Automating Identification of Unique Patterns, Mutation in Human DNA using Artificial Intelligence Technique

B.Mukunthan
Assistant Professor,
Department of Master of Computer Applications,
SVS College of Engineering,
Coimbatore, Tamil Nadu, India

Dr. N.Nagaveni
Associate Professor,
Department of Mathematics,
Coimbatore Institute of Technology,
Coimbatore, Tamil Nadu, India,

ABSTRACT
In molecular biology and genetic engineering, DNA sample identification is not considered as a biometric recognition technology mainly because it's not an automated process i.e. it takes more time to analyze the DNA samples. Mutation identification is still an exigent task as it's a manual process; mutations are changes in a genomic sequence caused by factors such as radiation, mutagenic chemicals, viruses, transposons. The automation of DNA feature extraction process achieved by applying neural network technique which has the advantage over conventional programming, in their ability to solve problem that do not have an algorithmic solution or the available solutions is too complex to be found is discussed in this paper, the proposed technique reduces the complication in precisely analyzing, interpreting the unique repeated patterns of human DNA. In this novel approach the perfect blend made of bioinformatics and neural networks technique results in efficient DNA pattern analysis algorithm with utmost prediction accuracy of unique repeated patterns and mutation, computed by number of correct identification of the target for a set of given inputs.

Keywords
Adaptive Resonance Theory, Simplified fuzzy ARTMAP, Competitive learning, NFPR-processor, Input Generator, Preprocessor, Separator, Discriminator, Comparator, DNA profiling, DNA sequence.

1. INTRODUCTION
Knowledge of DNA sequences has become indispensable for basic biological research. DNA sequencing is applied in various fields such as diagnostic, biotechnology, forensic biology and biological systematic. The DNA sequences of thousands of organisms have been decoded and stored in databases. The sequence information is analysed to determine genes that encode polypeptides, RNA genes, regulatory sequences, structural motifs, and repetitive sequences. A comparison of genes within a species or between different species can show similarities between protein functions, or relations between species. With the growing amount of data, it became impractical to analyse DNA sequences manually.

Neural networks learn by examples so that it can be trained with known examples of a problem to gain knowledge about it so the neural network can be effective to solve unknown or untrained instances of the problem if is aptly trained. A pattern [1] [2] is essentially an arrangement or an ordering, in which some organization of underlying structure can be said to exist i.e. a pattern can be referred to as a quantitative or structural description of an object or some item of interest. A set of patterns that share some common properties can be regarded as pattern class [3] in our case the unique repeated nucleotide sequence from the given Human DNA sample. The concept of applying Artificial Neural Systems (ANS) or Artificial Neural Networks (ANN) or simply Neural Networks in the field of DNA profiling is discussed in this paper.

2. ARTIFICAL NEURAL NETWORK TECHNIQUES
Neural Networks [4] can process information in parallel, at high speed, and in a distributed manner. Neural networks which are simplified models of the biological neuron system, is a massively parallel distributed processing system made up of highly interconnected neural computing elements that have the ability to learn and thereby acquire knowledge and make it available for use. Neural Network architectures [5] have been classified into various types based on their learning mechanisms and other features. Some classes of Neural Network refer to this learning process as training and the ability to solve a problem using the knowledge acquired as inference.

Neural Networks [6] exhibit mapping capabilities, i.e., they can map input patterns to their associated output patterns. Neural Networks architectures can be trained with known examples of a problem before they are tested for their inference. They can, therefore, identify new objects previously untrained. Neural Networks are robust systems and are fault tolerant. They can therefore, recall full patterns from incomplete, partial or noisy patterns.

In Competitive Learning method those neurons which respond strongly to input stimuli have their weights updated, when an input pattern is presented, all neurons in the layer compete and the winning neuron undergoes weight adjustment. Hence it is a "Winner-takes-all" strategy.

Adaptive resonance theory [7] employs a new principle of self organization based on competitive learning .Adaptive resonance theory nets are designed to be both stable and plastic. Neural networks suitable particularly for pattern classification problems in realistic environment is Neural- Fuzzy resonance mapping [8] , it is a vast simplification of fuzzy resonance mapping which possess reduced computational overhead and architectural redundancy when compared to fuzzy resonance mapping.

3. DNA PROFILING AND SEQUENCING

DNA profiling [9][20] also called DNA testing, DNA typing, or genetic fingerprinting, is a technique employed by forensic scientists to assist in the identification of individuals on the basis of their respective DNA profiles. DNA profiles [10] are encrypted sets of numbers that reflect a person's DNA makeup [17], which can also be used as the person's identifier. DNA sequencing theory addresses physical processes related to sequencing DNA .The term DNA sequencing [11] refers to sequencing methods for determining the order of the nucleotide bases—adenine, guanine, cytosine, thymine and uracil(rare case) in a molecule of DNA.

Single nucleotide poly-orphisms [15] are a DNA sequence variation occurring when a single nucleotide A, T, C, or G in the genome (or other shared sequence) differs between members of a species (or between paired chromosomes in an individual). The genome [12] [18] is the entirety of an organism's hereditary information which is encoded either in DNA or, for many types of virus, in RNA. For example, two sequenced DNA fragments from different individuals [16] [19], AAGCCTA to AAGCTTA, contain a difference in a single nucleotide. Various DNA Sequence Formats [13] available are:1) Plain sequence format 2) EMBL format 3) GCG format 4) GCG-RSF (rich sequence format 5) Gen Bank format 6) IG format 7) FASTA format. A sequence file in FASTA format of a given sample is used as an input to that is to be interpreted and analysed.

