CHAPTER - 4

HYBRID MODEL BASED ON EVOLUTIONARY COMPUTING, FUZZY LOGIC AND NEURAL NETWORKS

4.0 INTRODUCTION

Data mining consists of the techniques and tools that are utilized for extracting knowledge from the database. The database must be efficiently utilized for building a decision making model with maximum prediction accuracy. This is an era of information and huge data can be captured and stored at a relatively low cost. Such data bases are largely available in almost all fields including the field of medicine. Retrieving knowledge from the data base has increased enormously. Such data bases become valuable databases on the basis of the ability of extracting information to build a decision making model. Extracting information implies understanding the rules governing the data. Handling such huge data bases manually becomes an impossible task as the size of the data and the dimension of the data are very large and handling such data bases goes beyond human capacity (Chen and Hsu, 2006). Therefore for dealing with such data bases, the computing technology for automating the process is desired and hence more such efficient techniques are to be developed.

Techniques such as Artificial Neural Network and Fuzzy Logic when fused with evolutionary computing techniques have shown more capability while addressing highly complex real life problems. Genetic Algorithms (GAs) are the paradigm with best theoretical basis of the whole evolutionary computation. GAs are the stochastic search algorithms based on the principles of natural selection and genetics. They represent a powerful tool to solve complex optimization problems where the traditional methods cannot find an optimum solution. They provide an efficient, adaptive and robust search in optimization process that are usually applied to very large complex and multi nodal search space (Dehauri, 2010).
This chapter mainly focuses on the application of the hybrid systems to medical data bases. The hybrid systems namely Genetic neuro system, Neuro Fuzzy system and Genetic Neuro Fuzzy systems have been applied to a medical data set and the results have been discussed in detail to show the supremacy of the hybrid systems over the other techniques.

4.1 DIMENSIONALITY REDUCTION

Dimensionality Reduction or selecting a good subset of features is of great importance in applying Machine Learning (ML) techniques for classification of data, since a high dimensional feature set reduces the classification accuracy. Some of the features may be redundant and non-informative in a data base and hence different combinatorial set of features should be formed so that the best classification accuracy may be obtained. Feature selection process plays a major role in Machine Learning techniques while constructing a Machine Learning Model. Feature subsets can be obtained through lot of techniques such as Principal Components Analysis (PCA), Particle Swarm Optimization (PSO), Genetic Algorithm (Saqib et al 2014).

A Feature Subset Selection (FSS) is an operator $F_S$ or a map from ‘$m$’ dimensional feature space (input space) to ‘$n$’ dimensional feature space (output). Mathematically

$$F_S : R^{rxm} \rightarrow R^{rxn}$$  \hspace{1cm} (4.1)

where $m>n$. $R^{rxm}$ is any database matrix containing the original feature set having ‘$r$’ observations and $R^{rxn}$ is the reduced feature set containing ‘$r$’ observations in the subset selection. (Oluleya et al, 2014)

4.1.1 Principal Component Analysis

Principal Component Analysis is a well-known statistical technique. It is widely used in engineering and scientific disciplines, such as pattern recognition, data compression and coding, image processing, high-resolution spectrum analysis,
and adaptive beam forming. The PCA is based on the spectral analysis of the second-order moment matrix that statistically characterizes a random vector. In the zero mean case, this matrix, called the correlation matrix, becomes the covariance matrix (Ke-Lin Du, M. N. S. Swamy 2006). In the area of image coding, the PCA is known by the name of the Karhunen-Loeve transform (KLT), which is an optimal scheme for data compression based on the exploitation of correlation between neighbouring pixels or groups of pixels. The PCA is directly related to Singular Value Decomposition (SVD), and the most common way to perform the PCA is via the SVD of the data matrix (J Qiu 2012). However, the capability of the SVD is limited for very large data sets. Pre-processing usually maps a high-dimensional space to a low-dimensional space with minimum information loss. The process is known as feature extraction. The PCA is a well-known feature-extraction method (Ke-Lin Du, M.N.S. Swamy 2006). The PCA allows the removal of the second-order correlation among given random processes. By calculating the eigenvectors of the covariance matrix of the input vector, the PCA linearly transforms a high-dimensional input vector into a low-dimensional one whose components are uncorrelated.

