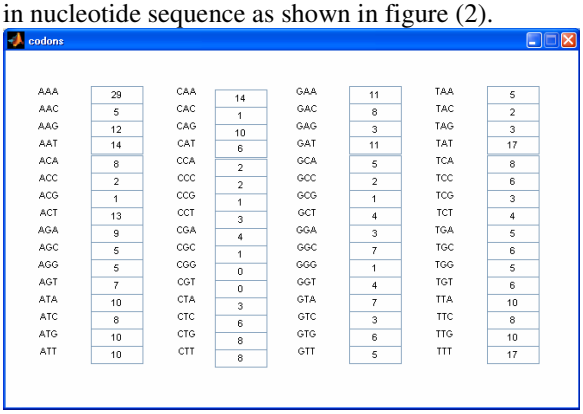

Fig 2; Count codons of sequence (acc.no. is NM_000619)

this profile is enough information to identify a protein. Using the amino acid composition, atomic composition, and molecular weight, user can a search public databases for similar proteins, converting it to an amino acid sequence and determine its amino acid composition as shown in figure (3).

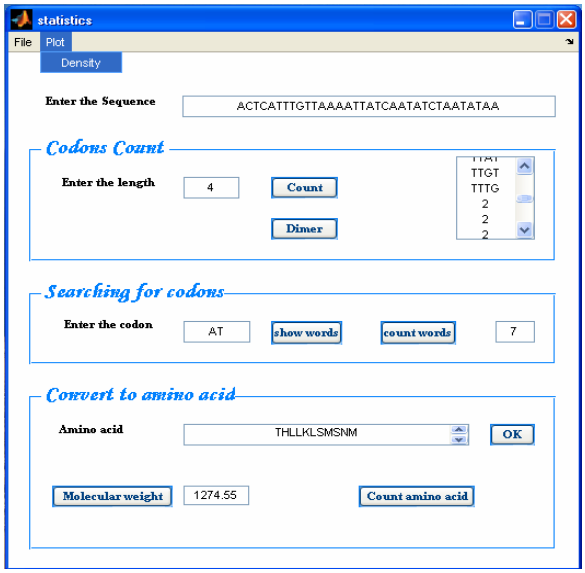

Fig 3; statistics form with sequence input and show count codon with length of the codon, and count any word of sequence ,amino acid that is converted from the sequence and the molecular weight of amino acid.

User also can Plot monomer densities and combined monomer densities in a graph of any given sequence as shown in figure (4).

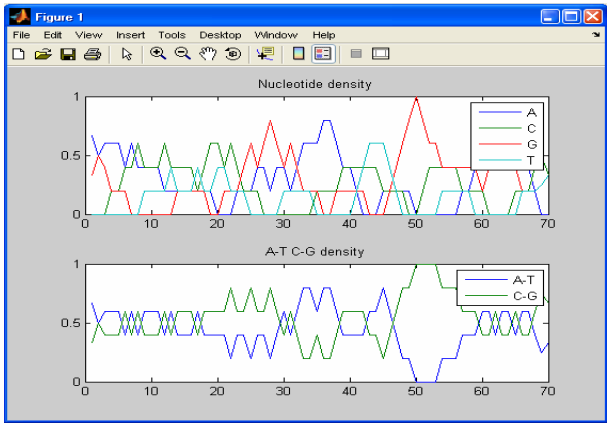

Fig 4; plot of monomer density of sequence (Gaagacacactacctgactcctgcaggaggatgaaacaagcctttcaggggcc cgtgcagaaggaactgc)

In BIOINFTool, the user can access the web to get any sequence by two ways either by taking the accession numbers of the sequences from the user as shown in figure (5) or by taking the sequence from the user as shown in figure (6) and retrieve the sequences in open reading frames as shown in figure (7), and by using substitution matrices (pam50, pam250, blosum30, blosum62 and blosum80) we can get global alignment (NW) or/and local alignment (SW) as shown in figure (8. a) and (3.b) and get a score for both them . It can draw a dot plot for them as shown in figure (9).