4. NEURAL-FUZZY PATTERN RECOGNITION PROCESSOR

4.1 Learning Input Generator

The input generator is used for input normalization and it represents the presence of particular feature in the input patterns and its absence. Various conditions for generating normalized learning input are shown in table 1 below.Learning Inputs

$$LIN_{i,\,n} = I_1, I_2 \ldots, I_p \qquad (1)$$

Where $0.1 \le i \le 0.5, \quad 0.1 \le n \le 0.5$

and $p = 4$

Table 1 Conditions for learning input normalization

	Condition	Learning Input	Category
Case 1	i ≠ n or i=n=0.1 and n<=0.5	$LIN_{i,\,n}$ = i, n, 1- i, 1-n e.g. $LIN_{0.1,\,0.1}$ = 0.1, 0.1, (1- 0.1), (1-0.1) $LIN_{0.1,\,0.1}$ = 0.1, 0.1, 0.9, 0.9 $LIN_{0.2,\,0.5}$ = 0.2, 0.5, (1- 0.2), (1-0.5) $LIN_{0.2,\,0.5}$ = 0.2, 0.5, 0.8, 0.5	Category=L(logical)
Case 2	i = n and 0.1> i, n <0.5	$LIN_{i,\,n}$ = i, 1-i, 1-n, n e.g. $LIN_{0.2,\,0.2}$= 0.2, (1-0.2), (1- 0.2), 0.2 $LIN_{0.2,\,0.2}$ = 0.2, 0.8, 0.8, 0.2 $LIN_{0.3,\,0.3}$= 0.3, (1-0.3), (1- 0.3), 0.3 $LIN_{0.3,\,0.3}$= 0.3, 0.7, 0.7, 0.3	Category=ILL(illogical)
Case 3	i =n=0.5	$LIN_{i,\,n}$ = i, n, 1- i, 1-n e.g. $LIN_{0.5,\,0.6}$= 0.5, 0.6, (1- 0.5), (1+ 0.5) $LIN_{0.5,\,0.6}$= 0.5, 0.6, 0.4, 0.6	Category=ILL (illogical)

4.2 Activation Function Generator

When coded input patterns from input generator are presented to NFPR-Processor all output nodes become active to varying degrees. The output activation denoted by ACFj referred to as the activation function for the j^{th} output node. Where LIN is the learning input and LIW_j is the corresponding learning input weights.

$$ACFj = \frac{|LIN \wedge LI\,Wj|}{\alpha + |LI\,Wj|} \qquad (2)$$

Here α is kept as a small value close to 0 it's about 0.0000001.The node which registers the highest activation function is deemed Winner node

$$\text{i.e. } Winnernode = \max(ACFj) \qquad (3)$$

In the event of more than one node emerging as the winner owing to the same activation function value some mechanism such as choosing a node with the smallest index may be devised to break the tie.

4.3 Match Function Generator

The match function which helps to determine whether the network must adjust its learning parameters is given by equation 4 .The match function in association with the vigilance parameter decides on whether a particular output node is good enough to encode a given input pattern or whether a new output node should be opened to encode the same.

$$MAFj = \frac{|LIN \wedge LI\,Wj|}{|LIN|} \qquad (4)$$

The network is said to be in a state of resonance, if the match function value exceeds vigilance parameter. However, for a node

to exhibit resonance, it is essential that it not only encodes the given input pattern but should also represent the same category as that of the input pattern. The network is said to be in state of mismatch reset if the vigilance parameter exceeds match function, Such a state only means that the particular output node is not fit enough to learn the given input pattern and thereby cannot update its weights even though the category of the output node may be the same as that of the input pattern. This is so, since the output node has fallen short of the expected encoding granularity indicated by the vigilance parameter. If match function is greater than vigilance parameter and category of input pattern is not same with the learning input, the vigilance parameter is updated as below and the values of WFI and CFI is shown in table 2.

$$\rho = \text{MAF} + \delta \qquad (\delta = 0.001) \tag{5}$$

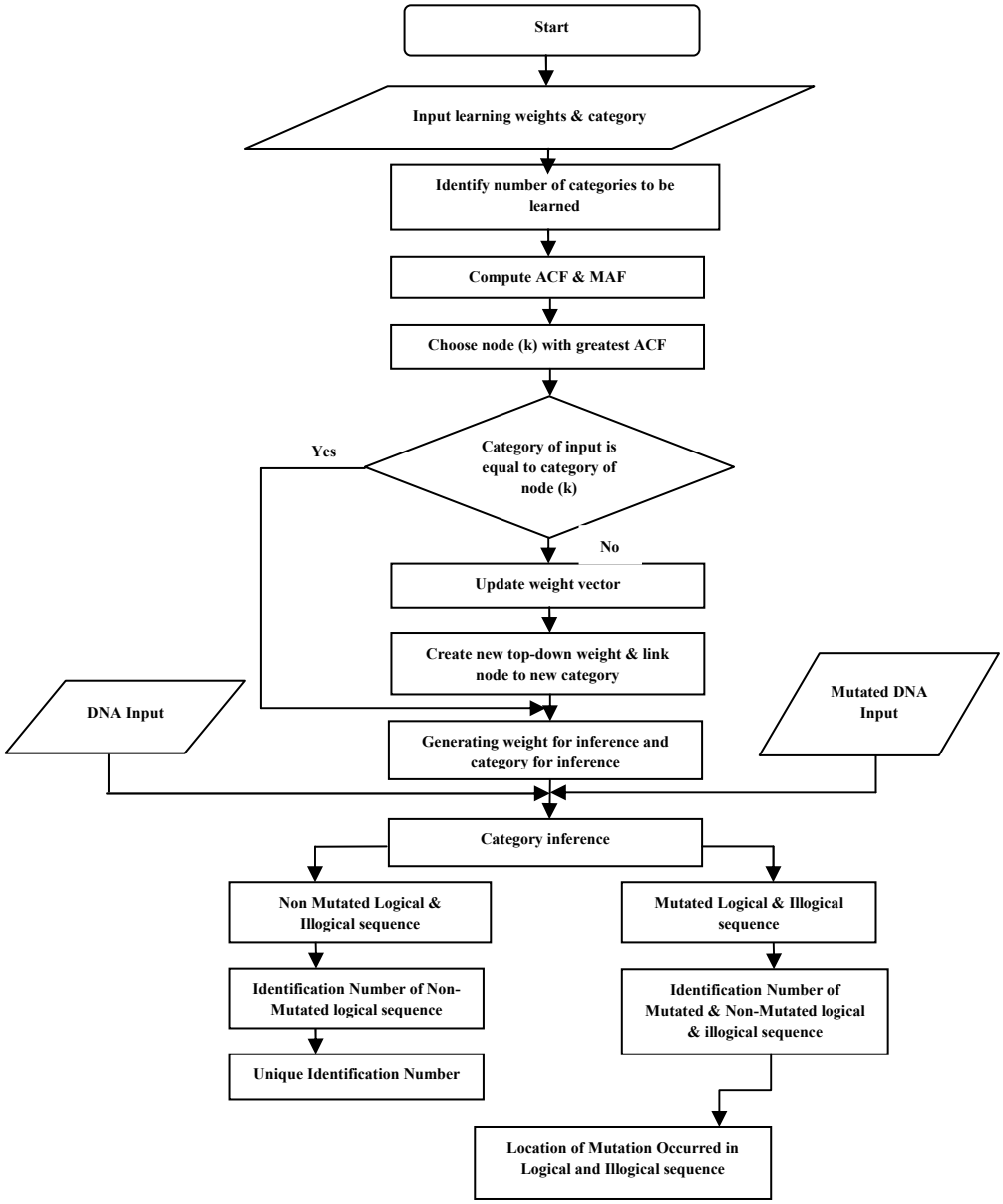


Figure 1 Flow chart of NFPR system

The weight updating equation of an output node j when it proceeds to learn the given input pattern *I* is given by