The PCA is often based on optimization of some information criterion, such as the maximization of the variance of the projected data or the minimization of the reconstruction error. The aim of the PCA is to extract m orthonormal directions $\mathbf{w}_i \in \mathbb{R}^n$, $i = 1, 2, \ldots, m$, in the input space that account for as much of the data’s variance as possible. Subsequently, an input vector $\mathbf{x} \in \mathbb{R}^n$ may be transformed into lower m-dimensional space without losing essential intrinsic information. The vector $\mathbf{x}$ can be represented by being projected onto the m-dimensional subspace spanned by $\mathbf{w}_i$ using the inner products $\mathbf{x}^T \mathbf{w}_i$. This achieves dimensionality reduction.

The PCA finds those unit directions $\mathbf{w} \in \mathbb{R}^n$ along which the projections of the input vectors, known as the principal components (PCs), $\mathbf{y} = \mathbf{x}^T \mathbf{w}$, have the largest variance

$$E_{PCA}(\mathbf{w}) = E[\mathbf{y}^2] = \mathbf{w}^T \mathbf{C} \mathbf{w} = \frac{\mathbf{w}^T \mathbf{C} \mathbf{w}}{||\mathbf{w}||^2}$$ (4.2)
where \( \overline{w} = \frac{w}{\|w\|} \). \( E_{PCA}(w) \) is a positive-semi-definite function. Setting \( \frac{\partial E_{PCA}}{\partial w} = 0 \), we get

\[
C_w = \frac{w^T C w}{\|w\|^2} w
\]

(4.3)

It can be verified that the solutions to (4.3) are \( w = \alpha c_i, i = 1, 2, \ldots, n \), where \( \alpha \in \mathbb{R} \). When \( \alpha = 1 \), \( w \) becomes a unit vector.

We now examine the positive-definiteness of the Hessian of \( E_{PCA}(w) \) at \( w = c_i \). Multiplying the Hessian by \( c_j \) leads to

\[
H(c_i)c_j = \begin{cases} 
0 & i = j \\
(\lambda_i - \lambda_j)c_j & i \neq j 
\end{cases}
\]

(4.4)

Thus, \( H(w) \) has the same eigenvectors as \( C \) but with different eigenvalues. \( H(w) \) is positive-semi-definite only when \( w = c_1 \). As a result, \( w \) will eventually point in the direction of \( c_1 \) and \( E_{PCA}(w) \) takes its maximum value.

By repeating maximization of \( E_{PCA}(w) \) but limiting \( w \) orthogonal to \( c_1 \), the maximum of \( E_{PCA}(w) \) is equal to \( \lambda_2 \) at \( w = \alpha c_2 \). Following this deflation procedure, all the \( m \) principal directions \( \overline{w}_i \) can be derived. The projection \( y_i = x^T \overline{w}_i, i = 1, 2, \ldots \), are the PCs of \( x \). This linear dimensionality reduction procedure is the KLT, and the result for two-dimensional input data is illustrated in Figure 4.1.

![Figure 4.1 Illustration of PCA in two dimensions.](image-url)
Each data point is accurately characterized by its projections on the two principal directions \( \mathbf{w}_1 \) and \( \mathbf{w}_2 \),

where \( \mathbf{w}_1 = \mathbf{w}_1/\mathbf{w}_1 \) and \( \mathbf{w}_2 = \mathbf{w}_2/\mathbf{w}_2 \). If the data is compressed to the one dimensional space, each data point is then represented by its projection on the eigen vector \( \mathbf{w}_1 \).

A linear LS estimate \( \hat{x} \) can be constructed for the original input \( x \)

\[
\hat{x} = \sum_{i=1}^{m} y_i \mathbf{w}_i
\]  

(4.5)

The process can be treated as data reconstruction. The reconstruction error \( e \) is the difference between the original and reconstructed data

\[
e = x - \hat{x} = \sum_{i=m+1}^{n} y_i \mathbf{w}_i
\]  

(4.6)

Naturally, \( e \) is orthogonal to \( \hat{x} \). Each principal component \( y_i \) is a Gaussian with zero mean and variance \( \sigma_i^2 = \lambda_i \). The variances of \( x \), \( \hat{x} \), and \( e \) can be, respectively, expressed as

\[
E[||x||^2] = \sum_{i=1}^{n} \sigma_i^2 = \sum_{i=1}^{m} \lambda_i
\]  

(4.7)

\[
E[||\hat{x}||^2] = \sum_{i=1}^{m} \sigma_i^2 = \sum_{i=1}^{m} \lambda_i
\]  

(4.8)

\[
E[||e||^2] = \sum_{i=m+1}^{n} \sigma_i^2 = \sum_{i=m+1}^{n} \lambda_i
\]  

(4.9)

When we use only the first \( m_1 \) among the extracted \( m \) PCs to represent the raw data, we need to evaluate the error by replacing \( m \) by \( m_1 \) (Du and Swami 2006).