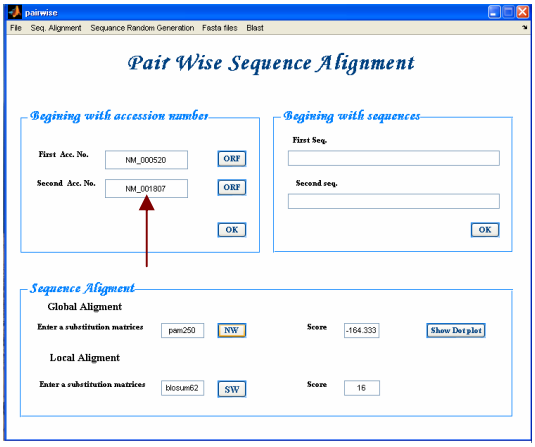

Fig 5, Pair wise sequence alignment, beginning with accession number.

Fig 6, Pairwise sequence alignment, beginning with sequences.

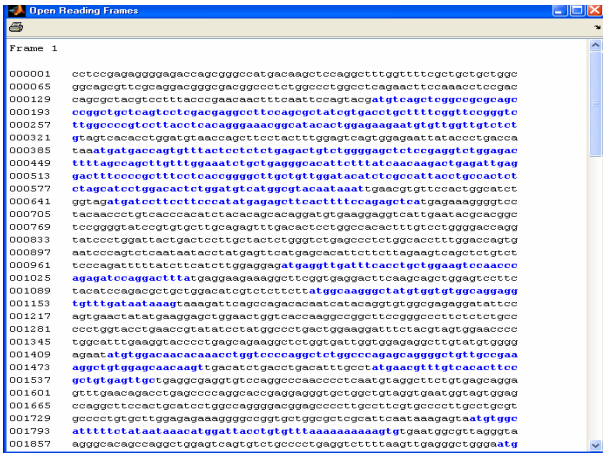

Fig 7, open reading frame of sequence NM_001807.

Fig 8. a, Global alignment (NW) by using pam250 substitution matrices.

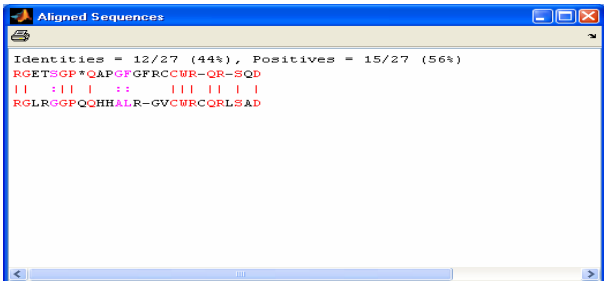

Fig 8. b, Local alignment (SW) by using blosum62 substitution matrices

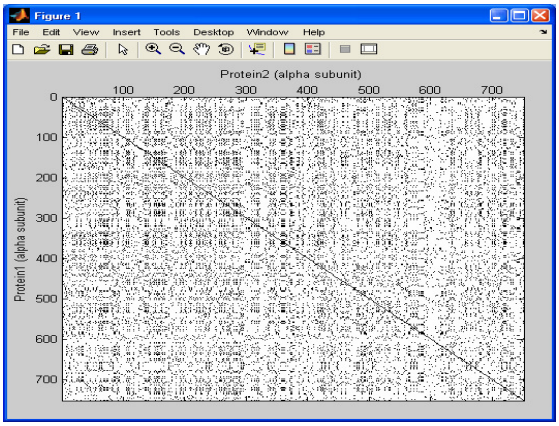

Fig 9, dot plot of two similar sequences

It can compare two proteins using randomization test as shown in figure (10). It can read and save a file in fasta format as shown in figure (11.a, b). It can generate a random sequence from a finite alphabet of RNA, DNA and Protein as shown in figure (12. a, 12. b, 12. c).