$$\text{WFI j}^{\text{new}} = \beta\,(\text{LIN} \wedge \text{WFIj}^{\text{old}}) + (1-\beta)\text{WFIj}^{\text{old}}$$
$$\text{where} \quad 0 \le \beta \le 1 \tag{6}$$

Once the network has been trained, the inference of patterns, logical or illogical i.e. the categories to which the patterns belong may be easily computed. This is accomplished by passing the input pattern into the preprocessor and then to the input layer. All the output nodes compute the activation functions with respect to the input. The winner, node with the highest activation function, is chosen. The category to which output node belongs is the one

to which given input pattern is classified by the network.

$$CIFj = \frac{\left| PPO \wedge WFIj \right|}{\left| WFIj \right|} \quad (7)$$

Table 2 Generating weights for inference, category for inference from learning inputs

Nucleotide Pair		A,A	A,U	T,A	T,T	T,U	G,A	G,G	G,U	C,A	C,C	C,U	U,A	U,U
Category		L	L	L	ILL	L	L	ILL	L	L	ILL	L	L	ILL
Fuzzy Equivalent		0.1,0.1*	0.1,0.5*	0.2,0.1*	0.2,0.8**	0.2,0.5*	0.3,0.1*	0.3,0.7**	0.3,0.5*	0.4,0.1*	0.4,0.6**	0.4,0.5*	0.5,0.1*	0.5,0.6***
Complement of Learning Input		0.9,0.9	0.9,0.5	0.8,0.9	0.8,0.2	0.8,0.5	0.7,0.9	0.7,0.3	0.7,0.5	0.6,0.9	0.6,0.4	0.6,0.5	0.5,0.9	0.5,0.4
Augmented Input / Learning Input(LI)		0.1,0.1, 0.9,0.9	0.1,0.5, 0.9,0.5	0.2,0.1, 0.8,0.9	0.2,0.8, 0.8,0.2	0.2,0.5, 0.8,0.5	0.3,0.1, 0.7,0.9	0.3,0.7, 0.7,0.3	0.3,0.5, 0.7,0.5	0.4,0.1, 0.6,0.9	0.4,0.6, 0.6,0.4	0.4,0.5, 0.6,0.5	0.5,0.1, 0.5,0.9	0.5,0.6, 0.5,0.4
δ		0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
ρ		0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5, 0.600+ δ	0.601	0.601	0.601
β		1	1	1	1	1	1	1	1	1	1	1	1	1
Activation Function	ACF(1)	0.9999	0.7999	0.9375	0.7999	0.9999	0.9333	0.8751	0.9999	0.9285	0.9230	0.9999	0.9230	0.8461
	ACF(2)	~	~	~	~	0.8499	0.5999	0.8999	0.8888	0.6111	0.8888	0.9375	0.6249	0.9375
	ACF(3)	~	~	~	~	~	~	~	~	~	~	~	~	0.7499
Highest Ignition Function		ACF(1)	ACF(1)	ACF(1)	ACF(2)	ACF(1)	ACF(1)	ACF(2)	ACF(1)	ACF(1)	ACF(2)	ACF(1)	ACF(1), ACF(2)	ACF(2)
Match Function	MAF(1)	1.0000	0.8000	0.7500	0.6000	0.7500	0.7000	0.6000	0.7000	0.6500	0.6000	0.6500	0.6000	0.5500
	MAF(2)	~	~	~	~	0.8500	0.6000	0.9000	0.8000	0.5500	0.8000	0.7500	0.5000	0.7500
	MAF(3)	~	~	~	~	~	~	~	~	~	~	~	~	0.7500
Category Match / Mismatch		Match	Match	Match		Match	Match		Match	Match	Mismatch	Match	Match	
					Match			Match			Match			Match
Learning Input Weights Updated/ Not Updated/ Added	LIW(1)	=LI	Updated	Updated	Not Updated	Updated	Updated	Not Updated	Updated	Updated	Not Updated	Updated	Not Updated (<ρ)	Not Updated
	LIW(2)	~	~	~	=LI	Not Updated	Not Updated	Updated	Not Updated	Not Updated	Updated	Not Updated	Not Updated (<ρ)	Updated
	LIW(3)	~	~	~	~	~	~	~	~	~	~	~	Added	Not Updated
Learning Input Weights & Category	LIW(1) L	0.1,0.1, 0.9,0.9	0.1,0.1, 0.9,0.5	0.1,0.1, 0.8,0.5	0.1,0.1, 0.8,0.5	0.1,0.1, 0.8,0.5	0.1,0.1, 0.7,0.5	0.1,0.1, 0.7,0.5	0.1,0.1, 0.7,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	WFI(1)=0.1,0.1, 0.6,0.5 CFI(1)=L
	LIW(2) ILL	~	~	~	0.2,0.8, 0.8,0.2	0.2,0.8, 0.8,0.2	0.2,0.8, 0.8,0.2	0.2,0.7, 0.7,0.2	0.2,0.7, 0.7,0.2	0.2,0.7, 0.7,0.2	0.2,0.6, 0.6,0.2	0.2,0.6, 0.6,0.2	0.2,0.6, 0.6,0.2	WFI(2)=0.2,0.6, 0.5,0.2 CFI(2)=ILL
	LIW(3) L	~	~	~	~	~	~	~	~	~	~	~	0.5,0.1, 0.5,0.9	WFI(3)=0.5,0.1, 0.5,0.9 CFI(3)=L

A-ADENINE, T-THYMINE, G-GUANINE, C-CYTOSINE,U-URACIL,ACF=ACTIVATION FUNCTION, MAF=MATCH FUNCTION, CFI= CATEGORY FOR INFERENCE, WFI= WEIGHT FOR INFERENCE , L=LOGICAL, ILL=ILLOGICAL ρ=VIGILANCE PARAMETER *- CASE1,**-CASE2,***CASE3,=LI-EQUAL TO LEARNING INPUT,<ρ=LESS THAN RHO

If CIF (1) or CIF (3) is greater than CIF (2) the inferred category is logical else if CIF (2) is greater than CIF (1) and CIF (3) then inferred category is illogical. For the DNA inputs of fasta format whose category is logical the corresponding seven consecutive nucleotide base in the DNA sample is chosen as single logical sequence and DNA inputs whose category is illogical, two consecutive nucleotide base is considered as an illogical sequence with base pair thirty two as shown in table4.