4.2 EVOLUTIONARY COMPUTATION TECHNIQUES

Evolutionary Algorithms are a class of stochastic search and optimization techniques guided by natural selection and genetics. They are population based algorithms by simulating the natural evolution of biological systems. (Zitzler, 1999). For an optimization problem in a domain, if the calculus is difficult to implement or is inapplicable, search methods such as EAs can be used. Individuals in a population
compete and exchange information with one another. There are three basic genetic operations, namely, crossover1, mutation, and selection. The procedure of a typical Evolutionary Algorithm is given in figure 4.2.

1. Set $t = 0$.
2. Randomize initial population $P(0)$.
3. Repeat:
   a) Evaluate fitness of each individual of $P(t)$.
   b) Select individuals as parents from $P(t)$ based on fitness.
   c) Apply search operators (crossover and mutation) to parents, and generate $P(t+1)$.
   d) Set $t = t + 1$.
until the termination criterion is satisfied.

**Figure 4.2 Evolutionary Algorithm**

EAs are stochastic processes performing searches over a complex and multimode space. They have the following advantages:

EAs can solve hard problems reliably and fast. They are suitable for evaluation functions that are large, complex, non-continuous, non-differentiable, and multimodal. The EA approach is a general-purpose one, that can be directly interfaced to existing simulations and models and EAs are extendable and easy to hybridize. EAs are directed stochastic global search. They can always reach the near optimum or the global maximum and possess inherent parallelism by evaluating multi points simultaneously. EAs employ a structured, yet randomized, parallel multipoint search strategy that is biased toward reinforcing search points of high fitness. The evaluation function must be calculated for all the individuals of the population, thus resulting in a computation load that is much higher than that of a simple random search or a gradient search.
4.2.1 Terminologies:

Some terminologies that are used in the EA literature are listed below. These terminologies are an analogy to their biological counterparts.

**Population:** A set of individuals in a generation is called a population, \( P(t) = \{x_1, x_2, \ldots, x_{NP}\} \), where \( x_i \) is the \( i^{th} \) individual, and \( NP \) is the size of the population. The initial population is usually generated randomly, while the population of other generations are generated from some selection / reproduction procedure.

**Chromosome:** Each individual \( x_i \) in a population is a single chromosome. A chromosome, sometimes called a genome, is a set of parameters that define a solution to the problem under consideration. The chromosome is often represented as a string in EAs. Biologically, a chromosome is a long, continuous piece of DNA that contains many genes, regulatory elements and other intervening nucleotide sequences. Normal members of a particular species all have the same number of chromosomes. For example, human body cells contain 46 diploid chromosomes, that is, they have two set of chromosomes, one set of 46 chromosomes from the mother and the other set of 46 chromosomes from the father. Chromosomes are used to encode a biological organism.

**Gene:** In EAs, each chromosome \( x \) comprises of a string of elements \( x_i \), called genes, i.e. \( x = [x_1, x_2, \ldots, x_n] \), where \( n \) is the number of genes in the chromosome. Each gene encodes a parameter of the problem into the chromosome. A gene is usually encoded as a binary string or a real number. In biology, genes are entities that parents pass to offspring during reproduction. These entities encode information essential for the construction and regulation of proteins and other molecules that determine the growth and functioning of the organism.

**Allele:** The biological definition for an allele is any one of a number of alternative forms of the same gene occupying a given position called a locus on a chromosome. In the EA terminology, the value of a gene is indicated as an allele.
**Genotype**: A genotype is biologically referred to the underlying genetic coding of a living organism, usually in the form of DNA. The genotype of each organism corresponds to an observable, known as a phenotype. In EAs, a genotype represents a coded solution, that is, an individual’s chromosome.

**Phenotype**: Biologically, the phenotype of an organism is either its total physical appearance and constitution or a specific manifestation of a trait. A phenotype is determined by genotype or multiple genes and influenced by environmental factors. The concept of phenotypic plasticity describes the degree to which an organism’s phenotype is determined by its genotype. A high level of plasticity means that environmental factors have a strong influence on the particular phenotype that develops. The ability to learn is the most obvious example of phenotypic plasticity. As another example of phenotypic plasticity, sports can strengthen muscles. However, some organs have very low phenotypic plasticity; for example, the color of human eyes cannot be changed by environment. The mapping of a set of genotypes to a set of phenotypes is referred to as a genotype–phenotype map. In EAs, a phenotype represents a decoded solution.