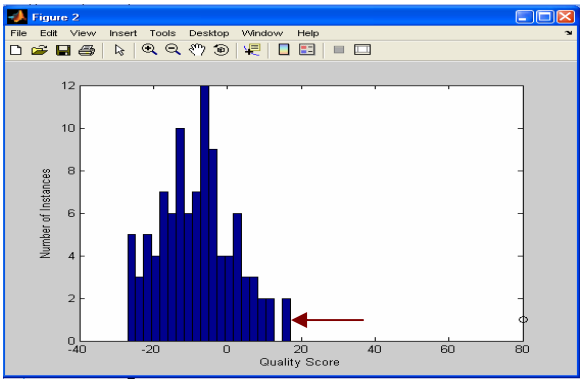

Fig 10, Randomization test between two protein sequences (seq1=ggtgcacacaagatagatagacacaccagagagataatataggagag) and (seq2=ggtgcacacaagatagatagacacaccagagagataatataggagaga)

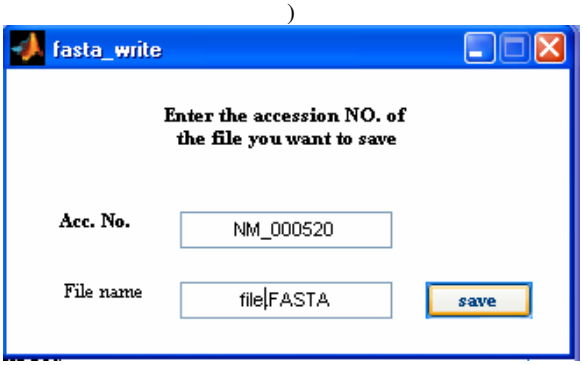

Fig 11.a, save fasta file

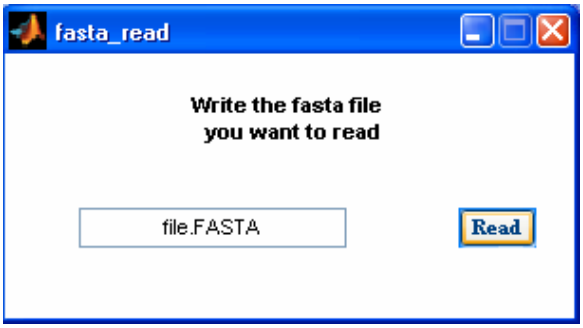

Fig 11.b, read fasta file

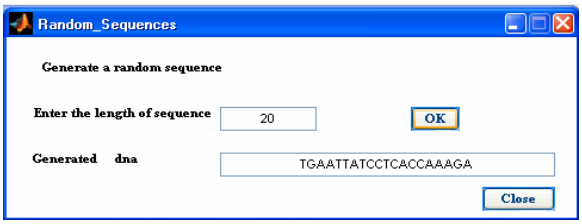

Fig 12.a, Generate a random DNA sequence

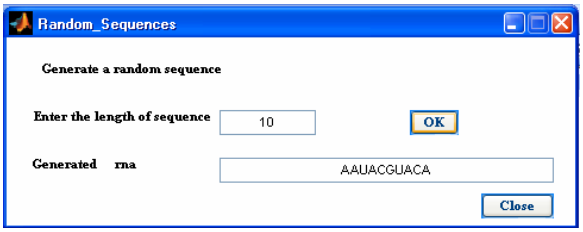

Fig 12. b, Generate a randomRNA sequence

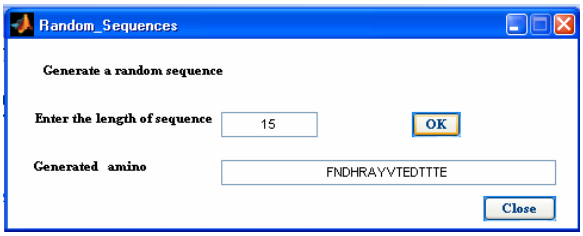

Fig 12.c, Generate a random protein sequence

In the BIOINFTool, user must have the file of micro array which contain the image result from the experiment, enter the file name to read its data, then make some processing on them like: to get the gene names of certain region, enter the range of genes, this is shown in figure (13).