Logical sequence (LS):

$$LSp,s,k = Lseq\ p,s,1, Lseq\ p,s,2,..., Lseq\ p,s,k$$
$$\text{Where} \quad p,s = 1 \text{ to } \infty$$
$$\text{and} \quad k = 1 \text{ to } 7 \quad (8)$$

The separator outputs which are logical in their category are fed to the discriminator (D1) where identification number is computed using the equation below as shown in table 5.

$$D1_{ps} = \sum_{k=1}^{7} k(Lseq_{ps,k})^{k} \quad (9)$$

$$ps = 1 \text{ to} \infty$$

Illogical sequence (IS): $IS_{p,s} = ILseq_{s}, ILseq_{s},..., ILseq_{\infty}$ (10)

The separator outputs which are illogical in their category are fed to the discriminator (D2) where the identification number of illogical sequence is generated.

$$D2_{p,s} = ILseq_{s}^{m}$$
$$\text{where } p,s,m = 1 \text{to} \infty \quad (11)$$

m = Number of times nucleotide base is repeated

The comparator unit compares the identification numbers of all logical sequences of mutated and non-mutated DNA inputs from Discriminator (D1) and illogical sequences of mutated and non-

mutated sequences from (D2) to identify the location of mutation in the given sample.

DNA SAMPLE: *HUMAN-1* [BASE PAIR =32, SEQUENCE =25]

AATGTGTTGTGTGACCCCTCAAAATCTCTCAAATGTGTT
TTTACACTCCGTTGGTAATATGGAATGTGTTAAAGTTGC
TACCCGGGGTTTTTTAATGTGTCTCT
TGTGACCCCTCAAAATCTCTCAAATGTGTTTTTACACTC
CGTTGGTAATATGGAATGTGTTAAAGTTGCTACCCGGG
GTTTTTTAATGTGTCTCT

Table 3 Performance of proposed system for various numbers of epochs

Vigilance Parameter (ρ):0.5				
S. no	**Learning Vector (Number of Epochs)**	**Number of Learning Inputs**	**Learning Time (Seconds)**	**Accuracy%**
1	25	25	62.49	100%
2	13	13	34	100%

(No. of Epochs=25(For All Possible Combinations) and No. of Epochs=13)

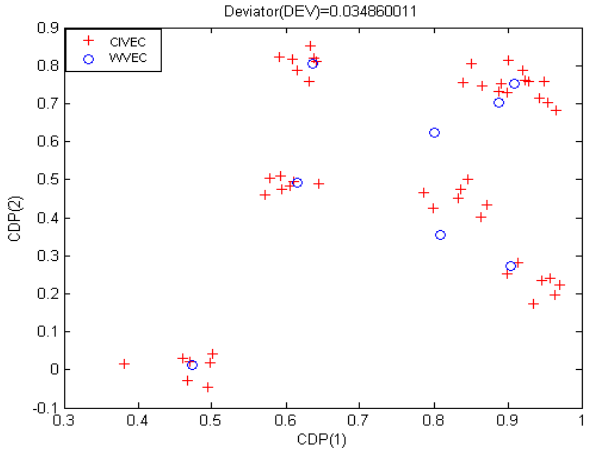
Table 4 Generating weights for inference, category inference function from learning Inputs

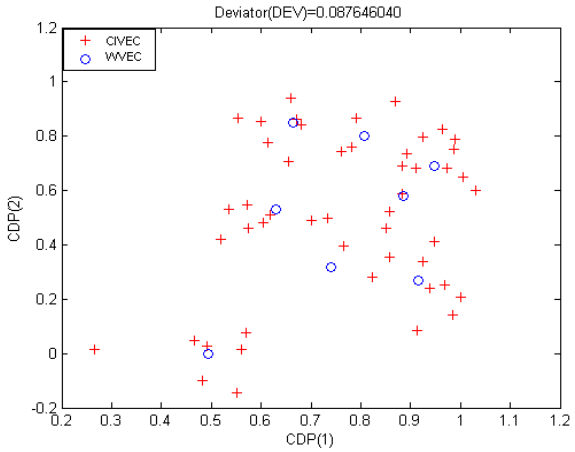
		DNA INPUTS OF *HUMAN-1*											
PPI (Preprocessor Input)		**A,A* 0.1,0.1**	**T,G* 0.2,0.3**	**C,C** 0.4,0.4**	**C,C** 0.4,0.4**	**C,C** 0.4,0.4**	**C,T* 0.4,0.2**	**T,C* 0.2,0.4**	**A,A* 0.1,0.1**	**T,T** 0.2,0.2**	**T,T** 0.2,0.2**	**T,T** 0.2,0.2**	**T,A* 0.2,0.1**
PPO (Preprocessor Output)		0.1,0.1, 0.9,0.9	0.2,0.3, 0.8,0.7	0.4,0.6, 0.6,0.4	0.4,0.6, 0.6,0.4	0.4,0.6, 0.6,0.4	0.4,0.2, 0.6,0.8	0.2,0.4, 0.8,0.6	0.1,0.1, 0.9,0.9	0.2,0.8, 0.8,0.2	0.2,0.8, 0.8,0.2	0.2,0.8, 0.8,0.2	0.2,0.1, 0.8,0.9
WFI	**WFI(1) / CFI(1)-L**	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5	0.1,0.1, 0.6,0.5
	WFI(2) / CFI(2)-ILL	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2	0.2,0.6, 0.5,0.2
	WFI(3) / CFI(3)-L	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9	0.5,0.1, 0.5,0.9
CIF	**CIF(1)**	1.0000	1.0000	0.9230	0.9230	0.9230	1.0000	1.0000	1.0000	0.7692	0.7692	0.7692	1.0000
	CIF(2)	0.6000	0.6000	1.0000	1	000	0.7333	0.8666	0.6000	1.0000	1.0000		0.6666
	CIF(3)	0.8000	0.7500	0.7000	0.7000	0.7000	0.9000	0.7000	0.8000	0.5000	0.5000	0.5000	0.8500
GIC		CIF(1)	CIF(1)	CIF(2)	CIF(2)	CIF(2)	CIF(1)	CIF(1)	CIF(1)	CIF(2)	CIF(2)	CIF(2)	CIF(1)
IC	**LOGICAL**	L	L				L	L	L				L
	ILLOGICAL			ILL	ILL	ILL				ILL	ILL	ILL	
Categorized Sequence		**0.1,0.1,0.2, 0.3,0.2,0.3, 0.2**	**0.2,0.3,0.2, 0.3,0.2,0.3, 0.1**	**0.4,0.4**	**0.4,0.4**	**0.4,0.4**	**0.4,0.2,0.4, 0.1,0.1,0.1, 0.1**	**0.2,0.4,0.2, 0.4,0.2,0.4, 0.1**	**0.1,0.1,0.2, 0.3,0.2,0.3, 0.2**	**0.2,0.2**	**0.2,0.2**	**0.2,0.2**	**0.2,0.1,0.4, 0.1,0.4,0.2, 0.4**