**Fitness**: Fitness in biology refers to the ability of an individual of certain genotype to reproduce. The set of all possible genotypes and their respective fitness values is called a fitness landscape. Fitness function is a particular type of objective function that quantifies the optimality of a solution, i.e. a chromosome, in an EA. It is used to map an individual’s chromosome into a positive number. Fitness is the value of the objective function for a chromosome $x_i$, namely $f(x_i)$. After the genotype is decoded, the fitness function is used to convert the phenotype’s parameter values into the fitness. Fitness is used to rate the solutions.

**Natural Selection**: Natural selection is believed to be the most important mechanism in the evolution of biological species. It alters biological populations over time by propagating heritable traits affecting individual organisms to survive and reproduce. It adapts a species to its environment. Natural selection does not distinguish between its two forms, namely, ecological selection and sexual selection, but it is concerned with those traits that help individuals to survive the environment and to reproduce. Natural selection causes traits to become more prevalent when they contribute to fitness.
Natural selection is different from artificial selection. Genetic drift and gene flow are two other mechanisms in biological evolution. Genetic flow, also known as genetic migration, is the migration of genes from one population to another.

**Genetic Drift**: Genetic drift is a contributing mechanism in biological evolution. As opposed to natural selection, genetic drift is a stochastic process that arises from random sampling in the reproduction. It changes allele frequencies (gene variations) in a population over many generations and affects traits that are more neutral. The genes of a new generation are a sampling from the genes of the successful individuals of the previous one, but with some statistical error.

Drift is the cumulative effect over time of this sampling error on the allele frequencies in the population, and traits that do not affect reproductive fitness change in a population over time. Like selection, genetic drift acts on populations, altering allele frequencies and the predominance of traits. It occurs most rapidly in small populations and can lead some alleles to become extinct or become the only alleles in the population, thus reducing the genetic diversity in the population.

**Termination Criterion**: The search process of an EA will terminate when a certain termination criterion is met. Otherwise a new generation will be produced and the search process continues. The termination criterion can be selected as a maximum number of generations, or the convergence of the genotypes of the individuals. Convergence of the genotypes occurs when all the bits or values in the same positions of all the strings are identical, and crossover has no effect for further processes. Phenotypic convergence without genotypic convergence is also possible. For a given system, the objective values are required to be mapped into fitness values so that the domain of the fitness function is always greater than zero (Du and Swami, 2006).

### 4.3 GENETIC PROGRAMMING

In artificial intelligence, genetic programming (GP) is an evolutionary algorithm-based methodology inspired by biological evolution to find computer programs that perform a user-defined task. Essentially GP is a set of instructions and
a fitness function to measure how well a computer has performed a task. It is a specialization of genetic algorithms (GA) where each individual is a computer program. It is a machine learning technique used to optimize a population of computer programs according to a fitness landscape determined by a program's ability to perform a given computational task.

GP evolves computer programs, traditionally represented in memory as tree structures.[5] Trees can be easily evaluated in a recursive manner. Every tree node has an operator function and every terminal node has an operand, making mathematical expressions easy to evolve and evaluate. Thus traditionally GP favors the use of programming languages that naturally embody tree structures.

**Genetic Operators**: The main operators used in evolutionary algorithms such as GP are crossover and mutation.

**Crossover**: Crossover is applied on an individual by simply switching one of its nodes with another node from another individual in the population. With a tree-based representation, replacing a node means replacing the whole branch. This adds greater effectiveness to the crossover operator. The expressions resulting from crossover are very different from their initial parents.

**Mutation**: Mutation affects an individual in the population. It can replace a whole node in the selected individual, or it can replace just the node's information. To maintain integrity, operations must be fail-safe or the type of information the node holds must be taken into account. For example, mutation must be aware of binary operation nodes, or the operator must be able to handle missing values.

### 4.4 EVOLUTIONARY STRATEGIES

In computer science, an evolution strategy (ES) is an optimization technique based on ideas of adaptation and evolution. It belongs to the general class of evolutionary computation or artificial evolution methodologies.
**Methods**: Evolution strategies use natural problem-dependent representations, and primarily mutation and selection, as search operators. In common with evolutionary algorithms, the operators are applied in a loop. An iteration of the loop is called a generation. The sequence of generations is continued until a termination criterion is met.