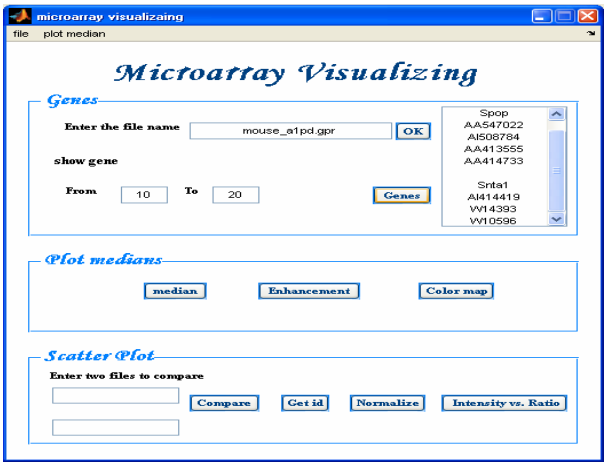

Fig 13; Micro array visualization form of this program

If there are two columns in the microarray data structure labeled 'F635 Median - B635' and 'F532 Median - B532'. These columns are the differences between the median foreground and the median background for the 635 nm channel and 532 nm channels respectively. These give a measure of the actual expression levels.

To compare between them, user must have the files of micro array (mouse_a1pd.gpr –mouse_a1_wt.gpr). A simple way to compare the two channels is with a log plot as shown in figure (14) .Points that are above the diagonal in this plot correspond to genes that have higher expression levels.

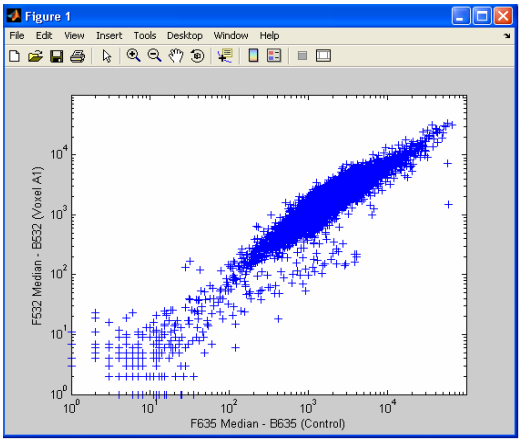

Fig 14; compare between the median foreground and the median background for the 635 nm channel and 532 nm channel

Also, if the user has the file of micro array which contain the image result from the experiment, our tool takes the file name to read its data, then it will show number of genes of this file, to get name of any gene, you can enter its number, then the program will show its name as shown in figure (15)

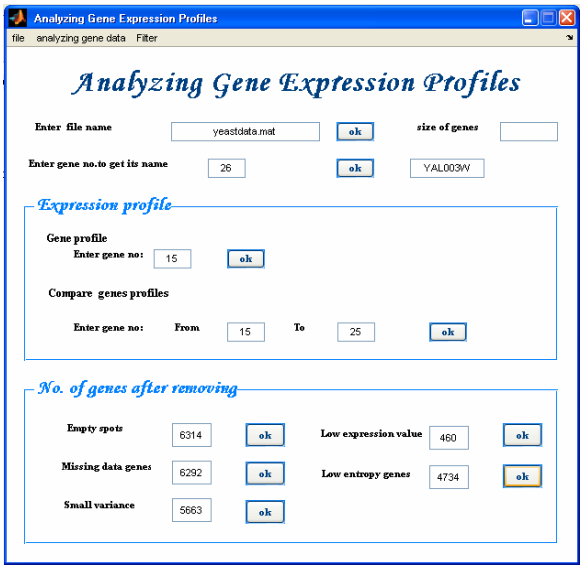

Fig 15; Analyzing gene expression profile, the file name

To get a behavior of any genes expression, user can enter its number, and then our tool will show its profile as shown in figure (16.a), to compare between several genes expression profile user must enter the range of these genes then the BIOINFTool will show the comparison of the profiles of these genes as shown in figure (16. b).

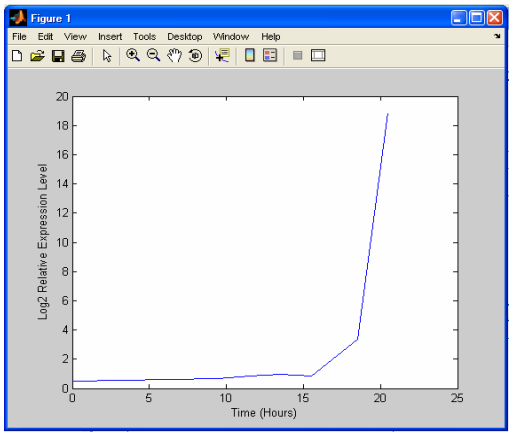

Fig 16. a; Log2 gene expression level over time (behavior of gene that has number 15)