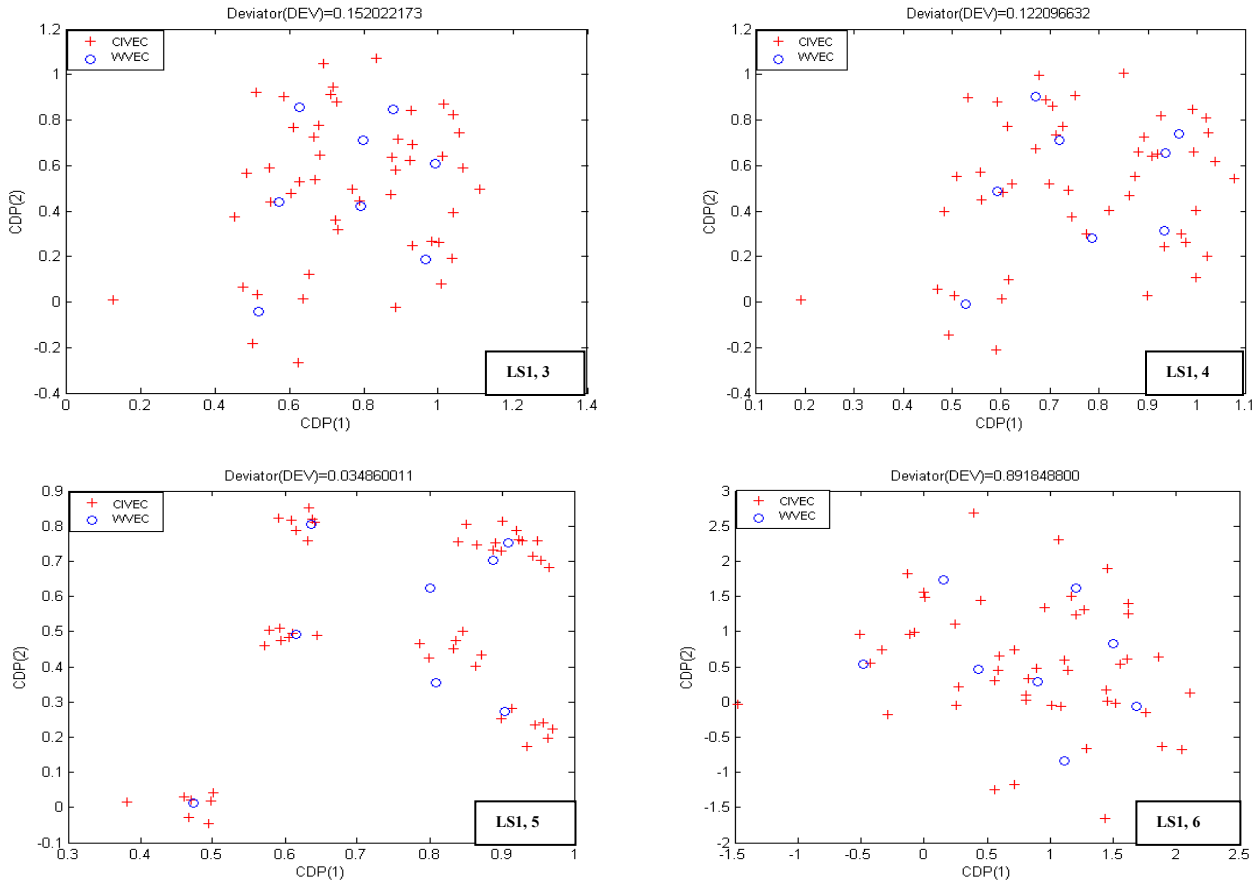
CIF-Category Inference Function,*- CONDTION 1,**-CONDITION 2,***-CONDITION 3,IC-Inferred Category

LS1, 1

LS1, 2







(CDP=Clustered Data Points, CIVEC=Cluster of Input Vectors,WVEC=Weight Vectors)
Figure 2 MATLAB Output for Logical sequence ($LS_{(1,1)}$-$LS_{(1,6)}$) Showing $LS_{(1,1)}$ and $LS_{(1,5)}$ are unique