As far as real-valued search spaces are concerned, mutation is normally performed by adding a normally distributed random value to each vector component. The step size or mutation strength (i.e. the standard deviation of the normal distribution) is often governed by self-adaptation. Individual step sizes for each coordinate or correlations between coordinates are either governed by self-adaptation or by covariance matrix adaptation (CMA-ES).

The (environmental) selection in evolution strategies is deterministic and only based on the fitness rankings, not on the actual fitness values. The resulting algorithm is therefore invariant with respect to monotonic transformations of the objective function. The simplest evolution strategy operates on a population of size two: the current point (parent) and the result of its mutation. Only if the mutant's fitness is at least as good as the parent one, it becomes the parent of the next generation. Otherwise the mutant is disregarded. This is a (1 + 1)-ES. More generally, \( \lambda \) mutants can be generated and compete with the parent, called (1 + \( \lambda \))-ES. In (1, \( \lambda \))-ES the best mutant becomes the parent of the next generation while the current parent is always disregarded. For some of these variants, proofs of linear convergence (in a stochastic sense) have been derived on unimodal objective functions.

Contemporary derivatives of evolution strategy often use a population of \( \mu \) parents and also recombination as an additional operator, called (\( \mu/\rho^+ \), \( \lambda \))-ES. This makes them less prone to get stuck in local optima.
4.5 APPLICATION OF HYBRID SYSTEM TO MEDICAL DATA SET

Intelligent hybrid models which will overcome the limitations of the existing techniques of Soft Computing methods to improve the disease classification accuracy have been developed. For the purpose of model development, the PIMA Indian Diabetes Dataset from the UCI Depository of Machine learning (ML) Data base is considered.

Genetic Algorithm (GA) is mainly used to optimize the feature subset. Next, GA is fused with ANN & FL to form the hybrid models GA–ANN & GA-FL. Finally GA is combined with ANFIS which is a combination of ANN and FL to formulate the hybrid Model GA-ANFIS. The performance capabilities of the hybrid models, GA – ANN, GA – FL & GAANFIS have been enhanced due to clubbing of GA, the powerful optimization tool.

All the models are applied on the same data set to prove the efficacy of hybrid models, when clubbed with evolutionary computing techniques. A number of useful performance metrics such as accuracy sensitivity, specificity and the area under the Receiver Operating Characteristic Curve (ROC) are computed. The results are analysed and the performances of the techniques are compared. The experimental results do clearly show that the hybrid intelligent techniques are more effective in classifying medical data.

4.5.1. GA – FL Model

In this model, GA is deployed to select the feature subset and after which, the features that are not selected by the GA are removed from the dataset and then FL is applied for classification of data. Figure 4.4 illustrates the GA-FL model development.
4.5.2 Genetic Artificial Neural Network (GA - ANN) Model

GA is a global search technique used in computing and is characterised as a global search heuristics. In this study GA is applied for selection of the feature subset. In addition to this, GA is used to optimize the number of nodes in the hidden layer. The representation of the candidate solution is done by using a string of 12 bits where the first 8 bits represent the input features and the last 4 bits represent the number of nodes in the hidden layer. Any \( i \) th bit in the string of 12 bits can take the value '1' or '0'. Within the first 8 bits when \( i \) takes the value 1 it indicates the \( i \)th input variable is selected and if \( i \) takes 0 it indicates the \( i \)th input variable is not selected. For example the string of 12 bits 101110101001 indicates the selection of the input variables one, three, four, five, seven and the number of nodes chosen in the hidden layer is 9. In this study ANN is trained with back propagation algorithm. Mean Square Error (MSE) of the test data set is considered as the fitness function for the Genetic Algorithm to select the chromosome of the
input features which is a candidate solution for the subsequent generation. Chromosomes with smaller MSE will have a better chance to be selected to survive for the next generation. This is done through the conversion of MSE into ranking. The higher the ranking value is, the chance of the chromosome getting selected for the next generation is better.

\[
\text{fitness} = \left( \frac{\text{rank}}{\sum \text{rank}} \right) \times 100 \quad (4.12)
\]

Rank based Roulette wheel selection procedure is applied to select two chromosomes as parent and after which cross over operator is applied. This produces two new off springs that inherit genetic information from their parents. Then the next step is to replace the worst rank chromosomes which mimics the biological evolution process where most fit chromosome will survive for next generation. Then next, the mutation operation is applied.