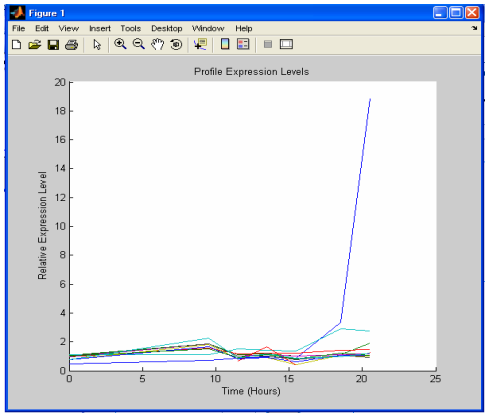

Fig 16.b; expression level over time (behavior of range 15 - 25)

***In the BIOINFOtool*,** user can create a phylogenetic tree by two methods: *Firstly*, we can enter species in the range of 10 species and their accession numbers as shown in figure (17. a) and the **BIOINFOtool** will create a phylogenetic tree with the data as shown in figure (17. b). *Secondly,* we can create a phylogenetic tree by writing the name of the file containing the data such as pf00002.tree as shown in figure (18. a), the **BIOINFOtool** will create a phylogenetic tree as shown in figure (18. b).

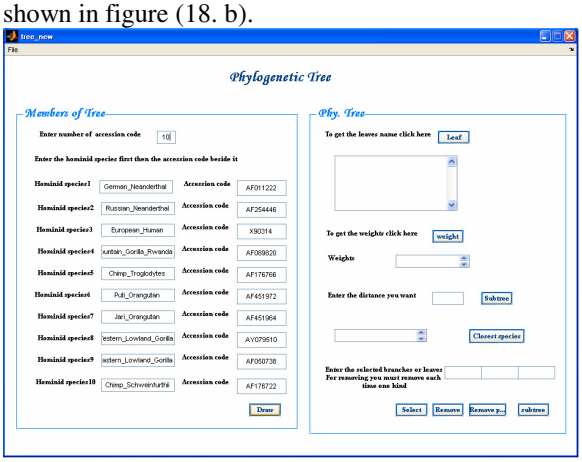

Fig 17. a, Phylogenetic tree form with ten species

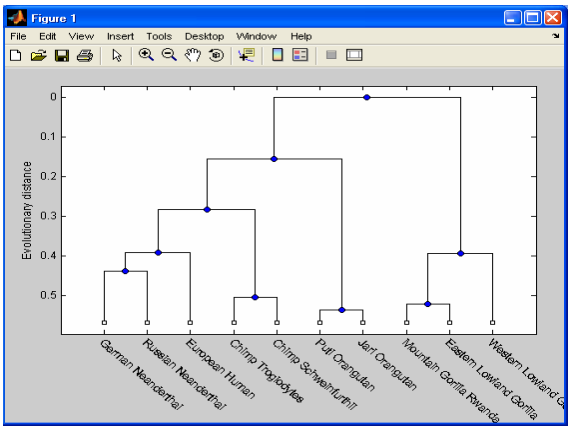

Fig 17. b, Phylogeneitc tree of ten species in the form in figure 17. a

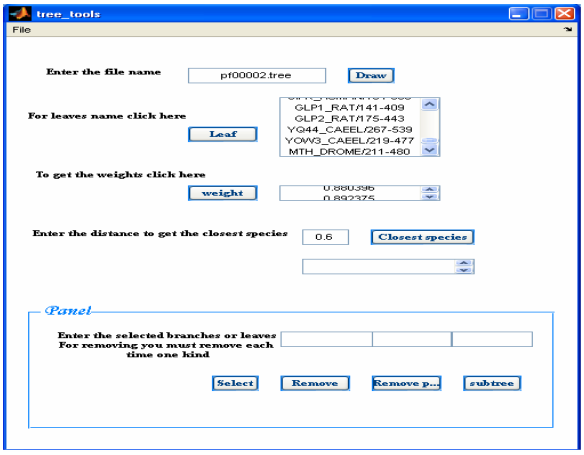

Fig 18. a, Phylogenetic tree form with a file name (pf00002.tree) as input is taken from the user.