Table 5 Discriminator (D1) outputs for non-mutated logical sequence of *human-1*

Logical Sequence	Human (p)	Sequence (s)	Discriminator (D1) Inputs ($LS_{p,s,k}$)	Logical Sequence							Identification Number ($D1_{p,s}$)	Unique Identification number (UIN_p)
				$Lseq_{p,s,k}$ (k=1)	$Lseq_{p,s,k}$ (k=2)	$Lseq_{p,s,k}$ (k=3)	$Lseq_{p,s,k}$ (k=4)	$Lseq_{p,s,k}$ (k=5)	$Lseq_{p,s,k}$ (k=6)	$Lseq_{p,s,k}$ (k=7)		
LS1	**1**	**1**	$LS_{1,1,k}$	**0.1**	**0.1**	**0.2**	**0.3**	**0.2**	**0.3**	**0.2**	**0.182464**	**0.182464 (REPEATED PATTERN)**
LS2	**1**	**2**	$LS_{1,2,k}$	**0.2**	**0.3**	**0.2**	**0.3**	**0.2**	**0.3**	**0.1**	0.442375	
LS3	**1**	**3**	$LS_{1,3,k}$	**0.4**	**0.2**	**0.4**	**0.1**	**0.1**	**0.1**	**0.1**	0.672457	
LS4	**1**	**4**	$LS_{1,4,k}$	**0.2**	**0.4**	**0.2**	**0.4**	**0.2**	**0.4**	**0.1**	0.671137	
LS5	**1**	**5**	$LS_{1,5,k}$	**0.1**	**0.1**	**0.2**	**0.3**	**0.2**	**0.3**	**0.2**	**0.182464**	
LS6	**1**	**6**	$LS_{1,6,k}$	**0.2**	**0.1**	**0.4**	**0.1**	**0.4**	**0.2**	**0.4**	0.475453	
LS7	**1**	**7**	$LS_{1,7,k}$	**0.4**	**0.3**	**0.2**	**0.2**	**0.3**	**0.3**	**0.2**	0.627014	
LS8	**1**	**8**	$LS_{1,8,k}$	**0.1**	**0.1**	**0.2**	**0.1**	**0.2**	**0.3**	**0.3**	0.150465	
LS9	**1**	**9**	$LS_{1,9,k}$	**0.1**	**0.1**	**0.2**	**0.3**	**0.2**	**0.3**	**0.2**	**0.182464**	
LS10	**1**	**10**	$LS_{1,10,k}$	**0.2**	**0.1**	**0.1**	**0.1**	**0.3**	**0.2**	**0.2**	0.239480	
LS11	**1**	**11**	$LS_{1,11,k}$	**0.3**	**0.4**	**0.2**	**0.1**	**0.4**	**0.4**	**0.4**	0.731645	
LS12	**1**	**12**	$LS_{1,12,k}$	**0.3**	**0.2**	**0.2**	**0.2**	**0.2**	**0.2**	**0.2**	0.411033	
LS13	**1**	**13**	$LS_{1,13,k}$	**0.1**	**0.1**	**0.2**	**0.3**	**0.2**	**0.3**	**0.2**	**0.182464**	

5. IDENTIFCIATION OF MUTATION IN THE SAMPLE

Mutation [21] is a change of DNA sequence within a gene or chromosome of an organism resulting in the creation of a new character or trait not found in the parental type [22]. The mutation results when a change occurs in a chromosome, either through an alteration in the nucleotide sequence of the DNA coding for a gene or through a change in the physical arrangement[23] [24] of a chromosome.

There are many different types of mutations; a point mutation (base pair substitution) is a simple change in one base of the gene sequence. In this case, the entire meaning of the sentence has been altered with a one letter change. In neutral or silent mutation, another one letter point mutation has occurred. However, the meaning of the sentence has not been altered. In a frame shift mutation, one or more bases are inserted or deleted into the sequence of the gene, the equivalent of adding or removing letters in a sentence, adding or removing one letter changes each subsequent word. This type of mutation can make the DNA meaningless and often results in shortened and functionless protein. Mutations that result in missing DNA are called deletions. These can be small, or longer deletions that affect a large number of genes on the chromosome. Deletions can also cause frame-shift mutations. Mutations that result in the addition of extra DNA are called insertions. Insertions can also cause frame-shift mutations, and generally result in a nonfunctional protein.

In an inversion mutation, an entire section of DNA is reversed. A small inversion may involve only a few bases within a gene, while longer inversions involve large regions of a chromosome containing several genes.

5.1 Various Types of Mutation identification in Human-1 sample

Before Mutation:

LS1/RS **LS2** **IS1** **IS1** **IS1**
AATGTGT TGTGTGA C C C
LS3 **LS4** **LS5/RS** **IS2** **IS2**
CTCAAAA TCTCTCA AATGTGT T T
IS2 **LS6** **LS7** **LS8**
T TACACTC CGTTGGT AATATGG
LS9/RS **LS10** **LS11** **IS3** **IS3**
AATGTGT TAAAGTT GCTACCC G G
IS3 **LS12** **LS13/RS** **LS14**
G GTTTTTT AATGTGT CTCTXXX

Case 1:-

After Point Mutation in the sample:

LS1/RS **LS2** **IS1** **IS1** **IS1**
AATGTGT TGTGTGA C C C
LS3 **LS4** **LS5/RS** **IS2** **IS2**
CTCA **C** AA TCTCTCA AATGTGT T T
IS2 **LS6** **LS7** **LS8**
T TACACTC CGTTGGT AATATGG
LS9/RS **LS10** **LS11** **IS3**
AATGTGT TAAAGTT GCTACCC G
IS3 **IS3** **LS12** **LS13/RS** **LS14**
G G GTTTTTT AATGTGT CTCTXXX

In case 1 the point mutation occurred in logical sequence ($LS_{1,3}$) by the mutant C that can be identified with the change in identification number of $LS_{1,3}$ where identification number of illogical sequence remains unaltered as in table 6 below.

Table 6 point mutation

Logical Sequence (LS)	Identification Number (Before Mutation)	Identification Number (After Mutation)
$LS_{1,1}$	0.182464	0.182464
$LS_{1,2}$	0.442375	0.442375
$LS_{1,3}$	0.672457	**0.723607**
$LS_{1,4}$	0.672577	0.672577
$LS_{1,5}$	0.182464	0.182464
$LS_{1,6}$	0.475453	0.475453
$LS_{1,7}$	0.627014	0.627014
$LS_{1,8}$	0.151905	0.151905
$LS_{1,9}$	0.182464	0.182464
$LS_{1,10}$	0.236024	0.236024
$LS_{1,11}$	0.731645	0.731645
$LS_{1,12}$	0.412474	0.412474
$LS_{1,13}$	0.182464	0.182464
Illogical Sequence (IS)	**Identification Number (Before Mutation)**	**Identification Number (After Mutation)**
$IS_{1,1}$	0.064000	0.064000
$IS_{1,2}$	0.008000	0.008000
$IS_{1,3}$	0.027000	0.027000

[Result: Change in polypeptide sequence might change the shape or function of the protein, depending on where in the sequence occurs]

Case 2:-

After Frame shift mutation [Insertion] in the sample:

LS1/RS **LS2** **IS1** **IS1** **IS1**
AATGTGT TGTGTGA C C C
LS3 **LS4** **LS5/RS** **IS2** **IS2**
CTCAAAA TCTCTCA AATGTGT T T
IS2 **LS6** **LS7** **LS8**
T TACACTC CGTTGGT AATATGG
LS9/RS **LS10** **LS11** **LS12**
AATGTGT TAAAGTT GCTACCC G **C** GGGTT
IS3 **IS3** **IS3** **LS13** **LS14**
T T T TAATGTG TCTCTXX

In case 2 the frame shift mutation (insertion) occurred in one of the $IS_{1,3}$ by the mutant C which alters both the logical sequence ($LS_{1,12}$) and illogical sequence ($IS_{1,3}$) that can be identified by the change in identification number of both logical sequence ($LS_{1,12}$) and illogical sequence ($IS_{1,3}$) as in table 7.