**Cross Over and Mutation**

In this work a single point cross over is applied. In cross over operation, the first step is to locate the random cross over point. Then all bits after the location are swapped between the parents to produce two off springs. The crossover and mutation operations are illustrated in figures 4.6 and 4.7.

![Random crossover location](image)

**Figure 4.4 Single point crossover operation**
The two offspring will then replace the worst two chromosomes provided the fitness function of the offspring are better than the fitness of the parents.

The steps followed in the development of GA – ANN hybrid algorithm are:

1. Data preparation including setting of population size, maximum generation and mutation rate.

2. Creation of initial population: A population is a binary string of ‘1’s and ‘0’s. The first 8 bits indicates the input feature is selected or not according to its value is 1 or 0 and the last 4 bits together indicate the number of nodes chosen in the hidden layer and the maximum value it can assume is 15.

3. Decoding each chromosome in the population indicating the selected features and the hidden node size. Features that are not related are removed from all three sets, training, testing and validating data set.

4. The Network training with the training data set using LM Algorithm.

5. Validating the data and testing the data and obtaining MSE of the test data set.

6. Application of cross over operation to generate offspring.

7. Repeating the steps 3 to 6 until the objective (MSE, the desired value) is reached.
The methodology is illustrated in figure 4.6

**Figure 4.6 GA – ANN Model**
4.5.3 Genetic Algorithm – Adaptive Neuro Fuzzy Inference System (GA – ANN- ANFIS) Model

In the GA-ANFIS method, the GA is used to select the best subset of features. Then from the data set, only selected features are chosen to train the ANFIS model. After selecting number of membership function for each and every input. If the MSE for testing is achieved, the procedure is stopped, otherwise designing the appropriate number of membership function continues till desired accuracy is obtained.

The methodology is illustrated in figure 4.7

Figure 4.7 GA – ANN-ANFIS Model

4.5.4 Results

The performances of the four models are compared using their classification accuracy, Mean Squared Error, Sensitivity, Specificity and Precision values. The results obtained are presented in table 4.1.
The classification accuracies of the hybrid models GA-ANN, GA-FL are found to be higher than the classification accuracies obtained through the one of the soft computing technique genetic algorithm and the hybrid models developed in combination of two of the main techniques of soft computing methods is able to do the classification of the data better than the classification task performed by when one technique is used for classification. However, the hybridisation of the three techniques when applied on data for classification, yields better classification.

### Table 4.1 Performances of the Models

<table>
<thead>
<tr>
<th>Classifier</th>
<th>SE</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Precision</th>
<th>Classification Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GA</td>
<td>0.4663</td>
<td>70.68</td>
<td>81.76</td>
<td>56.94</td>
<td>78.95</td>
</tr>
<tr>
<td>GA - FL</td>
<td>0.4183</td>
<td>72.88</td>
<td>83.04</td>
<td>59.72</td>
<td>80.43</td>
</tr>
<tr>
<td>GA - ANN</td>
<td>0.4374</td>
<td>74.58</td>
<td>83.04</td>
<td>60.27</td>
<td>80.87</td>
</tr>
<tr>
<td>GA-ANN-ANFIS</td>
<td>0.3066</td>
<td>84.12</td>
<td>93.67</td>
<td>75.61</td>
<td>87.60</td>
</tr>
</tbody>
</table>

The ROC curves for the GA-ANN and GA-ANN-ANFIS models are shown in figures 4.8 and 4.9.

![Figure 4.8 ROC curve for GA-ANN model](image)

![Figure 4.9 ROC curve for GA-ANN-ANFIS model](image)
4.6 CONCLUSION

In this Chapter, the hybrid models GA–ANN, GA–FL & GA–ANN–ANFIS have been designed, developed and experimented for classification of PIMA Indian diabetes data set. The expert system, ANFIS which is a hybrid technique, combining ANN and FL is very effective for disease classification is evident from the fact that it has performed better in giving high percentage of classification accuracy compared to the performance of its constituent methods as this model has both ANNs’ learning ability and FL’s qualitative approach. In the development of GA-ANFIS model, GA is utilized to optimize the number of input features and then the data set is presented to the ANFIS model and it has been demonstrated that when GA is clubbed with Neuro Fuzzy Model, it enhances the classifiers accuracy significantly.