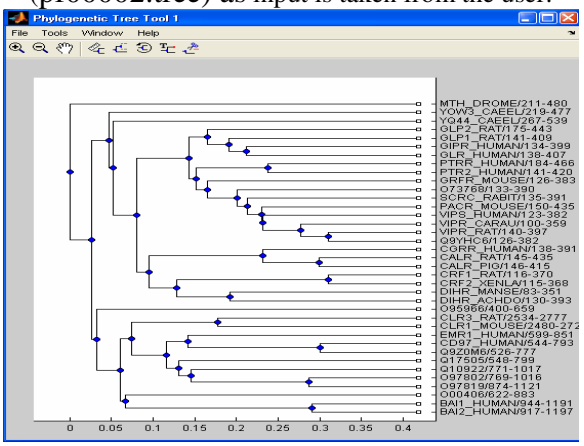

Fig 18. b; Phylogeneitc tree of file (pf00002.tree) in the form in figure 18. a.

IV. Discussion

Some other toolboxes have been developed in Matlab for bioinformatics related analyses. These include PHYLLAB (Rzhetsky A., Morozov P., 2001) and MATARRAY (Venet D., (2003) and MBEEToolbox as well as bioinformatics toolbox developed by MATHWORKS. Other examples can be found at MATLAB CENTRAL (http://www.mathworks/matlabcentral/). PHYLAB mainly focuses on is on creating a maximum likelihood tree based on Bayesian principles using a Markov chain Monte Carlo method to compute posterior parameter distribution. MATARRAY is focused on the analysis of gene expression data from micro arrays but does not address molecular evolution. The bioinformatics toolbox provides a range of bioinformatics functions, including some related to molecular evolution .MBETool provides an extensible, functional framework for users with more specialized requirements to explore and analyze aligned nucleotide or protein sequences.

BIOINFTool provides a wide range of molecular evolution related functions and phylogenetic methods. It is different from other tools as it covers all user work in the field of bioinformatics as consists of four main modules in one package.

V. Conclusion

The BIOINFTool is an effort to introduce a friendly software tool for bioinformatics and data analysis as it is marked by its easy graphical interface. It gives a good environment for the analysis of molecular biology and evolution. It offers a substation set of frequently used functions to manipulate sequences, to calculate genetic distances, to infer phylogenetic tree and to perform a useful micro array gene expression analysis. In summary, BIOINFTool is a useful tool that can aid in the exploration, interpretation and visualization of data in the field of molecular biology.

Acknowledgment

The authors wish to thank an anonymous reviewer for highlighting several works of relevance to this study.

References

Gingeras, T.R. and Roperts, R.J. (1980) Steps toards computer analysis of nucleotide sequences. Science, 209, 1322-1328.

Sellers, P.H. (1980) The theory and computation of evolutionary distances: pattern recognition. J.Algorithms, 1,359-373.

Smith, T.F. and Waterman, M.S. (1981a) Comparison of biosequences. Adv.Appl. Math., 2,482-489.

Smith, T.F. and Waterman, M.S. (1981b) Identification of common molecular subsequences. J. Mol. Biol., 147, 195-197.

Lipman, D., J. and pearson, W.R. (1985) Rapid and sensitive protein similarity searches. Science, 227, 1435-1441.

Mount, D. W. (2003) Bioinformatics Sequence and Genome Analysis http://pevsnerlab.kennedykrieger.org/bioinfo_course.htm.

Needleman, S.B. and Wunsch, C.D. (1970) A gerenal method applicable to the search for similarities in the

amino acid sequence of two protein. J.Mol. Biol., 48, 443-453.

Rzhetsky A., Morozov P. (2001) Markov chain Monte Carlo computation of confidence intervals for substitution- rate variation in proteins. Pac symp Biocomput, 6:203-214.
Venet D., (2003) MatARRAY: a Matlab toolbox for microarray data. Bioinformatics, 19:659-660.
MATLABCentral
[http://www.mathworks.com/matlabcentral/]