[Result: Change in polypeptide sequence might change the shape or function of the protein, depending on where in the sequence occurs.**]**

Case 3:-

After Point mutation [Neutral or Silent] in the sample:

LS1/RS **LS2** **IS1** **IS1** **IS1**
AATGTGT TGTGTGA C C C
LS3 **LS4** **LS5/RS** **IS2** **IS2**
CTCAAAA TCTCTCA AATGTGT T T
IS2 **LS6** **LS7** **LS8**
T TACACTC CGTTGGT AATATGG
LS9/RS **LS10** **LS11** **IS3**
AATGTGT TAAAGTT GCTACCC G
IS3 **IS3** **IS3** **LS12** **LS13/RS**
G G G GTTTTTT AATGTGT
LS14
CTCTXXX

Table 7 frame shift mutation [insertion]

Logical Sequence (LS)	Identification Number (Before Mutation)	Identification Number (After Mutation)
$LS_{1,1}$	0.182464	0.182464
$LS_{1,2}$	0.442375	0.442375
$LS_{1,3}$	0.672457	0.672457
$LS_{1,4}$	0.672577	0.672577
$LS_{1,5}$	0.182464	0.182464
$LS_{1,6}$	0.475453	0.475453
$LS_{1,7}$	0.627014	0.627014
$LS_{1,8}$	0.151905	0.151905
$LS_{1,9}$	0.182464	0.182464
$LS_{1,10}$	0.236024	0.236024
$LS_{1,11}$	0.731645	0.731645
$LS_{1,12}$	0.412474	**0.745974**
$LS_{1,13}$	0.182464	**0.243464**
Illogical Sequence (IS)	**Identification Number (Before Mutation)**	**Identification Number (After Mutation)**
$IS_{1,1}$	0.064000	0.064000
$IS_{1,2}$	0.008000	0.008000
$IS_{1,3}$	0.027000	**0.008000**

[Result: No change in polypeptide sequence, possible consequence for the organism =none**].**
In case 3 the point mutation is occurred in same $IS_{1,3}$ as case 2 but with mutant G that only alters the illogical sequence ($IS_{1,3}$) and not any of the Logical sequences that can be identified only using the change in identification number of illogical sequence ($IS_{1,3}$) table 8.

Table 8 point mutation [neutral or silent]

Logical Sequence (LS)	Identification Number (Before Mutation)	Identification Number (After Mutation)
$LS_{1,1}$	0.182464	0.182464
$LS_{1,2}$	0.442375	0.442375
$LS_{1,3}$	0.672457	0.672457
$LS_{1,4}$	0.672577	0.672577
$LS_{1,5}$	0.182464	0.182464
$LS_{1,6}$	0.475453	0.475453
$LS_{1,7}$	0.627014	0.627014
$LS_{1,8}$	0.151905	0.151905
$LS_{1,9}$	0.182464	0.182464
$LS_{1,10}$	0.236024	0.236024
$LS_{1,11}$	0.731645	0.731645
$LS_{1,12}$	0.412474	0.412474
$LS_{1,13}$	0.182464	0.182464
Illogical Sequence (IS)	**Identification Number (Before Mutation)**	**Identification Number (After Mutation)**
$IS_{1,1}$	0.064000	0.064000
$IS_{1,2}$	0.008000	0.008000
$IS_{1,3}$	0.027000	**0.008100**

Case 4:-
After Frame shift mutation in the sample:

LS1/RS **LS2** **IS1** **IS1** **IS1**
AATGTGT TGTGTGA C C C
LS3 **LS4** **LS5/RS** **IS2** **IS2**
CTCAAAA TCTCTCA AATGTGT T T
IS2 **LS6** **LS7** **LS8**
T TACACTC CGTTGGT AATATGG
LS9/RS **LS10** **LS11** **IS3** **IS3**
AATGTGT TAAAGTT GCTACCC G G
IS3 **LS12** **LS13/RS** **LS14**
G **GTTT TTA** ATGTGTC TCTXXX

In case 4 the frame mutation [deletion] occurred in logical sequence ($LS_{1,\ 12}$) by the removal of mutant T and can be identified with the change in identification number of logical sequence ($LS_{1,\ 12}$) with no alteration in any of the illogical sequence as in table 9 .

Table 9 frame shift mutation [deletion]

Logical Sequence (LS)	Identification Number (Before Mutation)	Identification Number (After Mutation)
$LS_{1,1}$	0.182464	0.182464
$LS_{1,2}$	0.442375	0.442375
$LS_{1,3}$	0.672457	0.672457
$LS_{1,4}$	0.672577	0.672577
$LS_{1,5}$	0.182464	0.182464
$LS_{1,6}$	0.475453	0.475453
$LS_{1,7}$	0.627014	0.627014
$LS_{1,8}$	0.151905	0.151905
$LS_{1,9}$	0.182464	0.182464
$LS_{1,10}$	0.236024	0.236024
$LS_{1,11}$	0.731645	0.731645
$LS_{1,12}$	0.412474	**0.410945**
$LS_{1,13}$	0.182464	**0.291402**
Illogical Sequence (IS)	**Identification Number (Before Mutation)**	**Identification Number (After Mutation)**
$IS_{1,1}$	0.064000	0.064000
$IS_{1,2}$	0.008000	0.008000
$IS_{1,3}$	0.027000	0.027000

[Result: Change in polypeptide sequence might change the shape or function of the protein, depending on where in the sequence occurs.**]**

Case 5:-
In case 5, the inversion mutation occurred in logical sequence ($LS_{1,\ 10}$) by replacing TAAAGTT with mutant TTGAAAT that can be identified with the change in identification number of logical sequence ($LS_{1,\ 10}$) alone with no alteration in any of the illogical sequence as in table 10 below.
After Inversion mutation in the sample:

LS1/RS **LS2** **IS1** **IS1** **IS1**
AATGTGT TGTGTGA C C C
LS3 **LS4** **LS5/RS** **IS2** **IS2**
CTCAAAA TCTCTCA AATGTGT T T
IS2 **LS6** **LS7** **LS8**
T TACACTC CGTTGGT AATATGG
LS9/RS **LS10** **LS11** **IS3**
AATGTGT **TTGAAAT** GCTACCC G
IS3 IS3 **LS12** **LS13/RS** **LS14**
G G GTTTTTT AATGTGT CTCTXXX

Table 10 inversion mutation

Logical Sequence (LS)	Identification Number (Before Mutation)	Identification Number (After Mutation)
$LS_{1,1}$	0.182464	0.182464
$LS_{1,2}$	0.442375	0.442375
$LS_{1,3}$	0.672457	0.672457
$LS_{1,4}$	0.672577	0.672577
$LS_{1,5}$	0.182464	0.182464
$LS_{1,6}$	0.475453	0.475453
$LS_{1,7}$	0.627014	0.627014
$LS_{1,8}$	0.151905	0.151905
$LS_{1,9}$	0.182464	0.182464
$LS_{1,10}$	0.236024	**0.361546**
$LS_{1,11}$	0.731645	0.731645
$LS_{1,12}$	0.412474	0.412474
$LS_{1,13}$	0.182464	0.182464
Illogical Sequence (IS)	**Identification Number (Before Mutation)**	**Identification Number (After Mutation)**
$IS_{1,1}$	0.064000	0.064000
$IS_{1,2}$	0.008000	0.008000
$IS_{1,3}$	0.027000	0.027000

6. CONCLUSION

As an attempt to automate the genetic finger printing the Neural-fuzzy Pattern Recognition System discussed in the above work assists forensic scientists by generating unique identification number for individuals from their DNA sample. The proposed system also helps to identify the location of occurrence mutation in the given mutated DNA sample, for instance, gene mutations which triggers HNPCC tumor that could not be detected even by PCR-SSCP can be easily detected by subjecting the sample to gene sequencing process and analyzed using above system.

Further development can be extended by training patterns in DNA protein that can be represented by suitable fuzzy equivalent in order to classify and predict the protein structure in the protein folding problem. The above technique can be used in the areas where feature extraction is to be done in genetic engineering with suitable modification.

7. ACKNOWLEDGEMENT

We like to thank Senior Scientist, Dr.K.Thangarasu of Bio-Rad Laborataries, Banaglore, India and Dr.K.Somasundaram of Amirtha University, Coimbatore, India for their assistance in conducting the experiment.

8. REFERENCES

[1] Richard O. Duda, Peter E.Hart, David G. Stork, "Pattern classification"-Second Edition", John Wiley and sons, 2006.

[2] Robert Schalkoff, "Pattern Recognition: Statistical, Structural and Neural Approaches, 2007, John Wiley and sons.

[3] Donald R. Tveter. "The Pattern Recognition Basis of Artificial Intelligence", IEEE Press, New York, page 117.

[4] John Hertz, Anders Krogh, and Richard G. Palmer. "Introduction to the Theory of Neural Computation", Addison Wesley, Redwood City, A, 2008.

[5] Stephen J. Hanson, Jack D. Cowan, and C. Lee Giles, "Advances in Neural Information Processing Systems", volume 5, Morgan Kaufmann San Mateo CA, 2009.

[6] "Advances in Neural Networks issn-2006", Third international symposium on neural networks, Springer Berlin Heidelberg, New York publications.

[7] Carpenter, G.A. and S. Grossberg, "A Massively Parallel Architecture for a self-organizing Neural Pattern Recognition Machine", Computer Vision, Graphics and Image Processing, 37, PP. 54-115.

[8] Carpenter, G.A. and S. Grossberg, and J.H. Reynolds (2010), "ARTMAP: Supervised Real Time Learning and Classification of Non-stationary Data by a Self-organizing Neural Network". Vol. 4, pp. 565-588.

[9] Stephen Krawetz, David D.Womble , "Introduction to Bioinformatics A Theoretical and Practical Approach", Human Press Inc.,

[10] David W.Mount, David W. Mount, "Bio informatics Sequence and Genome analysis"- Second Edition, Cold Spring Harbor Laboratory Press, New York.

[11] Des Higgins, willie Taylor, "Bioinformatics Sequence, Structure and data banks", Oxford University Press, 2000.

[12] "Bioinformatics for geneticists", Michael R.Barnes , Second Edition, John Wiley & Sons Ltd.

[13] Andreas D. Buxevanis, "Bioinformatics-A practical Guide to the Analysis of genes and proteins", second edition, A John wiley & sons, Inc., Publication.

[14] Norah Rudin, Keith Inman, "An Introduction forensic DNA Analysis", 2002-CRC Press.

[15] .Computational Intelligence and Bio inspired Systems, 8th international work conference on artificial neural networks, iwann-2005proceedings.

[16] Julie A. Ayala-Gross, "DNA Analysis: The best method for Human Identifications", National University, San Diego – 2001.

[17] Joe Nickell and John F.Fischar, "Crime Science Methods of Forensic Detection", 1999. University Press of Kentucky.

[18] John O. Savino, Brent E Turvey , "Rape Investigation Hand book", 2005, Elsevier Inc.,

[19] David E. Newton, "DNA Evidence and Forensic science"- 2008 facts on file, Inc. http://www.factsonfile.com.

[20] Jorg T. Epplen Thomas Lubjuhn, Birkhauser, "DNA Profiling and DNA Finger Printing", Verlag Publication.

[21] Charles L.Valon, "New developments in Mutation Research", Nova science publishers Inc New York, 2007.

[22] "Oxidative Damage to Nucleic Acids", Springer science press, New York.

[23] Richard G. H. Cotton, Edward Edkins, Sue Forrest "Mutation detection", IRL Press at Oxford University Press.

[24] Graham R. Taylor "Laboratory methods for the detection of mutations and polymorphisms in DNA", CRC Press, 2007 - Science.

[25] Phipps Arabie, Lawrence J. Hubert, and Geert De Soete, editors, "Clustering and Classification", World Scientific, River Edge, NJ.

[26] S.N Sivanandam, "Introduction to neural networks and MATLAB-6.0", Tata McGraw-Hill publishing company, 2006.